

Air Force Institute of Technology

AFIT Scholar

Faculty Publications

2023

Atmospheric Meteorological Effects on Forecasting Daily Lightning Occurrence at Cape Canaveral Space Force Station

Jon Saul [*]

Air Force Institute of Technology

Torrey J. Wagner

Air Force Institute of Technology

Eric G. Mbonimpa

Air Force Institute of Technology

Brent T. Langhals

Air Force Institute of Technology

Follow this and additional works at: <https://scholar.afit.edu/facpub>



Part of the [Meteorology Commons](#)

Recommended Citation

Saul, Jon [*]; Wagner, Torrey J.; Mbonimpa, Eric G.; and Langhals, Brent T., "Atmospheric Meteorological Effects on Forecasting Daily Lightning Occurrence at Cape Canaveral Space Force Station" (2023). *Faculty Publications*. 1404.

<https://scholar.afit.edu/facpub/1404>

This Conference Proceeding is brought to you for free and open access by AFIT Scholar. It has been accepted for inclusion in Faculty Publications by an authorized administrator of AFIT Scholar. For more information, please contact AFIT.ENWL.Repository@us.af.mil.

Atmospheric Meteorological Effects on Forecasting Daily Lightning Occurrence at Cape Canaveral Space Force Station

Jon Saul
*Data Analytics Certificate
Program*
*Graduate School of Engineering
and Management*
Air Force Institute of Technology
Wright-Patterson AFB, OH
jon.saul.1@spaceforce.mil

Torrey Wagner*
*Data Analytics Certificate
Program*
*Graduate School of Engineering
and Management*
Air Force Institute of Technology
Wright-Patterson AFB, OH
torrey.wagner@afit.edu
**corresponding author*

Eric Mbonimpa
*Data Analytics Certificate
Program*
*Graduate School of Engineering
and Management*
Air Force Institute of Technology
Wright-Patterson AFB, OH
eric.mbonimpa@afit.edu

Brent Langhals
*Data Analytics Certificate
Program*
*Graduate School of Engineering
and Management*
Air Force Institute of Technology
Wright-Patterson AFB, OH
brent.langhals@afit.edu

Abstract— As the Cape Canaveral Space Force Station and Kennedy Space Center increase their launch rate, any process that could assist in the automation of the currently-manual lightning forecast would be valuable. This work examines the possibility of machine-learning assistance with the daily lightning forecast which is produced by the 45th Weather Squadron. A dataset consisting of 34 lightning, pressure, temperature and windspeed measurements taken from 334 daily weather balloon (rawinsonde) launches in the timeframe 2012-2021 was examined. Models were created using recursive feature elimination on logistic regression and XGClassifier algorithms, as well as Bayesian and bandit optimization of neural network (NN) hyperparameters. The modeling process was repeated after eliminating 13 features related to windspeed. The best performing models on both datasets were the optimized NN models, with an F1 metric of 0.79 on the full dataset and 0.66 on the reduced dataset. For comparison, a model that predicted randomly achieved $F1 = 0.47$ on this dataset. The addition of 13 windspeed-related features more than doubled the complexity of the 21-feature no-wind model while increasing model performance by 13 percentage points. A notable inference from the statistical modeling is that the most important feature from both datasets was the Thompson convective index, which is related to temperature, dewpoint, relative humidity and lapse rate.

Keywords: *lightning forecast, launch operations, neural network, lightning indices*

Regular research paper

I. INTRODUCTION & BACKGROUND

Lightning hazard is a major safety concern at Cape Canaveral Space Force Station (CCSFS) and Kennedy Space Center (KSC) Florida. This eastern range (ER) is located in the thunderstorm capital of the United States. The 45th Weather Squadron (WS) currently provides weather support

to both governmental and commercial launches at CCSFS and KSC. Rawinsondes, balloon-borne meteorological sensors, collect data daily at 0500/0600 local time and 1700/1800 local time (1000 and 2200 Zulu time/UTC). The dataset included lightning occurrence, meteorological characteristics such as wind speed, lapse rate, relative humidity, temperature, and thermodynamic and kinematic indices such as Convective Available Potential Energy (CAPE) & K-index. With the recent significant increases in the ER launch rate, the 45th WS forecasting methods will need to transition to automated products, where applicable. The objective is to reduce required labor by automating lightning forecasts. The forecasted lightning can either be cloud-to-ground lightning, or intra-cloud lightning which does not reach the ground.

