Air Quality Quantification in Taiwan Using Machine Learning Techniques in Apache Spark Platform

MSc Research Project
Data Analytics

Sreenand Kandath
Student ID: X18137636

School of Computing
National College of Ireland

Supervisor: Dr. Catherine Mulwa

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name	:	Sreenand Kandath	
Student ID	:	X18137636	
Programme	:	MSc Data Analytics	Year: 2019-2020
Module	:	MSc Data Analytics - Research Project	
Supervisor	:	Dr. Catherine Mulwa	
Submission Due Date:		12/12/2019	
Word Count	:	7537 words	Page Count : 25

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:

Date : 13/12/2019

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Air Quality Quantification in Taiwan Using Machine Learning Techniques in Apache Spark Platform

Sreenand Kandath
X18137636

Abstract

In this era where oxygen is sold in bottles due to the deteriorated air quality outside, the exigency to reduce the air pollution, which proportionally increases the air quality is very high. This research is based on the historical air quality data of the Island of Taiwan. Various machine learning algorithms were manoeuvred to predict the PM2.5 with various meteorological factors which act as the main component in calculating the air quality index (AQI) . This project was able to determine the prediction accuracy of various regression models and ensemble models in Apache Spark environment and compare the performance of each model relative to the performance efficiency and root mean square error. Linear regression, neural network regression, decision forest, decision tree with boosted decision tree models - AdaBoost and gradient boosted trees were modelled in both Apache spark client and cluster environment. Through multiple comparison parameters ensemble models with Boosted Tree were found to be the best models in predicting the air quality index with a prediction accuracy of 80%.

Keywords : Apache Spark, AdaBoost, Neural Network Regression, PM2.5

1 Introduction

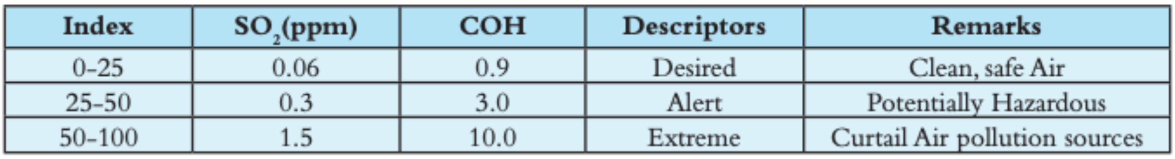
Air pollution is treated as one of the fatal health emergency all around the world. Eradicating air pollution is impossible, the only way to safeguard ourselves from the harmful effects of air pollution is by controlling the air pollutants and reducing it to the minimum magnitude possible. This is possible only with the equal participation of both the government and the public. This research concentrates on the Island of Taiwan, where the AQI index lies on the vulnerable zone. Historical air quality data of Taiwan was used to predict the AQI with the help of Apache spark in both client and cluster mode.

1.1 Motivation and Background

Asian countries are struggling against air pollution. It is estimated that around 537,000 premature deaths are happening in Asian countries because of air pollution. Even though air pollution is higher in urban areas, the people who suffer the most falls under the below poverty line and the people who lives in deteriorated air quality areas (Haq and Schwela, 2008). The Island of Taiwan is situated in the southern coast of China with a population of over 18 million. From the past few decades the face of Taiwan changed due to rapid industrialization and economic development, but this came with a boon – the air pollution caused by combustion of fossil fuels and electronic waste. Due to the high population density the risk of health hazards on the people is considered as a major problem in the recent years. (Chow, Watson and Chaung, 2012). Air pollution in an area is measured using the AQI (Air Quality Index) value. The AQI was implemented as an easy way to convey the people the quality of air around them. AQI is measured taking eight parameters into consideration (PM10, PM2.5, NO2, SO2, CO, O3, NH3, and Pb) , the AQI makes it possible to calculate real time air quality and which in turn helps the people understand about the air quality around them in any given time (Kumar, 2016). Table 1 presents how the AQI value is taken into consideration. Areas having below 25 AQI value is considered as clean and safe air zone (Central pollution control board, 2014) .

This research focuses on creating a model using various Decision tree and Regression models in Apache Spark to predict the AQI level in Taiwan. Machine learning algorithms are used widely in the prediction of AQI level, since the robustness of a model depends upon the quantity of data, performing these machine learning techniques in a distributed computing platform like spark using PySpark and scala with machine learning libraries will increase the accuracy and precision of the prediction by reducing the processing time. This research also focuses on the performance of distributed computing platforms when integrated with the machine learning algorithms.

Table 1 : Air Quality Analysis



Index	SO$_2$(ppm)	COH	Descriptors	Remarks
0-25	0.06	0.9	Desired	Clean, safe Air
25-50	0.3	3.0	Alert	Potentially Hazardous
50-100	1.5	10.0	Extreme	Curtail Air pollution sources

1.2 Research Question

The existing models focuses on forecasting and predicting the air quality for the near future, whereas this research concentrates on predicting the air quality considering various meteorological factors and understanding the factors that leads to the deteriorating air quality.

RQ : *“ To what extent can prediction of air quality be improved using machine learning methodologies (Multiple linear regression, ElasticNet regression, Lasso regression, Neural Network regression, Random forest, ExtraTrees regressor, AdaBoost and Gradient Boost Trees) to support the Taiwan government improve/enhance the life span of the people ? ”*

Currently the air quality index in Taiwan is above the limit (50–100) set by the World Health Organization.

Sub RQ : *“How can the performance of the machine learning models (Linear regression, Decision forest and gradient boost) be enhanced when implemented in Apache spark cluster mode ?”*

1.3 Objectives and Contributions

This project made it possible to understand about the Air quality in Taiwan and how the AQI index is calculated. Various reports and literatures were analysed to gather knowledge about the deteriorating air quality in Taiwan and the existing measure taken to improve the air quality.. Table 2 presents the implemented objectives in this project which helped solve the research question and the sub research question mentioned in section 1.2.

Table 2 : Objectives

Obj	Description	Evaluation Method
Obj 1	Literature review on the deteriorating air quality in Taiwan and existing air quality prediction models .	
Obj 2	Data preprocessing and performing exploratory analysis to calculate the AQI index by Understanding the Meteorological factors influencing the Air Quality Index in Taiwan.	
	Sub Obj (i): Preparing the data for analysis by transferring the data from local to MySQL and then to HDFS using Sqoop.	
	Sub Obj (ii) : Perform detailed exploratory analysis using univariate and bivariate analysis.	Box plot, Correlation matrix, Histograms.
	Sub Obj (iii): Find out the meteorological factors affecting the PM2.5 level.	Correlation Matrix
Obj 3	Implementing AQI prediction model in Apache Spark client Mode. Sub Obj (a): Building models using Regression Techniques 1) Linear regression 2) Neural network regression 3) Elastic Net regression Sub Obj (b): Building model using Various Tree Models 4) Random Forest 5) Extra Trees 6) AdaBoost	
Obj 4	Implementing AQI prediction model in Apache Spark cluster Mode. Sub Obj (i) Linear Regression (ii) Random Forest (ii) Gradient Boosted Trees (GBT)	
Obj 5	Evaluation of the Developed Models.	RMSE, MSE, MAE and regression scores
Obj 6	Comparing the performance of Apache Cluster mode and Client Mode.	Processing time

Contributions : The major contribution of this project is the air quality prediction model for the Island of Taiwan using Linear regression, Neural network regression, ElasticNet regression and ensemble models like Random forest, ExtraTrees, AdaBoost and Gradient boost techniques. This project also discusses about the performance of above machine learning techniques in Apache spark client and cluster environment.

The rest of the technical report consist of various sections where further process of the projects are deeply explained. Chapter 2 has the related works followed by chapter 3 which explains the altered methodology and the design architecture, chapter 4 where the data collection and pre-processing steps are explained followed by Chapter 5 which includes the implementation, evaluation and results obtained. Chapter 6 is the discussion and comparison of existing and adopted models, finally chapter 7 includes the conclusion and future work.

2 Related Work

Various researchers have conducted research on air quality prediction using different machine learning models which includes classification models, regression models and time-Series. In this section few works done by researchers are discussed in order to provide back bone to the research. The section begins by reviewing the need to develop a AQI prediction model followed by different models implemented by various scholars and researchers.

2.1 Review on Air Pollution and Its Harmful Effects On Humans

Air pollution is considered as a key factor in many long term diseases, The harmful air pollutants are so light that these particles can enter the human respiratory system through the nasal cavity and cause serious issues to the respiratory system. Many believe that air pollution can cause only skin and eye related issues due to the lack of knowledge, apart from these short term damages there are lot of medical problems triggered by air pollution which can even lead to death (Cohen *et al.*, 2005). According to a study conducted by WHO (World Health Organization) , the death toll due to air pollution is greater than that of the deaths due to AIDS. In urban areas air pollution is found to be more harmful, new born babies and kids below 5 years are affected by air pollution due to their high breathing capacity which forces them to intake more oxygen due to the developing respiratory system. Morning exercises and physical activates escalates the rate of air pollutants being penetrated into the body, similarly people who are suffering from high BP (Blood Pressure) and diabetes are more viable to be affected by air pollution. (Sierra-Vargas and Teran, 2012).

It is often believed that human are the root cause of air pollution. According to (Davidson, Phalen and Solomon, 2005) natural disasters can also act as a catalyst in the increasing rate of air pollution. A catastrophic forest fire in Canada in 2002 increased the rate of PM2.5 in the atmosphere to a great extent, Other natural disasters like the dust storm in Africa in the year 2001 lead to the rise in air pollution in the surrounding areas as the winds can carry the dust particles very long distance.

A group of researchers from the University of Washington orchestrated a study on a group of women for a duration of 9-10 years . Women with no history of cardiovascular or respiratory problems in their early stage of life were selected. After following their medical records for the next 9 years it was found out that the women who were diagnosed with respiratory and cardiovascular illness lived in areas were the air was polluted to a great extent (Bernstein *et al.*, 2004).

2.2 An Effective Method To Reduce The Harmful Effects Of Air Pollution

As discussed above air pollution can be extremely dangerous, it has to be controlled and reduced to the maximum extent possible. This is only possible with the help of people. People living in the rural areas are less known about the facts of air pollution and the main source of knowledge is through newspapers and even in some areas there is lack of newspaper circulation. According to (Bladen and Karan, 1976) people living in the urban areas have much better knowledge about the hazardous gases and air pollutants when compared to the rural people. In Malaysia when people were asked about the air quality around them, the feedback was surprising, almost all the people claimed that the air around them is clean and safer to breathe, the air in the surrounding areas were tested and found out that AQI index lies in the danger zone. When this was conveyed to the public , people came forward to take part in air pollution prevention measures. From this it is understood that when people are given right information about the consequences of the air pollution, they will try to reduce and control air pollution. (Anderson, 2009).

