

Training flood forecasting models using hydrologically diverse data

Everett Snieder

A Dissertation submitted to the Faculty of Graduate Studies
in Partial Fulfilment of the Requirements for
the Degree of Doctor of Philosophy

Civil Engineering
York University
Toronto, Ontario

September 2024

© Everett Snieder 2024

Abstract

Floods constitute a major threat to populations and infrastructure. Flood frequency and severity are projected to increase due to factors such as climate change and urbanisation. Flood early warning systems (FEWS), which rely on models that predict streamflow, provide relevant groups (e.g., transportation authorities, police, schools, etc.) with advance notice of flood risk, allowing them to respond early and minimise flood damage. However, Canada has inadequate flood forecasting infrastructure and lacks a national system. To address the projected flood risk, there is a crucial need to improve modelling capabilities in Canadian watersheds.

The following manuscript-based dissertation presents a series of case studies that propose methodological improvements to hydrological modelling frameworks, focusing on the selection of training data for flood forecasting studies. The studies also demonstrate applications of machine learning (ML) to improve model accuracy and reliability. These studies span a wide range of spatiotemporal conditions, ranging from forecasting flash-flooding at sub-hourly frequencies in small urban watersheds, to nationwide, multi-day forecasts. Each of the four studies exploits the benefits of hydrologically diverse training data for improving model performance as FEWS.

The first manuscript introduces a novel ensemble model framework for traditional, physics-based urban stormwater models. The method leverages the concept of equifinality to generate ensembles with diverse parameter value estimates, which are shown to outperform traditionally calibrated models. The second and third manuscripts evaluate pure ML-based ensembles. The second manuscript simplifies flood forecasting by framing it as a binary problem. The Synthetic Minority Oversampling TEchnique (SMOTE) algorithm is applied to increase the proportion of flood samples in the dataset, which is shown to improve flood forecast accuracy at the expense of an increased number of false positives. Next, three ensemble algorithms and five classifiers are systematically compared; extreme learning machines (ELMs) and support vector machines (SVMs) are found to be the strongest classifiers. The third manuscript proposes an improvement to popular ensemble algorithms, which consists of embedding synthetic oversampling within the ensemble loop, to increase the covariance of ensemble member predictions. The fourth manuscript evaluates a novel cluster-based training data selection framework for regionally trained deep learning models. The method is used to show how data from

hydrologically dissimilar basins is more useful to improving performance in a target basin, compared to data from the target basin itself, or proximal basins.

Collectively, the results of these studies advance techniques for achieving improvements in model performance, especially during high streamflow conditions, which are most important to flood warning systems.

Acknowledgements

Foremost thank you to my partner Caitrin, for taking a chance on me despite having no meaningful income, savings, or “a real job”.

Once again, I would like to thank my supervisor, Dr. Khan. It’s been 7 years since we were first introduced, and I am so grateful for the immeasurable patience you have exerted as you encouraged me to wrap up projects.

Thank you to my dissertation supervisory committee members, Drs. Gingerich and Butler, and examination committee members Drs. Olusola and Tolson, who’s invaluable feedback has improved the quality and clarity of this work.

Another thank you to Dr. Alobaidi for the collaboration on Chapter 3, and who’s perspectives on diversity-in-learning inspired the direction of much of my research.

Special thanks you to the broader Civil Engineering research group, with special mentions to Josephine Morgenroth, Mike DeSantis, Karen Abogadil, Rahma Shakir for excellent discussion and collaboration.

Lastly, I would like to acknowledge York University and NSERC, who provided much of the funding for this research. Additionally, special mention to the NRC-INRAP and MITACS programs, who provided the funding to pursue research projects within and beyond this dissertation.

Table of contents

Abstract	ii
Acknowledgements	iv
List of contents	v
List of figures	xii
List of tables	xxiv
List of abbreviations	xxvi
1 Introduction	1
1.1 Research motivation	1
1.2 Background	1
1.2.1 Hydrological modelling approaches	2
1.2.2 Hydrological model development	4
1.2.3 Ensemble learning	7
1.2.4 Deep learning and the shift towards regional learning	8
1.3 Gaps in research	8
1.4 Research objectives	9
1.5 Data sources	10
1.5.1 Locations	11
1.5.2 Toronto and Region Conservation Authority	11
1.5.3 Province of Ontario	12
1.5.4 Government of Canada	12
1.5.5 HYSETS	12
1.6 Research significance and practical implications	13
1.7 Dissertation outline	13
2 A novel ensemble algorithm based on hydrological event diversity for urban rainfall-runoff model calibration and validation	17
2.1 Preface	17
2.2 Abstract	19

2.3	Introduction	19
2.4	Methods	26
2.4.1	SWMM model setup	26
2.4.1.1	Study area and model creation	26
2.4.1.2	Rainfall event discretisation	28
2.4.1.3	Sensitivity analysis	29
2.4.1.4	Model calibration	32
2.4.2	SWMM ensembles	33
2.4.2.1	Identification of diverse hydrological events using unsupervised clustering	35
2.4.2.2	Combination methods	36
2.4.3	Validation techniques	39
2.5	Results	40
2.5.1	Event discretisation and selection	40
2.5.2	Sensitivity analysis	41
2.5.3	Single event calibration outcome	47
2.5.4	Comparison of ensemble member configurations	48
2.5.5	Evaluation of combination techniques	51
2.6	Discussion	52
2.7	Conclusion	56
3	Advancing Flood Early Warning Systems: Ensemble Learning based Clas- sifiers for Urban Flood Forecasting	58
3.1	Preface	58
3.2	Abstract	59
3.3	Introduction	59
3.4	Methodology	64
3.4.1	Study region and data	64
3.4.2	Data preprocessing	65
3.4.2.1	Candidate input feature set	65
3.4.2.2	Labelling	66
3.4.2.3	Data partitioning	68

3.4.2.4	Feature selection	69
3.4.2.5	Resampling	69
3.4.2.6	Classification and regression comparison	71
3.4.3	Ensemble learning model comparison	72
3.4.3.1	ML models	72
3.4.3.2	Ensemble learning methods	73
3.4.4	Performance metrics	74
3.5	Results and discussion	76
3.5.1	Resampling hyperparameter grid-search	76
3.5.2	Comparison between classification and regression models	78
3.5.3	Ensemble model intercomparison	79
3.5.3.1	ROC and PRC curves	82
3.5.3.2	Watershed comparison	83
3.6	Conclusions	85
4	Resampling and ensemble techniques for improving ANN-based high flow forecast accuracy	87
4.1	Preface	87
4.2	Abstract	88
4.3	Introduction	89
4.4	Early investigations	92
4.4.1	Study area	92
4.4.2	Individual learner description	94
4.5	Review and description of methods for handling imbalanced target datasets . . .	98
4.5.1	Resampling techniques	100
4.5.1.1	Random undersampling	100
4.5.1.2	Random oversampling	101
4.5.1.3	Synthetic minority oversampling technique for regression	101
4.5.2	Ensemble-based techniques	103
4.5.2.1	Randomised weights and biases	104
4.5.2.2	Bagging	105
4.5.2.3	Adaptive boosting for regression	106

4.5.2.4	Least squares boosting	107
4.5.3	Hybrid methods	108
4.5.4	Model implementation and evaluation	109
4.5.4.1	Performance assessment	111
4.5.4.2	K-fold cross-validation	112
4.6	Results	113
4.6.1	Comparison of resampling and ensemble methods	118
4.6.2	Limitations and future work	123
4.7	Conclusion	123
5	A diversity centric strategy for the selection of spatio-temporal training data for LSTM-based streamflow forecasting	125
5.1	Preface	125
5.2	Abstract	127
5.3	Introduction	127
5.4	Methods and Data	131
5.4.1	Input and target variables	131
5.4.2	Basin selection	131
5.4.3	Sequence-to-sequence LSTM models	134
5.4.4	Performance metrics	135
5.4.5	Clustering	136
5.4.6	Experiments	139
5.4.6.1	Experiment 1: evaluating streamflow and basin redundancy in training datasets	139
5.4.6.2	Experiment 2: cross-comparison of 2 clusters of basins	139
5.5	Results and discussion	141
5.5.1	Experiment 1a: cluster-based temporal undersampling	141
5.5.2	Experiment 1b: cluster-based spatial undersampling	144
5.5.3	Experiment 2: cross-comparison of 2 clusters of basins	144
5.6	Conclusion	151
6	Conclusion	152
6.1	Summary	152

6.2	Contributions	153
6.3	Limitations of research	154
6.4	Future work	155
A	Code Availability	193
B	List of Publications	194
C	Supplementary SWMM studies	196
C.1	Supplementary information for Chapter 3	196
C.1.1	Description of SWMM-MATLAB interface	196
C.1.2	Hydrological events	199
C.2	GA-based calibration and ANN-based real-time bias correction for SWMM . . .	202
C.2.1	Introduction	202
C.2.2	Methods	202
C.2.3	Results	204
C.2.4	Conclusion	205
C.3	Investigating event selection for GA-based SWMM rainfall-runoff model calibra- tion	206
C.3.1	Introduction	206
C.3.2	Study area	207
C.3.3	Methods	207
C.3.3.1	Event discretisation and selection	207
C.3.3.2	Genetic algorithm-based optimisation	209
C.3.3.3	Performance criteria	209
C.3.3.4	Multi-model prediction	211
C.3.3.5	Model validation	212
C.3.4	Results and discussion	212
C.3.4.1	Unsupervised event clustering	212
C.3.4.2	Comparison of single model and multi-model predictions	213
C.3.5	Conclusion	214
C.4	Spatial bootstrapping for model-free estimation of subcatchment parameter un- certainty for a semi-distributed rainfall runoff model	216

C.4.1	Introduction	216
C.4.2	Methods	216
C.4.3	Results	218
C.4.4	Conclusion	219
D	Supplementary ensemble studies	220
D.1	Supplementary information for Chapter 5	220
D.1.1	Input variable selection results	220
D.2	Pseudocode	221
D.3	Evaluation of boosting for flood forecasting	224
D.3.1	Introduction	224
D.3.2	Methods	224
D.3.3	Results	225
D.3.4	Conclusion	225
E	Supplementary large-sample studies	228
E.1	Supplementary information for Chapter 6	228
E.1.1	Static attribute statistics	228
E.1.2	Sensitivity of silhouette scores to minimum cluster size	229
E.1.3	Two-cluster attributes	231
E.2	Additional two-cluster comparison performance metrics	232
E.3	Large-scale analysis of machine learning based flow forecasting models for South- ern Ontario	233
E.3.1	Introduction	233
E.3.2	Methods	234
E.3.3	Results and discussion	235
E.3.4	Conclusion	237
E.4	Large-scale evaluation of temporal trends in ANN behaviour for daily flow fore- casts in Canadian catchments	242
E.4.1	Introduction	242
E.4.2	Methods	242
E.4.3	Results and discussion	244
E.4.4	Conclusion	246

E.5	Cluster-based hyperparameter optimisation for LSTM-based flow forecasting in Canadian catchments	247
E.5.1	Introduction	247
E.5.2	Methods	247
E.5.3	Results and discussion	247
E.5.3.1	Hyperparameter transfer across locally trained models	247
E.5.3.2	Hyperparameter transfer across regionally trained models	252
E.5.4	Conclusion	253
E.6	A large sample study of the effects of upstream hydrometeorological input features for LSTM-based daily flow forecasting in Canadian catchments	254
E.6.1	Introduction	254
E.6.2	Methods	254
E.6.3	Results and discussion	255
E.6.3.1	Conclusion	257

List of Figures

1.1	Conceptual schematics of (A) a generic basin with 3 sub-basins, (B) the corresponding process-based model diagram (B) artificial neural network neural diagram (C), and hybrid model (C).	4
1.2	Graphical comparison of methods used in each of four manuscripts. Subplots from left to right indicate the training basin(s), the corresponding hydrograph(s), the resampled streamflow data, and the model or ensemble configurations. (A) shows the discretisation of hydrological events from a continuous record, which are used to train an ensemble of SWMM models. (B) shows SMOTE resampling applied to high flows, used to train ensembles. (C) shows SMOTE used iteratively, to create more diverse ensembles. (D) shows data from additional basins, which is undersampled and used to train a single model.	15
2.1	Location of catchments within Southern Ontario (left), Spring Creek South (top-right), and Taylor Creek South (bottom-right) SWMM models, including subcatchment boundaries (grey), watercourses (blue), flow gauges (red triangles), and raingauges (purple squares).	27
2.2	Complete novel ensemble framework including event discretisation, event clustering, single model optimisation, and ensemble combination.	34
2.3	SCS event features plotted against one another in scatter-plots. Histograms for each event feature are along the diagonal. Colours indicate the cluster ID; Cluster 1 events are green and Cluster 2 events are purple.	42
2.4	TCS event features plotted against one another in scatter-plots. Histograms for each event feature are along the diagonal. Colours indicate the cluster ID; Cluster 1 events are green and Cluster 2 events are purple.	43
2.5	Scatterplots showing median sensitivity of total flow (left) and peak flow (right) of the uncalibrated model to Cluster 1 (x-axis) and 2 (y-axis) events for SCS (a) and TCS (b) catchments. Whiskers indicate range of sensitivities for all events in each cluster. Only parameters with sensitivity values greater than 0.1 are labelled.	44

2.6	Four GA-based calibration routines for the SCS (a) and TCS (b) catchments including: Baseline-Uncertainty, Double-Uncertainty, Sensitive-Parameter, and Insensitive-Parameter calibration. Calibrations to events in Cluster 1 and 2 are identified by the green and purple annotations.	45
2.7	KGE values for the SCS (a) and TCS (b) catchments, colourised from good (blue) to poor (red) performance, obtained for each single-event calibration routine (along the diagonal), cross-validated against each other calibration event (upper and lower triangles). Events are sorted by cluster.	46
2.8	Difference in KGE between the ensemble and mean single model predictions across ensemble size for the SCS (a) and TCS (b) catchments. EC1, EC2, and ECB are shown in green, purple, and red, respectively. Note that the ECB contains twice as many predictors as the EC1, EC2 configurations.	49
2.9	Variance of KGE across increasing ensemble size for uniform weighted combined predictions for the ECB (red), EC1 (green), and EC2 (purple) for the SCS (a) and TCS (b) catchments. Events are ordered by cluster and colour coded event IDs are shown in the top-right corners.	51
2.10	Validation performance of the SCS (a) and TCS (b) catchments for single model (red), uniform weighting (blue), variable weighting (green), MLR-based stacking (yellow), and ANN-based stacking (purple). Calibration performance is indicated by a grey bar.	53
3.1	Don, Etobicoke, and Humber watersheds (cyan with transparent white fill), watercourses (blue), precipitation gauges (yellow squares), stream gauges (yellow triangles), target gauge (green triangle), and flood vulnerable areas (red). Hydrometric features outside the study watersheds are displayed in muted colours for spatial context. Earthstar Geographics satellite imagery used as basemap. . .	64
3.2	Hydrographs with stage data for 2017 at the target gauges for Don (top), Etobicoke (middle), and Humber (bottom). Typical stage values are blue and flood events are highlighted in red.	67
3.3	Stage duration curves for target gauges for Don (top), Etobicoke (middle), and Humber (bottom). The dashed red line corresponds to the 99.7th percentile stage. . .	67

3.4	Example of flood labelling for a persistence window 3. Raw timeseries (a), flood labels based on a static threshold (b), smoothed and trimmed flood labels (c) with 2-timestep gap filled, and 1-timestep flood removed. The static flood threshold and flood samples are indicated by the dashed red line and red points, respectively.	68
3.5	Number of flood samples with an increasing MFD for Etobicoke (left), Humber (middle), and Don (right).	68
3.6	Train (a) and test (b) performance of bagged ANN classifiers for varying SMOTE (indicated by colours) and RUS (x-axis) resampling rates. The red square indicates the original, non-resampled training data.	77
3.7	Classification (blue) and regression (red) model for Don, Etobicoke, and Humber River watersheds according to the FN, F_1 , F_2 , and G-Mean metrics. Uncertainty bands are generated by through bootstrap resampling of the training data.	78
3.8	Flood forecast performance for the Don River LR, SVM, DT, ELM, and ANN models (subplot columns) as used in bagging (blue), stacking (green), and boosting (yellow), according to FNR, F_1 , F_2 , and G-Mean (subplot rows). Transparent uncertainty envelopes are obtained through bootstrapping the training set. .	79
3.9	Flood forecast performance for Etobicoke Creek LR, SVM, DT, ELM, and ANN models (subplot columns) as used in bagging (blue), stacking (green), and boosting (yellow), according to FNR, F_1 , F_2 , and G-Mean (subplot rows). Transparent uncertainty envelopes are obtained through bootstrapping the training set. .	80
3.10	Flood forecast performance for the Humber River LR, SVM, DT, ELM, and ANN models (subplot columns) as used in bagging (blue), stacking (green), and boosting (yellow), according to FNR, F_1 , F_2 , and G-Mean (subplot rows). Transparent uncertainty envelopes are obtained through bootstrapping the training set.	80

3.11	ROC curves for the Don (top row), Etobicoke (middle), and Humber (bottom) for LR, SVM, DT, ELM, and ANN models (left to right columns), and bagging (blue), stacking (green) and boosting (yellow) ensemble algorithms. The number of false positives is shown along the x-axis, and the absolute and relative numbers of true positives are shown along the left and right y-axes. AUC values are listed in the respective ensemble algorithm colour and subplot.	83
3.12	PRC curves for the Don (top row), Etobicoke (middle), and Humber (bottom) for LR, SVM, DT, ELM, and ANN models (left to right columns), and bagging (blue), stacking (green) and boosting (yellow) ensemble algorithms. The TPR is shown along the x-axis, and the PPV and FDR are shown along the left and right y-axes.	84
3.13	Classifier performance, plotted as TP against FP (left) and PPV against TPR (right), with ideal performance in the top-right and top-left corners, respectively. The fifteen unique ensemble configurations are denoted by a unique symbol and the Etobicoke, Don, and Humber models are coloured in green, red, and blue, respectively.	84
4.1	Bow (a) and Don (b) River basins upstream of Calgary and Toronto, respectively. Surface watercourses and waterbodies are shown in blue. The target stage monitoring stations are red while upstream hydrometeorological monitoring stations (stage, precipitation, and temperature) are yellow. Aerial imagery obtained from © Esri [Esri, 2020]. Surface water and watershed boundaries obtained from © Scholars GeoPortal [DMTI Spatial Inc., 2019, 2014c,a,b] and the © TRCA [TRCA, 2018]	93
4.2	Histograms of observed stage for the (a) Bow River 6-hour stage and (b) Don River hourly stage. The dashed red line indicates the fixed threshold used to distinguish between typical and high stage values.	95
4.3	Observed and individual learner stage predictions for the Bow River system for all 10 years of available stage (a) and a 3 month subset which contains particularly high stage (b), to better distinguish between the two hydrographs. The dashed red line indicates the fixed threshold used to distinguish between typical and high stage values.	97

4.4	Observed and individual learner stage predictions for the Don River system for all 10 months of available stage (a) and a 14 day subset which contains particularly high stage (b), to better distinguish the two hydrographs. The dashed red line indicates the fixed threshold used to distinguish between typical and high stage values.	98
4.5	Baseline model residuals versus observed stage for the Bow (a) and Don (b) River systems. Inverse frequency versus gradient across 4 time steps for the Bow (c) and Don (d) River target variables. Colouring indicates normalised scatter point density.	99
4.6	Test MSE across ensemble size for RWB (red), Bagging (blue), AdaBoost (yellow), and LSBoost (green) for the Don (a) and Bow River (b).	113
4.7	Overall (blue), typical stage (red), and high stage (yellow) CE (a) and PI (b) for the Bow River models.	114
4.8	Overall (blue), typical stage (red), and high stage (yellow) CE (a) and PI (b) for the Don River models.	115
4.9	Change in (absolute) performance of CE (a), CE_{TS} (b), CE_{HS} (c), PI (d), PI_{TS} (e), PI_{HS} (f) produced by combinations of resampling (listed along the x-axis) and ensemble (listed along the y-axis) methods for the Bow River models.	119
4.10	Change in (absolute) performance of CE (a), CE_{TS} (b), CE_{HS} (c), PI (d), PI_{TS} (e), PI_{HS} (f) produced by combinations of resampling (listed along the x-axis) and ensemble (listed along the y-axis) methods for the Don River models.	120
4.11	MSE ratio between Bagging and SMOTER-Bagging models for the Bow River calibration (a), Bow test (c), Don River calibration (b), and Don test (d) partitions across high stage threshold values ranging from the 50th to 90th percentile stage values.	121
4.12	Test MSE ratio between Bagging and SMOTER-Bagging models for the Bow (a) and the Don (b) Rivers across ensemble size.	122

5.1	Left column, from top to bottom: unclustered (a), clustered (c), and undersampled streamflow (e) for basin 01AD003 (HYDAT ID; located along St. Francis River in New Brunswick). Right column, from top to bottom: unclustered (b), clustered (d), and undersampled basins (f). Cluster colours are arbitrary and there is no connection between temporal (a, c, e), and spatial (b, d, f) cluster colours. World Gray Canvas basemap in (b, d, f) provided by ESRI.	138
5.2	Scatterplot matrix illustrating the temporal clustering results for Basin 01AD003 streamflows ($K=6$). Markers are colourised by cluster and kernel smoothed histograms are shown along the diagonal. Axis labels q , q_{grd1} , q_{grd3} , $\sin$, and $\cos$, are shortform for q_t , $(q_{t-3} - q_t)/3$, $q_{t-1} - q_t$, $\sin^{-1}(t/365)$, and $\cos^{-1}(t/365)$, respectively	142
5.3	Hydrographs for Basins 01AD003 (top) and 07AF002 (bottom) with observations from October 1989–October 1991, colourised by cluster. Cluster colours are arbitrarily assigned.	142
5.4	CDFs showing model performance according to NSE (top row) and PI (bottom row) for RUS (red), two CUS_Q configurations (blue and green) and no resampling baseline (black). Subplots (a), (b), (c), and (d), (e), (f) correspond to forecasts of 1, 2, and 3 days respectively.	143
5.5	CDFs for NSE (top row) and PI (bottom row) for models trained to various basin subsets, which include a baseline (black line, includes all basins), cluster-based undersampling (coloured solid lines), and random undersampling (coloured dashed lines). The number of basins sampled from each cluster in the CUS_B configurations is equal to B/K . Subplots (a), (b), (c), and (d), (e), (f) correspond to forecasts of 1, 2, and 3 days respectively.	145
5.6	Clustering ($K=2$) result for basins across Canada. For each cluster, 32 square markers indicate basins selected for the baseline and evaluation sets in the two-cluster experiments. World Street Map basemap provided by ESRI.	146
5.7	Scatterplot matrix illustrating the five most important features for clustering, as determined using the RF surrogate model. Markers are colourised by cluster and kernel smoothed histograms are shown along the diagonal.	146

5.8	CDFs for models evaluated on C0 basins according to NSE (top row) PI (bottom row). Each row compares models trained to three different training datasets: a baseline that includes 32 basins in the respective cluster, the baseline plus 32 basins in the same cluster, and the baseline plus 32 basins in the other cluster. Subplots (a), (b), (c), and (d), (e), (f) correspond to forecasts of 1, 2, and 3 days respectively.	147
5.9	CDFs for models evaluated on C1 basins according to NSE (top row) PI (bottom row). Each row compares models trained to three different training datasets: a baseline that includes 32 basins in the respective cluster, the baseline plus 32 basins in the same cluster, and the baseline plus 32 basins in the other cluster. Subplots (a), (b), (c), and (d), (e), (f) correspond to forecasts of 1, 2, and 3 days respectively.	147
5.10	Boxplots showing the NSE and PI of models evaluated on 32 C0 basins for experiment 2a (a, b), and 2b (b, c), and 32 C1 basins for experiment 2a (e, f) and 2b (g, h). for a lead time of 3 days. The baseline model, which is trained uniquely using the evaluation basins, is indicated in bold. Blue, red, and green colours indicate models trained on C0, C1, and both types of basins.	148
5.11	Model performance (PI) across increasing numbers of training basins for models evaluated on 4 C0 basins (a) and 4 C1 basins (b). Pie chart markers illustrate the proportion of C0 and C1 basins used in each training dataset. The coloured dashes along the top of each subplot indicate the which cluster produced the better addition to the training dataset.	150
C.1	Process diagram showing how Python (green), MATLAB (red), and SWMM (blue) are used to create SWMM model files, and perform model calibration. . .	198
C.2	Sample discretised hyetographs (blue) and class 1 (green, left column) and class 2 (purple, right column) hydrographs for the SCS (a) and TCS (b) catchments.	199
C.3	SWMM model of the Fourteen Mile Creek watershed.	202
C.4	Configuration of SWMM, ANN, and hybrid models.	203
C.5	Objective function versus GA generations for calibration of SWMM models on six different rainfall events.	204
C.6	Event-based calibration performance for SWMM, ANN, and hybrid models. . . .	205

C.7	NSE (top row) and PI (bottom row) of the three models and six events considered. The events along the Y-axis are used for calibration while the X-axis are used for validation. The diagonal shows the calibration performance of each event whereas the remaining cells show the validation performance.	206
C.8	Fourteen Mile Creek SWMM model. Subcatchment boundaries (black lines), subcatchment routing (grey lines), watercourses (blue lines), storage units (green squares), weirs (yellow lines), and the stream gauge location (red triangle). The basemap (OpenStreetMap) illustrates landcover, roads, and Lake Ontario.	208
C.9	Scatterplot matrix for clustering indicator variables; classes are distinguished by colour.	214
C.10	Validation performance of SWMM configurations for single model (red), uniform weighting (blue), variable weighting (green), MLR-based stacking (yellow), and ANN-based stacking (purple). Calibration performance is indicated by a grey bar.	215
C.11	(A) Taylor Creek tributary (blue lines), stream gauge (red triangle), rain gauge (blue square); (B) Digital Terrain Model; (C) delineated Taylor Creek catchment area (pink); (D) Taylor Creek subcatchments (violet); (E) Impervious (black) and pervious (green) landcover overlaid by subcatchment boundaries (pink); (F) SWMM model as shown in GUI (back routing network), overlaying the catchment boundary.	217
C.12	Pervious (green) and impervious (black) landcover at a 1x1m resolution for an arbitrary residential neighbourhood in Taylor Creek, overlaid by 100x100m grid, illustrating block bootstrapping resampling procedure.	218
C.13	Model performance (KGE) across increasing subcatchment resolution for four block bootstrap sizes (colours).	219
C.14	Model performance (KGE) across increasing block sizes for four subcatchment resolutions (colours).	219
D.1	Map showing locations of upstream stream (yellow), target stream (red) and precipitation (green) gauges for the Don River. The watershed area is marked in grey and hydrographic features are shown in dark blue.	224
D.2	Graphical description of each of the three boosting methods considered	225

D.3	(a) Hydrographs of two runoff events for observed (black), no boosting baseline (blue), GradBoost.ls (green), AdaBoost.rt.rs (red), and AdaBoost.rt.rw (yellow). (b) Evolution of cost function weights across boosting iterations for each of the three boosting methods.	226
D.4	Violin plots illustrating the training (top) and test (bottom) performance of each configuration, according to NSE (left) and PI (right).	226
D.5	Poster presented at the AGU 2019 General Assembly.	227
E.1	Silhouette scores (left) for minimum cluster sizes of 2 (a), 8 (c), 16 (e), 32 (g), and 64 (i). Respective clustering results (right) for K values (corresponding to maximum silhouette score) of 8 (b), 7 (d, f), and 2 (h, j). World Gray Canvas basemap provided by ESRI.	230
E.2	Boxplots showing the α – NSE (left) and β – NSE (right) of models evaluated on 32 C0 basins for experiment 2a (a, b), and 2b (b, c), and 32 C1 basins for experiment 2a (e, f) and 2b (g, h). for a lead time of 3 days. The baseline model, which is trained uniquely using the evaluation basins, is indicated in bold. Blue, red, and green colours indicate models trained on C0, C1, and both types of basins.	233
E.3	Boxplots showing the MSE (left) and OFR (right) of models evaluated on 32 C0 basins for experiment 2a (a, b), and 2b (b, c), and 32 C1 basins for experiment 2a (e, f) and 2b (g, h). for a lead time of 3 days. The baseline model, which is trained uniquely using the evaluation basins, is indicated in bold. Blue, red, and green colours indicate models trained on C0, C1, and both types of basins.	234
E.4	Locations of major floods in Southern Ontario that occurred between 1900 and 2021. The floods shown are from the Canadian federal disaster database, thus are only those that meet the Government of Canada’s definition of a disaster [Government of Canada and Public Safety Canada, 2021].	234
E.5	Diagram of data-processing pipeline.	235
E.6	Performance of models on the train (y-axis) and test (x-axis) datasets according to NSE (left) and PI (right). Points along and below the diagonal are considered to have good generalisation. Colour illustrates the density of points.	236

E.7	Spatial patterns in performance of each watershed. Colour indicates performance according to NSE (left) and PI (right). Points are located at the centroids of each catchment.	236
E.8	Performance of models according to NSE (top) and PI (bottom) against increasing numbers of available flow (left), precipitation (middle), and temperature (right) stations.	237
E.9	Hydrological characteristics, including mean flow (MEAN_MMHA), annual flow seasonality (ANNUAL_SEASONALITY), high flow frequency (HIGH_Q_FREQ), high flow duration (HIGH_Q_DUR), low flow duration (LOW_Q_DUR), low flow frequency (LOW_Q_FREQ), flow duration curve slope (SLOPE_FDC), and baseflow index (BASEFLOW_INDEX).	238
E.10	Physical watershed characteristics, including mean slope (SLOPE), perviousness fraction (PERV), volumetric volume of coarse fragments fraction (CFVO), bulk density of fine earth fraction (BDOD), proportion of clay (CLAY), and (SAND), silt (SILT), and soil organic carbon content (SOC).	239
E.11	Correlogram illustrating linear relationships between watershed characteristics. .	240
E.12	Feature importance of watershed characteristics for predicting model performance of Southern Ontario watersheds.	241
E.13	Model performance against annual flow seasonality (left) and high flow duration (right).	241
E.14	Example of target flow (red) overlaid by partitioning schemes for ‘fixed training data’ (yellow), ‘recent training data’ (blue), and ‘all training data’ (green). Models are evaluated in 1-year time-continuous segments (purple) in increments of one year.	243
E.15	Example of target flow (red) overlaid by sliding training window (green) used to incrementally train RF models, which are used to quantify temporal feature importance. An example of sequential feature rankings is shown in the top left. .	243
E.16	Temporal trends in model performance for ‘fixed’ (yellow), ‘recent’ (blue), and ‘all’ (green) training configurations, according to NSE (labelled CE, top left), NSE on high flow (labelled CE _{hf} , top right), PI (labelled CP, bottom left), and MVE (bottom right). Lines indicate mean performance across basin, while the filled uncertainty envelope shows 95th percentiles of the basin set.	244

E.17	Temporal trends in feature importance for flow (red), precipitation (blue), and temperature (yellow).	245
E.18	Mean feature importance of flow (top) and temperature (bottom) shown spatially. The colour gradient from yellow to purple correspond illustrates most important to least important, respectively.	245
E.19	Mean feature importance of precipitation (top) across Canada, and zoomed subplots showing for basins in western British Columbia (bottom left), Southern Ontario (bottom middle), and Atlantic Canada (bottom right). The colour gradient from yellow to purple correspond illustrates most important to least important, respectively.	246
E.20	K-Means clustering results (K=6) for basins across Canada. Enlarged markers with white borders were selected, based on Euclidean distance to cluster centroid, for the analysis of hyperparameter optimisation and transfer.	248
E.21	Cross-comparison of the five most important clustering features, including elevation, annual precipitation, daily precipitation, monthly precipitation, and slope, according to an RF trained to predict the K-Means clustering results. The diagonal contains smoothed histograms for each feature. Points are coloured by cluster.	249
E.22	Relative change in model performance (MSE) between target basins (x-axis) with non-optimum hyperparameters, and with hyperparameters donated from each source basin (y-axis). Basins are grouped based on cluster and groups are divided by white margins. Red to yellow to green indicates that the hyperparameter transfer produced worse, no impact, and better performance, respectively.	250
E.23	Six basins with homogeneous (a) and heterogeneous (b) hydrological characteristics, per the K-Means clustering results. Basin footprints are underlain in transparent blue.	251
E.24	The performance on six 'diverse' basins of models trained individually to each basin (red), a model trained to all six basins (blue), a model trained to 60 basins (6 from each cluster, including the test basins; yellow), and a model trained to the same 60 basins with RUS training data (green).	251

E.25	The performance on six 'similar' basins of models trained individually to each basin (red), a model trained to all six basins (blue), a model trained to 60 similar basins (60 from the same cluster, including the test basins; yellow), and a model trained to 60 diverse basins (6 from each cluster, including the test basins; green).	252
E.26	Example of hydrological network extracted from the nested basin boundaries.	254
E.27	Generic S2S LSTM diagram illustrating a forecast of timesteps $t+0$ to $t+m$ based on a lookback period of $t-1$ to $t-n$.	254
E.28	Cumulative distribution functions for NSE and PNSE for single-basin MLR (red), persistence-based (blue), and variance-based (yellow) basin regularisation.	255
E.29	Relationship between NSE and PNSE for persistence-based (blue) and variance-based regularisation (yellow).	256
E.30	Cumulative distribution functions for NSE and PNSE for single-basin MLR without (red) and with upstream data (blue), S2S without (yellow), and with upstream data (purple).	256
E.31	Absolute change in 3-day PNSE, ranging from blue (improvement), to white (no change), to red (deterioration), between S2S models without and with upstream data.	256
E.32	Basin-wise permutation-based feature importance. Features colourised by time-series (red), physical attribute (green), hydrometeorological statistic (blue) and one-hot encoding (yellow).	257

List of Tables

1.1	Tabular comparison of methods used in each of four manuscripts. The ‘basins’ column indicates whether one or multiple basins are used to calibrate each model(s). The ‘preprocessing technique’ describes the clustering or resampling method used to generate training datasets. The ‘models’ column indicates whether predictions are made using one or multiple models.	14
2.1	Summary of catchment properties and select SWMM model parameters for SCS and TCS. Gauge IDs are obtained from the TRCA [TRCA, 2020].	26
2.2	Calibration parameter names and sections in SWMM, units, relative uncertainty, and upper and lower constraints. Constraints are based on physically possible limits, which are determined from published ranges and SWMM input value constraints. Detailed descriptions of each parameter are available in the SWMM user manual [Rossman, 2015].	32
2.3	Mean performance rank and variance of calibration models, single model predictions, and all combination techniques for the SCS catchment. Bold font indicates the best column-wise scores, ignoring the CAL case.	52
2.4	Mean performance rank and variance of calibration models, single model predictions, and all combination techniques for the TCS catchment. Bold font indicates the best column-wise scores, ignoring the CAL case.	52
3.1	Summary of watershed attributes.	65
3.2	Comparison between classification and regression ANNs.	72
3.3	Confusion matrix illustrating how true and false positives and negatives form the basis for the TPR, FPR, PPV, TNR, and accuracy.	74
3.4	Mean performance of ensemble algorithms for each watershed and metric. Best performance between ensemble algorithms is indicated in bold.	81
3.5	Mean performance of ensemble members for each watershed and metric. Best performance between ensemble members is indicated in bold.	82
4.1	Target variable statistics for the Bow and Don River watersheds.	94
4.2	Input variables for the Bow and Don Rivers.	96
4.3	Individual learner ANN model description used for both watersheds.	96
4.4	Summary of ensemble methods and hyperparameters.	110

4.5	Mean CE and PI scores for all, typical, and high stage for the Bow River ensembles; the highest scores are shown in bold and the calibration scores are italicised and enclosed by parentheses.	116
4.6	Mean CE and PI scores for all, typical, and high stage for the Don River ensembles; the highest scores are shown in bold and the calibration scores are italicised and enclosed by parentheses	117
5.1	Basin static attribute statistics (mean, standard deviation, 10th, 50th, and 90th percentile) calculated for 2363 Canadian catchments.	133
5.2	Dynamic features used as model inputs.	134
5.3	Streamflow resampling configurations in Experiment 1a. Labels denote the resampling type, number of clusters (K ; for CUS), and resampling range (ϕ).	140
5.4	Basin resampling configurations in Experiment 1b. Labels denote the resampling type, number of clusters (K ; for CUS _B), and resampling range (ϕ).	140
5.5	LSTM size and training datasets used in Experiment 2. Configurations are grouped by experiment	141
6.1	Summary of chapters	152
C.1	Values of event cluster centroids and total number of events for SCS and TCS catchments.	200
C.2	Statistical summary of discretised hydrological events for the SCS catchment. . .	201
C.3	Statistical summary of discretised hydrological events for the TCS catchment. . .	201
C.4	GA calibration parameters and uncertainty ranges.	203
C.5	Discretised event dates, statistics, and classification result.	204
C.6	GA Calibration Parameters and Uncertainty Ranges	210
C.7	Discretised event dates, statistics, and classification result.	213
C.8	Ranked performance and variance for KGE, PFE, and MVE.	213
D.1	List of 25 most useful inputs identified using the PC IVS algorithm for the Bow and Don River watersheds, selected from the set of candidate inputs. Input variables are encoded in the following format "station ID"_"variable"_"static"_"lagged timesteps". Variable abbreviations "WL" and "Precip" refer to water level (stage) and precipitation.	220
E.1	Mean and standard deviation of basin attributes for 2 cluster configuration. . . .	231
E.2	LSTM hyperparameter ranges and search distributions	247

List of abbreviations

AdaBoost[-RT]	Adaptive Boosting [Regression Threshold]
ANN	Artificial Neural Network
AUC	Area Under Curve
Bagging	Bootstrap Aggregation
CE	Coefficient of Efficiency
ELM	Extreme Learning Machine
DDM	Data-Driven Model
DT	Decision Tree
DTM	Digital Terrain Model
FEWS	Flood Early Warning Systems
FDR	False Discovery Rate
FP[N]	False Negative [Rate]
FP[R]	False Positive [Rate]
GA	Genetic Algorithm
HS	High Stage
IVS	Input Variable Selection
KFCV	K-Fold Cross Validation
KGE	Kling-Gupta Efficiency
KNN	K-Nearest Neighbours
KS	Kolmogorov Smirnov
LM	Levenburg Marquardt
LR	Logistic Regression
LSBoost	Least-Squares Boosting
LSTM	Long-Short Term Memory
MIT	Minimum Inter-event Time
ML	Machine Learning
MLP	Multi-Layer Perceptron
MLR	Multiple Linear Regression
MSE	Mean Squared Error
MVE	Mean Volume Error
NSE	Nash-Sutcliffe Efficiency

NSERC	National Science and Engineering Research Council
PC	Partial Correlation
PCA	Principal Component Analysis
PFE	Peak Flow Error
PI	Persistence Index
PPV	Positive Predictive Value
PRC	Precision Recall Curve
RF	Random Forest
RMSE	Root Mean Squared Error
ROC	Routine Operator Characteristic
ROI	Region Of Interest
ROS	Random OverSampling
RUS	Random UnderSampling
RWB	Randomised Weights and Biases
SCS	Spring Creek South
SMOTE[R]	Synthetic Minority Oversampling Technique (Regression)
SOM	Self-Organising Map
SVM	Support Vector Machine
SWMM	StormWater Management Model
TCS	Taylor Creek South
TRCA	Toronto and Region Conservation Authority
TP[N]	True Negative [Rate]
TP[R]	True Positive [Rate]
TS	Typical Stage

1 Introduction

This section introduces the research motivation towards this manuscript-based dissertation, followed by some general technical background, research objectives, significance, and an outline of the document. The technical background is aimed at a general audience, while each manuscript chapter contains introductions in greater detail.

1.1 Research motivation

Flooding is the most severe natural disaster in Canada, causing billions in damage every year [Burn and Whitfield, 2016]. Flooding in Canada is expected to worsen under climate change, which is projected to increase the frequency of extreme hydrological events [Grum et al., 2006]. Beyond preventative measures such as hydraulic infrastructure, flood early warning systems (FEWS) are considered to be the most effective ways to mitigate flood damage [Creutin and Borga, 2003]. FEWS are used to operate flood infrastructure and provide advanced notice of flood risk, which is critical to evacuation efforts [Munawar et al., 2021].

The following manuscript-based dissertation contains a series of case studies that propose methodological improvements to hydrological modelling frameworks. These studies comprise a wide variety of hydrological modelling frameworks and techniques.

Since being proposed roughly 30 years ago [Hsu et al., 1995], ML has been increasingly applied for hydrological modelling. While the topic received consistent attention in research, until recently [Nevo et al., 2022], there were relatively few operational applications of ML for FEWS. Much of the resistance to adopt ML models in operational settings can be attributed to the ‘black-box’ characterisation of ML models, as well as their (supposed) poor ability to extrapolate [Nearing et al., 2021].

The following body of work is motivated by improving the efficacy of ML-based streamflow forecasting models. Improvements are focused on training data selection, with respect to model performance on flood level flows.

1.2 Background

The following sections are intended to provide a brief primer to hydrological modelling, with a focus on aspects that are relevant to the research objectives in Sec. 1.4. In brief, these include

modelling paradigms, and model training.

1.2.1 Hydrological modelling approaches

Hydrological models, specifically rainfall-runoff models, typically aim to simulate the rainfall-runoff transformation. Such models come in a variety of forms, but all tend to use hydrometeorological input data to predict runoff (as either streamflow or river stage). The accuracy of these models is evaluated using performance metrics, which are a function of observed and predicted runoff values. There are many different performance metrics, ranging from simple mean squared error (MSE), to specialised peak flow metrics; a review of performance evaluation is provided in [Bennett et al., 2013].

Approaches to hydrological models are often grouped into two paradigms, 'process-based' and 'data-driven' [Sabzipour et al., 2023]. Comparatively, process-based models have a long history and are much more widely used in practice compared to pure data-driven models. While data-driven models are steadily increasingly in popularity and have recently been shown to convincingly outperform process-based models [Kratzert et al., 2019b, Sabzipour et al., 2023, Mosaffa et al., 2022]. The following three sections provide a brief overview of process-based, data-driven, and hybrid modelling approaches, followed by generalised descriptions of data partitioning, model calibration, and evaluation techniques.

Process-based models

Process-based models, also referred to as process-based or traditional models, are meant to be representative of hydrological processes and simulate the movement of water through some hydrological system. A comprehensive review of the evolution of process-based numerical models in catchment hydrology is reviewed by Paniconi and Putti [2015]. Process-based models exist on a broad range of complexity, both in terms of process representation and spatiotemporal resolution [Horton et al., 2022]. For example, a simple 'lumped' model may contain only a few parameters for an entire catchment. In contrast, a detailed 'distributed' model may have thousands of parameters. Similarly, models may only simulate a single process, or dozens of interconnected processes. The Stormwater Management Model, which is the modelling platform featured in chapter 2, is a semi-distributed model in which catchments are divided into subcatchments, which have homogenous properties. Each subcatchment is treated as a con-

control volume with inputs (e.g., precipitation, overland flow, etc.), outputs (infiltration, evapotranspiration, outflow, etc.), and storage (e.g., reservoirs, cisterns, etc.). At each simulation timestep, water is routed between each of these elements. It is widely agreed that there is no universally ‘best’ hydrological model [Horton et al., 2022]. Nevertheless, the literature is rich with comparisons between process-based models with varying degrees of complexity [Mai et al., 2022, Ruelland et al., 2008, Khakbaz et al., 2012, de Lavenne et al., 2016]. In recent years, the rising popularity of purely data-driven hydrological models, discussed in the following section, has also resulted in numerous comparisons between process-based and data-driven approaches [Mai et al., 2022, Kratzert et al., 2019b, Lees et al., 2021].

Data-driven models

In contrast, data-driven models are purely statistical. While data-driven is a term that encompasses a broad spectrum of statistic models, the work herein is focused uniquely on machine learning models. Rather than attempting to model the underlying hydrological system that relate hydrological variables, machine-learning models are only interested in the mathematical transformation between inputs and outputs. Such an approach has obvious advantages and limitations. Some advantages include better performance compared to process-based models [Mai et al., 2022], flexible input requirements [Jin et al., 2024], improved regionalisation [Sabzipour et al., 2023]. However, data-driven models come with disadvantages, such as their limited hydrological interpretability [Lees et al., 2022] and their ability to extrapolate [Nguyen et al., 2023].

Hybrid approaches There is a vast modelling landscape that lies between process-based and data-driven models, aimed at leveraging advantages from each approach. For example, Höge et al. [2022] use machine-learning models to estimate internal states in conceptual hydrological models, which leverages the strong accuracy of ML without sacrificing model interpretability.

Comparison of model types

Figure 1.1(A) illustrates a generic basin with 3 sub-basins, and examples of (B) a process-based model (semi-distributed), (C) a data-driven model (ANN), and (D) a hybrid approach (where an internal process of a process-based model is replaced with an ANN). In the process-based model, basins and hydrological routing are represented in the model structure. Input features, denoted by purple and blue nodes, drive mass-balance equations. Sub-basin attributes

such as soil types and topography inform how the model is stored within, or exits, each sub-basin. In contrast, the data driven model has a model structure that is not informed by hydrology. While the input data is the same, the data-driven model is not parameterised with hydrological processes. Instead, the model is simply trained to learn the input-output transformation. That said, there does exist an indirect relationship between model topology and hydrological processes. For example, predicting more complex hydrological processes may require a data-driven model with a greater number of parameters. Finally, the hybrid model features a similar structure to the process-based model in (B), however, the routing between upstream and downstream catchments is replaced by an ANN module.

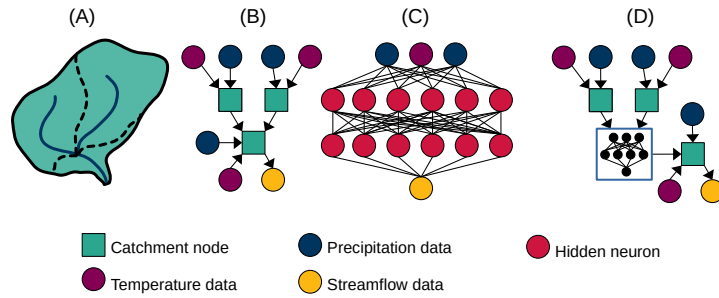


Figure 1.1: Conceptual schematics of (A) a generic basin with 3 sub-basins, (B) the corresponding process-based model diagram (B) artificial neural network neural diagram (C), and hybrid model (C).

1.2.2 Hydrological model development

A comprehensive background of hydrological modelling is beyond the scope of this work; the literature is rich with reviews and comparisons of approaches to process-based hydrological modelling [Horton et al., 2022, Devia et al., 2015, Pechlivanidis et al., 2011, Mai et al., 2022], and data-driven modelling [Zounemat-Kermani et al., 2021, Maier and Dandy, 2000]. Instead, the following sections provide a background on topics that are fundamental to the main chapters of this work. These topics include diversity, dataset partitioning for training and testing hydrological models, and as well as overfitting, model generalisation, and diversity-in-learning.

Hydrological diversity Hydrological diversity is a term that appears regularly in literature [Delaigue et al., 2022, Merino-Martín et al., 2012, Gauch et al., 2021], but lacks a concrete definition. The expression is broadly used to describe hydrologically diverse conditions (e.g., high flow versus low flow periods), or processes (e.g., snowmelt dominated basin versus arid basin).

In ensemble literature, diversity is used with respect to the error distributions of ensemble members [Brown et al., 2005a], which is a key attribute of ensemble modelling, which is outlined in Sec. 1.2.3. Training models to diverse datasets is among the best ways to generate ensemble diversity. Thus, there is a link between the two uses of the term: hydrologically diverse training datasets can be used to generate ensembles with diverse error distributions.

In this work, diversity is broadly used to describe hydrological diversity, both in time and space. In each chapter, diversity is characterised using some form of classification, which is one of many tools for identifying diverse conditions. Given a set of hydrological data, diverse subsets of data are formed by maximising the representation of each cluster. In contrast, a subset that belongs to a single cluster is not considered diverse.

Dataset partitioning Models must be validated to ensure that they meet performance expectations. Hydrological models are typically validated temporally or spatially. Temporal validation is done using historic hydrometric data collected in the basin being modelled. In contrast, spatial validation is performed on models developed for ungauged basins. The studies featured in Chapters 4–5 exclusively focus on prediction in gauged basins, therefore unless otherwise stated, validation refers to temporal validation.

Validation requires that data be partitioned into training and testing datasets. The training dataset is used to tune model parameters and the test dataset is considered to be unseen by the model and is used to verify the performance of the model if deployed [Bowden et al., 2002]. Training and testing are terms used for data-driven models, whereas for process-based models, training and testing datasets are typically referred to as calibration and validation datasets, respectively [Höge et al., 2018]. The following paragraphs use the terminology associated with data-driven models.

Process-based models are typically parameterised based on existing hydrological datasets. Due to various sources of uncertainty, process-based models rarely perform accurately without tuning model parameters, within some estimated uncertainty range, such that the model output more closely matches observations [Beven and Binley, 1992, Beven, 2012, Knighton et al., 2014].

The parameters of data-driven parameters are randomly initialised, and the models are trained to match observations. As a result, training is a necessary step for data-driven models. In

contrast, process-based models, while benefiting from calibration, may exhibit reasonable performance without any tuning.

There is a huge number of training algorithms available for process-based [Arsenault et al., 2014, Shahed Behrouz et al., 2020] and data-driven models [Anh et al., 2023] alike. While there are some overlaps between the algorithms used for data-driven and process-based models, not all algorithms are available to both types. For example, gradient-descent-based algorithms that are popular in machine learning rely on a differentiable cost function, which process-based models often lack. In contrast, gradient-free, evolutionary optimisation algorithms are often used for process-based models, but despite their capacity for competitive performance, are rarely used for tuning machine learning models compared to gradient-based alternatives [Morse and Stanley, 2016, Whitlam et al., 2021].

Generalisation and overfitting

Generalisability refers to a model’s ability to perform on unseen data. Since this cannot be calculated directly, the generalisation error is calculated on the test dataset [Advani et al., 2020, Hastie et al., 2009]. Overfitting occurs when the model is trained to the extent that it fits noise or idiosyncrasies contained in the training dataset [Dietterich, 1995]. Since these are typically not present in the test dataset, they can result in high test error (or poor generalisation), despite low training error. Models with high degrees of freedom are most prone to overfitting. Comparatively, data-driven models typically have greater degrees of freedom than process-based models, thus are more at risk to overfitting [Höge et al., 2018].

Peak flows are challenging to predict accurately. Several studies have documented models, that despite perform well on average, underestimating peak flows [Schoppa et al., 2020, Sudheer et al., 2003, Abrahart et al., 2007]. The problem of poor peak flow performance cannot be solved using a conventional training approach; tuning a model to the point that it achieves good peak flow performance typically results in severe overfitting. This prediction efficiency is in part attributable to the use of standard cost functions such as SSE, which have been shown to produce models that underpredict peak streamflows [Gupta et al., 2009, Mizukami et al., 2019]. Such models tend to perform well on typical flows, with a small number of streamflow samples contributing to a large proportion of the error [Clark et al., 2021].

Thus, training models for FEWS requires techniques that emphasise accuracy on flood obser-

vations, while avoiding overfitting. There are many approaches available to mitigate the risk of overfitting, which can be grouped into modifications to the training procedure, and modifications to the training dataset [Anctil and Lauzon, 2004a]. The chapters herein focus on the latter. Increasing the diversity of hydrological conditions to which a model is exposed during training is likely to improve its performance on future conditions [Zheng et al., 2022]. There are many methods for increasing the diversity in the training dataset.

Often, due to limits of available historic data, increasing the timespan of the training dataset is not always possible [Freni et al., 2009, Adamowski et al., 2012]. In other cases, such as for heuristic optimisation of complex process-based models, training on multiple years of data is too computationally expensive.

1.2.3 Ensemble learning

Ensemble learning is a machine learning technique in which multiple models (called base learners) are trained, then combined to generate predictions. Such approaches have been extensively applied to hydrological problems [Zounemat-Kermani et al., 2021]. The success of ensemble learning is attributable to the error covariance between predictors, which is cancelled out during combination [Brown et al., 2005a]. There are many strategies for achieving diverse ensemble, a taxonomy and summary are provided in [Brown et al., 2005a]. Some approaches for creating diversity-in-learning include modifying the training data [Abraham, 2003, Li et al., 2016, Alobaidi et al., 2015, Anctil and Lauzon, 2004b], changing the architecture [Masisi et al., 2008, Belayneh et al., 2016], and, changing the starting point in the hyperparameter space [Opitz and MacLin, 1997, Opitz and Maclin, 1999, Maqsood et al., 2004], and changing the hyperspace traversal [Fiszelew et al., 2007].

Among these, modifying the training dataset of each ensemble member is among the most widely used approach and is used in some of the most popular ensemble algorithms such as Boosting and Bagging [Zounemat-Kermani et al., 2021, Brown et al., 2005a]. In contrast to subsampling permutations of a single dataset, deep-learning methods are increasingly used to incorporate training data from beyond the region of interest [Kratzert et al., 2019a, Anderson and Radić, 2022].

1.2.4 Deep learning and the shift towards regional learning

Within hydrology, there is a growing trend towards deep-learning approaches. In this context, deep-learning is used to include ANNs with more than a single hidden layer, and structures that are more complex than simple feed-forward networks. Examples of deep learning architectures include recurrent neural networks (RNNs), long-short term memory networks (LSTMs), graph neural networks (GNNs), and convolutional neural networks (CNNs).

Typically, traditional applications of machine learning in hydrology train models or ensembles to single basins. In contrast, deep learning approaches train models to data from multiple basins, which is referred to as regional learning [Kratzert et al., 2019a]. In the context of diversity, diversity is achieved by increasing the training dataset to span a wide variety of hydrological conditions, which reduces the need for large ensembles. Of course that is not to say that deep learning models do not benefit from ensemble frameworks, in fact, adapting ensemble algorithms to deep learning is an area of active research interest [Ganaie et al., 2022]. However, their benefit comparatively less than simpler ML models such as shallow ANNs, SVMs and DTs, which are sometimes referred to as ‘weak learners’, which tend to have more bias and variance than deep learning models, thus their predictions benefit more from ensemble combination [Brown et al., 2005a]. Additionally, deep learning models are much more computationally expensive than alternative formats [Ganaie et al., 2022].

1.3 Gaps in research

While flood forecasting is frequently a common application for hydrological models, there are often few considerations made during model development to ensure models will perform accurately on flood-level flows. Standard modelling approaches often result in poor model performance on high flows, which is due to the inherent infrequency of flood events. In most cases, there are very few flood level flows within hydrometric datasets. Decades worth of data may only contain a few flood observations. Thus the challenge of calibrating hydrological models becomes achieving a high degree of accuracy on a small number of extreme values, while maintaining adequate performance in terms of generalisation on unseen data, extrapolation, and typical flow performance.

The flood forecasting challenge is exacerbated by the fact that popular training cost functions such as the root mean-squared error (RMSE) and Nash-Sutcliffe Efficiency (NSE) often

result in inadequate performance on high streamflows [Ehret and Zehe, 2011, Gupta et al., 2009]. The Kling-Gupta Efficiency (KGE) is widely seen as an improvement over such metrics [Gupta et al., 2009], however, its use can still produce poor performance on flood events [Knoben et al., 2019]. While uncommonly used compared to residual-based metrics, classification metrics provide an alternative tool for evaluating FEWS performance. The rates of false positive and false negative flood warnings provide a much better sense for FEWS performance compared to a score like NSE. Ultimately, special consideration must be given to flood level flows during both model training and evaluation. Streamflow forecasting models also often exhibit heteroscedastic residuals, with residuals increasing as streamflow or stage increases [Fleming et al., 2015]. There is a need for improved tools for reducing or eliminating this effect, which could in turn improve model capabilities with respect to FEWS.

There is a need to advance methods for training hydrological models to improve their accuracy on flood level flows. However, under traditional training methods, low training accuracy on high flows typically causes inaccurate performance on low flows, or poor model generalisation. Modifying the training dataset, to include greater hydrological diversity - especially of high flows, presents a potential solution to this problem.

1.4 Research objectives

The overarching objective of this body of work is to advance training procedures and data selection hydrological models as to improve their performance on flood level flows. This work leverages hydrological diversity to improve the accuracy and generalisability of ensembles and regionally trained models. In the following dissertation, novel methods are developed for three different modelling frameworks: physically-based hydrological models, ensembles of shallow ML models, and for deep learning models.

The objectives of the first manuscript are (i) to apply unsupervised clustering to characterise discretised, time-continuous hydrological events, (ii) to develop SWMM ensembles in which each ensemble member is calibrated to a different hydrological event, (iii) conduct a cross-comparison of models calibrated to each event cluster, and (iv) compare ensemble performance with traditional multi-event calibrated models.

The objectives of the second manuscript are (i) to determine whether training binary classi-

fiers, which predict whether or not to issue a flood warning, using preprocessed flood labels can yield better results than traditional, regression-based models; (ii) to provide a systematic comparison between three ensemble algorithms and five classifiers, as to provide recommendations on model selection; and (iii) to evaluate the trade-off between false and true positives, as to better understand the positive prediction rates of each classifier-ensemble combination. It is absolutely essential that emergency managers and flood authorities understand the trade-off between reducing missed flood alarms, while increasing the rates of false alarms.

The objectives of the third manuscript are (i) to develop a systematised framework for embedding resampling methods within ensemble algorithms, as to increase the error covariance between members and improve generalisation; and (ii) to evaluate the proposed scheme with respect to its ability to improve high streamflow performance, using explicit high streamflow metrics.

Lastly, the objectives of the fourth manuscript are (i) to apply temporal and spatial clustering to regional training datasets as a way to characterise hydrological diversity, (ii) apply cluster-informed resampling to determine whether predictions in a region of interest benefit from training to ‘hydrologically similar’ data, or whether similar data is simply redundant. (iii) the effects of adding ‘hydrologically dissimilar’ data (with respect to the region of interest) are also evaluated, to determine whether it has a positive impact on the performance of the trained model.

1.5 Data sources

This section provides a brief overview of the datasets used throughout the thesis. The following subsections, which are organised by the respective data collection/management authority and sorted by increasing spatial domain, outline the various data sources accessed throughout the dissertation. The work in this thesis overwhelmingly consists of hydrometeorological forcing data, which serves as the input data for rainfall-runoff models, and is necessary to train all the flood forecasting models featured in this work. Some spatial data was required to create SWMM models in Chapter 2, and spatial basin characteristics were required to for the regionally trained ANNs in Chapter 5.

One note on the data used in this work: all forecasting models are trained using strictly his-

torical hydrometeorological observations input data and no forecasted products, specifically precipitation. While there are numerous examples in literature of models that do not use such inputs [Tyrallis et al., 2021, Shada et al., 2022, Toth, 2009, Tiwari and Chatterjee, 2011, Girihagama et al., 2022], state-of-the-art operational streamflow forecasts do utilise precipitation forecasts [Nevo et al., 2022]. The use of such forecasts in the models featured in this work would likely improve their accuracy, especially at long lead times.

1.5.1 Locations

The locations of forecasts in this work were selected primarily based on historic flooding. Locations are largely urban, as urban areas are more populated, thus more vulnerable to flood damage. The characteristically high impervious landcover in urban areas also contributes to flood events that evolve rapidly [Zanchetta et al., 2022]. Chapters 2-4 all feature watersheds in the City of Toronto that have historically experienced large flood events, most notably in 2013 [Rincón et al., 2018]. Chapter 4 additionally includes the Bow River in downtown Calgary, which experienced devastating flooding in 2013 [Khan and Valeo, 2016]. Chapter 5 does not focus on specific locations, instead using randomly sampled basins from across Canada. This was done to improve the generalisability of the findings of the study.

1.5.2 Toronto and Region Conservation Authority

Chapters 2, 3, and 4 use hydrometric data published by the Toronto and Region Conservation Authority [Toronto and Region Conservation Authority, 2020b]. The datasets include precipitation, stage, and streamflow at a 15-minute temporal resolution. Streamflow data is available from 2007 onwards at the longest operating station, but most streamflow records begin as of 2012. Some streamflow and precipitation records are highly discontinuous, presumably due to station damage or maintenance.

The TRCA also publishes watershed, streamflow, and flood vulnerable area maps. The flood vulnerable area maps are used to identify target locations for point-source flow forecasts. For each target, watershed data is used to identify relevant predictions (e.g., upstream flow). The streamflow and watershed datasets are also used to validate the creation of SWMM models in Chapter 2. All the spatial and dynamic datasets mentioned above are hosted on the TRCA’s open data portal and available for download free of charge [TRCA, 2020]. They were most

recently accessed in January 2022. An API is also provided to access measurements in real-time.

1.5.3 Province of Ontario

Land cover data from the Southern Ontario Land Resource Information System (SOLRIS) is used to parameterise SWMM models in Chapter 2. The SOLRIS dataset includes a variety of land-use categories (e.g., types of vegetation, transportation, etc.), which are reclassified for the purpose of the study to ‘permeable’ or ‘impermeable’, which is a key SWMM parameter. The SOLRIS was accessed in September 2021 and is publicly available for download [[Province of Ontario and Ministry of Natural Resources and Forestry, 2022](#)].

1.5.4 Government of Canada

The Government of Canada hosts a trove of data. The Water Survey of Canada hosts the HYDAT database that includes streamflow and stage measurements at hourly and/or daily rates for gauges across the country. These records date as far back as 1950. Hydrometric data for the Bow River basin is used in Chapter 4. Environment Canada hosts meteorological data that includes temperature and precipitation timeseries’, which is also used in Chapter 4. The Government of Canada also hosts high-resolution DTMs that are used to delineate the SWMM models in Chapter 2. Historic HYDAT data is publicly available at a daily resolution [[Government of Canada, 2020](#)], whereas hourly data is available for purchase. Environment Canada meteorological data is publicly available at an hourly resolution. The DTM datasets used are also publicly available [[Government of Canada, 2014](#)]. The data used in this work was downloaded in September 2019.

1.5.5 HYSETS

HYSETS is an open-source aggregation of hydrometric and climatic datasets that spans all of North America [[Arsenault et al., 2020](#)]. The Water Survey of Canada and Environment Canada datasets are among them. Hydrometric data is aggregated on a basin basis at a daily temporal resolution. The dataset also contains static basin attributes, such as topographic and Landover descriptors. HYSETS is used in Chapter 5. HYSETS is currently hosted on the Open Science Framework [[Arsenault et al., 2020](#)]; the data used in this work was downloaded in March 2021.

1.6 Research significance and practical implications

Meeting the objectives outlined above has numerous practical implications. Advancing prediction frameworks for process-based models has the potential to improve model accuracy across a broader range of hydrological conditions. It also provides a new method for leveraging highly accurate ML models while retaining physically meaningful and interpretable base models. The advancement of such methods is especially important given that applications of process-based models outnumber pure data-driven models in operational contexts [Alizadeh et al., 2021, Emerton et al., 2016, Abrahart et al., 2012]; however, that is steadily changing, especially with the release of Google’s operational, ML-driven forecasting platform [Nevo et al., 2022, Nearing et al., 2023]. This work is a promising compromise between both paradigms.

The next contribution is a comprehensive cross-comparison of ensemble algorithms, with respect to their effectiveness on flood level flows. These are evaluated alongside SMOTE (synthetic minority oversampling technique), which has been widely used in broader machine learning applications, but rarely for streamflow forecasting. SMOTE is used to generate synthetic high flow training samples. Generating synthetic high flows provides the model with more training data, while minimising the risk of overfitting to the small number of observed flows.

Finally, proposed clustering-based resampling methods for constructing training datasets for regional learning have the potential to drastically reduce the computational effort needed to train and evaluate deep-learning models. The proposed methods can also be used to guide optimum data selection with respect to regions of interest, from increasingly large hydrological databases.

1.7 Dissertation outline

Chapter 1.5 introduces the hydrological datasets used within each of the four manuscript chapters. This chapter also covers data cleaning and preprocessing techniques. Chapters 2-5 are comprised of standalone manuscripts with their own introductions, methods, results, discussions, and conclusions. Each of these chapters includes a preface, publication details, and author contributions.

The first of four manuscript chapters, Chapter 2 aims to address training data selection for

process-based SWMM models for two basins in Toronto, ON. This study proposes a novel variation of multi-event model training, in which an ensemble of SWMM models are trained to a variety of hydrological events. Artificial neural networks are used to combine ensemble predictions.

Chapter 3 frames flood forecasting as a binary classification problem. Projecting continuous streamflow data into a binary space simplifies the learning challenge and has the potential to improve the accuracy of flood warning systems. A synthetic oversampling technique is also employed to improve flood flow classification. The proposed framework applied in three Toronto ON basins and systematically evaluated using a cross-comparison of machine learning model architectures and ensemble frameworks.

Chapter 4 advances the resampling technique used in Chapter 3 by embedding it within ensemble learning algorithms. Integrating resampling procedures within ensemble algorithms increases the diversity of training data used by ensemble members, thus improving model generalisation. This study is conducted on two basins, with predictions generated for Toronto and Calgary.

Chapter 5 includes an application of a deep learning model. Rather than training an ensemble of models to diverse hydrological conditions, a single deep-learning model is trained to large, hydrologically diverse datasets. This study aims to understand the extent to which hydrologically dissimilar basins are useful to model training and generalisability.

Chapter	Basin(s)	Preprocessing technique	Model(s)
Chapter 2	Single	Timeseries clustering (SOM)	Multi
Chapter 3	Single	SMOTE	Multi
Chapter 4	Single	SMOTER (embedded)	Multi
Chapter 5	Multiple	Timeseries and spatial clustering (KMeans)	Single

Table 1.1: Tabular comparison of methods used in each of four manuscripts. The ‘basins’ column indicates whether one or multiple basins are used to calibrate each model(s). The ‘pre-processing technique’ describes the clustering or resampling method used to generate training datasets. The ‘models’ column indicates whether predictions are made using one or multiple models.

Collectively, the chapters span a wide variety of scales, regions, and model types. A common theme throughout these chapters is the use of different data selection strategies for increasing the diversity of training datasets. A simplified, visual comparison of the methods used in each chapter is provided in Fig. 1.2 and summarised in Table 1.2. Rows (A) to (D) cor-

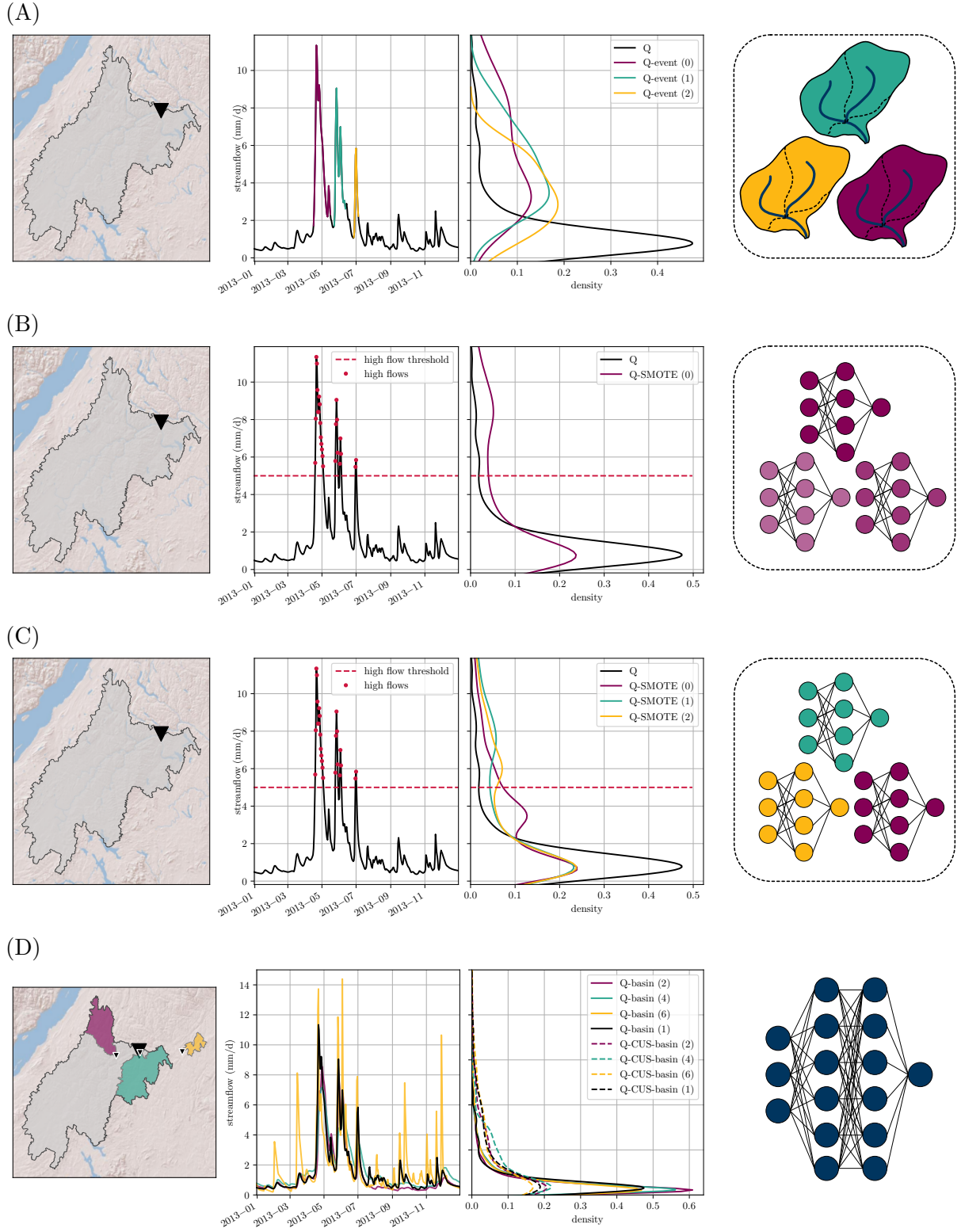


Figure 1.2: Graphical comparison of methods used in each of four manuscripts. Subplots from left to right indicate the training basin(s), the corresponding hydrograph(s), the resampled streamflow data, and the model or ensemble configurations. (A) shows the discretisation of hydrological events from a continuous record, which are used to train an ensemble of SWMM models. (B) shows SMOTE resampling applied to high flows, used to train ensembles. (C) shows SMOTE used iteratively, to create more diverse ensembles. (D) shows data from additional basins, which is undersampled and used to train a single model.

respond to chapters 2 to 5. In the first column of Fig. 1.2, rows (A) to (C) use training data from a single basin, while (D) features a model trained to multiple basins. The second column shows the streamflow record for each basin(s). In row (A), streamflow events are extracted from a continuous record. Similarly, in rows (B) and (C), high streamflow samples are extracted from a continuous record; unlike with the process-based models in (A), high streamflow samples do not need to be time-continuous for modelling purposes, therefore an event discretisation method is not needed. Finally, row (D) shows the streamflow records for all four training basins. The third column shows the histogram of raw streamflow (black) and histograms of resampled streamflow (coloured). In (A), the histograms are those of each discretised rainfall-runoff event. In (B) and (C), the histograms of the SMOTE resampled streamflow is shown, which contains a higher proportion of high streamflows. In (C), multiple coloured histograms correspond to the repeated application of SMOTE resampling within an ensemble framework. Finally, (D) shows the histograms of streamflow resampled based on K-means clustering, which is applied to obtain a balanced distribution of streamflow states. Finally, the fourth column illustrates the model(s) used for prediction. (A) features multiple process-based models, each trained to a different rainfall-runoff event. (B) features an ensemble of models, each trained on the SMOTE resampled streamflow. (C) also shows an ensemble, but the colours indicate higher diversity, as the SMOTE resampled streamflow is repeated for each ensemble member. Finally, (D) shows a single deep-learning model, which is trained to the resampled streamflow records of all the basins.

Chapter 6 summarises major findings from each chapter and provides some general conclusions. Opportunities for future work are also presented.

A single reference section is provided for the entirety of the research. Appendix A includes descriptions and links to the code used in each chapter. Appendix B includes a list of journal publication and conference presentations that resulted from this research. Appendices C, D, and E include summaries of conference presentations related to Chapters 2, 4, and 5, respectively. Conference details are included in the preface of each chapter.

2 A novel ensemble algorithm based on hydrological event diversity for urban rainfall-runoff model calibration and validation

2.1 Preface

Unlike other chapters in this dissertation, which focus on pure ML models, this first work is focused on semi-distributed physics-based model Storm Water Management Model (SWMM). SWMM is an open source engine developed by the US Environmental Protection Agency. The work originated from a National Research Council Artificial Intelligence Industry Partnership (INRAP) fund, with Civica Infrastructure Inc., a hydrological software company based out of Vaughan, Ontario. The early objective of the INRAP project was to develop a method to apply real-time corrections to a detailed SWMM model of Fourteen Mile Creek, situated in Oakville, Ontario. The model was provided in an uncalibrated state, therefore, the focus shifted from bias correction to model calibration. Due to the very high spatiotemporal resolution of the model, calibration to continuous, multi-year records was incredibly computationally expensive. Consequently, the new focus of the investigation became event discretisation and selection for automated calibration routine. Subsequently, an ANN-based bias correction procedure was developed for the Fourteen Mile Creek model. Ultimately, the discrete bias correction framework was abandoned in favour of an ensemble approach, in which the ANN is used to assimilate the predictions from multiple models, as well as real-time hydrometric observations. These topics are documented throughout the conference publications listed below. More models were required in order to properly validate the proposed method. An automated routine was developed to generate SWMM models using raw input data. Ultimately, three models were generated for catchments within Toronto and the surrounding area. These models were comparatively less complex than the Fourteen Mile Creek model, which facilitated computationally expensive experiments featured in the final manuscript. The manuscript was published in Journal of Hydrology in 2023 as follows:

Snieder, E., Khan, U. T. (2023) A novel ensemble algorithm based on hydrological event diversity for urban rainfall-runoff model calibration and validation, Journal of Hydrology, 619, 129193, <https://doi.org/10.1016/j.jhydrol.2023.129193>.

As mentioned above, early work on this topic was accepted to the 2020 EGU General Assembly (virtual presentation), the 2021 Canadian Society for Civil Engineering Annual Conference (virtual presentation and conference publication), and the 2021 EGU General Assembly (virtual presentation). The respective citations are as follows:

Snieder, E., Khan, U. T. (2020). Improved real-time SWMM flow forecasts using two machine learning approaches. Poster presentation at the EGU General Assembly. Virtual Display.

Snieder, E., Khan, U. T. (2021). Investigating event selection for GA-based SWMM rainfall-runoff model calibration. Manuscript and virtual presentation at the CSCE Annual Conference.

Snieder, E., Khan, U. T. (2021). Spatial bootstrapping for model-free estimation of subcatchment parameter uncertainty for a semi-distributed rainfall runoff model. Virtual presentation at the EGU General Assembly.

Conference publications were peer-reviewed and are included with minor revisions for the sake of fluidity. Results from conference posters presentations (which ranged from 2 to 15 minutes), that are not included in any manuscripts, are transcribed and developed into technical notes.

Contributions to this chapter are as follows: **E. Snieder:** conceptualisation; data curation; formal analysis; investigation; methodology; visualisation; writing - original draft. **U. T. Khan:** conceptualisation; funding acquisition; supervision; writing - editing, revisions.

2.2 Abstract

Hydrological models such as the stormwater management model (SWMM) are essential tools for managing water resources. Model performance typically relies on calibration to historic rainfall-runoff events. However, traditional calibration techniques suffer many issues, such as ad-hoc event selection, ignoring model parameter uncertainty, and poor validation techniques. We introduce a novel ensemble framework that relies on data-driven event selection of calibration and validation events to achieve robust model generalisation.

The proposed framework is applied to SWMM models for two urbanised catchments in Southern Ontario, Canada. The framework uses unsupervised machine learning to cluster rainfall-runoff events. Events in different clusters have diverse characteristics; sampling events from each cluster ensures a diverse set of events for calibration. Validation events can be sampled in the same manner, which ensures similarity between calibration and validation data. We study the effects that the calibration event cluster has on model sensitivity and generalisation. Only minor differences in model parameter sensitivity were observed between different clusters. However, the cluster(s) used for calibration were found to be very important for model generalisation. Next, we generate ensemble predictions, which use multiple SWMM models, each calibrated to different events. Ensemble performance is compared to the single-model baseline and across increasing numbers of ensemble members. Results demonstrate improvements in generalisation and decrease in prediction variance across increasing ensemble size. Finally, we compare several different techniques for ensemble combination, which increase in complexity from simple averaging to ANN-based stacking augmented with exogenous input features. The ANN-based stacking performs the best according to two out of three evaluation criteria. Simple averaging, while inferior in performance to stacking, consistently outperforms the single model baseline. The novel framework proposed in this work provides a data-driven, adaptable alternative to conventional calibration frameworks that have been proven to improve model performance.

2.3 Introduction

Physics-based hydrological models simulate dynamic rainfall-runoff processes. These rainfall-runoff models typically use rainfall observations to predict flow at the outlet of a catchment. Such models have a wide range of applications, such as flood early warning systems and stormwa-

ter infrastructure design. For any model application, it is important that predictions are accurate under a wide range of hydrological conditions. Stormwater management models (SWMM), a popular semi-distributed rainfall-runoff modelling platform, rely on robust calibration techniques to fine-tune uncertain model parameters and ensure accurate performance under different conditions. The efficacy of calibration can be attributed to at least three factors: improved model parameterisation, the method of searching for an optimum parameter set, and the selection of calibration and validation data [Beven, 2012, James, 2005]. Recent research has focused primarily on advancing the former two factors, while the selection of calibration and validation datasets is often overlooked in modelling studies [Niazi et al., 2017]. Robust validation is essential to verify the generalisation of the calibrated model. Models tend to generalise best on statistically similar data [Zheng et al., 2018, Bowden et al., 2002, Zheng et al., 2022]. While this can be easily achieved through resampling methods [Toth, 2009, Anttil and Lauzon, 2004a], many physics-based models such as SWMM require time-continuous data [Zheng et al., 2022], making it difficult to apply such methods.

There are two common formats for time-continuous datasets: using a long period of data, or partitioning data into time-continuous segments. These formats are commonly referred to as continuous and event-based calibrations, respectively [Hossain et al., 2019]. A review of SWMM calibration studies (between 1987 and 2014) found 16 instances of continuous calibration, 20 single-event, 52 multi-event, and 25 unspecified [Niazi et al., 2017]. The advantages and disadvantages of each approach are well documented in existing studies [Shamsi and Koran, 2017, Hossain et al., 2019, Niazi et al., 2017]. For example, model initial conditions only need to be calculated a single time for a continuous model run, whereas in an event-based format, they need to be calculated for each individual event. Continuous modelling runs are also simpler than event-based to configure [Shamsi and Koran, 2017]. However, implementing event-based calibration routines is becoming increasingly easy to automate. This is thanks to advances in modelling platforms and the development of wrappers available in multiple programming languages, which allow for multiple events, hot-start files, and model validation to be done programmatically [Leutnant et al., 2019, McDonnell et al., 2020].

Event-based calibration frameworks have several advantages. Firstly, research has shown that thoughtful selection of a few informative events can outperform continuous or randomly selected events [Singh and Bárdossy, 2012, Hossain et al., 2019]. That being said, it is possible

to adapt weighting schemes to continuous data to achieve similar improvements by increasing the weight allocated to informative events. Secondly, event-based calibration routines also simplify the curation of calibration and validation datasets that are both statistically similar and hydrologically diverse, which is difficult to achieve with two continuous segments of data. The challenge with event-based calibration becomes selecting the best events out of those discretised from a continuous record. Recent studies have demonstrated a wide range of approaches to event selection, which typically involve ranking events based on characteristics such as the return period, mean intensity, or total rainfall. [Blackler and Guo \[2014\]](#) distinguish early spring, spring, summer, and intense summer rainfall events; their study noted that long, low intensity events achieved stronger performance on the calibration data compared to short, intense events. [Broekhuizen et al. \[2020\]](#) evaluate a multi-event calibration scheme in which events are selected from a set of discretised events based on one of rainfall, rainfall intensity, mean flow, total flow relative to total rainfall, or rainfall duration. In their calibration scheme, events that fall above and below a runoff threshold are used in a two-stage calibration technique, consisting of first calibrating to one type of event, then another. Their approach was found to outperform a single stage multi-event approach. Similarly, [Talei and Chua \[2012\]](#) select events based on lag time and hydrograph shape. [Jean et al. \[2018\]](#) select events with the greatest total precipitation depth and mean precipitation intensity; their study found that selecting events based solely on total precipitation resulted in an underestimation of extreme flow frequency, and only by using a multi-criteria approach to event selection, which included mean intensity, did the selected events produce satisfactory calibration performance. [Singh and Bárdossy \[2012\]](#) demonstrate a method for identifying critical hydrological events using a statistical depth function. Models calibrated to critical events are found to perform comparably to models calibrated to the entire record, and better than models calibrated to random events. While this method is an improvement over simple ad-hoc methods, their approach only uses a single criterion (either antecedent precipitation or discharge). Recently, [Zheng et al. \[2022\]](#) proposed a novel framework for achieving statistical similarity between the calibration and validation datasets, in which a model is run on a continuous stretch of data and the model output is deterministically sampled for calibration and validation, which circumvents the need for time-continuous datasets. However, sampling time-discontinuous sets of calibration and evaluation data risks artificially high model performance compared to evaluations on time-continuous data in cases where flow is highly auto-correlated [[Padiyedath Gopalan](#)

et al., 2019, Veloz, 2009].

A common issue with the approaches reviewed above is that they tend to neglect the hydrological diversity of rainfall-runoff events [Awol et al., 2018]. Consider a rainfall event with 20 millimetres of total rainfall; this event may be short and intense, long and mild, occur in the spring or the fall, occur with dry or wet antecedent conditions. Current practices overwhelmingly rely on a single criterion, resulting in calibrated models that are not guaranteed to perform well on unseen events (which may have very different characteristics). Thus, there is an opportunity for improving calibration frameworks by explicitly identifying hydrologically diverse events, which will have a favourable impact on model generalisation.

