Weather Data Analysis for Enhanced Flight Visibility Management

Project submitted to the

SRM University – AP, Andhra Pradesh

for the partial fulfillment of the requirements to award the degree of

Bachelor of Technology

In

Computer Science and Engineering

School of Engineering and Sciences

Submitted by

Krishna Bogineni (AP21110010619)

Ravindranath Nayudu (AP21110010621)

Adnan Khan (AP21110011217)

Nizamuddin Tummalagodu (AP21110010552)

Under the Guidance of

(Dr. Rajiv Senapati)

SRM University–AP

Neerukonda, Mangalagiri, Guntur

Andhra Pradesh – 522 240

Nov, 2023

Certificate

Date: 28-Nov-23

This is to certify that the work present in this Project entitled "**Weather Data Analysis for Enhanced Flight Visibility Management**" has been carried out by **Krishna Bogineni, Ravindranath Nayudu, Adnan Khan and Nizamuddin Tummalagodu** under my/our supervision. The work is genuine, original, and suitable for submission to the SRM University – AP for the award of Bachelor of Technology/Master of Technology in **School of Engineering and Sciences**.

Dr. Rajiv Senapati

Supervisor

Acknowledgements

I would like to thank everyone who assisted in creating the weather analysis data for improved aviation visualization.

Additionally, I thank Ravi Senpati, my project supervisor who guided and assisted me during this period. Ravi Senpati was an important yet underestimated contributor to the writing of this report. Her expertise in commenting, encouraging, as well as giving remarks helped in proper completion and presentation of the work.

I appreciate my educational institution whose involvement led me to write this paper. Successful attainment of project requires library and computer facilities that provide collated data from wide range essential for huge data set.

Finally, I need to acknowledge my fellow staff members and other scholars whom we came across to make it complete. They had a lot of differences which added impetus to this mission.

Furthermore, I will be extremely thankful for my family members and close friends who stayed beside me through the most difficult moments of this project. They had always been inspiring and pushing me to believe in myself.

Lastly, I would like express my gratitude to everyone who has in one way or another contributed towards making this a realizable venture regardless of how small they are. I would like to take this opportunity to thank them because they indeed made a great difference in attaining success in this project.

Table of Contents

Abstract

This project report is focused on the area of data mining, which deals with "weather data analysis with improved flight visibility." The improvement of aviation safety is the core objective of this research project, and it seeks to explore utilizing data mining techniques in analyzing weather information that will aid in managing flight This report provides a detailed description of the methods of dealing with weather threats in aviation, while using lots of massive data, and obtaining relevant information from that data. The importance of project advisor's guidance and support throughout the challenging research process cannot be overstated. In addition, the peer collaboration and access to institutional resources contributed immensely to the success of this project. This report constitutes an extensive probe into the data mining-aviation nexus with useful suggestions on visibility management strategy for better flight safety.

Description

Weather dynamics exert strong force upon the overall air traffic complexity making the process more safe or dangerous for flight operation in the complicated environment of aviation. The challenge is managing the visibility during critical atmospheric circumstances such as low visibility situations especially in adverse weather where visibility is compromised. A data mining project with this title "Weather Data Analysis for Improved flight visibility management" sets out to delve into the complexities involved in data-based improvements of aviation security.

Aviation is ever changing and necessitates a frequent shift of the instruments for its management. Weather data is very huge as many people know and it is only in this context that the use of data mining techniques may yield meaning inferences from the huge amounts of weather data out there today. We look for a pattern that will help to illustrate how changing weather elements influence the extent of visual ranges.

The importance of this project is in its ability to facilitate the evolution, invention, and use of advanced tools and techniques which can assist aviation specialists to make justified choices while coping with difficult weather situations. We seek to comprehend and solve visibility management problems on flights by using data mining.

1. Introduction

Algorithm used

1.1 Apriori

The Apriori algorithm is a classic algorithm in the field of data mining and association rule learning. It is used to discover interesting relationships, patterns, and associations within large datasets. Specifically, the Apriori algorithm is employed for mining frequent itemsets and generating association rules.

Here's a brief overview of the key concepts associated with the Apriori algorithm:

1. Itemset: - The collection of items or elements is known as an itemset. For example, in the area of market basket analysis (one of the applications for the Apriori algorithm), products purchased by a customer would be the items.

2. Support: - The support of an itemset represents how often it's present in the dataset. The ratio of transactions that comprise the itemset for the dataset defines this index. An essential consideration in establishing the importance of an itemset is its support index.

3. Frequent Itemset: - Frequent itemset means a set of items that occurs with at least or equal to a minimally acceptable support. The Apriori algorithm starts from individual items, then it is increased in step by step until the whole set is covered for every possible length.

4. Association Rules: - An association rule is an expression of the form "if X, then Y," with X and Y being itemsets. These suggest that if item x is observed on the same transaction, then there would be an increased chance of seeing item y.

5. Confidence: - The confidence level is a parameter which expresses reliability of association rule. The proportion of transactions with X and Y over transactions with X is used in calculation as it shows high confidence or high association.

The Apriori algorithm is widely used in various applications, including market basket analysis, recommendation systems, and bioinformatics, where discovering associations among items is valuable for decision-making and pattern recognition.

1.2 Correlation Coefficient

Correlation coefficient is a measure used in data mining and statistic that defines linear relationship for two variable. The correlation is used for measuring the extent of change between two variables. The significance of a correlation coefficient as an essential indicator of relations among parameters in the data set cannot be overstated.

The most commonly used correlation coefficient is the Pearson correlation coefficient, often denoted by the symbol "r." It ranges from -1 to 1, where:

1: This shows that there is absolute positive linear correlation.
0: Indicates no linear relationship (correlation).
-1: Perfect negative linear association is indicated.

The correlation coefficient is useful for several purposes:

Identifying Relationships: An estimated correlation coefficient of close to 1 or -1 reveals a tight linear match between variables among which analysis identifies relationships in the data.

Feature Selection: The use of correlation analysis can help to narrow down on important features that have strong relation with the dependent variable or high correlation amongst themselves. Removing redundant, highly correlated features can help in simplifying model development.

Assessing Model Assumptions: Understanding the relationships between the input features is also very critical when building predictive models that are interpret and validated.

One should, however, bear in mind that simply because two things are related does not mean that one causes the other. Just because the two variables correlate does not automatically imply causation between those two. Besides, the Pearson's correlation is tailored to determine a straight line relationship only which disregards the other non-straight relationship forms. In the case of non-linear relationships, other correlation measures like the Spearman rank correlation may be utilized.

1.3 Decision Tree

One of the widely utilized machine learning algorithms within data mining and data analysis is the decision tree. Supervised learning algorithm applicable to classification and regressions. The decision tree model is constructed by dividing the datasets into smaller parts using specific feature values that eventually creates a tree-like structure with decisions.

Here are the key concepts associated with decision trees in the context of data mining:

1. Node: - A node in a decision tree is a decision point. It is equal to an attribute and the limit point.

2. Root Node: - The root node is the highest order node at the top of a decision tree. It is the initial branch or attribute of the partition.

3. Internal Node: - The subsequent decisions for particular features lead to internal nodes in a decision tree. Branching continues from these nodes towards other secondary branches.

4. Leaf Node: - The leaf nodes are the last nodes of a decision tree. They are the terminal leaf nodes of decisions that denote the last predicted value or class for a particular branch.

5. Splitting: - In splitting one divides these datasets along an attribute in relation of a given cut-point level. It involves selecting one point between different features that provide the highest discrimination of data in separate groups of outcomes.

6. Decision Criteria: - For each node, the corresponding decision criteria are established after comparing a certain characteristic with a fixed limit. Decision tree algorithm chooses attributes and thresholds optimally separating the data points into different classes.

7. Entropy and Information Gain: - For instance, most decision trees rely on quantities like entropy and information gain in choosing the best traits of a split.4 Informaton gain measures how much a feature reduces unenrty with prediction and hence classifiction.

8. Pruning: - In decision tree learning pruning is applied to avoid overfitting. This requires cutting off those branches or nodes that do not add significant improvements to bettering the model's prediction accuracy on unknown data.

Since decision tree is understandable and can be seen, it helps in identifying how the model arrives at its decision-making. In addition, they are versatile and applicable for both qualitative and quantitative data. Decision trees can be included within the ensemble model that comprises several decision trees towards improving prediction accuracy.

2. Methodoloy

1.1 Libraries Used

1.1.1 Pandas

Pandas is used for data manipulation and analysis. It helps in reading, cleaning, transforming, and analyzing structured data, which is essential for organizing and processing weather-related datasets.