PM2.5 (particulates with diameter smaller than 2.5 μm) are kind of air pollutants that are originated mainly from the combustion of fossil fuels. It is observed that the concentration of PM2.5 is greater in areas where there is heavy traffic. When a study was conducted in Northern Taiwan to study the relationship between PM10 and PM2.5 it was found that the correlation between these two were high during the time period September – April, when the airflow was low with very less moisture. Taiwan being the second densely populated city with a wide growth it is a proven fact that the rise in PM2.5 is due to the intervention of fossil fuels (Liu, 2002) .

2.3 An Investigation on Apache Spark Machine Learning Process

Apache Spark is a computing engine which can run in both cluster and client mode. Spark is fully developed by Apache, it is one of the most reliable, robust and fastest distributed computing engine. While compared to the predecessor Hadoop, Spark uses in-memory computing rather than disk-based engines like the ones present in Hadoop. Apache Spark is widely used for data analysis tasks, there are various libraries and packages developed for spark to catalyst the data analysis process. Tools like MLib, Spark SQL and GraphX are few of the packages developed for data analysis in spark. From the tests conducted it was found that spark performed 100x faster than Hadoop during machine learning tasks. Apache spark also performs well in data streaming tasks with very less latency (Shoro and Soomro, 2015).

Machine learning is an iterative computing task and requires reloading the data each time when one process is passed (Lamba *et al.*, 2014).Most of the traditional big data frameworks like Hadoop map reduce are limited to this feature. In Apache spark with the presence of in-memory distribution, the data can be refreshed with a very low latency time, this is made possible by storing the data in cache instead of the disk. In spark when data is loaded each partition acts as a segment of the data and will be saved in available cache memory and the rest in disk . During machine learning tasks the training data will be stored in the slave node, this increases the speed of the process by accessing the required slave node other than accessing the entire dataset and enables parallel computing (Lin, Lee and Lin, no date).

MLlib is a library developed to perform machine learning tasks in the spark environment, MLlib acts as a platform in spark providing fast and robust common algorithms to run in a distributed mode. Currently the MLlib can perform various algorithms such as Linear models, Naïve Bayes, Classification, Regression and ensembles. Another significant feature of MLlib is the pipeline structure included in the framework. The pipeline is constructed in such a way that the users can use their own customized algorithms by swapping out the pre-loaded algorithms which enables high level tuning, this is made possible by a package spark.ml The MLlib is designed in such a way that the features provided by spark are utilized to the maximum, spark supports various tasks that are required by the spark.ml packages like high level data integrity using SQL and graph processing using the GraphX API (Meng *et al.*, 2016).

2.4 Critique On Existing Methods, Models and Techniques on Air Quality Index

The Air Quality in Beijing was predicted using support vector regression and random forest regression, both the models performed well, but the random forest regression model performed better when the sample dataset was of larger size (Liu *et al.*, 2019). The models developed by Liu et al was able to bring down the *Root Mean Square Error to 7.66* for SVR and *9.60* for Random Forest by checking the correlation between the predictor variables used to predict the PM2.5. (Corani, 2005) used pruned Neural network and Lazy learning algorithms to predict the Air quality in Milan. The data set had 4 years of historical air quality data and prediction model was developed on the basis of seasonal analysis which found that PM10 showed a rise during the winter than the summer season. The Author used feed-forward neural networks to compare the performance with prunes neural networks and lazy learning. It was observed that the FFNN model showed tendency to overfit the model when the number of parameters were increased. The LL and PNN performed almost the same while PNN requires complex development with high computational speed while LL was easy to model and convenient to interpret. (Samadianfard *et al.*, 2013) proposed a model using MLR (Multiple Linear Regression) to predict the ozone level in Spain. The research corresponds to the prediction of AQI index since ozone is one of the main factor in calculating the AQI index. Few limitations were found for the MLR model, over estimation was one of them. The author had also implemented GEP model which showed a much better prediction accuracy than MLR when NMSE, R and FA2 were taken into consideration. MLR performed better when the time frame was short. This implies that MLR model performs less efficiently when working with historical data with long time frames. Multiple Additive Regression (MART) , LSTM and RNN models were used to predict the PM2.5 value in Tehran. The LSTM model outperformed the the other two models with a *R-Squared value of 0.8* and *RMSE 8.91* and MART models performed with the least *R-Squared value of 0.54*, the Mart model used bagging technique and the final model had hundreds and thousands of branches. This was done on the assumption that randomization could obtain a better result (Kaimian *et al.*, 2019).

2.5 Identified Gaps and Conclusion

The above reviews explains that there is a great need to develop a AQI prediction model for the Island of Taiwan as proven from Section 2.5 . (Samadianfard *et al.*, 2013) built a model using Multiple Linear Regression which performed well, but with a drawback of showing poor performance with long time-frame.(Corani, 2005) implemented PNN and FFNN models which are complex neural network models to predict the AQI. The models by (Kaimian *et al.*, 2019) concludes that LSTM outperformed the MART models with highest R-Squared value of 0.80. Section 2.6 conveys the performance of Apache Spark and how machine learning can be implemented in Spark.

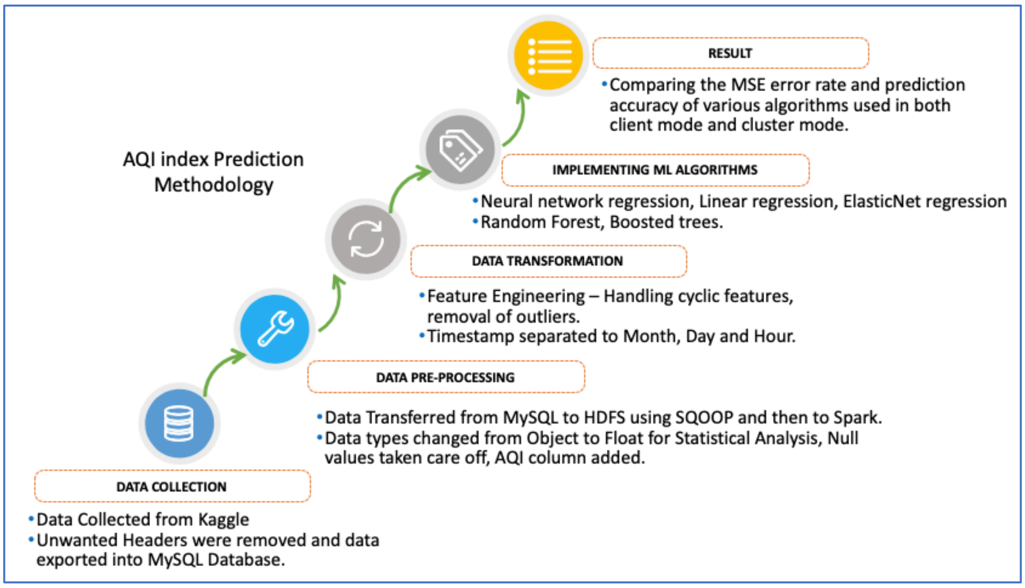
Based on literature the gaps identified are rectified by implementing boosted and regularized regression techniques as well as simple neural network regression to solve the air quality issue in Taiwan.

3 Air Quality Prediction Methodology and Design Specifications

3.1 Air Quality Prediction Methodology Approach

The methodology section includes the modified methodology approach for this research. The research follows KDD methodology, the traditional design is modified according to the steps followed in this research as presented in figure 1. KDD is one of the methodology used in the area of machine learning to develop different models. Data mining is a part of the KDD process. Following a clearly modified methodology approach helps in building a robust model by proper implementation of various stages at the proper time, The model can only be developed with prior understanding of the domain and client goals (Santos, 2018) .The modified methodology consist of five stages.

The methodology is structured in such a way that each stage is given equal importance, starting from data collection which is the first stage of the research, second is the data pre-processing, third data transformation which includes feature engineering, fourth the implementation stage and finally the result .



- Comparing the MSE error rate and prediction accuracy of various algorithms used in both client mode and cluster mode.
- Neural network regression, Linear regression, ElasticNet regression
- Random Forest, Boosted trees.
- Feature Engineering – Handling cyclic features, removal of outliers.
- Timestamp separated to Month, Day and Hour.
- Data Transferred from MySQL to HDFS using SQOOP and then to Spark.
- Data types changed from Object to Float for Statistical Analysis, Null values taken care off, AQI column added.
- Data Collected from Kaggle
- Unwanted Headers were removed and data exported into MySQL Database.

Figure 1 : Air quality index prediction methodology

3.2 Design Specification

The process flow architecture was designed for the prediction of AQI in Taiwan to increase the robustness and accuracy of the research. The architecture follows the methodology in Section 3.2 and has a three tier structure. The first layer in the architecture is data composition layer where the data is imported and pre-processed followed by the logic layer which acts as the core of the entire project and finally the third layer called as the client side is where the results obtained from the implementation part is presented and discussed using various tools. The main goal of the research is to predict the AQI with respect to the concentration of PM2.5 in the atmosphere of Taiwan which is only possible with a strong architecture design as presented in figure 2.

