Oklahoma Weather Modeling Predictions

Matthew Ferguson, Ryan Coffin, and Helmuth Gonzalez, Group D

***Abstract* — Our team developed a set of models that predict next day precipitation using climate data. Model data is sourced from the Global Historical Climatology Network daily (GHCNd), a climate database composed of daily weather summaries from a network of international land surface stations. GHCNd provides a range of variables for modeling including temperature, precipitation, and windspeed.[1] We use data from GHCNd station USW00013967 at Will Rodgers World Airport to train two sets of machine learning models. The first machine learning model set uses a decision tree to classify the occurrence of next day rain. The second model set uses linear regression to estimate the amount of precipitation on the following day. Models were trained on 6893 days of weather event data. The best classification accuracy of the decision tree models was 78%. The best coefficient of determination of the linear regression models was 0.054. Further progress on machine learning models for next day precipitation predictions can be made with humidity data.**

***Keywords* — Predictive analytics, Regression analysis, Classification tree analysis, Metamaterials**

I. Introduction

CLIMATE and weather models help us to accurately predict local precipitation and better understand climate. Specifically, we are exploring the relationship between daily weather measurements and next day precipitation to better understand the climate and weather of Oklahoma. Oklahoma has regular thunderstorms and tornadoes, so weather models are helpful in predicting critically severe weather.

The National Oceanic and Atmospheric Administration (NOAA) and the National Centers for Environmental Information (NCEI) provide data access to GHCNd data. GHCNd is an integrated network of over 100,000 land surface stations and collects weather data across 180 countries. Dozens of measurement types are made at a station daily, and station measurements often date back decades. These records are a treasure trove of climate data that can be leveraged as training and test sets for machine learning models.

Our team produced two sets of machine learning models by extracting features which highly correlated with next day precipitation from GHCNd data. The first model set are decision tree classification models which predicts whether or not rain will occur as a binary outcome (yes, no). The second model set uses linear regression models to predict the amount of precipitation on the following day.

II. Weather Station Data

The selected weather station for analysis is USW00013967 and is located in the state of Oklahoma (U.S.) in Oklahoma City's Will Rodgers World Airport. We selected this station specifically because it will enable us to better understand our local climate and weather.

Additionally, USW000013967 has collected data since December 1941 and our snapshot of this data occurs March 2022. The GHCNd dataset for USW00013967 contains more than 50 potential variables within its approximate 340,000 data entries.[3] The vast amount of data offers many features, events, and targets for training machine learning models. Many features do not extend over the full period as new instruments and measurements are introduced and old ones retired (e.g., cloud coverage only being available from 1960s, -1980's). Exploration of the performance of the models when using measurements from different time periods is key. The most accurate models were constructed using 6,894 data points spanning from 1965 to 1983 which ran contrary to our original intuition that modern data would yield the best modeling features.

III. Data Preparation

A. Data Processing

USW00013967 GHCNd data was downloaded using NOAA's public data access. The data is organized into three untitled columns and contains information regarding the type of measurements (e.g., type of precipitation), values (e.g., 0.3 millimeters of rain), and dates. One date is repeated multiple times for each measurement type and value. In this format, a single date may appear in as many as 52 rows. We processed the data using Python's Pandas DataFrame pivot functions to transform the original data format into rows of events by date and columns of features. The result was a table containing at least 55 columns: a date column and a column for each of the possible types of measurements, and the two target variables. After this transformation each date (and the date's measurements) can be processed by machine learning algorithms as a single event consisting of up to 52 features. See Table II for a definition of dataset features.

Two feature metavariables were created, PRECIPFLAG (PFLAG) and PRECIPAMT (PAMT). These variables are the basis for our two target variables, NEXTDAYPRECIPFLAG (NPFLAG) and NEXTDAYPRECIPAMT (NPAMT). PFLAG and NPFLAG are binary indicators of rain, either rain occurred and the flag equals 1 or rain did not occur and the flag equals zero. PAMT and NPAMT are variables that measure total precipitation as the sum of daily rainfall (PRCP) and daily snowfall. Our target variables NPFLAG and NPAMT are offset by one day such that a date event will have precipitation values from the next day to use as a training target.