A. Literature Review

Supervised machine learning algorithms, such as neural networks, random forests, logistic regression, and decision trees have shown promising improvements in lightning prediction accuracy. Additionally, data from many of the sensors used in this work have been used to forecast the probability of lightning activity in previous studies [1]. Studies can be differentiated based on various factors, including their primary goal (classification or prediction with specific lead time), the utilized features, identification of important features, and the spatiotemporal characteristics of the dataset.

Bates et al.'s study at six locations in Australia found that logistic regression outperformed other classifiers, such as random forests, linear discriminant analyses and quadratic discriminant analyses. The results demonstrated the superior predictive ability of logistic regression in distinguishing between non-lightning and lightning days using large-scale atmospheric variables. The identified predictors primarily

included measures of instability, lifting potential, and atmospheric water content. The logistic regression was able to achieve hit rates above 90%. Other metrics used, such as false-alarm ratio and AUC were also improved with the logistic regression model [2]. It is unknown how this model compares to a randomly-predicting model on their dataset.

Zhu et al. developed a support vector machine algorithm that classified cloud-to-ground and intracloud lightning with an accuracy of 97% at a site in Argentina, outperforming existing lightning detection networks [3]. The data used originated from the Cordoba Marx Meter Array across 4 days in 2018, consisting of 10 electric field change sensors with fast and slow channels. The data used for training and testing the classification model is the fast-channel waveforms, which can detect small-amplitude pulses and provide a full range of amplitudes for model training. The significance of accurately classifying lightning discharges as cloud-to-ground or intracloud in this paper lies in ensuring safety, as cloud-to-ground discharges possess the potential to cause harm to individuals and property. For this work it is also unknown how this model compares to a randomly-predicting model on their dataset.

Leal and Matos's study in Brazil used six-year (2015-2020) data from ground-based weather stations, including air temperature, humidity, pressure, and wind speed to predict lightning occurrence within one hour. Their results showed that their decision tree predicted over 70% of lightning occurrences [4].

Mzila et al. predicted 80% of lightning occurrences in South Africa using deep neural networks [5]. They used meteorological parameters such as the temperature, air pressure and lightning data from South Africa weather service. Two different neural network architectures were applied to the prediction task, namely standard neural networks and radial basis function network. Both architectures exhibited comparable performance in terms of prediction accuracy.

Essa et al. found that the Long-Short-Term-Memory Recurrent neural networks (LSTM) model outperformed other models [6]. The study used historical Cloud-to-ground Lightning Data from the South African Lightning Detection Network (SALDN) for the year 2018. The dataset contains nearly 20 million lightning observations grouped into three-hour intervals. Three machine learning models are evaluated: Autoregressive (AR), Auto Regressive Integrated Moving Average (ARIMA), and LSTM Recurrent Neural Network. The models were tested for their ability to predict the number of Cloud-to-ground lightning flashes in South Africa for a three-hour time horizon using the Mean Absolute Percentage Error (MAPE) metric. The LSTM's MAPE was approximately 3700 versus AR and ARIMA's MAPE values of 15,312 and 15,080, respectively.

