Marine and Coastal Fog: Forecasts and Evaluations

Zheqi Chen

A DISSERTATION SUBMITTED TO

THE FACULTY OF GRADUATE STUDIES

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

GRADUATE PROGRAM IN

EARTH AND SPACE SCIENCE AND ENGINERRING

YORK UNIVERSITY

TORONTO, ONTARIO

April 2025

Abstract

Accurate marine fog forecast is of importance for human activities in the coastal regions and over the ocean. However, it is still a challenge due to lack of observation and unsatisfactory model performance. This dissertation includes three studies on forecasting marine fog. The first study has summarized the performances of three different models during the FATIMA (Fog And Turbulence Interactions in the Marine Atmosphere) field campaign on Sable Island, Nova Scotia, including two WRF (Weather Research and Forecasting) models and a COAMPS (Coupled Ocean/Atmosphere Mesoscale Prediction System) model. It is found that the models can perform differently on fog prediction with similar errors in temperature, dew point, relative humidity, wind speed and wind direction. Additional tests show that the WRF model can be improved by adjusting the horizontal and vertical domains, and a spin-up time is necessary for forecasting fog.

The second study has compared the liquid water content and droplet concentration observations from the field campaign to WRF variables in two periods. It is found that the WRF model produces liquid water contents up to 0.6 g m^{-3} while the observation has up to 0.3 g m^{-3}, partly due to the updraft speed in the microphysics scheme being too high and causing a high activation rate towards droplets. The surface temperature of the island, not included in the GFS (Global Forecast System) data but assigned by WPS (WRF Preprocessing System) is also too high,

causing the fog to incorrectly dissipate in daytime.

The last study has used XGBoost, a machine learning model to post-process WRF output. XGBoost is trained with the ERA5 (ECMWF Reanalysis v5) data of St. John's, Newfoundland and Labrador, and Yarmouth, Nova Scotia. XGBoost can predict fog or clear with given 2 m temperature and relative humidity, 10 m U and V winds, land surface pressure, their corresponding values one hour ago, hour, day and month. Tests using forecasts of the features from WRF in 2024 found that XGBoost improved the recall by up to 0.07 without decreasing the precision, compared to using WRF only. This shows the potential to combine the strength of both models.

Acknowledgements

First of all, I would like to thank my supervisors, Peter Taylor and Yongsheng Chen, for giving me the chance of participate in this research as my PhD study. They have been guiding me throughout my entire graduate program. I would also like to thank my committee members, George Isaac and John Liu, and Piyush Teeloku, who is not only a student in our research group, but also a friend of mine. Financial support for my Research Assistantship has been provided by grants from NSERC and ECCC to my supervisors, and, in my final year, from York University support of Prof. Taylor's research. Lastly, I cannot be more grateful to my parents for their support of everything.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Motivation

Fog can seriously affect transportation and public safety. Flights can be delayed or cancelled, and traffic accidents can occur as a result of fog. Inaccurate fog forecasts cost U.S. aviation up to $875 million in additional operational costs annually (Riordan and Hansen, 2002). Therefore, accurate and in-time forecasts of fog are essential to the public safety, economy and society. In coastal cities, marine fogs can last for days, which will have more impact than radiation fogs that mostly will dissipate in daytime. However, forecasts of marine fog still suffer from the lack of observations over the open sea, parameterization uncertainties, and coarse vertical resolution in numerical models for representing boundary layer processes (Yang et al., 2018). The goal of this dissertation is trying to improve fog forecasting, by understanding the problems existing in the current numerical models in Chapter 2 and Chapter 3, and by introducing machine learning to fog forecasting in Chapter 4. A table of abbreviations is provided in Appendix A.

This dissertation is an extension of my MSc thesis, Chen (2021), which studied the marine fog on Sable Island, NS in summer 2018 with the Weather Research and Forecasting (WRF) model

(Skamarock et al., 2019). It compared the variables between WRF and NAV CANADA (NAVCAN) reports at the airport. It was found that the visibility calculated from WRF, using liquid water content and a fixed droplet number concentration of 100 cm^{-3}, was much lower than the reported values (Fig. 1.1), despite that the errors of the other quantities were small. The model used has a relatively low resolution of 10 km. The present dissertation will have better comparisons in more detail in Chapter 2 and 3.

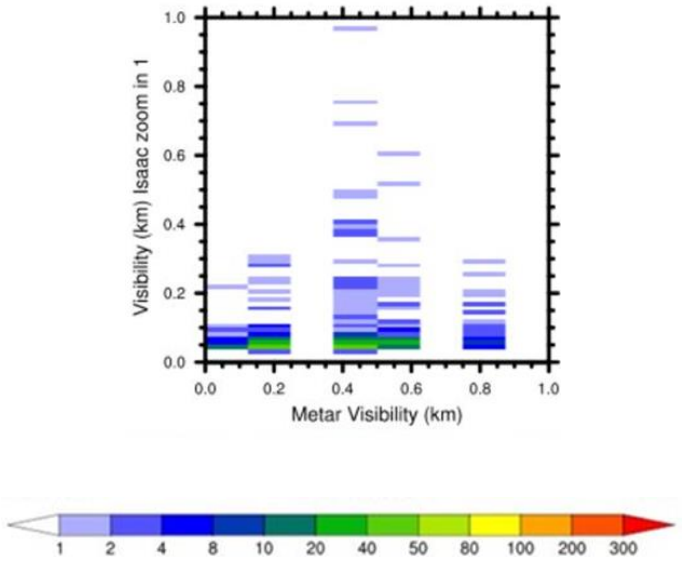


Figure 1.1 Correlation plot from Chen (2021) between reported (METAR) visibility and calculated visibility up to 1 km for June, July and August 2018. Calculated visibility uses the equation from Isaac et al. (2020), with the results from WRF. The colour bar is the number of hours. For reported visibility at 0.2, 0.4 or 0.6 km, calculated visibility is mostly lower.

Chen (2021) also discussed the possible cause of this difference between reported and calculated visibility, claiming that the mechanism of water droplets deposition due to turbulence was missing in the model. Chen (2021) implemented the change from Taylor et al. (2021) and

successfully decreased the amount of liquid water near the surface. Since this is a hypothetical approach, the present dissertation will not continue this discussion. However, the present dissertation will have further discussions on the physical parameterizations and the amount of liquid water in WRF in Chapter 3.

As the last experiment, Chen (2021) used a domain with a high resolution of 400 m in Fig. 1.2. It had a few discussions on the impact of the land on model outputs. The model was able to catch the diurnal cycle of temperature of the land, but it missed more fog due to higher temperatures dissipating the fog. This dissertation will use a resolution of 1 km, which is more practical, but the land of the island can still be recognized in the model. The present dissertation will also discuss the temperature issue of the land.

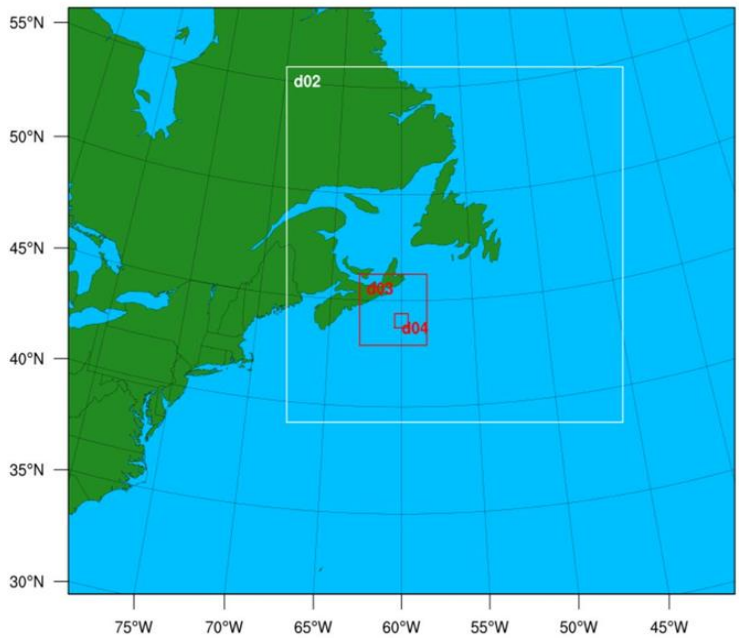


Figure 1.2 Domains used in Chen (2021). The location of Sable Island is recognized as a water surface in domain 1 and 2. The land of the island is recognized in domains 3 and 4.

1.2 Fog

Fog and cloud both result from the condensation of water vapour. However, fog forms within the planetary boundary layer, which makes it more difficult to forecast than cloud. In meteorological practice, and from the World Meteorological Organization (WMO) definition, fog is observed when the visibility is reduced to below 1 km due to small water droplets in the air. Fog is specifically associated with near-surface layers containing water droplets that have an average diameter of 10-20 μm, and a concentration up to 1 g m^{-3} (de La Fuente et al., 2007).

Most fogs that occur over inland areas are due to the cooling of the surface by emitting longwave radiation at night, and they will usually dissipate in daytime when solar radiation heats the land surface. They are called radiation fog (Fig. 1.3). A clear sky condition is favorable because the clouds will absorb the longwave radiation and emit some of it back to the surface, decreasing the rate of cooling. Wind speed should be low since it will introduce vertical mixing which reduces the cooling at the very bottom (Environment and Climate Change Canada (ECCC), 2013).

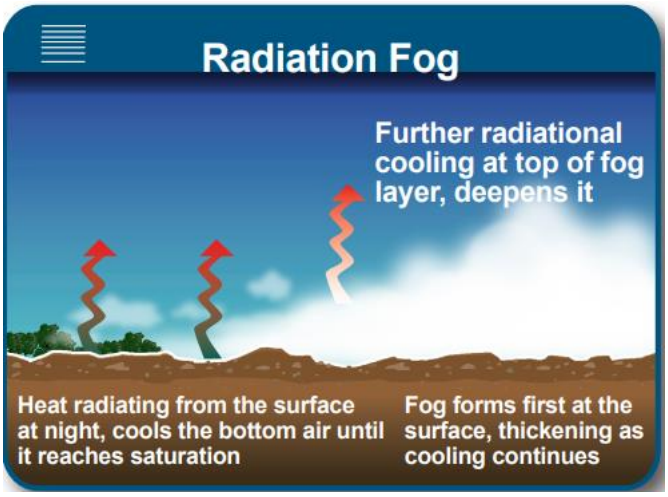


Figure 1.3 Mechanism of a radiation fog (from ECCC, 2013).

Marine fog that occurs over the ocean and coastal regions can last for days and usually involves the advection of warm, moist air originating from a warmer water surface area moving over a colder water surface area, which is called an advection fog (Fig. 1.4). The difference between sea air temperature and sea surface temperature (SST) during an advection fog is typically between 0.5 ˚C to 3 ˚C (Yang et al., 2018). Compared to radiation fog, winds should be stronger, approximate 8 to 17 knots. Lighter winds produce less dense fog while stronger winds can create low stratus clouds (ECCC, 2013). However, turbulence within the fog is still very low (Telford and Chai, 1993).

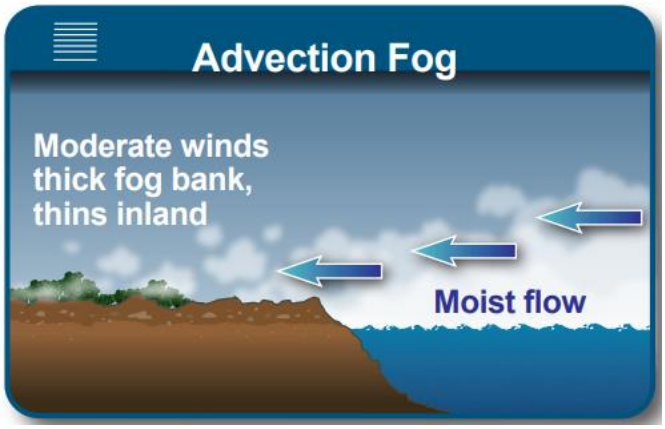


Figure 1.4 Mechanism of an advection fog (from ECCC, 2013).

For numerical weather prediction (NWP) models, the difficulty of fog forecasting comes from the fact that numerous feedbacks in fog can occur, all of which are sensitive to the spatial resolution, initial conditions, and physical parameterizations within the model. Among the parameterizations, the land surface, microphysics, radiation, and boundary layer, each plays a particularly important role during the formation, maintenance, and eventual demise of fog

(Wilson and Fovell, 2018). Gultepe et al. (2006) stated that NWP models need accurate liquid water content and droplet number concentration to provide accurate visibility forecasts.

Another challenge for marine fog comes from a lack of observations. As mentioned above, marine fog occurs due to the advection of marine air. Therefore, accurate observation of the original area of the air mass is also important. Popular global models such as the Global Forecast System (GFS) and the European Centre for Medium-range Weather Forecasts (ECMWF) can provide input data for NWP models for a large area, but they usually have a resolution of 0.5˚ to 0.25˚, while for a pointwise forecast the required resolution should be higher. The High-Resolution Rapid Refresh (HRRR) model from the National Oceanic & Atmospheric Administration (NOAA) is a regional model with a resolution of 3 km, with data assimilation, which has the potential to improve the results. However, it is not popular among the users of WRF, partly because it is regional, and WRF does not have official support to it. Blaylock (2017) managed to initialize WRF with HRRR boundary conditions. A separate high-resolution SST input is also an option, but not many archives are available.

Koračin and Dorman (2017) produced the first world-wide marine fog occurrence analysis based on the International Comprehensive Ocean-Atmosphere Data Set (ICOADS) for 1950 – 2007. There are 12 maxima with high marine fog occurrences. The maximum fog occurrence centre in the ocean is over the Kuril Islands in the NW Pacific Ocean. The Grand Banks area in NW Atlantic in June-July-August is the second highest marine fog maximum occurrence, reaching 45.0%, defined as the percentage of the fog reports among all reports (Figure 1.5). The FATIMA

(Fog And Turbulence Interactions in the Marine Atmosphere) 2022 campaign in Chapter 2 and 3 had field campaigns on Sable Island as well as ship measurements over the Grand Banks.

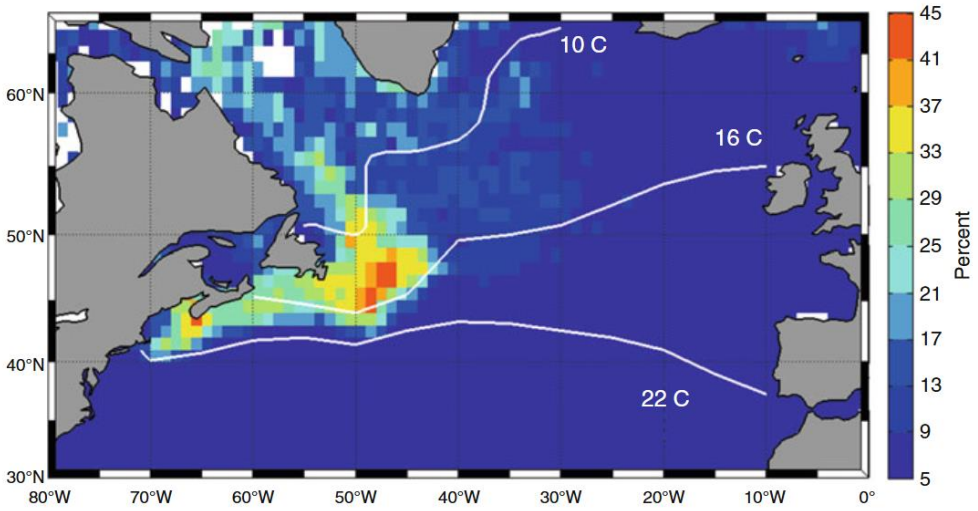


Figure 1.5 Marine fog occurrence of NW Atlantic in June-July-August at 1˚ resolution. White colour means no report. The lines are SST isotherms (from Koračin and Dorman, 2017).

1.3 WRF

The WRF model is a mesoscale NWP system designed for both atmospheric research and operational forecasts (Skamarock et al., 2019). The model serves a wide range of meteorological applications across scales from tens of meters to thousands of kilometers. The real case simulation (em_real) in WRF has been used for this study. More details of the physics in WRF will be introduced and discussed in Chapter 3.

There are a variety of methods to read the files and analyze the WRF output. The present dissertation uses the NCAR Command Language (NCL) which is an open source interpreted

language designed specifically for scientific data analysis and visualization. It is a powerful language for reading, writing, manipulating and visualizing scientific data. It also provides specific functions for WRF data analysis, such as calculating variables in any model levels, including temperature or relative humidity that are not directly given in the WRF model output (wrf_tk, wrf_rh), finding the nearest longitude latitude location for a specified gird point (wrf_user_ij_to_ll), and interpolating data at a desired height.

1.4 Machine Learning

Currently, there are increasing interests in using machine learning (ML) in weather forecasting. Unlike the traditional NWP models using equations to derive the future state from the current state, ML allows computers to learn from data without being explicitly programmed (Hsieh 2023). The same model, e.g., neural network, random forest or deep learning, can be used for totally different purposes from scam mail detection to weather forecasting, because ML models will try to find patterns with the given training data to make predictions without knowing the backgrounds. Compared to traditional NWP models they are claimed to be faster and more accurate. However, there are also doubts about machine learning, saying that the models have low interpretability.

Chapter 4 of the present dissertation will discuss more on this topic and will have one study that uses an ML model. Python programming language, which is popular in various fields including machine learning, will be used with external libraries such as numpy for numerical computing, pandas for data structure and sklearn for many functions related to ML. This work is partly a

collaboration with MSc student, Piyush Teeloku. The planning and data collection has been conducted collaboratively. However, we used different ML models for this approach and thus the results are different and independent. The analysis and interpretation in Chapter 4 are my original work.

1.5 Literature Review

Fog forecasting has always been an issue. On the traditional NWP side, Lin et al. (2017) have used the WRF model to predict the advection fog over Shanghai Pudong International Airport (ZSPD) with the highest resolution of 4 km. The lowest observed visibility from the METAR reports was 2 km. They have tried 8 different algorithms to calculate visibility, but all algorithms showed the visibility of lower than 1 km at the corresponding time. They have also tested different forecast lead time, different microphysics schemes and different PBL schemes. All the model results showed lower visibility than the observation.

Weston et al. (2022) have also used the WRF model to forecast marine fog in Namibia, Africa. The smallest domain had a resolution of 3 km. They have compared the model results with the observations from a field campaign conducted from August 23rd to September 12th, 2017, at the location on the coastline called Henties Bay. The measurements were at 20 m above sea level. During the period, the maximum observed LWC was mostly 0.2 g m^{-3} and sometimes their model could match the observations. However, their model had a few false alarms where the values could reach 0.4 g m^{-3}. The results of cloud droplet number concentration were close to observed values around 25 cm^{-3}.

There are different approaches on the ML model side. Gultepe et al. (2024) have used 3 different ML models to nowcast the visibility over the Grand Banks. They have used the measurements collected during the FATIMA Grand Banks campaign, which was conducted along with the Sable Island campaign, where the R/V Atlantic Condor was sailed in the NW Atlantic. They have used the temperature, dew point depression, relative humidity, wind speed, wind direction, pressure, as well as the standard deviations of temperature, relative humidity and pressure as the features, and visibility as the target. They have tried the lead times of 30 min and 60 min. In the range below 1 km, their model results were consistent over time between 0.2 to 0.4 km, which was not able to capture the sudden increase of observed visibility up to 0.8 km. In the range below 10 km where the observed values varied between 0.2 km to 10 km, although the ML models were able to predict the trend of change, the model values did not reach the extreme values as well. Also, the increase and decrease of visibility in the model happened later than the observation.

Vorndran et al. (2022) have used XGBoost to nowcast the radiation fog in Linden-Leihgestern, Germany which is characterized by one of the highest fog occurrences in Germany. They have used a binary classification between fog (visibility <= 1 km) and no fog (visibility > 1 km). They have used 9 years of data, with the features including 10 m pressure, 2 m and 10 m temperature, temperature inversion (2 m temperature minus 10 m temperature), temperature trend (current temperature minus temperature 3 hours ago), 2 m and 10 m relative humidity (RH), 2 m net radiation, 2 m visibility, 10 m wind speed, 10 m vertical turbulence intensity, dew point depression, solar zenith angle and current fog state. These features were used to nowcast the fog state with lead times of 60, 120, 180 and 240 min. They achieved the F1 score of 0.8 with a lead time of 60 min, but it decreased with a longer lead time. With a lead time of 240 min, they only

achieved a F1 score of 0.5.

There are also approaches to use ML to post-process the NWP outputs. Kim et al. (2022) has trained the XGBoost model with results from the Local Data Assimilation and Prediction System (LDAPS) model of the Korea Meteorological Administration. They used XGBoost regression to predict visibility as a numerical value. The XGBoost model showed better performance than the LDAPS model against observation. However, their visibility was in the range between 4 and 20 km. There can be problems for fog with extreme values lower than 1 km.

Thomasson (2023) has trained 3 ML models with observation data in Denmark and used them to post-process the results from the NWP model of the Danish Meteorological Institute, similar to the approach in the present dissertation. The NWP model had a consistent F1 score of 0.2 with different lead times, and although the ML models performed better with a higher F1 score (0.4 to 0.5) with a short lead time, they performed similar to the NWP model with a lead time of 6 hour or longer. This is probably due to that the visibility was divided into 8 different classes, with 5 classes lower than 1.5 km. The models were not able to distinguish the classes with small differences.

In summary, different approaches of fog forecasting have different problems. The present dissertation will discuss the problems of the WRF model, and propose an approach to post-process WRF outputs with machine learning.

Chapter 2

The FATIMA 2022 Campaign

2.1 Introduction

Before going into the fog forecast details, this chapter provides overviews of the FATIMA Sable Island campaign, and the models involved. It summarizes the model predictions of the models for the period of the campaign. The detailed comparisons and interpretations are my independent work.

2.1.1 FATIMA

FATIMA is a five-year project seeking leaps in the fundamental understanding of marine-fog life cycle via a cross-disciplinary approach. A field study (Fernando et al., 2025) was conducted in July 2022, with measurements on Sable Island. Another campaign was conducted at the same time, with ship measurements taken from a ship on the water over the Newfoundland Grand Banks. There were daily weather briefings as guides to the crews on-site for preparation of measurements, one part of which involved 1–2-day forecasts using WRF and the Coupled Ocean/Atmosphere Mesoscale Prediction System (COAMPS, Hodur et al., 2015) models. These were generally successful in forecasting weather conditions and, in particular, the likelihood of

fog and reduced visibility.

2.1.2 Sable Island

Sable Island, meaning “island of sand”, is a 30 km long and narrow (mostly < 1 km wide), crescent-shape sand bar in the Atlantic Ocean, about 150 km offshore from Nova Scotia, Canada. It is now a National Park and therefore, permissions from Parks Canada must be granted prior to any visit. In the past it was the site of many shipwrecks and was known as the graveyard of the Atlantic, due to the frequent fogs and sudden strong storms in the area. According to the ECCC Climate Normals, averaged over 30 years (1971-2000), there have been more than 200 hours with visibility ≤ 1 km per year in June and July, which is due to the contrasting effects of the cold Labrador Current and the warm Gulf Stream. On average there are 127 days out of the year that have at least 1 hour of fog.

ECCC operated an upper air station there for many years until August 2019, where radiosondes were carried aloft by hydrogen-filled weather balloons to altitudes beyond 40 km. Taylor et al. (1993) used the island as an offshore platform to study the surface features of offshore frontal passages in winter storms. It is an ideal location from which to study summer marine fog situations, since its long and narrow shape of land surface will have minimal impact on fogs. Therefore, the FATIMA project chose the island as part of its North Atlantic campaign. Since the ECCC weather station does not provide visibility reports, reports at the Sable Island airport provided by NAVCAN will be used for comparisons. Fig 2.1 shows a photo of the weather station on Sable Island on a foggy morning.

Figure 2.1 A photo of the ECCC weather station on Sable Island at 07:20 local time, September 26, 2020 (from Cheng et al., 2021). Photo by Jason Surette, Parks Canada.

2.1.3 NWP Models

In the campaign, different models have been used to forecast or hindcast the fog on Sable Island or over the Grand Banks. Two WRF models with different settings presented here are hindcasts and one COAMPS model from the U.S. Navy is the archive of forecasts.

The COAMPS model operated by Dr. Gabersek from U.S. Naval Research Laboratory utilized a terrain-following, height-based vertical coordinate. The experiment setup for Sable Island included three nested domains with decreasing horizontal grid spacing of 9, 3 and 1 km (Fig. 2.2a). In the vertical, 68 levels were used with 23 levels below 1 km, and the model top was at 28 km. During the campaign, COAMPS was initialized four times daily, at 00Z, 06Z, 12Z and 18Z, with each forecast lasting 48 hours. In this study, results from the forecasts initialized at 00Z and 12Z every day are used. GFS 0.5˚ x 0.5˚ forecasts were used as input data. Data assimilation with surface observations was also used. Since SST does not change significantly in a short-term forecast (Isaac et al. 2020), COAMPS was utilized by Dr. Gabersek in the atmosphere-only mode, uncoupled from the ocean to save computational resources.

The WRF model operated by me was version 4.2.1. It had 4 domains in total, with resolutions of 30 km, 10 km, 3.3 km and 1.1 km, respectively (Fig. 2.2b), zooming into the Sable Island area. During the campaign period, only the first two low-resolution domains were used for real time forecasts. After the campaign, simulations using all four domains were performed as hindcasts for research purposes. It will be called WRF (1.1 km) hereafter. To reduce computational cost, the vertical nested level option was used, with 37 vertical levels for domain 1 (30 km) and domain 2 (10 km), and 51 levels for domain 3 (3.3 km) and domain 4 (1.1 km). WRF uses eta level, which is the scaled pressure level with the surface being 1 and the top of the atmosphere being 0. The hindcasts of every day simulations used the GFS 0.25˚ x 0.25˚ final analysis (FNL) data as input. Domain 1 and domain 2 started at 12Z every day, 12 hours before the forecast day and ran for 36 hours, using the first 12-hour as a spin-up time. Domain 3 and domain 4 started at 00Z on the forecast day and ran for 24 hours.

The WRF model run by Dr. Dimitrova from University of Notre Dame was version 3.9.1 with four nested domains with resolutions of 12150 m, 4050m, 1350 m and 450 m (Fig. 2.2c). It implemented 50 pressure-based terrain-following vertical levels with a model top set at 50 hPa. The 0.25˚ x 0.25˚ GFS forecasts were used as input data. All domains started at 12Z and ran for 48 hours, with the first 12 hours of spin-up time discarded. This model will be called WRF (450 m) hereafter. This configuration was only used for hindcasts of the selected intensive operation periods (IOP), not forecasts. This is why the results only partially cover the field campaign.

Different GFS input data were used for different models since there was no plan for a comparison

during the campaign period. The results of the WRF (1.1 km) are hourly starting from July 5, and the results of the COAMPS model have a 3-hour interval starting from July 5. The results of the WRF (450 m) are hourly covering only 4 periods: 00Z July 7 to 12Z July 8, 00Z July 14 to 12Z July 15, 00Z July 17 to 00Z July 19, 00Z July 24 to 00Z July 27. The horizontal domains of each model are shown in Fig.2.2. Table 2.1 summarizes all 3 models.

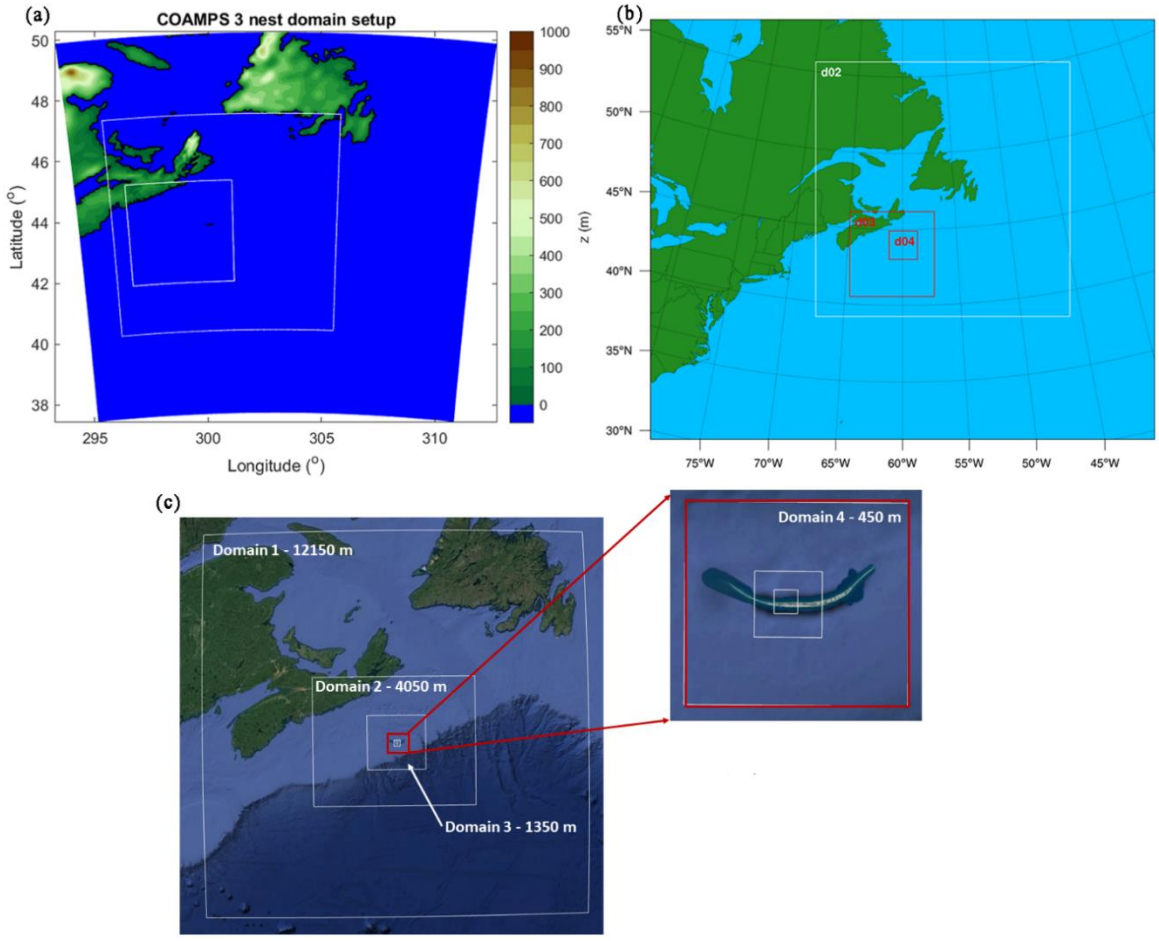


Figure 2.2 Domain settings used in (a) COAMPS, (b) WRF (1.1 km) and (c) WRF (450 m).

Table 2.1 Summary of the 3 models in Section 2.2.

	resolution (km)	vertical level	dataset	period
COAMPS	9, 3, 1	68	GFS 0.5° forecast	3-hourly July 5-31
WRF (1.1 km)	30, 10, 3.3, 1.1	51	GFS 0.25° FNL	hourly July 5-31
WRF (450 m)	12, 4, 1.3, 0.45	50	GFS 0.25° forecast	hourly selected period

2.2 Model Predictions

Fig.2.3 and Fig.2.4 show 2 m temperature and 2 m dew point of each model as well as the observation, respectively. With the high-resolution domains in all models, the models are able to predict the diurnal cycles of temperature on land, since the land surface will heat during the day and may have strong radiative cooling at night. In the research in Chen (2021) where a WRF model was run for Sable Island for the summer of 2018, with a coarse resolution of 10 km, the grid point of the island was considered as water, and the diurnal cycle of temperature was missing. The diurnal cycle was also missing in the York forecasts for daily briefings during the campaign in July 2022, where the two-domain set up was the same as Chen (2021). All simulations presented here have high enough resolutions to show the land of Sable Island, so the models are able to get the diurnal cycle of temperature. COAMPS has larger errors at the end of the month, and it has weaker diurnal cycles compared to the WRF models and the observations. The two WRF models can generally predict 2 m temperature well. The COAMPS model has dew point errors, up to 5°C near the end of the month. It is able to give good dew point predictions for some periods, but it is low for some other periods. WRF (1.1 km) has a better overall prediction,

but also has large, negative, nighttime errors for the period between July 10th and 14th. WRF (450 m) has generally similar results to WRF (1.1 km).

