

NUS DATA SCIENCE COMPETITION REPORT

TEAM 14



March 18, 2018

Terrence Tan, Ng MIN MIN, Wang Jie, Tong Hui Kang

Table of Contents

[﷟HYPERLINK "bookmark://\_Toc509139156" ￼](#_Toc509139156)2￼2

[﷟HYPERLINK "bookmark://\_Toc509139157" ￼](#_Toc509139157)3￼3

[﷟HYPERLINK "bookmark://\_Toc509139158" ￼](#_Toc509139158)3￼3

[﷟HYPERLINK "bookmark://\_Toc509139159" ￼](#_Toc509139159)4￼4

[﷟HYPERLINK "bookmark://\_Toc509139160" ￼](#_Toc509139160)4￼4

[﷟HYPERLINK "bookmark://\_Toc509139161" ￼](#_Toc509139161)5￼5

[﷟HYPERLINK "bookmark://\_Toc509139162" ￼](#_Toc509139162)5￼5

[﷟HYPERLINK "bookmark://\_Toc509139163" ￼](#_Toc509139163)5￼5

[﷟HYPERLINK "bookmark://\_Toc509139164" ￼](#_Toc509139164)5￼5

[﷟HYPERLINK "bookmark://\_Toc509139165" ￼](#_Toc509139165)7￼7

[﷟HYPERLINK "bookmark://\_Toc509139166" ￼](#_Toc509139166)7￼7

[﷟HYPERLINK "bookmark://\_Toc509139167" ￼](#_Toc509139167)9￼9

[﷟HYPERLINK "bookmark://\_Toc509139168" ￼](#_Toc509139168)10￼10

[﷟HYPERLINK "bookmark://\_Toc509139169" ￼](#_Toc509139169)10￼10

[﷟HYPERLINK "bookmark://\_Toc509139170" ￼](#_Toc509139170)10￼10

[﷟HYPERLINK "bookmark://\_Toc509139171" ￼](#_Toc509139171)11￼11

[﷟HYPERLINK "bookmark://\_Toc509139172" ￼](#_Toc509139172)11￼11

[﷟HYPERLINK "bookmark://\_Toc509139173" ￼](#_Toc509139173)12￼12

[﷟HYPERLINK "bookmark://\_Toc509139174" ￼](#_Toc509139174)14￼14

[﷟HYPERLINK "bookmark://\_Toc509139175" ￼](#_Toc509139175)15￼15

# Executive Summary

We have made rigorous use of the base model to provide a baseline estimate of the location of the rain gauge. In the process, we have developed a comprehensive data pipeline from the input data to the visualization of the rainfall predictions and the mapping the possible locations of rain gauge.

Halfway into the competition we have found a set of NEA rain gauge. We evaluated the base model w.r.t. available ground truths and the base model is sufficiently accurate in estimating the location of the rain gauge.

Our results are available here:

<https://1drv.ms/f/s!ArT5tVl4yeQBhtZZKKtrryNXgqYnBw>

week{t}.png refers to the week t rainfall predictions by our base model  
station{p}.png refers is the location prediction of the rain gauge p by model b

week{t}\_ab.png refers to the week t rainfall predictions by model a  
station\_ab{p}.png refers is the location prediction of the rain gauge p by model a

a and b are the different models that we use the train the model

# Introduction

Throughout the history of mankind, forecasting weather is the everlasting dream. Precipitation rate relates to many aspects of our daily life. Soil moisture rate, traffic accidence rate and surface hardness are directly affected by the amount of rainfall in the relevant region.

For this competition, we were provided with two main datasets:

1. *Gauge Measurements* - contains historical rain gauge readings for 52 weeks for year 2017, for 50 stations
2. *Backscattered Radar Signal* - contains 31 weeks, with each file containing the (x, y) coordinate, with the number of counts (in 5minute intervals) in 34 discretized bins

In this report, we will explore how we have made use of such data to create our base model as well as our risk and loss functions.