Mostajabi et al, used XGBoost algorithm to demonstrate that the model has a predictive capability for lightning occurrence lead times up to 30 min [7]. The predictors

included available surface weather variables, namely air pressure at station level, air temperature, relative humidity, and wind speed. Finally, Speranza conducted a study [8] that utilized surface data from Electric Field Mills (EFM) within a 50-mile radius of CCSFS to forecast lightning. The study employed various LSTM structure models, all of which achieved a prediction accuracy of above 70%. This research built on previous efforts by weather squadrons, specifically Venzke and Folsom [9] [10], who used simpler decision trees for lightning predictions at the time. However, their previous work relied on sensors located miles away and did not incorporate rawinsonde data.

The focus of this study is to investigate the ability of classical machine learning and neural network algorithms in predicting daily lightning occurrence at CCSFS, using the CCSFS morning rawinsonde data. We will explore various machine-learning models, including deep neural networks, logistic regression, and extreme gradient boosting, as they have demonstrated high accuracy in predicting lightning and thunderstorms in prior research. The study aims to present the results to the decision-makers of 45 WS to compare with existing systems. The findings of this study could potentially improve the current lightning prediction models and contribute to better decision-making regarding safety measures and operations at CCSFS.

B. Data Acquisition

Two sets of data were combined prior to analysis. The first set of data was obtained from the 14th Weather Squadron; the US Air Force unit responsible for collecting climatic weather data. The 14 WS provided summary rawinsonde data, including stability indices and mean flow data for each rawinsonde released at CCSFS from 2012 to 2021. The list of input variables with a brief description, mean, minimum value and maximum value can be found in Table 1. In addition, further information on these variables can be found on the Nation Weather Service (NWS) website on Skew-T derived parameters [11]. The second set of data was obtained from the 45 WS and provided daily observed lightning documentation. This data is utilized as the predictor (independent) variable and specifies whether lightning did or did not occur. The observed lightning data included lightning strikes for multiple locations on or around CCSFS. Lightning occurred during 34% of the days in the dataset.

Table 1: Descriptive statistics of model features, where wind-based variables are denoted by bold text.

Description	mean	min	max
Convective Available Potential Energy (J/kg)	1109.2	-794.9	4247.7
Lapse rate 700 mb to 500 mb (°C/km)	-0.00174	-0.00253	-0.00121
Lapse rate 850 mb to 500 mb (°C/km)	-0.00168	-0.00209	-0.00127
L convective index (°C)	-2.3	-9.2	14.6
K convective index (°C)	23.6	-38.18	42.4
Thompson convective index (°C)	26.0	-49.8	51.6
Vertical lapse rate + low-level moisture	41.5	8.6	55.2
The temperature at 500 mb (°C)	25.1	18.6	28.2
The temperature at 700 mb (°C)	17.2	9.4	21.6
The temperature at 850 mb (°C)	8.7	0.8	13.4
The temperature at 1000 mb (°C)	-6.6	-12.5	-2.7
Convective temp - Approximate temperature that the air near the ground must warm to for surface-based convection to develop (°C)	21.52	5.69	26.53
Relative Humidity 1000 mb to 700 mb (%)	67.40	29.78	98.35
Precipitable water (kg/m ²)	43.57	-403.3	704.46
Relative Humidity 700 mb to 500 mb (%)	48.60	1.716	98.63
Relative Humidity surface to 700 mb (%)	68.11	31.76	98.31
Averaged windspeed surface to 700 mb (kt)	12.52	2.74	30.43
Averaged wind speed 1000 mb to 700 mb in "east/west" direction (kt)	-0.585	-26.89	25.68
Averaged "north/south" wind speed 1000 mb to 700 mb (kt)	3.71	-22.03	26.27
850 mb "east/west" wind speed (kt)	-0.041	-30.83	35.8
700 mb "east/west" wind speed (kt)	2.97	-36.79	48.8
500 mb "east/west" wind speed (kt)	5.97	-30.63	50.32
250 mb "east/west" wind speed (kt)	13.22	-30.63	93.72
850 mb "north/south" wind speed (kt)	3.44	-29.82	29.07
700 mb "north/south" wind speed (kt)	3.27	-30.04	35.19
500 mb "north/south" wind speed (kt)	1.031	-22.65	33.84
250 mb "north/south" wind speed (kt)	-2.05	-43.73	53.35
Shear; surface to 6 km (kt)	0.00164	-0.00101	0.00515
Storm relative helicity; surface to 3 km (m²/s²)	5.12	-96.10	97.26
Wet bulb zero - pressure level where sounding is at zero degrees Celsius due to evaporational cooling (mb)	12698.17	5857.78	16365.11
Lowest FZ level - the lowest elevation where temperature equals 0 °C (mb)	15534.33	10662.73	18273.38
Energy helicity index (J·m ² /s ² ·kg)	0.067	-0.969	1.803
Bulk Richardson number, CAPE / 0 - 6km shear (J/kg·kt)	2.26E+06	1.48E+07	3.36E+00