1.1.2 matplotlib.pyplot

Matplotlib is a plotting library in Python that helps visualize data in various formats like line plots, scatter plots, histograms, etc. It's beneficial for creating visual representations of weather patterns and trends.

1.1.3 plotly.express

Plotly Express is another visualization library that provides an interface for creating interactive plots. It's useful for generating interactive graphs to explore weather data in-depth, allowing for zooming, hovering, and other interactive features.

1.1.4 mpl_toolkits.mplot3d

It is a submodule within Matplotlib that specifically deals with 3D plotting capabilities. It extends Matplotlib to support the creation of three-dimensional plots, allowing visualization of data in 3D space through various types of plots such as 3D scatter plots, surface plots, wireframes, and more.

1.1.5 matplotlib.patches

It is a toolset for creating and managing shapes like rectangles, circles, and polygons. It's used to highlight, annotate, or visually represent data in plots by adding various graphical shapes.

1.1.6 Seaborn

Seaborn is built on top of Matplotlib and provides a higher-level interface for drawing attractive and informative statistical graphics. It can be used for creating visually appealing and informative statistical plots to analyze weather-related patterns.

1.1.7 StandardScaler, PCA, KMeans

Scikit-learn is a machine learning library. StandardScaler helps in standardizing features by removing the mean and scaling to unit variance. PCA (Principal Component Analysis) is used for dimensionality reduction. KMeans is a clustering algorithm, potentially useful for grouping weather data into meaningful clusters based on certain features.

1.1.8 Numpy

Numpy is used for numerical computations in Python. It's handy for handling arrays and mathematical operations, which can be essential when dealing with numerical weather data.

1.1.9 %matplotlib inline

It is a Jupyter notebook command that displays Matplotlib plots directly within the notebook, ensuring the generated plots appear beneath the code that creates them without requiring additional commands for display.

1.1.10 mlxtend.frequent_patterns

Mlxtend is a library that includes various tools for machine learning. In this context, apriori and association_rules are used for association analysis and finding frequent itemsets within weather-related data.

1.1.11 Warnings

The warnings module helps manage or suppress warnings that might occur during data analysis or processing.

1.2 Algorithm

Input: - CSV file containing cleaned JFK weather data (`jfk_weather_cleaned.csv`)

Output: - Visualizations, analysis, and models for weather data exploration

STEP 1.

Import Required Libraries: - Import necessary Python libraries, including pandas, matplotlib, plotly, seaborn, scikit-learn, mlxtend, and others.

STEP 2.

Load Weather Data: - Read the cleaned JFK weather data from the CSV file into a Pandas DataFrame (`data`).

STEP 3.

Data Overview and Summary:

- Display the first few rows of the data using `data.head()`.
- Check for missing values with `data.isna().sum()`.
- Display summary statistics using `data.describe()`.

STEP 4.

Data Distribution Visualization: - Plot histograms for each column in the dataset using Seaborn and Matplotlib to visualize the data distribution.

STEP 5.

Data Scaling: - Use StandardScaler from scikit-learn to scale the non-date columns in the dataset.

STEP 6.

Visualize Scaled Data Distribution: - Plot histograms for each scaled column to visualize the distribution after scaling.

STEP 7.

Time Series Data Visualization: - Set the date column as the index and visualize the time series data for visibility using both Matplotlib and Plotly Express.

STEP 8.

Association Rules and Apriori Algorithm:

- Select relevant columns for association rules.

- Convert the data to binary format for Apriori algorithm.

- Apply Apriori algorithm using mlxtend.

- Print frequent itemsets and association rules.

STEP 9.

Correlation Analysis:

- Calculate and display the correlation matrix for selected weather variables.

- Define a custom function to calculate correlation coefficients for specific pairs.

- Calculate and print correlation coefficients for specific pairs ('DRYBULBTEMPF' vs. 'RelativeHumidity' and 'WindSpeed' vs. 'StationPressure').

- Visualize the correlation matrix using heatmaps.

STEP 10.

Attribute Analysis and Visualization:

- Calculate Euclidean distance between the target attribute ('VISIBILITY') and other attributes.

- Calculate Jaccard coefficients between the target attribute and other attributes.

- Visualize the distance matrix and Jaccard coefficient matrix.

STEP 11.

Principal Component Analysis (PCA) and K-means Clustering:

- Select relevant columns for clustering.

- Standardize the data and perform PCA.

- Plot the explained variance for different numbers of principal components.

- Apply K-means clustering and plot the error vs. the number of clusters.

STEP 12.

3D Clustering Visualization:

- Perform PCA for 3D clustering visualization.

- Apply K-means clustering and visualize clusters in a 3D scatter plot.

- Display variables contributing to each cluster.

STEP 13.

Decision Tree Induction:

- Define a class (`Node`) for representing nodes in a decision tree.

- Define functions for calculating information gain, entropy, and decision tree induction.

- Build a decision tree for predicting 'VISIBILITY' based on other weather attributes.

- Print and visualize the decision tree.

STEP 14.

Radar Chart for Mean Weather Variables:

- Select relevant columns for the radar chart.

- Calculate the mean for each column.

- Normalize mean values for better comparison.

- Plot a radar chart to visualize mean weather variables with respect to visibility.

3. Discussion

3.1 Data Ingestion:

	DATE	VISIBILITY	DRYBULBTEMPF	WETBULBTEMPF	DewPointTempF	RelativeHumidity	WindSpeed	WindDirection	StationPressure	SeaLevelPressure	Precip
0	2010-01-01 00:51:00	6.0	33	32	31	92	0	0	29.97	29.99	0.01
1	2010-01-01 01:51:00	6.0	33	33	32	96	0	0	29.97	29.99	0.02
2	2010-01-01 02:51:00	5.0	33	33	32	96	0	0	29.97	29.99	0.02
3	2010-01-01 03:51:00	5.0	33	33	32	96	0	0	29.95	29.97	0.02
4	2010-01-01 04:51:00	5.0	33	32	31	92	0	0	29.93	29.96	0.02
...	...	...	...	...	...	...	...	...	...	...	...
75078	2018-07-27 18:51:00	10.0	76	73	72	88	3	230	30.00	30.02	0.00
75079	2018-07-27 19:51:00	4.0	69	69	69	100	13	40	29.99	30.01	1.16
75080	2018-07-27 20:51:00	10.0	71	70	70	96	0	0	30.02	30.04	0.01
75081	2018-07-27 21:51:00	10.0	72	71	70	94	5	50	30.00	30.02	0.01
75082	2018-07-27 22:51:00	10.0	72	71	71	97	0	0	30.01	30.03	0.00

75083 rows × 11 columns

Here the data is imported om the file and displayed,
data = pd.read_csv("jfk_weather_cleaned.csv")