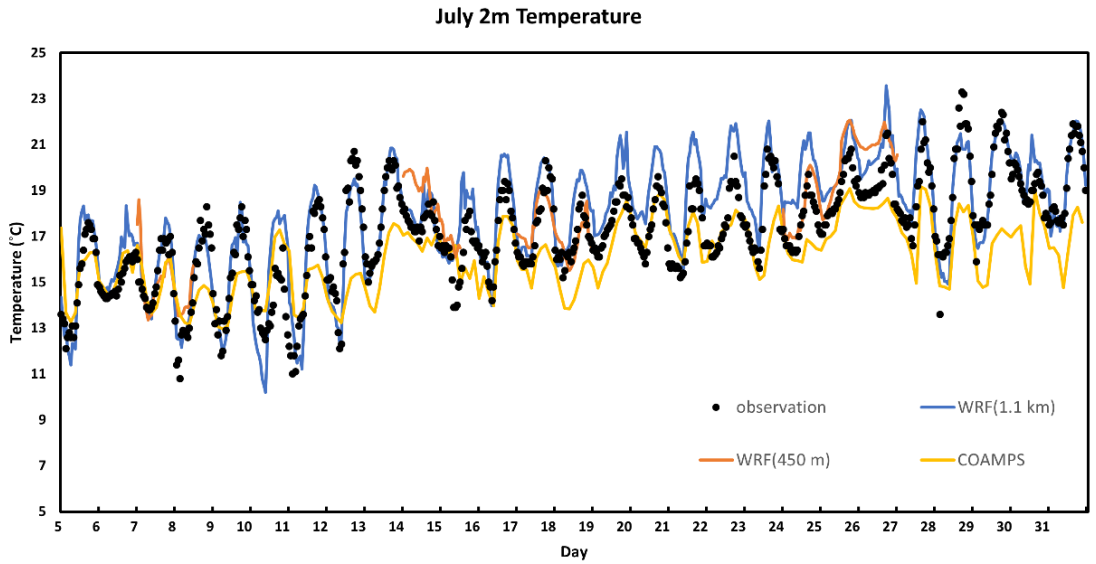


Figure 2.3 Hourly and 3-hourly 2 m temperature from the models compared with NAVCAN reports. Day labels along the Day axis are at 0000 UTC.

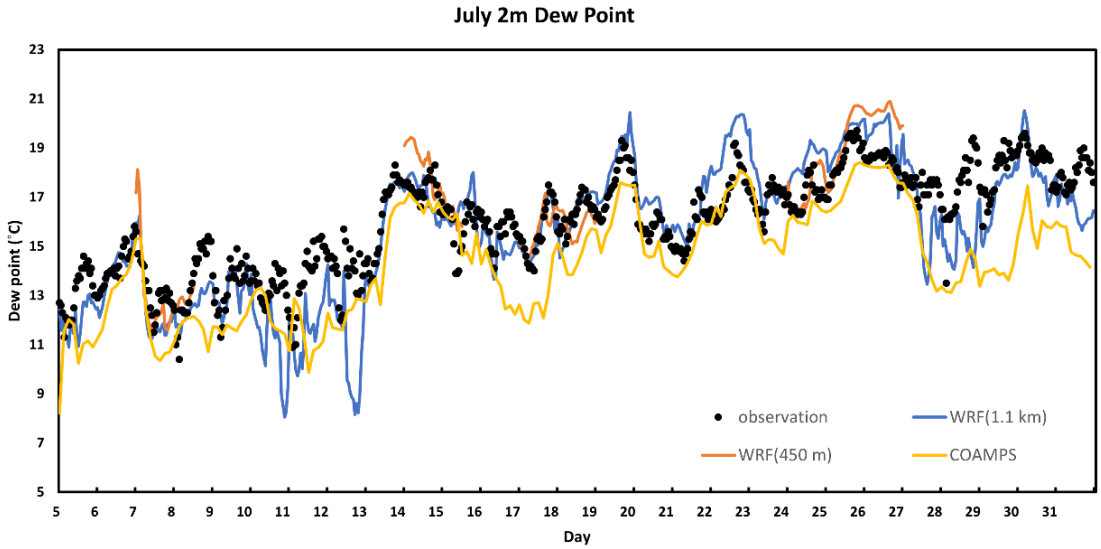


Figure 2.4 Hourly and 3-hourly 2 m dew point temperature from the models compared with NAVCAN reports. Day labels along the Day axis are at 0000 UTC.

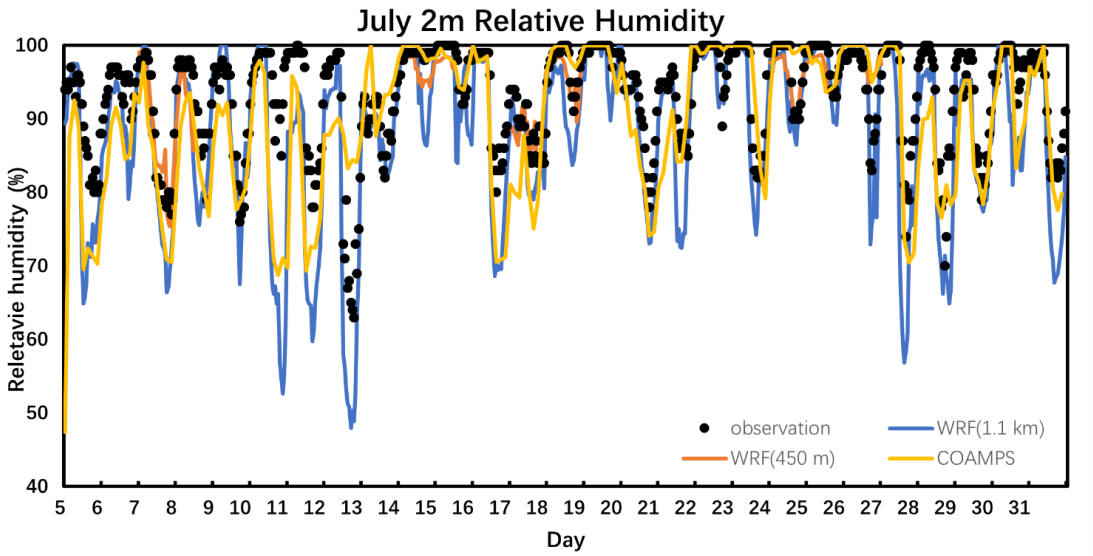


Figure 2.5 Hourly and 3-hourly 2 m relative humidity from the models compared with NAVCAN reports. Day labels along the Day axis are at 0000 UTC.

Fig.2.5 shows 2 m RH of each model as well as the observation. The RH for the WRF model predictions is calculated by the NCL function wrf_rh, using vapour pressure. For COAMPS, RH prediction is good at the end of the month, even though there are errors of temperature and dew point then. WRF (450 m) has good fits for its periods. The WRF (1.1 km) model can give good predictions when RH is relatively high, but when observed RH is lower than 80% the model sometimes gives lower values. In the periods when RH is lower in WRF (1.1 km) than the observation, model temperature does not seem to be higher, so the air mass should be dryer in the model.

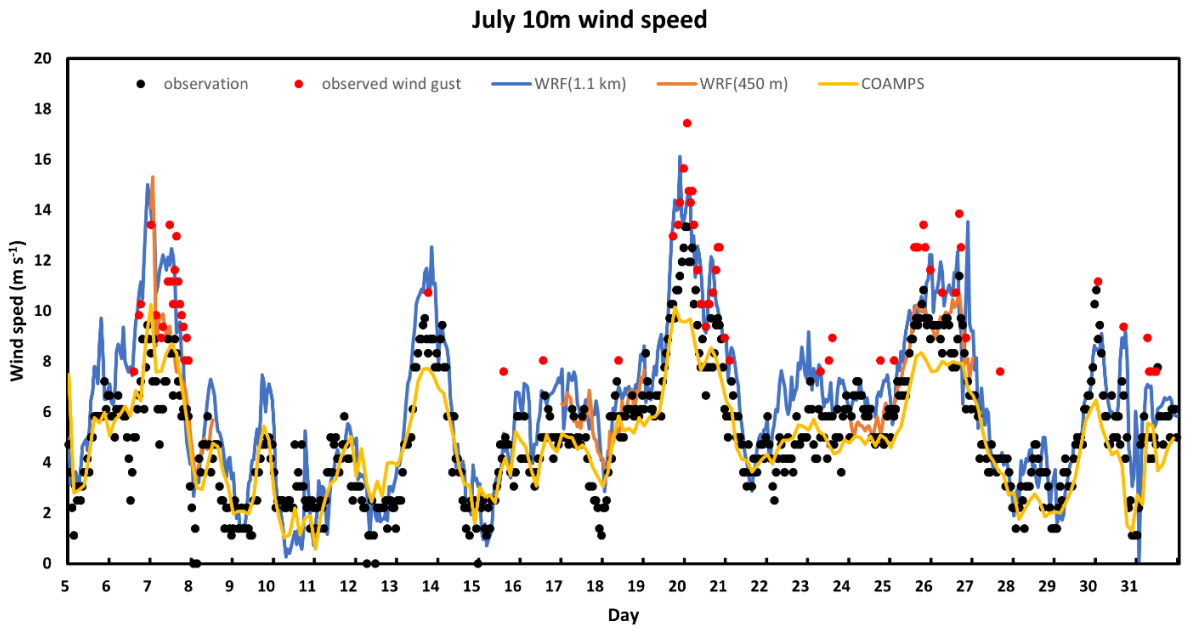


Figure 2.6 Hourly and 3-hourly 10 m wind speed from the models compared with NAVCAN reports. Day labels along the Day axis are at 0000 UTC.

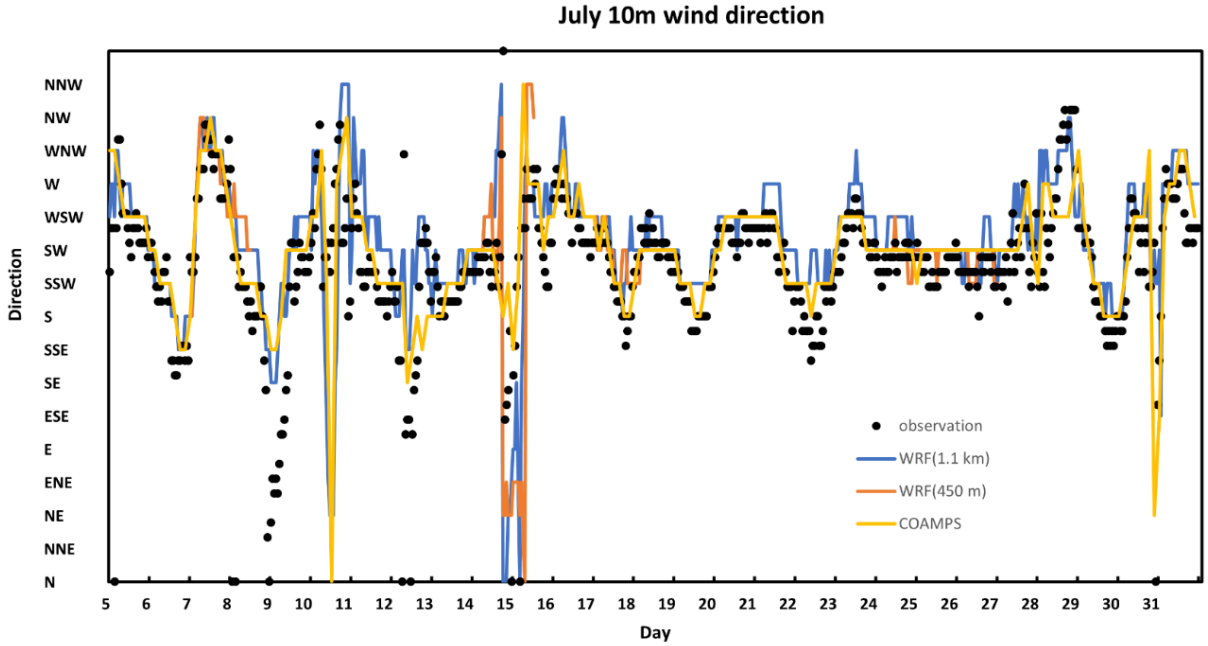


Figure 2.7 Hourly and 3-hourly 10 m wind direction from the models compared with NAVCAN reports. Day labels along the Day axis are at 0000 UTC.

Fig.2.6 and Fig. 2.7 show 10 m wind speed and 10 m wind direction of each model as well as the observation, respectively. COAMPS has good predictions of wind speed, and WRF (1.1km) seems to overestimate the wind speed. For the month of July, winds mainly came from the SW direction, which is predicted by the models. When the wind speed is low, e.g., on the 10th, 15th and 31st, models have larger bias in wind direction. The measurements of wind direction may be less accurate when the wind is weak.

The mean absolute errors relative to NAVCAN observations of 2 m temperature, 2 m dew points, 2 m relative humidity, 10 m wind speed and 10 m wind direction are calculated for different models and are shown in Table 2.2.

Table 2.2 Mean absolute errors of different quantities of the three simulations.

	T2 (˚C)	**TD2 (˚C)**	**RH2 (%)**	**WS10 (m/s)**	**WD10 (degree)**
WRF (1.1 km)	1.05	1.13	5.41	1.57	26.7
WRF (450 m)	1.11	0.98	2.48	1.12	21.3
COAMPS	1.45	1.71	5.66	0.99	21.1

WRF (1.1 km) has the lowest error of temperature but highest errors of wind speed and wind direction. WRF (450 m) has the best dew point and relative humidity prediction. COAMPS has the highest errors of temperature and dewpoint, but lowest errors of wind speed and wind direction.

2.3. Additional Tests with WRF

2.3.1 Methods

In this section, a WRF model with the same physics parameters of WRF (1.1 km) is used to simulate the month of July again, with adjusted domain settings. From Table 2.2, it is found that COAMPS has the best prediction of wind. For Sable Island in the summer, most of the winds come from the SW direction and the domain in COAMPS covers a large area of ocean in SW of the island with the finest resolution of 1 km. In addition, most fogs that occurred in this area are advection fogs due to the movement of air parcels. Therefore, it was decided to run the WRF model with domain settings similar to COAMPS, to see if this setting can improve the performance of WRF. With similar domain settings, it will be a more direct comparison between WRF and COAMPS.

The horizontal domains are adjusted to be similar to the domains of COAMPS, with the same resolution of 9 km, 3 km and 1 km, shown in Fig.2.8. The vertical domains are also adjusted, with 67 vertical levels at heights similar to those in COAMPS. Since WRF uses scaled pressure as the vertical axis (eta level) and COAMPS uses height, the levels are not exactly the same but as close as possible. The lowest levels at 1 m in COAMPS were not implemented since the WRF model cannot have levels lower than 2 m. Variables at 2 m are calculated using the similarity theory by relating the information at the lowest model level to the surface via a stability dependent, approximately log, profile of wind and other scalars (Skamarock et al. 2019). The mean heights of the lowest 5 levels in the new WRF setting are: 3.9 m, 12 m, 22 m, 34 m and 46 m in the previous setting, and 3.8 m, 9.5 m, 14 m, 20 m and 25 m in the new setting.

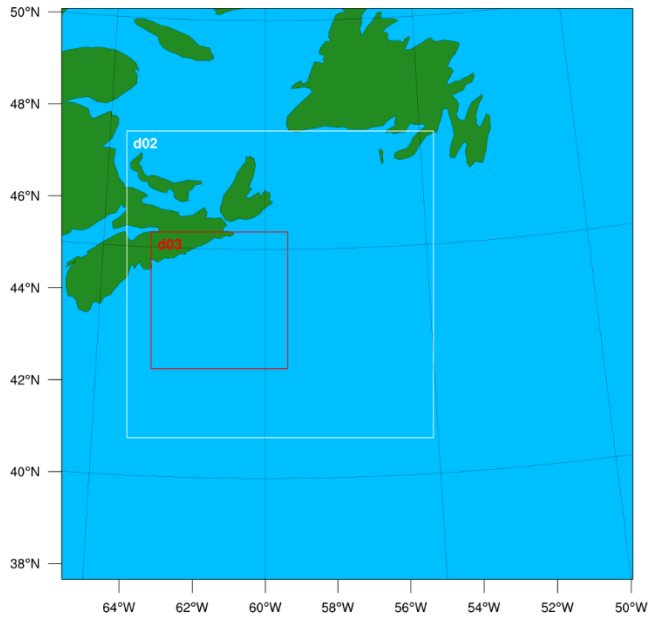


Figure 2.8 Domains used in the WRF simulation for additional tests, with resolutions of 9, 3 and 1 km, respectively.

Two sets of simulations have been done over the campaign period in July 2022. Both sets have adopted the above horizontal and vertical domain settings, but the simulation times are different. One set contains 12-hour simulations initialized at 00Z and 12Z every day with all results used. It is the same set up of COAMPS in this study. It is called WRF (0spin) hereafter, meaning that no spin-up time is used. Another set starts at 12 Z every day and runs for 36 hours, with the results of the first 12 hours discarded as spin-up. This is the same set up as WRF (1.1 km). It is called WRF (12spin) hereafter. To be consistent with COAMPS, here GFS 0.25˚ x 0.25˚ forecasts are used as input data, instead of the FNL for WRF (1.1 km). COAMPS used 0.5˚ x 0.5˚ forecasts, but there is only 0.25˚ x 0.25˚ available in the archive.

2.3.2 Results

Table 2.3 Mean absolute errors of different quantities of all five simulations.

	T2 (˚C)	**TD2 (˚C)**	**RH2 (%)**	**WS10 (m/s)**	**WD10 (degree)**
WRF (1.1 km)	1.05	1.13	5.41	1.57	26.7
WRF (450 m)	1.11	0.98	2.48	1.12	21.3
COAMPS	1.45	1.71	5.66	0.99	21.1
WRF (0spin)	1.04	1.08	5.11	0.92	22.8
WRF (12spin)	0.95	0.96	4.41	0.95	24.7

Table 2.3 adds the two new tests in this section to Table 2.2. Compared to WRF (1.1 km), both WRF (0spin) and WRF (12spin) have improvements on all quantities, especially on wind speed and wind direction, showing that the new domain setting covering more area SW of Sable Island can give better results. Compared to WRF (0spin), WRF (12spin) has better temperature, dew point and relative humidity predictions, but worse wind speed and wind direction.

Since mean absolute error can only show the errors with positive number, Fig. 2.9-2.11 show the histograms of the errors. Errors of T2, TD2 and WS10 are shown, of WRF (1.1 km), COAMPS and WRF (12spin). The errors of WD10 have higher uncertainties because the reported observations are in every 10 degrees, and therefore, the errors of WD10 are not shown. The histograms of COAMPS have lower frequencies because in this study COAMPS only has data every 3 hours. WRF (1.1 km) and WRF (12spin) have the same amount of data.

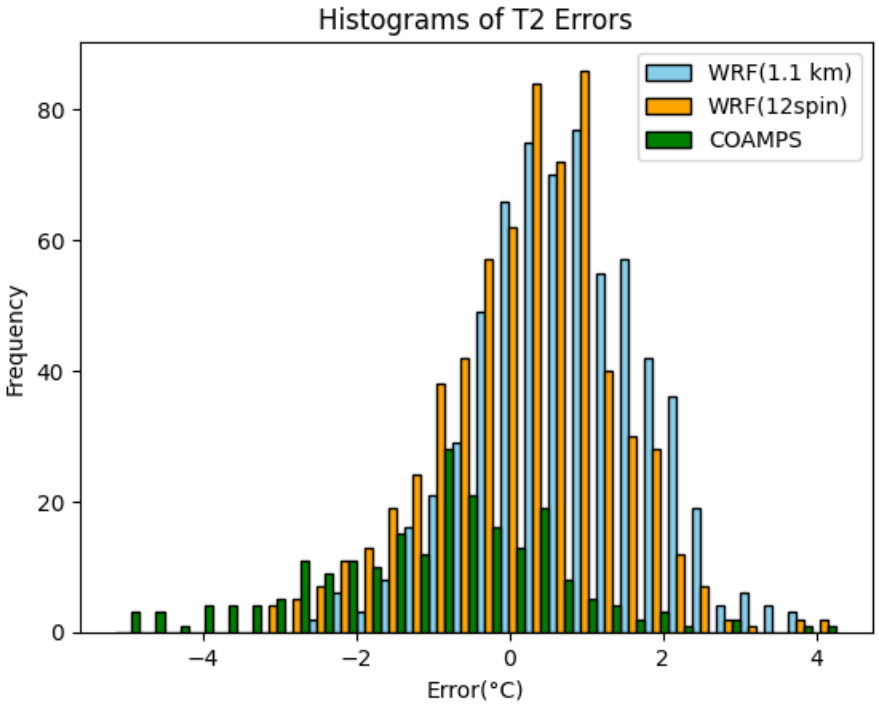


Figure 2.9 Histograms of 2 m temperature errors of the 3 models.

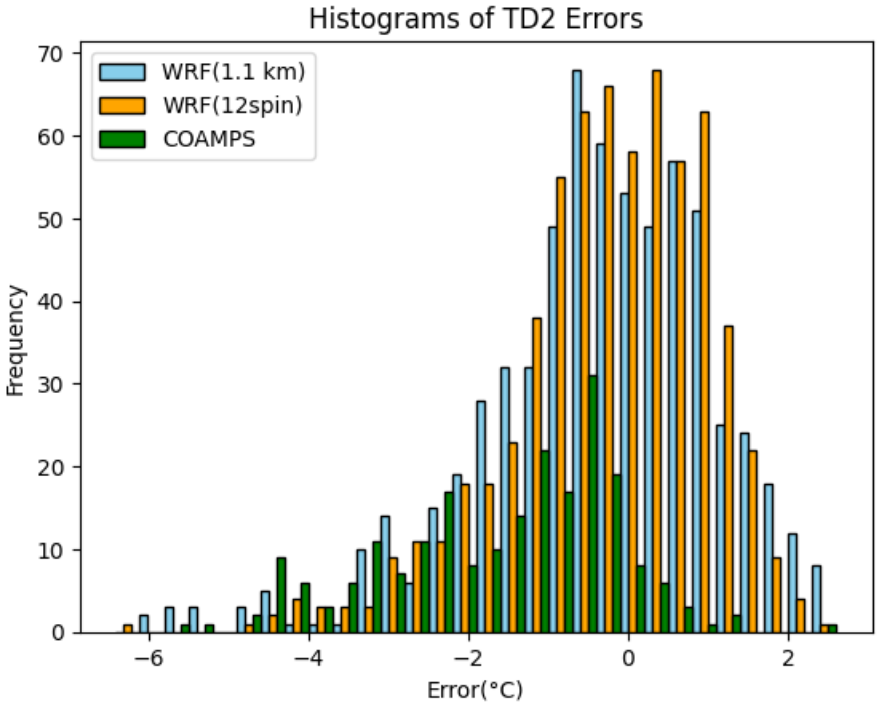


Figure 2.10 Histograms of 2 m dew point temperature errors of the 3 models.

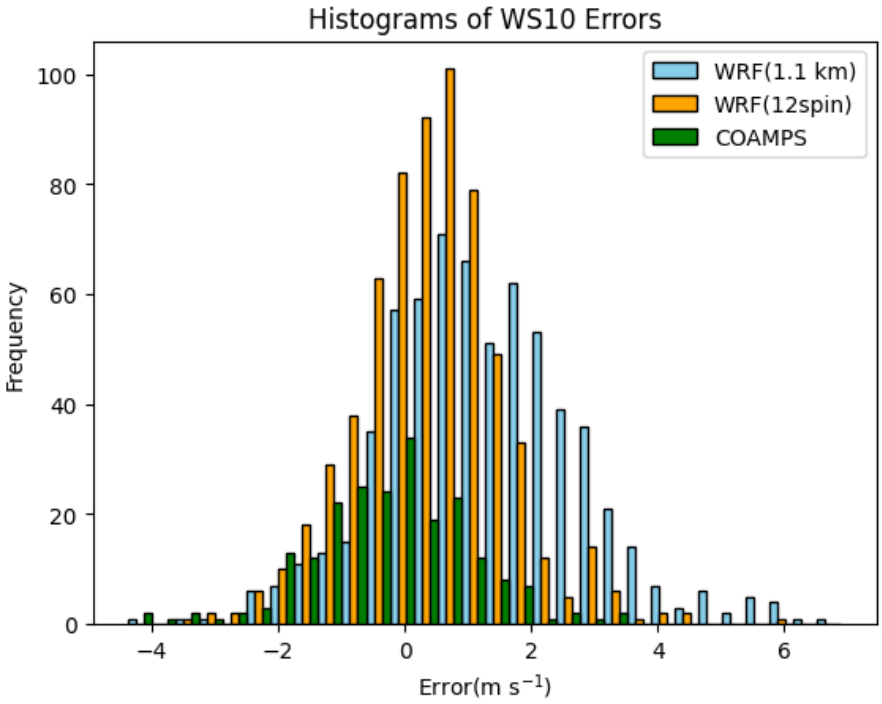


Figure 2.11 Histograms of 10 m wind speed errors of the 3 models.

For 2 m temperature, the difference between WRF (1.1 km) and WRF (12spin) is small. The peak of COAMPS is at the negative side, meaning that COAMPS has more predictions lower than the observation. For 2 m dew point, WRF (1.1 km) and WRF (12spin) are still similar. WRF (12spin) having higher frequencies than WRF (1.1 km) near 0 error shows a slight improvement. Similar to 2 m temperature, COAMPS predicts lower dew point values than the observation. For 10 m wind speed, WRF (12spin) and COAMPS have their peaks at 0 error. WRF (1.1 km) has more errors on the positive side. To summarize, the distributions of the errors are consistent with the mean absolute errors, showing that WRF (12spin) has a great 10 m wind speed improvement from WRF (1.1 km), and WRF (1.1 km) and WRF (12spin) have better T2 and TD2 predictions than COAMPS.

Since the distributions of the errors are normal, one can apply the student's t-test to show the significance of the differences with a score. Python function scipy.stats.ttest_ind is used for the test. A t-test calculates the arithmetic means between 2 samples:

$$t = \frac{\overline{x_1} - \overline{x_2}}{\sqrt{\frac{s_1^2}{N_1} + \frac{s_2^2}{N_2}}} \tag{2.1}$$

where $\overline{x_1}$ and $\overline{x_2}$ are the means, s_1 and s_2 are the standard deviations and N_1 and N_2 are the sizes of the samples. It is used to test a null hypothesis that the two samples were drawn from the same population. A p value is determined by the t score against a theoretical t-distribution by the default setting of the function. It is the probability that the difference between the two samples is simply by chance. Generally, a p value lower than chosen threshold, e.g., 5% or 1 %, means that the difference is statistically significant so that the null hypothesis is rejected and the two samples come from different populations (The Scipy community, 2025). In our case, it can be used to see if the differences in the errors between the models are simply by chance or they are significantly different. Table 2.4 summarizes the t scores and the p values between each model for the errors of T2, TD2 and WS10.

Most p values are extremely low, meaning that the differences are statistically significant. The only one that is less significant is the 2 m dew point between WRF (1.1 km) and WRF (12spin). From Fig. 2.10 the shape of the distribution is similar between the two models so the t-test suggests that the improvement of 2 m dew point error can be by chance. However, the improvements of other quantities are significant.

Table 2.4 T scores and p values between model errors of 2 m temperature, 2 m dew point and 10 wind speed.

	T2	TD2	WS10
WRF (1.1 km) vs WRF (12spin)	t score = 7.66	t score = -1.41	t score = 10.9
	p value = 3.53e-14	p value = 0.16	p value = 1.32e-26
COAMPS vs WRF (1.1 km)	t score = -15.1	t score = -11.2	t score = -14.5
	p value = 2.68e-38	p value = 2.33e-25	p value = 2.72e-39
COAMPS vs WRF (12spin)	t score = -10.9	t score = -12.6	t score = -7.05
	p value = 1.97e-23	p value = 4.17e-30	p value = 9.58e-12

2.4. Fog period

Fig. 2.12 shows the fog period with visibility < = 1 km of the five simulations in this study. COAMPS calculates visibility with 2 different methods. COAMPS (L) calculates visibility based on the amount of liquid water (Stoelinga and Warner, 1999) and COAMPS (RH) uses relative humidity (Boudala et al., 2012). WRF does not provides visibility values directly, so the values are calculated using the equation from Stoelinga and Warner (1999), and cloud water mixing ratio at the lowest model level. Based on the observational study of several fog events by Kunkel (1984), Stoelinga and Warner (1999) calculated visibility in km with a simple equation

$$\text{visibility} = \frac{-\ln(0.02)}{\beta} \tag{2.2}$$

where $\beta = 144.7C^{0.88}$ in km^{-1} is the extinction coefficient for cloud liquid water and C is the mass concentration of the liquid water in g m^{-3}. The meteorological optical range (MOR) in this

equation is 0.02. However, WMO suggests a value of 0.05, assuming that the apparent contrast of a black object is 0.05, when an observer can just see and recognize it against the horizon (WMO, 2020). The results below use the value of 0.02, which shouldn't bring too much difference since only the fog or no fog condition is being assessed.

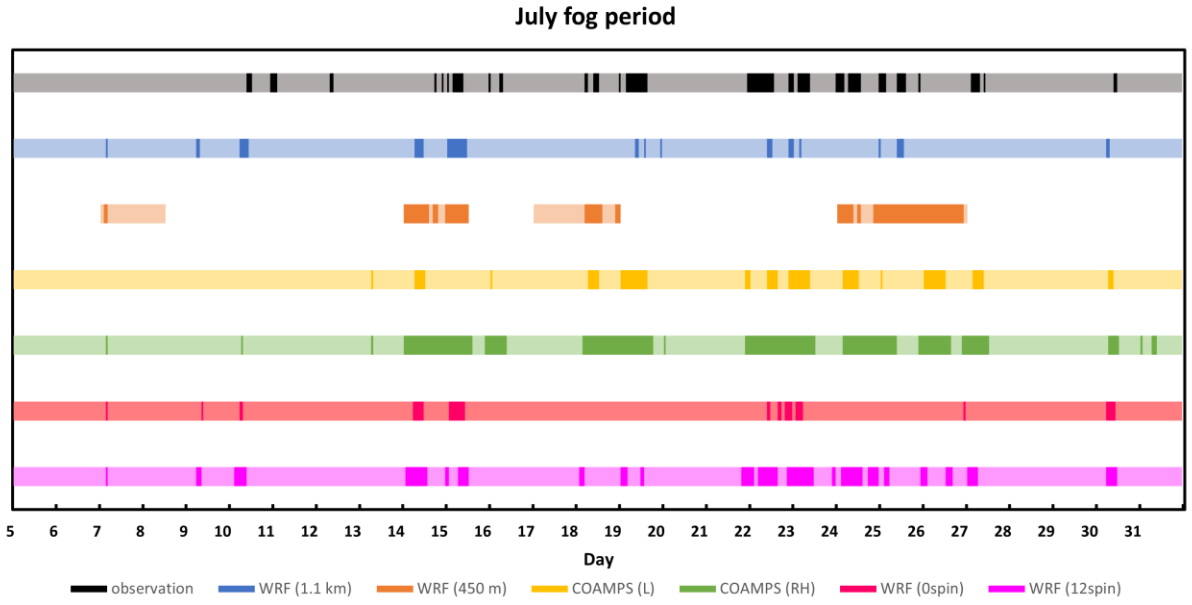


Figure 2.12 Fog period predicted by the five simulations in this study from July 5 to July 31. Darker color in a row means visibility <= 1 km. Day labels along the Day axis are at 0000 UTC.

By only changing how visibility is calculated, COAMPS (L) and COAMPS (RH) give very different results, with COAMPS (RH) having more fog events than COAMPS (L). As Dimitrova et al. (2021) pointed out, different algorithms can give very different results of visibility. WRF (450 m) also gives more fog than WRF (1.1 km). Between the two tests in Section 2.3, WRF (12spin) has more fog than WRF (0spin), with the only difference between the 2 simulations being the operation time.

Table 2.5 Fog occurrence of the five simulations.

	precision	**recall**	**F1 score**
WRF (1.1 km)	0.63	0.29	0.39
WRF (450 m)	0.43	0.94	0.59
COAMPS (L)	0.74	0.59	0.66
COAMPS (RH)	0.53	0.86	0.65
WRF (0spin)	0.50	0.21	0.30
WRF (12spin)	0.51	0.56	0.54

Table 2.5 shows some scalar attributes calculated based on the fog occurrence contingency table of each simulation. Precision, recall and F1 score are calculated, which are the measures of success of prediction for imbalanced data based on true positive, false negative and false positive (Pedregosa et al., 2011). Precision is a measure of the fraction of positive items among actually returned items, which is calculated as

$$Precision = \frac{true\ positive}{true\ positive + false\ positive}. \tag{2.3}$$

Recall is a measure of the fraction of items that were returned among all items that should have been returned, which is calculated as

$$Recall = \frac{true\ positive}{true\ positive + false\ negative}\ . \tag{2.4}$$

A system with high recall but low precision returns most positive items, but the proportion of returned results that are incorrectly labeled is high. A system with high precision but low recall returns few positive items but most of them are correct. An ideal system should have both high precision and high recall, which will return most of the positive items and most of which are correct. Therefore, the F1 score is introduced, which takes both perspectives into consideration:

$$F1 = 2 * \frac{Precision * Recall}{Precision + Recall}. \tag{2.5}$$

Between the tests in Section 2.2, WRF (1.1 km) has a very low recall, which results in its low F1 score. WRF (450 m) has a very high recall but a low precision, which means that the model has a lot of false positives. It has a F1 score of 0.59, but it only simulates some IOPs when there is fog in the observation, so the model performance is not tested with no fog periods. The two different calculations of COAMPS have very close F1 scores, which are both higher than the two WRF models. COAMPS (L) has higher precision while COAMPS (RH) has higher recall, so the F1 score can be indicative of the overall performance of a model.

