Forecasting Air Quality based on Traffic and Weather Data

BRETHE EASY GROUP

INTRODUCTION

Allergic rhinitis
Hay fever

Backgrond

Before heading out, it's common to check:

- Weather
- Traffic congestion
- Air quality

Flowers pollen

Could we predict future air quality to provide timely reminders for travelers, like weather forecasts?

PM (Particulate matter)

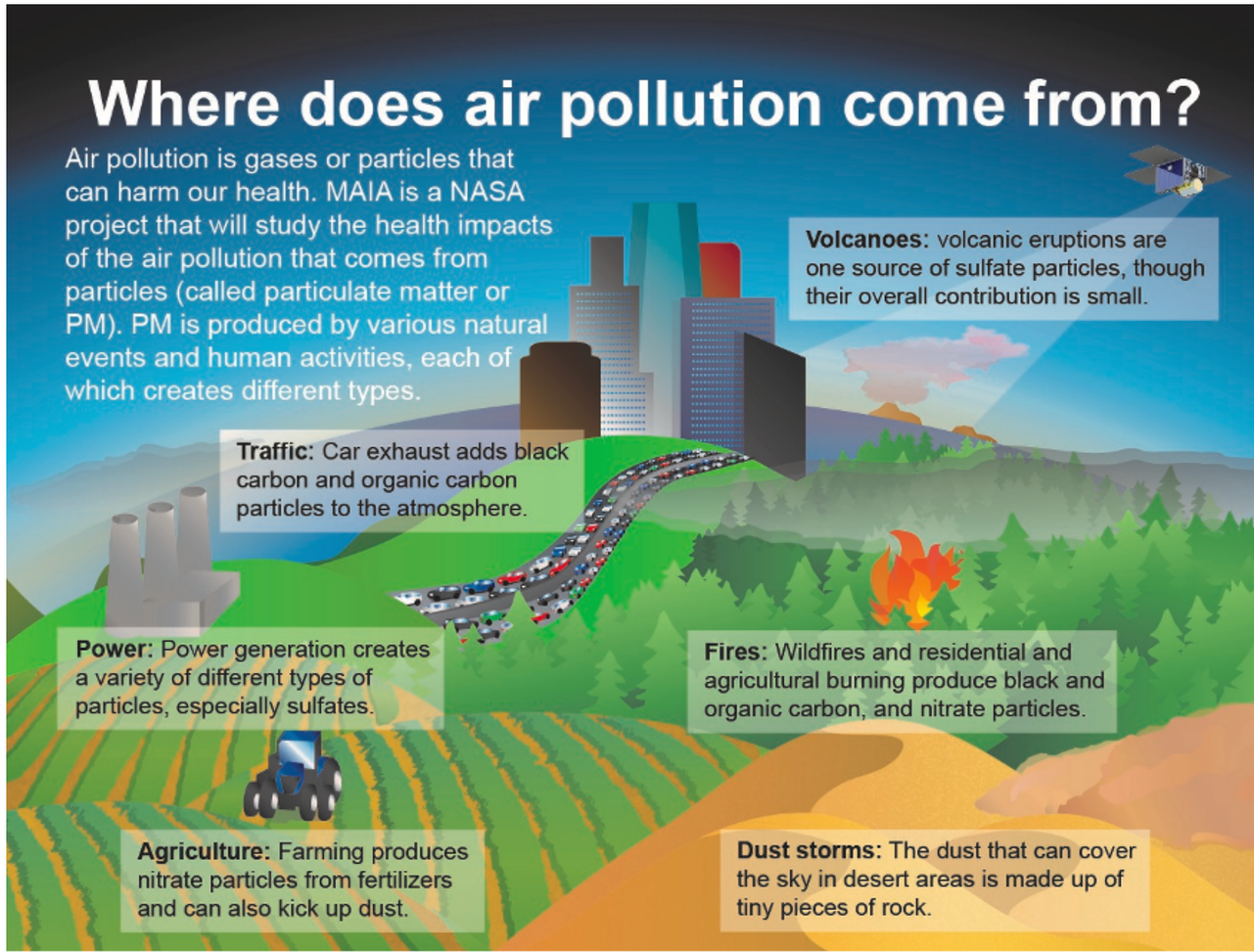
A major component of air pollution

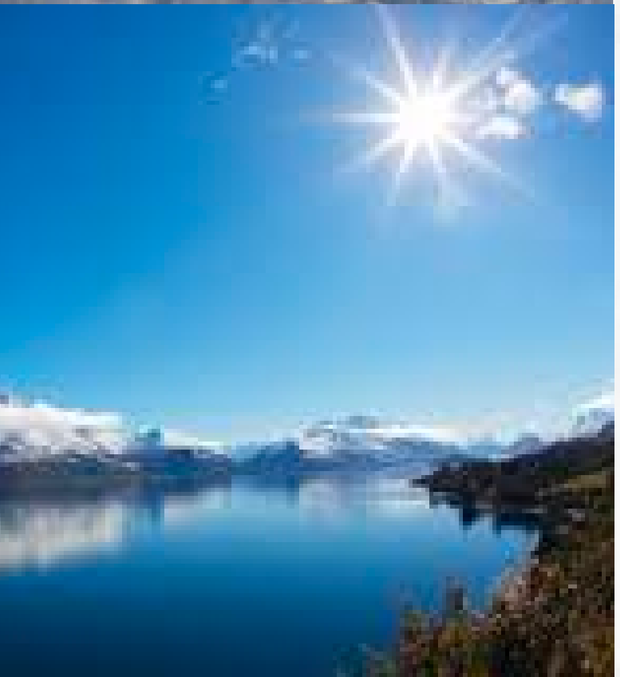
- PM2.5
- road dust, pollen

WEATHER CHANGES

TRAFFIC

AIR QUALITY





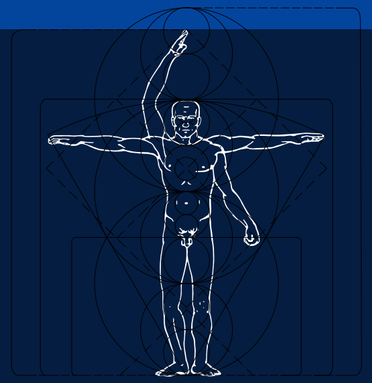
2-3%

28.7%

Our research in Auckland revealed a 2-3% increase in PM levels over a year, exceeding the World Health Organization's safety standards. Even in areas with relatively good air quality, like Queenstown, PM levels were 28.7% above the average, affecting people's breathing health.

AIR QUALITY PREDICTION

Collecting and analyzing data on traffic, weather, and more



Build models that can alert people to changes in air quality in advance

By providing timely and accurate air quality information, we aim to foster healthier, more sustainable urban living

Our Target Audience

Academia

Government

The Public

New Zealand's Ministry for the Environment

FRAMEWORK & TASKS

Alvin

Collecting, processing, and analyzing data

Leslie

Builds 3 **DEMO** version models

Morgan

Builds 3 **PRO** version models

Bella

Develops our Air Quality Forecast website

Three iconic cities :

1. Auckland
2. Wellington
3. Christchurch

DATA

Data Collection

Data in traffic, weather, and air quality

Time range: From the beginning of 2020 to the end of 2023 (except Auckland)

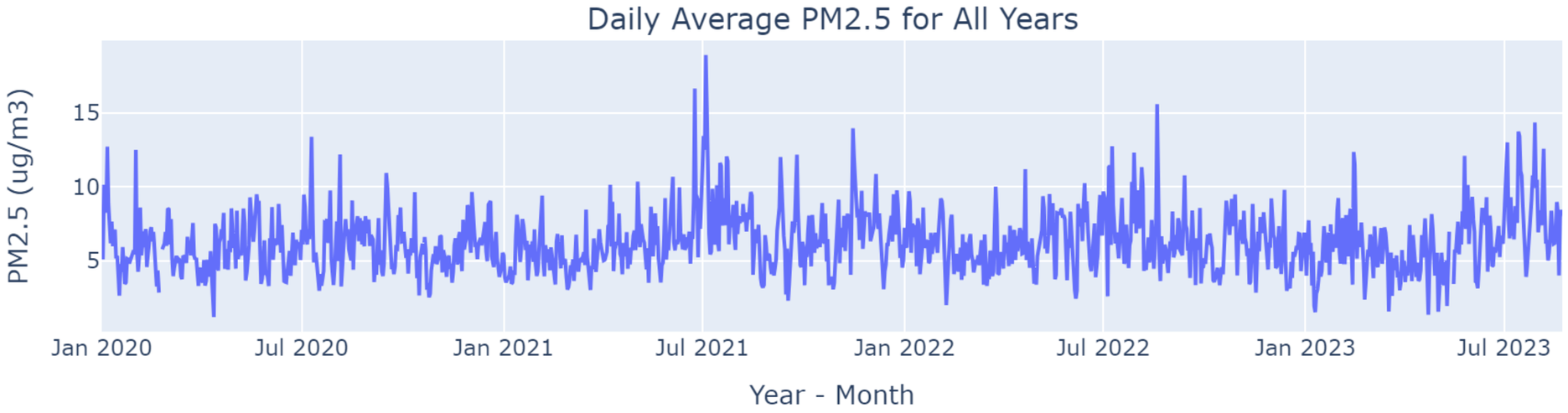
Regional range: Auckland, Wellington, Christchurch

Data source:

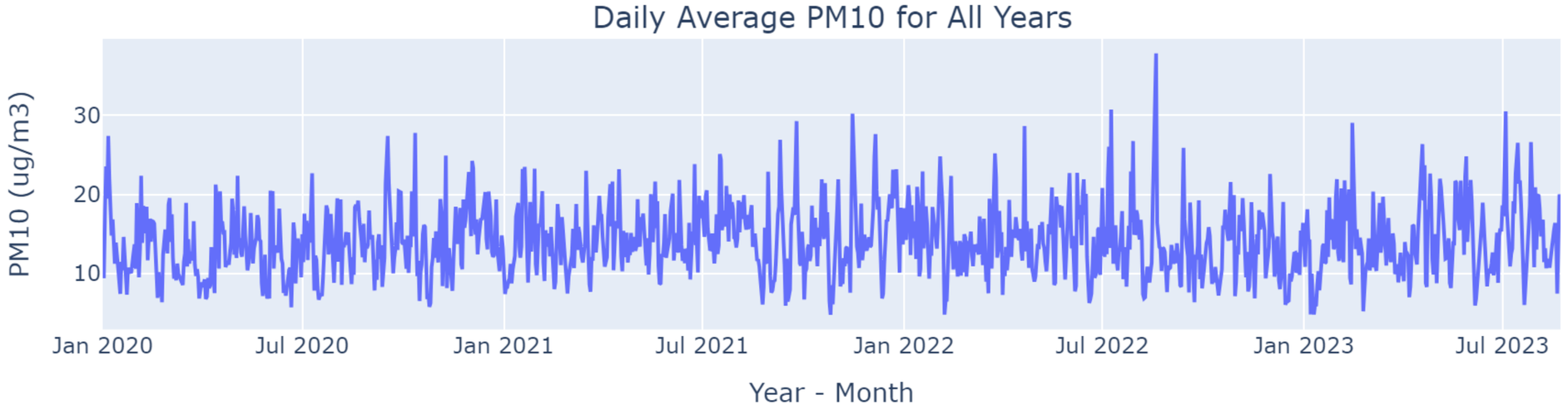
- New Zealand Transport Agency (NZTA)
- National Institute of Water and Atmospheric Research (NIWA)
- Auckland Council
- Greater Wellington Regional Council (GWRC)
- Environment Canterbury (Canterbury Regional Council)

