



NTUC LHUB CAPSTONE
SCTP ASSOCIATE AI ML DEVELOPER

**SINGAPORE
DENGUE FEVER**

PROJECT DONE BY MUHAMAD FADZIL BURHAN

CONTENT

01

INTRODUCTION AND OVERVIEW

02

TOPIC RATIONALE SELECTION

03

SYSTEM ARCHITECTURE AND DESIGN

04

CODING AND IMPLEMENTATION HIGHLIGHTS

05

TIMELINE, TESTING AND VALIDATION RESULTS

06

VISUALIZATION DEMONSTRATION

07

Q&A

INTRODUCTION



This project focuses on analyzing Singapore's dengue fever situation using NEA datasets. Dengue is a serious public health issue, and our objective is to analyze trends, predict outbreaks, and provide insights that can help in preventive measures.

OVERVIEW



By applying data-driven approaches, including machine learning and statistical analysis, we aim to identify patterns in dengue cases, predict future outbreaks, and highlight high-risk areas. Our findings will be visualized to support public health officials and policymakers in making informed decisions to better combat dengue in Singapore.



TOPIC RATIONALE SELECTION

DENGUE FEVER WAS CHOSEN AS THE TOPIC FOR THIS ANALYSIS DUE TO ITS SIGNIFICANT IMPACT ON PUBLIC HEALTH IN SINGAPORE. AS A TROPICAL COUNTRY, SINGAPORE IS PARTICULARLY SUSCEPTIBLE TO DENGUE OUTBREAKS. THE DISEASE POSES A SERIOUS THREAT TO THE POPULATION, LEADING TO HOSPITALIZATION, MORTALITY, AND ECONOMIC LOSSES.

Objective n° 1

Identify key factors associated with dengue outbreaks in Singapore: Explore the relationship between dengue cases and various environmental, socio-economic, and demographic factors.



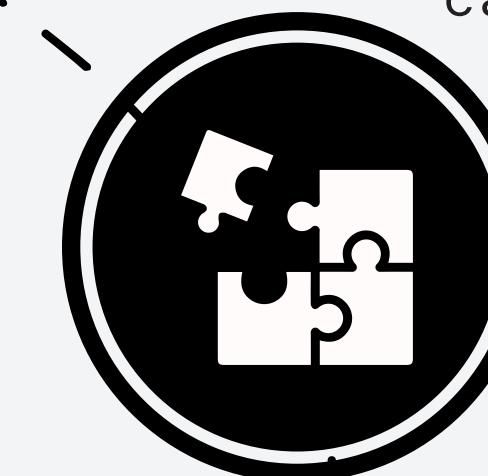
Objective n° 2

Develop predictive models for dengue outbreaks: Utilize machine learning techniques to create models that can forecast dengue cases based on historical data and relevant predictors.



Objective n° 3

Assess the effectiveness of dengue prevention and control measures: Evaluate the impact of government initiatives, such as mosquito control programs and public health campaigns, on reducing dengue incidence.



SYSTEM ARCHITECTURE AND DESIGN

Mission						
0	1	choa chu kang avenue 2	1.378437	103.743747	1	
1	30	choa chu kang avenue 2 (block 296)	1.377421	103.740759	1	
2	25	choa chu kang avenue 2 (block 296a)	1.377133	103.740941	1	
3	23	choa chu kang avenue 2 (block 296b)	1.376838	103.741413	1	
4	13	choa chu kang avenue 2 (block 296c)	1.376565	103.742240	1	
...	
394	1	telok blangah heights (block 61)	1.275763	103.811716	81	
395	1	towner road (block 103)	1.320730	103.861588	82	
396	1	whampoa west (block 34)	1.320412	103.863726	82	

- Our mission is to create a robust data pipeline that processes NEA dengue data efficiently, enabling accurate predictions and actionable insights. By leveraging Jupyter Notebooks, machine learning models, and effective data cleaning and processing techniques, we aim to improve public health outcomes by providing reliable tools for predicting dengue outbreaks in Singapore.

- Our vision is to empower decision-makers with data-driven insights and visualizations that lead to more informed and proactive responses to dengue outbreaks. By building a transparent, scalable system, we aspire to make dengue prediction accessible and impactful, ultimately reducing the incidence and severity of dengue in Singapore.

