Predicting Winter Fog over Complex Terrain using Machine Learning

Grace Liu
University of Utah

UUCS-24-002

School of Computing
University of Utah
Salt Lake City, UT 84112 USA

16 April 2024

Abstract

Fog forms in high-elevation complex terrain as frequently as it does over bodies of water but is less understood and harder to predict. Forecasting winter cold fog over complex terrain is particularly difficult due to the complex interactions between land, water, snow cover, and atmospheric conditions in the process of fog formation. Traditional physical and numerical models have a limited ability to represent various conditions associated with fog formation; thus, fog prediction remains a challenge in weather prediction. This study aims to evaluate the effectiveness of machine learning methods in predicting winter fog over complex terrain, specifically the city of Heber in northern Utah. We utilize 10 years of surface meteorological observations. Emphasis will be placed on examining various baseline methods for their effectiveness in machine learning to help produce meaningful forecasts.

PREDICTING WINTER FOG OVER COMPLEX TERRAIN USING MACHINE LEARNING

by

Grace Liu

A Senior Honors Thesis Submitted to the Faculty of
The University of Utah
In Partial Fulfillment of the Requirements for the

Honors Degree in Bachelor of Science

In

Data Science

Approved:

Zhaoxia Pu

Zhaoxia Pu, PhD
Thesis Faculty Supervisor

Mary Hall

Mary Hall, PhD
Director, School of Computing

Thomas C. Henderson

Thomas Henderson, PhD
Honors Faculty Advisor

Monisha Pasupathi, PhD
Dean, Honors College

May 2024

ABSTRACT

Fog forms in high-elevation complex terrain as frequently as it does over bodies of water but is less understood and harder to predict. Forecasting winter cold fog over complex terrain is particularly difficult due to the complex interactions between land, water, snow cover, and atmospheric conditions in the process of fog formation. Traditional physical and numerical models have a limited ability to represent various conditions associated with fog formation; thus, fog prediction remains a challenge in weather prediction. This study aims to evaluate the effectiveness of machine learning methods in predicting winter fog over complex terrain, specifically the city of Heber in northern Utah. We will utilize 10 years of surface meteorological observations. Emphasis will be placed on examining various baseline methods for their effectiveness in machine learning to help produce meaningful forecasts.

TABLE OF CONTENTS

INTRODUCTION

Fog is defined by low visibility (less than 1 km according to the glossary of the American Meteorological Society) and can significantly impact the safety of aviation and other outdoor traffic[1]. The formation of fog is a complex process and involves interactions ranging from microphysical to planetary-scale[2,3]. Due to the complicated factors that contribute to the formation of fog, it is difficult to forecast fog accurately[4].

Fog forecasting remains a challenge in weather prediction because traditional physical and numerical models have a limited ability to represent various conditions associated with fog formation. Various machine learning methods have been applied to visibility prediction such as decision tree induction[5], tree-based machine learning methods[6], random forest and K-nearest neighbor methods[7]. Deep learning methods in particular have shown to be promising for fog prediction[8].

The framing of the prediction problem has implications for the explainability of models, the most appropriate data pre-processing, and specific algorithms used. Physics-informed machine learning can result in more robust models[9], but experiments ranging from a deep hybrid model[10] to explicable forecasting approaches that could be integrated into deep learning techniques[11] have produced reasonable results. The prediction problem can be framed as regression or classification, which influences what pre-processing is most effective[12]. Deep learning ensembles can outperform individual models as investigated in this survey of deep learning ensembles[13].

As for forecasting and understanding fog, new efforts have been made to utilize deep learning to predict fog[14–16] and other meteorological variables[17–19] in addition to ongoing efforts to understand the physical science of fog[2,20]. Previous work has applied deep learning to specific types of fog such as coastal fog[21] and sea fog[22–24]. However, few studies have been done on fog that forms below 0°C over complex terrain, especially mountainous cold fog.

Meteorological data records are commonly time series data. Recurrent neural networks (RNNs) and long short-term memory (LSTM) models are particularly well-suited to time series data. LSTMs have been used to predict COVID-19 cases[25,26]. Further variants of RNNs have been used for visibility prediction such as transductive LSTMs[27] and autoregressive RNNs[28]. Although LSTMs have not been applied to MesoWest data for visibility prediction, they have been used to predict other variables[29–31]. For our methods, we also focus on winter fog at stations with complex terrain in Utah as studied by the Cold Fog Amongst Complex Terrain campaign[32].

This thesis work aims to apply modern machine learning techniques to surface meteorological observations from MesoWest to gain familiarity with the data, various techniques for time series analysis, and a variety of machine learning methods. We present the results of baseline testing used for nowcasting to examine how well surface meteorological data can be used to identify the occurrence of fog.

METHODS AND DATA

The goal of this project is to apply a variety of machine learning techniques to surface meteorological observations from MesoWest for the nowcasting of fog, revealing the meteorological processes important to the formation of mountainous cold fog and the variables most crucial to accurate detection of such fog. Model options include linear classifiers, probabilistic methods, decision tree-based methods, and neural networks. Naive forecasts are used as a baseline model to compare more complex models against.

2.1. Data

Meteorological data was obtained from the MesoWest station at the Heber Valley Airport (KHCR) with the aim of studying fog over complex terrain. The data was downloaded through Synoptic data (https://download.synopticdata.com/). 10 years of data ranging from 2012-2023 were downloaded then filtered to winter seasons by only using data from December-February, resulting in ten winter seasons. This resulted in a total of 63,517 rows of data measured at roughly 20-minute intervals for the Heber station.

After determining what variables were usable, the data was pre-processed by fixing formatting inconsistencies and sea level pressure data that was erroneously swapped with pressure data. We attempted to use data from the Community Collaborative Rain, Hail and Snow (CoCoRaHS) Network to provide precipitation data at the Heber Valley site as precipitation can be a useful variable to consider when forecasting fog, but the data became very sparse when matched spatially with the MesoWest data, so we ran the baseline tests without precipitation data. Having precipitation data available also would have enabled the exclusion of precipitation fog, which forms via different mechanisms from radiation fog, one of the primary types of fog studied by CFACT.

Various normalization methods were applied to normalize the data into equally important features. Results are presented for no normalization, Gaussian standardization, and min-max scaling. For the experiments presented, four variables were used to make predictions: air temperature, relative humidity, pressure, and wind speed. This set of variables is commonly used in operational forecasting and represents a robust set of features that describe the state of the atmosphere.