EDA for Auckland

PM2.5

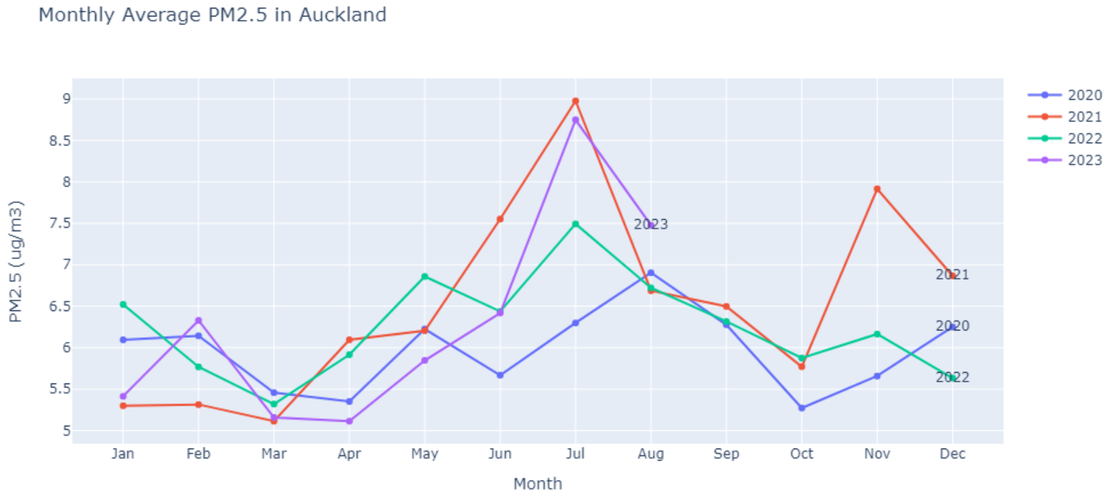


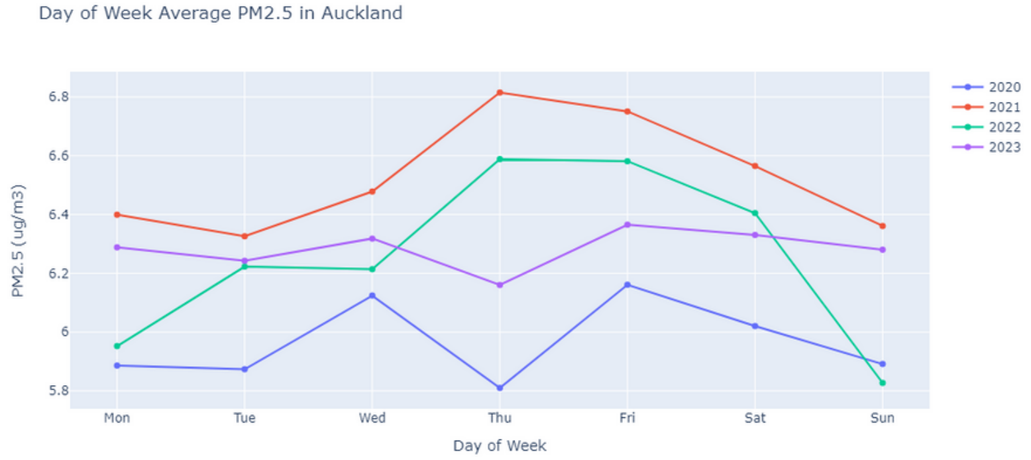
PM10



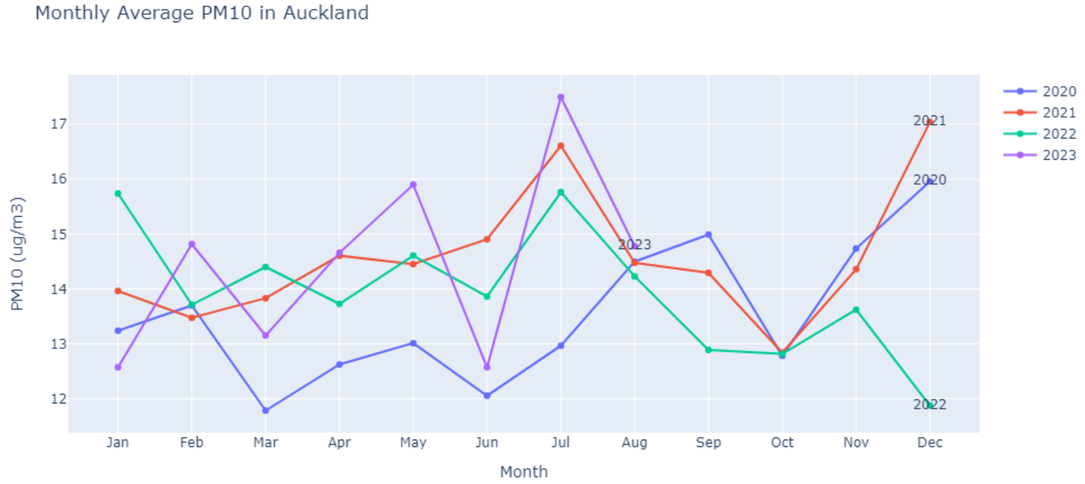
EDA for Auckland

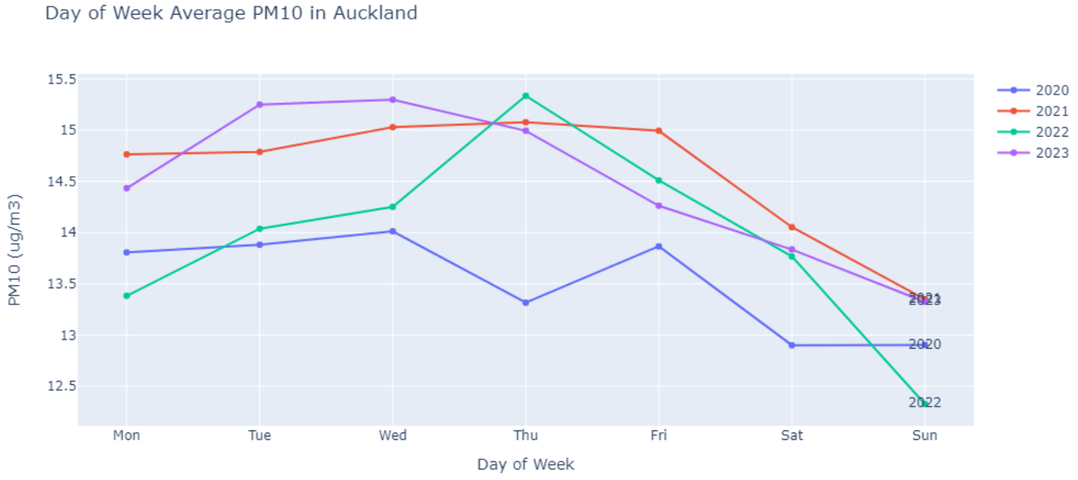
PM2.5





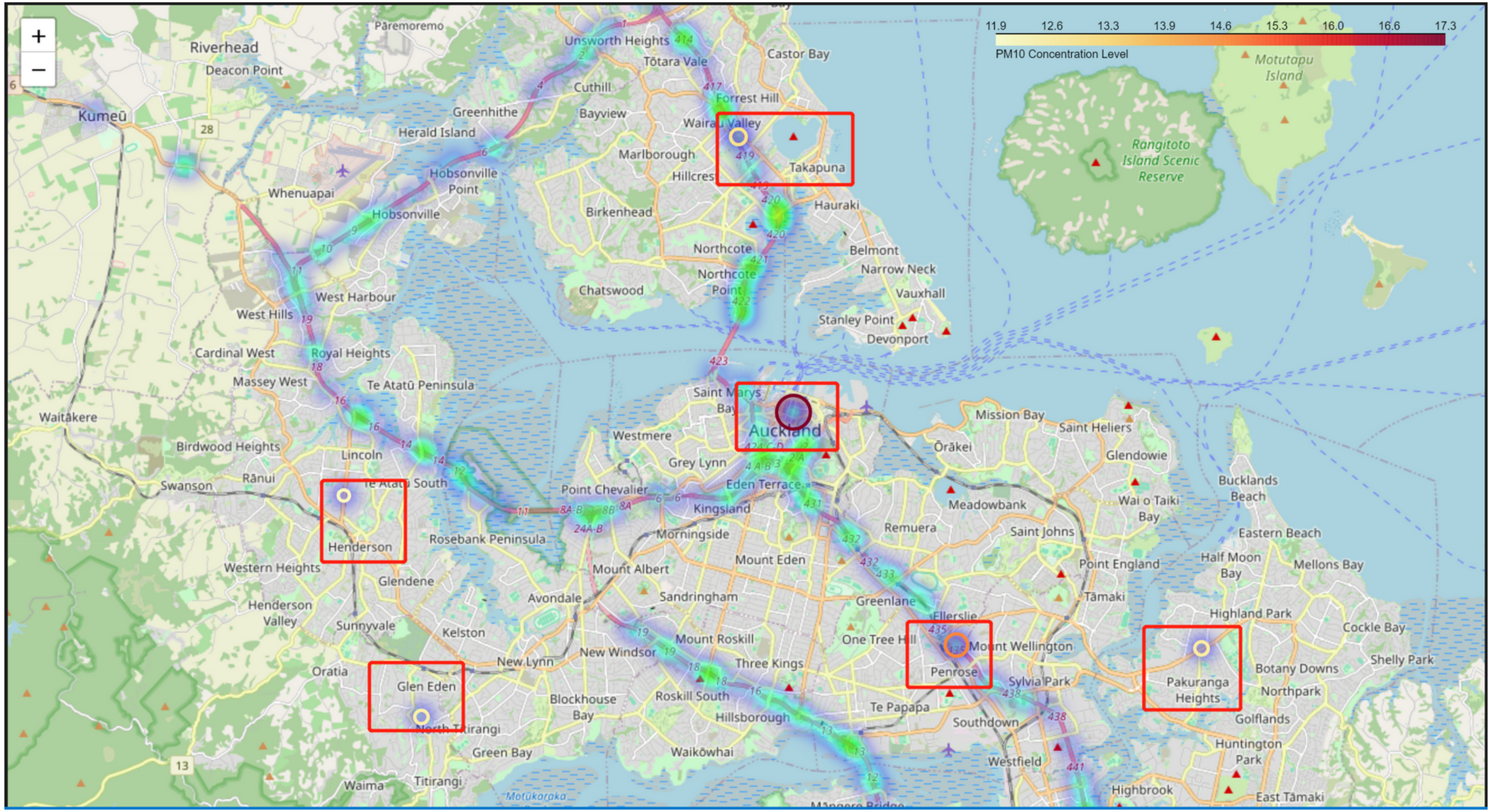
PM10





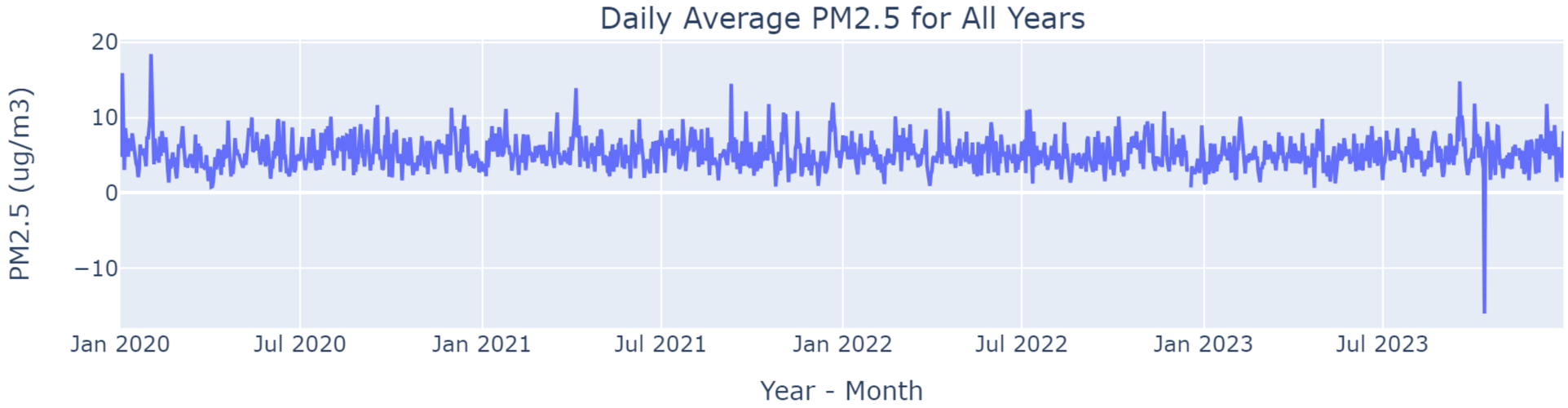
EDA for Auckland: Geo-Spacial Analysis

Average AADT and Average PM10

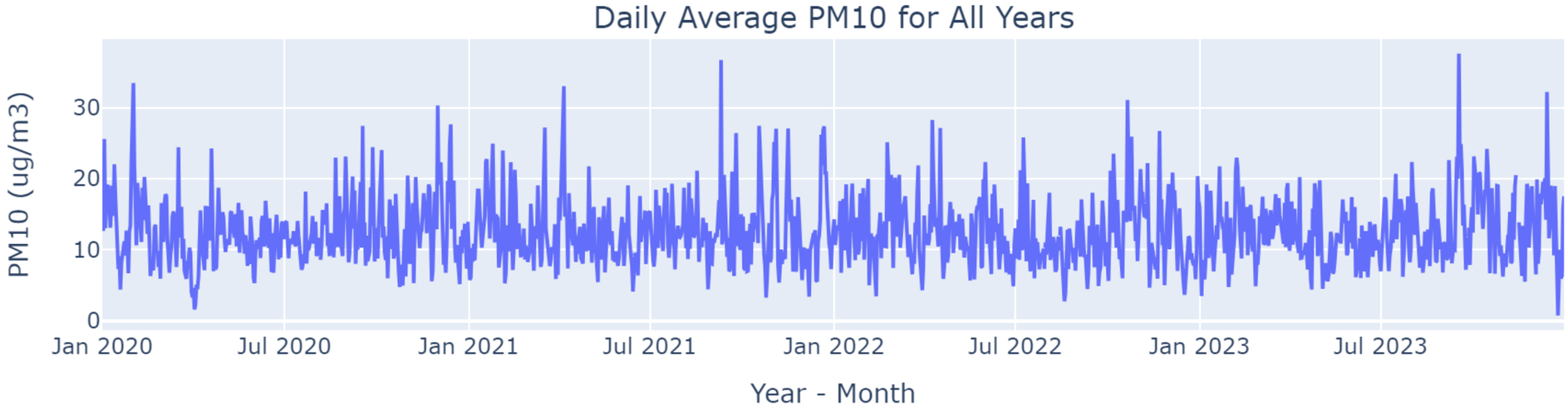


EDA for Wellington

PM2.5

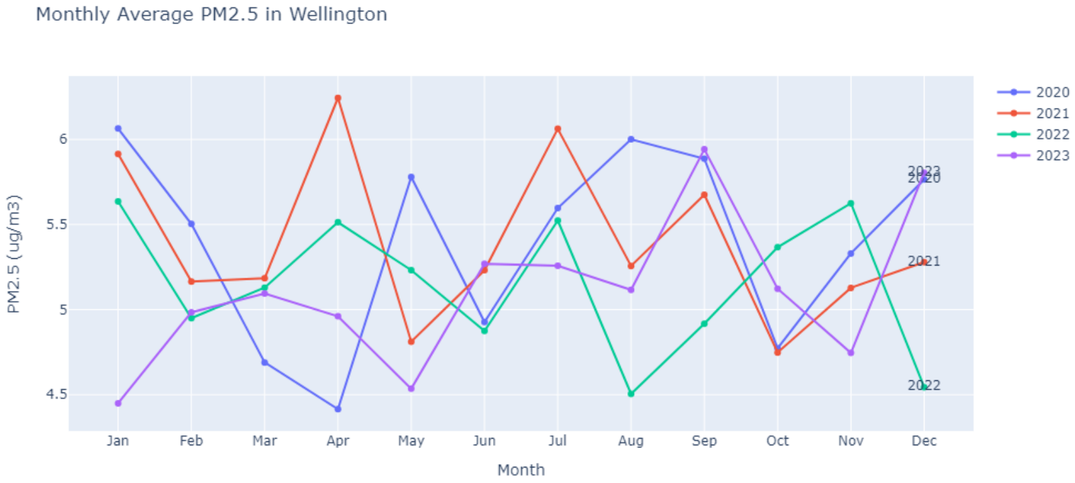


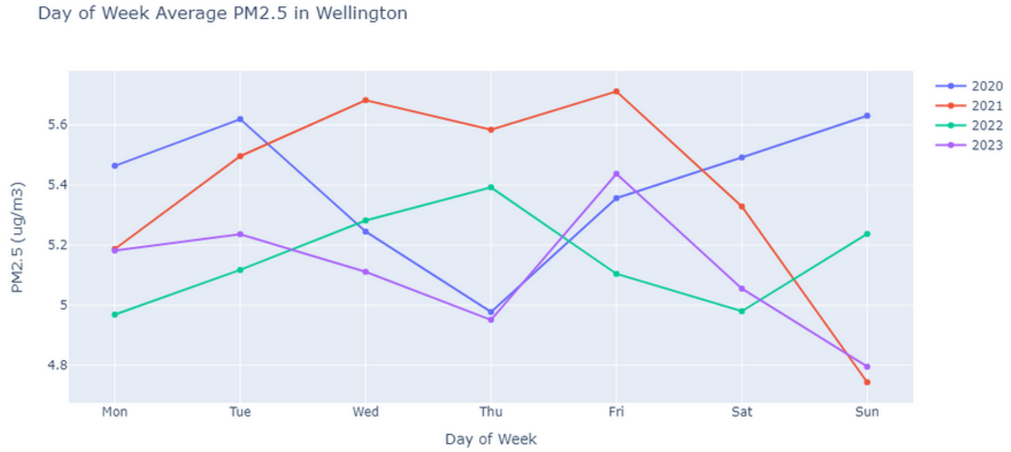
PM10



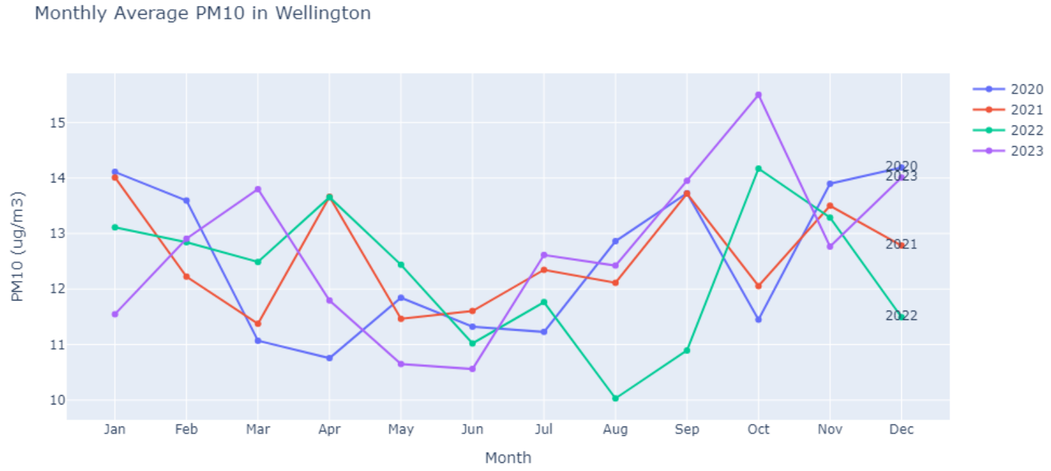
EDA for Wellington

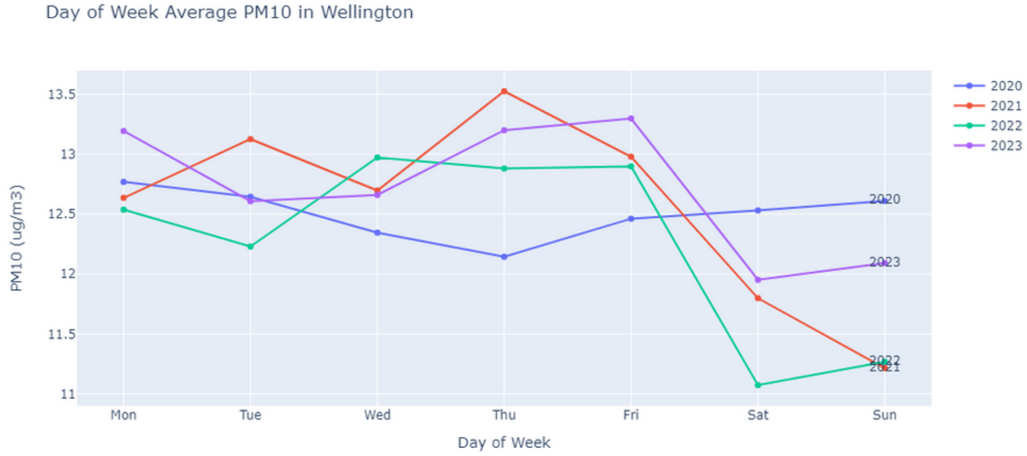
PM2.5





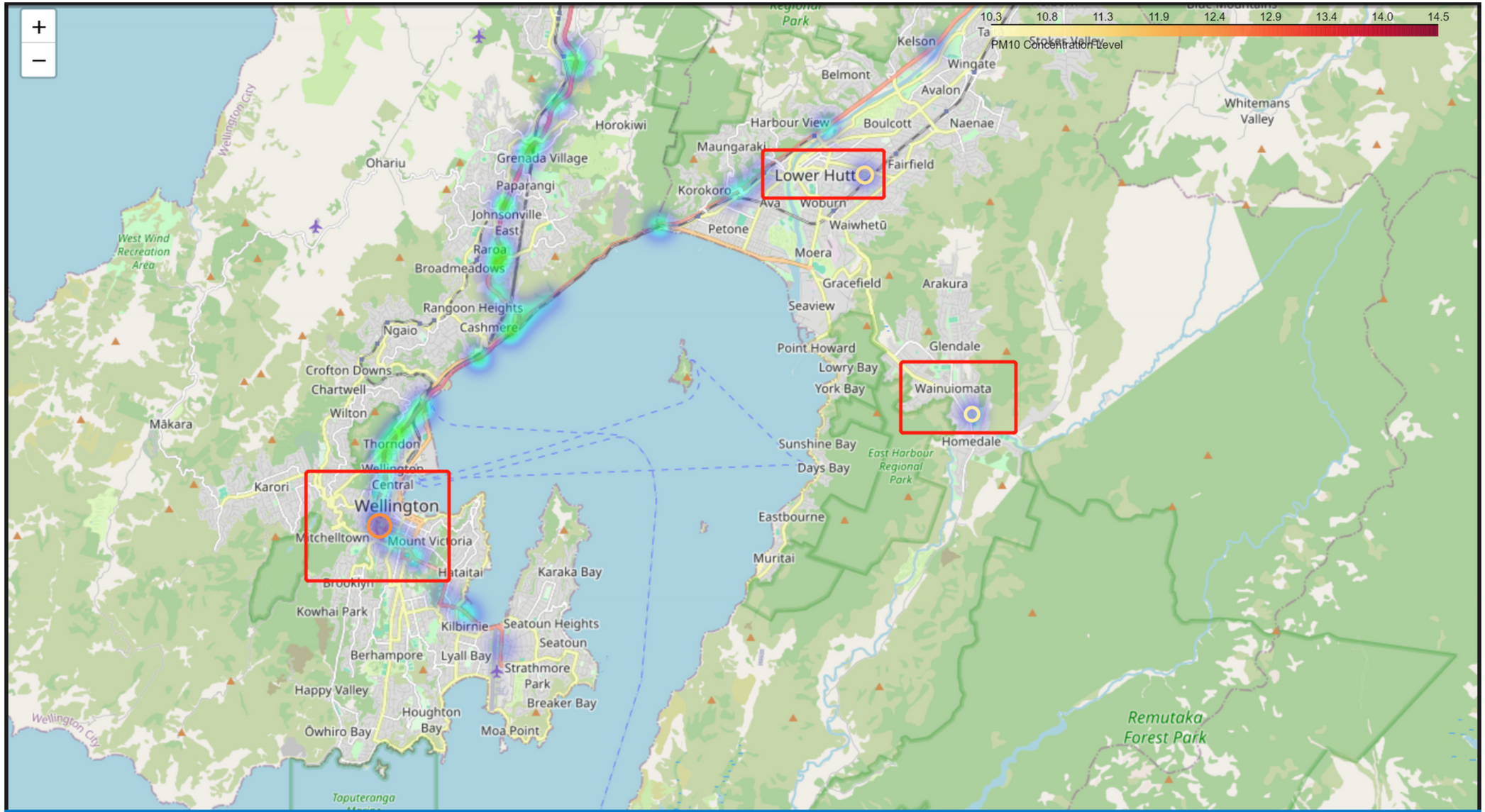
PM10





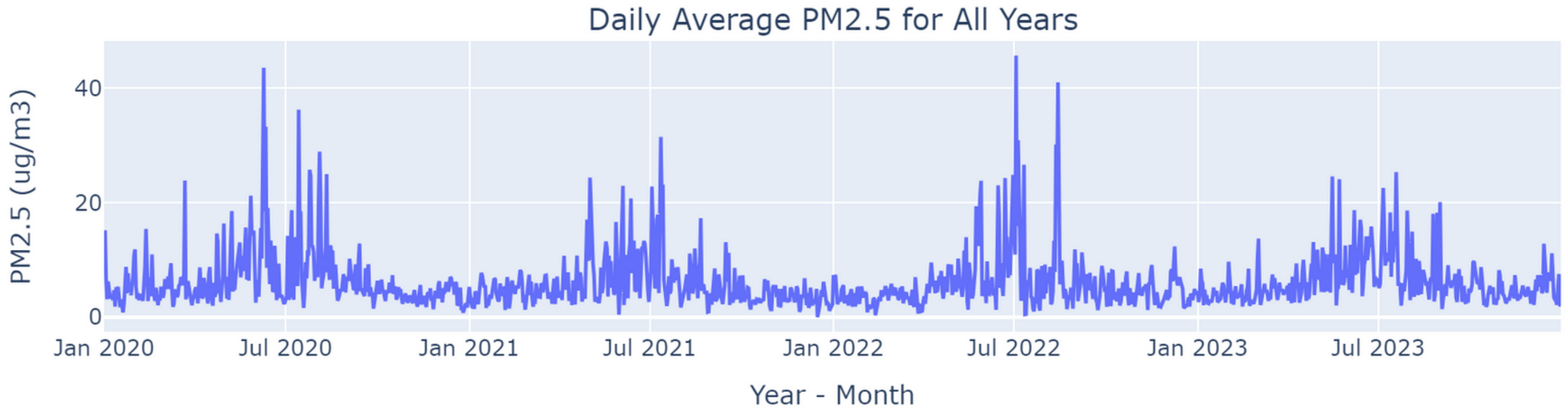
EDA for Wellington: Geo-Spacial Analysis