We propose that calibration and validation events be identified using unsupervised clustering, which is a simple yet proven method for data partitioning. Clustering techniques have been widely used in research to classify individual observations, for the dual purpose of ensuring that all flow regimes are equally represented (i.e., an equal number of low and high flows), and ensuring similar statistical properties between the calibration and validation data sets [Bowden et al., 2002, Anctil and Lauzon, 2004a]. Toth [2009] used a self-organising map (SOM) to classify streamflow and rainfall data (with several lagged copies of each), evaluating the effect of varying the number of clusters has on overall model performance. The study also links the SOM-based classifications to physical hydrological phenomena (e.g., a given class may correspond to the rising limb of the hydrograph). Bowden et al. [2002] compared genetic algorithm (GA)-based and SOM-based data splitting methods, finding that they both outperformed traditional approach, in which data is split arbitrarily without any consideration for the statistical properties of each split. Similarly, Anctil and Lauzon [2004a] use an SOM to create subsets of data with statistically similar properties, and draw an even number of samples from each cluster to form their training dataset. All the examples above were conducted using machine learning (ML) models. However, as far as our knowledge extends, no works have evaluated such methods on physics-based models such as SWMM. In order to use these methods for physics-based models, they must be modified to accommodate time-continuous segments of data, which is a typical requirement for physics-based models [Guo et al., 2020]. We propose a novel adaptation of clustering methods for time-continuous data, which will allow us to identify diverse rainfall events for calibration as well as statistically similar events for validation.

Even with an improved method for identifying calibration events, event selection is inherently uncertain, which manifests as uncertain calibration results. Typically, studies that calibrate models to multiple events fail to account for this uncertainty by aggregating multiple events into a single timeseries [Krebs et al., 2013, Awol et al., 2018], or optimising the mean performance on multiple events [Broekhuizen et al., 2020, Arriero Shinma and Ribeiro Reis, 2014]. We propose creating SWMM ensembles as a way to handle calibration parameter uncertainty. An ensemble approach preserves the equally valid model parameter sets generated by calibration to different events, because predictions are aggregated as opposed to parameter estimates. A multi-model approach is motivated by ensemble theory, which has proven both theoretically and empirically that predictions produced by ensembles consistently outperform single models [Brown et al., 2005a]. The success of an ensemble is attributable to the degree of disagreement between the ensemble members, which is referred to as diversity. Diversity can be quantified as the covariance between the error distributions of ensemble members [Brown et al., 2005a]. The benefits of diversity can be explained with the bias-variance-covariance decomposition of mean squared error (MSE); since the covariance can be negative, greater covariance between models can reduce overall MSE. One way to create diversity among ensemble members is by varying the data used to calibrate each member [Brown et al., 2005a]. For event-based SWMM calibration, we suggest a novel ensemble approach in which each model in an ensemble of SWMM models is calibrated to a unique event, creating diversity between members, hence reducing overall error and improving model performance. The predictions of the ensemble member are combined to form discrete predictions, which has been shown to improve model generalisation for physics-based models [Shamseldin et al., 1997, Beven, 2012].

To support our proposed use of SWMM ensembles, the following paragraphs provide a brief review of comparable ensemble approaches in the literature and present ensemble taxonomy that is used throughout the manuscript. Ensembles can consist of model of different types or models of the same type, referred to as heterogeneous and homogeneous, respectively [Zahmatkesh et al., 2015]. Heterogenous ensembles have been widely studied and are often found to outperform any single standalone model [Kumar et al., 2015, Kauffeldt et al., 2016]. Their success can be attributed to the diversity created by varying assumptions and structures of different modelling platforms. In contrast, homogeneous ensembles differ in that they all having the same model structure, but different parameter values. Homogeneous ensembles have

received comparatively less attention in research. Physics-based hydrological model parameters are highly uncertain, which translates to uncertain predictions. Within a homogeneous ensemble framework, multiple different and equally valid model parameter sets are capable of high degrees of diversity and diverse parameter sets contribute to high ensemble accuracy [Brown et al., 2005a]. Examples of homogeneous ensembles of physics-based hydrological models are relatively uncommon in literature compared to examples of standalone models or heterogeneous ensembles. Nevertheless, research on homogenous ensembles has shown promising improvements to performance. Seibert and Beven [2009] create physics-based ensembles in which each member is trained to randomly sampled data; the ensemble approach is found to outperform the single model. Similarly, Liu et al. [2014] use rainfall-runoff models as members for the AdaBoost ensemble algorithm [Freund and Schapire, 1995], noting considerable improvements in performance.

Regardless of what type of models are used as ensemble members, literature has shown that ensemble generalisation is dependent on the specific combination technique used. Most studies that compare combination techniques do so using ML models as the ensemble base learners [Ancil and Lauzon, 2004a, Abrahart, 2003, Tyralis et al., 2021, Alobaidi et al., 2019]. There is a need to understand whether the improvements in performance achieved by combination techniques can be extended to homogeneous ensembles of physics-based models. Combining predictions obtained from different parameter sets may be seen as counter-intuitive, especially when parameters are assumed to be time invariant, which are expected to have a single correct value. However, significant work in hydrology has demonstrated that there is no single optimum parameter sets; instead, there may be numerous, realistic parameter sets that are equally suitable for a given model [Awol et al., 2018, Mediero et al., 2011, Beven, 2012, Xu et al., 2019]. Combination strategies range from simple averaging to more complex methods such as stacking, in which a statistical model used as the combiner [Breiman, 1996b]. Several studies that compare combination strategies have found that simple averaging is "hard to beat in practice", but can be improved upon using methods such as stacking (using a model to combine predictions) if configured correctly [Tyralis et al., 2021, Kumar et al., 2015].

The review above highlights the need for a data-driven framework for model calibration that ensures calibration and validation datasets are time-continuous, statistically similar, and each contain diverse hydrological conditions. Our proposed framework features an innovative event

clustering technique that we use to identify a set of calibration events. We use a novel ensemble algorithm, in which ensemble members are independently calibrated to different events, to exploit the effects that ensemble diversity have on generalisation. The final step of our framework is to combine predictions, which done using four approaches, two of which have not yet been evaluated for such ensembles.

To develop and evaluate our proposed framework, we set the following objectives:

- i To develop a multi-criteria clustering approach to select hydrologically diverse, time-continuous events for calibrating urban rainfall-runoff model. This component builds on clustering methods used in ML research and investigates their efficacy on traditional physics-based models (i.e., SWMM), and explicitly measures the effect of event selection on model performance.
- ii To fully realise the benefit of a diverse set of calibration events, we propose an alternative approach to traditional multi-event calibration. Predictions are made using homogeneous ensembles of multiple SWMM models, each calibrated to a different event, in order to create diversity between members, and hence improving model generalisation and performance.
- iii Study the effects that each cluster have on model behaviour using a traditional sensitivity analysis and by evaluating the performance of non-diverse ensemble, that is, ensembles that only models calibrated events belonging to one cluster.
- iv To investigate effects that ensemble size and combination techniques have on generalisation. While these approaches have been used in ML applications, they have received very little attention for homogeneous, physics-based models. Our study of ensemble combination must feature techniques that are ranging in complexity.

The proposed framework is applied to two case-studies in Southern Ontario, Canada, and is compared to baseline results from a single-event SWMM model using multiple performance criteria.

2.4 Methods

The methods are organised into three sections that describe the SWMM models, the data-driven framework, and validation techniques. The SWMM model description includes the standard methods used for event discretisation, calibration, and sensitivity analysis. The ensemble section outlines our novel framework for event selection and ensemble prediction. Lastly, we describe a robust validation framework to verify the proposed methods.

2.4.1 SWMM model setup

2.4.1.1 Study area and model creation

Two catchments, located within the Etobicoke Creek and Don River watersheds in the Greater Toronto Area, in Southern Ontario, Canada were selected for this study. These catchments were selected based primarily on their flood vulnerability and the heavily urbanised and rapidly changing landcover, as identified by the Toronto and Region Conservation Authority (TRCA). An advantage of these specific catchments is the availability of high resolution forcing (15-minute) and spatial data (2m), openly provided by the TRCA. The catchments under study have outlets at Spring Creek South (SCS) and Taylor Creek South (TCS), which are in the Etobicoke Creek and Don River watersheds, respectively. Table 2.1 provides catchment properties; both catchments can be characterised as urbanised based on impervious landcover, they are close to one another geographically, have a similar size, and their models contain a similar number of subcatchments. Because of their similarity, they can be used to verify the proposed framework on one another.

Table 2.1: Summary of catchment properties and select SWMM model parameters for SCS and TCS. Gauge IDs are obtained from the TRCA [TRCA, 2020].

parameters	units	SCS	TCS
outlet	[#]	HY059	HY062
num_subcatchments	[-]	10	9
total_area	[ha]	3.85e+03	3.32e+03
mean_area	[ha]	385	369
mean_imperv	[%]	73.9	56.4
conduit_length	[km]	45.3	28.5

The SWMM models used in this study were created from scratch using openly available spatial and temporal datasets. Fig. 2.1 shows the conduits, subcatchments, raingauges, and outfalls of each model. Models were constructed beginning with a user-specified streamflow gauge, which was used the outfall in SWMM. A 2-metre resolution digital terrain model (DTM) [Gov-

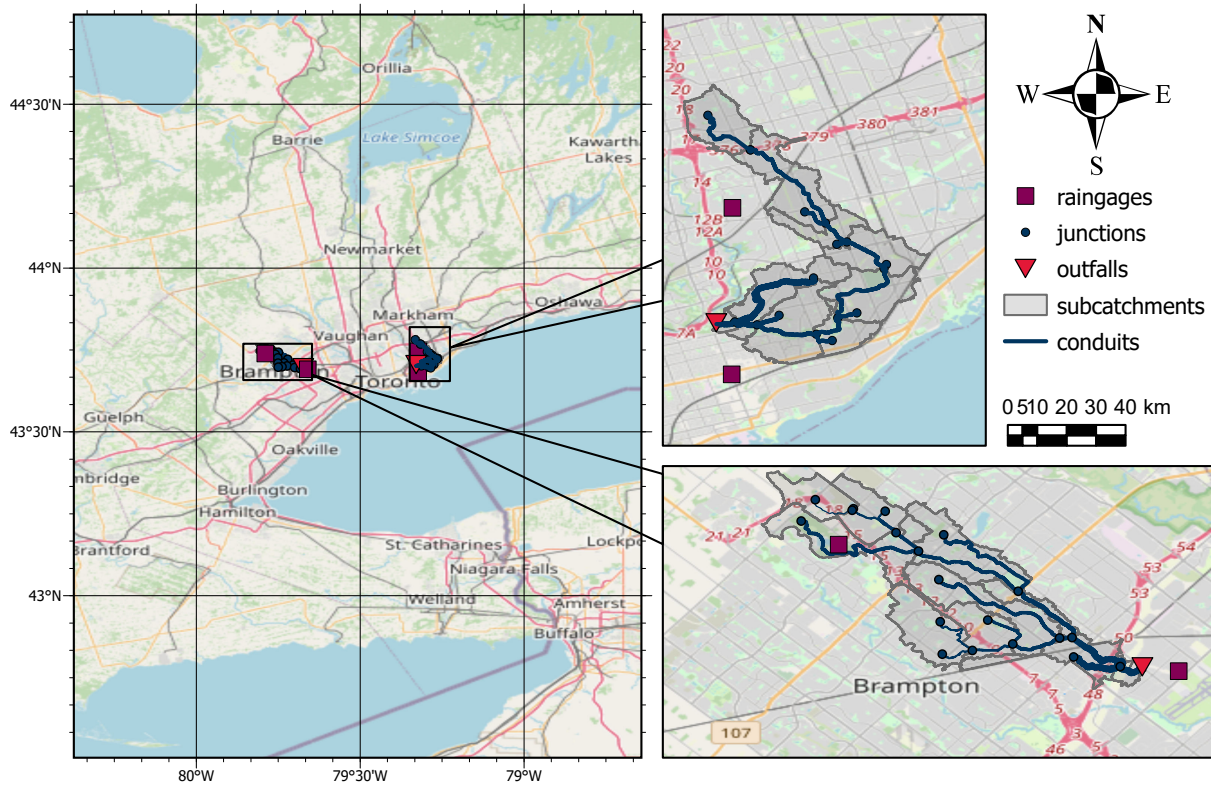


Figure 2.1: Location of catchments within Southern Ontario (left), Spring Creek South (top-right), and Taylor Creek South (bottom-right) SWMM models, including subcatchment boundaries (grey), watercourses (blue), flow gauges (red triangles), and raingauges (purple squares).

ernment of Canada, 2014] was used to calculate flow accumulation pathways and catchment boundaries, using a traditional delineation approach procedure in ArcGIS Pro [O’Callaghan and Mark, 1984, Morris and Heerdegen, 1988]. Stream burning was performed based on mapped hydrological features maintained by the TRCA [2020], which is an enforcement technique that improves the accuracy of the spatial analysis by subtracting known flowpaths from the DTM surface prior to calculating DTM derived flowpaths and catchments to encourage agreement between derived and observed flowpaths [Lindsay, 2016]. Flow accumulation pathways were further pruned to better match the existing watercourses. The catchments delineated for the SCS and TCS gauges were each discretised into subcatchments based on the nodes of the dendritic network of flowpaths. Streams were assumed to have a simple rectangular geometry; stream widths were estimated using methods described in Mulvihill et al. [2009], in which empirical regression equations are used to relate drainage area to stream geometry. These equations were found to be relatively accurate and were developed using streamflow data from the southeast coast of Lake Ontario, geographically near the catchments herein. Most channel and subcatchment parameters were calculated using spatial datasets including soil [Agriculture

and Agri-Food Canada, 2022], DTM [Government of Canada, 2014], and land cover [Province of Ontario and Ministry of Natural Resources and Forestry, 2022]. raingauges, obtained from the TRCA [2020], were assigned to subcatchments based on the minimum Euclidean distance between the gauge locations and subcatchment centroids, which resulted in two unique gauges for each catchment, listed in Table 2.1. The process above was used to generate shapefiles representing outfalls, conduits, junctions, subcatchments, and raingauges, which were exported to MATLAB to create SWMM models. Rainfall data and observed flow data was processed in MATLAB and converted to binary model files using SWMM. Events containing flags for snow and ice conditions in the records obtained from Environment Canada [2022] were removed from the selection. These events were not included because cold weather processes (e.g., snow accumulation) were not parameterised in the models. Snow events were omitted for several reasons: snow ploughing and storage is complex in urban areas and the data is unavailable, sensors often become obstructed during winter months, and cold weather events do not always produce runoff [Aquafor Beech, 2016]. Despite the omission of snow in this study, our proposed methods could be used with models in which snow processes are represented. A warm-up period of 24-hours was generated for each event under study to accommodate varying initial conditions for each event-based simulation. We developed a wrapper for SWMM in MATLAB which allowed us to run simulations and edit model parameters from the MATLAB environment. Several considerations were taken to improve the computational efficiency of the analyses conducted. The geometry of subcatchments and conduits are simplified (vertices removed) in order to minimise file write times during calibration. Binary rainfall and warmup files were generated for each event under evaluation, to reduce simulation runtimes. A complete list of routines is provided in Appendix C.1.1 and Fig. C.1 illustrates how MATLAB was used to interact with SWMM. The complete SWMM wrapper is available on [GitHub](#).

All computations were conducted on a workstation with the following specifications: CPU:AMD Ryzen 3700x; RAM: 32GB DDR4; GPU: NVIDIA GeForce GTX 1660. Software versions are Python 3.7.9 within ArcGIS Pro 2.7.0, MATLAB 2021a, and SWMM 5.1; 14 threads were used for the simulations.

2.4.1.2 Rainfall event discretisation

Rainfall and stage data were collected at point-source gauges at a 15-minute resolution for a period beginning in May 2002 and June 2006 for SCS and TCS, respectively, and ending

in May 2021. The mean aerial weighted rainfall was used to perform the rainfall event discretisation and selection for each catchment. Isolating data used to calibrate and validate the SWMM model is a two-step process: event discretisation and selection. Event discretisation is described in the paragraph below, while selection is included in Sec. 2.4.2.1, in the context of the ensemble framework.

Event discretisation is the process of identifying rainfall-runoff events from a continuous record. The most common and simplest procedure for discretisation uses a minimum inter-event time (MIT), during which there is negligible rainfall, to identify the start and end points of events [Nojumuddin et al., 2018]. While sophisticated discretisation approaches have been developed in previous research [Thiesen et al., 2019], the MIT-based approach was chosen due to its widespread use in research [Nojumuddin et al., 2018, Jean et al., 2018, Behera et al., 2010] and applied settings [Aquafor Beech, 2016]. The US-EPA and Ontario Ministry of the Environment, Conservation and Parks recommend an MIT of 12 hours [Aquafor Beech, 2016]. However, it is also important to consider catchment characteristics and runoff response while selecting an MIT value [Joo et al., 2013]. The hydrographs of the target stream gauges in each catchment have long receding limbs, therefore, a longer MIT of 24-hours was selected to ensure that the entire event hydrograph is captured in the simulation. A long MIT permits enough time for the receding limb of the event hydrograph to approach its baseflow value at the beginning of the event, uninterrupted by new rainfall events. Consequently, rainfall events are larger in magnitude and duration compared to events discretised using a shorter MIT. A buffer time of 24 hours was appended to the end of each rainfall event to ensure that the runoff response is fully captured. Lastly, events with a total rainfall of less than 5 mm were excluded from the analysis, as events of this size may not produce any runoff in urban catchments such as those used in this study [Aquafor Beech, 2016].

2.4.1.3 Sensitivity analysis

The sensitivity of rainfall-runoff model parameters is often analysed to develop a sense of model behaviour and fine-tune calibration [Swathi et al., 2019, Perin et al., 2020]. These analyses are used to distinguish sensitive and insensitive parameters, as to select only the most sensitive parameters for calibration. Existing research has shown that there is no globally sensitive set of parameters for hydrological models, and that parameter sensitivity should be calculated on a case by case basis [Knighton et al., 2016]. However, with the emergence of auto-

mated calibration techniques, the usefulness of removing insensitive parameters from calibration routines should be questioned. The sensitivity analysis presented in this study has a dual purpose. Foremost, we wanted to assess difference in parameter sensitivity between events in different clusters. Additionally, we wanted to test whether removing insensitive parameters offers any improvements to model calibration in terms of computational time or performance. The sensitivity analysis in this research was conducted for each individual event.

The sensitivity analysis was conducted using a one-at-a-time approach, which has been widely used in rainfall-runoff modelling [Shin and Choi, 2018, Xu et al., 2019, Shahed Behrouz et al., 2020, Campolongo et al., 2007]. Instead of analysing the sensitivity of individual parameter values, parameters are individually adjusted across all subcatchments. For example, to analyse slope, the slope of all subcatchments is adjusted by some fixed amount, as opposed to varying each subcatchment independently. This drastically reduces computational costs at the expense of information on the sensitivity of individual parameter values, while still being generally effective in identifying sensitive parameters [Barco et al., 2008, Shahed Behrouz et al., 2020, Campolongo et al., 2007]. Total volumetric flow and peak flow were used as model response variables for the sensitivity analysis, which are given by the following respective equations:

$$\hat{q}_v = \sum (\hat{q}\Delta t) \quad (1)$$

$$\hat{q}_p = \max(\hat{q}) \quad (2)$$

where $\hat{q}_v$, $\hat{q}$, Δt , $\hat{q}_p$, are the predicted total flow volume, predicted flow timeseries, timestep, and predicted peak flow. The timeseries predicted by SWMM, $\hat{q}$, are formulated as a function of a parameter set X and observed rainfall p :

$$\hat{q} = f(X, p) \quad (3)$$

Sensitivity is measured by adjusting each parameter relative to their initial value x_0 by fraction α :

$$I_\alpha = \frac{(f(x_0(1 + \alpha), p)) / f(x_0, p)}{\alpha} \quad (4)$$

where I_α is the sensitivity of some parameter x , with an initial value x_0 that is modified by a

fraction α when subject to a precipitation event p .

The parameters selected for calibration, sections, units, relative uncertainty, and upper and lower constraints used in this research are listed in Table 2.2. Calibration parameters were selected based on a combination of expert knowledge and parameter estimate uncertainty. Where a physical constraint on a parameter exists (e.g., the upper limit of percent impervious landcover), we truncated parameter values to meet the constraint and recalculated parameter sensitivity based on the subcatchment area weighted mean.

Ten values for α were considered, which increase linearly from the lower bound of parameter uncertainty to the upper bound. For example, the percent imperviousness, which has a relative uncertainty of 0.2, is evaluated for α values of $\pm 0.04, 0.08, 0.12, 0.16$, and 0.20 . The overall sensitivity I of the model to a given rainfall event p was taken as the absolute mean across a total of 10 (denoted as M) values of α considered:

$$I = \frac{1}{M} \sum_{j=1}^M |I_{\alpha_j}| \quad (5)$$

Following the sensitivity analysis, we conduct calibrations using different configurations of parameters, to test whether removing insensitive parameters is useful. The study of calibration configurations was conducted on one event from each cluster for each catchment. The configurations used in this study are as follows:

- i Baseline-Uncertainty calibration
- ii Double-Uncertainty calibration
- iii Sensitive-Parameter calibration
- iv Insensitive-Parameter calibration

We refer to calibration using expert-knowledge informed uncertainty as Baseline-Uncertainty. The Double-Uncertainty configuration uses the expert-based relative uncertainty multiplied by a factor of 2, which is used to assess whether further improvements to model accuracy can be achieved by expanding the parameter search beyond expert judgement. Lastly, the Sensitive-Parameter and Insensitive-Parameter use the five most sensitive and insensitive parameters,

respectively, as determined by the sensitivity analysis.

Table 2.2: Calibration parameter names and sections in SWMM, units, relative uncertainty, and upper and lower constraints. Constraints are based on physically possible limits, which are determined from published ranges and SWMM input value constraints. Detailed descriptions of each parameter are available in the SWMM user manual [Rossman, 2015].

parameter description	parameter name [units]	relative uncertainty	lower bound	upper constraint
characteristic width	Width [m]	2.00	0	1e+06
mean subcatchment slope	PercentSlope [frac]	0.30	0	5
mean impervious landcover	PercentImperv [%]	0.20	0	100
impervious landcover roughness	N_Imperv [-]	0.10	0	0
pervious landcover roughness	N_Perv [-]	0.50	0	0.8
impervious depression storage	S_Imperv [mm]	0.20	0	2.5
pervious depression storage	S_Perv [mm]	0.50	0	7.6
Green & Ampt suction head	Suction [mm]	0.50	0	320
saturated hydraulic conductivity	Ksat [mm hr ⁻¹]	0.50	0	120
initial soil moisture deficit	IMD [frac]	0.30	0	0.4

2.4.1.4 Model calibration

As mentioned above, SWMM predictions are a function of a calibration parameter set and rainfall event. Event-based calibration is given by the following general expression:

$$x \leftarrow \operatorname{argmax}(g(q, f(X, p))) \quad (6)$$

where $g()$ is the objective function given in Eqn. 7 below, $f()$ is the SWMM model with parameter set X (bound by uncertainty and physical constraints), forcing rainfall data p , and observed flow q . The Kling-Gupta Efficiency (KGE) was used as the objective function in this work [Gupta et al., 2009]:

$$\text{KGE} = 1 - \sqrt{(r - 1)^2 + (\alpha - 1)^2 + (\beta - 1)^2} \quad (7)$$

where:

$$r = \frac{\operatorname{cov}(q, \hat{q})}{\sigma_q \sigma_{\hat{q}}} \quad (8)$$

$$\alpha = \frac{\sigma_q}{\sigma_{\hat{q}}} \quad (9)$$

$$\beta = \frac{\mu_{\hat{q}}}{\mu_q} \quad (10)$$

where σ and μ denote the standard deviation and mean statistics, respectively, for flow q and predicted flow $\hat{q}$.

A wide variety of optimisation algorithms have been applied for SWMM calibration [Niazi et al., 2017]. We used a standard GA for because of its popularity, competitive performance when compared against other common optimisation algorithms, and because the optimisation algorithm itself is not a focus of our research [Jeon et al., 2014, Niazi et al., 2017].

The GA is an evolutionary algorithm that searches for a global minimum by iteratively breeding new populations of parameter estimates. Each new population is created using crossover and mutation, which are applied to the best performing populations in each generation. Important hyperparameters for the GA include population size, mutation rate, crossover fraction, to which we assigned values of 40, 0.01 (default), and 0.8 (default), respectively. The GA terminates when it meets one of several stopping criteria, which included maximum number of generations, function tolerance, and maximum stall generations, for which we used values of 100, 1E-4, and 5, respectively. When the GA does not improve the objective function by the tolerance for five consecutive generations, the algorithm terminates. GA hyperparameter sensitivity was assessed using trial and error. Increasing the population size beyond 40 had little effect on the optimisation, and the algorithm typically terminated due to the objective function levelling out at approximately 40 generations, well before the maximum number. The GA was implemented using the native GA algorithm in MATLAB, which is included in the optimisation toolbox [MATLAB, 2021].

2.4.2 SWMM ensembles

Typically, SWMM predictions are generated using a single model, which is calibrated to one or multiple events. In this study, we propose a data-driven ensemble framework for SWMM, that uses models that are independently calibrated to individual events as base learners. The novel framework is depicted in Fig. 2.2, which begins with discretising events from a continuous record. Next, diverse events are identified using our novel clustering method described in Sect. 2.4.2.1. A set of events are sampled based on their cluster to individually calibrated SWMM models, which form ensembles. Finally, ensemble predictions are combined using one of four combination methods described in Sect. 2.4.2.2, two of which are innovative in that they can utilise real-time hydrometric data to improve ensemble accuracy. Our proposed

framework will provide a data-driven solution to calibration event selection, preserve multiple model parameter sets, and the ensemble is expected to improve generalisation.

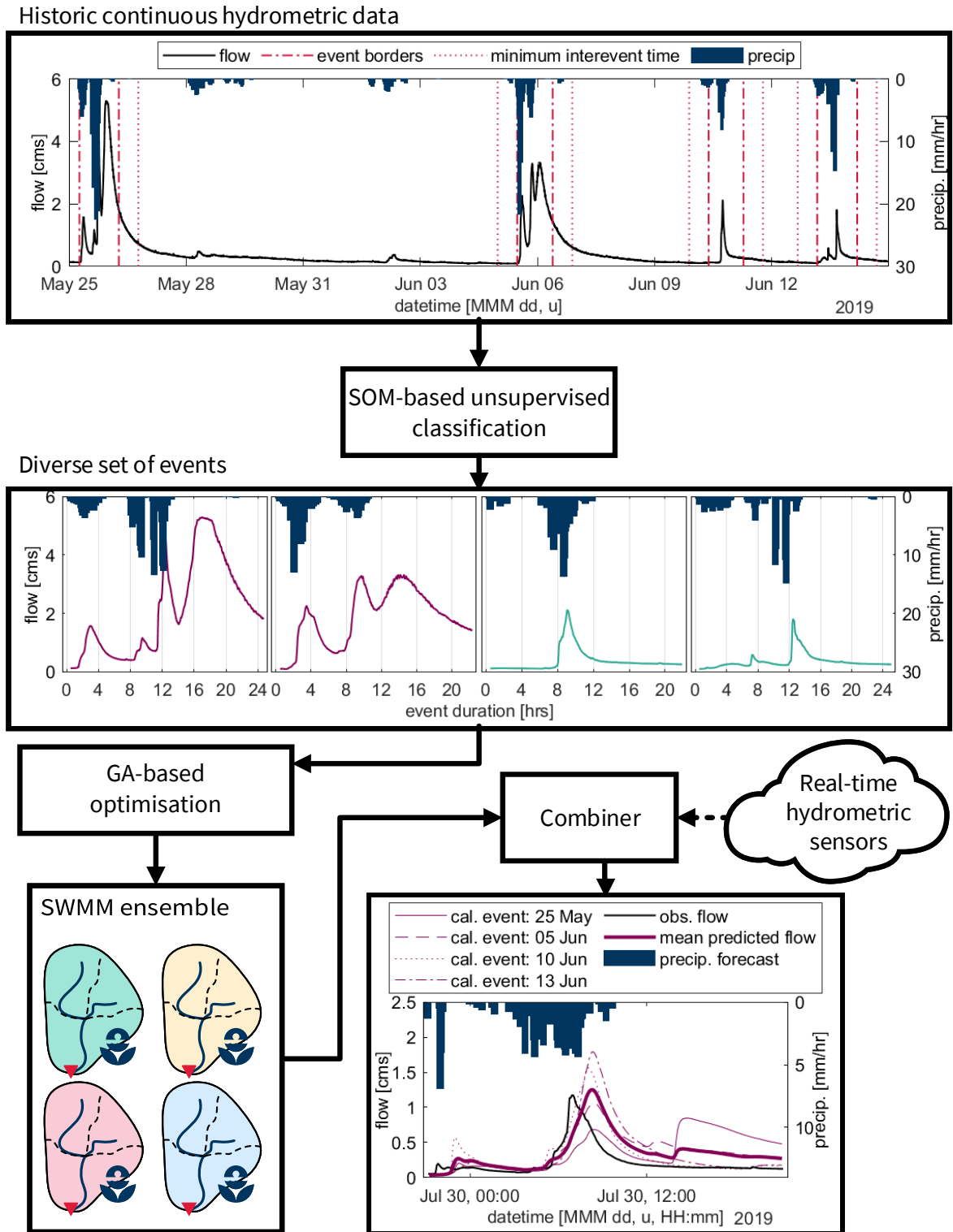


Figure 2.2: Complete novel ensemble framework including event discretisation, event clustering, single model optimisation, and ensemble combination.

2.4.2.1 Identification of diverse hydrological events using unsupervised clustering

Clustering is proposed as a way to meet our first objective of identifying diverse, time-continuous events for model calibration and validation. We apply clustering to characteristics derived from a set of rainfall-runoff events. While event selection is typically conducted using only rainfall-based event characteristics [Broekhuizen et al., 2020, Szelag et al., 2022, Sun et al., 2014], we simultaneously considered both rainfall and flow characteristics [Toth, 2009], which encourages hydrologically diverse events. Our novel approach uses clusters events based on event characteristics including: total precipitation P , peak intensity i_p , mean intensity i_μ , duration D , date, peak flow q_p , flow volume Q , hydrograph centroid coordinates q_{ct} and q_{cy} , and hyetograph centroid coordinates p_{ct} and p_{cy} . The selection of these specific features as the criteria with which to distinguish diverse rainfall-runoff events is supported in existing studies on event selection [Barco et al., 2008, Hossain et al., 2019, Blackler and Guo, 2014]. Additionally, the event date (day of year) was also used, as the time of year is important to consider in distinguishing between different hydrological events [Hossain et al., 2019, Barco et al., 2008, Blackler and Guo, 2014]. Including the date among the event statistics enables the clustering algorithm to distinguish between annual seasons, as events of similar magnitudes occurring at different times of the year may have different dominant hydrological processes. The day of year of each event is converted to radians, then the Sine and Cosine are taken for the features, which is a way to pre-process dates that preserves the cyclical nature of the days of the year [Abraham et al., 2001].

We used an SOM to cluster rainfall-runoff events, which is an unsupervised neural network-based classification technique. Several studies have demonstrated the use of SOMs for hydrological timeseries clustering as a pre-processing technique for ML models [Toth, 2009, Anctil and Lauzon, 2004a]. Typically, examples of clustering-based training data selection will select an even number of samples from each cluster, discouraging model bias towards particular flow conditions. In our framework, this is achieved by selecting the same number of events from each cluster. Maximising the number of clusters is preferable, as it would increase the diversity of represented flow conditions and in turn, the diversity of the calibrated ensemble members. However, a disadvantage of more clusters is that it may result in clusters that only contain one or two events. Thereby, if we want to draw a balanced number of events from each

cluster, we are constrained by the number of events in the smallest cluster. Consequently, we use only two clusters in our study. Using two clusters results in an ample number of events in cluster, enabling us to study the effects of model performance across an increasing ensemble size and complete our third objective, something that would not be possible with a greater number of clusters. Our proposed event selection scheme is as follows:

- i Cluster events into K clusters, in which K is constrained by the number of events in the smallest cluster (N)
- ii Select N events from each cluster, prioritising events nearest the cluster centroids
- iii Calibrate individual SWMM models to each event ($K \times N$ events if using an even number from each cluster)
- iv Generate discrete predictions by combining predictions from all SWMM models

Our second objective is to exploit the diversity created by clustering events by creating a homogenous ensemble of SWMM models. To assess differences in performance between ensembles that use events from different clusters, we evaluated events in both clusters using: (1) only models calibrated to Cluster 1 events; (2) only models calibrated to Cluster 2 events; and (3) using models calibrated to an equal number of events in each cluster. Ensemble predictions that use events from a single cluster are denoted as ECx where x is the numeric cluster label, and ensembles that use a balanced number of events from each cluster are denoted ECB. This will allow us to confirm whether models calibrated to a given type of event will perform best on the same type of event, while also quantify the effects of including a balanced amount of information on each event cluster. These configurations were also studied across an increasing ensemble size, in order to better understand how many individual models are required for satisfactory performance.

2.4.2.2 Combination methods

To create an ensemble, a set of M models are combined to form a single prediction, according to Eqn. 11 below:

$$\hat{q} = h(\hat{q}_1, \hat{q}_2, \dots, \hat{q}_M) \quad (11)$$

where $h()$ is the combiner function that combines predictions $\hat{q}$ for models calibrated to events 1 to M .

Our third objective is to investigate the performance of different ensemble combination techniques. We evaluate four common combination techniques; two weighted approaches: uniform and non-uniform weighting, and two stacking approaches: multiple linear regression (MLR), and artificial neural network (ANN)-based stacking [Shamseldin et al., 1997, Tyralis et al., 2021]. The stacking combiners have the option to leverage real-time hydrometric data, to further improve prediction accuracy.

Weighted combination is among the most popular techniques for aggregating or combining ensemble predictions [Sharkey, 1996]. Uniform weighting is the most common weighting scheme and has been described as "hard to beat in practice" [Tyralis et al., 2021]. There are methods for tuning the ensemble combination weights, which are capable of outperforming uniform weighting schemes [Tyralis et al., 2021]. In this research, we evaluate both uniform and non-uniform weighting schemes. For the weighted scheme, the weight allocated to each ensemble member is calculated as the mean cross-validation performance on the events used to calibrate the other ensemble members. In a weight vector of length M , the weight of an individual model is given by Eqn. 12:

$$w_m = \frac{1}{M-1} \sum_{\substack{i=1 \\ i \neq m}}^M g(q, \hat{q}_i) \quad (12)$$

The weight vector is then normalised such that its sum is equal to one:

$$w_{norm} = \frac{w}{|w|} \quad (13)$$

The remaining two combination methods use stacking, which is a combination technique in which a model is trained to combine other models (commonly referred to as a meta-learner). Originally, stacking was proposed using multiple linear regression (MLR) [Breiman, 1996b], however, a wide variety of different models can be used as the meta-learner. More complex models, such as ANNs, can map non-linear relationships between input data (SWMM predictions) and target data (observed flows), providing greater compensation for potential deficien-

cies of the SWMM models [Sharkey, 1996]. However, a more complex meta-learner introduces the risk of overfitting, hence poor generalisation. To determine whether increased complexity of the meta-learner is warranted, this work evaluates both a linear model and ANN.

The input datasets used to train the stacking meta-learners are constructed by first concatenating the predictions obtained for each event from each SWMM model. One advantage of the stacking approach is the ability to supplement the meta-learner with historic and exogenous data [Bento et al., 2021]. The predictions of each SWMM model in the ensemble are used as the input feature set. Next, we augment the input feature set using flow (lagged by a forecast lead-time of 12-hours) and aerial mean precipitation data. Five lagged recurrences of each flow and precipitation are used in the feature set; lag times are determined based on maximising the cross-correlation function between each of the exogenous input features and the observed flow [Tiwari and Chatterjee, 2010a].

The configuration of the linear model is trivial, but some consideration is required for the configuring the ANN. Best-practices are employed to encourage strong ensemble generalisation. Specifically, we use stop-training and Bagging, which have both been shown to improve ANN generalisation [Anctil and Lauzon, 2004a]. Due to being highly flexible models, ANNs are prone to overfitting. Stop-training involves withholding a subset of ANN training data from the calibration procedure to use as a calibration termination criterion. During training, if the performance of the withheld data does not improve across multiple ANN parameter adjustments, training is terminated. The number of consecutive parameter adjustments with no improvements until training termination is referred to as the patience. For our ANNs, we withheld 50% of the training data for stop-training and used a patience value of 6. The stop-training data is allocated using block-splitting; the training data is divided into 10 time-continuous blocks and alternately assigned for training and stop-training. Block splitting, in contrast to random sampling, preserves the time dependency structure contained within the flow data [Li et al., 2010]. Bagging is a widespread ensemble technique in which the training data for each ensemble member is resampled from the original training set, with replacement and the average model predictions are used. Bagging is successful due to the diversity obtained from the varying training data for each ensemble member, which is the same mechanism that motivated our proposed ensemble framework. Grid-search optimisation was used to evaluate ensemble sizes of 2^n for n values of 1 to 7. Ensembles were evaluated based on 50%

randomly selected validation data. Consistent with previous studies [Shu and Ouara, 2007, Zaier et al., 2010], increases in ensemble size were found to improve generalisation. Performance typically stabilised at an ensemble size of 32, the largest size of 128 was used, as it ensures sufficient resampling and only has a marginal impact on computation time. Grid-search optimisation was also used to determine the ANN model architecture, which consists of a single layer of 8 hidden neurons. The Levenberg-Marquardt backpropagation algorithm is used to train the individual ANNs. A robust validation procedure, described in the upcoming section, further helps to ensure that the stacking models are parameterised properly.

2.4.3 Validation techniques

While comparing different calibration procedures, the objective function (KGE, described in Eqn. 7) is used, which ranges from an optimum value of 1 to $-\infty$. While there is no inherent reference point for KGE values, predictions that are more accurate than the historic observed mean produce a KGE greater than -0.41 and are considered acceptable, whereas KGE values less than -0.41 are considered unacceptable [Knoben et al., 2019]. For the comparison of ensemble combination techniques, additional performance metrics are used, including the peak flow error (PFE), which is the difference in the maximum observed and maximum simulated flows, and the mean volume error (MVE), which is the difference in observed and simulated total flow volume during the event. The latter two are defined as follows:

$$\text{PFE} = \frac{\max(q) - \max(\hat{q})}{\max(q)} \quad (14)$$

$$\text{MVE} = \frac{\sum(q - \hat{q})}{\sum(q)} \quad (15)$$

where q and $\hat{q}$ are the same as described earlier. In physics-based hydrological model calibration literature, validation is often performed on a single event, or altogether absent [Niazi et al., 2017, Cristina et al., 2017]. This is problematic, especially for applications such as flood forecasting, since a physics-based model calibrated to a particular event(s) is not guaranteed to maintain its accuracy on other events, especially events with flow conditions that are dissimilar to those contained in the calibration data [Beven, 2012].

Since multiple SWMM models are being calibrated in this study, each model is evaluated on each event, excluding the event used for calibration. For example, if there are 10 discretised events, each event will be evaluated using the 9 remaining models, and thus have 9 performance scores. For ensemble predictions, which use predictions from multiple SWMM models, it is necessary to consider different combinations of models that can be used to generate predictions for each validation event, as different combinations of predictors will produce different ensemble predictions. For example, consider a set of 10 events, which we want to evaluate using ensemble predictions that contain 4 members. For each event, there are 9 models available with which to produce ensemble predictions, resulting in 126 unique combinations of models that can generate predictions (given by $9C4$). Ideally, we would evaluate every single unique combination of predictors, however, this becomes very computationally expensive for methods such as stacking, where a unique meta-learner is trained to each unique combination. Another problem that arises is that trying every possible combination does not scale well with increasing numbers of available models. In cases with more than 100 unique combinations, we randomly sample 100 combinations with which to make predictions. Since different subsets of models do not perform the same, using resampled subsets quantifies the associated variability in performance.

2.5 Results

This section presents the results of the event selection, sensitivity analysis, calibration, and ensemble predictions.

2.5.1 Event discretisation and selection

The outcome of the SOM-based unsupervised clustering approach is illustrated in Fig. 2.3 for SCS and 2.4 for TCS, in which clusters are distinguished by colour. Sample hydrographs and cluster summaries are also provided in Sec. C.1.2. Differentiating characteristics can be identified from the bottom-most rows and from the histograms along the diagonals in the scatter-plot matrices. For both catchments, events are segregated based primarily on total flow and precipitation; Cluster 1 events (green) tend to be small and frequent ($n=259$, 250 for SCS and TCS), while events in Cluster 2 (purple) tend to be large, and infrequent ($n=45$, 34 for SCS and TCS). Additionally, events in Cluster 1 tend to have a shorter duration and lower mean intensity compared to those in Cluster 2. The day of year can be read from the Sine and Co-

sine components, which form a ring as shown in Figs. 2.3 and 2.4 when plotted against one another. The midpoint along the Sine axes and upper limit of the Cosine axes correspond to January 1 and the day of year increases in the clockwise direction, with one rotation corresponding to a year. There is no clear discrimination based on event date, though Cluster 2 events tend to occur mostly during summer to late fall. The event date is expected to have a larger influence on the clustering outcome as the amount of available data and the number of clusters increases. Since the numeric cluster ID is arbitrarily assigned by the unsupervised learning algorithm, events are reclassified based on the ascending rank of the total flow of each cluster centroid, to achieve a consistent numeric notation between catchments.

Cluster properties are summarised in Table C.1 and two representative events from each cluster are illustrated in Fig. C.2. A set of events for the analyses presented in the following sections is generated by sampling 10 events from each cluster. Events are selected based on minimising their Euclidian distance between each event and the corresponding cluster centroid. Ensuring events are representative of their cluster encourages dissimilarity between events in different clusters; otherwise, there is a risk selecting events that are near the cluster boundary and thus, not diverse. The analyses presented in the following sections are conducted using strictly the subsets of 20 events (10 from each cluster) for each catchment. The properties of these sampled events are provided in Tables C.2 and C.3 for the SCS and TCS subcatchments, respectively.

2.5.2 Sensitivity analysis

The following one-at-a-time sensitivity analysis was conducted to determine whether there are significant differences in parameter sensitivity between Cluster 1 and 2 events. Parameter sensitivities for events in Clusters 1 and 2 are plotted against one another in Fig. 2.5. Each parameter has a unique colour-symbol combination, and the uncertainty lines indicate the full range of total flow (left) and peak flow (right) sensitivities across the 10 events in each cluster.

The sensitivity analysis results are in complete agreement for both catchments. The subcatchment imperviousness and width are two most sensitive parameters for peak flow and volume. Peak flow is also sensitive to changes in the overland flow Manning's N for impervious surfaces and subcatchment slope. This result indicates that runoff from impervious surfaces is the dominant process contributing to the runoff response of both catchments. The model sen-

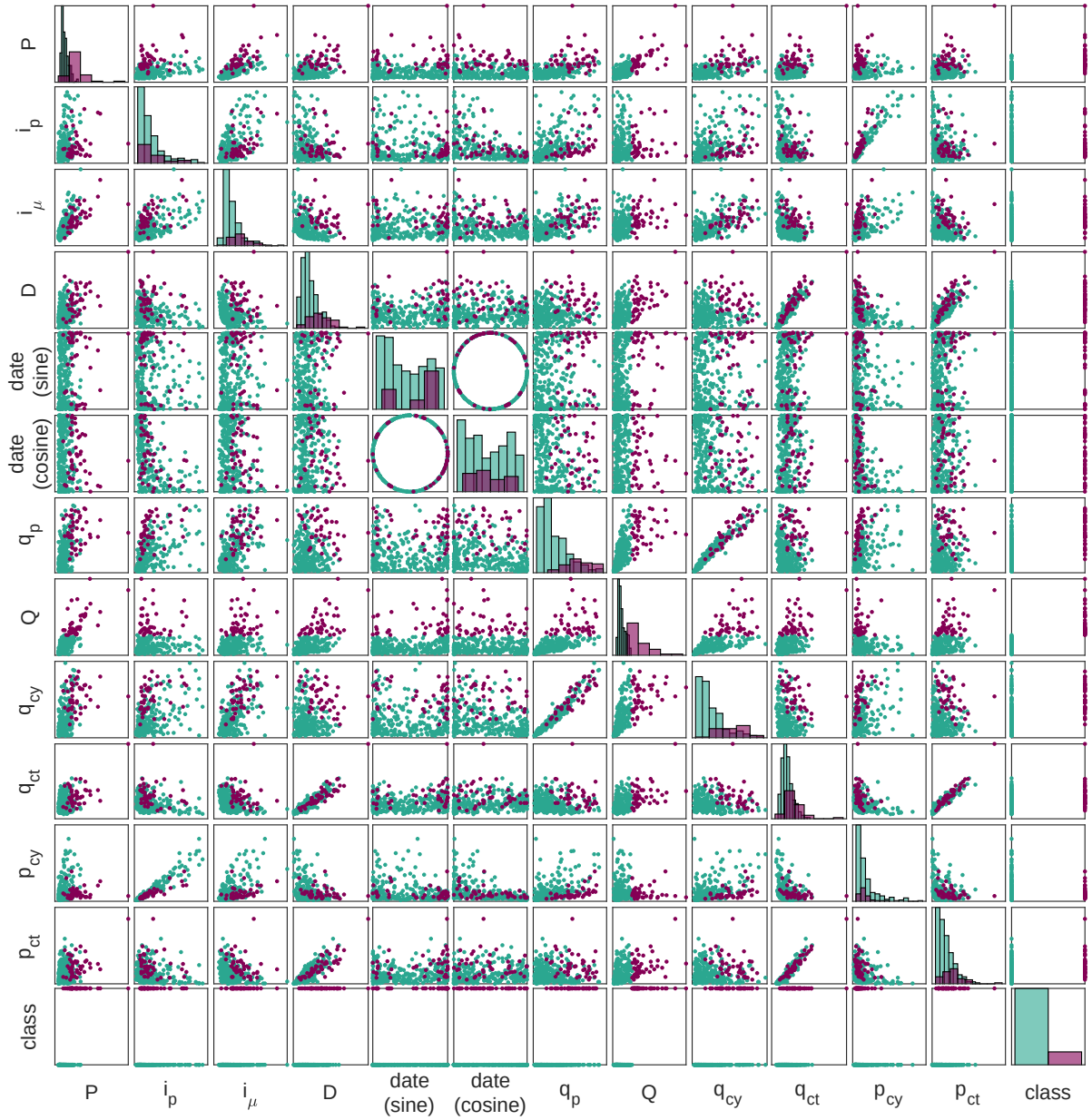


Figure 2.3: SCS event features plotted against one another in scatter-plots. Histograms for each event feature are along the diagonal. Colours indicate the cluster ID; Cluster 1 events are green and Cluster 2 events are purple.

sitivity to subcatchment slope, width, and Manning’s N are likely disproportionately high, due to the lack of a modelled minor stormwater system, the parameters of which would likely also impact peak flow. Notably, the antecedent moisture content (IMD parameter), which is widely considered to be a very important parameter for rainfall-runoff model performance [Tramblay et al., 2010], is insensitive to both event clusters and catchments.

The position of parameter sensitivity values relative to the 1:1 line illustrates whether it is more sensitive to Clusters 1 or 2: parameters below the 1:1 line are more sensitive to Cluster

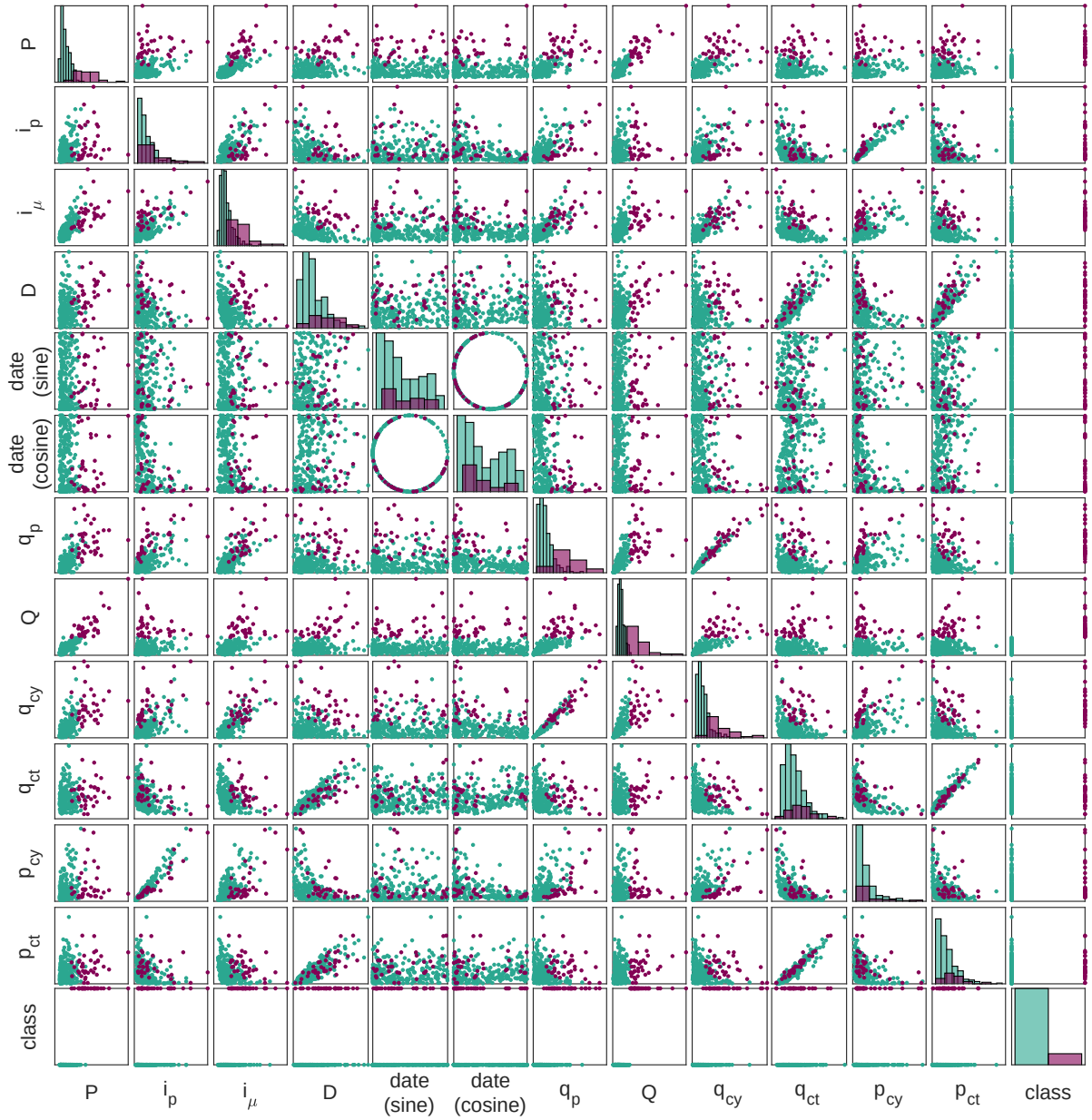


Figure 2.4: TCS event features plotted against one another in scatter-plots. Histograms for each event feature are along the diagonal. Colours indicate the cluster ID; Cluster 1 events are green and Cluster 2 events are purple.

1 events and parameters above the line are more sensitive to Cluster 2 events. There are no drastic differences in parameter sensitivity between the two clusters. Parameter sensitivities are centred about the 1:1 line, indicating sensitivity to types of events, though parameters are all marginally more sensitive to Cluster 1 events. Sensitivities are also more variable to different Cluster 1 events, illustrated by the uncertainty lines typically being broader than they are tall. The high sensitivity of the percent impervious parameter is consistent with recent research by [Awol et al. \[2018\]](#) for a SWMM model in a nearby catchment with similar land

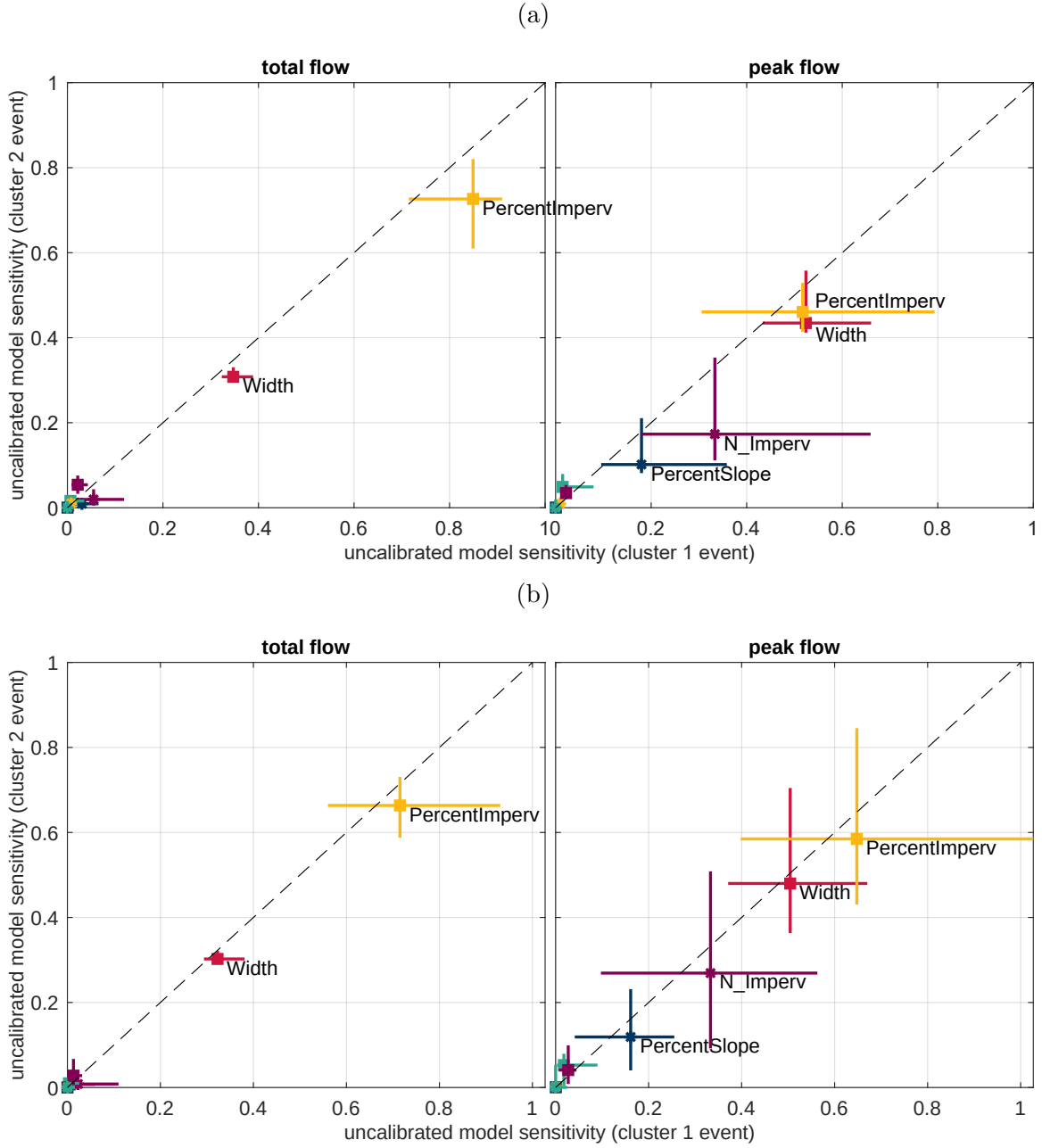


Figure 2.5: Scatterplots showing median sensitivity of total flow (left) and peak flow (right) of the uncalibrated model to Cluster 1 (x-axis) and 2 (y-axis) events for SCS **(a)** and TCS **(b)** catchments. Whiskers indicate range of sensitivities for all events in each cluster. Only parameters with sensitivity values greater than 0.1 are labelled.

cover and geography.

Sensitivity analyses often serve to identify insensitive parameters, in order to remove them from GA-based model calibration. However, our results illustrate that this does not yield any improvements in terms of calibration time or performance. Fig. 2.6 illustrates the maximisation of KGE plotted against time (recorded at each new GA generation) for the Baseline-Uncertainty calibration (expert knowledge-based) and for the three other configurations: Double-

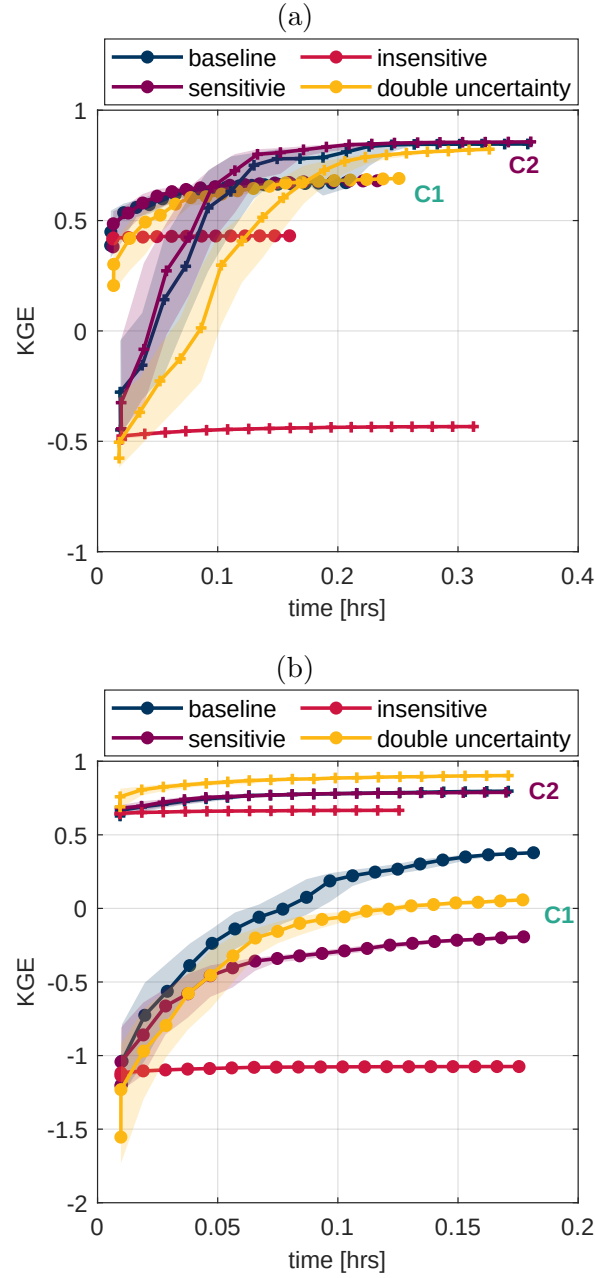


Figure 2.6: Four GA-based calibration routines for the SCS (a) and TCS (b) catchments including: Baseline-Uncertainty, Double-Uncertainty, Sensitive-Parameter, and Insensitive-Parameter calibration. Calibrations to events in Cluster 1 and 2 are identified by the green and purple annotations.

Uncertainty, Sensitive-Parameter, and Insensitive-Parameter calibration. The scores of each calibration routine are shown as a function of time for calibrations to each type of event. These are included in a single plot, with the event cluster identified by a right-aligned label. For both catchments, the models obtain better performance on Cluster 2 events (larger, infrequent) compared to Cluster 1 events, meaning that the model is better suited to large rainfall-runoff events. In all cases, the calibrated models obtain a positive KGE value.

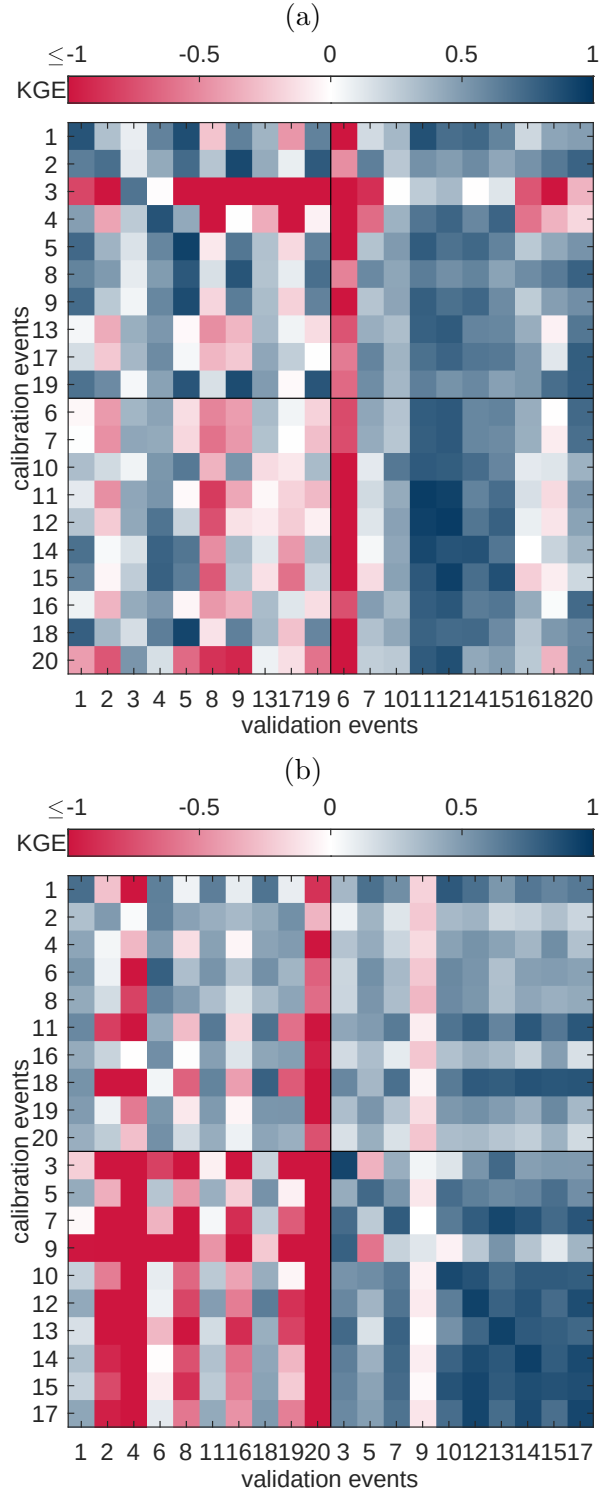


Figure 2.7: KGE values for the SCS **(a)** and TCS **(b)** catchments, coloured from good (blue) to poor (red) performance, obtained for each single-event calibration routine (along the diagonal), cross-validated against each other calibration event (upper and lower triangles). Events are sorted by cluster.

The baseline and Sensitive calibrations perform very similarly for three out of four events, suggesting that insensitive parameters can in some cases be omitted from calibration, as is

done in [Hossain et al., 2019, Awol et al., 2018, Shahed Behrouz et al., 2020, Niazi et al., 2017]. The outlier, in which the Sensitive parameter configuration achieves a poor calibration score, suggests a limitation of the one-at-a-time sensitivity analysis. Comparisons between the Double-Uncertainty and Baseline configurations confirm that the expert knowledge-based uncertainty bounds are well-defined, and the GA cannot achieve a much better solution compared to the Baseline configuration; in fact in one case, the Double-Uncertainty case is worse than the Baseline, which is attributable to the much larger search space. The Insensitive-Parameter calibration validates the outcome of the sensitivity analysis, as the GA was unable to produce any noteworthy improvements.

2.5.3 Single event calibration outcome

The outcome of all the single-event calibrations and cross-validation is shown in Fig. 2.7. The heatmaps show the performance of all 20 models, each calibrated to a single event, on every event within the set. The colours show the KGE of each combination; blue to red indicate good to poor performance. The diagonal of the matrix indicates the KGE achieved during calibration, in other words, the event used for calibration and validation is the same. Strong cross-validation of a model calibrated to a given event is indicated by a row dominated by blue colours, indicating a high KGE when cross-validated on other events.

From the lower-right and upper-left quadrants of these figures, models calibrated to Cluster 2 events tend to perform better on events in their own cluster compared to the performance of events calibrated on Cluster 1 events. The models calibrated to Cluster 2 events also exhibit better validation across clusters, illustrated by the KGE values in the upper-right quadrant, which are similar to those obtained on their respective cluster. However, the opposite is not true. As shown in the lower-left quadrants, models calibrated to Cluster 1 events do not perform as well as they do on events in their respective cluster.

As expected, the highest KGE on any given event is achieved during calibration (along the diagonal), though there are some exceptions. The calibration KGE for events 6 (SCS) and 4 (TCS) are surpassed by models calibrated to different events, indicating that a more optimum parameter set may exist than that returned from the GA optimisation. However, the difference in performance is marginal, and given that none of the other calibrated models achieve satisfactory performance on those events. This suggests that unsatisfactory performance may

not be due to the optimisation algorithm itself, but rather that a better solution does not exist within the optimisation search space.

Colour trends in the vertical direction indicate how well different models perform on a given event. For example, squares in the vertical direction show that none of the calibrated models achieved acceptable performance on the event, as is the case for event 6 (SCS), and 20 (TCS). Likewise, dominant red colouring along the horizontal direction, such as for events 3 (SCS) and 9 (TCS) indicate that models calibrated to those events do not perform well on any other events. There are some cases, such as events 3 (SCS) and 3 (TCS), where the model achieves a good KGE value during calibration but fails to generalise to most other events within the set. Meanwhile, many other models achieved satisfactory performance across the set of events, indicating that the model parameterisation and uncertainty are not at fault. Instead, poor cross-validation may be due to the GA overfitting the SWMM model. These results demonstrate the poor cross-validation performance of single-event SWMM models as well as the high variability in performance across events, especially across different clusters. The poor cross-validation performance highlights the opportunity for ensemble techniques to improve generalisation. Ensembles are formed by combining the predictions obtained from models that are each calibrated to different events, which is discussed in the following section.

2.5.4 Comparison of ensemble member configurations

Here we present a comparison in performance between the single model and uniformly combined ensemble predictions. The purpose of this analysis is to evaluate whether we can improve SWMM generalisation using simple ensembles and to determine what role that the type of events, as determined using clustering, have on generalisation. Specifically, we consider uniformly weighted ensembles using only models calibrated to events belonging to a single cluster, using models calibrated on events from both clusters, and varying numbers of ensemble members. As a reminder, the notations EC x and ECB are used for ensemble predictions based on models calibrated to a specific cluster and an equal number of events from each cluster, respectively.

Fig. 2.8 illustrates the changes in performance of the EC1, EC2, and ECB relative to the mean single model performance. The mean single model performance is calculated as the mean cross-validation performance of each model on the event being evaluated, which is in-

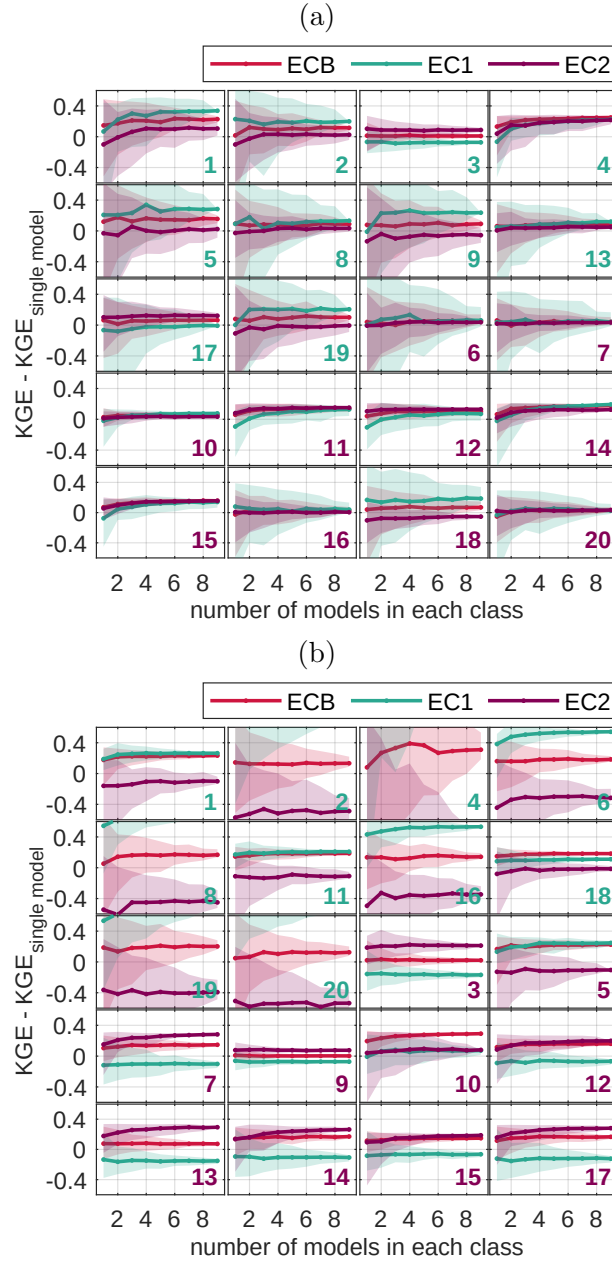


Figure 2.8: Difference in KGE between the ensemble and mean single model predictions across ensemble size for the SCS (a) and TCS (b) catchments. EC1, EC2, and ECB are shown in green, purple, and red, respectively. Note that the ECB contains twice as many predictors as the EC1, EC2 configurations.

indicated by the ID in the bottom right of each subplot, which is also colourised according to the event cluster. The calculation of mean single event performance of course does not include the model calibrated on the event being evaluated. Single cluster ensembles EC1 and EC2 tend perform best when evaluated on events in their respective cluster, with only a few exceptions, such as events 3, 18, 17 (SCS), and 5 (TCS). Ensembles consistently outperform the mean single model predictions, except for single-cluster events on their validation events in the

opposite cluster (e.g., EC1 evaluated on cluster 2 events), which usually results in a decrease in performance. The performance of the ECB ensembles is consistently between the EC1 and EC2 configurations. There are several cases where the ECB ensembles match the performance of the better single-cluster ensembles, such as for events 11, 12 (SCS), and 1, 5, 15 (TCD). Finally, there is a lone instance where the ECB ensembles outperforms the others, on event 10 (TCS). The mean KGE of all ensembles tends to remain constant or increase as with an increasing number of predictors. The KGE of some configurations increases by as much as 0.1 to 0.2 between ensembles with the fewest and most predictors, such as for event 1, 4 (SCS), and 7, 10, 17 (TCS).

The shaded area in Fig. 2.8 shows the spread in performance of different combinations of calibration events. To better visualise the effects that event combination has on model performance, the variance of performance is calculated for each ensemble size and illustrated in Fig. 2.9. While the variance in performance ranges by an order of magnitude between events, the variance of all 3 configurations consistently decreases as the number of models increases. The variance of the ensemble models approaches zero as the number of predictors approaches the number of models available in the set of sampled events. Comparably, if a model were trained on every event on record, there would be no need for event selection. These plots demonstrate how the variance in performance exponentially decays as information from more events linearly increases. For calibration, this result highlights the high degree of uncertainty when making predictions based on a single event, with exponential improvements in prediction consistency as the calibration event set grows.

The EC1 predictions typically have the highest variance for the SCS catchment, whereas the EC2 predictions have the highest variance for TCS. This result suggests that the variability in model performance owed to event selection is unique to individual catchments, despite both catchments in this study being similar in size, location, and land use. Most importantly, Fig. 2.9 illustrates that including information from several events for predictions is essential for achieving stable performance. For example, selecting a subset of two events for calibration produces highly variable performance, when the events are sampled from a set of events that have similar characteristics, as determined by the SOM. According to this figure, stable performance is only achieved with ensembles of models calibrated to 6 different events, or 12 (6 from each cluster) for the ECB configuration.

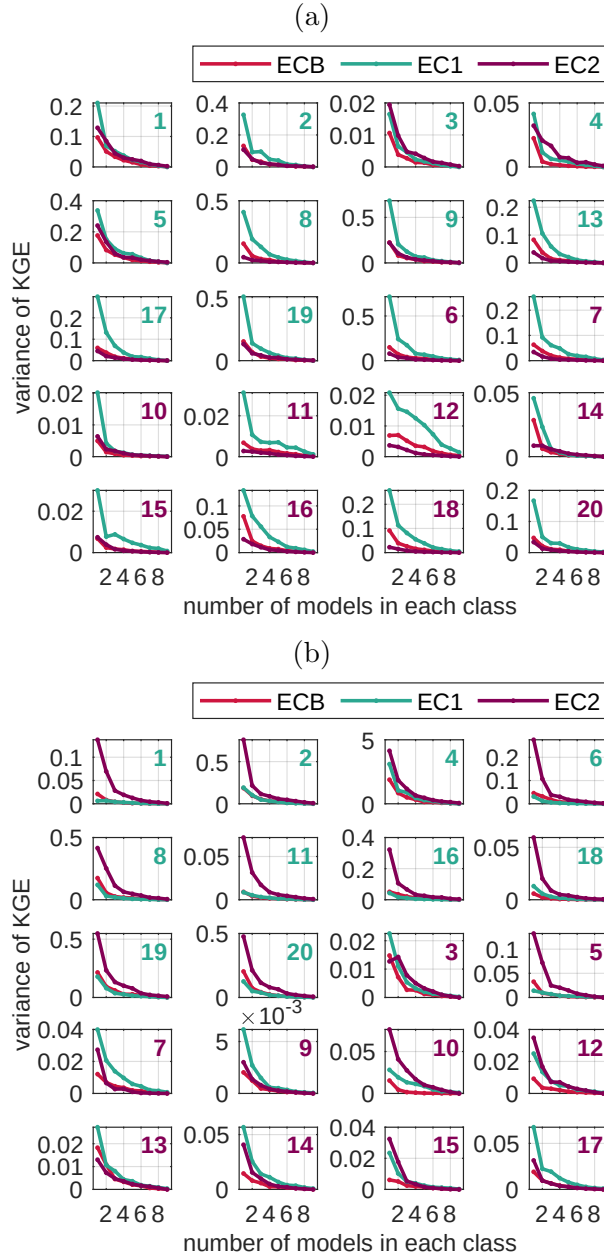


Figure 2.9: Variance of KGE across increasing ensemble size for uniform weighted combined predictions for the ECB (red), EC1 (green), and EC2 (purple) for the SCS (a) and TCS (b) catchments. Events are ordered by cluster and colour coded event IDs are shown in the top-right corners.

2.5.5 Evaluation of combination techniques

Next, we present a comparison of different methods of combining the predictions of ensemble members. The following analyses include only ECB configurations, in order to not bias performance towards any given event class. Results are illustrated in Fig. 2.10, which compares the performance for each combination technique based on three criteria: KGE, MVE, and PFE. For reference, the calibration and single model performances are also shown. The

mean performance rank and variance are also summarised in Tables 2.3 and 2.4. Given the variability of performance between different events, configurations are ranked for each event and the mean rank is reported. The ranks of PFE and MVE are calculated based on the absolute value of error. The variance is also reported; while the uniformly weighted ensemble has almost no variance, the other combination methods can introduce variability. The best combination technique have the lowest rank and lowest variance. The calibration performance, while included in the rank analysis, is only provided for reference and cannot be used in practice. Out of the four combination techniques and three performance criteria, ANN-based stacking produces the best overall results, achieving the strongest rank in 3 out of 6 cases (3 metrics for 2 catchments). In fact, ANN-based stacking exhibits comparable or better performance than the calibration model on events 17, 6, 7, 16, 18 (SCS), 11, 19 (TCS), demonstrating very strong generalisation. The non-uniformly weighted combination also does notably well, achieving the strongest PFE rank for both catchments.

Table 2.3: Mean performance rank and variance of calibration models, single model predictions, and all combination techniques for the SCS catchment. Bold font indicates the best column-wise scores, ignoring the CAL case.

	kge rank	kge var	pfe rank	pfe var	mve rank	mve var
CAL	1.95	0	2.65	0	2.4	0
SINGLE	5.55	1.75E-1	4.35	2.23E-1	4.95	7.62E-2
UNIFORM	4.3	3.80E-4	3.7	4.20E-4	4.75	1.54E-4
WEIGHTED	3.5	4.76E-4	2.9	4.61E-4	3.75	1.98E-4
STACK_MLR	2.7	8.31E-4	3.95	1.27E-3	2.65	1.23E-3
STACK_ANN	3	6.24E-3	3.45	1.07E-2	2.5	5.31E-3

Table 2.4: Mean performance rank and variance of calibration models, single model predictions, and all combination techniques for the TCS catchment. Bold font indicates the best column-wise scores, ignoring the CAL case.

	kge rank	kge var	pfe rank	pfe var	mve rank	mve var
CAL	1.25	0	2.65	0	1.35	0
SINGLE	5.55	1.45E-1	4.1	3.31E-1	4.35	9.73E-2
UNIFORM	3.75	2.42E-4	3.8	6.64E-4	3.7	1.47E-4
WEIGHTED	4.35	1.33E-3	3.6	1.92E-3	4.9	8.41E-4
STACK_MLR	3.25	9.71E-3	3.75	1.11E-2	3.25	5.74E-3
STACK_ANN	2.85	1.90E-2	3.1	2.31E-2	3.45	1.13E-2

2.6 Discussion

The clustering approach presented in this research provides a means of achieving diverse, time-continuous, and statistically similar calibration and validation datasets. We consider this to be a substantial improvement over traditional approaches to event selection, which over-

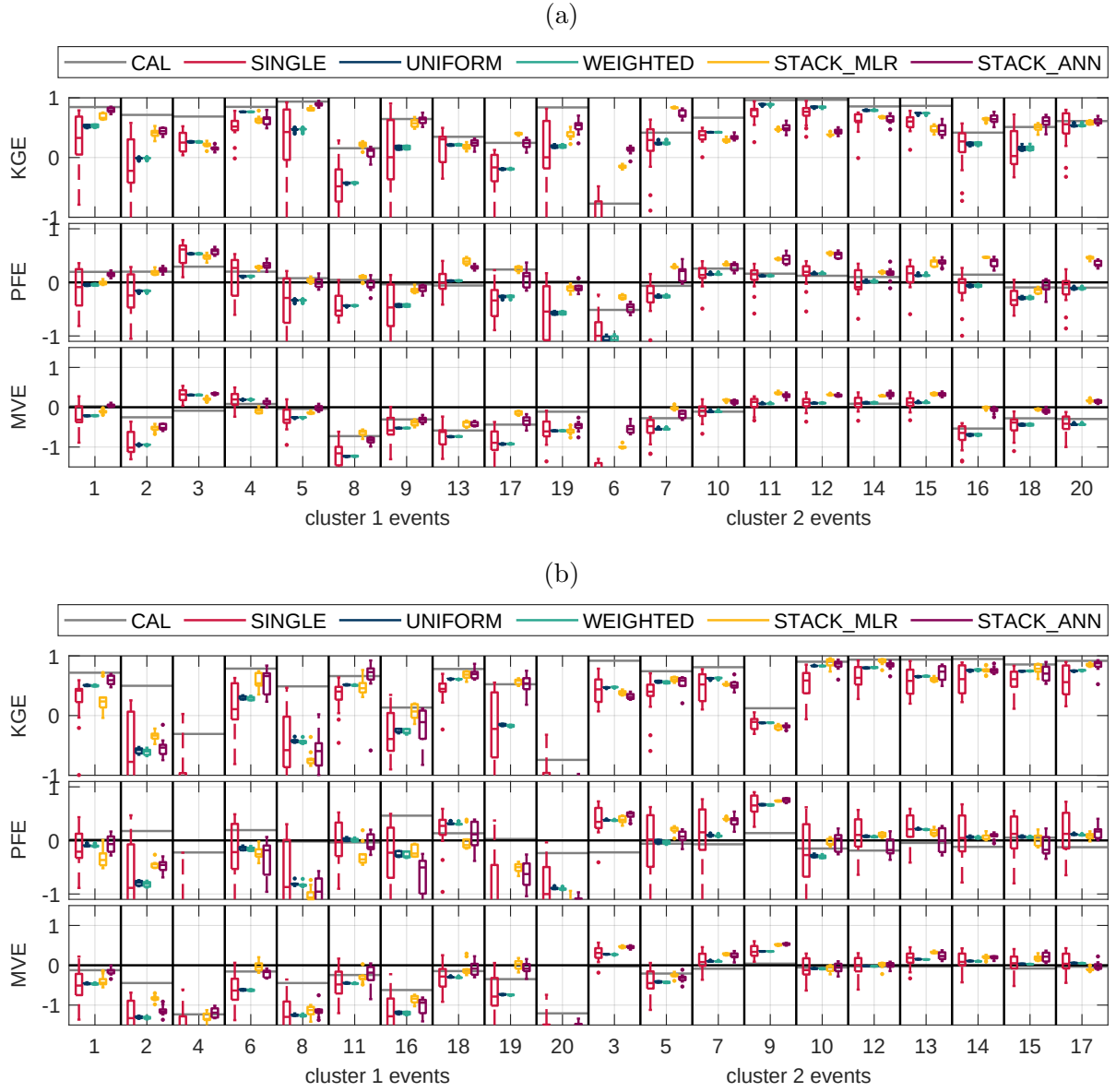


Figure 2.10: Validation performance of the SCS **(a)** and TCS **(b)** catchments for single model (red), uniform weighting (blue), variable weighting (green), MLR-based stacking (yellow), and ANN-based stacking (purple). Calibration performance is indicated by a grey bar.

whelmingly rely on subjective methods and/or single criterion. Our clustering method can be easily adapted to use different clustering algorithms or event characteristics. For example, expanding the clustering criteria to include features such as temperature may be useful in cold climates, which would encourage the distinction between cold-weather and warm-weather events, which typically involve different hydrological processes. To demonstrate the motivation for our proposed framework, we first illustrated the poor cross-validation performance of models calibrated to single events. While models tend to validate well on events with similar characteristics (in the same cluster), they perform poorly on events in the other cluster. This

also showed how calibration to different events produces different optimised parameter sets.

We introduced homogeneous ensembles as a technique to utilise the parameter sets obtained from calibrating a SWMM model to different events. Ensembles consisting of models calibrated to events in a given cluster perform well when validated on events in the same cluster, and poorly across clusters. Ensembles calibrated to a balanced number of events in each cluster perform somewhere in between, but typically much better than the mean single model. The single-cluster ensembles demonstrate the benefits and disadvantages of non-diverse calibration events. This finding can be applied to achieve balanced performance on different events or performance improvements on specific flow conditions. Alternatively, this ensemble framework can be expanded to a modular format, in which ensemble members are selected as predictors based on anticipated flow conditions. For example, if a large event is anticipated, predictions could be made using only the models calibrated to similar events. We also show how increasing the ensemble size generates minor improvements in performance as well as a reduction in performance variance. The performance of ensembles with three or fewer models tends to have high variance, which decreases as more and more events are included. We recommend maximising the number of events for calibration (reserving statistically similar events for validation) and caution against using less than 4 events, as model performance may be very sensitive to the events used. The improvements produced by our proposed method are consistent with previous studies that implicitly identify diverse events [Jean et al., 2018, Singh and Bárdossy, 2012]. However, the methods proposed in previous studies do not guarantee that selected events are hydrologically diverse and therefore are not guaranteed to improve model generalisation. In contrast, using clustering ensures a diverse set of calibration events.