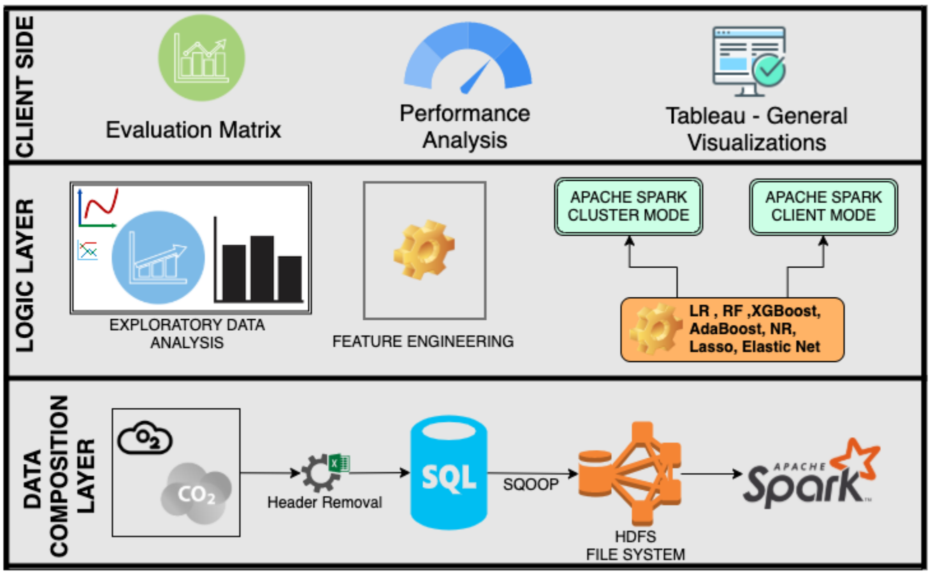


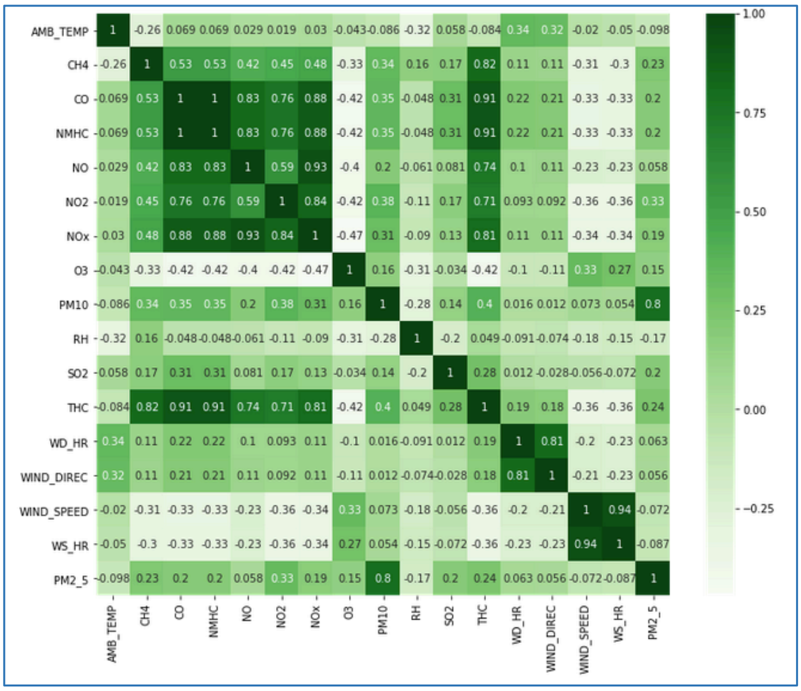
Figure 2 : Design architecture

4 Data Collection, Exploratory Data Analysis and Feature Selection

The Taiwan Air Quality data is collected from Kaggle, dataset consists of 23 columns and 218640 rows which includes the observations from 25 observatories in Taiwan. The data is recorded in one hour gap from each of the 25 monitoring stations. Once the data is downloaded unwanted headers are removed and is exported into *MySQL* database. The data stored in the *MySQL* database is transferred to *the HDFS (Hadoop distributed file system)* using *sqoop* pipeline. Since the data will be stored as a text format in HDFS, proper measures were taken to import the data as data frame into *PySpark* . The data types were changed from object to float for related fields and null values were removed permanently which reduced the number of rows to 214000 .

Exploratory data analysis were performed to get a clear picture of the data. The anomalies, patterns and hypothesis are checked during this stage. The identified anomalies and problems are rectified using feature selection and feature engineering techniques. Since the data was timely data, time series method was used to find the outliers in the data. Once the data was transformed into a data frame, feature engineering techniques were performed to find out the features that has most impact on *PM2.5*. *PM2.5* was taken as the target variable since it is considered as one of the most harmful pollutant in Taiwan (Section 2.2). AQI columns were added which calculated the AQI and categorized it as *GOOD, MODERATE* or *BAD*.

The dataset contains 17 variables, which includes different meteorological factors and various pollutants rate which constitutes the raise in *PM2.5*. Some of these variables are highly correlated with each other. The variables *NOx, CO, NMHC, THC* are omitted as they are highly correlated with the variable '*NO'* as detailed in figure 3. *PM10* is highly correlated with the predictor variable *PM2.5*. This variable cannot be removed because it has valuable information about the predictor variable.



	AMB_TEMP	CH4	CO	NMHC	NO	NO2	NOx	O3	PM10	RH	SO2	THC	WD_HR	WIND_DIREC	WIND_SPEED	WS_HR	PM2_5
AMB_TEMP	1	-0.26	0.069	0.069	0.029	0.019	0.03	-0.043	-0.086	-0.32	0.058	-0.084	0.34	0.32	-0.02	-0.05	-0.098
CH4	-0.26	1	0.53	0.53	0.42	0.45	0.48	-0.33	0.34	0.16	0.17	0.82	0.11	0.11	-0.31	-0.3	0.23
CO	0.069	0.53	1	1	0.83	0.76	0.88	-0.42	0.35	-0.048	0.31	0.91	0.22	0.21	-0.33	-0.33	0.2
NMHC	0.069	0.53	1	1	0.83	0.76	0.88	-0.42	0.35	-0.048	0.31	0.91	0.22	0.21	-0.33	-0.33	0.2
NO	0.029	0.42	0.83	0.83	1	0.59	0.93	-0.4	0.2	-0.061	0.081	0.74	0.1	0.11	-0.23	-0.23	0.058
NO2	0.019	0.45	0.76	0.76	0.59	1	0.84	-0.42	0.38	-0.11	0.17	0.71	0.093	0.092	-0.36	-0.36	0.33
NOx	0.03	0.48	0.88	0.88	0.93	0.84	1	-0.47	0.31	-0.09	0.13	0.81	0.11	0.11	-0.34	-0.34	0.19
O3	-0.043	-0.33	-0.42	-0.42	-0.4	-0.42	-0.47	1	0.16	-0.31	-0.034	-0.42	-0.1	-0.11	0.33	0.27	0.15
PM10	-0.086	0.34	0.35	0.35	0.2	0.38	0.31	0.16	1	-0.28	0.14	0.4	0.016	0.012	0.073	0.054	0.8
RH	-0.32	0.16	-0.048	-0.048	-0.061	-0.11	-0.09	-0.31	-0.28	1	-0.2	0.049	-0.091	-0.074	-0.18	-0.15	-0.17
SO2	0.058	0.17	0.31	0.31	0.081	0.17	0.13	-0.034	0.14	-0.2	1	0.28	0.012	-0.028	-0.056	-0.072	0.2
THC	-0.084	0.82	0.91	0.91	0.74	0.71	0.81	-0.42	0.4	0.049	0.28	1	0.19	0.18	-0.36	-0.36	0.24
WD_HR	0.34	0.11	0.22	0.22	0.1	0.093	0.11	-0.1	0.016	-0.091	0.012	0.19	1	0.81	-0.2	-0.23	0.063
WIND_DIREC	0.32	0.11	0.21	0.21	0.11	0.092	0.11	-0.11	0.012	-0.074	-0.028	0.18	0.81	1	-0.21	-0.23	0.056
WIND_SPEED	-0.02	-0.31	-0.33	-0.33	-0.23	-0.36	-0.34	0.33	0.073	-0.18	-0.056	-0.36	-0.2	-0.21	1	0.94	-0.072
WS_HR	-0.05	-0.3	-0.33	-0.33	-0.23	-0.36	-0.34	0.27	0.054	-0.15	-0.072	-0.36	-0.23	-0.23	0.94	1	-0.087
PM2_5	-0.098	0.23	0.2	0.2	0.058	0.33	0.19	0.15	0.8	-0.17	0.2	0.24	0.063	0.056	-0.072	-0.087	1

Figure 3 : Correlation matrix

5 Implementation, Evaluation and Results Of Air Quality Prediction Models

5.1 Introduction

Once we know the features to be selected, next step is the implementing the model. In this project the models will be deployed in two platforms - Apache spark client and cluster platforms. The Apache spark will be installed in Ubuntu eco system using Virtual Box.

5.2 Implementation and Evaluation of Pre-processed Data and Feature Engineering

5.2.1 Removal and imputing of NA Values

Taking care of the NA values is an important task while building regression models. In this project the pre-processed dataset is checked for NA values in each columns and is plotted in a graph as presented in figure 4. It can be noted that there is a great descent from column *THC* to column *WS_HR* . This means that there are more number of NA values in columns UVB, *RAIN_COND*, *PH_RAIN*, *NMHC*, *CH4* and *THC* compared to the remaining columns in the dataset. The columns with high number of NA values are considered first, *UVB* column has more than 13513 NA values, so it is better to drop this column. Similarly other columns which have NA ratios greater than 0.5 are removed. The column *WS_HR* which denotes the wind speed average for every hour is an important feature and cannot be removed.

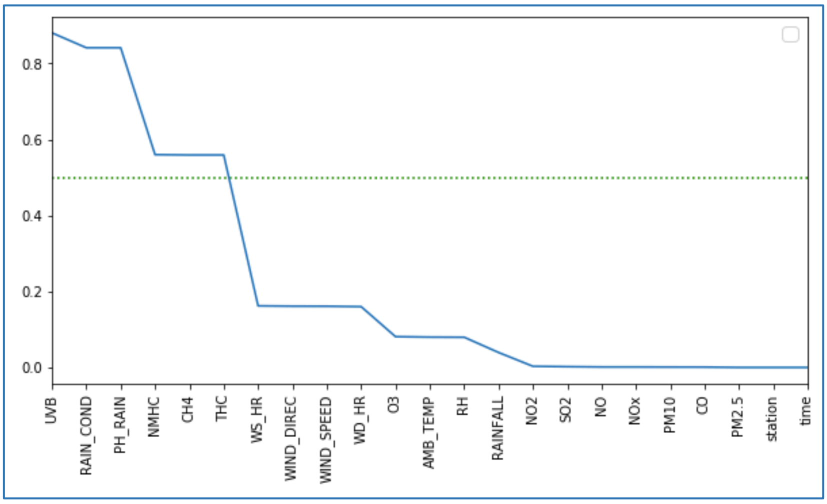


Figure 4 : NA ratio

The skewness of the column WS_HR was checked and found that it was right skewed as presented in figure 5.