data later the data is described,

```
In [4]: data.isna().sum()
```

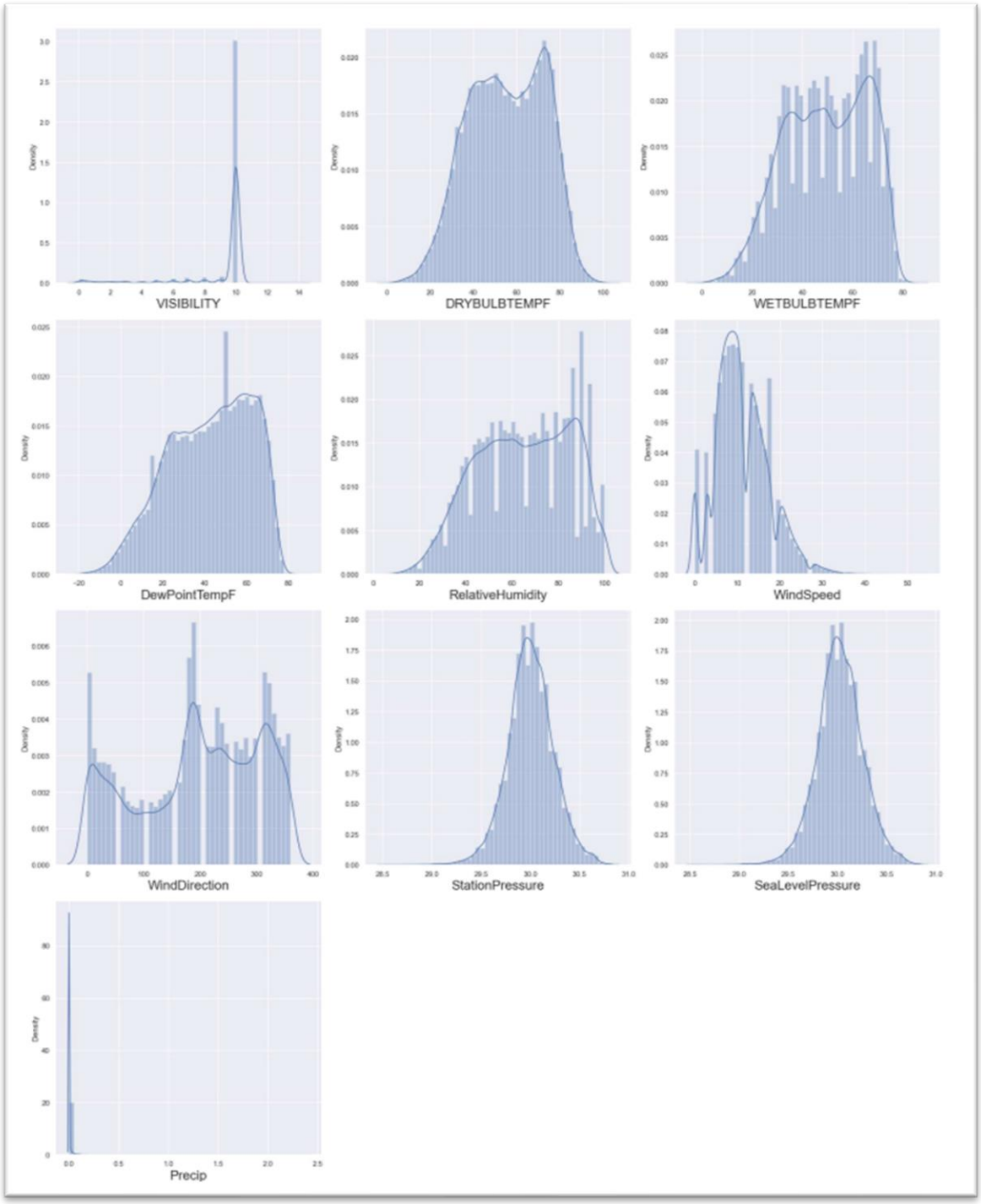
```
Out[4]: DATE                0
        VISIBILITY          0
        DRYBULBTEMPF        0
        WETBULBTEMPF        0
        DewPointTempF       0
        RelativeHumidity    0
        WindSpeed           0
        WindDirection       0
        StationPressure     0
        SeaLevelPressure    0
        Precip              0
        dtype: int64
```

```
In [5]: data.describe()
```

Out[5]:

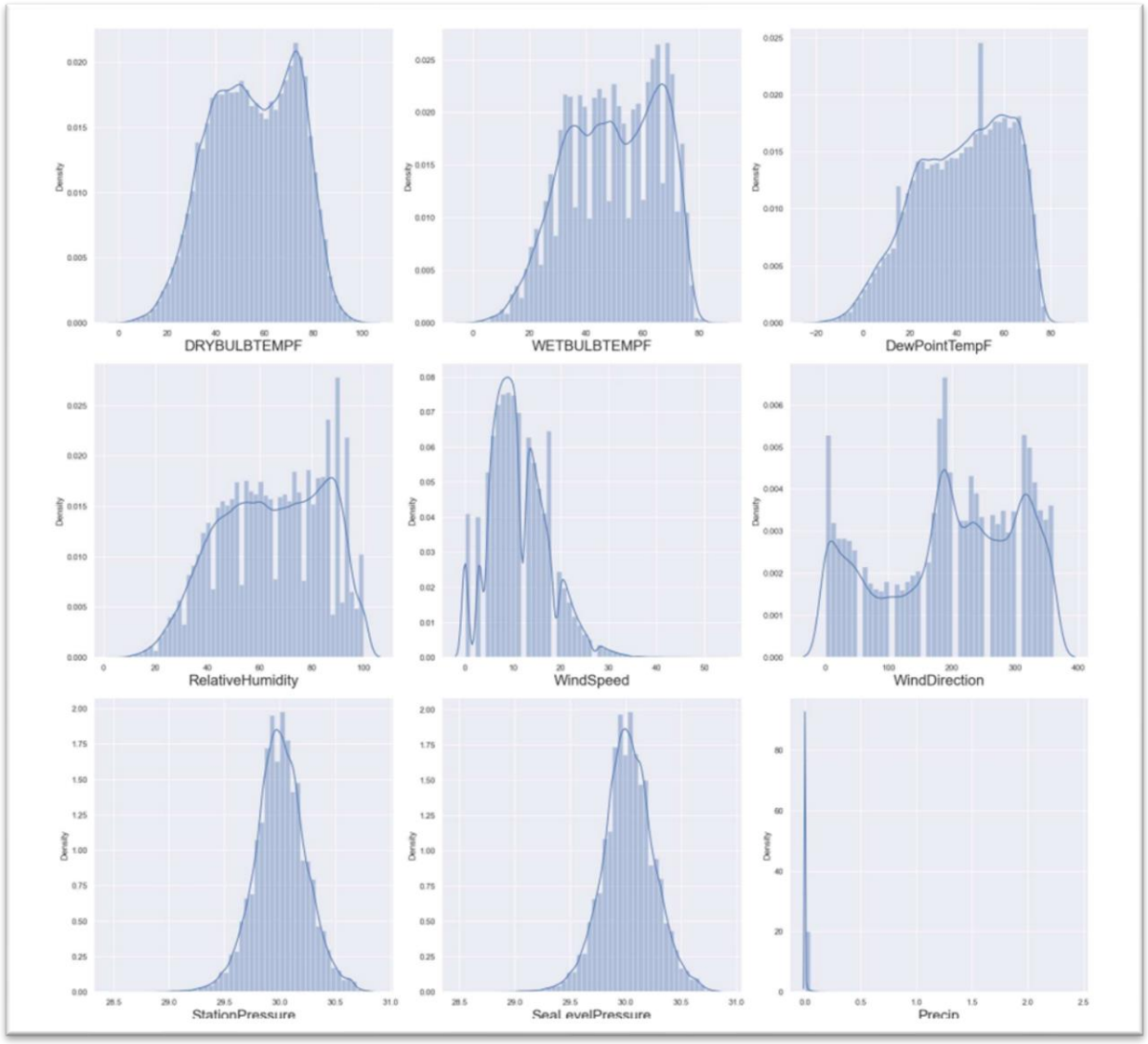
	VISIBILITY	DRYBULBTEMPF	WETBULBTEMPF	DewPointTempF	RelativeHumidity	WindSpeed	WindDirection	StationPressure	SeaLevelPressure
count	75083.000000	75083.000000	75083.000000	75083.000000	75083.000000	75083.000000	75083.000000	75083.000000	75083.000000
mean	9.211896	55.355527	49.327544	42.424024	64.812075	11.253240	196.550751	30.005579	30.026049
std	2.202311	17.394334	16.182867	19.577957	19.898962	6.101048	107.692804	0.235172	0.234069
min	0.000000	1.000000	-1.000000	-19.000000	8.000000	0.000000	0.000000	28.520000	28.540000
25%	10.000000	42.000000	36.000000	27.000000	49.000000	7.000000	110.000000	29.860000	29.880000
50%	10.000000	56.000000	50.000000	44.000000	66.000000	10.000000	200.000000	30.000000	30.020000
75%	10.000000	70.000000	64.000000	59.000000	82.000000	15.000000	290.000000	30.150000	30.170000
max	14.000000	102.000000	85.000000	84.000000	100.000000	53.000000	360.000000	30.830000	30.850000

3.2 Data Cleaning:



These subplots, each displaying the distribution of values for a specific column (excluding the 'DATE' column) in the dataset using seaborn's `distplot`. This visual representation allows for a quick overview of the data distribution for each variable in a clear and organized manner.