A histogram of next-day precipitation and amounts is shown below. Interestingly, the image suggests that the probability of no next day precipitation on any given day is approximately 77% (5,367 no next-day precipitation entries over 6,893 data entries), like the highest accuracy achieved in our best decision tree for this analysis.

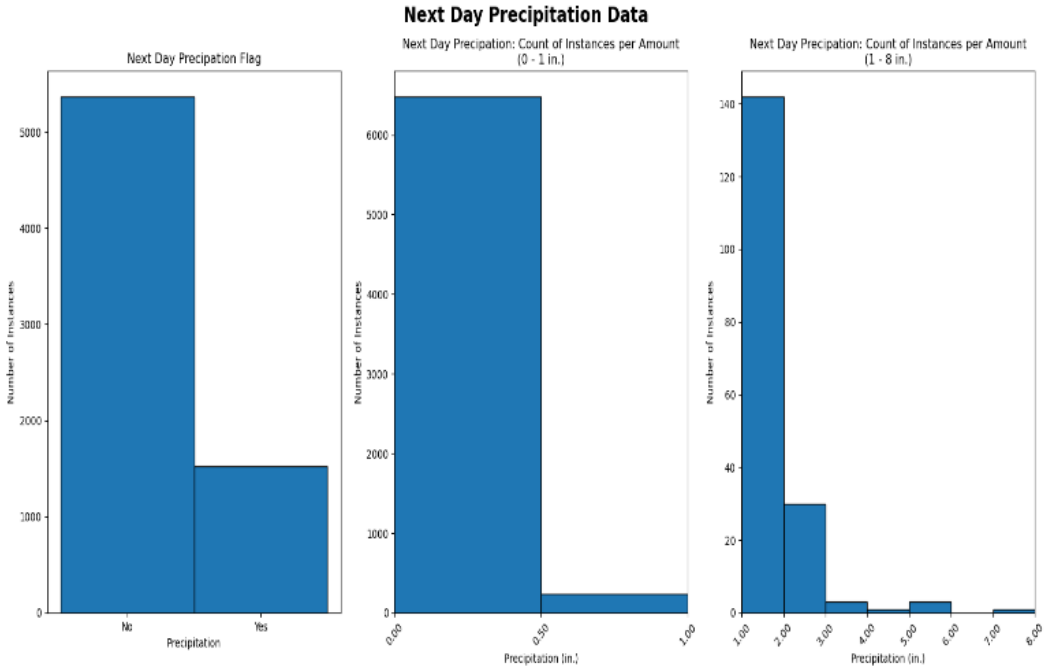

Fig. 1. Next Day Precipitation

Due to the disparity favoring no precipitation, the number of instances per amount for next-day precipitation had to be shown in the two graphs for different ranges: 0 to 1 inch (middle image) and 1 inch to 8 inches (right-most image). The histogram of next day precipitation is shown in Figure 1.

A Data Quality Report and a correlation matrix were created using the processed data. The features which most highly correlated with the target variables were selected for training. The selected features are discussed in Correlation Matrix Section. All events missing these selected features were deleted from the sample set. The remaining data set ("clean" data) contained 6983 events with high correlation features and no missing values. Therefore, no imputation of values was necessary. Table I below offers a description of the processed and clean datasets.

TABLE I
Data Nomenclatures and Descriptions

Nomenclature	Description
Processed Data	Processed data is the output of Python's Pandas pivot method applied to the raw GHCNd USW00013967 data, prior to deleting events (i.e., dates) missing selected high correlation features.
Clean Data	Clean data is the data set which results from deletion of partial events from the processed data. Events (i.e., dates) that did not contain selected highly correlated features were deleted.

B. Data Quality Report

The Data Quality Reports (Table III, and Table IV) were prepared by using the processed and clean event data to compute the following feature specific statistics: cardinality, mean, median, count of entries at the median (N-Median), mode, count of entries at the mode (N-Mode), standard deviation (Deviation), minimum, maximum, count of "zero" entries (Zeroes), and count of missing entries (Missing). Table II below provides a definition of each variable presented in the correlation matrix. [4]

The most notable outcome of the data quality report is the division in the data of measurement types. Cloud coverage and sunshine are only obtainable in the 1960s-1980s and were highly correlated with our target variables. However, windspeed was also highly correlated with our target variables and only available from the 1990's onwards. Therefore, when we proceeded to the modeling phase, we generated models from two separate clean data sets. In the end, the older cloud coverage and sunshine data created higher overall model performance. As a result, we present the Data Quality Report of the clean dataset containing cloud coverage and sunshine data in addition to the Data Quality Report of the processed data.

