Rain Forecast and Maintenance Optimisation across WA

Project 4 Group 10

Hossein Falsafi, Foluke Daramola, Hieu Lam

Objectives

1. Identify stations providing sufficient data for machine learning and forecasting

2. Simulate rainfall using historical data

3. Develop optimal maintenance routes based on geographical proximity

4. Visualise the station on the map as well as the maintenance route trips

Tools

Data CRUD
(Creat,Read,Update,delete)

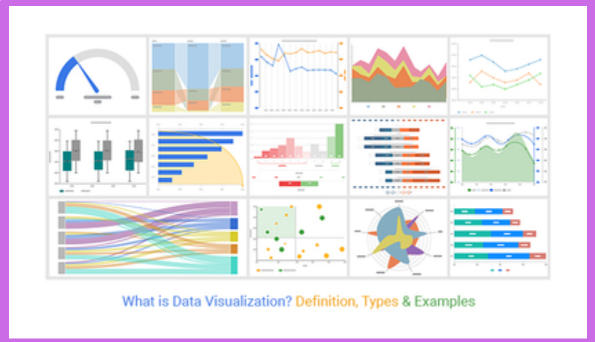
- Python Pandas (Panda, NumPy, Pathlib, API request)
- Python Matplotlib

Rainfall analysis

- Spark
- Neural Networking (TensorFlow / Keras, Tuner)

Clustering of 225 stations from North to South)

- Elbow Method
- K-means clustering algorithm from Scikit-learn



Data Visulisation

- JavaScript
- HTML (D3 Library, Leaflet, Plotly)
- CSS

METHODOLOGY

01 COLLECT DATA FROM RELIABLE SOURCES

02 IDENTIFY STATION LOCATIONS

03 EXTRACT DATA FROM STATIONS USING API

04 APPLIED KMEANS CLUSTERING TO CLUSTER STATIONS BASED ON THEIR GEOGRAPHICAL COORDINATES

05 UTILISE MACHINE LEARNING(NN) FOR RAINFALL FORECASTING

06 CONVERT DATA INTO JSON FORMAT

07 DEVELOP A MAP USING HTML AND JAVASCRIPT

Data Required for Analysis

KEY FACTORS

01 RAINFALL(HISTORICAL DATA)

02 HUMIDITY

03 ATMOSPHERIC PRESSURE

04 LOW-PRESSURE SYSTEMS

05 WIND SPEED AND DIRECTION

06 CLOUD COVER

07 WEATHER PATTERNS

08 GEOGRAPHICAL COORDINATE

CRUD

DATA RETRIEVAL

Data source:[WA Government Department of Primary Industries and Regional Development]