## Given Data

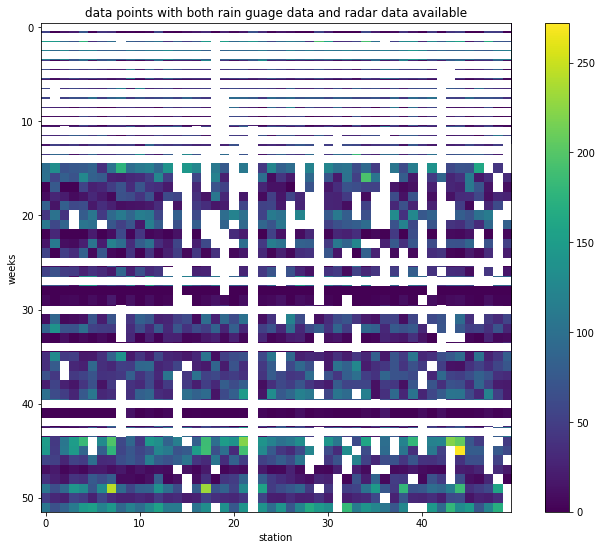
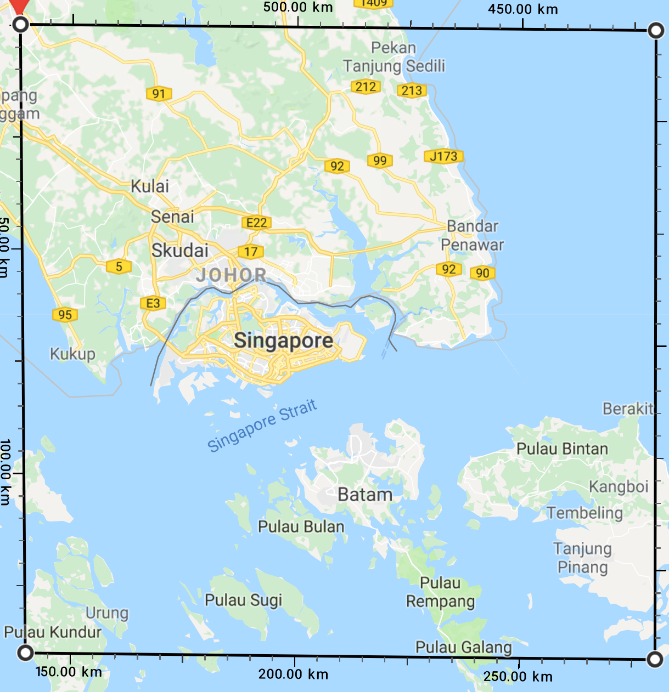
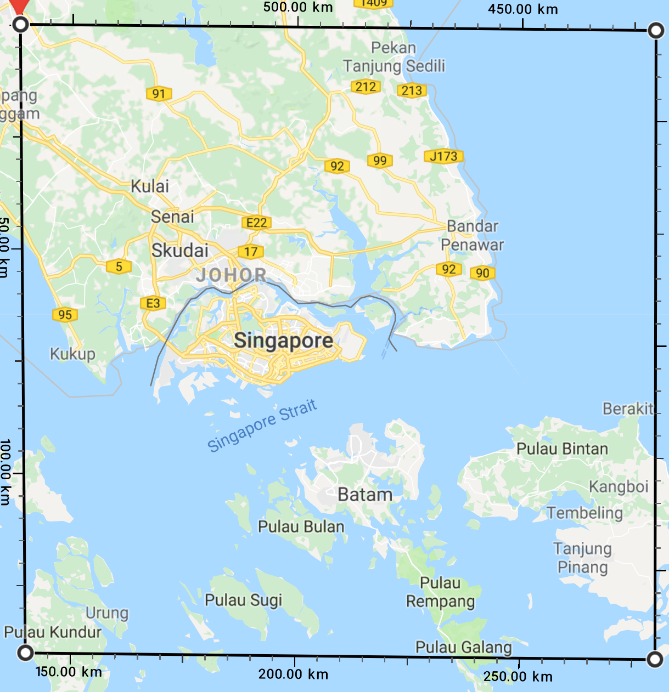


Figure 2.1.1

Figure 2.1.1 shows the sparsity of the dataset, with only 31 weeks of radar data given out of 52 weeks. The white pixels indicate NaN values, while white horizontal lines represent missing information on the particular weeks in the radar dataset.

In addition, we also realized that for one particular rain gauge (i.e gauge 22), it does not have any value collected in the 31 weeks of radar data. This means that we effectively have 49 rain gauges.