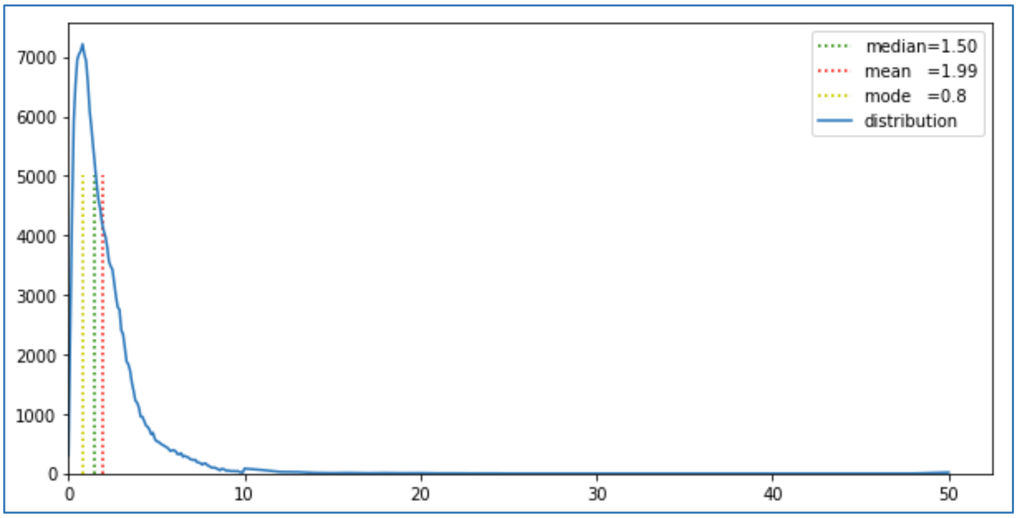


Figure 5 : Skewness of wind speed average

The NA values in this column was filled with the median value and from the boxplot presented in figure 6 it is understood that there are outliers in this column. This is not an issue because wind is a natural phenomenon and varies from time to time.

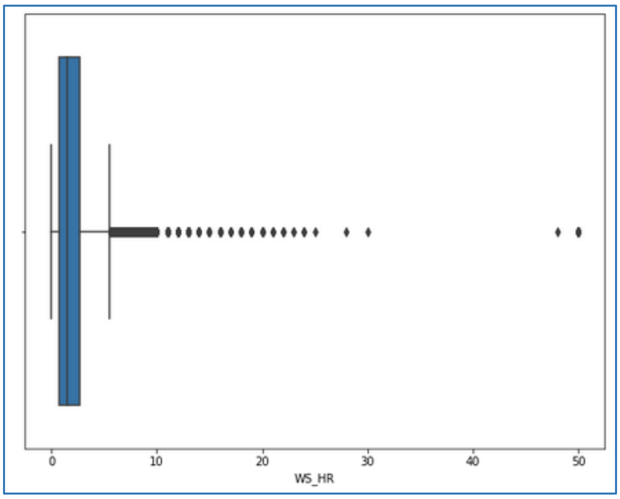


Figure 6 : Box plot of average wind speed

5.2.2 Feature Engineering

- **Date Extraction** : Day, Month, Week day, Hour were extracted from the Timestamp using the datetime time library *(pandas.to_datetime)* and pandas data frame.
- **Unit Conversion** : Since the data consist of many features which includes various pollutants, these pollutants are measured in different units and this needs to made into a general unit for the model to predict accurately.
  For example : *Taiwan['O3'] = Taiwan['O3']/1000* Here O3 value was recorded in unit PPB(Parts per billion), which was converted to PPM(Parts per million).
- **Handling Outliers** : Outliers cause overfitting when the deployed, to remove this the data was visualized based on various features using different methods. Since the data is time series data, time series plot was used to understand the drop and rise in various pollutants. Some of the visualisations made to understand the outliers are presented below in figure 7.

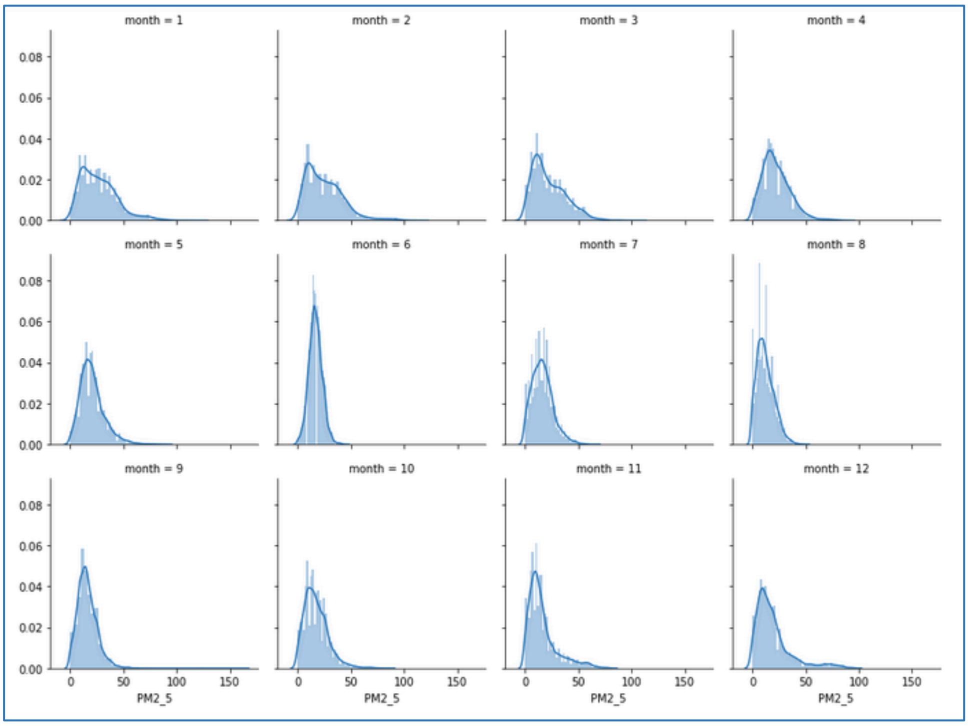


Figure 7 : Outliers plot

- **Handling Cyclical features**[1] : Features like hour and month are cyclical features which may cause confuse during the model deployment, to avoid this these features are converted to circular instead of linear using trigonometric functions, this is usually done to reduce the MSE error rate.

The columns Hour and month are cyclic features and the machine learning algorithms cannot calculate the distance between hour 0 and hour 24 precisely, The algorithm treats theses values as distant from each other. Month 1 and 12 are close to each other but the model does not depict it as such. To consolidate this issue, feature engineering as mentioned in section 4.2 is performed and the values in Hour and Month column are made close to each other in a cyclic format as presented in figure 8 using sin and cos functions.

[1] http://blog.davidkaleko.com/feature-engineering-cyclical-features.html

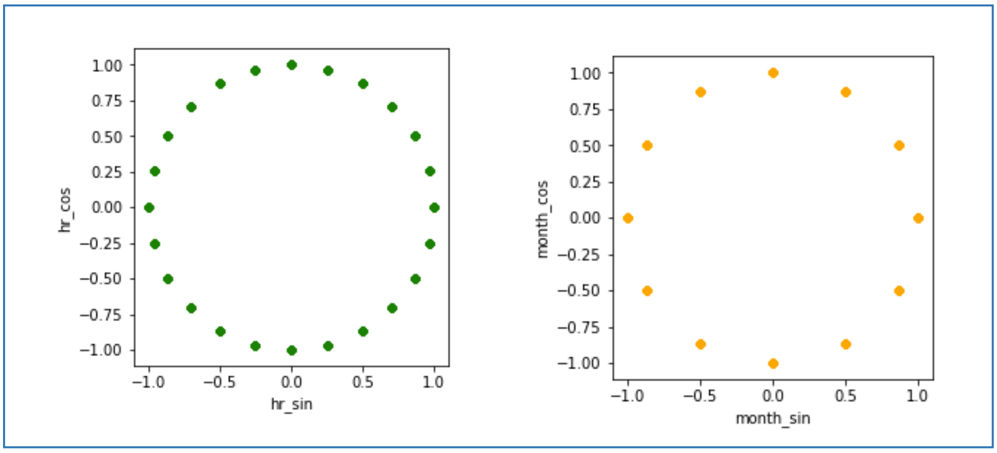


Figure 8 : Cyclic conversion of hour and month

5.3 Implementation of Linear Regression Model

Linear regression studies the linear relationship between independent variables and dependent variables . In linear regression the dependent variables are always continues and the independent variable can be continues or categorical or both. The linear regression follows the equation :

$$Y = a + b1 * \ X1 + b2 * X2 + \ldots\ldots + bn * Xn$$

Where *a* is considered as the *y- intersect* of the line and *b* is the *slope*. Every linear regression problem will have a line and that is decided by estimating the values of Y – dependent variable and values of X – independent variable. The multi variable regression allows the computation of multiple independent variable to estimate the dependent variable (Astrid Schneider, Gerhard Hommel, 2010).

i. Linear Regression Model *Client Mode)*

After feature engineering and exploratory analysis, 23 features were selected to predict AQI. PM2.5 is taken as the dependent variable to predict the AQI as supported by literature in section 2.4 . Linear regression was performed using the *LinearRegression* package imported from the *sklearn.linear_model* library . The data was split into train and test on a ratio of 7 : 3 with manually set random state. After the prediction the regression score was found to be 0.698. The measure values were plotted against the predicted values using scatterplot. This model was implemented to satisfy the sub objective(a) 1 of the objective 3 in section 1.3 .

ii. Linear Regression Model *(Cluster Mode)*

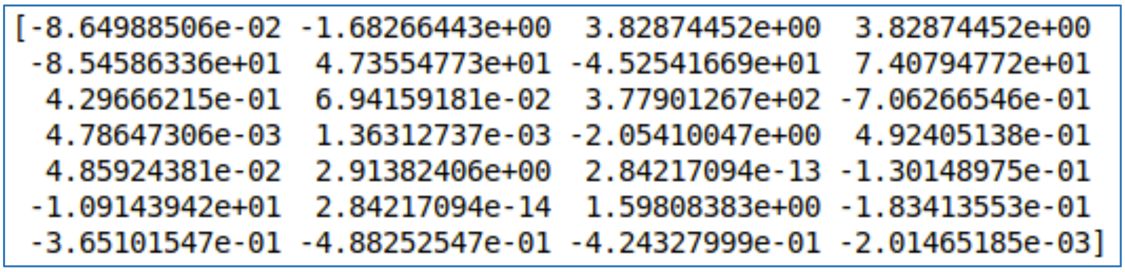
The features were converted into an array of vectors using Vectors package from *pyspark.ml.linalg* and using *VectorAssembler* from *pyspark.ml.feature* and the linear regression model was developed using *LinearRegression* package from *pyspark.ml.regression* library. By the successful implementation of this model the sub objective (i) of objective 4 from section 1.3 is accomplished.