TABLE II
Feature Definitions

Feature	Definition
ACMH (percentage)	Average cloudiness from midnight to midnight.
ACSH (percentage)	Average cloudiness from sunrise to sunset.
PRCP (tenths of mm))	Amount of rainfall.
PRECIPAMT (inches)	Total amount of precipitation.
PRECIPFLAG (binary)	Precipitation indicator
PSUN (percentage)	Daily percent of possible sunshine.
TMAX (tenths of deg C)	Maximum temperature
TMIN (tenths of deg C)	Minimum Temperature
TSUN (minutes)	Daily total sunshine.
WSF2 (tenths of meters per second)	Fastest 2-minute wind speed
WSF5 (tenths of meters per second)	Fastest 5-minute wind speed
WTS	Sum of weather type columns, a metavariable.

TABLE III
Data Quality Report of Raw NOAA Data

	Date	ACMH	ACSH	PRCP	PSUN	TMAX	TMIN	TSUN	WSF2	WSF5	PRECIP FLAG	PRECIP AMT	NEXT DAY PRECIPFLG	NEXT DAY PRECIP AMT
Cardinality	29262	11	11	309	101	123	124	819	52	76	2	564	2	564
Mean	19816900.1	49.153	53.346	23.792	66.79	219.309	96.1943	478.069	99.340	122.547	0.208	0.097	0.208	0.097
Median	19820215.5	50	60	0	80	233	100	564	98	116	0	0	0	0
N-Median	0	886	690	21015	77	430	480	69	593	508	23165	20997	23164	20997
Mode	19411214	0	100	0	100	311	211	0	94	103	0	0	0	0
N-Mode	1	1294	2034	21015	1111	593	799	1442	677	568	23165	20997	23164	20997
Deviation	231337.916	33.291	36.644	86.333	33.9	104.445	98.1814	271.922	32.209	40.679	0.406	0.343	0.406	0.343
Minimum	19411214	0	0	0	0	-161	-255	0	27	0	0	0	0	0
Maximum	20220308	100	100	1935	100	450	289	876	443	720	1	7.618	1	7.618
Zeroes	0	1294	1497	21015	559	87	520	1442	0	2	23165	20997	23164	20997
Missing	0	19128	19127	2168	22368	2173	2168	17328	18760	18769	0	2170	1	2170

TABLE IV
Data Quality Report of Preprocessed NOAA Data

	Date	ACMH	ACSH	PRCP	PSUN	TSUN	WTS	PRECIP FLAG	PRECIP AMT	NEXT DAY PRECIPFLG	NEXT DAY PRECIP AMT
Cardinality	6893	11	11	208	101	147	8	2	294	2	294
Mean	19740540	49.206	53.563	22.158	66.795	494.295	0.954	0.221	0.090	0.221	0.090
Median	19740612	50	60	0	80	570	0	0	0	0	0
N-Median	1	603	454	5367	77	75	3740	5367	5367	5367	5367
Mode	19650102	0	100	0	100	0	0	0	0	0	0
N-Mode	1	878	1408	5367	1111	543	3740	5367	5367	5367	5367
Deviation	54908.554	33.263	36.714	83.498	33.891	259.687	1.254	0.415	0.333	0.415	0.333
Minimum	19650102	0	0	0	0	0	0	0	0	0	0
Maximum	19831231	100	100	1913	100	876	8	1	7.531	1	7.531
Zeroes	0	878	1025	5367	558	543	3740	5367	5367	5367	5367
Missing	0	0	0	0	0	0	0	0	0	0	0

C. Correlation Matrix

The correlation matrix we developed can be seen in Table V. The correlation matrix uses processed data and reveals the features that most correlate (or anti-correlate) to next day precipitation targets. Average cloudiness from midnight to midnight (ACMH) had the highest correlation (0.319) to NPFLAG. ACMH correlated 0.181 to NPAMT. This is a surprising outcome as ACMH is a manually collected datapoint.