  
Figure 2.1.2

Moreover, we were also given information of the radar coverage which was made up of 480 squares by 480 squares, with each square box having a spatial resolution of side 292m. With the further information of the starting left most grid’s location at 1.98 degrees latitude and 103.338 degrees longitude, we derived the coverage area as seen in Figure 2.1.2.

## Data Collected

### Data.gov.sg

Understanding that the locations of the stations would be within Singapore, we scrapped geolocation data on rain gauge using DataGov rainfall API. An issue arise when only rain gauges which detect rainfall will be displayed. This means that there could be a possibility of missing out rain gauges. Figure 2.2.1.1 describes codes to find these missing stations.



Figure 2.2.1.1

### NEA Weather Forecast

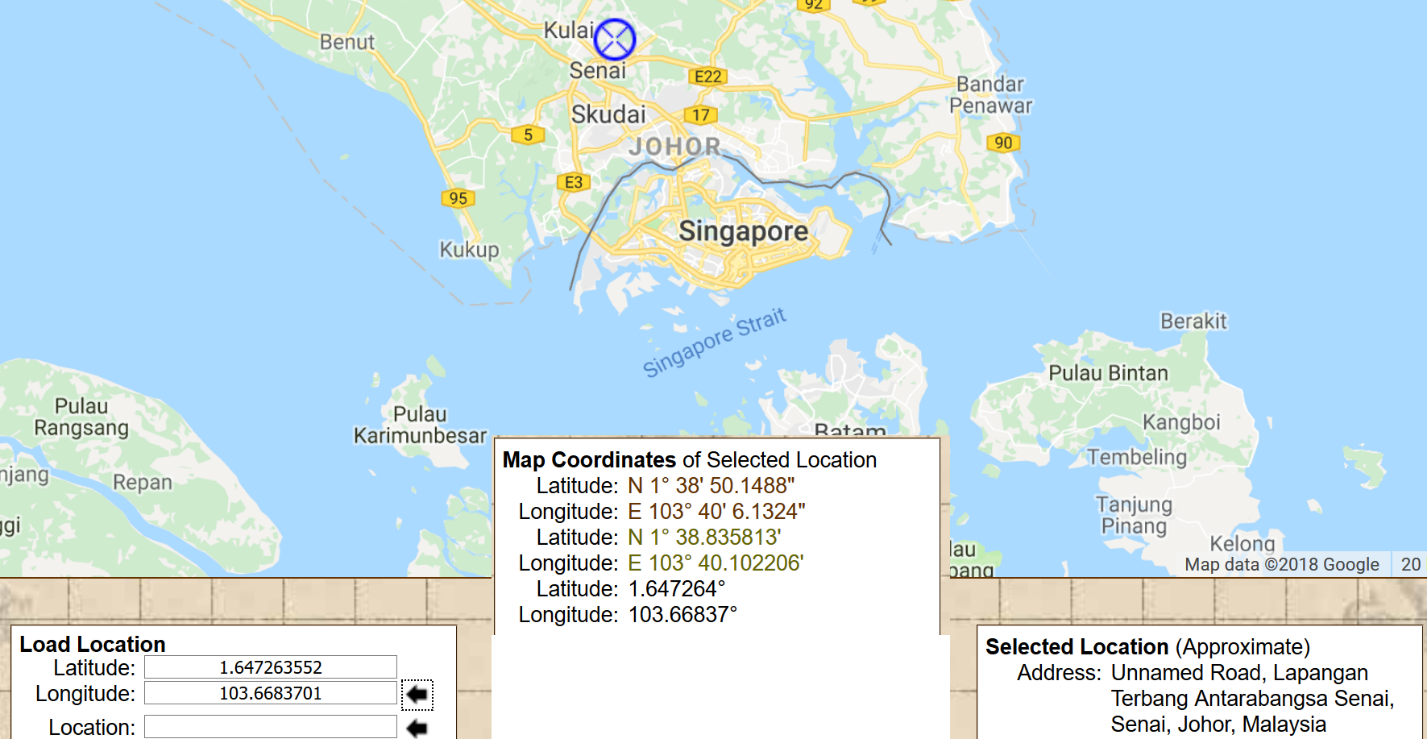
Moreover, we also scrapped rainfall data from NEA to establish ground truth and streamline the search for the exact location of the 50 stations provided.