2.2. Nowcasting Methodology

All models will classify sequences of data as “foggy” or “not foggy” at various time intervals into the future. Nowcasts (predicting the label at the end of the sequence) in addition to 1-, 6-, and 24-hour forecasts can be predicted; results for nowcasting are presented. For these results, each model uses sequences with a length of 1, meaning the models predict whether there was fog present or not for one row of data, which are spaced roughly 20 minutes apart. Deep learning results are not presented, but models designed for sequential data such as RNNs and LSTMs can easily use longer sequence lengths in order to better learn from the temporal patterns in time series data. The results will be evaluated against either standard National Weather Service definitions of fog or relaxed thresholds to address imbalances in the data.

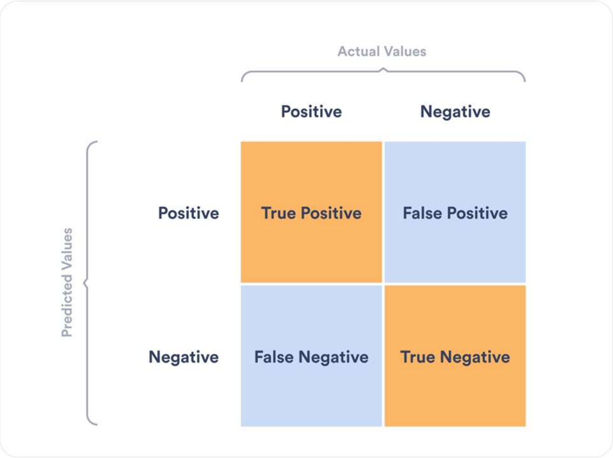
For our most comprehensive baseline tests, a “near fog” subset of the data is used. Near fog is classified when the visibility is below 5 kilometers and relative humidity exceeds 70%. Fog is then classified by visibility below 3 kilometers and relative humidity exceeding 80%. This can be viewed as a realistic forecasting situation where meteorologists might only create forecasts when conditions look promising for fog. This classification of data results in 3066 “near fog” and 2966 “fog” data points, nearly a 50:50

ratio, and can potentially reveal what meteorological conditions lead to fog. The near fog subset of the data was split into training and validation splits of roughly equal size.

2.3. Evaluation

Model performance should not be evaluated purely on prediction accuracy, especially when the data is unbalanced. The majority of the data does not detect fog, so predicting the most common or baseline label will yield a high accuracy. As an alternative, the accuracy could be calculated only for time steps labeled as foggy. This accuracy is called the true positive (TP) or hit rate.

In a confusion matrix, the predictions for each time step are grouped based on the actual and predicted label. Many forecasting metrics are derived from this basic display of results. False alarm ratio (FAR, FP/(FP+TN)), probability of detection (POD, TP/(TP+FN)), and critical success index (CSI, TP/(TP+FN+FP)) are often used in forecasting and provide more information than accuracy alone. For regression methods that output an actual visibility prediction that is thresholded to result in a predicted label, root mean square error (RMSE), mean absolute error (MAE), and coefficient of determination (R^2) can be calculated.



		Actual Values	
		Positive	Negative
Predicted Values	Positive	True Positive	False Positive
	Negative	False Negative	True Negative

Fig. 1. Confusion matrix.

Finally, we also use Area Under the Receiver Operating Characteristic Curve (ROC AUC) to evaluate the experiments. ROC AUC is the area under the Receiver Operating Characteristics curve, which plots the TP rate against FP rate. A score of 1 corresponds to perfectly correct performance, 0 perfectly incorrect, and 0.5 a model with no discriminative power between the positive and negative class. We will take a score between 0.5 and 0.7 to be poor, 0.7 and 0.8 acceptable, 0.8 and 0.9 excellent, and 0.9 and 1.0 outstanding. Mean average precision (mAP) is a related metric that is less sensitive to extremely imbalanced data. mAP values can also range from 0 to 1, with 1 being a perfect score.

2.4. Methods

The prediction problem can be formulated as regression or classification. An overview of methods is provided below.

2.4.1. Linear Regression

Ordinary least squares linear regression is one of the simplest models with straightforward calculations and interpretable results. Multiple linear regression creates a linear model for one target variable with multiple input variables. Linear methods should be used as a baseline to determine whether there are simple linear relationships in the data before moving onto more complex models.

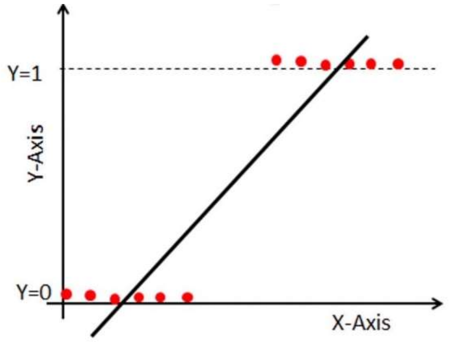


Fig. 2. Example of a linear regression model.

2.4.2. Gaussian Processes

Gaussian process regression and classification are both probabilistic methods. Gaussian process classification directly produces a class label for the data and produces slightly different results from Gaussian process regression.

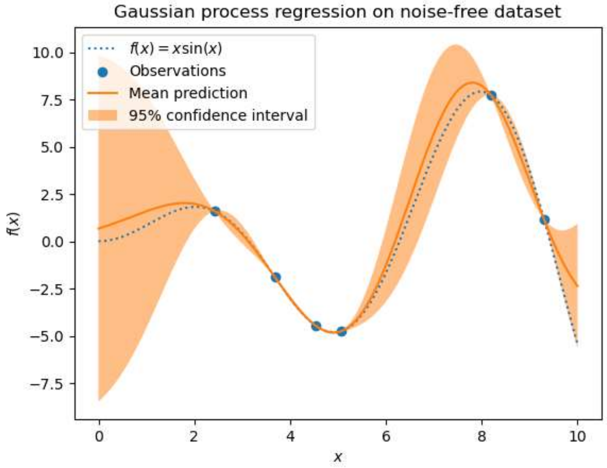


Fig. 3. Visualization of Gaussian process regression.

