UNIVERSITI MALAYA

FACULTY OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY

WIE3007 DATA MINING AND WAREHOUSING

SESSION 2022/2023 SEMESTER 1

Group Assignment (25%)

Group 10

Name	Matric Number
Emily Choo Yu Xin	17205147/1
Phong Jia Wei	17207406/1
Tan Jia Chyi	17205011/2
Soong Pei Chze	17204232/1
Gan Jia Soon	17206343/1

Lecturer: Dr. Riyaz Ahamed Ariyaluran Habeeb Mohamed

Table of Contents

1. Introduction

Climate change is listed among the biggest health risks by the World Health Organization (WHO). Similarly, air pollution is also listed as the biggest environmental health threat. This is due to the fact that air pollution, no matter indoor or outdoor, has caused an estimated death of 7 million per year. (Campbell-Lendrum & Prüss-Ustün, 2019).

According to the World Health Organization (2019), air pollution is the air being contaminated by any gases, liquid and solid particles that may alter the composition whether indoors or outdoors. Example of contamination pollutants as quoted from the World Health Organization (2019) are as follow, "particulate matter (PM2.5 and PM10), carbon monoxide (CO), ozone (O3), black carbon (BC), sulphur dioxide and nitrogen oxides (NOx)".

Currently, our transportation, electricity generation, industry, and food production systems are all powered by various types of energy that mainly contribute to air pollution. (Campbell-Lendrum & Prüss-Ustün, 2019). Moreover, Kinney (2018) mentioned that the combustion of fossil fuels (which emits carbon dioxide, black carbon, and ozone precursors) and agricultural production are the primary causes of human-caused changes in the global climate system (emitting methane).

Kinney (2018) suggested that the issues regarding climate change and air pollution can be connected with their factors and solutions. Therefore, it is important to research this area as air pollution has contributed to climate change and causes many other consequences such as health issues. As emphasised by Campbell-Lendrum & Prüss-Ustün (2019), the persistence of air pollution may lead to the outspread of noncommunicable illnesses such as lung and heart diseases. Nature may be disrupted by the thinning of ozone layers as a side effect of air pollution.

With the climate change problems arising, humans should be alerted to nature's destruction. This project aims **to identify the patterns and insights from the air pollution data**. Through this project, we can analyse the data to have an understanding of the major pollutants and gas emission and help in identifying which of them contribute significantly to air pollution. This may assist in decision-making for the efforts of solving air pollution. Another objective for our project is **to predict whether a state is classified as the state with high**

pollution based on the pollutants and gas emissions of the states such as PM2.5,PM10,NO2,O3,CO,SO2, which are all the major actors in urban air pollution. Hence, we have trained several models and compared the model's accuracy among them in order to get the most accurate result for the prediction.

For this project, the dataset chosen is DEAP: Deciphering Environmental Air Pollution from Kaggle. It is a large dataset using Spatio-temporal containing details about urban air pollution collected for 2 years in the United States. The table below shows the description for each column in the dataset.

Table 1.1: Dataset Description

Column Title	**Description**
Date	Date of the sample collected
City	City of the sample location
County	County of the sample location
State	State of the sample location
Population Staying at Home	People staying at home were sampled for domestic emission
Population Not Staying at Home	People not staying at home
mil_miles	Vehicle travel distance sampled
past_week_avg_miles	Average of miles that the vehicle travelled in the past week
Minimum, Maximum, Median, Variance and Count (for each criterion)	Minimum, maximum, median, variance and count of each pollutant and meteorological feature:

	Pollutants: PM2.5, PM10, NO2, O3, CO, SO2 **Meteorological Features**: Temperature, Pressure, Humidity, Dew, Wind Speed, Wind Gust

2. Data Pre-processing (Practical)

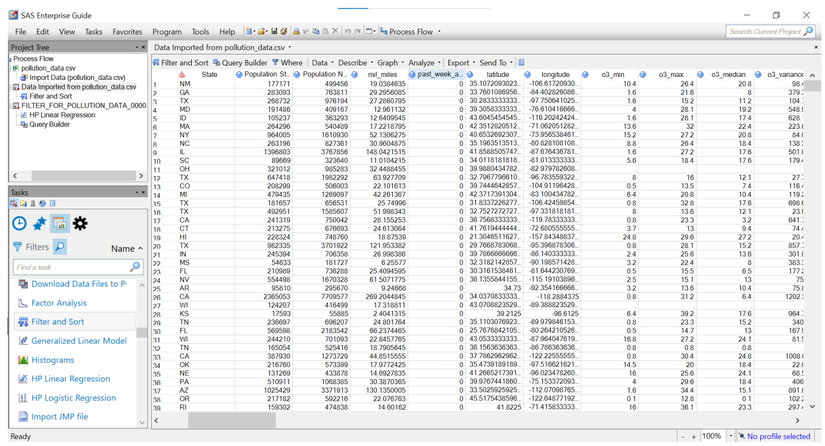
After reviewing the chosen dataset ,the project team realised that there are some missing values, outliers and null values in the dataset. Hence, data pre-processing such as data cleaning has to be applied in this dataset. Data cleaning is the process of removing incorrect, duplicate, incomplete data or missing values (Kara Sherrer, June 30 2022). Data cleaning improves the quality of data as well as any business decisions that draw from the data for further analysis. There are a lot of data cleaning tools available in the market nowadays such as Open Refine, Trifacta Wrangler, WinPure and others (Alex McFarland, April 27 2022). However, in this project , the data preprocessing models that we chose are SAS Enterprise Guide and SAS Enterprise Miner.

2.1 SAS Enterprise Guide

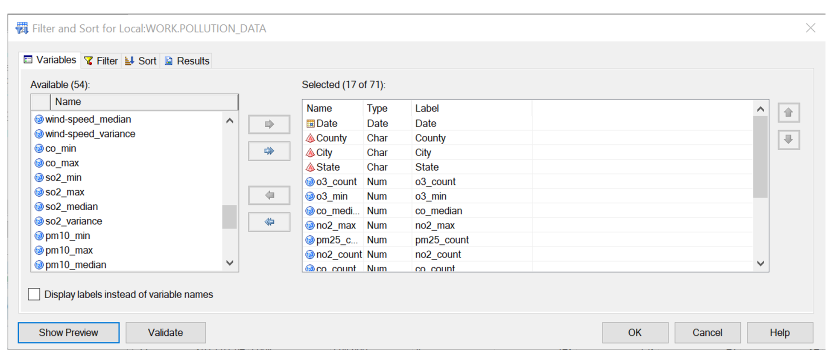
SAS Enterprise Guide is a user interface to Statistical Analysis System (SAS). It can be used for basic SAS programming. Furthermore, the tasks in the system can be used to generate SAS programs for the user to manipulate data, describe data, visualise data and perform statistical analysis on it. Normally, Enterprise Guide acts as the 'general store' of SAS as it offers something for everyone and provides general and simple reporting or even analysis. In Enterprise Guide, there are existing features such as **Filter and Sort** or **Query Builder** to perform the simple data cleaning tasks. The dataset in the format of Microsoft Excel can be easily imported into SAS EG for further used. It is good to use for small analysing purposes. The unstructured and missing data can be cleaned by using the Filter and Sort features located in the SAS EG client interface. Filter is used to remove outliers and retain the useful values, at the same time able to exclude the missing values. Moreover, the values can be listed in an ascending or descending order in the result table. Data reduction can be applied in Query Builder. After cleaning the data, it's time to reduce the amount of data and only select the useful one according to the business problem. Query Builder can once again specify the data based on the analysis purpose.

2.1.1 Data Pre-Processing with Filter and Sort in SAS Enterprise Guide

<table>
<tr><th>No.</th><th>Steps</th><th>Screenshots and Explanation</th></tr>
<tr><td>1</td><td>Import the dataset into SAS Enterprise Guide >Select Filter and Sort from the Task (located at the bottom left)</td><td>The below shows the original dataset with some missing and null values.

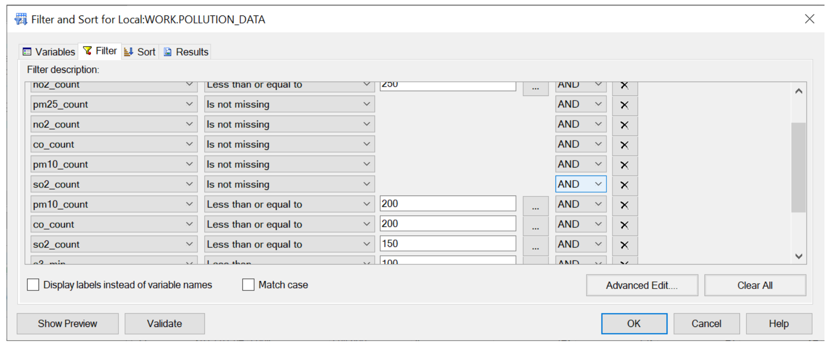


</td></tr>
<tr><td>2</td><td>Select the Variables</td><td>There are a lot of variables such as different types of pollutants or meteorological features in this dataset. In this project that is related to air pollution, our prediction part would focus more on the effect of different types of pollutants in each state. Hence, the variables that are selected are related to pollutants. For instance, O3 count, O3 minimum and maximum values or the median.

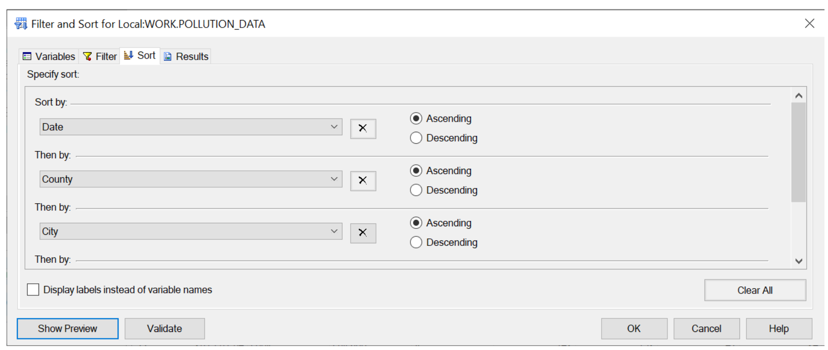


</td></tr>
</table>

<table>
<tr>
<td>3</td>
<td>Filter the Variables</td>
<td>The variables selected are able to set the range and filter the unwanted values or remove the outliers. By using the relationship such as less than or equal to , greater than , is not missing , is missing ,etc

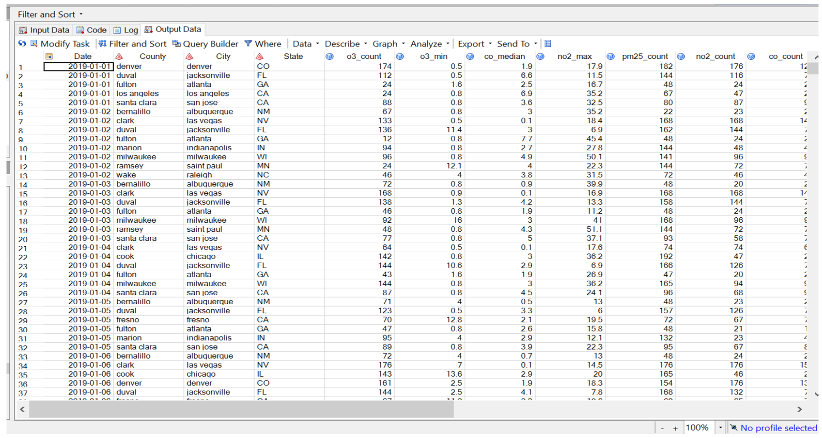


</td>
</tr>
<tr>
<td>4</td>
<td>Sort the Variables</td>
<td>The variables are sorted by date, county, state and city.



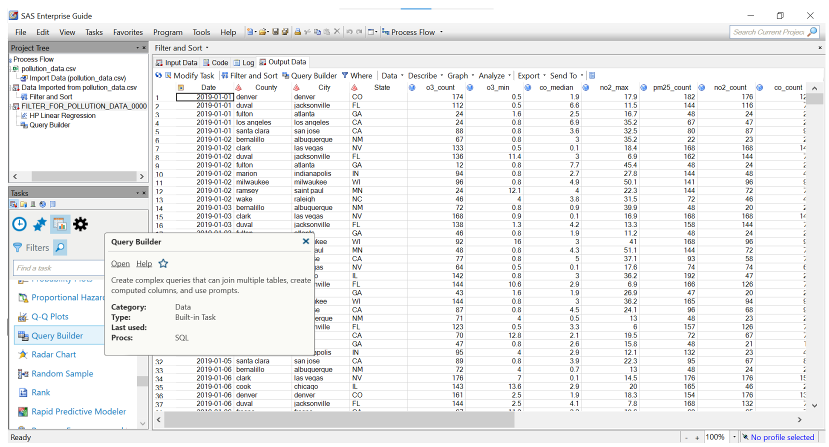
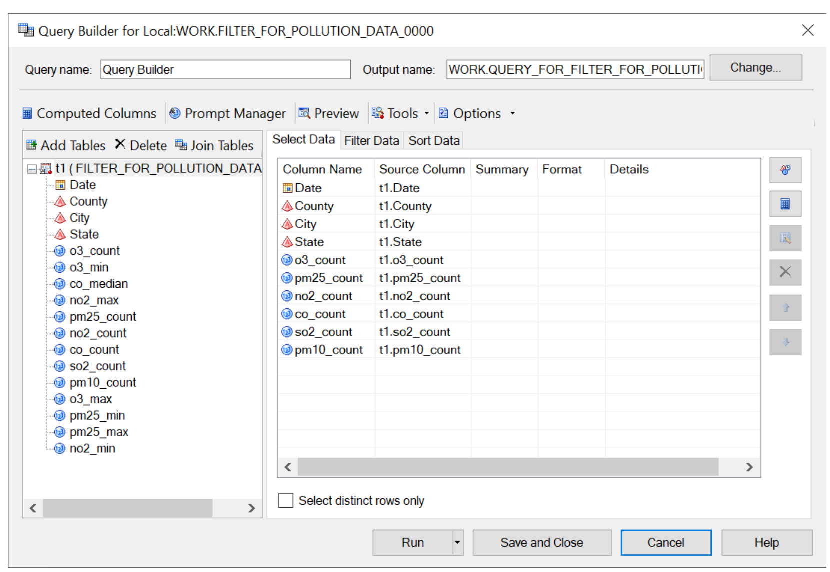
</td>
</tr>
</table>