https://weather.agric.wa.gov.au/

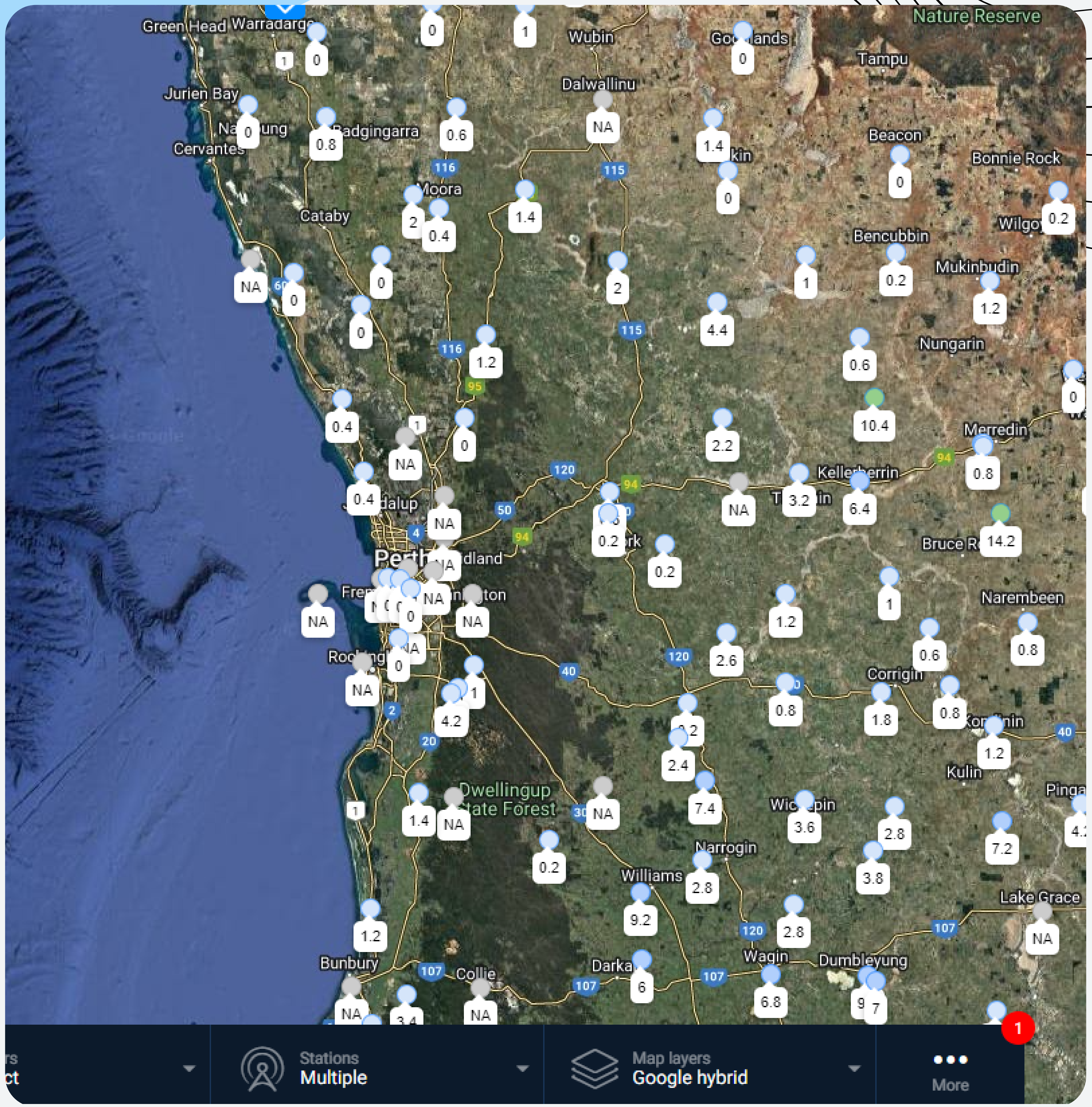


Fig 1. Image Showing

API

The API requests covered a range of factors including covering rainfall, humidity, average air temperature, and average wind speed.

API Key initially incurred charges, prompting our team to reach out to DPPIR.

API KEY

Obtain list of stations and sent API requests for the required rain forecasting data based on that list,

ACQUIRE DATA

Collected 365 days of data from 225 weather stations across WA. Cleaned and saved the data as stationslist.csv, preparing it for machine learning.

DATA COLLECTION

Machine Learning

1-RAIN FORECAST

1-Read and Transform Data: Utilised PySpark to read and transform the locally stored raindata.csv into a PySpark data frame using Spark.