Average AADT and Average PM10

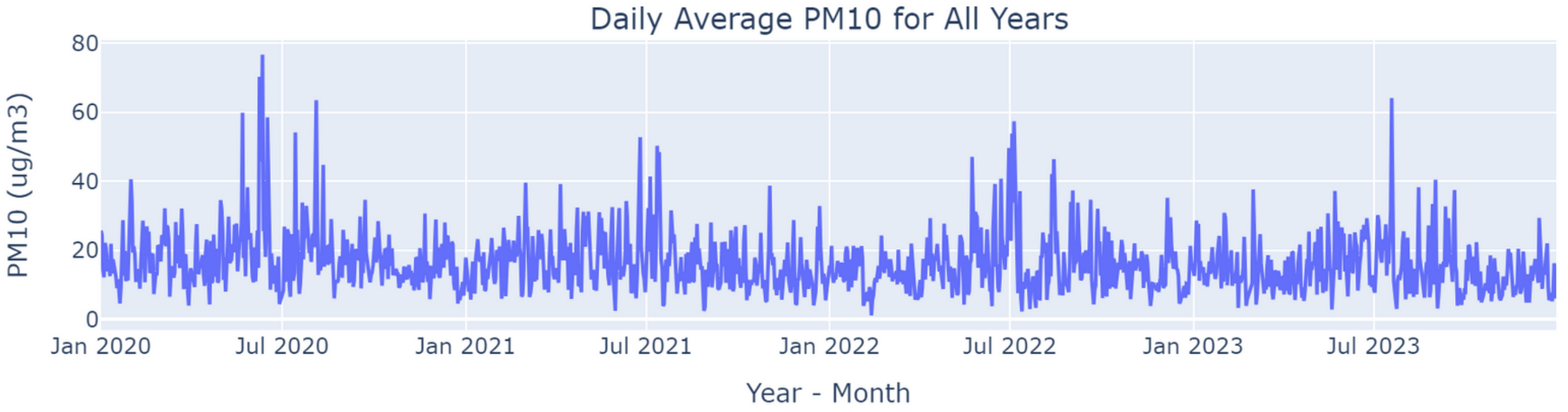


EDA for Christchurch

PM2.5

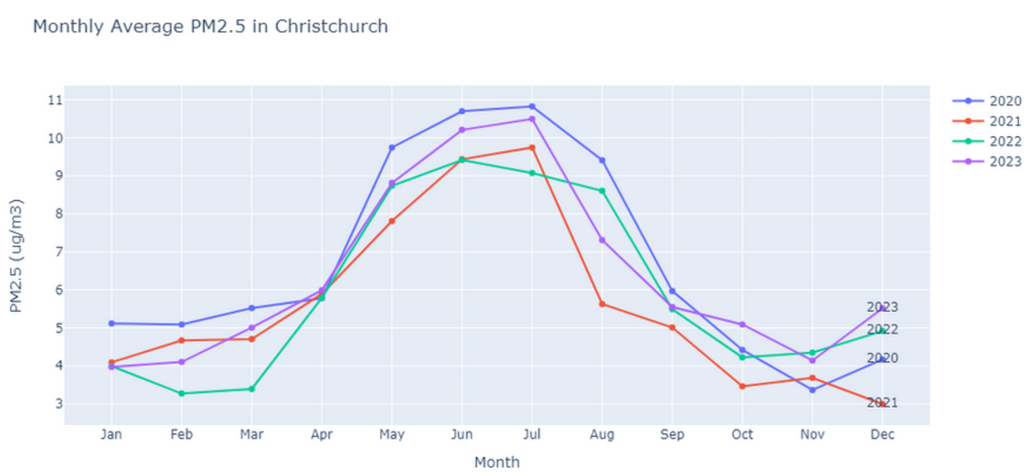


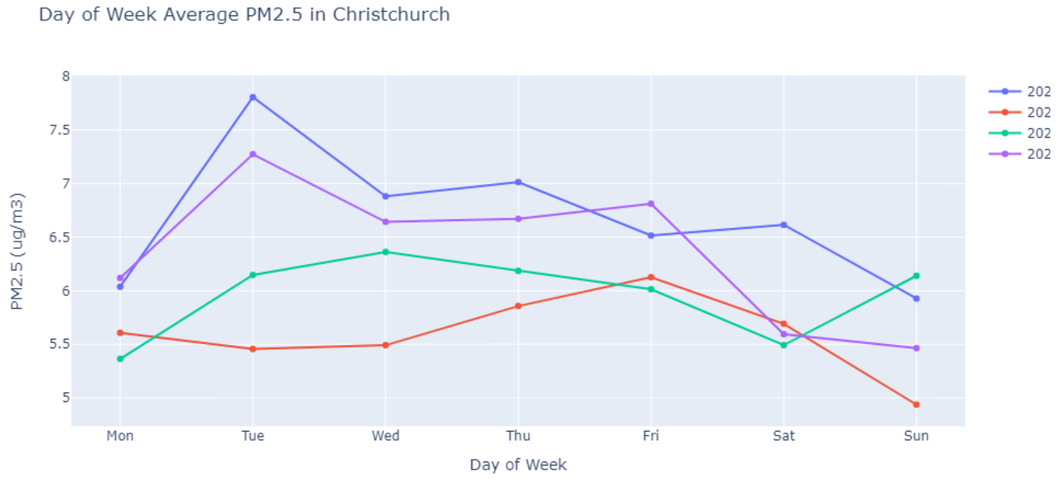
PM10



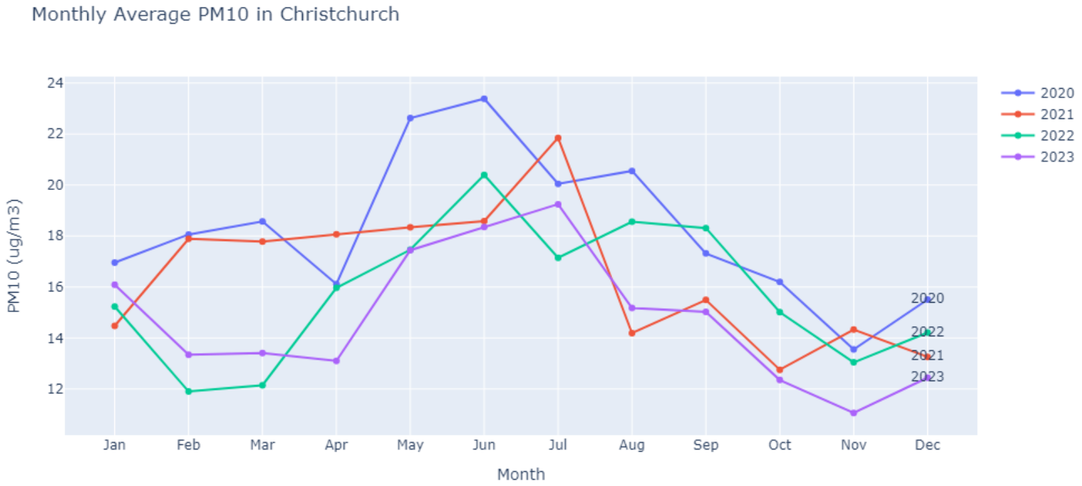
EDA for Christchurch

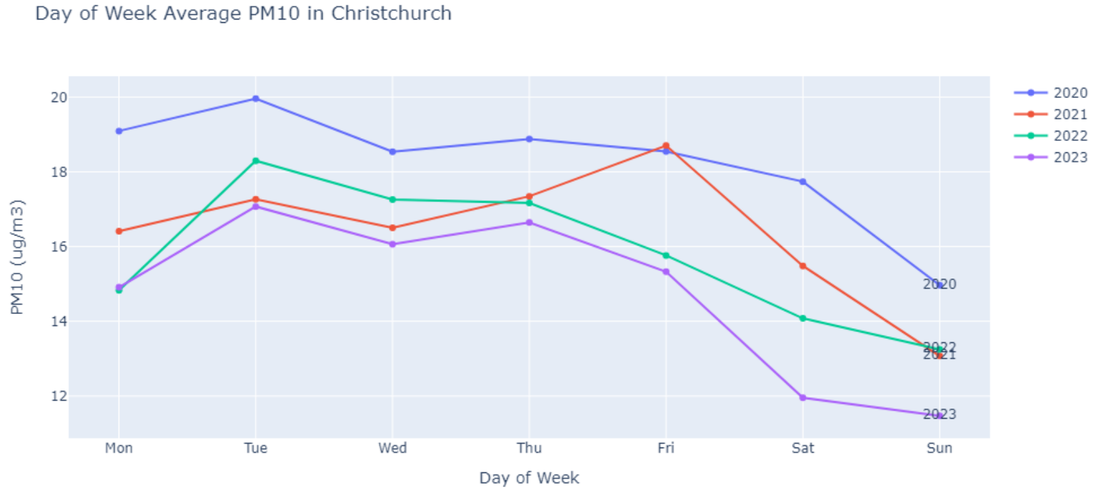
PM2.5





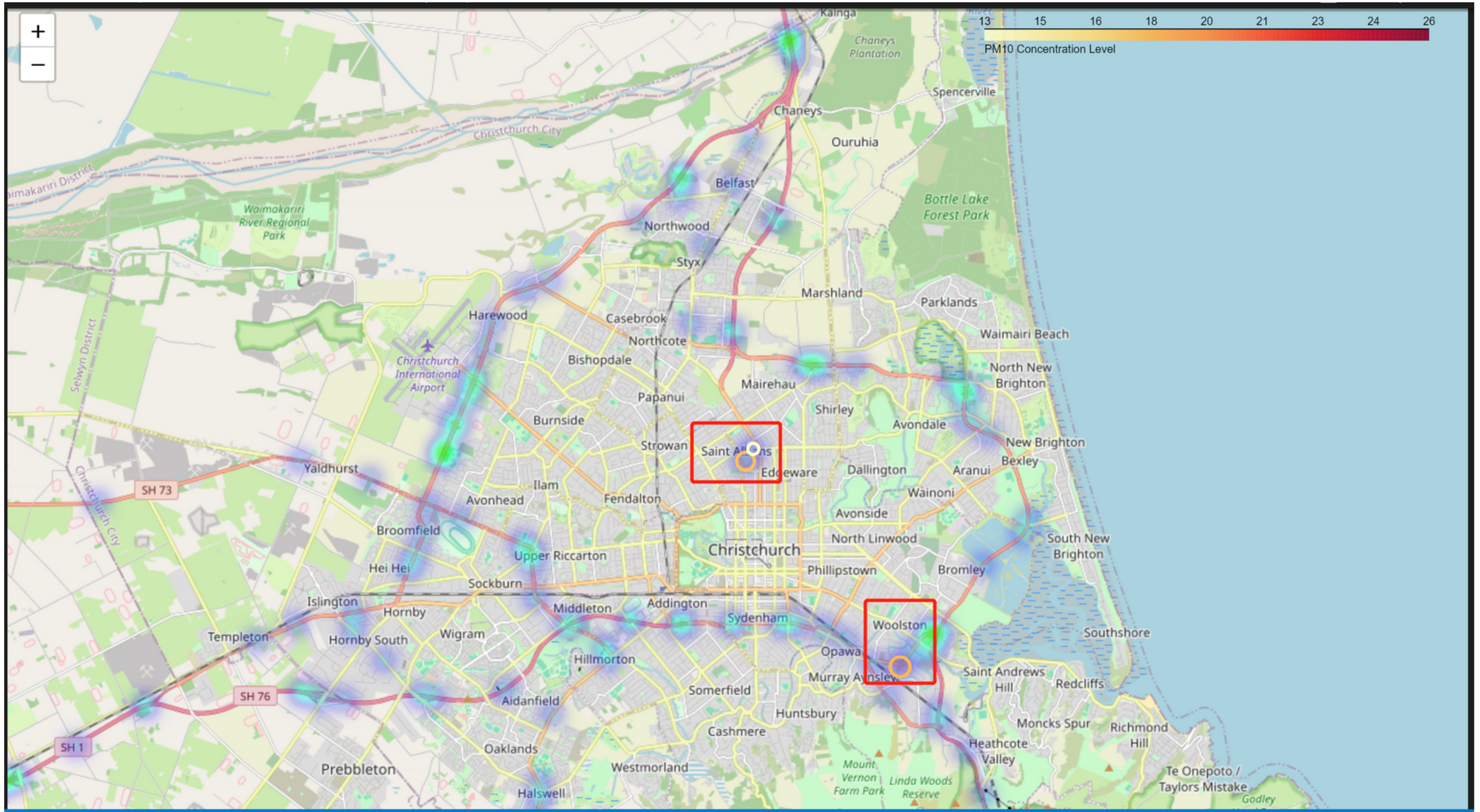
PM10





EDA for Christchurch: Geo-Spacial Analysis

Average AADT and Average PM10



Feature Engineering

Month & Year

```
# Add 'Month' and 'Year' columns
air_df01 = air_df.copy()
air_df01['Year'] = [d.year for d in air_df01['Date']]
air_df01['Month'] = [d.strftime('%b') for d in air_df01['Date']]
years = air_df01['Year'].unique()
months = ['Jan', 'Feb', 'Mar', 'Apr', 'May', 'Jun', 'Jul', 'Aug', 'Sep', 'Oct', 'Nov', 'Dec']
air_df01
```