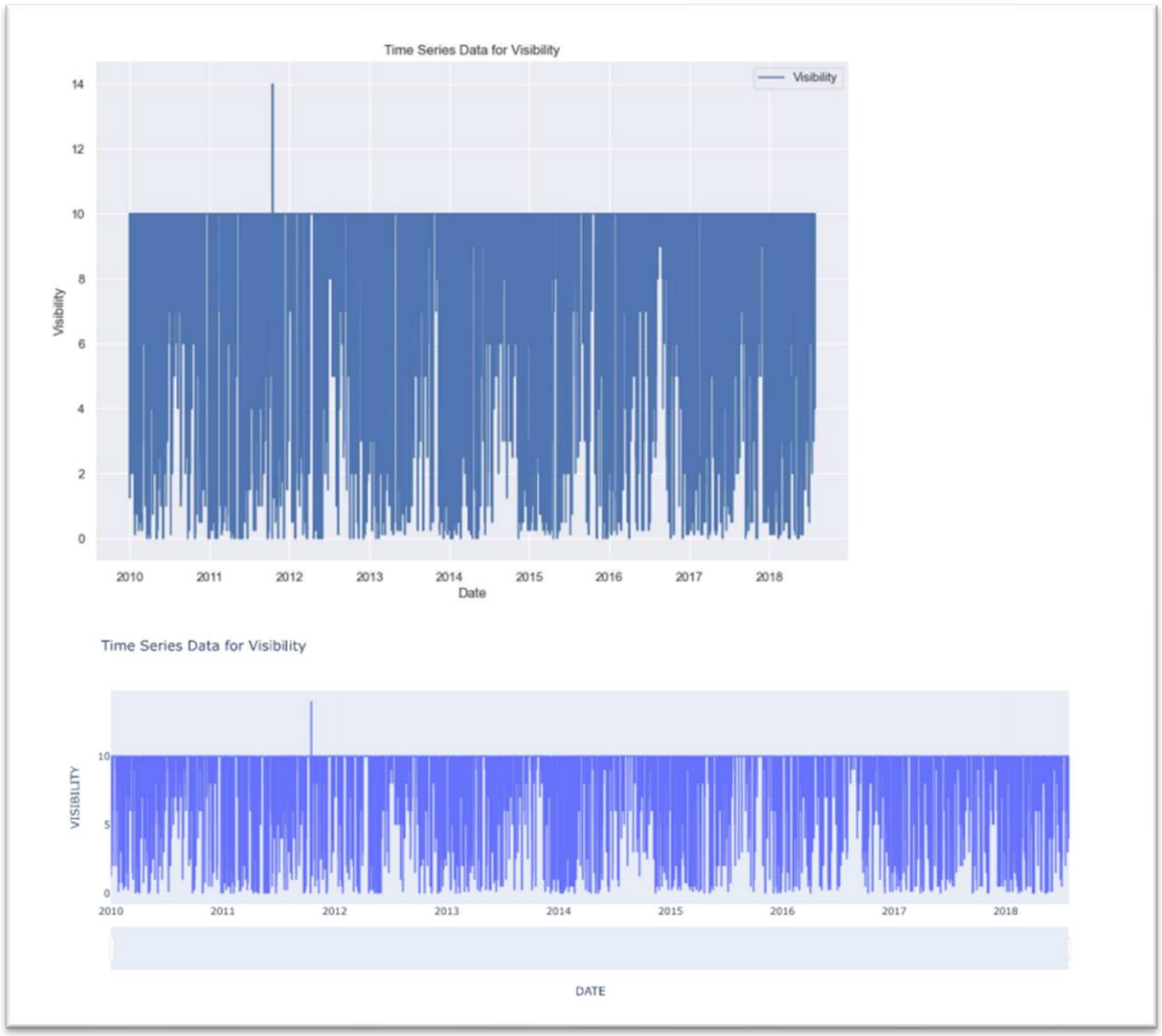
3.3 Data Intigration:



After Scaling the data and removing the date and Visibility the outcome is,

Here the grid of subplots, each displaying the distribution of values for a specific column in the dataset (`X`). The seaborn `distplot` function is used to visualize the data distributions, providing a concise overview of the variable behaviors.

3.4 Data Transformation:



The code uses a pandas DataFrame (`data`) with weather information from JFK airport, plotting the time series of visibility using both Matplotlib and Plotly Express. Matplotlib provides a traditional line plot, while Plotly Express creates an interactive line plot with a range slider for exploring the visibility trends over time.

3.5 Apriori algorithm & Frequent Itemset :

```
Frequent Itemsets:
    support     itemsets
0  0.907569        (0.0)
1  0.857584       (10.0)
2  0.100955       (30.0)
3  0.105483        (8.0)
4  0.108187        (9.0)
5  0.824501  (10.0, 0.0)


Association Rules:
  antecedents consequents  antecedent support  consequent support   support  \
0      (10.0)       (0.0)            0.857584            0.907569  0.824501
1       (0.0)      (10.0)            0.907569            0.857584  0.824501

   confidence      lift  leverage  conviction  zhangs_metric
0    0.961423  1.059338  0.046184    2.395988       0.393317
1    0.908472  1.059338  0.046184    1.555978       0.606014


Correlation Matrix:
                  VISIBILITY  DRYBULBTEMPF  WETBULBTEMPF  DewPointTempF  \
VISIBILITY          1.000000      0.063499     -0.034205      -0.129985
DRYBULBTEMPF        0.063499      1.000000      0.970013       0.888192
WETBULBTEMPF       -0.034205      0.970013      1.000000       0.969145
DewPointTempF      -0.129985      0.888192      0.969145       1.000000
RelativeHumidity   -0.465327      0.126035      0.351903       0.559575
WindSpeed           0.020778     -0.159370     -0.214081      -0.261974
WindDirection       0.173371     -0.125621     -0.201306      -0.270603
StationPressure     0.194537     -0.212059     -0.237525      -0.257653
SeaLevelPressure    0.194505     -0.209948     -0.235166      -0.257096
Precip             -0.293458     -0.000320      0.041163       0.077397

                  RelativeHumidity  WindSpeed  WindDirection  StationPressure  \
VISIBILITY                -0.465327   0.020778       0.173371         0.194537
DRYBULBTEMPF               0.126035  -0.159370      -0.125621        -0.212059
WETBULBTEMPF               0.351903  -0.214081      -0.201306        -0.237525
DewPointTempF              0.559575  -0.261974      -0.270603        -0.257653
RelativeHumidity           1.000000  -0.277406      -0.369105        -0.195860
WindSpeed                 -0.277406   1.000000       0.359374        -0.266817
WindDirection             -0.369105   0.359374       1.000000        -0.141398
StationPressure           -0.195860  -0.266817      -0.141398         1.000000
SeaLevelPressure          -0.199092  -0.267428      -0.141826         0.997309
Precip                     0.193205   0.057189      -0.075277        -0.119309

                  SeaLevelPressure    Precip
VISIBILITY                0.194505 -0.293458
DRYBULBTEMPF             -0.209948 -0.000320
WETBULBTEMPF             -0.235166  0.041163
DewPointTempF            -0.257096  0.077397
RelativeHumidity         -0.199092  0.193205
WindSpeed                -0.267428  0.057189
WindDirection            -0.141826 -0.075277
StationPressure           0.997309 -0.119309
SeaLevelPressure          1.000000 -0.122211
Precip                   -0.122211  1.000000
```

The code utilizes Apriori algorithm for frequent itemset and association rule mining on weather data. It selects specific weather columns, converts data to binary format, identifies frequent itemsets with minimum support, generates association rules based on lift, and prints the resulting itemsets, rules, and correlation matrix among selected variables.