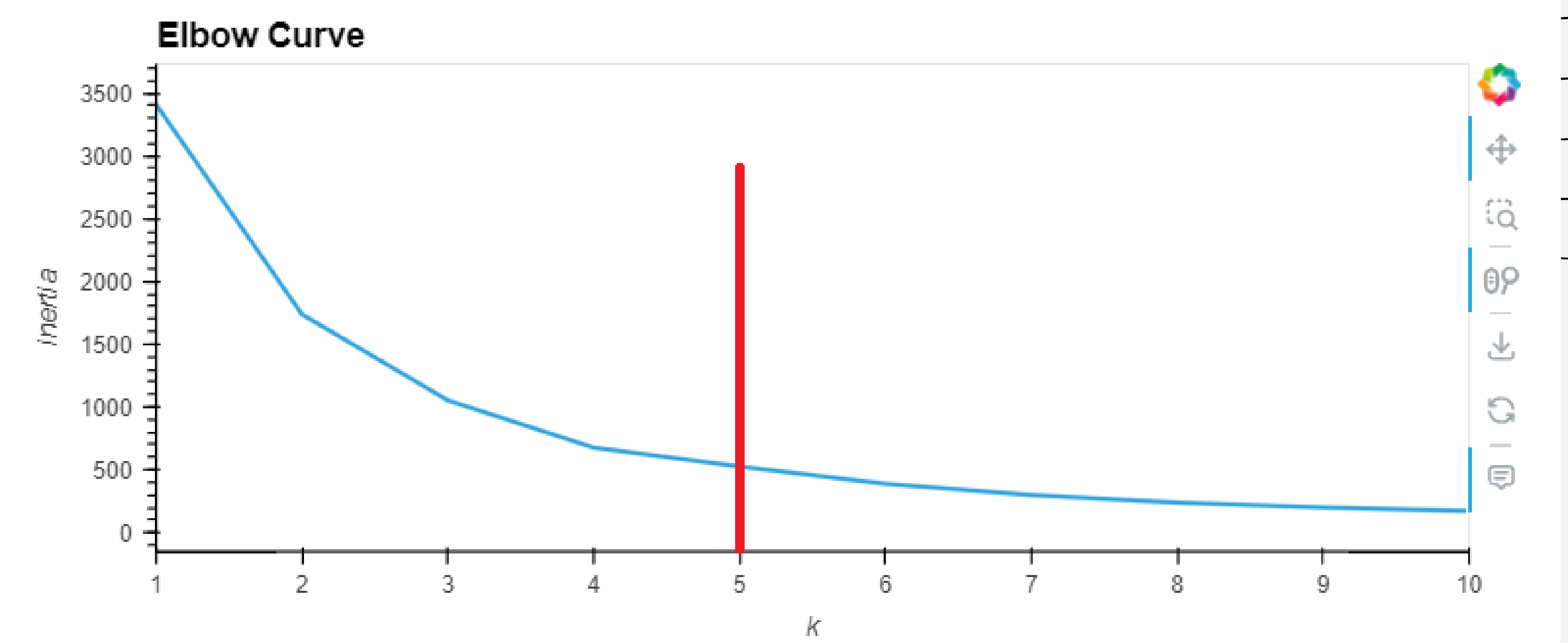
2-Rainfall Prediction Model: Trained a rainfall prediction model for 365 days employing scikit-learn, TensorFlow, and Keras_Tuner.

3- The create_model function defines the neural network architecture for Keras Tuner, allowing customisation of activation functions, neuron counts, and hidden layers. Keras Tuner is initialised with the Hyperband algorithm, using the create_model function. The search is conducted with tuner.search() over 50 epochs, optimizing hyperparameters for enhanced model performance.