2.4.3. Logistic Regression

Logistic regression can be used for binary classification when labels are desired instead of regressed values. Instead of ordinary least squares, logistic regression uses maximum likelihood estimation to predict the likelihood of each label.

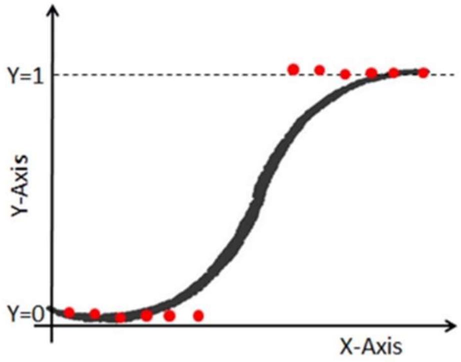


Fig. 4. Example of logistic regression applied to the same data as Fig. 2.

2.4.4. Tree-Based Methods

A decision tree represents a decision-making process, with each node representing a test of the data until a decision or label is reached. While individual decision trees have limited capabilities and are prone to overfitting, ensemble methods like random forests greatly reduce the likelihood of overfitting and produce robust models. XGBoost is a highly optimized tree-based library that can handle large amounts of data.

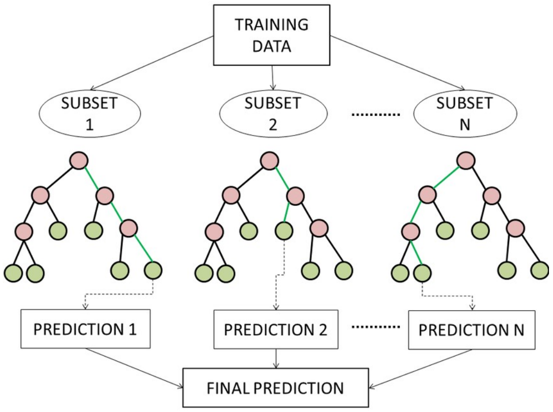


Fig. 5. Visualization of a random forest classifier, which consists of an ensemble of decision trees.

2.4.5. Deep Learning

RNNs are used to detect patterns in sequential or time series data. LSTMs and gated recurrent unit (GRU) models have architectures better suited to learning long-term relationships[33] such as the physical laws that govern the atmosphere. Basic feedforward networks can be used to test whether recurrent networks are able to learn temporal patterns from the data better than more basic models.

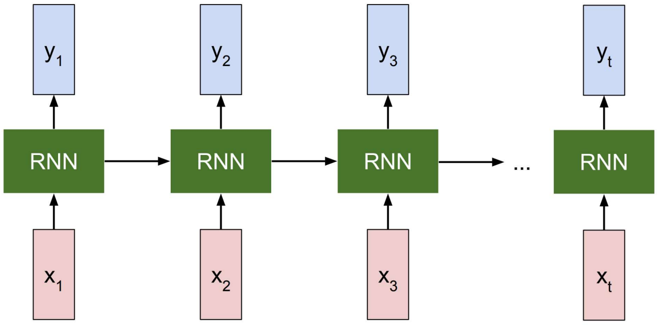


Fig. 6. Visualization of RNN architecture.

RESULTS

The validation results for ten models will be presented: a baseline model that predicts the majority label, linear regression, Gaussian process regression (GPR), GPR without normalizing the target variable, Gaussian process classification (GPC), logistic regression, random forest classifier with an optimized max depth, XGBoost with default parameters, XGBoost with parameters found in the "Get Started" guide (https://xgboost.readthedocs.io/en/stable/get_started.html), and XGBoost with hyperparameters optimized by the Bayesian Optimization package in Python.

Presented in the tables below are confusion matrix entries (true negatives, false positives, false negatives, true positives), validation accuracy, probability of detection, false alarm ratio, critical success index, and ROC AUC for each model.

TABLE I

BASELINE RESULTS: NO NORMALIZATION

	TN	FP	FN	TP	VA	PoD	FAR	CSI	ROC AUC
Majority	1646	0	1362	0	0.547	0	0	0	0.5
linr	870	776	356	1006	0.624	0.739	0.471	0.471	-
gpr	832	814	310	1052	0.626	0.772	0.495	0.483	-
gpr_nonorm	716	930	179	1183	0.631	0.869	0.565	0.516	-
gpc	964	682	615	747	0.569	0.548	0.414	0.365	0.555
lr	0	1646	0	1362	0.453	1.000	1.000	0.453	0.493
rfc_7	754	892	156	1206	0.652	0.885	0.542	0.535	0.714
xgboost_def	1081	565	639	723	0.600	0.531	0.343	0.375	0.695
xgboost_getstarted	688	958	105	1257	0.647	0.923	0.582	0.542	0.711
xgboost_opt	683	963	94	1268	0.649	0.931	0.585	0.545	0.712

TABLE II

BASELINE RESULTS: GAUSSIAN STANDARDIZATION

	TN	FP	FN	TP	VA	PoD	FAR	CSI	ROC AUC
Majority	1646	0	1362	0	0.547	0	0	0	0.5
linr	870	776	356	1006	0.624	0.739	0.471	0.471	-
gpr	1024	622	569	793	0.604	0.582	0.378	0.400	-
gpr_nonorm	1024	622	569	793	0.604	0.582	0.378	0.400	-
gpc	1098	548	560	802	0.632	0.589	0.333	0.420	0.709
lr	1118	528	564	798	0.637	0.586	0.321	0.422	0.670
rfc_7	755	891	159	1203	0.651	0.883	0.541	0.534	0.714
xgboost_def	1081	565	639	723	0.600	0.531	0.343	0.375	0.695
xgboost_getstarted	688	958	105	1257	0.647	0.923	0.582	0.542	0.711
xgboost_opt	683	963	94	1268	0.649	0.931	0.585	0.545	0.712

TABLE III

BASELINE RESULTS: MIN-MAX SCALING

	TN	FP	FN	TP	VA	PoD	FAR	CSI	ROC AUC
Majority	1646	0	1362	0	0.547	0	0	0	0.5
linr	870	776	356	1006	0.624	0.739	0.471	0.471	-
gpr	970	676	469	893	0.619	0.656	0.411	0.438	-
gpr_nonorm	971	675	473	889	0.618	0.653	0.410	0.436	-
gpc	1049	597	436	926	0.657	0.680	0.363	0.473	0.705
lr	1121	525	565	797	0.638	0.585	0.319	0.422	0.669
rfc_7	756	890	158	1204	0.652	0.884	0.541	0.535	0.714
xgboost_def	1081	565	639	723	0.600	0.531	0.343	0.375	0.695
xgboost_getstarted	688	958	105	1257	0.647	0.923	0.582	0.542	0.711
xgboost_opt	683	963	94	1268	0.649	0.931	0.585	0.545	0.712

DISCUSSION

4.1. Linear Regression

For linear regression, the data can be visually inspected for linear correlations between the predictors and response variables. The correlations for individual predictors are weak, as fog formation does not depend strongly on any one variable.