Lastly, we evaluate four ensemble combination techniques in order to determine whether they can be used to generate further improvements in model performance, using both the single-model and calibration performance as baselines. The performance of combiners is consistent between both the SCS and TCS catchments. The ANN-based stacking method, which allows for the addition of exogenous data at the combination stage, is the best combiner on average across all catchments and performance criteria. The effects of exogenous input variables in model combiners are consistent with a recent power load forecast study by Bento et al. [2021], which found that models with exogenous inputs outperform those without, and ensembles with exogenous inputs outperform standalone counterparts.

The weighted combination achieves the highest mean PFE rank, while the stacked models tend to over-predict peak flows. The stacked ensembles also tend to exhibit lower, or positive volume error compared to the other combination techniques. We attribute this to the fact that they are not constrained to mass conservation laws, unlike the convex combined models. The strong performance of the non-uniformly weighted ensembles is consistent with previous works. Tyralis et al. [2021] found that a convex weighted combination of machine learning models outperformed single model and uniformly weighted model performance. Similarly, Kumar et al. [2015] found that a similar method, constrained multiple linear regression, performed the best out of six different combination strategies.

Based on these results, unconstrained combination techniques such as stacking are well-suited to overcome certain physics-based model behaviour such as total volume underestimation, however, our results showed that this resulted in slight overestimation of peak flow magnitudes. Of course, part of the success of the stacking-based methods is due to its use of exogenous sensor data for prediction, which is not always available in real-time. The stacked combiner can be configured with or without exogenous data, however, we expect to see a slight decrease in performance without this data. The uniformly weighted and non-uniformly weighted combinations exhibit almost no variance associated with event selection, whereas the stacking methods introduce some variance in performance, it remains much lower than the variance in performance between models calibrated to different single events. The stacking variability is owed to the calibration of the combiner (i.e., MLR or ANN).

These results highlight how a simple, uniform combiner can produce considerable improvements over the single model scenario, while eliminating the uncertainty associated with single event selection. Stacked combinations create further improvements, which we attribute to their ability to incorporate real-time data and overcome volume balance errors.

The main limitation of this work presented above is that the methods were only applied to two low complexity SWMM models. More complex SWMM models may produce even greater variability in performance between different events, due to having higher degrees of freedom. Therefore, future studies should consider SWMM models with varying degrees of complexity. Similarly, a large-scale study should include catchments that represent a broad range of hydrological conditions, which would be used to further test the proposed methods. Another

limitation of this work is the use of two event clusters. This study used two clusters to ensure that enough events are available for the validation scheme. Increasing the number of clusters will improve the diversity of the calibration event set further and is expected to produce further improvements to model generalisation. The factor limiting the number of clusters is the amount of available historic data; more data increases the likelihood of multiple occurrences of infrequent events, resulting in larger cluster sizes. A final limitation of this work is that snow months were removed from the model simulations and analysis.

2.7 Conclusion

The primary objective of this paper was to develop an improved method for calibrating urban rainfall-runoff models in such a way that preserves time-continuity, identifies statistically similar calibration and validation datasets, and achieves strong generalisation. We accomplish these objectives using a novel ensemble framework that relies on unsupervised clustering to identify calibration and validation events. Within our ensemble, models are distinguished by the events to which they are calibrated, creating diversity, and improving generalisation. The main findings of our paper are summarised as follows:

- First, we conduct a sensitivity analysis, which indicates that there is little difference in parameter sensitivities between event clusters. We used the 5 most sensitive parameters to demonstrate how filtering insensitive parameters has no effect on GA-based model calibration.
- A systematic cross-validation analysis revealed that single models generalise poorly, especially on dissimilar events, as indicated by the clustering results.
- Ensembles were created using events from a single cluster, which were found to cross-validate well on similar events, but poorly across clusters. Meanwhile, ensembles with an even number of events in each cluster are a compromise in performance between the two single cluster ensembles, while still outperforming the mean performance of the non ensemble models.
- Increasing the number of models in each ensemble was shown to improve performance and reduce variance associated with the random selection of ensemble members.
- Several different ensemble combination techniques were compared, including uniform,

weighted, MLR-based stacking, and ANN-based stacking. Out of four methods, ANN-based stacking was found to produce the strongest performance across KGE, and MVE, while weighted combination produced the best PFE.

- All SWMM ensemble techniques outperform the mean single-model performance in every metric and catchment except one, demonstrating the value of the proposed methods.

3 Advancing Flood Early Warning Systems: Ensemble Learning based Classifiers for Urban Flood Forecasting

3.1 Preface

The following chapter is the product of a collaboration with Dr. Alobaidi from McGill university. Dr. Alobaidi provided foundational code featuring a cross-comparison of several ensemble algorithms and classifier models. The code, which was originally developed for a geotechnical application, was adapted for hydrometric data and developed as to include a suite of pre- and post-processing procedures. The motivation for applying a classification framework to flood forecasting lies in the simplification of the learning problem; by reducing the complexity of hydrological states represented in the target data, flood forecasting might become more accurate. The content of this chapter was submitted to the Journal of Hydrology as follows:

Snieder, E., Alobaidi, M., Khan, U. T. (2024). Advancing flood early warning systems: ensemble learning based classifiers for urban flood forecasting. Submitted to the Journal of Hydroinformatics.

Contributions to this chapter are as follows: **E. Snieder:** data curation; formal analysis; investigation; visualisation; code - pre- and post-processing, analysis; writing - original draft.

M. Alobaidi: coding - ensemble models; writing - editing, revisions **U. T. Khan:** conceptualisation; funding acquisition; supervision; writing - editing, revisions.

3.2 Abstract

Flood early warning systems (FEWS), which rely primarily on flood forecasting models, are becoming increasingly important for mitigating damage to natural and built infrastructure. Machine learning (ML) provides flexible modelling solutions to modelling challenges such as FEWS. In this work, a flood forecasting approach is developed as a pure classification problem to directly predict flood events. The advantages of the proposed approach are demonstrated by appropriately discretising the streamflow or stage into binary states, instead of treating them as continuous variables. This distinction is highlighted through a juxtaposition of regression and classification approaches, each used to generate flood alerts. The research also features a systematic cross-comparison of five ML models and three ensemble learning frameworks to provide additional insights on modelling applicability. Another contribution of this research is the exploration of the balance between true and false positives, examined in light of diverse resampling strategies and classification thresholds. Classification performance is shown to be primarily dependent on the base learner; Extreme Learning Machines and Support Vector Machines show the best performance, whereas Logistic Regression and Decision Trees perform relatively poorly, irrespective of their ensemble model pairing. Between the ensemble methods, boosted models consistently outperform their counterparts.

3.3 Introduction

A flood early warning system (FEWS) provides communities with advanced notification of a suspected flood hazard and is considered to be among the best ways to mitigate flood damage [Yilmaz et al., 2010, Muñoz et al., 2021]. Such systems are especially important in urban areas, where hydrological processes are concentrated in both space and time, and built infrastructure interferes with natural processes [Niemczynowicz, 1999]. A FEWS typically relies on streamflow models to determine whether to issue a flood warning. FEWS can be potentially developed using data from instruments such as rain gauges, water level sensors, and flow rate sensors among other weather sensors. Real-time monitoring is required in FEWS, which relies on continuously receiving required data, where it is processed and used in flood forecasting [Pengel et al., 2013, Nguyen et al., 2021, López et al., 2020]. The literature is rich with research developing process-based modelling for predicting streamflow, which attempts to simulate the general evolution of streamflow and its properties in time, with respect to well-established input features [Tokar and Markus, 2000, Perrin et al., 2003, Gallice et al., 2015].

This modelling approach can be used to capture river flood conditions and dynamics with satisfactory results. It is also important to make a distinction between the intended FEWS framework and that involved in the flood inundation system (FIS). In this work, the intended problem of interest, FEWS, is related to providing advanced notice of potential flooding [Alobaidi et al., 2020]. It aims to predict floods and provide alerts before floodwaters arrive. On the other hand, the main goal of FIS is to map the extent and depth of flooding [Chang et al., 2018]. FEWS and FIS can sometimes be identified as the same due to their interconnected roles in flood management [Agbehadji et al., 2023]. Both systems involve flood monitoring and data analysis, although at varying level and using different intelligence, leading to an overlap in their specific functions and purposes as described in [Mentzafou et al., 2023]. Consequently, the specific term FEWS is used in this work to identify prediction in time, which may be expanded to a spatiotemporal setting. A problem which essentially stems from prediction in space, even if it is expanded to a spatiotemporal setting, is referred to as FIS.

The formulation of the streamflow forecasting problem can be carried out using physics-informed approaches [Biswas et al., 2020]. Although such deterministic models can adequately capture the nuanced dynamics of river flooding under different hydro-meteorological conditions, they are inherently constrained by the need for precise estimates of the features governing streamflow [Ahmed et al., 2021]. In contrast, data-driven models (DDMs) can assimilate diverse cases of feature space, which provides a more flexible solution to streamflow forecasting. Using DDM frameworks, accurate short- and long-term streamflow prediction techniques have been demonstrated in the early literature [Hsu et al., 1995, Tiwari and Chatterjee, 2010b]. Over the past few decades, this flexibility has led to a growing adoption of DDMs, specifically machine learning (ML) models, for important applications in hydrology. ML models emphasise prediction accuracy, although often at the cost of comprehensive interpretability of the model, which has become a more favoured trade-off in hazard problems such as flood prediction [Tyralis et al., 2021]. Recently, large-scale studies have showcased the applicability of ML models for streamflow prediction, achieving improved performance over classical physics-based and mainstream statistical techniques [Mosavi et al., 2018, Kratzert et al., 2019b, Xiang and Demir, 2020, Nearing et al., 2021]. However, despite their suitability for streamflow prediction, utilised ML models are not always designed to maximise their efficacy in FEWS. The majority of existing research in the development of FEWS predominantly adheres to a con-

ventional paradigm where predicting streamflow is the main objective and peak streamflows are an incidental product [Alobaidi et al., 2020, Snieder et al., 2021]. In most cases, a regression model is used to predict streamflow as a continuous variable. One problem with this approach is that models are rigorously optimised to minimise objective functions, such as the Mean Squared Error (MSE), which intrinsically lack a flood-centric focus [Maier et al., 2010, Snieder et al., 2021]. Model validation and selection are also aligned with these metrics, in that they do not focus on flood events. Therefore, the resulting models are not always well-suited for accurate FEWS.

The popularity of cost functions such as MSE for optimisation is partly owed to the fact that they are often the default setting in implementations of regression models. While some implementations allow the use of custom cost functions, constraints such as backpropagation requiring a differentiable cost function limit the degree of customisation. An obvious solution to this problem is to use a weighted cost function to allocate greater importance to flood events during training; however, a weighted cost function can result in sporadic model behaviour on low streamflows, which reduces the reliability of the predicted streamflow and subsequent flood warnings [Snieder, 2019]. Another issue with the standard approach is that the rates of false and missed alarms are not given appropriate attention, which may be highly variable between different modelling approaches. In other words, they are not usually used to provide a direct measure of FEWS efficacy. A potential solution to these problems is to reframe streamflow forecasting as a pure classification problem. That is, using hydrometeorological data to forecast a categorical flood label directly.

Formulating streamflow forecasting as a classification problem presents several advantages over conventional regression methods. First, classification models explicitly optimize the accuracy of flood warnings, enabling a more direct trade-off between false positives and false negatives than conventional modelling approaches. In contrast, traditional regression models aim to capture the continuum of the streamflow, or river states, which shifts their training objective away from critical flood events. Second, the classification paradigm streamlines the learning problem by categorizing river states into meaningful discrete classes, such as flood and non-flood. This representation potentially reduces the model’s dependency on extraneous features that are not germane to flood prediction. Lastly, there are scenarios where classification models are intrinsically necessary. For instance, when flood data is ordinal in nature—such as

information crowdsourced through social media [Jongman et al., 2015]—or when flood conditions are ascertained through image-processing techniques [Munawar et al., 2021]. Traditional regression models cannot be used in this case.

Despite the advantages outlined above, relatively few studies have examined classification models for timeseries based (or FEWS) prediction in hydrology, unlike their prevalence in FIS frameworks. Miao et al. [2016] employs a binary classifier to the long-term output of a physics-based rainfall-runoff model. The classifier acts as a surrogate model that is used to issue flood warnings in ungauged basins, since the physics-based model is too computationally expensive to run in real-time. A recent study by Muñoz et al. [2021] presents a pure data-driven approach, in which stage data is labelled as “no alert”, “pre-alert”, or “alert”, in order to train classifiers for FEWS using temperature, streamflow, and precipitation timeseries’ as input features. The study compares the performance of models across several lead times for a watershed in Ecuador, finding classification approaches can outperform other models designed to extract flood event warnings. Similarly, the work in Rasheed et al. [2022] utilises long short-term memory (LSTM)-based classifiers to issue early flood warnings in ungauged basins. While these studies establish the feasibility of using classification models in FEWS, they also highlight a multitude of unanswered questions concerning the development and performance evaluation of such models. At this point, it is important to reiterate the distinction between FEWS and FIS, which further established the contribution of the present work. In studies involving the development of FEWS, river flood events (as appropriately defined in the present work) are quite rare with respect to the available streamflow record [Alobaidi et al., 2020]. Absence of abundant flood events introduces what is called “class-imbalance” on a severe level, which hinders establishing a diverse dataset that can be used to develop a classifier with reasonable generalization ability [Hakim et al., 2023, Hadi et al., 2024]. This has implicitly contributed to conventionally formulating streamflow-based flood prediction problems as regression based in FEWS. Moreover, unlike FIS, the feature space available for FEWS is considerably more limited. This limitation constrains the complexity of machine learning model architectures that can be effectively employed. In such scenarios, the application of deep learning approaches would be ill-advised. This is where ensemble learning approaches, a topic actively explored in the general applied machine learning literature, become particularly relevant. Ensemble learning, which is an ML paradigm that employs multiple models

(commonly known as base learners), has gained considerable attention in hydrology due to its demonstrated capability to outperform standalone models [Alobaidi et al., 2020, Zounemat-Kermani et al., 2021]. For instance, ensemble approaches have been successfully applied to rainfall-runoff modelling and water quality prediction, showcasing their robustness and predictive accuracy. More on ensemble learning is presented further in this work. Despite this success, there remains a notable gap in the literature: ensemble methods have yet to be systematically evaluated in the context of streamflow classification tasks. To this extent, the main contributions and objectives of this work are (i) to develop the required formulation stages and important definitions which should be addressed in order to successfully frame flood prediction as a classification problem in the context of FEWS, (ii) to conduct a systematic cross-comparison of ML model types and ensemble algorithms, and (iii) to use a classification metrics and tools to evaluate models based on the trade-off between missed and false alarms.

The first objective determines whether classifiers are viable candidates for improved flood prediction performance compared to the typical regression approach. This is carried out by directly comparing the performance of regression and a proposed classification framework involving bagged artificial neural networks (ANNs) following a similar feasibility analysis approach in the broad literature [Zounemat-Kermani et al., 2021, Maier et al., 2010]. The second objective is accomplished by evaluating three ensemble learning frameworks and five base learners. As there are hundreds of variations of ensemble learning models and base learners available, the scope of this work utilizes a curation of architectures that span a broad range of complexities and training procedures for effective evaluation. A cross-comparison of each method produced a total of fifteen model configurations. The final objective is attained by evaluating models based on specialised classification metrics for imbalanced datasets, as well as visual measures. Specialised metrics were used to develop a preprocessing scheme and evaluate ensemble combinations. The preprocessing scheme undersampled non-flood, and synthetically oversampled flood streamflow samples. The proposed approach is then demonstrated over three urban watersheds in Toronto, Canada: Etobicoke Creek, Humber River, and Don River.

3.4 Methodology

The following subsections outline the three watersheds under study, ensemble methods, and experimental configurations. Some background is provided on preprocessing methods, five types of ML models, and three ensemble methods used in this study. The aim of this section is not to provide a comprehensive review of each ML method, which are all well-documented in the referenced works.

3.4.1 Study region and data

This study was conducted using hydrometeorological data collected from three watersheds overlapping the city of Toronto, Canada: Etobicoke Creek, Humber River, and Don River. These watersheds were selected because they are important socioeconomic centres, densely populated, and have all experienced flooding in recent years.

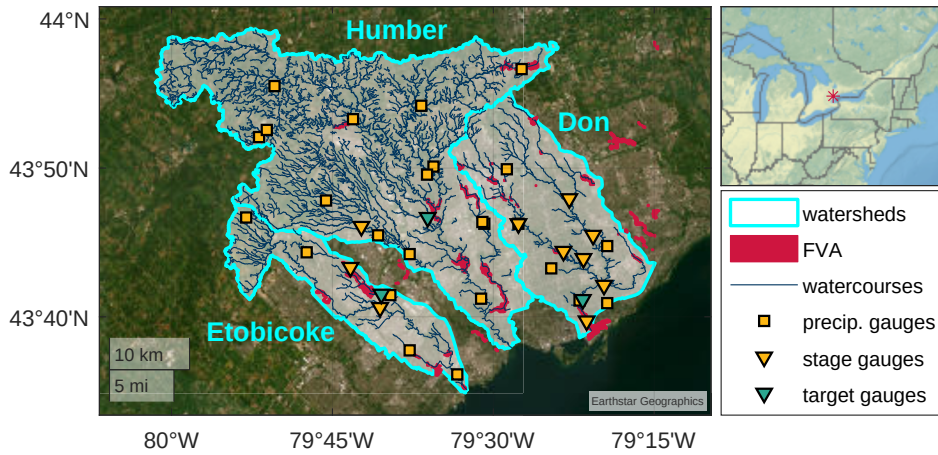


Figure 3.1: Don, Etobicoke, and Humber watersheds (cyan with transparent white fill), watercourses (blue), precipitation gauges (yellow squares), stream gauges (yellow triangles), target gauge (green triangle), and flood vulnerable areas (red). Hydrometric features outside the study watersheds are displayed in muted colours for spatial context. Earthstar Geographics satellite imagery used as basemap.

The watershed boundaries, watercourses, and gauge locations are illustrated in Fig. 3.1. There are many similarities between the three watersheds, whose properties are summarised in Table 3.1, including their geographical location and downstream urbanised land use. Urbanised watersheds were used due to their characteristically high flood vulnerability. Having used three similar watersheds allowed for the cross-validation of results obtained for each individual wa-

tershed. Out of the three watersheds, the Humber River watershed is distinct in that it has large rural headwaters, resulting in a much larger time-to-concentration compared to the other two watersheds. Hydrometeorological input timeseries’ for streamflow, stage, and precipitation within each watershed were obtained from the local watershed authority, the Toronto and Region Conservation Authority (TRCA) [TRCA, 2020]. The stage at a single gauge in each watershed was selected as the target for prediction. The three target gauges were selected based on their proximity to flood vulnerable areas (FVAs), which were identified by the TRCA [TRCA, 2020], and illustrated in Fig. 3.1. In some cases, the target gauges were near, but not completely within the FVAs. The remaining hydrometeorological gauges within each watershed form the respective candidate input sets.

Table 3.1: Summary of watershed attributes.

name	area [km ²]	mean slope [deg]	imperviousness [frac]
Don	358.0	1.92	0.76
Etobicoke	212.0	1.00	0.60
Humber	911.0	2.09	0.28

3.4.2 Data preprocessing

3.4.2.1 Candidate input feature set

Timeseries data were obtained for each gauge for the period of 2013 to 2020 at a 15-minute resolution. Data from November to April and November to December were removed from the Bow and Don River datasets, prior to any analysis; these periods are associated with ice conditions. The years 2013 and 2014 were unavailable for Etobicoke creek, but otherwise the timeseries contained relatively few missing samples. A previous study of the Don River found that accurate predictions were challenging for previously developed ML approaches beyond a 6-hour lead time, because the lead time exceeded the lag time between upstream and downstream stations [Snieder et al., 2020]. Therefore, a lead time of 3-hours was selected for all three watersheds in this work. The input and target data was downsampled from a 15-minute resolution to match the lead time of the forecast. Continuous variables were downsampled by calculating the mean stage, streamflow, and total precipitation. Flood labels were downsampled such that any 3-hour window containing any positive labels were labelled as floods. It was important to label flood data prior to downsampling, as calculating the 3-hour mean value can smooth out peak flows, resulting in the loss of flood events with very short durations.

The candidate input set was augmented such that each of the hydrometeorological timeseries was lagged by 1 timestep to imitate the forecast, and recurrent lagged copies were included up to a maximum lag of 36 hours. Including lagged copies of candidate input features is common practice and provides the model with recent hydrological states while also capturing the lag time that exists between the target gauge and other gauges within each watershed [Nanda et al., 2016]. A maximum lag amount of 36-hours is excessive and results in a high-dimensional input set; therefore, feature selection was used to extract the most useful features.

3.4.2.2 Labelling

Assigning categorical labels to stage observations is a critical part of training a forecasting model for FEWS. The first step is to identify a stage threshold associated with flooding. Flood stage thresholds can be obtained from a terrain-derived inundation extent or using statistics, such as a specific percentile [Fleming et al., 2015, Nanda et al., 2016]. A number of gauges within the watersheds under study area exhibited high-stage warning thresholds, which were determined by the local conservation authority [TRCA, 2020]. An analysis of the available stage data found that the published high-stage thresholds corresponded approximately to the 99.9th percentile. In the present work, a slightly more conservative percentile of 99.7th was selected for classifying high stage samples at all stream gauges, which ensured a sufficient quantity and temporal distribution of flood labels. Stage hydrographs are illustrated in Fig. 3.2 for each of the three watersheds, with values above the 99.7th percentile highlighted in red. Typical of urban and semi-urban watersheds, all three hydrographs exhibited ‘flashy’ behaviour, characterised by sharp rising and receding limbs. Such volatile changes over short periods makes flood forecasting particularly challenging [Snieder et al., 2020]. The ‘flashiness’ can be detected based on the difference between the 15-minute and daily stage-discharge curves in Fig. 3.3, which suggests flood events in these watersheds typically last less than a day [Searcy, 1959].

Flood samples were identified by using the high stage threshold value to discriminate between floods and non-floods [Razali et al., 2020, Snieder et al., 2021, Nevo et al., 2022]. This exercise was improved using a few simple preprocessing steps, illustrated in Fig. 3.4, which increased the similarity between categorical flood labels and actual flood events. First, a binary time-series was created by comparing stage values against the high-stage threshold. We then defined a minimum flood duration (MFD), which was used to fill gaps in flood events and elim-

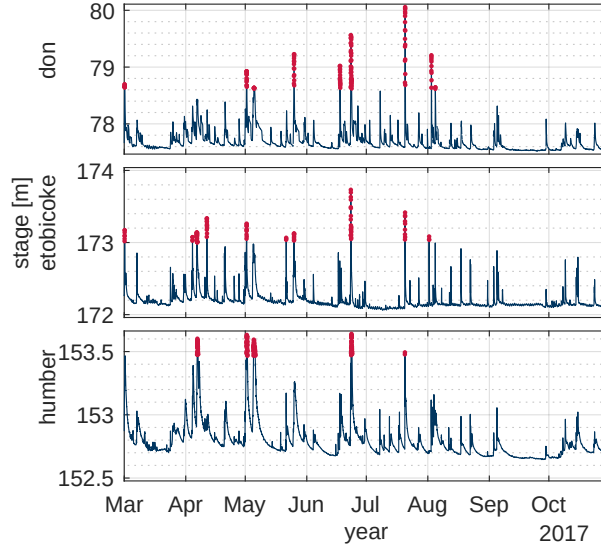


Figure 3.2: Hydrographs with stage data for 2017 at the target gauges for Don (top), Etobicoke (middle), and Humber (bottom). Typical stage values are blue and flood events are highlighted in red.

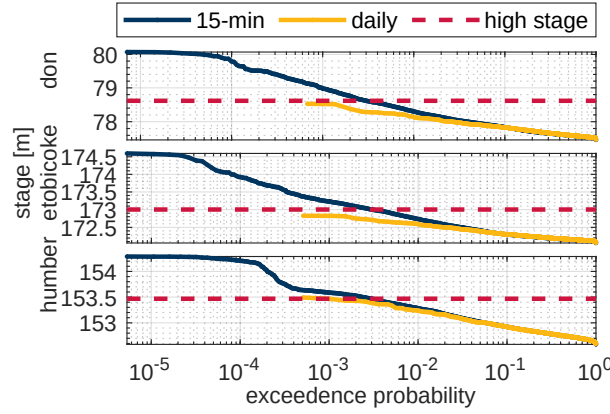


Figure 3.3: Stage duration curves for target gauges for Don (top), Etobicoke (middle), and Humber (bottom). The dashed red line corresponds to the 99.7th percentile stage.

inate brief spikes in stage. The MFD was used to smooth flood events. That is, all points between two flood samples that were separated by no more than the MFD were also labelled as floods. This is important as there may be gaps (samples where the stage is below the threshold) during a flood event that should be labelled as floods. Next, flooding typically occurs after persistent high stage, lasting several hours. Flood events shorter than the MFD, identified by $[0,1]$ sequences in the binary timeseries, were labelled as non-floods, ensuring that brief spikes in river stage were not confused with flood events. This was particularly important when using stage data with a relatively high temporal resolution, such as the 15-minute data used in this study. A disadvantage of this labelling approach was that it can reduce the total number of flood labels available for training. To illustrate this effect, the number of flood

samples are plotted against increasing MFD values in Fig. 3.5. The Etobicoke and Humber curves indicates that there were many short events based on the convex shape of the curves [Hsieh, 2021], whereas the concave shape of the Don curve indicate longer flood events. An MFD of 4 was selected for this research, equivalent to one hour, in part to ensure sufficient availability of flood samples with which to train and validate the models.

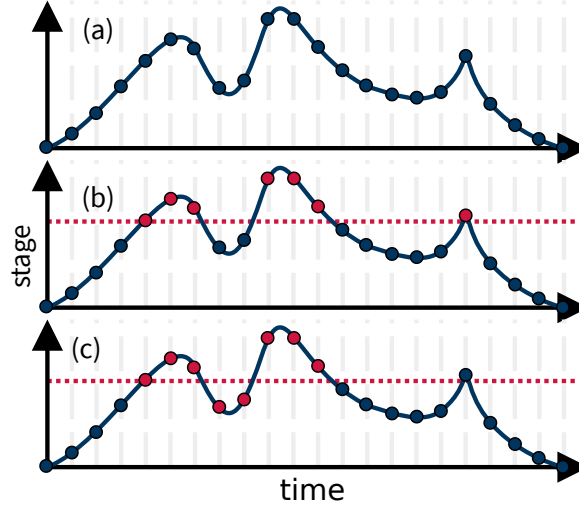


Figure 3.4: Example of flood labelling for a persistence window 3. Raw timeseries (a), flood labels based on a static threshold (b), smoothed and trimmed flood labels (c) with 2-timestep gap filled, and 1-timestep flood removed. The static flood threshold and flood samples are indicated by the dashed red line and red points, respectively.

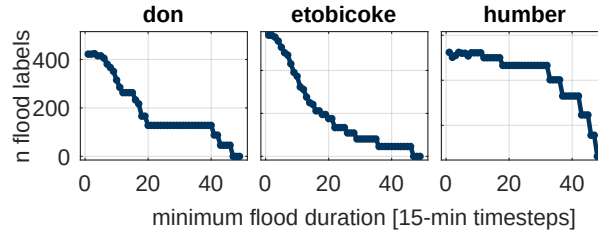


Figure 3.5: Number of flood samples with an increasing MFD for Etobicoke (left), Humber (middle), and Don (right).

3.4.2.3 Data partitioning

The input and target datasets were divided into continuous training and testing data partitions with an equal number of flood instances. The partitions are time-continuous blocks to ensure preservation of temporal structures within the data, such as autocorrelation. The length of the blocks was determined such that each set contains 50% of the flood labels. The training dataset was used for resampling and feature selection, which are described in previous and following subsections. Following best practice, every feature set was normalised between 0

and 1, with normalisation parameters having been obtained from the training samples.

3.4.2.4 Feature selection

Feature selection (also called input variable selection) is a crucial preprocessing step for ML-based models [May et al., 2011]. Reducing the size of the feature set has several benefits, including improvements in model calibration through simplification of the learning task, reducing computational demand, and reducing data requirements [May et al., 2011]. Since this study compares several model types, a model-free feature selection method was used [Snieder et al., 2020], as an embedded feature selection method risks creating bias towards the model type used to calculate feature importance. Minimum-Redundancy Maximum-Redundancy (mRMR) is an efficient model-free feature selection algorithm that selects inputs based on mutual information with the target variable [Hanchuan Peng et al., 2005], and has been routinely used for applications in hydrology [Zahmatkesh et al., 2015, Karamouz et al., 2021, Ren et al., 2020].

The inner mechanism of mRMR hinges on two stages. In the first stage, mRMR ranks candidate features by their individual relevance to the target variable, quantified by mutual information index. Features with higher mutual information are given priority. In the second stage, mRMR introduces redundancy considerations by iteratively adding features that contribute the most unique information while maintaining a minimal overlap with previously selected features. This ensures that the selected feature set is not only relevant but also diverse, preventing the inclusion of redundant features. One of the distinct advantages of the mRMR method is its ability to handle high-dimensional datasets effectively. By selecting a subset of the most informative features, it reduces the dimensionality of the data, which can lead to improved model performance and reduced computational complexity. Furthermore, mRMR's model-free nature allows it to be applied across various machine learning algorithms and domains, making it a versatile tool for feature selection. Consequently, for each unique candidate input set in this study, inputs with the highest 20 relevancy scores were selected. The feature selections of the mRMR algorithm were found to be consistent with the findings of a previous study that compared four different feature selection algorithms in the Don watershed [Snieder et al., 2020].

3.4.2.5 Resampling

Flood data are usually highly imbalanced (called skewness when referring to continuous stage data), which introduces added difficulty of achieving accurate flood forecasts. Failure to properly account for data imbalance can result in biasing the model performance towards the majority class (i.e., non-flood samples). Labelling high-stage based on the 99.7th percentile produced a ratio of non-flood to flood samples of 332:1. Additional labelling criteria, such as the high-stage persistence method described above, increases this imbalance even further. There are several approaches described in the literature for handling heavily imbalanced data, broadly consisting of resampling the training data or modifying the training algorithm [Galar et al., 2012]. Undersampling non-flood samples and oversampling flood samples can both be used to reduce the imbalance of the training dataset [Snieder et al., 2021]. Modifying the learning algorithm includes methods such as weighted objective functions, which increases the relative weight of flood samples in the cost function used to train the models [Muñoz et al., 2021]. This work used resampling, which has the benefits of reducing computational demand through undersampling, and increasing the diversity of the training dataset using synthetic oversampling [Snieder et al., 2021].

The Synthetic Minority Oversampling TEchnique (SMOTE) is the first of two resampling techniques used to counter the effects of class imbalance [Chawla et al., 2002]. SMOTE increases the size of the minority class by generating synthetic samples, which are created by linearly interpolated between a point and a neighbouring point, as determined using K-Nearest Neighbours (KNN). The point and interpolation distance are both determined randomly. SMOTE has been widely used for imbalanced learning problems within ML literature [Cateni et al., 2014, Haixiang et al., 2017], and its use is rapidly growing in hydrology [Snieder et al., 2021, Sufi Karimi et al., 2019, Razali et al., 2020, Atieh et al., 2017, Wu et al., 2020, Tang et al., 2022] for regression and classification problems. In order to avoid low diversity among the synthetically oversampled minority class (i.g., flood samples), and to further tune the model bias towards the minority class, SMOTE was combined with random undersampling (RUS) of the majority class, as proposed by Chawla et al. [2002]. RUS serves to further bias model performance away from the majority class, and when combined with SMOTE it represents an improvement over either standalone resampling technique [Chawla et al., 2002].

The evaluation of SMOTE-RUS showed that a ratio of flood to non-flood samples of 1:1 does not produce the best performance. Our initial analysis found that there was an upper limit to

the amount of generalisable information that can be obtained through synthetic oversampling, and too much oversampling will result in overfitting. Similarly, too much undersampling eliminates diversity in the majority class, resulting in an incomplete representation of non-flood samples. Therefore, we conducted an exhaustive two-dimensional grid-search to determine optimum SMOTE and RUS resampling rates and better understand the trade-off between flood and non-flood prediction accuracy. The notation for resampling rate used in this study is relative to the size of each class in the unaltered dataset. For example: using SMOTE and RUS resampling rates of 12 and 20, respectively, applied to a dataset containing 100 flood and 10,000 no-flood observations will produce a resampled dataset with 1,200 flood 500 non-flood samples.

The grid-search of resampling parameters included SMOTE rates of 0 to 10 in increments of 1, and RUS rates of 10^a for values of 0 to 2 in increments of 0.5. In total, the SMOTE-RUS parameter search contained 55 unique training datasets, which were evaluated on each watershed. Since conducting this search on every ensemble configuration featured in this study would be computationally expensive, a single ensemble configuration was used to evaluate the resampling training data. ANNs with bagging ensembles were chosen for the resampling experiment, as they are among the most common model type and ensemble for water resources applications [Zounemat-Kermani et al., 2021, Mosavi et al., 2018]. Their popularity has been attributed to strong and reliable performance, which fosters confidence in the results of the grid-search. A description of both methods, along with other model and ensemble types featured in this research, are included in the following paragraphs.

3.4.2.6 Classification and regression comparison

To assess the viability of a classification approach to flood forecasting, compared binary classification models with traditional regression models. The model parameters were mirrored when possible (e.g., ANN architecture, ensemble size, input data), except for non-transferrable parameters listed in Table 3.2. In both cases, the ANN contained a single layer of five hidden neurons and a hyperbolic tangent sigmoid activation function. The training data for both model types was resampled using the optimum SMOTE-RUS rates determined using the grid-search described above. The continuous output of the regressor was converted to binary values based on the high stage threshold used to label the binary model targets, allowing for both models to be evaluated using classification metrics.

Table 3.2: Comparison between classification and regression ANNs.

Model	Classifier	Regressor
Target data	Binary flood label	Continuous stage data
Output activation function	Logistic	Linear
Optimisation function	Cross-entropy	Levenburg-Marquardt

3.4.3 Ensemble learning model comparison

In order to aid in model selection and better understand the capabilities of classification models for flood forecasting, a comparison was conducted for five different ML models and three ensemble algorithms, which are outlined in the following two subsections.

3.4.3.1 ML models

This section provides a brief description of the five base learners included in this study: Logistic regression (LR), Support Vector Machine (SVMs), Decision Trees (DTs), ANNs, and Extreme Learning Machines (ELMs). was the simplest model considered, consisting of a linear regression model passed through a logistic function. Recent examples of logistic regression in water resources are found in [Szelag et al., 2020, De Coste et al., 2021]. Support Vector Machines (SVMs) learn a hyperplane to separate classes. A kernel is used to transform input features to a higher dimensional space, enabling nonlinear classification. SVMs have been used extensively in hydrology and are thoroughly reviewed in [Raghavendra. N and Deka, 2014]. Recently, several studies have combined SVMs with decomposition techniques for improving flood forecasting [Kambalimath S and Deka, 2021, Chen et al., 2021, Shada et al., 2022]. ELMs are a variant of single-layer feed-forward ANNs, distinguished by their training procedure [Huang et al., 2006]. ELM parameters are trained in a single computation using the Moore–Penrose generalised inverse operation, in contrast to backpropagation which is typical for ANNs. ELMs exhibit competitive performance when compared against other ML models, but can be trained in a fraction of the time. Several recent streamflow forecasting studies have noted strong performance of ELMs when compared against other state-of-the-art ML models [Ren et al., 2020, Zhou et al., 2018]. DTs are models with a nodal tree-like structure. Because of their simple structure, DTs are typically used within ensemble frameworks, which significantly improves their performance. The Random Forest (RF) ensemble algorithm has been widely applied to water resources problems Tyralis et al. [2019b], while other ensemble algorithms have received comparatively little attention. ANNs have been extensively ap-

plied to streamflow forecasting, and review papers can be found in [Yaseen et al., 2015, Maier et al., 2010]. ANNs are overwhelmingly the most popular ML model architecture for modelling river systems [Maier et al., 2010]. While recent research has largely focused on advancing specialised ANN variations such as graph neural networks [Liu et al., 2022] and long short-term memory storage ANNs [Kratzert et al., 2019a, Ouma et al., 2022], our study used the basic multilayer perceptron (MLP) feed-forward network. Similarly to a recent large-sample study [Tyralis et al., 2021], the hyperparameters for each model were set to the recommended values within MATLAB 2021a.

3.4.3.2 Ensemble learning methods

The generalisability of the models described above can be improved using an ensemble learning framework. Ensemble learning refers to the use of multiple models for prediction, and can typically be distinguished by how they train each ensemble member and how member predictions are combined. Ensemble learning has consistently been found to improve performance when compared to that of individual models [Zounemat-Kermani et al., 2021]. A key factor in the success of ensemble learning is error diversity; that is, the degree of disagreement between ensemble members [Brown et al., 2005a]. Ensemble learning frameworks use diversity generating techniques to improve model generalisability. Detailed reference to diversity-in-learning for ensemble learning methods can be found in [Brown et al., 2005a, Díez-Pastor et al., 2015b, Gong et al., 2019]. The present research uses three ensemble algorithms: bagging, stacking, and boosting. Brief descriptions of each algorithm are provided below; a more detailed review of each modelling approach can be found in Zounemat-Kermani et al. [2021].

Bagging, short for bootstrap aggregating, involves training a series of predictors on diverse training sets, which are generated by resampling the original training data with replacement [Breiman, 1996a]. Aggregated predictions are obtained through the use of the mean or majority vote of the predictors. Stacking, short for stacked generalisation, is an ensemble learning approach in which an ensemble integration model is trained to combine ensemble members [Wolpert, 1992]. Any suitable model may be used as the combiner. Recently, stacking has been shown to be capable of outperforming simple model averaging for combining data-driven streamflow forecasting models [Tyralis et al., 2021]. Boosting is an ensemble algorithm in which an initial model is trained, followed by a sequence of models; each additional model in the sequence is trained based on the error of the previously trained models [Freund and

Table 3.3: Confusion matrix illustrating how true and false positives and negatives form the basis for the TPR, FPR, PPV, TNR, and accuracy.

	predicted flood	predicted no flood	
observed flood	TP	FN	$\text{TPR} = \frac{\text{TP}}{\text{TP} + \text{FN}} = 1 - \text{FNR}$
observed no flood	FP	TN	$\text{FPR} = \frac{\text{FP}}{\text{TN} + \text{FP}} = 1 - \text{TNR}$
	$\text{PPV} = 1 - \text{FDR} = \frac{\text{TP}}{\text{TP} + \text{FP}}$	$\text{TNR} = \frac{\text{TN}}{\text{TN} + \text{FP}}$	$\text{Acc.} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}}$

Schapire, 1996]. Adaptive boosting (AdaBoost), which is the variant used in this research, trains each new model with resampled training data. The training data for each model after the first in the ensemble are resampled based on a weight vector that is calculated using the residuals of the ensemble prediction using models trained up to that point.

3.4.4 Performance metrics

In regression, evaluation of flood streamflows typically uses purposeful performance metrics, such as the Nash-Sutcliffe Efficiency Index on High Streamflows [Crochemore et al., 2015, Mizukami et al., 2019] or, complex visual measures [Ehret and Zehe, 2011, Ewen, 2011]. In binary classification, model performance is based on the numbers of correct and incorrect classifications.

The most basic way to quantify classifier performance is with the accuracy, which is simply the sum of correct classifications (flood and no flood) over the total number of samples. However, accuracy alone is insufficient for evaluating classifiers, especially in the case of highly imbalanced flood data, for which a model could miss every flood while still maintaining a high accuracy. Consequently, performance metrics are based on the number of correct floods, missed floods, correct non-floods, and incorrect non-floods predictions, which are denoted as true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN), respectively. These are visualised in Table 3.3, along with their respective rates (TPR, TNR, FPR, and FNR, respectively), accuracy, positive predictive value (PPV), and false discovery rate (FDR). There are many alternate names for these criteria; in this work, we refer to TPR and PPV as recall and precision, respectively.

Using measures simultaneously is redundant [Branco et al., 2016]. Instead, these rates form the basis for several criteria, some of which being better suited for heavily imbalanced classification problems [Branco et al., 2016, Muñoz et al., 2021]. Of the criteria suited for imbalanced problems, this study used the F-measure and G-Mean in Eqns. 16 and 17, respectively.

These measures can also manage the trade-off in performance between simple measures. For example, the F-measure is a function of both recall and precision, accounting for the trade-off between them and eliminating the need to simultaneously consider both criteria:

$$F_{\beta} = \frac{(1 + \beta)^2 \cdot TPR \cdot PPV}{\beta^2 \cdot PPV + TPR} \quad (16)$$

F_{β} is the F-measure and β can be tuned to reflect the relative importance of recall and precision, such that recall is β times more important than precision. Since the selection trade-off between recall and precision is subjective, this work considered values of 1 and 2. F_1 weights recall and precision equally, whereas in F_2 , recall is twice as important as precision.

Model accuracy is not a reliable metric for flood forecasting. An improved metric is the G-Mean, which balances the accuracies on the true and negative classes regardless of class size [Kubat, Miroslav; Matwin, 1997]:

$$\text{G-Mean} = \sqrt{\text{TPR} + \text{TNR}} \quad (17)$$

Visual tools including the receiver operating characteristic (ROC) and precision recall (PRC) curves were also used to assess model performance. Binary classifiers typically output decimal values between 0 and 1, which are often interpreted as the probability of a true prediction. By default, classifiers use a discrimination threshold of 0.5 to distinguish between positive and negative predictions. Lowering the discrimination threshold increases the FPR and TPR, while increasing the threshold decreases these rates. ROC curves illustrate the trade-off between FPs and TPs across increasing threshold values. ROC curves have an ideal point in the top left of the plot, and can be condensed to a single value, the area under the curve (AUC). A trimmed AUC [Aggarwal and Sathe, 2017] was used, which only used FP values between 0 and 400 for the AUC calculation; models behaving beyond this range are considered impractical and irrelevant to FEWS. Similarly to ROC, PRC curves visualise the trade-off between precision and recall, making them better suited than ROC curves for imbalanced data such as river floods [Saito and Rehmsmeier, 2015]. In contrast to the ROC curve, PRC curves have an ideal point at the top right of the plot.

3.5 Results and discussion

3.5.1 Resampling hyperparameter grid-search

The outcome of the SMOTE and RUS hyperparameter grid-search is shown in Fig. 3.6 in the form of performance on the resampled training data (a) and unmodified test data (b). The results for all three watersheds agreed with one another, so we can discuss findings in a general manner. According to Fig. 3.6(a), model performance improved according to all metrics with increasing SMOTE and RUS rates. This was to be expected, as both rates increased the relative number of flood samples in the training dataset, and the performance metrics selected focus on flood forecasting capability.

However, the improvements observed on the training dataset did not all transfer to the test dataset, as shown in Fig. 3.6(b). This disparity in performance was not due to overfitting, but rather due to the different train and test distributions; the training dataset contained proportionally more flood observations, which may not provide a reliable sense of model performance. For example, the models may have performed well on synthetic flood samples, but that did not guarantee strong performance on observed data. Nevertheless, overall preprocessing tended to have a favourable effect on the test performance. Increasing SMOTE and RUS rates improved the TPRs. However, the F-scores show that while SMOTE is associated with a slight improvement in F-scores, increasing RUS rates was very detrimental to F-scores. These poor F-scores were due to increases in FPs, at more than twice a rate of the increases in TPs. In some cases, undersampling with no SMOTE produced slight improvements in both F-scores, however, comparatively larger performance gains were obtained using SMOTE and no RUS.

Respective SMOTE and RUS rates of 4 and $10^{0.5}$ were selected to preprocess the training data in subsequent experiments, which were found to produce the most favourable trade-off between all metrics while also reducing the computational expense of the experiments. The SMOTE rate was sufficiently large to generate improvements in F-score and G-Mean, while the RUS rate was small enough that it did not negatively impact the F-scores, which remained similar to the scores obtained with no RUS resampling.

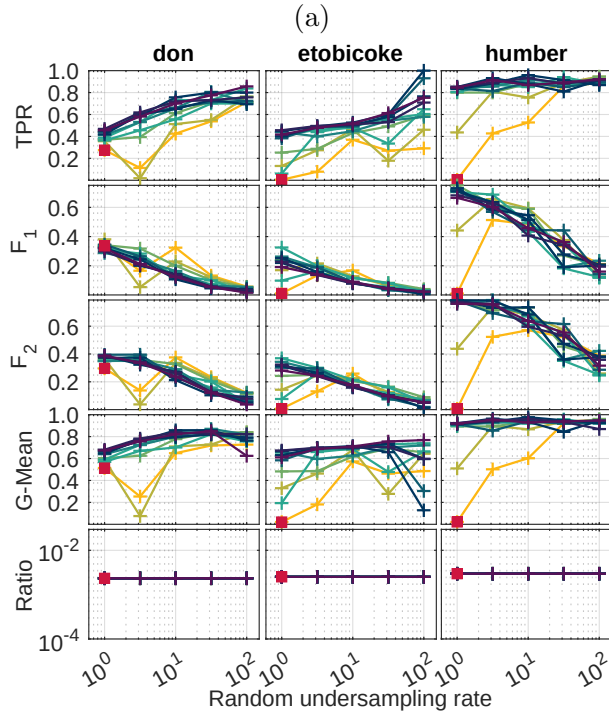
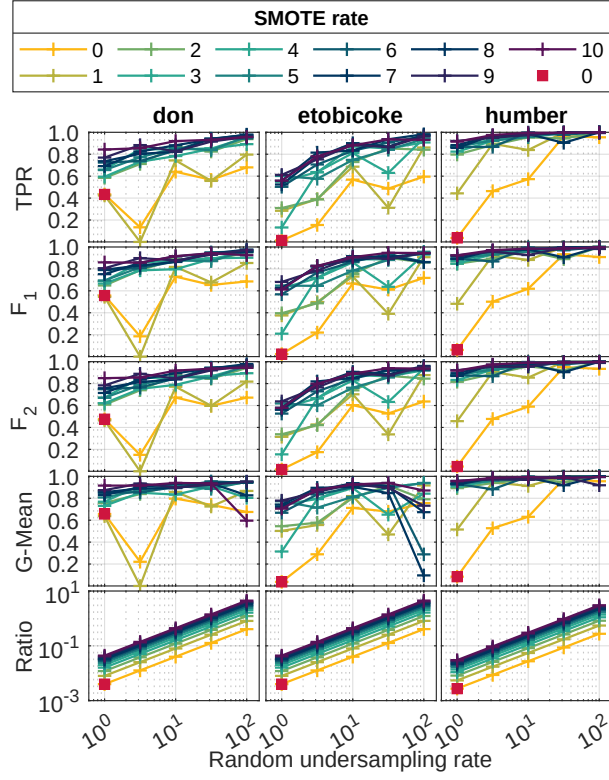


Figure 3.6: Train (a) and test (b) performance of bagged ANN classifiers for varying SMOTE (indicated by colours) and RUS (x-axis) resampling rates. The red square indicates the original, non-resampled training data.

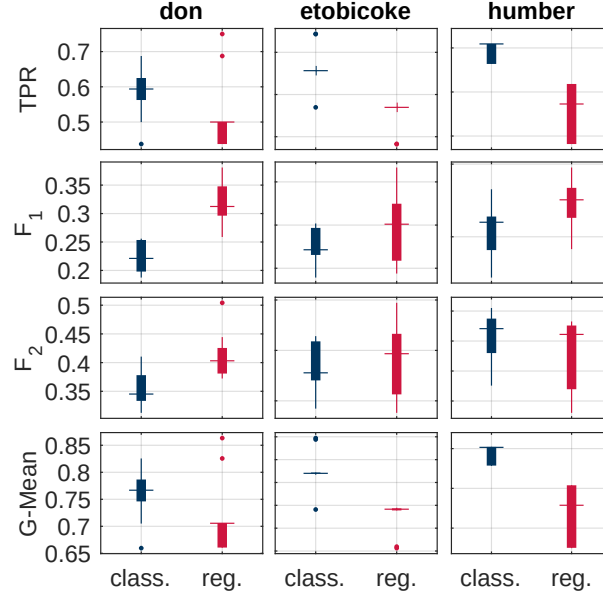


Figure 3.7: Classification (blue) and regression (red) model for Don, Etobicoke, and Humber River watersheds according to the FN, F_1 , F_2 , and G-Mean metrics. Uncertainty bands are generated by through bootstrap resampling of the training data.

3.5.2 Comparison between classification and regression models

As mentioned in Sec. 3.3, there are several potential advantages to a classification approach to FEWS, one of which is simplifying the learning task. In this section, we compared the performance of a binary classifier to a typical regression approach, in which flood events were obtained from the simulated stage predictions. Fig. 3.7 illustrates differences in performance of each configuration. The uncertainty range of the boxplots was generated by 100 training instances, each with randomised initial model parameters and trained to bootstrapped training data. The classifier consistently outperformed the regression model according to the TPR and G-Mean metrics, whereas the regression model tended to have slightly better F-scores. The F-score disparity was attributed to the tendency of classifiers to produce slightly more FPs compared to regression models, as the latter have a strong tendency to underpredict flood flows without the use of specialised cost functions [Zhao et al., 2011, Toth, 2016]. This experiment demonstrated that a classification-based approach to flood forecasting is capable of improved performance, depending on the selected metrics, when compared to a typical regression model, warranting further experiments to guide model selection.

3.5.3 Ensemble model intercomparison

In order to aid in model selection and to gain an improved understanding of the performance of classifiers for FEWS, we conducted an intercomparison of base learners and ensemble algorithms. A systematic approach, evaluating each combination of learner and algorithm, was used to understand how learners perform across ensemble algorithms, and vice versa.

First, a grid-search of the number of base learners was conducted, which is a critical hyperparameter in ensemble learning. Figs. 3.9 - 3.10 illustrate test performance across ensemble sizes ranging from 1 to 128, increasing in squared increments. Increasing the ensemble size promotes ensemble diversity, thereby was expected to improve generalisation. However, in practice, increasing the ensemble size was at times detrimental to model performance (e.g., boosting with SVMs in Fig. 3.9 and 3.8).

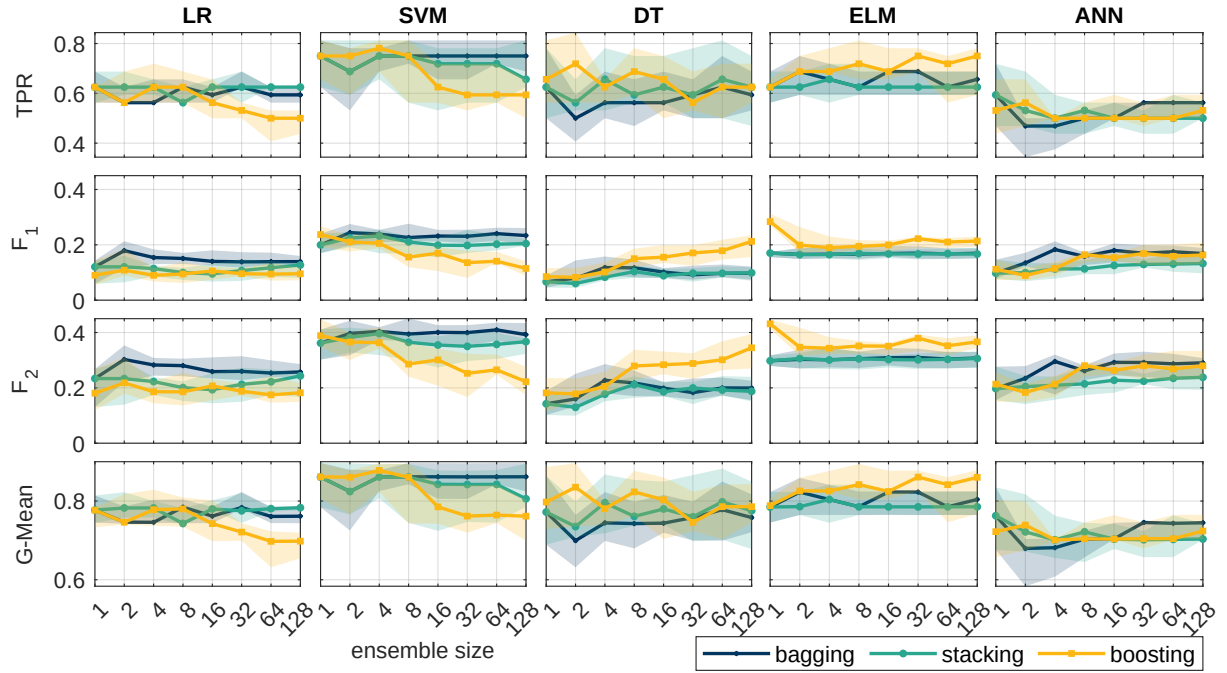


Figure 3.8: Flood forecast performance for the Don River LR, SVM, DT, ELM, and ANN models (subplot columns) as used in bagging (blue), stacking (green), and boosting (yellow), according to FNR, F_1 , F_2 , and G-Mean (subplot rows). Transparent uncertainty envelopes are obtained through bootstrapping the training set.

The performance of the ensembles of DTs and ANNs consistently improved with ensemble size across all metrics, which was most pronounced by the F-scores. In contrast, the performance of the SVM and LR boosting ensembles tended to decrease across an increasing ensemble size, suggesting that these model types are not well-suited at fitting the training data generated by

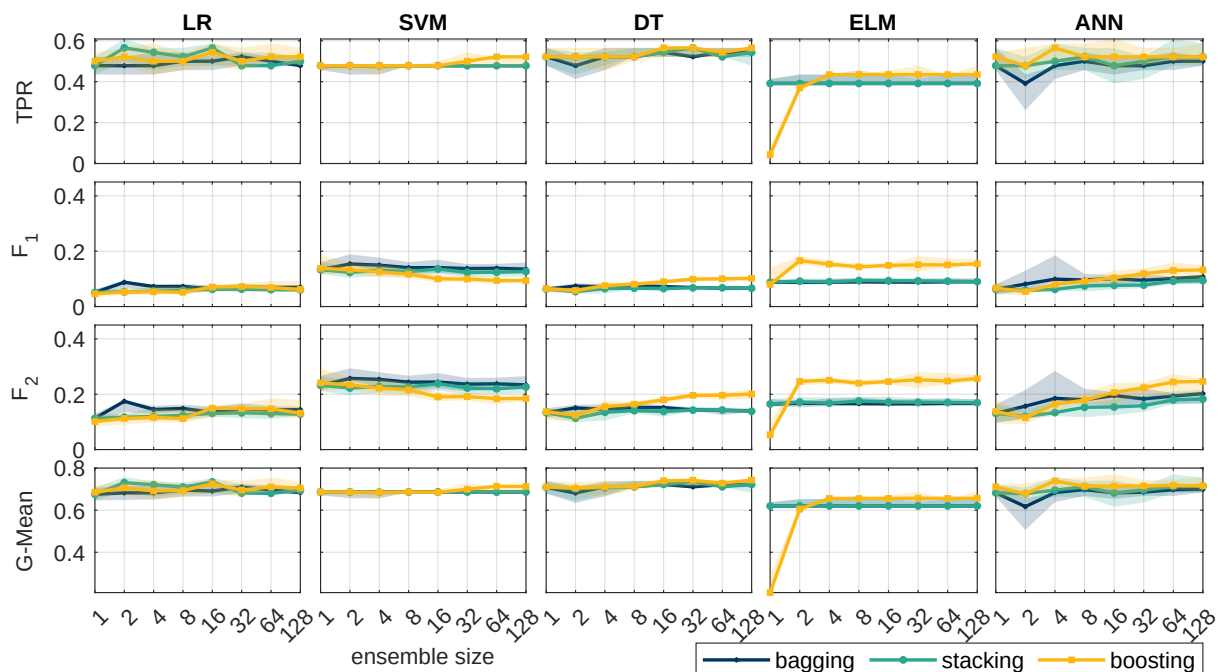


Figure 3.9: Flood forecast performance for Etobicoke Creek LR, SVM, DT, ELM, and ANN models (subplot columns) as used in bagging (blue), stacking (green), and boosting (yellow), according to FNR, F_1 , F_2 , and G-Mean (subplot rows). Transparent uncertainty envelopes are obtained through bootstrapping the training set.

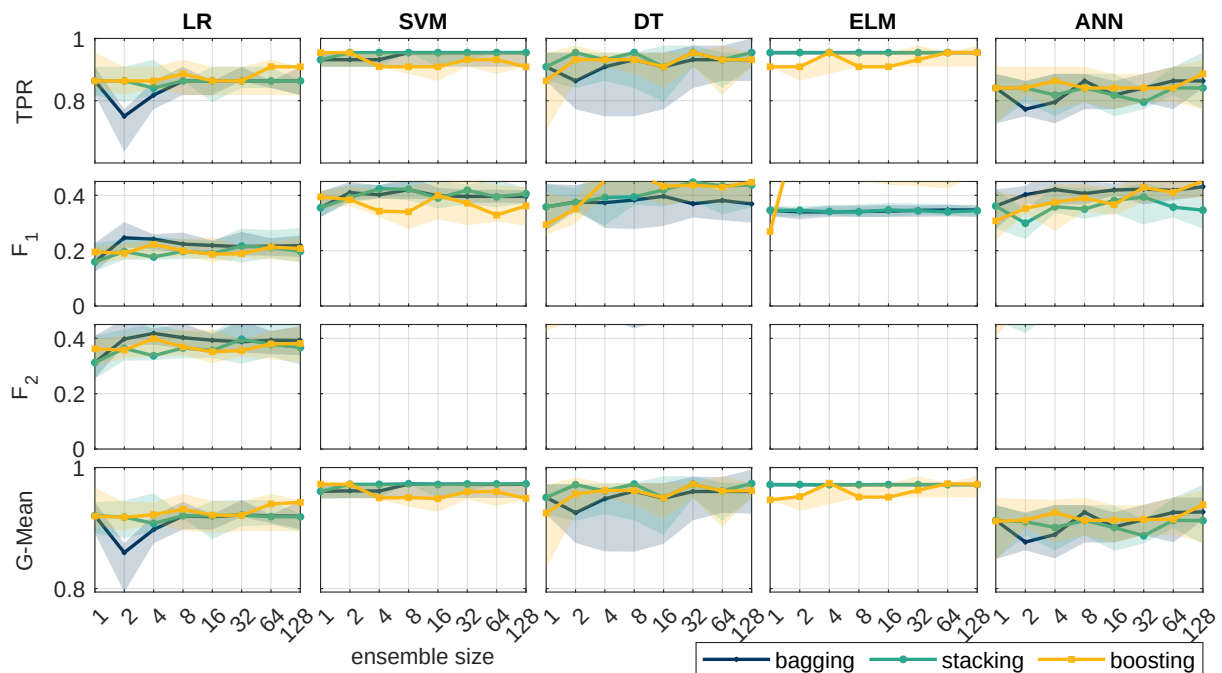


Figure 3.10: Flood forecast performance for the Humber River LR, SVM, DT, ELM, and ANN models (subplot columns) as used in bagging (blue), stacking (green), and boosting (yellow), according to FNR, F_1 , F_2 , and G-Mean (subplot rows). Transparent uncertainty envelopes are obtained through bootstrapping the training set.

the reweighting mechanism of the boosting algorithm. This is consistent with previous work, which associated oversized boosted ensembles with overfitting [Liu et al., 2014]. The bagging and stacking algorithms were comparatively less sensitive to ensemble size than boosting. The performance of bagged ensembles improved slightly with size, whereas stacked ensembles were typically indifferent to size. A notable outlier configuration was the boosted ELM. For the Don River, it exhibited very strong F-scores with a small ensemble size — unlike the other two watersheds, for which the ELM exhibited a decrease in F-score performance across ensemble size. For all three watersheds, applying ELMs in a boosting algorithm results in drastically better performance than within stacking or bagging algorithms.

The mean performance of each ensemble member model and ensemble algorithm is shown in Tables 3.4 and 3.5, respectively. The scores were obtained by averaging the performance of each bootstrapped training instance, selecting the highest-performing ensemble size for each configuration, and then averaging the performance of each learner or ensemble configuration. According to these tables, boosted ensembles and SVMs consistently achieved the best average performance. The performance of ELMs, while high in boosted ensembles, was lowered by averaging across all three ensemble algorithms.

Table 3.4: Mean performance of ensemble algorithms for each watershed and metric. Best performance between ensemble algorithms is indicated in bold.

metric	watershed	bagging	stacking	boosting
TPR	Don	0.656	0.656	0.688
	Etobicoke	0.487	0.504	0.526
	Humber	0.914	0.914	0.932
F ₁	Don	0.179	0.153	0.202
	Etobicoke	0.103	0.0909	0.122
	Humber	0.368	0.366	0.445
F ₂	Don	0.309	0.279	0.333
	Etobicoke	0.191	0.174	0.219
	Humber	0.568	0.563	0.615
G – Mean	Don	0.802	0.802	0.819
	Etobicoke	0.688	0.699	0.715
	Humber	0.951	0.951	0.960

The high performance of SVMs can be attributed to its complexity, which allows for diverse members in an ensemble framework. In contrast, ensembles comprised of ANNs are comparatively more complex than SVMs, and may not achieve the same degree of ensemble diversity, as all the ensemble members will converge to the same solution. In contrast, ensembles consisting of LRs, may not be complex enough to fit the non-linear relationship between input

Table 3.5: Mean performance of ensemble members for each watershed and metric. Best performance between ensemble members is indicated in bold.

metric	watershed	LR	SVM	DT	ELM	ANN
TPR	Don	0.625	0.760	0.667	0.698	0.583
	Etobicoke	0.543	0.493	0.558	0.406	0.529
	Humber	0.879	0.955	0.947	0.955	0.864
F ₁	Don	0.138	0.238	0.145	0.208	0.161
	Etobicoke	0.0743	0.143	0.0814	0.117	0.111
	Humber	0.229	0.414	0.441	0.456	0.426
F ₂	Don	0.255	0.398	0.261	0.349	0.271
	Etobicoke	0.152	0.245	0.165	0.200	0.210
	Humber	0.404	0.623	0.640	0.636	0.605
G – Mean	Don	0.783	0.867	0.804	0.829	0.755
	Etobicoke	0.723	0.696	0.734	0.633	0.717
	Humber	0.929	0.973	0.969	0.973	0.926

and target features [Hipni et al., 2013]. The success of boosting and SVMs in this study is particularly notable due to the relative infrequency with which these methods appear in the literature. The strong performance of the boosted ensembles can be attributed to their mechanism to explicitly generate diversity, which cannot be said about bagging or stacking [Brown et al., 2005a]. Nevertheless, a recent review of ensemble techniques in hydrology found that boosting represents only 20% of ensemble applications. While DTs are commonly used in ensembles, SVMs and ELMs are used in fewer than 15% and 10% of ensemble frameworks, respectively [Zounemat-Kermani et al., 2021]. The strong performance of ELMs is consistent with several recent studies, which found that ELMs outperformed ANNs for streamflow forecasting [Yaseen et al., 2018, Seo et al., 2018, Atiquzzaman and Kandasamy, 2016].

3.5.3.1 ROC and PRC curves

The ROC curves for each watershed and model configurations are illustrated in Fig. 3.11 below. The 128-model ensembles were used for each configuration, which were chosen because increased ensemble size seldom damaged performance. The shape of the ROC curves and AUC values largely followed the findings described above, with the boosted ELM and SVMs producing the best AUC values across watersheds. One exception was the LR ensembles for Etobicoke Creek, which produced competitive AUC values despite notably poor performance seen in Fig. 3.9. The shape of a steep then flat shape of the ROC curve suggests that the discrimination threshold of the model could be lowered without any reduction in TPs, which would in turn improve metrics such as PPV.

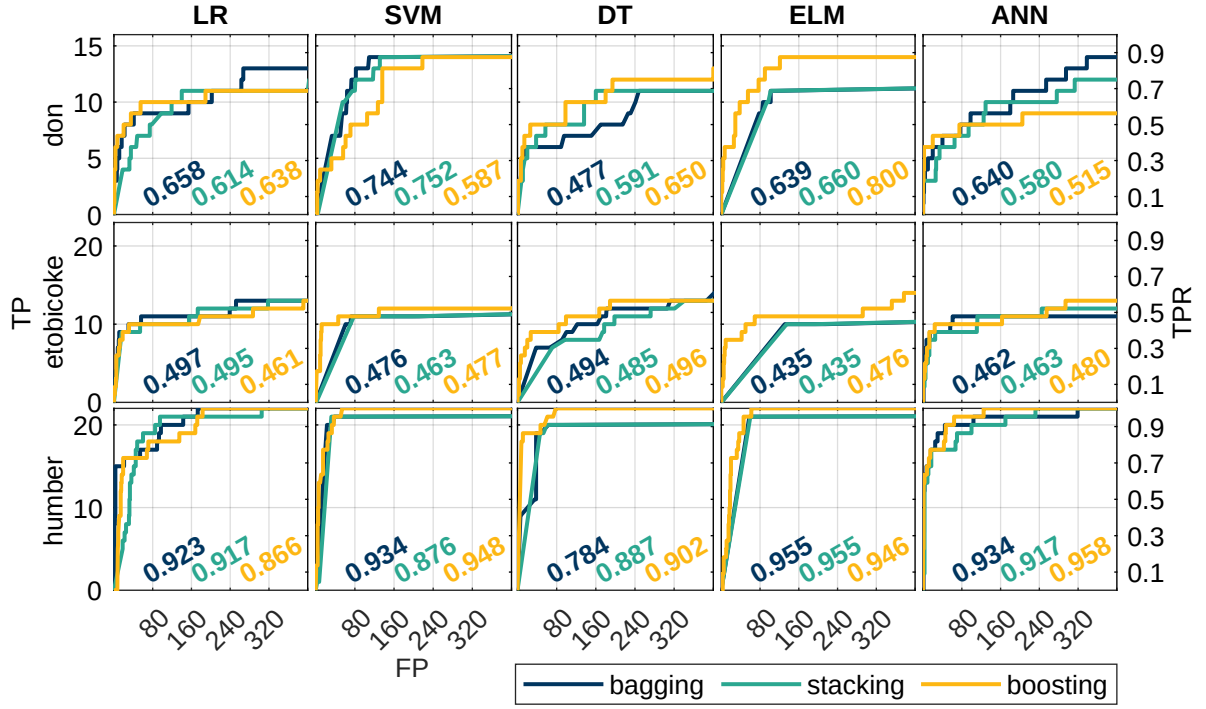


Figure 3.11: ROC curves for the Don (top row), Etobicoke (middle), and Humber (bottom) for LR, SVM, DT, ELM, and ANN models (left to right columns), and bagging (blue), stacking (green) and boosting (yellow) ensemble algorithms. The number of false positives is shown along the x-axis, and the absolute and relative numbers of true positives are shown along the left and right y-axes. AUC values are listed in the respective ensemble algorithm colour and subplot.

Next, Fig. 3.12 illustrates PRC curves. Some PRC curves began at a non-zero TPR value, which occurred when an incremental change in the threshold value influenced the predicted class of multiple samples simultaneously. This is due to under-dispersion of the predicted values and is an undesirable trait, since it limits the ability to fine-tune model performance using the classification threshold. The boosted ensembles, LR, and ANN models were the most well-dispersed methods. However, improved dispersion does not guarantee better FEWS performance. For example, the boosted DT was well dispersed for the Don River case, and the ROC/PRC curves were superior to the other ensemble methods for very small discrimination thresholds. However, the other ensemble learning methods outperformed the boosted DT on TPR values between 0.5 and 0.7, on Fig. 3.11, which is the range most likely to be selected for FEWS.

3.5.3.2 Watershed comparison

Given their proximity and similar characteristics (e.g., land cover, climate, rainfall), all three

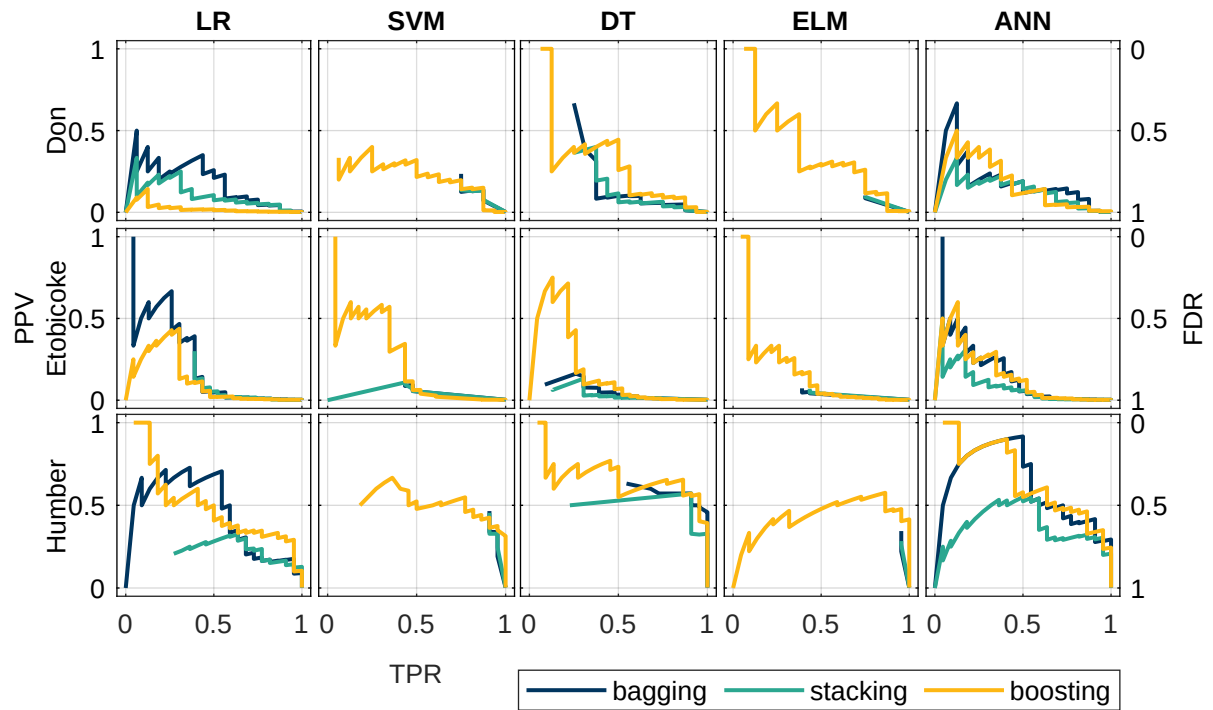


Figure 3.12: PRC curves for the Don (top row), Etobicoke (middle), and Humber (bottom) for LR, SVM, DT, ELM, and ANN models (left to right columns), and bagging (blue), stacking (green) and boosting (yellow) ensemble algorithms. The TPR is shown along the x-axis, and the PPV and FDR are shown along the left and right y-axes.

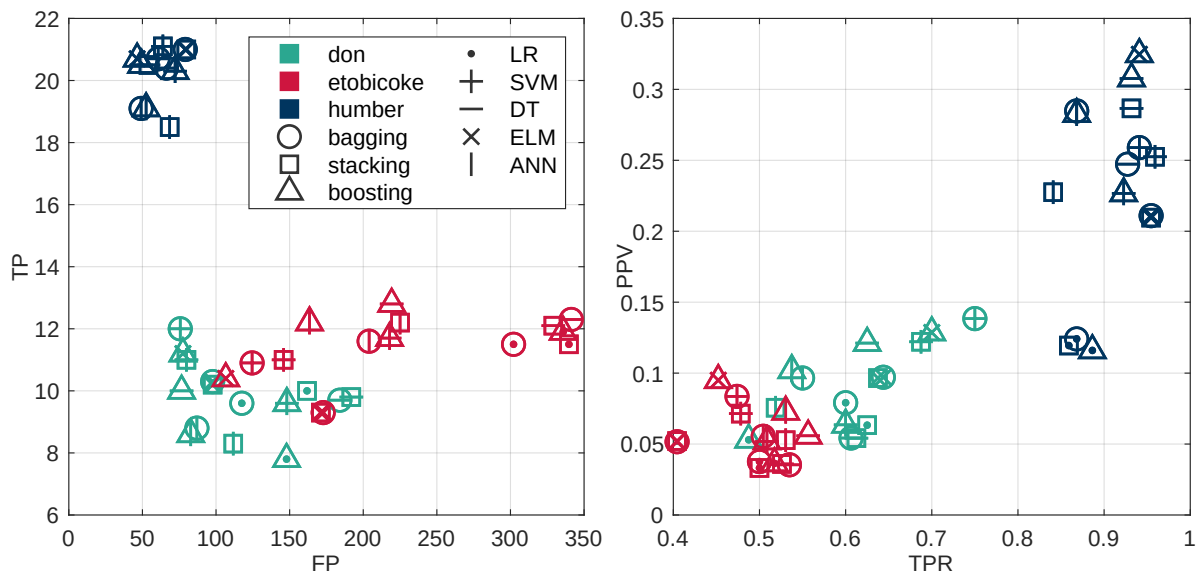


Figure 3.13: Classifier performance, plotted as TP against FP (left) and PPV against TPR (right), with ideal performance in the top-right and top-left corners, respectively. The fifteen unique ensemble configurations are denoted by a unique symbol and the Etobicoke, Don, and Humber models are coloured in green, red, and blue, respectively.

watersheds are expected to contain similar hydrological processes. Therefore, accurate flood forecasting is expected to be comparably challenging, thus we compare the performance scores

obtained for each watershed directly. Fig. 3.13 illustrates the performance of all fifteen ensemble configurations for all three watersheds on individual subplots. Ensemble configurations and watersheds are distinguished by symbol and colour, respectively. Between the three watersheds, the Etobicoke Creek models showed the poorest performance, followed by the Don, then the Humber models. The trend in performance can be attributed to watershed size and relative permeable land cover. Due to its size, the Humber had a greater amount of upstream gauges and a larger travel time between upstream and the target station. The Humber river was also the least urbanised of the three watersheds. The relatively permeable headwaters of the Humber reduced direct runoff generation, which increased the lag time between rainfall and runoff peaks. Both of these factors resulted in flood events that occur more gradually compared to the other two watersheds, making it easier to achieve accurate performance. There were also notable patterns between watersheds and ensemble configurations. The LR and DT models had the poorest performance for each watershed, and even in ensemble frameworks are unable to capture the flood dynamics of the system. Ensembles of LR and DT models were most likely to generate FPs, which was especially noticeable when looking at the spread of FP values for Etobicoke Creek models and PPV of Humber River.

3.6 Conclusions

This study analysed the effectiveness of a classification-based approach to streamflow forecasting for FEWS for three urbanised watersheds surrounding Toronto, Canada. RUS and SMOTE resampling were used to reduce the imbalance of flood classes. A two-dimensional grid-search was revealed that SMOTE is more effective than RUS in improving TPR performance while minimising FPs. The performance of a bagged ANN as a classifier and regressor was compared, in order to demonstrate the advantages of a pure classification approach. While both models exhibited similar performance, the binary model had a better G-Mean and FNR, at the expense of more FPs. Next, an exhaustive comparison of five ML models (LR, DT, SVM, ELM, and ANN) and three ensemble algorithms (bagging, stacking, and boosting) was performed, and it was found that the ELM and SVM models within a boosting ensemble framework tend to have the strongest classification performance overall. The boosting ensemble performance is particularly sensitive to the ensemble size, to a greater degree than the other ensemble algorithms. An analysis of the ROC and PRC curves found that the boosting algorithm performed well, which is consistent with the previous finding, but that the SVM

and ELMs were not very well dispersed. The ANN was comparably more dispersed, despite exhibiting poor performance in the Don. Comparing model performance across watersheds, it was determined that predictions in the Humber River were the most accurate, followed by the Don, then Etobicoke Creek, which was attributed to watershed size.

The findings of this work emphasize the advantages of a pure classification approach to flood forecasting. In particular, this work studies the trade-off between TPs and FPs across different resampling schemes and classification thresholds. Understanding this trade-off is essential for FEWS deployment. Next, this work highlights the importance of accounting for flood data imbalance, which may go unaddressed when using common regression-based metrics. Using classification metrics, that are specialised for imbalanced problems, this work highlights the need to correct class imbalance using methods such as SMOTE and RUS. Next, the exhaustive model comparison finds that boosting, SVMs, and ELMs offer the best FEWS performance for the three featured watersheds. While these models are relatively less common than methods such as ANNs in the literature, the results of this study demonstrated their aptitude for FEWS. Finally, the comparison of three similar urban watersheds reveals idiosyncratic behaviour, which suggests that model selection should be conducted on a case-by-case basis to maximise model performance.

The main limitation of this work is that it was conducted on three urban watersheds. Even in selecting similar watersheds, idiosyncratic behaviour was observed, which was attributed to watershed size and the availability and distance of upstream data. Therefore, these findings may not generalise to watersheds with different hydrological characteristics. Therefore, future work on classification-based approaches should use large-sample datasets in order to validate model types across a wider range of watersheds.

4 Resampling and ensemble techniques for improving ANN-based high flow forecast accuracy

4.1 Preface

The following chapter features the continued investigation of the ensemble research in Sec. 3. In this chapter, the SMOTE resampling method, which was used to generate synthetic flood data, was embedded within ensemble algorithms, to exploit the benefits of ‘diversity-in-learning’ [Brown et al., 2005b]. The content of this chapter was published in Hydrology and Earth System Sciences in 2021 as follows:

Snieder, E., Abogadil, K., and Khan, U. T. (2021). Resampling and ensemble techniques for improving ANN-based high-flow forecast accuracy, *Hydrology and Earth System Sciences*, 25, 2543–2566, <https://doi.org/10.5194/hess-25-2543-2021>.

Preliminary work on this topic was submitted to the 2019 AGU General Assembly (poster presentation), with the following citation:

Snieder, E., Khan, U. T. (2019). A comprehensive evaluation of boosting algorithms for artificial neural network-based flow forecasting models. Poster presentation at the AGU General Assembly. San Francisco, USA.

Conference publications were peer-reviewed and are included with minor modifications for the sake of fluidity. Results from poster presentations, which are not included in any other publications, are summarised as technical notes.

Contributions to this chapter are as follows: **E. Snieder:** conceptualisation; data curation; formal analysis; investigation; methodology; visualisation; writing - original draft. **K. Abogadil:** review of literature; writing - draft, editing. **U. T. Khan:** conceptualisation; funding acquisition; supervision; writing - editing, revisions.

4.2 Abstract

Data-driven flow forecasting models, such as Artificial Neural Networks (ANNs), are increasingly featured in research for their potential use in operational riverine flood warning systems. However, the distributions of observed flow data are imbalanced, resulting in poor prediction accuracy on high flows, both in terms of amplitude and timing error. Resampling and ensemble techniques have shown to improve model performance on imbalanced datasets. However, the efficacy of these methods (individually or combined) has not been explicitly evaluated for improving high flow forecasts. In this research, we systematically evaluate and compare three resampling methods: random undersampling (RUS), random oversampling (ROS), and synthetic minority oversampling technique for regression (SMOTER); and four ensemble techniques: randomised weights and biases, Bagging, adaptive boosting (AdaBoost), least squares boosting (LSBoost); on their ability to improve high stage prediction accuracy using ANNs. These methods are implemented both independently and in combined, hybrid techniques, where the resampling methods are embedded within the ensemble methods. This systematic approach for embedding resampling methods are novel contributions. This research presents the first analysis of the effects of combining these methods on high stage prediction accuracy. Data from two Canadian watersheds (the Bow River in Alberta, and the Don River in Ontario), representing distinct hydrological systems, are used as the basis for the comparison of the methods. The models are evaluated on overall performance, and on typical and high stage subsets. The results of this research indicate that resampling produces marginal improvements to high stage prediction accuracy, whereas ensemble methods produce more substantial improvements, with or without resampling. Many of the techniques used produced an asymmetric trade-off between typical and high stage performance; reduction of high stage error resulted in disproportionately larger error on typical stage. The methods proposed in this study highlight the diversity-in-learning concept and help support for future studies on adapting ensemble algorithms for resampling. This research contains many of the first instances of such methods for flow forecasting and moreover, their efficacy to address the imbalance problem and heteroscedasticity, which are commonly observed in high flow and flood forecasting models.

4.3 Introduction

Data-driven models such as artificial neural networks (ANNs) have been widely and successfully used over the last three decades for hydrological forecasting applications [Govindaraju, 2000-04-31, Abrahart et al., 2012, Dawson and Wilby, 2001]. However, some studies have noted that these models can exhibit poor performance during high flow (or stage) hydrological events [Sudheer et al., 2003, Abrahart et al., 2007, de Vos and Rientjes, 2009], with poor performance manifesting as late predictions (i.e., timing error), under-predictions, or both. For flow forecasting applications such as riverine flood warning systems, the accuracy of high stage predictions are more important than that of typical stage. One cause of poor model accuracy on high stage is the scarcity of representative sample observations available with which to train such models [Moniz et al., 2017a]. This is because stage data typically exhibits a strong positive skew, referred to as an imbalanced domain; thus, there may only be a small number of flood observations within decades of samples. Consequently, objective functions that are traditionally used for training ANNs (e.g., mean squared error (MSE), sum of squared error (SSE), etc.), that equally consider all samples, are biased towards values that occur most frequently and reflected by poor model performance on high flow or stage observations [Pisa et al., 2019]. Sudheer et al. [2003] also point out that such objective functions are not optimal for non-normally distributed data. This problem is exacerbated when such metrics are also used to assess model performance; regrettably, such metrics are the most widely used in water resources applications [Maier et al., 2010]. As a result, studies that assess models using traditional performance metrics risk overlooking deficiencies in high stage performance.