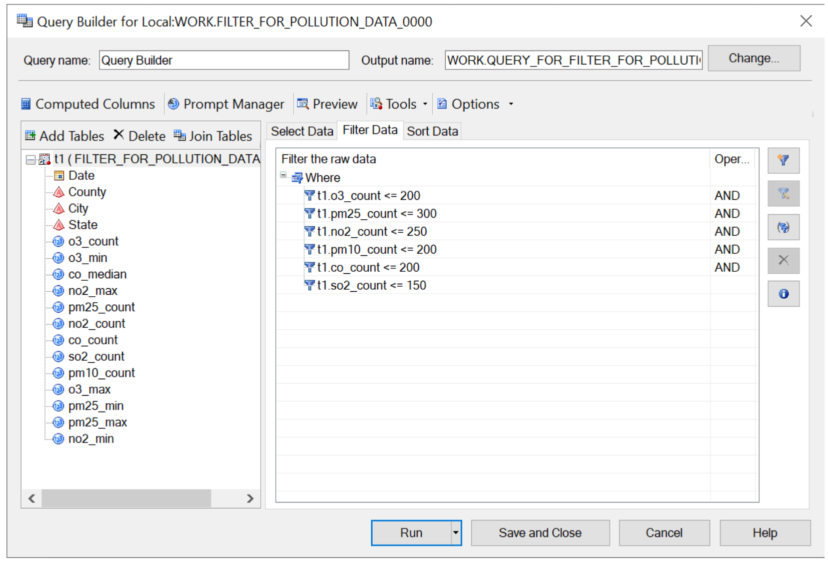
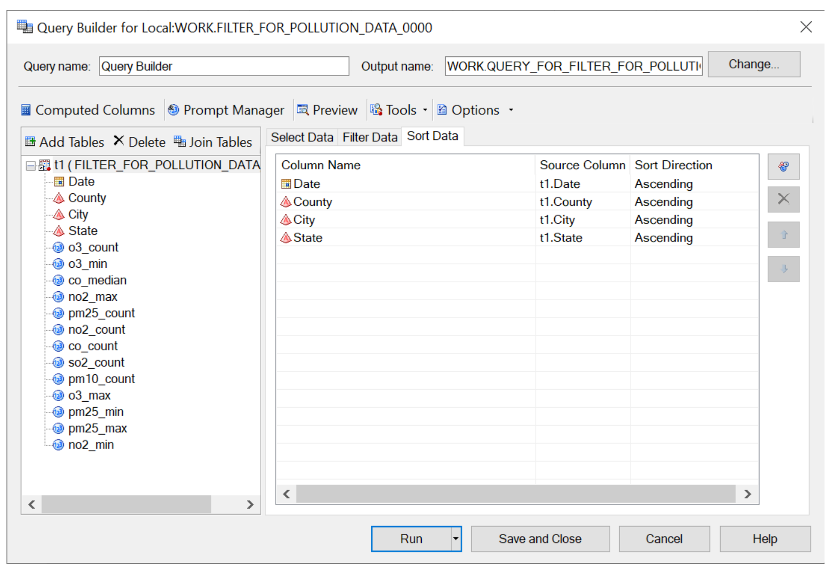
	Result	Data is filtered and the result shows there are no more missing values on it. The output tables show the values such as minimum, maximum, median and count for each pollutant only.



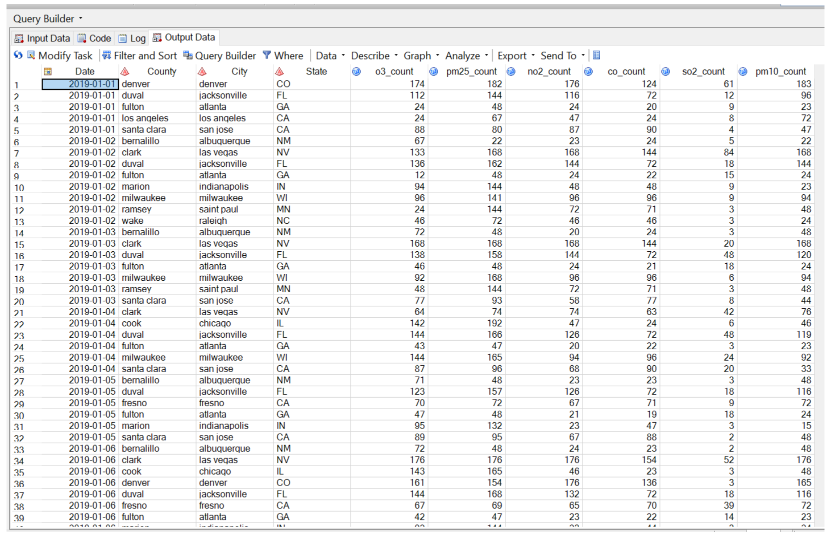
	Date	County	City	State	o3_count	o3_min	co_median	no2_max	pm25_count	no2_count
1	2019-01-01	denver	denver	CO	174	0.5	1.9	17.9	182	176
2	2019-01-01	duval	jacksonville	FL	112	0.5	6.6	11.5	144	116
3	2019-01-01	fulton	atlanta	GA	24	1.6	2.5	16.7	48	24
4	2019-01-01	los angeles	los angeles	CA	24	0.8	6.9	35.2	67	47
5	2019-01-01	santa clara	san jose	CA	88	0.8	3.6	32.5	80	87
6	2019-01-02	bernalillo	albuquerque	NM	67	0.8	3	35.2	22	23
7	2019-01-02	clark	las vegas	NV	133	0.5	0.1	18.4	168	168
8	2019-01-02	duval	jacksonville	FL	136	11.4	3	6.9	162	144
9	2019-01-02	fulton	atlanta	GA	12	0.8	7.7	45.4	48	24
10	2019-01-02	marion	indianapolis	IN	94	0.8	2.7	27.8	144	48
11	2019-01-02	milwaukee	milwaukee	WI	96	0.8	4.9	50.1	141	96
12	2019-01-02	ramsey	saint paul	MN	24	12.1	4	22.3	144	72
13	2019-01-02	wake	raleigh	NC	46	4	3.8	31.5	72	46
14	2019-01-03	bernalillo	albuquerque	NM	72	0.8	0.9	39.9	48	20
15	2019-01-03	clark	las vegas	NV	168	0.9	0.1	16.9	168	168
16	2019-01-03	duval	jacksonville	FL	138	1.3	4.2	13.3	158	144
17	2019-01-03	fulton	atlanta	GA	46	0.8	1.9	11.2	48	24
18	2019-01-03	milwaukee	milwaukee	WI	92	16	3	41	168	96
19	2019-01-03	ramsey	saint paul	MN	48	0.8	4.3	51.1	144	72
20	2019-01-03	santa clara	san jose	CA	77	0.8	5	37.1	93	58
21	2019-01-04	clark	las vegas	NV	64	0.5	0.1	17.6	74	74
22	2019-01-04	cook	chicago	IL	142	0.8	3	36.2	192	47
23	2019-01-04	duval	jacksonville	FL	144	10.6	2.9	6.9	166	126
24	2019-01-04	fulton	atlanta	GA	43	1.6	1.9	26.9	47	20
25	2019-01-04	milwaukee	milwaukee	WI	144	0.8	3	36.2	165	94
26	2019-01-04	santa clara	san jose	CA	87	0.8	4.5	24.1	96	68
27	2019-01-05	bernalillo	albuquerque	NM	71	4	0.5	13	48	23
28	2019-01-05	duval	jacksonville	FL	123	0.5	3.3	6	157	126
29	2019-01-05	fresno	fresno	CA	70	12.8	2.1	19.5	72	67
30	2019-01-05	fulton	atlanta	GA	47	0.8	2.6	15.8	48	21
31	2019-01-05	marion	indianapolis	IN	95	4	2.9	12.1	132	23
32	2019-01-05	santa clara	san jose	CA	89	0.8	3.9	22.3	95	67
33	2019-01-06	bernalillo	albuquerque	NM	72	4	0.7	13	48	24
34	2019-01-06	clark	las vegas	NV	176	7	0.1	14.5	176	176
35	2019-01-06	cook	chicago	IL	143	13.6	2.9	20	165	46
36	2019-01-06	denver	denver	CO	161	2.5	1.9	18.3	154	176
37	2019-01-06	duval	jacksonville	FL	144	2.5	4.1	7.8	168	132

2.1.2 Data Pre-Processing with Query Builder in SAS Enterprise Guide

No.	Steps	Screenshots and Explanation
1	**After Filter and Sort the data > Select Query Builder from the Task (located at the bottom left)**	The below shows the original dataset with some missing and null values. 
2	**Select the Variables**	Count for different types of pollutants is specified and required for further analysis. Hence, in Query Builder the pollutants count is selected and the minimum, maximum, and median variables are ignored. 

<table>
<tr>
<td>3</td>
<td>Filter the Variables</td>
<td>The count for each pollutant is filtered again according to the needs.

</td>
</tr>
<tr>
<td>4</td>
<td>Sort the Variables</td>
<td>The data is sorted in ascending order using the same variables as in Filter and Sort.

</td>
</tr>
</table>

	Result	Only the count of each type of pollutant remains.



	Date	County	City	State	o3_count	pm25_count	no2_count	co_count	so2_count	pm10_count
1	2019-01-01	denver	denver	CO	174	182	176	124	61	183
2	2019-01-01	duval	jacksonville	FL	112	144	116	72	12	96
3	2019-01-01	fulton	atlanta	GA	24	48	24	20	9	23
4	2019-01-01	los angeles	los angeles	CA	24	67	47	24	8	72
5	2019-01-01	santa clara	san jose	CA	88	80	87	90	4	47
6	2019-01-02	bernalillo	albuquerque	NM	67	22	23	24	5	22
7	2019-01-02	clark	las vegas	NV	133	168	168	144	84	168
8	2019-01-02	duval	jacksonville	FL	136	162	144	72	18	144
9	2019-01-02	fulton	atlanta	GA	12	48	24	22	15	24
10	2019-01-02	marion	indianapolis	IN	94	144	48	48	9	23
11	2019-01-02	milwaukee	milwaukee	WI	96	141	96	96	9	94
12	2019-01-02	ramsey	saint paul	MN	24	144	72	71	3	48
13	2019-01-02	wake	raleigh	NC	46	72	46	46	3	24
14	2019-01-03	bernalillo	albuquerque	NM	72	48	20	24	3	48
15	2019-01-03	clark	las vegas	NV	168	168	168	144	20	168
16	2019-01-03	duval	jacksonville	FL	138	158	144	72	48	120
17	2019-01-03	fulton	atlanta	GA	46	48	24	21	18	24
18	2019-01-03	milwaukee	milwaukee	WI	92	168	96	96	6	94
19	2019-01-03	ramsey	saint paul	MN	48	144	72	71	3	48
20	2019-01-03	santa clara	san jose	CA	77	93	58	77	8	44
21	2019-01-04	clark	las vegas	NV	64	74	74	63	42	76
22	2019-01-04	cook	chicago	IL	142	192	47	24	6	46
23	2019-01-04	duval	jacksonville	FL	144	166	126	72	48	119
24	2019-01-04	fulton	atlanta	GA	43	47	20	22	3	23
25	2019-01-04	milwaukee	milwaukee	WI	144	165	94	96	24	92
26	2019-01-04	santa clara	san jose	CA	87	96	68	90	20	33
27	2019-01-05	bernalillo	albuquerque	NM	71	48	23	23	3	48
28	2019-01-05	duval	jacksonville	FL	123	157	126	72	18	116
29	2019-01-05	fresno	fresno	CA	70	72	67	71	9	72
30	2019-01-05	fulton	atlanta	GA	47	48	21	19	18	24
31	2019-01-05	marion	indianapolis	IN	95	132	23	47	3	15
32	2019-01-05	santa clara	san jose	CA	89	95	67	88	2	48
33	2019-01-06	bernalillo	albuquerque	NM	72	48	24	23	2	48
34	2019-01-06	clark	las vegas	NV	176	176	176	154	52	176
35	2019-01-06	cook	chicago	IL	143	165	46	23	3	48
36	2019-01-06	denver	denver	CO	161	154	176	136	3	165
37	2019-01-06	duval	jacksonville	FL	144	168	132	72	18	116
38	2019-01-06	fresno	fresno	CA	67	69	65	70	39	72
39	2019-01-06	fulton	atlanta	GA	42	47	23	22	14	23

2.2 SAS Enterprise Miner

Another method tried for Data Preprocessing in this project is SAS Enterprise Miner with Replacement and Impute node. SAS Enterprise Miner is an advanced analytics data mining tool that helps in developing descriptive and predictive models. Data cleaning can be done in SAS Enterprise Miner with various nodes of different practices such as transformation, replacement, variable selection and others. In this project, we have utilised a replacement node followed by an impute node for Data preprocessing. Replacement Node is a data mining preprocessing node that is utilised to replace outliers for interval variables and unknown values for class variables by generating scoring code. The outliers and unknown values will be then treated as missing values. Impute Node is used to impute the missing values before the data are being fitted into the models. New variables with prefaced IMP_ will be created for variables with imputed missing values as the original variables will not be overwritten.

2.2.1 Data Pre-Processing with Replacement and Impute Node in SAS Enterprise Miner

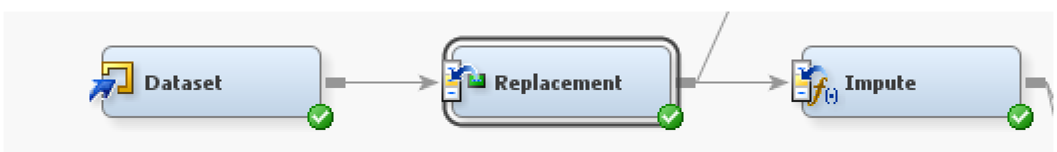


Figure 2.2: Nodes used in Data-Preprocessing

<table>
<tr><th>No</th><th>Steps</th><th>Screenshots and Explanation</th></tr>
<tr><td>1</td><td>Identify the Target, Input and Rejected Variables.</td><td>Target Variables: State
Input: count of pollutants (PM2.5, PM10, NO2, O3, CO, SO2)
<table>
<tr><th>Name</th><th>Role</th><th>Level</th><th>Report</th><th>Order</th><th>Drop</th><th>Lower Limit</th><th>Upper Limit</th></tr>
<tr><td>no2_count</td><td>Input</td><td>Interval</td><td>No</td><td></td><td>No</td><td>.</td><td>.</td></tr>
<tr><td>o3_count</td><td>Input</td><td>Interval</td><td>No</td><td></td><td>No</td><td>.</td><td>.</td></tr>
<tr><td>co_count</td><td>Input</td><td>Interval</td><td>No</td><td></td><td>No</td><td>.</td><td>.</td></tr>
<tr><td>pm25_count</td><td>Input</td><td>Interval</td><td>No</td><td></td><td>No</td><td>.</td><td>.</td></tr>
<tr><td>so2_count</td><td>Input</td><td>Interval</td><td>No</td><td></td><td>No</td><td>.</td><td>.</td></tr>
<tr><td>pm10_count</td><td>Input</td><td>Interval</td><td>No</td><td></td><td>No</td><td>.</td><td>.</td></tr>
<tr><td>County</td><td>Rejected</td><td>Nominal</td><td>No</td><td></td><td>No</td><td>.</td><td>.</td></tr>
<tr><td>pressure_min</td><td>Rejected</td><td>Interval</td><td>No</td><td></td><td>No</td><td>.</td><td>.</td></tr>
<tr><td>pressure_varia</td><td>Rejected</td><td>Interval</td><td>No</td><td></td><td>No</td><td>.</td><td>.</td></tr>
<tr><td>pressure_medi</td><td>Rejected</td><td>Interval</td><td>No</td><td></td><td>No</td><td>.</td><td>.</td></tr>
<tr><td>pressure_max</td><td>Rejected</td><td>Interval</td><td>No</td><td></td><td>No</td><td>.</td><td>.</td></tr>
<tr><td>so2_median</td><td>Rejected</td><td>Interval</td><td>No</td><td></td><td>No</td><td>.</td><td>.</td></tr>
<tr><td>so2_min</td><td>Rejected</td><td>Interval</td><td>No</td><td></td><td>No</td><td>.</td><td>.</td></tr>
<tr><td>City</td><td>Rejected</td><td>Nominal</td><td>No</td><td></td><td>No</td><td>.</td><td>.</td></tr>
<tr><td>so2_max</td><td>Rejected</td><td>Interval</td><td>No</td><td></td><td>No</td><td>.</td><td>.</td></tr>
<tr><td>pm25_max</td><td>Rejected</td><td>Interval</td><td>No</td><td></td><td>No</td><td>.</td><td>.</td></tr>
</table></td></tr>
</table>

Population_Not	Rejected	Nominal	No		No	.	.
no2_min	Rejected	Interval	No		No	.	.
State	Target	Nominal	No		No	.	.