II. METHOD

The cross-industry standard process for data mining (CRISP-DM) was applied using the phases of data understanding, data preparation, modeling, and evaluation [12]. All phases were conducted with the Scikit-learn 1.2.2 and Keras/TensorFlow 2.11 frameworks within a GPU-enabled Python version 3.9.16 environment.

The data were modeled with logistic regression, the XGClassifier decision tree algorithm, and binary classification neural networks. For the first two methods, recursive feature elimination (RFE) was used for feature selection. The impact of wind variables on the models was also investigated. In the logistic regression and the neural

networks approach there are two classes: 1 for lightning occurs and 0 for lightning doesn't occur.

A. Data Preparation

The summary rawinsonde data were manually reviewed for missing or incorrect data. For records containing missing or incorrect data, the entire record was removed from the dataset. After ensuring the data was clean, the data was split 80/20 into training and holdout/testing datasets. This split allows independent testing of the models and prevention of overfitting [13]. The models were constructed using the training dataset and model performances were calculated from the test/holdout dataset. The logistic regression and XGClassifier modeling were applied to this split dataset and the neural network modeling further subdivided the training dataset into 80% training and 20% validation.

B. Metrics

To evaluate the performance of the classification models, we used both the F1 score and confusion matrix metrics. The F1 score provides an overall measure of the model's performance, but it does not give us insight into the types of errors the model is making. In contrast, the confusion matrix provides information on the number of true positives (TP), false positives (FP), true negatives (TN), and false negatives (FN), giving a more detailed understanding of the model's performance. The F1 score is calculated as the harmonic mean of precision and recall, which are shown in Equations 1 & 2.

$$\text{precision} = \frac{TP}{TP + FP} \quad (1)$$

$$\text{recall} = \frac{TP}{TP + FN} \quad (2)$$

Therefore, presenting both the F1 score and confusion matrix together can provide a more complete picture of the performance of the classification model, as it penalizes extreme values of either precision or recall. F1 is calculated using Equation 3:

$$\text{F1 score} = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \quad (3)$$

In the dataset, the positive class is a lightning occurrence. In the case of lightning forecasting, it is critical that false negatives be minimized, where a model predicts "no lightning" when lightning actually occurred. False negatives are shown in the lower left corner of the confusion matrices presented Figure 2 and Figure 4 of this work.

C. Trivial Modeling

As a further part of data understanding, performance metrics are also calculated for two trivial models. The first is a model that predicts the majority class of "no lightning" or 0, also known as the no-information rate model. The second is a model that predicts randomly with equal probabilities. For these trivial logistical models, all datapoints were

utilized. The majority class model possessed an F1 = 0.40, and the random model had an F1 = 0.47.

D. Statistical Modeling

The statistical models applied in this work used recursive feature elimination (RFE) to iteratively select features, using the sklearn logistic regression and XGClassifier algorithms. The default values for logistic regression were used, and the max_depth parameter within XGClassifier was set to 1 to limit overfitting. The F1 metric on the training and holdout datasets were compared to the other modeling techniques.