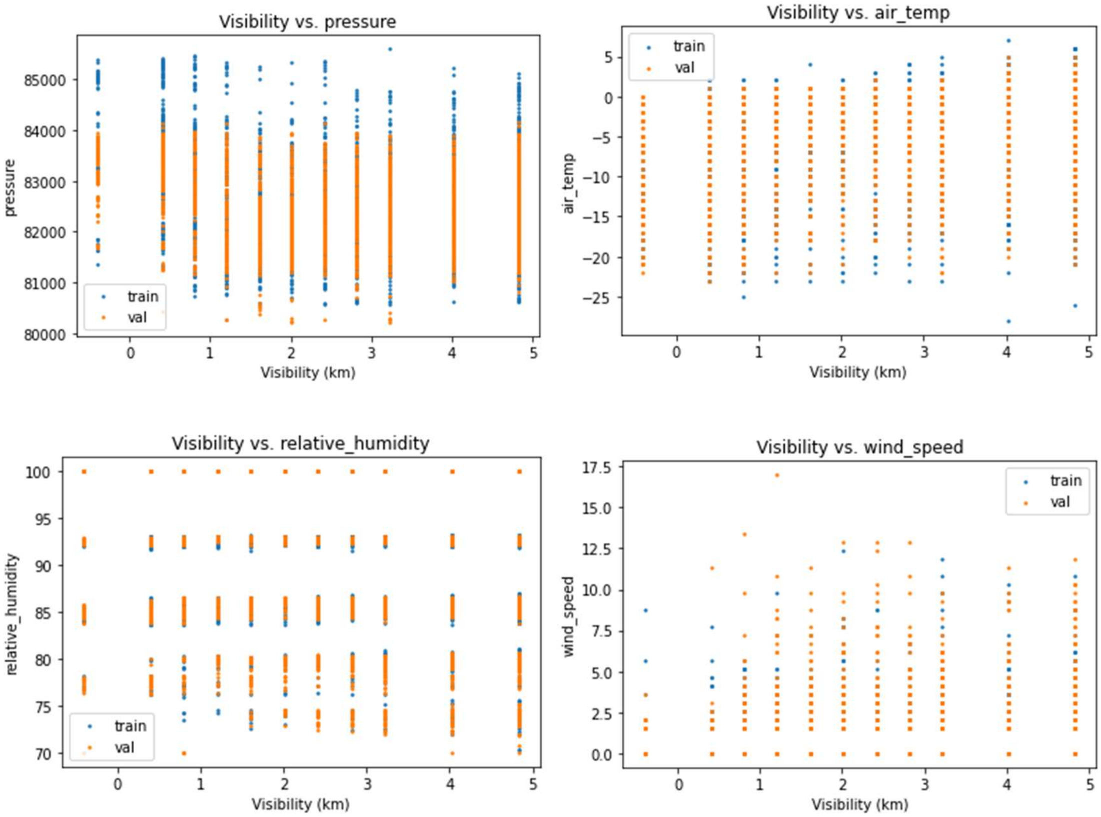


Fig. 7. Visibility plotted against the four feature variables. No standardization was used.