Real-time data-driven flow forecasting models frequently use antecedent input variables (also referred to as autoregressive inputs) for predictions. Several studies have attributed poor model prediction on high stage to model over-reliance on antecedent variables [Snieder et al., 2020, Abrahart et al., 2007, de Vos and Rientjes, 2009, Tongal and Booij, 2018]. Consequently, the model predictions are similar to the most recent antecedent conditions, sometimes described as a lagged prediction [Tongal and Booij, 2018]. In other words, the real-time observed stage at the target gauge is used as the predicted value for a given lead time. This issue is closely linked to the imbalanced domain problem as frequently occurring stage values typically exhibit low temporal variability compared to infrequent, high stage values; this phenomenon is further described in Sect. 4.4.

Improving the accuracy of high stage or flow forecasts has been the focus of many studies. Several studies have examined the use of preprocessing techniques to improve model performance. [Sudheer et al. \[2003\]](#) propose using a Wilson-Hilferty transformation to change the skewed distribution of stage data. The study found that transforming the target data reduces annual peak flow error produced by ANN-based daily flow forecasting models. [Wang et al. \[2006\]](#) evaluate three strategies for categorising streamflow samples, based on a fixed value flow threshold, unsupervised clustering, and periodicity; separate ANN models are trained to predict each flow category and combined to form a final prediction. The periodicity-based ANN, which detects periodicity from the autocorrelation function of the target variable, is found to perform the best out of the three schemes considered. [Fleming et al. \[2015\]](#) address the issue of poor high flow performance by isolating a subset of daily high flows by thresholding based on a fixed value. By doing so, traditional objective functions (e.g., MSE) become less influenced by the imbalance of the training dataset. ANN-based ensembles trained on high flows are found to perform well, though the improvements to high flow accuracy are not directly quantified, as the high flow ensemble is not compared directly to a counterpart trained using the full training dataset.

An alternative approach to improving high flow forecast accuracy has been to characterise model error as having amplitude and temporal components [[Seibert et al., 2016](#)]. [Abrahart et al. \[2007\]](#) use a specialised learning technique in which models are optimised based on a combination of root mean square error (RMSE) and a timing error correction factor, which is found to improve model timing for short lead-times, but have little impact on higher lead times. [de Vos and Rientjes \[2009\]](#) use a similar approach, in which models that exhibit a timing error are penalised during calibration. The technique is found to generally reduce timing error at the expense of amplitude error.

Finally, there is considerable evidence that ensemble-based and resampling techniques to improve prediction accuracy of infrequent samples [[Galar et al., 2012](#)]. Ensemble methods, such as bootstrap aggregating (Bagging) and boosting, are known for their ability to improve model generalisation. Such methods are widely used in classification studies and are increasingly being adapted for regression tasks [[Moniz et al., 2017b](#)]. However, ensemble methods alone do not directly address the imbalance problem, as they typically do not explicitly consider the distribution of the target dataset. Thus, ensemble methods are often combined with prepro-

cessing strategies to address the imbalance problem [Galar et al., 2012]. Resampling, which is typically used as a preprocessing method, can be used to create more uniformly distributed target dataset or generate synthetic data with which to train models [Moniz et al., 2017a]. Resampling also promotes diversity-in-learning when embedded in ensemble algorithms (rather than used as a preprocessing strategy). Examples of such combinations appear in machine learning literature, but are typically developed for ad hoc applications [Galar et al., 2012].

However, the efficacy of these methods (a combination of resampling strategies with ensemble methods) has not been systematically investigated for flow forecasting applications. While previous studies have provided comparisons of ensemble methods, none have explicitly studied their effects on high flow prediction accuracy, which has only received little attention within the context of the imbalance problem in general. Additionally, previous research uses resampling as a preprocessing technique, whereas in this research, resampling is embedded within the ensembles to promote diversity-in-learning. Thus, the main objective of this research is to develop a systematised framework for combining several different resampling and ensemble techniques with the aim to improve high flow forecasts using ANNs. Three resampling techniques: random undersampling (RUS), random oversampling (ROS), and synthetic minority oversampling technique for regression (SMOTER) and four ensemble algorithms: randomised weights and biases (RWB), Bagging, adaptive boosting for regression (AdaBoost), and least-squares boosting (LSBoost) will be investigated to address the issues related to high flow forecasts, i.e., the imbalanced domain problem and heteroscedasticity. Each combination of these methods will be explicitly evaluated on their ability to improve model performance on high stage (infrequent) data subsets along with the typical (frequent) data subsets. Such a framework and comparison, to address the imbalanced domain, has not been presented in existing literature. Lastly, while only selected resampling and ensemble techniques are presented, many of which are the first instances of their use for high flow forecasting, this proposed framework may easily be expanded to resampling and ensemble strategies beyond those included in this research.

The remainder of the manuscript is organised as follows: first, in Sect. 4.4 we present the baseline ANN flow forecast models, which are used as the individual learners for the ensembles, for two Canadian watersheds, followed by a performance analysis of these models to highlight the imbalance domain problem and illustrates the heteroscedasticity of baseline model

residuals. The two watersheds, with differing hydrological characteristics, but both prone to riverine floods, are the Bow River watershed (in Alberta), and the Don River watershed (in Ontario). Sect. 4.5 provides a review and applications of each resampling method and ensemble technique, followed by a description of the implementation of each approach in this research, and model evaluation methods. Lastly, Sect. 4.6 includes the results and discussion from the two case studies.

4.4 Early investigations

The following section provides descriptions for the two watersheds under study. The parametrisation of the single ANN models to predict stage in each watershed (referred to as the individual learners) is described. The output of the individual learners are used to exemplify the inability of these ANNs to accurately predict high stage (from both an amplitude and temporal error perspective) and to illustrate the imbalance problem.

4.4.1 Study area

The Bow and Don Rivers are featured as case studies in this research to evaluate methods for improving the accuracy of high stage data-driven forecasts. The Bow River, illustrated in Fig. 4.1 (a), begins in the Canadian Rockies mountain range and flows eastward through the City of Calgary, where it is joined by the Elbow River. The Bow River's flow regime is dominated by glacial and snowmelt processes which produce annual seasonality. The Bow River watershed has an area of approximately $7,700\text{km}^2$ upstream of the target stage monitoring station in Calgary and consists of predominantly natural and agricultural land cover. The City of Calgary has experienced several major floods (recently in 2005 and 2013) and improvements to flow forecasting models have been identified as a key strategy for mitigating flood damage [Khan et al., 2018, , 2014].

The Don River, illustrated in Fig. 4.1 (b), begins in the Oak Ridges Moraine and winds through the Greater Toronto Area until it meets Lake Ontario in downtown Toronto. The 360km^2 Don River watershed is heavily urbanised which results in the high stage seen in the River to be attributable to the direct runoff following intense rainfall events. Its urbanised landscape has also contributed to periodic historical flooding [Toronto and Region Conservation Authority, 2020a]. Persistent severe flooding (recently in 2005 and 2013) have motivated calls for further

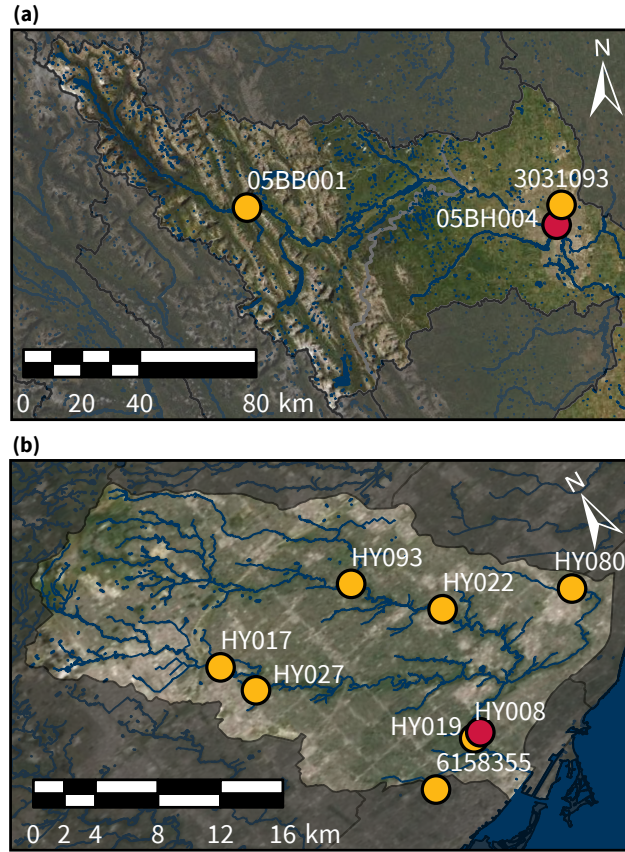


Figure 4.1: Bow (a) and Don (b) River basins upstream of Calgary and Toronto, respectively. Surface watercourses and waterbodies are shown in blue. The target stage monitoring stations are red while upstream hydrometeorological monitoring stations (stage, precipitation, and temperature) are yellow. Aerial imagery obtained from © Esri [Esri, 2020]. Surface water and watershed boundaries obtained from © Scholars GeoPortal [DMTI Spatial Inc., 2019, 2014c,a,b] and the © TRCA [TRCA, 2018]

mitigation strategies such as improved flow forecast models and early warning systems [Nirupama et al., 2014].

Data from November to April and November to December were removed from the Bow and Don River datasets, prior to any analysis; these periods are associated with ice conditions. The histograms in Figure 4.2 illustrate the imbalanced domains of the target stage for both rivers. A high stage threshold (Θ_{HS}) is defined, which is used to distinguish between typical and high stage. Stage values greater than the threshold are referred to as high stage (q_{HS}) while stage below the threshold, as typical stage (q_{TS}). Target stage statistics for the Bow and Don Rivers are provided for the complete stage distribution, as well as the q_{TS} and q_{HS} subsets, in Table 4.1.

The use of a fixed threshold for distinguishing between common (frequent) and rare (infre-

Table 4.1: Target variable statistics for the Bow and Don River watersheds.

River	Subset	Mean [m]	Min. [m]	Max. [m]	Skew. [-]	Var. [m ²]
Bow	q	1.28	0.92	3.07	1.18	0.067
	q _{TS}	1.18	0.92	1.47	0.21	0.022
	q _{HS}	1.69	1.47	3.07	1.85	0.039
Don	q	77.62	77.51	79.21	3.78	0.018
	q _{TS}	77.58	77.51	77.67	0.59	0.0017
	q _{HS}	77.82	77.68	79.21	2.99	0.034

quent) samples is used both in flow forecasting [Crochemore et al., 2015, Razali et al., 2020, Fleming et al., 2015] and in more general machine learning studies that are focused on the imbalance problem [Moniz et al., 2017a]. In this research, the high stage threshold is simply and arbitrarily taken as the 80th percentile value of the observed stage. The threshold value is ideally derived from the physical characteristics of the river (i.e., the stage at which water exceeds the bank or associated with a specified return period); unfortunately, this site-specific information is not readily available for the subject watersheds. An important consideration to make while selecting a Θ_{HS} value is that it produces a sufficient number of high stage samples; too few samples risks overfitting and poor generalisation. The distinction between typical and high stage is used in some of the resampling techniques in Sect. 4.5.1 and for assessing model performance in Sect. 4.5.4.

4.4.2 Individual learner description

The individual learner (sometimes called the base model, or base learner) for both systems use upstream hydro-meteorological inputs (stage, precipitation, and temperature) to predict the downstream stage (the target variable). The multi-layer perception (MLP) ANN is used as the individual learner for this study, and the selected model hyperparameters are summarised in Table 4.3. The MLP-ANN was chosen as the individual learner because it is the most commonly used machine learning architecture for predicting water resources variables in river systems [Maier et al., 2010]. The individual learner can be used for discrete value prediction or as a member of an ensemble, in which a collection of models are trained and combined to generate predictions. Each ANN has a hidden layer of 10 neurons; a grid-search of different hidden layer sizes indicated that larger numbers of hidden neurons have little impact on the ANN performance. Thus, to prevent needlessly increasing model complexity, a small hidden layer is favoured. The number of training epochs is determined using early-stopping

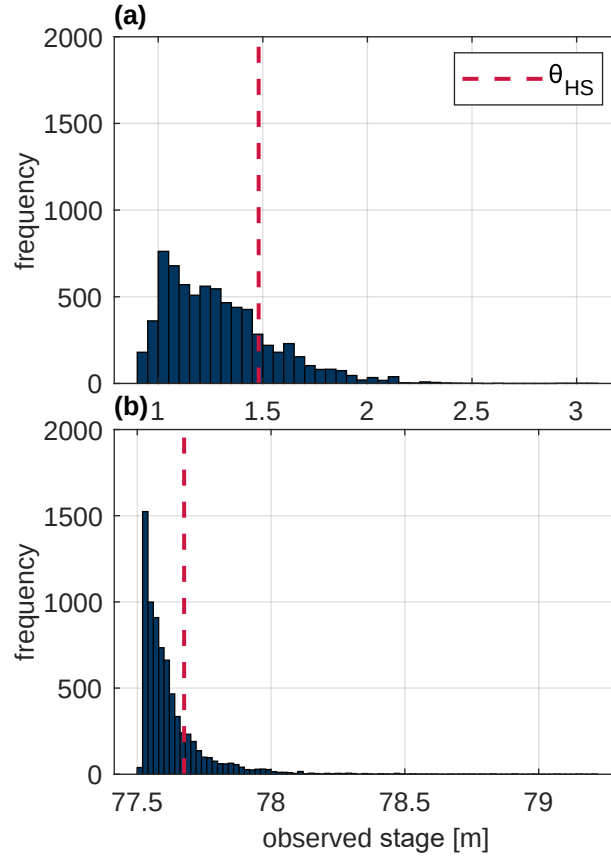


Figure 4.2: Histograms of observed stage for the (a) Bow River 6-hour stage and (b) Don River hourly stage. The dashed red line indicates the fixed threshold used to distinguish between typical and high stage values.

(also called stop-training), which is performed by dividing the calibration data into training and validation subsets; training data is used to tune the ANN weights and biases, whereas the validation performance is used to determine when to stop training [Ancil and Lauzon, 2004a]. For this study, the optimum number of epochs is assumed if the error on the validation set increases for 5 consecutive epochs. Early-stopping is a common technique for achieving generalisation and preventing overfitting [Ancil and Lauzon, 2004a]. Of the available data for each watershed, 60% is used for training, 20% for validation, and 20% for testing (the independent dataset). K-fold cross-validation (KFCV) is used to evaluate different continuous partitions of training and testing data, and is explained in greater detail in Sect. 4.5.4.2. The Levenberg–Marquardt algorithm was used to train the individual learners, because of its speed of convergence and reliability [Lauzon et al., 2006, Maier and Dandy, 2000, Tongal and Booij, 2018]. The full set of input and target variables used for both catchments are summarised in Table 4.2. For both rivers, the input variables are used to forecast the target variable 4 timesteps in advance, i.e., for the Bow River, the model forecasts 24 hours in the fu-

Table 4.2: Input variables for the Bow and Don Rivers.

Catchment	Variable	Station ID	Statistics	Data source	Lag times
Bow River	Water level	05BB001, 05BH004*	Max, min, mean 6-hour	Water Survey of Canada	0:11
6-hour timestep	Precipitation	031093	Cumulative 6-hour	City of Calgary	0:11
24-hour forecast	Temperature	031093	Max, min, mean 6-hour	City of Calgary	0:11
Don River	Water level	HY017, HY019*, HY022, HY080, HY093	Hourly	TRCA	0:5
1-hour timestep	Precipitation	HY008, HY927	Hourly	TRCA	0:11
4-hour forecast	Temperature	6158355	Hourly	Environment Canada	0:5

Table 4.3: Individual learner ANN model description used for both watersheds.

Model class	Artificial neural network
Architecture	Multi-layer perceptron
IVS	Partial correlation
Hidden neurons	10
Activation function	Tanh (hidden layer), Linear (output layer)
Training algorithm	Levenburg-Marquardt backpropagation
Stopping criteria	Validation dataset

ture, whereas for the Don River, the model forecasts 4 hours in the future. Some of the input variables used in the Bow River model, including the minimum, mean, and maximum statistics, are calculated by coarsening hourly data to a 6-hour timestep. Several lagged copies of each input variable are used, which is common practice for ANN-based hydrological forecasting models [Snieder et al., 2020, Abbot and Marohasy, 2014, Fernando et al., 2009, Banjac et al., 2015]. For example, to forecast x_t by 4 timesteps, x_{t-4} , x_{t-5} , x_{t-6} , etc. may be used as input variables, as these variables are recorded automatically, in real-time.

The Partial Correlation (PC) input variable selection (IVS) algorithm is used to determine the most suitable inputs for each model from the larger candidate set [He et al., 2011, Sharma, 2000]. Previous research for the Don and Bow Rivers found that PC is generally capable of removing non-useful inputs in both systems, achieving reduced computational demand and improved model performance [Snieder et al., 2020]. The simplicity and computational efficiency of the PC algorithm method makes it an appealing IVS algorithm for this application. The 25 most useful inputs amongst all the candidates listed in Table 4.2, determined by the PC algorithm, are used in the models for each watershed. A complete list of selected inputs is shown in Appendix D.1.1.

The Bow and Don River individual learners produce coefficients of Nash-Sutcliffe efficiency (CE) greater than 0.95 and 0.75, respectively. These scores are widely considered by hydrologists to indicate good performance [Crochemore et al., 2015]. However, closer investigation of the model performance reveals that high stage samples consistently exhibit considerable error.

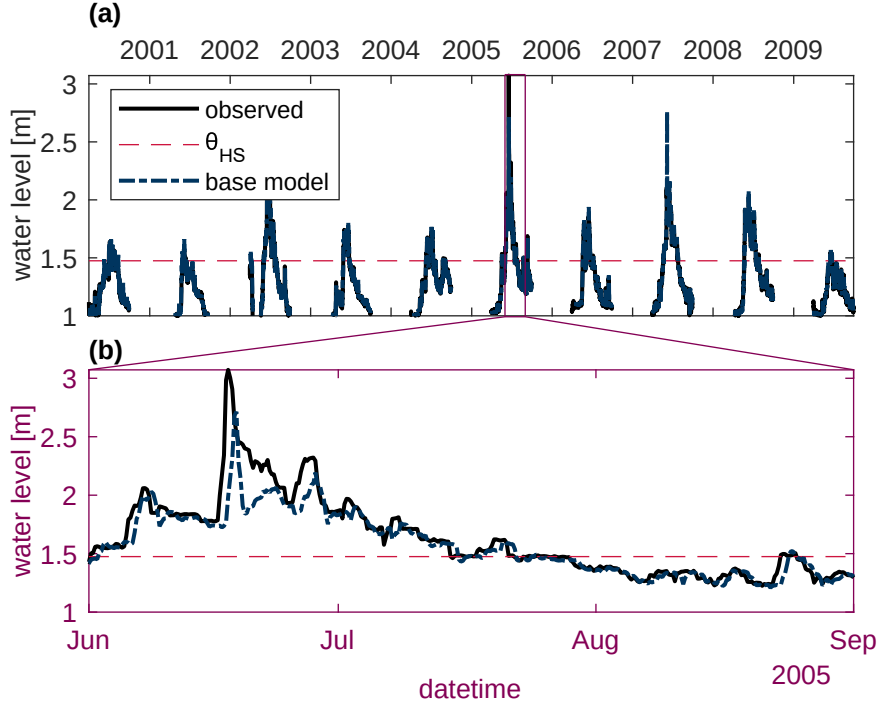


Figure 4.3: Observed and individual learner stage predictions for the Bow River system for all 10 years of available stage (a) and a 3 month subset which contains particularly high stage (b), to better distinguish between the two hydrographs. The dashed red line indicates the fixed threshold used to distinguish between typical and high stage values.

Such is plainly visible when comparing the observed hydrographs with the individual learner predictions, as shown in Figs. 4.3 and 4.4, for the Bow and Don Rivers, respectively. Plotting the individual learner residuals against the observed stage, as in Fig. 4.5 (a and b) illustrates how the variance of the residuals about the expected mean of 0 increases with the increasing stage magnitude; Fleming et al. [2015] also describe the heteroscedastic nature of flow prediction models. This region of high stage also exhibits amplitude errors in the excess of 1 meter, casting doubt on the suitability of these models for flood forecasting applications. In Fig. 4.5 (b and c) the normalised inverse frequency of each sample point is plotted against the stage gradient, illustrating how the most frequent stage values typically have a low gradient with respect to the forecast lead time, given by $(q_{t+L} - q_t)/L$. Note that the inverse frequency is determined using 100 histogram bins. Thus, when such a relationship exists, it is unsurprising that model output predictions are similar to the most recent autoregressive input variable. Previous work that analysed trained ANN models for both subject watersheds demonstrates how the most recent autoregressive input variable is the most important variable for accurate stage predictions [Snieder et al., 2020].

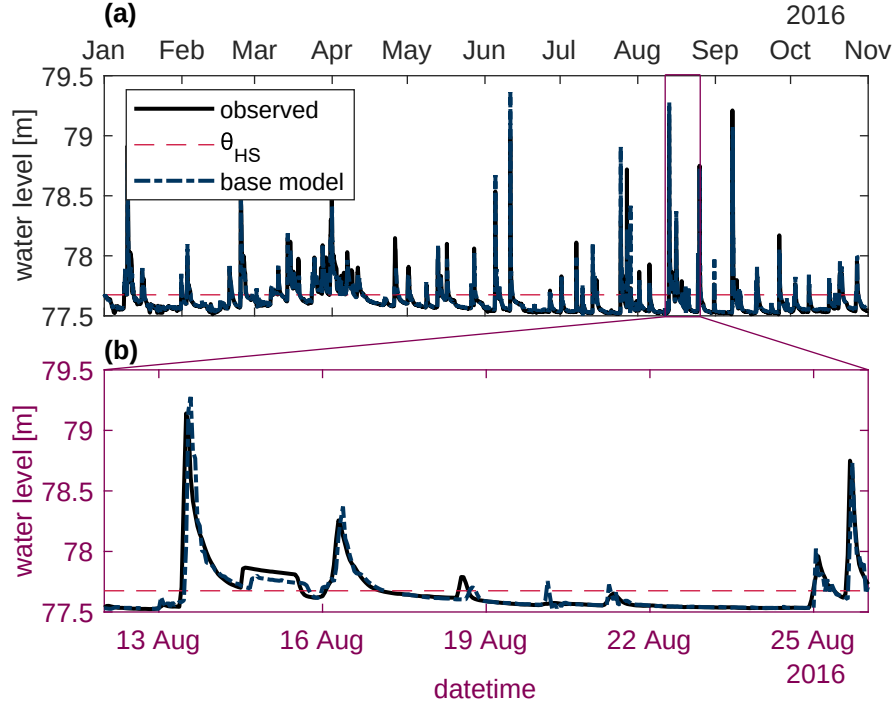


Figure 4.4: Observed and individual learner stage predictions for the Don River system for all 10 months of available stage (a) and a 14 day subset which contains particularly high stage (b), to better distinguish the two hydrographs. The dashed red line indicates the fixed threshold used to distinguish between typical and high stage values.

Without accounting for the imbalanced nature of stage data, data-driven models are prone to inadequate performance similar to that of the individual learners described above. Consequently, such models may not be suitable for flood related applications such as early flood warning systems. The following section describes, and reviews resampling and ensemble methods, which are proposed as solutions to the imbalance problem, which manifests as poor performance on high stage samples, relative to typical stage.

4.5 Review and description of methods for handling imbalanced target datasets

Many strategies have been proposed for handling imbalanced domains, which can be broadly categorised into three approaches: specialised preprocessing, learning methods, and combined methods [Haixiang et al., 2017, Moniz et al., 2018]. According to a comprehensive review of imbalanced learning strategies resampling and ensemble methods are among the most popular techniques employed [Haixiang et al., 2017]. Specifically, a review of 527 papers on imbalanced classification found that a resampling technique was used 156 times [Haixiang et al., 2017]. From the same review, 218 of the 527 papers used an ensemble technique such as Bagging or boosting. Many of the studies reviewed used combinations of available techniques and often

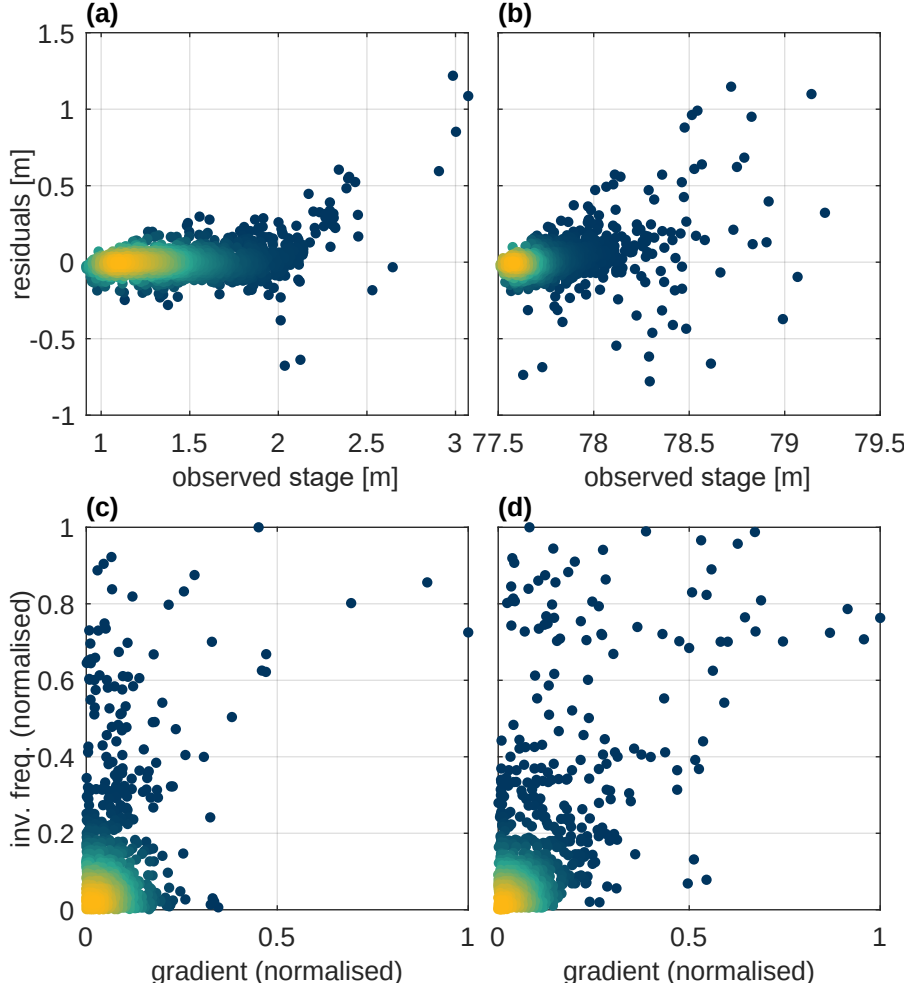


Figure 4.5: Baseline model residuals versus observed stage for the Bow (a) and Don (b) River systems. Inverse frequency versus gradient across 4 time steps for the Bow (c) and Don (d) River target variables. Colouring indicates normalised scatter point density.

propose novel hybrid approaches that incorporate elements from several algorithms. Since it is impractical to compare every unique algorithm that has been developed for handling imbalanced data, the scope of this research adheres to relatively basic techniques and combinations of resampling and ensemble methods. The following sections describe the resampling and ensemble methods used in this research. The review attempts to adhere to hydrological studies that feature each of the methods, however, when this is not always possible, examples from other fields are presented.

First, it is important to distinguish between the data imbalance addressed in this study and cost-sensitive imbalance. Imbalance in datasets can be characterised as a combination of two factors: imbalanced distributions of samples across the target domain and imbalanced user interest across the domain. Target domain imbalance is related solely to the native distribu-

tion of samples while cost-sensitivity occurs when costs vary across the target domain. While both types of imbalance are relevant to the flow forecasting application of this research, cost-sensitive methods are complex and typically involve developing a relationship between misprediction and tangible costs, for example, property damage [Toth, 2016]. Cost-sensitive learning is outside the scope of this research, which is focused on reducing high stage errors due to the imbalanced nature of the target stage data.

4.5.1 Resampling techniques

Resampling is widely used in machine learning to create subsets of the total available data with which to train models. Resampling is typically used as a data preprocessing technique [Brown et al., 2005a, Moniz et al., 2017a]. However, in our research, resampling is embedded in the ensemble algorithms, as to promote diversity amongst the individual learners. This following section discusses examples of resampling, whether used for preprocessing or used within the learning algorithm. Pseudocode for each resampling method is provided in Appendix D.2.

4.5.1.1 Random undersampling

RUS is performed by subsampling a number of frequent cases equal to the number of infrequent cases, such that there are an even amount in each category and achieve a more balanced distribution compared to the original set. As a result, all of the rare cases are used for training, while only a fraction of the normal cases are used. RUS is intuitive for classification problems; for two-class classification, the majority class is undersampled such that the number of samples drawn from each class is equal to the number of samples in the minority class [Yap et al., 2014]. However, RUS is less straightforward for regression, as it requires continuous data first to be categorised, as to allow for an even number of samples to be drawn from each category. Categories must be selected appropriately such that they are continuous across the target domain and each category contains a sufficient number of samples to allow for diversity in the resampled dataset [Galar et al., 2013]. Undersampling is scarcely used in hydrological forecasting applications, despite seeing widespread use in classification studies. Ruhana et al. [2016] demonstrate an application of fuzzy-based RUS for categorical flood risk support vector machine (SVM) based classification, which is motivated by the imbalanced nature of flood data. RUS is found to outperform both ROS and synthetic minority oversampling technique

(SMOTE) on average across 5 locations.

In this research, N available stage samples are categorised into N_{TS} typical stage and N_{HS} high stage based on the threshold Θ_{HS} . The undersampling scheme draws N_{HS} with replacement from each of the subsets, such that there are an equal number of each category. RUS can be performed with or without replacement; the former provides greater diversity when resampling is repeated several times, and thus this approach is selected for the present research.

4.5.1.2 Random oversampling

ROS simply consists of oversampling rare samples, thus modifying the training sample distribution through duplication [Yap et al., 2014]. ROS is procedurally similar to RUS, also aiming to achieve a common number of frequent and infrequent samples. Instead of subsampling the typical stage, high stage values are resampled with replacement so that the number of samples matches that of the typical stage set. The duplication of high stage samples in the training dataset increases their relative contribution to the model’s objective function during calibration. Compared to undersampling, oversampling is advantaged such that more samples in the majority class are utilised. The drawbacks of this approach are that there is an increased computational cost. There are few examples of ROS applications in water resources literature; studies tend to favour SMOTE, which is discussed in the following section. Saffarpour et al. [2015] use oversampling to address the class imbalance of binary flood data; surprisingly, oversampling was found to decrease classification accuracy compared to the raw training dataset. Recently, Zhaowei et al. [2020] applied oversampling for vehicle traffic flow, as a response to the imbalance of the training data.

For ROS, as with RUS, N available stage samples are categorised into N_{TS} typical stage and N_{HS} high stage samples based on the threshold Θ_{HS} . The oversampling scheme draws N_{TS} with replacement from each of the subsets, such that there are an equal number of each category. ROS is distinguished from RUS in that it produces a larger sample set that inevitably contains duplicated high stage values.

4.5.1.3 Synthetic minority oversampling technique for regression

SMOTER is a variation of the SMOTE classification resampling technique introduced by [Chawla et al., 2002] that bypasses excessive duplication of samples by generating synthetic samples, which unlike duplication, creates diversity within the ensembles. SMOTE is widely

considered as an improvement over simple ROS as the increased diversity help prevents overfitting [Ruhana et al., 2016]. For a given sample, SMOTE generates synthetic samples by randomly selecting one of k nearest points, determined using k -nearest neighbours (KNN), and sampling a value at a linear distance between the two neighbouring points. The original SMOTE algorithm was developed for classification tasks; Torgo et al. [2013] developed the SMOTER variation, which is an adaptation of SMOTE for regression. SMOTER uses a fixed threshold to distinguish between 'rare' and 'normal' points. In addition to oversampling synthetic data, SMOTER also randomly undersamples normal values, to achieve the desired ratio between rare and normal samples. The use of SMOTE in the development of models that predict river stage is only being recently attempted. Atieh et al. [2017] use two methods for generalisation: Dropout and SMOTER; these were applied to ANN models that predicted the flow duration curves for ungauged basins. They found that SMOTER reduced the number of outlier predictions, whereas both approaches resulted in the improved performance of the ANN models. Wu et al. [2020] used SMOTE resampling in combination with AdaBoosted sparse Bayesian models. The combination of these methods resulted in improved model accuracy compared to previous studies using the same dataset. Razali et al. [2020] used SMOTE with various Bayesian network and machine learning techniques, including decision trees, KNN and SVM. Each technique is applied to an imbalanced classified flood dataset (flood flow and non-flood flow categories); the SMOTE decision tree model achieved the highest classification accuracy. SMOTE decision trees have also been applied for estimating the pollutant removal efficiency of bioretention cells. Wang et al. [2019a] found that decision trees developed with SMOTE had the highest accuracy for predicting pollutant removal rates; the authors attribute the success of SMOTE to its ability to prevent the majority class from dominating the fitting process. Sufi Karimi et al. [2019] employ SMOTER resampling for stormwater flow prediction models. Their motivation for resampling is flow dataset imbalance and data sparsity. Several configurations are considered with varying degrees of oversampled synthetic and undersampled data. The findings of the study indicate that increasing the oversampling rate tends to improve model performance compared to the non-resampled model, while increasing the undersampling rate produces a marginal improvement. Collectively, these applications of SMOTE affirm its suitability for mitigating the imbalance problem in the flood forecasting models featured in this research.

SMOTER is adapted in this research following the method described by [Torgo et al., 2013]. One change in this adaptation is that rare cases are determined using the θ_{HS} value, instead of a relevancy function. Similarly, only high values are considered as 'rare', instead of considering both low and high values as rare, as in the original algorithm. Oversampling and undersampling are performed at rates of 400% and 0% respectively, as to obtain an equivalent number of normal and rare cases.

4.5.2 Ensemble-based techniques

Ensembles are collections of models (called individual learners), each with variations to the individual learner model type or to the training procedure [Alobaidi et al., 2019]. It is well established that ensemble-based methods improve model stability and generalisability [Alobaidi et al., 2019, Brown et al., 2005a]. Recent advances in ensemble learning have emphasised the importance of diversity-in-learning [Alobaidi et al., 2019]. Diversity can be generated both implicitly and explicitly through a variety of methods, some of which include varying the initial set of model parameters, varying the model topology, varying the training algorithm, and varying the training data [Sharkey, 1996, Brown et al., 2005a]. The largest source of diversity in the ensembles under study is attributable with varying the training data, which occurs both in the various resampling methods described above and the in some cases, the ensemble algorithms. Only homogeneous ensembles are used in this work, thus no diversity is obtained through varying the model topology or training algorithm [Zhang et al., 2018, Alobaidi et al., 2019]. Ensemble predictions are combined to form a single discrete prediction. Ensembles that are combined to produce discrete predictions have been proven to outperform single models by reducing model bias and variance, thus improving overall model generalisability [Brown et al., 2005a, Sharkey, 1996, Shu and Burn, 2004, Alobaidi et al., 2019]. This has contributed to their widespread application in hydrological modelling [Abrahart et al., 2012]. In some cases, ensembles are not combined, and the collection of predictions are used to estimate the uncertainty associated with the diversity between ensemble members [Tiwaree and Chatterjee, 2010a, Abrahart et al., 2012]. While this approach has obvious advantages, it is not possible for all types of ensembles, such as the boosting methods, which are also used in this research. Thus, this research combines ensembles to aid comparison across the different resampling and ensemble methods used.

There are many distinct methods for creating ensemble methods. The purpose of this paper is not to review all ensemble algorithms, but rather to compare four ensemble methods that commonly appear in literature: Bagging, adaptive boosting, and gradient boosting. A fourth method, randomised weights and biases, which does not qualify as an ensemble technique due to the absence of repeated resampling, is also included in the ensemble comparison because of its widespread use. While several studies have provided comparisons of ensemble methods, none of these studies have explicitly studied their effects on high stage prediction, nor their combination with resampling strategies, which is common in applications outside of flow forecasting.

Methods that aim to improve generalisability have shown promise in achieving improved prediction on high stage, which may be scarcely represented in training data. However, to the knowledge of the authors, no research has explicitly evaluated the efficacy of ensemble-based methods for improving high stage accuracy. Applications of ensemble methods for improving performance of imbalanced target variables have been thoroughly studied in classification literature. Several classification studies have demonstrated how ensemble techniques can improve prediction accuracy for imbalanced classes [Galar et al., 2012, López et al., 2013, Díez-Pastor et al., 2015b,a, Błaszczyński and Stefanowski, 2015]. Such methods are increasingly being adapted for regression problems, which is typically achieved by projecting continuous data into a classification dataset [Moniz et al., 2017b,a, Solomatine and Shrestha, 2004]. Pseudocode for each of the ensemble algorithms used in this research is provided in Appendix D.2.

4.5.2.1 Randomised weights and biases

While not technically a form of ensemble learning, repeatedly randomising the weights and biases of ANNs is one of the simplest and most common methods for achieving diversity among a collection of models, thus, it acts as a good comparison point for the proceeding ensemble methods [Brown et al., 2005a]. In this method, members are only distinguished by the randomisation of the initial parameter values (i.e., the initial weights and biases for ANNs in this research) used for training. For this method, an ensemble of ANNs is trained, each member having a different randomised set of initial weights and biases. Thus, when trained, each ensemble member may converge to different final weight and bias values. Ensemble members are combined through averaging. This technique is often used, largely to alleviate variability in training outcomes and uncertainty associated with the initial weight and bias parameteri-

sation [Shu and Burn, 2004, de Vos and Rientjes, 2005, Fleming et al., 2015, Barzegar et al., 2019]. Despite its simplicity, this method has been demonstrated to produce considerable improvements in performance when compared to a single ANN model, even outperforming more complex ensemble methods [Shu and Burn, 2004]. The weights and biases of each ANN are initialised using the default initialisation function in MATLAB, and an ensemble size of 20 is used.

4.5.2.2 Bagging

Bagging is a widely used ensemble method, first introduced in [Breiman, 1996a]. Bagging employs the bootstrap resampling method, which consists of sampling with replacement, to generate subsets of data on which to train ensemble members. The ensemble members are combined through simple averaging to form discrete predictions. Bagging is a proven ensemble method in flood prediction studies and has been widely applied and refined for, both spatial and temporal prediction, since its introduction by Breiman [1996a]. Chapi et al. [2017] use Bagging with Logistic Model Trees (LMT) as the individual learners to predict spatial flood susceptibility. The Bagging ensemble is found to outperform standalone LMTs, in addition to logistic regression and Bayesian logistic regression. For a similar flood susceptibility prediction application, Chen et al. [2019] use Bagging with Reduced Error Pruning Trees (REPTree) as the base learners. The Bagged models are compared to Random Subspace ensembles; both ensemble methods perform better than the standalone REPTree models, with the Random Subspace model slightly outperforming the Bagged ensemble. Anctil and Lauzon [2004a] compared five generalisation techniques in the development of ANNs for flow forecasting. They combined Bagging, boosting and stacking with stop training and Bayesian regularisation, making a total of nine model configurations. They found that stacking, Bagging, and boosting all resulted in improved model performance, ultimately recommending the use of the last two in conjunction with either stop training or Bayesian regularisation. Ouarda and Shu [2009] compared stacking and Bagging ANN models against parametric regression for estimating low flow quantile for summer and winter seasons and found higher performance in ANN models (single and ensemble) compared to traditional regression models [Ouarda and Shu, 2009]. Cannon and Whitfield [2002] applied Bagging to MLP-ANN models for predicting flow and found that Bagging helped create the best performing ensemble ANN. Shu and Burn [2004] evaluated six approaches for creating ANN ensembles for regional flood frequency

flood analysis, including Bagging combined with either simple averaging or stacking; Bagging resulted in higher performance compared to the basic ensemble method. In a later study, [Shu and Ouarda \[2007\]](#) used Bagging and simple averaging to create ANN ensembles for estimating regional flood quantiles at ungauged sites. Implementing Bagging is uncomplicated, a description of the algorithm is described in its original appearance [\[Breiman, 1996a\]](#). This research uses a Bagging ensemble of 20 members.

4.5.2.3 Adaptive boosting for regression

The AdaBoost algorithm was originally developed by [Freund and Schapire \[1996\]](#) for classification problems. The algorithm has undergone widespread adaptation and its popularity has led to the development of many variations, which typically introduce improvements in performance, efficiency, and expanded for regression problems. This study uses the AdaBoost.RT variation [\[Solomatine and Shrestha, 2004, Shrestha and Solomatine, 2006\]](#). Broadly put, the AdaBoost algorithm begins by training an initial model. The following model in the ensemble is trained using a resampled or reweighted training set, based on the residual error of the previous model. This process is typically repeated until the desired ensemble size is achieved or a stopping criterion is met. Predictions are obtained by weighted combination of the ensemble members, where model weights are a function of their overall error.

Similar to Bagging, there are many examples of AdaBoost applications for hydrological prediction. [Solomatine and Shrestha \[2004\]](#) compared various forms of AdaBoost against Bagging in models predicting river flows and found AdaBoost.RT to outperform Bagging. In a later study, the same authors compared the performance of AdaBoosted M5 tree models against ANN models for various applications, including predicting river flows in a catchment; they found higher performance in models that used the AdaBoost.RT algorithm compared to single ANNs [\[Shrestha and Solomatine, 2006\]](#). [Liu et al. \[2014\]](#) used AdaBoost.RT for calibrating process-based rainfall-runoff models, and found improved performance over the single model predictions. [Wu et al. \[2020\]](#) compared boosted ensembles against Bagged ensembles for predicting hourly streamflow and found the combination of AdaBoost (using resampling) and Bayesian model averaging gave the highest performance.

The variant of AdaBoost in this research follows the algorithm AdaBoost.RT proposed by [\[Solomatine and Shrestha, 2004, Shrestha and Solomatine, 2006\]](#). This algorithm has three

hyperparameters. The relative error threshold parameter is selected as the 80th percentile of the residuals of the individual learner and 20 ensemble members are trained. AdaBoost can be performed using either resampling or reweighting [Shrestha and Solomatine, 2006]; resampling is used in this research as it has been found to typically outperform reweighting [Seiffert et al., 2008]. Recently, several studies have independently proposed a modification to the original AdaBoost.RT algorithm by adaptively calculating the relative error threshold value for each new ensemble member [Wang et al., 2019b, Li et al., 2020]. This modification to the algorithm was generally found to be detrimental to the performance of the models in the present research, thus, the static error threshold described in the original algorithm description was used [Solomatine and Shrestha, 2004].

4.5.2.4 Least squares boosting

LSBoost is a variant of gradient boosting, which is an algorithm that involves training an initial model, followed by a sequence of models that are each trained to predict the residuals of the previous model in the sequence. This is in contrast to the AdaBoost method, which uses the model residuals to inform a weighted sampling scheme for subsequent models. The prediction at a given training iteration is calculated by the weighted summation of the already trained model(s) from the previous iterations. For LSBoost weighting is determined by a least-squares loss function; other variants of gradient boosting use a different loss function [Friedman, 2000].

Gradient boosting algorithms have previously been used to improve efficiency and accuracy for hydrological forecasting applications. Ni et al. [2020] use the gradient boosting variant XGBoost, which uses Decision Trees (DTs) as the individual learners, in combination with a Gaussian Mixture Model (GMM) for streamflow forecasting. The GMM is used to cluster streamflow data, and an XGBoost ensemble is fit to each cluster. Clustering streamflow data into distinct subsets for training is sometimes used as an alternative to resampling; its purpose is similar to that of resampling, which is to change the training sample distribution [Wang et al., 2006]. The combination of XGBoost and GMM is found to outperform standalone SVM models. Erdal and Karakurt [2013] developed gradient boosted regression trees and ANNs for predicting daily streamflow, and found gradient boosted ANNs to have higher performance than the regression tree counterparts. Worland et al. [2018] use gradient boosted regression trees to predict annual minimum 7-day streamflow at 224 unregulated sites; performance is

found to be competitive with several other types of data-driven models. [Zhang et al. \[2019\]](#) use the Online XGBoost gradient boosting algorithm for regression tree models to simulate streamflow, and found that it outperformed many other data-driven and lumped hydrological models. [Papacharalampous et al. \[2019\]](#) use gradient boosting with regression trees and linear models, which are compared against several other model types for physically-based hydrological model quantile regression post-processing. Neither of the gradient boosting models outperform the other regression models, and a uniformly weighted ensemble of all other model types typically outperforms any individual model type. These examples of gradient boosting affirm its capability for improving performance compared to the single model comparison, as well as other machine learning models. However, none of these studies use gradient boosting with ANNs as the individual learner. Moreover, these studies do not examine the effects of gradient boosting on model behaviour within the context of the imbalance problem. Therefore, we use LSBoost to study its efficacy for improving high stage performance.

The implementation of LSBoost in this research is unchanged from the original algorithm [\[Friedman, 2000\]](#). The algorithm has two hyperparameters; the learning rate which scales the contribution of each new model and the number of boosts. A learning rate of 1 is used and the number an ensemble size of 20 is used.

4.5.3 Hybrid methods

The resampling and training strategies reviewed above can be combined to further improve model performance on imbalanced data; numerous algorithms have been proposed in the literature that embed resampling schemes in ensemble learning methods. [Galar et al. \[2012\]](#) describes a taxonomy and presents a comprehensive comparison of such algorithms for classification problems. Many of these algorithms effectively present minor improvements or refinements to popular approaches. Alternative to implementing every single unique algorithm for training ensembles, the present research proposes employing a systematic approach to combine preprocessing resampling and ensemble training algorithms, in a modular fashion; such combinations are referred to as 'hybrid methods'. Hybrid methods hope to achieve the benefits of both standalone methods: improved performance on high stage while maintaining good generalisability. Thus, in this research, every permutation of resampling (RUS, ROS, and SMOTER) and ensemble methods (RWB, Bagging, AdaBoost, and LSBoost) is evaluated, re-

sulting in twelve unique hybrid methods. For resampling combinations with RWB ensembles, the resampling is performed once, thus, diversity is only obtained from the initialisation of the ANN. This combination is equivalent to evaluating each resampling technique individually, to provide a basis for comparison with resampling repeated for each ensemble member, as used in the other ensemble-based configurations. For combinations of resampling with Bagging, AdaBoost, and LSBoost, the resampling procedure is performed for training each new ensemble member. One non-intuitive hybrid case is the combination of SMOTER with AdaBoost, because the synthetically generated samples do not have predetermined error weights. A previous study has recommended assigning the initial weight value to synthetic samples [Díez-Pastor et al., 2015a]. However, this research proposes that synthetic sample weights are calculated in the same manner as the synthetic samples (e.g., based on the randomly interpolated point between a sample and a random neighbouring point). Thus, if two samples with relatively high weights are used to generate a synthetic sample, the new sample will have a similar weight.

The hyperparameters for each of the resampling and ensemble method employed in this study are listed in Table 4.4. Every ensemble uses the ANN described in Sect. 4.4.2 as the individual learner. The hyperparameters of the individual learner are kept the same throughout all of the ensemble methods to allow for a fair comparison [Shu and Burn, 2004] (excluding of course the number of epochs, which is determined through validation stop-training).

4.5.4 Model implementation and evaluation

All aspects of this work are implemented in MATLAB 2020a. The Neural Network Toolbox was used to train the baseline ANN models. The resampling and ensemble algorithms used in this research were programmed by the authors and available upon request; the pseudocode for each method is available in Appendix D.2.

Table 4.4: Summary of ensemble methods and hyperparameters.

Type	Complete name	Short form	Hyperparameters
Resampling	Random undersampling	RUS	Rare case threshold (θ_{HS}) = 80th percentile stage
	Random oversampling	ROS	Rare case threshold (θ_{HS}) = 80th percentile stage
	Synthetic minority oversampling technique	SMOTER	Rare case threshold (θ_{HS}) = 80th percentile stage
			Oversampling percentage = 400% Undersampling percentage = 0% K-nearest neighbours = 10
Ensemble	Randomized initial weights and biases	RWB	-
	Bootstrap aggregating	Bagging	Combination weighting: uniform
	Adaptive boosting (for regression using error thresholding)	AdaBoost	Error threshold = 80th percentile of base model error Resampling/reweighting= resampling
	Least squares boosting	LSBoost	Learning rate = 1 Combination weight = least squares

4.5.4.1 Performance assessment

The challenges of training models on imbalanced datasets outlined in Sect. 4.3 and evaluating model performance are one and the same: many traditional performance metrics (e.g., MSE, CE , etc.) are biased towards the most frequent stage values and the metrics are insensitive to changes in high stage accuracy. In fact, despite their widespread use, these metrics are criticised in literature. For example, ANN models for sunspot prediction produced a lower RMSE (equivalent to CE when used on datasets with the same observed mean) compared to conventional models, however were found to have no predictive value [Abrahart et al., 2007]. Similarly, CE values may be misleadingly favourable if there is significant observed seasonality [Ehret and Zehe, 2011]. CE is also associated with the underestimation of peak flows, volume balance errors, and undersized variability [Gupta et al., 2009, Ehret and Zehe, 2011]. Zhan et al. [2019] suggest that CE is sensitive to peak flows due to the square term. This assertion is correct while comparing two samples, however, when datasets are imbalanced, the errors of typical stage overwhelm those of high stage. Ehret and Zehe [2011] evaluate the relationship between phase error and RMSE using triangular hydrographs; their study shows how RMSE is highly sensitive to minor phase errors, however, when a hydrograph has a phase and amplitude error RMSE is much more sensitive to overpredictions compared to underpredictions.

The coefficient of efficiency (CE), commonly known as the Nash-Sutcliffe efficiency, is given by the following formula:

$$CE = 1 - \frac{\sum (q(t) - \hat{q}(t))^2}{\sum (q(t) - \bar{q})^2} \quad (18)$$

where q is the observed stage, $\hat{q}$ is the predicted stage, and $\bar{q}$ is the mean observed stage.

The persistence index (PI) is a measure similar to CE , but instead of normalising the sum of squared error of a model based on the observed variance, it is normalised based on the sum of squared error between the target variable and itself, lagged by the lead time of the forecast model (referred to as the naive model). Thus, the CE and PI range from an optimum value of 1 to $-\infty$, with values of 0 corresponding to models that are indistinguishable from the observed mean and naive models, respectively. Since both models use antecedent input variables with lag times equal to the forecast length, PI is a useful indicator for over-reliance on this input variable, which has been associated with peak stage timing error [de Vos and Rientjes, 2009]. Furthermore, the PI measure overcomes some of the weaknesses of CE , such as a

misleadingly high value for seasonal watersheds. Moreover, PI is effective in identifying when models become over-reliant on autoregressive inputs, as the model predictions will resemble those of the naive model. PI is given by the following formula:

$$PI = 1 - \frac{\sum (q(t) - \hat{q}(t))^2}{\sum (q(t) - q(t-L))^2} \quad (19)$$

where L is the lead time of the forecast.

In order to quantify changes in model performance on high stage, both the CE and PI measures are calculated for typical stage (TS) and high stage (HS) [Crochemore et al., 2015].

The resampling methods are expected to improve the high stage CE at the expense of CE for typical stage, while ensemble methods are expected to produce an outright improvement in model generalisation, reflected by reduced loss in performance between the calibration and test data partitions. Thus, the objective of this research is to find model configurations with improved performance on high stage while maintaining strong performance overall. TS and HS performance metrics are calculated based only on the respective observed stage. For example, the CE for high stage is calculated by:

$$CE_{HS} = 1 - \frac{\sum (q_{HS}(t) - \hat{q}_{HS}(t))^2}{\sum (q_{HS}(t) - \bar{q}_{HS})^2} \quad (20)$$

where q_{HS} is given by:

$$q_{HS} = q \mid q \geq \theta_{HS} \quad (21)$$

The performance for CE_{TS} , PI_{HS} , and PI_{TS} are calculated in the same manner, substituting $q_{TS}(t)$ for $q_{HS}(t)$ in Eq. 21 for HS calculations, and using Eq. 19 in place of Eq. 18 for PI calculations.

4.5.4.2 K-fold cross-validation

The entire available dataset is used for both training and testing by the use of KFCV, a widely used cross-validation method [Hastie et al., 2009, Bennett et al., 2013, Solomatine and Ostfeld, 2008, Snieder et al., 2020]. Ten folds are used in total; eight folds for calibration and two for testing. Of the eight calibration folds, six are used for training, while two are used for early-stopping. When performance is reported as a single value, it refers to the mean model

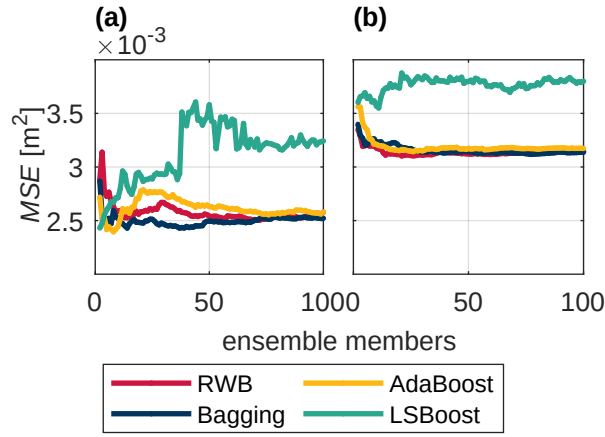


Figure 4.6: Test MSE across ensemble size for RWB (red), Bagging (blue), AdaBoost (yellow), and LSBoost (green) for the Don (a) and Bow River (b).

performance of the respective partition across K-folds. It is important to distinguish between the application of KFCV for evaluation (as used in this research) as opposed to using KFCV for producing ensembles, in which an ensemble of models is trained based on a KFCV data partitioning scheme [Duncan, 2014].

4.6 Results

This section provides a comparison of the performance of each of the methods described throughout Sect. 4.5 applied to the Bow and Don River watersheds, which are described in Sect. 4.4.1. Changes to model performance are typically discussed relative to the individual learner (see Sect. 4.4.2), unless explicit comparisons are specified. First, the results of a grid-search analysis of ensemble size is provided. Next, general overview and comparison of the results are presented, followed by detailed comparison of the resampling and ensemble methods. Finally, the effects that varying the HS threshold and ensemble size have on resampling and high stage performance are evaluated for the Bagging and SMOTER-Bagging models.

Fig. 4.6 illustrates the change in test performance as the ensemble size increases from 2 to 100 for each river. This grid-search is performed only for the base ensemble methods (RWB, Bagging, AdaBoost, and LSBoost) without any resampling. The Bow River results indicates that AdaBoost and LSBoost tend to favour a small ensemble size (2-15 members), whereas the generalisation of RWB and Bagging improves with a larger size (> 20 members). The performance of LSBoost rapidly deteriorates as the ensemble size grows, likely as the effects of overfitting become more pronounced. Similar results are obtained for the Don, except that

RWB, Bagging, and AdaBoost all improve with larger ensemble size, while LSBoost performs worse than all other ensembles, even for small ensemble sizes. Similar to the Bow, a larger ensemble size (>20 members) produces favourable MSE.

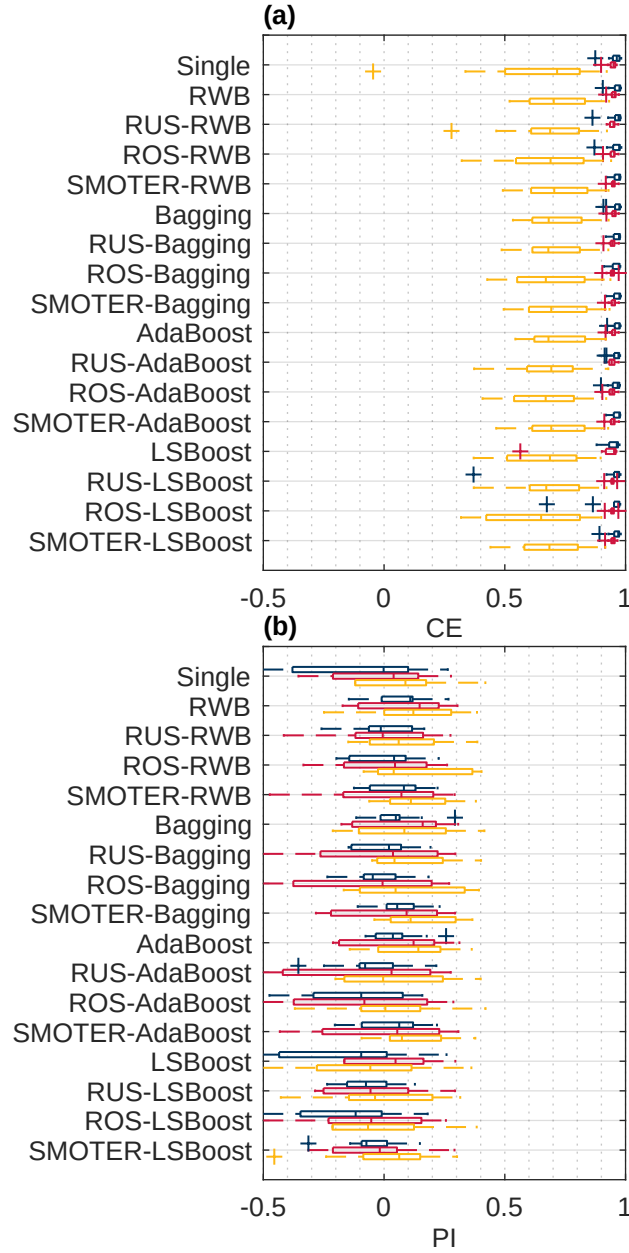


Figure 4.7: Overall (blue), typical stage (red), and high stage (yellow) CE (a) and PI (b) for the Bow River models.

Figs. 4.7 and 4.8 show the CE and PI box-whisker plots for the Bow and Don Rivers, respectively. These figures show the performance of the test dataset, across the K-folds, for each re-sampling, ensemble, and hybrid technique, as well as the individual learner. The performance metrics are calculated for the entire dataset, the HS values, and the TS values. Models with a larger range have more variable performance when evaluated across different subsets of the

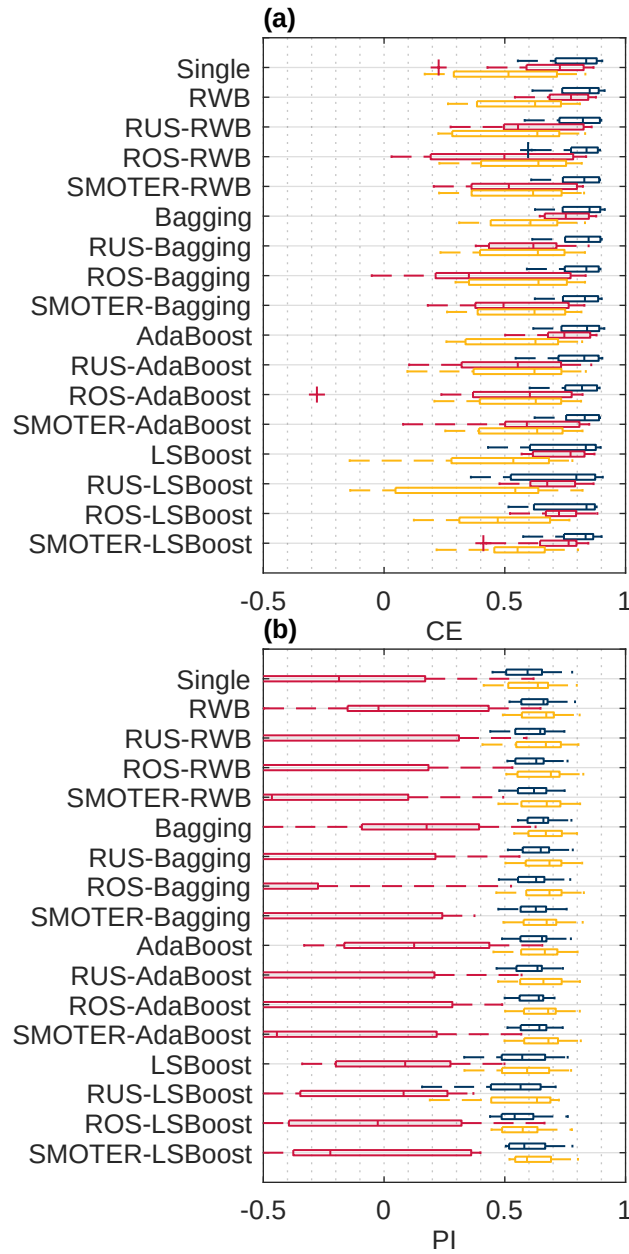


Figure 4.8: Overall (blue), typical stage (red), and high stage (yellow) CE (a) and PI (b) for the Don River models.

available data.

The average performance for each resampling, ensemble, and hybrid methods for the Bow and Don River models are shown in Tables 4.5 and 4.6, respectively, which list the CE and PI for the entire dataset, as well as the TS and the HS datasets. The ensemble results for each KFCV fold were combined using a simple arithmetic average. The results have been separated into different categories: each section starts with the ensemble technique (either RWB, Bagging, AdaBoost, or LSBoost), followed by the three hybrid variations (RUS-, ROS-, or

Table 4.5: Mean CE and PI scores for all, typical, and high stage for the Bow River ensembles; the highest scores are shown in bold and the calibration scores are italicised and enclosed by parentheses.

Label	CE	CE _{TS}	CE _{HS}	PI	PI _{TS}	PI _{HS}
Base model	(0.967) 0.954	(0.954) 0.944	(0.829) 0.617	(0.182) -0.166	(0.111) -0.0593	(0.227) -0.175
RWB	(0.974) 0.962	(0.96) 0.951	(0.865) 0.718	(0.331) 0.0731	(0.229) 0.0856	(0.392) 0.128
RUS-RWB	(0.972) 0.956	(0.954) 0.947	(0.863) 0.68	(0.286) -0.0505	(0.116) -0.013	(0.384) 0.015
ROS-RWB	(0.973) 0.957	(0.955) 0.947	(0.87) 0.681	(0.312) -0.0266	(0.125) 0.00468	(0.418) 0.0454
SMOTER-RWB	(0.974) 0.963	(0.957) 0.948	(0.871) 0.72	(0.329) 0.0524	(0.176) 0.0168	(0.417) 0.139
Bagging	(0.973) 0.961	(0.96) 0.952	(0.86) 0.709	(0.32) 0.0503	(0.234) 0.0886	(0.372) 0.0887
RUS-Bagging	(0.972) 0.961	(0.955) 0.945	(0.867) 0.715	(0.298) 0.00346	(0.119) -0.0403	(0.399) 0.116
ROS-Bagging	(0.973) 0.959	(0.954) 0.943	(0.873) 0.696	(0.312) -0.0374	(0.111) -0.0851	(0.425) 0.0896
SMOTER-Bagging	(0.974) 0.962	(0.957) 0.948	(0.873) 0.719	(0.333) 0.0511	(0.17) 0.018	(0.427) 0.144
AdaBoost	(0.974) 0.963	(0.96) 0.95	(0.865) 0.719	(0.327) 0.0465	(0.22) 0.0488	(0.389) 0.112
RUS-AdaBoost	(0.972) 0.959	(0.954) 0.942	(0.865) 0.693	(0.288) -0.0642	(0.107) -0.105	(0.39) 0.0509
ROS-AdaBoost	(0.972) 0.956	(0.951) 0.942	(0.872) 0.673	(0.291) -0.114	(0.052) -0.109	(0.424) -0.0307
SMOTER-AdaBoost	(0.974) 0.962	(0.957) 0.947	(0.872) 0.714	(0.331) 0.0259	(0.166) -0.00642	(0.425) 0.121
LSBoost	(0.974) 0.948	(0.958) 0.907	(0.869) 0.666	(0.328) -0.504	(0.189) -0.786	(0.403) -0.104
RUS-LSBoost	(0.97) 0.904	(0.952) 0.944	(0.854) 0.364	(0.246) -0.718	(0.0643) -0.0609	(0.35) -0.824
ROS-LSBoost	(0.973) 0.929	(0.952) 0.944	(0.875) 0.517	(0.304) -0.425	(0.0638) -0.0757	(0.435) -0.431
SMOTER-LSBoost	(0.973) 0.958	(0.954) 0.946	(0.868) 0.684	(0.3) -0.0522	(0.117) -0.0255	(0.401) 0.00239

SMOTER-). The calibration (training and validation) performance is indicated in parentheses and italics, followed by the test performance. Comparing both the calibration and test performance is useful since it provides a sense of overfitting, hence, generalisation. For example, an improvement in calibration performance and decrease in test performance suggests that the model has been overfitted. In contrast, improvements to both partitions indicates favourable model generalisation. The best performing model (based on testing performance) have been highlighted in bold text for each performance metric, CE and PI , for both watersheds.

Based on the CE values in Figs. 4.7 - 4.8 and Tables 4.5 - 4.6, the majority of the Bow and Don River models achieve “acceptable” prediction accuracy (as defined by D. N. Moriasi et al. [2007]). Values of CE_{TS} and CE_{HS} are both lower than the CE , which is to be expected, as the stage variance of each subset is lower than that of the set of all stage values. For the Bow River models, the CE and CE_{TS} values are consistently higher than the CE_{HS} ; this is attributable to the high seasonality of the watershed producing a misleadingly high value for CE due to the high variance of stage throughout the year, as discussed in Sect. 4.5.4.1. The CE_{HS} values also have higher variability compared to the overall CE and CE_{TS} , as shown in Fig. 4.7a. In contrast, for the Don River models, the difference in CE , CE_{TS} , and CE_{HS} is less pronounced; whereas the CE (for the entire dataset) is typically higher, as expected, than both the CE_{TS} and CE_{HS} , the difference between CE_{TS} and CE_{HS} is low, as demonstrated in the mean and range of the box-whisker plots in Fig. 4.8a. Unlike the Bow River, the Don River does not exhibit notable seasonality, resulting in smaller difference between the HS and

Table 4.6: Mean CE and PI scores for all, typical, and high stage for the Don River ensembles; the highest scores are shown in bold and the calibration scores are italicised and enclosed by parentheses

Label	CE	CE _{TS}	CE _{HS}	PI	PI _{TS}	PI _{HS}
Base model	(0.86) 0.781	(0.782) 0.664	(0.677) 0.511	(0.716) 0.592	(0.0197) -0.213	(0.74) 0.61
RWB	(0.873) 0.806	(0.814) 0.755	(0.705) 0.572	(0.744) 0.641	(0.165) 0.0944	(0.763) 0.654
RUS-RWB	(0.853) 0.792	(0.638) 0.588	(0.685) 0.555	(0.704) 0.615	(-0.585) -0.63	(0.746) 0.645
ROS-RWB	(0.864) 0.799	(0.629) 0.488	(0.715) 0.584	(0.726) 0.624	(-0.632) -0.991	(0.771) 0.665
SMOTER-RWB	(0.866) 0.795	(0.642) 0.552	(0.715) 0.57	(0.729) 0.618	(-0.573) -0.749	(0.771) 0.656
Bagging	(0.869) 0.808	(0.811) 0.757	(0.696) 0.581	(0.736) 0.65	(0.154) 0.0875	(0.755) 0.663
RUS-Bagging	(0.864) 0.805	(0.676) 0.609	(0.706) 0.585	(0.726) 0.638	(-0.433) -0.502	(0.764) 0.668
ROS-Bagging	(0.858) 0.795	(0.553) 0.271	(0.716) 0.584	(0.712) 0.618	(-1.14) -1.41	(0.771) 0.665
SMOTER-Bagging	(0.865) 0.798	(0.604) 0.526	(0.718) 0.581	(0.729) 0.623	(-0.705) -0.888	(0.774) 0.662
AdaBoost	(0.87) 0.803	(0.807) 0.744	(0.698) 0.567	(0.737) 0.637	(0.136) 0.0393	(0.758) 0.651
RUS-AdaBoost	(0.857) 0.787	(0.658) 0.53	(0.694) 0.553	(0.712) 0.613	(-0.51) -0.888	(0.754) 0.646
ROS-AdaBoost	(0.864) 0.793	(0.604) 0.516	(0.718) 0.575	(0.726) 0.616	(-0.725) -1.07	(0.773) 0.658
SMOTER-AdaBoost	(0.867) 0.801	(0.667) 0.578	(0.715) 0.584	(0.732) 0.629	(-0.46) -0.743	(0.771) 0.665
LSBoost	(0.869) 0.746	(0.813) 0.741	(0.696) 0.446	(0.736) 0.555	(0.169) 0.0719	(0.755) 0.567
RUS-LSBoost	(0.835) 0.715	(0.744) 0.685	(0.625) 0.419	(0.67) 0.513	(-0.128) -0.207	(0.697) 0.548
ROS-LSBoost	(0.871) 0.759	(0.761) 0.716	(0.708) 0.472	(0.738) 0.561	(-0.0738) -0.0931	(0.766) 0.579
SMOTER-LSBoost	(0.871) 0.787	(0.775) 0.695	(0.707) 0.537	(0.74) 0.599	(0.00723) -0.0914	(0.765) 0.62

TS .

Values of PI are typically lower than for CE for both watersheds. The Bow River models obtain PI values centred around 0 (see Fig. 4.7b), indicating that only some of the model configurations perform with greater accuracy than the naive model, meaning that a timing error exists. The box-whisker plots of each ensemble method do not show a clear trend (with respect to the mean value or range) when comparing the PI , PI_{TS} , and PI_{HS} : the mean and range are similar for all variants tested.

The Don River models have positive PI values of approximately 0.6, indicating a lower reliance on autoregressive input variables, when compared to the Bow River. And in contrast to the Bow River, there is a notable difference between the PI metrics: the PI_{TS} has a lower mean value and higher variance (see Fig. 4.8b) than the PI (for the entire dataset) and the PI_{HS} . These lower PI_{TS} are due to the low variability (steadiness) of the Don River TFs (see Fig. 4.4), and thus, the sum of squared error between the naive model and observed stage is also low, reducing the PI value. The low value of PI_{TS} is attributed to the quality of the naive model, not the inaccuracy of the ANN counterpart. Note that PI_{HS} are typically slightly higher than the overall PI : during high stage, there is greater variability, thus the naive model is less accurate, resulting in a higher PI score.

4.6.1 Comparison of resampling and ensemble methods

This section provide a detailed comparison of performance across the different resampling and ensemble methods. As expected, all three resampling methods (RUS, ROS, and SMOTER) typically increase HS performance, often at the expense of TS performance. Based on results shown in Table 4.5, the SMOTER- variations provide the highest performance for HS for the Bow River. SMOTER-RWB CE_{HS} is 0.72, an increase from 0.617 of the individual learner, whereas the SMOTER-Bagging PI_{HS} is 0.144, compared to -0.175 for the individual learner. These indicators suggest that the HS prediction accuracy has improved slightly using these SMOTER variations. The results shown in Table 4.6 for the Don River indicate that the best improvements for HS prediction accuracy is provided by the RUS-Bagging method: the CE_{HS} is 0.585 (an increase from 0.511 of the individual learner), and the PI_{HS} is 0.668 (an increase from 0.61 of the individual learner). While both these metrics show an improvement in HS prediction accuracy for the Don River, the improvements are relatively small compared to the performance improvement for the Bow River. ROS often exhibits poorer performance than SMOTER and RUS. Previous research has noted the tendency for ROS-based models to overfit, due to the high number of duplicate samples [Yap et al., 2014]. RUS, despite using considerable less training data for each individual learner, is not as prone to overfitting as ROS. The RUS-Bagging models consistently outperform the RUS-RWB models; this may be due to the repeated resampling, thus RUS-Bagging uses much more of the original training samples, while RUS-RWB only uses 20% of the original data.

Figures 4.9 and 4.10 show absolute changes in CE and PI relative to the individual learner for the Bow and Don Rivers, respectively, for the entire dataset, the TS and the HS . Performance is colourised in a 2D matrix to facilitate comparisons in performance between each resampling methods across ensemble types and vice versa. From these figures, it is apparent that SMOTER generally produces the largest improvements in HS performance, for both CE and PI , and for both watersheds. The SMOTER methods are also generally the least detrimental to TS performance for both watersheds, as compared to ROS and RUS. Notably, SMOTER is the only resampling method whose performance does not decrease when used in combination with LSBoost. However, the change in performance due to SMOTER is marginal compared to the models without resampling. For the Bow River, the largest improvements between the best models with no resampling and the best models with resampling for CE_{HS}

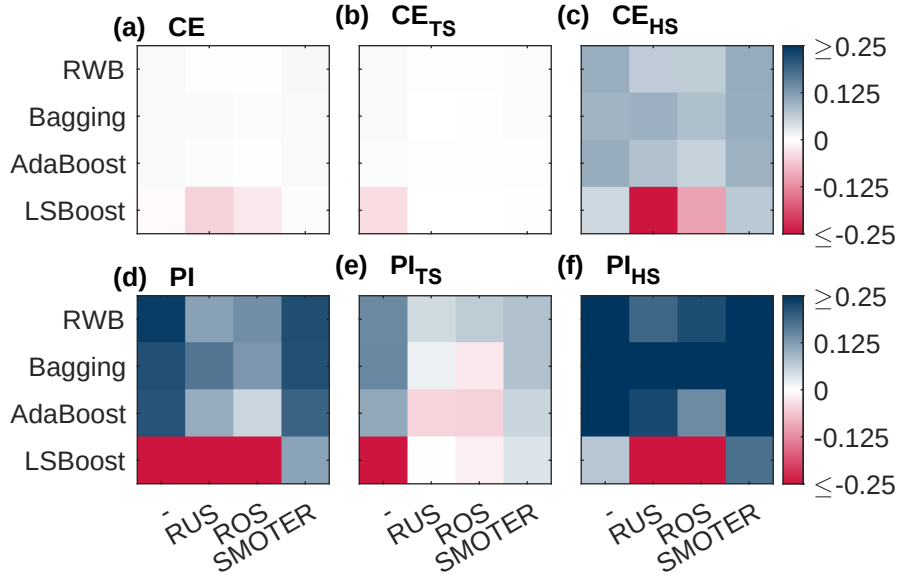


Figure 4.9: Change in (absolute) performance of CE (a), CE_{TS} (b), CE_{HS} (c), PI (d), PI_{TS} (e), PI_{HS} (f) produced by combinations of resampling (listed along the x-axis) and ensemble (listed along the y-axis) methods for the Bow River models.

and PI_{HS} are 0.001 and 0.016, respectively. For the Don River, the same improvements are 0.004 and 0.005, respectively. The remaining resampling methods (RUS and ROS) also generally tend to improve HS performance across the ensemble techniques; however this improvement is not consistent, as is the case with SMOTER, and the decrease in TS performance is also higher. Thus, while SMOTER provides consistent improvements over the non-resampling methods for CE and PI (entire, TS , and HS), RUS and ROS only provide minor improvements to HS performance.

When looking at the resampling methods, the RWB ensembles exhibit competitive performance compared to the other ensemble methods, despite their lower diversity. These ensembles represent a considerable improvement over the individual learner and often achieve higher performance compared to the other, more complex ensemble methods, as shown in Tables 4.5 and 4.6. This suggests that using RWB is useful for improving CE and PI performance (for entire, TS , and HS) as compared to the single, individual learner. For the Bow River, the RWB ensembles improve the PI for each case (PI , PI_{TS} , and PI_{HS}), whereas only improving CE_{HS} . For the Don River models, a notable increase in performance is seen for both CE and PI (entire and HS datasets); however, when combined with the resampling techniques (RUS, ROS, and SMOTER), the TS performance metrics exhibit poorer performance.

The Bagging ensembles also perform well, typically outperforming the RWB counterparts, and

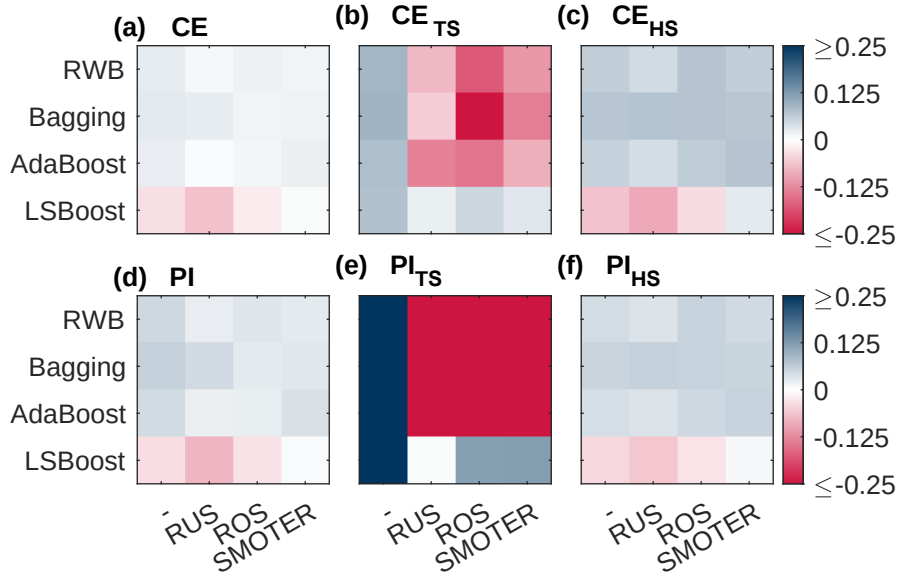


Figure 4.10: Change in (absolute) performance of CE (a), CE_{TS} (b), CE_{HS} (c), PI (d), PI_{TS} (e), PI_{HS} (f) produced by combinations of resampling (listed along the x-axis) and ensemble (listed along the y-axis) methods for the Don River models.

following the same trends described above. This is likely due to their repeated resampling, which achieves greater ensemble diversity compared to the RWB models, for which resampling only occurs once. This result is consistent with a previous comparison of Bagging and boosting [Shu and Burn, 2004]. Like RWB and Bagging, AdaBoost improves model performance compared to the individual learner, but is typically slightly poorer compared to RWB and Bagging, and has higher variability in terms of improvement to model performance across all model types and both watersheds. The RWB, Bagging, and Adaboost models consistently improve TS and HS performance compared to the individual learner regardless of whether they are combined with a resampling strategy.

The LSBoost models have the poorest HS performance out of all the ensemble methods studied. This is consistent across all resampling methods and both watersheds. In contrast, the change in performance for CE_{TS} and PI_{TS} is less detrimental when using LSBoost, suggesting that this method is not well-suited to improve HS performance. The LSBoost models are slightly overfitted, despite utilising the stop-training for calibrating the ANN ensemble members. This is indicated by the degradation in performance between the calibration and test dataset, a change which is larger than that seen in the other ensemble models. This is most noticeable for the RUS-LSBoost models for both the Bow and the Don Rivers, which are more prone to overfitting compared to other models, due to the smaller number of training samples.

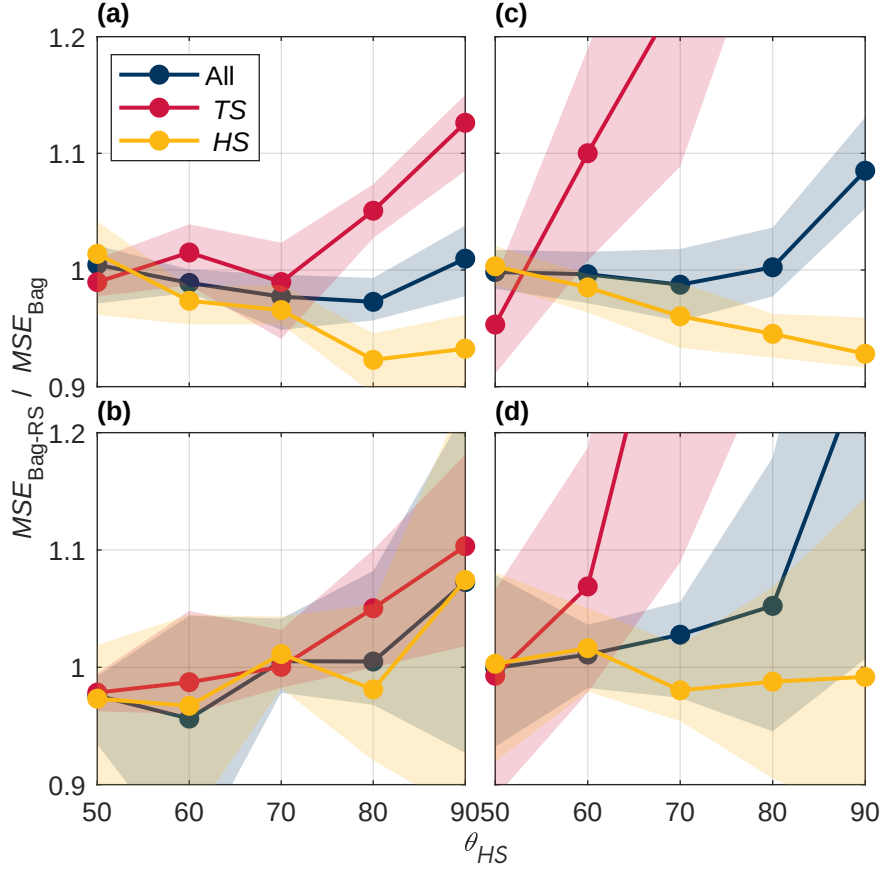


Figure 4.11: MSE ratio between Bagging and SMOTER-Bagging models for the Bow River calibration (a), Bow test (c), Don River calibration (b), and Don test (d) partitions across high stage threshold values ranging from the 50th to 90th percentile stage values.

The CE decreases from 0.97 to 0.902 for the Bow and 0.835 to 0.715 for the Don River; none of the other models that use RUS exhibit such a gap between train and test performance.

The overfitting produced by the boosting methods is consistent with previous research, which finds that boosting is sometimes prone to overfitting on real-world datasets [Vezhnevets and Barinova, 2007]. One reason that the improvements made by the boosting methods (AdaBoost and LSSBoost) are not more substantial may be due to the use of ANNs as individual learners. ANNs typically have more degrees of freedom compared to the decision trees that are most commonly used as individual learners; thus, the additional complexity offered by boosting does little to improve model predictions. Additionally, the boosting methods further increase the effective degrees of freedom of the predictions. Nevertheless, these methods still tend to improve performance over that of the individual learner. Ensembles of less complex models such as regression trees are expected to produce relatively larger improvements when relative to the single model predictions.