For the tests in Section 2.3, both WRF (0spin) and WRF (12spin) have lower precisions than WRF (1.1 km), but WRF (12spin) has a much higher recall compared to WRF (0spin), resulting a higher F1 score. Because recall is the measurement of the items that should have been returned, it means that WRF (12spin) can predict more fog with a spin up time. Compared to WRF (1.1 km), the precisions of WRF (0spin) and WRF (12spin) are lower because more items are returned and some of them can be false positives, but the overall performance of WRF (12spin) is better with a

higher F1 score. Therefore, it shows that WRF model performance can be improved with a different domain setting. The improvement can come from that domain with 1 km resolution in the new setting covers a larger area around the island than the domain with 1.1 km resolution in the old setting. The largest domain in the old setting covering the east coast of North America may not be important.

2.5 Discussion

This chapter is an overview of the model performances in the FATIMA 2022 Sable Island field campaign. Three models, a COAMPS model and two WRF models, are used for the campaign and the results are compared to NAVCAN reports at the Sable Island airport.

It is found that the errors of 2 m temperature, 2 m dew point, 2 m relative humidity, 10 m wind speed and 10 m wind direction are relatively small among the models, but because COAMPS has the best performance on 10 m wind, another experiment is to change the domain setting of WRF (1.1 km) to match the domain setting of COAMPS. The need of spin-up time is also tested, with two different spin-up time settings, WRF (0spin) and WRF (12spin).

Model performances of fog prediction are different. COAMPS has two algorithms of calculating visibility. The algorithm using liquid water has a higher precision and the algorithm using relative humidity has a higher recall. Both have similar high F1 scores. WRF (450 m) has a high recall but a very low precision, meaning that it has a lot of false positives. On contrast, WRF (1.1 km)

has a low recall but a relatively high precision, so it does not report enough fog. It is also found that model performance of WRF can be improved by only changing the domain setting, and a spin-up time is necessary for WRF to predict fog. Since WRF starts every simulation with no liquid water, the model needs time to generate liquid water from water vapour that is provided by the given initial conditions.

This chapter has only discussed the fog occurrence as fog or no fog, but fog thickness is another issue. Therefore, in the next chapter, two periods during the campaign will be used as case studies. Results of liquid water content and droplet number concentration collected during the campaign will be used to compared to the model results.

Chapter 3

Case Studies

3.1 Introduction

The last chapter has summarized the performances of the models during the whole campaign period. However, in many algorithms e.g., Equation 2.2, visibility is determined by the liquid water content (LWC) and the droplet number concentration (QNC). Therefore, it is necessary to see these quantities in both the model and the observation. In this chapter, the observations of LWC and QNC from the FATIMA Sable Island field campaign are compared to results from WRF. Different tests are performed with different features within the model. Two IOPs are chosen for case studies. IOP 5 was from July 13 12Z to July 15 12Z, and IOP 9 was from July 21 12Z to July 22 12Z. These two IOPs were described as “excellent fog events” with respect to the length of the events and the quality of the observations.

Fig. 3.1 shows the SST on July 13th from the WRF simulation. The SST data are from the GFS FNL data and by default WRF will not update the SST values during the run. The warm Gulf Stream and the cold Labrador Current merge around Nova Scotia. Winds from the SW direction bring warmer air to the location, causing the fog in this period.

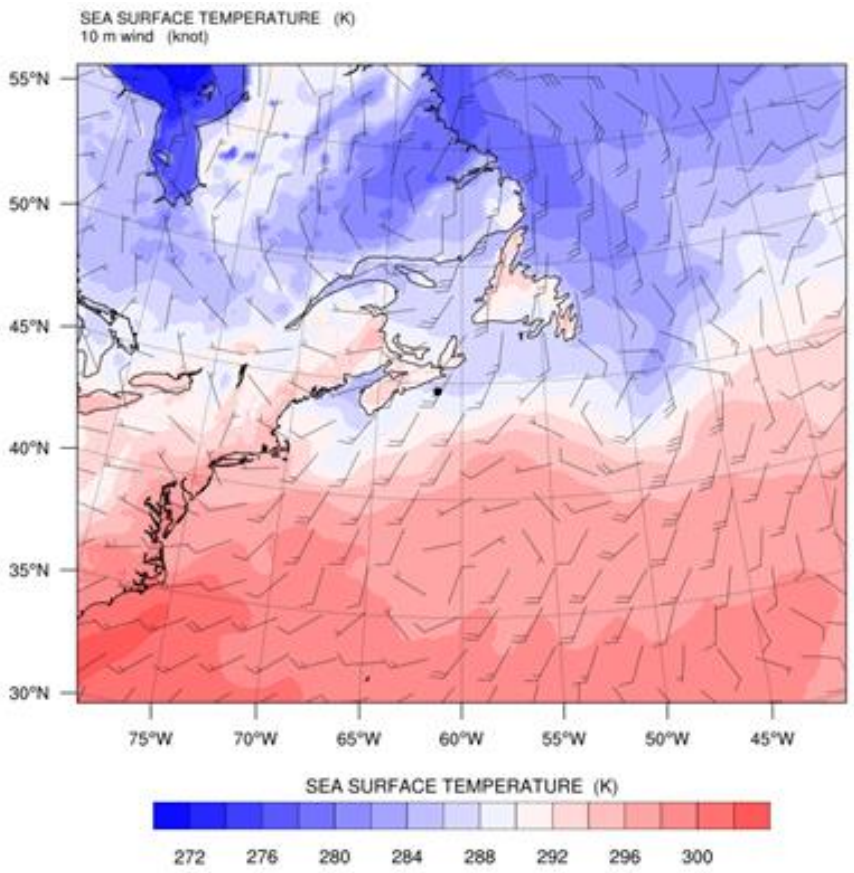


Figure 3.1 Sea surface temperature and 10 m wind on July 13th from GFS. Land grid points also have SST values, but they are not used in WRF. The black dot is the approximate location of Sable Island.

3.2 Observation

Fig. 3.2 shows the deployment of equipment in the campaign. There are three main towers with instruments at different heights. Observations from a FM-120 (fog monitor) instrument at 1.4 m are used, which is in the NE direction from the south tower, 147 m away. The instrument does not directly provide LWC. It measures the droplet spectrum between 3 μm and 50 μm and integrates it to get LWC and QNC. Additional observations of visibility from a Vaisala Forward Scatter

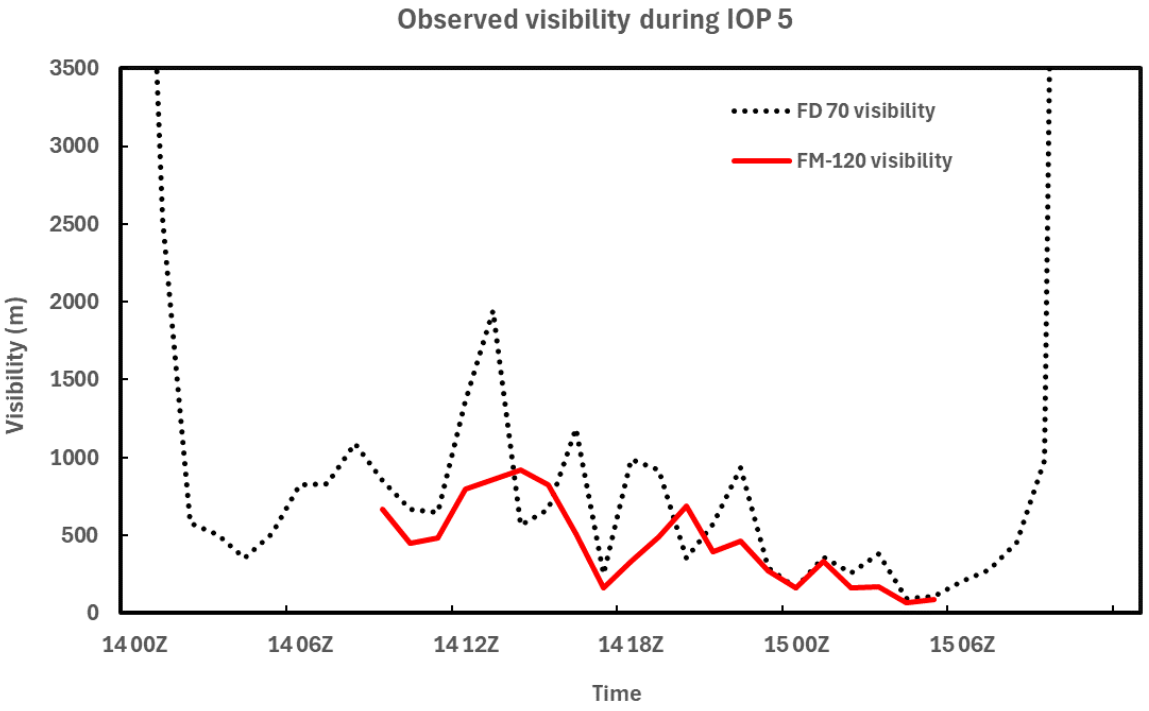
Sensor FD70 instrument at 2.5 m are used in this step to verify the results from the FM-120. The distance between the two instruments is 17 m. The values of LWC and QNC from the FM-120 are used to calculate visibility with the following equation from Isaac et al. (2020):

$$Vis = \frac{1.24 * \rho_w^{2/3}}{LWC^{2/3} N^{1/3}} \tag{3.1}$$

where ρ_w is density of water, LWC is liquid water content in kg m^{-3} and N is fog droplet number concentration in m^{-3}.

Figure 3.2 Sable Island instrument deployment during the campaign (from Fernando et al., 2025).

Fig. 3.3 shows the time series of calculated visibility from FM-120 and measured visibility from FD70 in the periods. The two measurements are comparable in some periods. There are differences because one is calculated from LWC and QNC while another one is measured from forward scattering. It is also possible that the fog was patchy, and the two instruments were in slightly different locations. The values of LWC and QNC shown in this chapter are every 15 seconds averaged to 1 hour.



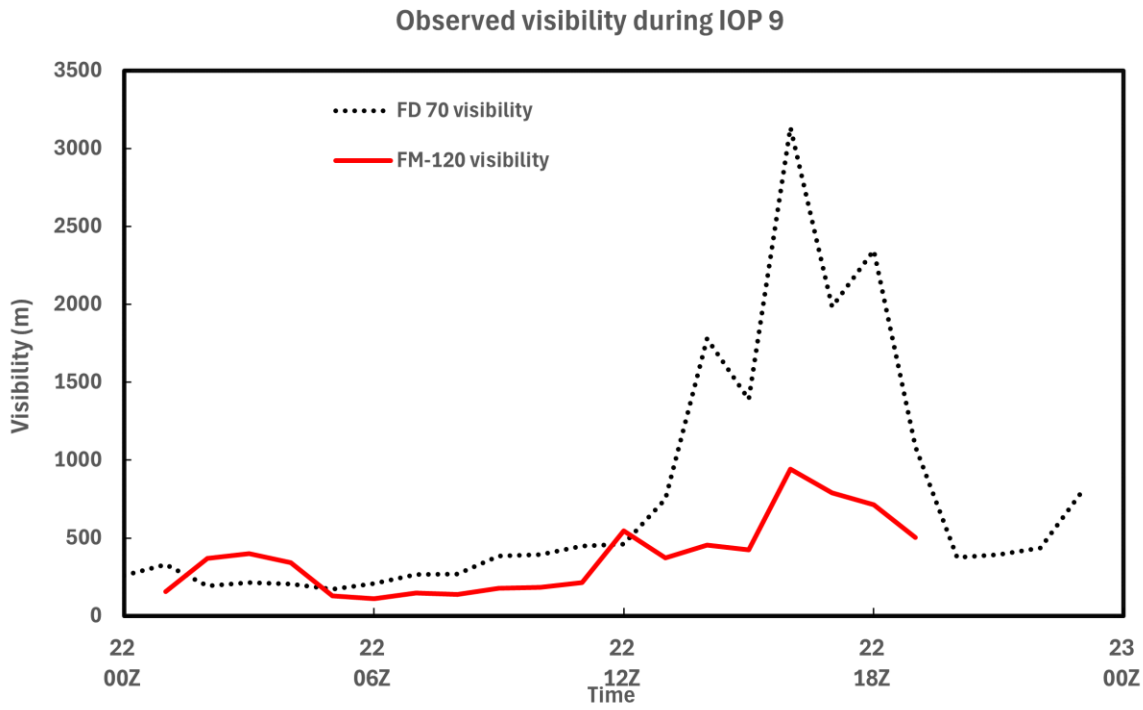


Figure 3.3 Time series of calculated visibility from FM-120 and measured visibility from FD70, in IOP5 and IOP 9. Time is in the format of DD HH. FM-120 did not cover the two full periods.

Additional observations used in this chapter include the temperature and relative humidity at 2 m from a Vaisala HMP-155 instrument at the central tower. The values are every 1 min averaged to 1 hour. There are soundings from weather balloons, with measurements of pressure, temperature, relative humidity, wind speed and wind direction. Sounding measurements were usually performed twice a day, with increased frequency during the IOPs. There were also wind measurements during the campaign, however, unfortunately those at 10 m were not functional for the periods so they are not used in comparisons. The observations from NAVCAN at the airport used in Chapter 2 are still used in this chapter for comparisons.

3.3 WRF Parameterization

Chapter 2 does not focus on the physical parameterizations in WRF because it serves as an overview. This chapter involves comparisons of different parameterizations, therefore, this section explains how the physical parameterizations work in WRF, and the schemes chosen in this chapter.

3.3.1 Introduction

In WRF, different physics components interact with each other. There are mainly five components: radiation, planetary boundary layer (PBL), surface, cumulus and microphysics. The radiation is called first to provide radiative fluxes to the surface scheme. Then the surface component will provide turbulent heat, moisture and momentum fluxes to the PBL component. The PBL and cumulus components both provide tendencies of atmospheric variables. The

microphysics component is called last to update the atmospheric state at the end of the model time step (Skamarock et al., 2019). Fig. 3.4 shows the interactions between different components. This study focuses on the PBL and the microphysics components. For each component there are different schemes for the users to choose from.

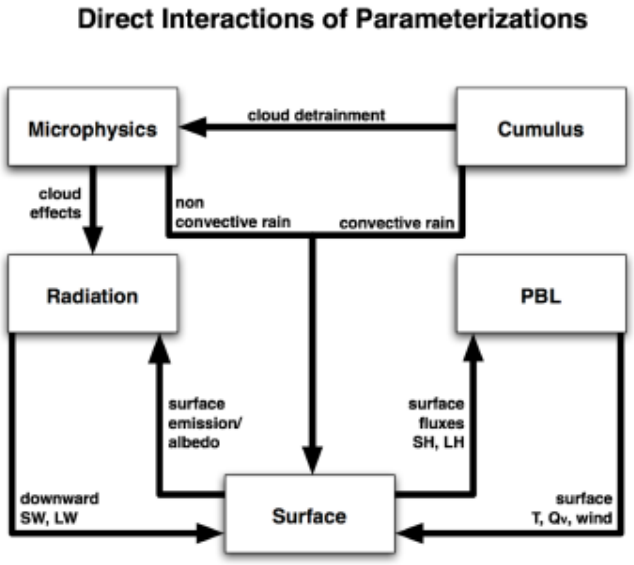


Figure 3.4 Interactions between different physics components in WRF (from Skamarock et al., 2019).

3.3.2 The PBL Component

In WRF, the PBL component is responsible for vertical sub-grid scale fluxes due to eddy transport in the whole atmospheric column, not just the boundary layer. Moisture, heat and momentum will be exchanged through the eddies that cannot be explicitly represented on grid scales. Therefore, the PBL component provides tendencies of temperature, moisture and momentum in the entire atmospheric column (Skamarock et al., 2019). There are two major

questions to consider when choosing a PBL scheme: the order of turbulence closure and whether it is a local or nonlocal approach.

In the equations of motion in a turbulent flow, the variables are expressed with mean and fluctuation parts. The mean component represents the time average of the background atmosphere, while the fluctuation term represents the deviations from the background mean state. With such decomposition and after time or ensemble averaging, mathematically the equations of motion always contain more unknown terms than known terms, where the unknown terms are typically an order higher than the known terms. Therefore, a n^{th}-order closure scheme will empirically relate the unknown terms of moment n+1 to lower moment known terms. N is usually an integer, but some schemes will have non-integer orders (Cohen et al., 2015).

The second question is whether the scheme is local or nonlocal. When determining the variables at one vertical level, local schemes only allow local gradients represented via the directly adjacent levels to affect this level, while nonlocal schemes can let multiple vertical levels affect it. Therefore, nonlocal schemes can resolve the deep convection in the atmosphere better. Local schemes can be improved by introducing higher orders of closure (Cohen et al., 2015). In general, nonlocal PBL schemes predict relatively stronger wind speeds and warmer temperatures at low levels of the boundary layer over night compared to local schemes (Jahn and Gallus, 2018).

The scheme to be used in this study is the Mellor-Yamada-Nakanishi-Niino (MYNN) scheme (Nakanishi and Niino, 2006 and 2009), which was used in Chen (2021) based on the comparisons

among many WRF parameterizations. It was first implemented into WRF in version 3.1. Since then, it has been extensively modified to improve forecast skill in support of the National Weather Service, the Federal Aviation Administration, and users in the renewable-energy industry (Olson et al, 2019). The MYNN scheme discussed here is based on the present-day version (version 4.2 and newer). It is a local scheme with two versions available: level 2.5 with an order 2.5 closure and level 3 with an order 3 closure. In WRF version 4.4 and before, they can be activated in the namelist with bl_pbl_physics = 5 for MYNN level 2.5 and bl_pbl_physics = 6 for MYNN level. In version 4.5 and newer, bl_pbl_physics = 5 chooses the MYNN schemes and a new option bl_mynn_closure chooses the corresponding order (=2.5 or 3.0). Compared to an order 2.0 closure, the MYNN level 2.5 scheme predicts TKE as an extra prognostic variable, while level 3 adds variances of potential temperature, moisture and their covariances (Skamarock et al., 2019). Chapter 2 uses the MYNN level 2.5 version.

3.3.3 The Microphysics Component

As the final component to be called in a time step, the microphysics component takes all tendencies from other physics components and resolves water vapour, cloud and precipitation processes. There is no doubt that this component is of importance to fog prediction in the model. Different microphysics schemes have different numbers of moisture variables. Some only have water vapour, cloud water and rain, while others can include snow, ice or hail.

Different variables can have different moments of order. Single-moment microphysics schemes only calculate the amount of the moisture variables (e.g., cloud water mixing ratio QCLOUD and

rain water mixing ratio QRAIN), while double-moment schemes also calculate the number of the variable (e.g., cloud water droplet number concentration QNCLOUD and rain water droplet number concentration QNRAIN).

The scheme to be used in this study is the Thompson microphysics scheme. Two versions are available in WRF: the original Thompson microphysics scheme (Thompson et al., 2008, activated with mp_physics = 8) which was also used in Chapter 2 and Chen (2021), and the Thompson aerosol-aware microphysics scheme (Thompson and Eidhammer, 2014, activated with mp_physics = 28). The original Thompson scheme is a hybrid scheme, with single-moment for each of water vapour, cloud water, snow and a graupel-hail categories, and double-moment for cloud ice and rain. The aerosol-aware scheme is an updated version so that it is double-moment for cloud water as well.

Since activation is important in determining the number concentration, the number concentration of aerosol is also calculated. Instead of considering all different aerosols, the scheme puts the aerosols into two groups: water-friendly aerosol and ice-friendly aerosol. The water-friendly aerosol is a category including sulfates, sea salts and organic matters, with black carbon also included in a newer version. Thompson and Eidhammer (2014) claimed that several studies have shown that chemical properties are not nearly as important as the aerosol number and size distribution. The water-friendly aerosol is also referred to cloud condensation nuclei (CCN). The Thompson microphysics scheme has an initial profile of CCN concentration. Weston et al. (2022) argued that the value at the surface is 300 cm^{-3} for all grid points but for a water surface it should

be adjusted.

Both Thompson schemes use a generalized gamma distribution for size distributions of their predicted water species except for snow:

$$N(D) = \frac{N_t}{\Gamma(\mu+1)}\lambda^{\mu+1}D^{\mu}e^{-\lambda D} \tag{3.2}$$

where N_t is the total number of particles in the distribution, D is the particle diameter, Γ is the gamma function, λ is the distribution's slope and μ is the shape parameter. For the original Thompson scheme with single-moment for cloud water, N_t = 100 cm^{-3} to represent "clean" air over the ocean, and Thompson et al. (2008) advised the users to set N_t according to the users' own data. Then an empirical relationship between N_t and the shape parameter μ_c for cloud water is derived as:

$$\mu_c = \min\left(\frac{1000}{N_t} + 2, 15\right) \tag{3.3}$$

where N_t is in 100 cm^{-3} and the slope λ is calculated as:

$$\lambda = \left[\frac{\pi}{6}\rho_w \frac{\Gamma(4+\mu)}{\Gamma(1+\mu)}\left(\frac{N_t}{LWC}\right)\right]^{\frac{1}{3}} \tag{3.4}$$

where ρ_w is water density and LWC is in kg cm^{-3}.

Equation 3.5 from is the governing conservation equation for all species mass mixing ratio or number concentration.

$$\frac{\partial \Phi}{\partial t} = -\frac{1}{\rho}\nabla\cdot(\rho \boldsymbol{U}\Phi) - \frac{1}{\rho}\frac{\partial(\rho V_{\Phi}\Phi)}{\partial z} + \delta\Phi + S_{\Phi} \tag{3.5}$$

where Φ is mass mixing ratio or number concentration, t is time, ρ is the air density, $\boldsymbol{U}$ is the 3D wind vector, z is height, V_Φ is the fall speed of Φ, $\delta\Phi$ represents the subgrid-scale mixing operator, and S_Φ represents the various microphysical process rate terms (Thompson and Eidhammer, 2014). It includes advection, deposition, mixing and various microphysical processes. The equation is the same for both versions of the scheme. Equation 3.5 and 3.6 describe the S_Φ term for cloud water droplet number concentration and water-friendly aerosol number concentration introduced in the aerosol-aware scheme.

$$\frac{dN_c}{dt} = -(\text{rain, snow, graupel collecting droplets}) - (\text{freezing into cloud ice}) - (\text{collide/coalesce into rain}) - (\text{evaporation}) + (\text{CCN activation}) + (\text{cloud ice melting}) \quad (3.5)$$

$$\frac{dN_{wfa}}{dt} = -(\text{rain, snow, graupel collecting aerosols}) - (\text{homogeneous nucleated deliquesced aerosols}) - (\text{CCN activation}) + (\text{cloud and rain evaporation}) + (\text{surface emissions}) \quad (3.6)$$

Droplet condensation only occurs when relative humidity reaches 100%. For the activation rate, which is the percentage of CCN becoming water droplets, both Thompson schemes calculate it based on five quantities: CCN concentration, updraft speed, ambient temperature, CCN mean diameter and hygroscopicity. Each quantity has a list of values, and they are used for a look-up table in the model. For example, the updraft speed has 0.01, 0.0316, 0.1, 0.316, 1.0, 3.16, 10.0,

31.6 and 100.00, in m s^{-1}. The minimum updraft speed of 0.01 m s^{-1} means that once the activation begins, even if model updraft is lower than 0.01 m s^{-1}, the scheme will still use this value when referring to the look-up table for CCN activation rate. Weston et al. (2022) pointed out that this updraft speed, which equates to a cooling rate of 0.23 K h^{-1}, is too high for fog and can overestimate fog droplet activation in the model.

3.3.4 Other Components

This subsection describes the schemes used for other components. They remain the same in the different runs described in the next section. The options below are the default of WRF, for physic_suite = 'CONUS'.

Cumulus parameterization is responsible for the sub-grid scale effects of convective and shallow clouds. Similar to the PBL component, it represents vertical fluxes but in coarser resolutions. It is suggested to use cumulus parameterization for domain resolution larger than 10 km, and it should not be used when it is smaller than 4 km (Skamarock et al., 2019). Between 10 km and 4 km is a grey zone that the user can choose to turn it on or not based on performance. In this study, the domain resolutions are 9 km, 3 km and 1 km. Therefore, only domain 1 with 9 km turns on the cumulus parameterization. The Tiedtke cumulus parameterization scheme (cu_physics = 6, Zhang et al., 2011) is chosen.

Radiation is important in fog since water droplets are efficient in absorbing and emitting

longwave radiation. In a radiation fog, after the lowest layer of fog has formed due to cooling, the fog droplet will absorb the longwave radiation. When the fog is thick enough, part of the radiation is emitted back to the surface, shutting down the surface cooling. On the other hand, the radiation emitted upward at the top does not come back, so that further cooling happens at the fog top. This allows fog top cooling to become the dominant mechanism for further fog growth (Wilson and Fovell, 2018).

The radiation component provides atmospheric temperature tendencies due to radiative flux. The Rapid Radiative Transfer Model for GCMs (RRTMG, Iacono et al., 2008) scheme is chosen for both longwave and shortwave schemes (ra_lw_physics = 4, ra_sw_physics = 4). Radiation processes in the model are expensive to run, therefore, it is not run for every model time step. It is recommended to set the time step as 1 minute per km of the resolution. The radiation component is run every 9 minutes given that the largest domain has the resolution of 9 km. The scheme also uses a look-up table for efficiency.

3.4 Experiment Setup

3.4.1 Model Setup

Since it is shown that domain setup of WRF (12spin) in Chapter 2 can have better results, the same domain setting is used in this chapter. It has 3 domains with resolutions of 9 km, 3 km and 1 km (Fig. 2.8), and 67 vertical levels are used. The experiment contains runs with different schemes and features in WRF. Model results are output every 1 minute and averaged to 1 hour. The WRF configuration used in this chapter is provided in Appendix B.

3.4.2 Different Runs

The control run is a setting with the original Thompson microphysics scheme and MYNN level 2.5 PBL scheme. They are the simplest versions of the schemes. This combination was also used in the last chapter. In the second run, the Thompson aerosol-aware microphysics scheme is used, and the PBL scheme remains unchanged. This is to test the importance of droplet number concentration in the model. In the third run, the MYNN level 3.0 PBL scheme is used, with the aerosol-aware microphysics scheme. MYNN level 3.0 will calculate the variance and covariance of potential temperature and liquid water mixing ratio, but it will require significantly higher computational cost. This run can test if it is worth using the option.

In the fourth run, the PBL and microphysics schemes are the same as the second case, with an additional code modification. The CCN concentration at the surface of the initial CCN profile is changed from 300 cm^{-3} to 100 cm^{-3}. The fifth and last run is an experimental run, meaning that the change is not suggested by the developers. There is a further modification to the fourth case. Since it is suspected that the minimum updraft speed of CCN activation is too high for fog, it is changed from 0.01 m s^{-1} to 0.1 m s^{-1} to see the sensitivity of condensed water to this number. A previous test to change the value to 0.001 m s^{-1} show no difference probably because it is out of the range of the look-up table. Table 3.1 summarizes the scheme used for all runs. All the runs also include a spin-up time of 12 hours.

Table 3.1 Summary of all five runs in Chapter 3.

	MYNN PBL	**Thompson microphysics**	**initial surface CCN**	**minimum updraft for CCN activation**
MYNN2MP8	level 2.5	original	300 cm^{-3}	0.01 m s^{-1}
MYNN2MP28	level 2.5	aerosol-aware	300 cm^{-3}	0.01 m s^{-1}
MYNN3MP28	level 3	aerosol-aware	300 cm^{-3}	0.01 m s^{-1}
initial100	level 2.5	aerosol-aware	100 cm^{-3}	0.01 m s^{-1}
initial100_w0.1	level 2.5	aerosol-aware	100 cm^{-3}	0.1 m s^{-1}

3.5 Results

3.5.1 IOP5

Weather condition

Fig 3.5 shows the night microphysics GEOS-16 satellite at the chosen times. A large band of fog has covered offshore Nova Scotia, including Sable Island. Only nighttime is shown since night microphysics only works during nighttime, although according to the RH observation in Fig. 3.7, the fog event was from 01Z, July 14 to 09Z, July 15. To better show the fog event in the model, results shown from Fig. 3.6 are from 15Z, July 13 to 15Z, July 15, not exactly the same as the IOP time (12 Z, July 13 – 12Z, July 15).

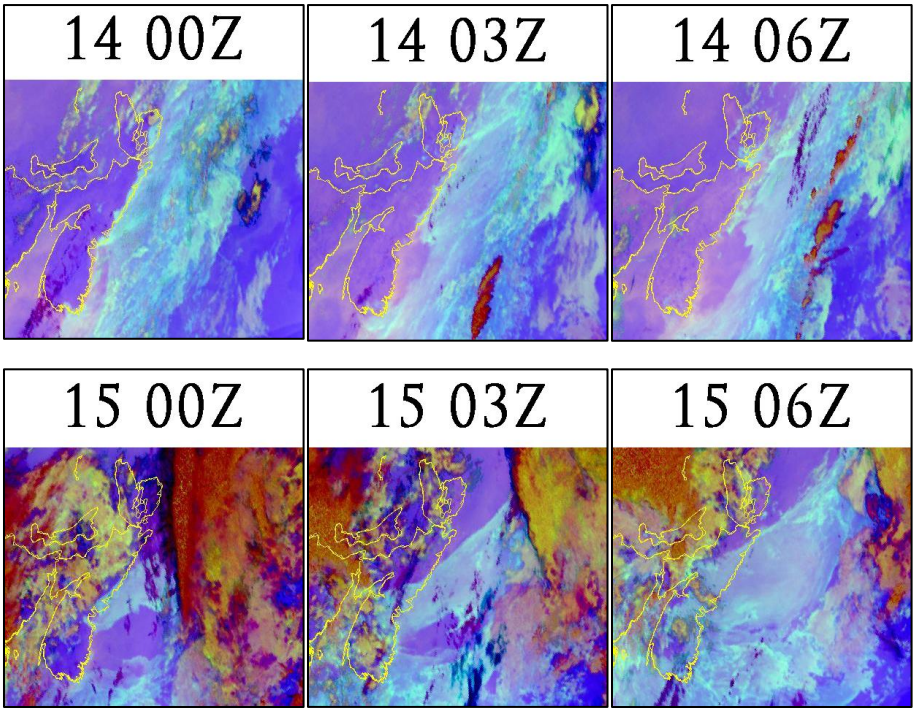


Figure 3.5 Night microphysics images from GOES-16 satellite with Nova Scotia on the left. Land mask is drawn with yellow lines. Time stamps are in the format of DD HH. The bright blue (aqua) colour means fog. The brown to red colour means high, thick cloud.

Fig. 3.6 – 3.9 show the 2 m temperature, 2 m relative humidity, 10 m wind speed and 10 m wind direction of the observations and different model runs. NAVCAN reports and the field measurements match very well, except that the field measurements give 100% RH on the 14th while NAVCAN reports have only up to 99%. All model runs produced close values of these variables. Between observations and the model results, WRF predicted wind speed and direction well. However, the model always has higher 2 m temperatures than the observed values. The model also predicted a fog starting from 21Z, July 13, 6 hours earlier than the observation, but model RH decreases on July 14 between 09Z and 21Z. Model RH also dropped rapidly after 11Z, July 15.

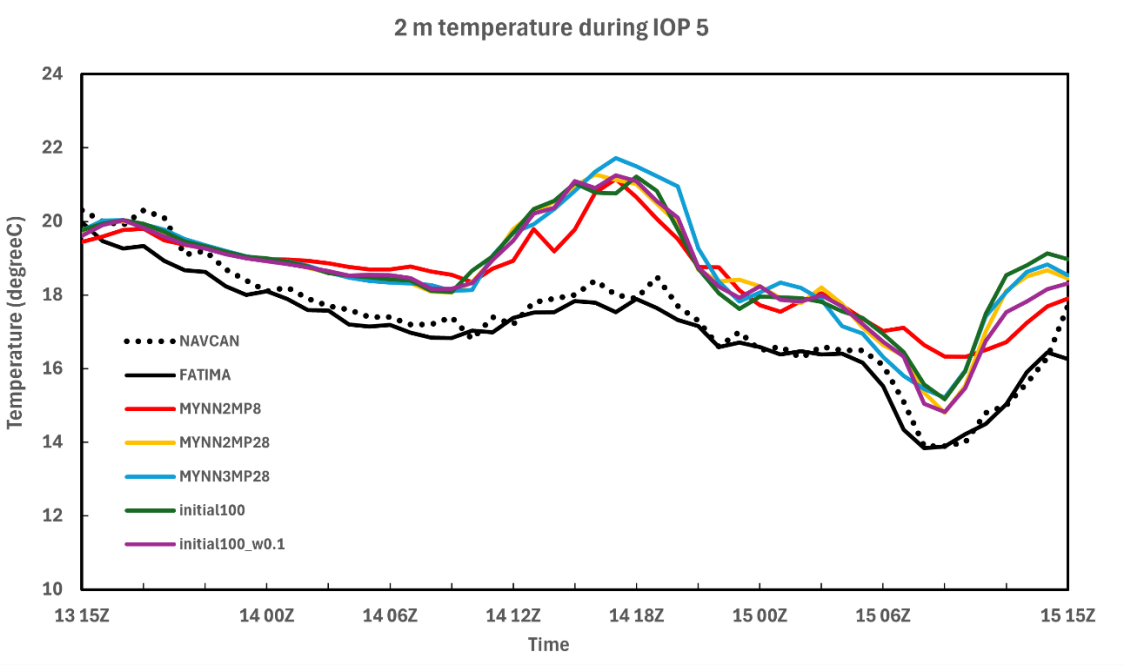


Figure 3.6 2 m temperature from the NAVCAN reports, field observations and five model runs during the period.