TABLE IV

MULTIPLE LINEAR REGRESSION COEFFICIENTS

Pressure	Air temp	RH	Wind speed
-0.0000773	0.0954	-0.0370	-0.0442

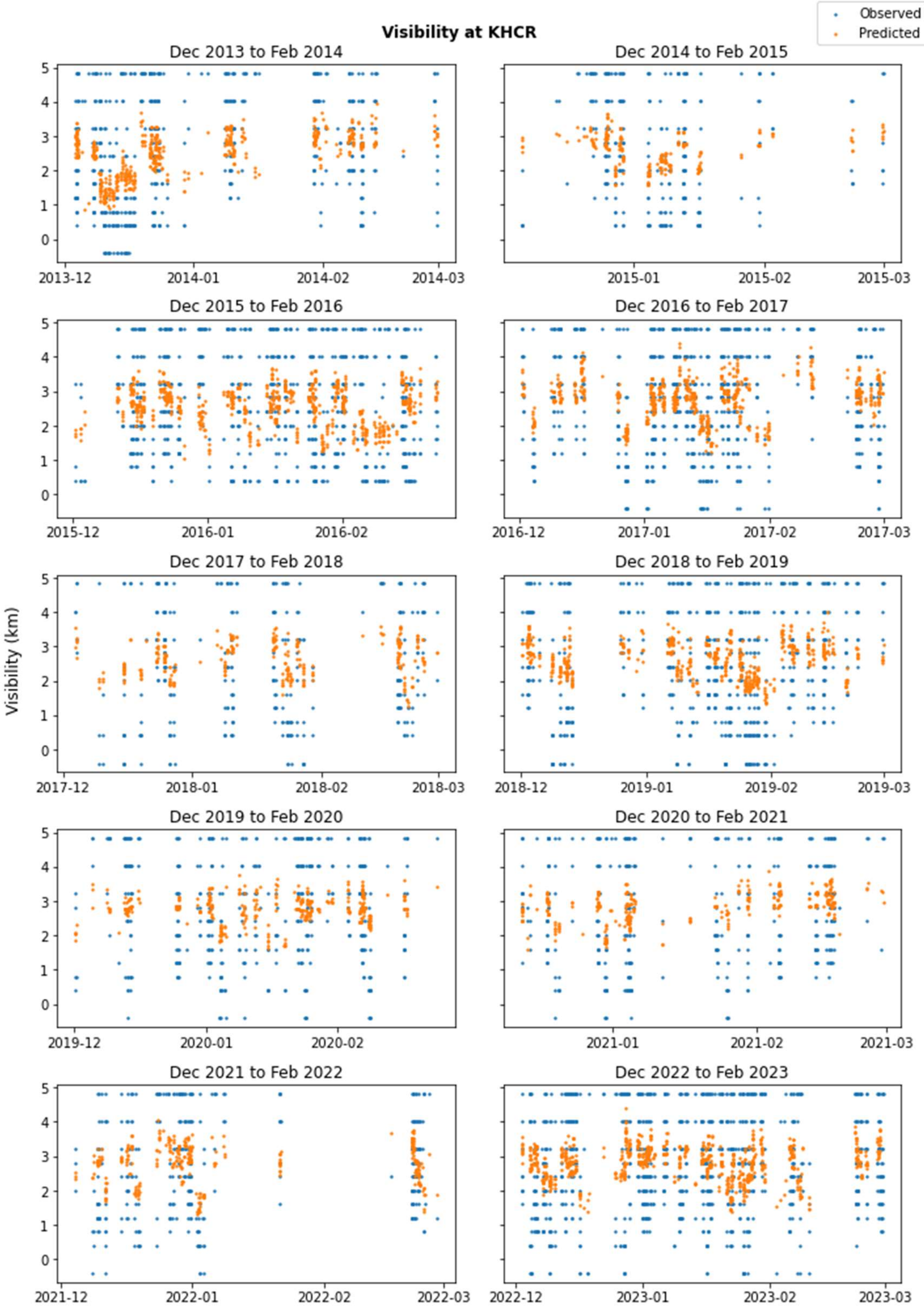


Fig. 8. Observed vs. predicted visibility for linear regression model. The first five winter seasons were used for training and next five for validation.

Additionally, the visibility predictions from the regression models can be visually compared against the true observed visibility. As expected, the performance and coefficients of linear regression do not change based on the normalization used due to the scale invariance of linear regression.

4.2. Gaussian Process Regression

Unlike linear regression, both Gaussian processes are sensitive to the normalization used. GPR yielded the highest CSI and validation accuracy with no normalization and without setting the method to standardize the predicted variable. With this configuration, the model predicted several erroneously low visibility values for the validation features. The visibility in the original MesoWest dataset contained negative values that we used as is, so scaling the data to have a minimum of 0 and the same maximum could have potentially reduced this specific error.

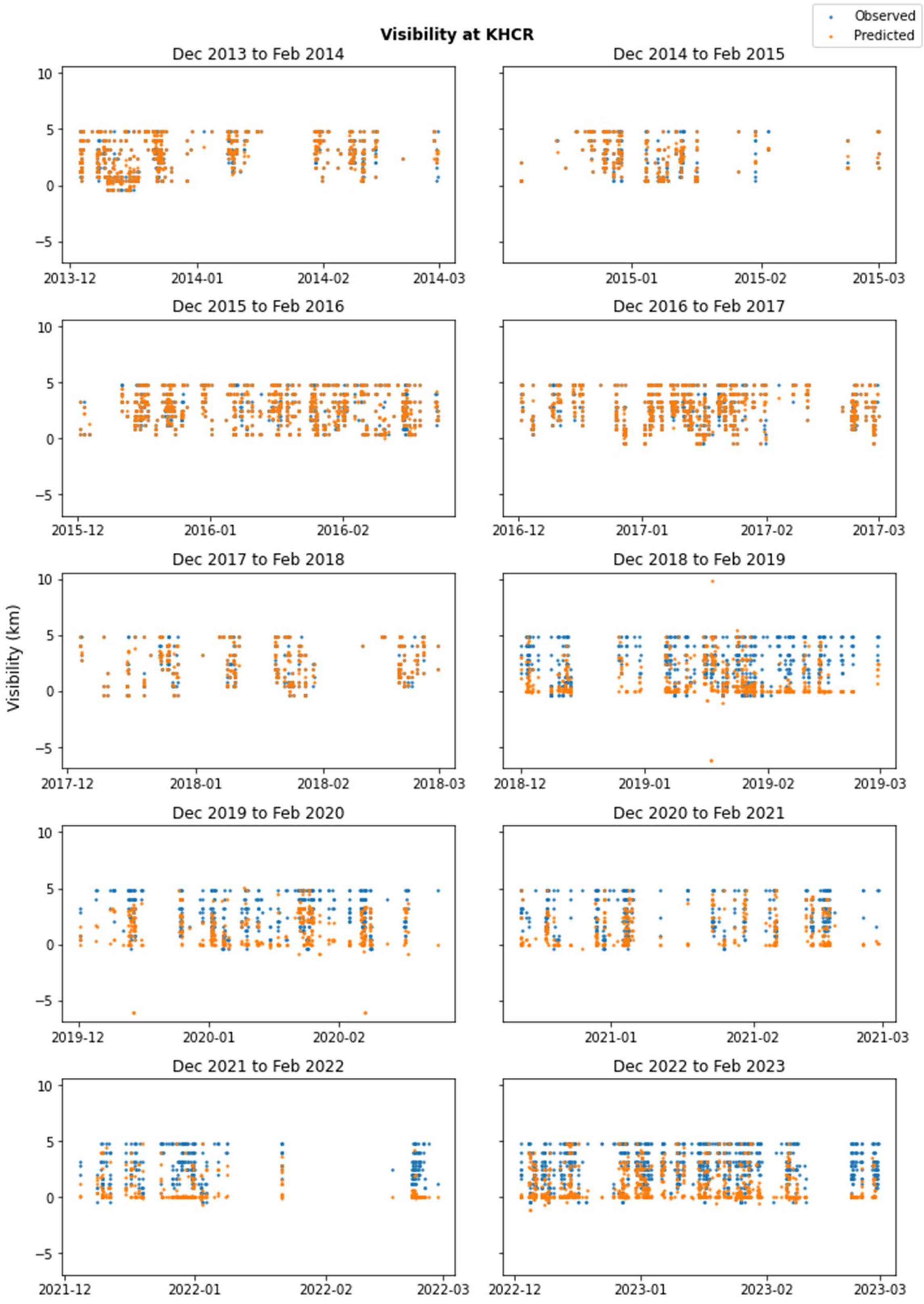


Fig. 9. Observed vs. predicted visibility for GPR.