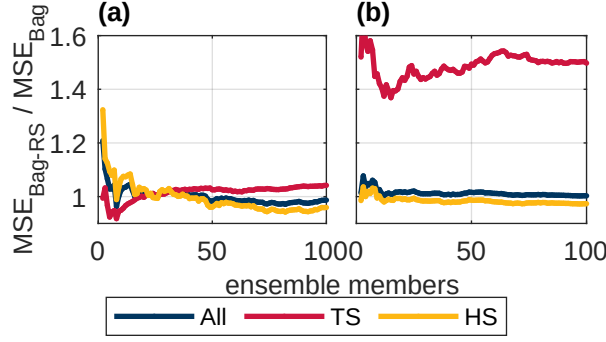


Figure 4.12: Test MSE ratio between Bagging and SMOTER-Bagging models for the Bow (a) and the Don (b) Rivers across ensemble size.

As discussed in Sect. 4.4.1, a fixed threshold is used to distinguish between high and typical stage values, which was set to 80% for the results presented above. Fig. 4.11 shows the effects of the fixed threshold increasing from the 50th to 90th percentile of the stage distribution. These plots show the relative effects of SMOTER-Bagging compared to simple Bagging; these configurations were selected for this comparison since they both exhibited relatively good, consistent performance. A performance ratio greater than 1 indicates that the SMOTER-Bagging model has greater error compared to the Bagging model, 1 indicates that they have the same performance, and less than 1, improved performance. Error is presented for all stage values as well as the TS and HS subsets. The calibration plots illustrate an asymmetric trade-off between HS and TS error. For a given θ_{HS} value, the error ratio of the TS subset increases more than the decline in HS error. More importantly, the improvements in HS performance obtained in calibration are considerably less pronounced in the test dataset, despite a loss in TS performance.

Fig. 4.12 illustrates the effects of varying the ensemble size, thus, number of resampling repetitions, for the SMOTER-Bagging model, relative to the simple Bagging model. The plot shows the relative improvement in HS produced by the SMOTER resampling as the ensemble size increases, reaching a steady value at an ensemble size of approximately 70 for both the Don and Bow systems. This is larger than that required for the simple Bagging model to reach steady performance, shown in Fig. 4.6, indicating that SMOTER requires more resampling than regular resampling with replacement (default in Bagging) in order to reach stable performance. Consistent with observations made from Fig. 4.11, an asymmetric trade-off between typical and high stage performance is noted, illustrated by the disproportionate increase

in error on typical stage, relative to the improvement on high stage.

4.6.2 Limitations and future work

A limitation of this study is the lack of a systematic case-by-case hyperparameter optimisation of the models. The individual learner parameters (e.g. topology, activation function, etc.) were constant across all ensemble members. Likewise, the ensemble hyperparameters were not optimised, but simply tuned using an ad-hoc approach. A systematic approach to hyperparameter optimisation for each model will likely yield improved model performance. However, hyperparameter optimisation on such a scale would be very computationally expensive. Similarly, the selection of the HS threshold may affect CE_{HS} and PI_{HS} performance, and the sensitivity of model performance of this threshold should be explored. This research featured resampling and ensemble methods for improving prediction accuracy across an imbalanced target dataset, i.e., the high stage. Further to imbalanced target data, flood forecasting applications commonly have imbalanced cost; for example, underprediction is typically more costly than overprediction. The use of cost-functions, such as asymmetric weighting applied to underpredictions and overpredictions, for flood forecasting has been shown to reduce underprediction of flooding [Toth, 2016]. Many cost-sensitive ensemble techniques (e.g., Galar et al. [2012]) have yet to be explored in the context of flood forecasting models and should be the focus of future work.

4.7 Conclusion

This research presented the first systematic comparison of the effects of combined resampling and ensemble techniques for improving the accuracy of flow forecasting models, specifically for high stage (infrequent) observations. Methods were applied to two Canadian watersheds, the Bow River in Alberta, and the Don River, in Ontario. This research attempts to address the widespread problem of poor performance on high stage when using data-driven approaches such as ANNs. Improving performance on high stage is essential for model applications such as early flood warning systems. Three resampling and four ensemble techniques are implemented as part of ANN flow forecasting models, for both watersheds. These methods are assessed independently and systematically combined in hybrid approaches, as to assess their efficacy for improving high stage performance. A major contribution of this paper is the comprehensive evaluation of these hybrid methods, most of which are the first instances in the water

resources field. While methodologies for these combination methods are available in existing machine learning literature, our proposed implementation of SMOTER-AdaBoost is a novel improvement. Results demonstrate that resampling methods, when embedded in ensemble algorithms, generally only produces a small improvement in high stage performance, based on CE and PI ; the SMOTER variation provided the most consistent improvements. An asymmetric trade-off between typical and high stage performance is observed, in which improved high stage performance produced disproportionately worse typical flow performance. Such a trade-off should be carefully considered while implementing these methods. Further research on this topic may explore the combination of cost-sensitive approaches with ensemble methods, which would allow for more aggressive penalisation of poor accuracy on high stage.

5 A diversity centric strategy for the selection of spatio-temporal training data for LSTM-based streamflow forecasting

5.1 Preface

Work on a series ‘large-sample’ studies, that is, including several hundred basins, began as a result of a collaboration between York University and Ontario’s Ministry of the Environment Conservation and Parks. My role on the project consisted of developing flood forecasting models for all basins (with available data) located along the Canadian shoreline of Lake Ontario. Following the project, the study area was expanded to include all of Southern Ontario, which permitted the study of spatiotemporal trends in model behaviour and performance. This early work was presented at the 2021 AGU General Assembly (virtual presentation). The publication of the HYSETS database in 2020, which contains clean hydroclimatic data for all of North America, allowed for the expansion of the study area to include all of Canada. The Canada-wide research was presented at several conferences, including the EGU General Assembly (2022, 2023) and the Canadian Water Resources Annual Conference (2022). The first of several planned manuscripts (which are beyond the scope of this dissertation) was submitted to Hydrology and Earth Systems Science as follows:

Snieder, E. and Khan, U. T. (2024, preprint). A diversity centric strategy for the selection of spatio-temporal training data for LSTM-based streamflow forecasting, Hydrology and Earth System Sciences Discussions, 2024, 1–23, <https://doi.org/10.5194/hess-2024-169>

The conference citations are as follows:

Snieder, E., Khan, U. T. (2021). Large-scale analysis of machine learning based flow forecasting models for Southern Ontario. Poster presentation at the AGU General Assembly. Hybrid Poster.

Snieder, E., Khan, U. T. (2022). Large-scale evaluation of temporal trends in ANN behaviour for daily flow forecasts in Canadian catchments. Podium presentation at the EGU General Assembly. Vienna, Austria.

Snieder, E., Khan, U. T. (2022). An ANN-based Canada-wide flood forecasting system. Podium presentation at the Canadian Water Resources Association National Conference. Canmore,

Canada.

Snieder, E., Khan, U. T. (2023). A large sample study of the effects of upstream hydrometeorological input features for LSTM-based daily flow forecasting in Canadian catchments. Poster presentation at the EGU General Assembly. Vienna, Austria.

Khan, U. T. Snieder, E. (2023). Cluster-based hyperparameter optimisation for LSTM-based flow forecasting in Canadian catchments. Podium presentation at the EGU General Assembly. Vienna, Austria.

Contributions to this chapter are as follows: **E. Snieder:** conceptualisation; data curation; formal analysis; investigation; methodology; visualisation; writing - original draft. **U. T. Khan:** conceptualisation; funding acquisition; supervision; writing - editing, revisions.

5.2 Abstract

Deep learning models are increasingly being applied to streamflow forecasting problems. Their success is in part attributed to the large and hydrologically diverse datasets on which they are trained. However, common data selection methods fail to explicitly account for hydrological diversity contained within training data. In this research, clustering is applied to temporal and spatial datasets in order to better understand the importance of hydrological diversity within regional training datasets. This study presents a novel resampling approach to creating hydrologically diverse datasets. First, the undersampling procedure is used to under-sample temporal data, and is used to show how the amount of temporal data needed to train models can be halved without any loss in performance. Next, it is applied to reduce the number of basins in the training dataset. While basins cannot be omitted from training without some loss in performance, we show how hydrologically dissimilar basins are highly beneficial to model performance. This is shown empirically for Canadian basins; models trained to sets of basins separated by thousands of kilometres outperform models trained to localised clusters. We strongly recommend an approach to training data selection that encourages a broad representation of diverse hydrological processes.

5.3 Introduction

Floods constitute a major threat to populations and infrastructure and are projected to increase in severity due to climate change and urbanisation. Flood early warning systems (FEWS), which rely on models that predict streamflow, provide advanced notice of flood risk and are considered amongst the best ways to mitigate flood damage. Many Canadian communities lack any sort of FEWS, making them vulnerable to flood damage. Over the past 3 decades, machine learning (ML) models have been increasingly applied for streamflow prediction and represent significant potential for improving the accuracy and coverage of FEWS in flood prone regions. Recently, several large sample studies have shown that ML models can consistently outperform traditional, physics-based hydrological models [Mai et al., 2022, Arsenault et al., 2023a, Kratzert et al., 2019b], which underscores their proficiency for FEWS.

ML model development has typically followed the same format as physics-based models, in that a single model is parameterised and calibrated on an individual basin, which is referred to as a locally trained model. The work by Kratzert et al. [2019b] demonstrated that the ac-

curacy of ML models can be improved by training a model to a set of basins, rather than an individual basin, which is referred to as a regionally trained model. Regional training relies on deep learning architectures such as long-short term memory networks (LSTMs), which have recently surged in popularity for streamflow forecasting applications and are considered to be state-of-the-art [Fang et al., 2022]. Recent advances in regional learning have focused on improvements to model architectures [Nevo et al., 2022, Giriagama et al., 2022], and benchmarking against traditional physics-based models [Lees et al., 2021, Arsenault et al., 2023a].

Broadly speaking, physics-based and ML hydrological models benefit from diverse training data, which improves their performance on future, unseen conditions. Locally trained models are only provided with temporally diverse data at a single point in space. In contrast, regionally trained models have been empirically shown to outperform locally trained models in several works [Kratzert et al., 2019b, Zhang et al., 2022], and their success is in part attributable to the spatial and temporal hydrological diversity contained in the multi-basin datasets on which they are trained [Kratzert et al., 2019b].

However, there is currently little guidance on optimum basin (spatial) selection of training data for these models. Often times, models are trained to complete large-sample datasets (e.g., CAMELS US). However, with increasingly large, global hydrological datasets, it is not always practical or feasible to train to all available data, especially when conducting computationally expensive tasks such as hyperparameter selection, which is required to achieve optimum model performance, or creating multimodel ensembles. Therefore, there is a need for improved guidance on efficient methods for training data selection, to maximise model performance and generalisation. Many of the deep learning advances in hydrology have utilised well-established large-sample basins (e.g., CAMELS-US, CAMELS-AUS). Canada is a country with a diverse hydrological landscape, characterised by coastal regions, mountain, urban, and exposed rock in the form of the Canadian shield. The effectiveness of regionally trained models has not yet been well established on highly diverse Canadian basins.

When training a model to predict streamflows in some region of interest, the goal is to select the most relevant spatial and temporal data, while avoiding data that has either no impact or a negative impact on model performance. Unsupervised clustering has been used in previous studies as a data-driven approach to identify spatial and temporal diversity within a

training dataset [Toth, 2009, Kratzert et al., 2024]. The application of clustering as a means to identify spatial and temporal diversity is in itself nothing new. Many studies have applied clustering to spatial and temporal data as a means to quantify hydrological diversity. However, the treatment of hydrological diversity generally follows one of two approaches: either, it is used to generate hydrologically diverse datasets, or datasets with homogeneous hydrological conditions. The former, aims to generalise models to a wide range of conditions, promoting balanced performance and good generalisation, while the latter aims to simplify the learning problem, improving performance in similar conditions. Both approaches have been used successfully for temporal streamflow clustering. Anctil and Lauzon [2004a] apply an SOM to streamflow data in a single basin to create a training dataset with a balanced representation of diverse hydrological states. In contrast, Toth [2009] use an SOM to classify streamflow into homogeneous subsets, on which individual models are trained and combined in a modular format. Their approach is found to improve overall prediction accuracy, which can be attributed to the error diversity of the collection of trained models. [Snieder et al., 2021] partition streamflows into typical streamflows and high streamflows, in order to undersample typical streamflows and oversample high streamflows, which is found to improve performance on the latter, which is desirable for FEWS applications. The same motivation has led to numerous applications of clustering on basins, particularly in regional training schemes. For example, Gauch et al. [2021] showed that implicitly increasing hydrological diversity of regional training datasets, by iteratively increasing the number of basins, as well as the amount of data in each basin, improves model generalisation. However, the study does not explicitly quantify hydrological diversity, in part due to the absence of a widely agreed upon metric for hydrological similarity [Oudin et al., 2010]. However, other studies have used clustering to estimate hydrological diversity, such that basin selection can explicitly account for hydrological diversity. These cases tend to use some form of clustering (either supervised or unsupervised) to quantify hydrological diversity within training data and the effects it has on model generalisation. Zhang et al. [2022] clustered a set of 35 mountainous basins in China based on hydro-climatic attributes, finding that a model trained to all available basin typically outperformed those trained to individual clusters. Hashemi et al. [2022] applied a similar approach by clustering basins into hydrological regions based on the ranges of indices derived from streamflow, precipitation, and temperature. As done in Zhang et al. [2022], their study compared locally and globally trained models, finding only minor differences in the performance between the

two. A common problem in comparing global and locally trained models is that these comparisons typically do not control for sample size. As a result, the improved performance of the global model can be impacted by the regularisation effect on the sample size. In other words, deep learning models trained to small datasets may be overfitted and thus, poorly generalised. Fang et al. [2022] accounts for this potential issue. Their study groups basins based on ‘ecoregions’, and evaluates the effects of additional training basins at three similarity intervals. Their study showed that counterintuitively, ‘far’ or ‘dissimilar’ basins amongst the training dataset often produced greater improvements in model performance when compared to the inclusion of ‘close’ basins. They speculate that distant basins provide an implicit regularisation effect, in other words, prevent overfitting. Fang et al. [2022] call for further investigation into the effect that hydrological diversity has on model generalisation and underline the need for a systematic approach. Kratzert et al. [2024] characterise hydrological diversity by applying K-Means clustering to basin attributes, finding that models trained to basins with similar hydrological characteristics outperform randomly sampled basin sets of the same sizes. However, in every case, they show that LSTMs trained to hundreds of basins outperform those trained to smaller subsets. They also demonstrate how regional learning improves performance on extreme events, thus for FEWS, as the training datasets contain a higher number of extreme events, spread across all basins. Many of these studies assume that similar basins are most useful to one another in the context of regional learning. We challenge this assumption, and seek to determine to what extent hydrologically similar data is beneficial for training.

The objective of this study is to study the effect that the formation of hydrologically diverse training datasets has on model performance and generalisation. Hydrological diversity is quantified using clustering, which is applied separately to streamflow (temporal) and basins (spatial). This topic is analysed throughout two experiments. In the first experiment, we evaluate the effects of removing non-diverse streamflow data from regional training datasets. The latter test is repeated, but by undersampling non-diverse basins, instead of streamflow values, from a larger subset. In the second experiment, we compare the effects of adding ‘similar’ and ‘dissimilar’ basins to a training dataset for some region of interest. The purpose of this second experiment is to compare the contribution of additional basins to model generalisation, with respect to their hydrological similarity to the evaluation set.

While numerous studies have applied clustering to streamflow and basins, to the extent of the

knowledge of the authors, the use of clustering to explicitly create spatial and temporally diverse training datasets is a novel approach. The outcome of these experiments has the potential to improve methods for the creating of training datasets for regionally trained models. This topic is investigated on sequence-to-sequence (Seq2Seq) LSTMs for daily streamflow forecasts of 1–3 days in Canadian basins. Basins are sampled from across Canada and models are trained using historic hydrometeorological data from the past 36 years.

5.4 Methods and Data

5.4.1 Input and target variables

This study uses data retrieved from the HYSETS dataset, which contains hydrometeorological data for over 14,000 basins across North America [Arsenault et al., 2020]. The target variable, future streamflow, is predicted using dynamic and static input features. The models in this study are autoregressive (AR), meaning past streamflow at the target gauge is used as one of the input features [Nearing et al., 2022]. Additional dynamic features include daily basin-averaged minimum temperature, maximum temperature, precipitation, and snow water equivalent (SWE), which are listed in Table 5.2. The static basin attributes, which are summarised in Table 5.1 in the appendix, allow the model to transfer learned information between basins. Some of the static attributes are included in the HYSETS database, while additional attributes are calculated based on the dynamic timeseries, which generally follow those included in the CAMELS dataset [Addor et al., 2017]. The additional attributes are calculated based on the timeseries’ data already contained within HYSETS and do not rely on any external databases. Finally, the input feature set also contains one-hot encoded basin labels [Lees et al., 2022], which enables the model to distinguish between streamflows in different basins. Static basin attributes are used both as input features for the streamflow forecasting model and in the basin clustering method, which are described in Sections 5.4.3 and 5.4.3, respectively.

5.4.2 Basin selection

This study only considers Canadian basins from the HYSETS database. Basins are removed if they have less than 80% data availability within any of the training, validation, and testing periods, which span a total of 36 years from Oct. to Sep. of 1982-1994, 1994-2006 and 2006-2018, respectively. The training partition is used to train the models, validation for fine-

tuning LSTM hyperparameters, and the test partition is used to calculate model performance. While records in some basins exist prior to 1982, it is imperative that the data used to train and evaluate basins be from the same time period and be of a similar size. Including records from before 1982 results in fewer basins to choose from, which tends to reduce the hydrological variability of the basin set. Next, some basins are removed, due to missing static attributes. These criteria produce a set of approximately 2000 basins, with highly variable attributes, according to Table 5.1.

Table 5.1: Basin static attribute statistics (mean, standard deviation, 10th, 50th, and 90th percentile) calculated for 2363 Canadian catchments.

	mean	std	10%	50%	90%
elevation (m)	703	740	93.5	380	1.89e+03
slope (deg)	6.12	6.52	0.596	3.02	16
gravelius (-)	1.69	0.359	1.32	1.61	2.17
aspect (deg)	172	89	59.2	162	302
land use forest (frac)	0.435	0.294	0.0269	0.45	0.826
land use grass (frac)	0.0843	0.143	0.0018	0.027	0.24
land use wetland (frac)	0.0448	0.0892	0	0.008	0.145
land use water (frac)	0.0142	0.0315	0	0.0036	0.0349
land use urban (frac)	0.0945	0.167	0.0011	0.0464	0.219
land use shrubs (frac)	0.0987	0.153	0.0001	0.0336	0.293
land use crops (frac)	0.223	0.284	0	0.0754	0.745
land use snow ice (frac)	0.00485	0.0325	0	0	0.0001
permeability (log; m2)	-13.9	1.18	-15.4	-14	-12.3
porosity (frac)	0.123	0.0641	0.0286	0.124	0.205
precip mean (mm/d)	2.74	1.19	1.29	2.71	3.93
precip high (mm/d)	13.7	5.94	6.44	13.6	19.6
precip low (mm/d)	1	0	1	1	1
precip mean (mm/mo)	83.2	36.2	38.6	82.5	120
precip mean (mm/y)	998	435	464	990	1.43e+03
precip high freq (d/yr)	3.93	0.528	3.21	4.06	4.46
precip high dur (d)	1.24	0.131	1.12	1.2	1.42
precip low freq (d/yr)	48.6	5.82	41.2	49	55.7
precip low dur (d)	5.5	2.78	3.53	4.65	8.26
baseflow index (-)	0.646	0.168	0.416	0.665	0.846
q mean (mm/d)	1.3	1.38	0.151	0.98	2.56
q high (mm/d)	11.7	12.5	1.36	8.82	23
q low (mm/d)	0.261	0.277	0.0302	0.196	0.512
q high freq (d/yr)	95.8	122	1	49	268
q high dur (d)	2.63	4.13	1	1.74	5.12
q low dur (d)	26.8	44.9	3.56	15.4	56.8
q zero freq (d/yr)	351	1.43e+03	0	0	698
q 95 (mm/d)	4.43	4.52	0.429	3.41	8.81
q 5 (mm/d)	0.162	0.316	0	0.0722	0.389
q adf (-)	-12.4	6.99	-19.4	-11.6	-5.16
q seasonality (-)	0.272	0.231	0.0543	0.192	0.65
q var (mm2/d2)	9.5	27.3	0.129	3.5	19
runoff ratio (-)	0.45	0.374	0.101	0.371	0.842
tmean seasonality (-)	0.782	0.0633	0.724	0.795	0.831
tmean alltime (c)	9.95	5.32	3.77	9.81	17.5
tmax annual mean (c)	34.6	3.05	30.8	35	38.1
tmin annual mean (c)	-20.8	9.73	-33	-21.3	-7.76

Table 5.2: Dynamic features used as model inputs.

Dynamic attributes		Source	Units
Discharge		Water Survey of Canada	mm (basin normal)
Precipitation	Environment and Climate Change Canada (Quality controlled)		mm
Temperature	Environment and Climate Change Canada (Quality controlled)		degC
SWE		ERA5	mm

5.4.3 Sequence-to-sequence LSTM models

LSTM models, with a Seq2Seq architecture [Cho et al., 2014], are used to generate forecasts at a daily resolution, at multiple lead times. Seq2Seq models are composed of an encoder and decoder; the encoder transforms an input sequence into a fixed length context vector, which is provided to the decoder, which outputs predictions. Recently, several studies have demonstrated the aptness of Seq2Seq models for predicting runoff at multiple lead times [Xiang et al., 2020, Giriagama et al., 2022, Zhang et al., 2022].

The Seq2Seq models in this study use hyperparameter values that are common for LSTM rainfall runoff models. The models in this study use a hidden layer size of 128 cells, a dropout rate of 0.2, a batch size of 32, and Adam optimisation with a decaying learning rate of 0.001 to 0.0001 across a total of 80 epochs. Input and output sequences of 7 and 3 days were used, respectively. While an input sequence of 365 days is commonly used for streamflow prediction [Kratzert et al., 2019b, Arsenault et al., 2023b], Gauch et al. [2021] noted that small sequences are better suited to small basin sets, and have been used in AR models [Nevo et al., 2022].

Typically, for regional training, the error terms of individual basins are normalised based on streamflow variance of that basin. Using the typical variance-based regularisation applied to the cost function produces a relative increase in weight applied to low variance basins. While this works well for non-AR models, AR models have a tendency to develop an over-reliance on recent streamflow observations, which can manifest in a positive timing error [Snieder, 2019]. The resulting models may be barely distinguishable from the naive model (i.e., the most recent streamflow observation). Highly seasonal, naturalised basins are most prone to this problem, as streamflow tends to change gradually with time, thus a model that outputs recent streamflow observations might be mistakenly seen as accurate. Specialised performance metrics such as the Persistence Index (PI) are often used to identify this problem in real-time forecasting applications [Nevo et al., 2022]. This is simply because the PI normalises error rel-

ative to the naive model, with a PI less than 0 corresponding to a non-informative forecast relative to real-time observations. The same is not the case for widely used Nash Sutcliffe Efficiency (NSE), which can be misleadingly high in the same cases [Knoben et al., 2019]. For this reason, we propose that basin persistence (i.e., mean squared deviation between observations at the current time and forecast time) be used to regularise the cost function for regionally trained models. Instead of placing more weight on low variance basins, persistence-based regularisation places more weight on basins that have low error between recent and future streamflow values. Failure to do so results in models that are not adequately trained in those basins, and produce non-informative forecasts (relative to the naive model).

In this study, basins are normalised using the formulation proposed in [Kratzert et al., 2019b] for NSE*, but substituting the basin variance for the persistence corresponding to the forecast lead time. The persistence-based cost function PI* is given by:

$$\text{PI}^* = \frac{1}{B} \sum_{b=1}^B \sum_{t=1}^T \frac{\sum (q_t - \hat{q}_t)^2}{\sum (p_b + \epsilon)^2} \quad (22)$$

in which q_t is the observed streamflow, $\hat{q}_t$ is the predicted streamflow, ϵ is a constant (0.1) that prevents the function from exploding to negative infinity [Kratzert et al., 2019b], and p_b corresponds to the persistence of an individual basin b in a set B basins, given by:

$$p_b = \sqrt{\sum_{t=1}^T (q_t - q_{(t-L)})^2} \quad (23)$$

in which L is the forecast lead time of $\hat{q}_t$.

5.4.4 Performance metrics

Models are evaluated using two performance metrics: NSE and PI. The NSE, given in Eqn. 24, is amongst the most widely used metrics for hydrological models and effectively normalises the mean squared model error based on streamflow variance. The PI, which is given in Eqn. 25 and used in the basin regularisation function described above, is a similar metric but instead of normalising squared residuals using the mean, it normalises forecasts based on the squared error between the streamflow at the current and forecast timesteps [Kitanidis and

Bras, 1980].

$$\text{NSE} = 1 - \frac{\sum_{t=1}^T (q_t - \hat{q}_t)^2}{\sum_{t=1}^T (q_t - \bar{q})^2} \quad (24)$$

$$\text{PI} = 1 - \frac{\sum_{t=1}^T (q_t - \hat{q}_t)^2}{\sum_{t=1}^T (q_t - q_{t-L})^2} \quad (25)$$

where $\hat{q}_t$ to the predicted streamflow, and $\bar{q}$ to the mean observed streamflow. q_{t-L} is the observed streamflow, shifted by the lead time L of the forecast such that it represents the real-time observable streamflow in an operational context. Both metrics range between $-\infty$ and 1, with 1 being perfect and values less than 0 indicating performance worse than each respective baseline.

5.4.5 Clustering

Clustering is a simple yet effective way to identify hydrologically diverse data for training streamflow forecasting models. This study uses clustering to identify two forms of hydrological diversity. First, it is applied to streamflow records of individual basins, to identify diverse streamflow conditions. Second, it is applied to static basin attributes, to identify basins with diverse hydrometeorological attributes. Note that both methods are independent of one another. This study uses the constrained K-means clustering algorithm [Bennett et al., 2000], which allows for the specification of a minimum cluster size. This avoids a problem that occurs with clustering streamflow, which is that infrequent flood streamflows typically produce a very small cluster, constraining the number of samples that are available when drawing an even number of samples from each cluster [Toth, 2009].

The first application of clustering is to identify hydrologically diverse streamflows. Previous studies have applied clustering to the input vectors of ML models [Anctil and Lauzon, 2004a, Abrahart and See, 2000]. However, such approaches do not guarantee that streamflow is the main variable by which clusters are discriminated. For that reason, we engineer a feature set for clustering streamflow based solely on the target streamflow data, which encourages diverse streamflow conditions between clusters. The engineered feature set includes streamflow (q_t), two streamflow gradient features (given as $(q_{t-3} - q_t)/3$ and $q_{t-1} - q_t$), and two day of year

features (given as $\sin(2\pi t/365)$ and $\cos(2\pi t/365)$ where t is the day of year 1,...,365). The sin and cos functions ensure that the day of year features are continuously changing from December to January. Features such as the streamflow gradient, encourages the representation of rising and falling limbs within the clusters - which would be indistinguishable based solely on streamflow value.

The second application is on basins, which are clustered based on their static attributes. Due to the large number of features (39) and collinearity between features, principal component analysis (PCA) is used to reduce the feature set to 8 principal components. Names and statistics of the static attribute set are provided in Table S1 in the supplementary information.

Both clustering applications are used to inform a simple resampling procedure that aims to maximise hydrological diversity. The cluster-based undersampling (CUS) procedure is as follows. Given N training examples (either streamflow samples or basins):

1. Select an undersampling rate ϕ as a fraction of N and a number of clusters K
2. Cluster records into K clusters with a minimum cluster size of $\phi N/K$
3. Sort samples based on distance to the cluster centroid
4. Select samples 1 to $\phi N/K$ from each cluster to form the training dataset

Each undersampling strategy is illustrated in Fig. 5.1. Subplots (a), (c), and (e) illustrate the raw streamflow timeseries (a), clustered streamflow (c), and undersampled streamflow (f). Note how the timeseries in (f) contains fewer typical streamflows and proportionally more high streamflows, compared to the continuous record in (a) and (c). Similarly, subplots (b), (d), and (f) show Canadian basins (a), clustered basins (d), and diverse, undersampled basins (f).

CUS applied to streamflow and basins are denoted as CUS_Q and CUS_B , respectively. Selecting an equal number of examples (streamflow or basins) from each cluster results in a balanced variety of hydrological conditions within the training set. There are several reasons why such a training dataset is desirable. First, balanced hydrological conditions encourage balanced performance across different streamflows, or basins. Models trained to imbalanced datasets, such as streamflow records in which low streamflows drastically outnumber high

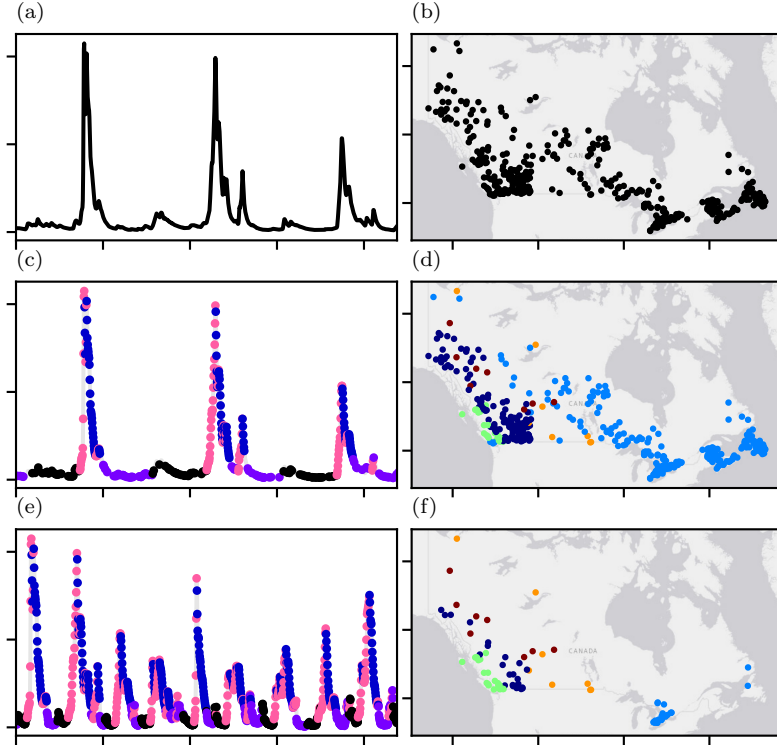


Figure 5.1: Left column, from top to bottom: unclustered (a), clustered (c), and undersampled streamflow (e) for basin 01AD003 (HYDAT ID; located along St. Francis River in New Brunswick). Right column, from top to bottom: unclustered (b), clustered (d), and undersampled basins (f). Cluster colours are arbitrary and there is no connection between temporal (a, c, e), and spatial (b, d, f) cluster colours. World Gray Canvas basemap in (b, d, f) provided by ESRI.

streamflows, may be biased towards low streamflow conditions (Snieder et al. [2021]). The same reasoning applies to basin selection. A regionally trained model may be biased towards areas with dense spatial coverage. By selecting an equal number of each ‘type’ of basin, we encourage balanced spread of hydrological characteristics in the training basin dataset, which translates to good generalisation across a broader range of basins.

Due to the large and diverse feature set, feature importance is calculated to interpret the dominant basin attributes that distinguish clusters. Since K-means does not inherently quantify feature importance, a random forest (RF) classifier is used as a surrogate for approximating feature importance. RFs contain an intrinsic importance metric that is commonly used in hydrology [Tyralis et al., 2019b]. An RF with 256 estimators, and a max depth of 6 is used. Since the RF is simply fitting the outcome of the constrained K-means clustering, the RF is

expected to achieve near perfect accuracy, without the need for hyperparameter tuning.

5.4.6 Experiments

5.4.6.1 Experiment 1: evaluating streamflow and basin redundancy in training datasets

In this experiment, CUS is applied to streamflow (experiment 1a) and separately applied to basins (1b). These experiments are designed to determine the extent to which non-diverse (i.e., redundant) data can be removed from training datasets, without any loss in model performance. The CUS generated datasets are compared with random undersampled (RUS) datasets, which consist of ϕN samples (sampled without replacement).

To evaluate CUS_Q (experiment 1a), a set of 64 randomly sampled basins is established. CUS_Q is applied to each basin individually, then merged to form the training dataset. Several resampling configurations are considered, including ϕ values of 0.25 to 0.50, and K values 6 and 12. These configurations are compared against two baseline models, which are trained to (1) the entire dataset and (2) an RUS, which are also undersampled at 0.25 and 0.50 to match the sample size of the CUS_Q datasets. The parameters for each training configuration are listed in Table 5.3.

The framework outlined above is replicated to evaluate the effects of spatial undersampling (experiment 1b). Beginning this time with a set of 128 randomly sampled basins, subsets of basins sampled for varying numbers of clusters. Basin subsets comparisons are made against a baseline model that is trained to the entire set of basins, and RUS subsets. The purpose of this experiment is to determine whether clustering can effectively be used to identify a subset of basins that is sufficiently hydrologically diverse such that it can be used to train a model capable of generalisability on the complete set. In this experiment, the basins are trained to the entire streamflow records (i.e., no temporal resampling). The parameters for each training configuration are listed in Table 5.4.

5.4.6.2 Experiment 2: cross-comparison of 2 clusters of basins

The next experiment is designed to determine to what extent hydrologically dissimilar basins are useful to one another for model training. In experiment 2a, basins are divided into two clusters (which are referred to as C0 and C1). The reasoning behind 2 clusters is to max-

Table 5.3: Streamflow resampling configurations in Experiment 1a. Labels denote the resampling type, number of clusters (K ; for CUS), and resampling range (ϕ).

label (K, ϕ)	unique basins	samples (per basin)	total samples (thousands)
baseline	64	4380	280
CUS _Q ($K=6, \phi=0.25$)	64	1095	70
CUS _Q ($K=6, \phi=0.50$)	64	2190	140
CUS _Q ($K=12, \phi=0.50$)	64	2190	140
RUS _Q ($\phi=0.50$)	64	2190	140

Table 5.4: Basin resampling configurations in Experiment 1b. Labels denote the resampling type, number of clusters (K ; for CUS_B), and resampling range (ϕ).

label (K, ϕ)	unique basins	samples (per basin)	total samples (thousands)
baseline	128	4380	560
CUS _B ($K=2, \phi=0.50$)	64	4380	280
CUS _B ($K=8, \phi=0.50$)	64	4380	280
CUS _B ($K=32, \phi=0.50$)	96	4380	420
RUS _B ($\phi=0.50$)	64	4380	280
RUS _B ($\phi=0.75$)	96	4380	420

imise the hydrological dissimilarity between basins in each cluster (based on the static basin attributes). In the first experiment, for each cluster, a baseline model is trained to 32 basins that belong to that cluster. Next, we compare the effects of adding 32 similar basins (labelled as ‘+similar’), with adding 32 dissimilar basins to the training set (i.e., from the other cluster; labelled as ‘+dissimilar’). In all cases, only the original 32 basins are evaluated (those used to train the baseline model); the performance of the additional training basins is not reported. This produces five unique training sets: 32 basins in cluster 0, 64 basins in cluster 0, 32 basins in cluster 1, 64 basins in cluster 1, and 32 basins in each cluster 0 and 1.

Next in experiment 2b, the configuration from 2a is repeated, but with the addition of cluster-based streamflow undersampling. This provides a comparison point between models trained in 32 basins without CUS_B and models trained on 64 basins with CUS_B, as both configurations have the same number of samples. The 32 basin configurations have greater temporal representation within the evaluation basins, while the 64 basin CUS_B configurations have greater spatial diversity, at the expense of temporal data. These comparisons reveal which is more useful to model generalisation: temporal data from within the subject basin, or data from outside the basin. One distinction in this experimental configuration is that it does not use one-hot-encoded basin labels, as many of the test basins are not included in the training dataset.

The labels are removed to ensure that the model does not develop any dependencies on them, since they are not available for the test basins.

Table 5.5: LSTM size and training datasets used in Experiment 2. Configurations are grouped by experiment

	cells	n basins (k=0)	n basins (k=1)	training data (years)	n samples (thousands)
code					
cluster(0)-n(32)	128	32	0	12	140
cluster(0)-n(64)	128	64	0	12	280
cluster(1)-n(32)	128	0	32	12	140
cluster(1)-n(64)	128	0	64	12	280
cluster(0,1)-n(64)	128	32	32	12	280
cluster(0)-n(32) half	64	32	0	12	140
cluster(0)-n(64) half	64	64	0	12	280
cluster(1)-n(32) half	64	0	32	12	140
cluster(1)-n(64) half	64	0	64	12	280
cluster(0,1)-n(64) half	64	32	32	12	280
cluster(0)-n(32) cus	128	32	0	6	70
cluster(0)-n(64) cus	128	64	0	6	140
cluster(1)-n(32) cus	128	0	32	6	70
cluster(1)-n(64) cus	128	0	64	6	140
cluster(0,1)-n(64) cus	128	32	32	6	140
cluster(0)-n(32) cus half	64	32	0	6	70
cluster(0)-n(64) cus half	64	64	0	6	140
cluster(1)-n(32) cus half	64	0	32	6	70
cluster(1)-n(64) cus half	64	0	64	6	140
cluster(0,1)-n(64) cus half	64	32	32	6	140

5.5 Results and discussion

5.5.1 Experiment 1a: cluster-based temporal undersampling

Examples of temporal clustering results are provided in Figs. 5.2 (for basin 01AD003) and 5.3 (basins 01AD003 and 07AF002, one from each basin cluster from Sec. 5.5.3). These results are for 6 clusters and minimum cluster sizes of 365. Although associations between clusters and specific hydrological characteristics can be expected to vary between individual basins, the results from basins 01AD003 and 07AF002 characterise four seasonal periods, as well as rising and receding limbs. Distinguishing between rising and falling limbs is consistent with previous studies that used streamflow clustering [Toth, 2009]. Ensuring that distinct seasons are represented in the clustering results is important, as streamflow drivers are known to change throughout the year.

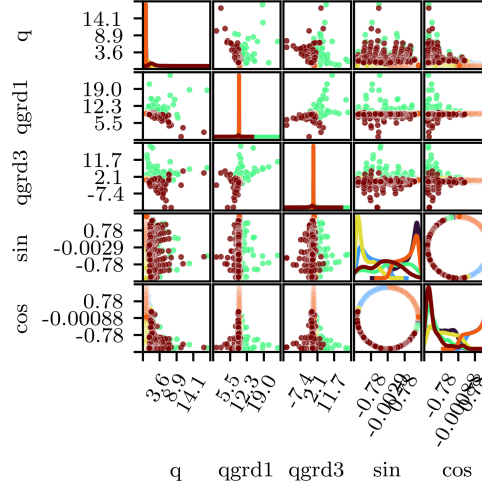


Figure 5.2: Scatterplot matrix illustrating the temporal clustering results for Basin 01AD003 streamflows ($K=6$). Markers are coloured by cluster and kernel smoothed histograms are shown along the diagonal. Axis labels q , $qgrd1$, $qgrd3$, $\sin$, and $\cos$, are shortform for q_t , $(q_{t-3} - q_t)/3$, $q_{t-1} - q_t$, $\sin^{-1}(t/365)$, and $\cos^{-1}(t/365)$, respectively

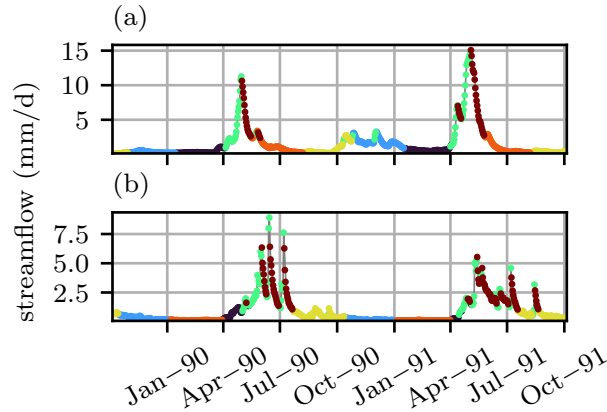


Figure 5.3: Hydrographs for Basins 01AD003 (top) and 07AF002 (bottom) with observations from October 1989–October 1991, coloured by cluster. Cluster colours are arbitrarily assigned.

The performance of models trained on a set of 64 randomly sampled basins is shown in Fig. 5.4 in terms of NSE (a-c) and PI (d-f) for cases without resampling, CUS_Q and temporal RUS. The cumulative density functions (CDFs), which have an optimum shape ‘J’, represent the proportion of basins (in the evaluation set) that fall below the performance along the x-axis. The baseline model (no resampling) performs reasonably well across basins, with 100%

and 75% of basins achieving an NSE greater than 0.5 at the 1-day and 3-day lead times, respectively. Roughly 95% of basins achieve a positive PI, indicating lower error than the naive model.

In comparison, the basin sets with CUS_Q at a rate of 0.5, (meaning that they use 6 out of 12 years of available training data), achieve the same level of performance as the baseline, for both the 6-cluster and 12-cluster cases. The RUS model trained to 6 years of randomly sampled data performs very poorly. Similarly, the CUS_Q model trained to 3 years of cluster-based data perform poorly, indicating that key hydrological processes are no longer sufficiently represented in the reduced training data. This also indicates that a lower limit of the extent to which CUS_Q can be used is somewhere between an undersampling rate of 0.25 and 0.5. Finally, in no cases do any undersampled configurations outperform the model trained to all data.

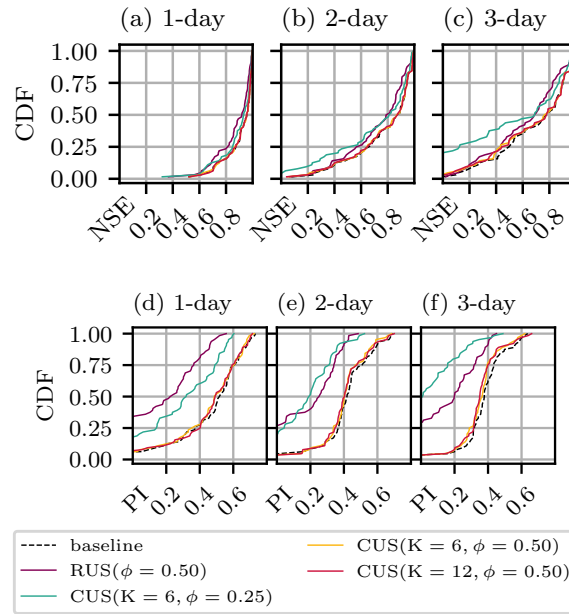


Figure 5.4: CDFs showing model performance according to NSE (top row) and PI (bottom row) for RUS (red), two CUS_Q configurations (blue and green) and no resampling baseline (black). Subplots (a), (b), (c), and (d), (e), (f) correspond to forecasts of 1, 2, and 3 days respectively.

These results highlight how a simple clustering method can be used to efficiently identify subsets of data that are sufficiently representative of the hydrological processes, which are identified by the clustering method, contained in each basin such that there is no loss in temporal generalisation. In addition, these results show that a significant proportion of hydrological

data within a continuous series is redundant and needlessly adds to the computational burden of training, which is especially relevant to computationally expensive tasks such as hyperparameter optimisation (HPO). Reducing the computational requirement of HPO speeds up model development, or allows for more extensive HPO, potentially improving model accuracy, thus FEWS reliability.

5.5.2 Experiment 1b: cluster-based spatial undersampling

In experiment 1b, CUS_B is used to select training basins. As with temporal undersampling, reducing the number of training basins required to train models has the potential to drastically reduce the computational demand of training, especially across large domains such as the thousands of gauged basins spread across Canada.

First, a baseline model is trained to a set of 128 randomly sampled basins. The basins are then grouped into K clusters, sampled at rates of 0.5 and 0.75. As with the previous experiment, RUS configurations are included at the same resampling rates, which in this case, consists of models trained to randomly sampled basin sets. The CDFs for each training configuration are shown in Fig. 5.5. Unlike with the temporal streamflow undersampling in experiment 1a, the basin subsets are unable to match the performance of the baseline model that is trained to the complete set of basins, which is visibly shown by the CDFs for PI. As expected, the configurations with a greater number of training basins perform most closely to the baseline. The CUS_B configurations narrowly outperform the RUS configurations with the same undersampling rates, most of all at a rate of 0.50.

5.5.3 Experiment 2: cross-comparison of 2 clusters of basins

To better understand the extent to which including hydrologically similar basins in the training dataset can benefit model performance, we consider an extreme case in which basins are grouped into 2 clusters, which are referred to as C0 and C1. The choice of 2 clusters is based on the maximum silhouette score, which is a commonly used measure of cluster cohesion [Rousseeuw, 1987]. An important note on the result of the maximum silhouette score is that the result depends on the use of constrained K-means clustering, which uses a minimum cluster size of 64. A lower minimum cluster size tends to increase the optimum number of clusters, details are provided in the supplementary information. Another reason for using two clusters is that it

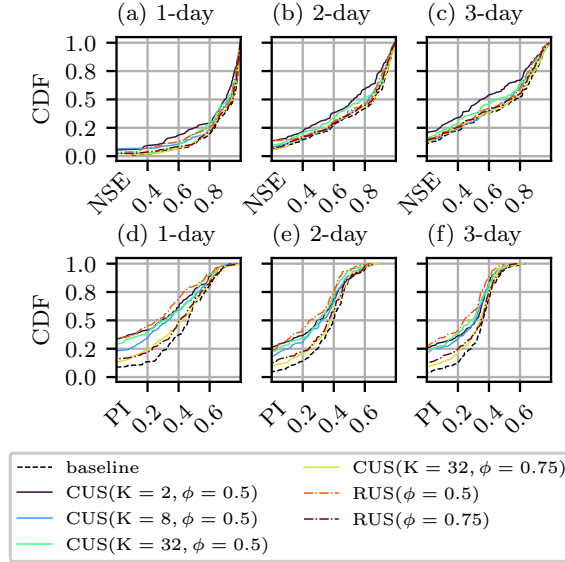


Figure 5.5: CDFs for NSE (top row) and PI (bottom row) for models trained to various basin subsets, which include a baseline (black line, includes all basins), cluster-based undersampling (coloured solid lines), and random undersampling (coloured dashed lines). The number of basins sampled from each cluster in the CUS_B configurations is equal to B/K . Subplots (a), (b), (c), and (d), (e), (f) correspond to forecasts of 1, 2, and 3 days respectively.

simplifies the analysis. For example, we compare the effects of similar, and dissimilar clusters; in contrast, a greater number of clusters would require a larger number of training configuration at varying degrees of hydrological similarity

Fig. 5.6 shows the spatial distribution of the cluster labels across Canada. C0 basins tend to be located at low elevations, along coastlines and east of the Rocky Mountain range. In contrast, the C1 basins are mainly confined to higher elevations in the Rocky Mountains. Mean basin attributes for each cluster are provided in Table S2 in the supplementary information.

The five most important features, identified by applying an RF to unsupervised clustering outcome, include elevation, slope, and landcovers, are shown in Fig. 5.7. The relevant features identified here by unsupervised clustering are consistent with the relevant descriptors deemed significant for determining hydrological similarity in the model-based method referenced in [Oudin et al., 2010].

First, for each cluster, a model is trained to a set of 32 basins from that cluster. To measure the value of adding hydrologically similar basins, 32 additional basins are added to the baseline training dataset. Finally, to measure the effects of dissimilar basins, 32 dissimilar basins are instead added to the baseline training set. Performance metrics are only calculated for the

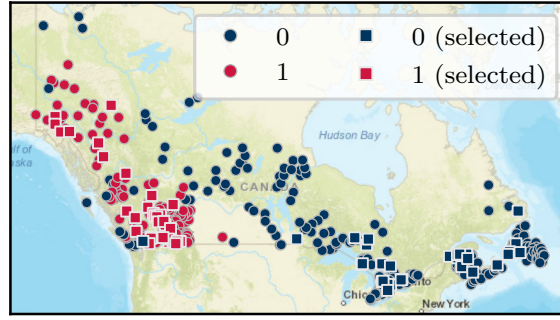


Figure 5.6: Clustering ($K=2$) result for basins across Canada. For each cluster, 32 square markers indicate basins selected for the baseline and evaluation sets in the two-cluster experiments. World Street Map basemap provided by ESRI.

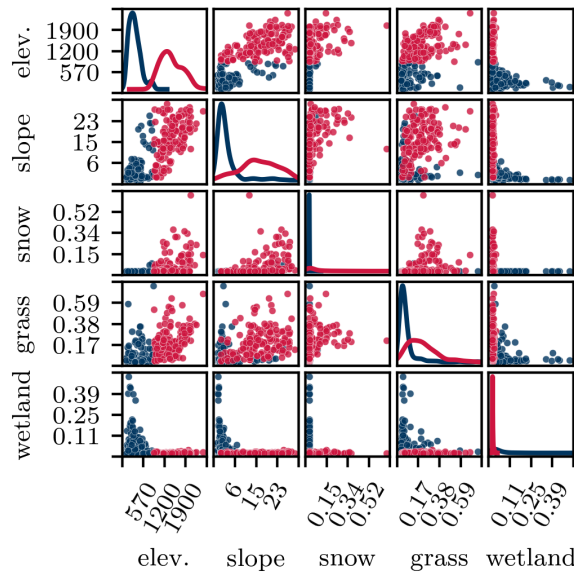


Figure 5.7: Scatterplot matrix illustrating the five most important features for clustering, as determined using the RF surrogate model. Markers are coloured by cluster and kernel smoothed histograms are shown along the diagonal.

baseline set of 32 basins. The above is repeated for each cluster.

Figs. 5.8 and 5.9 show the performance of the models evaluated on C0 and C1 basins, respectively. In C0 basins, more similar basins produce a notable improvement in performance across all lead times. Adding dissimilar basins instead produces even greater improvements, most notably, according to the PI. The same trends are seen with the C1 forecasts, with an even greater difference between the scores of the ‘+ similar’ and ‘+ dissimilar’ training sets.

In C0 basins, adding more similar, or dissimilar basins both improve the performance, with dissimilar basins producing larger improvements across all lead times and both metrics. A

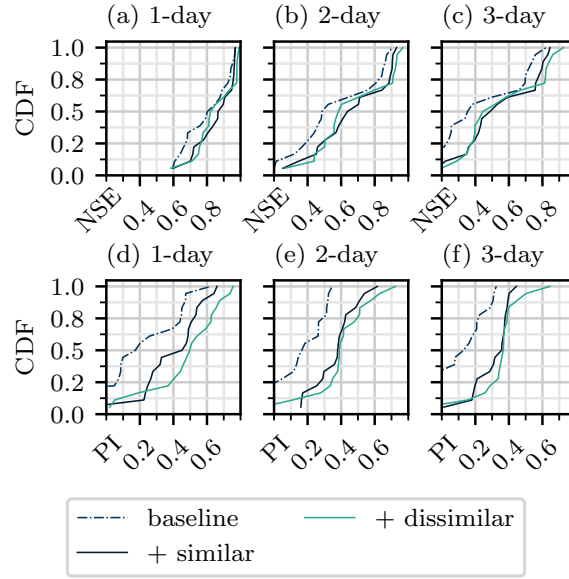


Figure 5.8: CDFs for models evaluated on C0 basins according to NSE (top row) PI (bottom row). Each row compares models trained to three different training datasets: a baseline that includes 32 basins in the respective cluster, the baseline plus 32 basins in the same cluster, and the baseline plus 32 basins in the other cluster. Subplots (a), (b), (c), and (d), (e), (f) correspond to forecasts of 1, 2, and 3 days respectively.

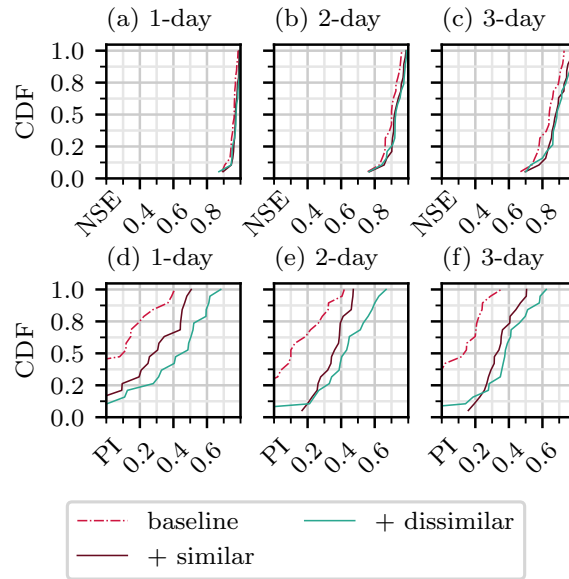


Figure 5.9: CDFs for models evaluated on C1 basins according to NSE (top row) PI (bottom row). Each row compares models trained to three different training datasets: a baseline that includes 32 basins in the respective cluster, the baseline plus 32 basins in the same cluster, and the baseline plus 32 basins in the other cluster. Subplots (a), (b), (c), and (d), (e), (f) correspond to forecasts of 1, 2, and 3 days respectively.

similar result is observed with the C1 basins, with the addition of dissimilar basins causing comparatively greater improvements in performance.

In experiment 2b the configurations from 2a are repeated, but incorporating CUS_Q, which is introduced in Sec. 5.5.1. Models from 2a and 2b are compared in Fig. ?? for C0 (a) and C1 (b) evaluation sets. Consistent with the results from experiment 1a in Sec. 5.5.1, CUS_Q is found to have very little impact on model test performance. This result indicates that for fixed training dataset size, data from outside the region of interest is much more useful to the training procedure than redundant data within the region of interest, identifiable using the clustering procedure.

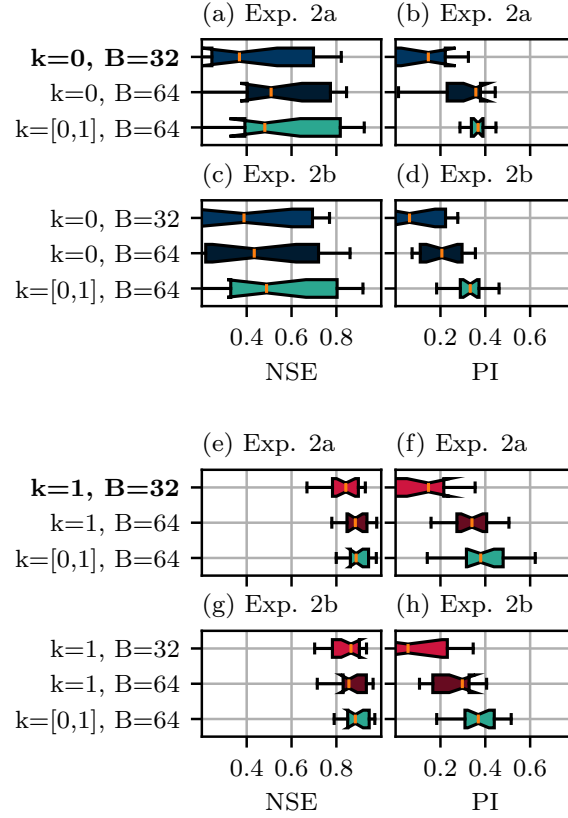


Figure 5.10: Boxplots showing the NSE and PI of models evaluated on 32 C0 basins for experiment 2a (a, b), and 2b (b, c), and 32 C1 basins for experiment 2a (e, f) and 2b (g, h). for a lead time of 3 days. The baseline model, which is trained uniquely using the evaluation basins, is indicated in bold. Blue, red, and green colours indicate models trained on C0, C1, and both types of basins.

Collectively, these results reveal that the addition of training basins with distinctive hydrological characteristics is more useful in terms of improving model performance when compared to the addition of basins with similar characteristics. This outcome might be counter-intuitive, since one could expect that training to additional basins that are most similar to the test set would be the most useful. This result also highlights the danger of training models to a hydrologically similar basin sets, which is a common approach in literature [Kratzert et al., 2024,

Hashemi et al., 2022].

One explanation for this result is that the input feature set is missing key explanatory variables. Two basins could have similar input vectors but different corresponding streamflow values. This difference may be explained by processes that are not captured within the input features. For example, two basins may appear to be hydrologically similar based on the available basin attributes, may have a different rainfall-runoff relationship, due to factors not included in the basin attributes, such as surficial geology, or the presence of hydraulic structures such as dams. While the LSTM should be able to distinguish between these two basins using the one-hot encoded basin labels, incomplete explanatory variables may inhibit the ability of models to transfer learned behaviour between basins.

Another explanation is that hydrologically similar basins contain a high degree of overlapping input-output patterns, reducing the amount of new information that can benefit predictions in the region of interest, in contrast to dissimilar basins. Information from dissimilar basins could be useful from a hydrological perspective, or simply provide a regularisation effect to the LSTM. Adding basins with distinct hydrological properties from some region of interest might occupy more neural pathways during model training compared to basins that have similar properties, which could mitigate overfitting to the region of interest. However, this explanation is not supported by the fact that constraining the number of cells in the network did not produce comparable regularisation.

A final explanation is that while similar basins provide examples of similar hydrological behaviour, dissimilar basins provide examples of what not to predict. A simple analogy from image classification, is that a model trained to classify photos of dogs might benefit more from being trained to some photos of cats, than to be trained to more photos of dogs.

Lastly, to better understand the effect that additions of similar and dissimilar basins have on model performance, we consider small, incremental additions. First, we begin with a model trained to 4 basins in a given cluster. Next, we consider two additions to the training dataset: 4 C0 basins, and 4 C1 basins. The addition that produces the best performance is retained, and the process is repeated. The models are retrained from scratch for each modification to the training dataset. The outcome is shown in Fig. 5.11 for models evaluated on 4 C0 basins (a) and 4 C1 basins (b). In both cases, the models benefit from the addition of 4 basins that

belong to the same cluster, however, afterwards there are no clear preference in terms of which basin clusters produce the best improvements. Despite some incremental additions hampering model performance, the performance on each evaluation improves across the larger training dataset, but the improvements decay exponentially, which is consistent with other studies that have looked at model performance across increasing training data [Kratzert et al., 2024, Gauch et al., 2021]. In Fig. 5.11(a), adding 4 similar basins to the 8 basin training dataset produces a large loss in performance, which highlights the lack of robustness against models trained to very small basin sets. Between C0 and C1 evaluations, C0 basins are more sensitive to new training data, and are comparatively more likely than C1 basins to exhibit worse performance after the addition of new training basins.

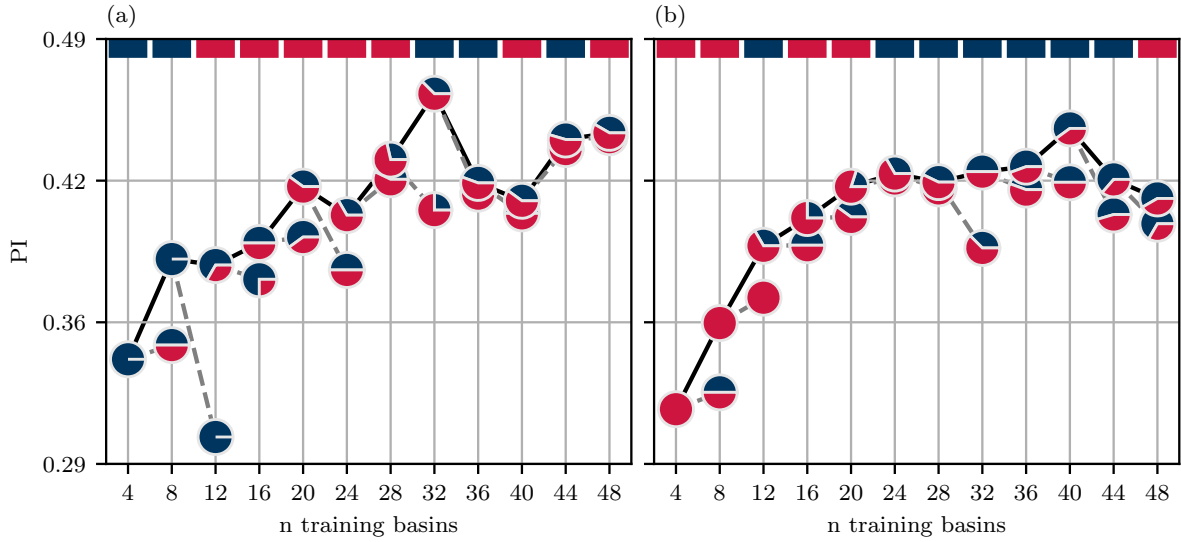


Figure 5.11: Model performance (PI) across increasing numbers of training basins for models evaluated on 4 C0 basins (a) and 4 C1 basins (b). Pie chart markers illustrate the proportion of C0 and C1 basins used in each training dataset. The coloured dashes along the top of each subplot indicate the which cluster produced the better addition to the training dataset.

The experimental results detailed above all assert the importance of hydrologically diverse, information rich training datasets. This is of particular importance in small regions of interest, where far away, dissimilar basins may be seen as not relevant to the training task. This study presents many opportunities for future work on curating datasets for hydrological models. While our study uses a simple clustering approach to quantify hydrological diversity, more sophisticated approaches, such as one based on mutual information, may further improve the results. Additionally, relatively little work exists on transferring models between different hydrological regions - which can potentially provide an improved starting point for model train-

ing, leading to better performance and more reliable FEWS.

5.6 Conclusion

The selection of training data is amongst the most important factors contributing to the performance of streamflow forecasting models. Our study showed that the performance of flow forecasting models relies on diverse training data, using a novel use of cluster-based resampling to identify and maximise temporal and spatial hydrological diversity within training datasets. In the first set of experiments, cluster-based undersampling was used to eliminate redundant temporal data from training datasets, drastically reducing the computational demand of model training. The next set of experiments showed how, given some region of interest, data from hydrologically dissimilar basins can be much more useful than data from similar basins. This result is counter to the intuitive approach to curating training basins for training, which is to train models to a group of hydrologically similar, or proximal basins. This outcome also highlights the need for large and hydrologically diverse training datasets. The latter can be combined with cluster-based temporal undersampling to generate diverse training sets that produce more performative models, for a fixed number of training observations, compared to models trained without cluster-based undersampling. Finally, temporal and spatial undersampling routines are combined to demonstrate how, for a fixed number of training samples, spatial hydrological diversity is much more beneficial than temporal diversity. These findings are critical to improving the reliability and accuracy of flood forecasting models, and minimising the effects of flooding.

6 Conclusion

6.1 Summary

The overarching objective of this body of work is to advance training procedures and data selection hydrological models as to improve their performance on flood level flows. In the following dissertation, novel methods are developed for three different modelling frameworks: physically-based hydrological models, ensembles of shallow ML models, and for deep learning models. The chapters and their attributes are summarised in Table 6.1 and in the paragraphs below.

Ch.	Name	Training data	Spatial scale	Temporal scale
2	SWMM	Event segmentation, clustering	2 urban basins	15M
3	Classification ensembles	SMOTE	3 basins	3H
4	Regression ensembles	SMOTE*	2 urban basins	H/D
5	Deep learning	Spatial and temporal clustering	100+ basins	D

Table 6.1: Summary of chapters

Chapter 2 presents a novel hybrid framework, that provides an alternative to traditional multi-event physics-based model calibration. Unsupervised K-means clustering was used to cluster continuous rainfall-runoff events based on event statistics (e.g., peak flow, duration, etc.). SWMM ensembles were formed by calibrating each ensemble member to a different hydrological event. Using an even number of events from each cluster is shown to have the best performance on average, while the scheme could also be used to bias model accuracy towards a desired type of event. Performance was improved further using ANN-based ensemble combination.

Chapter 3 demonstrates a binary classification approach to flood forecasting. The approach uses a novel labelling scheme that better reflects actual flood conditions, compared to traditional post-processing approaches. Classifying flood events during preprocessing is found to produce slightly better accuracy compared to post-processing ANN regression models. While ANN-type models are by far the most popular for flood forecasting, SVMs and ELMs were found to often yield higher accuracy and a more favourable tradeoff between true and false positive predictions. Presenting imbalanced classification metrics such as the tradeoff between true and false positive of models was shown to be invaluable for model selection, which is not captured using typical flood forecasting metrics such as the NSE.

The objectives of the third manuscript are (i) to develop a systematised framework for embedding resampling methods within ensemble algorithms, as to improve the error diversity between ensemble members; and (ii) to evaluate the proposed scheme with respect to its ability to improve high streamflow performance, using specialised metrics.

Chapter 4 advances ensemble algorithms, which include Bagging, AdaBoost, LSBoost, by integrating them with resampling methods. The added resampling step in each ensemble algorithms implicitly increases ensemble diversity, thus performance. Three resampling methods are considered: oversampling, undersampling, and SMOTER, all of which are aimed at increasing the proportion of flood samples in the training dataset. Each resampling method was cross-compared with each ensemble algorithm, for a total of 16 ensemble configurations, plus a baseline model. Ultimately, the modified ensemble models were found to have the capacity to marginally improve high flow performance. An aggressive resampling strategy can further improve high flow performance, but was found to rapidly degrade performance on typical streamflows, opening the possibility of false-positive alarms.

Lastly, Chapter 5 applied unsupervised constrained K-means clustering to group streamflow and basins into hydrologically similar groups. Streamflow training data was shown to be highly redundant, and data from the majority clusters could be removed with no adverse effect to model performance. In contrast, data from ‘hydrologically similar’ basins could not be removed without some loss in overall accuracy. Next, hydrologically dissimilar basins among the training data were shown to be hugely beneficial to model performance.

6.2 Contributions

This dissertation explored the development of methods for training flood forecasting models as to improve their performance as FEWS. Broadly, these chapters all focused on clustering and resampling methods for generating hydrologically diverse datasets, that promote accurate performance on flood level streamflows. Some notable contributions in the dissertation include:

1. A novel ensemble algorithm for SWMM, which accommodates equifinality and enables hybrid modelling.
2. Systematic cross-comparisons of ensemble algorithms and ML models for both classification and regression approaches to flood forecasting

3. Novel adaptations of several ensemble algorithms to improve their performance on skewed datasets
4. A new, persistence-based cost function for regionally trained deep learning models
5. A framework for generating spatiotemporally diverse training data for regionally trained deep learning models

6.3 Limitations of research

An obvious limitation of chapters 2–4 are that they were each only conducted for 2 to 3 basins. While case studies on small numbers of basins has been common throughout the past several decades, within the last few years there has been a shift towards large-sample studies, featuring dozens or hundreds of basins. Such studies allow for the validation of proposed improvements to be validated across a much larger spectrum of spatiotemporal conditions. Large-sample are made easier by the recent push for large-sample datasets, which not only reduce data wrangling requirements, but also allow for more standardised benchmarking. That being said, the aforementioned chapters use data at spatiotemporal resolutions finer than what is currently available in major large-sample datasets, which typically offer hydrometric data at daily resolutions.

Another limitation of chapters 2–4 is that they do not include any snow data. Winter months were omitted from these chapters due to the complex nature, and unavailability of data, of snow ploughing and removal. Since snow is often moved between catchments, it is difficult to form a reliable estimate of snowpack. This limitation is addressed in Chapter 5, in which SWE is used as an input feature.

Chapter 5 utilises a large-sample dataset with over 3000 Canadian basins, but is constrained to models trained on subsets of up to 128 basins, due to limitations of computational resources. With increasing dataset standardisation, there will likely be increasing numbers of pre-trained models, that can be fine-tuned at a fraction of the computational cost of training from scratch.

A limitation of Chapters 2–5 is that they all use two classes. There are various reasons for using two classes. In most cases, it simplifies the analysis and cross-comparison. In Chapter 5 it is a result of the silhouette score of the constrained K-means clustering results. In all cases, the work would be improved through the use of more clusters, which would increase the hy-

drological diversity of training data in Chapters 2, 4, and 5. In Chapter 3 it would increase the usefulness of the model output in terms of FEWS.

6.4 Future work

This section recommends avenues for future research, which reflect both outstanding research questions in each chapter and the current research landscape. These include:

1. Apply ensemble frameworks to LSTMs Throughout recent years, model application has shifted from ensembles with hundreds of base learners, to using single deep learning models. However, deep learning models still benefit from ensemble frameworks. The lessons learned in Chapters 3 and 4 can be extended to deep-learning regional models. The benefits of ensemble learning could be extended to deep learning models at minimal computational cost by first pretraining a single model, then performing fine-tuning within an ensemble framework.
2. Enhancements to spatiotemporal subsampling procedure to focus more on flood events With increasing evidence of the effectiveness of deep-learning approaches to flow forecasting, such should be central to future work. More specifically, Chapter 5 highlighted how regional learning can benefit from information-rich data subsets. This finding could potentially be exploited further by using resampling algorithms such as SMOTE, featured in 3 and 4, to generate training datasets with very large numbers of flood observations.
3. Benchmarking against operational FEWS There exists a large body of research devoted to demonstrating how machine learning models achieve higher accuracy compared to traditional, physics-based models. The physics-based models included in these comparisons are typically relatively simple lumped models that are generated for the sake of the comparison. Moreover, comparisons are typically made using non-specialised performance metrics. A more convincing approach, which has not yet been investigated on a large scale, is to compare data-driven FEWS with traditional, operational FEWS. Operational FEWS may differ from the physics-based models used in research in a variety of ways. For example, the underlying models might be more sophisticated, or the flood warning may rely on the operation of infrastructure, something that is not likely rep-

resented in a lumped model. Such a comparison poses challenges as historically issued flood warnings are not widely available and the format or underlying models might be considerably different between regions.

References

- G. o. A. . Respecting our rivers : Alberta’s approach to flood mitigation. Technical report, 2014.
- J. Abbot and J. Marohasy. Input selection and optimisation for monthly rainfall forecasting in Queensland, Australia, using artificial neural networks. *Atmospheric Research*, 138:166–178, Mar. 2014. ISSN 01698095. doi: 10.1016/j.atmosres.2013.11.002.
- R. J. Abrahart. Neural network rainfall-runoff forecasting based on continuous resampling. *Journal of Hydroinformatics*, 5(1):51–61, Jan. 2003. ISSN 1464-7141. doi: 10.2166/hydro.2003.0004.
- R. J. Abrahart and L. See. Comparing neural network and autoregressive moving average techniques for the provision of continuous river flow forecasts in two contrasting catchments. *Hydrological Processes*, 14(11-12):2157–2172, Aug. 2000. ISSN 0885-6087. doi: 10.1002/1099-1085(20000815/30)14:11/12<2157::AID-HYP57>3.0.CO;2-S.
- R. J. Abrahart, L. See, and P. E. Kneale. Investigating the role of saliency analysis with a neural network rainfall-runoff model. *Computers & Geosciences*, 27(8):921–928, Oct. 2001. ISSN 00983004. doi: 10.1016/S0098-3004(00)00131-X.
- R. J. Abrahart, A. J. Heppenstall, and L. M. See. Timing error correction procedure applied to neural network rainfall-runoff modelling. *Hydrological Sciences Journal*, 52(3):414–431, June 2007. ISSN 02626667. doi: 10.1623/hysj.52.3.414.
- R. J. Abrahart, F. Anctil, P. Coulibaly, C. W. Dawson, N. J. Mount, L. M. See, A. Y. Sham-seldin, D. P. Solomatine, E. Toth, and R. L. Wilby. Two decades of anarchy? Emerging themes and outstanding challenges for neural network river forecasting. *Progress in Physical Geography: Earth and Environment*, 36(4):480–513, Aug. 2012. ISSN 0309-1333. doi: 10.1177/0309133312444943.
- J. Adamowski, H. F. Chan, S. O. Prasher, and V. N. Sharda. Comparison of multivariate adaptive regression splines with coupled wavelet transform artificial neural networks for runoff forecasting in Himalayan micro-watersheds with limited data. *Journal of Hydroinformatics*, 14(3):731–744, July 2012. ISSN 1464-7141. doi: 10.2166/hydro.2011.044.

- N. Addor, A. J. Newman, N. Mizukami, and M. P. Clark. The CAMELS data set: Catchment attributes and meteorology for large-sample studies. *Hydrology and Earth System Sciences*, 21(10):5293–5313, Oct. 2017. ISSN 1607-7938. doi: 10.5194/hess-21-5293-2017.
- M. S. Advani, A. M. Saxe, and H. Sompolinsky. High-dimensional dynamics of generalization error in neural networks. *Neural Networks*, 132:428–446, Dec. 2020. ISSN 0893-6080. doi: 10.1016/j.neunet.2020.08.022.
- I. E. Agbehadji, T. Mabhaudhi, J. Botai, and M. Masinde. A systematic review of existing early warning systems’ challenges and opportunities in cloud computing early warning systems. *Climate*, 11(9):188, 2023.
- C. C. Aggarwal and S. Sathe. Model Combination Methods for Outlier Ensembles. In *Outlier Ensembles*, pages 187–205. Springer International Publishing, Cham, 2017. doi: 10.1007/978-3-319-54765-7_5.
- Agriculture and Agri-Food Canada. Soil Surveys for Ontario. <https://sis.agr.gc.ca/cansis/publications/surveys/on/index.html>, 2022.
- A. N. Ahmed, T. Van Lam, N. D. Hung, N. Van Thieu, O. Kisi, and A. El-Shafie. A comprehensive comparison of recent developed meta-heuristic algorithms for streamflow time series forecasting problem. *Applied Soft Computing*, 105, July 2021. ISSN 15684946. doi: 10.1016/j.asoc.2021.107282.
- B. Alizadeh, A. Ghaderi Bafti, H. Kamangir, Y. Zhang, D. B. Wright, and K. J. Franz. A novel attention-based LSTM cell post-processor coupled with bayesian optimization for streamflow prediction. *Journal of Hydrology*, 601:126526, Oct. 2021. ISSN 0022-1694. doi: 10.1016/j.jhydrol.2021.126526.
- M. H. Alobaidi, P. R. Marpu, T. B. Ouarda, and F. Chebana. Regional frequency analysis at ungauged sites using a two-stage resampling generalized ensemble framework. *Advances in Water Resources*, 84:103–111, 2015. ISSN 03091708. doi: 10.1016/j.advwatres.2015.07.019.
- M. H. Alobaidi, M. A. Meguid, and F. Chebana. Predicting seismic-induced liquefaction through ensemble learning frameworks. *Scientific Reports*, 9(1):11786, Dec. 2019. ISSN 2045-2322. doi: 10.1038/s41598-019-48044-0.

- M. H. Alobaidi, M. A. Meguid, and F. Chebana. Varying-parameter modeling within ensemble architecture: Application to extended streamflow forecasting. *Journal of Hydrology*, 582: 124511, Mar. 2020. ISSN 00221694. doi: 10.1016/j.jhydrol.2019.124511.
- F. Anctil and N. Lauzon. Generalisation for neural networks through data sampling and training procedures, with applications to streamflow predictions. *Hydrology and Earth System Sciences*, 8(5):940–958, 2004a. ISSN 1027-5606. doi: 10.5194/hess-8-940-2004.
- F. Anctil and N. Lauzon. Generalisation for neural networks through data sampling and training procedures, with applications to streamflow predictions. *Hydrology and Earth System Sciences*, 8(5):940–958, Oct. 2004b. ISSN 1607-7938. doi: 10.5194/hess-8-940-2004.
- S. Anderson and V. Radić. Interpreting Deep Machine Learning for Streamflow Modeling Across Glacial, Nival, and Pluvial Regimes in Southwestern Canada. *Frontiers in Water*, 4, 2022. ISSN 2624-9375.
- D. T. Anh, D. V. Thanh, H. M. Le, B. T. Sy, A. H. Tanim, Q. B. Pham, T. D. Dang, S. T. Mai, and N. M. Dang. Effect of Gradient Descent Optimizers and Dropout Technique on Deep Learning LSTM Performance in Rainfall-runoff Modeling. *Water Resources Management*, 37(2):639–657, 2023.
- Aquafor Beech. Runoff Volume Control Targets for Ontario. Technical report, Aquafor Beech, Guelph, 2016.
- T. Arriero Shinma and L. F. Ribeiro Reis. Incorporating multi-event and multi-site data in the calibration of SWMM. In *Procedia Engineering*, volume 70, pages 75–84. Elsevier Ltd, 2014. doi: 10.1016/j.proeng.2014.02.010.
- R. Arsenault, A. Poulin, P. Côté, and F. Brissette. Comparison of Stochastic Optimization Algorithms in Hydrological Model Calibration. *Journal of Hydrologic Engineering*, 19(7): 1374–1384, July 2014. ISSN 1084-0699, 1943-5584. doi: 10.1061/(ASCE)HE.1943-5584.0000938.
- R. Arsenault, F. Brissette, J.-L. Martel, M. Troin, G. Lévesque, J. Davidson-Chaput, M. C. Gonzalez, A. Ameli, and A. Poulin. A comprehensive, multisource database for hydrometeorological modeling of 14,425 North American watersheds. *Scientific Data*, 7(1):243, Dec.

2020. ISSN 2052-4463. doi: 10.1038/s41597-020-00583-2.

R. Arsenault, J.-L. Martel, F. Brunet, F. Brissette, and J. Mai. Continuous streamflow prediction in ungauged basins: Long short-term memory neural networks clearly outperform traditional hydrological models. *Hydrology and Earth System Sciences*, 27(1):139–157, Jan. 2023a. ISSN 1607-7938. doi: 10.5194/hess-27-139-2023.

R. Arsenault, J.-L. Martel, F. Brunet, F. Brissette, and J. Mai. Continuous streamflow prediction in ungauged basins: Long short-term memory neural networks clearly outperform traditional hydrological models. *Hydrology and Earth System Sciences*, 27(1):139–157, Jan. 2023b. ISSN 1607-7938. doi: 10.5194/hess-27-139-2023.

M. Atieh, G. Taylor, A. M.A. Sattar, and B. Gharabaghi. Prediction of flow duration curves for ungauged basins. *Journal of Hydrology*, 545:383–394, Feb. 2017. ISSN 00221694. doi: 10.1016/j.jhydrol.2016.12.048.

M. Atiquzzaman and J. Kandasamy. Prediction of hydrological time-series using extreme learning machine. *Journal of Hydroinformatics*, 18(2):345–353, Mar. 2016. ISSN 14647141. doi: 10.2166/hydro.2015.020.

F. S. Awol, P. Coulibaly, and B. A. Tolson. Event-based model calibration approaches for selecting representative distributed parameters in semi-urban watersheds. *Advances in Water Resources*, 118:12–27, Aug. 2018. ISSN 03091708. doi: 10.1016/j.advwatres.2018.05.013.

G. Banjac, M. Vařak, and M. Baotić. Adaptable urban water demand prediction system. *Water Supply*, 15(5):958–964, Oct. 2015. ISSN 1606-9749. doi: 10.2166/ws.2015.048.

J. Barco, K. M. Wong, and M. K. Stenstrom. Automatic Calibration of the U.S. EPA SWMM Model for a Large Urban Catchment. *Journal of Hydraulic Engineering*, 134(4):466–474, Apr. 2008. ISSN 0733-9429. doi: 10.1061/(ASCE)0733-9429(2008)134:4(466).

R. Barzegar, M. Ghasri, Z. Qi, J. Quilty, and J. Adamowski. Using bootstrap ELM and LSSVM models to estimate river ice thickness in the Mackenzie River Basin in the Northwest Territories, Canada. *Journal of Hydrology*, 577:123903, Oct. 2019. ISSN 00221694. doi: 10.1016/j.jhydrol.2019.06.075.

P. K. Behera, Y. Guo, R. S. Teegavarapu, and T. Branham. Evaluation of Antecedent Storm

- Event Characteristics for Different Climatic Regions based on Interevent Time Definition (IETD). In *World Environmental and Water Resources Congress 2010*, pages 2441–2450, Reston, VA, May 2010. American Society of Civil Engineers. ISBN 978-0-7844-1114-8. doi: 10.1061/41114(371)251.
- A. Belayneh, J. Adamowski, B. Khalil, and J. Quilty. Coupling machine learning methods with wavelet transforms and the bootstrap and boosting ensemble approaches for drought prediction. *Atmospheric Research*, 172–173:37–47, 2016. ISSN 01698095. doi: 10.1016/j.atmosres.2015.12.017.
- K. Bennett, P. Bradley, and A. Demiriz. Constrained K-means clustering. Technical Report MSR-TR-2000-65, Microsoft, May 2000.
- N. D. Bennett, B. F. Croke, G. Guariso, J. H. Guillaume, S. H. Hamilton, A. J. Jakeman, S. Marsili-Libelli, L. T. Newham, J. P. Norton, C. Perrin, S. A. Pierce, B. Robson, R. Sepelt, A. A. Voinov, B. D. Fath, and V. Andreassian. Characterising performance of environmental models. *Environmental Modelling and Software*, 40:1–20, May 2013. ISSN 13648152. doi: 10.1016/j.envsoft.2012.09.011.
- P. M. R. Bento, J. A. N. Pombo, M. R. A. Calado, and S. J. P. S. Mariano. Stacking Ensemble Methodology Using Deep Learning and ARIMA Models for Short-Term Load Forecasting. *Energies*, 14(21):7378, Nov. 2021. ISSN 1996-1073. doi: 10.3390/en14217378.
- K. Beven. *Rainfall-Runoff Modelling*. Wiley-Blackwell, Lancaster, Jan. 2012. ISBN 978-0-470-71459-1. doi: 10.1002/9781119951001.
- K. Beven and A. Binley. The future of distributed models: Model calibration and uncertainty prediction. *Hydrological Processes*, 6(3):279–298, July 1992. ISSN 10991085. doi: 10.1002/hyp.3360060305.
- N. K. Biswas, F. Hossain, M. Bonnema, A. Aminul Haque, R. K. Biswas, A. Bhuyan, and A. Hossain. A computationally efficient flash flood early warning system for a mountainous and transboundary river basin in bangladesh. *Journal of Hydroinformatics*, 22(6):1672–1692, 2020.
- G. E. Blackler and J. C. Y. Guo. Field Test of Paved Area Reduction Factors Using a Storm

- Water Management Model and Water Quality Test Site. *Journal of Irrigation and Drainage Engineering*, 140(4):05013001, Apr. 2014. ISSN 0733-9437. doi: 10.1061/(ASCE)IR.1943-4774.0000680.
- J. Błaszczyński and J. Stefanowski. Neighbourhood sampling in bagging for imbalanced data. *Neurocomputing*, 150(PB):529–542, 2015. ISSN 18728286. doi: 10.1016/j.neucom.2014.07.064.
- G. J. Bowden, H. R. Maier, and G. C. Dandy. Optimal division of data for neural network models in water resources applications. *Water Resources Research*, 38(2):2–1, 2002. ISSN 00431397. doi: 10.1029/2001WR000266.
- P. Branco, L. Torgo, and R. P. Ribeiro. A survey of predictive modeling on imbalanced domains. *ACM Computing Surveys*, 49(2):31, Aug. 2016. ISSN 15577341. doi: 10.1145/2907070.
- L. Breiman. Bagging predictors. *Machine Learning*, 24(2):123–140, Aug. 1996a. ISSN 0885-6125. doi: 10.1007/BF00058655.
- L. Breiman. Stacked regressions. *Machine Learning*, 24(1):49–64, July 1996b. ISSN 0885-6125. doi: 10.1007/BF00117832.
- I. Broekhuizen, G. Leonhardt, J. Marsalek, and M. Viklander. Event selection and two-stage approach for calibrating models of green urban drainage systems. *Hydrology and Earth System Sciences*, 24(2):869–885, 2020. ISSN 16077938. doi: 10.5194/hess-24-869-2020.
- G. Brown, J. Wyatt, R. Harris, and X. Yao. Diversity creation methods: A survey and categorisation. *Information Fusion*, 6(1):5–20, 2005a. ISSN 15662535. doi: 10.1016/j.inffus.2004.04.004.
- G. Brown, J. L. Wyatt, and P. Tiño. Managing diversity in regression ensembles. *Journal of Machine Learning Research*, 6:1621–1650, 2005b. ISSN 15337928.
- D. H. Burn and P. H. Whitfield. Changes in floods and flood regimes in Canada. *Canadian Water Resources Journal / Revue canadienne des ressources hydriques*, 41(1-2):139–150, Apr. 2016. ISSN 0701-1784. doi: 10.1080/07011784.2015.1026844.