✓ 0.0s

Day of Week & Is Weekend

```
# Add 'Day of Week' and 'Is Weekend' columns
day_of_week = {0: 'Mon', 1: 'Tue', 2: 'Wed', 3: 'Thu', 4: 'Fri', 5: 'Sat', 6: 'Sun'}
air_df01['Day of Week'] = air_df01['Date'].dt.weekday.map(day_of_week)
air_df01['Is Weekend'] = air_df01['Date'].dt.weekday > 4
air_df01
```

✓ 0.0s

Average Temperature & Total Traffic

```
# add daily average temperature column and total traffic column
air_df01['T(C)'] = air_df01[['Tmax(C)', 'Tmin(C)']].mean(axis=1)
air_df01['trafficCount'] = air_df01[['lightCount', 'heavyCount']].sum(axis=1)
air_df01
```

✓ 0.0s

Data Preprocessing - Missing Values

- Seasonal features: Seasonal mean
- Non-seasonal features: Median

```python
def seasonal_mean(ts, n, lr=0.7):
    """
    Compute the mean of corresponding seasonal periods
    ts: 1D array-like of the time series
    n: Seasonal window length of the time series
    """
    out = np.copy(ts)
    for i, val in enumerate(ts):
        if np.isnan(val):
            ts_seas = ts[i-1::-n]  # previous seasons only
            if np.isnan(np.nanmean(ts_seas)):
                ts_seas = np.concatenate([ts[i-1::-n], ts[i::n]])  # previous and forward
            out[i] = np.nanmean(ts_seas) * lr
    return out
```

Data Preprocessing - Outliers

- Detect outliers using IQR
- Replace outliers using Median

```
# Define a function to detect outliers using IQR
def detect_outliers(column):
    Q1 = column.quantile(0.25)
    Q3 = column.quantile(0.75)
    IQR = Q3 - Q1
    lower_bound = Q1 - 1.5 * IQR
    upper_bound = Q3 + 1.5 * IQR
    return (column < lower_bound) | (column > upper_bound)

outliers = air_df04.select_dtypes(exclude='object').apply(detect_outliers)
outliers
```

✓ 0.0s

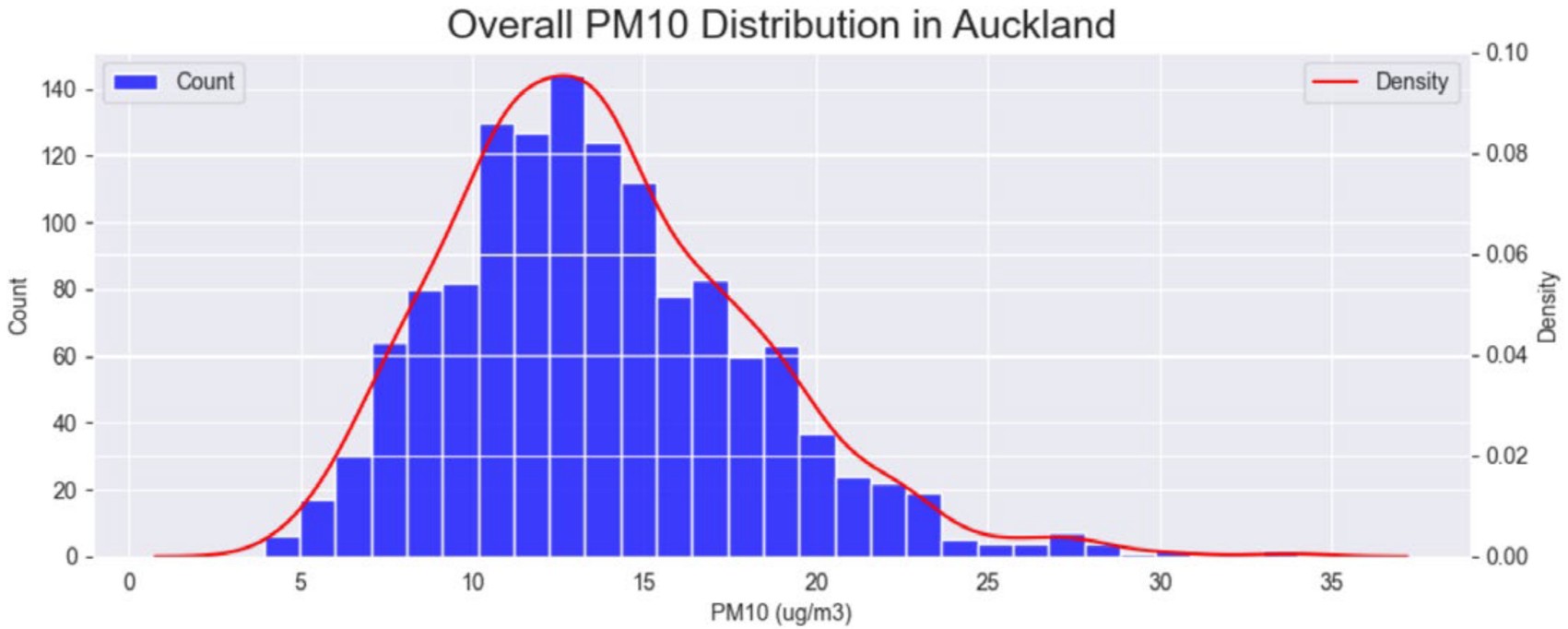
```
# change outliers as median
for column in air_df04.select_dtypes(exclude='object').columns:
    median_value = air_df04[column].median()
    outliers_column = outliers[column]
    air_df04.loc[outliers_column, column] = median_value

air_df04
```

✓ 0.0s

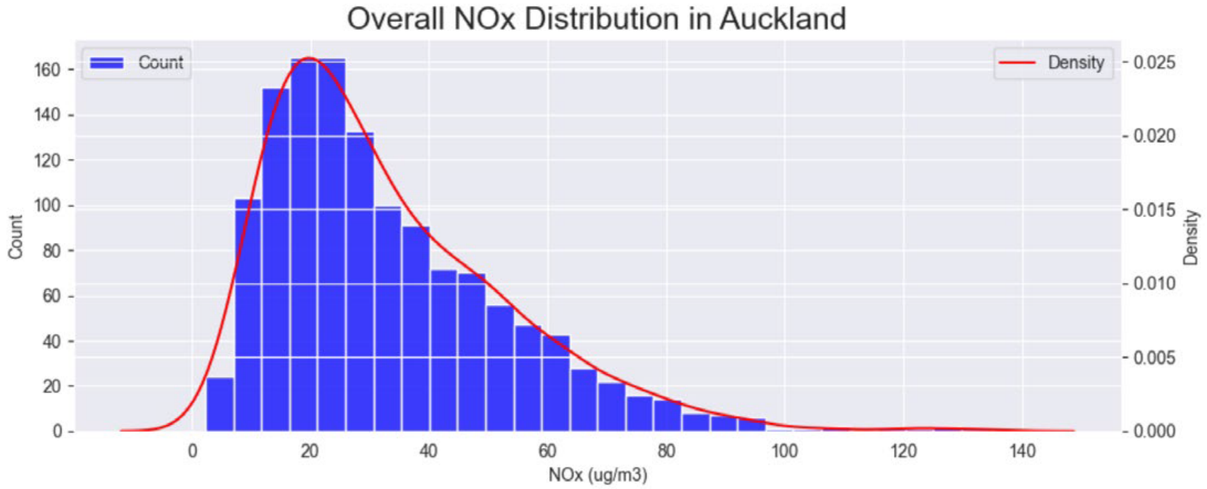
Data Preprocessing - Data Scaling

- Normal Distribution
- Skewed Normal Distribution
- Non-Normal Distribution

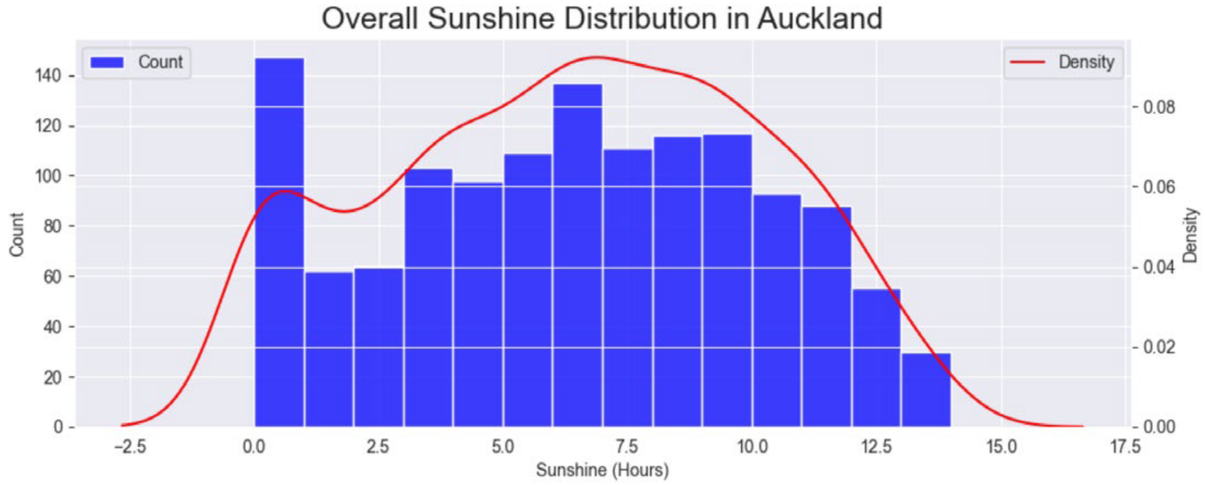


Normal distribution

Data Preprocessing - Data Scaling



Skewed Normal Distribution

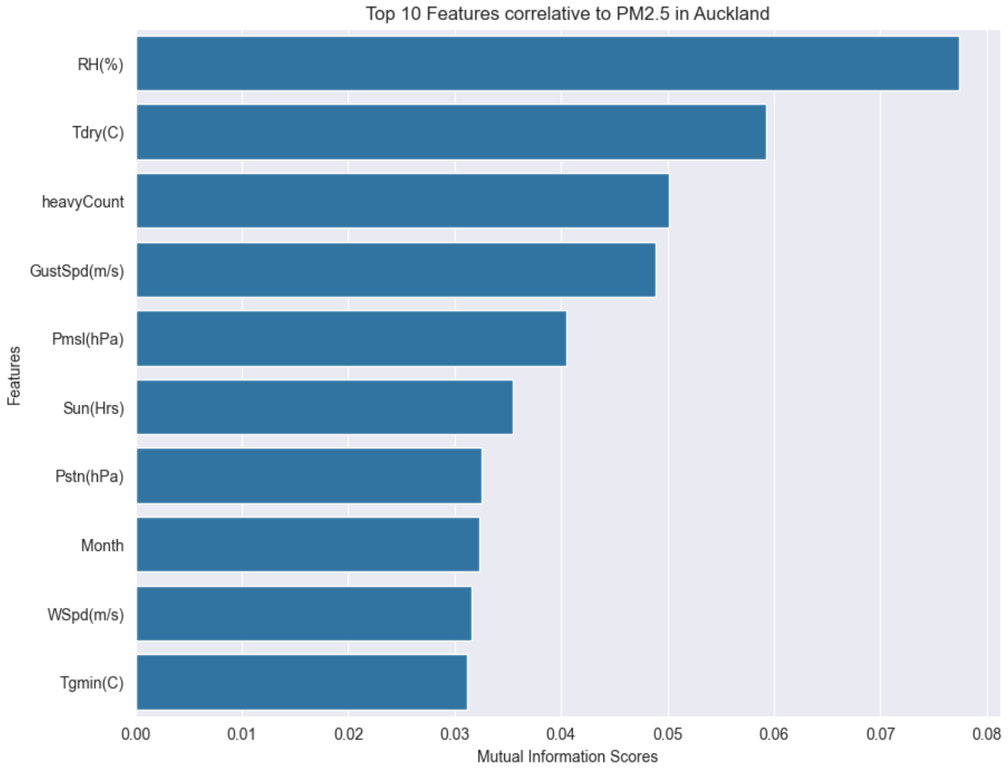


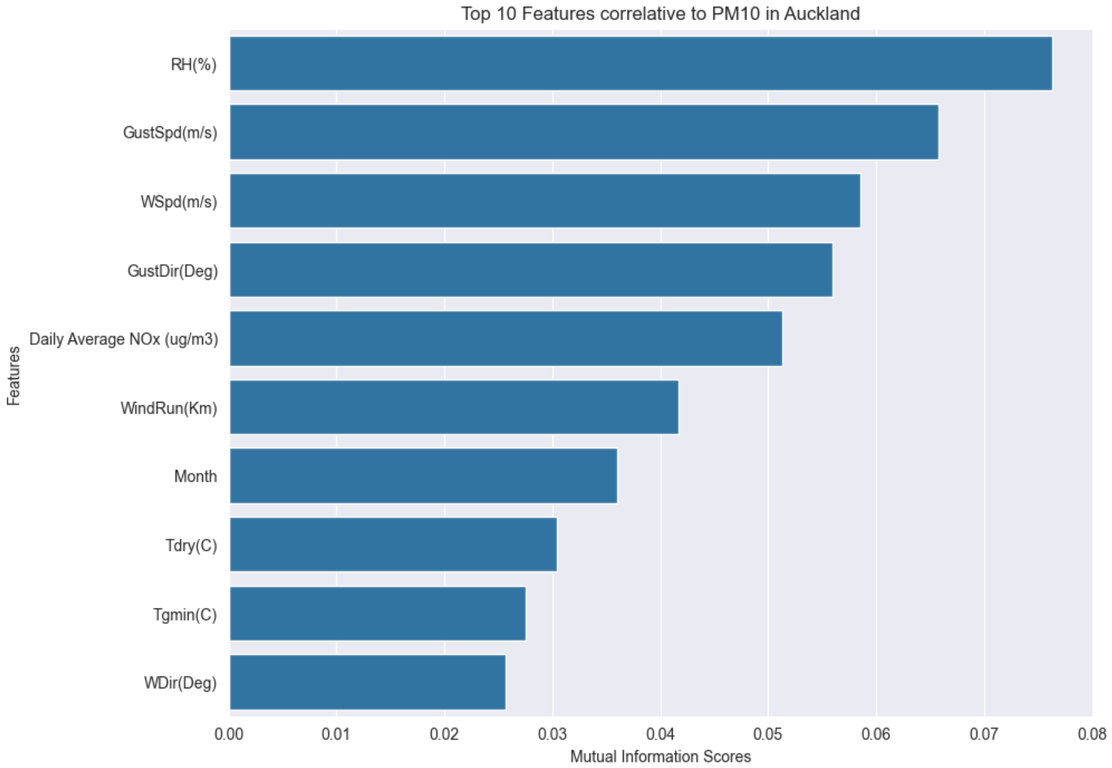
Non-Normal Distribution

Data scaling: MinMaxScaler

- Specific range: between 0 and 1 for neural network algorithm
- Maintaining the original distribution