2

Node: Replacement

Replace the outliers and unknown variables with Missing Value.

For Interval Variables, the Limit Methods is set to User Specify and the Replacement Upper Limit is identified after studying the datasets. This can help to remove the outliers as they will be replaced as Missing Values.

Columns: Label Mining Basic Statistics

Name	Use	Limit Method	Replacement Lower Limit	Replacement Upper Limit	Replace Method	Lower Replacement Value	Upper F Value
[illegible]	Default	Default	.	.	Default	.	
no2_count	Default	User Specified	.	250	Missing	.	
no2_max	Default	Default	.	.	Default	.	
no2_median	Default	Default	.	.	Default	.	
no2_min	Default	Default	.	.	Default	.	
no2_variance	Default	Default	.	.	Default	.	
o3_count	Default	User Specified	.	200	Missing	.	
o3_max	Default	Default	.	.	Default	.	
o3_median	Default	Default	.	.	Default	.	
o3_min	Default	Default	.	.	Default	.	
o3_variance	Default	Default	.	.	Default	.	
past_week_av	Default	Default	.	.	Default	.	
pm10_count	Default	User Specified	.	200	Missing	.	
pm10_max	Default	Default	.	.	Default	.	
pm10_median	Default	Default	.	.	Default	.	
pm10_min	Default	Default	.	.	Default	.	
pm10_variance	Default	Default	.	.	Default	.	
pm25_count	Default	User Specified	.	300	Missing	.	
pm25_max	Default	Default			Default		

For Class Variables, the replacement value is set to _MISSING_ to replace the unknown values detected.

Replacement Editor-WORK.OUTCLASS

Variable	Formatted Value	Replacement Value	Frequency Count	Type	Character Unformatted Value	Numeric Value
State	UT		667	C	UT	.
State	MA		666	C	MA	.
State	WA		666	C	WA	.
State	NV		663	C	NV	.
State	IN		659	C	IN	.
State	VA		649	C	VA	.
State	SC		550	C	SC	.
State	_UNKNOWN_	_MISSING_	.	C		.

Output and Result:

Limits and Replacement Values for Interval Variables

Variable	Replace Variable	Lower limit	Lower Replacement Value	Upper Limit	Upper Replacement Value
co_count	REP_co_count	.	.	200	.
no2_count	REP_no2_count	.	.	250	.
o3_count	REP_o3_count	.	.	200	.
pm10_count	REP_pm10_count	.	.	200	.
pm25_count	REP_pm25_count	.	.	300	.
so2_count	REP_so2_count	.	.	150	.

Replacement Values for Class Variables

Variable	Formatted Value	Type	Character Unformatted Value	Numeric Value	Replacement Value	Label
State	Unknown	C		.	_blank_	

Number of Replacement Done:

Replacement Counts

Obs	Variable	Label	Role	Train
1	State		TARGET	0
2	co_count	co_count	INPUT	416
3	no2_count	no2_count	INPUT	384
4	o3_count	o3_count	INPUT	1197
5	pm10_count	pm10_count	INPUT	692
6	pm25_count	pm25_count	INPUT	284
7	so2_count	so2_count	INPUT	3

<table>
<tr>
<td>3</td>
<td>Node: Impute

Impute the missing values.</td>
<td>The variables have their new labels with REP_pollutants count. The missing values will then be imputed in this stage.
<table>
<tr><th>Name</th><th>Use</th><th>Method</th><th>Use Tree</th><th>Role</th><th>Level</th></tr>
<tr><td>Population_Not</td><td>Default</td><td>Default</td><td>Default</td><td>Rejected</td><td>Nominal</td></tr>
<tr><td>Population_Sta</td><td>Default</td><td>Default</td><td>Default</td><td>Rejected</td><td>Nominal</td></tr>
<tr><td>REP_State</td><td>Default</td><td>Default</td><td>Default</td><td>Target</td><td>Nominal</td></tr>
<tr><td>REP_co_count</td><td>Default</td><td>Default</td><td>Default</td><td>Input</td><td>Interval</td></tr>
<tr><td>REP_no2_coun</td><td>Default</td><td>Default</td><td>Default</td><td>Input</td><td>Interval</td></tr>
<tr><td>REP_o3_count</td><td>Default</td><td>Default</td><td>Default</td><td>Input</td><td>Interval</td></tr>
<tr><td>REP_pm10_cou</td><td>Default</td><td>Default</td><td>Default</td><td>Input</td><td>Interval</td></tr>
<tr><td>REP_pm25_cou</td><td>Default</td><td>Default</td><td>Default</td><td>Input</td><td>Interval</td></tr>
<tr><td>REP_so2_coun</td><td>Default</td><td>Default</td><td>Default</td><td>Input</td><td>Interval</td></tr>
</table>
Output and Result:
Imputation Summary
Number Of Observations
<table>
<tr><th>Variable Name</th><th>Impute Method</th><th>Imputed Variable</th><th>Indicator Variable</th><th>Impute Value</th><th>Role</th><th>Measurement Level</th><th>Label</th><th>Number of Missing for TRAIN</th></tr>
<tr><td>REP_co_count</td><td>MEAN</td><td>IMP_REP_co_count</td><td>M_REP_co_count</td><td>57.2177</td><td>INPUT</td><td>INTERVAL</td><td>Replacement: co_count</td><td>11474</td></tr>
<tr><td>REP_no2_count</td><td>MEAN</td><td>IMP_REP_no2_count</td><td>M_REP_no2_count</td><td>72.2710</td><td>INPUT</td><td>INTERVAL</td><td>Replacement: no2_count</td><td>12422</td></tr>
<tr><td>REP_o3_count</td><td>MEAN</td><td>IMP_REP_o3_count</td><td>M_REP_o3_count</td><td>69.8013</td><td>INPUT</td><td>INTERVAL</td><td>Replacement: o3_count</td><td>2843</td></tr>
<tr><td>REP_pm10_count</td><td>MEAN</td><td>IMP_REP_pm10_count</td><td>M_REP_pm10_count</td><td>71.2049</td><td>INPUT</td><td>INTERVAL</td><td>Replacement: pm10_count</td><td>19323</td></tr>
<tr><td>REP_pm25_count</td><td>MEAN</td><td>IMP_REP_pm25_count</td><td>M_REP_pm25_count</td><td>91.7774</td><td>INPUT</td><td>INTERVAL</td><td>Replacement: pm25_count</td><td>746</td></tr>
<tr><td>REP_so2_count</td><td>MEAN</td><td>IMP_REP_so2_count</td><td>M_REP_so2_count</td><td>26.5141</td><td>INPUT</td><td>INTERVAL</td><td>Replacement: so2_count</td><td>20923</td></tr>
</table>
</td>
</tr>
</table>

3. Model Diagram and Explanation

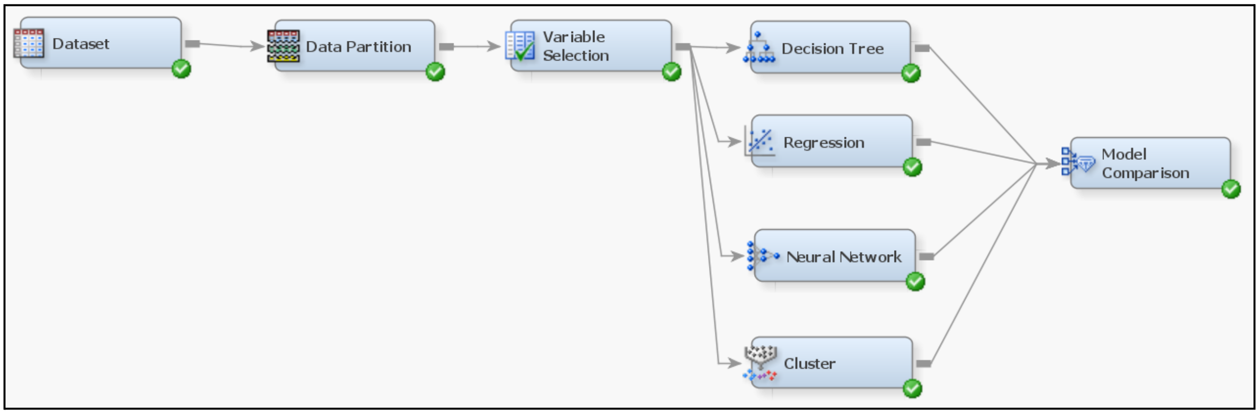


Figure 3.1: Model Diagram

The Figure 3.1 above shows the model diagram designed for this project. A model diagram is a visual representation of a data mining model showing the relationships between the variables in the models. This diagram is created in SAS Enterprise Miner, a software by SAS that provides a variety of modelling techniques and generates model diagrams with the user interface.

The **data preprocessing** which is the data cleaning method chosen is Filter and Sort , followed by Query Builder in SAS Enterprise Guide. Thus, we have exported the cleaned dataset in the format of SAS Table from SAS Enterprise Guide. A new Datasource is created in SAS Enterprise Miner and acts as the input in this model diagram. This dataset continues to be prepared by selecting targets within the first node. The Target chosen is the State while the Input chosen is the Count of the Pollutants, which include PM25, PM10, NO2, O3, CO, and SO2. Other variables are set to Rejected.

It then connects to the **Data Partition node**, allocating the data into 80% training that is used for preliminary model fitting and 20% validation that is used to assess the appropriateness of the model chosen. The partitioning method used is Stratified that all observations have the equal probability of being written to one of the partitioned dataset to help in improving the classification precision of the fitted models. Figure 3.2 below shows the data partitioned for 80% train and 20% validate.

Partition Summary

Type	Data Set	Number of Observations
DATA	EMWS4.Ids2_DATA	6168
TRAIN	EMWS4.Part_TRAIN	4923
VALIDATE	EMWS4.Part_VALIDATE	1245

Figure 3.2: Data Partition Summary

Next, the **Variable Selection node** is joined next to remove the irrelevant input to minimise the probability of overfitting and improve the prediction performance. The variables with R-squared values <0.05 will be rejected as they are less significant compared to others to the target variable in the model. For the figures below, Figure 3.3 shows the R-Square values chart for the input variables and Figure 3.4 shows the R-Square values.

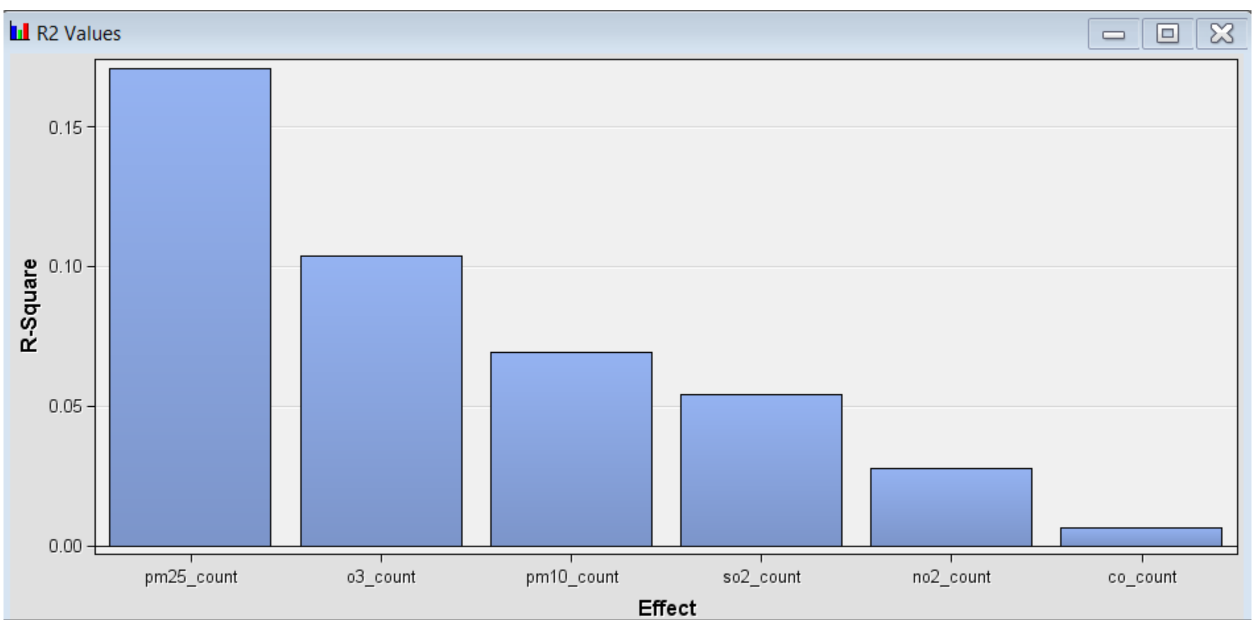


Figure 3.3: R-Square Values Chart

The DMINE Procedure

R-Squares for Target Variable: _DUMMY_TARGET_

Effect		DF	R-Square	
AOV16:	pm25_count	15	0.433824	
AOV16:	o3_count	15	0.253190	
AOV16:	no2_count	15	0.241851	
AOV16:	pm10_count	15	0.236024	
Var:	pm25_count	1	0.171037	
Var:	o3_count	1	0.103656	
AOV16:	co_count	15	0.093960	
Var:	pm10_count	1	0.069156	
AOV16:	so2_count	15	0.067555	
Var:	so2_count	1	0.054318	
Var:	no2_count	1	0.027582	R2 < MINR2
Var:	co_count	1	0.006591	R2 < MINR2

Figure 3.4: R-Square Values

Variable Selection

Variable Name	Role	Measurement Level	Type	Label	Reasons for Rejection
co_count	Rejected	Interval	Numeric		Varsel:Small R-square value
no2_count	Rejected	Interval	Numeric		Varsel:Small R-square value
o3_count	Input	Interval	Numeric		
pm10_count	Input	Interval	Numeric		
pm25_count	Input	Interval	Numeric		
so2_count	Input	Interval	Numeric		

Figure 3.5: Rejected Variable

The Figure 3.5 above shows the role of the variables either input or rejected in the Variable Selection node. CO_count and NO2_count are being rejected due to the low R-squared value.