3.6 Data Correlating:

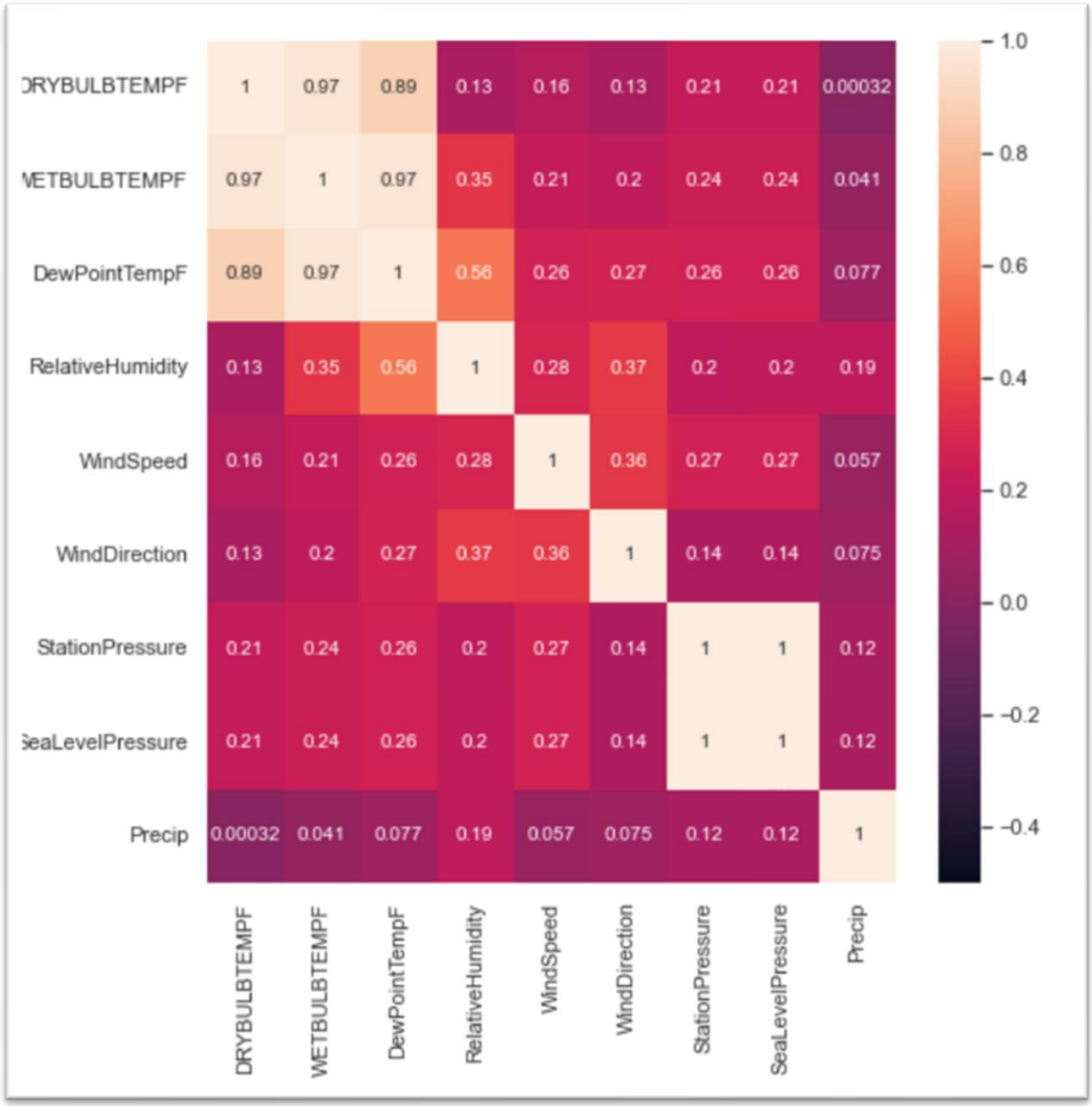
```
Correlation Matrix:
                  VISIBILITY  DRYBULBTEMPF  WETBULBTEMPF  DewPointTempF  \
VISIBILITY          1.000000      0.063499     -0.034205      -0.129985
DRYBULBTEMPF        0.063499      1.000000      0.970013       0.888192
WETBULBTEMPF       -0.034205      0.970013      1.000000       0.969145
DewPointTempF      -0.129985      0.888192      0.969145       1.000000
RelativeHumidity   -0.465327      0.126035      0.351903       0.559575
WindSpeed           0.020778     -0.159370     -0.214081      -0.261974
WindDirection       0.173371     -0.125621     -0.201306      -0.270603
StationPressure     0.194537     -0.212059     -0.237525      -0.257653
SeaLevelPressure    0.194505     -0.209948     -0.235166      -0.257096
Precip             -0.293458     -0.000320      0.041163       0.077397

                  RelativeHumidity  WindSpeed  WindDirection  StationPressure  \
VISIBILITY                -0.465327   0.020778       0.173371         0.194537
DRYBULBTEMPF               0.126035  -0.159370      -0.125621        -0.212059
WETBULBTEMPF               0.351903  -0.214081      -0.201306        -0.237525
DewPointTempF              0.559575  -0.261974      -0.270603        -0.257653
RelativeHumidity           1.000000  -0.277406      -0.369105        -0.195860
WindSpeed                 -0.277406   1.000000       0.359374        -0.266817
WindDirection             -0.369105   0.359374       1.000000        -0.141398
StationPressure           -0.195860  -0.266817      -0.141398         1.000000
SeaLevelPressure          -0.199092  -0.267428      -0.141826         0.997309
Precip                     0.193205   0.057189      -0.075277        -0.119309

                  SeaLevelPressure    Precip
VISIBILITY                0.194505 -0.293458
DRYBULBTEMPF             -0.209948 -0.000320
WETBULBTEMPF             -0.235166  0.041163
DewPointTempF            -0.257096  0.077397
RelativeHumidity         -0.199092  0.193205
WindSpeed                -0.267428  0.057189
WindDirection            -0.141826 -0.075277
StationPressure           0.997309 -0.119309
SeaLevelPressure          1.000000 -0.122211
Precip                   -0.122211  1.000000
Correlation between DRYBULBTEMPF and RelativeHumidity: 1.939887435883568e-08
Correlation between WindSpeed and StationPressure: -9.907107271888358e-06
```

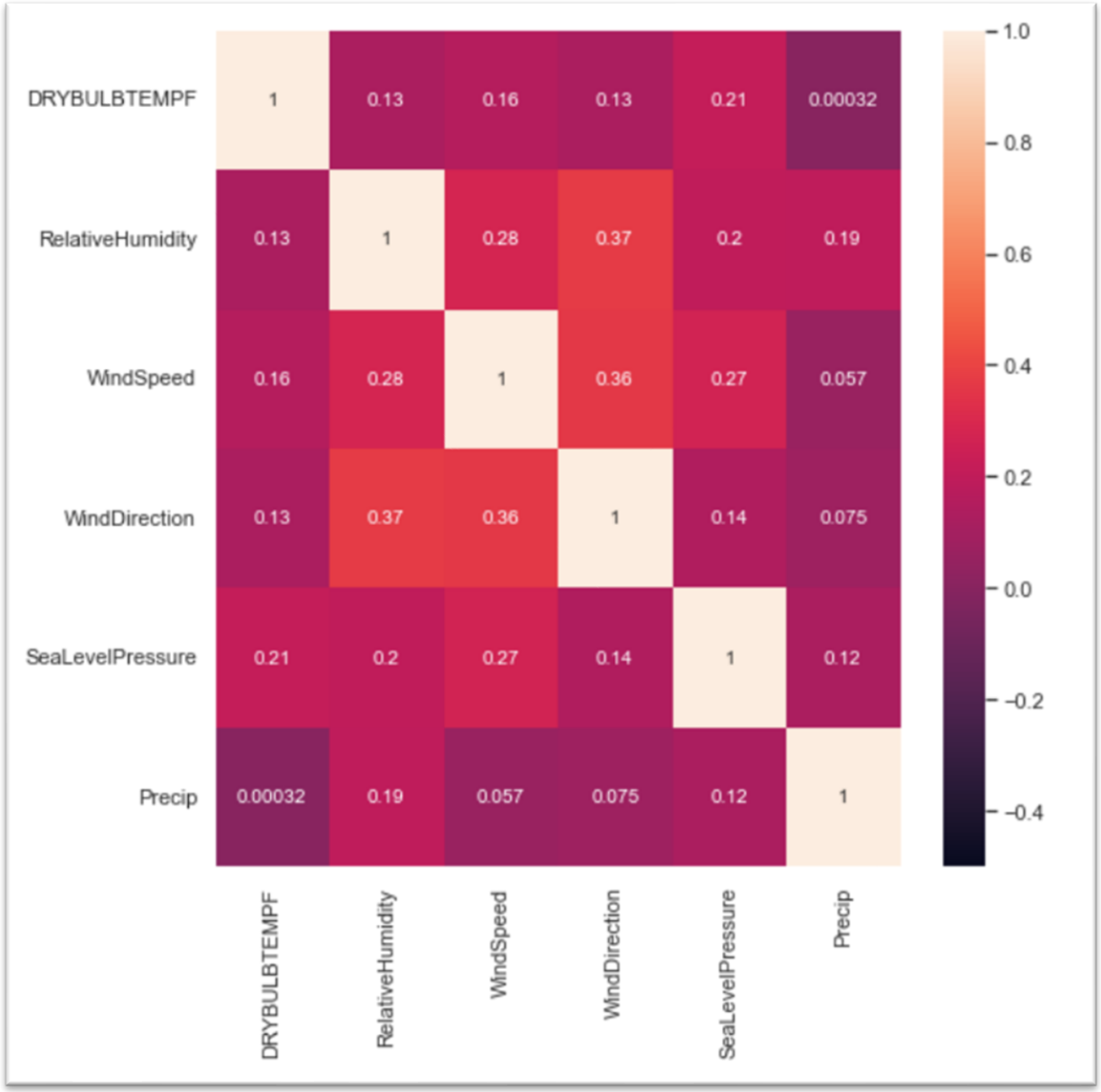
The code computes and displays the correlation matrix for selected weather variables. It then calculates and prints the correlation coefficients for temperature-humidity and wind speed-pressure pairs, offering insights into their relationships.

3.7 Correlation Heat Map:



This is a heatmap using Seaborn to visualize the absolute correlation matrix of the DataFrame `X`. It provides a concise representation of the correlation between different features, with annotations for each correlation coefficient. The note about the issue with matplotlib version 3.1.1 and its impact on annotation is included.