5.3.1 Evaluation and Result of Linear Regression Model

5.3.1.1 Linear regression using Apache Spark Client Mode

The actual level of PM2.5 versus the predicted level of PM2.5 is plotted in a graph as presented in figure 9. The *regression score* for the model shows the value around *0.6959*, with *Root Mean Square value of 7.50* , *Mean Absolute Error 5.56* and *R- squared value 0.70*. From table 3 it can be analyzed that the coefficients are showing negative values which means as the independent values are inversely proportional to the dependent variable.

Table 3: Coefficient matrix for linear regression



```
[-8.64988506e-02 -1.68266443e+00  3.82874452e+00  3.82874452e+00
 -8.54586336e+01  4.73554773e+01 -4.52541669e+01  7.40794772e+01
  4.29666215e-01  6.94159181e-02  3.77901267e+02 -7.06266546e-01
  4.78647306e-03  1.36312737e-03 -2.05410047e+00  4.92405138e-01
  4.85924381e-02  2.91382406e+00  2.84217094e-13 -1.30148975e-01
 -1.09143942e+01  2.84217094e-14  1.59808383e+00 -1.83413553e-01
 -3.65101547e-01 -4.88252547e-01 -4.24327999e-01 -2.01465185e-03]
```

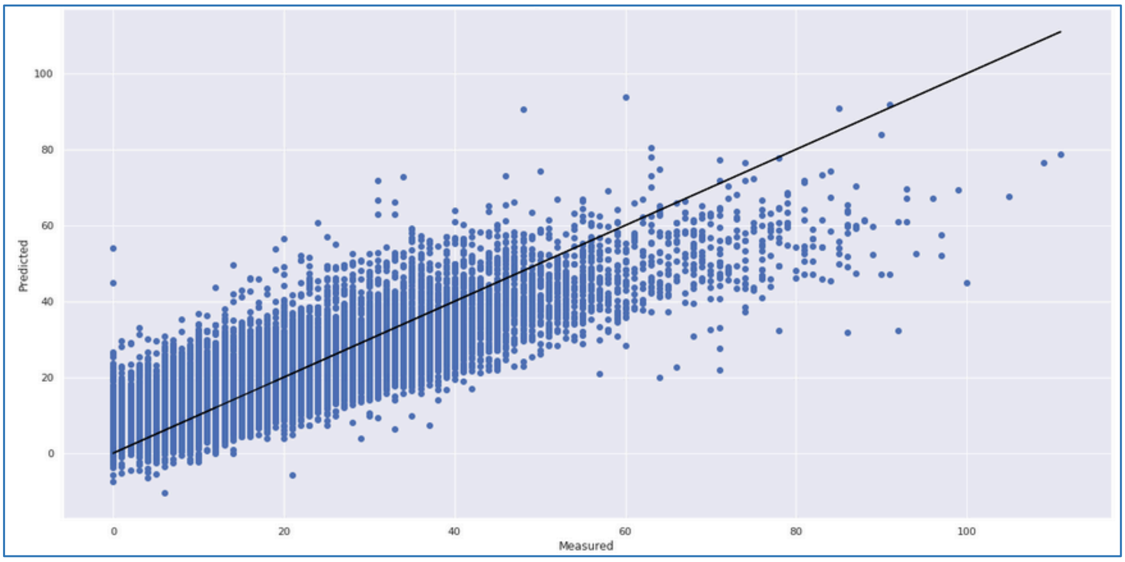


Figure 9 : Linear regression actual versus predicted plot

5.3.1.2 Linear regression using Apache Spark Client Mode

Linear regression when implemented in Apache Spark Client mode yielded the same result in respective of the Root Mean Square , Mean Absolute Error and R-Squared value. The performance speed of the model was comparatively slow in the client mode.

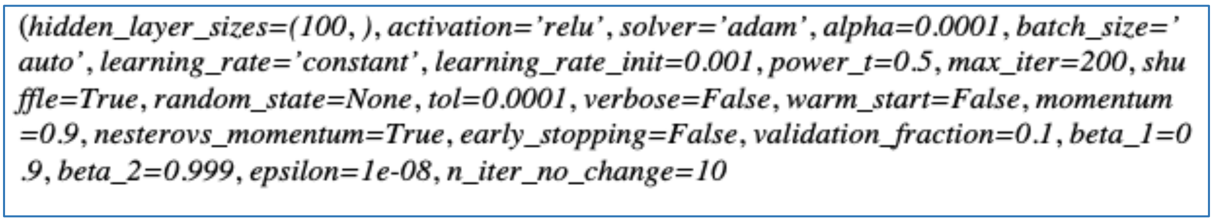
5.4 Implementation of Neural Network regression

Neural network regression is a type of deep regression technique. This neural network regression is performed using linear regression layer and is a type of vanilla regression. This type of regression is used to make the regression model more robust (Mesejo, Alameda-pineda and Horaud, no date). Neural network regression was performed using *MLPRegressor* package from *sklearn.neural_network* library. The advantage of neural network regression is the flexibility to fit into any model and choose the type of regression suitable for the situation. The regression score (0.762) in this model was somewhat similar to the linear regression model since Neural network regression has a layer of linear regression. This model is implemented in connection with the sub objective(a) 2 of objective 3 in section 1.3.

5.4.1 Evaluation and Results of Neural Network Regression Model

The Neural Network regression performed using Multi-Layer Perceptron was able to deliver a *regression score of 0.769* with a *root mean square 6.62* and *R -squared value 0.76*.

The time taken to build the Neural Network Regression was high compared to the time taken by Linear regression. The predicted values are plotted against the actual values as presented in figure 11 . The above mentioned values were produced when the MLP regressor was run in default hyper-parameters as presented in figure 10.



```
(hidden_layer_sizes=(100, ), activation='relu', solver='adam', alpha=0.0001, batch_size='
auto', learning_rate='constant', learning_rate_init=0.001, power_t=0.5, max_iter=200, shu
ffle=True, random_state=None, tol=0.0001, verbose=False, warm_start=False, momentum
=0.9, nesterovs_momentum=True, early_stopping=False, validation_fraction=0.1, beta_1=0
.9, beta_2=0.999, epsilon=1e-08, n_iter_no_change=10
```

Figure 10 : Default hyper-parameters of multi-layer perceptron

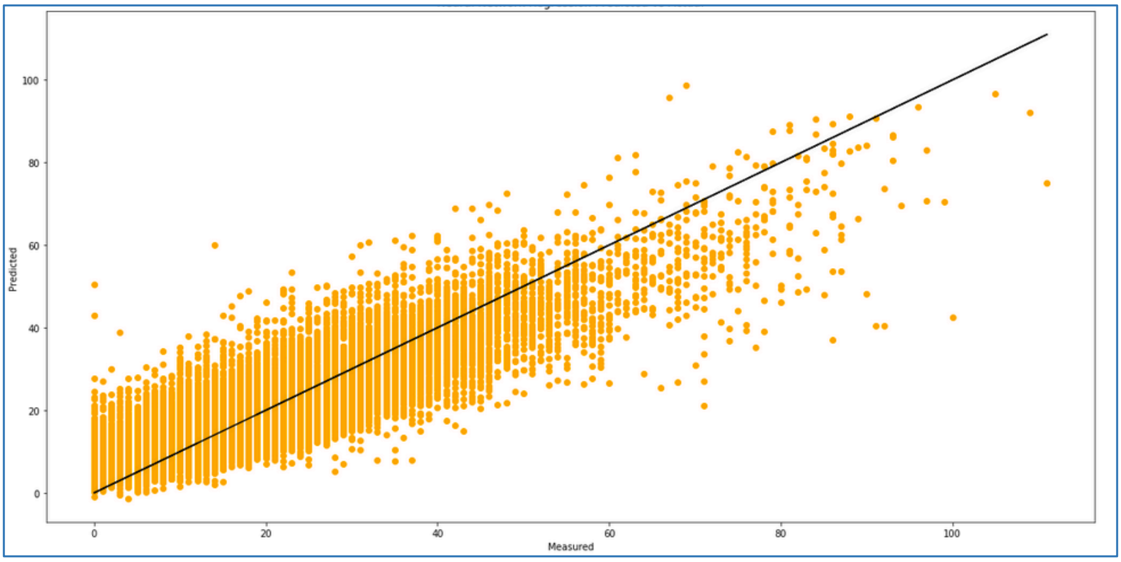


Figure 11 : Actual versus predicted plot for neural network regression

5.5 Implementation of Elastic Net Regression

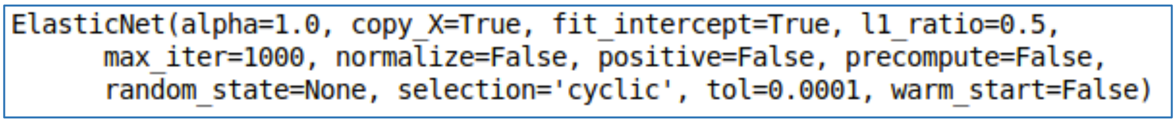
The Elastic Net is similar to Lasso Regression but works by combining both L1 and L2 regularization. Elastic Net basically checks for correlation and groups features which are correlated together and checks whether any of the variable in the group is a strong predictor, if there is any strong predictor, the entire group will be considered to build the model. Elastic regression was performed using *sklearn.linear_model* library using the *ElasticNet* package. Elastic Net can be represented mathematically as presented in Figure 12. This model accomplishes the sub objective(a) 3 of the objective 3 from section 1.3.

$$\min\left(\left|\left|\mathrm{Y}-\mathrm{X}\theta\right|\right|_2^2 + \lambda_1||\theta||_1 + \lambda_2||\theta||_2^2\right)$$

Figure 12 : Elasticnet regression equation

5.5.1 Evaluation and Result of ElasticNet Regression

Another regularized regression method ElasticNet showed a better performance compared to the normal Linear regression and Lasso regression, The model was set with default parameters as presented in figure 13 and the actual values were plotted against the predicted values as



```
ElasticNet(alpha=1.0, copy_X=True, fit_intercept=True, l1_ratio=0.5,
      max_iter=1000, normalize=False, positive=False, precompute=False,
      random_state=None, selection='cyclic', tol=0.0001, warm_start=False)
```

Figure 13 : Parameters for elasticnet regression

presented in figure 14. The model yielded a good fit with *root mean square value of 7.79*, *mean absolute error of 5.79* and *R squared value of 0.67*.