Effects Chosen for Target: _DUMMY_TARGET_

Effect		DF	R-Square	F Value	p-Value	Sum of Squares	Error Mean Square
Var:	pm25_count	1	0.171037	1015.335399	<.0001	112.154955	0.110461
Var:	so2_count	1	0.005442	32.514249	<.0001	3.568702	0.109758
Var:	pm10_count	1	0.000580	3.468940	0.0626	0.380553	0.109703
Var:	o3_count	1	0.000991	5.931483	0.0149	0.650050	0.109593

Figure 3.6: Effects of Chosen Variables

Figure 3.6 above shows the effect of variables chosen. As a result, O3_count, pm10_count, pm25_count and SO2_count as the pollutants will remain as the input to the algorithm model as they have higher R-squared value.

Next, **four algorithm models are implemented** in this project: Decision Tree, Regression, Neural Network, and Clustering, represented by the four nodes connected to the Variable Selection node.

The first algorithm we use is the **decision tree**. A decision tree is a non-parametric supervised learning algorithm, which is utilised for both classification and regression tasks. It has a hierarchical tree structure, which consists of a root node, branches, internal nodes and leaf nodes. The leaf nodes represent all the possible outcomes within the dataset.

Regression is another algorithm used in our project that is used to predict the probability of the target value based on the input variables. The regression type used is **Logistic Regression** with stepwise selection models that will add the effects that are important with the target and remove the effects that existed in the model that are not important with the target .

Next, the **neural network** model to recognize the hidden pattern and correlation in raw data. The network architecture chosen for the network training is Multilayer Perceptron that can accept numbers of input.

The fourth algorithm used is **clustering** that is used to place the objects into clusters suggested by the data. Segment is set as the model role that will be assigned to the cluster variables and Standardization is used to divide the variables values by standard deviation. The results and analysis will be shown under section Model Practical Implementation and Comparison below. Lastly, the four model nodes are connected to the **Model Comparison node** to compare the accuracy between the models selected and evaluate their performances.

4. Model Practical Implementation and Comparisons (Practical)

4.1 Decision Tree

Decision trees are a class of supervised learning in data mining techniques that separate a huge collection of heterogeneous records into smaller groups of homogenous records by applying the directed knowledge discovery (Ghoson, A. M. , 2011). Directed knowledge discovery is mainly focused on achieving the result as it will explain and analyse the target fields in terms of the input fields to figure out the patterns for the prediction of future events by using a chain of decision rules. Hence, decision trees can provide predictive and explanatory models as the decision tree model contains the decision rules to explain the reason for certain decisions.

Decision tree models are explanatory models which are made up of simple English rules so that the rules are clear and easily understandable by people. The models include a chain of decision rules that differentiate the records in different bins or classes called nodes. The topmost node in the tree is the root node (Tutorialspoint, 2022). Each node may have two or more children or maybe have no child, which is called leaf node. The dataset has to undergo data partition which separates the dataset into two parts: training and validate sets. The training set is a set of data used for learning by the model. The validate set is the data that is used to prevent biassed evaluation of models fitted on the training sets while tuning model hyperparameters (Samarth Agrawal, May 17 2021). Furthermore, the validate set plays a crucial role in model preparation , for instance feature selection. The test set is used to assess the performance and accuracy of fully-specified classifiers (Brownlee, J. , July 14 2017).

In this project , we implement 80% of training datasets and 20% of validation datasets in data partitions. Among the variables in this dataset, we will mainly focus on the pollutants count. We used the variable selection node to select the top 4 pollutants that impact our overall analysis, which is O3, pm10, pm25 and SO2 count. After that , we added a decision tree node in the diagram using SAS Enterprise Miner. The below shows the results of the node:

Tree

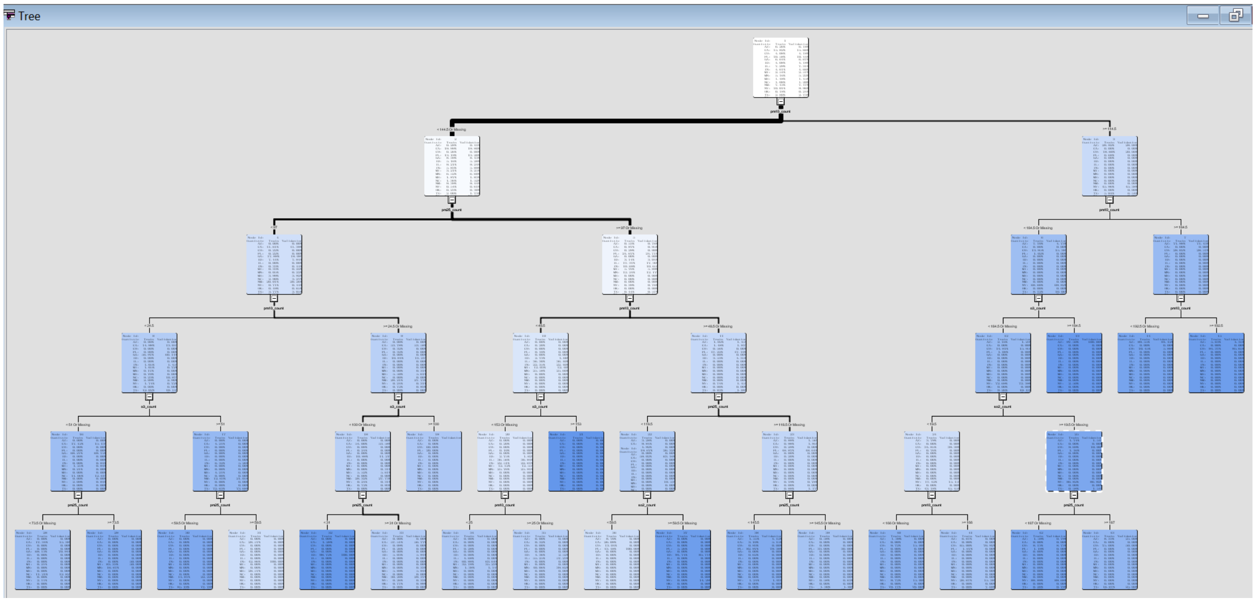


Figure 4.1.1 Tree

There are 43 nodes in the decision tree in Figure 4.1.1. The decision tree has 2 branches, binary splits and the tree depth is 5, while the decision tree has 5 generations.The nodes are coloured from light to dark, corresponding to high to low percentage of correctly classified observations. In the decision tree, the aim is to split until it reaches the maximum purity level .

Figure 4.1.2 shows node Id 1 which is the root node and also known as the parent node.

Node Id:	1	
Statistic	Train	Validation
AZ:	6.26%	6.18%
CA:	15.82%	15.66%
CO:	4.08%	4.18%
FL:	10.58%	10.44%
GA:	6.64%	6.67%
ID:	4.08%	4.18%
IL:	7.29%	7.31%
IN:	4.61%	4.66%
MI:	2.54%	2.57%
MN:	5.16%	5.22%
MS:	1.48%	1.45%
NC:	1.08%	1.20%
NM:	7.43%	7.47%
NV:	10.01%	9.96%
OK:	0.18%	0.24%
TX:	2.80%	2.73%
WI:	9.93%	9.88%
Count:	4923	1245

Figure 4.1.2 Root Node

The root node is the highest node in the tree structure and has no parent node. It is a global element that represents the entire message of the tree. The figure shows the original train and validate percentage for each country in the dataset before the splitting according to the values.

In this case, the tree splits pm10 count into two branches from the root node using the decision rule less than 144.5 or missing and larger than 144.5. It is clearly to be seen that the decision rule that indicates <144.5 or missing with the light colour has the greater count compared to another. After splitting the pm10 count , the tree will be continued with the other pollutants count.

Score Rankings Overlay: State

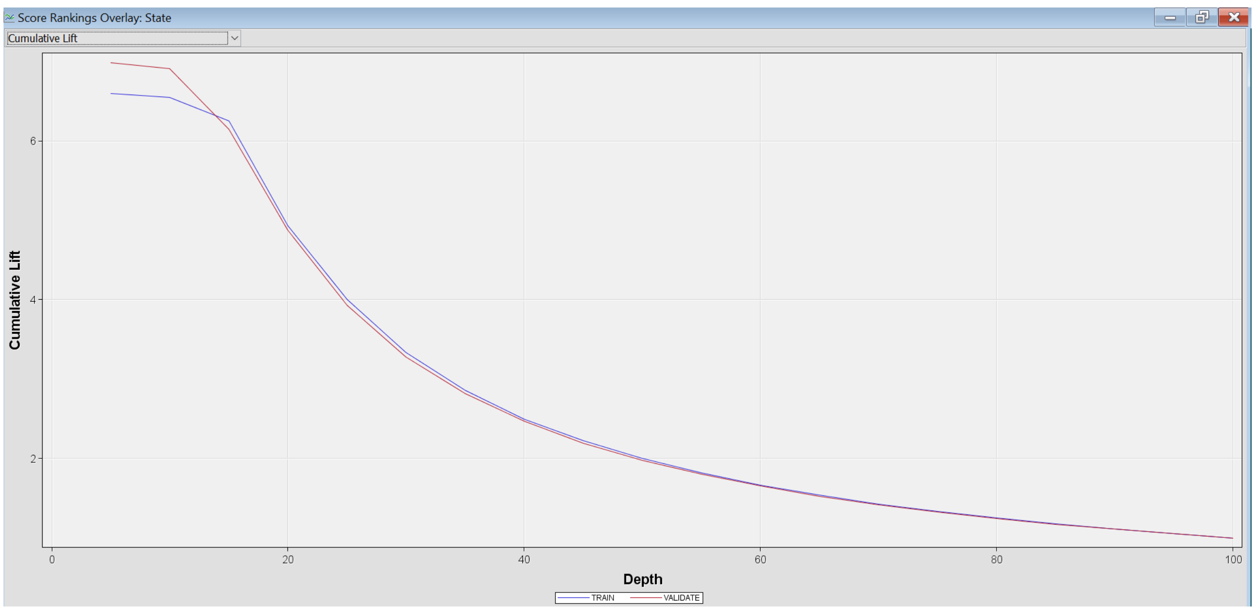


Figure 4.1.3 Score Ranking Overlay: State

Output

Train Sets	Validate Sets

Train Sets

Data Role=TRAIN Target Variable=State Target Label=' '

Depth	Gain	Lift	Cumulative Lift	% Response	Cumulative % Response	Number of Observations	Mean Posterior Probability
5	560.003	6.60003	6.60003	65.5579	65.5579	247	0.65558
10	554.956	6.49890	6.54956	64.5533	65.0566	246	0.64553
15	525.529	5.66554	6.25529	56.2756	62.1336	246	0.56276
20	393.437	0.96623	4.93437	9.5975	49.0129	246	0.09598
25	299.919	0.25467	3.99919	2.5296	39.7238	246	0.02530
30	233.311	0.00000	3.33311	0.0000	33.1077	246	0.00000
35	185.557	0.00000	2.85557	0.0000	28.3643	247	0.00000
40	149.898	0.00000	2.49898	0.0000	24.8223	246	0.00000
45	122.157	0.00000	2.22157	0.0000	22.0668	246	0.00000
50	99.959	0.00000	1.99959	0.0000	19.8619	246	0.00000
55	81.795	0.00000	1.81795	0.0000	18.0576	246	0.00000
60	66.655	0.00000	1.66655	0.0000	16.5538	246	0.00000
65	53.844	0.00000	1.53844	0.0000	15.2812	246	0.00000
70	42.820	0.00000	1.42820	0.0000	14.1862	247	0.00000
75	33.306	0.00000	1.33306	0.0000	13.2413	246	0.00000
80	24.981	0.00000	1.24981	0.0000	12.4143	246	0.00000
85	17.634	0.00000	1.17634	0.0000	11.6846	246	0.00000
90	11.104	0.00000	1.11104	0.0000	11.0359	246	0.00000
95	5.260	0.00000	1.05260	0.0000	10.4554	246	0.00000
100	0.000	0.00000	1.00000	0.0000	9.9330	246	0.00000

Validate Sets

Data Role=VALIDATE Target Variable=State Target Label=' '

Depth	Gain	Lift	Cumulative Lift	% Response	Cumulative % Response	Number of Observations	Mean Posterior Probability
5	598.897	6.98897	6.98897	69.0476	69.0476	63	0.66241
10	591.127	6.83232	6.91127	67.5000	68.2800	62	0.64553
15	514.214	4.59148	6.14214	45.3617	60.6814	62	0.43280
20	387.260	1.04350	4.87260	10.3093	48.1389	62	0.09598
25	292.861	0.19761	3.92861	1.9523	38.8128	63	0.02366
30	228.077	0.02066	3.28077	0.2041	32.4124	62	0.00000
35	181.718	0.02066	2.81718	0.2041	27.8323	62	0.00000
40	146.901	0.02066	2.46901	0.2041	24.3927	62	0.00000
45	119.407	0.02066	2.19407	0.2041	21.6763	63	0.00000
50	97.777	0.02066	1.97777	0.2041	19.5394	62	0.00000
55	80.063	0.02066	1.80063	0.2041	17.7894	62	0.00000
60	65.290	0.02066	1.65290	0.2041	16.3298	62	0.00000
65	52.594	0.02066	1.52594	0.2041	15.0756	63	0.00000
70	41.892	0.02066	1.41892	0.2041	14.0182	62	0.00000
75	32.610	0.02066	1.32610	0.2041	13.1012	62	0.00000
80	24.484	0.02066	1.24484	0.2041	12.2984	62	0.00000
85	17.201	0.02066	1.17201	0.2041	11.5789	63	0.00000
90	10.833	0.02066	1.10833	0.2041	10.9498	62	0.00000
95	5.133	0.02066	1.05133	0.2041	10.3866	62	0.00000
100	0.000	0.02066	1.00000	0.2041	9.8795	62	0.00000

Score ranking overlay is the plot that indicates the same set of axes to simultaneously display selected statistics for both training and validation data sets (SAS, 2006). Focus on the cumulative lift, the cumulative lift has the high values in the range of 6 to 7. The graph that indicates train and validate data declined smoothly for deeper depth. The initial lift of validation data is higher than the train data. However, both lines that indicate the training and validation data become nearer when the decision tree depth goes deeper.

The Fit Statistics

Fit Statistics

Target	Target Label	Fit Statistics	Statistics Label	Train	Validation	Test
State		NOBS	Sum of Frequencies	4923	1245	.
State		MISC	Misclassification Rate	0.310989	0.299598	.
State		MAX	Maximum Absolute Error	0.999147	1	.
State		SSE	Sum of Squared Errors	2119.736	527.9893	.
State		ASE	Average Squared Error	0.025328	0.024946	.
State		RASE	Root Average Squared Error	0.159148	0.157944	.
State		DIV	Divisor for ASE	83691	21165	.
State		DFT	Total Degrees of Freedom	78768	.	

Figure 4.1.4 Fit Statistics

The Fit Statistics table is a table that contains information related to model accuracy such as the model error details, sensitivity and specificity of the model. For instance, the data that is available in Fit statistics tables are sum of frequencies, misclassification rate, maximum absolute error, sum of squared errors, average squared error, root average squared error, divisor for Average Squared Error and total degrees of freedom. All these values are calculated from the 'Misclassification Matrix' table which is also known as the confusion table. In this project, Misclassification rate will be used in model comparison to find the model that has the highest accuracy in evaluating and predicting the seriousness of pollution in each state. If you focus on the misclassification error, the training and validation data is pretty low , whereas the two errors are not that significant. It is only approximately 0.011, equivalent to 1.1 % difference.This tells that there is less opportunity that this model overfits the training data and is good to classify the class on the validation data. However, this model has to compare with other models like regression or neural networks to choose the best model among them.