The average cloudiness from sunrise to sunset (ACSH) was the second most correlated (0.272) with NPFLAG. ACSH had a 0.152 correlation with NPAMT. PSUN and TSUN were most anticorrelated with the target variables (See Table V).

With the strongest correlations identified, events that did not contain ACMH, ACSH, TSUN, and PSUN data were deleted resulting in a clean dataset of 6,894 points – sufficient data points for training and testing models. It is important to note that, although not included in Table V, features WSF2 and WSF5, the fastest 2- and 5-minute windspeeds, respectively, were the next highest set of variables correlated to the targets. However, these features were excluded from the analysis as they produced lower accuracies and coefficients of determination when compared to the ones shown in Table V. Also, WSF2 and WSF5 were not recorded until 1993 which is outside the date range of our cloud coverage and sunshine data (which ended in 1983).

TABLE V
Correlation Matrix

	NEXTDAY PRECIPFLAG	NEXTDAY PRECIPAMT	ACMH	ACSH	PRCP	PSUN	TSUN	PRECIP FLAG	PRECIP AMT
NEXTDAYPRECIPFLAG	1.000	0.522	0.319	0.272	0.186	-0.268	-0.202	0.276	0.185
NEXTDAYPRECIPAMT	0.522	1.000	0.181	0.152	0.145	-0.122	-0.092	0.161	0.143
ACMH	0.319	0.181	1.000	0.945	0.292	-0.822	-0.748	0.502	0.300
PRECIPFLAG	0.276	0.161	0.502	0.436	0.511	-0.523	-0.418	1.000	0.522
ACSH	0.272	0.152	0.945	1.000	0.251	-0.798	-0.731	0.436	0.258
PRCP	0.186	0.145	0.292	0.251	1.000	-0.293	-0.237	0.511	0.995
PRECIPAMT	0.185	0.143	0.300	0.258	0.995	-0.304	-0.247	0.522	1.000
TSUN	-0.202	-0.092	-0.748	-0.731	-0.237	0.958	1.000	-0.418	-0.247
PSUN	-0.268	-0.122	-0.822	-0.798	-0.293	1.000	0.958	-0.523	-0.303

IV. Modeling

We created two sets of model types, a set of decision tree classification models, and set of linear models. Each type of model was trained using features from a single day and features from two days to predict next day precipitation. The decision tree classifier models predict if rain is present on the next day (yes/no) whereas the linear models predict next day precipitation amount.

The processed and cleaned data was stored in a Python DataFrame, and each of the features (represented by a column in the DataFrame) were normalized. A random seed was then used to repeatably and randomly divide the data such that 70% of the data would serve as the training set and the remaining 30% would serve as the test set.

Models were trained using the training data set. The test data set was input into our models and the predictions the model made were evaluated for performance. Features and other parameters were iteratively changed to determine the features and settings that optimize model performance. Notably, features from a single day and features from two days are used to train both model types and have varying impact on performance metrics. The performance metrics and impacts can be observed in Tables VI and VII.

A. *Classification*

We leverage Python's scikit-learn library to train decision tree classifier models. A classification model is a model which assigns an event (e.g., a date) to a target category or class (e.g., next-day precipitation) using relationships between features and targets. A decision tree is a special form of classification model that uses decision rules (or node questions) inferred by machine learning algorithms to predict classes of data. Given that features are highly correlated to targets, a decision tree is the perfect tool for predicting whether or not it will rain tomorrow.

Entropy and Gini classification criteria for information gain were used to split nodes. Entropy based decision trees are more computationally complex and more accurate than Gini based decision trees. A Gini tree is found to scale better then Entropy due to the design being less computationally complex. This application has a small number of models being trained so entropy-based information gain was the optimal criteria for our models. Tree depth was limited to three to prevent overfitting and poor generalization.

The strength of correlation was the chief determinant in selecting which features to initially include in model training. The features selected for iterative training on the decision tree classifier are ACMH, ACSH, PSUN, TSUN, WSF2, TMIN, WTS, and PRECIPFLAG.

