**Air Quality Prediction**

**Mohammed Ali Sheriff Shaik**

**A04278033**

**Data Analytics, Prof: Dr. Scott King**

**Problem Statement:**

**Air Quality is declining progressively due to the rapid emission of pollutants into the air, such as SO, NO2, PM2.5, PM10, etc. We try to predict the next day’s data using the current day’s data. The Air Quality index is a metric used to define the quality of the air and tell us which pollutant has a significant role in it.**

**Dataset:**

**Used Aotizhongxin city’s data as training dataset and Huairou city’s data for validation and testing dataset. Resampling train and test data for 24hrs. Imputing missing data using KNNImputer. It is the best way to handle missing data for this dataset as it has few missing values and fills with the nearest value.**

**Dataset Attributes:**

**year: year of data in this row  
month: month of data in this row  
day: day of data in this row  
hour: hour of data in this row  
PM2.5: PM2.5 concentration (ug/m^3)  
PM10: PM10 concentration (ug/m^3)  
SO2: SO2 concentration (ug/m^3)  
NO2: NO2 concentration (ug/m^3)  
CO: CO concentration (ug/m^3)  
O3: O3 concentration (ug/m^3)  
TEMP: temperature (degree Celsius)  
PRES: pressure (hPa)  
DEWP: dew point temperature (degree Celsius)  
RAIN: precipitation (mm)  
wd: wind direction  
WSPM: wind speed (m/s)  
station: name of the air-quality monitoring site**

**Data Preparation:**

**Step 1: Load the dataset using pandas**

**Step 2: Convert Year, month, day, and hour as a DateTime and set it as index.**

**Step 3: Check for Missing values in the data frame and fill it using KNNImputer.**

**Step 4: Resample the data to 24hrs for all the pollutants.**

**Step 5: Calculate the Air Quality Index, which is the maximum of all the pollutant features.**

**Step 7: Feature Engineering for selecting features with low correlation for running in the machine learning model.**

**Step 8: Splitting the datasets, Aotizhongxin city’s data will be used as the training data and Huaroi city’s data will be split into validation and testing data in a 70:30 ratio.**

**Step 9: Use Xgboost as a training method and evaluate the model using SHAP.**

**Air Quality Calculation:**

**A screenshot of a computer

Description automatically generated with low confidence**

**According to the Chinese Government:**

**Average data by the day for all the pollutants that are the mean of 24hr, data. For PM2.5, PM10, SO2, NO2, CO. For ‘O3’, we take the mean of every 8hr, and the average of those values will give the index for the day. The maximum of these values is considered the Air Quality Index.**

**Missing Values: Chart, histogram

Description automatically generated**

**In the heatmap we can see for highly correlated attributes the null values doesn’t show on the chart.**

**Dendrogram of training data:**

**Diagram

Description automatically generated**

**Dendrogram Plays a vital role in understanding the correlation between features, predictors and predictants. If they are grouped at the same level it represents they are highly correlated.**

**Feature Correlation:**

**Chart

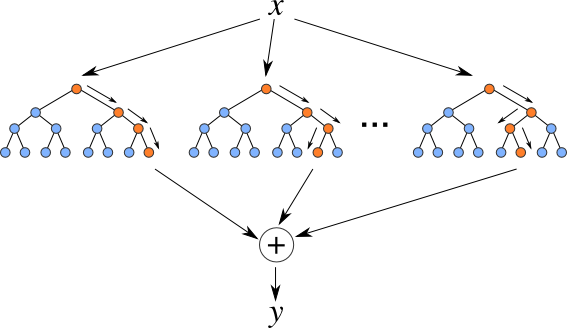
Description automatically generated**

**Pandas Profiler gives the report of which feature is correlated with which feature, helping in selecting features:**

**Graphical user interface, text, application, email

Description automatically generated**

**The final Features would be: PM2.5, Pres, WSPM, and DEWP. Temp is removed because it is highly correlated with all of the features.**

**Xgboost: Xgboost stands for Extreme Gradient boosting, it is a robust model. Additive tree model: add new trees that complement the already-built ones  Response is the optimal linear combination of all decision trees**

**Linear Regression:**

**Linear regression analysis predicts the value of one variable based on the value of another. The variable you wish to predict is referred to as the dependent variable. The variable you use to predict the other variable's value is known as the independent variable. Assumes data is Normally Distributed; check for Linearity. Predicted values are Normally Distributed.**

**Check for Linearity:**

**Calendar

Description automatically generated**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Model** | **HyperParameters** | **Mean Absolute Error** | **Mean Squared Error** | **R2\_Score** |
| **XGboost** | * **base\_score=0.5, booster='gbtree',** * **n\_estimators=20000,** * **objective = 'reg:linear',** * **early\_stopping\_rounds=100,** * **max\_depth=3,** * **learning\_rate=0.0001** | 0.0903 | 0.01284 | -1.077 |
| **Linear Regression** |  | 0.0673 | 0.0094 | -0.601 |

**Evaluating XGboost using SHAP(** **SHapley Additive exPlanations** **):**

**SHAP is a mathematical method for explaining machine learning model predictions. It is based on game theory concepts and can be used to explain any machine learning model's predictions by calculating the contribution of each feature to the prediction.**

**Chart, bar chart

Description automatically generated**

**A picture containing background pattern

Description automatically generated**

**Chart

Description automatically generated**

**Conclusion:**

**We tried to predict the Air Quality Index of the next day given the current day’s features(pollutants) such as SO2, NO, O3, CO etc. We used XGBoost and Linear Regression models on our data. By the measures of central tendency, we can say Linear regression works better than XGBoost. So, we evaluated the model to see what was wrong with our XGBoost model.**