The Leaf Statistics

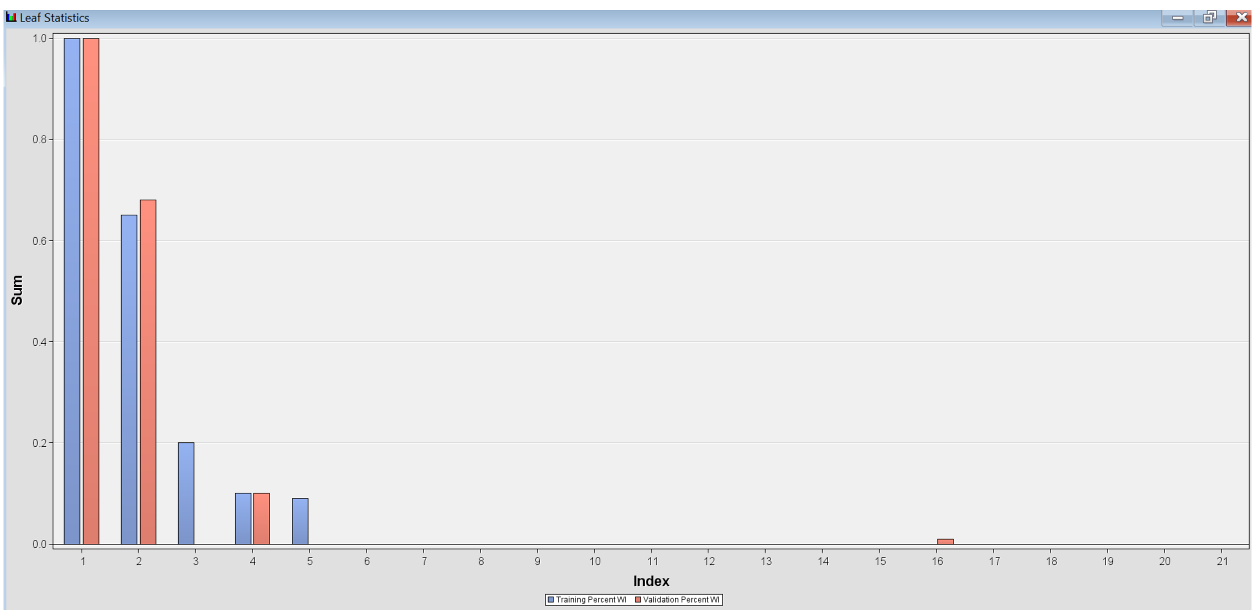


Figure 4.1.5 Leaf Statistics

The Leaf Statistics Plot is the bar chart graph that displays the summary statistics for the leaves of the currently selected subtree.

Output

Variable Importance

Variable Name	Label	Number of Splitting Rules	Importance	Validation Importance	Ratio of Validation to Training Importance
pm10_count		7	1.0000	1.0000	1.0000
pm25_count		7	0.6917	0.6793	0.9821
o3_count		4	0.3215	0.2858	0.8890
so2_count		2	0.2531	0.2582	1.0201

Figure 4.1.6 Variable Importance

Figure 4.1.6 indicates the variable importance of the decision tree result. Variable Importance is used to identify which predictors are the most useful to predict the response variable (SAS,2019). From the above figure, we can notice that pm10 count has the higher validation importance which is 1 compared to other pollutants count.

Event Classification Table

Train Set

Data Role=TRAIN Target=State Target Label=' '

False Negative	True Negative	False Positive	True Positive
34	4188	246	455

Validation Set

Data Role=VALIDATE Target=State Target Label=' '

False Negative	True Negative	False Positive	True Positive
12	1070	52	111

From the event classification table, the exact number of false negative, true negative, false positive and true positive cases in the training and validation data as predicted by the decision tree model. The misclassification rate is also shown in the fit statistics table.

Assessment Score Distribution

Data Role=TRAIN Target Variable=State Target Label=' '

Posterior Probability Range	Number of Events	Number of Nonevents	Mean Posterior Probability	Percentage
0.95-1.00	7	0	1.00000	0.1422
0.60-0.65	448	246	0.64553	14.0971
0.15-0.20	1	4	0.20000	0.1016
0.05-0.10	33	313	0.09538	7.0282
0.00-0.05	0	3871	0.00000	78.6309

Data Role=VALIDATE Target Variable=State Target Label=' '

Posterior Probability Range	Number of Events	Number of Nonevents	Mean Posterior Probability	Percentage
0.95-1.00	3	0	1.00000	0.2410
0.60-0.65	108	52	0.64553	12.8514
0.05-0.10	10	92	0.09553	8.1928
0.00-0.05	2	978	0.00000	78.7149

The above figure shows the assessment score distribution between train and validate sets. The percentage of the train is slightly different with the percentage score of validation sets.

4.2 Regression

Regression is a widely used supervised machine learning technique that predicts future outcomes or events. A regression model estimates and provides a mapping function that describes the connection or relationship between one or more independent variables and a response, dependent, or target variable. There are many different types of regression analysis techniques in machine learning, and their usage varies depending on the nature of the data.

Class Targets	
Regression Type	Logistic Regression
Link Function	Logit

Figure 4.2.1 Class Targets Configuration

The regression that we use in this project is Logistic Regression and the link function is the logit link function. Logistic Regression is a widely used supervised machine algorithm that uses the logistic function (also known as the sigmoid function) to model the probability of a certain class or event occurring. It is also a statistical method used for binary classification problems, where the goal is to predict a binary outcome based on a set of input features. The algorithm will try to find the best set of parameters (also known as weights) that maximizes the likelihood of the observed data.

SAS Enterprise Miner provides a user-friendly interface for creating and deploying logistic regression models. By default, logistic regression will attempt to predict the probability that a binary or ordinal target will acquire the event of interest as a function of one or more independent inputs (SAS Help Center, n.d.). Same as the previous model, we use 80% of the data for training and 20% of the data for validation.

Score Rankings Overlay: State

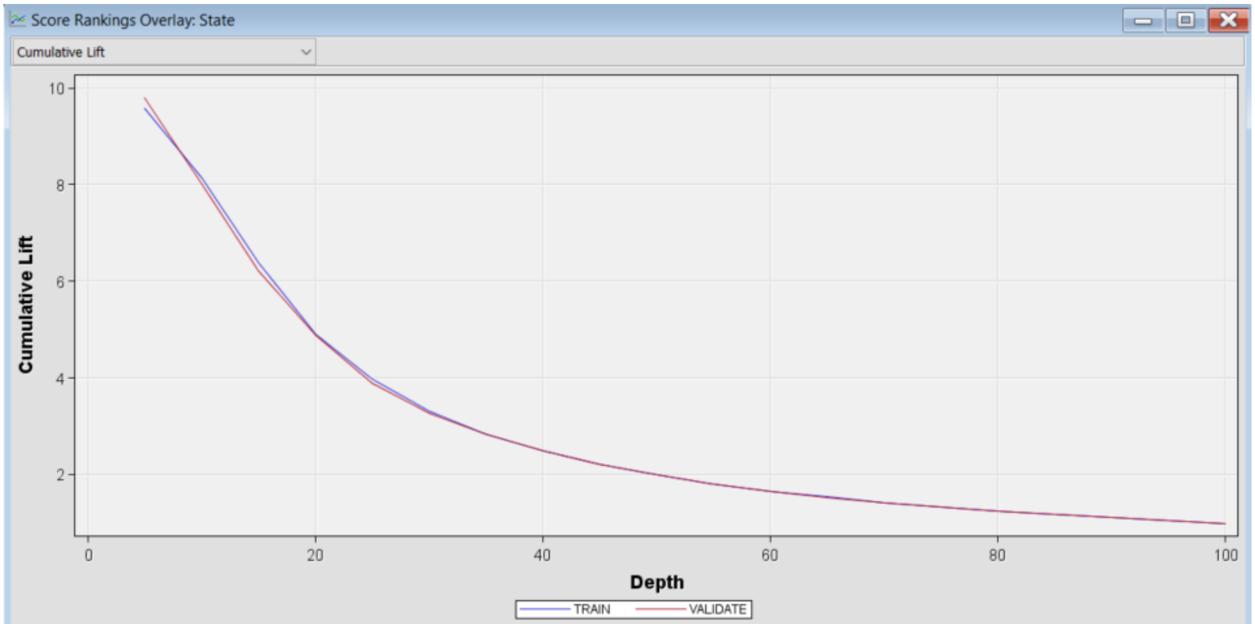


Figure 4.2.2 Score Rankings Overlay Diagram

Several statistics for each decile (group) of observations are presented on the vertical axis of a score rankings chart. The observations are sorted from highest expected profit to lowest expected profit for a nominal or ordinal aim. From the result, we can see that the cumulative lift for validate data has slighly higher lift values than the train dataset at the beginning of the first decile (depth). The cumulative lift for both train and validate data is closer as the depth goes deeper.

Fit Statistics

Target	Target Label	Fit Statistics	Statistics Label	Train	Validation	Test
State		AIC	Akaike's Information Criterion	6900.568	.	.
State		ASE	Average Squared Error	0.018181	0.017982	.
State		AVERR	Average Error Function	0.080541	0.076479	.
State		DFE	Degrees of Freedom for Error	78688	.	.
State		DFM	Model Degrees of Freedom	80	.	.
State		DFT	Total Degrees of Freedom	78768	.	.
State		DIV	Divisor for ASE	83691	21165	.
State		ERR	Error Function	6740.568	1618.675	.
State		FPE	Final Prediction Error	0.018217	.	.
State		MAX	Maximum Absolute Error	1	0.999999	.
State		MSE	Mean Square Error	0.018199	0.017982	.
State		NOBS	Sum of Frequencies	4923	1245	.
State		NW	Number of Estimate Weights	80	.	.
State		RASE	Root Average Sum of Squares	0.134835	0.134096	.
State		RFPE	Root Final Prediction Error	0.134972	.	.
State		RMSE	Root Mean Squared Error	0.134904	0.134096	.
State		SBC	Schwarz's Bayesian Criterion	7642.509	.	.
State		SSE	Sum of Squared Errors	1521.545	380.5863	.
State		SUMW	Sum of Case Weights Times Freq	83691	21165	.
State		MISC	Misclassification Rate	0.212066	0.214458	.

Figure 4.2.3: Fit Statistics

For the model comparison purpose, we will be focusing on the misclassification rate as the main statistics label to determine the best model. From the Fit Statistics result, we can observe that the difference for misclassification rate between the train and validation data is only 0.0024 which is equivalent to 0.24%. This means that the model is not overfitting or underfitting as the difference is not significant.

Event Classification Table

Data Role=TRAIN Target=State Target Label=' '

False Negative	True Negative	False Positive	True Positive
70	4298	136	419

Data Role=VALIDATE Target=State Target Label=' '

False Negative	True Negative	False Positive	True Positive
20	1091	31	103

Figure 4.2.4: Event Classification Table

From the event classification table, we are able to know about the exact number of false negative, true negative, false positive and true positive cases in the training and validation data as predicted by the logistic regression model. From the number of classification events, we can then calculate other classification matrices such as sensitivity and classification rate.

Type 3 Analysis of Effects

Effect	DF	Wald Chi-Square	Pr > ChiSq
o3_count	16	698.1782	<.0001
pm10_count	16	1165.3897	<.0001
pm25_count	16	1046.3760	<.0001
so2_count	16	670.4324	<.0001

Figure 4.2.5: Analysis of effect

By using the variable selection node, we have removed some variables which have low impact on the model. From the analysis of the effect table, we can further verify that all the 4 input variables selected by the variable selection node are having a high impact on the model by looking at the Wald Chi-Square result, in which none of them are having zero value.

4.3 Clustering

The next algorithm used in this assignment is clustering. Clustering is an unsupervised machine learning method to identify and group similar data points in a large dataset. In other words, clustering in data mining is to determine the group of objects which are similar to each other in the group but different from the object in other groups. In clustering, the datasets are divided into groups based on their similarity and each of the groups is labelled according to their data types (Sharma, R., 2022). To simplify, clustering is to take the input variables and group them according to our observations. For instance, if we have a group of students, we can cluster them based on things they have in common according to their inputs instead of the output variables. Clustering is used when we are analysing a large dataset as it can organise them into something useful without instruction. If we are not performing massive analysis, clustering is able to provide fast and accurate insights. Besides, clustering is helpful in data preparation when we are not sure of how many classes the data is divided into. Moreover, clustering can help to determine anomalies or outliers in the datasets. In this case, density-based spatial clustering of applications with noise (DBSCAN) is used to look for separate clusters that mark outliers in the datasets (explorium, 2022).

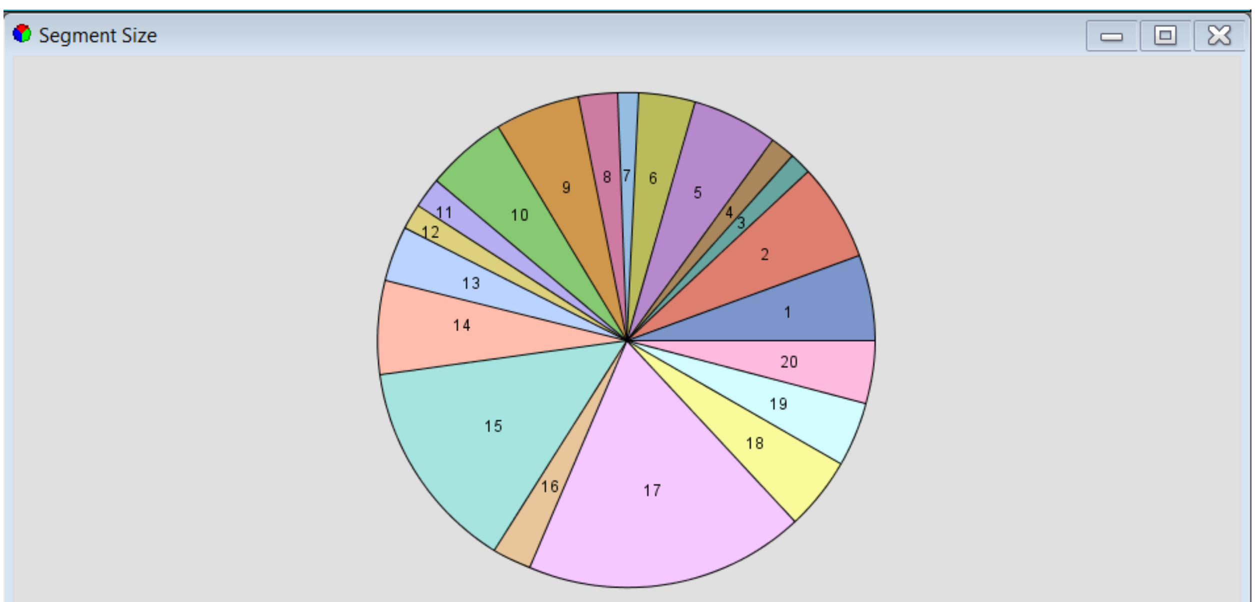


Figure 4.3.1: Clustering Segment Size

In this model, we implement a "Cluster" node to cluster the air pollution based on the counts of different types of pollutants which include O3, PM10, PM25 and SO2. Based on Figure 4.3.1, it is clearly seen that the "Cluster" node has created 20 clusters for this dataset.