Feature Selection - Auckland

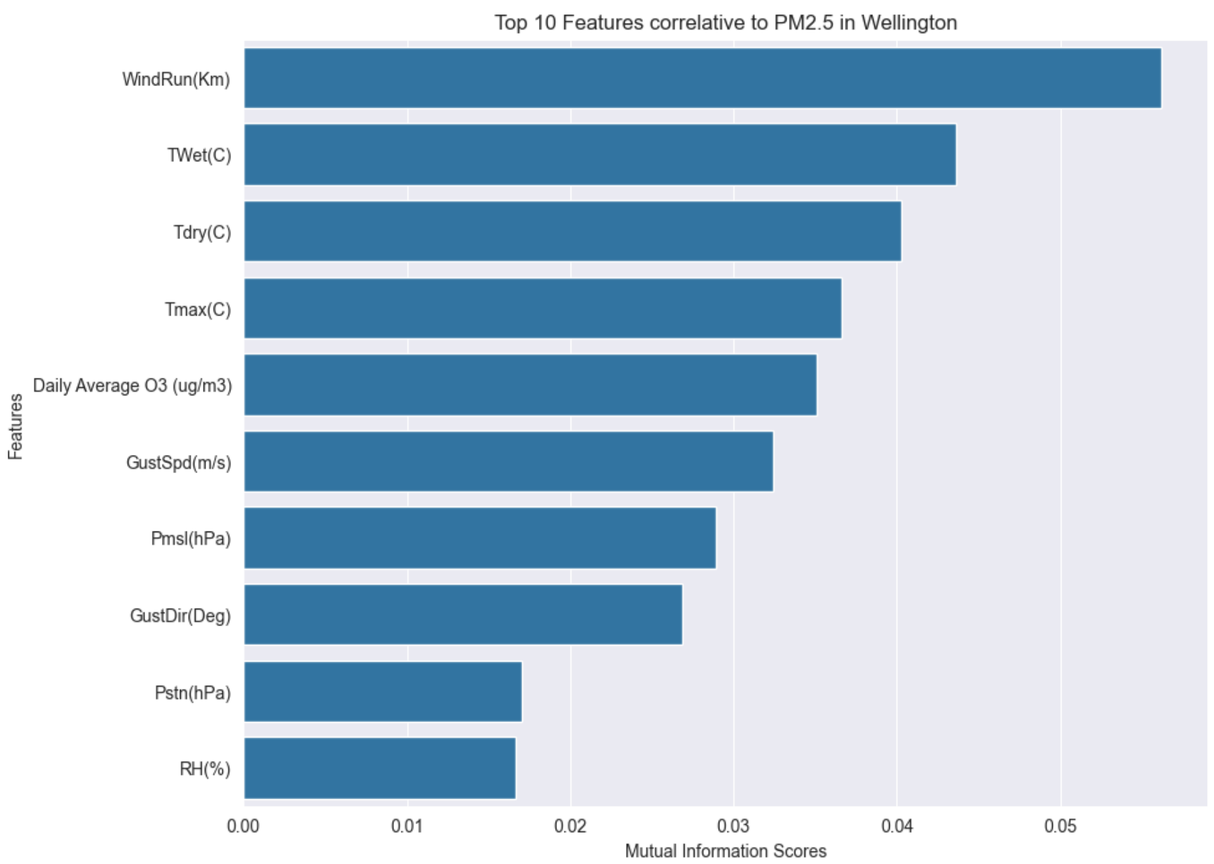




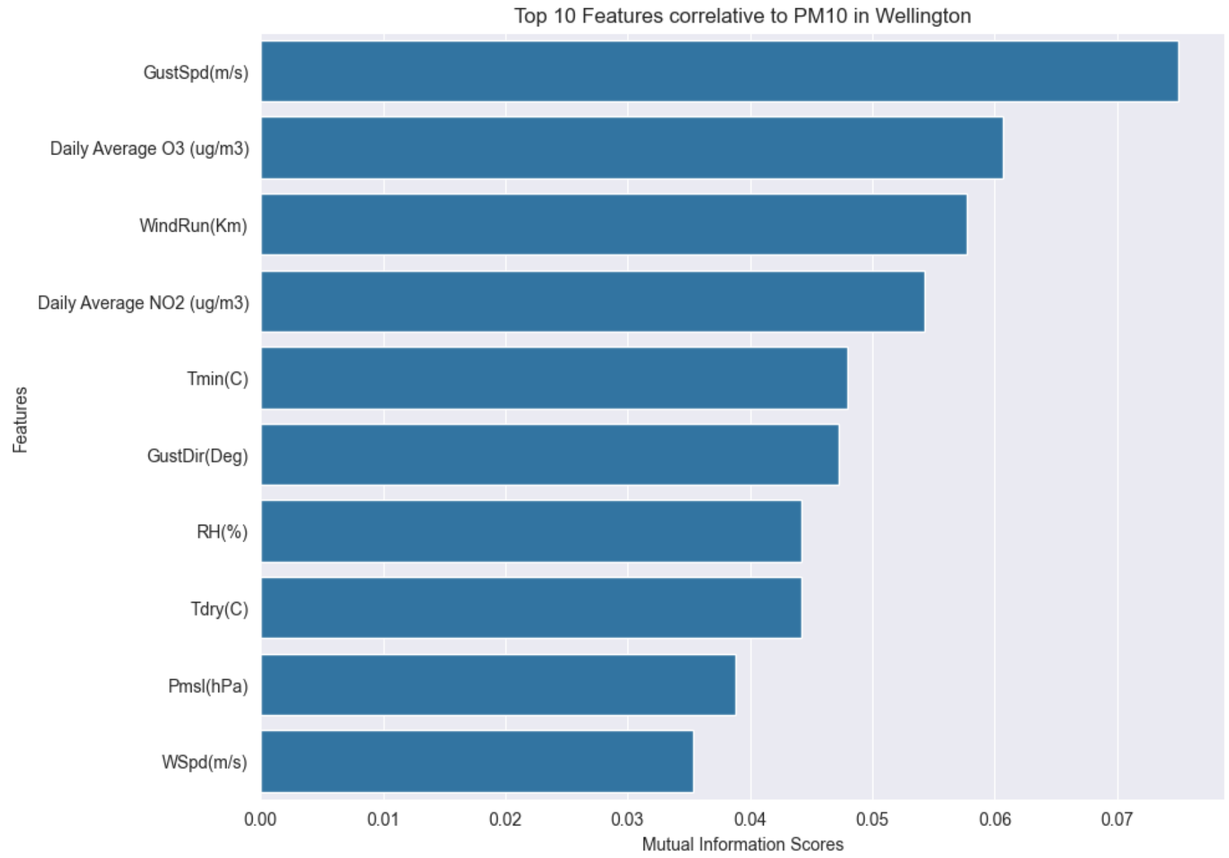
Top 1 correlative to PM2.5:
Relative Humidity (%)

Top 1 correlative to PM10:
Relative Humidity (%)

Feature Selection - Wellington

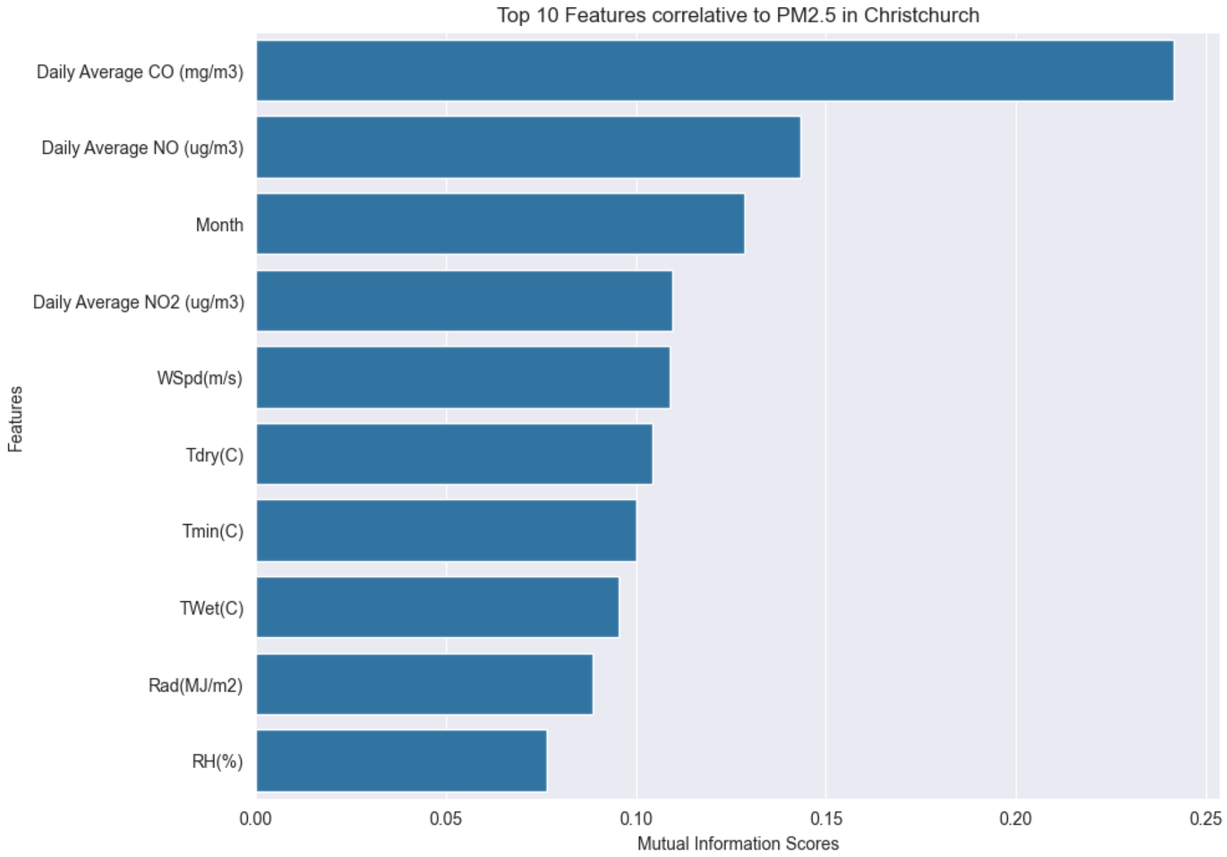


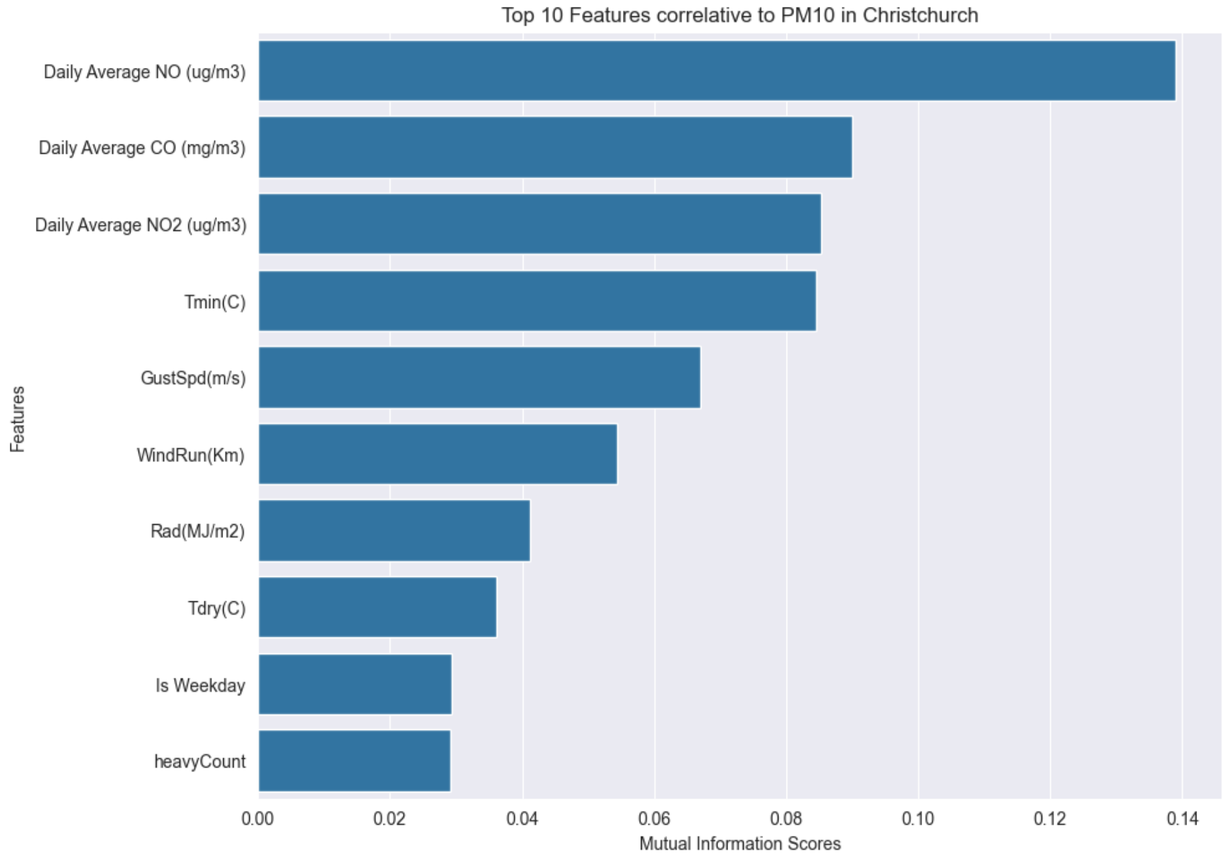
Top 1 correlative to PM2.5:
Wind Run (km)



Top 1 correlative to PM10:
Gust Speed (m/s)

Feature Selection - Christchurch





Top 1 correlative to PM2.5:
CO (mg/m3)

Top 1 correlative to PM10:
NO (ug/m3)

MODEL

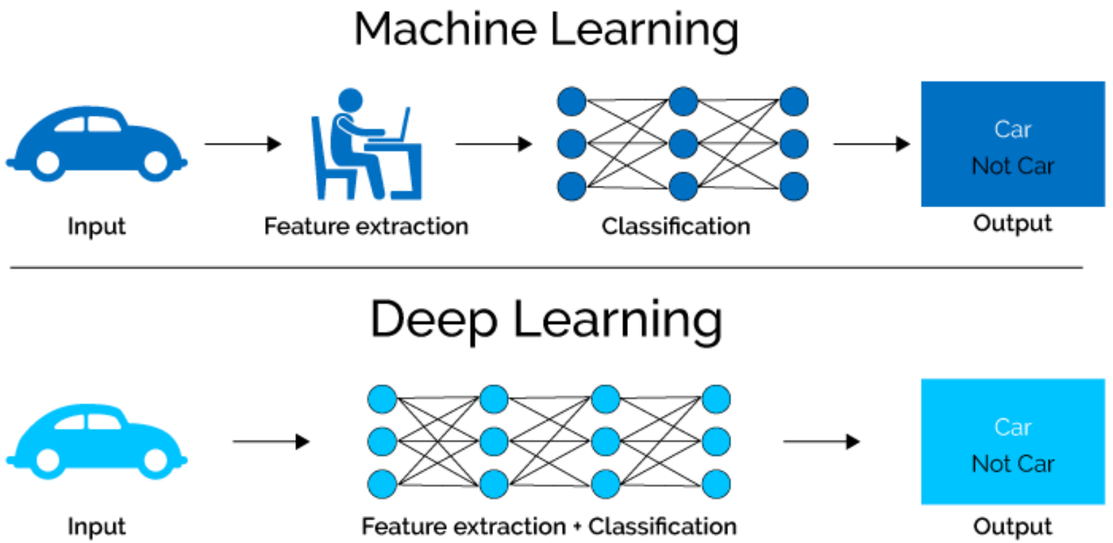
Time Serious Prediction

Machine Learning

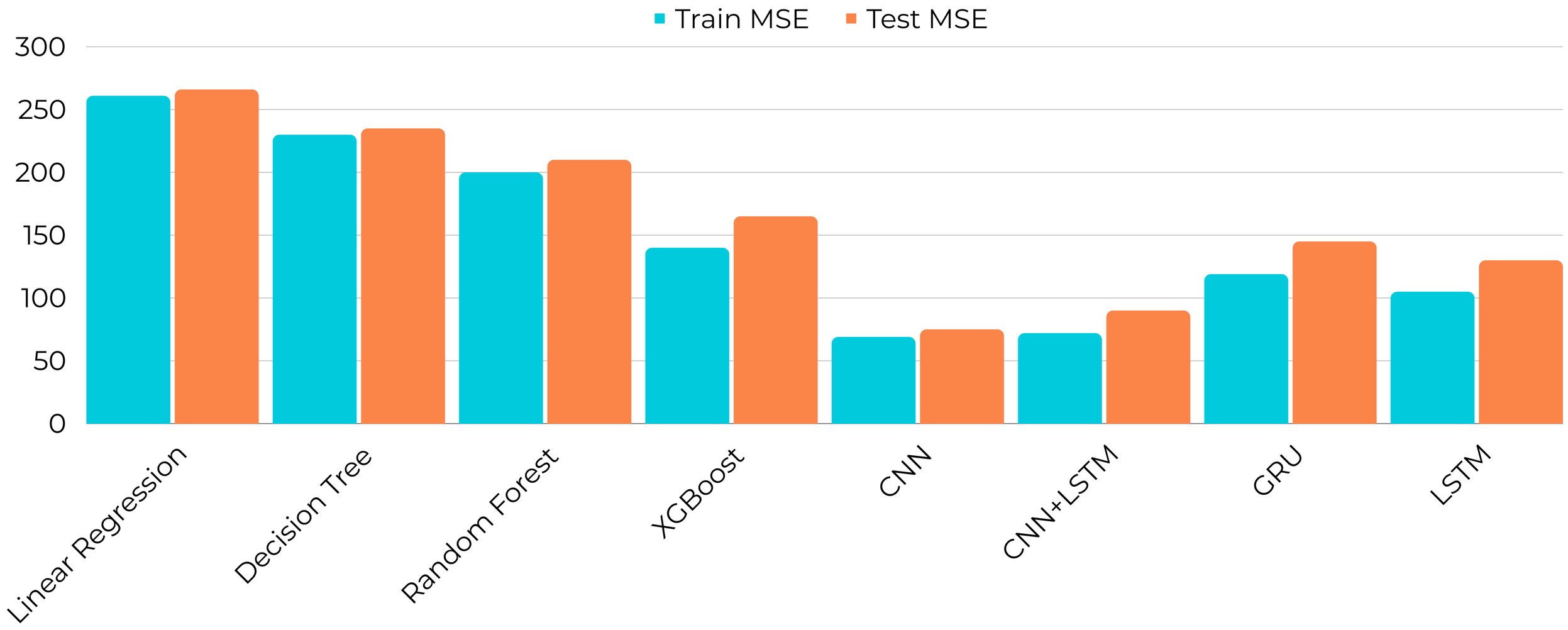
- Linear Regression
- Decision Tree
- Random Forest
- XGBoost

Deep Learning

- Convolutional Neural Networks
- LSTM(long short-term memory networks)
- GRU(Gated Recurrent Units)



Model Comparision



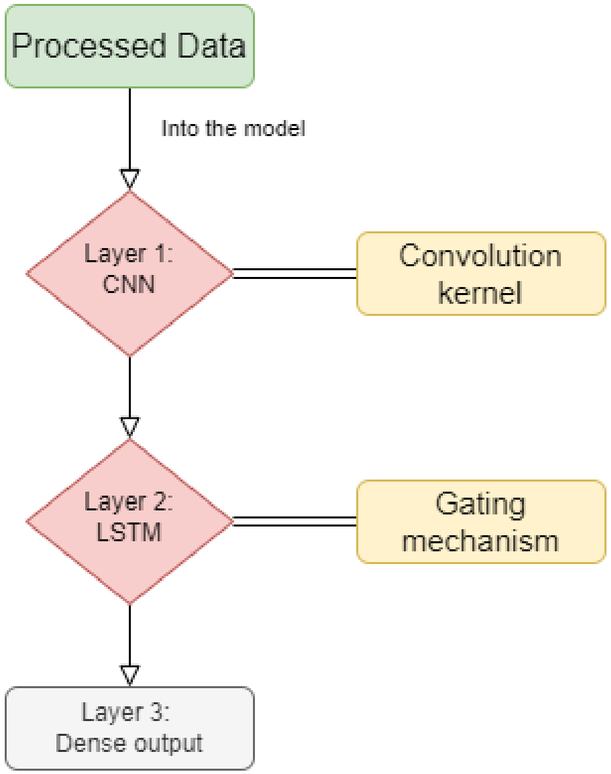
MSE is like a scoring standard, used to tell us how accurate the model's predictions are. The smaller the MSE, the closer the model's predictions are to the real situation.