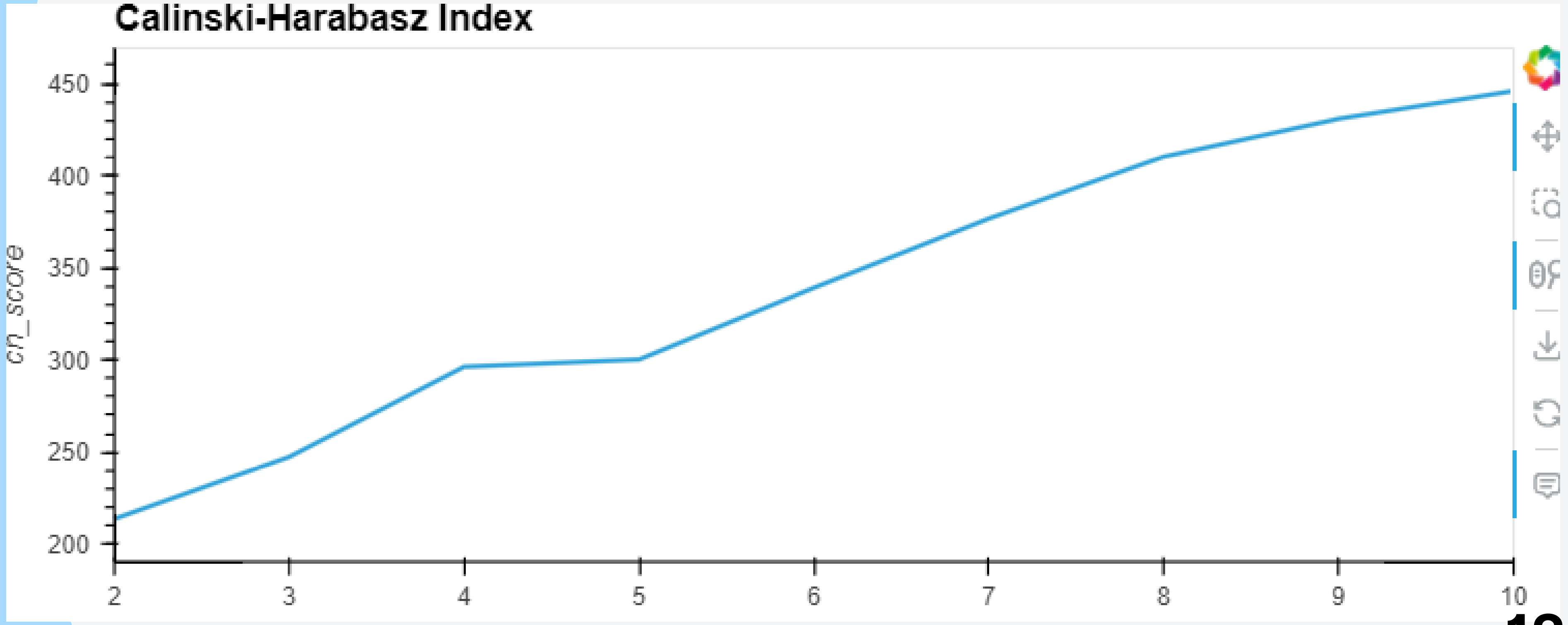
4-Model Accuracy: Despite experimenting with various activations and classifications, the model achieved an accuracy of approximately 64%

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense (Dense)               (None, 1)                 224