The CLUSTER Procedure
Ward's Minimum Variance Cluster Analysis

Eigenvalues of the Covariance Matrix

	Eigenvalue	Difference	Proportion	Cumulative
1	7518.66744	6413.35016	0.7773	0.7773
2	1105.31728	414.87432	0.1143	0.8915
3	690.44296	331.53202	0.0714	0.9629
4	358.91093		0.0371	1.0000

Root-Mean-Square Total-Sample Standard Deviation 49.17694

Root-Mean-Square Distance Between Observations 139.0934

Figure 4.3.2: The Cluster Procedure

Figure 4.3.2 illustrates the cluster procedure using Ward's Minimum Variance Cluster Analysis. In this analysis, the table of eigenvalues of the covariance matrix is displayed. These values are used in the computation of the cubic clustering criterion. The first two columns (eigenvalue and difference) show each eigenvalue of the variables and the difference between the eigenvalue and its successor. However, the last two columns (proportion and cumulative) display the individual and cumulative proportion of variation associated with each eigenvalue (SAS, 2017).

Variable Importance

Variable Name	Label	Number of Splitting Rules	Number of Surrogate Rules	Importance
o3_count		9	14	1.00000
pm10_count		5	18	0.99027
pm25_count		5	14	0.92289
so2_count		9	10	0.85721

Figure 4.3.3: Clustering Variable Importance

Figure 4.3.3 illustrates the variable importance of the cluster result. Variable Importance is used to indicate which predictors are the most useful to predict the response variable. It displays each variable that was used to generate the clusters and their relative importance. Hence, from figure 4.3.3, we can see that the variable o3_count has the highest importance with the value of 1 while so2_count has the lowest importance of 0.85721. The higher the importance, the more accurate the clustering is and thus, the closer the model represents reality.

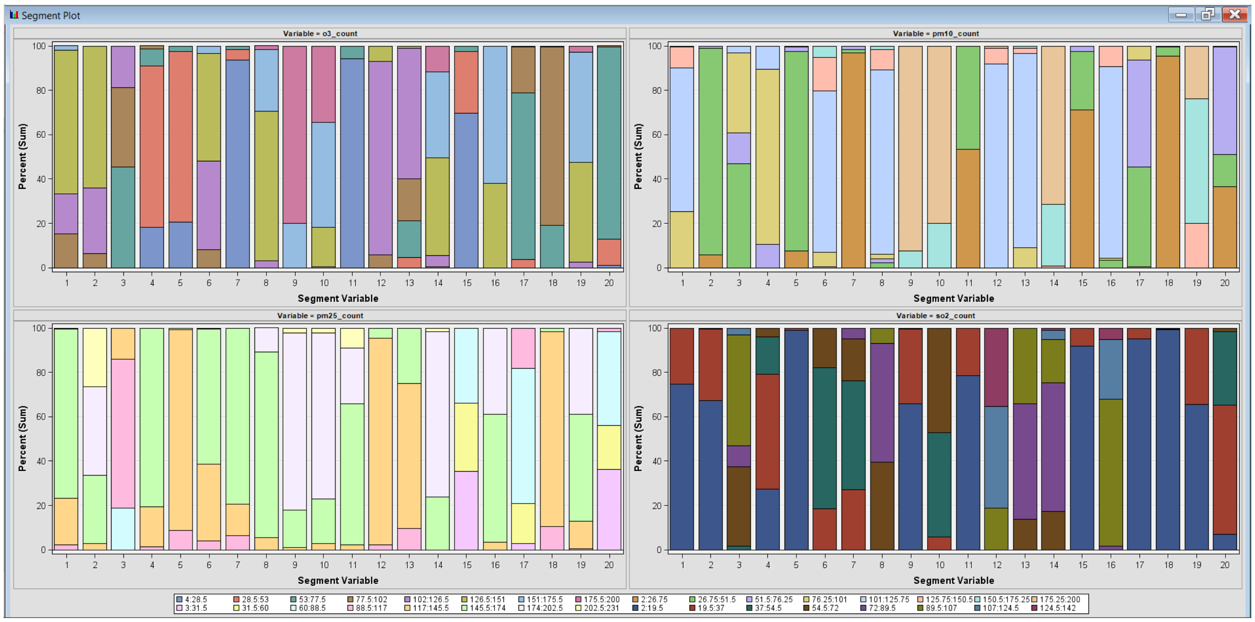


Figure 4.3.4: Segment Plot of Each Variable

Figure 4.3.4 shows the segment plot of the variables which include o3_count, pm10_count, pm25_count and so2_count. It clearly illustrates the distribution of each cluster for each variable. For example, about 97% of cluster 11 of o3_count is made up of the blue colour region which represents the count in the range of 4 to 28.5 in which the legend of the plot is shown at the bottom. On the other hand, let's take a look at So2_count, only 6% of cluster 5 of so2_count is made up of the brown colour region which represents the count in the range of 2 to 26.75.

Mean Statistics

Clustering Criterion	Maximum Relative Change in Cluster Seeds	Improvement in Clustering Criterion	Segment Id	Frequency of Cluster	Root-Mean-Square Standard Deviation	Maximum Distance from Cluster Seed	Nearest Cluster	Distance to Nearest Cluster	o3_count	pm10_count	pm25_count	so2_count
0.255185	0.068185	.	1	280	0.285462	1.378112	6	0.962158	129.9643	113.6643	155.4	15.68214
0.255185	0.068185	.	2	311	0.250029	1.117565	1	1.307623	130.8006	45.49196	185.0611	15.63344
0.255185	0.068185	.	3	64	0.378036	1.209495	13	1.181882	85.53125	69.04688	99.78125	83.92188
0.255185	0.068185	.	4	77	0.294523	1.156575	5	1.196433	45.15584	93.51948	157.6364	27.36364
0.255185	0.068185	.	5	286	0.165571	1.033658	11	0.864349	41.99301	45.36364	134.7692	3.541958
0.255185	0.068185	.	6	173	0.28256	1.339125	1	0.962158	128.0925	121.2601	153.3006	46.34104
0.255185	0.068185	.	7	63	0.29442	1.290285	11	1.130057	23.28571	21.74603	153.3968	46.88889
0.255185	0.068185	.	8	129	0.250736	1.52318	16	0.904242	146.8915	116.8605	165.3876	76.28682
0.255185	0.068185	.	9	275	0.210834	1.005119	19	0.789653	182.9273	188.3855	182.6727	15.90909
0.255185	0.068185	.	10	262	0.254546	1.150304	14	0.999976	165.7366	180.2099	177.7481	52.62214
0.255185	0.068185	.	11	88	0.244865	0.969832	5	0.864349	25.28409	33.46591	173.8523	13.19318
0.255185	0.068185	.	12	85	0.22041	1.270969	16	1.027912	117.2471	120.6118	142.0706	119
0.255185	0.068185	.	13	175	0.309839	1.240176	8	0.994328	101.7086	116.9257	142.2	83.91429
0.255185	0.068185	.	14	305	0.263596	1.650509	10	0.999976	153.5049	177.8164	177.9574	83.91148
0.255185	0.068185	.	15	686	0.263706	1.144379	17	1.090829	27.55394	30.24052	46.40525	8.103499
0.255185	0.068185	.	16	118	0.253694	1.381808	8	0.904242	156.0169	116.5	171.8983	104.6102
0.255185	0.068185	.	17	897	0.237102	1.363229	20	0.949553	71.58082	60.82497	70.14716	6.754738
0.255185	0.068185	.	18	236	0.172948	1.330935	5	0.944866	87.15254	23.88136	135.1695	4.118644
0.255185	0.068185	.	19	211	0.272472	1.53193	9	0.789653	151.6493	163.455	167.8436	15.8436
0.255185	0.068185	.	20	202	0.339619	1.163397	17	0.949553	64.42574	48.65842	49.5297	33.62376

Figure 4.3.5: Mean Statistics of Cluster

Figure 4.3.5 shows the mean statistics of each cluster. The frequency of a cluster indicates the number of observations in each cluster. The four variables used which are o3_count, pm10_count, pm25_count and so2_count have different values for each sector.

Variable	Highest Value	Lowest Value
o3_count	Cluster 9 - 182.9273	Cluster 7 - 23.28571
pm10_count	Cluster 9 - 188.3855	Cluster 7 - 21.74603
pm25_count	Cluster 2 - 185.0611	Cluster 15 - 46.40525
so2_count	Cluster 12 - 119	Cluster 5 - 3.541958

Table 4.3.1: Value of Variables

Table 4.3.1 above shows the highest and lowest value for each variable in each cluster. Overall, we can see that pm10_count of cluster 9 has the highest value among the other variables whereas so2_count of cluster 5 has the lowest value. Therefore, conclusion can be made as cluster 9 from pm10_count contribute the most to air pollution while cluster 5 from so2_count contribute the least to air pollution.

4.4 Neural Network

Neural Network is another algorithm used in this assignment. Neural network consists of input layer nodes, hidden layers nodes, and output layer nodes and each node has their associated weight and threshold. Neural networks depend on the training data to learn in order to improve their accuracy through the training process which the results can help in clustering and classifying the data (IBM, 2021).

There are 4 pollutant inputs and 1 target chosen for the model through variable selection as the preparation to train the model with a neural network. This is because a smaller number of important inputs can help in reducing the time required to train the neural network and improve the prediction result. Thus, the Neural Network Architecture chosen is multilayer perceptron (MLP) as it can accept various input, ignore irrelevant inputs than other architectures, has hidden layers and has connection between input layer, hidden layer and output layer. The maximum number of training iterations is set to 50 and after training and running the model, the number of hidden units used is defined as 5 as it gives better performance compared to others.

```
Dual Quasi-Newton Optimization

Dual Broyden - Fletcher - Goldfarb - Shanno Update (DBFGS)

Parameter Estimates            121
```

Figure 4.4.1: Dual Quasi-Newton Optimization

The optimization training technique is set as Default and thus, the technique will be selected based on the number of weights applied during the execution. Based on the Figure 4.4.1 above, the training technique selected to train the neural network is Quasi-Newton. It is selected as the best training technique as it has the lowest Average Error as defined in the model selection criterion. Since the weights applied during the execution is 121, thus, it is understandable that the Quasi-Newton technique is chosen as it can perform better in medium-sized networks with more number of iterations required.

Optimization Start			
Active Constraints	0	Objective Function	0.3057805919
Max Abs Gradient Element	0.0079458623		

Figure 4.4.2: Optimization Start

Optimization Results			
Iterations	50	Function Calls	128
Gradient Calls	58	Active Constraints	0
Objective Function	0.0840552886	Max Abs Gradient Element	0.008986729
Slope of Search Direction	-0.002494196		

Figure 4.4.3: Optimization Result

Optimization is the process of changing the attributes of the neural network such as learning rate and weights to reduce the loss with the use of optimizers to minimise the function (Chauhan, N. S., 2020). One of the performance measure criteria for neural networks is minimising the objective function. The objective function is the sum of total error and penalty function, divided by the total frequency (SAS, n.d.). From the Figure 4.4.2 and Figure 4.4.3 above, we can observe that the training has been done as the objective function after the optimization has decreased from 0.30578 to 0.08406. The number of iterations performed can be also viewed which is 50 as the maximum iteration set initially.