- F. Campolongo, J. Cariboni, and A. Saltelli. An effective screening design for sensitivity analysis of large models. *Environmental Modelling & Software*, 22(10):1509–1518, Oct. 2007. ISSN 13648152. doi: 10.1016/j.envsoft.2006.10.004.
- A. J. Cannon and P. H. Whitfield. Downscaling recent streamflow conditions in British Columbia, Canada using ensemble neural network models. *Journal of Hydrology*, 259(1-4): 136–151, Mar. 2002. ISSN 00221694. doi: 10.1016/S0022-1694(01)00581-9.
- S. Cateni, V. Colla, and M. Vannucci. A method for resampling imbalanced datasets in binary classification tasks for real-world problems. *Neurocomputing*, 135:32–41, July 2014. ISSN 18728286. doi: 10.1016/j.neucom.2013.05.059.
- L.-C. Chang, M. Z. M. Amin, S.-N. Yang, and F.-J. Chang. Building ann-based regional multi-step-ahead flood inundation forecast models. *Water*, 10(9):1283, 2018.
- K. Chapi, V. P. Singh, A. Shirzadi, H. Shahabi, D. T. Bui, B. T. Pham, and K. Khosravi. A novel hybrid artificial intelligence approach for flood susceptibility assessment. *Environmental Modelling & Software*, 95:229–245, Sept. 2017. ISSN 13648152. doi: 10.1016/j.envsoft.2017.06.012.
- N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer. SMOTE: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16:321–357, Jan. 2002. ISSN 10769757. doi: 10.1613/jair.953.
- S. Chen, M. Ren, and W. Sun. Combining two-stage decomposition based machine learning methods for annual runoff forecasting. *Journal of Hydrology*, 603:126945, 2021. ISSN 00221694. doi: 10.1016/j.jhydrol.2021.126945.
- W. Chen, H. Hong, S. Li, H. Shahabi, Y. Wang, X. Wang, and B. B. Ahmad. Flood susceptibility modelling using novel hybrid approach of reduced-error pruning trees with bagging and random subspace ensembles. *Journal of Hydrology*, 575:864–873, Aug. 2019. ISSN 00221694. doi: 10.1016/j.jhydrol.2019.05.089.
- K. Cho, B. van Merriënboer, D. Bahdanau, and Y. Bengio. On the Properties of Neural Machine Translation: Encoder-Decoder Approaches, Oct. 2014.
- M. P. Clark, R. M. Vogel, J. R. Lamontagne, N. Mizukami, W. J. M. Knoben, G. Tang,

- S. Gharari, J. E. Freer, P. H. Whitfield, K. R. Shook, and S. M. Papalexiou. The Abuse of Popular Performance Metrics in Hydrologic Modeling. *Water Resources Research*, 57(9): e2020WR029001, 2021. ISSN 1944-7973. doi: 10.1029/2020WR029001.
- J.-D. Creutin and M. Borga. Radar hydrology modifies the monitoring of flash-flood hazard. *Hydrological Processes*, 17(7):1453–1456, May 2003. ISSN 0885-6087. doi: 10.1002/hyp.5122.
- Ma. Cristina, A. Paule-Mercado, and C.-H. Lee. Calibration of the SWMM for a mixed land use catchment in Yongin, South Korea. In *Desalination and Water Treatment*, volume 63, pages 381–388, Kyoto, 2017. Desalination Publications. doi: 10.5004/dwt.2017.11441.
- L. Crochemore, C. Perrin, V. Andréassian, U. Ehret, S. P. Seibert, S. Grimaldi, H. Gupta, and J.-E. Paturel. Comparing expert judgement and numerical criteria for hydrograph evaluation. *Hydrological Sciences Journal*, 60(3):402–423, Mar. 2015. ISSN 0262-6667. doi: 10.1080/02626667.2014.903331.
- D. N. Moriasi, J. G. Arnold, M. W. Van Liew, R. L. Bingner, R. D. Harmel, and T. L. Veith. Model Evaluation Guidelines for Systematic Quantification of Accuracy in Watershed Simulations. *Transactions of the ASABE*, 50(3):885–900, 2007. ISSN 2151-0032. doi: 10.13031/2013.23153.
- C. W. Dawson and R. L. Wilby. Hydrological modelling using artificial neural networks. *Progress in Physical Geography: Earth and Environment*, 25(1):80–108, Mar. 2001. ISSN 0309-1333. doi: 10.1177/030913330102500104.
- M. De Coste, Z. Li, D. Pupek, and W. Sun. A hybrid ensemble modelling framework for the prediction of breakup ice jams on Northern Canadian Rivers. *Cold Regions Science and Technology*, 189:103302, Sept. 2021. ISSN 0165232X. doi: 10.1016/j.coldregions.2021.103302.
- A. de Lavenne, G. Thirel, V. Andréassian, C. Perrin, and M.-H. Ramos. Spatial variability of the parameters of a semi-distributed hydrological model. *Proceedings of the International Association of Hydrological Sciences*, 373:87–94, May 2016. ISSN 2199-899X. doi: 10.5194/piahs-373-87-2016.
- N. de Vos and T. Rientjes. Correction of Timing Errors of Artificial Neural Network Rainfall-Runoff Models. In *Practical Hydroinformatics*, pages 101–112. Springer, Berlin, Heidelberg,

2009. doi: 10.1007/978-3-540-79881-1_8.

N. J. de Vos and T. H. M. Rientjes. Constraints of artificial neural networks for rainfall-runoff modelling: Trade-offs in hydrological state representation and model evaluation. *Hydrology and Earth System Sciences Discussions*, 2(1):365–415, 2005. ISSN 1027-5606. doi: 10.5194/hessd-2-365-2005.

O. Delaigue, P. Brigode, V. Andréassian, C. Perrin, P. Etchevers, J.-M. Soubeyroux, B. Janet, and N. Addor. CAMELS-FR: A large sample hydroclimatic dataset for France to explore hydrological diversity and support model benchmarking. In *IAHS-2022 Scientific Assembly*, May 2022.

G. K. Devia, B. Ganasri, and G. Dwarakish. A Review on Hydrological Models. *Aquatic Procedia*, 4:1001–1007, Jan. 2015. ISSN 2214241X. doi: 10.1016/j.aqpro.2015.02.126.

T. Dietterich. Overfitting and undercomputing in machine learning. *ACM Computing Surveys*, 27(3):326–327, Sept. 1995. ISSN 0360-0300, 1557-7341. doi: 10.1145/212094.212114.

J. F. Díez-Pastor, J. J. Rodríguez, C. García-Osorio, and L. I. Kuncheva. Random Balance: Ensembles of variable priors classifiers for imbalanced data. *Knowledge-Based Systems*, 85: 96–111, Sept. 2015a. ISSN 09507051. doi: 10.1016/j.knosys.2015.04.022.

J. F. Díez-Pastor, J. J. Rodríguez, C. I. García-Osorio, and L. I. Kuncheva. Diversity techniques improve the performance of the best imbalance learning ensembles. *Information Sciences*, 325:98–117, Dec. 2015b. ISSN 00200255. doi: 10.1016/j.ins.2015.07.025.

DMTI Spatial Inc. Major Water Regions (MJWTR).
http://geo.scholarsportal.info/#r/details/_uri@=311685684, 2014a.

DMTI Spatial Inc. Intermittent Water (MNINR). http://geo.scholarsportal.info/#r/details/_uri@=2422157200, 2014b.

DMTI Spatial Inc. Minor Water Regions (MNWTR).
http://geo.scholarsportal.info/#r/details/_uri@=2840086328, 2014c.

DMTI Spatial Inc. Watersheds Region. http://geo.scholarsportal.info/#r/details/_uri@=2751227225, 2019.

- N. Duffy and D. Helmbold. Boosting methods for regression. *Machine Learning*, 47(2-3):153–200, 2002. ISSN 08856125. doi: 10.1023/A:1013685603443.
- A. Duncan. *The Analysis and Application of Artificial Neural Networks for Early Warning Systems in Hydrology and the Environment*. PhD thesis, University of Exeter, Sept. 2014.
- U. Ehret and E. Zehe. Series distance - An intuitive metric to quantify hydrograph similarity in terms of occurrence, amplitude and timing of hydrological events. *Hydrology and Earth System Sciences*, 15(3):877–896, June 2011. ISSN 10275606. doi: 10.5194/hess-15-877-2011.
- R. E. Emerton, E. M. Stephens, F. Pappenberger, T. C. Pagano, A. H. Weerts, A. W. Wood, P. Salamon, J. D. Brown, N. Hjerdt, C. Donnelly, C. A. Baugh, and H. L. Cloke. Continental and global scale flood forecasting systems. *WIREs Water*, 3(3):391–418, 2016. ISSN 2049-1948. doi: 10.1002/wat2.1137.
- Environment Canada. Weather, climate and hazards - Environment Canada. <https://www.canada.ca/en/services/environment/weather.html>, 2022.
- H. I. Erdal and O. Karakurt. Advancing monthly streamflow prediction accuracy of CART models using ensemble learning paradigms. *Journal of Hydrology*, 477:119–128, Jan. 2013. ISSN 00221694. doi: 10.1016/j.jhydrol.2012.11.015.
- Esri. World Imagery. https://services.arcgisonline.com/ArcGIS/rest/services/World_Imagery/MapServer, 2020.
- J. Ewen. Hydrograph matching method for measuring model performance. *Journal of Hydrology*, 408(1-2):178–187, June 2011. ISSN 00221694. doi: 10.1016/j.jhydrol.2011.07.038.
- K. Fang, D. Kifer, K. Lawson, D. Feng, and C. Shen. The Data Synergy Effects of Time-Series Deep Learning Models in Hydrology. *Water Resources Research*, 58(4):e2021WR029583, 2022. ISSN 1944-7973. doi: 10.1029/2021WR029583.
- T. Fernando, H. Maier, and G. Dandy. Selection of input variables for data driven models: An average shifted histogram partial mutual information estimator approach. *Journal of Hydrology*, 367(3-4):165–176, Apr. 2009. ISSN 00221694. doi: 10.1016/j.jhydrol.2008.10.019.
- A. Fiszlelew, P. Britos, A. Ochoa, H. Merlino, E. Fernandez, and R. Garcia-Martinez. Finding

- Optimal Neural Network Architecture Using Genetic Algorithms. *Advances in Computer Science and Engineering Research in Computer Science*, 27:15–24, 2007.
- S. W. Fleming, D. R. Bourdin, D. Campbell, R. B. Stull, and T. Gardner. Development and operational testing of a super-ensemble artificial intelligence flood-forecast model for a pacific northwest river. *Journal of the American Water Resources Association*, 51(2):502–512, Apr. 2015. ISSN 17521688. doi: 10.1111/jawr.12259.
- M. Frank, D. Drikakis, and V. Charissis. Machine-Learning Methods for Computational Science and Engineering. *Computation*, 8(1):15, Mar. 2020. ISSN 2079-3197. doi: 10.3390/computation8010015.
- G. Freni, G. Mannina, and G. Viviani. Urban runoff modelling uncertainty: Comparison among Bayesian and pseudo-Bayesian methods. *Environmental Modelling & Software*, 24(9):1100–1111, Sept. 2009. ISSN 13648152. doi: 10.1016/j.envsoft.2009.03.003.
- Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. In *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 904, pages 23–37. 1995. ISBN 978-3-540-59119-1. doi: 10.1007/3-540-59119-2_166.
- Y. Freund and R. E. Schapire. Experiments with a New Boosting Algorithm. In *Proceedings of the 13th International Conference on Machine Learning*, pages 148–156, 1996. ISBN 1-55860-419-7.
- J. H. Friedman. Greedy Function Approximation: A Gradient Boosting Machine. *Annals of Statistics*, 29(5):1189–1232, 2000.
- M. Galar, A. Fernandez, E. Barrenechea, H. Bustince, and F. Herrera. A Review on Ensembles for the Class Imbalance Problem: Bagging-, Boosting-, and Hybrid-Based Approaches. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 42(4):463–484, July 2012. ISSN 1094-6977. doi: 10.1109/TSMCC.2011.2161285.
- M. Galar, A. Fernández, E. Barrenechea, and F. Herrera. EUSBoost: Enhancing ensembles for highly imbalanced data-sets by evolutionary undersampling. *Pattern Recognition*, 46(12):3460–3471, Dec. 2013. ISSN 00313203. doi: 10.1016/j.patcog.2013.05.006.

- A. Gallice, B. Schaeffli, M. Lehning, M. B. Parlange, and H. Huwald. Stream temperature prediction in ungauged basins: Review of recent approaches and description of a new physics-derived statistical model. *Hydrology and Earth System Sciences*, 19(9):3727–3753, Sept. 2015. ISSN 16077938. doi: 10.5194/hess-19-3727-2015.
- M. A. Ganaie, M. Hu, A. K. Malik, M. Tanveer, and P. N. Suganthan. Ensemble deep learning: A review. *Engineering Applications of Artificial Intelligence*, 115:105151, Oct. 2022. ISSN 0952-1976. doi: 10.1016/j.engappai.2022.105151.
- M. Gauch, J. Mai, and J. Lin. The proper care and feeding of CAMELS: How limited training data affects streamflow prediction. *Environmental Modelling & Software*, 135:104926, Jan. 2021. ISSN 13648152. doi: 10.1016/j.envsoft.2020.104926.
- L. Giriagama, M. Naveed Khaliq, P. Lamontagne, J. Perdikaris, R. Roy, L. Sushama, and A. Elshorbagy. Streamflow modelling and forecasting for Canadian watersheds using LSTM networks with attention mechanism. *Neural Computing and Applications*, 2022. ISSN 14333058. doi: 10.1007/s00521-022-07523-8.
- Z. Gong, P. Zhong, and W. Hu. Diversity in Machine Learning. *IEEE Access*, 7:64323–64350, 2019. ISSN 21693536. doi: 10.1109/ACCESS.2019.2917620.
- Government of Canada. High Resolution Digital Elevation Model (HRDEM). <https://open.canada.ca/data/en/dataset/957782bf-847c-4644-a757-e383c0057995>, 2014.
- Government of Canada. Open Government. <https://open.canada.ca/en>, 2020.
- Government of Canada and Public Safety Canada. Floods. <https://www.publicsafety.gc.ca/cnt/mrgnc-mngmnt/ntrl-hzrds/fld-en.aspx>, 2021.
- R. S. Govindaraju. Artificial Neural Networks in Hydrology. II: Hydrologic Applications. *Journal of Hydrologic Engineering*, 5(2):124–137, 2000-04-31. ISSN 1084-0699. doi: 10.1061/(ASCE)1084-0699(2000)5:2(124).
- M. Grum, A. Jørgensen, R. Johansen, and J. Linde. The effect of climate change on urban drainage: An evaluation based on regional climate model simulations. *Water Science and Technology*, 54(6-7):9–15, Sept. 2006. ISSN 0273-1223. doi: 10.2166/wst.2006.592.

- D. Guo, F. Zheng, H. Gupta, and H. R. Maier. On the Robustness of Conceptual Rainfall-Runoff Models to Calibration and Evaluation Data Set Splits Selection: A Large Sample Investigation. *Water Resources Research*, 56(3), Mar. 2020. ISSN 0043-1397. doi: 10.1029/2019WR026752.
- H. V. Gupta, H. Kling, K. K. Yilmaz, and G. F. Martinez. Decomposition of the mean squared error and NSE performance criteria: Implications for improving hydrological modelling. *Journal of Hydrology*, 377(1-2):80–91, June 2009. ISSN 00221694. doi: 10.1016/j.jhydrol.2009.08.003.
- F. A. A. Hadi, L. Mohd Sidek, G. H. Ahmed Salih, H. Basri, S. S. Sammen, N. Mohd Dom, Z. Muhamad Ali, and A. Najah Ahmed. Machine learning techniques for flood forecasting. *Journal of Hydroinformatics*, page jh2024208, 2024.
- G. Haixiang, L. Yijing, J. Shang, G. Mingyun, H. Yuanyue, and G. Bing. Learning from class-imbalanced data: Review of methods and applications. *Expert Systems with Applications*, 73:220–239, May 2017. ISSN 09574174. doi: 10.1016/j.eswa.2016.12.035.
- D. K. Hakim, R. Gernowo, and A. W. Nirwansyah. Flood prediction with time series data mining: Systematic review. *Natural Hazards Research*, 2023.
- Hanchuan Peng, Fuhui Long, and C. Ding. Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Trans. Pattern Anal. Mach. Intell.*, 27(8):1226–1238, aug 2005. doi: 10.1109/TPAMI.2005.159.
- R. Hashemi, P. Brigode, P.-A. Garambois, and P. Javelle. How can we benefit from regime information to make more effective use of long short-term memory (LSTM) runoff models? *Hydrology and Earth System Sciences*, 26(22):5793–5816, Nov. 2022. ISSN 1027-5606. doi: 10.5194/hess-26-5793-2022.
- T. Hastie, R. Tibshirani, and J. Friedman. *Elements of Statistical Learning 2nd Ed.* Number 2. Springer New York, New York, NY, 2009. ISBN 978-0-387-84857-0. doi: 10.1007/978-0-387-84858-7.
- J. He, C. Valeo, A. Chu, and N. F. Neumann. Prediction of event-based stormwater runoff quantity and quality by ANNs developed using PMI-based input selection. *Journal of Hy-*

- drology*, 400(1-2):10–23, July 2011. ISSN 00221694. doi: 10.1016/j.jhydrol.2011.01.024.
- A. Hipni, A. El-shafie, A. Najah, O. A. Karim, A. Hussain, and M. Mukhlisin. Daily Forecasting of Dam Water Levels: Comparing a Support Vector Machine (SVM) Model With Adaptive Neuro Fuzzy Inference System (ANFIS). *Water Resources Management*, 27(10):3803–3823, Aug. 2013. ISSN 0920-4741. doi: 10.1007/s11269-013-0382-4.
- M. Höge, T. Wöhling, and W. Nowak. A Primer for Model Selection: The Decisive Role of Model Complexity. *Water Resources Research*, 54(3):1688–1715, Mar. 2018. ISSN 0043-1397. doi: 10.1002/2017WR021902.
- M. Höge, A. Scheidegger, M. Baity-Jesi, C. Albert, and F. Fenicia. Improving hydrologic models for predictions and process understanding using neural ODEs. *Hydrology and Earth System Sciences*, 26(19):5085–5102, Oct. 2022. ISSN 1607-7938. doi: 10.5194/hess-26-5085-2022.
- P. Horton, B. Schaeffli, and M. Kauzlaric. Why do we have so many different hydrological models? A review based on the case of Switzerland. *WIREs Water*, 9(1):e1574, 2022. ISSN 2049-1948. doi: 10.1002/wat2.1574.
- S. Hossain, G. A. Hewa, and S. Wella-Hewage. A Comparison of Continuous and Event-Based Rainfall-Runoff (RR) Modelling Using EPA-SWMM. *Water*, 11(3):611, Mar. 2019. ISSN 2073-4441. doi: 10.3390/w11030611.
- L. H. C. Hsieh. How does the effect fade over distance? An inquiry into the decay pattern of distance effect on property values in the case of Taipei, Taiwan. *Land*, 10(11), 2021. ISSN 2073445X. doi: 10.3390/land10111238.
- K.-l. Hsu, H. V. Gupta, and S. Sorooshian. Artificial Neural Network Modeling of the Rainfall-Runoff Process. *Water Resources Research*, 31(10):2517–2530, Oct. 1995. ISSN 00431397. doi: 10.1029/95WR01955.
- G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew. Extreme learning machine: Theory and applications. *Neurocomputing*, 70(1-3):489–501, Dec. 2006. ISSN 09252312. doi: 10.1016/j.neucom.2005.12.126.
- W. James. *Rules for Responsible Modeling*. Computational Hydraulics International, Guelph,

- Canada, 2005. ISBN 0-9683681-5-8.
- W. James, L. E. Rossman, W. Robert, and C. James. *User's Guide to SWMM5*. Computational Hydraulics International, Guelph, 2010. ISBN 978-0-9808853-5-4.
- M.-È. Jean, S. Duchesne, G. Pelletier, and M. Pleau. Selection of rainfall information as input data for the design of combined sewer overflow solutions. *Journal of Hydrology*, 565:559–569, Oct. 2018. ISSN 00221694. doi: 10.1016/j.jhydrol.2018.08.064.
- J.-H. Jeon, C.-G. Park, and B. Engel. Comparison of Performance between Genetic Algorithm and SCE-UA for Calibration of SCS-CN Surface Runoff Simulation. *Water*, 6(11):3433–3456, Nov. 2014. ISSN 2073-4441. doi: 10.3390/w6113433.
- A. Jin, Q. Wang, H. Zhan, and R. Zhou. Comparative Performance Assessment of Physical-Based and Data-Driven Machine-Learning Models for Simulating Streamflow: A Case Study in Three Catchments across the US. *Journal of Hydrologic Engineering*, 29(2):05024004, Apr. 2024. ISSN 1084-0699, 1943-5584. doi: 10.1061/JHYEFF.HEENG-6118.
- B. Jongman, J. Wagemaker, B. Romero, and E. de Perez. Early Flood Detection for Rapid Humanitarian Response: Harnessing Near Real-Time Satellite and Twitter Signals. *ISPRS International Journal of Geo-Information*, 4(4):2246–2266, Oct. 2015. ISSN 2220-9964. doi: 10.3390/ijgi4042246.
- J. Joo, J. Lee, J. Kim, H. Jun, and D. Jo. Inter-Event Time Definition Setting Procedure for Urban Drainage Systems. *Water*, 6(1):45–58, Dec. 2013. ISSN 2073-4441. doi: 10.3390/w6010045.
- S. Kambalimath S and P. C. Deka. Performance enhancement of SVM model using discrete wavelet transform for daily streamflow forecasting. *Environmental Earth Sciences*, 80(3):101, 2021. ISSN 18666299. doi: 10.1007/s12665-021-09394-z.
- M. Karamouz, R. Rahimi, and E. Ebrahimi. Uncertain Water Balance-Based Sustainability Index of Supply and Demand. *Journal of Water Resources Planning and Management*, 147(5):04021015, 2021. ISSN 0733-9496. doi: 10.1061/(asce)wr.1943-5452.0001351.
- K. S. Kasiviswanathan, J. He, K. P. Sudheer, and J. H. Tay. Potential application of wavelet

- neural network ensemble to forecast streamflow for flood management. *Journal of Hydrology*, 536:161–173, Apr. 2016. ISSN 00221694. doi: 10.1016/j.jhydrol.2016.02.044.
- A. Kauffeldt, F. Wetterhall, F. Pappenberger, P. Salamon, and J. Thielen. Technical review of large-scale hydrological models for implementation in operational flood forecasting schemes on continental level. *Environmental Modelling and Software*, 75:68–76, Jan. 2016. ISSN 13648152. doi: 10.1016/j.envsoft.2015.09.009.
- B. Khakbaz, B. Imam, K. Hsu, and S. Sorooshian. From lumped to distributed via semi-distributed: Calibration strategies for semi-distributed hydrologic models. *Journal of Hydrology*, 418–419:61–77, Feb. 2012. ISSN 00221694. doi: 10.1016/j.jhydrol.2009.02.021.
- U. T. Khan and C. Valeo. Short-Term Peak Flow Rate Prediction and Flood Risk Assessment Using Fuzzy Linear Regression. *Journal of Environmental Informatics*, 28(2):71–89, May 2016. ISSN 17262135. doi: 10.3808/jei.201600345.
- U. T. Khan, J. He, and C. Valeo. River flood prediction using fuzzy neural networks: An investigation on automated network architecture. *Water Science and Technology*, 2017(1): 238–247, Apr. 2018. ISSN 0273-1223. doi: 10.2166/wst.2018.107.
- P. K. Kitanidis and R. L. Bras. Real-time forecasting with a conceptual hydrologic model: 2. Applications and results. *Water Resources Research*, 16(6):1034–1044, Dec. 1980. ISSN 19447973. doi: 10.1029/WR016i006p01034.
- J. Knighton, E. White, E. Lennon, and R. Rajan. Development of probability distributions for urban hydrologic model parameters and a Monte Carlo analysis of model sensitivity. *Hydrological Processes*, 28(19):5131–5139, Sept. 2014. ISSN 08856087. doi: 10.1002/hyp.10009.
- J. Knighton, E. Lennon, L. Bastidas, and E. White. Stormwater detention system parameter sensitivity and uncertainty analysis using SWMM. *Journal of Hydrologic Engineering*, 21(8):05016014, Aug. 2016. ISSN 10840699. doi: 10.1061/(ASCE)HE.1943-5584.0001382.
- W. J. Knoben, J. E. Freer, and R. A. Woods. Technical note: Inherent benchmark or not? Comparing Nash-Sutcliffe and Kling-Gupta efficiency scores. *Hydrology and Earth System Sciences*, 23(10):4323–4331, Oct. 2019. ISSN 16077938. doi: 10.5194/hess-23-4323-2019.
- F. Kratzert, D. Klotz, M. Herrnegger, A. K. Sampson, S. Hochreiter, and G. S. Nearing. To-

- ward Improved Predictions in Ungauged Basins: Exploiting the Power of Machine Learning. *Water Resources Research*, 55(12):11344–11354, 2019a. ISSN 1944-7973. doi: 10.1029/2019WR026065.
- F. Kratzert, D. Klotz, G. Shalev, G. Klambauer, S. Hochreiter, and G. Nearing. Towards learning universal, regional, and local hydrological behaviors via machine learning applied to large-sample datasets. *Hydrology and Earth System Sciences*, 23(12):5089–5110, Dec. 2019b. ISSN 1607-7938. doi: 10.5194/hess-23-5089-2019.
- F. Kratzert, M. Gauch, D. Klotz, and G. Nearing. HESS Opinions: Never train an LSTM on a single basin. *Hydrology and Earth System Sciences Discussions*, pages 1–19, Jan. 2024. doi: 10.5194/hess-2023-275.
- G. Krebs, T. Kokkonen, M. Valtanen, H. Koivusalo, and H. Setälä. A high resolution application of a stormwater management model (SWMM) using genetic parameter optimization. *Urban Water Journal*, 10(6):394–410, Dec. 2013. ISSN 1573062X. doi: 10.1080/1573062X.2012.739631.
- S. Kubat, Miroslav; Matwin. Addressing the Curse of Imbalanced Training Sets: One-Sided Selection. *Fourteenth Int. Conf. Mach. Learn.*, 97:179–186, 1997.
- A. Kumar, R. Singh, P. P. Jena, C. Chatterjee, and A. Mishra. Identification of the best multi-model combination for simulating river discharge. *Journal of Hydrology*, 525:313–325, June 2015. ISSN 00221694. doi: 10.1016/j.jhydrol.2015.03.060.
- S. Kumar, D. R. Kaushal, and A. K. Gosain. Evaluation of evolutionary algorithms for the optimization of storm water drainage network for an urbanized area. *Acta Geophysica*, 67(1):149–165, 2019. ISSN 18957455. doi: 10.1007/s11600-018-00240-8.
- C. Kurian, K. Sudheer, V. Krishna Vema, and D. Sahoo. Effective flood forecasting at higher lead times through hybrid modelling framework. *Journal of Hydrology*, page 124945, Apr. 2020. ISSN 00221694. doi: 10.1016/j.jhydrol.2020.124945.
- N. Lauzon, F. Anctil, and C. W. Baxter. Clustering of heterogeneous precipitation fields for the assessment and possible improvement of lumped neural network models for streamflow forecasts. *Hydrology and Earth System Sciences*, 10(4):485–494, July 2006. ISSN 1607-7938.

doi: 10.5194/hess-10-485-2006.

- T. Lees, M. Buechel, B. Anderson, L. Slater, S. Reece, G. Coxon, and S. J. Dadson. Benchmarking data-driven rainfall-runoff models in Great Britain: A comparison of long short-term memory (LSTM)-based models with four lumped conceptual models. *Hydrology and Earth System Sciences*, 25(10):5517–5534, Oct. 2021. ISSN 1607-7938. doi: 10.5194/hess-25-5517-2021.
- T. Lees, S. Reece, F. Kratzert, D. Klotz, M. Gauch, J. De Bruijn, R. Kumar Sahu, P. Greve, L. Slater, and S. J. Dadson. Hydrological concept formation inside long short-term memory (LSTM) networks. *Hydrology and Earth System Sciences*, 26(12):3079–3101, 2022. ISSN 16077938. doi: 10.5194/hess-26-3079-2022.
- D. Leutnant, A. Döring, and M. Uhl. Swmmr - an R package to interface SWMM. *Urban Water Journal*, 16(1):68–76, Jan. 2019. ISSN 17449006. doi: 10.1080/1573062X.2019.1611889.
- J. Li, C. Zhang, X. Zhang, H. He, W. Liu, and C. Chen. Temperature Compensation of Piezo-Resistive Pressure Sensor Utilizing Ensemble AMPSO-SVR Based on Improved Adaboost.RT. *IEEE Access*, 8:12413–12425, 2020. ISSN 2169-3536. doi: 10.1109/ACCESS.2020.2965150.
- S. Li, K. Ma, Z. Jin, and Y. Zhu. A new flood forecasting model based on SVM and boosting learning algorithms. In *2016 IEEE Congress on Evolutionary Computation (CEC)*, pages 1343–1348. IEEE, July 2016. ISBN 978-1-5090-0623-6. doi: 10.1109/CEC.2016.7743944.
- Z. Li, Q. Shao, Z. Xu, and X. Cai. Analysis of parameter uncertainty in semi-distributed hydrological models using bootstrap method: A case study of SWAT model applied to Yingluoxia watershed in northwest China. *Journal of Hydrology*, 385(1-4):76–83, 2010. ISSN 00221694. doi: 10.1016/j.jhydrol.2010.01.025.
- J. B. Lindsay. The practice of DEM stream burning revisited. *Earth Surface Processes and Landforms*, 41(5):658–668, Apr. 2016. ISSN 10969837. doi: 10.1002/esp.3888.
- S. Liu, J. Xu, J. Zhao, X. Xie, and W. Zhang. Efficiency enhancement of a process-based rainfall-runoff model using a new modified AdaBoost.RT technique. *Applied Soft Computing*, 23:521–529, Oct. 2014. ISSN 15684946. doi: 10.1016/j.asoc.2014.05.033.

- Y. Liu, G. Hou, F. Huang, H. Qin, B. Wang, and L. Yi. Directed graph deep neural network for multi-step daily streamflow forecasting. *Journal of Hydrology*, 607, 2022. ISSN 00221694. doi: 10.1016/j.jhydrol.2022.127515.
- E. López, C. Vionnet, D. Villarreal, G. Contini, E. Veizaga, and C. Ferreira. A cost-effective redundant communication system for improving the reliability of a flood early warning system. *Journal of Hydroinformatics*, 22(4):856–875, 2020.
- V. López, A. Fernández, S. García, V. Palade, and F. Herrera. An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics. *Information Sciences*, 250:113–141, Nov. 2013. ISSN 00200255. doi: 10.1016/j.ins.2013.07.007.
- J. G. MacKinnon. Critical Values For Cointegration Tests. *Working Paper*, (1227), Jan. 2010.
- J. Mai, H. Shen, B. A. Tolson, É. Gaborit, R. Arsenault, J. R. Craig, V. Fortin, L. M. Fry, M. Gauch, D. Klotz, F. Kratzert, N. O’Brien, D. G. Princz, S. Rasiya Koya, T. Roy, F. Seglenieks, N. K. Shrestha, A. G. T. Temgoua, V. Vionnet, and J. W. Waddell. The Great Lakes Runoff Intercomparison Project Phase 4: The Great Lakes (GRIP-GL). *Hydrology and Earth System Sciences*, 26(13):3537–3572, July 2022. ISSN 1027-5606. doi: 10.5194/hess-26-3537-2022.
- H. R. Maier and G. C. Dandy. Neural networks for the prediction and forecasting of water resources variables: A review of modelling issues and applications. *Environmental Modelling and Software*, 15(1):101–124, Apr. 2000. ISSN 13648152. doi: 10.1016/S1364-8152(99)00007-9.
- H. R. Maier, A. Jain, G. C. Dandy, and K. Sudheer. Methods used for the development of neural networks for the prediction of water resource variables in river systems: Current status and future directions. *Environmental Modelling & Software*, 25(8):891–909, Aug. 2010. ISSN 13648152. doi: 10.1016/j.envsoft.2010.02.003.
- I. Maqsood, M. Khan, and A. Abraham. An ensemble of neural networks for weather forecasting. *Neural Computing and Applications*, 13(2):112–122, June 2004. ISSN 0941-0643. doi: 10.1007/s00521-004-0413-4.

- L. Masisi, F. V. Nelwamondo, and T. Marwala. The effect of structural diversity of an ensemble of classifiers on classification accuracy. In *IASTED International Conference on Modelling and Simulation (Africa-MS)*, pages 1–6, 2008.
- MATLAB. Genetic Algorithm. <https://www.mathworks.com/help/gads/ga.html>, 2021.
- R. May, G. Dandy, and H. Maier. Review of Input Variable Selection Methods for Artificial Neural Networks. In *Artificial Neural Networks - Methodological Advances and Biomedical Applications*. InTech, Apr. 2011. ISBN 978-953-307-243-2. doi: 10.5772/16004.
- B. McDonnell, K. Ratliff, M. Tryby, J. Wu, and A. Mullapudi. PySWMM: The Python Interface to Stormwater Management Model (SWMM). *Journal of Open Source Software*, 5(52):2292, Aug. 2020. ISSN 2475-9066. doi: 10.21105/joss.02292.
- L. Mediero, L. Garrote, and F. J. Martín-Carrasco. Probabilistic calibration of a distributed hydrological model for flood forecasting. *Hydrological Sciences Journal*, 56(7):1129–1149, Oct. 2011. ISSN 0262-6667. doi: 10.1080/02626667.2011.610322.
- A. Mentzafou, A. Papadopoulos, and E. Dimitriou. Flood generating mechanisms investigation and rainfall threshold identification for regional flood early warning. *Environmental Earth Sciences*, 82(10):242, 2023.
- L. Merino-Martín, M. Moreno-de las Heras, S. Pérez-Domingo, T. Espigares, and J. M. Nicolau. Hydrological heterogeneity in Mediterranean reclaimed slopes: Runoff and sediment yield at the patch and slope scales along a gradient of overland flow. *Hydrology and Earth System Sciences*, 16(5):1305–1320, May 2012. ISSN 1607-7938. doi: 10.5194/hess-16-1305-2012.
- Q. Miao, D. Yang, H. Yang, and Z. Li. Establishing a rainfall threshold for flash flood warnings in China’s mountainous areas based on a distributed hydrological model. *Journal of Hydrology*, 541:371–386, Oct. 2016. ISSN 00221694. doi: 10.1016/j.jhydrol.2016.04.054.
- N. Mizukami, O. Rakovec, A. J. Newman, M. P. Clark, A. W. Wood, H. V. Gupta, and R. Kumar. On the choice of calibration metrics for ”high-flow” estimation using hydrologic models. *Hydrology and Earth System Sciences*, 23(6):2601–2614, June 2019. ISSN 16077938. doi: 10.5194/hess-23-2601-2019.

- N. Moniz, P. Branco, and L. Torgo. Resampling strategies for imbalanced time series forecasting. *International Journal of Data Science and Analytics*, 3(3):161–181, May 2017a. ISSN 2364-415X. doi: 10.1007/s41060-017-0044-3.
- N. Moniz, P. Branco, L. Torgo, and B. Krawczyk. Evaluation of Ensemble Methods in Imbalanced Regression Tasks. In *Proceedings of Machine Learning Research*, volume 74, pages 129–140, 2017b.
- N. Moniz, R. Ribeiro, V. Cerqueira, and N. Chawla. SMOTEBoost for Regression: Improving the Prediction of Extreme Values. In *2018 IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA)*, pages 150–159. IEEE, Oct. 2018. ISBN 978-1-5386-5090-5. doi: 10.1109/DSAA.2018.00025.
- D. G. Morris and R. G. Heerdegen. Automatically derived catchment boundaries and channel networks and their hydrological applications. *Geomorphology*, 1(2):131–141, 1988. ISSN 0169555X. doi: 10.1016/0169-555X(88)90011-6.
- G. Morse and K. O. Stanley. Simple Evolutionary Optimization Can Rival Stochastic Gradient Descent in Neural Networks. In *Proceedings of the Genetic and Evolutionary Computation Conference 2016*, pages 477–484, Denver Colorado USA, July 2016. ACM. ISBN 978-1-4503-4206-3. doi: 10.1145/2908812.2908916.
- H. Mosaffa, M. Sadeghi, I. Mallakpour, M. Naghdizadegan Jahromi, and H. R. Pourghasemi. Chapter 43 - Application of machine learning algorithms in hydrology. In H. R. Pourghasemi, editor, *Computers in Earth and Environmental Sciences*, pages 585–591. Elsevier, Jan. 2022. ISBN 978-0-323-89861-4. doi: 10.1016/B978-0-323-89861-4.00027-0.
- A. Mosavi, P. Ozturk, and K.-w. Chau. Flood Prediction Using Machine Learning Models: Literature Review. *Water*, 10(11):1536, Oct. 2018. ISSN 2073-4441. doi: 10.3390/w10111536.
- C. I. Mulvihill, B. P. Baldigo, S. J. Miller, D. DeKoskie, and J. DuBois. *Bankfull Discharge and Channel Characteristics of Streams in New York State*. U.S. Department of the Interior, U.S. Geological Survey, Virginia, 2009. ISBN 2009-5144.
- H. S. Munawar, A. W. Hammad, and S. T. Waller. A review on flood management technolo-

- gies related to image processing and machine learning. *Automation in Construction*, 132, 2021. ISSN 09265805. doi: 10.1016/j.autcon.2021.103916.
- P. Muñoz, J. Orellana-Alvear, J. Bendix, J. Feyen, and R. Céleri. Flood early warning systems using machine learning techniques: The case of the tomebamba catchment at the southern Andes of Ecuador. *Hydrology*, 8(4), 2021. ISSN 23065338. doi: 10.3390/hydrology8040183.
- D. T. Myers, D. L. Ficklin, S. M. Robeson, R. P. Neupane, A. Botero-Acosta, and P. M. Avelaneda. Choosing an arbitrary calibration period for hydrologic models: How much does it influence water balance simulations? *Hydrological Processes*, 35(2):e14045, 2021. ISSN 1099-1085. doi: 10.1002/hyp.14045.
- T. Nanda, B. Sahoo, H. Beria, and C. Chatterjee. A wavelet-based non-linear autoregressive with exogenous inputs (WNARX) dynamic neural network model for real-time flood forecasting using satellite-based rainfall products. *Journal of Hydrology*, 539:57–73, Aug. 2016. ISSN 00221694. doi: 10.1016/j.jhydrol.2016.05.014.
- G. Nearing, D. Cohen, V. Dube, M. Gauch, O. Gilon, S. Harrigan, A. Hassidim, D. Klotz, F. Kratzert, A. Metzger, S. Nevo, F. Pappenberger, C. Prudhomme, G. Shalev, S. Shenzis, T. Tekalign, D. Weitzner, and Y. Matias. AI Increases Global Access to Reliable Flood Forecasts, Nov. 2023.
- G. S. Nearing, F. Kratzert, A. K. Sampson, C. S. Pelissier, D. Klotz, J. M. Frame, C. Prieto, and H. V. Gupta. What Role Does Hydrological Science Play in the Age of Machine Learning? *Water Resources Research*, 57(3):e2020WR028091, Mar. 2021. ISSN 19447973. doi: 10.1029/2020WR028091.
- G. S. Nearing, D. Klotz, J. M. Frame, M. Gauch, O. Gilon, F. Kratzert, A. K. Sampson, G. Shalev, and S. Nevo. Technical note: Data assimilation and autoregression for using near-real-time streamflow observations in long short-term memory networks. *Hydrology and Earth System Sciences*, 26(21):5493–5513, Nov. 2022. ISSN 1607-7938. doi: 10.5194/hess-26-5493-2022.
- S. Nevo, E. Morin, A. Gerzi Rosenthal, A. Metzger, C. Barshai, D. Weitzner, D. Voloshin, F. Kratzert, G. Elidan, G. Dror, G. Begelman, G. Nearing, G. Shalev, H. Noga, I. Shavitt,

- L. Yuklea, M. Royz, N. Giladi, N. Peled Levi, O. Reich, O. Gilon, R. Maor, S. Timnat, T. Shechter, V. Anisimov, Y. Gigi, Y. Levin, Z. Moshe, Z. Ben-Haim, A. Hassidim, and Y. Matias. Flood forecasting with machine learning models in an operational framework. *Hydrology and Earth System Sciences*, 26(15):4013–4032, Aug. 2022. ISSN 1607-7938. doi: 10.5194/hess-26-4013-2022.
- H. D. Nguyen, D. K. Dang, N. Y. Nguyen, C. Pham Van, T. T. Van Nguyen, Q.-H. Nguyen, X. L. Nguyen, L. T. Pham, V. T. Pham, and Q.-T. Bui. Integration of machine learning and hydrodynamic modeling to solve the extrapolation problem in flood depth estimation. *Journal of Water and Climate Change*, 15(1):284–304, Dec. 2023. ISSN 2040-2244. doi: 10.2166/wcc.2023.573.
- L. D. Nguyen, H. T. Nguyen, P. D. Dang, T. Q. Duong, and L. K. Nguyen. Design of an automatic hydro-meteorological observation network for a real-time flood warning system: a case study of vu gia-thu bon river basin, vietnam. *Journal of Hydroinformatics*, 23(2):324–339, 2021.
- L. Ni, D. Wang, J. Wu, Y. Wang, Y. Tao, J. Zhang, and J. Liu. Streamflow forecasting using extreme gradient boosting model coupled with Gaussian mixture model. *Journal of Hydrology*, 586:124901, July 2020. ISSN 00221694. doi: 10.1016/j.jhydrol.2020.124901.
- M. Niazi, C. Nietch, M. Maghrebi, N. Jackson, B. R. Bennett, M. Tryby, and A. Massoudieh. Storm Water Management Model: Performance Review and Gap Analysis. *Journal of Sustainable Water in the Built Environment*, 3(2):04017002, May 2017. ISSN 2379-6111. doi: 10.1061/JSWBAY.0000817.
- J. Niemczynowicz. Urban hydrology and water management – present and future challenges. *Urban Water*, 1(1):1–14, Mar. 1999. ISSN 1462-0758. doi: 10.1016/S1462-0758(99)00009-6.
- N. Nirupama, C. Armenakis, and M. Montpetit. Is flooding in Toronto a concern? *Natural Hazards*, 72(2):1259–1264, 2014. ISSN 0921030X. doi: 10.1007/s11069-014-1054-2.
- N. S. Nojumuddin, F. Yusof, and Z. Yusop. Determination of minimum inter-event time for storm characterisation in Johor, Malaysia. *Journal of Flood Risk Management*, 11:S687–S699, Feb. 2018. ISSN 1753318X. doi: 10.1111/jfr3.12242.

- J. F. O’Callaghan and D. M. Mark. The extraction of drainage networks from digital elevation data. *Computer Vision, Graphics, & Image Processing*, 28(3):323–344, Dec. 1984. ISSN 0734189X. doi: 10.1016/S0734-189X(84)80011-0.
- D. Opitz and R. Maclin. Popular Ensemble Methods: An Empirical Study. *Journal of Artificial Intelligence Research*, 11:169–198, Aug. 1999. ISSN 1076-9757. doi: 10.1613/jair.614.
- D. W. Opitz and R. F. MacLin. An empirical evaluation of bagging and boosting for artificial neural networks. In *IEEE International Conference on Neural Networks - Conference Proceedings*, volume 3, pages 1401–1405. IEEE, 1997. ISBN 0-7803-4122-8. doi: 10.1109/ICNN.1997.613999.
- T. B. M. J. Ouarda and C. Shu. Regional low-flow frequency analysis using single and ensemble artificial neural networks. *Water Resources Research*, 45(11), Nov. 2009. ISSN 00431397. doi: 10.1029/2008WR007196.
- L. Oudin, A. Kay, V. Andréassian, and C. Perrin. Are seemingly physically similar catchments truly hydrologically similar? *Water Resources Research*, 46(11), 2010. ISSN 1944-7973. doi: 10.1029/2009WR008887.
- Y. O. Ouma, R. Cheruyot, and A. N. Wachera. Rainfall and runoff time-series trend analysis using LSTM recurrent neural network and wavelet neural network with satellite-based meteorological data: Case study of Nzoia hydrologic basin. *Complex and Intelligent Systems*, 8(1):213–236, 2022. ISSN 21986053. doi: 10.1007/s40747-021-00365-2.
- S. Padiyedath Gopalan, A. Kawamura, H. Amaguchi, T. Takasaki, and G. Azhikodan. A bootstrap approach for the parameter uncertainty of an urban-specific rainfall-runoff model. *Journal of Hydrology*, 579:124195, Dec. 2019. ISSN 00221694. doi: 10.1016/j.jhydrol.2019.124195.
- C. Paniconi and M. Putti. Physically based modeling in catchment hydrology at 50: Survey and outlook. *Water Resources Research*, 51(9):7090–7129, 2015. ISSN 1944-7973. doi: 10.1002/2015WR017780.
- G. Papacharalampous, H. Tyralis, A. Langousis, A. W. Jayawardena, B. Sivakumar, N. Mamassis, A. Montanari, and D. Koutsoyiannis. Probabilistic Hydrological Post-Processing at

- Scale: Why and How to Apply Machine-Learning Quantile Regression Algorithms. *Water*, 11(10):2126, Oct. 2019. ISSN 2073-4441. doi: 10.3390/w11102126.
- I. G. Pechlivanidis, B. M. Jackson, N. R. McIntyre, and H. S. Wheeler. Catchment scale hydrological modelling: A review of model types, calibration approaches and uncertainty analysis methods in the context of recent developments in technology and applications. *Global Nest Journal*, 13(3):193–214, 2011.
- B. Pengel, V. Krzhizhanovskaya, N. Melnikova, G. Shirshov, A. Koelewijn, A. Pyayt, I. Mokhov, et al. Flood early warning system: sensors and internet. *IAHS Red Book*, 357: 445–453, 2013.
- R. Perin, M. Trigatti, M. Nicolini, M. Campolo, and D. Goi. Automated calibration of the EPA-SWMM model for a small suburban catchment using PEST: A case study. *Environmental Monitoring and Assessment*, 192(6):374, June 2020. ISSN 0167-6369. doi: 10.1007/s10661-020-08338-7.
- C. Perrin, C. Michel, and V. Andréassian. Improvement of a parsimonious model for stream-flow simulation. *Journal of Hydrology*, 279(1-4):275–289, Aug. 2003. ISSN 00221694. doi: 10.1016/S0022-1694(03)00225-7.
- I. Pisa, I. Santín, J. L. Vicario, A. Morell, and R. Vilanova. Data preprocessing for ANN-based industrial time-series forecasting with imbalanced data. In *European Signal Processing Conference*, volume 2019-Sept. European Signal Processing Conference, EUSIPCO, Sept. 2019. ISBN 978-90-827970-3-9. doi: 10.23919/EUSIPCO.2019.8902682.
- Province of Ontario and Ministry of Natural Resources and Forestry. Ontario GeoHub. <https://geohub.lio.gov.on.ca/documents/0279f65b82314121b5b5ec93d76bc6ba/about>, 2022.
- H.-p. Qin, Z.-x. Li, and G. Fu. The effects of low impact development on urban flooding under different rainfall characteristics. *Journal of Environmental Management*, 129:577–585, Nov. 2013. ISSN 03014797. doi: 10.1016/j.jenvman.2013.08.026.
- S. Raghavendra. N and P. C. Deka. Support vector machine applications in the field of hydrology: A review. *Applied Soft Computing*, 19:372–386, June 2014. ISSN 15684946. doi: 10.1016/j.asoc.2014.02.002.

- Z. Rasheed, A. Aravamudan, A. Gorji Sefidmazgi, G. C. Anagnostopoulos, and E. I. Nikolopoulos. Advancing flood warning procedures in ungauged basins with machine learning. *Journal of Hydrology*, 609:127736, June 2022. ISSN 00221694. doi: 10.1016/j.jhydrol.2022.127736.
- N. Razali, S. Ismail, and A. Mustapha. Machine learning approach for flood risks prediction. *IAES International Journal of Artificial Intelligence*, 9(1):73–80, 2020. ISSN 22528938. doi: 10.11591/ijai.v9.i1.pp73-80.
- K. Ren, W. Fang, J. Qu, X. Zhang, and X. Shi. Comparison of eight filter-based feature selection methods for monthly streamflow forecasting – Three case studies on CAMELS data sets. *Journal of Hydrology*, 586:124897, July 2020. ISSN 00221694. doi: 10.1016/j.jhydrol.2020.124897.
- D. Rincón, U. Khan, and C. Armenakis. Flood Risk Mapping Using GIS and Multi-Criteria Analysis: A Greater Toronto Area Case Study. *Geosciences*, 8(8):275, July 2018. ISSN 2076-3263. doi: 10.3390/geosciences8080275.
- L. A. Rossman. *Storm Water Management Model User’s Manual Version 5.1*. United States Environmental Protection Agency, Cincinnati, 2015.
- P. J. Rousseeuw. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53–65, Nov. 1987. ISSN 0377-0427. doi: 10.1016/0377-0427(87)90125-7.
- D. Ruelland, S. Ardoin-Bardin, G. Billen, and E. Servat. Sensitivity of a lumped and semi-distributed hydrological model to several methods of rainfall interpolation on a large basin in West Africa. *Journal of Hydrology*, 361(1-2):96–117, Oct. 2008. ISSN 00221694. doi: 10.1016/j.jhydrol.2008.07.049.
- K. Ruhana, K. Mahamud, M. Zorkeflee, and A. M. Din. Fuzzy Distance-based Undersampling Technique for Imbalanced Flood Data. In *5th International Conference on Computing and Informatics*, pages 509–513, Istanbul, 2016.
- B. Sabzipour, R. Arsenault, M. Troin, J.-L. Martel, F. Brissette, F. Brunet, and J. Mai. Comparing a long short-term memory (LSTM) neural network with a physically-based hydro-

- logical model for streamflow forecasting over a Canadian catchment. *Journal of Hydrology*, 627:130380, Dec. 2023. ISSN 0022-1694. doi: 10.1016/j.jhydrol.2023.130380.
- S. Saffarpour, M. G. Erechtkhova, P. A. Khaite, S. Y. Chen, and M. Herall. Short-term prediction of flood events in a small urbanized watershed using multi-year hydrological records. In *21st International Congress on Modelling and Simulation*, pages 2234–2240, Gold Coast, Australia, 2015.
- T. Saito and M. Rehmsmeier. The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets. *PLoS ONE*, 10(3), Mar. 2015. ISSN 19326203. doi: 10.1371/journal.pone.0118432.
- L. Schoppa, M. Disse, and S. Bachmair. Evaluating the performance of random forest for large-scale flood discharge simulation. *Journal of Hydrology*, 590:125531, Nov. 2020. ISSN 0022-1694. doi: 10.1016/J.JHYDROL.2020.125531.
- J. K. Searcy. Flow-duration curves. Technical report, U.S. Department of the Interior, 1959.
- J. Seibert and K. J. Beven. Gauging the ungauged basin: How many discharge measurements are needed? *Hydrology and Earth System Sciences*, 13(6):883–892, June 2009. ISSN 1607-7938. doi: 10.5194/hess-13-883-2009.
- S. P. Seibert, U. Ehret, and E. Zehe. Disentangling timing and amplitude errors in streamflow simulations. *Hydrology and Earth System Sciences*, 20(9):3745–3763, 2016. ISSN 16077938. doi: 10.5194/hess-20-3745-2016.
- C. Seiffert, T. M. Khoshgoftaar, J. Van Hulse, and A. Napolitano. Resampling or reweighting: A comparison of boosting implementations. In *Proceedings - International Conference on Tools with Artificial Intelligence, ICTAI*, volume 1, pages 445–451, 2008. ISBN 978-0-7695-3440-4. doi: 10.1109/ICTAI.2008.59.
- Y. Seo, S. Kwon, and Y. Choi. Short-term water demand forecasting model combining variational mode decomposition and extreme learning machine. *Hydrology*, 5(4):54, Sept. 2018. ISSN 23065338. doi: 10.3390/hydrology5040054.
- B. Shada, N. R. Chithra, and S. G. Thampi. Hourly Flood Forecasting Using Hybrid Wavelet-

- SVM. *Journal of Soft Computing in Civil Engineering*, 6(2):1–20, Apr. 2022. ISSN 25882872. doi: 10.22115/SCCE.2022.317761.1383.
- M. Shahed Behrouz, Z. Zhu, L. S. Matott, and A. J. Rabideau. A new tool for automatic calibration of the Storm Water Management Model (SWMM). *Journal of Hydrology*, 581, 2020. ISSN 00221694. doi: 10.1016/j.jhydrol.2019.124436.
- A. Y. Shamseldin, K. M. O'Connor, and G. Liang. Methods for combining the outputs of different rainfall-runoff models. *Journal of Hydrology*, 197(1-4):203–229, Oct. 1997. ISSN 00221694. doi: 10.1016/S0022-1694(96)03259-3.
- U. M. Shamsi and J. Koran. Continuous Calibration. *Journal of Water Management Modeling*, 25:414, 2017. ISSN 2292-6062. doi: 10.14796/jwmm.c414.
- A. J. C. Sharkey. On Combining Artificial Neural Nets. *Connection Science*, 8(3-4):299–314, Dec. 1996. ISSN 0954-0091. doi: 10.1080/095400996116785.
- A. Sharma. Seasonal to interannual rainfall probabilistic forecasts for improved water supply management: Part 1 — A strategy for system predictor identification. *Journal of Hydrology*, 239(1-4):232–239, Dec. 2000. ISSN 00221694. doi: 10.1016/S0022-1694(00)00346-2.
- M.-J. Shin and Y. Choi. Sensitivity Analysis to Investigate the Reliability of the Grid-Based Rainfall-Runoff Model. *Water*, 10(12):1839, Dec. 2018. ISSN 2073-4441. doi: 10.3390/w10121839.
- D. L. Shrestha and D. P. Solomatine. Experiments with AdaBoost.RT, an improved boosting scheme for regression. *Neural Computation*, 18(7):1678–1710, July 2006. ISSN 08997667. doi: 10.1162/neco.2006.18.7.1678.
- R. R. Shrestha, A. Bárdossy, and F. Nestmann. Analysis and propagation of uncertainties due to the stage–discharge relationship: A fuzzy set approach. *Hydrological Sciences Journal*, 52(4):595–610, Aug. 2007. ISSN 0262-6667. doi: 10.1623/hysj.52.4.595.
- C. Shu and D. H. Burn. Artificial neural network ensembles and their application in pooled flood frequency analysis. *Water Resources Research*, 40(9), Sept. 2004. ISSN 00431397. doi: 10.1029/2003WR002816.

- C. Shu and T. B. M. J. Ouarda. Flood frequency analysis at ungauged sites using artificial neural networks in canonical correlation analysis physiographic space. *Water Resources Research*, 43(7), July 2007. ISSN 00431397. doi: 10.1029/2006WR005142.
- S. K. Singh and A. Bárdossy. Calibration of hydrological models on hydrologically unusual events. *Advances in Water Resources*, 38:81–91, Mar. 2012. ISSN 03091708. doi: 10.1016/j.advwatres.2011.12.006.
- E. Snieder, R. Shakir, and U. Khan. A comprehensive comparison of four input variable selection methods for artificial neural network flow forecasting models. *Journal of Hydrology*, 583:124299, Apr. 2020. ISSN 00221694. doi: 10.1016/j.jhydrol.2019.124299.
- E. Snieder, K. Abogadil, and U. T. Khan. Resampling and ensemble techniques for improving ANN-based high-flow forecast accuracy. *Hydrology and Earth System Sciences*, 25(5):2543–2566, May 2021. ISSN 1607-7938. doi: 10.5194/hess-25-2543-2021.
- E. J. Snieder. *Artificial Neural Network-Based Flood Forecasting: Input Variable Selection and Peak Flow Prediction Accuracy*. PhD thesis, York University, Nov. 2019.
- D. P. Solomatine and A. Ostfeld. Data-driven modelling: Some past experiences and new approaches. *Journal of Hydroinformatics*, 10(1):3–22, Jan. 2008. ISSN 1464-7141. doi: 10.2166/hydro.2008.015.
- D. P. Solomatine and D. L. Shrestha. AdaBoost.RT: A boosting algorithm for regression problems. In *IEEE International Conference on Neural Networks - Conference Proceedings*, volume 2, pages 1163–1168, 2004. doi: 10.1109/ijcnn.2004.1380102.
- P. C. Steffen and J. Gomes. Clustering of historical floods observed on Iguaçu River, in União da Vitória, Paraná. *RBRH*, 23(23), Aug. 2018. ISSN 2318-0331. doi: 10.1590/2318-0331.231820170107.
- K. P. Sudheer, P. C. Nayak, and K. S. Ramasastri. Improving peak flow estimates in artificial neural network river flow models. *Hydrological Processes*, 17(3):677–686, Feb. 2003. ISSN 08856087. doi: 10.1002/hyp.5103.
- H. Sufi Karimi, B. Natarajan, C. L. Ramsey, J. Henson, J. L. Tedder, and E. Kemper. Comparison of learning-based wastewater flow prediction methodologies for smart sewer manage-

- ment. *Journal of Hydrology*, 577:123977, Oct. 2019. ISSN 00221694. doi: 10.1016/j.jhydrol.2019.123977.
- N. Sun, M. Hall, B. Hong, and L. Zhang. Impact of SWMM Catchment Discretization: Case Study in Syracuse, New York. *Journal of Hydrologic Engineering*, 19(1):223–234, Jan. 2014. ISSN 1084-0699. doi: 10.1061/(ASCE)HE.1943-5584.0000777.
- V. Swathi, K. Srinivasa Raju, V. Murari R.R., and S. Sai Veena. Automatic calibration of SWMM using NSGA-III and the effects of delineation scale on an urban catchment. *Journal of Hydroinformatics*, 21(5):781–797, Sept. 2019. ISSN 14651734. doi: 10.2166/hydro.2019.033.
- B. Szlag, R. Suligowski, J. Studzinski, and F. De Paola. Application of logistic regression to simulate the influence of rainfall genesis on storm overflow operations: A probabilistic approach. *Hydrology and Earth System Sciences*, 24(2):595–614, Feb. 2020. ISSN 1607-7938. doi: 10.5194/hess-24-595-2020.
- B. Szlag, A. Kiczko, G. Walek, E. Wojciechowska, and M. Stachura. The role of catchment characteristics , sewer network , SWMM model parameters in urban catchment management based on stormwater flooding : Modelling , sensitivity analysis , risk assessment. (March):1–27, 2022. doi: 10.5194/hess-2022-109.
- A. Talei and L. H. Chua. Influence of lag time on event-based rainfall-runoff modeling using the data driven approach. *Journal of Hydrology*, 438–439:223–233, 2012. ISSN 00221694. doi: 10.1016/j.jhydrol.2012.03.027.
- T. Tang, D. Jiao, T. Chen, and G. Gui. Medium- and Long-Term Precipitation Forecasting Method Based on Data Augmentation and Machine Learning Algorithms. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 15:1000–1011, 2022. ISSN 21511535. doi: 10.1109/JSTARS.2022.3140442.
- S. Thiesen, P. Darscheid, and U. Ehret. Identifying rainfall-runoff events in discharge time series: A data-driven method based on information theory. *Hydrology and Earth System Sciences*, 23(2):1015–1034, Feb. 2019. ISSN 16077938. doi: 10.5194/hess-23-1015-2019.
- M. K. Tiwari and C. Chatterjee. Uncertainty assessment and ensemble flood forecasting using

- bootstrap based artificial neural networks (BANNs). *Journal of Hydrology*, 382(1-4):20–33, Mar. 2010a. ISSN 00221694. doi: 10.1016/j.jhydrol.2009.12.013.
- M. K. Tiwari and C. Chatterjee. Development of an accurate and reliable hourly flood forecasting model using wavelet-bootstrap-ANN (WBANN) hybrid approach. *Journal of Hydrology*, 394(3-4):458–470, Nov. 2010b. ISSN 00221694. doi: 10.1016/j.jhydrol.2010.10.001.
- M. K. Tiwari and C. Chatterjee. A new wavelet-bootstrap-ANN hybrid model for daily discharge forecasting. *Journal of Hydroinformatics*, 13(3):500–519, 2011. ISSN 14647141. doi: 10.2166/hydro.2010.142.
- A. S. Tokar and M. Markus. Precipitation-Runoff Modeling Using Artificial Neural Networks and Conceptual Models. *Journal of Hydrologic Engineering*, 5(2):156–161, 2000. ISSN 1084-0699. doi: 10.1061/(asce)1084-0699(2000)5:2(156).
- H. Tongal and M. J. Booij. Simulation and forecasting of streamflows using machine learning models coupled with base flow separation. *Journal of Hydrology*, 564:266–282, Sept. 2018. ISSN 00221694. doi: 10.1016/j.jhydrol.2018.07.004.
- L. Torgo, R. P. Ribeiro, B. Pfahringer, and P. Branco. SMOTE for regression. In *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 8154 LNAI, pages 378–389, 2013. ISBN 978-3-642-40668-3. doi: 10.1007/978-3-642-40669-0_33.
- Toronto and Region Conservation Authority. Lower Don River West Remedial Flood Protection Project. <https://trca.ca/conservation/green-infrastructure/lower-don-river-west-remedial-flood-protection-project/>, 2020a.
- Toronto and Region Conservation Authority. Frequently asked questions. <http://beta.trcagauging.ca/faq>, 2020b.
- E. Toth. Classification of hydro-meteorological conditions and multiple artificial neural networks for streamflow forecasting. *Hydrology and Earth System Sciences*, 13(9):1555–1566, Sept. 2009. ISSN 1607-7938. doi: 10.5194/hess-13-1555-2009.
- E. Toth. Estimation of flood warning runoff thresholds in ungauged basins with asymmet-

- ric error functions. *Hydrology and Earth System Sciences*, 20(6):2383–2394, 2016. ISSN 16077938. doi: 10.5194/hess-20-2383-2016.
- Town of Oakville and AMEC. Public Information Centre # 2 Flood Mitigation Opportunities Study Fourteen Mile Creek & M c Craney Creek Systems Municipal Class Environmental Assessment, 2014.
- Y. Trambly, C. Bouvier, C. Martin, J. F. Didon-Lescot, D. Todorovik, and J. M. Domergue. Assessment of initial soil moisture conditions for event-based rainfall-runoff modelling. *Journal of Hydrology*, 387(3-4):176–187, 2010. ISSN 00221694. doi: 10.1016/j.jhydrol.2010.04.006.
- TRCA. Watersheds TRCA. <https://data.trca.ca/dataset/watersheds-trca>, 2018.
- TRCA. TRCA Open Data. <https://data.trca.ca/>, 2020.
- H. Tyralis, G. Papacharalampous, A. Burnetas, and A. Langousis. Hydrological post-processing using stacked generalization of quantile regression algorithms: Large-scale application over CONUS. *Journal of Hydrology*, 577:123957, Oct. 2019a. ISSN 00221694. doi: 10.1016/j.jhydrol.2019.123957.
- H. Tyralis, G. Papacharalampous, and A. Langousis. A Brief Review of Random Forests for Water Scientists and Practitioners and Their Recent History in Water Resources. *Water*, 11(5):910, Apr. 2019b. ISSN 2073-4441. doi: 10.3390/w11050910.
- H. Tyralis, G. Papacharalampous, and A. Langousis. Super ensemble learning for daily streamflow forecasting: Large-scale demonstration and comparison with multiple machine learning algorithms. *Neural Computing and Applications*, 33(8):3053–3068, Apr. 2021. ISSN 14333058. doi: 10.1007/s00521-020-05172-3.
- S. D. Veloz. Spatially autocorrelated sampling falsely inflates measures of accuracy for presence-only niche models. *Journal of Biogeography*, 36(12):2290–2299, Dec. 2009. ISSN 03050270. doi: 10.1111/j.1365-2699.2009.02174.x.
- A. Vezhnevets and O. Barinova. Avoiding Boosting Overfitting by Removing Confusing Samples. In *Machine Learning: ECML 2007*, volume 4701 LNAI, pages 430–441.

- Springer Berlin Heidelberg, Berlin, Heidelberg, 2007. ISBN 978-3-540-74957-8. doi: 10.1007/978-3-540-74958-5_40.
- B. Wan and W. James. SWMM calibration using genetic algorithms. *Global Solutions for Urban Drainage*, pages 1–14, Sept. 2002. ISSN 2292-6062. doi: 10.14796/jwmm.r208-07.
- J. Wang, Z. Liang, X. Jiang, B. Li, and L. Chen. Bayesian Theory Based Self-Adapting Real-Time Correction Model for Flood Forecasting. *Water*, 8(3):75, Feb. 2016. ISSN 2073-4441. doi: 10.3390/w8030075.
- R. Wang, X. Zhang, and M. H. Li. Predicting bioretention pollutant removal efficiency with design features: A data-driven approach. *Journal of Environmental Management*, 242:403–414, 2019a. ISSN 10958630. doi: 10.1016/j.jenvman.2019.04.064.
- S.-H. Wang, H.-F. Li, Y.-J. Zhang, and Z.-S. Zou. A Hybrid Ensemble Model Based on ELM and Improved AdaBoost.RT Algorithm for Predicting the Iron Ore Sintering Characters. *Computational Intelligence and Neuroscience*, 2019:1–11, Jan. 2019b. ISSN 1687-5265. doi: 10.1155/2019/4164296.
- W. Wang, P. H. Gelder, J. K. Vrijling, and J. Ma. Forecasting daily streamflow using hybrid ANN models. *Journal of Hydrology*, 324(1-4):383–399, 2006. ISSN 00221694. doi: 10.1016/j.jhydrol.2005.09.032.
- S. Whitelam, V. Selin, S.-W. Park, and I. Tamblyn. Correspondence between neuroevolution and gradient descent. *Nature Communications*, 12(1):6317, Nov. 2021. ISSN 2041-1723. doi: 10.1038/s41467-021-26568-2.
- D. H. Wolpert. Stacked generalization. *Neural Networks*, 5(2):241–259, Jan. 1992. ISSN 08936080. doi: 10.1016/S0893-6080(05)80023-1.
- S. C. Worland, W. H. Farmer, and J. E. Kiang. Improving predictions of hydrological low-flow indices in ungaged basins using machine learning. *Environmental Modelling & Software*, 101:169–182, Mar. 2018. ISSN 13648152. doi: 10.1016/j.envsoft.2017.12.021.
- Y. Wu, Y. Ding, and J. Feng. SMOTE-Boost-based sparse Bayesian model for flood prediction. *Eurasip Journal on Wireless Communications and Networking*, 2020(1):78, 2020. ISSN 16871499. doi: 10.1186/s13638-020-01689-2.

- Z. Xiang and I. Demir. Distributed long-term hourly streamflow predictions using deep learning – A case study for State of Iowa. *Environmental Modelling and Software*, 131:104761, Sept. 2020. ISSN 13648152. doi: 10.1016/j.envsoft.2020.104761.
- Z. Xiang, J. Yan, and I. Demir. A Rainfall-Runoff Model With LSTM-Based Sequence-to-Sequence Learning. *Water Resources Research*, 56(1):e2019WR025326, 2020. ISSN 1944-7973. doi: 10.1029/2019WR025326.
- Z. Xu, L. Xiong, H. Li, J. Xu, X. Cai, K. Chen, and J. Wu. Runoff simulation of two typical urban green land types with the Stormwater Management Model (SWMM): Sensitivity analysis and calibration of runoff parameters. *Environmental Monitoring and Assessment*, 191(6), 2019. ISSN 15732959. doi: 10.1007/s10661-019-7445-9.
- B. W. Yap, K. A. Rani, H. A. A. Rahman, S. Fong, Z. Khairudin, and N. N. Abdullah. An Application of Oversampling, Undersampling, Bagging and Boosting in Handling Imbalanced Datasets. *Lecture Notes in Electrical Engineering*, 285 LNEE:13–22, 2014. ISSN 18761119. doi: 10.1007/978-981-4585-18-7_2.
- Z. M. Yaseen, A. El-Shafie, O. Jaafar, A. Afan, N. Sayl, H. A. Afan, and K. N. Sayl. Artificial intelligence based models for stream-flow forecasting: 2000–2015. *Journal of Hydrology*, 530: 829–844, Nov. 2015. ISSN 00221694. doi: 10.1016/j.jhydrol.2015.10.038.
- Z. M. Yaseen, M. F. Allawi, A. A. Yousif, O. Jaafar, F. M. Hamzah, and A. El-Shafie. Non-tuned machine learning approach for hydrological time series forecasting. *Neural Computing and Applications*, 30(5):1479–1491, 2018. ISSN 09410643. doi: 10.1007/s00521-016-2763-0.
- K. K. Yilmaz, R. F. Adler, Y. Tian, Y. Hong, and H. F. Pierce. Evaluation of a satellite-based global flood monitoring system. *International Journal of Remote Sensing*, 31(14):3763–3782, Aug. 2010. ISSN 0143-1161. doi: 10.1080/01431161.2010.483489.
- C.-C. C. Young, W.-C. C. Liu, and M.-C. C. Wu. A physically based and machine learning hybrid approach for accurate rainfall-runoff modeling during extreme typhoon events. *Applied Soft Computing*, 53:205–216, Apr. 2017. ISSN 15684946. doi: 10.1016/j.asoc.2016.12.052.
- Z. Zahmatkesh, M. Karamouz, and S. Nazif. Uncertainty based modeling of rainfall-runoff:

- Combined differential evolution adaptive Metropolis (DREAM) and K-means clustering. *Advances in Water Resources*, 83:405–420, Sept. 2015. ISSN 03091708. doi: 10.1016/j.advwatres.2015.06.012.
- I. Zaier, C. Shu, T. B. Ouarda, O. Seidou, and F. Chebana. Estimation of ice thickness on lakes using artificial neural network ensembles. *Journal of Hydrology*, 383(3-4):330–340, Mar. 2010. ISSN 00221694. doi: 10.1016/j.jhydrol.2010.01.006.
- A. D. L. Zanchetta, P. Coulibaly, and V. Fortin. Forecasting High-Flow Discharges in a Flashy Catchment Using Multiple Precipitation Estimates as Predictors in Machine Learning Models. *Hydrology*, 9(12):216, Dec. 2022. ISSN 2306-5338. doi: 10.3390/hydrology9120216.
- C. Zhan, J. Han, L. Zou, F. Sun, and T. Wang. Heteroscedastic and symmetric efficiency for hydrological model evaluation criteria. *Hydrology Research*, 50(5):1189–1201, Oct. 2019. ISSN 0029-1277. doi: 10.2166/nh.2019.121.
- H. Zhang, Q. Yang, J. Shao, and G. Wang. Dynamic Streamflow Simulation via Online Gradient-Boosted Regression Tree. *Journal of Hydrologic Engineering*, 24(10):04019041, Oct. 2019. ISSN 1084-0699. doi: 10.1061/(ASCE)HE.1943-5584.0001822.
- Y. Zhang, S. Ragetti, P. Molnar, O. Fink, and N. Peleg. Generalization of an Encoder-Decoder LSTM model for flood prediction in ungauged catchments. *Journal of Hydrology*, 614:128577, Nov. 2022. ISSN 0022-1694. doi: 10.1016/j.jhydrol.2022.128577.
- Z.-L. Zhang, X.-G. Luo, Y. Yu, B.-W. Yuan, and J.-F. Tang. Integration of an improved dynamic ensemble selection approach to enhance one-vs-one scheme. *Engineering Applications of Artificial Intelligence*, 74:43–53, Sept. 2018. ISSN 09521976. doi: 10.1016/j.engappai.2018.06.002.
- H. Zhao, A. P. Sinha, and G. Bansal. An extended tuning method for cost-sensitive regression and forecasting. *Decision Support Systems*, 51(3):372–383, 2011. ISSN 01679236. doi: 10.1016/j.dss.2011.01.003.
- Q. Zhaowei, L. Haitao, L. Zhihui, and Z. Tao. Short-Term Traffic Flow Forecasting Method With M-B-LSTM Hybrid Network. *IEEE Transactions on Intelligent Transportation Systems*, pages 1–11, July 2020. ISSN 1524-9050. doi: 10.1109/TITS.2020.3009725.

- F. Zheng, H. R. Maier, W. Wu, G. C. Dandy, H. V. Gupta, and T. Zhang. On Lack of Robustness in Hydrological Model Development Due to Absence of Guidelines for Selecting Calibration and Evaluation Data: Demonstration for Data-Driven Models. *Water Resources Research*, 54(2):1013–1030, Feb. 2018. ISSN 00431397. doi: 10.1002/2017WR021470.
- F. Zheng, J. Chen, H. R. Maier, and H. Gupta. Achieving Robust and Transferable Performance for Conservation-Based Models of Dynamical Physical Systems. *Water Resources Research*, 58(5), May 2022. ISSN 19447973. doi: 10.1029/2021WR031818.
- J. Zhou, T. Peng, C. Zhang, and N. Sun. Data Pre-Analysis and Ensemble of Various Artificial Neural Networks for Monthly Streamflow Forecasting. *Water*, 10(5):628, May 2018. ISSN 2073-4441. doi: 10.3390/w10050628.
- M. Zounemat-Kermani, O. Batelaan, M. Fadaee, and R. Hinkelmann. Ensemble machine learning paradigms in hydrology: A review. *Journal of Hydrology*, 598:126266, July 2021. ISSN 00221694. doi: 10.1016/j.jhydrol.2021.126266.

Appendix A Code Availability

The code for chapters 2-5 is hosted in repositories with the following URLs:

Chapter 2: <https://github.com/everettsp/swmm.m>

Chapter 3-4: <https://github.com/everettsp/ensembles.m>

Chapter 5: <https://github.com/everettsp/lsh-forecasting>

Chapters 4 and 3 are coded purely in MATLAB. The SWMM models in 2 are generated from raw data using Python to utilise the ArcPy's hydrology toolbox, while the SWMM wrapper and analysis are coded in MATLAB. Chapter 5 is coded completely in Python.

Appendix B List of Publications

Journal Articles

Snieder, E. and Khan, U. T. (Preprint, 2024) A diversity centric strategy for the selection of spatio-temporal training data for LSTM-based streamflow forecasting, Hydrology and Earth System Sciences Discussions.

Snieder, E., Alobaidi, M., Khan, U. T. (Submitted, 2024). Advancing flood early warning systems: ensemble learning based classifiers for urban flood forecasting. Submitted to the Journal of Hydroinformatics.

Snieder, E., Khan, U. T. (2023). A novel ensemble algorithm based on hydrological event diversity for urban rainfall–runoff model calibration and validation. Journal of Hydrology.

Snieder, E., Abogadil, K., Khan, U. T. (2021). Resampling and ensemble techniques for improving ANN-based high-flow forecast accuracy. Hydrology and Earth System Sciences. 2543–2566.

Conference Manuscripts and Presentations

Snieder, E., Khan, U. T. (2024). Towards improved spatio-temporal selection of training data for LSTM-based flow forecasting models in Canadian basins. Poster presentation at the EGU General Assembly. Vienna, Austria.

Khan, U. T. Snieder, E. (2023). Cluster-based hyperparameter optimisation for LSTM-based flow forecasting in Canadian catchments. Podium presentation at the EGU General Assembly. Vienna, Austria.

Snieder, E., Khan, U. T. (2023). A large sample study of the effects of upstream hydrometeorological input features for LSTM-based daily flow forecasting in Canadian catchments. Poster presentation at the EGU General Assembly. Vienna, Austria.

Snieder, E., Khan, U. T. (2022). An ANN-based Canada-wide flood forecasting system. Podium presentation at the Canadian Water Resources Association National Conference. Canmore, Canada.

Snieder, E., Khan, U. T. (2022). Large-scale evaluation of temporal trends in ANN behaviour for daily flow forecasts in Canadian catchments. Podium presentation at the EGU General Assembly. Vienna, Austria.

Snieder, E., Khan, U. T. (2021). Large-scale analysis of machine learning based flow forecasting models for Southern Ontario. Poster presentation at the AGU General Assembly. Hybrid Poster.

Snieder, E., Khan, U. T. (2021). Investigating event selection for GA-based SWMM rainfall–runoff model calibration. Manuscript and virtual presentation at the CSCE Annual Confer-

ence.

Snieder, E., Khan, U. T. (2021). Spatial bootstrapping for model-free estimation of subcatchment parameter uncertainty for a semi-distributed rainfall runoff model. Virtual presentation at the EGU General Assembly.

Snieder, E., Khan, U. T. (2020). Improved real-time SWMM flow forecasts using two machine learning approaches. Poster presentation at the EGU General Assembly. Virtual Display.

Snieder, E., Khan, U. T. (2019). A comprehensive evaluation of boosting algorithms for artificial neural network-based flow forecasting models. Poster presentation at the AGU General Assembly. San Francisco, USA.

Snieder, E., Shakir, R., Khan, U. T. (2019). Comparison of four input variable selection methods for artificial neural network-based flood forecasting models. Manuscript and podium presentation at the CSCE Annual Conference. Laval, Canada.

Snieder, E. & Khan, U. T. (2019). Assessment of peak flow error metrics and error weights for data-driven riverine flow forecasting models. Poster presented at the EGU General Assembly. Vienna, Austria.

Appendix C Supplementary SWMM studies

This section includes the supplementary information for Chapter 2, and several other projects that were submitted to conferences in the form of presentations, posters, or conference papers, but not included in the manuscript.

C.1 Supplementary information for Chapter 3

C.1.1 Description of SWMM-MATLAB interface

A hierarchical list of functions programmed for this study is included below. These are based around creating a SWMM class object in MATLAB with class functions to import SWMM models within MATLAB (through shapefiles or existing INP files), run simulations, modify simulation and model parameters, create binary RFF and HSF files. Another class, event data, is effectively a collection of hydrological timeseries that are discretised from a continuous record. This class is often an input to the SWMM class functions and has respective class functions for calculating event characteristics and clustering, which are used to generate the SWMM ensembles. The complete SWMM wrapper presented in this work is available on [GitHub](https://github.com/everettsp/swmm.m) (<https://github.com/everettsp/swmm.m>).

SWMM model (MATLAB CLASS OBJECT)

```
└─ model = read_inp(inp_file)
└─ model = create_inp(shapefiles)
└─ inp_file = write_inp(model)
└─ simulated_flow = run_simulation(model, event_data)
    │
    │   modify simulation start and end dates
    │   └─ set rff and hsf file pointers
    │       └─ create temporary system batch file to call SWMM executable
└─ map = plot(model)
└─ model_summary = summarise(model)
└─ model = calibrate(model, event_data, calibration_variables)
    │
    │   simulated_flow = run_simulation(model, event_data)
    │   └─ score = evaluate_model(results, event_data)
    │       └─ model = adjust_model_parameters(model, score)
└─ rff_file, hsf_file = prerun(model, event_data)
    │
    │   rff_file = create_rff(model, event_data)
    │       │
    │       └─ run_simulation(model, event_data)
    │   └─ hsf_file = create_hsf(model, event_data)
    │       └─ run_simulation(model, event_data)
```

SWMM event data (MATLAB TIMESERIES OBJECTS)

```
└─ event_data = discretise(continuous_precipitation, continuous_flow)
└─ event_features = summarise(event_data)
└─ event_clusters = cluster_events(event_features)
```

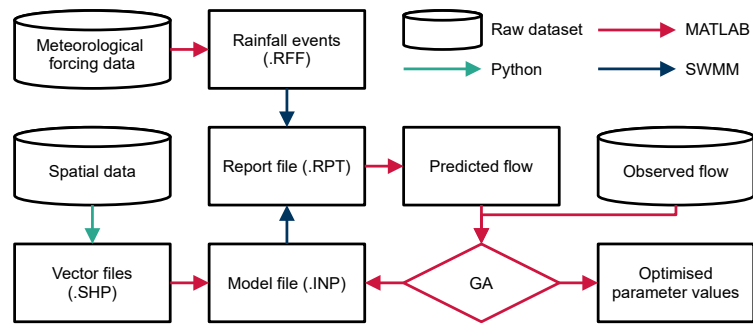


Figure C.1: Process diagram showing how Python (green), MATLAB (red), and SWMM (blue) are used to create SWMM model files, and perform model calibration.