Because they do not provide probability estimates for each class, only predicted values for visibility that are then thresholded to produce class labels, the regression models do not receive ROC AUC scores. However, other metrics such as Pearson correlation, MSE, and MAE can be calculated.

TABLE V

REGRESSION METRICS FOR VALIDATION PREDICTIONS

	Pearson correlation	MSE	MAE
Linear regression	0.290	2.18	1.27
GPR	0.0186	7.03	2.20

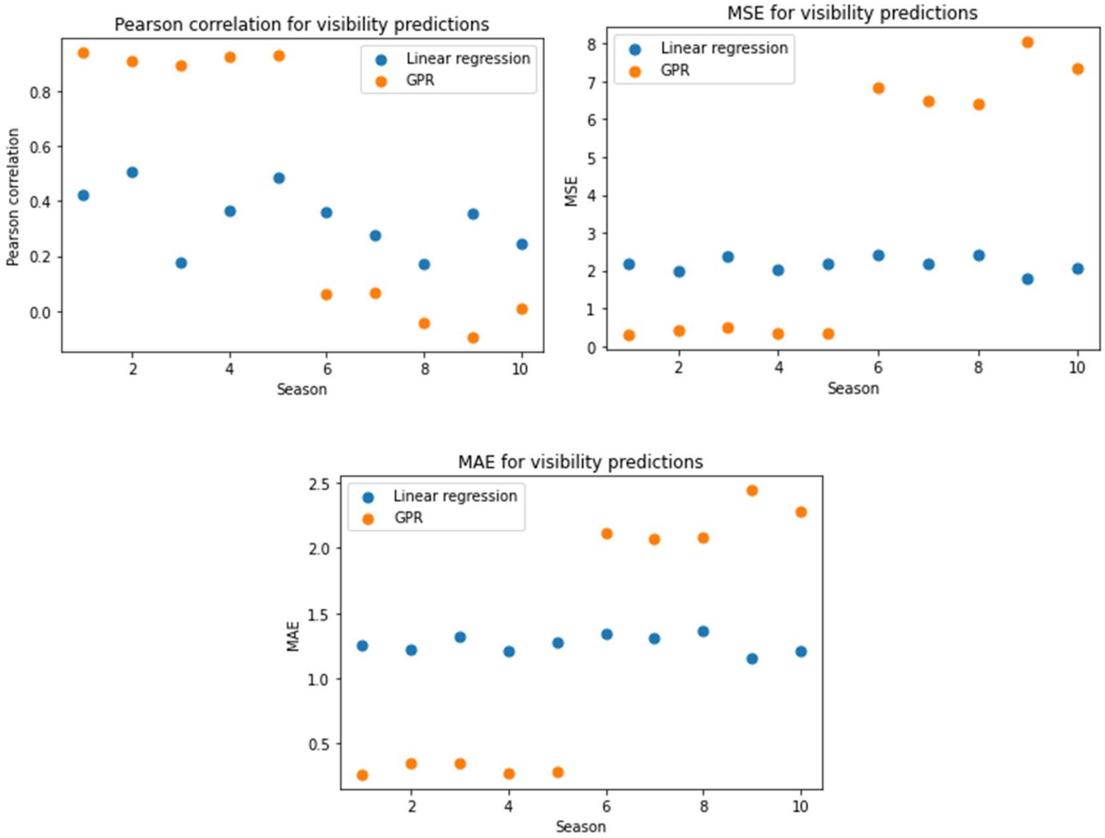


Fig. 10. Regression metrics calculated by season.

As seen in Fig. 10, GPR overfits to the training data and, as a result, performs more poorly than linear regression on the validation data. However, GPR has a higher accuracy after the predicted values are used to classify fog. These two regression models have different strengths, whether predicting more realistic visibility values or discriminative power between fog and near fog. To improve on the basic GPR model tested here, stricter penalties or ensembles could be used to reduce overfitting.

4.3. Gaussian Process Classification

GPC results in the more accurate classification when the data is normalized, with higher probability of detection and CSI but also slightly higher false alarm ratio under min-max scaling. The ROC AUC score is comparable to that of the decision tree-based methods, which have the highest scores.

4.4. Logistic Regression

Logistic regression is also sensitive to the normalization scheme due to the default L2 regularization used. Without normalization, the penalty causes the model to classify all examples as foggy. This model receives a low CSI but outperforms logistic regression with normalization, highlighting a bias toward positive classifications. The results of logistic regression are very similar for the two normalization types used.

4.5. Random Forest Classification

The results for a single random forest classifier are presented for each type of data normalization. These three classifiers had nearly identical results with differences likely due to small numerical instabilities, as the random state was set to be the same for each normalization experiment. The default number of trees of 100 was used, and the max depth was tuned for each type of normalization. This hyperparameter search can be visualized using validation curves like the following. All three validation curves were nearly identical due to scale invariance and result in the max depth set to 7 nodes.

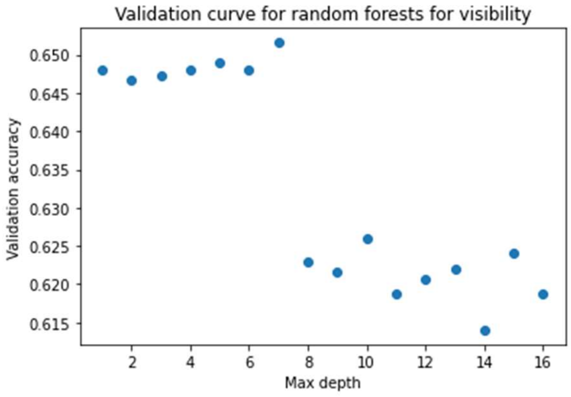


Fig. 11. Validation curve for RFC with no normalization.

The following figure shows validation curves for the unbalanced, unfiltered data, where 6.78% of the data had visibility below 3km and RH above 70%. The most relaxed visibility threshold of 10km resulted in a more archetypal validation curve where increasing the expressivity is beneficial at lower max depths but overfitting takes over at some point in the search. The best max depth falls at around 4 nodes.