WE DECIDE

CNN+LSTM

FOR
PROFESSIONAL
MODEL

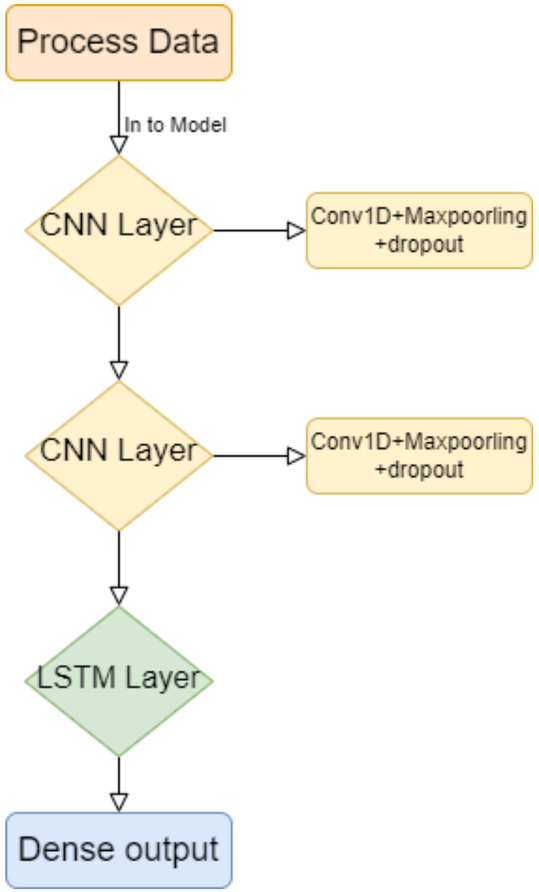
Model Structure(Version 1.1)



Filters input data, extracts features

Controls information flow

(Version 1.2)



More CNN layers &add dropout

change optimizer

Adjust the learning rate of neurons

After we did some feature engineering

We Pick Version 1.1 As The Demonstration Model

Model Connecting Speed

Demo

Version 1.1

> 1.39 Times Faster

Professional

Version 1.2

Professional Model Optimazation Version 1.2

01

Feature Engineering:

Enhancing input data quality through preprocessing techniques to improve model accuracy.

02

Model Architecture Tuning:

Adjusting CNN and LSTM layers, including the number of layers, neurons, and kernel sizes to balance model complexity and performance.

03

Regularization Techniques:

Implementing dropout, L1/L2 regularization to prevent overfitting and improve the model's generalization ability.

04

Hyperparameter Optimization:

Using techniques like grid search or random search to find the optimal set of model parameters, including learning rate, batch size, and epoch number, to enhance prediction accuracy.

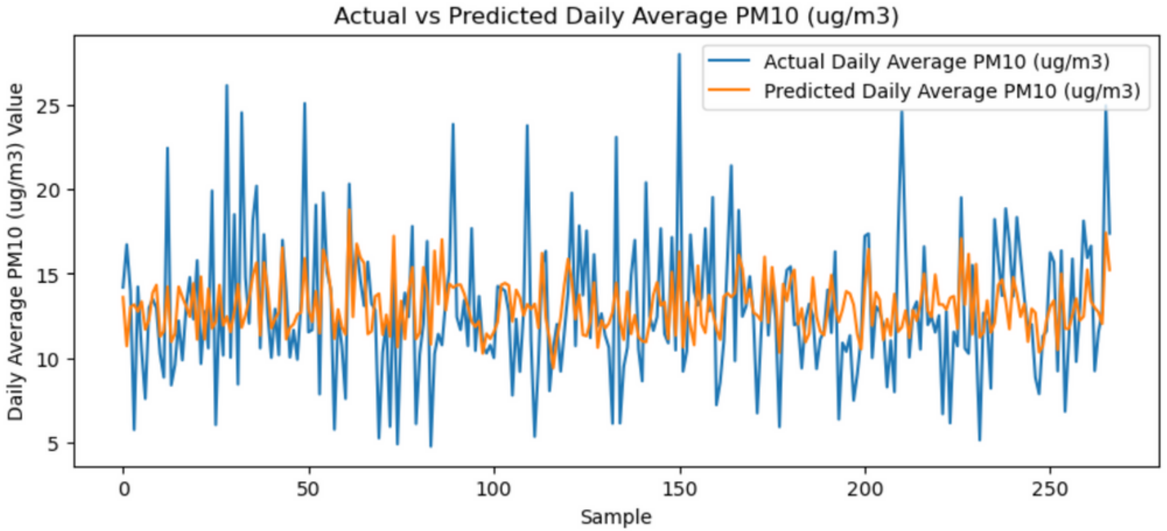
Professional Model Example in Auckland

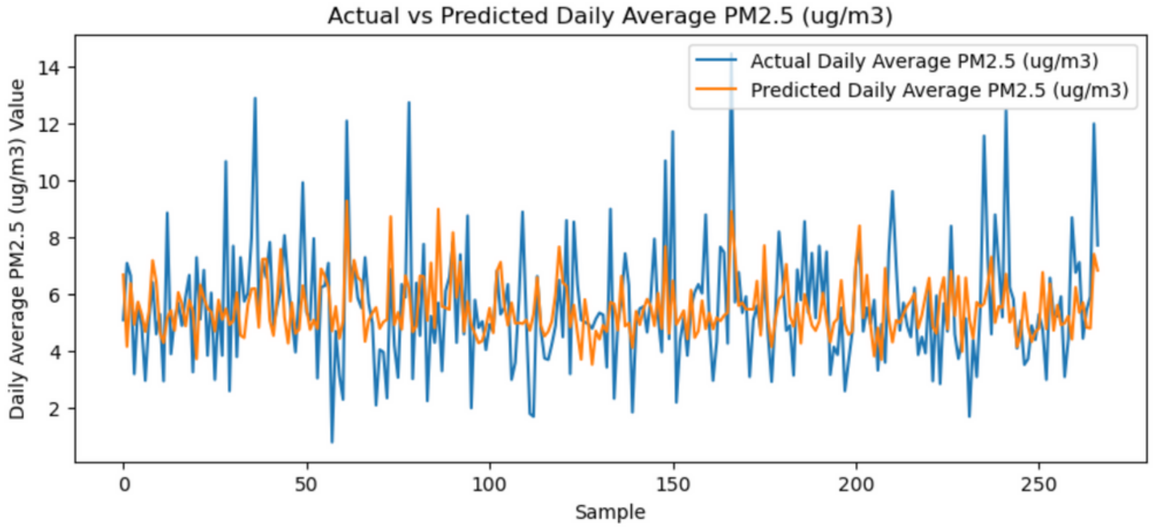
Mean Squared Error(MSE):34.179 --> 25.50
Mean Absolute Error(MAE):3.99 --> 3.34
Train R^2 Score:0.38 --> 0.4255
Test R^2 Score: 0.183 --> 0.1998

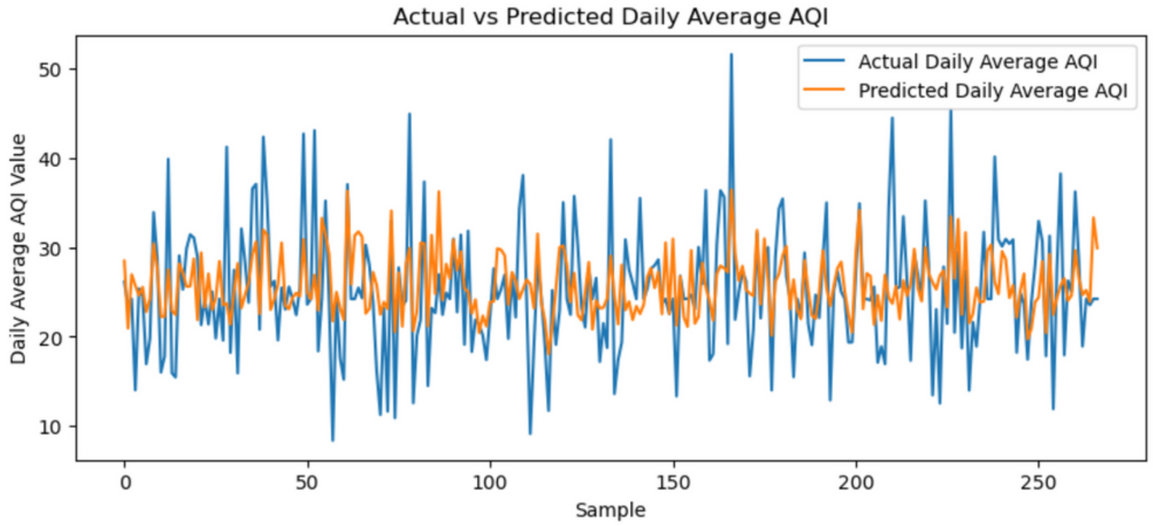
Training R2 score can explain approximately **42.55%** of the **variability** of the target variable. This shows that the model has some predictive power on the training set, but still has enough potential to continue to optimize in the future.

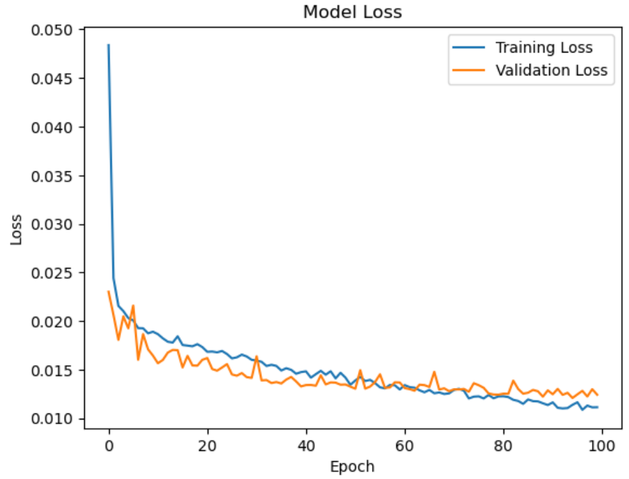
The test R2 score of the test set is about **18.38%,** which indicates that the model's prediction ability for unseen data is a little weak and the model's generalization ability is limited.

Version 1.2 model performance in Auckland region

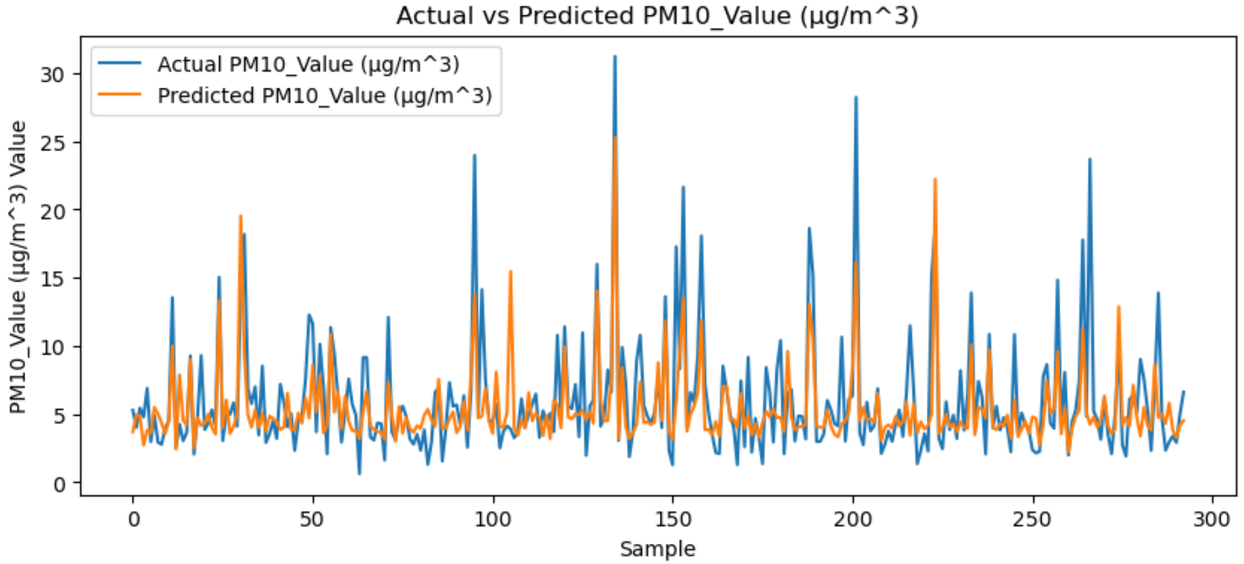


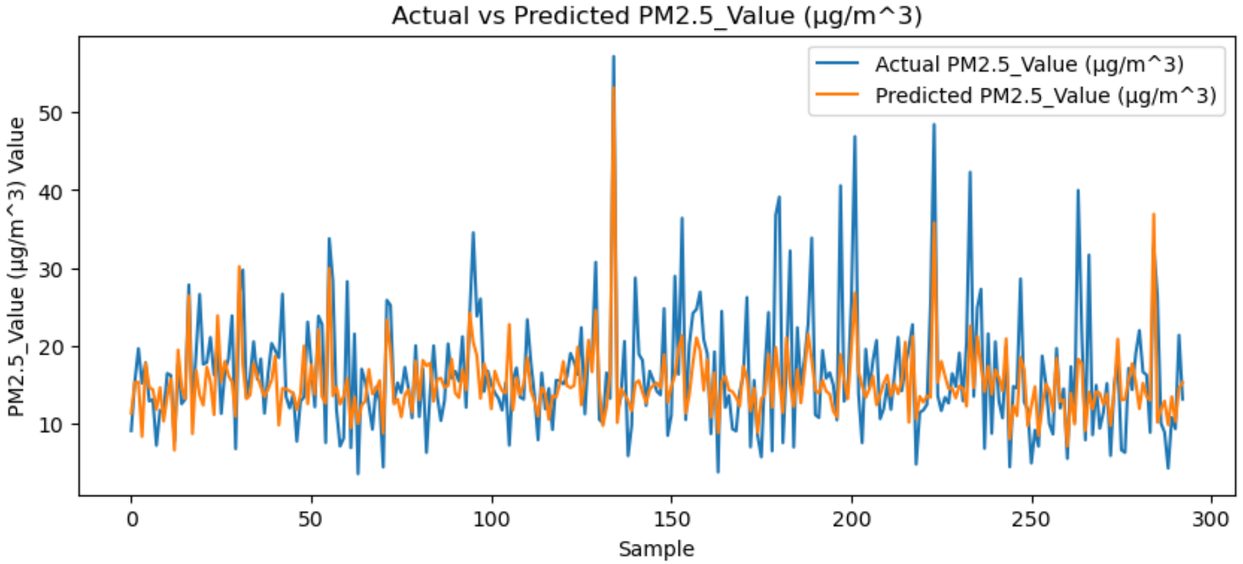




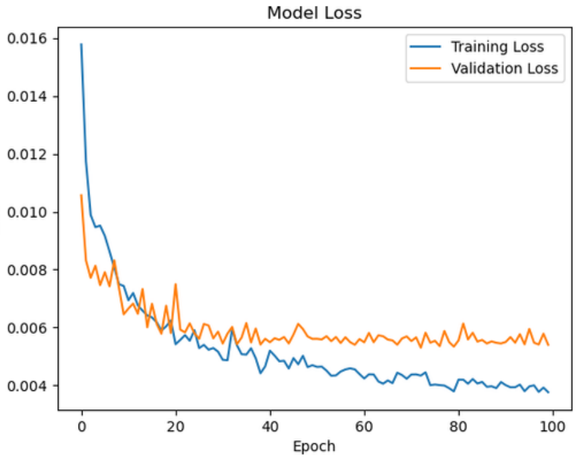


Version 1.2 model performance in ChristChurch region





Mean Squared Error(MSE):13.889 --> 12.11
Mean Absolute Error(MAE):3.190 --> 2.508
Train R^2 Score:0.72 --> 0.77
Test R^2 Score: 0.40 --> 0.47



WHAT WE FOUND

The Christchruch Model is the best.

All models still have potential for improvement.