C.1.2 Hydrological events

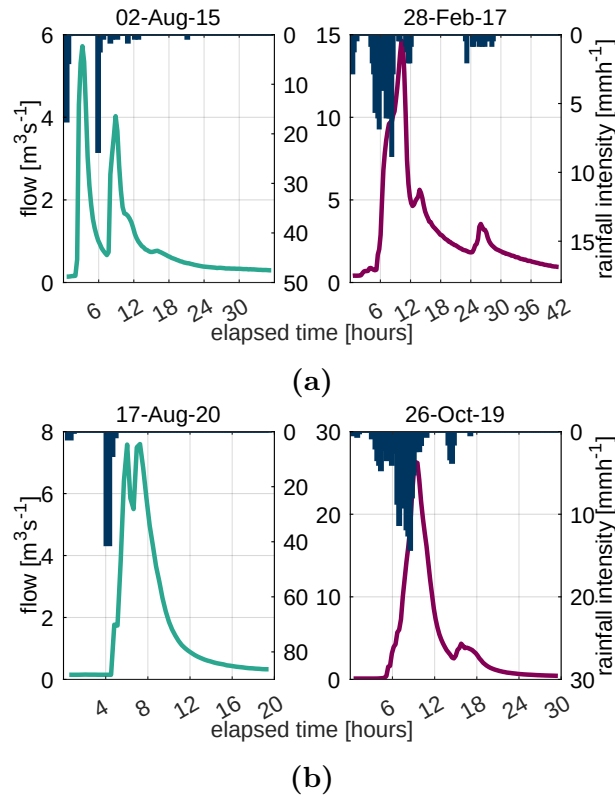


Figure C.2: Sample discretised hyetographs (blue) and class 1 (green, left column) and class 2 (purple, right column) hydrographs for the SCS (a) and TCS (b) catchments.

Table C.1: Values of event cluster centroids and total number of events for SCS and TCS catchments.

description	short-form	units	scs cluster 1	scs cluster 2	tcs cluster 1	tcs cluster 2
total precipitation	P	[mm]	31.6	12.1	37	11.8
peak precip. intensity	i _p	[mm hr ⁻¹]	11.3	9.96	22.4	11.4
mean precip. intensity	i _μ	[mm hr ⁻¹]	0.787	0.487	1.14	0.506
duration	D	[hr]	29.3	14.6	22.6	13.4
date (sine)	date (sine)	[−]	0.355	-0.13	-0.221	-0.3
date (cosine)	date (cosine)	[−]	-0.0243	-0.0639	-0.353	-0.209
peak flow	q _p	[m ³ s ⁻¹]	11	4.64	18.4	5.44
total flow volume	Q	[m ³ s ⁻¹]	4.22e+05	1.02e+05	4.25e+05	9.62e+04
hydrograph centroid (flow)	q _{cy}	[m ³ s ⁻¹]	2.6	1.11	4.8	1.4
hydrograph centroid (time)	q _{ct}	[hr]	19.4	12.4	12.9	10.9
hyetograph centroid (depth)	p _{cy}	[mm]	0.512	0.623	1.2	0.733
hyetograph centroid (time)	p _{ct}	[hr]	11.9	6.45	8.69	5.94
total events		[count]	259	45	250	34

Table C.2: Statistical summary of discretised hydrological events for the SCS catchment.

event label	start date [dd-mm-yy HHMM]	duration [hrs]	total precip. [mm]	peak intensity [mm/hr]	peak flow [cms]	class
1	28-Apr-13 17:45	10	9.58	3.94	3.73	1
2	16-Jun-13 05:45	16.3	7.85	10.5	6.92	1
3	05-Jul-13 06:15	8.5	5.43	3.15	8.18	1
4	21-Oct-13 03:30	25.8	6.43	3.15	3.18	1
5	30-May-15 11:15	32	58.7	33.1	9.59	2
6	07-Jun-15 20:45	39.5	30.1	13.4	13.6	2
7	02-Aug-15 16:30	17.3	14.6	23.6	5.72	1
8	02-Feb-16 18:45	20.5	10.6	3.2	2.82	1
9	10-Jan-17 05:15	24	23.7	7.2	11.3	2
10	28-Feb-17 20:30	25	23.6	8.8	14.6	2
11	03-Apr-17 08:15	28.8	20.7	6.8	12.3	2
12	17-Aug-17 13:45	14.5	12.3	13.2	8.9	1
13	21-Jan-18 21:45	30.3	23	4	7.72	2
14	19-Feb-18 18:15	18.8	21.1	4.8	9.51	2
15	03-Apr-18 12:30	28.3	34.5	10.8	15.2	2
16	05-Jul-18 18:45	3.75	14.6	44.2	11	1
17	30-Mar-19 03:30	29.5	35.5	9.2	7.53	2
18	05-Jun-19 05:30	29.3	10.3	4.74	3.62	1
19	26-Oct-19 18:45	14.3	32.3	12	16.2	2
20	12-Apr-20 19:45	13	10	3.19	3.31	1

Table C.3: Statistical summary of discretised hydrological events for the TCS catchment.

event label	start date [dd-mm-yy HHMM]	duration [hrs]	total precip. [mm]	peak intensity [mm/hr]	peak flow [cms]	class
1	09-Dec-12 20:00	12.8	10.6	5.6	2.3	1
2	20-May-13 21:45	20.8	17.2	28.8	9.45	1
3	10-Jun-13 08:30	19.5	25.4	8	13	2
4	27-Jul-13 15:00	9	21.6	56	4.49	1
5	06-Oct-13 23:15	17.8	10.2	6.4	4.34	1
6	27-Jun-15 11:15	30.5	42.4	6.72	9.9	2
7	19-Jun-17 23:15	38	13.7	4.25	3.05	1
8	07-Aug-18 19:00	20.5	20.5	9.8	34.1	2
9	21-Aug-18 06:30	31.5	44.3	40.3	16.6	2
10	02-Oct-18 02:45	17	9.75	6.2	3.56	1
11	01-Nov-18 10:15	46.8	37.9	8.8	11.3	2
12	26-Nov-18 06:45	22.8	24.8	4	8.35	2
13	14-Apr-19 08:15	26.8	29.4	13.6	19.1	2
14	18-Apr-19 20:00	43.8	29	7.2	5.81	2
15	17-Aug-19 06:00	8.25	17	12.8	6.3	1
16	26-Oct-19 19:30	16	38.2	14.3	26.3	2
17	30-Oct-19 12:15	32.8	43.6	7.95	10.8	2
18	17-May-20 23:30	25.5	9.52	3.39	3.25	1
19	17-Aug-20 11:00	3.5	13.8	41.2	7.6	1
20	23-Dec-20 22:30	33	23.1	3.6	2.31	1

C.2 GA-based calibration and ANN-based real-time bias correction for SWMM

C.2.1 Introduction

Flood damage is increasing due to rapid urbanisation and climate change effects. Flow forecasting models are among the best ways to mitigate flood damage. Traditionally, physically-based rainfall runoff models have been used for forecasting. Such models can be challenging to calibrate, as they may contain thousands of uncertain parameters. This project studies predictions made by SWMM, a popular semi-distributed rainfall-runoff model developed by the U.S. Environmental Protection Agency.

Data-driven models, which rely on machine learning techniques for characterising rainfall runoff systems, have been subject to extensive research throughout the past three decades; though, their operational use remains scarce. Despite widespread use of both model paradigms, there are relatively few studies in which they are explicitly compared. The Fourteen Mile Creek watershed, illustrated in Figure 1, is the focus of this study. The watershed is 70 ha in size, covers natural forest and urban areas, and was identified by the town of Oakville as the highest priority for flood mitigation measures [1]. One year of data (2019) is available at a 5-minute temporal resolution, including radar derived areal rainfall estimates and flow levels at one gauge, illustrated above.

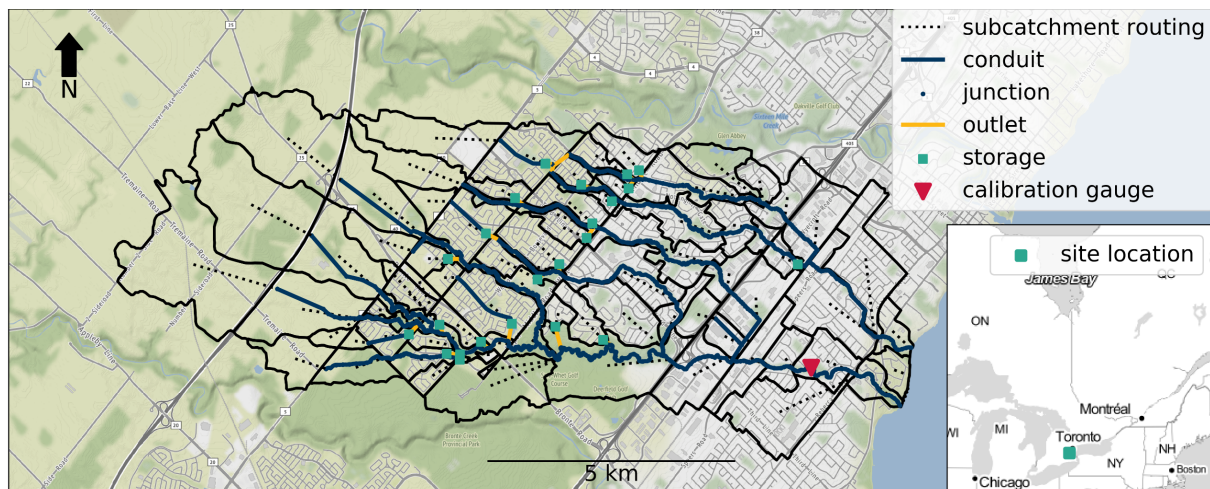


Figure C.3: SWMM model of the Fourteen Mile Creek watershed.

The objectives of this work are: (i) to calibrate SWMM and ANN flow forecasting models. (ii) Develop a hybrid approach that combines the benefits of both SWMM and ANN models, which include: strong performance, real-time adaptability, and a basis in physical hydrology. (iii) Assess the generalisability of each of the three model types through cross-validation on rainfall events that were not used for calibration.

C.2.2 Methods

The SWMM model is relatively large, featuring more than 70 catchments, 800 irregular conduit cross-sections, 20 stormwater ponds, and radar-derived rainfall input. The model was ob-

Table C.4: GA calibration parameters and uncertainty ranges.

attribute	description	uncertainty
Width	Characteristic width	200%
PercentSlope	Average slope	25%
PercentImperv	Percent impervious land cover	20%
N Imperv	Impervious area roughness	10%
N Perv	Pervious area roughness	50%
S Imperv	Impervious depression storage	20%
S perv	Pervious depression storage	50%
Suction	Suction head at the wetting front	50%
Ksat	Saturated hydraulic conductivity	50%
IMD	Initial moisture deficit	25%
Roughness	Channel roughness	25%

tained in an uncalibrated state. Calibration is performed using a Genetic Algorithm (GA), maximising NSE as the objective (cf. [Wan and James, 2002]). Calibration of models of this size are computationally intensive as there are over 1000 model parameters to optimise, which is why the model is calibrated on an event basis over a continuous simulation. A summary of the relative uncertainties associated with select SWMM parameters, determined based on expert judgement, is shown in Table C.4. The GA searches for optimum values from a uniformly distributed initial population.

ANNs, which can take advantage of real-time flow observations for short-term predictions and adaptive learning, are developed in this research. This study uses simple multi-layer perception ANNs, which are the most frequently used ANN architecture for water resources applications [Maier et al., 2010], to correct SWMM predictions with a lead time of 4-hours. Lagged rainfall and flow timeseries are used as the input variables for this model [Maier et al., 2010, Snieder et al., 2020]. In order to improve model generalisation and prevent overfitting, a small hidden layer consisting of only 3 hidden neurons is used. ANN generalisation is also improved by using an ensemble of ANNs, which use randomly sampled data used for training and as a validation stopping criterion.

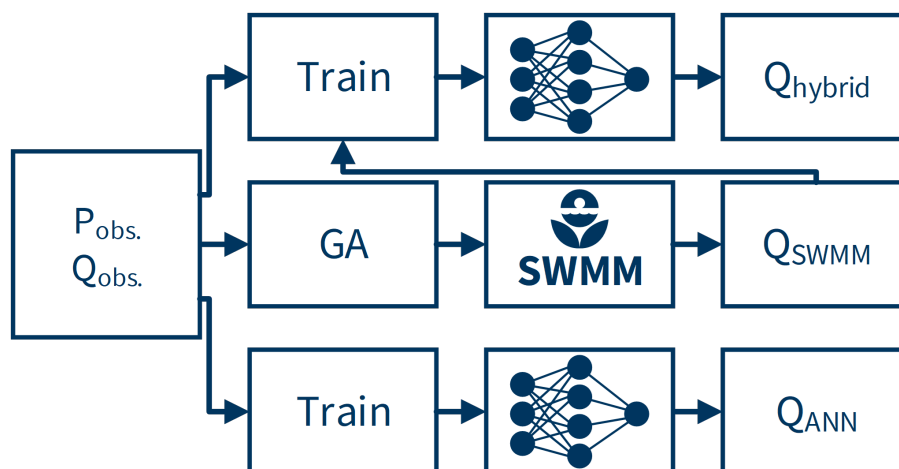


Figure C.4: Configuration of SWMM, ANN, and hybrid models.

Table C.5: Discretised event dates, statistics, and classification result.

start date	duration [hours]	total rainfall [mm]	peak intensity [mm/hr]
25-May-19	32.8	24.7	20
05-Jun-19	26.7	22.3	21.6
15-Jul-19	54.1	24.2	16.1
01-Oct-19	72.8	30.1	36.5
26-Oct-19	28.7	32.2	10.9
30-Oct-19	51.8	30	5.1

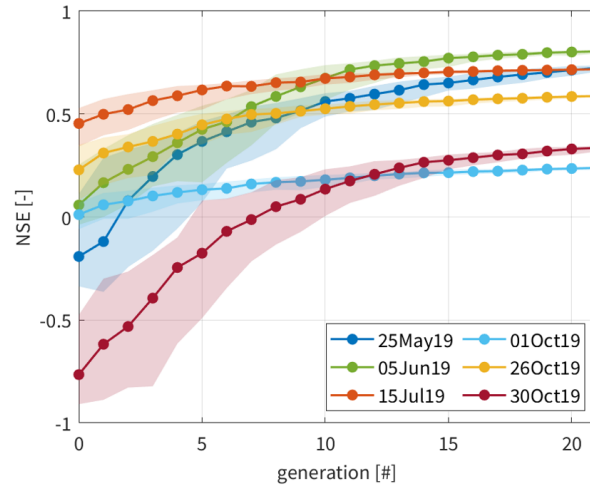


Figure C.5: Objective function versus GA generations for calibration of SWMM models on six different rainfall events.

Hybrid models are emerging as a popular method for combining the advantages of physically-based and data-driven models [Young et al., 2017, Kurian et al., 2020]). The hybrid model uses lagged rainfall and flow timeseries as inputs, and also uses the SWMM flow predictions as a model input parameter, as shown in Fig. C.4.

All three model types are calibrated on each rainfall event shown in Table C.5, and their performance is evaluated on each of the 5 remaining events (referred to as validation events). The model performance is assessed using two common criteria, the Nash-Sutcliffe Efficiency (NSE) and Persistency Index (PI), which both quantify model performance with a value from $-\infty$ to 1, having an optimum value of 1.

C.2.3 Results

The GA is used to optimise the NSE of a SWMM model to 6 different rainfall events, summarised in Table C.5 and illustrated in Fig. C.6. Fig. C.5 shows the increase in NSE produced by the GA, from an initial range of approximately -0.75 to 0.50, to a calibrated range of 0.25 to 0.80.

While the SWMM models achieve satisfactory performance on the calibration events, as shown in Figure 5 below, the models do not perform well when evaluated on other rainfall events. The pure ANN model achieves much stronger performance on the calibration events, but ex-

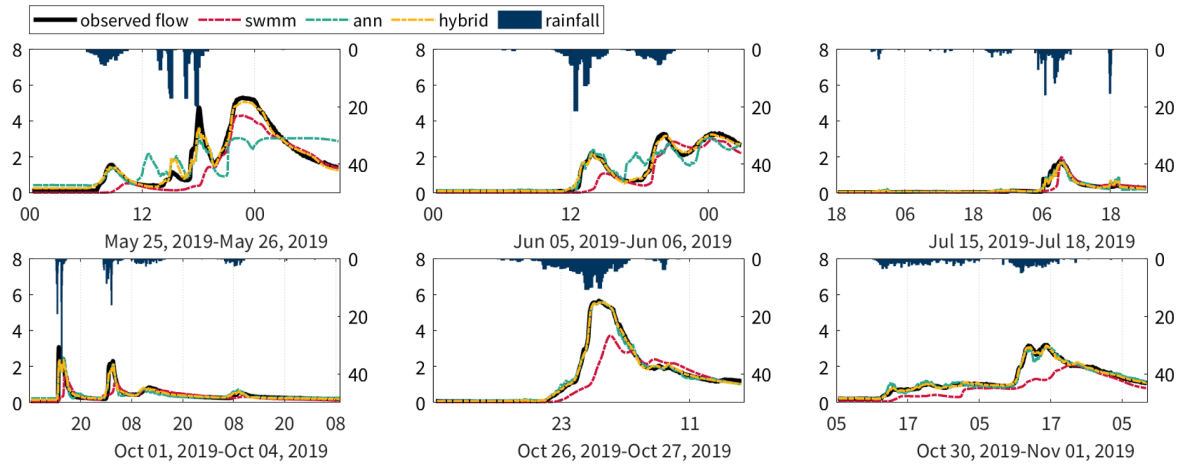


Figure C.6: Event-based calibration performance for SWMM, ANN, and hybrid models.

hibits similarly poor generalisation to the SWMM model on validation events. The hybrid models achieve excellent performance on the calibration events and very good generalisation. In particular, the models calibrated on the 05-Jun-19 events achieves positive NSE and PI values across all validation events.

C.2.4 Conclusion

This study demonstrated that GA can be used for optimisation of a large scale semi-distributed rainfall-runoff model in SWMM. ANN-based and hybrid (SWMM-ANN) models were developed to improve real-time predictions. The pure SWMM and ANN models exhibit poor cross-validation performance, whereas the hybrid model performed reasonably well. Ongoing research is evaluating the impacts of multi-event and continuous calibrations on model generalisation and improving the hybrid model performance for lead-times larger than 4-hours.

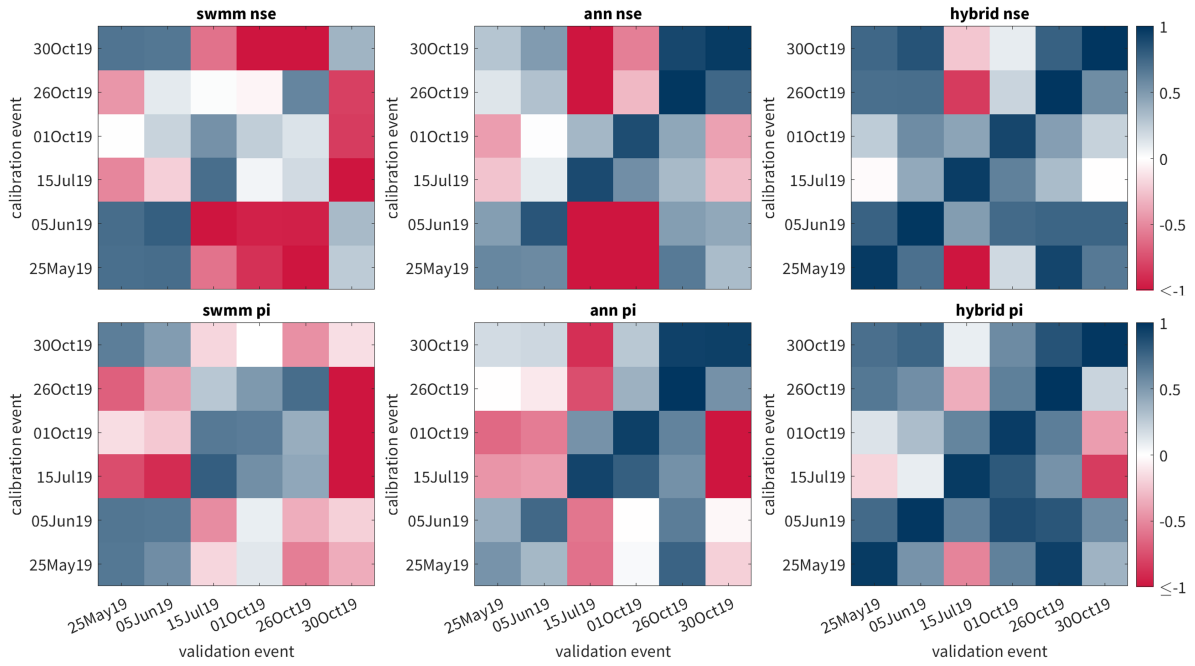


Figure C.7: NSE (top row) and PI (bottom row) of the three models and six events considered. The events along the Y-axis are used for calibration while the X-axis are used for validation. The diagonal shows the calibration performance of each event whereas the remaining cells show the validation performance.

C.3 Investigating event selection for GA-based SWMM rainfall-runoff model calibration

C.3.1 Introduction

Physics-based models are commonly used to simulate dynamic hydrological processes in order to forecast future conditions. SWMM is a popular semi-distributed hydrological discrete-time simulation engine developed and freely distributed by the United States Environmental Protection Agency (US-EPA). SWMM is capable of simulating processes including surface runoff, groundwater, flow routing, infiltration, snowmelt, surface ponding, and water quality routing; many of these can be simulated using a variety of methods. As is typical for physics-based rainfall-runoff models, SWMM models are parameterised based mainly on measurable watershed characteristics. Many model parameters are highly uncertain, which is attributable to factors such as measurement error, non-stationary parameters, and unknown degrees of spatio-temporal heterogeneity [Li et al., 2010]. In fact, the uncertainty of model parameters is among the most important uncertainty sources to consider for hydrological modelling [Shin and Choi, 2018, Li et al., 2010].

Typically, SWMM parameters are calibrated within their range of uncertainty in order to make predictions that more closely reflect observable values, which is quantified in the form of one or several performance criteria [Krebs et al., 2013]. Automated calibration techniques have been demonstrated throughout the past two decades, nevertheless, many aspects, or the entire model calibration is still performed manually in an ad-hoc fashion [Niazi et al., 2017].

Several studies have demonstrated the application of methods such as generalised likelihood uncertainty estimate (GLUE) and genetic algorithms (GA) for calibrating model parameters. However, model performance and generalisation remain heavily dependent on the selection of training data, which is typically done manually. A model calibrated to a particular event(s) is not guaranteed to maintain its accuracy on other events [Beven, 2012]. Therefore, selection of calibration and validation events is an important consideration, regardless of the calibration method. Multi-model approaches have emerged as a popular way to account for the dynamic variability of the rainfall-runoff relationship [Beven, 2012]. Such approaches can simultaneously consider multiple possible calibrated parameter sets. Considering the statements above, the objectives of this paper are to (1) investigate data-driven methods for identifying calibration events for SWMM calibration and (2) comparing different multi-model combination strategies.

C.3.2 Study area

Fourteen Mile Creek (FMC) is in and upstream of the town of Oakville ON, Canada. FMC has a total area of 31km², which is discretised into 61 subcatchments. The watershed is characterised as semi-urban, containing predominantly agricultural land-use upstream, while downstream consists of commercial and medium-density residential. The watercourses in the FMC model representation are highly detailed, as reflected by hundreds of irregular cross-sections used to represent the main creek. The FMC has historically experienced flooding and has been subject of several flood mitigation projects over the past decade [Town of Oakville and AMEC, 2014]. The FMC SWMM model was obtained in an uncalibrated state, evidenced by the low precision and variability of parameter values. Cold weather conditions were removed from the dataset based on historic data obtained from Environment Canada [Environment Canada, 2022]. as cold weather processes (e.g., snow accumulation) are not parameterised in the SWMM model. To accommodate varying initial conditions for each event-based simulation, Hot Start Files (HSF) are produced for each hydrological event using a period of one week prior to the event start.

C.3.3 Methods

The following section provides a brief review of state-of-the-art methods for event selection, followed by a description of the methods used for model calibration, performance evaluation, and multi-model combination.

C.3.3.1 Event discretisation and selection

The selection of rainfall event(s) is an important consideration, as it will influence how the model performs on future events, or how well the model generalises. It is widely known that calibrated models will perform best on data that resembles the calibration data. Thus, event selection strategies tend to reflect the model application; for flood forecasting, a popular application, high flow rainfall-runoff events are selected for calibration. Several studies have used ad-hoc approaches for identifying rainfall events. Qin et al. [2013] create synthetic events in 3 categories, with varying characteristics such as return period, duration, rainfall intensity,

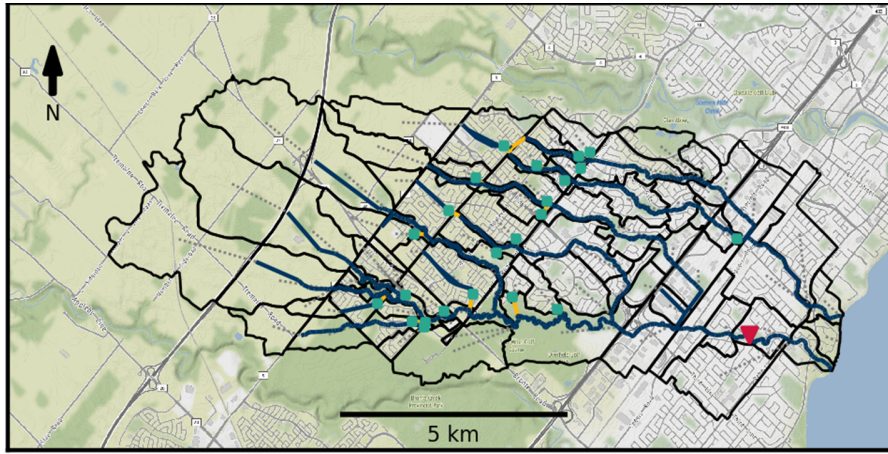


Figure C.8: Fourteen Mile Creek SWMM model. Subcatchment boundaries (black lines), subcatchment routing (grey lines), watercourses (blue lines), storage units (green squares), weirs (yellow lines), and the stream gauge location (red triangle). The basemap (OpenStreetMap) illustrates landcover, roads, and Lake Ontario.

rainfall amount, and time-to-peak. In this study, grouping rainfall events is motivated by the unique rainfall-runoff relationship across different rainfall characteristics, which is confirmed by simulation results. Blackler and Guo [2014] distinguish early spring, spring, summer, and intense summer rainfall events; this study noted that long, low intensity events achieved stronger calibration performance compared to short, intense events. Broekhuizen et al. [2020] evaluate a multi-event calibration scheme in which events are selected from a set of discretised events based on one of rainfall, rainfall intensity, mean flow, total flow relative to total rainfall, or rainfall duration. In their calibration scheme, events that fall above and below a runoff threshold are used in a two-stage calibration technique, which is found to outperform a single stage multi-event approach.

Data-driven hydrological models, often considered to be counterparts to physics-based models such as SWMM, often employ sophisticated data splitting methods for identifying data for model calibration and validation. It is well-known that data used for calibration and validation should have similar statistical properties [Zheng et al., 2018]. Unsupervised clustering provides an objective way to identify groups of events, each with distinct characteristics. Clustering techniques have previously been used to classify individual observations, for the dual purpose of ensuring that each all flow regimes are equally represented (i.e., an equal number of low and high flows), and to ensure similar statistical properties between the calibration and validation data [Bowden et al., 2002, Anctil and Lauzon, 2004a, Toth, 2009]. Toth [2009] used an SOM to classify streamflow and rainfall data (with several lagged copies of each), evaluating the effect that varying the number of clusters has on overall model performance. The study also links the SOM-based classifications to physical hydrological phenomena (e.g., a given class may correspond to the rising limb of the hydrograph). Bowden et al. [2002] compared GA-based and SOM-based data splitting methods, finding that they both outperformed the traditional approach, in which data is split arbitrarily without any consideration for the statistical properties of each split. Similarly, Anctil and Lauzon [2004a] use an SOM to create

data subsets with statistically similar properties.

However, because physics-based models require time-continuous data [Guo et al., 2020], methods that classify individual samples such as those mentioned above are unsuitable; instead, a clustering approach must be adapted to classify complete rainfall-runoff timeseries', rather than individual observations as has been done in previous works. Steffen and Gomes [2018] demonstrate a framework for classifying flood severity using based on a set of indicators in order to better understand flood formation mechanics. In our study, a comparable timeseries clustering approach is employed, however, in our case for calibration and validation data selection.

As mentioned above, clustering is conducted using features derived from rainfall-runoff events that have first been discretised from continuous data, which are summarised in Table C.7. For clustering, both rainfall and streamflow event statistics are used in order to achieve a subset of events with both diverse streamflow and forcing rainfall event characteristics [Toth, 2009]. The event statistics (called feature set) proposed in this study include total precipitation (P), peak intensity (i_P), mean intensity (i_μ), duration (D), date ($date$), peak flow (q_p), flow volume (Q), hydrograph centroid coordinates (q_{ct}) and (q_{cy}), and hyetograph centroid coordinates (p_{ct}) and (p_{cy}),.

The feature set has a high dimensionality (11) relative to the number of events, or samples (25). Thus, before clustering, principal component analysis (PCA) is used for dimensionality reduction. This combination of dimensionality reduction and clustering has been widely used in literature to reduce the effects of the curse of dimensionality [Frank et al., 2020, Li et al., 2016, Steffen and Gomes, 2018] K-means clustering is used to cluster the 5 first principal components (PCs).

C.3.3.2 Genetic algorithm-based optimisation

A variety of optimisation algorithms have been applied for SWMM calibration [Niazi et al., 2017]. A recent comparison of three evolutionary optimisation algorithms found that the GA outperforms the other algorithms [Kumar et al., 2019]. The GA is an evolutionary technique that searches for a global minimum by iteratively breeding new populations of parameter estimates. Each new population is created by crossing-over and mutating the best population of each previous generation. The initial population is generated by randomly sampling within the uncertainty range of each individual parameter. The uncertain parameters subject to calibration, and their degrees of uncertainty, are identified based on expert knowledge and are listed in Table C.6 below. The uncertainty value indicates the uncertainty of each parameter relative to its initial value. The lower and upper constraints are identified based on physically realistic values for each parameter (irrespective of site conditions). For example, the upper constraint for saturated hydraulic conductivity (KSat) is 120 mm/hr, which is largest value listed in the SWMM manual [James et al., 2010].

C.3.3.3 Performance criteria

The Kling-Gupta Efficiency (KGE) is used as the objective function for the GA [Gupta et al.,

Table C.6: GA Calibration Parameters and Uncertainty Ranges

class	attribute	units	uncertainty	lower constraint	upper constraint
subcatchments	Width	m	± 4.0	0	10^6
subcatchments	PercentSlope	%	± 0.5	0	5
subcatchments	PercentImperv	%	± 0.4	0	100
subareas	N_Imperv	-	± 0.2	0	0.024
subareas	N_Perv	-	± 1.0	0	0.8
subareas	S_Imperv	mm	± 0.4	0	2.54
subareas	S_Perv	mm	± 1.0	0	7.62
infiltration	Suction	mm	± 1.0	0	320
infiltration	Ksat	mm/hr	± 1.0	0	120
infiltration	IMD	-	± 0.5	0	0.4

2009]:

$$\text{KGE} = 1 - \sqrt{(r - 1)^2 + (\alpha - 1)^2 + (\beta - 1)^2} \quad (\text{C.1})$$

where:

$$r = \frac{\text{cov}(q, \hat{q})}{\sigma_q \sigma_{\hat{q}}} \quad (\text{C.2})$$

$$\alpha = \frac{\sigma_q}{\sigma_{\hat{q}}} \quad (\text{C.3})$$

$$\beta = \frac{\mu_{\hat{q}}}{\mu_q} \quad (\text{C.4})$$

where σ and μ denote the standard deviation and mean statistics, respectively, for flow q and predicted flow $\hat{q}$.

$$x \leftarrow \text{argmax}(g(q, f(X, p))) \quad (\text{C.5})$$

where x is the parameter set obtained by the maximisation of the objective function (KGE), $g()$. The objective function quantifies the similarity between the observed flow Q for a given rainfall-runoff event i , and the predicted flow, $f()$. The predicted flow is calculated in SWMM and is a function of all possible model parameter sets X (bound by their uncertainty range) and the precipitation for the same event, P .

Two additional performance criteria are used for model evaluation: the mean volume error (MVE) and peak flow error (PFE)., which are calculated as follows:

$$\text{PFE} = \frac{\max(q) - \max(\hat{q})}{\max(q)} \quad (\text{C.6})$$

$$\text{MVE} = \frac{\sum(q - \hat{q})}{\sum(q)} \quad (\text{C.7})$$

C.3.3.4 Multi-model prediction

A multi-model approach is used in this study, that treats collections of SWMM models, each calibrated to distinct hydrological events, as an ensemble; discrete predictions are obtained by combining the output of each ensemble member. Combining predictions from different parameter sets may be seen as counter-intuitive, especially for time-invariant parameters which are expected to have one true value, significant work in hydrology has demonstrated that there is no single optimum parameter sets; instead, there may be numerous parameter sets that are equally suitable for a given model [Awol et al., 2018, Mediero et al., 2011, Beven, 2012]. Multi-model predictions are supported by pure literature on model ensembles, which asserts that predictions produced by ensembles are known to outperform single models; the success of ensemble is attributable to the degree of disagreement between the ensemble members, also known as ensemble diversity [Brown et al., 2005a]. Combining the predictions from multiple rainfall-runoff events has been demonstrated improving model generalisation in literature [Shamseldin et al., 1997, Beven, 2012]. Most commonly, multi-model approaches combined different types of rainfall-runoff models [Zahmatkesh et al., 2015]. Several studies have demonstrated the effectiveness of combining homogeneous models, calibrated to different observations. Seibert and Beven [2009] use a Monte Carlo simulation to randomly select observations for model calibration and obtain predictions by taking the weighted mean of the 100 best performing models. The multi-model approach is found to outperform the single model.

Our proposed ensemble consists of models that are each calibrated to different rainfall-runoff events; ensemble diversity, which can be measured as the covariance between the predictions of ensemble members, is achieved implicitly by including diverse training data in the form of rainfall events with distinct characteristics. To create an ensemble, n models are calibrated by using a GA so find a solution for Eqn. C.5. Predictions are computed for each individually calibrated model and combined, which is generally expressed by:

$$\hat{q} = h(\hat{q}_1, \hat{q}_2, \dots, \hat{q}_M) \quad (\text{C.8})$$

where $h()$ is the combiner function that combines predictions $\hat{q}$ for models calibrated to events 1 to M .

where Q_i is the runoff corresponding to event i , $h()$ is the model combiner (e.g., equal model weighting, ANN, etc.), and $f(x_n, P_i)$ is a SWMM model with input parameters x_n and forcing precipitation for event i , P_i . There are a wide variety of model combination techniques; the following subsections outline two common methods: weighting combination and stacking.

Weighted combination is among the most popular techniques for aggregating ensemble predictions [Sharkey, 1996]. Uniform weighting is the most common weighting scheme, but various techniques exist for determining non-uniform model weights. We evaluate both uniformly weighted and non-uniformly weighted schemes; for the latter, we utilise the predictions of each calibrated model on the concatenated calibration timeseries.

$$w \leftarrow \text{argmax}(g(Q, \hat{Q})) \quad (\text{C.9})$$

$$\hat{Q} = \sum_{i=1}^M w_i f(X_i, P) \quad (\text{C.10})$$

The weights are determined using the Nelder-Mead simplex direct search algorithm.

Stacking is a generalisation technique in which a model is trained to combine other models (sometimes referred to as a meta-learner). Linear regression is the most common stacking model type [Breiman, 1996a]. However, more complex model types such as artificial neural networks are capable of mapping non-linear relationships between input data (SWMM predictions) and target data (observed flows), providing greater compensation for potential deficiencies of the SWMM models [Sharkey, 1996]. We include both a linear model and ANN in our comparison of combination methods. The configuration of the linear model is trivial; more consideration is necessary for the ANN parameterisation. State-of-the-art modelling practices are used to ensure the ANN is well-generalised. Specifically, stop-training and Bagging are employed; the Levenburg-Marquardt backpropagation is used for training. Grid-search optimisation is performed for the number of Bagged models and hidden layer architecture, which identified optimum sizes of 128 models and 8 hidden neurons.

A benefit of stacking is the ability to incorporate exogenous input variables such as precipitation forecasts and observed flow. In this research, we utilise aerial weighted rainfall for the FMC watershed and observed flow (lagged by 36 hours).

C.3.3.5 Model validation

In order to assess the ability for the model(s) to generalise, we assess model performance using events that are isolated from those use for predictions. For SM predictions, the validation performance for each event is calculated using the remaining 11 events. For the MM predictions, each possible combination is considered; for example, for 6 class 1 events (numbered 1-6) and 6 class 2 events (7-12), validating event 1 is performed using all 6 unique combinations of 5 events from each class ([2-11], [2-10,12], [2-9,11-12], [2-8,10-12], [2-7,9-12], and [2-6, 8-12]). High variability between each combination of selected events is indicative of high sensitivity to the event selection procedure.

C.3.4 Results and discussion

The following sections outline the outcome of the unsupervised event clustering and presents a comparison of the single and multi-model methods outlined above.

C.3.4.1 Unsupervised event clustering

The outcome of the unsupervised event clustering is illustrated in Fig. C.9, which shows the relationships between select indicator variables using scatterplots, and a summary is included

Table C.7: Discretised event dates, statistics, and classification result.

event start [ddMMyyHHmm]	duration [hours]	precip. [mm]	peak intensity [mm/hr]	mean flow rate [m3/s]	peak flow rate [m3/s]	class -
12-05-19 1505	39.5	13	2.8	1.2	1.8	2
25-05-19 0615	12.3	26.5	22.5	2.1	5.3	2
05-06-19 1130	9.8	23.6	21.7	1.9	3.3	2
10-06-19 0915	9.4	8.3	8.3	0.4	2.1	1
13-06-19 0400	12.6	10.4	14.9	0.4	1.8	1
20-06-19 0800	15.1	9.8	5.8	0.4	1.9	1
24-06-19 2220	3.3	11.5	10.8	0.6	2.5	1
29-07-19 2110	12.1	10.8	6.9	0.2	1.2	1
26-10-19 1945	13.7	33.7	11.9	1.8	5.6	2
30-10-19 1235	33.1	31.7	5.5	1.4	3.2	2
27-11-19 0715	5.3	6.1	6.5	0.3	0.6	1
29-12-19 1200	21.3	16.4	5.8	2.4	5.4	2

Table C.8: Ranked performance and variance for KGE, PFE, and MVE.

	KGE rank	KGE var.	PFE rank	PFE var.	MVE rank	MVE var.
CAL	1.083	-	2.167	-	2.250	-
SINGLE	4.000	0.156	3.167	0.119	3.250	0.119
UNIFORM	2.583	0.001	4.000	0.001	3.250	0.001
WEIGHTED	4.167	0.116	4.667	0.101	3.250	0.091
STACK_MLRX	4.917	0.044	3.917	0.038	4.750	0.037
STACK_ANNX	4.250	0.094	3.083	0.072	4.250	0.105

in Table C.7.

The final row and columns show the class values. The resulting classes can be described as large and small events, in terms of flow. There is clear stratification of classes for the total flow (Q), while the stratification is less defined for precipitation (P) and intensity (i_μ). Large events tend to occur in spring and wall, while small events occur throughout the year. Six events from each class are selected for model calibration and validation; models are calibrated to individual models, which are used together according to the methodologies described above.

C.3.4.2 Comparison of single model and multi-model predictions

As mentioned above, this research compares single model with several multi-model prediction techniques; the latter include uniformly weighted, weighted, multi-variable linear regression-based stacking, and artificial neural network-based stacking. The comparison is illustrated in Figure C.10 and summarised in Table C.8.

The range of the boxes indicates the range in performance due to the selection of 5 events from each of classes 1 and 2. In , the mean performance (ranked) and variance are calculated for each event, then averaged across all events. The highest-ranking performance and lowest variance are indicated in bold. The uniformly weighted model predictions have the strongest KGE and MVE performance, whereas the STACK_ANNX model has the strongest PFE.

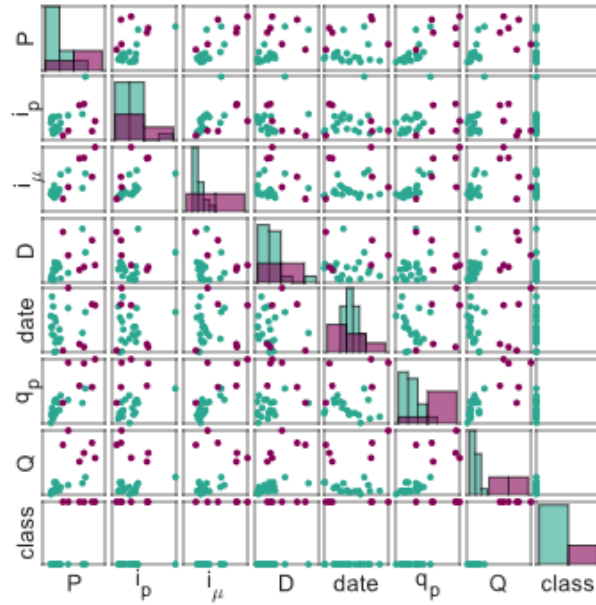


Figure C.9: Scatterplot matrix for clustering indicator variables; classes are distinguished by colour.

The variance in corresponds to the variance associated with the specific event(s) used for the single or multi-model prediction; for example, for validating event 1, the single model performance is calculated for events 2 to 12; the variance is calculated for the distribution of these 11 performance values. The uniformly weighted model has the lowest variance compared to all other methods, while the single model predictions typically have the highest, regardless of the performance metric. This means that model validation performance is highly sensitive to calibration event selection; combining predictions eliminates this variance.

Next, we compare the effects of three subsets of events for multi-model prediction: 5 events from each class (as presented above), using 5 class 1 events, and 5 class 2 events. Validation events belonging to class 1 and 2 are separated into left and right subplots, respectively. Generally, predictions made using models calibrated to a given class perform best on events of the same class; this is observed across different classes and performance criteria.

These trends in performance can be used to guide event selection for calibration procedures. Balanced performance is achieved using a combination of diverse event classifications; if improved performance is desired on events of a given class, predictions can be made using events calibrated based on events in the same class.

C.3.5 Conclusion

This research outlines a novel approach, based on unsupervised clustering, for objectively selecting rainfall-runoff events for SWMM calibration. Once clustered, an event number of events from each class are selected for calibration, ensuring statistically diverse training data. Next, several multi-model prediction strategies, including two weighting schemes and two stacking methods. Overall, the uniformly weighted model combinations have the best performance and lowest sensitivity to specific event selection, across three criteria. Uniformly

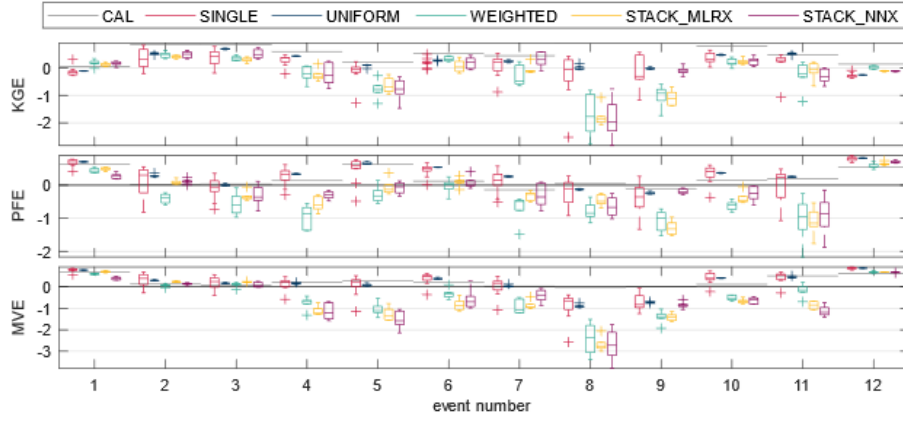


Figure C.10: Validation performance of SWMM configurations for single model (red), uniform weighting (blue), variable weighting (green), MLR-based stacking (yellow), and ANN-based stacking (purple). Calibration performance is indicated by a grey bar.

weighted predictions are compared for predictions made using only events from a single event class and predictions made using an even number of predictions from each class. Predictions made using a single class tend to perform best on validation events with the same class, and worst on events from the other class; the multi-class performance typically lies between the two.

Recommendations for future work include a more robust comparison of multi-event calibration methods (e.g., using multi-objective optimisation) within the proposed framework for event selection. Furthermore, a detailed analysis of calibrated parameter values for models calibrated to distinct events will provide a greater understanding and physics-based support for parameter estimates.

C.4 Spatial bootstrapping for model-free estimation of subcatchment parameter uncertainty for a semi-distributed rainfall runoff model

C.4.1 Introduction

Hydrological model parameter estimation is highly uncertain. Uncertainty estimation often relies heavily on expert judgement. Uncertainty estimates that are too large introduces risks of equifinality [Beven, 2012] and unrealistic parameter values, thus uninterpretable model representations. This work proposes a data-driven alternative, spatial bootstrapping, as a novel approach method for quantifying aggregation uncertainty in semi-distributed hydrological models.

Data and study area: The proposed method is evaluated for a catchment in Toronto Ontario. The study area is the Taylor Creek catchment, a tributary of the Don River that flows through heavily urbanised Toronto into Lake Ontario. This area has historically experienced flash flooding, thus could benefit from more accurate models.

SWMM is a semi-distributed hydrological modelling platform, meaning that a single catchment is discretised into subcatchments. The optimum level of spatial aggregation for semi-distributed rainfall runoffs is a function of the spatiotemporal resolution of the and heterogeneity of the underlying descriptor data, forcing data, model time step, as well as the frequencies of the hydrological signatures present in the area under study. Since the optimum aggregation level is not know, models were created at varying spatial resolutions, by adjusting the follow accumulation threshold used to extract the surface flow network. The flow accumulation grid, which is derived from the DTM, provides a spatially continuous estimate of upstream drainage contribution. A lower flow accumulation threshold results in smaller subcatchments, and vice versa. Varying the level of subcatchment aggregation also permits the study the variability of uncertainty estimation across different subcatchment sizes. Larger subcatchments are expected to be more uncertain, while the uncertainty asymptotically approaches 0 as the subcatchment size approaches the cell size (neglecting measurement uncertainty). SWMM models for Taylor Creek were generated from scratch using the approach and data detailed in Sec. 2.

Model performance is evaluated using the Kling-Gupta Efficiency (KGE; Gupta et al. [2009]) for a 1-year simulation. Since the aim of this work is to estimate uncertainty ranges, models are not calibrated. Performance distributions are obtained by propagating parameter uncertainty through the model using a Monte-Carlo simulation (n=100).

C.4.2 Methods

Bootstrapping is a common statistical technique that uses resampling with replacements to estimate the accuracy of a sample distribution. Semi-distributed hydrological models are parameterised by aggregating descriptors (e.g., landcover) over the footprint of each subcatchment. Such descriptors exist exist at varying spatial resolutions and accuracy levels. Typically, the aggregated model parameter values are tuned within some estimated range. However, boot-

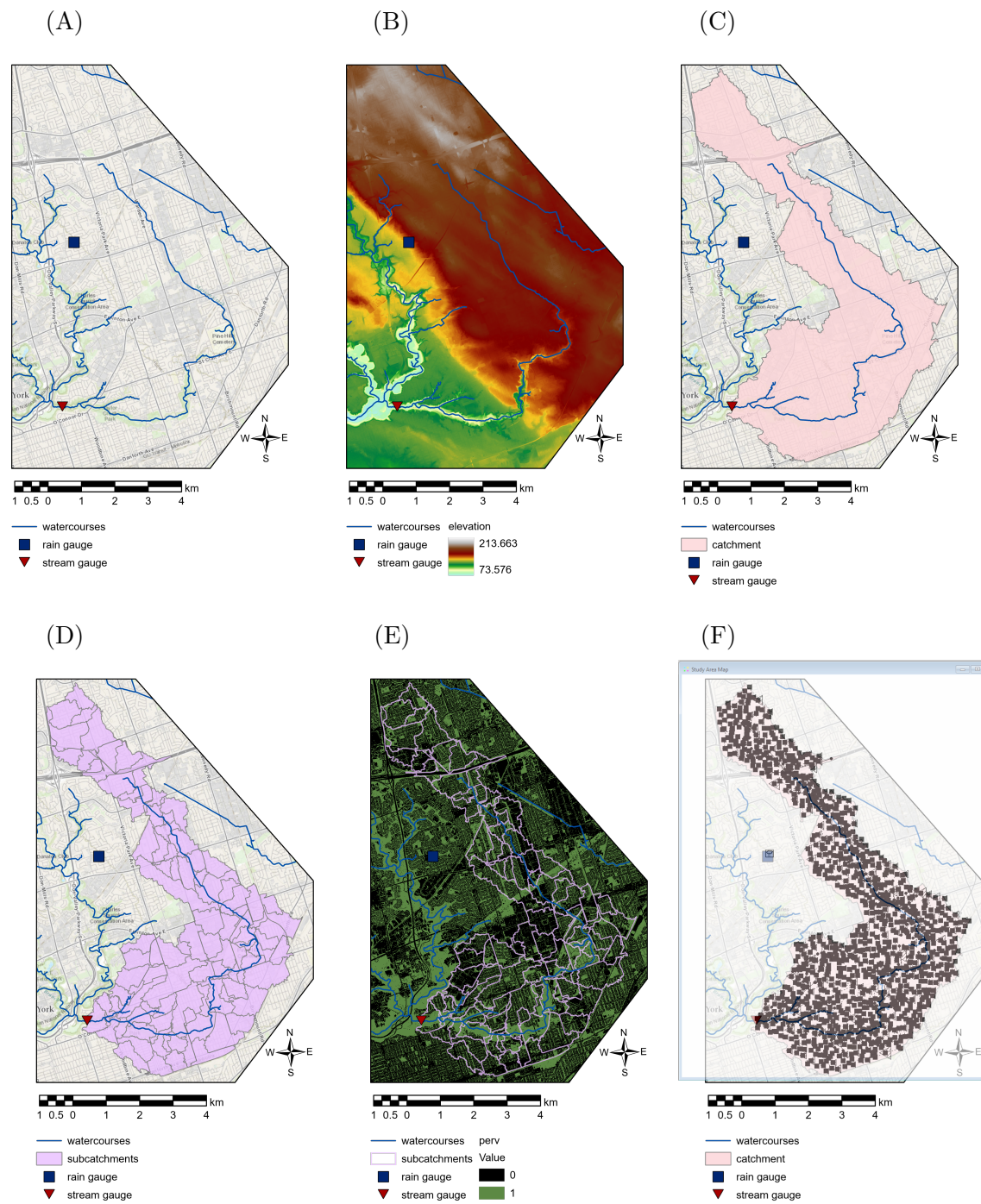


Figure C.11: (A) Taylor Creek tributary (blue lines), stream gauge (red triangle), rain gauge (blue square); (B) Digital Terrain Model; (C) delineated Taylor Creek catchment area (pink); (D) Taylor Creek subcatchments (violet); (E) Impervious (black) and pervious (green) land-cover overlaid by subcatchment boundaries (pink); (F) SWMM model as shown in GUI (back routing network), overlaying the catchment boundary.

strapping provides a way to estimate the uncertainty range using the descriptors directly. In this work, bootstrapping is used to provide new estimates of semi-distributed model parameters, which are propagated through the model to quantify uncertainty of model predictions, and thus, model accuracy.

Spatial data can be highly correlated and therefore, a simple bootstrap resampling approach should not be used [Kasiviswanathan et al., 2016]. Instead, the resampling approach should preserve autocorrelation within the spatial domain. Block bootstrap resampling is one such approach. In block bootstrap resampling, continuous blocks are resampled, instead of individual cells, preserving the autocorrelative pattern [Kasiviswanathan et al., 2016]. The size of the blocks should match the spatial shifts associated with the highest autocorrelation scores. However, spatial patterns may exist at multiple frequencies. For example, the impervious landcover data featured in this case study may contain spatial patterns at the individual building level, or at the block level. An example of 100m blocks overlaying impervious data is shown in Fig. C.12, in which there are patterns of individual buildings, and city blocks of buildings. To account for the varying frequencies of spatial patterns within the descriptor data, a range of block sizes are considered, ranging from 20x20m to 1000x100m.

C.4.3 Results

Model performance, according to KGE, is shown across increasing numbers of subcatchments in Fig. C.13 for several block sizes. The uncertainty of model performance, which is illustrated by the filled envelopes, is calculated by propagating the uncertainty of the 'Percent Impervious' subcatchment parameter ($n=100$ resamples). The solid line indicates mean performance. The mean KGE increases from 0.2 to nearly 0.4 from 1 to 1000 subcatchments, before a sharp decline at 10000 subcatchments. As expected, the uncertainty of model performance decreases inversely to the number of subcatchments. Larger block sizes produce the greatest uncertainty estimates.

To achieve a similar deviation from the spatial mean, the uniform uncertainty is set to $\pm 40\%$ 44% of the 'random uncertainty' models have poorer performance than the worst bootstrap-based model

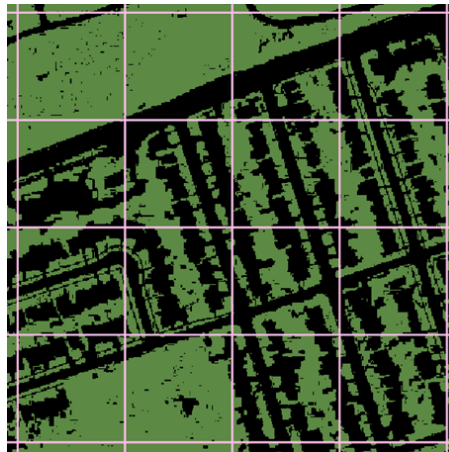


Figure C.12: Pervious (green) and impervious (black) landcover at a 1x1m resolution for an arbitrary residential neighbourhood in Taylor Creek, overlaid by 100x100m grid, illustrating block bootstrapping resampling procedure.

Fig. C.14 illustrates the same results, but performance is plotted against block size. This figure clearly illustrates the high uncertainty of the 1 and 9 subcatchment models at large

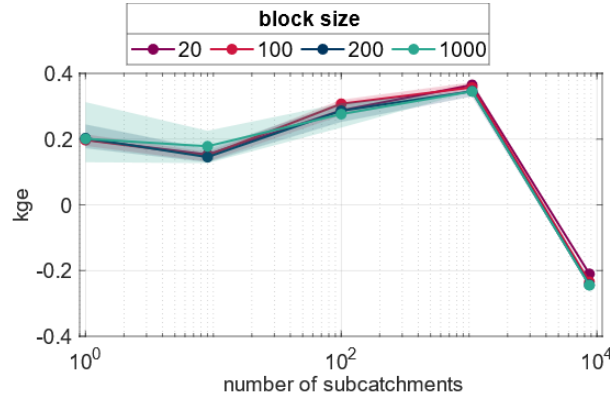


Figure C.13: Model performance (KGE) across increasing subcatchment resolution for four block bootstrap sizes (colours).

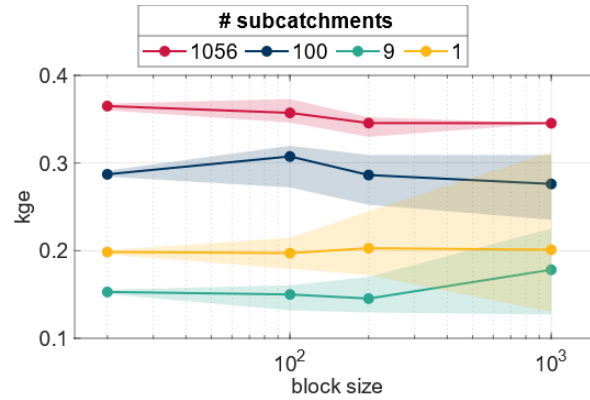


Figure C.14: Model performance (KGE) across increasing block sizes for four subcatchment resolutions (colours).

block sizes. With the proposed uncertainty estimation, the 1-subcatchment model is capable of achieving the same performance as the 100-subcatchment model, within their respective uncertainty range - demonstrating how an uncertainty estimation technique based on model scale can inform improved model calibration techniques. However, both models are outperformed by the 1000-subcatchment model, which achieves good performance despite having very low aggregation uncertainty. This is likely a reflection of the temporal resolution forcing data and highly complex hydrological characteristics in the Taylor Creek catchment.

C.4.4 Conclusion

This study featured a novel application of block bootstrap resampling for model-free estimation of subcatchment parameters for semi-distributed hydrological models. This proof of concept included a single catchment and single model parameter. Future work on this topic should include a larger number of catchments, and uncertainty applied to a greater number of model parameters. An interesting topic would be to apply such a framework to distributed model inputs such as radar-based rainfall estimates.

Appendix D Supplementary ensemble studies

This section includes the supplementary information for Chapter 4, and several other projects that were submitted to conferences in the form of presentations, posters, or conference papers, but not included in the manuscript.

D.1 Supplementary information for Chapter 5

D.1.1 Input variable selection results

Table D.1: List of 25 most useful inputs identified using the PC IVS algorithm for the Bow and Don River watersheds, selected from the set of candidate inputs. Input variables are encoded in the following format "station ID"_"variable"_"statistic"_"lagged timesteps". Variable abbreviations "WL" and "Precip" refer to water level (stage) and precipitation.

rank	Bow	Don
1	05BH004 WL Mean L4	HY022 WL Mean L4
2	05BB001 WL Max L4	HY008 Precip Sum L4
3	05BB001 WL Min L12	HY019 WL Mean L4
4	05BH004 WL Mean L5	HY008 Precip Sum L5
5	Calgary Temp Max L4	HY027 Precip Sum L4
6	05BB001 WL Max L6	HY017 WL Mean L4
7	05BH004 WL Mean L15	HY022 WL Mean L5
8	Calgary Precip Sum L5	HY008 Precip Sum L8
9	Calgary Temp Min L10	HY027 Precip Sum L6
10	Calgary Precip Sum L11	HY017 WL Mean L5
11	05BH004 WL Max L4	HY027 Precip Sum L5
12	05BH004 WL Min L4	HY008 Precip Sum L10
13	05BH004 WL Max L7	HY019 WL Mean L7
14	Calgary Precip Sum L7	HY080 WL Mean L4
15	05BB001 WL Min L15	HY008 Precip Sum L11
16	05BH004 WL Min L8	HY008 Precip Sum L6
17	Calgary Precip Sum L10	HY080 WL Mean L6
18	05BH004 WL Max L12	HY027 Precip Sum L7
19	Calgary Precip Sum L6	HY022 WL Mean L6
20	05BB001 WL Max L5	HY027 Precip Sum L8
21	Calgary Temp Min L15	HY022 WL Mean L7
22	05BH004 WL Min L6	HY080 WL Mean L5
23	05BH004 WL Mean L6	HY017 WL Mean L6
24	05BH004 WL Max L5	HY080 WL Mean L7
25	05BB001 WL Min L9	HY019 WL Mean L6

D.2 Pseudocode

Algorithm 1 Random undersampling

Require:

Set S containing X input features and Y observations, $(x_1, y_1), \dots, (x_m, y_m)$
High stage threshold, θ
 $S_{TS} = S$ where $Y < \phi_{TS}$
 $S_{HS} = S$ where $Y \geq \phi_{HS}$
 $S'_{TS} \leftarrow \text{sample}(S_{TS}, N_{HS})$
 $S'_{HS} \leftarrow \text{sample}(S_{HS}, N_{HS})$
 $S' = S'_{TS} \cup S'_{HS}$

Algorithm 2 Random oversampling

Require:

Set S containing X input features and Y observations, $(x_1, y_1), \dots, (x_m, y_m)$
High stage threshold, θ
 $S_{TS} = S$ where $Y < \phi_{TS}$
 $S_{HS} = S$ where $Y \geq \phi_{HS}$
 $S'_{TS} \leftarrow \text{sample}(S_{TS}, N_{TS})$
 $S'_{HS} \leftarrow \text{sample}(S_{HS}, N_{TS})$
 $S' = S'_{TS} \cup S'_{HS}$

Algorithm 3 SMOTER

Require:

Set S containing X input features and Y observations, $(x_1, y_1), \dots, (x_m, y_m)$
High stage threshold, θ_{HS}

Ensure:

$\phi_{HS}/(1 - \phi_{HS}) \in \mathbb{Z}$
 $N_{synth} \leftarrow \phi_{HS}/(1 - \phi_{HS}) - 1$
 $S_{TS} = S$ where $Y < \phi_{TS}$
 $S_{HS} = S$ where $Y \geq \phi_{HS}$
for $s_i \in S_{HS}$ **do**
 $nn_i = \text{kNN}(S, k)$
 for $j = 1, 2, \dots, N_{synth}$ **do**
 $s_j = nn_i(\text{randi}(1, k))$ ▷ randomly select one nearest neighbour
 $s_{diff} = s_i - s_j$
 $gap = \text{rand}(0, 1)$ ▷ randomly select a point between sample and nearest neighbour
 $s_{synth,i,j} = s_i + s_{diff} \times gap$
 end for
end for
 $S' = S \cup S_{synth}$ ▷ merge original and synthetic data

Algorithm 4 Bagging with resampling

Require:

Set S containing X input features and Y observations, $(x_1, y_1), \dots, (x_m, y_m)$
Learner, $f()$
Number of iterations, T
Resampling function, **resample()**
for $t = 1, 2, \dots, T$ **do**
 $S'_t, D'_t \leftarrow \text{resample}(S, D_t)$
 train $f(S'_t, D'_t)$ ▷ train learner using resampled examples
end for

Algorithm 5 AdaBoost.RT with resampling

Require:

Set S containing X input features and Y observations, $(x_1, y_1), \dots, (x_m, y_m)$
Learner, $f()$
Number of iterations, T
Resampling function, **resample()**
Relative error threshold ϕ
 $D_1(i) \leftarrow \frac{1}{m}$ **for** $i = 1, \dots, m$ ▷ initialise weights array
for $t = 1, 2, \dots, T$ **do**
 $S', D'_t \leftarrow \text{resample}(S, D_t)$
 train $f_t(S'_t, D'_t)$ ▷ train learner using resampled examples and weights
 $\epsilon_t = \sum D_t(i), i = \left| \frac{(f_t(x_i) - y_i)}{y_i} \right| > \phi$ ▷ calculate error rate
 $\beta_t = \epsilon_t^2$
 $D_{t+1}(i) = \frac{D_t(i)}{Z_t} \times \begin{cases} \beta_t, & \text{if } \left| \frac{(f_t(x_i) - y_i)}{y_i} \right| \leq \phi \\ 1, & \text{otherwise.} \end{cases}$ ▷ update weights for next boosting iteration
 $D_{t+1} = \text{normalise}(D_t)$
end for

Algorithm 6 LSBoost with resampling

Require:

Set S containing X input features and Y observations, $(x_1, y_1), \dots, (x_m, y_m)$

Learner, $f()$

Number of iterations, T

Resampling function, **resample**()

Learning rate ν $\{0 < \nu \leq 1\}$

$\hat{Y}_0 = \bar{Y}$

for $t = 1, 2, \dots, T$ **do**

$R_t = Y - \hat{Y}_{t-1}$

$S' \leftarrow \mathbf{resample}(S)$

$\triangleright$ resample input features and residuals

$R'_t = Y' - \hat{Y}_0 + \sum_{t=1}^T \rho_t f_t(X')$ $\triangleright$ calculate the residuals corresponding the resampled data

train $f_t(X', R'_t)$

$\triangleright$ train learner to latest residuals

$\rho_t = \operatorname{argmin}_{\rho} \sum [\hat{R}_t - \rho R_t]^2$

$\hat{Y}_t = \hat{Y}_{t-1} + \nu \rho_t f_t(X)$

end for

D.3 Evaluation of boosting for flood forecasting

D.3.1 Introduction

This case study features Toronto’s Don River watershed, which is illustrated in Fig. D.1. The Don River watershed has an area of 36,000 ha and 1.4 million residents, making it one of the most urbanised watersheds in Canada. Historically, the Don has experienced frequent and severe flooding. ANN based flood forecasting models for this watershed have been shown to produce accurate forecasts for short lead times (1-3 hours) [Snieder et al., 2020]; however, model performance is considered only satisfactory at larger lead times [Mosavi et al., 2018]. Boosting, which is a widely used ensemble-based meta-algorithm for increasing predictor accuracy, is proposed to improve predictions for the Don River [Duffy and Helmbold, 2002]. The following case study compares the effects of common boosting algorithms applied to ANN-based flow forecasting models.

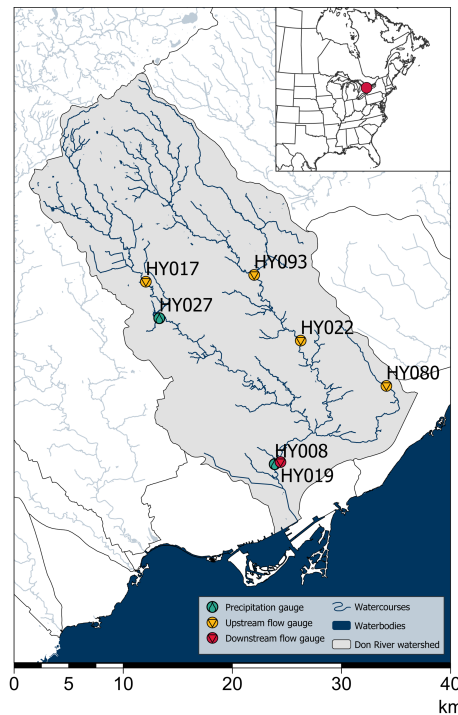


Figure D.1: Map showing locations of upstream stream (yellow), target stream (red) and precipitation (green) gauges for the Don River. The watershed area is marked in grey and hydrographic features are shown in dark blue.

The primary objective of this case study is to compare the effectiveness of three boosting algorithms for improving the performance of flow forecasting models in the Don River, especially with respect to high flows.

D.3.2 Methods

This work uses multilayer perceptron feedforward ANNs as the base learners for the boosting algorithms. Models are trained and tested in a 5-fold cross-validation format. Three boosting algorithms are considered: two variants of AdaBoost.RT [Shrestha and Solomatine, 2006, Shrestha et al., 2007], which use either resampling (.rs) or reweighting (.rw), as well as the

least squares (.ls) variant of GradBoost [Friedman, 2000]. Reviews and pseudocode for each boosting method are included in Sections 4.5.2 and D.2, respectively. A graphical description of each boosting method is also provided in Fig. D.2.

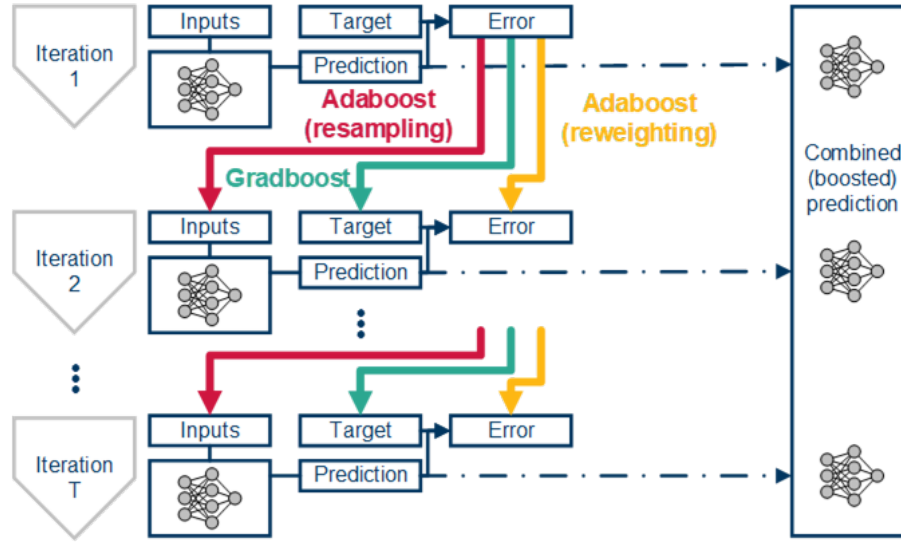


Figure D.2: Graphical description of each of the three boosting methods considered

D.3.3 Results

The effects of boosting are shown in the test event hydrographs in Fig. D.3 (A). The Gradboost models matches observed peak flows closer than the other models. The head maps in Fig. D.3 (B) illustrate the evolution of the weighted cost function across boosting iterations for each timestep in the event. Despite proportionally higher weight being applied to the rising limb and peaks of each event, the prediction error tends to persist across boosting iterations.

Performance of the boosted model ensembles is illustrated in the violin plots below for Nash-Sutcliffe Efficiency (NSE) and Persistence Index (PI). Collectively, the three boosting approaches have relatively indifferent impacts on the overall performance. The Adaboost models have the best performance, with mean NSE and PI values approximately 0.05 better than the base model.

This case study was presented as a poster, shown in Fig. D.5, at the AGU 2019 general assembly.

D.3.4 Conclusion

Despite their popularity in literature, the boosting algorithms produced only minor improvements to the flow forecasting models. Future work on this topic will aim to leverage methods for increasing the diversity of the ensemble member error, through embedded resampling techniques.

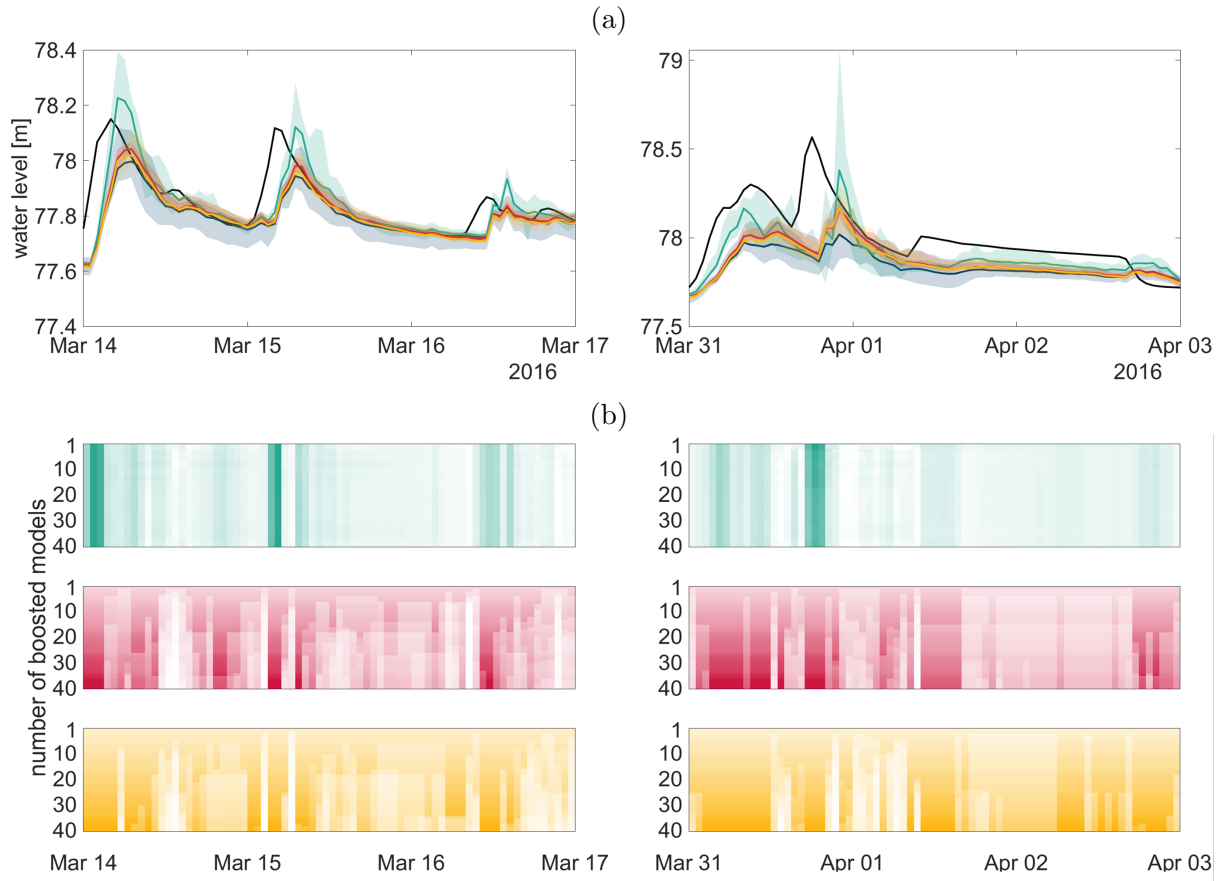


Figure D.3: (a) Hydrographs of two runoff events for observed (black), no boosting baseline (blue), GradBoost.ls (green), AdaBoost.rt.rs (red), and AdaBoost.rt.rw (yellow). (b) Evolution of cost function weights across boosting iterations for each of the three boosting methods.

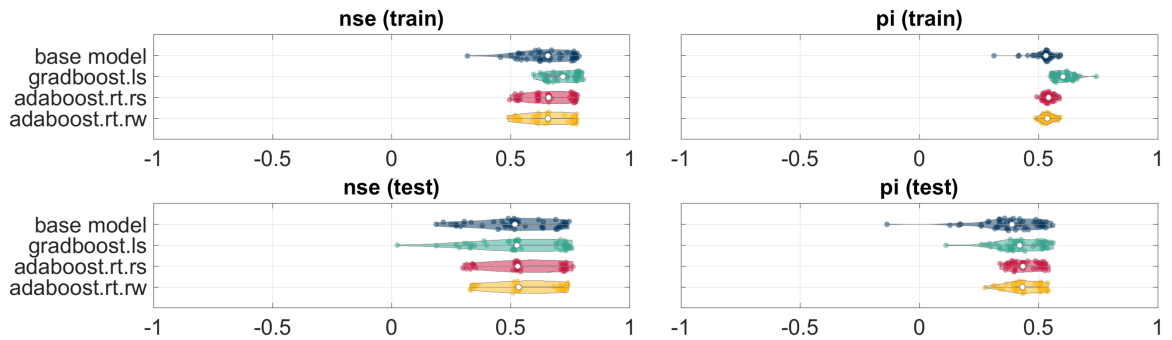


Figure D.4: Violin plots illustrating the training (top) and test (bottom) performance of each configuration, according to NSE (left) and PI (right).

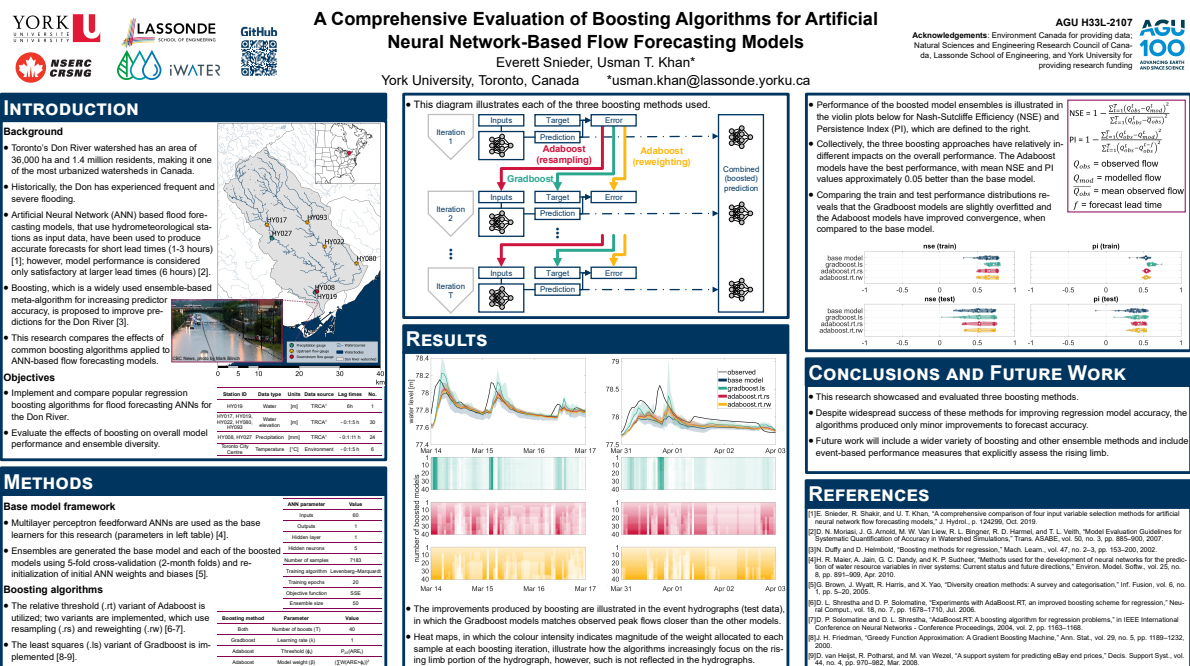


Figure D.5: Poster presented at the AGU 2019 General Assembly.

Appendix E Supplementary large-sample studies

This section includes the supplementary information for Chapter 5, and several other projects that were submitted to conferences in the form of presentations, posters, or conference papers, but not included in the manuscript.

E.1 Supplementary information for Chapter 6

E.1.1 Static attribute statistics

Attributes from [Arsenault et al. \[2020\]](#) are included directly in the dataset, whereas attributes from [Addor et al. \[2017\]](#) and [MacKinnon \[2010\]](#) are calculated from the hydrometeorological timeseries data in HYSETS. Seasonality attributes are calculated for flow and temperature using an additive seasonal decomposition for a frequency of one year. The seasonality is then given as:

$$\text{Seasonality} = 1 - \frac{\sum (w_t - \bar{w}_t)^2}{\sum ((w_t + s_t) - \overline{(w_t + s_t)})^2} \quad (\text{E.1})$$

where s_t and w_t are the seasonal and residual (i.e. white noise) terms of additive seasonal decomposition. Negative seasonality values are truncated to 0, resulting in seasonality values ranging between 0 and 1, corresponding to no and high seasonality.

E.1.2 Sensitivity of silhouette scores to minimum cluster size

The clustering outcome and silhouette scores are highly dependent on the minimum cluster size. Unconstrained k-means clustering results in an optimum K value of 8, however, some of these clusters do not contain enough basins to use for validation. A comparison of silhouette scores across the numbers of clusters for minimum cluster sizes of 2, 8, 16, 32, 64, is provided in Fig. [E.1](#) below. Although greater numbers of clusters produce more hydrologically diverse sets of basins, there are simply not enough basins in each cluster to form a training dataset with balanced hydrological conditions (i.e., an equal number of basins from each cluster).

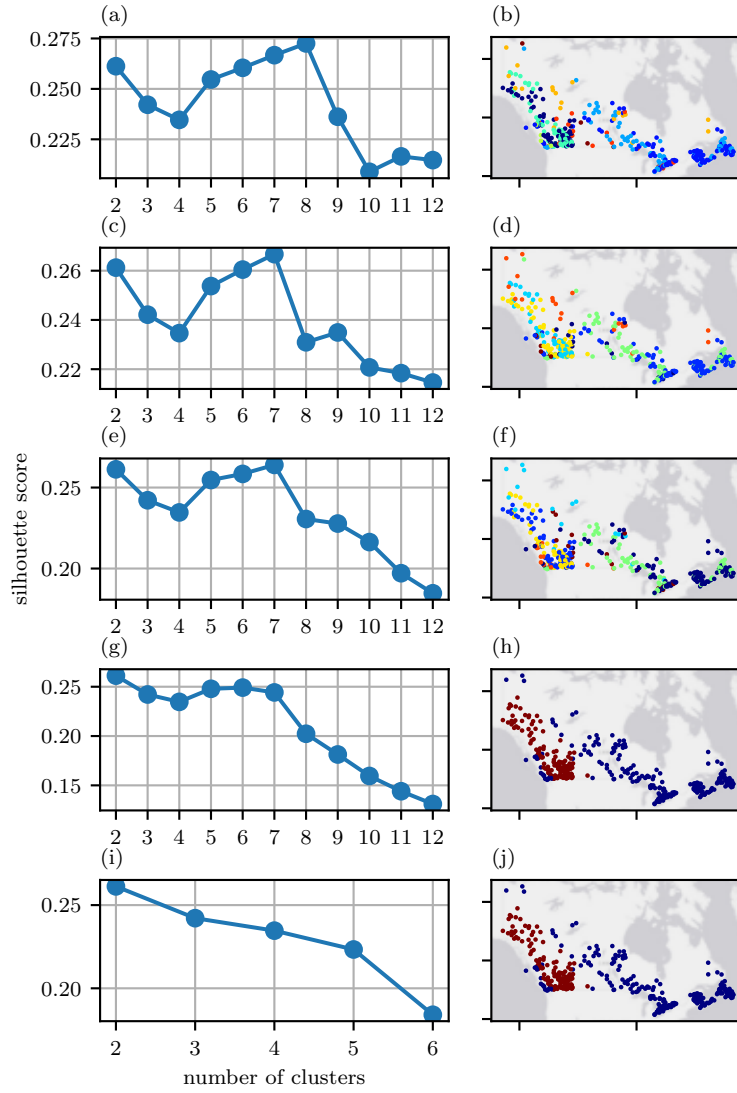


Figure E.1: Silhouette scores (left) for minimum cluster sizes of 2 (a), 8 (c), 16 (e), 32 (g), and 64 (i). Respective clustering results (right) for K values (corresponding to maximum silhouette score) of 8 (b), 7 (d, f), and 2 (h, j). World Gray Canvas basemap provided by ESRI.