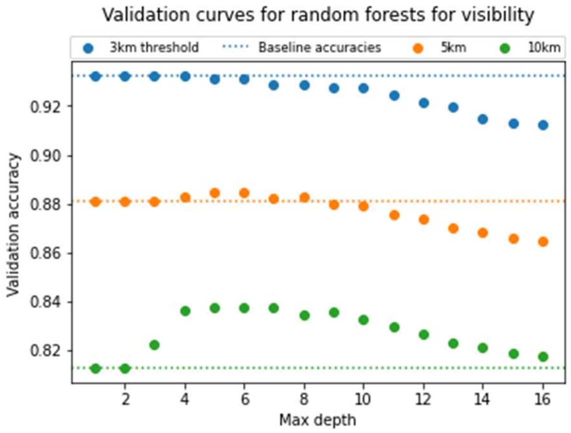


Fig. 12. Validation curves for various thresholds over the full dataset.

To examine the difference between a max depth of 3 and 4, their ROC curves and AUC scores can be visualized, with logistic regression shown for comparison. A perfect

ROC curve would form a 90° angle at the top-left corner, resulting in an area under the curve of 1.

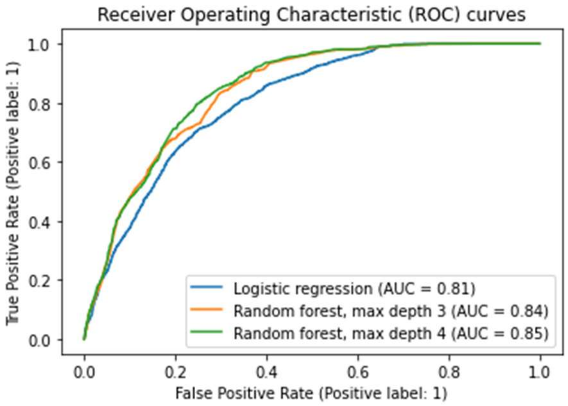


Fig. 13. ROC curves for 3 classifiers over the full dataset.

4.6. XGBoost

XGBoost utilizes gradient boosted decision trees and can be used to obtain a powerful baseline for any dataset due to its efficiency and relative ease of use. The models xgboost_def and xgboost_getstarted were trained using all default parameters and parameters used in the tutorial on the XGBoost website, “Get Started with XGBoost,” respectively. They are provided as a comparison to the third model xgboost_opt, which was trained using the hyperparameters selected by Bayesian Optimization.

For this model, the number of estimators, max depth, learning rate, and min_split_loss (minimum loss required to create a split in the tree) were tuned by an optimizer with the number of random initial points set to 5 and further steps of optimization to 25. As the same random seed was set for the three types of normalization, the same hyperparameters were used for essentially identical results. Interestingly, the CSI and ROC AUC for the optimized XGBoost model were barely higher than the model based on the quickstart tutorial.

TABLE VI

PARAMETERS USED FOR XGBOOST MODELS

	n_estimators	max_depth	learning_rate	min_split_loss
xgboost_def	1	6	0.3	0
xgboost_getstarted	2	2	1	0
xgboost_opt	762	2	0.01	5.0

CONCLUSIONS

Fog formation is rooted in both microphysics and large-scale atmospheric dynamics, making it difficult to forecast by any currently used methods, including machine learning. However, the nonlinear expressiveness of machine learning may be able to better capture the chaotic behavior of fog and the atmosphere than current rule-based or numerical approaches and result in better atmospheric modeling.

In this study, 10 models were evaluated on a “near fog” subset of winter data at the Heber Valley Airport. Overall, the tree-based methods performed the best with the highest ROC AUC and CSI, and all models were able to improve on the unskilled ROC AUC of 0.5 of the baseline model. The best values of ROC AUC obtained fall into the low end of 0.7 and 0.8, which we take to be acceptable classification. The 10 models were also tested with longer sequence lengths and on forecasting into the future, but ROC AUC fell below acceptable values. The models would need to be separately tuned for this prediction problem in order to detect fog at the same acceptable levels, and this could result in more operationally useful models for the forecasting of cold fog. Finally, this study only uses surface-based meteorological observations. Future work should also explore the use of other meteorological data, such as atmospheric soundings, satellite observations, and NWP model outputs, as data sources.

In addition to a basic set of machine learning baselines, we explored various deep learning models. We used a basic feedforward network, vanilla RNN, and simple LSTM in addition to more extensive experimentation with LSTMs, all of which struggled to capture the imbalanced dataset. Future steps to improve on these models could include reformulating them for the filtered data with balanced classes. In computer science research, different metrics might be used, and the deep learning models would likely be run on a GPU to be able to run experiments more quickly and use longer sequence lengths to determine the ideal value. The models could be greatly expanded in size and expressiveness after carefully designing an architecture for the problem data and goals.

Moreover, more sophisticated architectures like attention-based transformers could be used. The PyTorch transformer module was prohibitively complex to use in this overview, but it would likely produce better results than the baseline models here or improved LSTM models. The data can also be processed using dimensionality reduction methods like PCA, methods to handle unbalanced data such as undersampling and oversampling, or artificial augmentation using generative adversarial network (GAN) models for more robust models.

ACKNOWLEDGEMENTS

This study is supported by the NSF-funded project Cold Fog Amongst Complex Terrain (CFACT, PI: Dr. Zhaoxia Pu). I would like to thank Drs. Zhaoxia Pu and Shandian Zhe for their guidance and expertise.

REFERENCES

1. Gultepe, I. *et al.* Fog Research: A Review of Past Achievements and Future Perspectives. *Pure Appl. Geophys.* **164**, 1121–1159 (2007).
2. Golding, B. W. A Study of the Influence of Terrain on Fog Development. *Mon. Weather Rev.* **121**, 2529–2541 (1993).
3. Pu, Z., Chachere, C. N., Hoch, S. W., Pardyjak, E. & Gultepe, I. Numerical Prediction of Cold Season Fog Events over Complex Terrain: the Performance of the WRF Model During MATERHORN-Fog and Early Evaluation. *Pure Appl. Geophys.* **173**, 3165–3186 (2016).
4. Pu, Z. Surface Data Assimilation and Near-Surface Weather Prediction over Complex Terrain. in *Data Assimilation for Atmospheric, Oceanic and Hydrologic Applications (Vol. III)* (eds. Park, S. K. & Xu, L.) 219–240 (Springer International Publishing, Cham, 2017). doi:10.1007/978-3-319-43415-5_10.
5. Bartoková, I., Bott, A., Bartok, J. & Gera, M. Fog Prediction for Road Traffic Safety in a Coastal Desert Region: Improvement of Nowcasting Skills by the Machine-Learning Approach. *Bound.-Layer Meteorol.* **157**, 501–516 (2015).