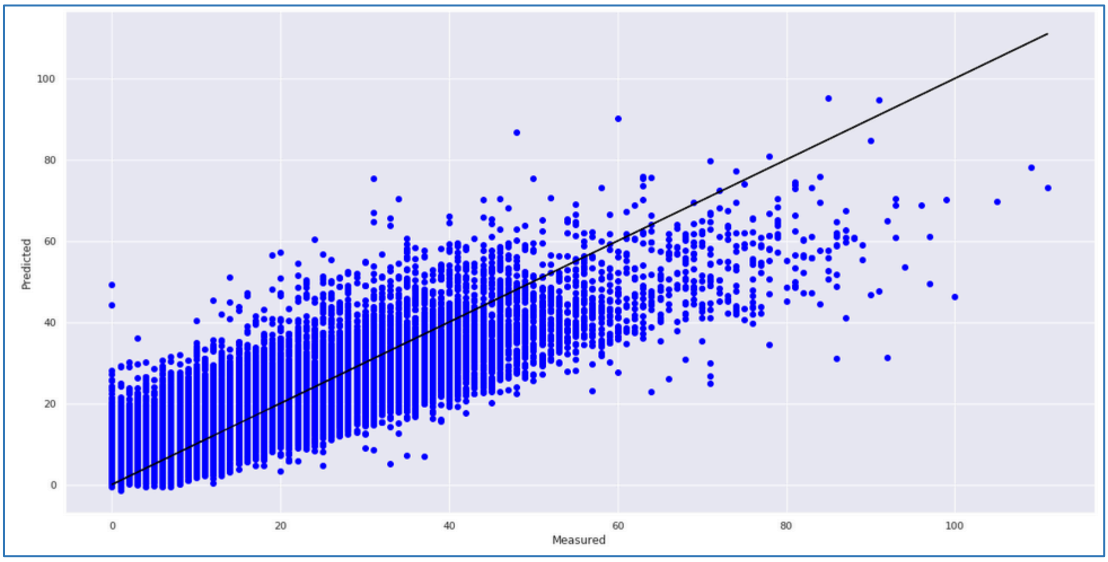


Figure 14 : Predicted values versus actual values for elasticnet regression

5.6 Implementation of Random Forest

i. Random Forest Using Apache spark client mode

Random forest is a type of ensemble model which is widely used in the machine learning models. Random forest can be considered as a classification as well as a regression model. There is a pre-built dimensionality reduction feature in Random forest which selects the strong predictor variable from a large number of input variables. In this project random forest was implemented using *RandomForestRegressor* package from *sklearn.ensmbler* library. This model satisfies the sub objective(b) 4 of the objective 3 in section 1.3.

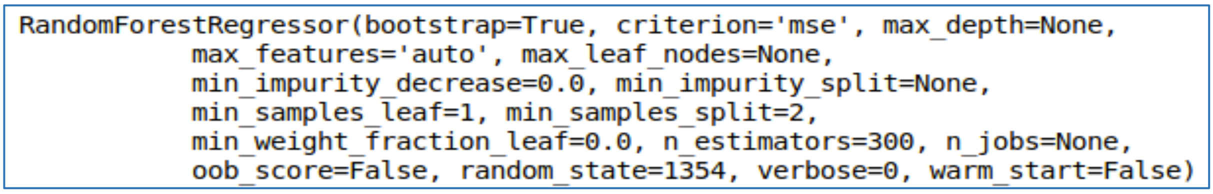
ii. Random Forest Using Apache spark cluster mode

Random forest was implemented in Apache spark cluster environment using spark machine learning libraries such as *RandomForestRegressor* from *pyspark.ml.regression* and for evaluation purpose *RegressionEvaluator* from *pyspark.ml.evaluation* libraries were used. This model satisfies the sub objective(ii) of the objective 4 in section 1.3.

5.6.1 Evaluation and Result of Random Forest Regression

5.6.1.1 Random Forest Using Apache Spark Client Mode

The Random Forest Regression model performed better than the previous models and had a *regression score of 0.792*. The hyper parameters configuration is presented in Figure 15. The random state was changed multiple times until the prediction rate was found to be the maximum.



```
RandomForestRegressor(bootstrap=True, criterion='mse', max_depth=None,
           max_features='auto', max_leaf_nodes=None,
           min_impurity_decrease=0.0, min_impurity_split=None,
           min_samples_leaf=1, min_samples_split=2,
           min_weight_fraction_leaf=0.0, n_estimators=300, n_jobs=None,
           oob_score=False, random_state=1354, verbose=0, warm_start=False)
```

Figure 15 : Parameters of random forest

The Actual values were plotted against the predicted values as presented in figure 16 where the noise is reduced compared to the scatterplots of the other models. The model yielded a *Root Mean Square 6.18* , *Mean Absolute Error 4.54* and *R squared value of 0.79*.

Feature Importance : The important features are for developing the Random Forest Regression model is presented in figure 17. The feature selection shows the how valuable each variable was in building the AQI prediction model.

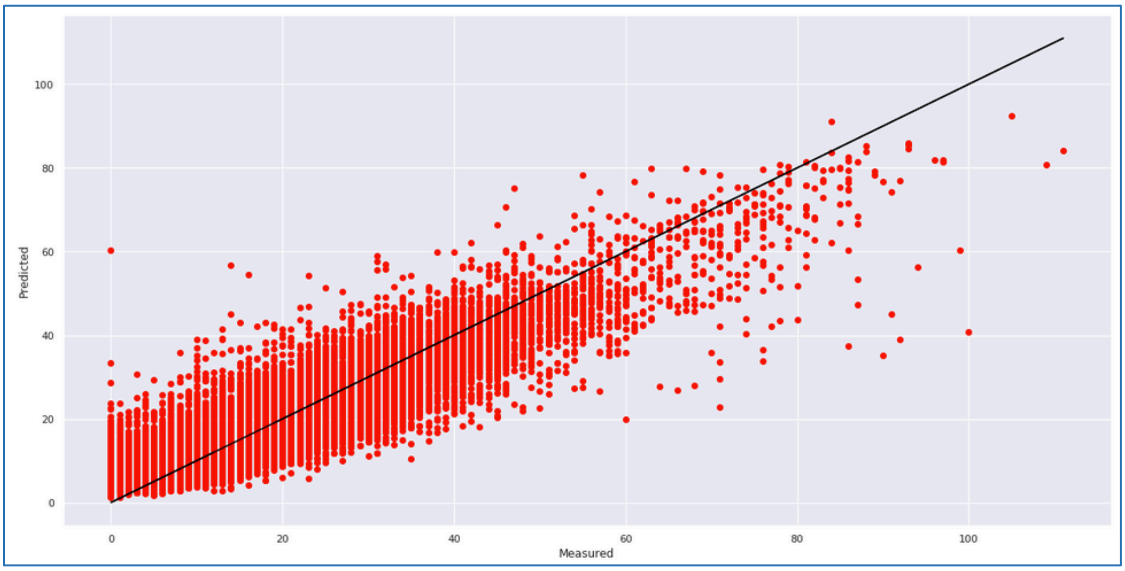


Figure 16 : Actual values versus predicted values for random forest regression

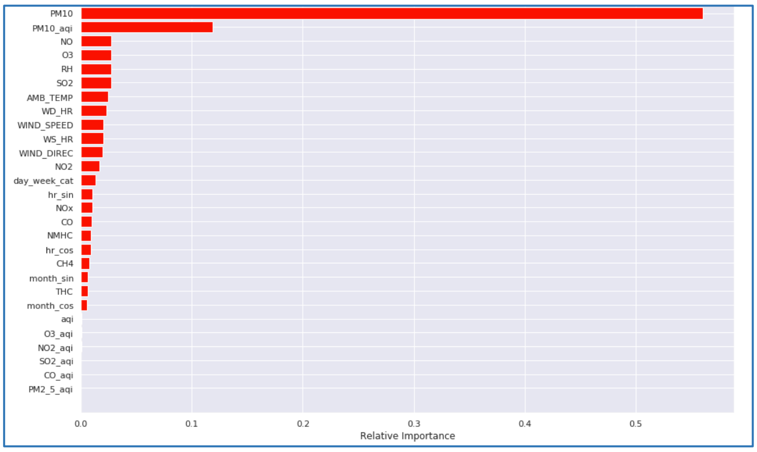


Figure 17 : Feature importance in random forest regression

5.6.1.2 Random Forest Using Apache Spark Cluster Mode

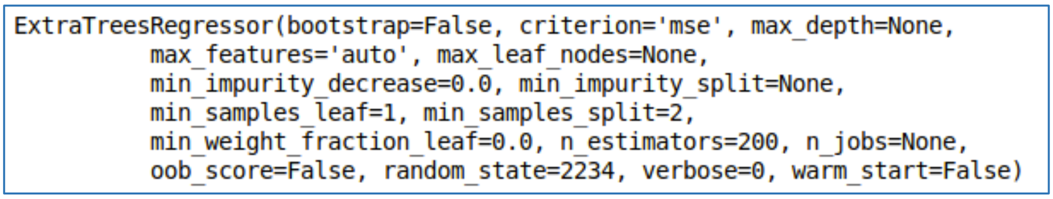
The results when Random forest model was implemented in Apache spark cluster environment was different compared to the model implemented in client mode. The performance of the model deteriorated with a Root Mean Square value of 7.641, Mean Absolute error of 5.740 and R-Squared value of 0.682. It was also found that the model was structured with 20 trees.

5.7 Implementation of ExtraTrees Regressor

ExtraTrees is another types of Random Forest, but in the case of ExtraTrees the trees are built using random splits of the subset of the features and not depending on the best splits. it does not bootstrap observations (meaning it samples without replacement), and nodes are split on random splits, not best splits. This model satisfies the sub objective(a) 5 of the objective 3 in section 1.3.