Demo Model Process

The demo model process - Random Forest

Efficiency in handling classification and regression problems

Traffic Count (trafficCount):

We can use quartiles as the basis for categorization:

Low traffic (0): Less than the 25th percentile

Medium traffic (1): Between the 25th and 75th percentiles

High traffic (2): Greater than the 75th percentile

Rainfall (Rain.mm.):

Most values are 0, indicating no rainfall.

Values greater than 0 are categorized as 1, indicating rainfall.

Date processing: Extract year, month, and day from the Timestamp column.

So the features' columns are: 'Year', 'Month', 'Day', 'Categorized_Traffic', 'Tmax.C.', 'Tmin.C.', 'Categorized_Rain', and the labels' columns are:'PM2.5','PM10'

MSE for AQI prediction: About 39.15

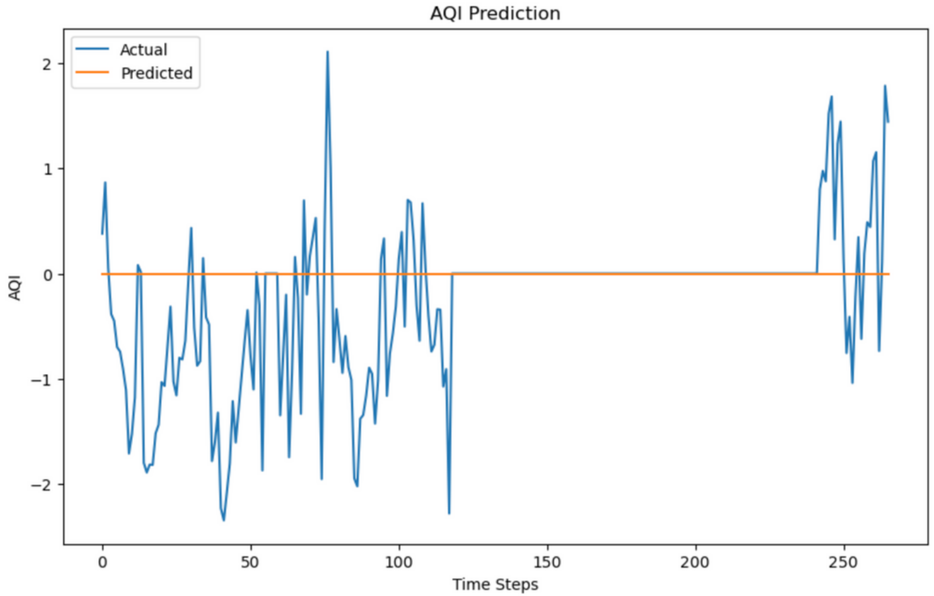
MSE for PM2.5 prediction: About 2.38

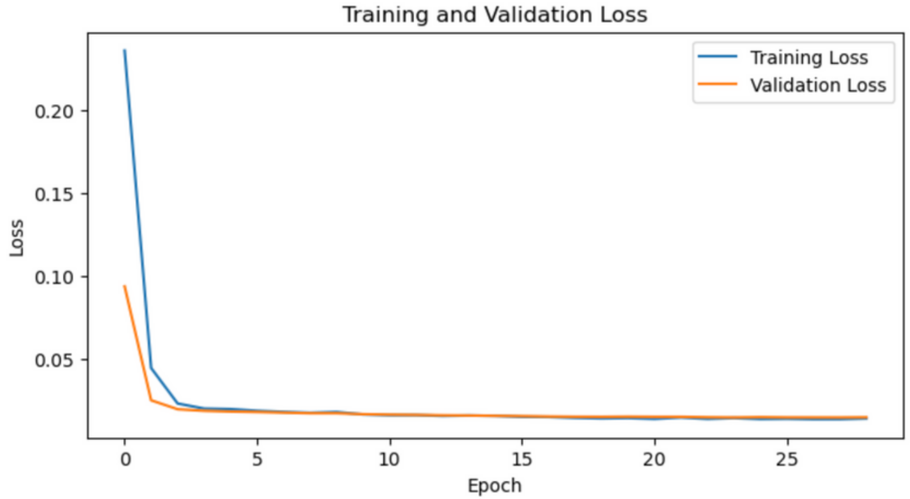
R^2 score for AQI prediction: About 0.32

R^2 score for PM2.5 prediction: About 0.30

The demo model process - LSTM

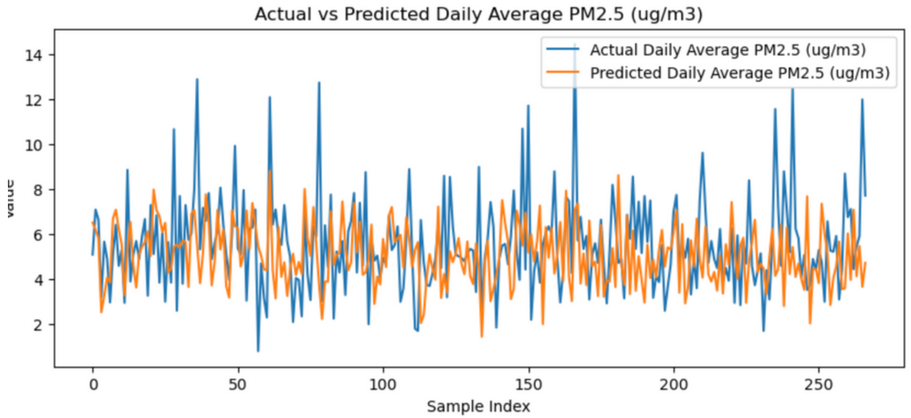
Capturing temporal correlation between data

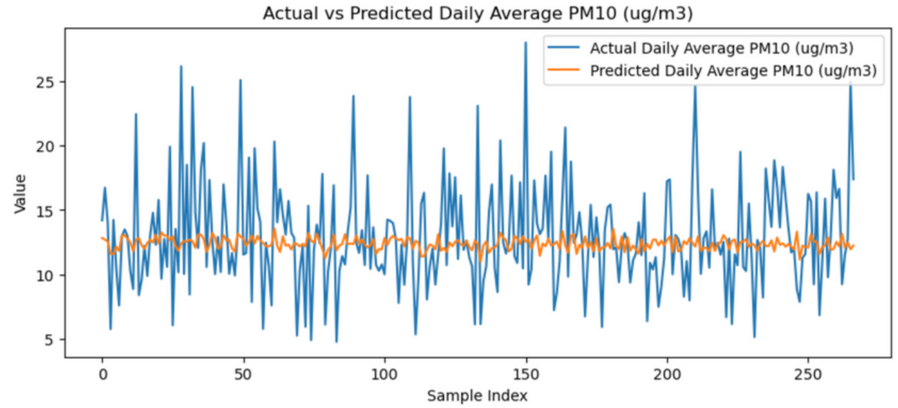


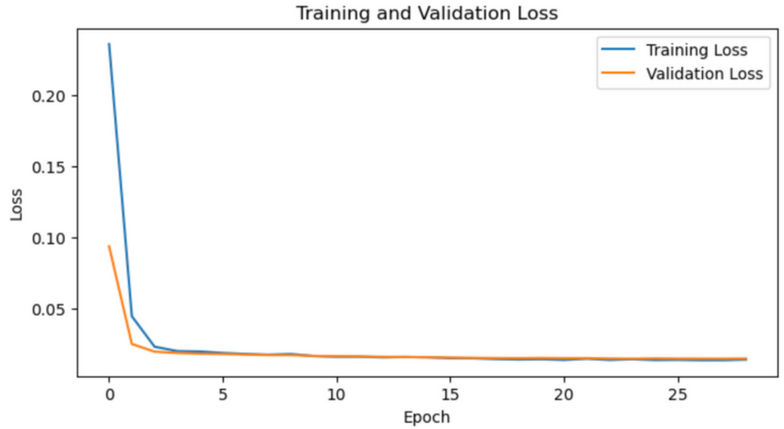


The demo model process - LSTM

Capturing temporal correlation between data





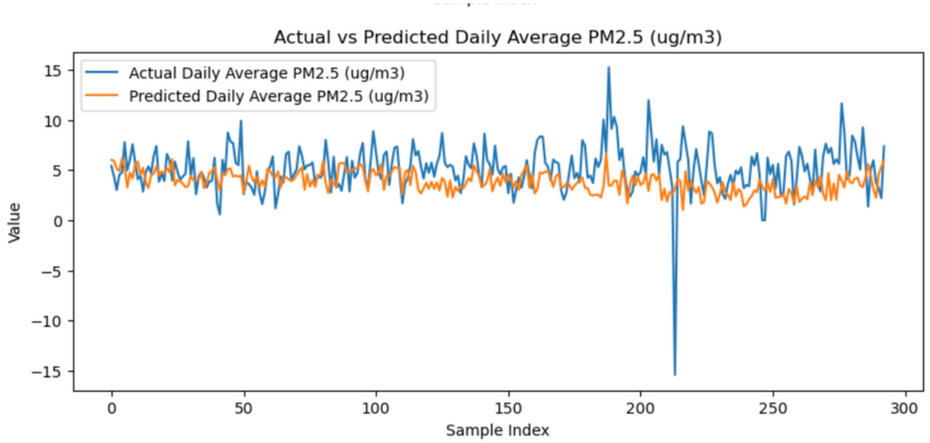


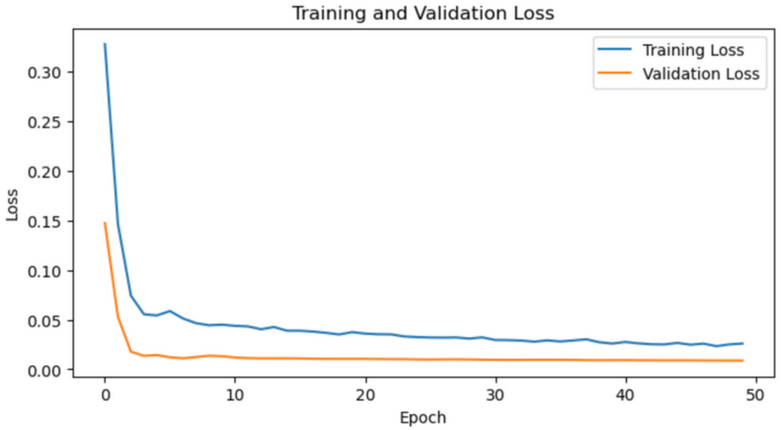

```
9/9 [==============================] - 0s 899us/step
Mean Squared Error (MSE):  11.333658112660647
Mean Absolute Error (MAE):  2.4264163834473886
Test Loss: 0.011941026896238327
```

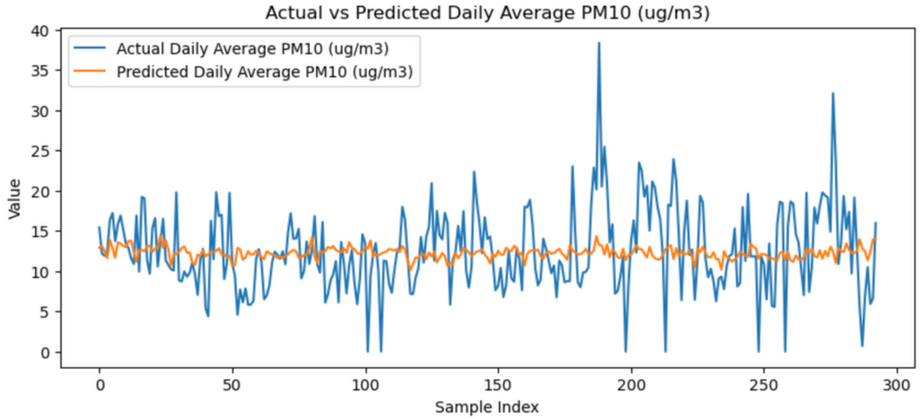
Demo model performance in Auckland

The demo model process - LSTM

Capturing temporal correlation between data





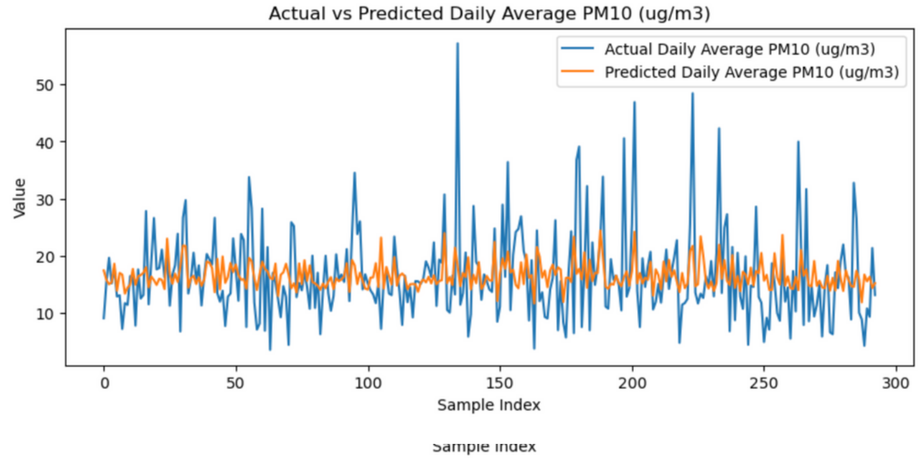


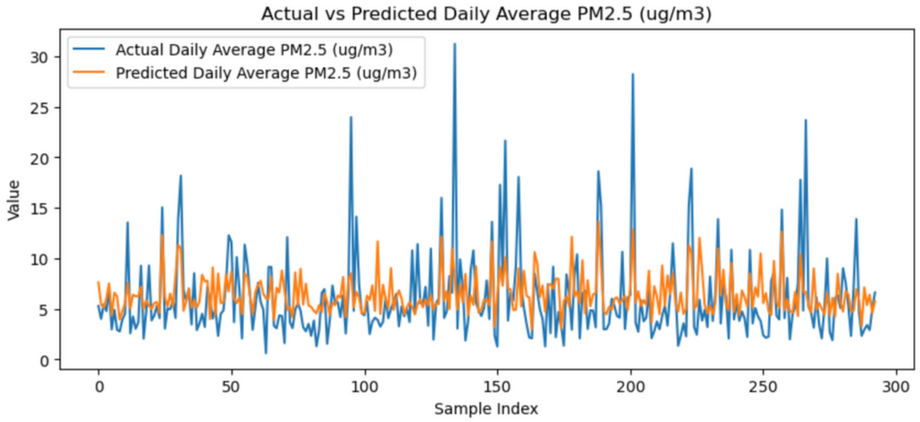

```
10/10 [==============================] - 0s 575us/step
Mean Squared Error (MSE):  17.007743475960208
Mean Absolute Error (MAE):  2.9917813688170782
Test Loss: 0.01234412845224142
```

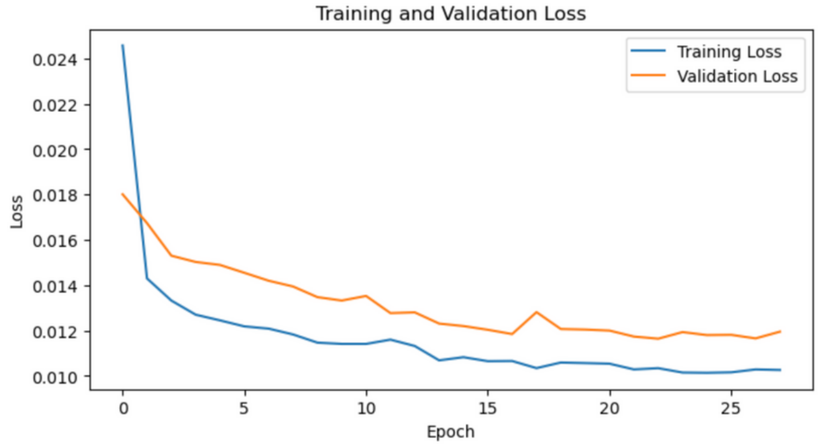
Demo model performance in Wellington

The demo model process - LSTM

Capturing temporal correlation between data








```
Mean Squared Error (MSE):  34.17945206214422
Mean Absolute Error (MAE):  3.9903637838122
Test Loss: 0.008400311693549156
```

Demo model performance in Christchurch

WEBSITE

Our Services

We have created a model interaction website for people who do not understand machine learning and coding

Services 01

Conduct air quality prediction

Services 02

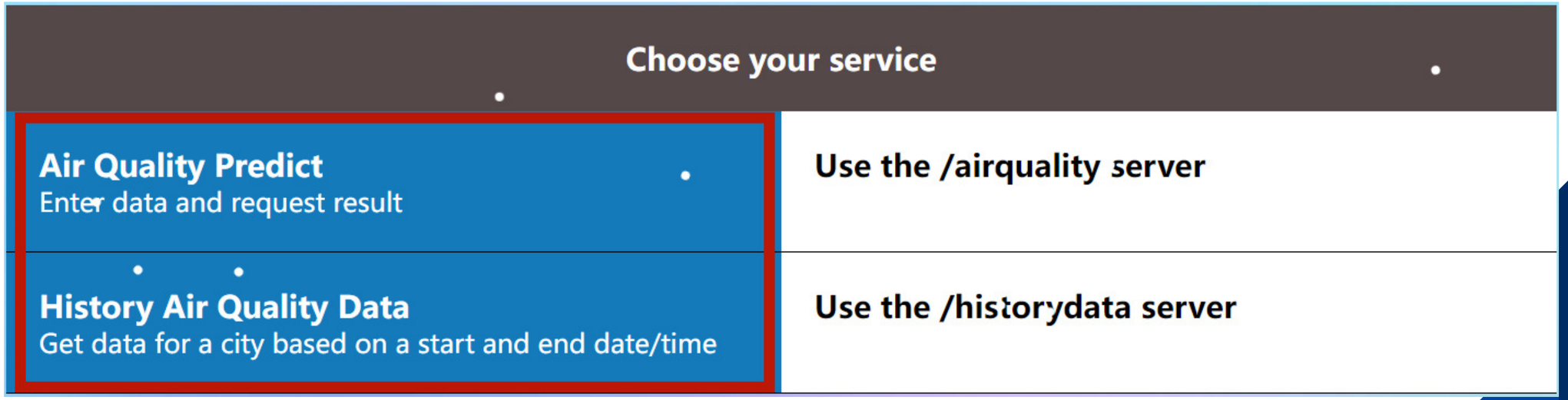
Conduct historical air quality data queries

Services 03

Allow users to customize features and export annual air quality data.