### Methodology to Convert Distance to Latitude and Longitude

Initially, we tried to get the location of the weather stations by taking the (x,y) value from the data sets. This was done by attempting to convert its distance from the origin of the top left corner that was located at 1.980 deg latitude and 103.338 deg longitude. We managed to convert them by multiplying 292 to their x and y values respectively and applying the following formulas.

Longitude = 103.338 + 292(y)(cos(radian(latitude))/111320  
Latitude = 1.98-(x\*292/110574)

However, most of the points ended up in Malaysia as seen in figure 2.2.3.1.

  
Figure 2.2.3.1

## What we need to give

Expected coordinates of the rain gauge as a csv file. In addition, we feel that is also relevant to describe how close the alternatives are, which is important to illustrate how sure are we of our results.

Defining the Risk/Loss Function

We want to minimize the root mean square error (RSME) of the predicted rainfall and the actual rainfall. We decided on root mean square error because we want to penalize larger differences between the actual and predict rainfall.

For each possible location of the rain gauge, we first take the difference between the predicted rainfall value suggested by the radar bin values and the actual rainfall value measured by the rain gauge.|  
 a = np.add(radars.as\_matrix(),-1\*np.array(actual))  
  
Then we take the root mean square of the difference.  
 b = np.sqrt(np.nanmean(np.power(a,2),axis=1))

To find out the most probable place of the rain guage we find the location where the RSME is the smallest. We are also interested in the RMSE as well.  
 argmin\_ = np.argmin(b)

min\_rsme = np.min(b)

In the handling of the missing data, we ignored them in the calculation. Since we are taking the mean of the errors, the number NaN values will not bias the mean.

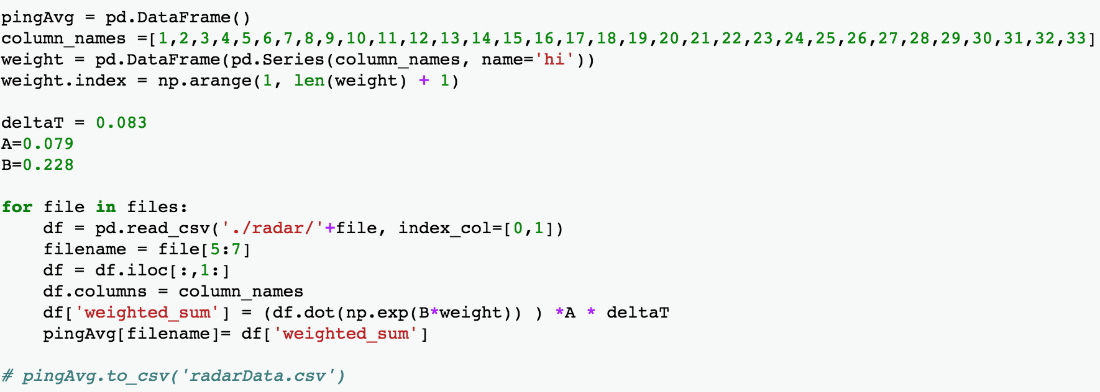
We will still plot for the entire radar plot to show that our method will accurately suggest the possible location of the rain gauge to fall in Singapore. However, in our submission, we will apply a geofence to the possible coordinates, so that our submission makes sense.