Model performance was then measured by calculating the classification accuracy. As the ratio of correct classifications to total classifications, model accuracy is the percentage of test set data points that were correctly categorized. Accuracy is mathematically expressed in Equation (1).

$$Accuracy = \frac{|TestSetCorrectlyPredicted|}{|TestSet|} \quad (1)$$

Different decision tree criteria for information gain-based splits (entropy and Gini) and different combinations of features were iteratively tested to maximize classification accuracy. Classification accuracy ranged between 68% and 78%. The iteration with the highest accuracy (78%) was an entropy-based decision tree consisting of variables ACMH, ACSH, and PRECIPFLAG (daily binary feature pertaining precipitation). Accuracies for all classification models are shown in Table V and our best decision trees using a single day and two days of feature data are shown in Figures 2 and 3. Wind speed-based models used more modern measurement instruments however produced lower classification accuracies. Oklahoma has unique weather famous for tornadoes, so it is not surprising to see the wind speed was a good predictor of precipitation. Cloudiness and sunshine however appear more directly correlated with precipitation outcome. Notably the inclusion of two days of features decreased model classification accuracy.

TABLE VI
Classification Model Accuracies

Classification Model	Training Features	Model Accuracy per Prediction	
		Next – Day	Two – Day
Entropy	ACMH, ACSH, PRECIPFLAG	77.6%	73.9%
Gini	ACMH, ACSH, PRECIPFLAG	77.6%	73.6%
Entropy	WSF2, PRECIPFLAG, WTS, TMIN	71.8%	68.2%
Gini	WSF2, PRECIPFLAG, WTS, TMIN	71.6%	67.8%