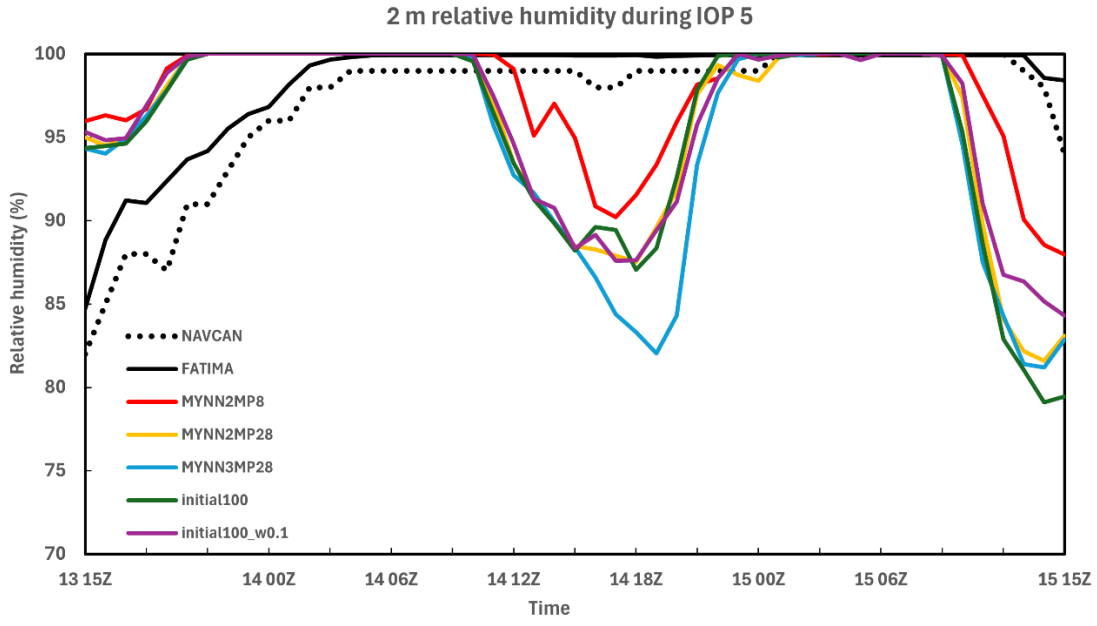


Figure 3.7 2 m relative humidity from the NAVCAN reports, field observations and five model runs during the period.

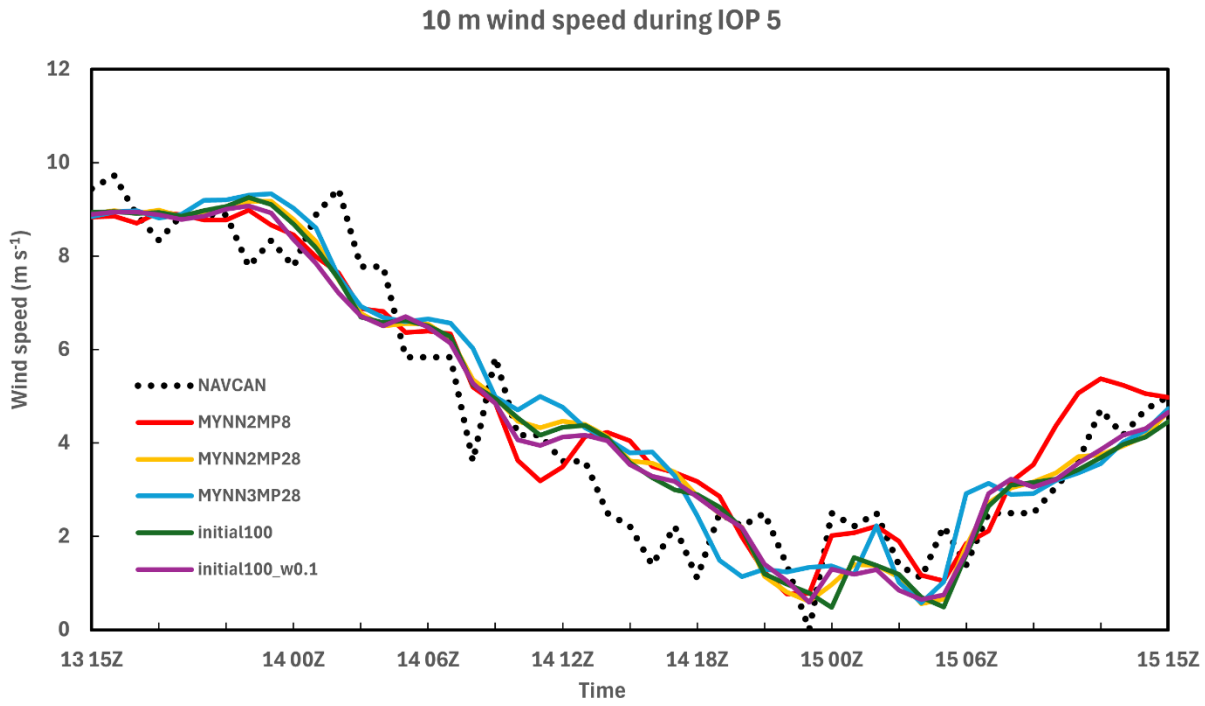


Figure 3.8 10 m wind speed from the NAVCAN reports and five model runs during the period.

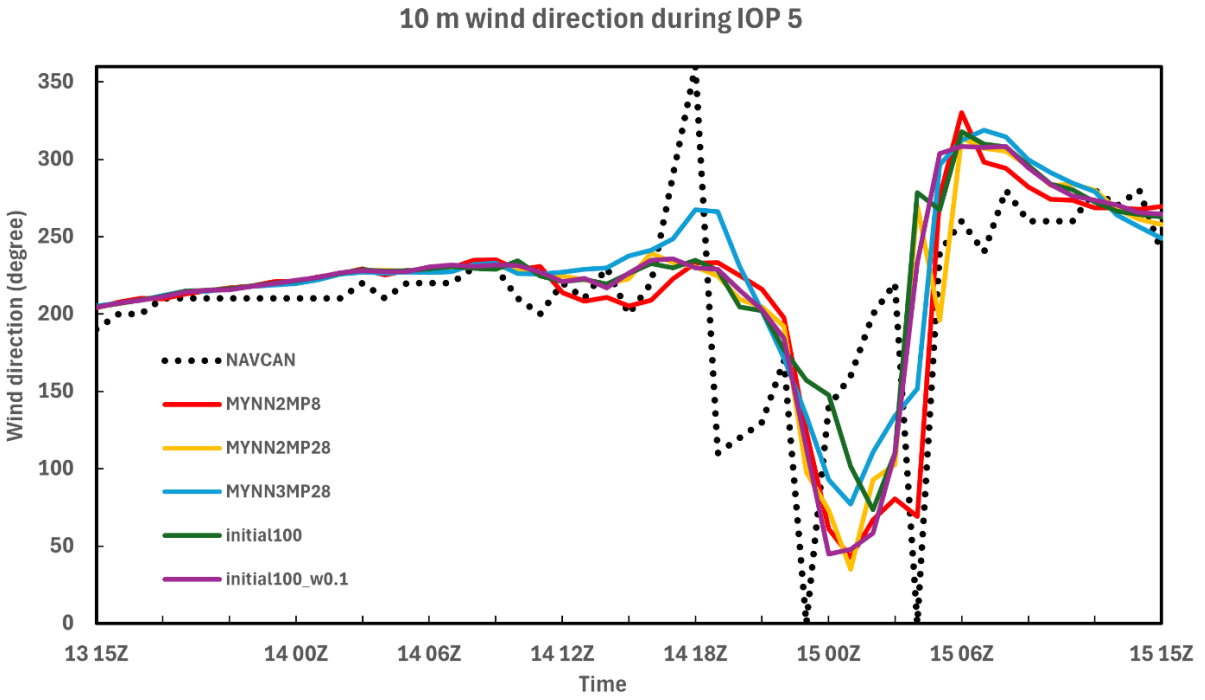


Figure 3.9 10 m wind direction from the NAVCAN reports and five model runs during the period.

Fig. 3.10 and Fig. 3.11 show the profiles of temperature and relative humidity at 06Z, July 14. All model runs have very close temperature profiles. In the model, temperature near the surface is higher than the sounding, which is consistent with the 2 m temperature in Fig. 3.6. From about 150 m to 1200 m, the model is colder than the observed values. From 1200 m to 3000 m, the two are very close.

For RH, the model can successfully catch the fog top at 600 m with near 100% RH. Above that, RHs in all model runs are lower compared to the sounding. MYNN2MP8 has a higher RH above the fog top, up to 1500 m. At the same height, it also has a lower temperature. The low temperature has a lower saturation vapour mixing ratio and thus lead to a higher RH. From Fig. 3.6 and Fig. 3.7, on July 14, between 09Z and 12Z when the air at does not reach saturation, the 2 m RH in MYNN2MP8 is higher than other model runs, while its 2 m temperature is lower. Therefore, it is consistent that a lower temperature results in a higher RH. On the other hand, above 600 m MYNN3MP28 has a lower RH, and its temperature is slightly higher too. However, the air in the model is still dryer than the sounding because the difference in RH is large but not the temperature.

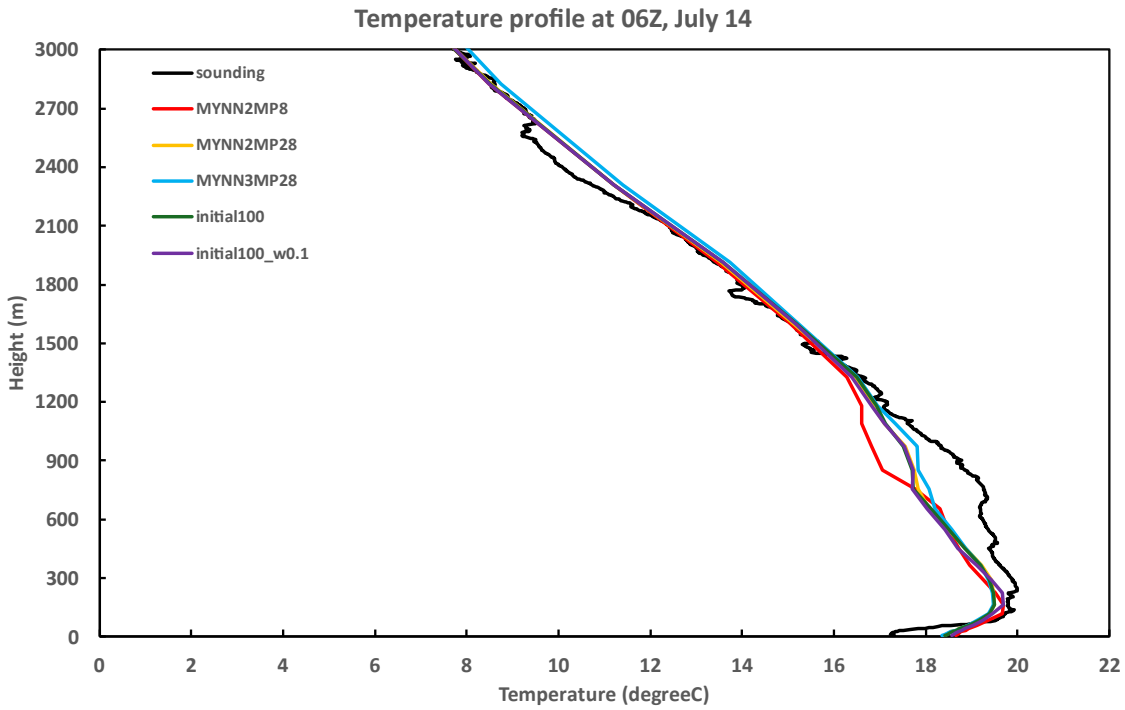


Figure 3.10 Temperature profiles from the weather balloon and five model runs at 06Z, July 14.

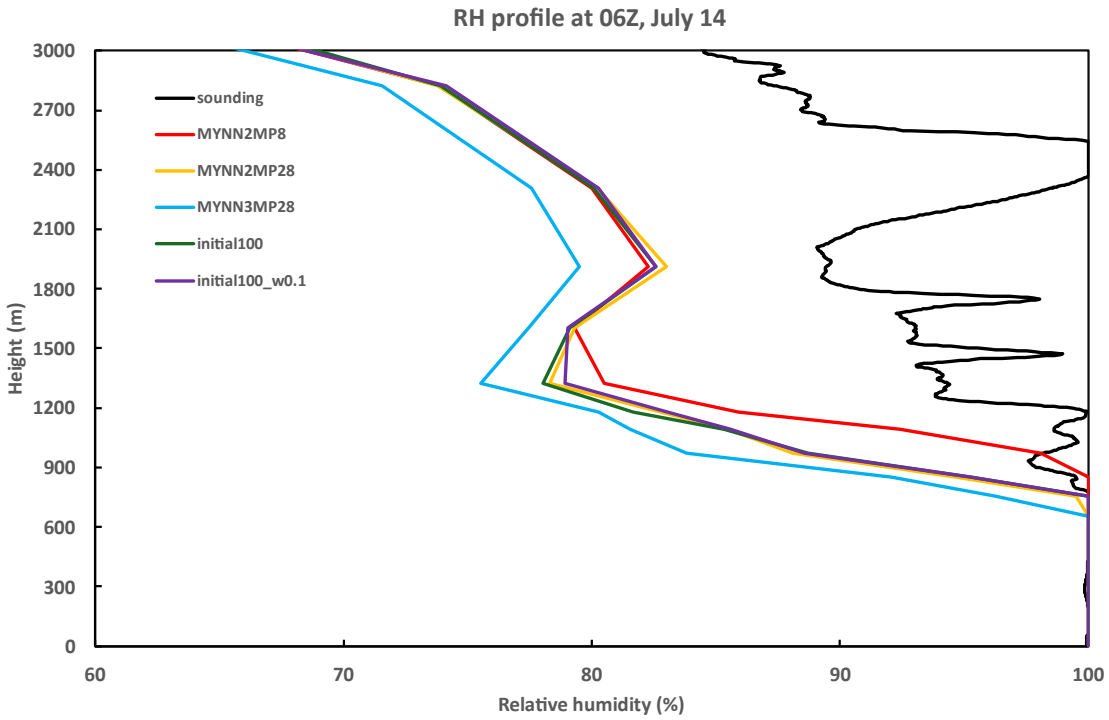


Figure 3.11 Relative humidity profiles from the weather balloon and five model runs at 06Z, July 14.

LWC and QNC

Fig. 3.12 and Fig. 3.13 show the LWC and QNC from the FM-120 observations at 1.4 m and the model results at 3.7 m. During IOP5, the instrument was only operated in 2 periods: 15Z, July 13 to 00Z, July 14 and 09Z, July 14 to 05Z, July 15. Based on the RH observation, the fog event occurred at 03Z, July 14, and started to dissipate from 14Z, July 15. During the second period when FM-120 data is available, LWC and QNC are increasing. Except for the period when the model misses the fog, the overall LWC in the model is higher than the observation. The observation has the highest value of 0.3 g m^{-3} but the model consistently has 0.4 g m^{-3} in the first half of the period which seems to be a false alarm. Except the first WRF run MYNN2MP8 which has a fixed QNC of 100 cm^{-3} when liquid water appears, other runs have much higher QNC than the observation. In the observation, QNC can increase from 20 cm^{-3} to up to 80 cm^{-3}, but the model has values higher than 250 cm^{-3}.

MYNN2MP8 also has a peak of LWC at 10Z, July 15, which is later than other runs. From Fig. 3.6 2 m temperature in MYNN2MP8 is higher than the others. Therefore, there can be less condensation due to a higher temperature. Although the fixed QNC in MYNN2MP8 is much lower than other runs, it does not seem to make a significant difference. In the first half of the period, initial100_w0.1 has the highest LWC and QNC, which means that the modified minimum updraft speed in the Thompson microphysics scheme does affect condensation. A higher updraft will increase the amount of liquid water in the air, which agrees with the results of Weston et al. (2022). Other than that, not much difference is found among all the five model runs.

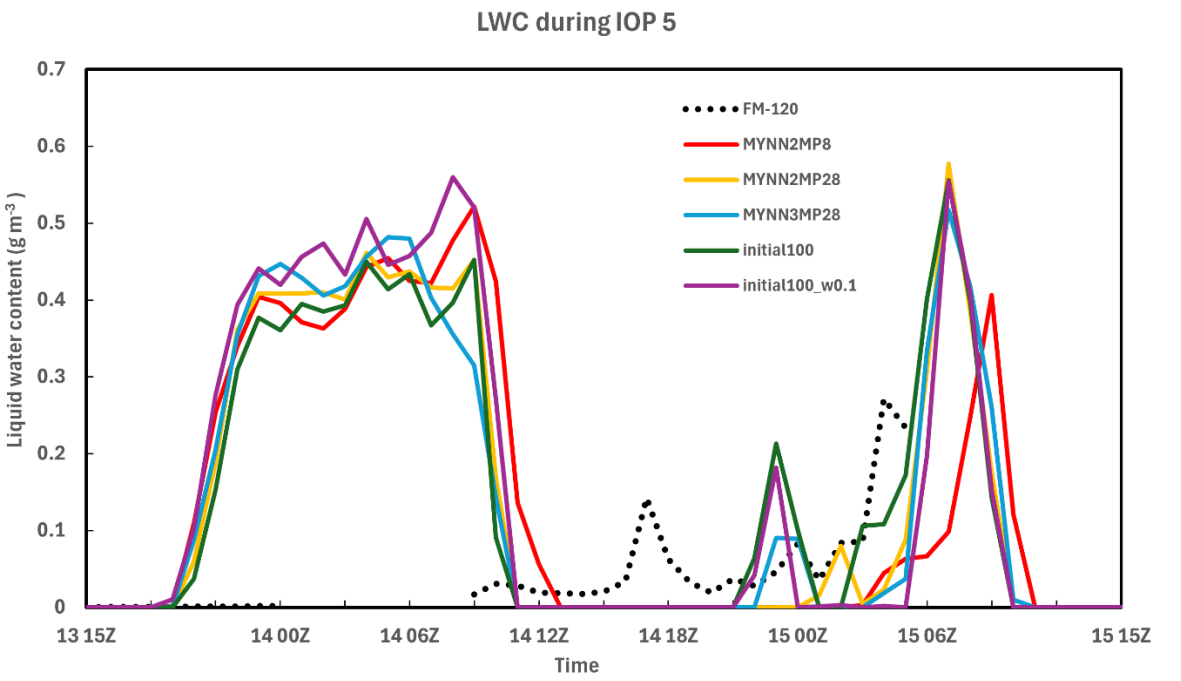


Figure 3.12 Liquid water content in g m^{-3} from FM-120 observation at 1.4 m and five model runs at 3.7 m during the period.

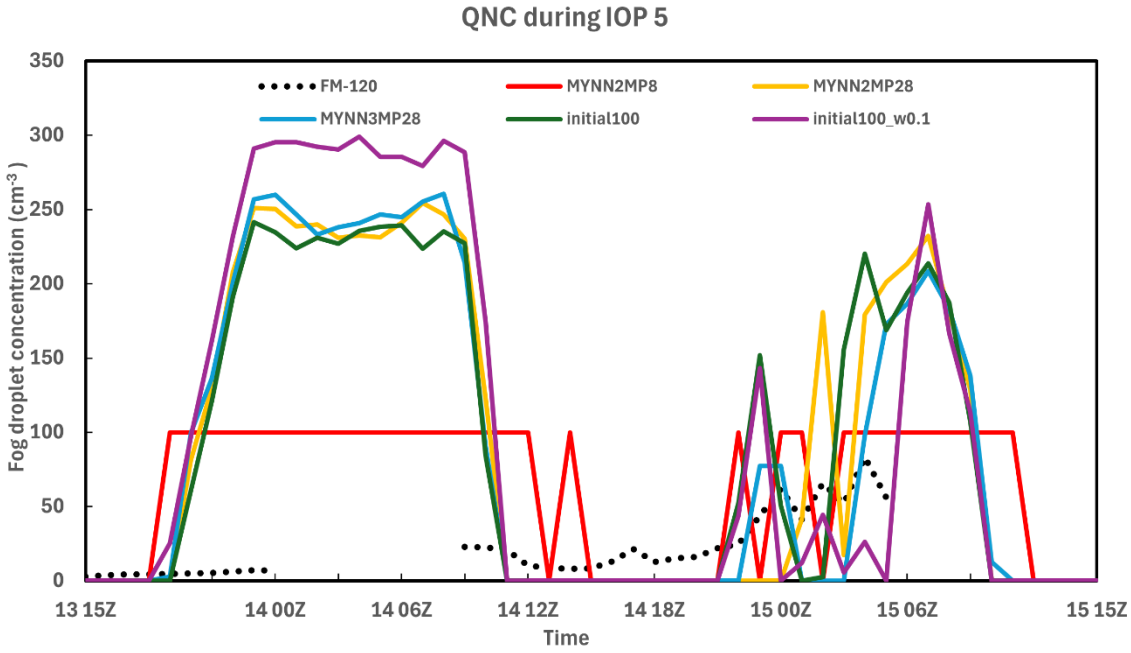


Figure 3.13 Droplet concentration in cm^{-3} from FM-120 observation at 1.4 m and five model runs at 3.7 m during the period.

3.5.2 IOP9

Weather condition

Fig 3.14 shows the satellite images of night microphysics in the early morning of July 22nd. According to the RH observation, the fog started to dissipate from July 22 16Z and seemed to recover in late afternoon at 21Z. Fog in the model dissipated and formed again earlier than the observation. To show this, different from the IOP period (July 21 12Z - July 22 12Z), the results below are from July 22 00Z to July 23 00Z to cover the fog period. From Fig. 3.15 to Fig. 3.18, there are similarities and differences between IOP 5 and this IOP. NAVCAN reports and the field measurements are still close, except that RH decreases more in the NAVCAN reports in the second half period. 2 m temperature in the model is still higher than the observation. In IOP 5, wind speed changes from 10 m s^{-1} to near 0, and there is also a change in direction. Model results are close to the observation. However, in IOP 9, winds are more consistent over time. Model wind speed is higher than the observation, and model wind direction is more westly.

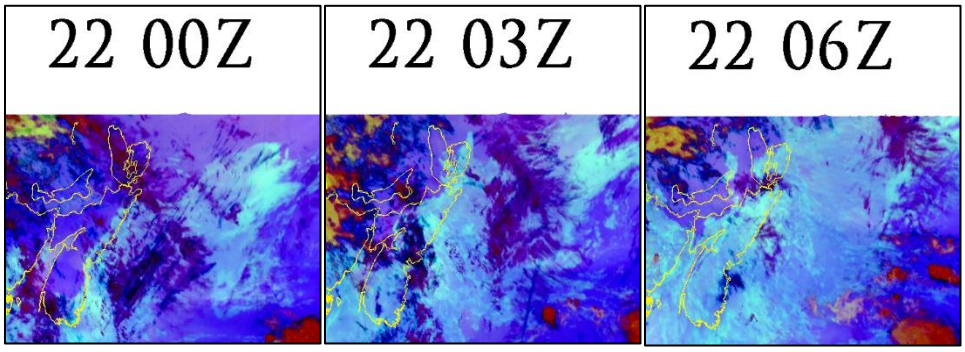


Figure 3.14 Night microphysics images from GOES-16 satellite with Nova Scotia on the left. Land mask is drawn with yellow lines. Time stamps are in the format of DD HH. The bright blue (aqua) colour means fog.

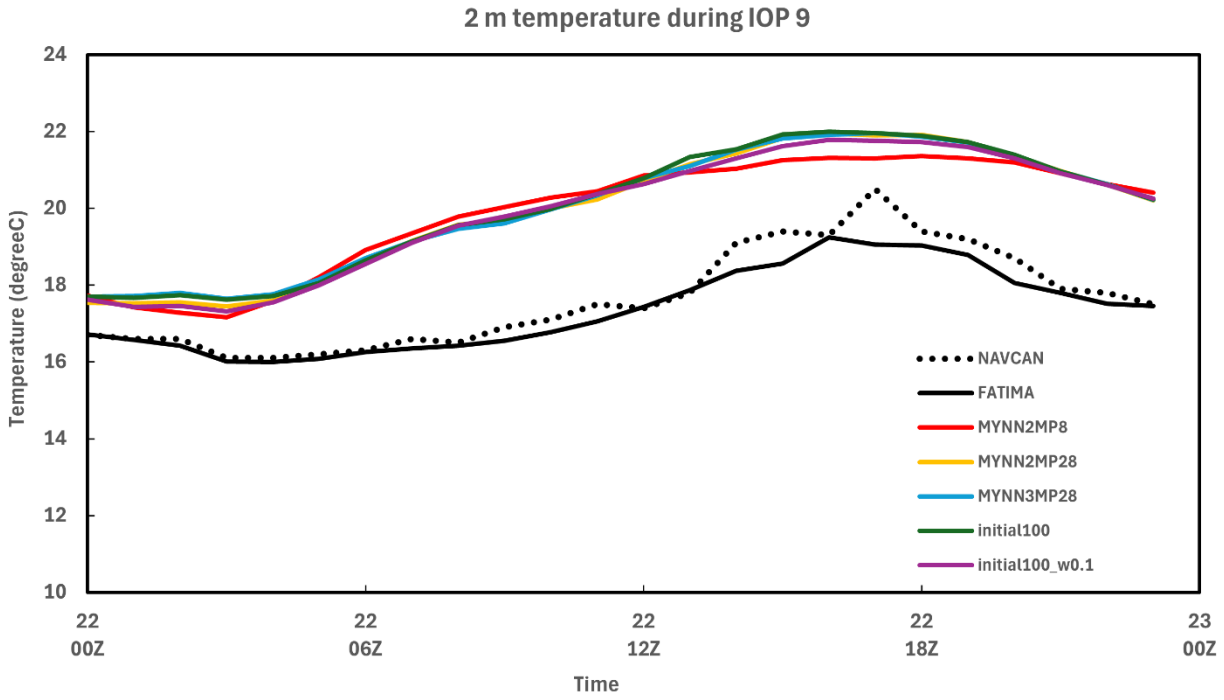


Figure 3.15 2 m temperature from the NAVCAN reports, field observations and five model runs during the period.

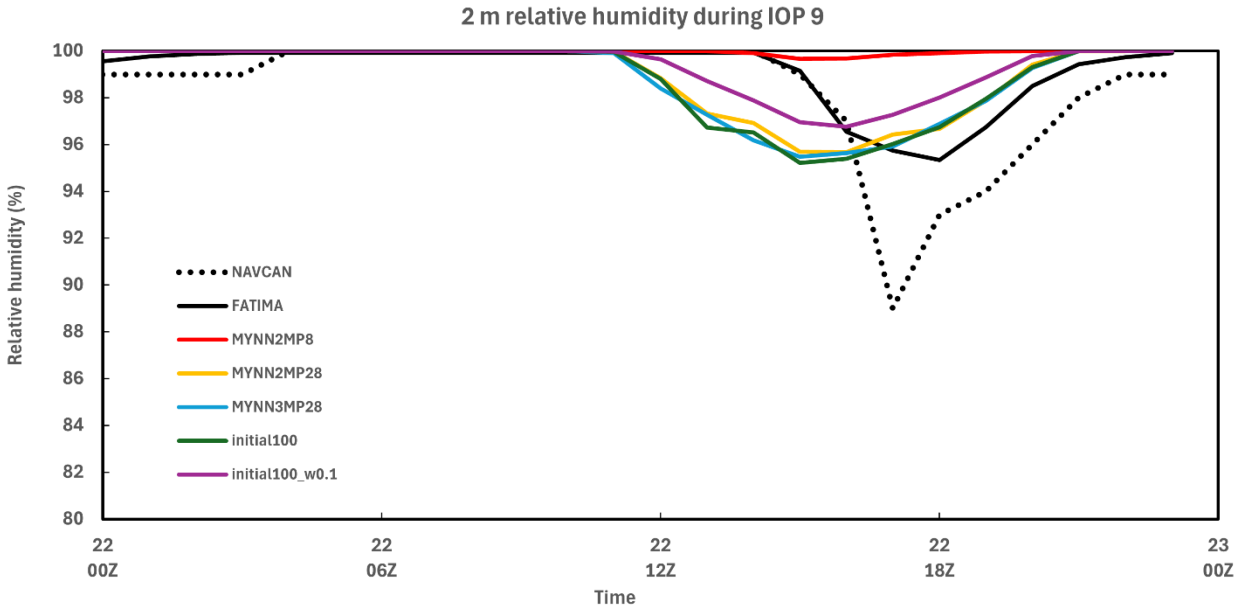


Figure 3.16 2 m relative humidity from the NAVCAN reports, field observations and five model runs during the period.

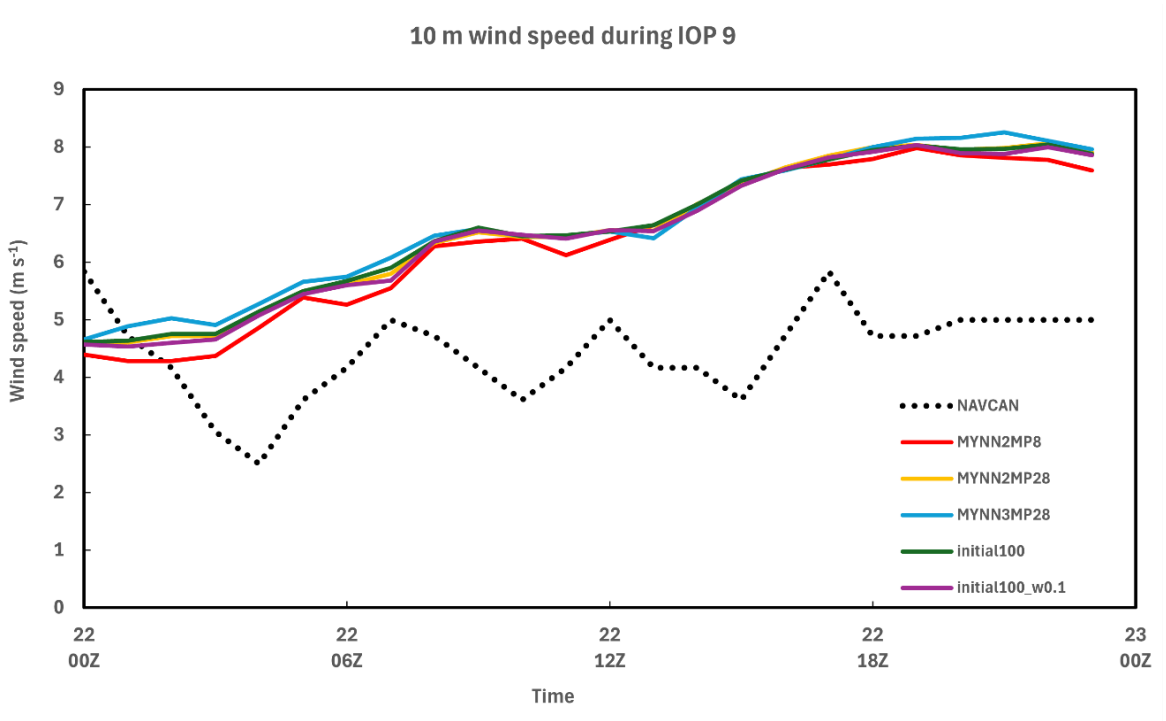


Figure 3.17 10 m wind speed from the NAVCAN reports and five model runs during the period.

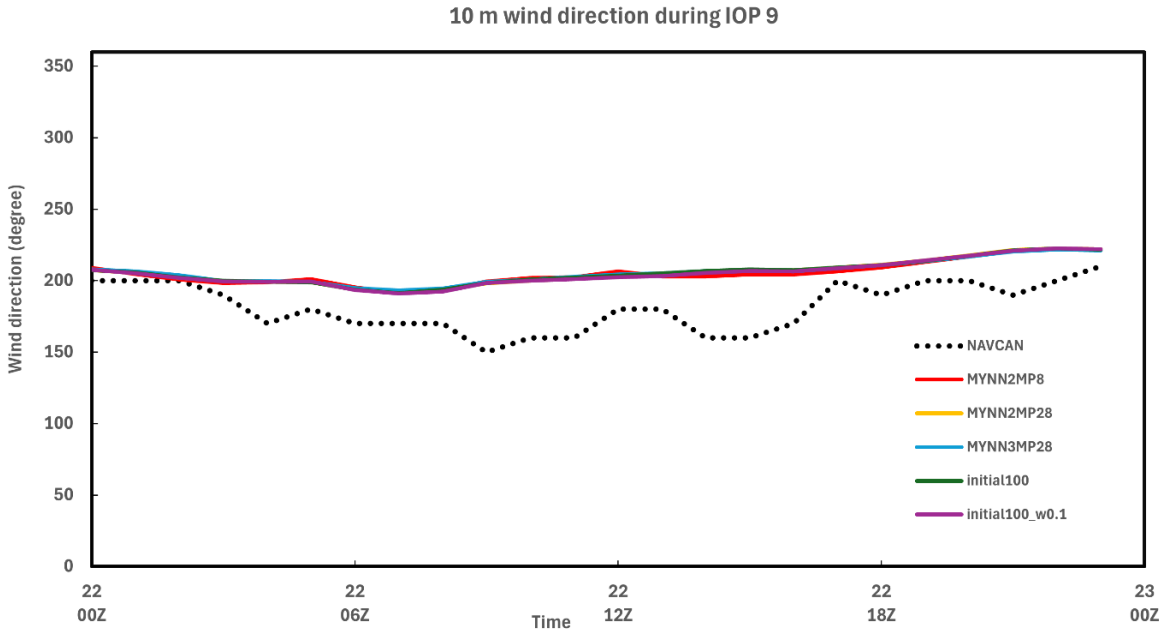


Figure 3.18 10 m wind direction from the NAVCAN reports and five model runs during the period.