E. Neural Network Modeling

The Python Keras API was utilized to construct a neural network for this problem. Keras requires the selection of a loss function, optimizer, optimization parameter, and optional regularization technique. Since the neural network is a binary classification neural network the loss function chosen for this neural network is binary cross-entropy. Dropout was selected as the regulation technique parameter [14]. Adam & SGD algorithms were both evaluated as options for the optimizer.

Class weights were specified as the NN models had difficulty avoiding defaulting to the trivial majority case model. Also, to correct instability noted in the final stages of training, an adaptive learning rate was applied that halved the learning rate every 30 epochs.

For the maximum number of neurons, we use guidance from Widrow, who proposed that the number of recommended datapoints P is the number of weights (neurons * (inputs + 1)) divided by the desired error level, according to Equation 4 [15]. For a 20% error level and 334-row dataset, the recommendation was 2 neurons for dataset #1 and 3 neurons for dataset #2. As a result, the capacity of the neural network was limited to within an order of magnitude of this recommendation, and potential overfitting was monitored.

$$P = \frac{\text{neurons} * (\text{inputs} + 1)}{\text{error}} \quad (4)$$

The Adaptive Experimentation Platform (Ax) library was installed into Python, and used to tune the neural network model’s hyperparameters. Bayesian optimization was used for numeric features, and Bandit optimization was used for categorical features. Within Ax, the hyperparameters tuned and their search space is defined on the left side of Table 2. Ax converged on a solution after 30 optimization loops.

Hyperparameter	Range	Dataset #1	Dataset #2
Learning Rate	0.001 – 0.5	0.041	0.101
Class weight for “1” class	1-2	1.34	1.65
Dropout Rate	0.0–0.5	0.0	0.0

# Hidden layers	1-4	1	1
Neurons/layer	1-15	6	1
Batch size	8-128	76	79
Activation Function	ReLU, SeLU	ReLU	SeLU
Optimizer	Adam, SGD	SGD	SGD

Table 2. Hyperparameters for the Ax-optimized NN models

III. Analysis & Results

Performance metrics are presented in this section for the three variations of modeling (logistic regression, XGBoost & neural network) on dataset #1 and dataset #2. For comparison, the trivial majority-class and random models yielded F1 metrics of 0.40 and 0.47, respectively.

A. Dataset #1 Modeling – All Features

Figure 1 shows the F1 metric for the training and holdout datasets, across a range of input features. All features were used as inputs to the NN, and RFE feature selection was used on the logistic regression, and XGClassifier models. As visible in Figure 1, the optimized NN had the highest performance on the holdout dataset with F1 = 0.79. This model’s performance on the training and test/holdout datasets are designated by the black arrow in Figure 1. Even when limited to $max_trees = 1$, XGClassifier had significant overfitting issues as the number of features increased. The set of optimal neural network hyperparameters for this dataset is presented in Table 2 in the previous section.

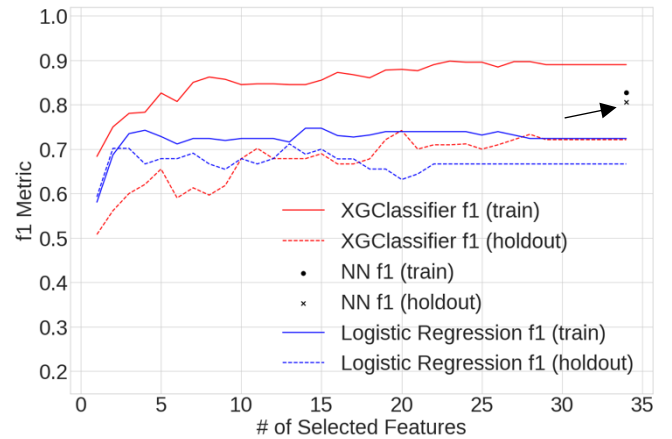


Fig. 1. XGClassifier, logistic regression, and neural network modeling results from dataset #1. The F1 metric for both the training and holdout datasets are presented. The arrow indicates the best model; the optimized NN model.