5.7.1 Evaluation and Result of ExtraTrees Regression Model

The ExtraTrees model relatively performed better than the Random Forest model with a regression score of 0.795 , Root Mean Square value of 6.14 , Mean Absolute score of 4.55 and R squared value 0.80. Various random sampling were used to test the model and most of the times the same result was produced. The Parameters used for the model are presented in figure 18. The Actual versus predicted plot is presented in figure 19 which further brings the scattered plots closer to the best fit line compared to the previous models.



```
ExtraTreesRegressor(bootstrap=False, criterion='mse', max_depth=None,
          max_features='auto', max_leaf_nodes=None,
          min_impurity_decrease=0.0, min_impurity_split=None,
          min_samples_leaf=1, min_samples_split=2,
          min_weight_fraction_leaf=0.0, n_estimators=200, n_jobs=None,
          oob_score=False, random_state=2234, verbose=0, warm_start=False)
```

Figure 18 : Hyper parameters for extratrees regression model

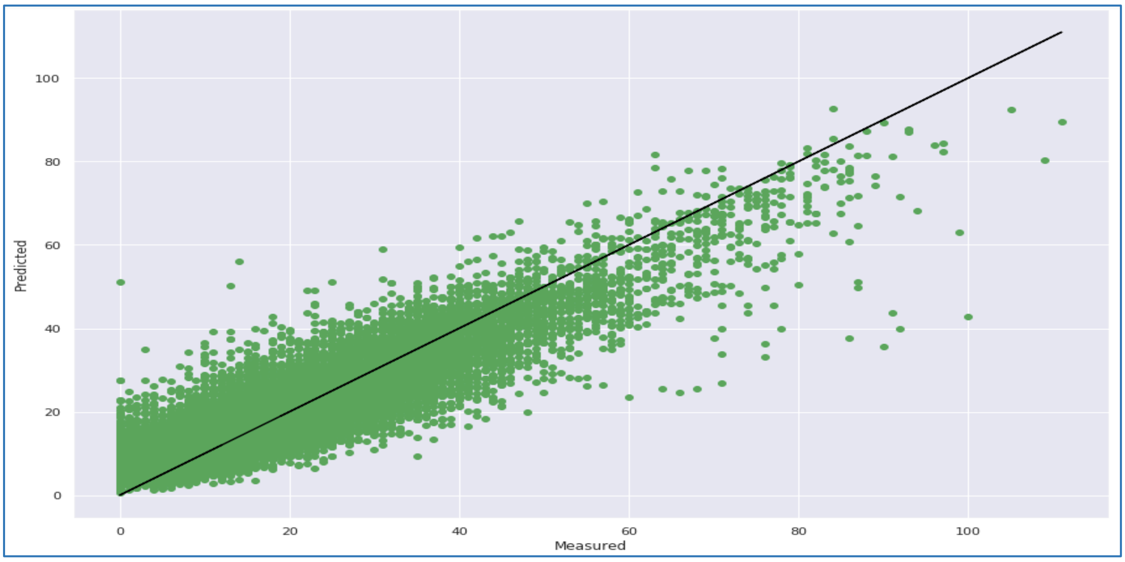


Figure 19 : Actual versus predicted plot for extratrees regression model

The feature importance below graph for ExtraTrees regression model is presented in Figure 20. The graphs depicts that the most valuable feature in this model is the PM10_aqi feature, which was manually created from PM10 values. Unlike the Random forest regression in this model more than two variables contributed more than average to the prediction.

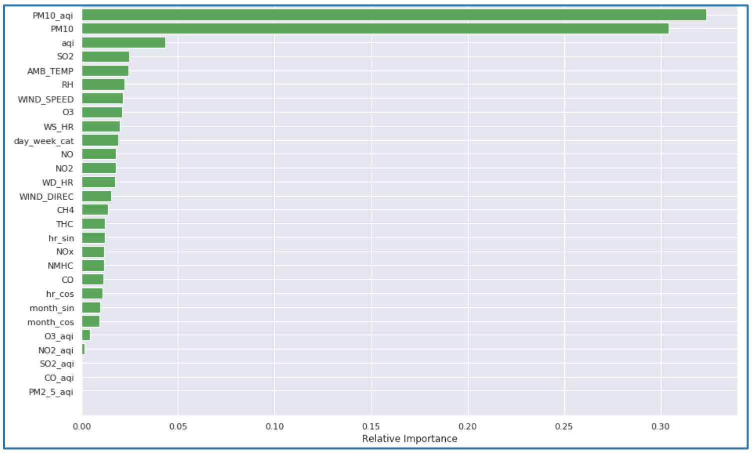


Figure 20 : Feature importance for extratrees regression

5.8 Implementation of Boosting Models

5.8.1 Implementation of AdaBoost

AdaBoost is a type of boosting method which enables the model to get hold of the non-linear relationships. AdaBoost considers the noise in the variables very seriously and it was proved that AdaBoost performs better when the data supplied to the model was large. AdaBoost was implemented using *sklearn.ensemble* library and *AdaBoostRegressor* package. The boosting formula for AdaBoost is presented in Figure 21. This model satisfies the sub objective(b) 6 of the objective 3 in section 1.3.

$$H(x) = \operatorname{sign}\left(\sum_{t=1}^{T} \alpha_t h_t(x)\right).$$

Figure 21 : AdaBoost equation

5.8.1.1 Evaluation and Result of AdaBoost Model

AdaBoost model performed very well compared to the normal decision tree, Boosting was applied to the normal decision tree and the prediction accuracy increased by 20 %. The Normal decision tree had a *regression score of 0.546* whereas the AdaBoost model gave a *regression score 0.801*. When the scatterplot of both the models are compared a great difference can be observed the scatterplots for both the models are given below in Figure 22 and 23 respectively. The AdaBoost model produced a *Root mean square error of 6.02 , Mean Absolute error of 4.37 and R squared value of 0.80.*

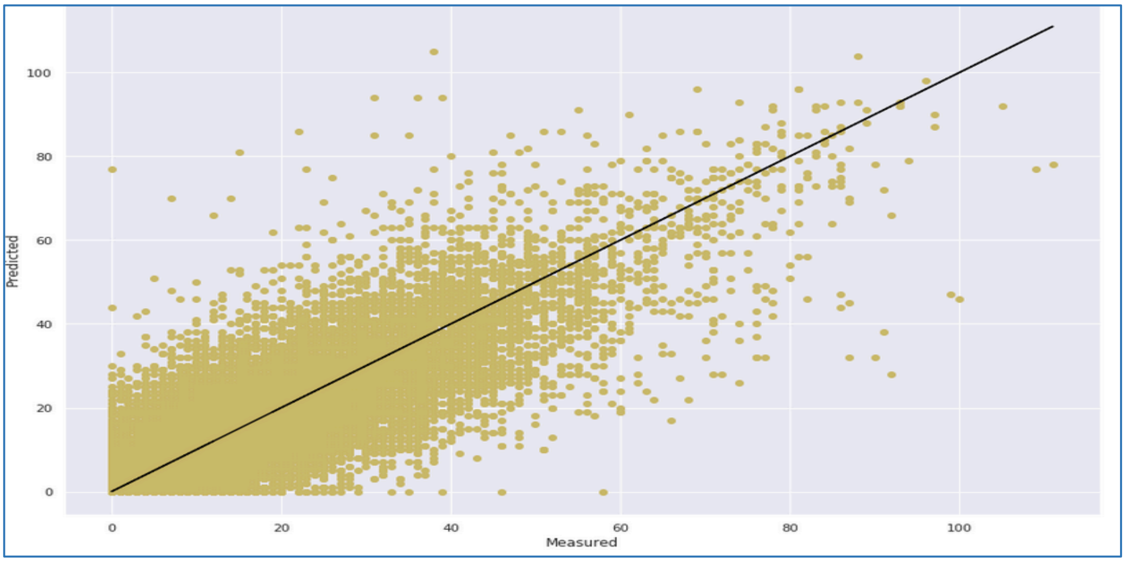


Figure 22 : Actual versus predicted plot for normal decision tree model

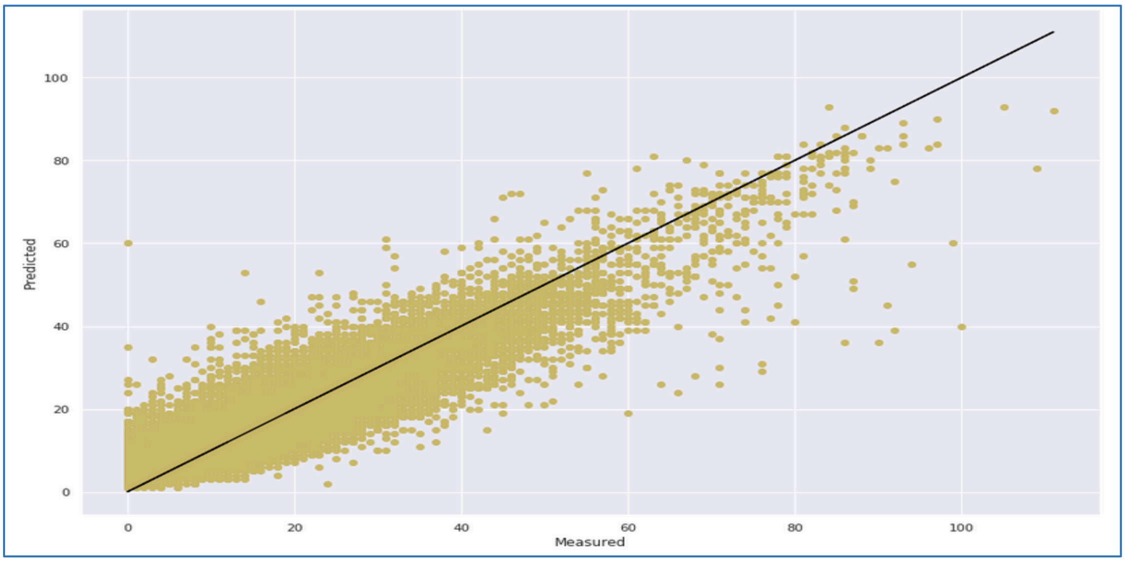


Figure 23 : Actual versus predicted values plot for AdaBoost model

5.8.2 Implementation of Gradient Boosted Trees (Cluster Mode)

Gradient boosting trees is another type of boosting algorithm slightly different than the AdaBoost model discussed above. GBT works in such a way that only one tree is built at a time and the following trees help correct the errors performed while building the previous tree. The gradient boosted trees are very sensitive to noisy data and since prior caution should be taken in removing the outliers before implementing the model. The GBT model was implemented using *GBTRegressor* from *pyspark.ml.regression* package. This model satisfies the sub objective(iii) of the objective 4 in section 1.3.

5.8.2.1 Evaluation and Result of Gradient Boosted Trees

The GBT algorithm was found to be better performing in the case of all the implemented in Apache spark cluster environment. The model yielded a *root mean squared error of 7.11* with a *R-squared value of 0.73 and mean absolute error of 5.310*.