Vision			
NUMBER OF ROWS : 9			
[8]:			
# Dataset Info			
df.info()			
<class 'pandas.core.frame.DataFrame'>			
RangeIndex: 399 entries, 0 to 398			
Data columns (total 9 columns):			
#	Column	Non-Null Count	Dtype
0	Number_Of_Cases	399 non-null	int64
1	Street_Address	399 non-null	object
2	Latitude	399 non-null	float64
3	Longitude	399 non-null	float64
4	Cluster_Number	399 non-null	int64
5	Recent_Cases_In_Cluster	399 non-null	int64
6	Total_Cases_In_Cluster	399 non-null	int64
7	Date	399 non-null	int64

CODING AND IMPLEMENTATION HIGHLIGHTS

Data Sourcing, Cleaning, and Processing

- Data Sourcing: NEA data is pulled from public APIs or downloaded as CSV files. This data includes both historical dengue cases and environmental data.
- Data Cleaning: In this step, we handle missing values, inconsistencies, and outliers. For example, incomplete records are either filled using interpolation or removed to ensure the quality of the data.
- Data Processing: This involves feature extraction, where relevant variables such as weekly dengue cases, weather patterns, and locations are extracted. We then perform normalization or scaling as needed to prepare the data for modeling.

Design Choices

- Machine Learning Models: For predicting dengue outbreaks, we use machine learning models such as Linear Regression, Decision Tree, Adam, Scikit Learn and etc. These models are selected for their ability to handle non-linear relationships and complex interactions between variables.
- Data Storage: Data is stored in a relational database for easy querying and management. For scalability, we also consider cloud storage solutions, enabling larger datasets to be processed efficiently.
- Visualization Tools: For data visualization, we use tools like Python ipynb libraries. These platforms allow us to create interactive dashboards with heat maps, trend lines, and predictive charts, making the insights accessible to users without technical expertise.

Data Flow

1. Ingestion: Raw data from NEA is collected.
2. Cleaning: The data is cleaned and preprocessed.
3. Modeling: Cleaned data is fed into machine learning models to predict future dengue outbreaks.
4. Storage: Processed data and model outputs are stored in databases.
5. Visualization: Results are visualized through dashboards and reports.

TIMELINE



1ST STAGE

Gathered NEA datasets, performed initial data cleaning, and conducted exploratory data analysis (EDA).

EDA

2ND STAGE

Processed data by handling missing values and engineered features for machine learning models.

PREPOCESSING

3RD STAGE

Built and trained models (e.g., Random Forest, Neural Networks) and evaluated their performance using metrics like accuracy, precision, and recall.