3.8 Data Selection (Heat Map):



The code generates a heatmap using Seaborn to visualize the absolute correlation matrix of the DataFrame `X_`. This heatmap provides a clear overview of the correlations between different features, with annotations for each correlation coefficient. The note about the issue with matplotlib version 3.1.1 and its impact on annotation is included.

3.9 Euclidean Distance:

	VISIBILITY	DRYBULBTEMPF	WETBULBTEMPF	DewPointTempF	\
VISIBILITY	0.000000	13512.410409	11875.964901	10620.940589	
DRYBULBTEMPF	13512.410409	0.000000	2026.325986	4316.392707	
WETBULBTEMPF	11875.964901	2026.325986	0.000000	2431.407206	
DewPointTempF	10620.940589	4316.392707	2431.407206	0.000000	
RelativeHumidity	16287.005544	7253.325858	7097.817481	7962.819852	
WindSpeed	1851.992527	13194.271787	11596.373528	10451.042723	
WindDirection	59161.410634	49251.359078	50700.355186	52619.721008	
StationPressure	5728.644202	8432.156565	6916.221572	6367.109376	
SeaLevelPressure	5734.225848	8427.421554	6911.782862	6364.014737	
Precip	2594.544972	15897.909421	14223.614886	12801.166025	

	RelativeHumidity	WindSpeed	WindDirection	\
VISIBILITY	16287.005544	1851.992527	59161.410634	
DRYBULBTEMPF	7253.325858	13194.271787	49251.359078	
WETBULBTEMPF	7097.817481	11596.373528	50700.355186	
DewPointTempF	7962.819852	10451.042723	52619.721008	
RelativeHumidity	0.000000	15904.760608	48190.873482	
WindSpeed	15904.760608	0.000000	58447.468440	
WindDirection	48190.873482	58447.468440	0.000000	
StationPressure	10992.468811	5409.191745	54350.001551	
SeaLevelPressure	10987.674429	5414.503681	54345.283421	
Precip	18575.522979	3505.977098	61410.790835	

	StationPressure	SeaLevelPressure	Precip
VISIBILITY	5728.644202	5734.225848	2594.544972
DRYBULBTEMPF	8432.156565	8427.421554	15897.909421
WETBULBTEMPF	6916.221572	6911.782862	14223.614886
DewPointTempF	6367.109376	6364.014737	12801.166025
RelativeHumidity	10992.468811	10987.674429	18575.522979
WindSpeed	5409.191745	5414.503681	3505.977098
WindDirection	54350.001551	54345.283421	61410.790835
StationPressure	0.000000	7.334262	8220.678709
SeaLevelPressure	7.334262	0.000000	8226.285206
Precip	8220.678709	8226.285206	0.000000

The code calculates the Euclidean distance between the target attribute ('VISIBILITY') and other specified attributes in the DataFrame 'data'. It constructs a distance matrix, where each element represents the Euclidean distance between corresponding attribute pairs. The resulting distance matrix is converted into a DataFrame for easy visualization and analysis of the relationships between the attributes.

3.10 Jaccard Coefficient:

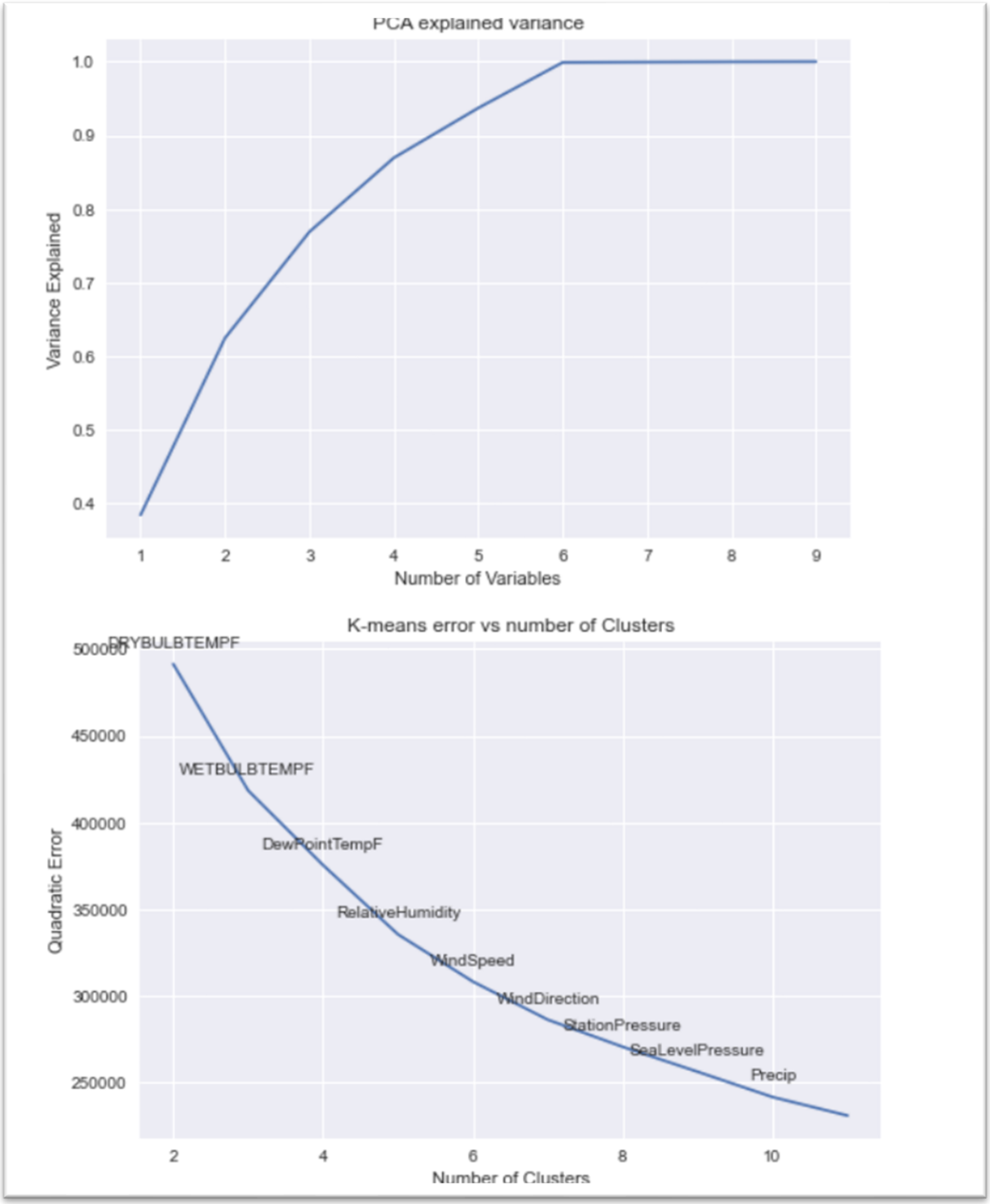
	VISIBILITY	DRYBULBTEMPF	WETBULBTEMPF	DewPointTempF	\
VISIBILITY	1.000000	0.099099	0.127660	0.108108	
DRYBULBTEMPF	0.099099	1.000000	0.807692	0.680328	
WETBULBTEMPF	0.127660	0.807692	1.000000	0.817308	
DewPointTempF	0.108108	0.680328	0.817308	1.000000	
RelativeHumidity	0.037736	0.882353	0.777778	0.649573	
WindSpeed	0.166667	0.407767	0.500000	0.417476	
WindDirection	0.036364	0.077519	0.078947	0.068702	
StationPressure	0.000000	0.006601	0.006969	0.006579	
SeaLevelPressure	0.000000	0.006623	0.006993	0.006601	
Precip	0.038462	0.000000	0.005780	0.005263	

	RelativeHumidity	WindSpeed	WindDirection	StationPressure	\
VISIBILITY	0.037736	0.166667	0.036364	0.000000	
DRYBULBTEMPF	0.882353	0.407767	0.077519	0.006601	
WETBULBTEMPF	0.777778	0.500000	0.078947	0.006969	
DewPointTempF	0.649573	0.417476	0.068702	0.006579	
RelativeHumidity	1.000000	0.400000	0.085470	0.006873	
WindSpeed	0.400000	1.000000	0.066667	0.008197	
WindDirection	0.085470	0.066667	1.000000	0.004184	
StationPressure	0.006873	0.008197	0.004184	1.000000	
SeaLevelPressure	0.006897	0.008230	0.004202	0.883721	
Precip	0.000000	0.007692	0.008065	0.000000	

	SeaLevelPressure	Precip
VISIBILITY	0.000000	0.038462
DRYBULBTEMPF	0.006623	0.000000
WETBULBTEMPF	0.006993	0.005780
DewPointTempF	0.006601	0.005263
RelativeHumidity	0.006897	0.000000
WindSpeed	0.008230	0.007692
WindDirection	0.004202	0.008065
StationPressure	0.883721	0.000000
SeaLevelPressure	1.000000	0.000000
Precip	0.000000	1.000000

The code calculates Jaccard coefficients between the target attribute ('VISIBILITY') and other specified attributes in the DataFrame 'data'. It constructs a Jaccard coefficient matrix, where each element represents the Jaccard coefficient between corresponding attribute pairs. The resulting matrix is converted into a DataFrame for easy visualization and analysis of similarity relationships between the attributes.