Fig. 3.19 and Fig. 3.20 show the profiles of temperature and relative humidity at 12Z, July 22. Again, near the surface model temperature is higher than the observation, with 3 °C warmer. From Fig. 3.16 2 m RH is starting to drop in most model runs except MYNN2MP8, but all runs still manage to get 100% RH above that. All model runs can catch the fog top at about 300 m but there are differences above that. The reason why the model can still predict the fog top despite the higher near-surface temperature can be that the air is very stable due to the inversion, and therefore, the low updraft cannot transfer the heat from the surface to the fog to dissipate it.

Between 300 m to 1200 m, model runs except MYNN2MP8 are warmer than the sounding and thus resulting a lower RH in this level. At 500 m, MYNN2MP8 has a RH of about 90% while the other runs have about 85%, so although different from the sounding, the lower temperature in MYNN2MP8 at this level still has an impact on the RH. However, although the temperature in MYNN2MP8 is close to the sounding, its RH profile follows the trends of other model runs, so the air in the model is dryer at this level.

Between 1200 m and 1800 m, the model has a higher RH but the temperature is still close to the sounding, so the air is moister at this level. Above 1800 m, RH in the sounding starts to vary but the temperature is close to the model results.

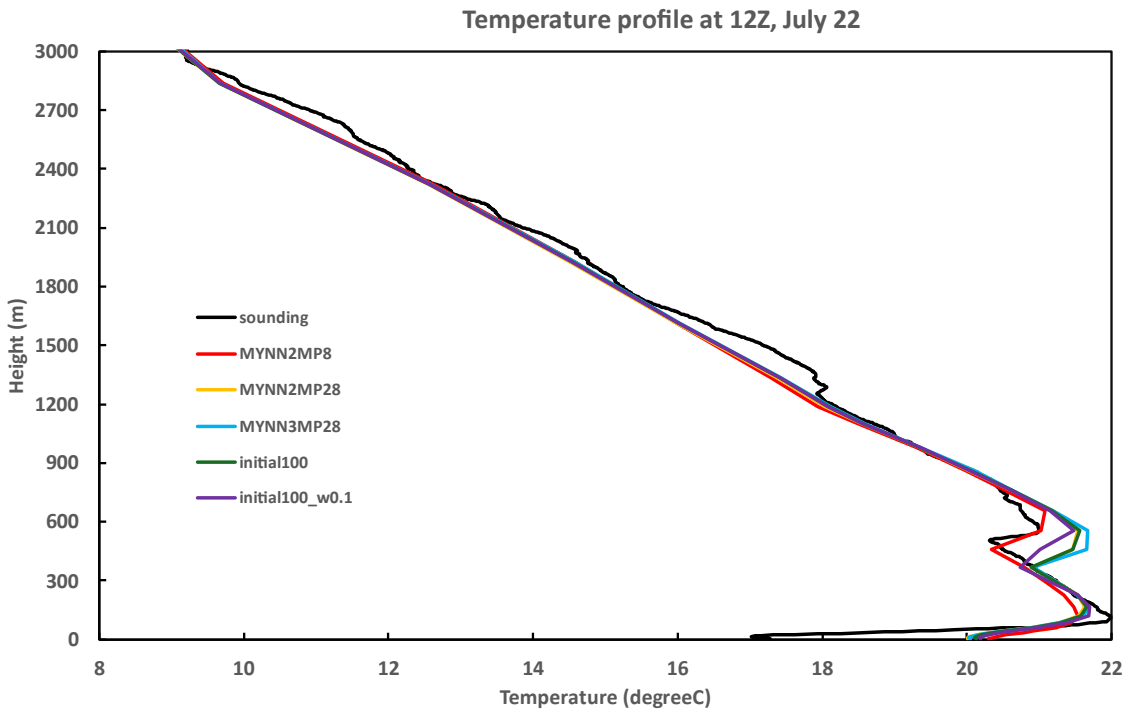


Figure 3.19 Temperature profiles from the weather balloon and five model run at 12Z, July 22.

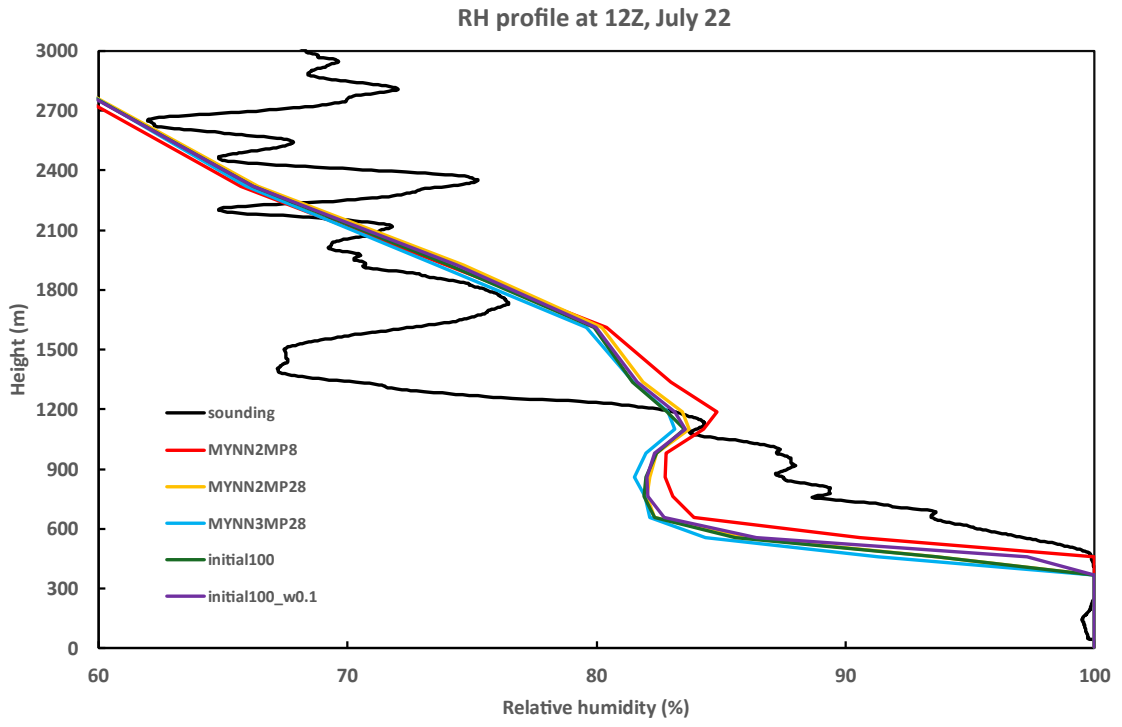


Figure 3.20 Relative humidity profiles from the weather balloon and five model runs at 12Z, July 22.

LWC and QNC

Fig. 3.21 and Fig. 3.22 show the LWC and QNC from different runs during IOP 9. According to the RH, the fog occurred at the beginning of the period and dissipated from 16Z, with a second event starting from 22Z. Compared to IOP 5, larger differences can be seen among different model runs. MYNN2MP8 has the highest amount of liquid water most of the time and unlike other runs, its fog does not dissipate in the second half of the period. The fixed value of droplet concentration of 100 cm^{-3} is higher than the observation.

With the droplet concentration being calculated instead of being a fixed value, other model runs have higher LWC and QNC than MYNN2MP8. Both QNC and LWC are much higher than the observation. In those runs the fog also dissipated with no liquid water near surface in the afternoon, but there should be liquid water according to the observation. There is little difference between MYNN2MP28 and MYNN3MP28 even though MYNN3MP28 is using a PBL scheme with higher moments to calculate more variables.

With the initial CCN being lower, initial100 gives lower LWC and QNC in the first half of the simulation, which is reasonable since initial100 only decreases the initial number. It shows that lowering the number of CCN can decrease LWC and QNC, because there are less CCN available to condense. Initial100_w0.1 has the most highest LWC and QNC in the first half of the simulation, which is consistent with IOP 5.

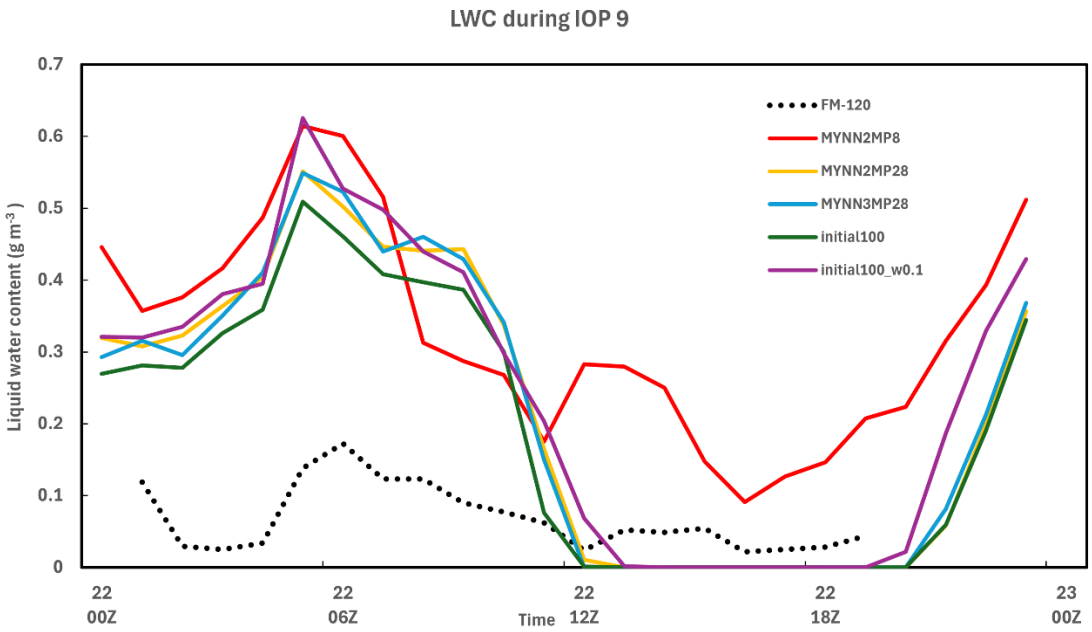


Figure 3.21 Liquid water content in g m^{-3} from FM-120 observation at 1.4 m and five model runs at 3.7 m during the period.

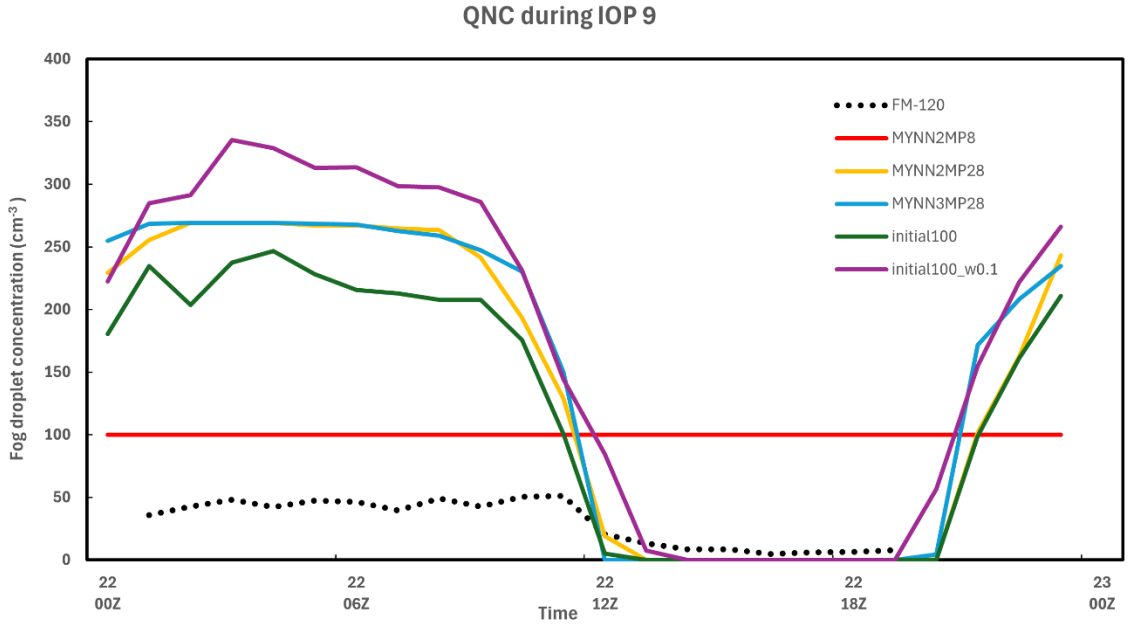


Figure 3.22 Droplet concentration in cm^{-3} from FM-120 observation at 1.4 m and five model runs at 3.7 m during the period.

3.6 Discussion

Results among different model runs

There are differences between different model runs, with the differences being smaller in IOP 5 and larger in IOP 9. MYNN2MP8 has a fixed QNC over time, whose values are half of the other runs calculating QNC, but the differences in LWC are not that high. In Fig. 3.21, MYNN2MP8 has the highest LWC in the beginning of IOP 9, 0.1 to 0.2 g m^{-3} higher than other runs, but it becomes lower than the others with a similar difference after 08Z, July 22. This is probably because in the scheme, they are treated separately. LWC in g m^{-3} presented in this chapter is converted from the cloud water mass mixing ratio (QCLOUD) in kg kg^{-1}, the actual variable used in the model. QCLOUD and QNC are independent. QNC determines the shape parameter μ in Equation 3.3, which then changes the size distribution in Equation 3.4. It does not affect QCLOUD directly. Both QCLOUD and QNC follow the same governing equation in Equation 3.5 independently, with only the size distribution involved in each of the governing equations. Therefore, a very high QNC does not have a very significant impact on LWC.

Between MYNN2MP28 and MYNN3MP28, the difference is very small. MYNN3MP28 calculates the variances and covariances of potential temperature and moisture, but it seems that the results are similar to the parameterized values used in the run with MYNN2.

Initial100 decreases the initial value of CCN concentration in the simulation, and it can be seen that in the first half of the two IOPs, its LWC and QNC values are on the lower end among all runs, which is more significant in IOP 9. The number of condensed droplets is determined by the

activation rate and the current CCN concentration. Indeed, reducing CCN concentration will also affect the activation rate since the activation rate is determined by 5 quantities including CCN concentration itself. The Thompson microphysics scheme allows the use of an input climate aerosol data. However, the test in Weston et al. (2022) showed that the result was not satisfactory. The Thompson microphysics scheme uses a fake source of CCN by continuously adding the value, which is the surface emission term in Equation 3.6, and it can also be a problem.

Initial100_w0.1 changes the minimum updraft speed to 0.1 m s^{-1}, which will increase the activation rate. In both IOPs, it has high LWC and QNC values. Weston et al. (2022) argued that the original value of 0.01 m s^{-1} is still too high for fog. They also tested with the updraft being 0.1 m s^{-1}. We have also tested with a lower value of 0.001 m s^{-1}, it did not give any differences. Because the activation rate is determined using a look-up table, it is possible that the value of 0.001 m s^{-1} is not included in the table so it is ignored. Therefore, a major code change will be needed if one wants to tune the scheme for better fog forecasting. Thompson and Eidhammer (2014) has acknowledged the limitation of the use of updraft speed in fog. In the comments in the code, it is proposed that they can use temperature and its tendency as proxy for updraft speed.

Another potential problem of the model can be the size distribution. A gamma distribution is widely used in microphysics schemes, but it may not be accurate. The size distribution of the droplets not only has an important effect of the radiation balance (Weston et al., 2022) but can also affect the gravitational settling of the droplets since larger droplets should fall faster than smaller droplets. In the Thompson microphysics scheme, gravitational settling is the only way to

remove liquid water droplets from the column without transforming to other forms of water. Taylor et al. (2021) argued that surface deposition of droplets due to turbulence should be included in the model.

The difference between the model and the observation

The most significant difference between the model and the observation is the temperature near surface. In both IOPs, it is found that the fog in the model has dissipated during daytime while it should have persisted according to the observation. 2 m temperature in the observation and the model are both above ground level, not above sea level so they are at the same elevation. Profiles show that the temperature difference becomes smaller aloft, and all model runs can predict the fog top well. Therefore, only the near-surface temperature is high, and it only dissipates the fog near-surface. From Fig. 3.23 one can see that there is fog in the surrounding sea area but not over the land. The downwind area also has less liquid water.

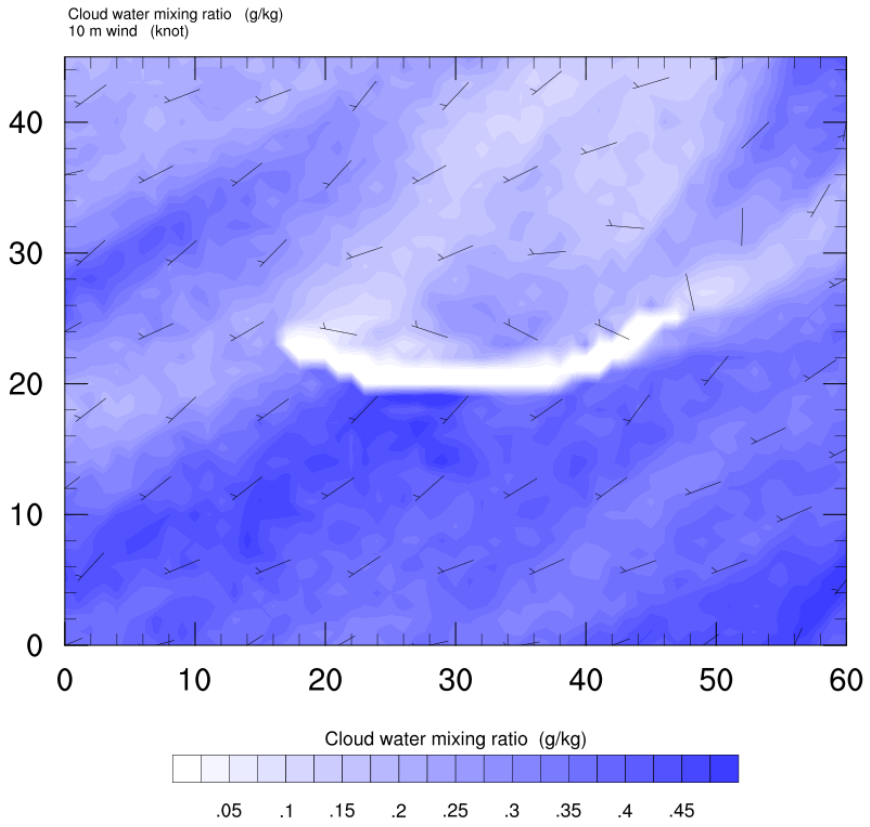


Figure 3.23 Cloud water mixing ratio at 3.7m and 10 m wind of MYNN2MP28 at 18Z, July 14 in domain 3, zoomed in to the area around Sable Island. Numbers on the axis are numbers of model grid points from lower left corner. Grid size is 1 km.

Figure 3.24 shows the skin temperature at 18Z, July 14, which is the temperature at the land and sea surface. They are the GFS data processed by the WRF Pre-processing System (WPS). The WPS program has three processes. First, the geogrid process will construct the domain according to the user's settings. Second, the ungrib process will extract the variables from the provided input dataset such as GFS. Last, the metgrid process will combine the user domain with the variables extracted, to produce input files for WRF.

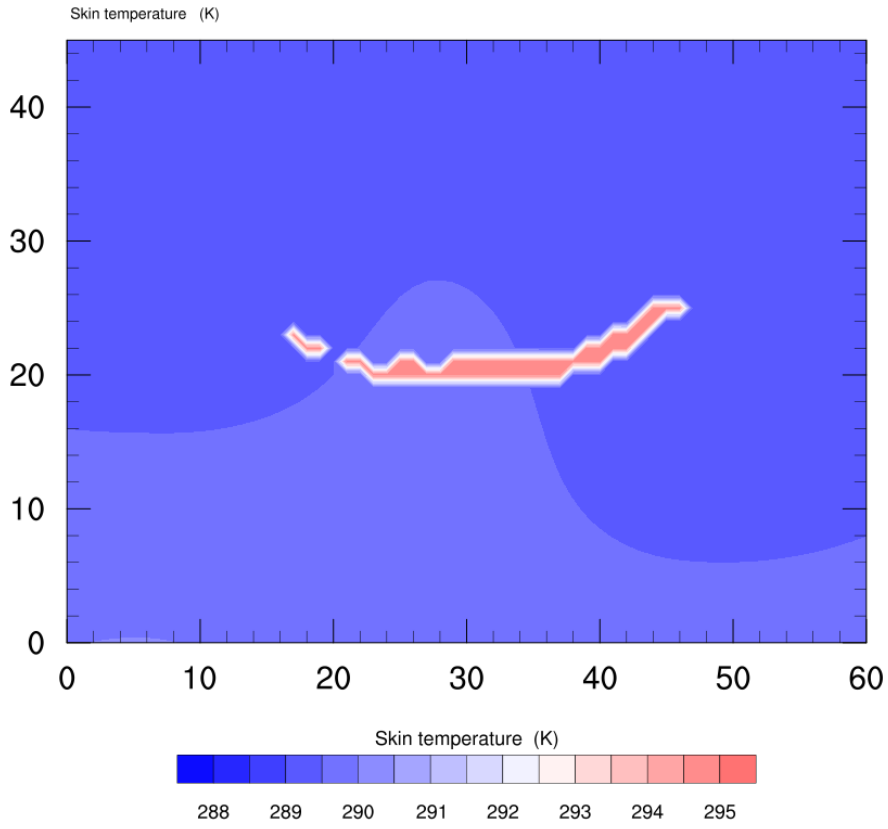


Figure 3.24 Skin temperature from the input file to WRF at 18Z, July 14 in domain 3, zoomed in to the area around Sable Island. Numbers on the axis are numbers of model grid points from lower left corner. Grid size is 1 km.

In domain 3 with a resolution of 1 km, the land of the island is recognized by the model with a higher skin temperature than the surrounding water. The input land skin temperature is 294 K, higher than the observed value at of 18 °C at 2 m. However, since Sable Island is a small island, the GFS dataset with a resolution of 0.25˚ x 0.25° does not contain the skin temperature of it. Therefore, the land is added by WPS in the first process when using a high resolution, then in the third process WPS assigns a skin temperature to it, which seems to be too high compared to the observation. Then this higher temperature burned off the fog in the model, while according to the

observation the fog did not completely dissipate.

This is similar to the “hot lakes” problem in one of the videos of the WRF User’s Tutorial by the National Center for Atmospheric Research (NCAR), “Advance Usage of the WPS” (Duda 2021). For each model grid point, its meteorological variables are usually interpolated by WPS from up to 16 surrounding points in the dataset. However, inland water bodies are so small that their skin temperature values are not included in the SST dataset, therefore, their values are often extrapolated from the nearby oceans, which turns out to be too warm. In the case of Sable Island, all surrounding points are water surface whose skin temperature are not applied to the land, so the skin temperature of the land will be extrapolated from the nearest land, i.e., the main island of Nova Scotia which has a stronger diurnal cycle and higher daytime temperature.

Unfortunately, this seems difficult to be fundamentally fixed. Duda (2021) had to manually assign the previous 5-day average of the 2 m air temperature as the initial skin temperature for the lake. A similar idea can be applied to Sable Island. Due to the diurnal cycle, one may need to use the average of the previous days at the same hours. Another possibility is to use the surrounding water skin temperature for the land. According to Fig. 3.23, there are fog areas surrounding the island, so it can get more fog in the daytime. However, it will require further studies on how much improvement it can bring, and how different it will be from simply using a coarser resolution that does not recognize the land.

The results over water seem to be more accurate. Because at 18Z, July 14 which is the local afternoon, night microphysics of the satellite was not working, Fig. 3.25 shows the cloud water mixing ratio values at 06Z, July 14. There is white colour over Sable Island due to plotting but there should be high liquid water according to Fig. 3.12.

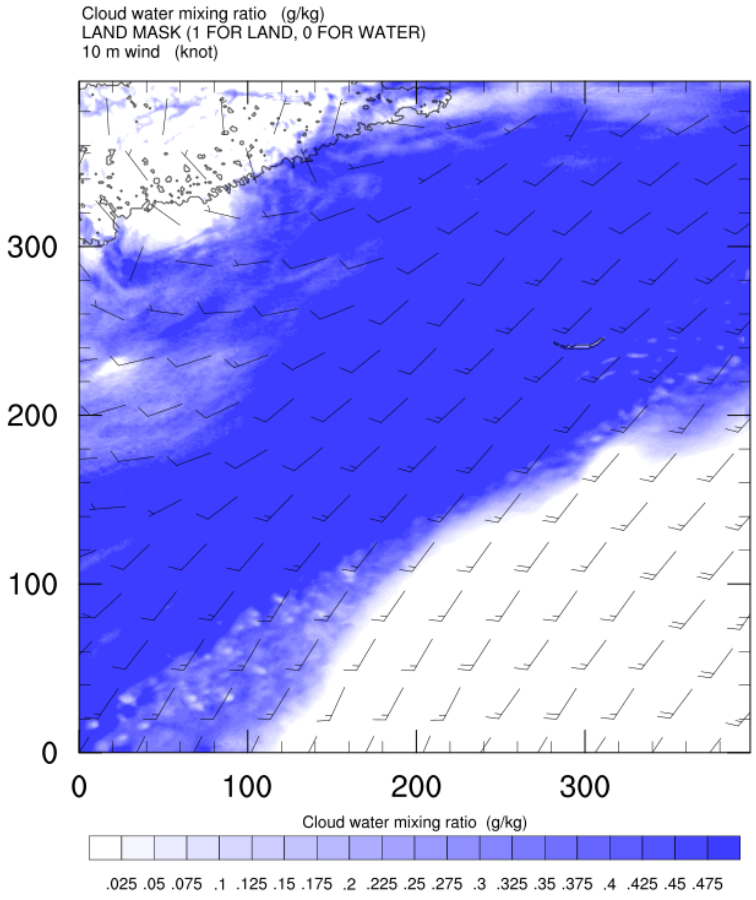


Figure 3.25 Cloud water mixing ratio at 3.7 m and 10 m wind of MYNN2MP28 at 06Z, July 14 in domain 3. Numbers on the axis are numbers of model grid points from lower left corner. Grid size is 1 km.

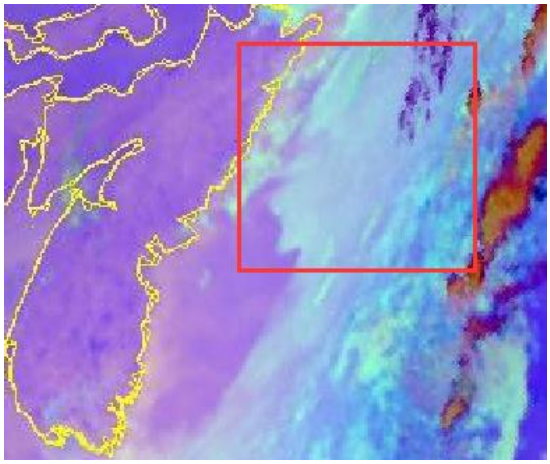


Figure 3.26 Night microphysics images from GOES-16 satellite at 06Z, July 14. Land mask is drawn with yellow lines. The bright blue (aqua) colour means fog. The brown to red colour on the means high, thick cloud. The approximate area of domain 3 in WRF is in the red square with the map projection being different.

Fig. 3.26 is a larger version of Fig. 3.5 at 06Z, July 14, at offshore NS. Compared to Fig. 3.25, the model is accurate in that there is fog on the coastline of NS. The bottom right corner in Fig. 3.25 shows a clear area. Because the map projection is different, that area should correspond to the top right corner in the red square in Fig. 3.26 with some purple colour meaning clear. That is not a perfect match because the purple colour is patchy while the clear area in Fig. 3.25 seems to be large, but it shows that the model predictions over water should be more accurate than the pointwise prediction on Sable Island.

This chapter has studied two IOPs during the FATIMA Sable Island campaign. By analyzing the LWC and QNC values, it has revealed some possible problems in the microphysics scheme, the

WRF model as well as the WPS program. Different model runs do not have fundamental improvements on the results. The problems are difficult to fix without any major code changes. In the next chapter, an ML model will be combined with the WRF model to try to improve fog forecasting.

Chapter 4

The Machine Learning Approach

4.1 Introduction

In recent years, machine learning has become a popular topic in various fields, including meteorology. GraphCast from Google (Lam et al., 2023) can predict hundreds of weather variables over 10 days at 0.25˚ resolution globally in under one minute and is said to "significantly outperforms the most accurate operational deterministic systems on 90% of 1380 verification targets, and its forecasts support better severe event prediction, including tropical cyclones, atmospheric rivers, and extreme temperatures". It uses the historical data from ECMWF Reanalysis v5 (ERA5, Hersbach et al., 2023) dataset in the years of 1979-2017 as training data. It takes the current weather state and the state 6 hours ago as input and predicts the weather state 6 hours ahead, then it can use its own prediction as input and predict further in time. These models have the potential to outperform current NWP models and have long-term forecasts for up to 10 days. However, the ERA5 dataset does not include cloud liquid water at a level but only the total column cloud liquid water. Since these deep learning models rely purely on machine learning, they cannot produce variables that do not exist in the training data, which makes them difficult to forecast. In addition, even if one wants to train a model to predict fog, the

training will be difficult, and the results will be unsatisfactory without a large amount of observation data

With fewer data available, one option is to nowcast. Gultepe et al. (2024) have used ML to nowcast marine fog visibility 30 and 60 min ahead on the Grand Banks using FATIMA measurements. In our case, we want to study the power of ML making predictions 24 hours ahead. The previous chapters have analyzed the WRF model and pointed out some of its problems, therefore, we try to combine ML with the prediction from WRF to see if the performance can be improved. The Python code used in this study is attached in Appendix C.

4.2 Training Data

In this study, we will use an ML model to perform a binary classification prediction between fog and clear situations. Regression which can directly predict the value of visibility is not used, because the visibility value of fog is extreme. The visibility can range from 0 to 24.1 km (maximum value of NAVCAN report) but only the fog periods with visibility $\leq$ 1 km are of interest. An ML model can learn the trend of rising or dropping of visibility, but predicting the extreme visibility values is more difficult (Lam et al., 2023). Therefore, only predicting the situation of either fog or clear is easier and more appropriate.

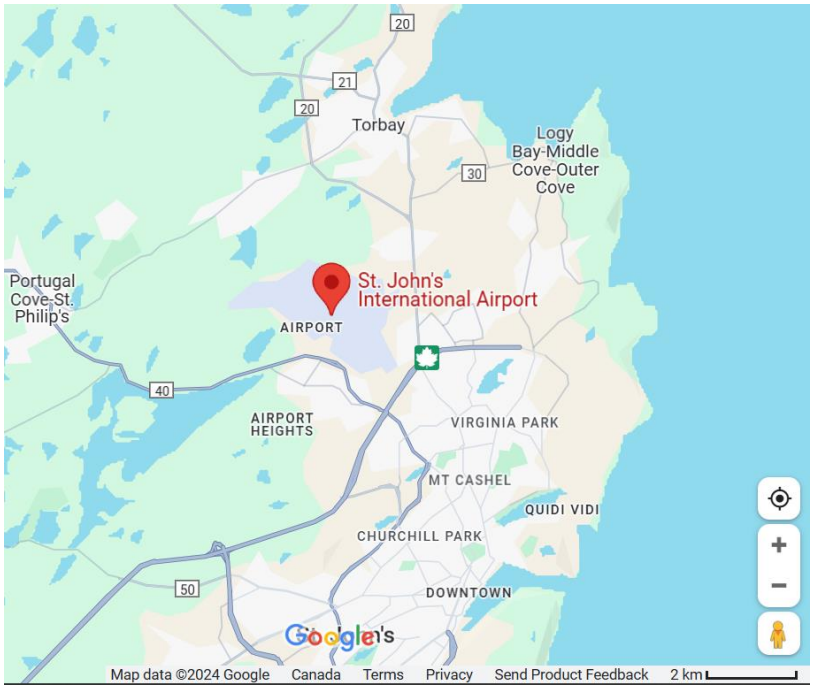


Figure 4.1 Map of St. John's showing the location of St. John's International Airport (from Google Maps).