User Interface

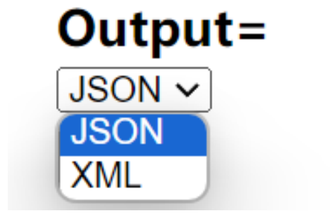
This is the main functional interface, where users can choose the desired functions

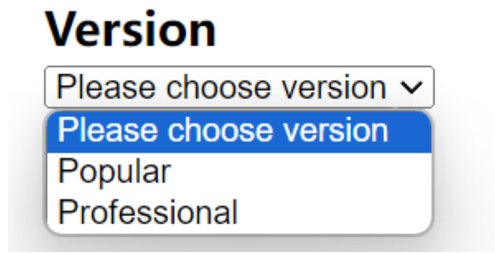


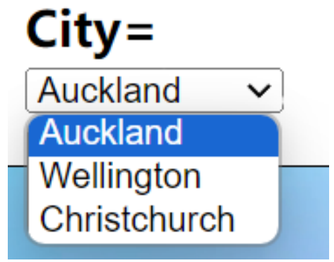
Air Quality Prediction

Choose your service	
Air Quality Predict Enter data and request result	**Use the /airquality server**
History Air Quality Data Get data for a city based on a start and end date/time	**Use the /historydata server**

Request from the airquality service	
Output Specify the data format used to return your query (JSON default)	**Output=** JSON
Version You can choose the version you want here. The features of the popular version will be smaller and faster than the professional version, but the accuracy is less	**Version** Please choose version
City Choose the city	**City=** Auckland





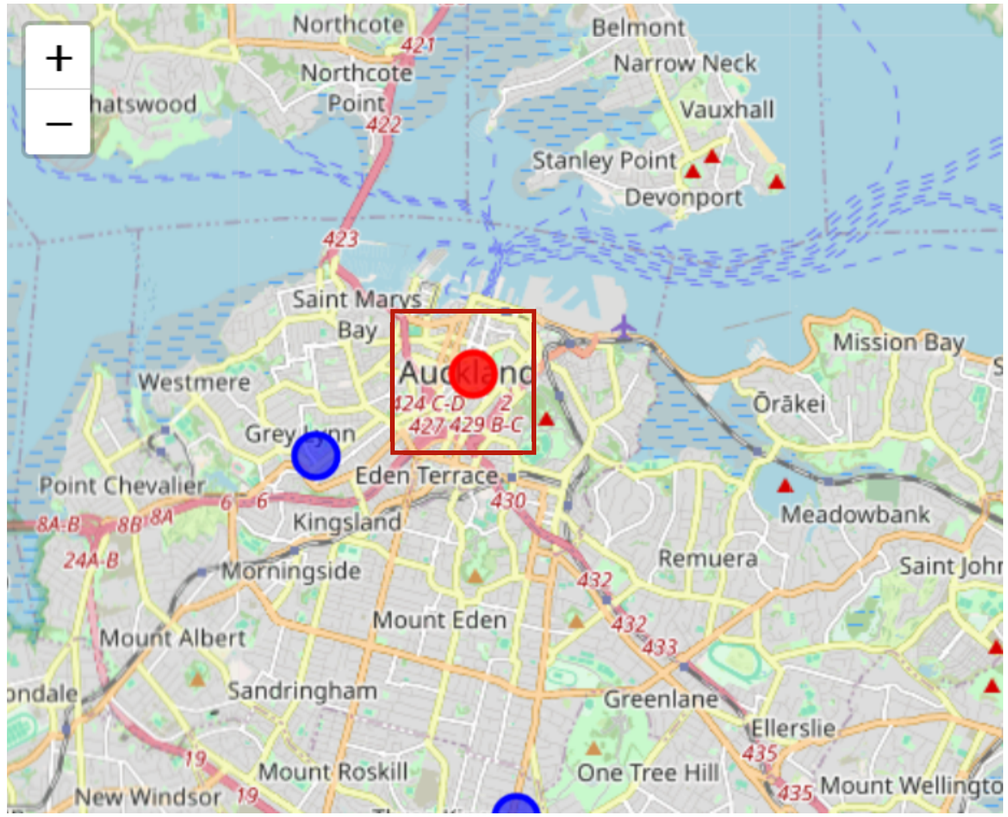


Popular Version

Date Predicted time point	**Date=** Example:2024-01-01
WDir Wind direction (degrees).	**WDir=** Realtime WDir is 245 degrees.
WSpd Wind speed (Default m/s).	**WSpd=** Realtime WSep is 4.63 m/s.
Rainfall Liquid equivalent precipitation rate (default mm/hr).	**Rainfall=** Realtime Rainfall is 0 mm/hr.
RH Relative humidity (%).	**RH=** Realtime Relative humidity is 49 %.

Tmax Maximum Temperature (°C).	**Tmax=** Today's Maximum Temperature is
Tmin Minimum Temperature (°C).	**Tmin=** Today's Minimum Temperature is
LightCount Light car traffic flow	**LightCount=** The small car traffic volume is 143294.5.
HeavyCount Heavy car traffic flow	**HeavyCount=** The heavy car traffic volume is 7317.5.

Auckland Real-time Data From The Following Station



WAKA KOTAHI NZ TRANSPORT AGENCY OPEN DATA

CW4778 Auckland

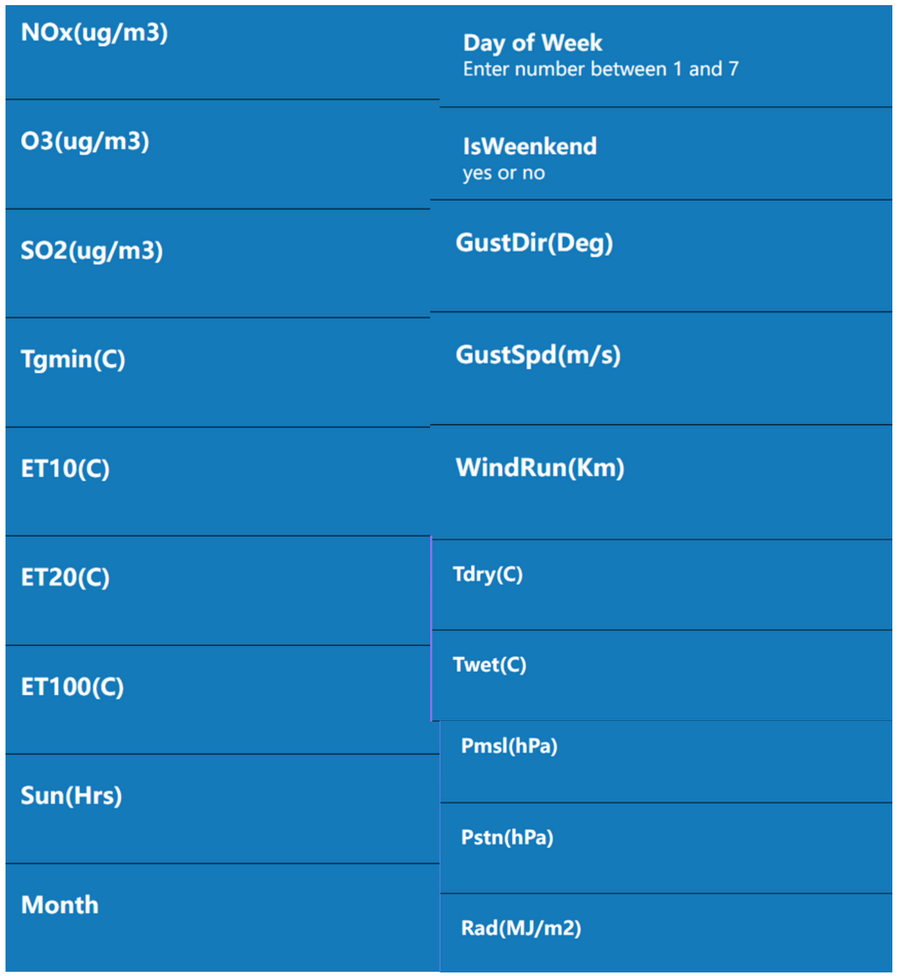
Network (ID)	APRSWXNET/CWOP (65)
Latitude	-36.86283
Longitude	174.73867
Elev (ft)	150.0
Station Start	2023-01-27

Output Result


```
{
  "prediction": {
    "city": "Auckland",
    "pm10": 14.2944240570068 36,
    "pm2.5": 17.936315536499023
  }
}
```


- CITY
- PM10
- PM2.5

Professional Version



- 19 additional features need to be inputted
- We are also unable to find the API that provides this data
- Required more time to run but more accurate

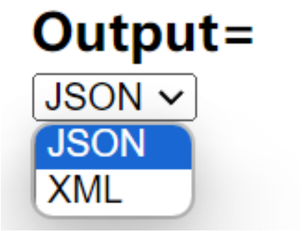
Output Result

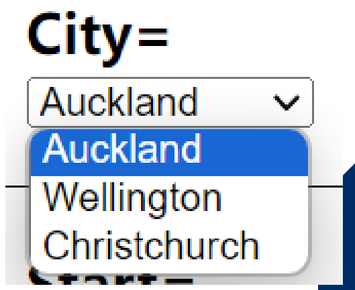

```
{
  "prediction": {
    "AQI": 67.62008666992188,
    "city": "Auckland",
    "pm10": 16.652309417724609,
    "pm2.5": 2.636292219161873
  }
}
```


- CITY
- AQI
- PM10
- PM2.5

History Data

History Air Quality Data Get data for a city based on a start and end date/time	**Use the /historydata server**
Request from the historydata service	
Output Specify the data format used to return your query (JSON default)	**Output=** JSON
City Choose the city	**City=** Auckland
Start YYYY-MM-DD timestamp of the start of the timeseries.	**Start=** Example: 2020-01-01
End YYYY-MM-DD timestamp of the end of the timeseries.	**End=** Example:2021-01-01





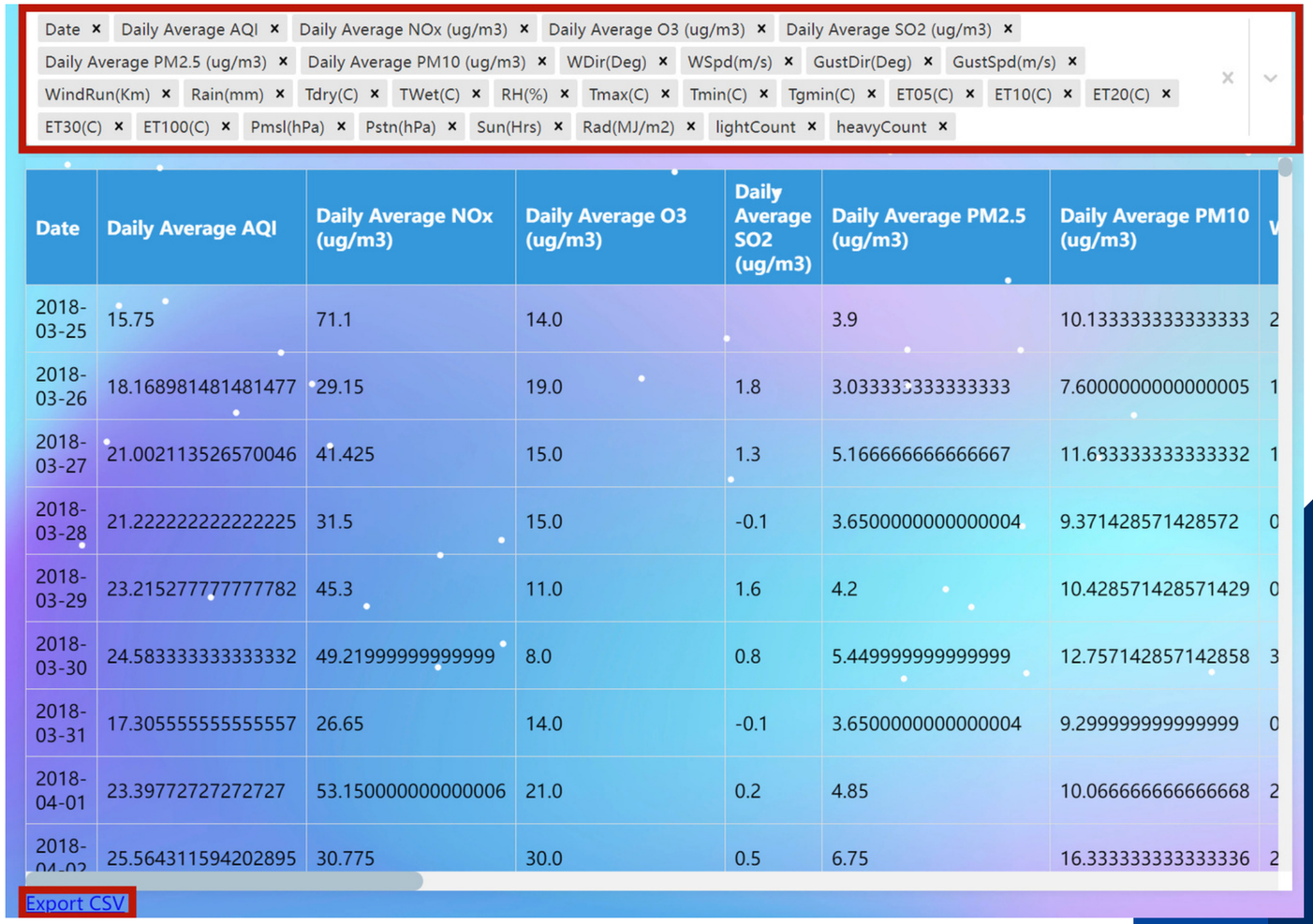
Output Result


```
{
  "Daily Average AQI": 36.01388888888889,
  "Daily Average NOx (ug/m3)": 20.98,
  "Daily Average O3 (ug/m3)": 44,
  "Daily Average PM10 (ug/m3)": 16.47142857142857,
  "Daily Average PM2.5 (ug/m3)": 7.8,
  "Daily Average SO2 (ug/m3)": 0.2,
  "Date": "2020-01-01",
  "ET05(C)": "-",
  "ET10(C)": "19.7",
  "ET100(C)": "19.1",
  "ET20(C)": "19.6",
  "ET30(C)": "-",
  "GustDir(Deg)": "224",
  "GustSpd(m/s)": "8.2",
  "Pmsl(hPa)": 1018.6,
  "Pstn(hPa)": 1016.7,
  "RH(%)": "79.0",
  "Rad(MJ/m2)": 19.03,
  "Rain(mm)": 0,
  "Sun(Hrs)": "6.9",
  "TWet(C)": "16.6",
  "Tdry(C)": 18.8,
  "Tgmin(C)": "9.3",
  "Tmax(C)": 23.2,
  "Tmin(C)": 13.2,
  "WDir(Deg)": "317",
  "WSpd(m/s)": "1.8",
  "WindRun(Km)": "270",
  "heavyCount": 358.25,
  "lightCount": 8631.990267639903
}
```

Customize Dataset

Flexible Feature Selection

Support Exporting



Date	Daily Average AQI	Daily Average NOx (ug/m3)	Daily Average O3 (ug/m3)	Daily Average SO2 (ug/m3)	Daily Average PM2.5 (ug/m3)	Daily Average PM10 (ug/m3)	W
2018-03-25	15.75	71.1	14.0		3.9	10.133333333333333	2
2018-03-26	18.168981481481477	29.15	19.0	1.8	3.0333333333333333	7.6000000000000005	1
2018-03-27	21.002113526570046	41.425	15.0	1.3	5.166666666666667	11.693333333333332	1
2018-03-28	21.222222222222225	31.5	15.0	-0.1	3.6500000000000004	9.371428571428572	0
2018-03-29	23.215277777777782	45.3	11.0	1.6	4.2	10.428571428571429	0
2018-03-30	24.583333333333332	49.21999999999999	8.0	0.8	5.449999999999999	12.757142857142858	3
2018-03-31	17.305555555555557	26.65	14.0	-0.1	3.6500000000000004	9.299999999999999	0
2018-04-01	23.39772727272727	53.150000000000006	21.0	0.2	4.85	10.066666666666668	2
2018-04-02	25.564311594202895	30.775	30.0	0.5	6.75	16.333333333333336	2

Future Work Of The Project

1. Comparing and trying the methods of Transformer.
2. Further works on the feature engineering.
3. Filtering for better quality datasets.
4. Trying to integrate three cities's datasets into one whole dataset, and build one whole model.