3.11 Plotting K-means error vs number of Clusters:



The code performs Principal Component Analysis (PCA) and K-means clustering on weather data. It plots the cumulative explained variance for PCA and the K-means clustering error versus the number of clusters. The annotations on the K-means plot

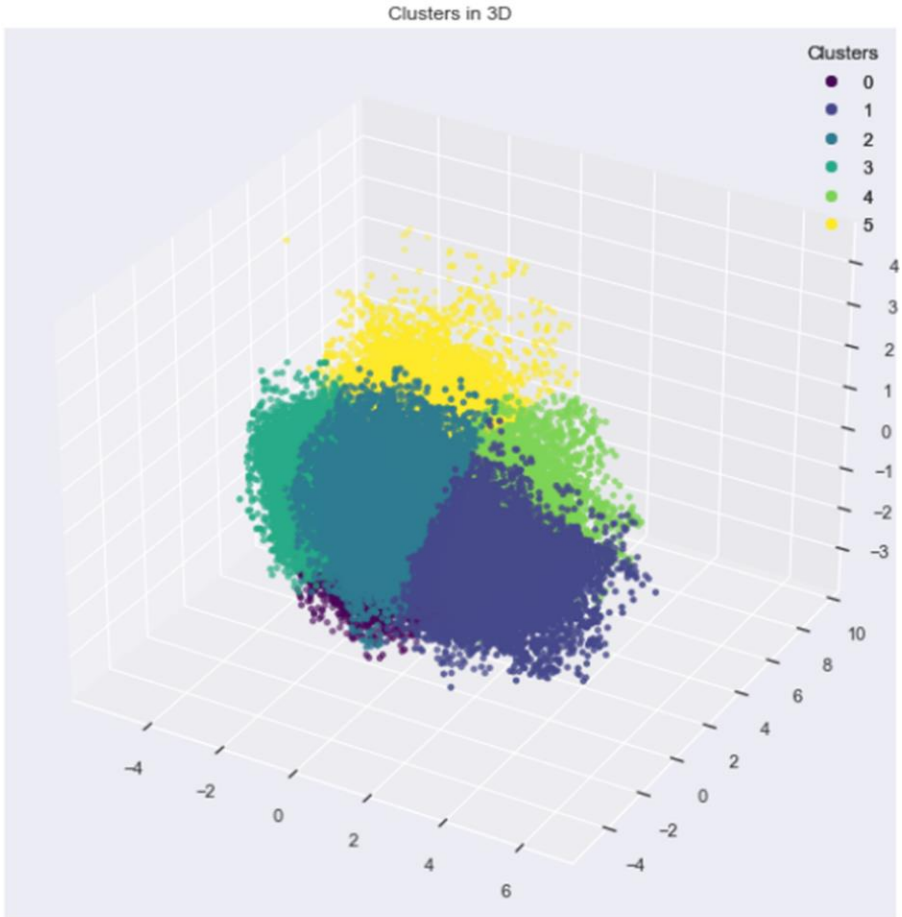
indicate the corresponding weather variables, providing insights into the optimal number of clusters and their variables' contributions.

3.12 3D-Variables contribution Clustering:

Variables contributing to each cluster:

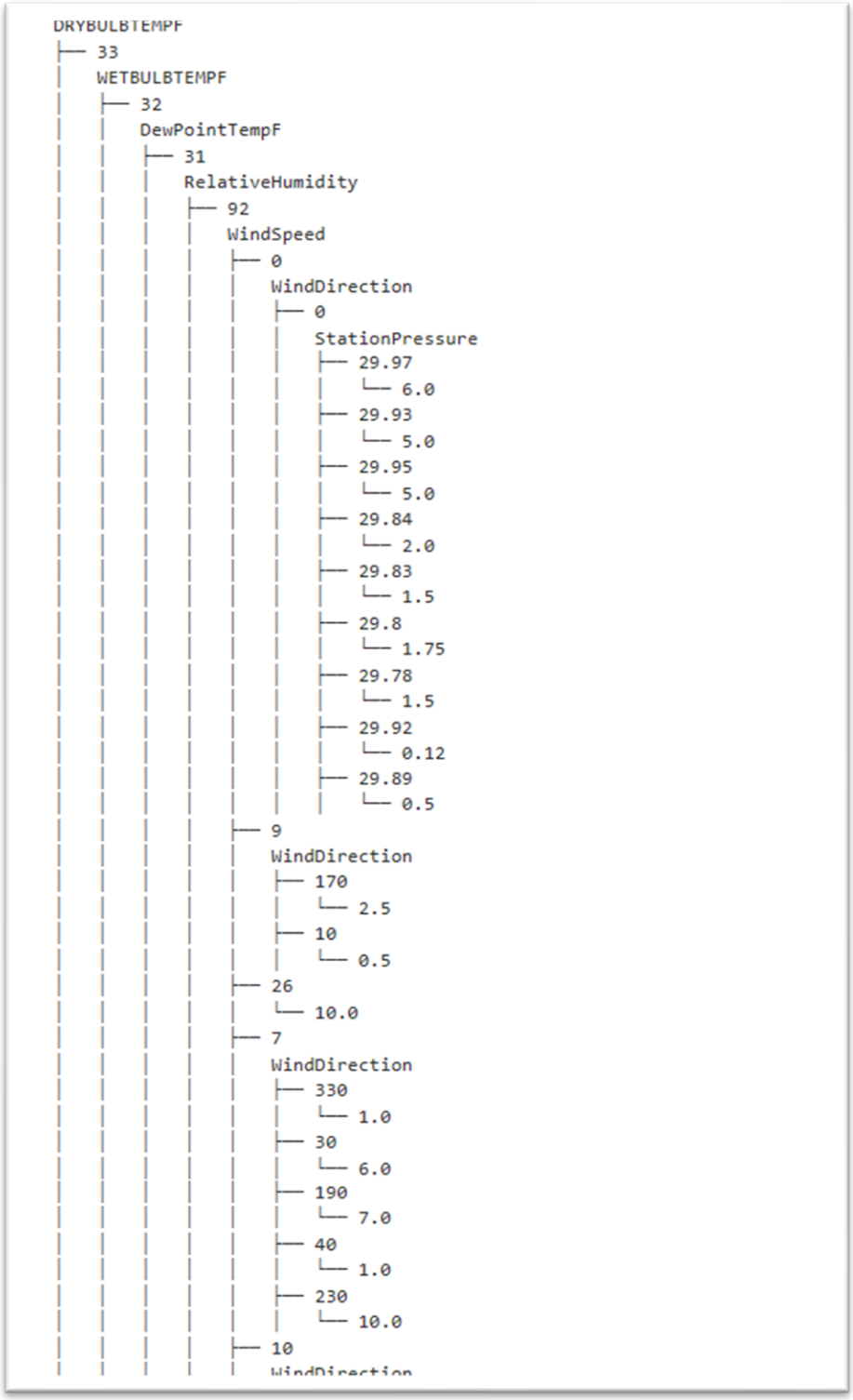
Cluster	DRYBULBTEMPF	WETBULBTEMPF	DewPointTempF	RelativeHumidity	\
0	1.03	0.85	0.62	-0.53	
1	-1.19	-1.33	-1.41	-0.93	
2	-0.36	-0.24	-0.10	0.42	
3	0.91	1.02	1.08	0.72	
4	-0.82	-0.97	-1.06	-0.82	
5	-0.45	-0.27	-0.07	0.69	

Cluster	WindSpeed	WindDirection	StationPressure	SeaLevelPressure
0	0.32	0.57	-0.12	-0.12
1	0.01	0.24	1.37	1.37
2	-0.67	-0.70	0.66	0.65
3	-0.51	-0.54	-0.26	-0.26
4	0.93	0.90	-0.32	-0.32
5	0.52	0.03	-1.54	-1.54