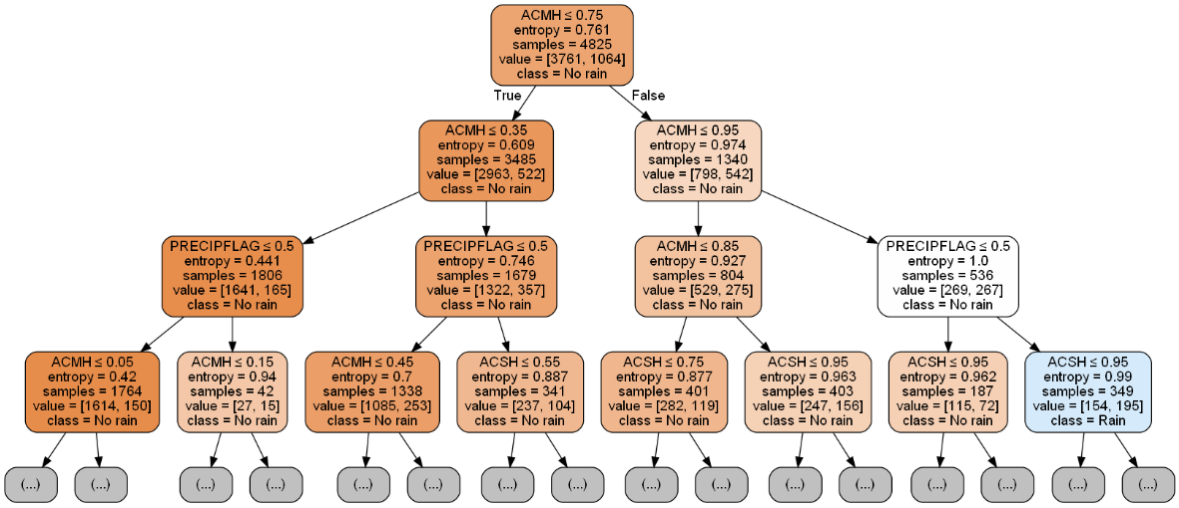

Fig. 2. Entropy Decision Tree using a Single Day

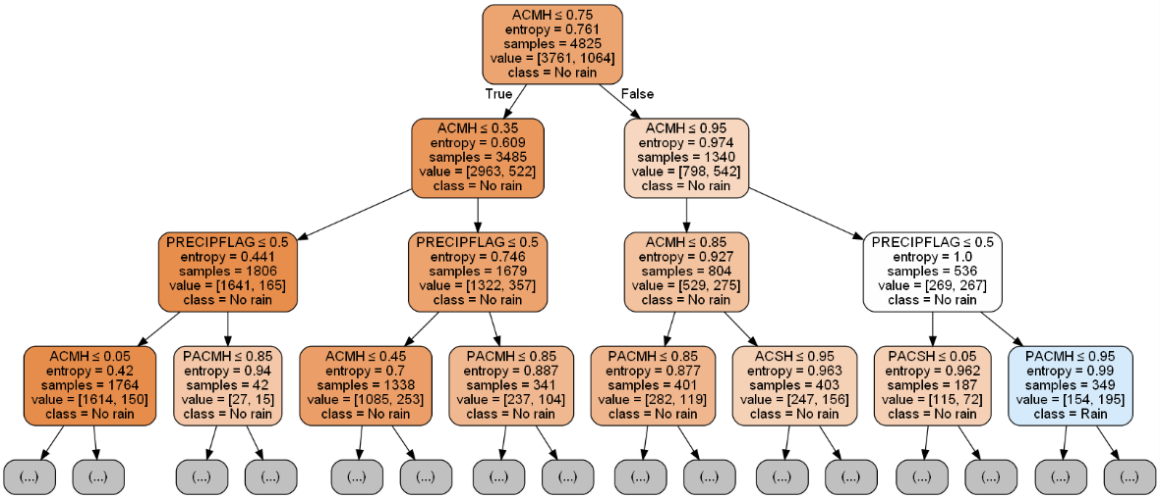

Fig. 3. Entropy Decision Tree using Two Days'

B. LINEAR MODELS

We leverage Python's scikit-learn library to train several linear models. Linear models describe target variables in terms of a linear combination of features. Linear models excel when trained with continuous numerical data. Linear regression models assume that a linear relationship exists between input and output variables to produce predictive models. Ridge regression with built-in cross-validation (RidgeCV) is another linear model type. However, RidgeCV introduces bias to decrease variance and address multicollinearity unlike linear regression models.

Two metrics were used to assess the performance of our linear models: Mean Squared Error (MSE) and the coefficient of determination. The MSE is a measure of model error Mathematically, MSE is the average error squared representing the Euclidean distance of a point from its true value (e.g., amount of true next day precipitation) to a model's estimated value (the predicted amount of next day rain). The form of mean square error is shown equation (2):

$$MSE = \frac{1}{|TestSet|} \sum_{1}^{|TestSet|} (NDP - MO)^2 \qquad (2)$$

The coefficient of determination, or goodness of fit, is used to analyze the impact one variable or set of variables has on another. Simply put, the coefficient of determination is the percentage of data points that fall on a line during linear regression; the higher the coefficient of determination, the more data points fall on the line and the more accurate the linear regression. The features we trained on iteratively to optimize model performance are ACMH, ACSH, PRCP, PRECIPAMT, PRECIPFLAG, PSUN, TSUN, WSF2, TMIN, and WTS. The features we trained on which resulted in the highest coefficients of determination for our models are ACMH, ACSH, PRCP, PRECIPAMT, PRECIPFLAG, PSUN, TSUN,

Model performance metrics, MSE and coefficients of determination, for linear models are in the table below (Table VII). Notably the inclusion of features spanning two days increased performance metrics of our linear models.

TABLE VII
Regression Model Accuracies

Linear Model	Training Features	Accuracy Metric	Model Accuracy per Prediction	
			Next – Day	Next – Day
Linear Regression	ACMH, ACSH, PRCP, PRECIPAMT, PRECIPFLAG, PSUN, TSUN	Coefficient of determination (R^2)	0.0491	0.0533
		Mean Squared Error (MSE)	0.00269	0.00269
RidgeCV		R^2	0.0483	0.0526
		MSE	0.00269	0.00269
Linear Regression	WSF2, PRCP, WTS, TMIN	R^2	0.0430	0.0439
		MSE	0.00211	0.00211
RidgeCV		R^2	0.0430	0.0438
		MSE	0.0211	0.00211

V. Discussion

We are modestly pleased to have a classification accuracy of 78% for our classification model, and a coefficient of determination of 0.0533 on our best linear model. We are interested in exploring how we can build from these models to improve performance metrics.