 dense_1 (Dense)             (None, 11)                22

 dense_2 (Dense)             (None, 25)                300

 dense_3 (Dense)             (None, 25)                650

 dense_4 (Dense)             (None, 29)                754

 dense_5 (Dense)             (None, 7)                 210

 dense_6 (Dense)             (None, 5)                 40

 dense_7 (Dense)             (None, 9)                 54

 dense_8 (Dense)             (None, 9)                 90

 dense_9 (Dense)             (None, 1)                 10

=================================================================
Total params: 2354 (9.20 KB)
Trainable params: 2354 (9.20 KB)
Non-trainable params: 0 (0.00 Byte)
_________________________________________________________________
```

2-SITE VISIT ROUTE TRIPS



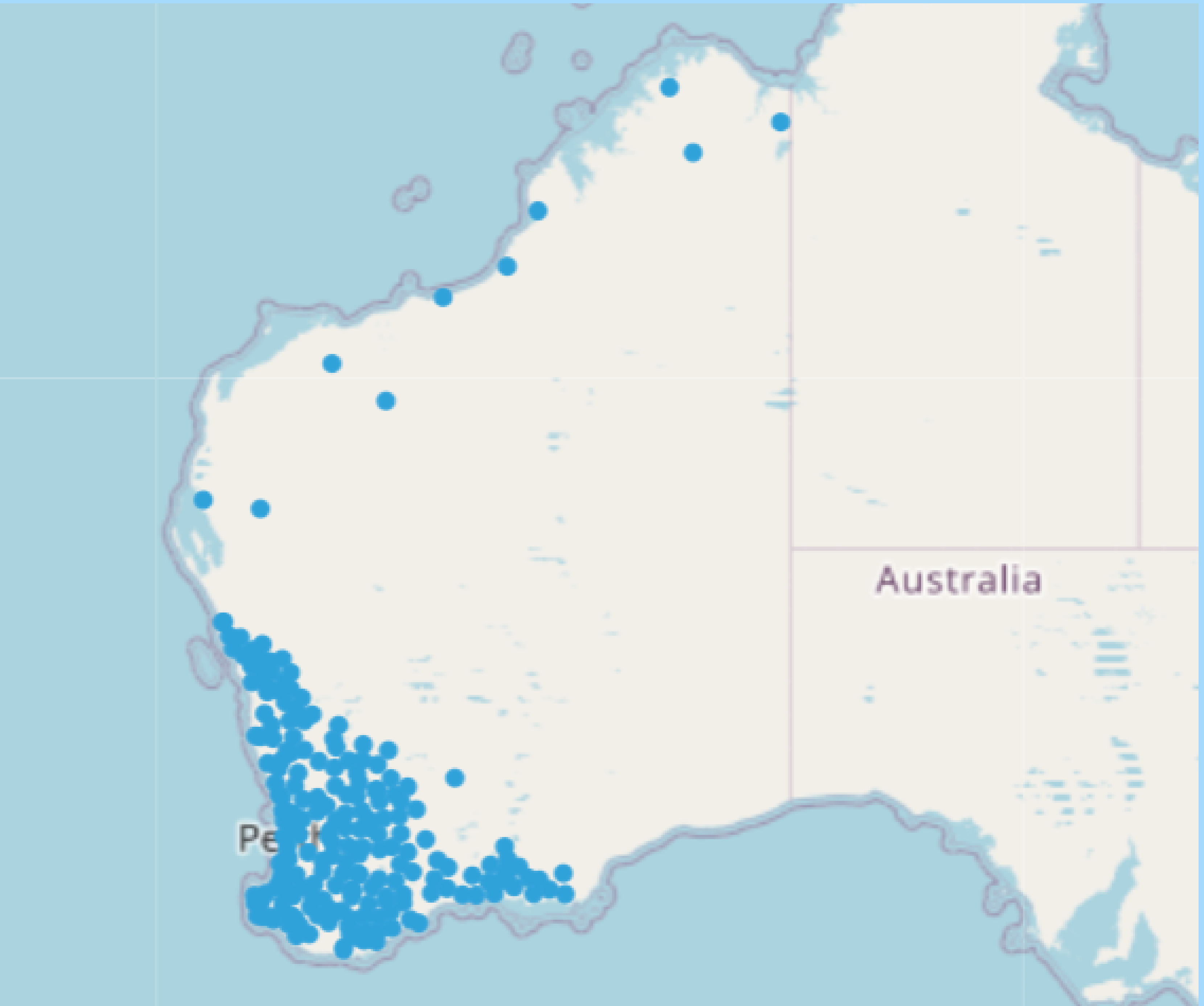
1) Inertia: In this case, K=5 the inertia of 525 suggests that the data points are relatively tight within each of the five clusters.

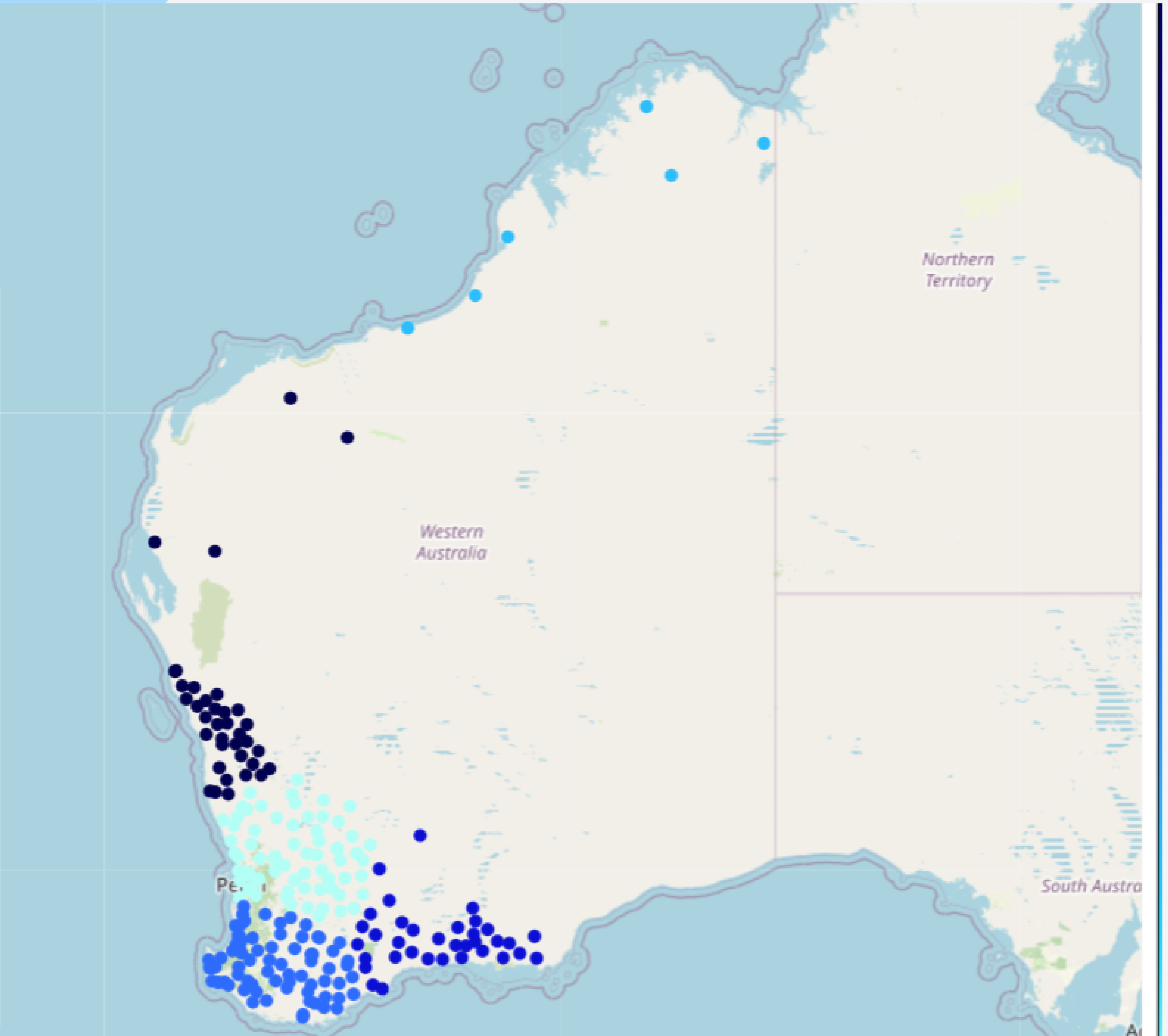


2) Calinski-Harabasz Index:
For the case of K=5 this index is 300. While it is not extremely high, it is still a reasonable value, suggesting that the clusters are distinguishable from each other.

2-SITE VISIT ROUTE TRIPS

After finding the K=5 the best number of clusters, K-means clustering was applied to group weather stations based on their geographical coordinates, yielding predictions of station clusters.



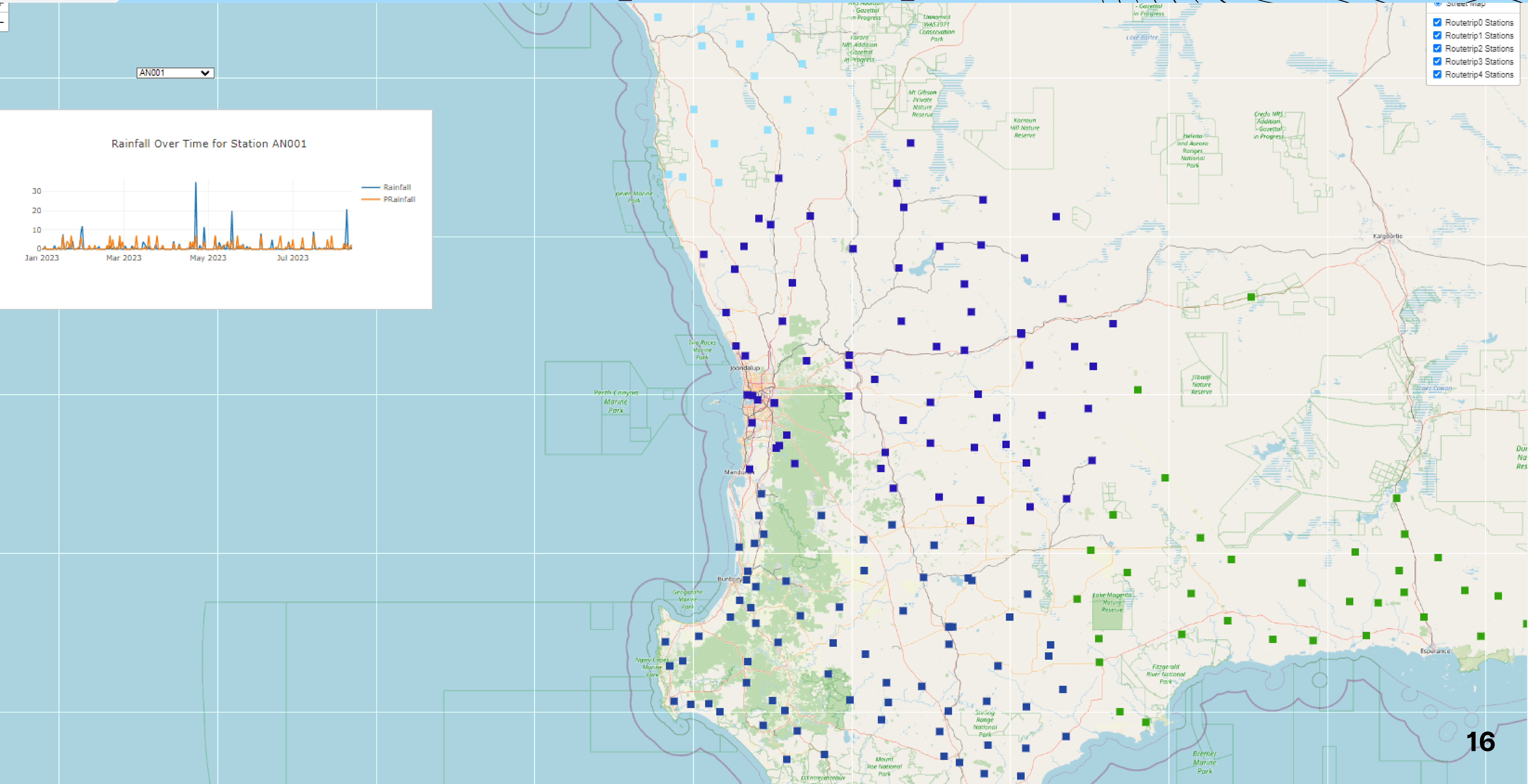


Visualisation

JASONIFY DATA