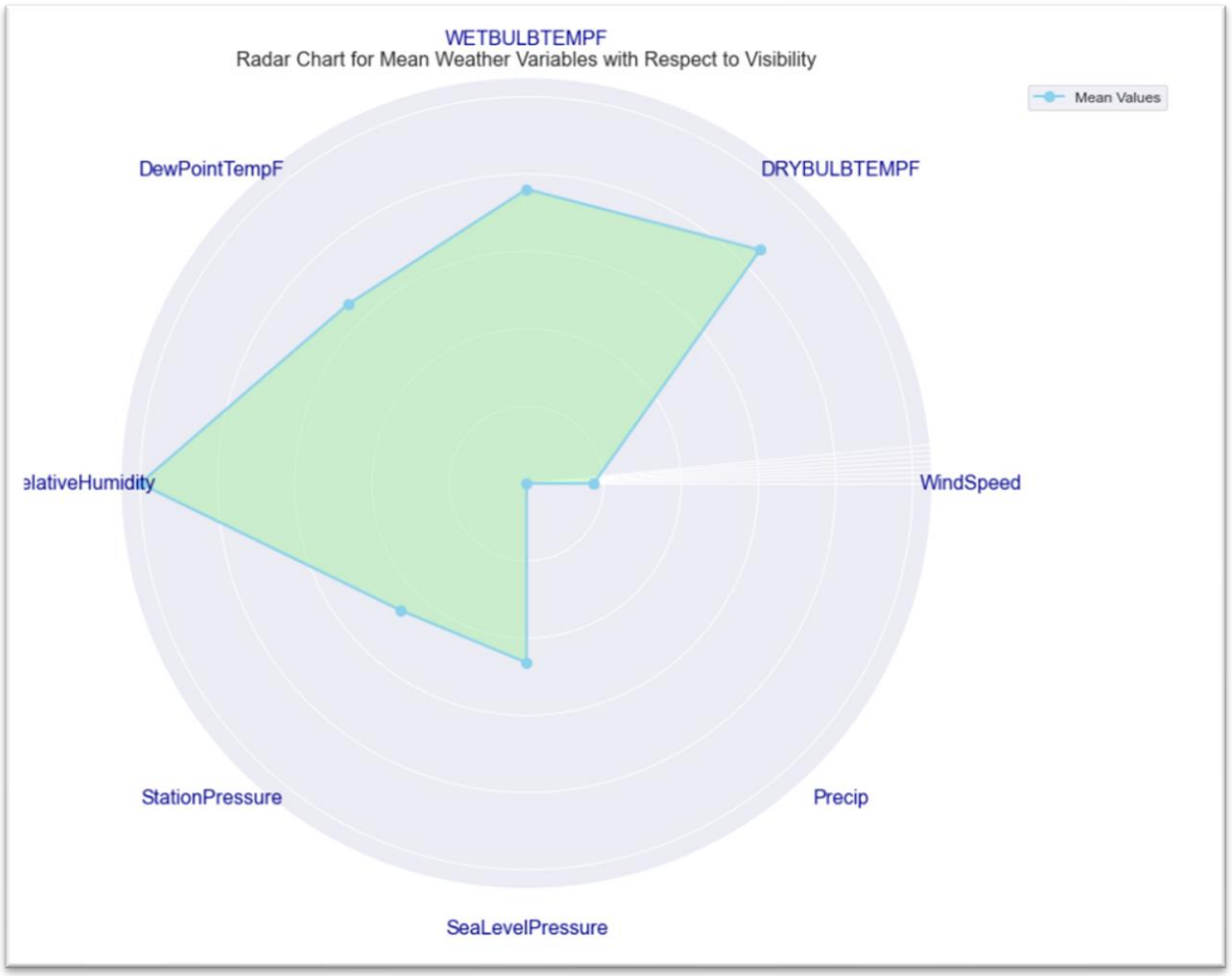
The code performs Principal Component Analysis (PCA) and K-means clustering on weather data, visualizing the clusters in a 3D plot. Each point represents a weather sample colored by its assigned cluster. The code also prints the variables contributing most to each cluster, providing insights into the key weather features defining each cluster.

3.13 Decision Tree Induction:



The provided code implements a basic decision tree algorithm for classification. It defines a `Node` class representing nodes in the decision tree and includes functions for calculating information gain, entropy, decision tree induction, and printing the resulting decision tree. The decision tree is built using weather data attributes to predict the 'VISIBILITY' target attribute. The tree is then printed in a hierarchical structure.

3.14 Graph Polar-Radar Chart:



The code generates a radar chart visualizing the mean values of selected weather variables with respect to visibility. It normalizes the mean values, plots them on a polar axis, and adds headings at vertex points. The chart provides a compact representation of the weather variable trends and their impact on visibility.

4. Concluding Remarks

In summary, investigating, as well as evaluating weather data about JFK has been a thorough expedition consisting of varied data mining and charting methods leading towards significant observations. The project was aimed at improving our knowledge of weather effects pertaining to the management of visibility during flights.

An initial data overview showed them having prepared and clean data that made sense as a basis for more of such analysis. The EDA techniques played a vital role in revealing the distribution and nature of meteorological elements. Moreover, they acted as a basis for the study that followed, enabling more clarity on the matter to be reached.

A complex study was conducted using techniques of machine learning and data mining including Apriori algorithm, association rule mining, PCA, and K-means clustering that enriched our comprehension of the dataset. Such techniques identified frequently occurring items, correlations, and clusters in the data, illuminating the intricate connections among various weather traits.

Furthermore, the process helped generate a predictive model for visibility with good explanatory value, revealing the variables affecting visibility under diverse climatic conditions.

A number of attribute analyses such as distancing calculations and Jaccard coefficients were incorporated in the exploration revealing the interconnections and contrasts observed among the various traits.

In addition, the project was summarized into a radar chart visualization that captured average weather variable such as visibility and gave a clear, but informative explanation of the dataset's central tendency.

Finally, the extensive study of JFK weather through different data mining methods has not only enhanced knowledge about weather processes but also it has application in aviation with regard to flight visibility. Diversity in applied methodologies broadens the weather data analysis spectrum and sets grounds for future use in the aviation as well as other industries.

5. Future Work

1. Integration of Artificial Intelligence (AI) and Machine Learning (ML):

Advanced Forecasting: AI algorithms can process vast amounts of historical weather data, identify patterns, and make predictions more accurately than traditional methods. Machine learning models can continuously learn and adapt, improving forecasting accuracy over time.

Risk Assessment: AI helps in assessing and predicting extreme weather events, such as hurricanes or floods. By analyzing historical data, machine learning models can identify potential risks, allowing for better preparation and mitigation strategies.

2. Enhanced Data Collection Methods:

Drones and Weather Balloons: Unmanned aerial vehicles (drones) and weather balloons equipped with sensors can be deployed to gather data from specific locations or altitudes that are challenging to reach otherwise. This enables a more granular understanding of atmospheric conditions.

Advanced Sensing Technologies: Integration of advanced sensors, such as LiDAR (Light Detection and Ranging) and radar technologies, provides detailed information on precipitation, wind patterns, and atmospheric composition. This enhances the accuracy of weather models and predictions.

3. Collaboration Among Stakeholders:

Shared Data Repositories: Establishing centralized repositories for weather data encourages collaboration by providing a common platform for meteorologists, researchers, and other stakeholders. This ensures that everyone is working with the most up-to-date and comprehensive information.

Improved Coordination: Enhanced collaboration among meteorologists, aviation authorities, and technology providers enables a more coordinated response to weather events. This is particularly crucial for aviation safety, as accurate and timely weather information is vital for flight planning and operations.

By integrating AI and ML, leveraging advanced data collection methods, and fostering collaboration among stakeholders, the field of meteorology can significantly improve its ability to predict and respond to weather-related challenges, ultimately enhancing safety and minimizing the impact of extreme weather events.

References

1. [Author name(s)], Year. Title. Journal Name. Volume and Page Numbers. DOI Link.
2. The following files are part of the NOAA Weather Data - JFK Airport hosted on IBM Developer Data Asset eXchange.

    Homepage:
    https://developer.ibm.com/exchanges/data/all/jfk-weather-data/
    Download link:
    http://s3.us-south.cloud-object-storage.appdomain.cloud/dax-assets-dev/dax-noaa-weather-data-jfk-aiport/1.1.2/noaa-weather-data-jfk-airport.tar.gz
    File list:
    -- jfk_weather.csv: the core dataset as obtained from NOAA.
    -- jfk_weather_cleaned.csv: data cleaned for non-numeric data and redundant fields or empty fields. NULLs have been replaced with the closest previous value.