….

The objectives 3 and 4 mentioned in section 1.3 has been successfully implemented in this section. The implemented models with their respective evaluating parameters are briefly explained above. As mentioned in the section 1.3, only three models were implemented in Apache spark cluster mode due to the limited availability of machine learning libraries in Apache spark.

6 Discussion and Comparison

6.1.1 Comparison of Implemented Models

The objective to develop various models to predict the Air Quality Index in Taiwan is accomplished, The models implemented showed good performance overall, The applied models were compared with respective to the *Root Mean Square, Mean Absolute Error* and *R-Squared* Value as shown in Table 1. Since the *pyspark.ml* libraries supports only limited machine learning libraries, not all the models implemented in client mode were able to be developed in cluster mode.

From Table 4 it can be clearly seen that the AdaBoost model outperformed the rest of the models, acquiring the highest accuracy of 80%, Even Extra Trees regressor yielded a precision of 80%, the *mean absolute error* rate was slightly high for this model. Elastic Net regression performed the least in client mode when compared to the other models even though regularization was applied. Overall the models implemented with the boosted tree structure performed better compared to the normal regression models because regression models takes the statistical assumptions into consideration. When Linear Regression model and Random forest model were implemented in cluster mode the performance was not up to the mark compared to the results of the same models when implemented in client mode. This is due to the lack of flexibility of the features in cluster mode. All the features are taken as a group of vectors instead of considering them separately. Even with this limitation Gradient boosted tress performed really well in cluster mode compared to the Linear regression and Random forest models .

Table 4 : Comparison of Implemented Models

Model	**Mode**	**RMSE**	**MAE**	**R Squared**
Linear Regression	Client	7.50	5.56	0.70
	Cluster	7.41	5.49	0.70
Neural Network Regression	Client	6.87	5.11	0.74
Elastic Net Regression	Client	7.79	5.79	0.67
Random Forest	Client	6.18	4.54	0.79
	Cluster	7.64	5.7	5.74
Extra Trees Regression	Client	6.14	4.55	0.80
AdaBoost	Client	6.02	4.37	0.80
GBT	Cluster	7.11	5.32	0.72

6.1.2 Performance Comparison of Apache Spark Environment

To satisfy the final objective (Objective 6) of this project, the performance speed of various process are compared when run in Apache spark client mode and cluster mode. From figure

26, it can is understood that the cluster mode performance was much better when compared to the client mode except in the data import process and Linear regression . In client mode the data was imported directly from the local whereas in cluster mode the data was imported from HDFS, which increased the latency time. The models which require more computational space and cache memory like Random forest and Boosted Model were able to be modelled in very less time in cluster mode compared to the client mode. This is made possible with architecture of Apache Spark (Section 2.4), even though the model was not a good fit compared to the models developed in client mode.

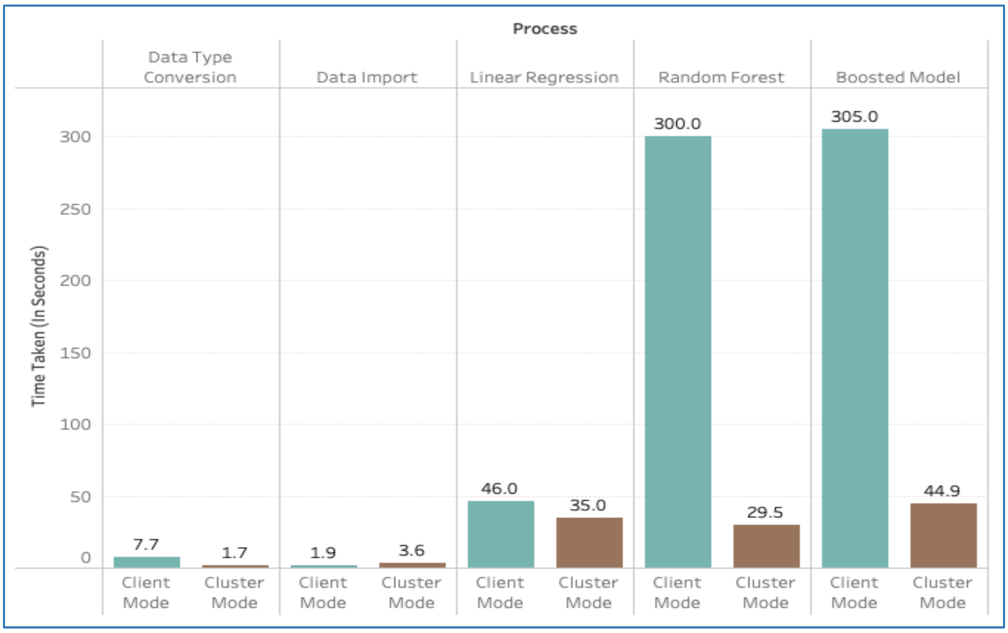


Figure 24 : Performance graph

7 Conclusion and Future Work

This project discusses about the various machine learning models which can be used to predict the Air Quality Index in Taiwan, which would help the people as well as the government of Taiwan to raise awareness about the air quality depending on various meteorological factors. This research also proved that Boosted tree techniques can outperform the existing air quality Prediction models. In addition to this, the research also contributes to the big data field with the successful implementation of machine learning models in Apache Spark and found that the machine learning algorithms when implemented in Apache Spark cluster environment showed better performance speed during the implementation of models which require more cache memory which is made possible with the in-memory feature in spark. The novelty of this research is the implementation of various regularization techniques in Linear regression model, even though the Boosted Decision tree models yielded a better result, these regularization models produced better results than the existing Regression Models.

In the future, the efficiency of the Boosted Tree algorithms implemented in the Apache spark cluster environment can be increased by providing a more complex and bigger dataset. Various challenges were faced during the research, where the historical AQI data consisted of columns with more NA values which were forced to remove for better building of the model, A better dataset with less missing values would increase the accuracy of the prediction models. The limited machine learning models available in the spark machine learning package restricted the access to implement regularized regression techniques in Apache spark cluster mode.

Acknowledgement

I would like to express my sincere gratitude to my supervisor Dr. Catherine Mulwa who guided me throughout the project , which helped me in completing this research successfully. I extend my gratitude to my parents and all the teaching staffs who taught different modules in my Masters programme at National College Of Ireland . Thank you very much.

References

Anderson, H. R. (2009) 'Air pollution and mortality: A history', *Atmospheric Environment*. doi: 10.1016/j.atmosenv.2008.09.026.

Astrid Schneider, Gerhard Hommel, and M. B. (2010) 'Linear Regression Analysis', *Evaluation of Scientific Publications*, 107(44), pp. 776–782. doi: 10.3238/arztebl.2010.0776.

Bernstein, J. A. *et al*. (2004) 'Health effects of air pollution', *Journal of Allergy and Clinical Immunology*. doi: 10.1016/j.jaci.2004.08.030.

Bladen, W. A. and Karan, P. P. (1976) 'Perception of Air Pollution in a Developing Country', *Journal of the Air Pollution Control Association*, 26(2), pp. 139–141. doi: 10.1080/00022470.1976.10470235.

Central pollution control board (2014) 'Central Pollution Control Board', *National air quality index*.

Chow, J. C., Watson, J. G. and Chaung, C. Y. (2012) 'Air Pollution in the Republic of China (Taiwan) APCA NOTE-BOOK', 2470(1983). doi: 10.1080/00022470.1983.10465640.

Cohen, A. J. *et al*. (2005) 'The global burden of disease due to outdoor air pollution', *Journal of Toxicology and Environmental Health - Part A*, 68(13–14), pp. 1301–1307. doi: 10.1080/15287390590936166.

Corani, G. (2005) 'Air quality prediction in Milan: Feed-forward neural networks, pruned neural networks and lazy learning', *Ecological Modelling*, 185(2–4), pp. 513–529. doi: 10.1016/j.ecolmodel.2005.01.008.

Davidson, C. I., Phalen, R. F. and Solomon, P. A. (2005) 'Airborne particulate matter and human health: A review', *Aerosol Science and Technology*, 39(8), pp. 737–749. doi: 10.1080/02786820500191348.

Haq, G. and Schwela, D. (2008) 'Urban Air Pollution in Asia Foundation Course on Air Quality Management in Asia', (January).

Kaimian, H. *et al*. (2019) 'Evaluation of different machine learning approaches to forecasting PM2.5 mass concentrations', *Aerosol and Air Quality Research*, 19(6), pp. 1400–1410. doi: 10.4209/aaqr.2018.12.0450.

Kumar, N. (2016) 'Air Quality Index – A Comparative Study for Assessing the Status of Air Quality Air Quality Index – A Comparative Study for Assessing the Status of Air Quality', (May). doi: 10.5958/2321-581X.2015.00041.0.

Lamba, A. *et al*. (2014) 'USES OF CLUSTER COMPUTING TECHNIQUES TO PERFORM BIG', 1(7), pp. 5804–5808.

Lin, Chieh-yen, Lee, C. and Lin, Chih-jen (no date) 'Large-scale Logistic Regression and Linear Support Vector Machines Using Spark'.

Liu, C. (2002) 'Effect of PM2 . 5 on AQI in Taiwan', 17(2), pp. 29–37.

Liu, H. *et al*. (2019) 'Air quality index and air pollutant concentration prediction based on machine learning algorithms', *Applied Sciences (Switzerland)*, 9(19). doi: 10.3390/app9194069.

Meng, X. *et al*. (2016) 'MLlib : Machine Learning in Apache Spark', 17, pp. 1–7.

Mesejo, P., Alameda-pineda, X. and Horaud, R. (no date) 'A Comprehensive Analysis of Deep Regression ´ ``', pp. 1–17.

Samadianfard, S. *et al*. (2013) 'Comparative analysis of ozone level prediction models using gene expression programming and multiple linear regression', *Geofizika*, 30(1), pp. 43–74.

Santos, A. A. M. F. (2018) 'KDD , SEMMA AND CRISP-DM : A PARALLEL OVERVIEW Ana Azevedo and M . F . Santos'.

Shoro, A. G. and Soomro, T. R. (2015) 'Big Data Analysis : Ap Spark Perspective BigDataAnalysisApSparkPerspective', (January).

Sierra-Vargas, M. P. and Teran, L. M. (2012) 'Air pollution: Impact and prevention', *Respirology*, 17(7), pp. 1031–1038. doi: 10.1111/j.1440-1843.2012.02213.x.