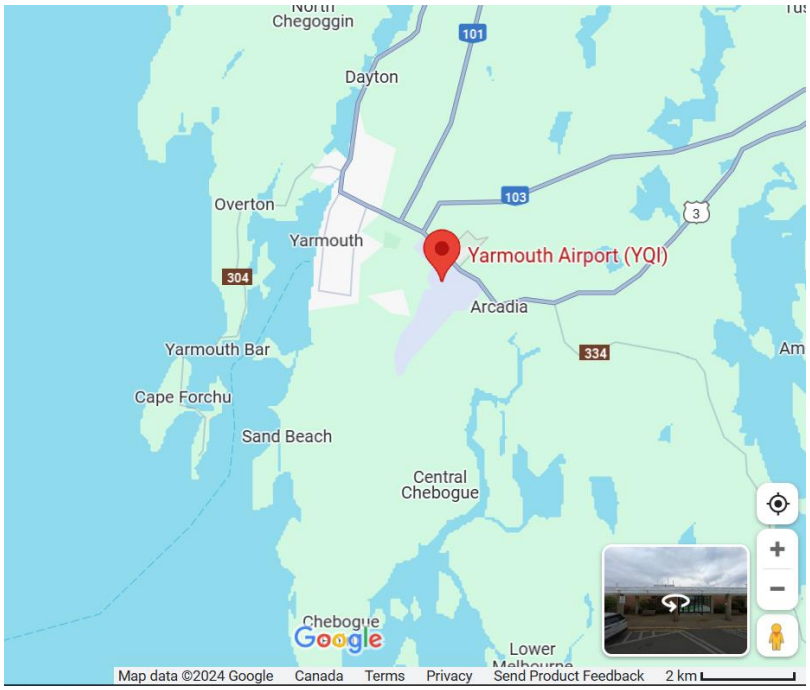


Figure 4.2 Map of Yarmouth showing the location of Yarmouth Airport (from Google Maps).

The chosen locations are St. John's, NL and Yarmouth, NS, which are at the east tip of NL and the south-west tip of NS, respectively. Visibility measurements are collected by NAVCAN at the corresponding airports. Fig. 4.1 and Fig. 4.2 show the map of the cities. Both airports are relatively close to the sea. Unfortunately, since the visibility report on Sable Island is incomplete with a lot of blank periods, Sable Island is not chosen for this ML study. The period of historical data is from 2012 to 2023, because 2012 is the earliest year with available visibility reports. Only the 5 months from April to August are used since before April blowing snow can also affect reported visibility, and there are less fogs after August. Fig. 4.3 and Fig. 4.4 show the hours of fog in each month. St. John's has more fogs in April and May while Yarmouth has more fogs in July and August. Since the two locations have different properties, they must be trained separately with their own data, meaning that there will be two independent models. Model performance will be poor if it is applied to other locations e.g., Sable Island.

As shown in Fig. 4.5 and Fig. 4.6, NAVCAN has never reported a visibility of 1.4 km, so it could be a clear border to distinguish between fog and clear. The visibility measurements are made by human observers looking at objects at a distance, which can be different for daytime and nighttime. At nighttime, background light will be an important factor. However, only fog or clear is of concern in this study and human observers should be reliable for distinguishing between the two. Therefore, it is decided that the "fog" class is when visibility is <= 1.2 km and "clear" class is when visibility is > 1.2 km, instead of the formal definition of 1 km. Tests using 1 km to distinguish between fog and clear have been done and they were less accurate. According to the report, rain alone does not reduce the reported visibility to below 1.2 km, and there are a lot of reports of rain combined with fog. Therefore, this study does not exclude any rain conditions.

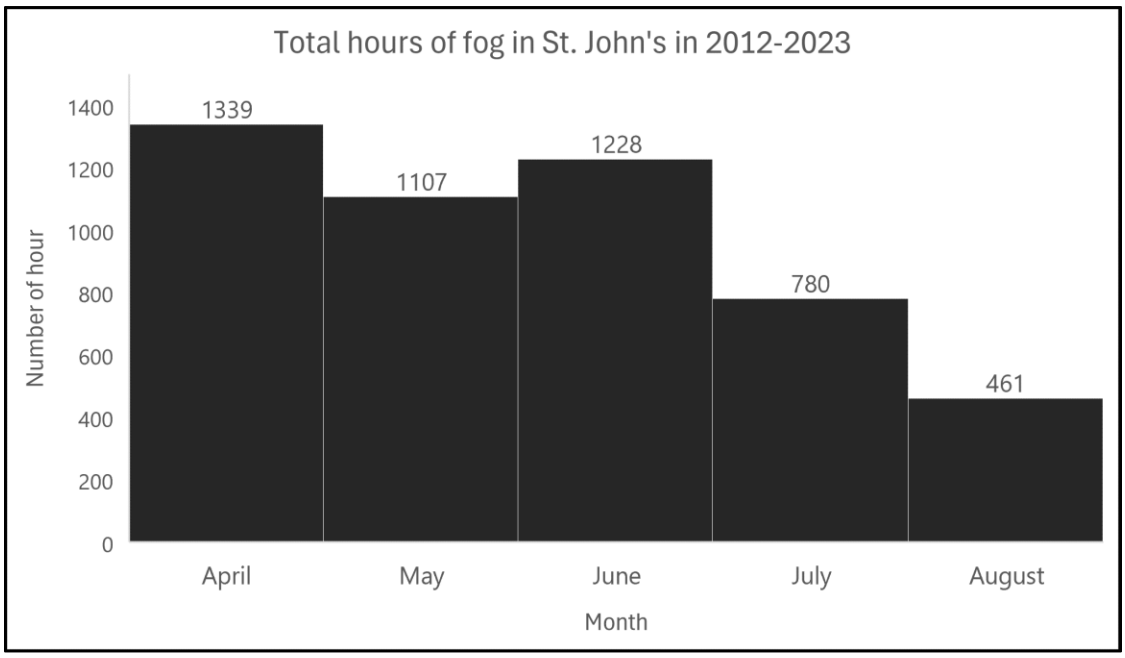


Figure 4.3 Hours of fog (with visibility <= 1.2 km) in April-August in St. John's. It has more fog than Yarmouth in April and May.

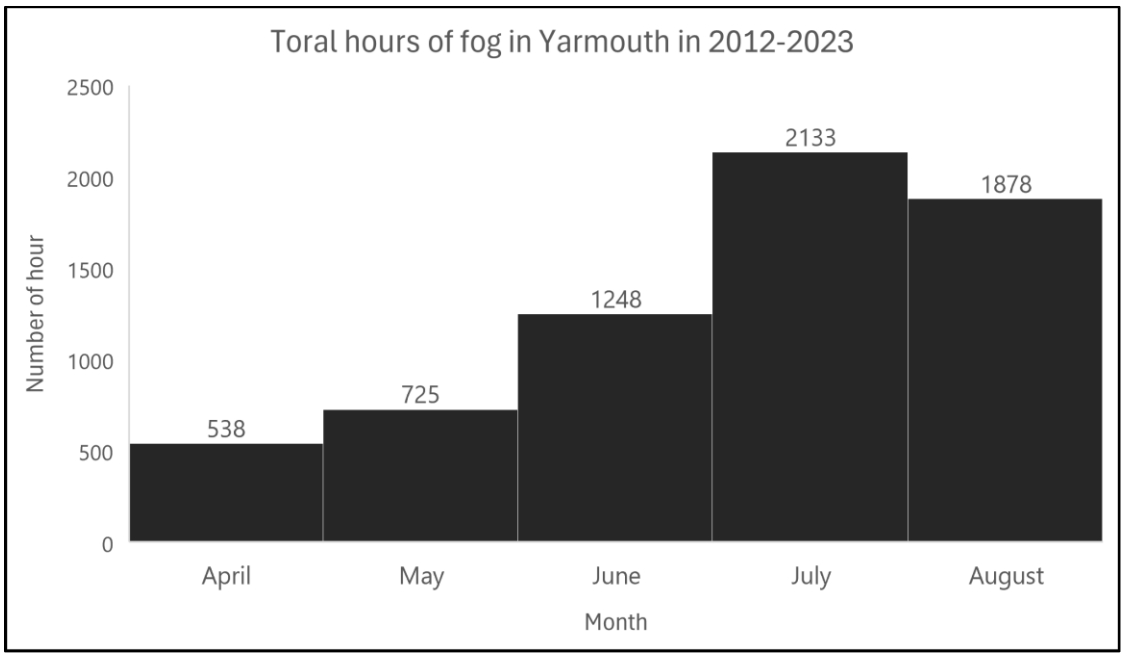


Figure 4.4 Hours of fog (with visibility <= 1.2 km) in April-August in Yarmouth. It has more fog than St. John's in July and August.

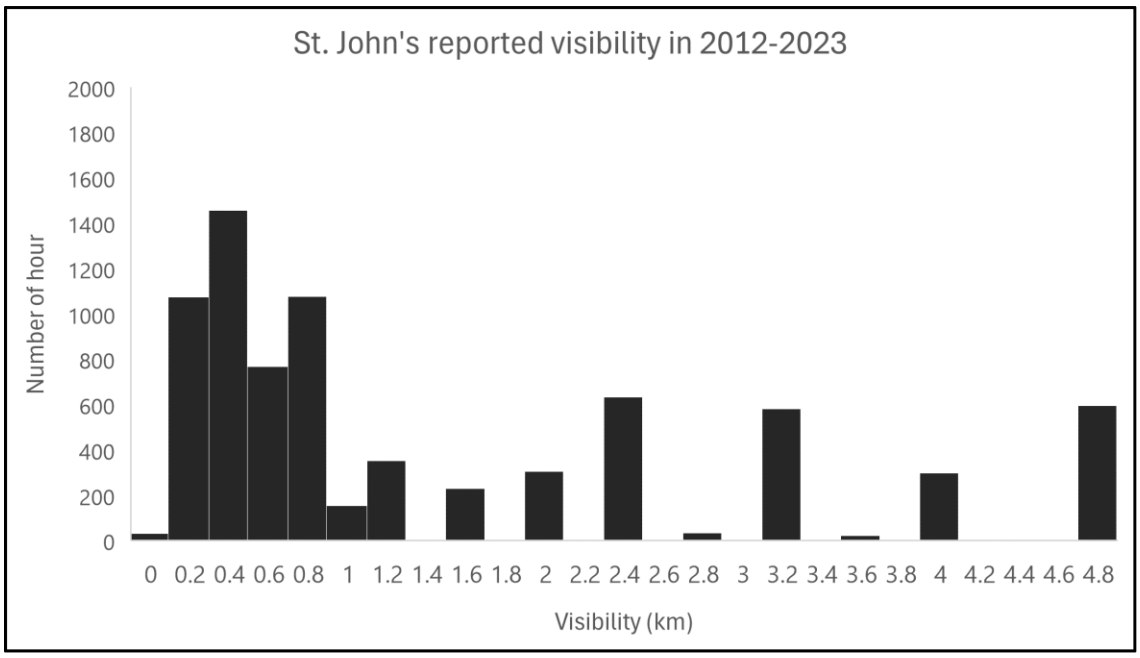


Figure 4.5 Histogram of reported visibility in St. John's in the year 2012-2023, from April to August each year. Only values lower than 5 km are shown.

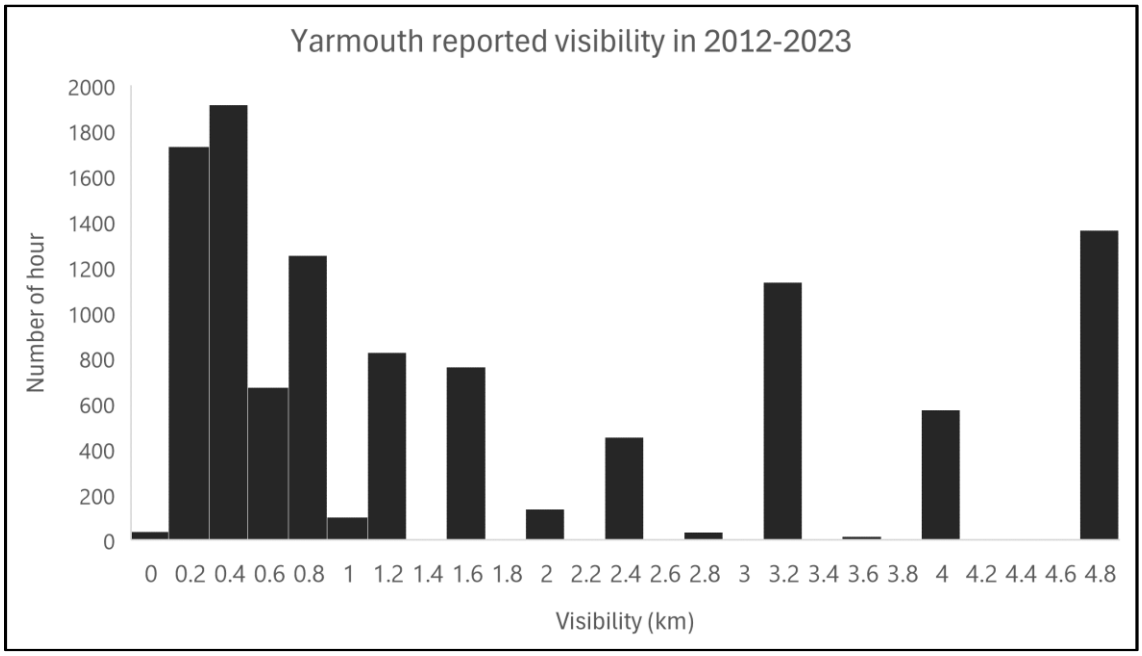


Figure 4.6 Histogram of reported visibility in Yarmouth in the year 2012-2023, from April to August each year. Only values lower than 5 km are shown.

The features used are 2 m temperature, 2 m relative humidity, 10 m U wind, 10 m V wind and land surface pressure, as well as month, day in the month and hour. Day of year is not tested but also a possible option. For each of the meteorological variables, its value one hour before the predict time (lag 1) is also used as a feature. It is possible since our goal is to use the 24-hour forecast of the variables from WRF, the lag 1 variables also come from WRF forecasts. To summarize, the features include 5 meteorological variables at the forecast time, 5 meteorological variables 1 hour before the forecast time and 3 variables of time in UTC. In the training process, the meteorological variables are extracted from ERA5 historical data that are downscaled by WPS at resolution of 9 km. The horizontal domain setting of WPS is shown in Fig. 4.7. The same domains will be used for WRF simulations later. Since the two locations are at a larger land where there exists surrounding land skin temperature in the ERA5 dataset, their values should be interpolated by WPS successfully.

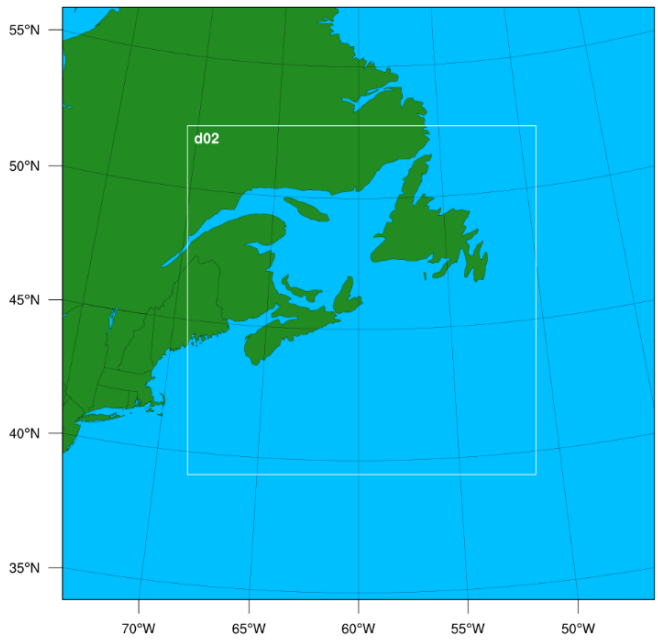


Figure 4.7 Domain setting for WPS to downscale the ERA5 data. The resolutions are 27 km for domain 1 and 9 km for domain 2.

A correlation heatmap of both locations is shown in Fig. 4.8, whose values by default are the Peason correlation coefficients (Pandas 2024). For any variable X and Y, it is calculated as

$$\rho_{X,Y} = \frac{cov(X,Y)}{\sigma_X \sigma_Y} = \frac{E[(X-\bar{X})(Y-\bar{Y})]}{\sigma_X \sigma_Y} \tag{4.1}$$

where $cov(X,Y)$ is the covariance, E is the expectation, $\bar{X}$ and $\bar{Y}$ are the means, and σ_X and σ_Y are the standard deviations of X and Y, respectively. This correlation can only show how strong the linear relationship is between 2 variables, while an ML model is capable to handle non-linear relationships. Therefore, although some variables have low correlations they are still used. The two locations show some similarities and some differences. Lag 1 variables always have high correlation as the regular variables since their values do not change rapidly in one hour. T2 (2 m temperature) has a low correlation with Hour. One reason can be that their relationship is not linear, in UTC 0 and 23 are not the coldest and hottest time in a day. Another reason can be that the dataset includes the temperature from April to August. The change of T2 in one day is small in the 5 months of data that are used.

The correlation between visibility and other variables are of interest. T2 has positive correlation with visibility in St. John's, so a higher temperature will result in higher visibility (= less fog), but the correlation between the two is negative in Yarmouth. This can be explained by Fig. 4.7 and Fig. 4.8 where St. John's has less fog in hotter months but Yarmouth has more, since T2 has high correlation with Month. Also, U wind has very low correlation with visibility in Yarmouth, but it is high in St. John's, so the direction of wind is important.

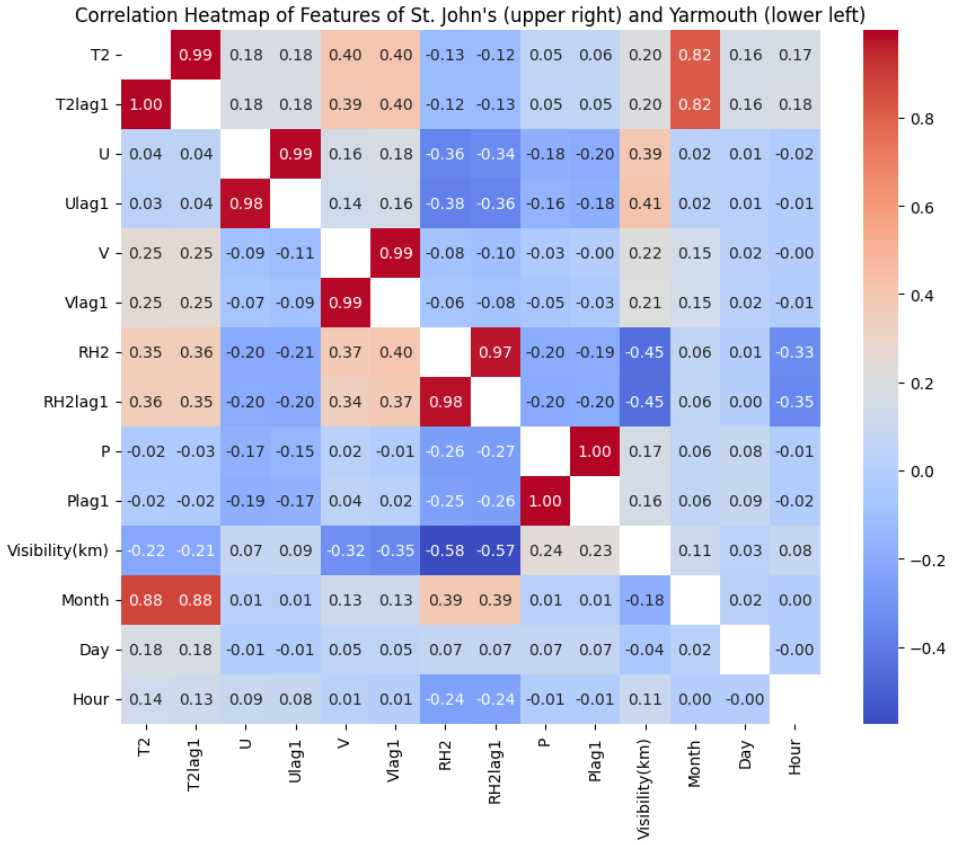


Figure 4.8 Correlation heatmap of the features and observed visibility in km from training data of both locations in 2012-2023. Data of St. John's are shown in upper right and Yarmouth in lower left. Values in the diagonal is hidden because the correlation with itself is always equal to 1.

4.3 Model Building

4.3.1 Model Description

The ML model used in this study is XGBoost (eXtreme Gradient Boosting, Chen and Guestrin, 2016). It has gained popularity from winning data science competitions and being a fast algorithm. It is a variant of the gradient boosting algorithm which is based on decision trees. A decision tree is a tree-like model of decisions and their consequence (Hsieh 2023). Fig. 4.9 shows

a simple example of a decision tree. Once a tree is constructed, one can use the data of RH and T to know the condensation situation by simply following the tree. Each block is called a node. The first node “RH=100%?” is a root node. When a node is split into further nodes, it becomes a parent of the new nodes. The node “T > 0?” is the parent node of “water” and “ice”. The nodes “water” and “ice” are the children of “T>0?”. When a node has no children, it is called a leaf, so “water”, “ice”, and “no condensation” are leaves. The tree in Fig. 4.9 also has a depth of 2, with the root node not being counted. The flow chart structure makes decision trees highly interpretable.

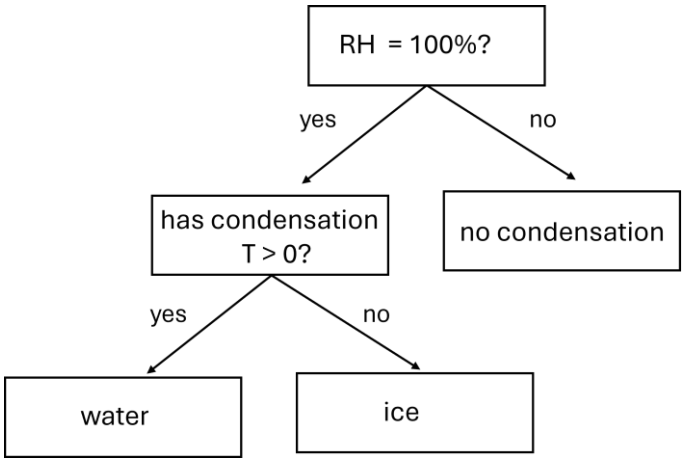


Figure 4.9 A simple example of a decision tree. “water”, “ice” and “no condensation” are leaves.

In machine learning, the algorithm CART (classification and regression trees) is used to construct trees. The goal of the tree is to minimize an objective function. In the case of XGBoost, the objective function is expressed as $L = l + \Omega$ where l is the loss function and Ω is the regularization term (Chen and Guestrin, 2016). The loss function is the sum of the differences between each provided actual value and model prediction. A lower loss means better model

predictions and thus better model performance. Different problems use different calculation of loss function. For example, a regression problem can use the simple mean square error as the loss function. The binary_logistic objective in XGBoost is chosen, which uses the logistic regression method for classification. The loss function, logistic loss of the model is:

$$l = \sum_i -y_i \ln(p_i) - (1 - y_i) \ln(1 - p_i) \tag{4.2}$$

where y_i and p_i are the true and predicted value of the ith data point, respectively (XGBoost developers 2022). In the case of binary classification, y_i as the provided target, is either 0 or 1, with 0 called a negative incidence and 1 called a positive incidence. The model prediction p_i is a probability between 0 and 1. Each raw model prediction $\hat{y}_i$ defined in Equation 4.5 will be transformed into the probability of the positive class between 0 and 1, using a sigmoid function $p_i = \frac{1}{1+e^{-\hat{y}_i}}$. With a perfect prediction where y_i and p_i are both 1 or they are both 0, the loss is 0. The loss of the model is the sum of the loss calculated for each data point. Fig. 4.13 will show the values of the loss function in this study.

If only the loss function is used, ideally the tree can grow aggressively until every single leaf contains only one class to reach 0 loss. This can lead to overfitting because a perfect tree for the training data is highly unreliable for new data. Therefore, the regularization term Ω is used to balance the fitting and growing. From (Chen and Guestrin, 2016), it is expressed as:

$$\Omega = \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2 + \alpha \sum_{j=1}^{T} |w_j| \tag{4.3}$$

where T is the number of leaves and w_j is the weight of the jth leaf defined in Equation 4.4. Values of them are changing during the training process so they are called parameters. On the other hand, Gamma (γ) is the minimum loss reduction required to make a further split. Lambda (λ) is called the L2 regularization term on weights. The third term alpha (α) is not mentioned in Chen and Guestrin (2016) but in the code, which is called the L1 regularization term on weights and it works in a similar way as lambda. These are the some of the hyperparameters that are set by the users and are constant during the training process.

Every leaf in the tree has a weight, which is calculated as:

$$w_j = -\frac{\sum_1^i g_i}{\sum_1^i h_i + \lambda} \tag{4.4}$$

where i is the number of data points in the leaf, g_i (gradient) and h_i (hessian) are the first and second order derivatives of the loss function l (Equation 4.2 without the summation) with respect to model output p_i of the ith data point in the leaf. Therefore, the weight of a leaf is determined by the number of data points in the leaf as well as the loss of its data points. It is used to see how well the leaf is classifying the data points and if a further split is needed.

However, decision trees are considered as “weak learners”, relative to “strong learners” which are more complex and more accurate models such as neural networks (Hsieh 2023). Methods are developed to combine the weak learners for more complex problems. Random forest constructs many trees as an ensemble, which has been proven to be able to compete well against strong learners. On the other hand, another approach called boosting constructs one tree at a time. Fig.

4.10 shows an example of boosting. The goal is to label the blue cross signs and the red plus signs. In every round, it will construct one decision tree. Member 1 makes a separation, but there are two incorrect points. Therefore, member 2 focuses on fixing the error of member 1, but it introduces new incorrect points. Member 3 will try to fix the error of member 2. Finally, by combing all 3 members with majority voting, all points are correctly labelled (Hsieh 2023).

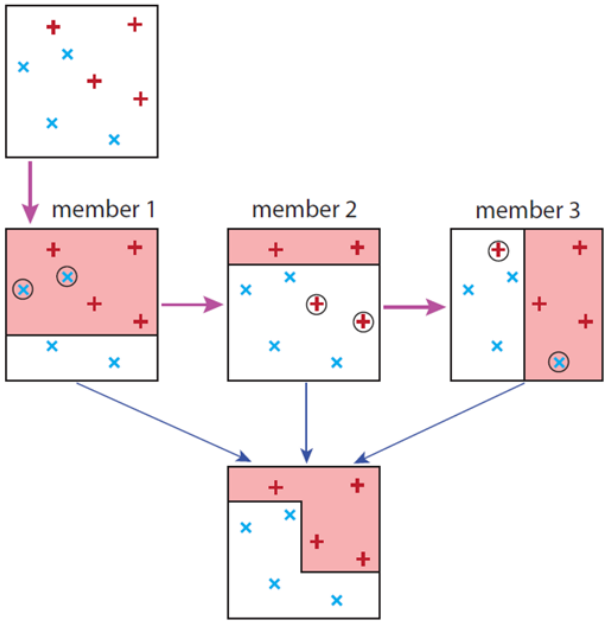


Figure 4.10 An example of boosting (from Hsieh 2023).

Gradient Boosting is an improvement from boosting, by using gradient descent to upgrade the solution from the previous member. XGBoost is a variant of gradient boosting with some unique features. For example, it can randomly select a subset of predictors to prevent overfitting (Hsieh 2023). Mathematically, the gradient process of XGBoost can be expressed as the following (XGBoost developers 2022).

$$\hat{y}_i^{(0)} = 0$$

$$\hat{y}_i^{(1)} = f_1(x_i) = \hat{y}_i^{(0)} + f_1(x_i)$$

$$\hat{y}_i^{(2)} = f_1(x_i) + f_2(x_i) = \hat{y}_i^{(1)} + f_2(x_i)$$

...

$$\hat{y}_i^{(t)} = \sum_{k=1}^{t} f_k(x_i) = \hat{y}_i^{(t-1)} + f_t(x_i) \tag{4.5}$$

where $\hat{y}_i^{(t)}$ is the model prediction at iteration t, $f_t(x_i)$ is the output of tree f_t with the input x_i at iteration t. Therefore, the prediction at any iteration is the summation of all previous iteration. The problem is to find the tree f_t, which should decrease the objective function from the last iteration. Therefore, from Chen and Guestrin (2016) an objective function at iteration t is:

$$L^{(t)} = \sum_{i=1}^{n} l\left(y_i, \hat{y}_i^{(t-1)} + f_t\right) + \Omega(f_t) \tag{4.6}$$

Since the fog case is much rarer than the clear case, a separate hyperparameter called scale_pos_weight is used, whose value should be set as (# of negative instances)/ (# of positive instances). In our case the class 0 is clear and the class 1 is fog, based on visibility <= 1.2 km. The corresponding value of scale_pos_weight is 7.97 for St. John's and 5.75 for Yarmouth based on data from 2012 to 2023. When the model has misclassified a fog event as clear, the loss of this data point (Equation 4.2 without the summation) will be multiplied by the value of scale_pos_weight. It can force the model to focus more on the rarer positive (fog)

instances, instead of getting low values of loss from the large number of true negatives.

4.3.2 Cross Validation

Cross validation (CV) is a technique to use the entire dataset to validate the model when data is not plentiful (Hsieh 2023). It divides the data into a number of approximately equal segments. One segment is reserved as "validation data", while the others are used for model training. The process will repeat until all segments have been used as validation data once. Each round a validation performance score is calculated, and a final score is usually the average of all rounds. This technique is used to tune the hyperparameters of the model, including γ, λ and α mentioned in the last section. A number of model runs is made with different values of hyperparameters, then each run will have a performance score. The run with the best score is the one with the optimal hyperparameters.

Different split methods can be used for CV. However, since weather data is in temporal order and the variables are autocorrelated, the ML model must take this into consideration during the process, otherwise the validation score will be overestimated while the model can still perform poorly in operational applications. Vorndran et al. (2022) have tried different split methods and pointed out the pitfalls in the training and evaluation of fog forecasting with ML algorithms.

StratifiedShuffleSplit (SSS) is a popular split method for unbalanced data where the numbers of data in each class are very different. Data points are randomly put into segments, while the ratio

between different classes is kept the same. In this case, between fog and clear. This method avoids the situation where the model does not see any data of rarer class of fog. However, Vorndran et al. (2022) has shown that this method will lead to an overestimated validation score.

TimeSeriesSplit (TSS) is a method where newer data come in later rounds. The dataset is divided into segments according to time. In the first round, only the first two segments are used, with the first one being the training set and the second one being the validation set. In the second round, the first two segments will work as the training set and the third segments is the validation set. This method ensures that the model can use newer data to validate the training with older data, which is more natural when dealing with a meteorological dataset.

The two split methods will be applied to the whole dataset twice. Firstly, it will be split into the CV set and the internal test set. The internal test set will be used later to validate the model after training, which will contain 8% of the total data set (1 year out of 12 years). Case 1 (SSS) randomly gets 8% of the data for the internal test set. Case 2 (TSS) uses the last year, 2023 as the internal test set. Since the data only covers the months from April to August, it is reasonable to use the data of one year as the test.

The split method is also applied during CV. Since there are 11 years of data in the CV set, it is easy to understand to use 1 year (9%) of the data (1 year out of 11 years) as validation. For case 1 with SSS, in each round 91% of the data is used for training and 9% of the data is used for validation, with the ratio between the classes being the same, and the split is randomly chosen. It

can have overlapping samples that are chosen for testing for more than once, however, it shouldn't be a problem with a large enough data set (Scikit-learn developers 2007-2025). It runs for 10 rounds. Fig. 4.11 shows an example of the split.

Case 2 splits the CV set into 11 segments, with each segments containing data of one year. The CV will have also 10 rounds to cover all segments. Fig. 4.12 shows an example of the CV with TSS.

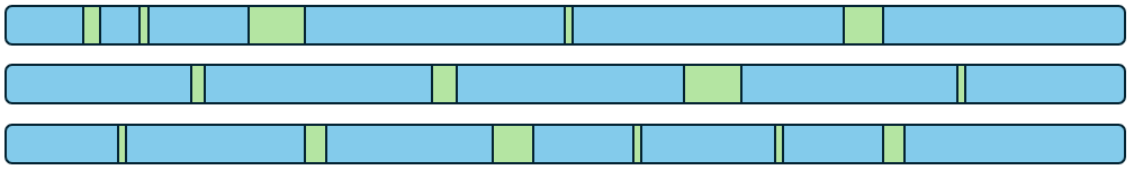


Figure 4.11 An example of CV using StratifiedShuffleSplit. Blue is training data and green is test data.

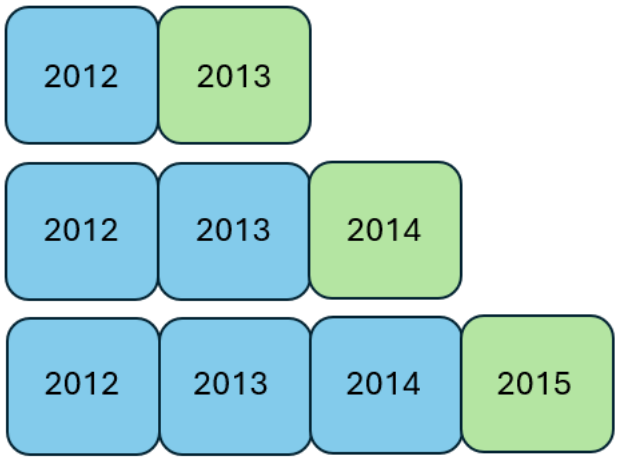


Figure 4.12 An example of CV using TimeSeriesSplit. Blue is training data and green is test data.