When balancing the tradeoff between complexity and performance, the best dataset #1 model is judged to be the Ax-optimized NN model. Its metric of F1 = 0.79 on the test/holdout dataset far exceeded the performance of any of the logistic regression or XGClassifier models. For the NN

model, the confusion matrix for both the training and holdout datasets is presented in Figure 2.

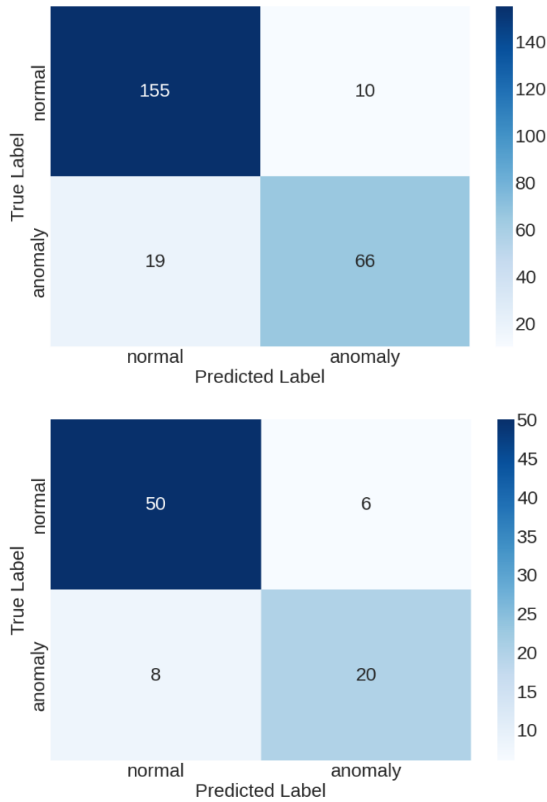


Fig. 2. Ax-optimized NN model performance, as illustrated by the confusion matrix for the training dataset (top) and test dataset (bottom).

B. Dataset #2 Modeling – Without Wind Features

Figure 3 presents the F1 metric for the training and holdout datasets on models created from dataset #2. The same process was followed to create neural network models, and also logistic regression and XGClassifier using RFE feature selection. Similar to dataset #1, the Ax-optimized neural network provided the highest performance with F1 = 0.66 on the test / holdout dataset. This model’s performance on the training and test/holdout datasets are designated by the black arrow in Figure 3. As visible in Figure 3, the NN model significantly outperformed logistic regression and the XGClassifier algorithms.

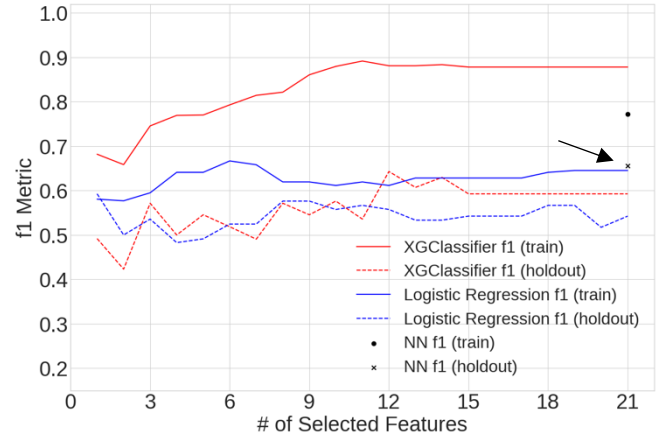


Fig. 3. XGClassifier, Logistic Regression, and neural network modeling results from dataset #2. The f1 metric for both the training and holdout datasets are presented. The arrow indicates the best model; Ax-optimized NN with all features.