6. Kim, B.-Y., Belorid, M. & Cha, J. W. Short-Term Visibility Prediction Using Tree-Based Machine Learning Algorithms and Numerical Weather Prediction Data. *Weather Forecast.* **37**, 2263–2274 (2022).
7. Alhathloul, S. H., Mishra, A. K. & Khan, A. A. Low visibility event prediction using random forest and K-nearest neighbor methods. *Theor. Appl. Climatol.* (2023) doi:10.1007/s00704-023-04697-6.
8. Schultz, M. G. *et al.* Can deep learning beat numerical weather prediction? *Philos. Trans. R. Soc. Math. Phys. Eng. Sci.* **379**, 20200097 (2021).
9. Kashinath, K. *et al.* Physics-informed machine learning: case studies for weather and climate modelling. *Philos. Trans. R. Soc. Math. Phys. Eng. Sci.* **379**, 20200093 (2021).
10. Grover, A., Kapoor, A. & Horvitz, E. A Deep Hybrid Model for Weather Forecasting. in *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* 379–386 (Association for Computing Machinery, New York, NY, USA, 2015). doi:10.1145/2783258.2783275.
11. Peláez-Rodríguez, C., Marina, C. M., Pérez-Aracil, J., Casanova-Mateo, C. & Salcedo-Sanz, S. Extreme Low-Visibility Events Prediction Based on Inductive and Evolutionary Decision Rules: An Explicability-Based Approach. *Atmosphere* **14**, 542 (2023).
12. Castillo-Botón, C. *et al.* Machine learning regression and classification methods for fog events prediction. *Atmospheric Res.* **272**, 106157 (2022).
13. Peláez-Rodríguez, C. *et al.* Deep learning ensembles for accurate fog-related low-visibility events forecasting. *Neurocomputing* **549**, 126435 (2023).

14. Palvanov, A. & Cho, Y. I. VisNet: Deep Convolutional Neural Networks for Forecasting Atmospheric Visibility. *Sensors* **19**, 1343 (2019).

15. Fabbian, D., Dear, R. de & Lellyett, S. Application of Artificial Neural Network Forecasts to Predict Fog at Canberra International Airport. *Weather Forecast.* **22**, 372–381 (2007).

16. Miao, K. *et al.* Application of LSTM for short term fog forecasting based on meteorological elements. *Neurocomputing* **408**, 285–291 (2020).

17. Salman, A. G., Heryadi, Y., Abdurahman, E. & Suparta, W. Single Layer & Multi-layer Long Short-Term Memory (LSTM) Model with Intermediate Variables for Weather Forecasting. *Procedia Comput. Sci.* **135**, 89–98 (2018).

18. Qing, X. & Niu, Y. Hourly day-ahead solar irradiance prediction using weather forecasts by LSTM. *Energy* **148**, 461–468 (2018).

19. Yu, Y., Cao, J. & Zhu, J. An LSTM Short-Term Solar Irradiance Forecasting Under Complicated Weather Conditions. *IEEE Access* **7**, 145651–145666 (2019).

20. Gultepe, I. *et al.* A Review of Coastal Fog Microphysics During C-FOG. *Bound.-Layer Meteorol.* **181**, 227–265 (2021).

21. Kamangir, H., Krell, E., Collins, W., King, S. A. & Tissot, P. Importance of 3D convolution and physics on a deep learning coastal fog model. *Environ. Model. Softw.* **154**, 105424 (2022).

22. Xiang, Y. *et al.* Intelligent model for offshore China sea fog forecasting. Preprint at https://doi.org/10.48550/arXiv.2307.10580 (2023).

23. Zhou, Y., Chen, K. & Li, X. Dual-Branch Neural Network for Sea Fog Detection in Geostationary Ocean Color Imager. *IEEE Trans. Geosci. Remote Sens.* **60**, 1–17 (2022).

24. Jeon, H.-K., Kim, S., Edwin, J. & Yang, C.-S. Sea Fog Identification from GOCI Images Using CNN Transfer Learning Models. *Electronics* **9**, 311 (2020).

25. Kırbaş, İ., Sözen, A., Tuncer, A. D. & Kazancıoğlu, F. Ş. Comparative analysis and forecasting of COVID-19 cases in various European countries with ARIMA, NARNN and LSTM approaches. *Chaos Solitons Fractals* **138**, 110015 (2020).

26. Wang, P., Zheng, X., Ai, G., Liu, D. & Zhu, B. Time series prediction for the epidemic trends of COVID-19 using the improved LSTM deep learning method: Case studies in Russia, Peru and Iran. *Chaos Solitons Fractals* **140**, 110214 (2020).

27. Karevan, Z. & Suykens, J. A. K. Transductive LSTM for time-series prediction: An application to weather forecasting. *Neural Netw.* **125**, 1–9 (2020).

28. Jonnalagadda, J. & Hashemi, M. Forecasting Atmospheric Visibility Using Auto Regressive Recurrent Neural Network. in *2020 IEEE 21st International Conference on Information Reuse and Integration for Data Science (IRI)* 209–215 (2020). doi:10.1109/IRI49571.2020.00037.

29. Wen, L., Zhou, K., Yang, S. & Lu, X. Optimal load dispatch of community microgrid with deep learning based solar power and load forecasting. *Energy* **171**, 1053–1065 (2019).

30. Li, Y., Zhu, Z., Kong, D., Xu, M. & Zhao, Y. Learning Heterogeneous Spatial-Temporal Representation for Bike-Sharing Demand Prediction. *Proc. AAAI Conf. Artif. Intell.* **33**, 1004–1011 (2019).

31. Narmadha, S. & Vijayakumar, V. Spatio-Temporal vehicle traffic flow prediction using multivariate CNN and LSTM model. *Mater. Today Proc.* **81**, 826–833 (2023).

32. Pu, Z. *et al.* Cold Fog Amongst Complex Terrain. *Bull. Am. Meteorol. Soc.* **1**, (2023).

33. Hochreiter, S. & Schmidhuber, J. Long Short-Term Memory. *Neural Comput.* **9**, 1735–1780 (1997).

Name of Candidate: Grace Liu

Date of Submission: April 23, 2024