```
var alldata = {
  "AN001": {
    "stationCode": "AN001",
    "lat": -29.063612,
    "lon": 114.997161,
    "Routetrip": 4,
    "Date": ["2023-01-01",...,"2023-01-02","2023-08-10","2023-08-11","2023-08-12","2023-08-13"
    "Rain": [0.0,0.0,...,0.0,0.0,9.0,0.4,0.0,0.0,0.6,0.2,0.2,3.2,20.6,0.0,0.0,1.0,0.4],
    "PerdictRain": [0.0,0.0,...,0.0,0.0,9.0,0.4,0.0,0.0,1,0.2,0.2,3.2,18,0.0,0.0,1.0,0.4]
  },
  "AM001": {
    "stationCode": "AM001",
    "lat": -34.270827,
    "lon": 118.268523,
    "Routetrip": 2,
    "

};
```

JAVASCRIPT/ HTML/CSS



Conclusion/ Limitations

CONCLUSION

1. We successfully identified 225 weather stations, and our machine learning model achieved a 64% accuracy in rainfall prediction, marking a significant step forward.

2. Leveraging historical data, we acquired daily rainfall information for each weather station over a one-year period, enabling us to simulate rainfall patterns effectively.

3. Utilizing K-Means clustering with a factor of 5, we designed five optimal maintenance routes across Western Australia, considering geographical proximity for efficient station upkeep.

4. Employing mapping tools, we successfully visualized all weather stations on the map, along with the calculated optimal maintenance routes, enhancing the overall accessibility and understanding of our weather station network.

LIMITATION

We aimed to boost the accuracy of the NN model by adjusting key settings, like :
1- Activation functions ('relu', 'tanh', 'sigmoid'),
2- The number of neurons changed in the first layer (1-30, step 2), hidden layer count (1-20),
3- Output activation, loss compile function changed (binary cross-entropy loss)
, and
4- Training epochs (50-100)

However, accuracy didn’t improve more than 64% due to the data limitations. The dataset provided by the DPIRD only included 4 out of all the required key factors.