Notably, there was no humidity data present in our original data set. A literature review of weather data modeling was conducted which suggested that an increase in precipitation prediction performance could be achieved by using humidity/dew point data. As an initial look, we proceeded to access daily data from the U.S. Climate Reference Network/U.S. Regional Climate Reference Network (USCRN/USRCRN) which contained humidity, ground soil temperature, and surface moisture measurements. With the new features, a correlation matrix was built. Notably, humidity, ground soil temperature, and surface moisture measurements were extremely correlated to amount of precipitation. Our next steps certainly involve analyzing this data set further.[2]

With more than 100,000 weather stations in 180 countries around the world, the GHCNd database contains a wealth of data and information. Another direction for future progress could be to scale the methods applied here to all GHCNd station datasets. However, for such an application Gini criterion for information gain-based splits is likely preferred over entropy due to its faster computation time for decision tree classifiers. Another challenge is that weather stations will not have the same measurement types, nor will they have the same strength of correlation between features and targets. Instruments and measurements will change at different times for each station. However, feature selection based on high correlation can be somewhat automated and those features can be iteratively passed to classification and linear models until performance metrics are maximized. For next steps we recommend incorporation of humidity, ground soil temperature, and surface moisture measurements to increase model accuracy and model coefficient of determination and then scaling production of models across many stations.

VI. Conclusion

. Our team has proven that the GHCNd database can be used to create machine learning models that predict local weather. However, classification accuracy and coefficient of determination results were lower than expected. Our best classification accuracy is 78%, and rain occurs on 23% of days. Our model is only marginally more accurate than one that assumes rain never occurs (77% accuracy). Furthermore, our best coefficient of determination is 0.0533. This is a modest to small value for goodness of fit.

Notably, we have succeeded in determining which features from Oklahoma weather stations most highly correlate to next day precipitation. Those features are cloud coverage, current day precipitation (amount and occurrence), and amount of sunshine. In more modern data sets wind speed also highly correlates to next day precipitation. Follow up work should expand to other GHCNd stations located in Oklahoma to validate our findings. Oklahoma suffers regularly from thunderstorms and tornadoes so there is a particularly great need for models which predict amount of precipitation and severe weather events.

Acknowledgment

The team would like to provide an acknowledgement to Professor Creed Jones from Virginia Tech in Blacksburg, VA. Professor Jones supplied critical information on model designs, decision trees, and critical predictive accuracies. The work he provided assisted in the team's advancement on decision predictive machine learning.

APPENDIX

A. DECISION TREE WEATHER PYTHON CODE

```
from sklearn import tree
import numpy as np
import pandas as pd
import pydot
Training_Features= ['ACMH','ACSH','PRECIPFLAG']
odf = pd.read_excel(r'C:\Users\Matt\Desktop\Model 2 Events.xlsx')
df=(odf-odf.min())/(odf.max()-odf.min())
cm=df.corr()
train_df=df.sample(frac=0.70, random_state=200)
test_df=df.drop(train_df.index)
Test_Answers=test_df['NEXTDAYPRECIPFLAG'].values.tolist()
Feature=train_df[Training_Features].values.tolist()
Target=train_df['NEXTDAYPRECIPFLAG'].tolist()
criteria="gini"
clf=tree.DecisionTreeClassifier(criterion=criteria)
clf=clf.fit(Feature, Target)
Fnames=Training_Features
Tnames='No rain', 'Rain'
dot_data=tree.export_graphviz(
            clf, out_file=None, feature_names=Fnames,
            class_names=Tnames, filled=True, rounded=True,
             special_characters=True, max_depth=3)
(graph,) =pydot.graph_from_dot_data(dot_data)
graph.write_png(r'C:\Users\Matt\Desktop\Weather_Tree_' + criteria +'.png')
ndf=test_df[Training_Features]
ndf=ndf.to_numpy()
results=[]
for row in ndf:
    prediction=clf.predict(row.reshape(1, -1))
    results.append(prediction[0])
mses = ((np.array(Test_Answers)-np.array(results))**2).mean()
accuracy= (len(results)-np.sum(abs(np.array(Test_Answers)-
np.array(results))))/len(results)
```