The grid search method is used to run the model for all combinations of hyperparameters in a given range. For each combination, the model will be trained and validated with the given training and validation splits. There are many hyperparameters in XGBoost, but those to be tuned here include max_depth, min_child_weight, subsample, lambda, alpha, and gamma. Max_depth and min_child_weight are the two main hyperparameters of XGBoost. Max_depth controls the maximum number of layers that a tree can have. A larger value will lead to a more complex model and more likely to overfit. On the other hand, min_child_weight controls the threshold determining if a leaf will split further or not, based on its weight defined in Equation 4.4. A smaller value will make the model easier to get deeper, and is more likely to lead to overfit.

The hyperparameter called subsample will let the model use a randomly chosen portion of the data set in one round, which is also a method to prevent overfitting. Lambda λ, alpha α and gamma γ are used in the regularization term in Equation 4.3, to further avoid overfitting. Increasing these three values will lead to a more conservative model. Since grid searching for all hyperparameters will cost a large amount of time, it is suggested that one can tune the hyperparameters in the order of: firstly max_depth and min_child_weight together, then gamma, followed by subsample, lastly lambda and alpha together (Jain 2025). Table 4.1 summarizes the hyperparameters used in this study.

Since we are more interested in predicting fog which has much fewer hours than the clear situation, the F1 score of fog is chosen as the evaluation score, so that the CV will choose the combination of hyperparameters with the highest F1 score of fog as the optimal setting. Table 4.2

summarizes the optimal hyperparameters acquired from CV.

Table 4.1 Summary of hyperparameters in this study.

Hyperparameter	**Description**
max_depth	Maximum layers of the trees.
min_child_weight	Minimum threshold for a leaf to make a split.
subsample	Fraction of the dataset to be used in one iteration.
lambda λ	L2 regularization term on weights in Equation 4.3.
alpha α	L1 regularization term on weights in Equation 4.3.
gamma γ	Minimum loss reduction required to make a further split in Equation 4.3.
scale_pos_weight	Control the balance of positive and negative classes. Not tuned in CV.

Table 4.2 Optimal hyperparameters acquired from CV.

St. John's						
	max_depth	min_child_weight	subsample	lambda	alpha	gamma
Case 1 (SSS)	12	1	0.7	4	1	1
Case 2 (TSS)	9	7	0.9	4	2	2
Yarmouth						
Case 1 (SSS)	13	2	0.6	3	0.4	0
Case 2 (TSS)	7	11	1	4.4	0.3	1.5

The absolute values of the numbers in Table 4.2 do not have significant meanings because different dataset will require different model structures. However, comparisons can be made between the two cases using the same dataset. For both locations, case 1 (SSS) has a larger max_depth and a smaller min_child_weight than case 2 (TSS) meaning that it is a more complex model, and more likely to overfit. In addition, in case 1 the three regularization terms alpha, lambda and gamma are either equal or lower than in case 2, with the only exception being that alpha for Yarmouth is 0.1 higher for case 1. Therefore, case 1 with StratifiedShuffleSplit is a model easier to overfit, according to the structure of the tree and the regularization terms. On the other hand, subsample in case 1 is lower than in case 2, which means that case 1 is trying to introduce more randomness in the training data to avoid overfitting.

4.3.3 Model Training

After the cross validation, the model will need to be trained with the optimal hyperparameters again with a full training dataset. In this section, the CV set is further split into 2 sets, the training set and the validation set. The validation set is used for early stopping, which is a technique to prevent overfitting. During training, in each round the model will try to get a lower objective function score (sum of l in Equation 4.2 and Ω in Equation 4.3) than the previous round. However, having a model that is trained very well to the training dataset only will lead to overfitting. Therefore, in each round the loss function is also calculated for the validation set. Once the score of the validation set has stopped decreasing for a few rounds, which means that the model cannot get a better score for a new dataset, the model will stop training before the specified number of rounds. For case 1 using SSS, the validation set is the shuffled stratified 9%

of the CV set (8% of the whole dataset), and for case 2 using TSS, the validation set is the most recent 9% of the CV set, i.e., data in 2022. Table 4.3 shows the split of the whole data set of this study.

Table 4.3 Summary of data splitting in this study. The training set and the validation set are used together for CV.

	training	**validation**	**internal test**
Case 1 (SSS)	shuffled 84%	shuffled 8%	shuffled 8%
Case 2 (TSS)	2012-2021	2022	2023

As mentioned above, the model will try to decrease the objective function during training, which however, is not highly interpretable. In order to monitor model performance, another function called evaluation metric, is computed and shown during training. This is only for the users and thus one can use any metric. In XGBoost, the default evaluation metric for classification is logloss, which is the logistic loss in Equation 4.2. By using this metric, the users can directly see the change of the loss function in the model. Error rate is one of the other metrics available which is simply the number of wrong predictions divided by the number of all predictions and it has a higher interpretability.

Fig. 4.13 shows the logistic loss against the round of training, for both locations in the two cases. In each round, the model constructs a new tree and separates the data in the training set only

based on the features. The model does not see the provided targets (fog condition) when separating the data. The loss is calculated with model predictions of fog condition and the provided targets. The loss of the model is decreasing as the training progresses, meaning that the model is improving its predictions on the training set. The validation set is used for early stopping mentioned above. The loss of the validation set also decreases, meaning that what the model has learnt from the training set at the current round can be applied to another dataset.

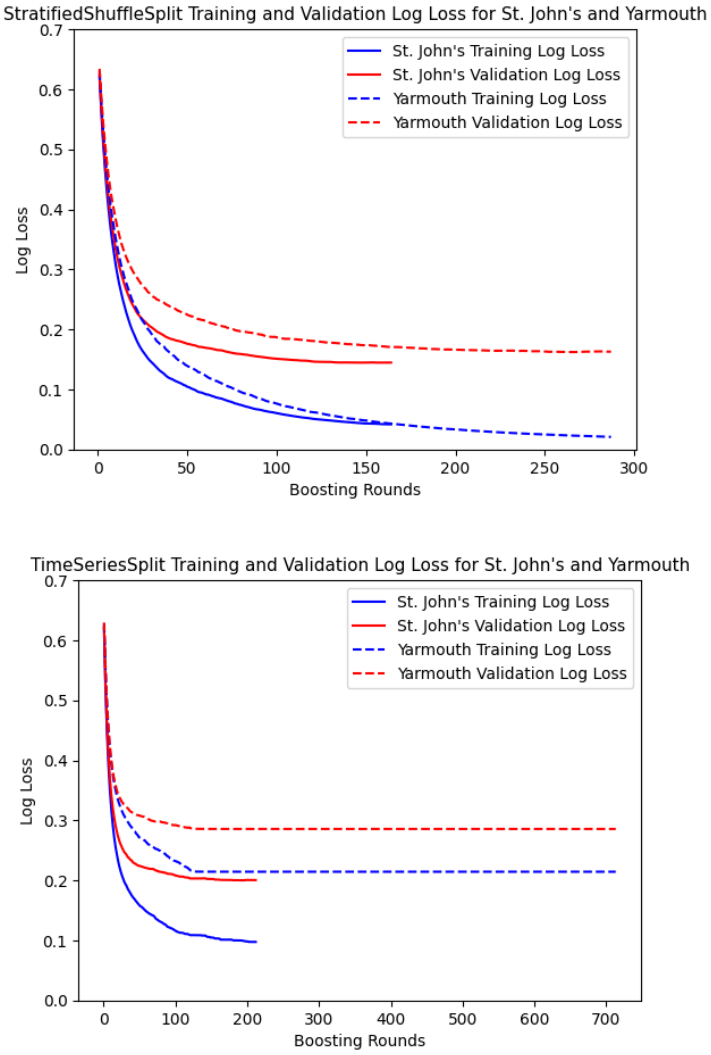


Figure 4.13 Logistic loss of training and validation against boosting rounds of case 2 for both locations.

In case 1, both locations have close rounds, and their final training loss is lower than in case 2. Especially for Yarmouth, the final training loss is 0.02 in case 2 with fewer rounds, much lower than 0.2 in case 1. St. John's has the training loss of 0.1 in case 1, but it is 0.03 in case 2. A lower loss means better predictions for that dataset. So far it seems like case 1 has a better fit for the dataset, and Yarmouth is more difficult to predict than St. John's.

4.3.4 Internal and External Tests

In Section 4.3.2, an internal test set has been split from the whole dataset. It will be used to validate model performance. After the model has been trained, all the features, 2m temperature, 2 m relative humidity, 10 m U wind, 10 m V wind, land surface pressure and their values 1 hour ago as well as month, day and hour, in the internal test set will be put into the model to get predictions which will then be compared to the actual values in the internal test set. Table 4.4 summarizes the scores for the two cases. Since the models are outputting continuous probabilities from 0 to 1, a threshold is needed to separate the two classes clear and fog. Because the internal test set is split from a known dataset, the thresholds in the table can be adjusted manually to get the best F1 scores. This is the process to tune the thresholds, and the same thresholds of each case will be applied to the external test with new data. Precision is the number of true positives divided by the sum of true positives and false positives. Recall is the number of true positives divided by the sum of true positives and false negatives. F1 score is the harmonic mean of precision and recall. Detailed explanations of precision, recall and F1 score are provided in Chapter 2.

Table 4.4 Scores of both cases in the internal test.

	St. John's				Yarmouth			
	precision	recall	F1	threshold	precision	recall	F1	threshold
Case 1 (SSS)	0.75	0.76	0.76	0.7	0.77	0.76	0.77	0.6
Case 2 (TSS)	0.75	0.72	0.73	0.8	0.64	0.67	0.65	0.8

When tuning the threshold to get the best F1 score, it will be the one balancing precision and recall since the F1 score is the harmonic mean of the two, which is the reason why the two are close in all cases. In case 1, the F1 scores between the two locations are very close, while in case 2, St. John's works much better than Yarmouth, which is consistent with the final loss in Fig. 4.13. Case 1 also has higher F1 score than case 2 for both locations. According to the scores, case 1 seems to perform better than case 2 again.

The internal test set is a part of the 2012-2023 ERA5 data. Although it can show the model performance when it sees new reanalysis data, we also want to see how it performs when used for prediction in an operational environment. Therefore, we test the model with new data. The WRF model has been run at 12Z every day for 36 hours long, then the hourly results of the last 24 hours are used. The importance of a spin-up time has been shown in Chapter 2. The initial and boundary conditions for WRF are from GFS forecasts issued at 12Z. The physical parameterization of WRF is the MYNN2MP28 case in the last chapter. Then the hourly 2 m temperature, 2 m relative humidity, 10 m U wind, 10 m V wind and land surface pressure, as well

as month, day and hour, are used as features to the ML model for hourly prediction of fog for the corresponding 24 hours. The period of test data is from April to August 2024.

This approach is essentially post-processing the WRF output. WRF will produce liquid water based on the numerical equations. On the other hand, the ML model will take the WRF prediction of the features and predict fog based on historical data. Therefore, it is not necessary to use WRF. For example, the GFS forecasts already include all the chosen features, but they are provided every 3 hours. Spatial resolution will probably matter, but the training data is provided with the ERA5 0.25˚ x 0.25˚ dataset so a high resolution may not make significant differences.

Table 4.5 Scores of both cases and WRF in the external test.

	St. John's			Yarmouth		
	precision	recall	F1	precision	recall	F1
Case 1 (SSS)	0.63	0.65	0.64	0.50	0.64	0.56
Case 2 (TSS)	0.62	0.69	0.66	0.48	0.63	0.55
WRF only	0.62	0.62	0.62	0.47	0.60	0.53

Table 4.5 shows the scores of both cases and WRF. WRF score is based on the prediction of liquid water in the lowest model level. It has class clear when there is no liquid water and class fog is when there is any amount. Surprisingly, the two cases have very close F1 scores, despite

their different results in the internal test. For all three scores, precision, recall and F1, St. John's is higher than Yarmouth, therefore, it shows that the forecast for Yarmouth is more difficult. In St. John's, precisions of case 1 (SSS), case 2 (TSS) and WRF only are very close. Case 1 (SSS) has a lower recall than case 2 (TSS), but the two XGBoost cases are higher than WRF. In Yarmouth, both cases have slightly higher precisions and recalls than WRF. As a result, in both locations both cases have marginally higher F1 scores than WRF.

The scores in the external test are lower than the internal test, because the external test is using WRF forecasts as input which can have more errors than the internal test using ERA5 data. However, case 1 (SSS) has an F1 score of 0.76 in the internal test for both locations, but in the external test, the scores are 0.65 for St. John's and 0.56 for Yarmouth, with a decrease by 0.12 and 0.20 respectively. Case 2 (TSS) has an F1 score of 0.73 for St. John's and 0.65 for Yarmouth, which deceases to 0.66 and 0.55 in the external test, with the differences being 0.07 and 0.10, respectively. This shows that case 1 (SSS) has overestimated its performance. It shows better results in the internal test but has the same level of performance as case 2 (TSS) when dealing with new data. The reason for the two cases performing similarly in the external test can be that this study itself is simple, with only 5 meteorological variables with their lag1 values used, and the time variables. Case 1 (SSS) is easier to overfit according to the hyperparameters and the training loss, therefore, if the study involves a more complex model with more features and more data, the model using StratifiedShuffleSplit can overfit more, and eventually perform worse than TimeSeriesSplit. Therefore, only case 2 (TSS) will be discussed further in the next section.

4.4 Result Interpretation

Table 4.6 Confusion matrices of both locations from case 2 and WRF in the external test.

		St. John's				Yarmouth			
		Case 2		WRF only		Case 2		WRF only	
observation		fog	clear	fog	clear	fog	clear	fog	clear
	fog	404	179	360	223	375	218	357	236
	clear	246	2843	225	2864	401	2625	410	2669

In the external test, Table 4.5 shows that case 2 (TSS) has higher recall scores than WRF. Table 4.6 shows the confusion matrices of case 2 and WRF. Again, the prediction of WRF is the prediction of liquid water in the lowest model level. It has class clear when there is no liquid water and class fog is when there is any amount. In St. John's, case 2 has more true positives (404 to 360) and less false negatives (179 to 223) than WRF only, with slightly more false positives (246 to 223). The improvement of recall results from more true positives and less false negatives. In Yarmouth, compared to WRF only, case 2 has more true positives (375 to 357), less false positives (401 to 410) and more false negatives (218 to 236), also resulting in a higher recall.

One can use different methods to interpret the model, to see how the model relates the features with the target. Since XGBoost uses an ensemble of trees and each tree focuses on fixing the errors of the previous, showing a single tree will not give any reasonable explanations. Therefore, it is better to interpret the model from the features across the trees. XGBoost has a function that can show feature importances. However, it is found that the results are not consistent with

different calculation algorithms. Therefore, the SHAP (SHapley Additive exPlanations) value method (Lundberg and Lee, 2017) is used, which involves Shapley values from game theory. Features are considered as the players in the game, and the function of the model as the rule. The SHAP value of each feature shows how the feature contributes to the model prediction from the expected value. Since in our case, class fog has the value 1 and class clear has the value 0, the expected value is simply number of fogs divided by the total amount of data. Fig. 4.14 and Fig. 4.15 are the summary plots of SHAP values of the two locations. The colour from blue to red means the value of the feature from low to high. The thickness of the bar means the density of the data points. The x-axis means the impact of the feature on model output and the values are probabilities. Negative SHAP values mean that the feature has impact towards a prediction of 0 (clear). Features are ordered by the average of the absolute value of the SHAP values, so that the top has the most impact and the bottom has the least impact.

For both locations RH2 is the most important one among all the features. The long, narrow and red part of RH2 on the positive side means that high values of RH2 will cause the model to more likely give a positive (fog) prediction, and some data points have very high impact. On the other side, most of the medium-to-low values of RH2 have negative impacts, but their impacts are not very high. Since the values of RH2 do not change very quickly with time, RH2lag1, being the second most important feature, shows similar patterns as RH2 but less impactful.

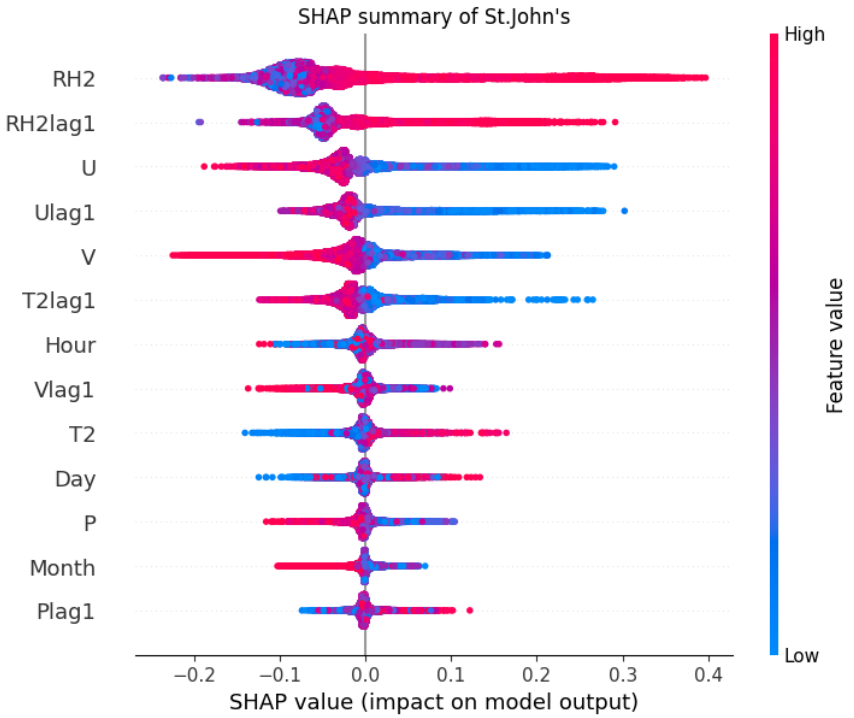


Figure 4.14 A SHAP summary plot of St. John's.

Figure 4.15 A SHAP summary plot of Yarmouth.

U wind and V wind are different between the two locations, and they are consistent with the correlation heatmap in Fig. 4.8, which can be explained by the direction of the advection. In St. John's, U and Ulag1 are ranked the third and forth, while V and Vlag1 are the fifth and the eighth. Either U or V, lower wind speed with blue colour will lead to a prediction of fog, and higher wind speed with red colour will lead to a prediction of clear. Therefore, an NE wind improves the possibility of a fog. From Fig. 4.1, the ocean is in the NE direction of St. John's, and the NAVCAN measurements at the St. John's International Airport is at 140.5 m above mean sea level. When the NE wind brings the moisture from the ocean to the land, the air climbs uphill and cools down, which increases the possibility of condensation in the air. On the other hand, Yarmouth has the opposite where higher U and higher V both lead to fog, and they are less important. The situation of Yarmouth is similar to Sable Island, where the SW wind in the summer brings the heat and moist air from the Gulf stream to the location. The elevation of the Yarmouth Airport is 42.90 m, therefore, there is also condensation due to air moving upward.

The two features of temperature, T2 and T2lag1 are interesting. In both locations, higher T2 suggests fog more likely, but it is more significant in Yarmouth than in St. John's. Although temperature will change in a day, but Month should be more important for the dataset including 5 months. According to Fig. 4.3 and Fig. 4.4, Yarmouth has more fogs in hotter months of July and August while St. John's has more in colder months, so T2 in Yarmouth has included the impact of Month while in St. John's it is countered to become less significant. T2lag1 does not have significant impacts in Yarmouth, except some outliers, but it is more important in St. John's also with outliers. Those blue outliers indicate that some low T2lag values has large impacts on fog prediction.

Hour is more important in Yarmouth, with the mixed values indicating a higher chance of fog, because Hour is in UTC, and the maximum 23 is followed by the minimum 0 of the next day. From the perspective of radiation, the land will have the lowest temperature in early morning when UTC is at noon. Pressure is not important in predicting fog. These SHAP values show that although XGBoost does not understand the physics involved, what it has learnt from the historical data is reasonable and can be interpreted.

Since RH2 and RH2lag1 are the two most important features, the performance of the ML model relied highly on the RH prediction of WRF. If WRF has errors on the prediction, the ML model will be misled. It is necessary to see how the model actually gives prediction. Table 4.7 shows two examples of predictions on June 24 and August 16, 2024, in Yarmouth. First column is hour in UTC on the corresponding day. WRFRH is the predicted RH of WRF in %, and WRFpred is classified as fog when it has any amounts of liquid water. OBSRH is the reported RH from the historical hourly report. OBSVIS is the reported visibility in km. OBS is classified with OBSVIS <= 1.2 km.

On June 24th, reported RH was very high and the visibility was low in the first few hours, although WRF had very high RH, it did not have fog because water can only condense at 100% RH in WRF. The ML model correctly predicted fog with the given WRF RH, because based on historical data, it does not require the RH to reach 100% to give a prediction of fog. From 21Z to 23Z QC was still 0 although RH was 100% due to that QC is at the lowest model level while RH at 2 m is calculated using similarity theory. This is an example of the model successfully

corrected the WRF output. This correction can decrease the number of false negatives and improve the recall.

Table 4.7 Relative humidity and fog predictions on July 24th and August 16th, 2024, in Yarmouth. OBSVIS is in km. WRFpred is based on the appearance of liquid water in the lowest level.

2024-06-24	WRFRH	WRFpred	OBSRH	OBSVIS	OBS	MLpred	2024-08-16	WRFRH	WRFpred	OBSRH	OBSVIS	OBS	MLpred
0	98.9516	clear	100	0.2	fog	fog	0	100	fog	100	24.1	clear	clear
1	99.2935	clear	100	0.2	fog	fog	1	100	fog	100	24.1	clear	clear
2	99.8605	clear	100	0.4	fog	fog	2	100	fog	100	24.1	clear	clear
3	99.4167	clear	100	0.4	fog	fog	3	100	fog	100	24.1	clear	clear
4	99.2197	clear	100	0.6	fog	fog	4	100	fog	100	0.2	fog	clear
5	100	clear	100	0.4	fog	fog	5	100	fog	100	0.4	fog	clear
6	100	fog	100	0.4	fog	fog	6	100	fog	100	0.4	fog	clear
7	100	fog	100	0.4	fog	fog	7	100	fog	100	0.2	fog	clear
8	100	fog	100	0.4	fog	fog	8	100	fog	100	0.2	fog	clear
9	100	fog	100	0.2	fog	fog	9	100	fog	100	1	fog	fog
10	100	clear	100	0.2	fog	fog	10	100	fog	100	0.2	fog	clear
11	100	clear	100	0.2	fog	fog	11	97.66085	clear	100	0.4	fog	clear
12	100	clear	100	0.8	fog	fog	12	89.54343	clear	100	0.8	fog	clear
13	98.8269	clear	100	1.2	fog	fog	13	74.68627	clear	100	4	clear	clear
14	98.653	clear	100	1.6	clear	fog	14	68.6583	clear	100	11.3	clear	clear
15	94.0415	clear	100	4.8	clear	fog	15	68.58209	clear	97	11.3	clear	clear
16	91.2184	clear	98	3.2	clear	clear	16	74.11628	clear	85	24.1	clear	clear
17	91.1493	clear	100	0.8	fog	clear	17	71.66692	clear	92	11.3	clear	clear
18	92.9003	clear	100	1.2	fog	clear	18	69.51038	clear	86	11.3	clear	clear
19	96.4197	clear	100	1.2	fog	clear	19	78.2293	clear	100	4.8	clear	clear
20	99.7939	fog	100	1.6	clear	fog	20	83.57793	clear	100	6.4	clear	clear
21	100	clear	100	0.4	fog	fog	21	85.62091	clear	100	1.6	clear	clear
22	100	clear	100	1.2	fog	fog	22	86.113	clear	100	0.2	fog	clear
23	100	clear	100	4.8	clear	fog	23	95.21206	clear	100	4	clear	clear

On August 16th, in the first few hours, although the reported RH was 100%, visibility was still at the maximum of 24.1 km. The WRF model accurately predicted RH of 100% and it had fog which is different from the reported visibility. From 4Z, reported visibility was low and it matched the prediction of WRF. However, the ML model predicted clear instead, with only one

hour fog at 9 a.m. In this example, there was a difference between OBSRH and OBSVIS.

The ML model is trained with RH2 from the ERA5 dataset and reported visibility. Indeed, in the training dataset from 2012 to 2023, for RH2 >= 98%, 36% of the data is reported as clear in Yarmouth, while in St. John's there are 45%. This is an example where the ML model reports clear even though the RH2 is 100%, possibly due to that there are a number of training data points where high RH is associated with clear weather. It is not clear if this is an error of the observation.

4.5 Discussion

This chapter has established an approach of combining machine learning and WRF output. XGBoost is chosen as the ML model. It is provided with meteorological variables including 2 m temperature, 2 m relative humidity, U wind, V wind and pressure, with their values one hour ago, and hour, day and month as features. The ML model is trained with the ERA5 dataset, from April to August in 2012-2023. Two locations, St. John's and Yarmouth have been chosen, and they are trained separately.

During the training, the data needs to be split. Two split methods, StratifiedShuffleSplit (SSS) and TimeSeriesSplit (TSS), have been used. SSS splits the data randomly, with each segment having the same ratio of different classes. TSS splits the data only according to time. The whole dataset from 2012 to 2023 is split into a CV set and an internal test set. The CV set is further

divided into 10 segments for cross validation. Two cases, one using SSS and one using TSS acquired the optimal hyperparameters from CV. Then the model is trained again using the whole CV set, with a portion of the data used as validation for early stopping. The trained model is tested with the internal test set, and StratifiedShuffleSplit performs better than TimeSeriesSplit.

To simulate the daily operational usage, output of the features (2 m temperature, 2 m RH etc.) from the WRF model is used for the external test for 2024. The WRF model is run to get the corresponding variables for the day, which are then used as input for the XGBoost. From the results, St. John's is easier to predict than Yarmouth, meaning that it is easier for XGBoost to distinguish fog and clear in St. John's. It is possible that the mechanism of fog in Yarmouth is more complex, and the current given features are not able to capture the difference between fog and clear. The two split methods have similar scores in the external test, despite their difference in the internal test. In other words, StratifiedShuffleSplit has overfit the training data and what it has learnt from it cannot be applied to a new dataset. Therefore, it has overestimated its performance in the internal test which is part of the known dataset.

Predicted liquid water from the WRF output has also been extracted for comparison. It is found that using XGBoost to predict fog based on WRF output of features performs slightly better than the liquid water prediction from WRF itself. In St. John's, XGBoost has a significantly higher recall, which means more true positives and less false negatives.

Further model interpretation has been done with the SHAP value, which can calculate the

contribution of each feature to the model output. RH2 and RH2lag1 are the two most important features of the model. Other features except pressure have different importances between the two locations, which shows that it is reasonable to include different features, and that the locations should be trained individually.

When investigating the model predictions in detail, there are some periods when WRF alone cannot correctly give a prediction of fog because the RH does not reach 100%, although it is very close. The ML model can fix this during training because fog is not tied to 100% RH. Some WRF post-processing software are able to do the same, but a ML model can be applied to any NWP models, not only WRF. There are also periods when WRF has 100% RH and the reported RH is also 100%, but the reported visibility is high. It is not clear if the difference between reported RH and visibility is an observation error. It is possible that the two measurements are at different elevation. With the current approach it will decrease the connection between RH and fog in the ML model, resulting in a wrong prediction. For further improvements, it is possible to include more features in the vertical so that the ML model can understand the difference between reported RH and visibility.

This is a simple approach to combine machine learning and WRF, with a limited number of features and 11 years of data including only 5 months from 2 locations, but it shows the potential to use NWP models to cover the area where machine learning cannot be used alone for various reasons. In the case of marine fog, since the mostly widely used dataset ERA5 does not include near-surface liquid water, the popular deep learning models cannot predict it, while NWP models

can have liquid water from its calculation. However, as discussed in the previous chapters, the prediction of liquid water of NWP models is still difficult. The use of ML can post-process the NWP model outputs by using statistics based on previous events and improve the results. The disadvantage is that, unlike the deep learning models such as GraphCast, the ML model with this approach cannot outperform the NWP model too much since its prediction is based on the NWP model prediction. Quality and quantity of the training data are also important, but the observation of marine fog is still limited. Once the model is trained, it is very fast for the ML model to give a prediction. Therefore, this approach has its potential and could be used to supply material for forecasts.

Chapter 5

Conclusion

5.1 Summary

Marine fog is of importance for human activities in the coastal regions and over the ocean. However, currently marine fog forecasting still has difficulties. This dissertation includes several studies on forecasting marine fog in the Northern Atlantic.

The first study in Chapter 2 has summarized the forecasts of 3 different models, including two WRF models with different settings and one COAMPS model, during the FATIMA 2022 field campaign on Sable Island, NS. Comparing the model results to the reported weather from NAVCAN at the Sable Island airport, it is found that all the models have similar ranges of errors for 2 m temperature, 2 m dew point, 2 m relative humidity, 10 m wind speed and 10 m wind direction. However, model performances are different for fog forecast. COAMPS has the highest scores for fog period prediction. An additional test has been done on the WRF model, which shows that with adjusted horizontal and vertical domains, the WRF fog prediction was improved. The test also shows the importance of the use of spin-up time for forecasting fog, which is needed to generate liquid water in the model.

The study in chapter 3 has taken a closer look at the FATIMA Sable Island field campaign. The field measurements from a FM-120 instrument of LWC and QNC during the campaign are compared to the WRF model values. Different settings in the WRF model have been tested in two IOPs. It is found that although tuning the settings has an impact on the values, the model predicts much higher LWC and QNC than the observed values. It is found that the updraft speed in the Thompson microphysics scheme can be one of the problems. The activation rate of CCN turning into droplets is determined by 5 quantities with the updraft speed being one of them. However, the minimum updraft speed in the scheme is too high in a fog, resulting in an activation rate that is also too high. Another problem is that fogs in WRF tend to dissipate over land in daytime even though the observation shows that there is still liquid water over the land. In the two study periods, model 2 m temperature is always higher than observation, but temperature aloft matches the sounding. Since Sable Island is a small island, the temperature of the land is not included in the GFS input data with a resolution of 0.25° x 0.25°. The pre-processing software WPS prepares the geographic data of the land and assigns the temperature of the land by extrapolating the land temperature from Nova Scotia which is higher. The model seems to perform better for fog in the surrounding water.

These two studies have pointed out the problems existing in the WRF model. NCAR has announced that they are focusing on a new NWP model called MPAS (Model for Prediction Across Scales, Skamarock et al. 2012). The current WRF model will still be maintained and supported. However, this doesn't mean that these studies are worthless. One can test whether the new model still has the same problems or not, since many functions in MPAS are integrated from the WRF model.

The last study is an approach to combine WRF output with an ML model. The XGBoost model is chosen, with training data between 2012 and 2023, in months from April to August. The features are from the ERA5 dataset, including 2 m temperature, 2 m relative humidity, 10 m U wind, 10 m V wind and land surface pressure, and their lag1 values (values one hour before), as well as hour, day and month. Using a binary classification method, the target is either clear or fog. The visibility data used is from St. John's international airport and Yarmouth airport. It is decided that the class fog is when reported visibility <= 1.2 km and class clear is when reported visibility > 1.2 km. Two split methods, StratifiedShuffleSplit (SSS) and TimeSeriesSplit (TSS) are compared in the internal test and the external test. The internal test uses a part of the training data to validate model performance, and SSS performs better than TSS. The external test simulates daily operation usage of forecasting. The WRF model is used first to predict the variables needed in the next 24 hours, then the predicted variables are used as input for XGBoost to give predictions of fog or clear. The external test covers the year of 2024, from April to August. It is found that the two split methods have similar performance in the external test despite their difference in the internal test, meaning that SSS has overestimated its performance. However, with the input of features (T2, RH2 etc.) from WRF, both methods have slightly higher performances than WRF predictions of liquid water, with higher recalls. Further model interpretation using the SHAP values show that model prediction relies heavily on the input RH2 and RH2lag1. A close inspection on the model prediction in the external test shows that the model can fix the error of WRF to some extent. WRF can only produce liquid water when its RH reaches 100%, so it can miss a fog event even though its RH is very high. The ML model, trained with historical data that connects fog with RH that is very high but not 100%, can predict fog accurately and thus reducing false negatives and increasing true positives, which increases the recall. However, it is

also found that in the reported data, visibility is very high despite a 100% RH. Because of this, sometimes the model will also predict clear with a RH of 100%. It is difficult to say if the report has errors, but it shows the importance of data quality for machine learning.

NWP models have a long lead time, but they have problems such as high LWC. ML models are not good at dealing with extreme values over time, and the accuracy decreases rapidly with longer lead times. Therefore, our approach of combining an NWP model and an ML model is trying to have the advantages from both sides. Although our features include hour, day and month, our ML model is not really doing a time series forecast. It does not connect a data point with another data point one hour ago or one hour later. Since each data point is independent, the forecast at any time from the NWP model can be input to the ML model. This solves the problem of lead time. On the other hand, the ML model is used to correct the NWP model, based on historical weather conditions. As presented in Chapter 4, it can predict fog with a RH lower than 100%, which can also solve the limitation of the NWP model. I believe that this approach can improve fog forecasting.