E.1.3 Two-cluster attributes

Table E.1: Mean and standard deviation of basin attributes for 2 cluster configuration.

	cluster 0 mean	cluster 0 std	cluster 1 mean	cluster 1 std
elevation (m)	307.05	179.09	1432.84	327.84
slope (deg)	3.11	3.82	16.15	6.84
gravelius (-)	2.11	0.5	1.86	0.37
aspect (deg)	155.91	88.78	169.09	100.34
land use forest (frac)	0.58	0.27	0.6	0.2
land use grass (frac)	0.04	0.08	0.2	0.13
land use wetland (frac)	0.04	0.07	0.0	0.0
land use water (frac)	0.06	0.07	0.02	0.02
land use urban (frac)	0.04	0.1	0.01	0.01
land use shrubs (frac)	0.09	0.11	0.09	0.05
land use crops (frac)	0.14	0.26	0.02	0.11
land use snow ice (frac)	0.0	0.01	0.05	0.1
permeability (log; m2)	-14.19	1.17	-14.07	0.9
porosity (frac)	0.1	0.06	0.11	0.06
precip mean (mm/d)	2.68	1.23	2.14	1.43
precip high (mm/d)	13.39	6.15	10.69	7.14
precip low (mm/d)	1.0	0.0	1.0	0.0
precip mean (mm/mo)	81.2	37.74	64.45	43.81
precip mean (mm/y)	974.38	452.9	773.45	525.76
precip high freq (d/yr)	3.57	0.5	3.33	0.66
precip high dur (d)	1.16	0.11	1.23	0.07
precip low freq (d/yr)	45.0	5.12	45.65	6.79
precip low dur (d)	3.99	1.63	4.64	0.98
baseflow index (-)	0.71	0.13	0.78	0.1
q mean (mm/d)	1.8	1.54	1.86	1.79
q high (mm/d)	16.18	13.88	16.77	16.13
q low (mm/d)	0.36	0.31	0.37	0.36
q high freq (d/yr)	69.38	91.85	47.28	83.23
q high dur (d)	2.67	2.69	2.35	2.54
q low dur (d)	29.66	34.52	42.33	35.39
q zero freq (d/yr)	183.47	970.46	135.96	1064.88
q 95 (mm/d)	6.13	5.27	6.42	5.57
q 5 (mm/d)	0.21	0.27	0.24	0.29
q adf (-)	-14.24	5.33	-11.23	4.58
q seasonality (-)	0.3	0.2	0.51	0.27
q var (mm2/d2)	11.02	26.07	10.2	21.08
runoff ratio (-)	0.6	0.38	0.87	0.57
tmean seasonality (-)	0.8	0.1	0.76	0.11
tmean alltime (c)	3.69	3.81	3.31	3.94
tmax annual mean (c)	31.03	2.27	31.17	2.83
tmin annual mean (c)	-30.24	9.02	-31.26	10.03

E.2 Additional two-cluster comparison performance metrics

The NSE α and β decompositions are taken from [Gupta et al., 2009]:

$$\alpha - \text{NSE} = \sigma_{\hat{q}} - \sigma_q \quad (\text{E.2})$$

where $\sigma_{\hat{q}}$ and σ_q are the standard deviations of the simulated and observed streamflow.

$$\beta - \text{NSE} = \frac{\mu_{\hat{q}} - \mu_q}{\sigma_q} \quad (\text{E.3})$$

where $\mu_{\hat{q}}$ and μ_q are the means of the simulated and observed streamflows.

The overfitting ratio, which quantifies the relative difference in train and test error, is calculated based on the MSE:

$$\text{MSE} = \frac{1}{n} \sum (q_t - \hat{q}_t)^2 \quad (\text{E.4})$$

The overfitting ratio is given as:

$$\text{OVERFITTING RATIO} = 1 - \frac{\text{MSE}_{\text{test}}}{\text{MSE}_{\text{train}}} \quad (\text{E.5})$$

where MSE_{test} and $\text{MSE}_{\text{train}}$ are the MSE scores on the test, and train partitions, respectively. The overfitting ratio ranges $-\infty$ to an ideal value of 1, which corresponds to 0 error on the test dataset. Scores of 0 indicate the same error on train and test data.

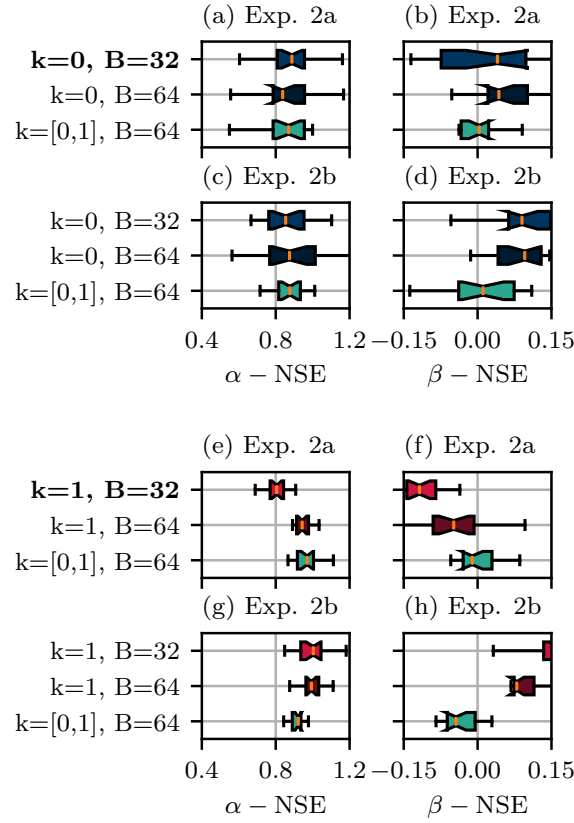


Figure E.2: Boxplots showing the $\alpha - \text{NSE}$ (left) and $\beta - \text{NSE}$ (right) of models evaluated on 32 C0 basins for experiment 2a (a, b), and 2b (b, c), and 32 C1 basins for experiment 2a (e, f) and 2b (g, h). for a lead time of 3 days. The baseline model, which is trained uniquely using the evaluation basins, is indicated in bold. Blue, red, and green colours indicate models trained on C0, C1, and both types of basins.

E.3 Large-scale analysis of machine learning based flow forecasting models for Southern Ontario

E.3.1 Introduction

Machine learning is increasingly used for modelling the rainfall runoff relationship [Mosavi et al., 2018]. Despite widespread advances in ML-based approaches to flood forecasting, there are no operational ML-based forecasting applications in Ontario, Canada. Most research advances ML-based techniques for small numbers of watersheds [Tyralis et al., 2019a], which creates a gap in knowledge surrounding the transferability and interpretability of results between regions with different hydrometeorological conditions.

The objectives of this work are (i) to develop an automated, large-scale ML-based flow forecasting platform for Southern Ontario, Canada. (ii) to evaluate the transferability and interpretability of model behaviour across varying quantities of data availability and watershed characteristics.

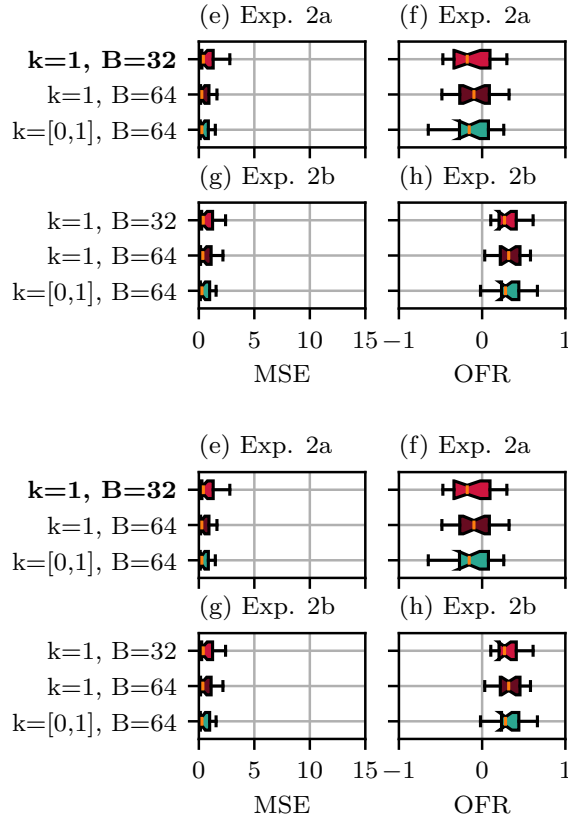


Figure E.3: Boxplots showing the MSE (left) and OFR (right) of models evaluated on 32 C0 basins for experiment 2a (a, b), and 2b (b, c), and 32 C1 basins for experiment 2a (e, f) and 2b (g, h). for a lead time of 3 days. The baseline model, which is trained uniquely using the evaluation basins, is indicated in bold. Blue, red, and green colours indicate models trained on C0, C1, and both types of basins.

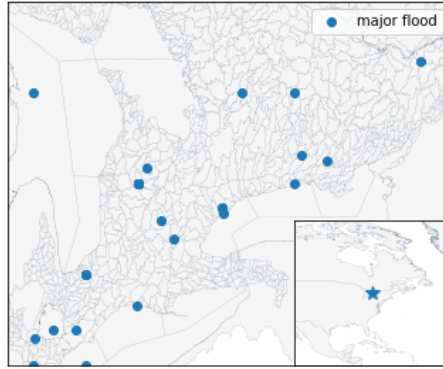


Figure E.4: Locations of major floods in Southern Ontario that occurred between 1900 and 2021. The floods shown are from the Canadian federal disaster database, thus are only those that meet the Government of Canada’s definition of a disaster [Government of Canada and Public Safety Canada, 2021].

E.3.2 Methods

This study considers all watersheds within Southern Ontario (those that fall within the 45.5th parallel). Out of more than 700 watersheds in this region, 229 contained sufficient flow data

(2+ years) to be trained and evaluated. The following framework was used to generate models for each watershed. (1) for each watershed, a feature set is generated using flow data within the watershed and meteorological data from stations within the watershed. If the watershed does not contain any meteorological stations, the nearest station is used. (2) inputs are lagged at several recurrences to account for temporal patterns, and min-max normalised between 0 and 1. (3) The records are divided into 50-50 time-continuous splits for training and testing data. (4) Individual ML models are trained and evaluated. This process is shown in Fig. E.5.

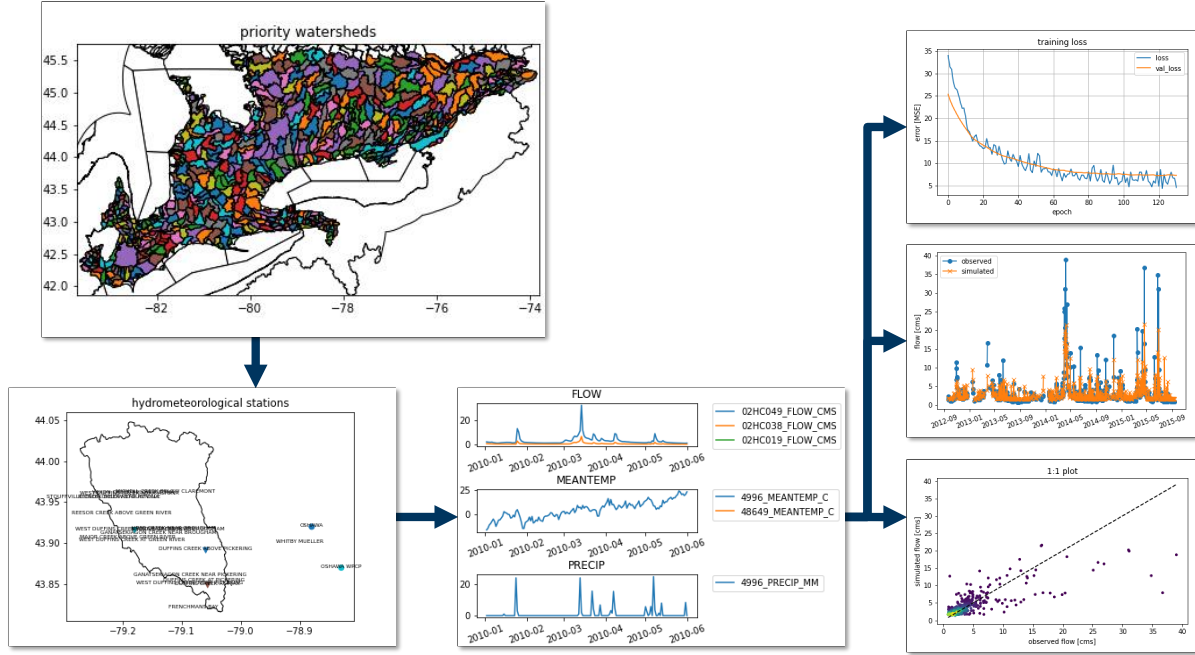


Figure E.5: Diagram of data-processing pipeline.

The models consist of feed-forward MLP ANNs. Bagged ensembles ($m=10$), which means that each model is trained to a resampled (with replacement) training dataset. Early-stopping with a patience value of 10 is applied to 50% of the training data (randomly selected). A network architecture of X-5-5-1 is used, where x is the number of inputs for a given watershed, with ReLu activation functions between the input and hidden layers. A drop-out rate of 0.5 is used for further regularisation. Adam backpropagation is used with a learning rate of 0.001, minimising MSE. Watershed attributes for each watershed are calculated using the respective hydrometeorological data and aggregated spatially continuous data.

The models are evaluated using two performance metrics: the NSE and PI. The NSE is the MSE normalised by the mean observed flow, while the PI is the MSE normalised by the error of the naive model.

E.3.3 Results and discussion

The outcome of training each ANN is illustrated in Fig. E.6, which illustrates model generalisation by comparing model performance on the training dataset and the test dataset. Good generalisation is indicated when the test performance is equal or better to the training perfor-

mance. According to Fig. E.6, most models are well generalised, which is attributable to the regularisation techniques used (Bagging, dropout, and early-stopping).

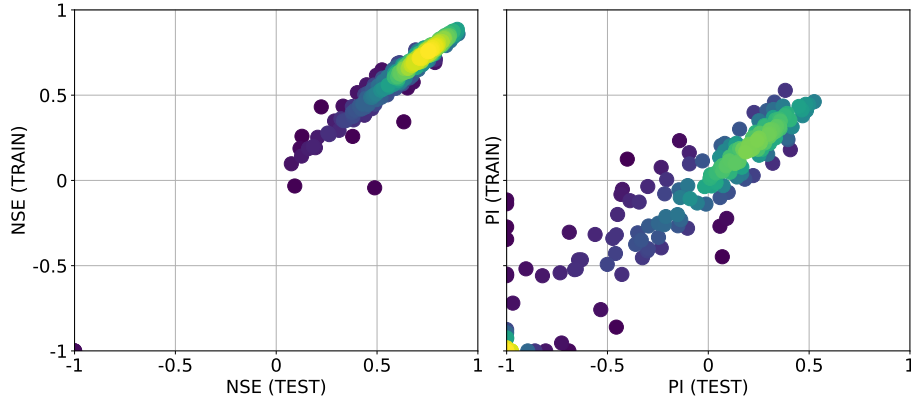


Figure E.6: Performance of models on the train (y-axis) and test (x-axis) datasets according to NSE (left) and PI (right). Points along and below the diagonal are considered to have good generalisation. Colour illustrates the density of points.

The spatial distribution of model performance is shown in Fig E.7 below. The NSE tends to be good (>0.80) in all areas except for in the Greater Toronto Area (GTA), which is highly urbanised, thus, features a comparatively faster runoff response compared to non-urbanised watersheds. The PI has an inverse spatial trend; north watersheds with a high NSE tend to have a low PI.

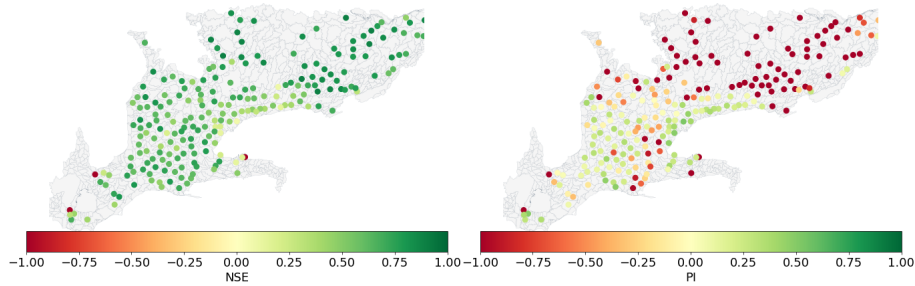


Figure E.7: Spatial patterns in performance of each watershed. Colour indicates performance according to NSE (left) and PI (right). Points are located at the centroids of each catchment.

There is a variable number of stream gauges and meteorological stations available for each watershed. Fig. E.8 groups model performance based on the number of available gauges for flow, precipitation, and temperature. Generally, there is not a significant change in model performance across increasing numbers of stations, except for the number of flow stations according to PI. The PI improves considerably across 1-3 gauges. Poor performance with 4 or 5 gauges is not reliable, as there is only one of each case.

In order to better understand spatial patterns in model performance, we calculate hydrological and physical characteristics for each watershed. These largely mirror those included in the CAMELS dataset [Addor et al., 2017]. Figs. E.9 and E.10 illustrate hydrological and physical characteristics for Southern Ontario watersheds.

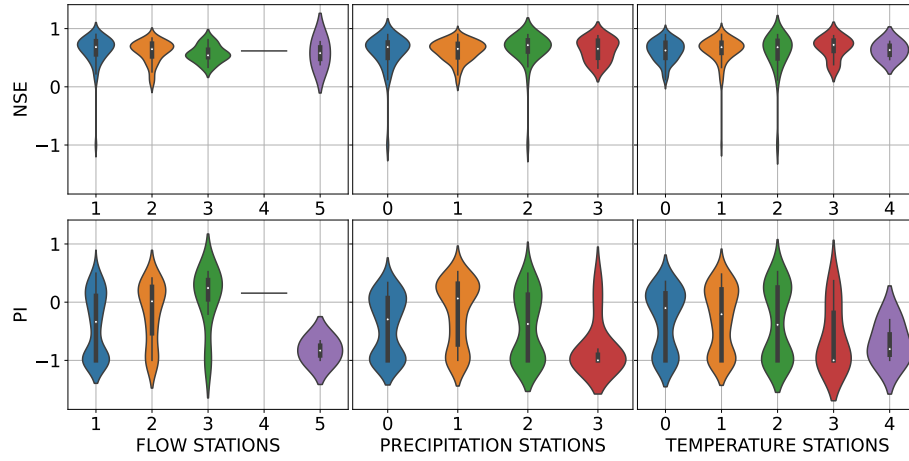


Figure E.8: Performance of models according to NSE (top) and PI (bottom) against increasing numbers of available flow (left), precipitation (middle), and temperature (right) stations.

The correlogram in Fig. E.11 summarises collinearities between watershed characteristics. Naturally, there is a negative correlation between soil textures, as well as between high flow frequency and baseflow index.

An RF is used to determine which watershed characteristics have the most influence on model performance. This is measured by evaluating the RF’s ability to predict flood forecasting model performance, based on watershed characteristics, then measuring the feature importance of each characteristic. Seasonality, high flow duration, and low flow frequency are found to be the best indicators of model performance. Feature importance values with respect to NSE and PI are shown in E.12.

The performance scores of NSE and PI for each model are plotted against two important characteristics in Fig. E.13, annual seasonality, and high flow duration. There is a positive correlation between NSE and each characteristic, whereas PI tends to be higher in watersheds with low seasonality and short high flow durations. The NSE of models tends to improve over increasing seasonality, while the PI declines. This result indicates an over-dependence on autoregressive input features. In contrast, watersheds with low seasonality tend to have comparatively better PI performance, as they don’t develop such a dependency.

E.3.4 Conclusion

This work looked at trends in model performance across varying data availability (in terms of number of stations). We showed that increased upstream flow data considerably improved the PI of models. Next, we illustrated that spatial patterns in model performance is closely tied to flow seasonality. Results highlighted the risk of over-dependence on autoregressive input features in highly seasonal watersheds. Future work on this topic will consider regularisation techniques to eliminate this dependence, by adapting training methods based on watershed characteristics.

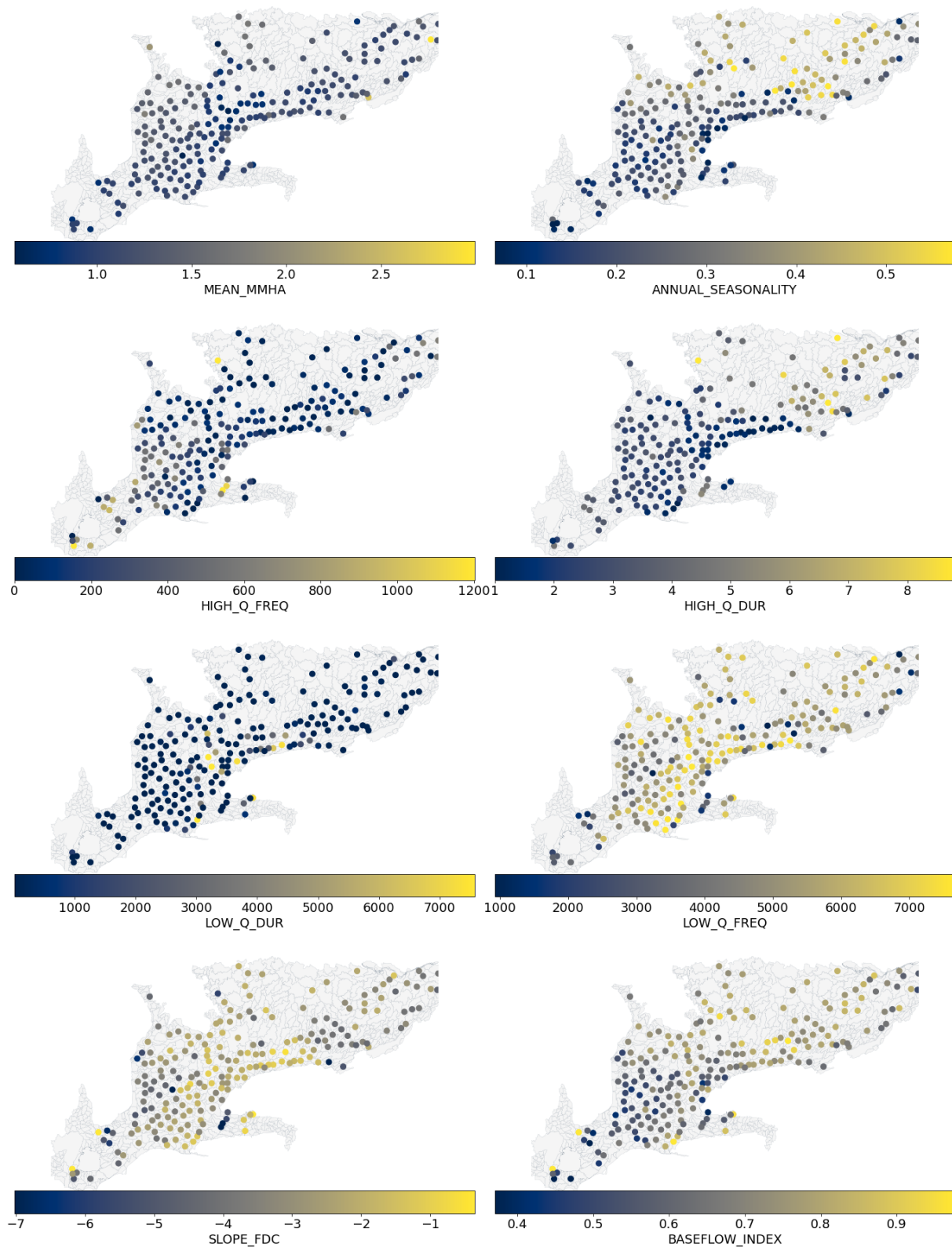


Figure E.9: Hydrological characteristics, including mean flow (MEAN_MMHA), annual flow seasonality (ANNUAL_SEASONALITY), high flow frequency (HIGH_Q_FREQ), high flow duration (HIGH_Q_DUR), low flow duration (LOW_Q_DUR), low flow frequency (LOW_Q_FREQ), flow duration curve slope (SLOPE_FDC), and baseflow index (BASEFLOW_INDEX).

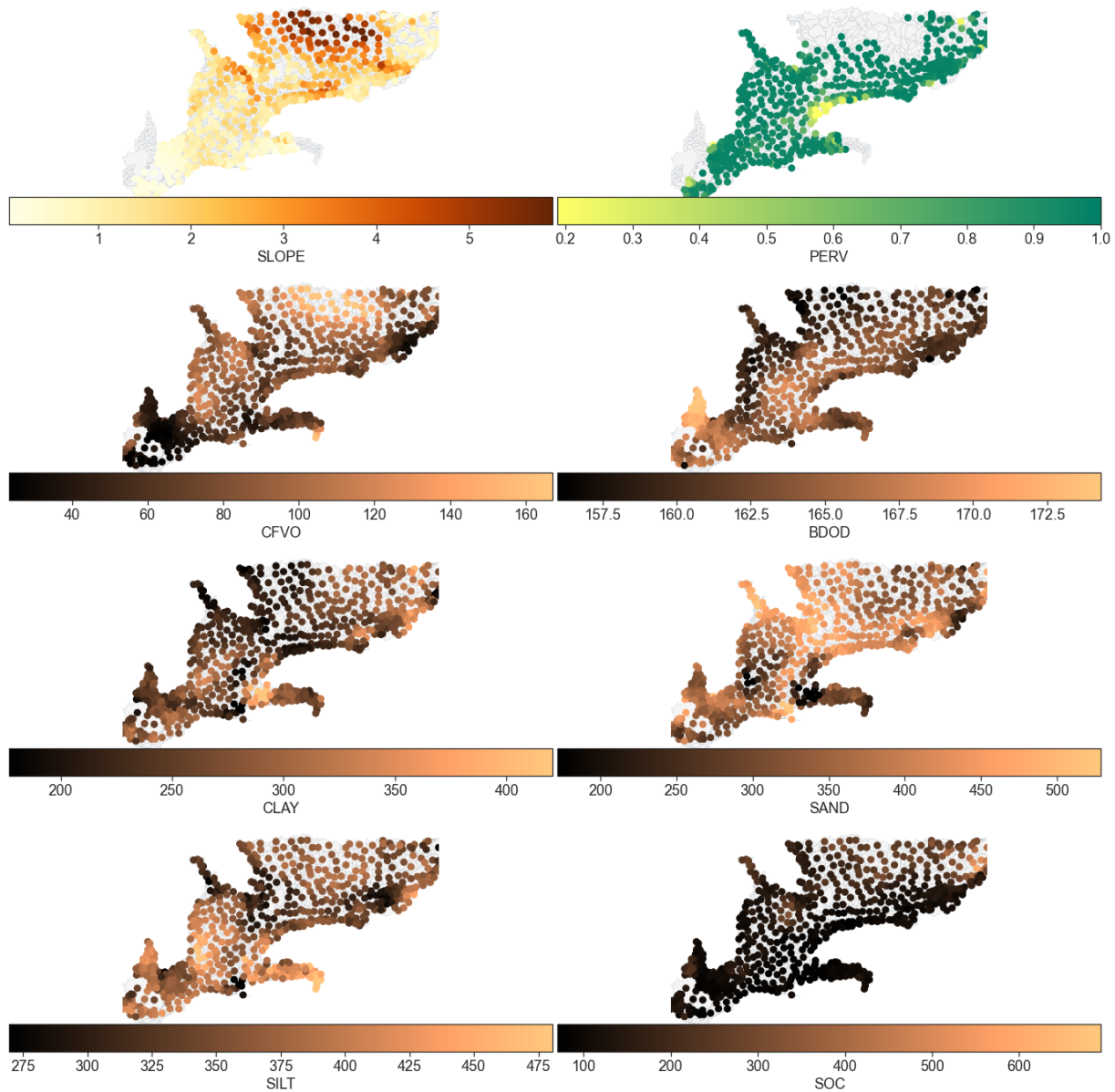


Figure E.10: Physical watershed characteristics, including mean slope (SLOPE), perviousness fraction (PERV), volumetric volume of coarse fragments fraction (CFVO), bulk density of fine earth fraction (BDOD), proportion of clay (CLAY), and (SAND), silt (SILT), and soil organic carbon content (SOC).

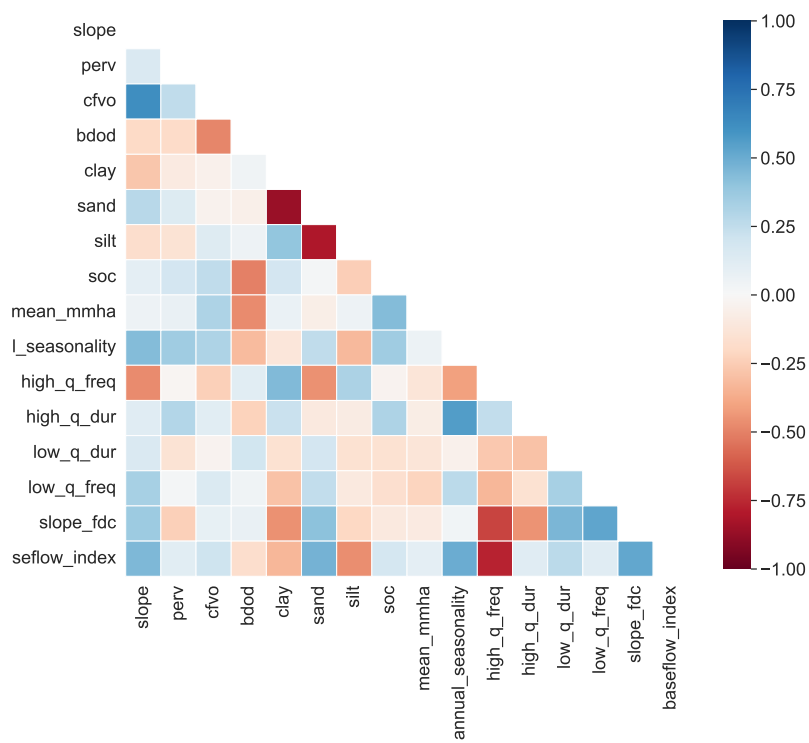


Figure E.11: Correlogram illustrating linear relationships between watershed characteristics.

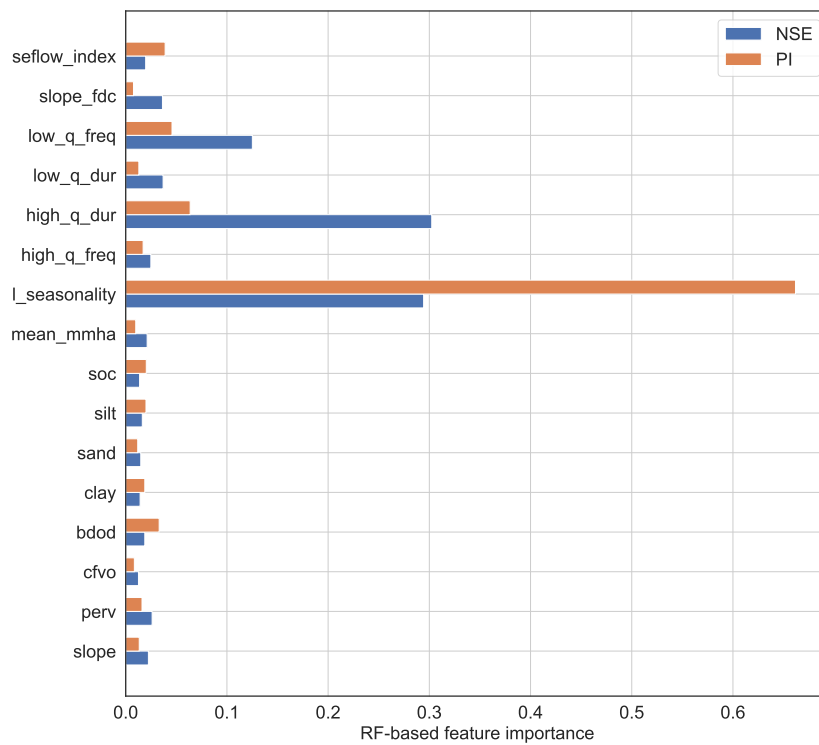


Figure E.12: Feature importance of watershed characteristics for predicting model performance of Southern Ontario watersheds.

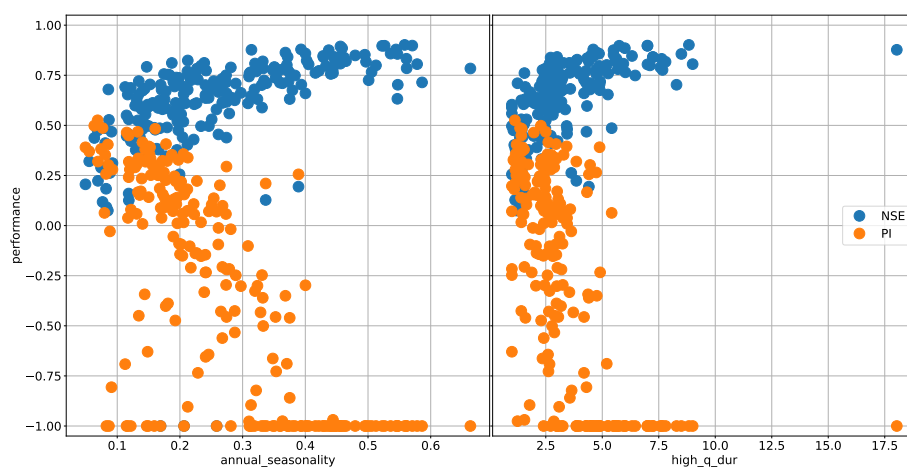


Figure E.13: Model performance against annual flow seasonality (left) and high flow duration (right).

E.4 Large-scale evaluation of temporal trends in ANN behaviour for daily flow forecasts in Canadian catchments

E.4.1 Introduction

Many hydrological systems are non-stationary, which is commonly attributed to factors such as climate change and urbanisation [Wang et al., 2016, Tyralis et al., 2019a, Yaseen et al., 2015]. Machine learning models typically assume stationarity (i.e., hydrological processes do not change in time). Failure to properly account for stationarity can lead to non-optimum model parameterisation [Myers et al., 2021].

The objectives of this study are (i) to evaluate temporal trends in ML-based streamflow forecast performance and (ii) to evaluate calculate temporal trends in feature importance. (i) is accomplished by evaluating models in 1-year time-continuous increments, under different training configurations, to assess whether there is a decay in performance in time, due to non-optimum model parameters. (ii) is accomplished by quantifying feature importance using random forests trained to 1-year time-continuous increments.

E.4.2 Methods

This work uses the HYSETS dataset, which is a hydrometeorological dataset for North America assembled by Arsenault et al. [2020] and described in Sec. 1.5. In this study, daily hydrometeorological timeseries’ are used to forecast streamflow at a lead time of 1 day.

ANNs are trained to individual basins. A hyperparameter optimisation on a subset of randomly sampled basins informed an architecture of 21-16-1 input-hidden-output layers, ReLu activation functions, Adam optimisation with a learning rate of 0.0001, a dropout rate of 0.2, and early stopping with a patience value of 10 on 25% of the training data.

To evaluate temporal trends in model behaviour, we consider three training configurations. (1) ‘fixed training data’, in which models are trained to a fixed 20-year period and evaluated incrementally on 1-year test periods. (2) ‘recent training data’ where models are trained to the 20-year period prior to the test periods. (3) ‘all training data’ where models are trained using periods starting at the beginning of the fixed periods, up to each test period. These three configurations are illustrated in Fig. E.14. While the Fig. shows 1-year evaluation periods, the experiment was conducted using 2-year partitions, due to the high computational expense of the experiment.

Temporal trends in feature importance are evaluated using a similar partitioning scheme, which are used to train RF models to measure feature importance. Unlike ANNs, RFs intrinsically measure feature importance, and are comparatively much quicker to train. They are also less likely to overfit on small, 1-year training partitions. The feature selection method is described in Fig E.15.

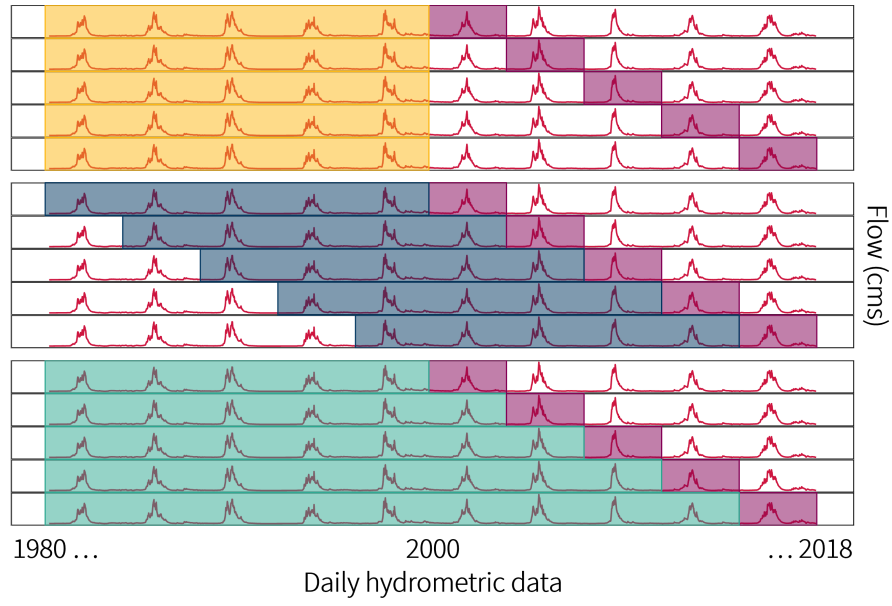


Figure E.14: Example of target flow (red) overlaid by partitioning schemes for ‘fixed training data’ (yellow), ‘recent training data’ (blue), and ‘all training data’ (green). Models are evaluated in 1-year time-continuous segments (purple) in increments of one year.

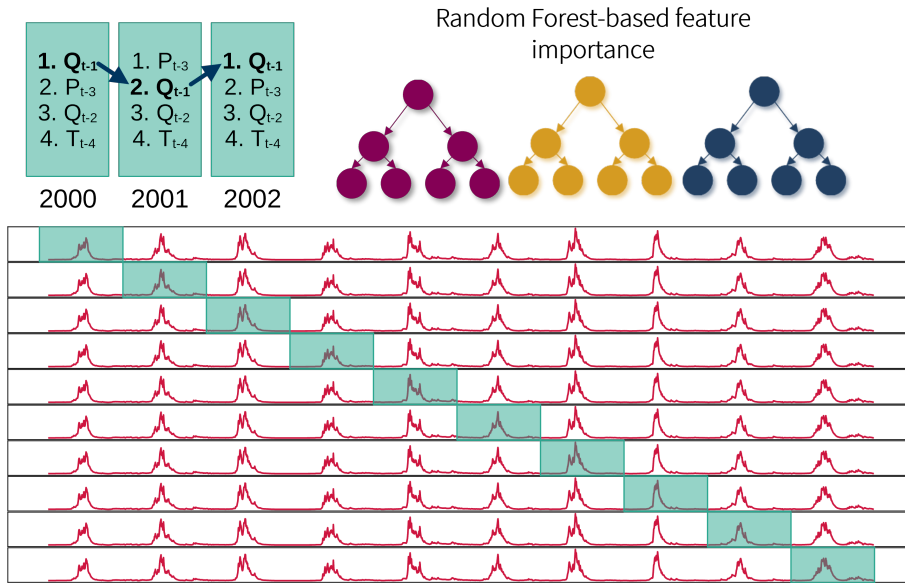


Figure E.15: Example of target flow (red) overlaid by sliding training window (green) used to incrementally train RF models, which are used to quantify temporal feature importance. An example of sequential feature rankings is shown in the top left.

E.4.3 Results and discussion

Temporal trends in model performance are shown in Fig. E.16 according to four metrics. While performance is expected to change year to year, as some years will experience larger runoff events than others, the NSE and PI metrics are normalised on the observed flow for each respective evaluation period, which improves the comparability of scores between years. Overall, the model performance tends to be reasonably stable year-to-year. There is little to separate the models trained to a fixed period of data, with those that were trained with more recent data. One exception is the PI, where the different training configurations diverge as of approximately 2010. The two configurations that use recent data tend to retain good performance, whereas the model trained to a fixed period sees a deterioration in performance, providing evidence towards worsening performance under non-stationarity. All models experienced a drop in performance during the 2012-2013 test period, most notably according to the high flow NSE. However, the model trained on all available historic data performed the best during this period, according to the PI metric.

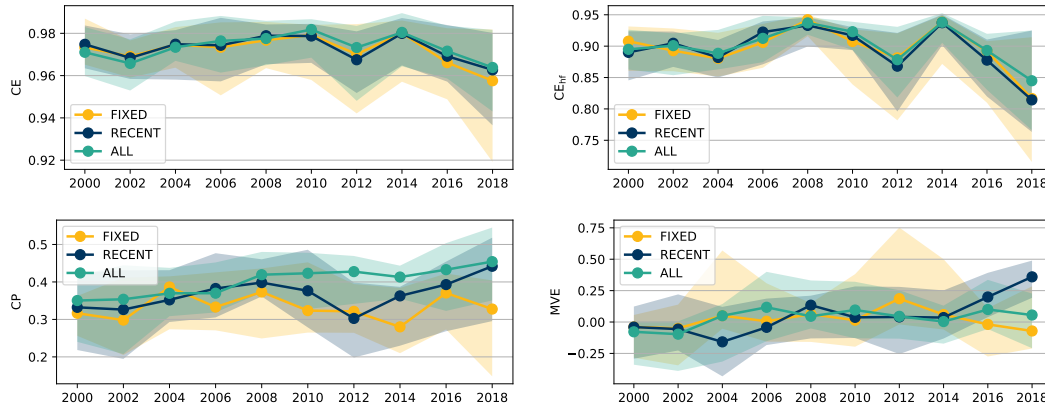


Figure E.16: Temporal trends in model performance for 'fixed' (yellow), 'recent' (blue), and 'all' (green) training configurations, according to NSE (labelled CE, top left), NSE on high flow (labelled CE_{hf} , top right), PI (labelled CP, bottom left), and MVE (bottom right). Lines indicate mean performance across basin, while the filled uncertainty envelope shows 95th percentiles of the basin set.

Aggregated feature importance, across all basins, is shown in Fig. E.17. Consistent with previous feature selection work [Snieder, 2019], flow is the most important feature, followed by meteorological features. Feature importance is relatively constant, aside from a minor increase in meteorological feature importance from years 2025 to 2018.

Figs. E.18 and E.19 show temporally aggregated feature importance on a basin basis. According to Fig. E.18, flow and temperature tend to have consistent importance across Canada. In contrast, the importance of precipitation features has hot-spots in three regions: Western British Columbia, Southern Ontario, and Atlantic Canada. This result is consistent with hydrological knowledge of Canada, as these regions are known for having particularly long or intense rainfall events.

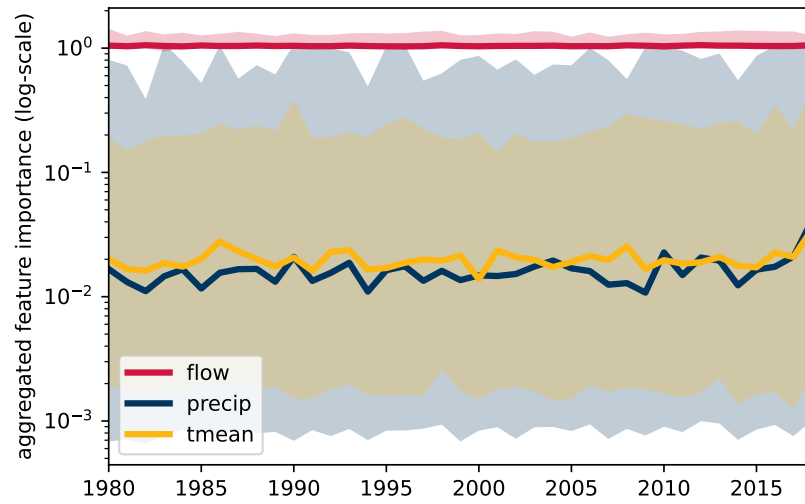


Figure E.17: Temporal trends in feature importance for flow (red), precipitation (blue), and temperature (yellow).

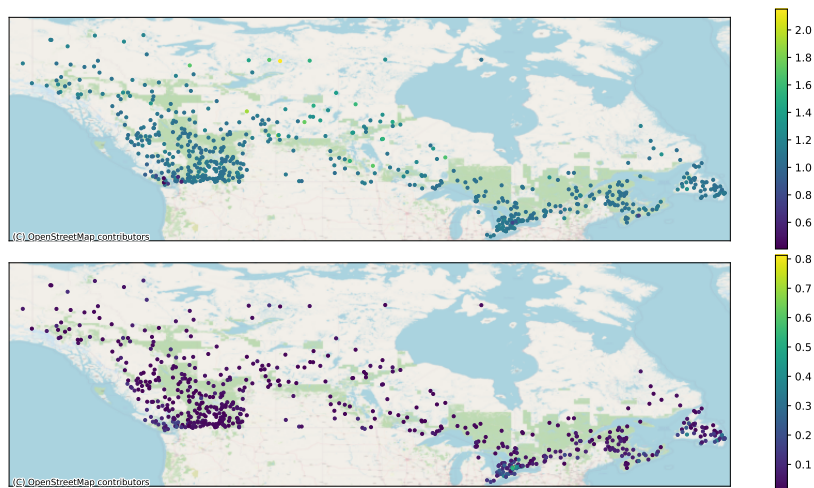


Figure E.18: Mean feature importance of flow (top) and temperature (bottom) shown spatially. The colour gradient from yellow to purple correspond illustrates most important to least important, respectively.

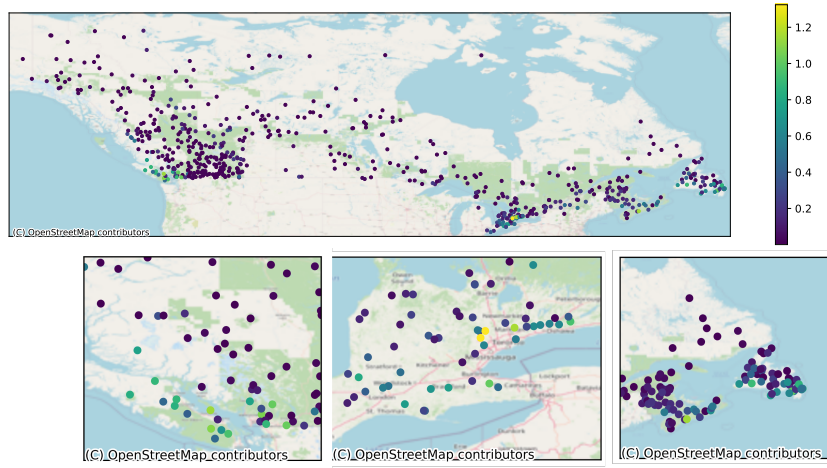


Figure E.19: Mean feature importance of precipitation (top) across Canada, and zoomed subplots showing for basins in western British Columbia (bottom left), Southern Ontario (bottom middle), and Atlantic Canada (bottom right). The colour gradient from yellow to purple correspond illustrates most important to least important, respectively.

E.4.4 Conclusion

Future work should look more in depth at which basins experience performance decay over time, and whether that can be attributed to non-stationarity of the input features, target features, or their relationships. The comparatively strong performance of the ‘all training data’ models make the case for training models using all available streamflow records, suggesting that there is little value in dropping very old data, in this case from 30 years prior, despite the non-stationarity of rainfall-runoff systems.

Spatial analysis of feature importance for models of basins across Canada reveals hot-spots where precipitation is comparatively more important than in other regions. This is an important consideration in selecting precipitation forcing data, which can take many forms (e.g, point-source, radar-based, etc.) and contain varying degrees of uncertainty.

E.5 Cluster-based hyperparameter optimisation for LSTM-based flow forecasting in Canadian catchments

E.5.1 Introduction

The performance of ML-based flow forecasting models is heavily dependent on the selection of hyperparameters [Khan et al., 2018]. The selection of hyperparameters for ML-based models is often done based on reference studies, or based on a simple grid-search. These approaches neglect the hydrological characteristics of the system(s) under study. The complexity of ML models can be expected to scale with the complexity of the hydrological system. Hyperparameter selection is also a function of the amount of available training data. High complexity models are prone to overfitting small training records, making it an important consideration for hyperparameter selection [Gauch et al., 2021].

Usage of fixed model parameters across large basin samples reveals spatial patterns in model performance. The following work aims to determine the extent to which poor hyperparameter specification is responsible for these clusters of poor performing models.

E.5.2 Methods

The interplay between hyperparameters and hydrological complexities is analysed under two modelling frameworks: for locally trained, and regionally trained models.

For locally trained models, we want to determine whether model hyperparameter transfer is related to the hydrological characteristics of the basins. K-Means clustering, based on 37 static basin attributes, is used to group basins that have similar hydrological characteristics. Next, the hyperparameters for six basins in each cluster are optimised using Gaussian-Processes Bayesian optimisation, which is much faster than an exhaustive hyperparameter grid-search. The optimum hyperparameter sets for each cluster are then transferred to basins with similar and distinct characteristics (i.e., in the same cluster and in other clusters).

Table E.2: LSTM hyperparameter ranges and search distributions

Name	Description	Range	Distribution
Dropout rate	Regularisation technique	0 – 0.5	Uniform
Number of cells	LSTM cells in single hidden layer architecture	1 – 64	Log-uniform
Batch size	Samples used in each parameter adjustment	32 – 512	Uniform
Learning rate	Size of parameter adjustment	1E-4 – 1E-1	Log-uniform
Sequence length	‘Look back’ period for each prediction	2 – 365	Log-uniform
Epochs	Number of passes through the training dataset	10 – 400	Uniform

E.5.3 Results and discussion

E.5.3.1 Hyperparameter transfer across locally trained models

The outcome of basin clustering is shown in Fig. E.20. Clusters tend to be closely grouped in space, despite no spatial attributes, due to the spatial autoregressive nature of hydrological and physical characteristics.

The most dominant clustering features are shown in E.21, according to the implicit feature importance values of an RF trained to the clustering results. Topographic and precipitation features have the largest influence on the clustering result.

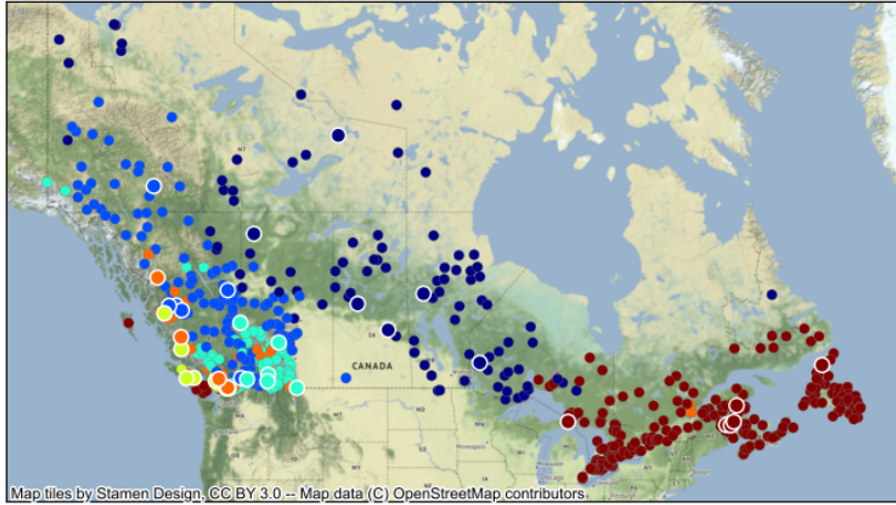


Figure E.20: K-Means clustering results ($K=6$) for basins across Canada. Enlarged markers with white borders were selected, based on Euclidean distance to cluster centroid, for the analysis of hyperparameter optimisation and transfer.

Fig. E.22 illustrates the hyperparameter transfer across basins with similar, and distinct hydrological characteristics. Each axis is grouped by cluster, resulting in 36 tiles. Tiles along the diagonal correspond to hyperparameter transfer between basins in the same cluster. The diagonal of individual cells represents equal source and target basins, thus, will have no change in MSE. According to this figure, there are no significant patterns related to hyperparameter transfer between similar basins, or basins in other clusters. One basin (972) exhibits a decrease in performance with almost any transferred hyperparameter set. In contrast, one basin (1477) exhibits a strong decrease in performance with transferred hyperparameter sets, indicating that the hyperparameter optimisation converged poorly. However, in most cases, transferring optimum hyperparameters between basins, whether or not they have similar hydrological characteristics, has only a small impact on model error. The 36 hyperparameter sets (each optimised for individual basins) are highly variable, which means that there are broad ranges of acceptable hyperparameters. This result may be due to the numerous regularisation techniques employed during training (dropout, early stopping), and may not hold without the use of such techniques.

Next, hyperparameter transfer between regional models is considered. Due to the computational burden of training regional models, the following results include only a single hyperparameter, the number of hidden cells. As with locally trained models, this experiment aims to determine the extent to which hyperparameters are a function of the hydrological diversity, thus complexity, contained within a training dataset.

Fig. E.23 shows the selection of 6 similar, and 6 diverse basins, based on the K-Means clustering results.

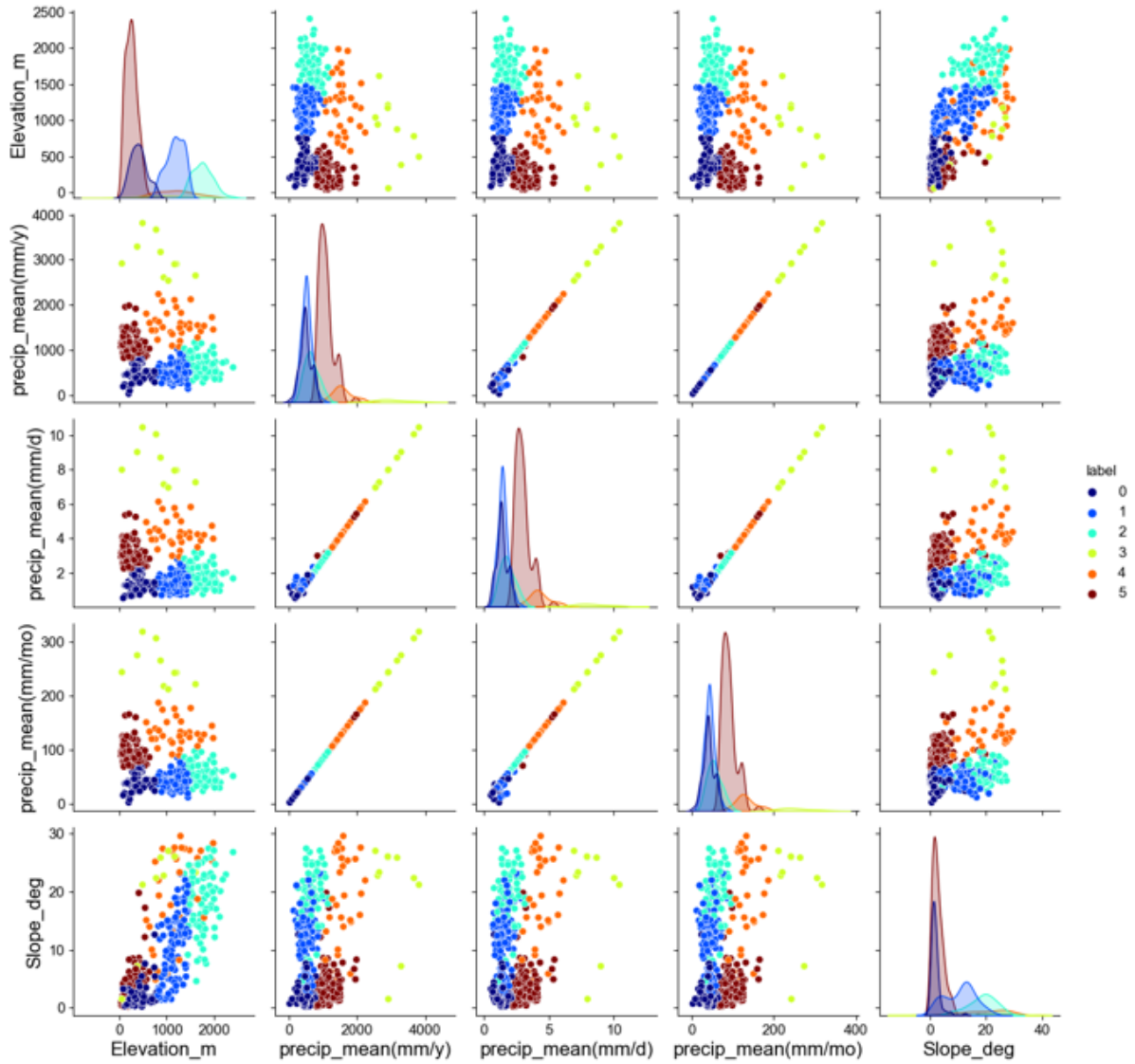


Figure E.21: Cross-comparison of the five most important clustering features, including elevation, annual precipitation, daily precipitation, monthly precipitation, and slope, according to an RF trained to predict the K-Means clustering results. The diagonal contains smoothed histograms for each feature. Points are coloured by cluster.

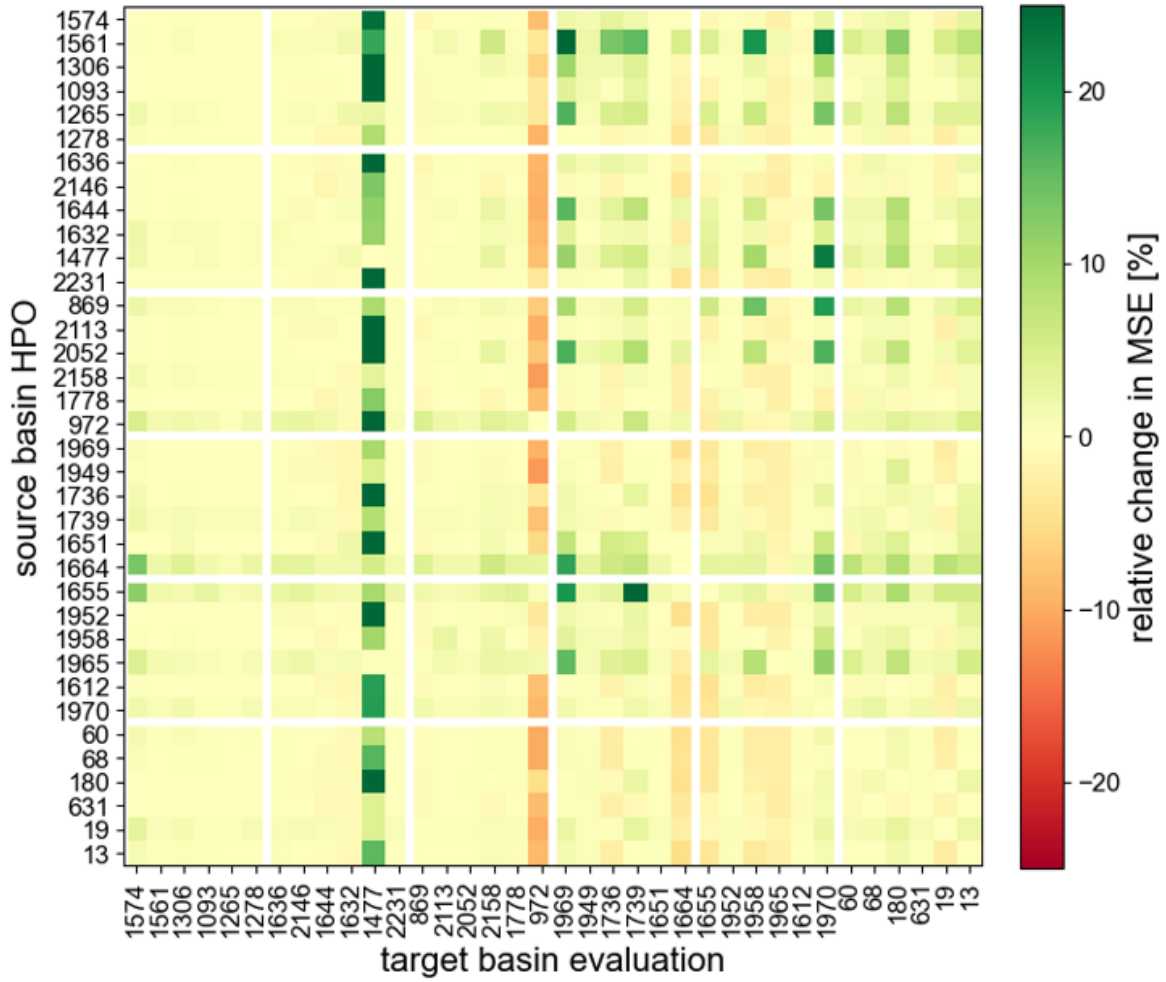


Figure E.22: Relative change in model performance (MSE) between target basins (x-axis) with non-optimum hyperparameters, and with hyperparameters donated from each source basin (y-axis). Basins are grouped based on cluster and groups are divided by white margins. Red to yellow to green indicates that the hyperparameter transfer produced worse, no impact, and better performance, respectively.

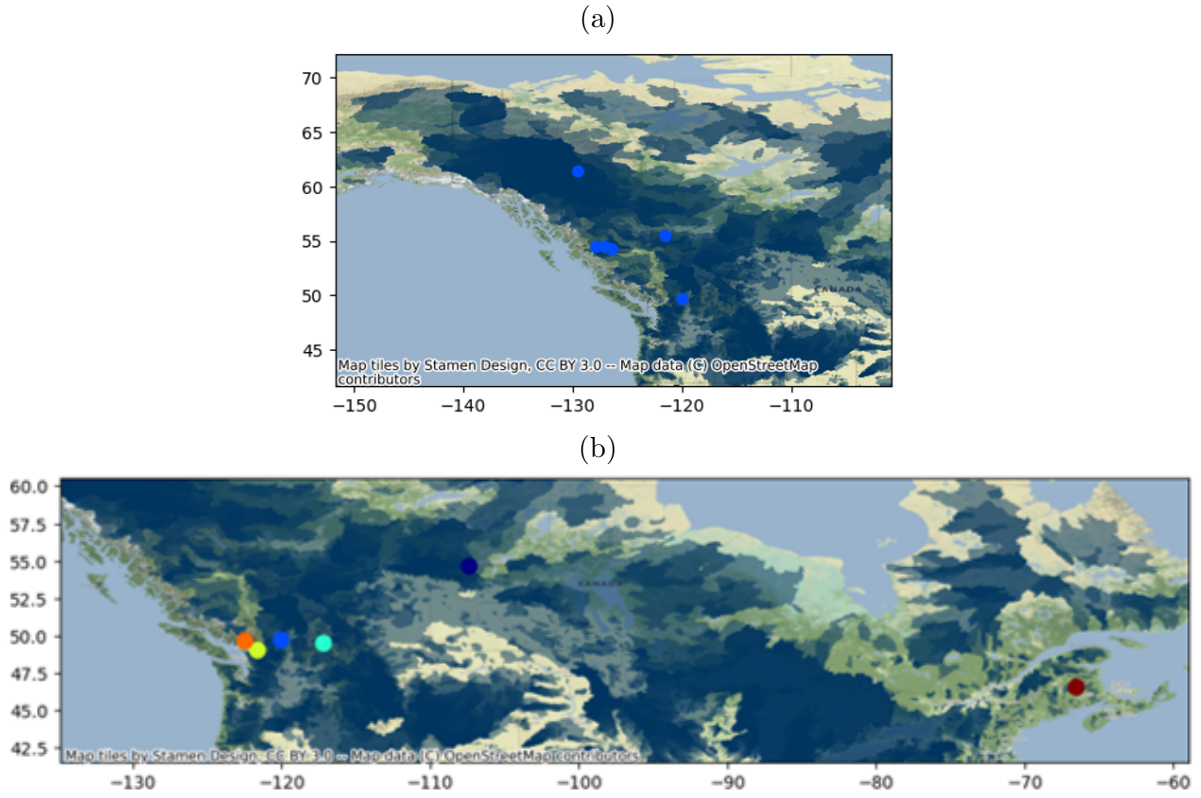


Figure E.23: Six basins with homogeneous (a) and heterogeneous (b) hydrological characteristics, per the K-Means clustering results. Basin footprints are underlain in transparent blue.

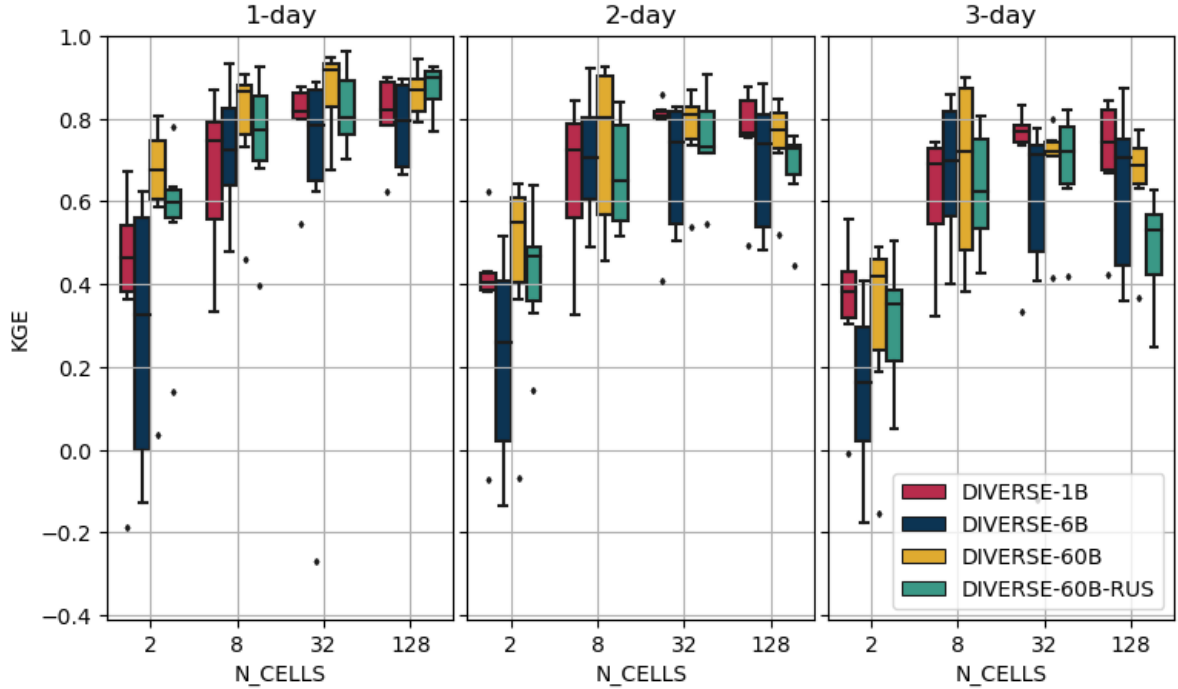


Figure E.24: The performance on six 'diverse' basins of models trained individually to each basin (red), a model trained to all six basins (blue), a model trained to 60 basins (6 from each cluster, including the test basins; yellow), and a model trained to the same 60 basins with RUS training data (green).

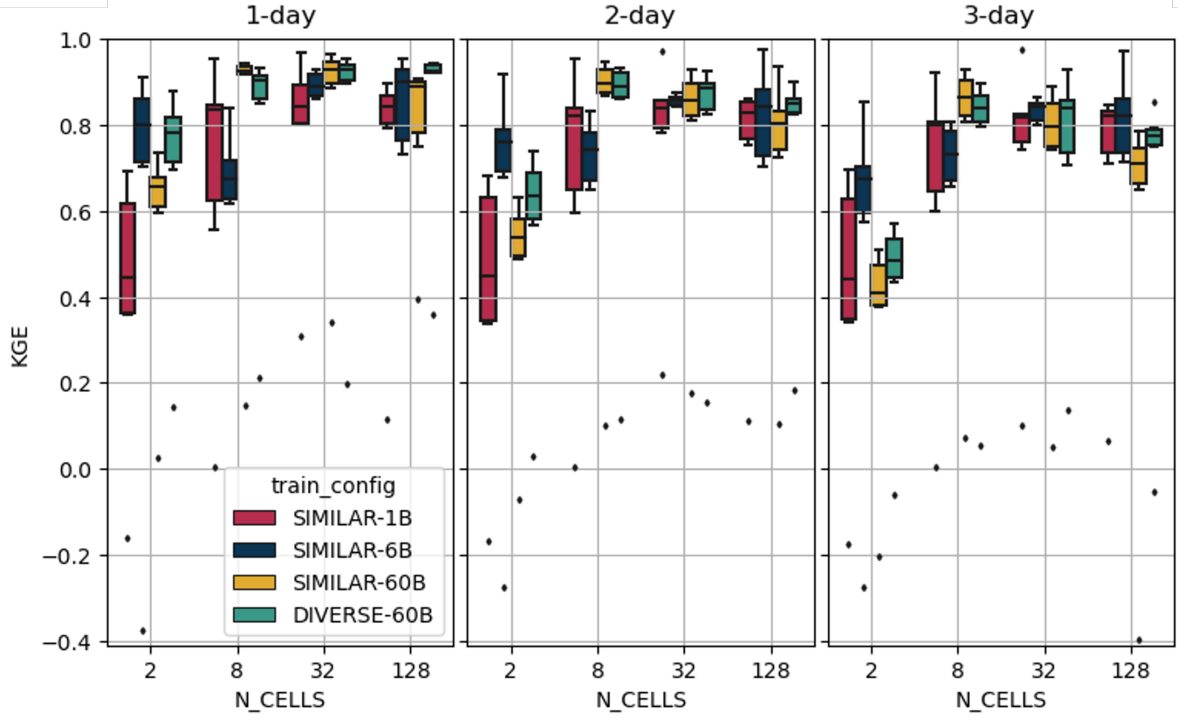


Figure E.25: The performance on six 'similar' basins of models trained individually to each basin (red), a model trained to all six basins (blue), a model trained to 60 similar basins (60 from the same cluster, including the test basins; yellow), and a model trained to 60 diverse basins (6 from each cluster, including the test basins; green).

E.5.3.2 Hyperparameter transfer across regionally trained models

The performance of several model configurations in terms of KGE, evaluated on six hydrologically diverse basins, is shown in Fig. E.24. Irrespective of training data, model performance tends to increase from 2 to 128 hidden cells, plateauing at 32 cells. When a single model is trained to the 6 diverse basins, it performs poorer than six models trained to the basins individually. However, when the training dataset is increased from 6 to 60 diverse basins (inclusive of the 6 basins being evaluated; 10 basins from each cluster), the model performance improves considerably, typically beyond that of the locally trained models. To determine whether this is due to the regularisation effect of a large training size (i.e., the model is less overfitted), the set of 60 basins is randomly undersampled such that it contains the same sample size as the 6 basin configurations. Despite containing only 10% of the data from the test basins compared to the other configurations, model trained to undersampled data exhibits competitive performance. This result highlights the value of hydrologically diverse datasets. These benefits are shown to be insensitive to different numbers of hidden cells, and training dataset sizes.

The previous experiment is replicated in Fig. E.25, but models are evaluated on a set of 6 hydrologically similar basins. Compared to the case with dissimilar basins, the model trained to six similar basins performs much better compared to locally trained models. These models are then compared against models trained with 60 similar, and 60 diverse basins (inclusive of the 6 basins being evaluated). Both larger training datasets tend to produce improvements in performance, regardless of whether the additional basins are hydrologically similar or not.

While these experiments highlight the importance of hydrological diversity within the dataset is essential for training models, too much diversity with respect to the size of the training dataset can result in poor performance. In cases with very few basins, it is much better to include similar examples. As the number of basins grow, so does the benefit of including hydrologically diverse basins.

E.5.4 Conclusion

Acceptable HP ranges for locally trained models are quite broad HP performance (hidden cells) was found to be consistent across an increasing number of training basins Hydrological diversity typically beneficial – but models trained to small, diverse basin sets can struggle to capture the variable hydrological processes included. Future work on this topic should include basins trained to larger basin subsets and varying degrees of hydrological diversity.

E.6 A large sample study of the effects of upstream hydrometeorological input features for LSTM-based daily flow forecasting in Canadian catchments

E.6.1 Introduction

Flooding poses an increasing threat to safety and infrastructure in Canada. Damage can be mitigated through flood warnings, which are driven by flow forecasting models. Regional training has emerged as an effective way to train flow forecasting models [Kratzert et al., 2019a, Nevo et al., 2022]. Existing work in long short-term memory (LSTM)-based flow forecasting use meteorological input features to predict flow, whereas autoregressive LSTMs have received comparatively less attention [Nearing et al., 2021]. While the inclusion of real-time observations has the potential to improve prediction accuracy, it also introduces new learning challenges. The primary objective of this work is to advance the efficacy of deep-learning flow forecasting models in the context of flood early warning systems.

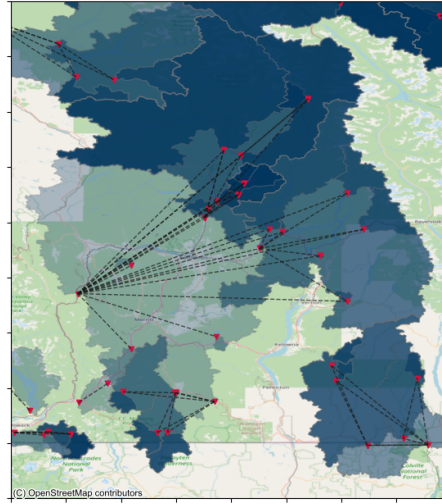


Figure E.26: Example of hydrological network extracted from the nested basin boundaries.

E.6.2 Methods

The HYSETS database was used as the basis for this work [Arsenault et al., 2020]. Of the 14000+ catchments in the HYSETS database, approximately 400 Canadian basins were found to have enough data for training and testing. A subset of 50 basins that each have at least one upstream basin are selected for this study.

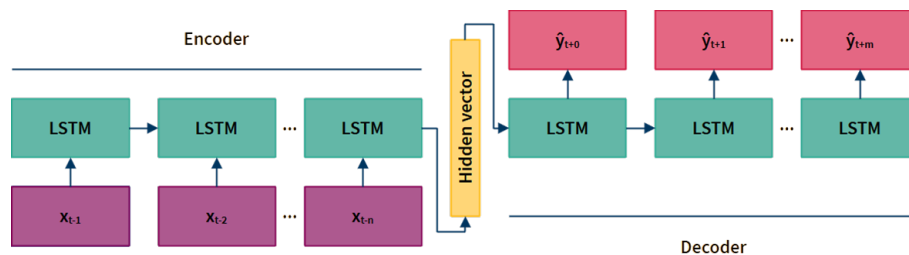


Figure E.27: Generic S2S LSTM diagram illustrating a forecast of timesteps $t+0$ to $t+m$ based on a lookback period of $t-1$ to $t-n$.

Sequence to sequence (S2S) LSTMs were used to predict discharge at lead times of 1 to 3 days. A heuristic hyperparameter search found optimum parameters including a hidden layer size of 128 cells, dropout rate of 0.4, batch size of 32, and a learning rate decreasing from 10-3 to 10-4 over 120 epochs.

Discharge at the target gauge, basin-averaged meteorological data (temperature, precipitation, SWE), discharge, precipitation, and SWE of upstream gauges, static basin data, and one-hot encoded basin labels. The number of upstream features corresponds to the number of maximum number of upstream features in the set of basins. Static features include a combined total of 37 terrain descriptors (e.g., fraction forest, fraction urban, slope, elevation) and hydrometeorological statistics (e.g., high/low frequencies, seasonality, baseflow, annual/monthly means).

Continuous 12-year periods were used for training, validation, and testing. Models performance was evaluated using Nash-Sutcliffe Efficiency (NSE) and persistent NSE (PNSE) [Nevo et al., 2022]. The latter uses the last known discharge as a benchmark, rather than the mean observed value. At a daily resolution, PNSE will typically be a stricter criterion, as variance typically exceeds the MSE between the observed and naïve forecast at a daily resolution.

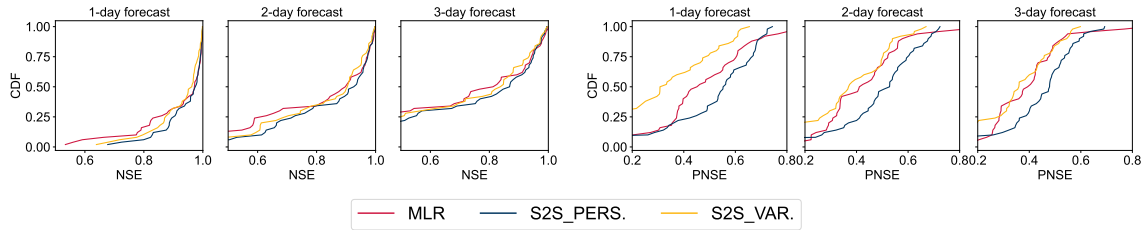


Figure E.28: Cumulative distribution functions for NSE and PNSE for single-basin MLR (red), persistence-based (blue), and variance-based (yellow) basin regularisation.

E.6.3 Results and discussion

We evaluate the persistent NSE for regularisation, to improve performance relative to the naive model, in order to counteract the tendency of ML models to produce naive forecasts. For sequential forecasts, this weighting scheme weights each lead time differently, accounting for the persistence within each basin. The effect of typical variance-based and proposed persistence-based regularisation is shown in Fig. E.29.

While the inclusion of upstream features improve the performance of the linear model, as shown in Fig. E.30, they have relatively less of an effect on the S2S models. This suggests that the S2S models are capable of extracting more information from the downstream gauges, thus receive less value from upstream basins. The inclusion of upstream produce improvements in a subset of larger basins and tend not to be detrimental to the other basins. The MLR_US model outperforms the S2S models in some cases, which could be addressed through fine-tuning.

Differences in performance between models with and without upstream data are shown spa-

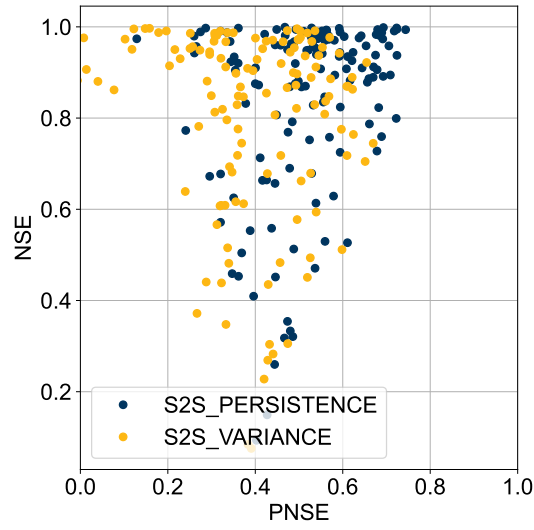


Figure E.29: Relationship between NSE and PNSE for persistence-based (blue) and variance-based regularisation (yellow).

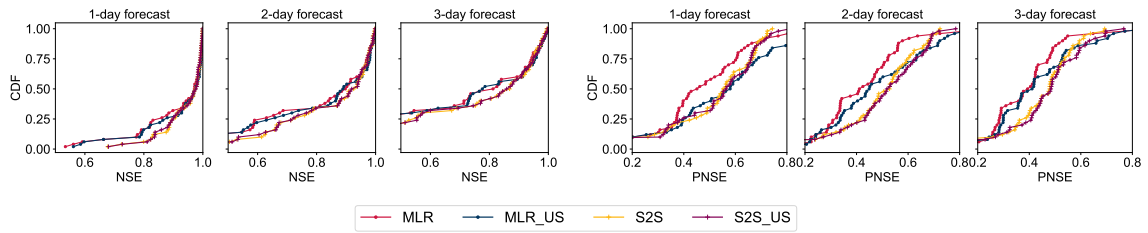


Figure E.30: Cumulative distribution functions for NSE and PNSE for single-basin MLR without (red) and with upstream data (blue), S2S without (yellow), and with upstream data (purple).

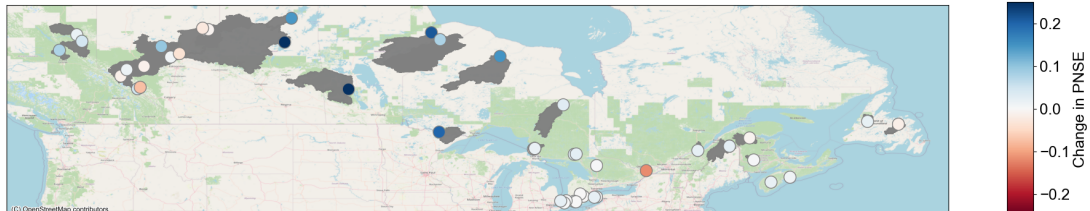


Figure E.31: Absolute change in 3-day PNSE, ranging from blue (improvement), to white (no change), to red (deterioration), between S2S models without and with upstream data.

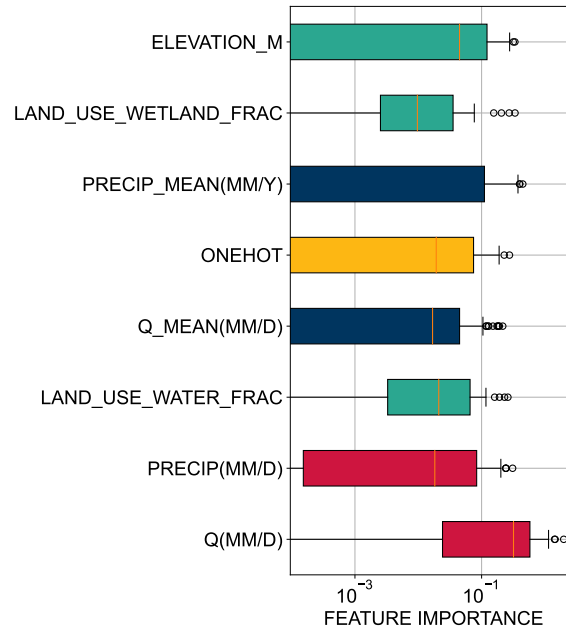


Figure E.32: Basin-wise permutation-based feature importance. Features colourised by time-series (red), physical attribute (green), hydrometeorological statistic (blue) and one-hot encoding (yellow).

tially in Fig. E.31. The largest improvements in PI are observed for higher lead-times in larger watersheds, as shown below. In contrast, smaller watersheds tend to exhibit anywhere from slight improvements to degradation in performance.

The results of permutation-based feature ablation are shown in Fig. E.32. Consistent with studies of locally trained models, dynamic discharge (Q) and precipitation (PRECIP) are the most important features. Elevation is the most important static feature, which is used by the model to distinguish between different basins. Upstream features are not included among these results as their importance varies widely between basins, and many basins only have a single upstream feature, meaning that the remaining upstream features are empty placeholders.

E.6.3.1 Conclusion

Persistence-based regularisation was evaluated and found to be very effective for improving average basin performance, especially large, autoregressive basins. Across the entire Canadian hydrometric network, there are relatively few basins with upstream gauges that contain sufficient overlapping data for effective training. Large basins tend to benefit the most from upstream data. In contrast, small basins are typically indifferent to the inclusion of upstream data, and in some cases exhibit a slight decrease in performance.

A limitation of this study is the daily resolution of the publicly available Canadian hydrometric database. There is a need for increased large-sample datasets at a sub-daily resolution, which would allow for focused improvements to forecast accuracy in smaller watersheds. Future work will expand the model comparison to include simple ML models and fine-tuned

deep learning models. Additionally, we will develop modular approaches for including upstream and autoregressive features, increasing model resiliency against dropped input signals.