B. LINEAR REGRESSION PYTHON CODE

```
from sklearn import linear_model as linmod
import pandas as pd
import numpy as np
odf = pd.read_excel(r'C:\Users\Matt\Desktop\Model 2 Events.xlsx') #import
preproc data
df=(odf-odf.min())/(odf.max()-odf.min()) #normalize
Training_Features=
['PRCP','ACMH','ACSH','TSUN','PSUN','PRECIPAMT','PRECIPFLAG']
train_df=df.sample(frac=0.70, random_state=200) #split data w/ seed
test_df=df.drop(train_df.index)
Test_Answers=test_df['NEXTDAYPRECIPAMT'].values.tolist()
Feature=train_df[Training_Features].values.tolist()
Target=train_df['NEXTDAYPRECIPAMT'].tolist()
trainX = np.array(Feature)
trainY = np.array(Target)
mlr = linmod.LinearRegression()
mlr.fit(trainX, trainY)
R_Squared=mlr.score(trainX, trainY)
print("W = ", mlr.intercept_, mlr.coef_)
ndf=test_df[Training_Features]
ndf=ndf.to_numpy()
results=[]
for row in ndf: #predict the rain amount for each row in the test data
    prediction=mlr.predict(row.reshape(1, -1))
    results.append(prediction[0])
mses = (1/len(results))*np.sum(((np.array(Test_Answers)-
np.array(results))**2))
```

REFERENCES

[1] ***National Centers for Environmental Information***, Global Historical Climatology Network daily (GHCNd), United States, *US National Oceanic and Atmospheric Administration (NOAA) Daily*, 2022. Accessed on: Abbrev. Mar. 20, 2022. [Online]. Available: www.ncei.noaa.gov/products/land-based-station/global-historical-climatology-network-daily

[2] ***National Centers for Environmental Information***, *Quality Controlled Datasets*, United States, *US National Oceanic and Atmospheric Administration (NOAA) Daily*, 2022. Accessed on: Abbrev. Mar. 22, 2022. [Online]. Available: www.ncei.noaa.gov/access/crn/qcdatasets.html

[3] ***NOAA by Station***, USW00013967, United States, *US National Oceanic and Atmospheric Administration (NOAA) Daily*, 2022. Accessed on: Abbrev. Mar. 10, 2022. [Online]. Available: www.ncdc.noaa.gov/pub/data/ghcn/daily/by_station/

[4] ***Menne, M.J., I. Durre, B. Korzeniewski, S. McNeal, K. Thomas, X.Yin, S. Anthony, R. Ray, R.S. Vose, B.E.Gleason, and T.G. Houston****, 2012: Global Historical Climatology Network - Daily (GHCN-Daily), Version 3.[III, e.g. Version 3.12]. NOAA National Climatic Data Center.* http://doi.org/10.7289/V5D21VHZ [access date].

Matthew Ferguson was born in Fairfax Virginia in 1993. He graduated with a Bachelor of Science in mechanical engineering from Virginia Tech in Blacksburg, VA in 2015. Matt is currently pursuing a computer engineering Master's of Engineering from Virginia Tech.

He worked with GE Aviation starting in 2013 as a Propulsion Engineer creating thermodynamic simulations of jet engines. He entered the reliability industry in 2016 as a Manufacturing Engineer with PCC aerostructures designing reliability improvements for aerospace manufacturing environments. In 2018 he began working with Northop Grumman as a Reliability Engineer where he models the reliability of the E-3 AWACS. Matt hopes to leverage machine learning to create improvements in aviation safety and reliability.

Ryan Coffin was born in Mount Kisco, New York, 1996. He has graduated with an associate in engineering science from Dutchess Community College in Poughkeepsie, NY in 2018, and with a Bachelor of Science in electrical engineering from State University of New York at New Paltz, NY in 2020. Mr. Coffin is currently engaging in a Master of Engineering from Virginia Tech in Blacksburg, VA.

He is currently an Electrical Engineer for the Northrop Grumman Corporation's Defense System's Engineering team in Oklahoma City, Oklahoma.

HELMUTH E. GONZALEZ received the B.S. degree in mechanical engineering and the B.S. degree in aerospace engineering from the University of Florida and the M.S. degree in reliability engineering from the University of Maryland.

He is currently the Senior Principal Reliability Engineer for Northrop Grumman Corporation's Defense System's E-3 Sustainment Engineering team in Oklahoma City, Oklahoma, USA. He was a Lead Design Engineer with GE Appliances for approximately eight years and was involved in various projects ranging from the redesign of complex components to the improvement of customer appeal through craftsmanship. He later joined United Technologies Corporation's (UTC) Carrier division where he served as the functional engineering expert on Reliability Engineering matters, challenges, and projects. He is a certified GE Reliability Practitioner and a Lean Six Sigma Green Belt. He has authored one technical publication.