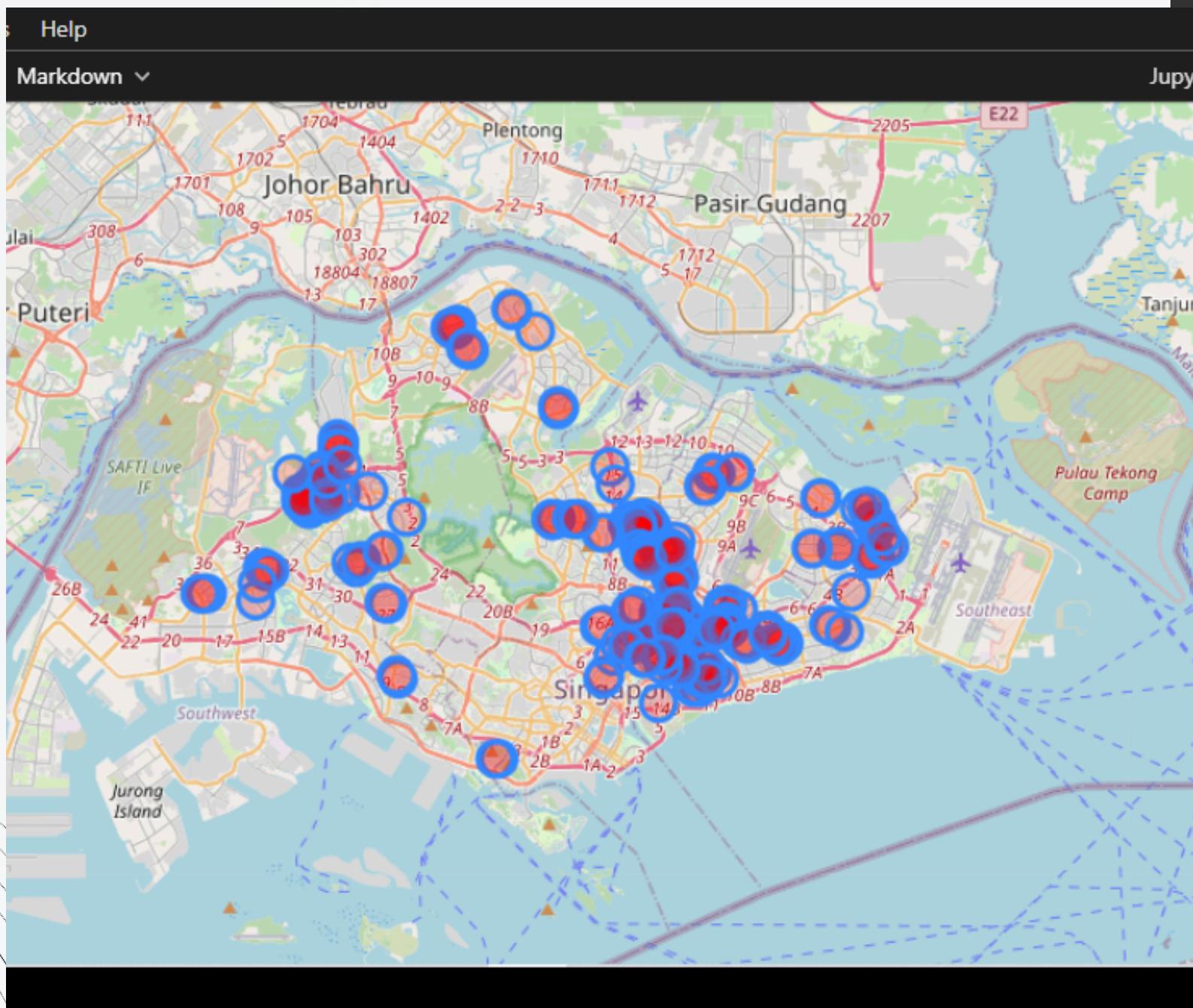
MODELING

4TH STAGE

Finalized visualizations, created dashboards, and prepared the final presentation.