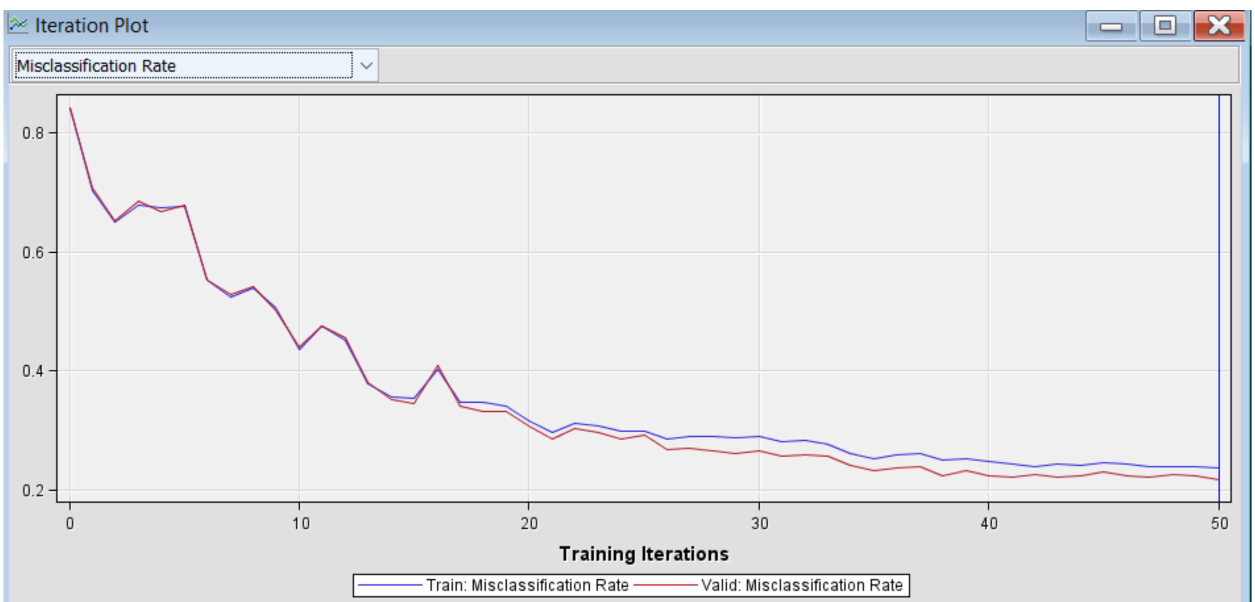


Figure 4.4.4: Iteration Plot of Neural Network based on Misclassification Rate

Iter	Restarts	Function Calls	Active Constraints	Objective Function	Function Change	Gradient Element	Step Size	Search Direction
1	0	8	0	0.25340	0.0524	0.0358	2.000	-0.0573
2	0	11	0	0.24037	0.0130	0.0134	0.0407	-1.182
3	0	13	0	0.22521	0.0152	0.0153	0.159	-0.205
4	0	15	0	0.21580	0.00941	0.0187	0.225	-0.169
5	0	18	0	0.20971	0.00609	0.00925	0.0772	-0.161
6	0	20	0	0.19902	0.0107	0.00442	0.130	-0.117
7	0	22	0	0.18447	0.0146	0.0138	0.259	-0.103
8	0	25	0	0.17590	0.00856	0.0135	0.139	-0.128
9	0	28	0	0.17023	0.00567	0.0168	0.124	-0.0894
10	0	30	0	0.15784	0.0124	0.00547	0.100	-0.190
11	0	32	0	0.15133	0.00651	0.0117	0.525	-0.0591
12	0	34	0	0.14032	0.0110	0.00811	0.177	-0.0922
13	0	36	0	0.13380	0.00652	0.00755	0.453	-0.0607
14	0	38	0	0.13105	0.00275	0.0163	0.389	-0.0420
15	0	40	0	0.12657	0.00449	0.00307	0.340	-0.0202
16	0	42	0	0.12374	0.00282	0.00648	0.610	-0.0184
17	0	44	0	0.12149	0.00225	0.00712	0.552	-0.0128
18	0	46	0	0.11829	0.00320	0.00429	0.579	-0.0097
19	0	49	0	0.11596	0.00233	0.00617	0.677	-0.0063
20	0	51	0	0.11401	0.00195	0.00430	0.502	-0.0116
21	0	53	0	0.11135	0.00266	0.00496	0.584	-0.0083
22	0	55	0	0.10846	0.00289	0.00448	0.585	-0.0114
23	0	57	0	0.10562	0.00284	0.00798	1.349	-0.0054
24	0	60	0	0.10403	0.00159	0.00598	0.345	-0.0083
25	0	62	0	0.10212	0.00191	0.00598	0.368	-0.0108
26	0	65	0	0.10118	0.000934	0.00273	0.146	-0.0097
27	0	67	0	0.09999	0.00120	0.00452	0.444	-0.0053
28	0	70	0	0.09916	0.000828	0.00412	0.271	-0.0052
29	0	72	0	0.09832	0.000837	0.00924	0.383	-0.0054
30	0	75	0	0.09786	0.000462	0.00489	0.225	-0.0041
31	0	79	0	0.09651	0.00135	0.00510	0.579	-0.0048
32	0	82	0	0.09582	0.000691	0.00862	0.272	-0.0051
33	0	84	0	0.09464	0.00118	0.00854	0.512	-0.0034
34	0	86	0	0.09362	0.00102	0.0137	0.554	-0.0053
35	0	88	0	0.09302	0.000607	0.0236	0.336	-0.0076
36	0	92	0	0.09159	0.00142	0.00836	0.479	-0.0063
37	0	95	0	0.09098	0.000620	0.00907	0.213	-0.0067
38	0	97	0	0.09015	0.000822	0.00346	0.326	-0.0048
39	0	99	0	0.08987	0.000281	0.00996	0.656	-0.0031
40	0	103	0	0.08892	0.000951	0.00898	0.467	-0.0041
41	0	106	0	0.08840	0.000523	0.00613	0.218	-0.0046
42	0	108	0	0.08771	0.000685	0.00638	0.485	-0.0027
43	0	110	0	0.08744	0.000275	0.00582	0.994	-0.0017
44	0	114	0	0.08657	0.000868	0.00512	0.691	-0.0026
45	0	116	0	0.08628	0.000294	0.0109	0.484	-0.0045
46	0	118	0	0.08580	0.000473	0.00612	0.239	-0.0031
47	0	120	0	0.08523	0.000575	0.00350	0.299	-0.0040
48	0	122	0	0.08492	0.000307	0.00960	0.556	-0.0026
49	0	124	0	0.08452	0.000397	0.00375	0.197	-0.0039
50	0	126	0	0.08406	0.000469	0.00899	0.398	-0.0025

Figure 4.4.5: Iteration Process Table with the Objective Function

The Figure 4.4.4 above shows the Iteration Plot based on the Misclassification Rate versus optimization iteration and Figure 4.4.5 that shows the Iteration Process Table with the Objective Function. The Misclassification Rate for Training should decrease as the number of iteration increases. From the graph plotted in Figure 4.4.4, we can see that the Misclassification Rate for validation dataset decreases initially and slightly increases at some iteration. This shows that the network is being trained to the random noise components of the training dataset.

On the other hand, the number of iterations plotted on x axis with last value 50 at it is the maximum number of training iteration sets.From the Figure 4.4.4, we can see that a blue vertical line is plotted where training iteration is 50. This shows that the iteration on number 50 has the minimum error function for the validation data set. This can be proved by referring to the Figure 4.4.5 above, it shows that the objective function is decreasing through the iteration process and has the lowest objective function value as 0.08406 in 50th iterations.

Fit Statistics

Target	Target Label	Fit Statistics	Statistics Label	Train	Validation	Test
State		DFT	Total Degrees of Freedom	78768	.	.
State		DFE	Degrees of Freedom for Error	78647	.	.
State		DFM	Model Degrees of Freedom	121	.	.
State		NW	Number of Estimated Weights	121	.	.
State		AIC	Akaike's Information Criterion	7276.671	.	.
State		SBC	Schwarz's Bayesian Criterion	8398.857	.	.
State		ASE	Average Squared Error	0.020043	0.019133	.
State		MAX	Maximum Absolute Error	1	1	.
State		DIV	Divisor for ASE	83691	21165	.
State		NOBS	Sum of Frequencies	4923	1245	.
State		RASE	Root Average Squared Error	0.141572	0.138324	.
State		SSE	Sum of Squared Errors	1677.38	404.9593	.
State		SUMW	Sum of Case Weights Times Freq	83691	21165	.
State		FPE	Final Prediction Error	0.020104	.	.
State		MSE	Mean Squared Error	0.020073	0.019133	.
State		RFPE	Root Final Prediction Error	0.141789	.	.
State		RMSE	Root Mean Squared Error	0.141681	0.138324	.
State		AVERR	Average Error Function	0.084055	0.082097	.
State		ERR	Error Function	7034.671	1737.583	.
State		MISC	Misclassification Rate	0.235629	0.216064	.
State		WRONG	Number of Wrong Classifications	1160	269	.

Figure 4.4.6: Fit Statistics for Neural Network

From the Figure 4.4.6 above that shows the Fit Statistics, we can see that the number of estimated weights is 121 which shows that the model used for training is in medium sized as mentioned above. Too large of a model used will result in long training time and less accurate results. Suitable model weights can help in ensuring better performance of the model with only important variables chosen. Besides, the Misclassification Rate for the train and validate model are 0.23563 and 0.21606 respectively. The lower misclassification rate shows better performance mode.

Table 4.4.1: Event Classification Table

Train Set				**Validation Set**			
Data Role=TRAIN Target=State Target Label=' '				Data Role=VALIDATE Target=State Target Label=' '			
False Negative	True Negative	False Positive	True Positive	False Negative	True Negative	False Positive	True Positive
98	4336	98	391	22	1104	18	101

From the Table 4.4.1 above, we can see the exact number of false negative, true negative, false positive and true positive predicted by the neural network. This also describes the predicted number of successes compared with the number of successes actually observed.

4.5 Model Comparisons

From the earlier sections, there are four models created with four nodes in SAS Enterprise Miner. The models are Decision Tree, Regression, Neural Network and Cluster. They can contribute to decision-making by the related stakeholders or organisations regarding air pollution. These models show various results that strive to solve similar objectives. Therefore, it would be wise to compare the models used to find the best model out of the four used.

In SAS Enterprise Miner, the data mining process applies Sample, Explore, Modify, Model and Assess (SEMMA). It has a useful function node that can compare the models. The function is known as the Model Comparison node under the Assess category. This node can review and compare the performance of the connected models with data mining measures for this project. (SAS Help Center, n.d.). The Model Comparison Node enables users to evaluate the performance of various models by generating resulting tables and graphs.

As mentioned in the Model Diagram and Explanation section, the four nodes are Decision Tree, Regression, Neural Network and Cluster. All these nodes are connected to the Model Comparison node to run the comparison analysis. To set the comparison in this project, the model selection grid selection statistic is set to Default and the selection table is set to validation. According to SAS Help Center (n.d.), validation data is chosen as the model selection when it is available. Then, run the node to observe the results.

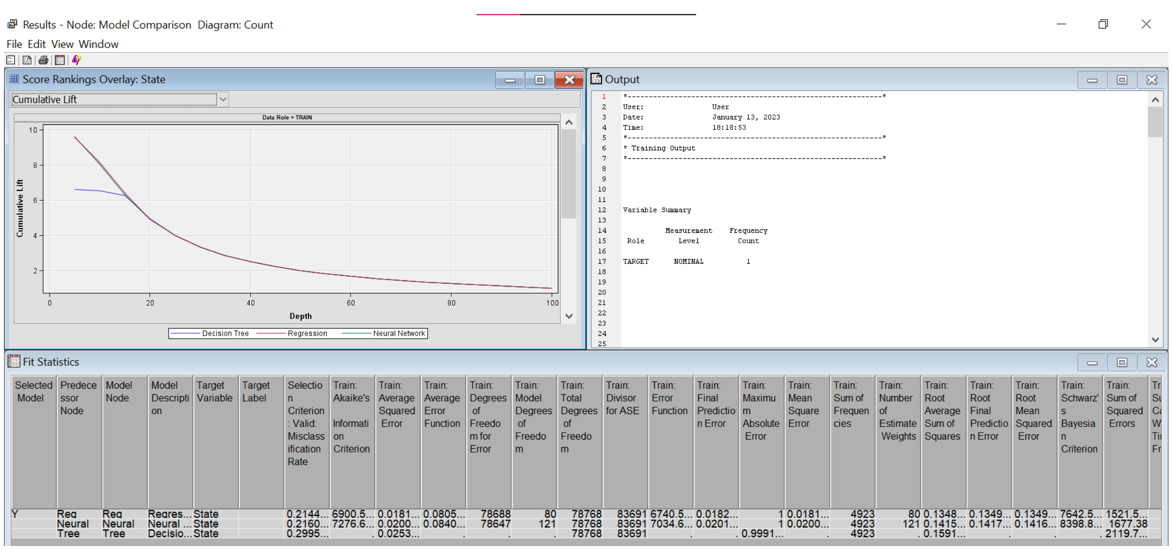


Figure 4.5.1: Result of Model Comparison Node

Figure 4.5.1 above shows the result that appeared after running the Model Comparison Node. The result shows three windows of comparison which are the Fit Statistics Table, Score Rankings Overlay Charts by State and Output of all the windows. The details of each window are analysed below.

Fit Statistics

Selected Model	Predecessor Node	Model Node	Model Description	Target Variable	Target Label	Selection Criterion: Valid: Misclassification Rate	Train: Akaike's Information Criterion	Train: Average Squared Error	Train: Average Error Function	Train: Degrees of Freedom for Error	Train: Model Degrees of Freedom	Train: Total Degrees of Freedom	Train: Divisor for ASE	Train: Error Function	Train: Final Prediction Error	Train: Maximum Absolute Error	Train: Mean Square Error	Train: Sum of Frequencies	Train: Number of Estimate Weights	Train: Root Average Sum of Squares	Train: Root Final Prediction Error	Train: Root Mean Squared Error	Train: Schwarz's Bayesian Criterion	Train: Sum of Squared Errors	Tr Su Ca W Ti Fr
Y	Reg	Reg	Regres...	State		0.2144...	6900.5...	0.0181...	0.0805...	78688	80	78768	83691	6740.5...	0.0182...	1	0.0181...	4923	80	0.1348...	0.1349...	0.1349...	7642.5...	1521.5...	
	Neural	Neural	Neural ...	State		0.2160...	7276.6...	0.0200...	0.0840...	78647	121	78768	83691	7034.6...	0.0201...	1	0.0200...	4923	121	0.1415...	0.1417...	0.1416...	8398.8...	1677.38	
	Tree	Tree	Decisio...	State		0.2995...	.	0.0253...	.	.	.	78768	83691	.	.	0.9991...	.	4923	.	0.1591...	.	.	.	2119.7...	

Figure 4.5.2: Fit Statistics Table Window

Next, Figure 4.5.2 shows the Fit Statistics Table Window from the Model Comparison results earlier. It consists of various statistical measure values for the Regression, Neural Network and Decision Tree to do model comparisons. From this table under selection statistics, it can be seen that the Selection Criterion in the table above is labelled at the Valid Misclassification Rate. Hence, this project uses Misclassification Rate to determine the accuracy of the models listed previously and choose the best model. The reason is according to SAS Help Center (n.d.), when Selection Statistics is Default, since the target (State) is categorical and there is no profit/loss matrix, it will use the Misclassification Rate.

Fit Statistics
Model Selection based on Valid: Misclassification Rate (_VMISC_)

Selected Model	Model Node	Model Description	Valid: Misclassification Rate	Train: Average Squared Error	Train: Misclassification Rate	Valid: Average Squared Error
Y	Reg	Regression	0.21446	0.018181	0.21207	0.017982
	Neural	Neural Network	0.21606	0.020043	0.23563	0.019133
	Tree	Decision Tree	0.29960	0.025328	0.31099	0.024946

Figure 4.5.3: Output for Fit Statistics Table on Model Selection

Referring above, Figure 4.5.3 shows the output for Fit Statistics Table on the model selection. It lists out the details of the Average Squared Error and Misclassification Rate of both the valid and train dataset table for the respective model nodes connected to the comparison node, which are the Regression, Neural Network and Decision Tree. Since it was mentioned earlier that the comparison criteria selected by SAS Enterprise Miner is the Valid Misclassification Rate, that will be the one we use to compare the models.

According to SAS Help Center (n.d.), Misclassification Rate is a statistical model in which the smallest Misclassification Rate value indicates the best model. The data is viewed as valid instead of the train as the model selection table is set as validation for this node. Thus, for the result, we will see the values in ascending order for the Valid Misclassification Rate criterion. From Figure 4.5.3, when the output result is arranged in ascending order from most accurate model to least, it will be Regression, Neural Network and Decision Tree. Therefore, **Regression** is considered the best model, it is the **smallest** for the **Valid Misclassification Rate** criterion.

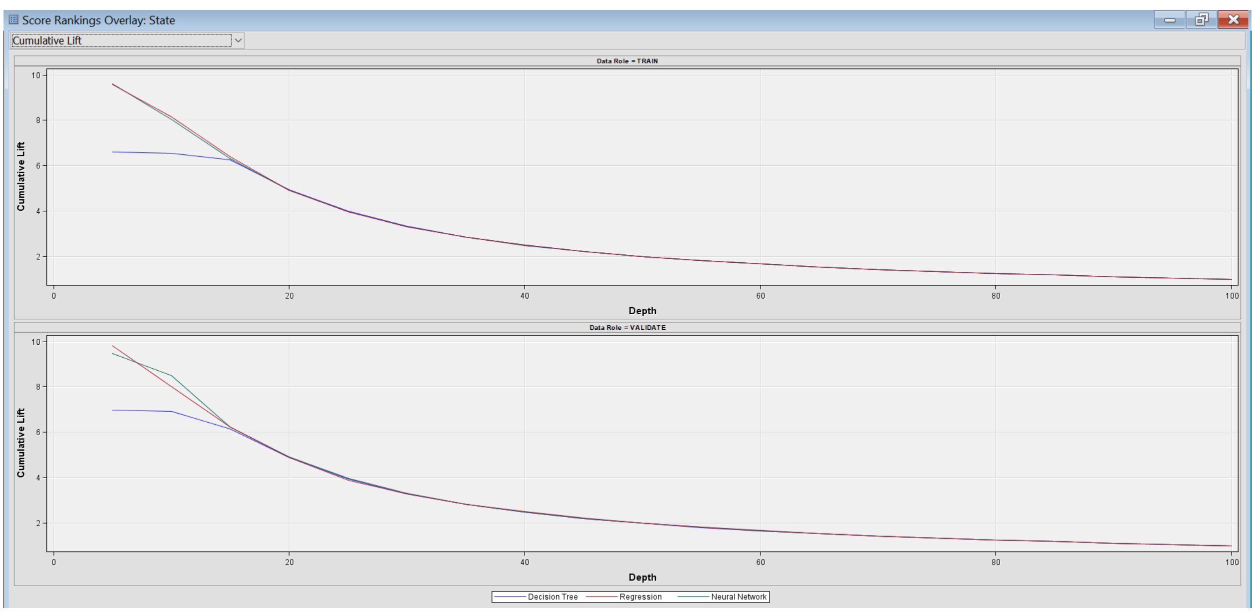


Figure 4.5.4: Score Rankings Overlay by State, Cumulative Lift Charts

Figure 4.5.4 above shows Cumulative Lift line charts for the Score Rankings Overlay. SAS Help Center (n.d.) has defined cumulative lift as “the cumulative ratio of % Captured Responses within each decile to the baseline % Response” and that the best model is seen as the **greatest value**. From Figure 4.5.4, **Regression** (red line) seems to have a **higher cumulative lift**

value than the other models at the very beginning by 20 percent of respondents. As it is seen that most of the lines are very close in the charts of Figure 4.5.4 after 20 percent of respondents, it is difficult to differentiate the cumulative lifts of the models. Hence, the result can be further confirmed by toggling the view to the table that plots the charts.