The best dataset #2 model is judged to be the Ax-optimized NN model. Its metric of F1 = 0.66 on the test/holdout dataset exceeded the performance of any of the logistic regression or XGClassifier models. The confusion matrix for both the training and holdout datasets is presented in Figure 4.

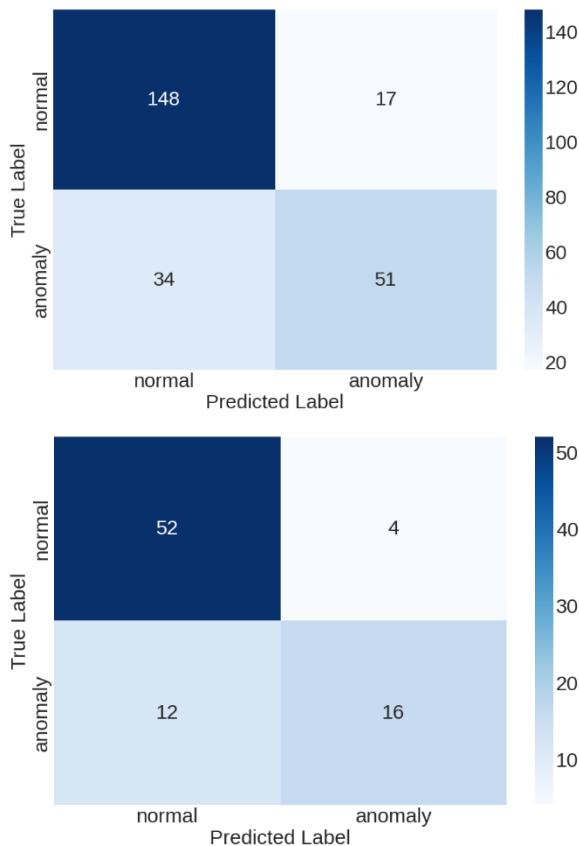


Fig. 4. XGClassifier 8-feature confusion matrix for the training dataset (top) and test dataset (bottom).

C. Model Evaluation

From the statistical modeling there were notable inferences available on both datasets. From dataset #1 (full dataset), the 2-feature logistic regression model on achieved the highest performance of any of the statistical models; $F1 = 0.72$ on the holdout dataset. The 2 features included in this model were the Thompson index, and the north/south component of wind speed at a pressure of 850 mb. For dataset #2 (without wind features) the 4-feature logistic regression model was interesting, as it was a simple model that achieved close to the highest performance of any of the statistical models; $f1 = 0.68$ on the holdout dataset. The 4 features included in this model via the RFE process were the Thompson index, the temperature at a pressure of 850 mb, the averaged relative humidity between 500-700 mb, and the average relative humidity from the surface to 700 mb pressure.

One of the main attributes of the model is consistently forecasting lightning occurrence. The current method employed at the 45 WS is a forecaster who determines whether lightning is going to occur later in the day by using many available tools and intuition. As such, different forecasts are produced as influenced by the on-duty forecaster. A machine-learning model will output reproducible results independent of individual forecasters. Using the model, the forecaster will also be able to output a

probability of lightning occurring later in the day and the probabilities will be reproducible.

Ultimately, the output of the model could have application as a part of the morning weather briefing to inform range customers about impending lightning that may affect their range operation for the day. In addition, the model will save daily FTE-hours used to forecast lightning occurrences.

IV. Conclusion

In this work, a dataset consisting of 34 lightning, pressure, temperature and windspeed measurements was analyzed from 334 daily weather balloon / rawinsonde launches. Models were created using recursive feature elimination on logistic regression and XGClassifier algorithms, and also using Bayesian and bandit optimization of neural network hyperparameters. The best performing models on both datasets were the optimized NN models, with an $F1 = 0.79$ on the full dataset and $F1 = 0.66$ on the reduced dataset. The addition of 13 windspeed-related features more than doubled the complexity of the 21-feature no-wind model while increasing model performance by 4%. A notable inference from the statistical modeling is that the most important feature from both datasets was the Thompson index. Future work could include expanding the set of input features and comparing machine learning performance to manual forecaster performance.