VISUALIZATION

TESTING AND VALIDATION RESULTS



```
[38]: model = Sequential()
model.add(Dense(64, activation='relu', input_dim=X_train.shape[1]))
model.add(Dense(32, activation='relu'))
model.add(Dense(1))

model.compile(optimizer='adam', loss='mean_squared_error')
model.fit(X_train_scaled, y_train, epochs=100, batch_size=32)

y_pred = model.predict(X_test_scaled)
mse = mean_squared_error(y_test, y_pred)
print("Mean Squared Error:", mse)
```

```
Epoch 1/100
10/10 5s 5ms/step - loss: 32569.8594
Epoch 2/100
10/10 0s 5ms/step - loss: 37327.6367
Epoch 3/100
10/10 0s 5ms/step - loss: 38299.4883
Epoch 4/100
10/10 0s 3ms/step - loss: 31007.2383
Epoch 5/100
10/10 0s 4ms/step - loss: 41306.6836
Epoch 6/100
```

```
[34]: dt = DecisionTreeRegressor()
#fitting the score and model score
dt.fit(X_train, y_train)
dt.score(X_train, y_train)

[34]: 1.0

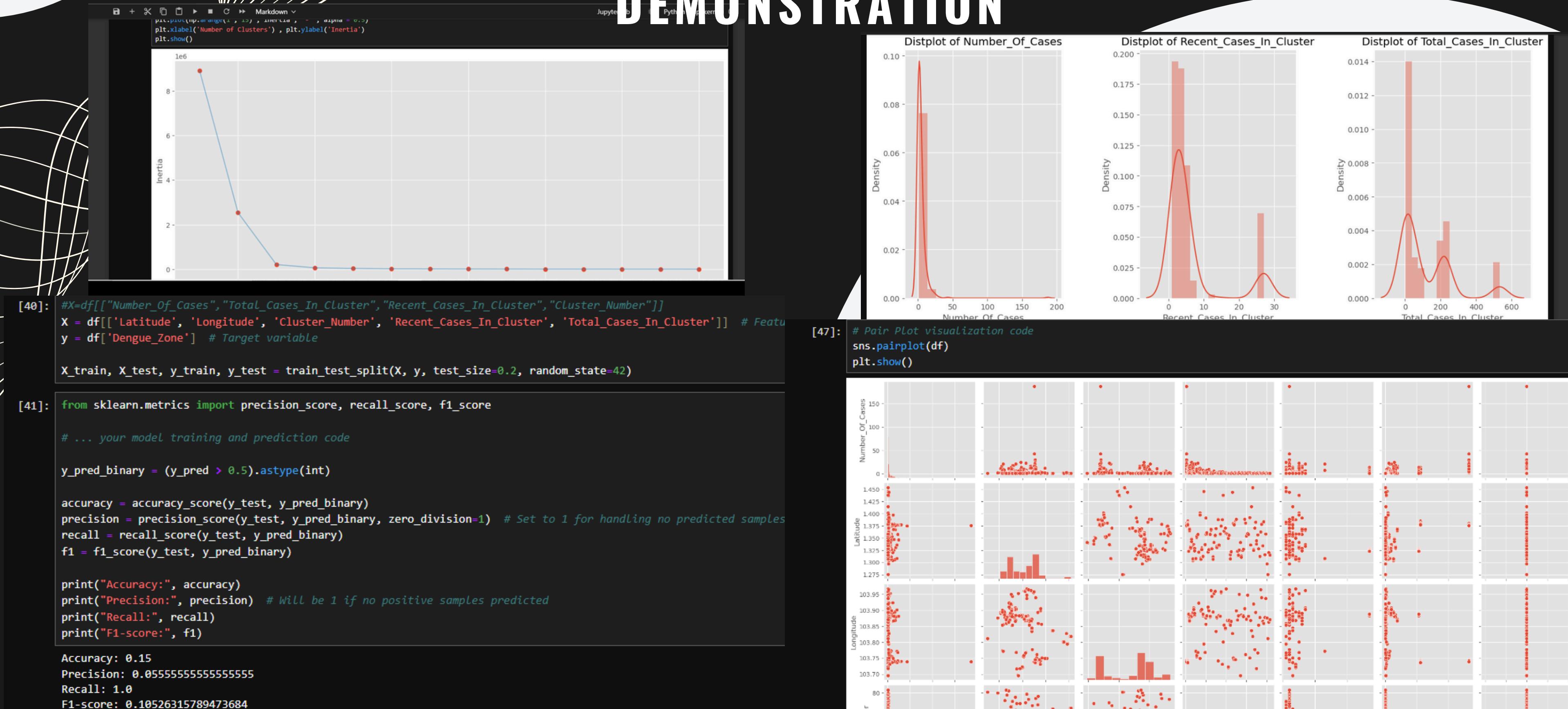
[35]: #prediction of test values
dt.predict(X_test)

[35]: array([ 38.,   3., 225., 37., 225., 225.,   2., 212., 21., 212., 533.,
       3., 225., 225., 212., 225.,   2.,   6., 18.,   2., 225., 533.,
       2., 14., 533., 2., 212., 17.,   9., 212.,   2.,   2.,   5.,
      225., 4., 212., 212.,   9., 77., 212.,   4., 225., 225., 533.,
      225., 2., 77., 58., 533., 65., 225.,   5.,   3., 225.,   3.,
      225., 5., 4., 225., 77., 212.,   58., 65., 225., 212., 225.,
      7., 3., 38., 225., 3., 212., 533., 225.,   2., 11., 21.,
      3., 65., 5.])

[36]: print("The model predicts will be=")
print(dt.predict([[60,500,100,50]])[0])
```

The model predicts will be= 225.0

VISUALIZATION DEMONSTRATION



THANK'S FOR WATCHING

Q&A session

Github:

<https://github.com/mFadzilb/Dengue-clusters-CSV-by-NEA-Singapore>

LinkIn:

www.linkedin.com/in/mfadzilb-82333222