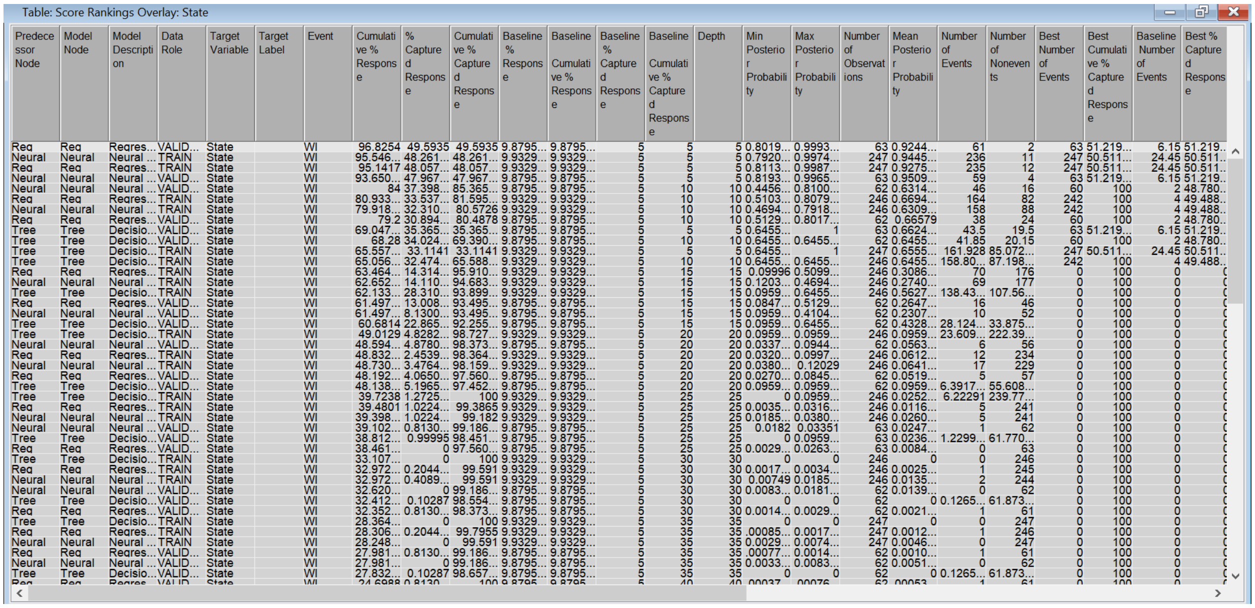


Figure 4.5.5: Score Rankings Overlay by State, Table (Sorted by Cumulative Lift) 1

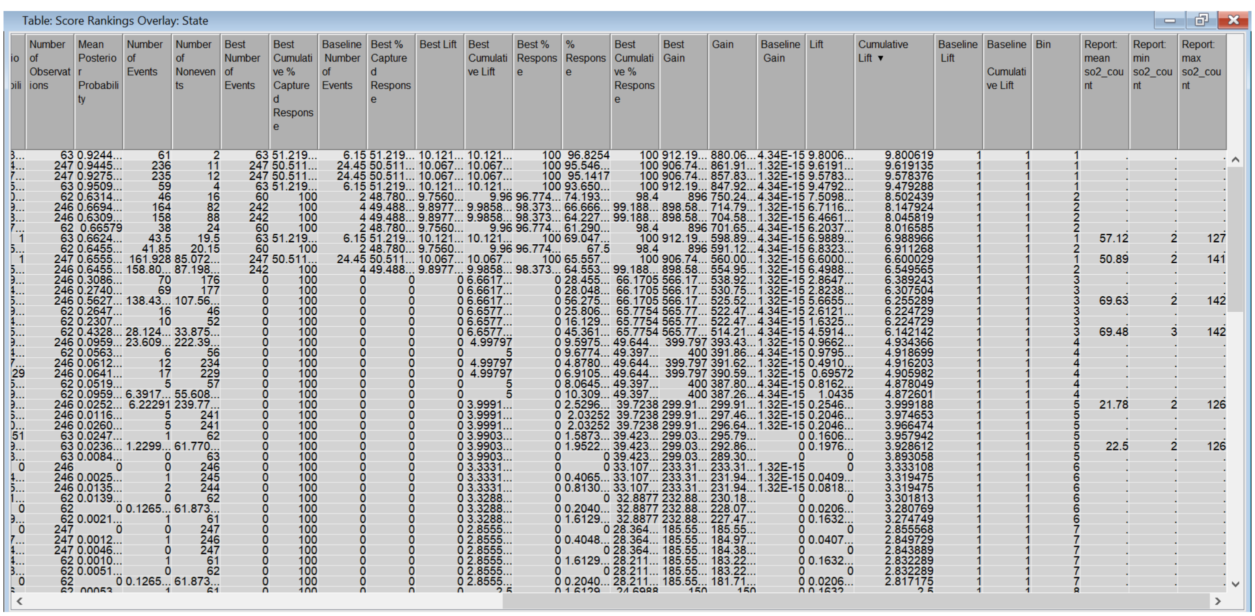


Figure 4.5.6: Score Rankings Overlay by State, Table (Sorted by Cumulative Lift) 2

Figure 4.5.5 and Figure 4.5.6 above shows the table of Score Rankings Overlay by State that has been sorted with Cumulative Lift in descending order (highest to lowest). The screenshots are placed in two figures as the table view is too long and needs to scroll to view. From the table, the highest valid cumulative lift is 9.800619 from Regression, which is the same as the graph. By descending order, it will be the Regression, Neural Network and Decision Tree. Therefore, **Regression** is the best model among the four models created as it has the **greatest Cumulative Lift** result.

The model comparison will be based on which model has multiple top criteria. In this project and based on earlier analysis within this section, **Regression** has the most criteria achieved, which shall be assumed as the best model. In this project, the regression is set as **Logistic Regression**.

Although there is another algorithm used, which is the cluster node, it does not appear in the Model Comparison node result. It was later discovered that in SAS Enterprise Miner, the Cluster node is a function under the Explore category based on SAS Data Mining SEMMA. The limitation faced is that the Model Comparison node only includes the algorithms under the Model category of SAS Data Mining SEMMA. Thus, Cluster is excluded in the model comparison as we do not have the factual support of analysis figures to compare it with other models.

5. Conclusion and Future Work

In conclusion, through this project, we have analysed the data to have an understanding of the major pollutants and gas emissions. Based on the variable selection node result, PM2.5 is the top pollutant which will affect our model result significantly. For another objective of this assignment, the models can classify and predict whether a state is classified as the state with high pollution based on the pollutants and gas emissions of the states such as PM2.5, PM10, NO2, O3, CO and SO2, with different misclassification rate. In short, the objectives are achieved through the identification and implementation of the highest accuracy machine learning models by using SAS Enterprise Miner.

At the beginning of our project, data preprocessing is done before the model implementation to remove all the outliers and missing values in the dataset. SAS Enterprise Guide (SAS EG) is used to carry out this process. The features in SAS EG which are 'Filter and Sort' and 'Query Builder' are implemented. 'Filter and Sort' feature is used to remove outliers and missing values in the dataset and sort them either in ascending or descending order while the 'Query Builder' feature is used for data reduction to reduce the excessive amount of data and remain the suitable data according to the business problems identified. For this dataset used, we only retain the count of each pollutant such as O3, NO2, SO2, PM10, PM25 and CO. All the other variables like minimum, maximum and median of the pollutants are not in use.

After data preprocessing, the data is loaded into the SAS Enterprise Miner (SAS EM) for the following data mining process. The data then undergoes a data partition process to allocate the data into 80% training and 20% validation. Training is for preliminary model fitting while validation is to test the appropriateness of the model selected. Then, it continues with variable selection in which only O3_count, SO2_count, PM10_count and PM25_count are used for the following algorithms. This is because variable selection rejects CO_count and NO2_count as they have a R-square value of less than 0.05.

There are a total of 4 algorithms used in this dataset to train the model to achieve our objective and solve the problem statement in our project. We have set the 'state' variable as the target and the 'O3_count', 'SO2_count', 'PM10_count' and 'PM25_count' variables as the input. This is to analyse which state has the highest number of pollutants which contribute the most to air pollution. The first algorithm used is the decision tree. This algorithm achieves high accuracy with 94.31% for the training dataset and 94.86% accuracy for the validation dataset. This percentage indicates how the decision tree algorithm accurately classified the state with a different pollutant count.

Besides, the second algorithm used is logistic regression. The logistic regression produces a misclassification rate of 0.24% which means the model is not overfitting or underfitting as the difference is small and not significant.

Furthermore, the third algorithm used is clustering. The dataset has been clustered into 20 clusters with 'O3_count' having the higher variable importance compared to the other variables. The higher the importance, the more accurate the clustering is and thus, the closer the model represents reality. From the mean statistics result for clustering, we can conclude that cluster 9 from pm10_count contribute the most to air pollution while cluster 5 from so2_count contributes the least to air pollution.

Finally, the last algorithm used is the neural network. The architecture chosen is the multilayer perceptron (MLP) as it can accept various inputs (O3_count, SO2_count, PM10_count and PM25_count) and ignore irrelevant inputs. The misclassification rate of the Neural Network is considered low which shows that the model has high accuracy as the lower the misclassification rate, the higher the model accuracy.

After implementing all the algorithms proposed previously, the model comparison is carried out to compare and evaluate the performance of the models. This is because the results generated by the four different algorithms are almost the same. In model comparisons, the two variables 'Misclassification Rate' and 'Cumulative Lift' are used to evaluate the models. The best model is evaluated with the criteria of lowest misclassification rate and highest cumulative lift value. For both 'Misclassification Rate' and 'Cumulative Lift', the evaluated best model is Logistic Regression. However, clustering is not included in the model comparisons as it is an unsupervised learning model and SAS software does not include the clustering in the result of the model comparison node.

In this assignment, we used 'State' as the target variables and pollutant count as the input variables. Therefore, in the future, the model will be further trained for different target and input variables. For example, the target variable will change from state to county or city. The purpose of doing this is to determine which county or city has higher pollution and the authorities can take further action in that county or city to reduce the effect of air pollution. Besides, the input variables can also be changed from count to mean or median to identify whether it will produce the same result as count. In addition, we will implement more models as for now we have only three models for model comparison and three are not enough to find the most accurate model. For instance, we will implement models like the Auto Neural model to find the optimal configuration for the neural network model, Ensemble model to create a new model by taking a function of posterior probabilities (for class targets) or the predicted values (for interval targets) from multiple models and Rule Induction model to build classification models to improve the classification of rare events in the target variable. By implementing more models, we can accurately predict the results and achieve the objective proposed.

6. References

Bhattacharyya, M. (2022). *DEAP: Deciphering Environmental Air Pollution*. https://www.kaggle.com/datasets/mayukh18/deap-deciphering-environmental-air-pollution

Brownlee, J. (July 14, 2017). What is the Difference Between Test and Validation Datasets? Retrieved from: https://machinelearningmastery.com/difference-test-validation-datasets/

Campbell-Lendrum, D., & Prüss-Ustün, A. (2019). Climate change, air pollution and noncommunicable diseases. *Bulletin of the World Health Organization*, *97*(2), 160. https://doi.org/10.2471/BLT.18.224295

Chauhan, N. S. (2020). Optimization algorithms in neural networks. KD Nuggets. Retrieved from https://www.kdnuggets.com/2020/12/optimization-algorithms-neural-networks.html#:~:text=The%20process%20of%20minimizing%20

Data Mining - decision tree induction. Tutorials Point. (n.d.). Retrieved January 11, 2023, from https://www.tutorialspoint.com/data_mining/dm_dti.htm#

exporium. (2022, Dec 25). Clustering - When you should use it and avoid it. requirements. *upGrad.* Retrieved from https://www.explorium.ai/blog/clustering-when-you-should-use-it-and-avoid-it/

Ghoson, A. M. (2011). Decision tree induction & clustering techniques in SAS enterprise miner, SPSS Clementine, and IBM intelligent miner A comparative analysis. *International Journal of Management & Information Systems (IJMIS)*, *14*(3). https://doi.org/10.19030/ijmis.v14i3.841

IBM. (2021). What are neural networks? Retrieved from https://www.ibm.com/topics/neural-networks

Kinney, P. L. (2018). Interactions of climate change, air pollution, and human health. *Current environmental health reports*, *5*(1), 179-186. https://doi.org/10.1007/s40572-018-0188-x

Samarth Agrawal. (May 17, 2021). How to split data into three sets(train,validation, and test) And why? Retrieved from: https://towardsdatascience.com/how-to-split-data-into-three-sets-train-validation-and-test-and-why-e50d22d3e54c

SAS. (2017, Aug 30). Cubic clustering criterion. Retrieved form https://documentation.sas.com/doc/en/emref/14.3/n1dm4owbc3ka5jn11yjkod7ov1va.htm

#:~:text=The%20cubic%20clustering%20criterion%20(CCC,evaluated%20by%20Monte%20Carlo%20methods.

SAS. (2017, Sep). SAS/STAT 14.3 User's guide: The cluster procedure. Retrieved form https://support.sas.com/documentation/onlinedoc/stat/143/cluster.pdf

Sharma, R. (2022, Aug 31). Cluster analysis in data mining: Applications, methods & requirements. *upGrad.* Retrieved from https://www.upgrad.com/blog/cluster-analysis-data-mining/#:~:text=and%20K%2Dmedoids%3F-,What%20is%20Clustering%20in%20Data%20Mining%3F,the%20similarity%20of%20the%20data.

What's new in SAS enterprise miner 5.2. (2006, April 21). Retrieved January 12, 2023, from https://support.sas.com/documentation/whatsnew/91x/emgui52whatsnew900.htm

World Health Organization. (2019). *What is Air Pollution?* https://cdn.who.int/media/docs/default-source/searo/wsh-och-searo/what-is-air-pollution-2019.pdf?sfvrsn=6dcc13ee_2