Authors' Note: The views and opinions of authors expressed herein do not necessarily state or reflect those of the U.S. Government or any agency thereof. Reference to specific commercial products does not constitute or imply its endorsement, recommendation, or favoring by the U.S. Government. The authors declare this is a work of the U.S. Government and is not subject to copyright protections in the United States. This article has been cleared with case number 88ABW-2023-0423.

REFERENCES

- [1] K. Bala, D. K. Choubey and S. Paul, "Soft computing and data mining techniques for thunderstorms and lightning prediction: a survey," in 2017 International conference of Electronics, Communication and Aerospace Technology (ICECA), 2017.
- [2] B. C. Bates, A. J. Dowdy and R. E. Chandler, "Lightning prediction for Australia using multivariate analyses of large-scale atmospheric variables," *Journal of applied meteorology and climatology*, pp. 525-534, 2018.
- [3] Y. Zhu, P. Bitzer, V. Rakov and Z. Ding, "A machine-learning approach to classify cloud-to-ground and intracloud lightning," *Geophysical Research Letters*, p. e2020GL091148, 2021.
- [4] A. F. Leal and W. L. N. Matos, "Short-term lightning prediction in the Amazon region using ground-based weather station data and machine learning techniques," in 2022 36th International Conference on Lightning Protection (ICLP), 2022.

- [5] S. Mzila, A. Ajoodha and H. Hunt, "Prediction of Lightning Occurrence in South Africa Using Neural Network and Meteorological Data," Available at SSRN 4331675, 2022.
- [6] Y. Essa, H. G. Hunt and R. Ajoodha, "Short-term prediction of lightning in southern africa using autoregressive machine learning techniques," in 2021 IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS), IEEE, 2021, pp. 1-5.
- [7] A. Mostajabi, D. L. Finney, M. Rubinstein and F. Rachidi, "Nowcasting lightning occurrence from commonly available meteorological parameters using machine learning techniques," Npj Climate and Atmospheric Science, p. 41, 2019.
- [8] D. Speranza, "Lightning Prediction Using Recurrent Neural networks," Air Force Institute of Technology, Wright-Paterson AFB, 2019.
- [9] K. C. Venzke, "Development of Predictors for Cloud-to-Ground Lightning Activity Using Atmospheric Stability Indices," Air Force Institute of Technology, Wright-Patterson AFB, 2001.
- [10] M. I. Folsom, "Developing a Forecast Tool for Cloud-to-Ground Lightning in the North Central and Northeastern United States," Air Force Institute of Technology, Wright-Patterson AFB, 2004.
- [11] Nation Weather Service, "Skew-T Parameters," Nation Weather Service, [Online]. Available: https://www.weather.gov/source/zhu/ZHU_Training_Page/convective_parameters/skewt/skewtinfo.html#SKEW2. [Accessed 10 08 2022].
- [12] IBM Corporation, "IBM SPSS Modeler CRISP-DM Guide," IBM Corporation, 2011.
- [13] A. Géron, "Overfitting the Training Data," in Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow, Sebastopol, O'Reilly Media, Inc., 2019, p. 28.
- [14] Keras, "Layer weight regularizers," [Online]. Available: <https://keras.io/api/layers/regularizers/>. [Accessed 16 08 2022].
- [15] B. Widrow, "ADALINE and MADALINE," in 1st International Conference on Neural Networks, San Diego, CA, 1987.
- [16] A. Géron, in Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow, Sebastopol, O'Reilly Media, Inc., 2019, p. 31.
- [17] G. James, D. Witten, T. Hastie and R. Tibshirani, in An Introduction to Statistical Learning with Applications in R, New York, Springer, 2017, p. 250.