5.2 Future Directions

As the AI technology is being developed at a surprisingly high speed, the use of machine learning in weather forecasts has great potential. After publishing GraphCast (Lam et al., 2023) which is claimed to outperform most of the NWP models, Google has published a new model called GenCast (Price et al., 2024). Also trained with the ERA5 data, it generates an ensemble of 15-days global forecasts and is claimed to be better than GraphCast. NVIDIA also has its weather

model call Earth-2 (NVIDIA 2023) which also trained with ERA5 data. However, as discussed in the previous chapter, although these weather models using machine learning are fast and can perform better than NWP models, their weakness is that they can only produce the variables that they are trained with. Also, unless specifically trained, they cannot have a high resolution of hundreds of meters which is possible in NWP models.

On the side of NWP models, the MPAS model mentioned in the last section uses a hexagonal Voronoi mesh. Compared to the traditional rectangular grid in WRF, it can have a smoother transition between higher and lower resolution nesting. It is claimed to be well-suited for higher-resolution, mesoscale atmosphere and ocean simulations.

In the foreseeable future, traditional NWP models will not be completely replaced by ML models, but they can have different roles for weather forecasts, with ML models producing general and low-resolution predictions and NWP models being more specific to a location or a variable. Especially for marine fog, due to the lack of observations, it is difficult to train an ML model that can have satisfactory predictions. This is the reason for having a test of combining ML and NWP models in Chapter 4, instead of continuing deeper research on WRF following the previous chapters. I hope this dissertation can give the readers some ideas on the future of weather forecasting and the relationship between NWP and ML models.

Bibliography

Boudala, F. S., Isaac, G. A., Crawford, R. W. and Reid J., 2012: Parameterization of Runway Visual Range as a Function of Visibility: Implications for Numerical Weather Prediction Models, *Journal of Atmospheric and Oceanic Technology*, **29**(2), 177-191

Blaylock, B. K., 2020: How to Initialize WRF with HRRR Boundary Conditions. https://home.chpc.utah.edu/~u0553130/Brian_Blaylock/hrrr.html

Brownlee, J., 2020: How to Configure XGBoost for Imbalanced Classification. *Machine Learning Mastery*. https://machinelearningmastery.com/xgboost-for-imbalanced-classification/

Cohen, A. E., Cavallo, S. M., Coniglio, M. C., and Brooks, H. E., 2015: A Review of Planetary Boundary Layer Parameterization Schemes and Their Sensitivity in Simulating Southeastern U.S. Cold Season Severe Weather Environments. *Weather and Forecasting*, **30**(3), 591-612. https://doi.org/10.1175/WAF-D-14-00105.1

Chen, T., and Guestrin, C., 2016. XGBoost: A Scalable Tree Boosting System. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '16). Association for Computing Machinery, New York, NY, USA, 785–794. https://doi.org/10.1145/2939672.2939785

Chen, Z., 2021: Studying marine fog using the Weather Research and Forecast (WRF) model, MSc thesis, York University, 91pp.

Cheng, L., Chen, Z., Taylor, P. A., Chen, Y., and Isaac, G. A., 2021: Fog over Sable Island, CMOS Bulletin SCMO, June 21, 2021, https://bulletin.cmos.ca/fog-over-sable-island/

de La Fuente, Y., Delage, Y., Desjardines, S., MacAfee, A., Pearson, G., and Ritchie, H., 2007: Can Sea Fog be Inferred from Operational GEM Forecast Fields? *Pure appl. geophys*., **164**, 1303-1325. https://doi.org/10.1007/s00024-007-0220-9

Dimitrova, R., Sharma, A., Fernando, H.J.S., Gultepe, I., Danchovski, V., Wagh, S., Bardoel, S. L. and Wang, S., 2021: Simulations of Coastal Fog in the Canadian Atlantic with the Weather Research and Forecasting Model. *Boundary-Layer Meteorol* **181**, 443–472. https://doi.org/10.1007/s10546-021-00662-w

Duda, M., 2021: Advanced Usage of the WRF Preprocessing System. https://youtu.be/xibGSLSaj6w?si=aHIrd83iEMZN0J3V

Environment and Climate Change Canada, 2013: National marine weather guide. https://www.publications.gc.ca/site/eng/9.629465/publication.html

Fernando, H. J. S., and Coauthors, 2025: Fatima-GB: Searching Clarity within Marine Fog. *Bull. Amer. Meteor. Soc*., **106**, E971–E1016, https://doi.org/10.1175/BAMS-D-23-0050.1

Gultepe E., Wang S., Blomquist B., Fernando H. J. S., Kreidl O. P., Delene D.J. and Gultepe I., 2024:Machine learning analysis and nowcasting of marine fog visibility using FATIMA Grand Banks campaign measurements. *Front. Earth Sci*. **11**:1321422. doi: 10.3389/feart.2023.1321422

Gultepe I, Muller, M.D., and Boybeyi, Z. 2006: A new visibility parametrization for warm fog applications in numerical weather prediction models. *J. Appl. Meteorol*. **45**, 1469–1480. https://doi.org/10.1175/JAM2423.1

Hersbach, H., Bell, B., Berrisford, P., Biavati, G., Horányi, A., Muñoz Sabater, J., Nicolas, J., Peubey, C., Radu, R., Rozum, I., Schepers, D., Simmons, A., Soci, C., Dee, D., Thépaut, J-N. (2023): ERA5 hourly data on single levels from 1940 to present. Copernicus Climate Change Service (C3S) Climate Data Store (CDS), DOI: 10.24381/cds.adbb2d47 (Accessed on 04-10-2024)

Hsieh, W. W. (2023). Introduction to environmental data science (1st ed.). Cambridge University Press.

Hodur, R.M., J. Pullen, J. Cummings, X. Hong, J.D. Doyle, P. Martin, and M.A. Rennick. 2002. The Coupled Ocean/Atmosphere Mesoscale Prediction System (COAMPS). *Oceanography* **15**(1):88–98, https://doi.org/10.5670/oceanog.2002.39.

Iacono, M. J., Delamere, J. S., Mlawer, E. J., Shephard, M. W., Clough, S. A., and Collins, W. D., 2008: Radiative forcing by long–lived greenhouse gases: Calculations with the AER radiative transfer models. *J. Geophys. Res*., **113**, D13103. doi:10.1029/2008JD009944

Isaac, G. A., Bullock, T., Beale, J., and Beale, S., 2020: Characterizing and Predicting Marine Fog Offshore Newfoundland and Labrador. Weather and Forecasting, **35**(2), 347-365. https://doi.org/10.1175/WAF-D-19-0085.1

Jahn, D. E., and W. A. Gallus, 2018: Impacts of Modifications to a Local Planetary Boundary Layer Scheme on Forecasts of the Great Plains Low-Level Jet Environment. *Wea. Forecasting*, **33**, 1109–1120, https://doi.org/10.1175/WAF-D-18-0036.1.

Jain, A., 2025: XGBoost Parameters Tuning: A Complete Guide with Python Codes. *Analytics Vidhya*. https://www.analyticsvidhya.com/blog/2016/03/complete-guide-parameter-tuning-xgboost-with-codes-python/

Kim, B., Belorid, M., and Cha, J, W., 2022: Short-Term Visibility Prediction Using Tree-Based Machine Learning Algorithms and Numerical Weather Prediction Data. *Wea. Forecasting,* **37**, 2263–2274, https://doi.org/10.1175/WAF-D-22-0053.1.

Koračin, D., Dorman, C. E., 2017: Marine Fog: Challenges and Advancements in Observations, Modeling, and Forecasting. Springer.

Kunkel, B. A., 1984: Parameterization of droplet terminal velocity and extinction coefficient in fog models. *J. Climate Appl. Meteor*., **23**, 34–41.

Lam, R., et al., Learning skillful medium-range global weather forecasting. Science **382**,1416-1421(2023). DOI:10.1126/science.adi2336

Lin, C., Zhang, Z., Pu, Z., and Wang, F., 2017: Numerical simulations of an advection fog event over Shanghai Pudong International Airport with the WRF model. *Journal of Meteorological Research*, **31**, 874-889. https://doi.org/10.1007/s13351-017-6187-2

Lundberg, S. (2017). A unified approach to interpreting model predictions. arXiv preprint arXiv:1705.07874.

Nakanishi, M., and Niino, H., 2006: An Improved Mellor–Yamada Level-3 Model: Its Numerical Stability and Application to a Regional Prediction of Advection Fog. *Boundary-Layer Meteorology*. **119**, 397-407. https://doi.org/10.1007/s10546-005-9030-8

Nakanishi, M., and Niino, H., 2009: Development of an Improved Turbulence Closure Model for the Atmospheric Boundary Layer. *Journal of the Meteorological Society of Japan*, **87**, 895-912. https://doi.org/10.2151/jmsj.87.895

NVIDIA, 2023: Earth-2 MIP. https://nvidia.github.io/earth2mip/

Pedregosa, F., Varoquaux, Ga"el, Gramfort, A., Michel, V., Thirion, B., Grisel, O., … others. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, **12**(Oct), 2825–2830.

Price, I., et al., 2024: GenCast: Diffusion-based ensemble forecasting for medium-range weather. https://doi.org/10.48550/arXiv.2312.15796

Riordan, D. and Hansen, B. K., 2002: A fuzzy case-based system for weather prediction. *Eng. Int. Syst.*, **3**, 139–146.

Skamarock, W. C., Klemp, J. B., Dudhia, J., Gill, D. O., Liu, Z., Berner, J., Wei Wang, J. G. Powers, M.G. Duda, D., M. Barker and Huang, X., 2019: A Description of the Advanced Research WRF Model Version 4 (No. 771 NCAR/TN-556+STR). http://dx.doi.org/10.5065/1dfh-6p97

Skamarock, W. C., J. B. Klemp, M. G. Duda, L. D. Fowler, S. Park, and T. D. Ringler, 2012: A Multiscale Nonhydrostatic Atmospheric Model Using Centroidal Voronoi Tesselations and C-Grid Staggering. Mon. Wea. Rev., **140**, 3090–3105, https://doi.org/10.1175/MWR-D-11-00215.1.

Scikit-learn developers, 2007-2025: StratifiedShuffleSplit. https://scikit-learn.org/1.6/modules/generated/sklearn.model_selection.StratifiedShuffleSplit.html

Stoelinga, M. T. and Warner, T. T., 1999: Nonhydrostatic, Mesobeta-Scale Model Simulations of Cloud Ceiling and Visibility for an East Coast Winter Precipitation Event, *Journal of Applied Meteorology and Climatology*, **38**(4), 385-404, https://doi.org/10.1175/1520-0450(1999)038%3C0385:NMSMSO%3E2.0.CO;2

Taylor, P. A., Chen, Z., Cheng, L., Afsharian, S., Weng, W., Isaac, G. A., Bullock, T., and Chen, Y., 2021: Surface deposition of marine fog and its treatment in the Weather Research and Forecasting (WRF) model, *Atmos. Chem. Phys*., **21**, 14687–14702, https://doi.org/10.5194/acp-21-14687-2021

Taylor, P. A., Salmon, J. R. & Stewart, R.E., 1993: Mesoscale observations of surface fronts and low pressure centres in Canadian East Coast winter storms. *Boundary-Layer Meteorology*, **64**, 15–54. https://doi.org/10.1007/BF00705661

Telford, J. W. and Chai, S. K., 1993: Marine Fog and Its Dissipation over Warm Water. *J. Atmos. Sci.*, **50**, 3336-3349.

The SciPy community, 2025: scipy.stats.ttest_ind. https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.ttest_ind.html

Thomasson, A., 2023: Improving Visibility Forecasts in Denmark Using Machine Learning Post-processing, Dissertation. Retrieved from https://urn.kb.se/resolve?urn=urn:nbn:se:uu:diva-500551

Thompson, Gregory, Paul R. Field, Roy M. Rasmussen, William D. Hall, 2008: Explicit Forecasts of Winter Precipitation Using an Improved Bulk Microphysics Scheme. Part II: Implementation of a New Snow Parameterization. *Mon. Wea. Rev*., **136**, 5095–5115. doi:10.1175/2008MWR2387.1

Thompson, Gregory, and Trude Eidhammer, 2014: A study of aerosol impacts on clouds and precipitation development in a large winter cyclone. *J. Atmos. Sci*., **71**(10), 3636-3658. doi:10.1175/JAS-D-13-0305.1

The NCAR Command Language (Version 6.6.2) [Software]. (2019). Boulder, Colorado: UCAR/NCAR/CISL/TDD. http://dx.doi.org/10.5065/D6WD3XH5

Olson, J. B., Kenyon, J. S., Angevine, W. A., Brown, J. M., Pagowski, M., and Suselj, K., 2019: A Description of the MYNN-EDMF Scheme and the Coupling to Other Components in WRF–ARW. *NOAA Technical Memorandum* OAR GSD-61. https://doi.org/10.25923/n9wm-be49

Vorndran, M., Schütz, A., Bendix, J., and Thies, B. (2022). Current training and validation weaknesses in classification-based radiation fog nowcast using machine learning algorithms. *Artif. Intell. Earth Syst*. **1**(2), e210006. doi:10.1175/aies-d-21- 0006.1

Weston, M. J., Piketh, S. J., Burnet, F., Broccardo, S., Denjean, C., Bourrianne, T., and Formenti, P.: Sensitivity analysis of an aerosol-aware microphysics scheme in Weather Research and Forecasting (WRF) during case studies of fog in Namibia, *Atmos. Chem. Phys*., **22**, 10221–10245, https://doi.org/10.5194/acp-22-10221-2022

World Meteorological Organization, 2020: Guide to Instruments and Methods of Observation (WMO-NO. 8). https://community.wmo.int/en/activity-areas/imop/wmo-no_8

Wilson, T. H., and Fovell, R. G., 2017: Modeling the Evolution and Life Cycle of Radiative Cold Pools and Fog. *Weather and Forecasting*, **33**(1), 203-220. DOI: 10.1175/WAF-D-17-0109.1

Xgboost developers, 2022: Introduction to Boosted Trees. *XGBoost Tutorials*. https://xgboost.readthedocs.io/en/stable/tutorials/model.html

Yang, L., Liu, J.-W., Ren, Z.-P., Xie, S.-P., Zhang, S.-P., and Gao, S.-H., 2018: Atmospheric conditions for advection- radiation fog over the western Yellow Sea. *Journal of Geophysical Research: Atmospheres*, **123**(10), 5455–5468. https://doi.org/10.1029/2017JD028088

Zhang, C., Wang, Y., and Hamilton, K., 2011: Improved representation of boundary layer clouds over the southeast pacific in ARW–WRF using a modified Tiedtke cumulus parameterization scheme. *Mon. Wea. Rev*., **139**, 3489–3513. doi:10.1175/MWR-D-10-05091.1

Appendix A

Table of Abbreviations

Abbreviation	Definition	Abbreviation	Definition
COAMPS	Coupled Ocean/Atmosphere Mesoscale Prediction System	NAVCAN	NAV CANADA
CV	Cross Validation	NCAR	National Center for Atmospheric Research
ECCC	Environment and Climate Change Canada	NCL	NCAR Command Language
ECMWF	European Centre for Medium-range Weather Forecasts	NOAA	National Oceanic & Atmospheric Administration
ERA5	ECMWF Reanalysis v5	NWP	Numerical Weather Prediction
FATIMA	Fog And Turbulence Interactions in the Marine Atmosphere	QNC	Cloud Droplet Number Concentration in cm^{-3}
FNL	Final Analysis	PBL	Planetary Boundary Layer
GFS	Global Forecast System	SSS	Stratified Shuffle Split

HRRR	High-Resolution Rapid Refresh	SST	Sea Surface Temperature
IOP	Intensive Observation Period	TSS	Time Series Split
LWC	Liquid Water Content in gm^{-3}	WPS	WRF Pre-processing System
ML	Machine Learning	WRF	Weather Research and Forecasting model

Appendix B

WRF Configuration in Chapter 3

```
&time_control
 run_days                  = 2,
 run_hours                 = 18,
 run_minutes               = 0,
 run_seconds               = 0,
 start_year                = 2022, 2022, 2022,
 start_month               = 07,   07,   07,
 start_day                 = 13,   13,   13,
 start_hour                = 00,   00,   00,
 end_year                  = 2022, 2022, 2022,
 end_month                 = 07,   07,   07,
 end_day                   = 15,   15,   15,
 end_hour                  = 18,   18,   18,
 interval_seconds          = 21600,
 input_from_file           = .true.,.true.,.true.,
 history_interval          = 180,  180,   1,  1,
 frames_per_outfile        = 10000, 10000, 10000,
```

```
restart                             = .false.,
restart_interval                    = 9999,
io_form_history                     = 2
io_form_restart                     = 2
io_form_input                       = 2
io_form_boundary                    = 2
/
&domains
time_step                           = 30,
time_step_fract_num                 = 0,
time_step_fract_den                 = 1,
max_dom                             = 3,
e_we                                = 150,   241,   400,
e_sn                                = 150,   241,   400,
e_vert                              = 66,    66,    66,
p_top_requested                     = 5000,
wif_input_opt                       = 1,
num_metgrid_levels                  = 34,
num_metgrid_soil_levels             = 4,
eta_levels=1,0.9991135,0.998623497,0.99803047,
           0.997150521,0.99699502,0.99500502,0.994654294,0.99144918,
           0.988378889, 0.983023785,0.977507905,0.969705271,0.96155065,
           0.95206733,0.941481662,0.930303505,0.918056291,0.907513992,
           0.897044302,0.880773985,0.871601049,0.861914428,0.840937255,
           0.805241551,0.778471087,0.729219268,0.683024553,0.664846992,
           0.585617374,0.564201579,0.542157588,0.496230844,0.472375929,
           0.447948496,0.397453322,0.371428989,0.344918868,0.317949238,
```

0.290548095,0.277745068,0.262745068,0.234571334,0.214571334,
0.18724357,0.17724357,0.16724357,0.15724357,0.148158676,
0.120345567,0.118841103,0.098841103,0.09432808,0.08932808,
0.075657653,0.067657653,0.062657653,0.059657653,0.049657653,
0.039657653,0.029868546,0.020868546,0.013868546,0.0089868546,
0.0059868546,0

dx = 9000, 3000, 1000,
dy = 9000, 3000, 1000
grid_id = 1, 2, 3,
parent_id = 0, 1, 2,
i_parent_start = 1, 30, 40,
j_parent_start = 1, 30, 55,
parent_grid_ratio = 1, 3, 3,
parent_time_step_ratio = 1, 3, 3,
feedback = 0,
smooth_option = 0,
/
&physics
mp_physics = 28, 28, 28,
cu_physics = 6, 0, 0,
ra_lw_physics = 4, 4, 4,
ra_sw_physics = 4, 4, 4,
bl_pbl_physics = 5, 5, 5,
bl_mynn_closure = 2.5,
bl_mynn_cloudmix = 1, 1, 1,
sf_sfclay_physics = 5, 5, 5,
sf_surface_physics = 2, 2, 2,

```
radt                                = 9,    9,    9,
bldt                                = 0,    0,    0
cudt                                = 5,    0,    0,
icloud                              = 1,
icloud_bl                           = 1,
num_land_cat                        = 21,
/
&fdda
/
&dynamics
hybrid_opt                          = 2,
w_damping                           = 1,
diff_opt                            = 1,      1,      1,
km_opt                              = 4,      4,      4,
diff_6th_opt                        = 0,      0,      0,
diff_6th_factor                     = 0.12,   0.12,   0.12,
base_temp                           = 290.
damp_opt                            = 3,
zdamp                               = 5000.,  5000.,  5000.,
dampcoef                            = 0.2,    0.2,    0.2,
khdif                               = 0,      0,      0,
kvdif                               = 0,      0,      0,
non_hydrostatic                     = .true., .true., .true.,
moist_adv_opt                       = 1,      1,      1,
moist_adv_dfi_opt                   = 1,      1,      1,
scalar_adv_opt                      = 1,      1,      1,
gwd_opt                             = 1,
```

```
use_q_diabatic                  = 1,
/
&bdy_control
spec_bdy_width                   = 5,
specified                       = .true.
/
&grib2
/
&namelist_quilt
nio_tasks_per_group = 0,
nio_groups = 1,
/
```

Appendix C

Code of Chapter 4

```
#@title Import relevant modules
!pip install scikit-learn==1.5.2
import numpy as np
import pandas as pd
from matplotlib import pyplot as plt
import seaborn as sns
import dask.dataframe as dd
from sklearn.preprocessing import OrdinalEncoder
from sklearn.model_selection import
    train_test_split,GridSearchCV,cross_val_score,TimeSeriesSplit
from sklearn.metrics import
    confusion_matrix,recall_score,f1_score,precision_score,classification_report,make_scorer
    ,roc_auc_score
import xgboost as xgb
from sklearn.utils import class_weight
import shap
```

```
# The following lines adjust the granularity of reporting.
pd.options.display.max_rows = 10
pd.options.display.float_format = "{:.1f}".format
print("Imported modules.")

# @title Clean Data
# Load the CSV file into a DataFrame
# Define the path and pattern for the CSV files

# Read all CSV files into a Dask DataFrame
# Compute the Dask DataFrame to get a Pandas DataFrame
combined_sj = dd.read_csv('trainStjohn*.csv',dtype={'Vis': 'object'}).compute()
combined_ym = dd.read_csv('trainYarmouth*.csv',dtype={'Vis': 'object'}).compute()

# Remove rows where the first column is blank and set time index
sj_cleaned = combined_sj[combined_sj['Vis'].notnull()]
sj_cleaned = sj_cleaned.set_index('Time')

ym_cleaned = combined_ym[combined_ym['Vis'].notnull()]
ym_cleaned = ym_cleaned.set_index('Time')

# Save the cleaned DataFrame back to a CSV file
sj_cleaned.to_csv('cleaned_file_sj.csv', index=False)
ym_cleaned.to_csv('cleaned_file_ym.csv', index=False)

# @title Correlation Map
# Calculate the correlation matrix
```

```
sj_corr = sj_cleaned.drop(['Vis','Location','TD2','TD2lag1'], axis=1)
ym_corr = ym_cleaned.drop(['Vis','Location','TD2','TD2lag1'], axis=1)

sj_matrix = sj_corr.corr()
ym_matrix = ym_corr.corr()

correlation_matrix = sj_matrix

for i in range(len(correlation_matrix.columns)):
  for j in range(correlation_matrix.shape[0]):
    if i == j:
      correlation_matrix.iloc[i,j] = np.nan
    if i > j:
      correlation_matrix.iloc[i,j] = ym_matrix.iloc[i,j]

# Plot the correlation heatmap
plt.figure(figsize=(10, 8))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', fmt=".2f")
plt.title("Correlation Heatmap of Features of St. John's (upper right) and Yarmouth (lower left)")
plt.show()

# @title Add train data
X_sj, y_sj =
        sj_cleaned[['T2','T2lag1','U','Ulag1','V','Vlag1','RH2','RH2lag1','P','Plag1','Month','Day','H
        our']], sj_cleaned[['Vis']]
X_ym, y_ym =
        ym_cleaned[['T2','T2lag1','U','Ulag1','V','Vlag1','RH2','RH2lag1','P','Plag1','Month','Day','
        Hour']], ym_cleaned[['Vis']]
```

```
# Encode y to numeric
y_sj_encoded = OrdinalEncoder(categories=[['clear','fog']]).fit_transform(y_sj)
y_ym_encoded = OrdinalEncoder(categories=[['clear','fog']]).fit_transform(y_ym)


# Separate train and validation data
X_sj_CV, X_sj_test, y_sj_CV, y_sj_test =
        train_test_split(X_sj,y_sj_encoded,test_size=0.08,shuffle=False)
X_ym_CV, X_ym_test, y_ym_CV, y_ym_test =
        train_test_split(X_ym,y_ym_encoded,test_size=0.08,shuffle=False)


# @title Cross validation St. John's
# Define hyperparameters
cvparams_sj = {
    "max_depth":[1],
    "min_child_weight":[12],
    "gamma":[0],
    "subsample":[0.7],
    "lambda":[8],
    "alpha":[3],
}



cv_sj = xgb.XGBClassifier(objective="binary:logistic",
tree_method="hist",
enable_categorical=True,
learning_rate=0.1,
```

```
n_estimators=300,
scale_pos_weight=7.97,
)

grid_search_sj = GridSearchCV(estimator=cv_sj,
param_grid=cvparams_sj,
cv=TimeSeriesSplit(n_splits=10),
n_jobs=1,
verbose=3,
scoring=make_scorer(f1_score,labels=[1],average='weighted')
)

grid_search_sj.fit(X_sj_CV,y_sj_CV)
print("Best hyperparameters:", grid_search_sj.best_params_)
print("Best score:", grid_search_sj.best_score_)

# @title Cross validation Yarmouth
# Define hyperparameters
cvparams_ym = {
    "max_depth":[2],
    "min_child_weight":[11],
    "gamma":[6],
    "subsample":[1],
    "alpha":[1,2,3,4,5],
    "lambda":[1,2,3,4,5],
}
```

```
cv_ym = xgb.XGBClassifier(objective="binary:logistic",
tree_method="hist",
enable_categorical=True,
learning_rate=0.1,
n_estimators=300,
scale_pos_weight=5.75,
)

grid_search_ym = GridSearchCV(estimator=cv_ym,
param_grid=cvparams_ym,
cv=TimeSeriesSplit(n_splits=10),
n_jobs=1,
verbose=3,
scoring=make_scorer(f1_score,labels=[1],average='weighted')
)

grid_search_ym.fit(X_ym_CV, y_ym_CV)
print("Best hyperparameters:", grid_search_ym.best_params_)
print("Best score:", grid_search_ym.best_score_)

# @title Train model St. John
good_params_sj = grid_search_sj.best_params_

good_params_sj.update({
    "objective":"binary:logistic",
    "tree_method":"hist",
```

```
    "learning_rate":0.1,
    "scale_pos_weight":7.97,
})

X_sj_train, X_sj_valid, y_sj_train, y_sj_valid = train_test_split(X_sj_CV, y_sj_CV,
        test_size=0.09, shuffle=False)

# Create classification matrices
dtrain_sj = xgb.DMatrix(X_sj_train, y_sj_train)
dvalid_sj = xgb.DMatrix(X_sj_valid, y_sj_valid)
evals_sj = [(dtrain_sj,"train"),(dvalid_sj,"validation")]
evals_result_sj = {}

model_sj = xgb.train(
  params=good_params_sj,
  dtrain=dtrain_sj,
  num_boost_round=114514,
  evals=evals_sj,
  verbose_eval=False,
  early_stopping_rounds=20,
  evals_result=evals_result_sj,
  callbacks=[xgb.callback.EvaluationMonitor(period=50)]
)

# @title Train model Yarmouth
good_params_ym = {
    "objective":"binary:logistic",
```

```
    "tree_method":"hist",
    "learning_rate":0.1,
    "scale_pos_weight":5.75,
    "max_depth":7,
    "min_child_weight":11,
    "subsample":1,
    "lambda":4.4,
    "alpha":0.3,
    "gamma":1.5
}



X_ym_train, X_ym_valid, y_ym_train, y_ym_valid = train_test_split(X_ym_CV, y_ym_CV,
        test_size=0.09, shuffle=False)

# Create classification matrices
dtrain_ym = xgb.DMatrix(X_ym_train, y_ym_train)
dvalid_ym = xgb.DMatrix(X_ym_valid, y_ym_valid)
evals_ym = [(dtrain_ym,"train"),(dvalid_ym,"validation")]
evals_result_ym = {}

model_ym = xgb.train(
  params=good_params_ym,
  dtrain=dtrain_ym,
  num_boost_round=114514,
  evals=evals_ym,
  verbose_eval=False,
```

```
    early_stopping_rounds=20,
    evals_result=evals_result_ym,
    callbacks=[xgb.callback.EvaluationMonitor(period=50)]
)

# @title Training and Validation Log loss
train_logloss_sj = evals_result_sj['train']['logloss']
validation_logloss_sj = evals_result_sj['validation']['logloss']
train_logloss_ym = evals_result_ym['train']['logloss']
validation_logloss_ym = evals_result_ym['validation']['logloss']

epochs_sj = range(1, len(train_logloss_sj) + 1)
epochs_ym = range(1, len(train_logloss_ym) + 1)

plt.plot(epochs_sj, train_logloss_sj, 'b-', label='St. John\'s Training Log Loss')
plt.plot(epochs_sj, validation_logloss_sj, 'r-', label='St. John\'s Validation Log Loss')
plt.plot(epochs_ym, train_logloss_ym, 'b--', label='Yarmouth Training Log Loss')
plt.plot(epochs_ym, validation_logloss_ym, 'r--', label='Yarmouth Validation Log Loss')

plt.title('TimeSeriesSplit Training and Validation Log Loss for St. John\'s and
        Yarmouth',fontsize=11)
plt.xlabel('Boosting Rounds')
plt.ylabel('Log Loss')
plt.ylim(0, 0.7)
plt.legend()
plt.show()
```

```
# @title SHAP

explainer = shap.TreeExplainer(model_sj,X_sj_train,model_output='probability')
shap_values = explainer(X_sj_train)

shap.initjs()

shap.plots.beeswarm(shap_values,max_display=13,show=False)
plt.title("SHAP summary of St.John's")
plt.show()

shap.plots.scatter(shap_values[:, 'T2'], color=shap_values[:,'Month'],ymax=0.25, ymin=-0.15,
        show=False)
plt.title("T2 SHAP values")
plt.show()

shap.plots.scatter(shap_values[:, 'T2lag1'], color=shap_values[:,'Month'],ymax=0.25, ymin=-
        0.15, show=False)
plt.title("T2lag1 SHAP values")
plt.show()

explainer2 = shap.TreeExplainer(model_ym,X_ym_train,model_output='probability')
shap_values2 = explainer2(X_ym_train)

shap.initjs()

shap.plots.bar(shap_values2,max_display=13,show=False)
```

```
plt.title("SHAP summary of Yarmouth")
plt.show()

shap.plots.beeswarm(shap_values2,max_display=13,show=False)
plt.title("SHAP summary of Yarmouth")
plt.show()

shap.plots.scatter(shap_values2[:, 'T2'], color=shap_values2[:,'T2lag1'],ymax=0.25, ymin=-0.15,
        show=False)
plt.title("T2 SHAP values")
plt.show()

shap.plots.scatter(shap_values2[:, 'T2lag1'], color=shap_values2[:,'T2'],ymax=0.25, ymin=-0.15,
        show=False)
plt.title("T2lag1 SHAP values")
plt.show()

shap.plots.waterfall(shap_values2[25000,:],max_display = 13)

# @title Internal test St. John's
dtest_sj = xgb.DMatrix(X_sj_test,y_sj_test)
pred_proba_sj = model_sj.predict(dtest_sj)

# Convert to binary predictions
pred_sj = (pred_proba_sj >= 0.8).astype(int)

# Overall results
```

```
print('Overall')
print(classification_report(y_sj_test,pred_sj),'\n')

# @title Internal test Yarmouth
dtest_ym = xgb.DMatrix(X_ym_test,y_ym_test)
pred_proba_ym = model_ym.predict(dtest_ym)

# Convert to binary predictions
pred_ym = (pred_proba_ym >= 0.8).astype(int)

# Overall results
print('Overall')
print(classification_report(y_ym_test,pred_ym),'\n')

# @title 2024 Test St. John's
sj_2024 = dd.read_csv('testStjohn*.csv',dtype={'QC': 'float64','Day': 'float64',
      'Hour': 'float64',
      'Month': 'float64'})

# Compute the Dask DataFrame to get a Pandas DataFrame
df2024_sj = sj_2024.compute()
# Remove rows where the first column is blank
test2024_sj = df2024_sj[df2024_sj['Vis'].notnull()]

test2024_sj = test2024_sj.set_index('Time')

X_sj_2024, y_sj_2024 =
```

```
    test2024_sj[['T2','T2lag1','U','Ulag1','V','Vlag1','RH2','RH2lag1','P','Plag1','Month','Day','
    Hour']], test2024_sj[['Vis']]

# Encode y to numeric
y_sj_2024 = OrdinalEncoder(categories=[['clear','fog']]).fit_transform(y_sj_2024)

d2024_sj = xgb.DMatrix(X_sj_2024)

pred_proba_2024sj = model_sj.predict(d2024_sj)

# Convert to binary predictions
pred2024_sj = (pred_proba_2024sj >= 0.8).astype(int)

# Overall results
print('Overall')
print(classification_report(y_sj_2024,pred2024_sj),'\n')

# @title 2024 Test Yarmouth
ym_2024 = dd.read_csv('testYarmouth*.csv',dtype={'QC': 'float64','Day': 'float64',
    'Hour': 'float64',
    'Month': 'float64'})

# Compute the Dask DataFrame to get a Pandas DataFrame
df2024_ym = ym_2024.compute()
# Remove rows where the first column is blank
test2024_ym = df2024_ym[df2024_ym['Vis'].notnull()]
```

```
test2024_ym = test2024_ym.set_index('Time')


X_ym_2024, y_ym_2024 =
        test2024_ym[['T2','T2lag1','U','Ulag1','V','Vlag1','RH2','RH2lag1','P','Plag1','Month','Day',
        'Hour']], test2024_ym[['Vis']]


# Encode y to numeric
y_ym_2024 = OrdinalEncoder(categories=[['clear','fog']]).fit_transform(y_ym_2024)


d2024_ym = xgb.DMatrix(X_ym_2024)


pred_proba_2024ym = model_ym.predict(d2024_ym)


# Convert to binary predictions
pred2024_ym = (pred_proba_2024ym >= 0.8).astype(int)


# Overall results
print('Overall')
print(classification_report(y_ym_2024,pred2024_ym),'\n')
```