Base Model

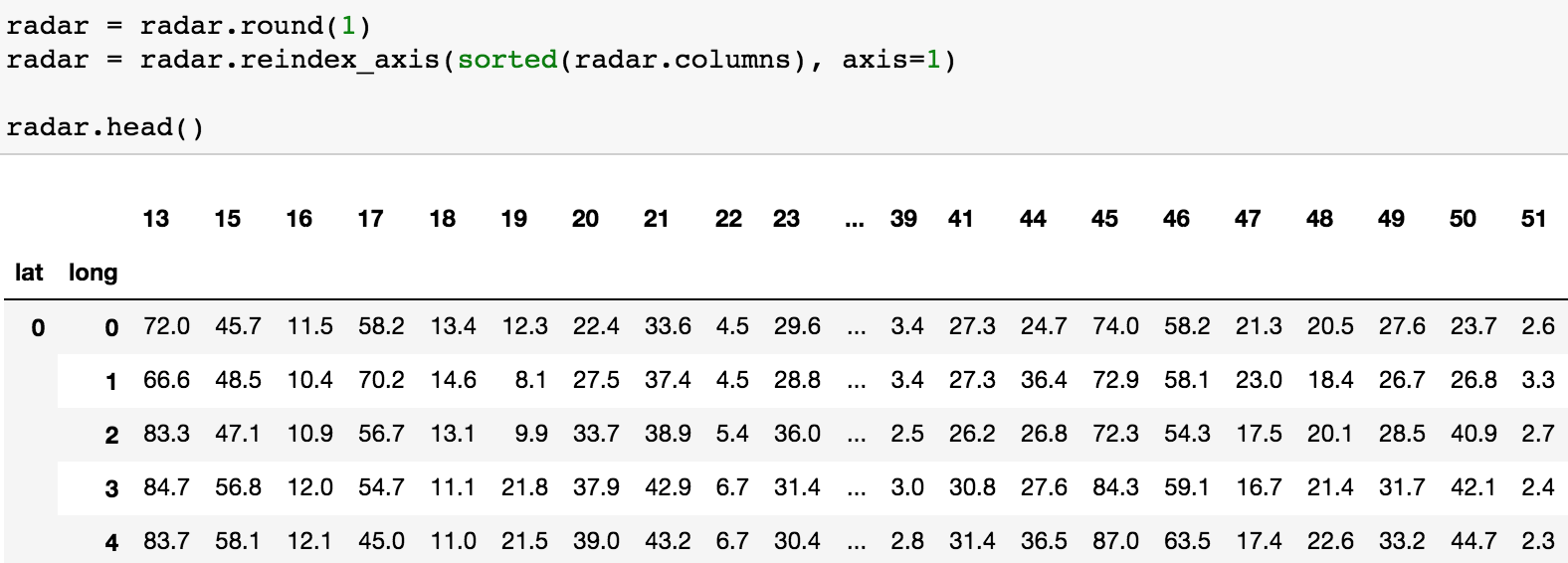
## Methodology

We use the parameters from the competition slides: A=0.079 ,B=0.228

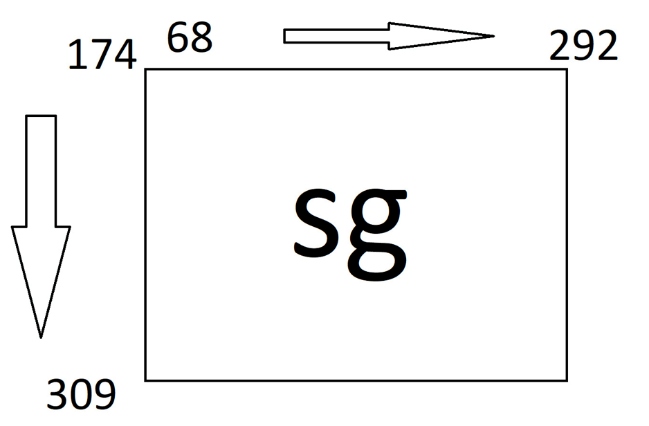
We ignored the bin , as we have found out that it greatly overestimates the amount of rainfall that we will be getting.

  
Figure 4.1.1

Our first step would be to gather all the *R(x,y)* for each of the 31 weeks and save them into another CSV file: radarData.csv. This is achieved by doing a dot product of each dataframe and the exponential function of B and bins, k. An overview of the data set is shown:

  
Figure 4.1.2

One problem surfaced when we simply used the risk function to calculate the (x,y) positions of the rain gauge stations without scaling the possible positions of them in the first place. Hence, we narrowed down the search by acknowledging the possible start and end latitude and longitude respectively of the Singapore map. This is hypothesized as such, defining the latitude range = [68, 292] and longitude range to be [174, 309]:

  
Figure 4.1.3

Using the risk loss function, we can calculate the best possible x and y location by minimizing root mean squared error (defined above). We use JavaScript Google Map API to plot them out on the Singapore maps in Figure 4.1.4.

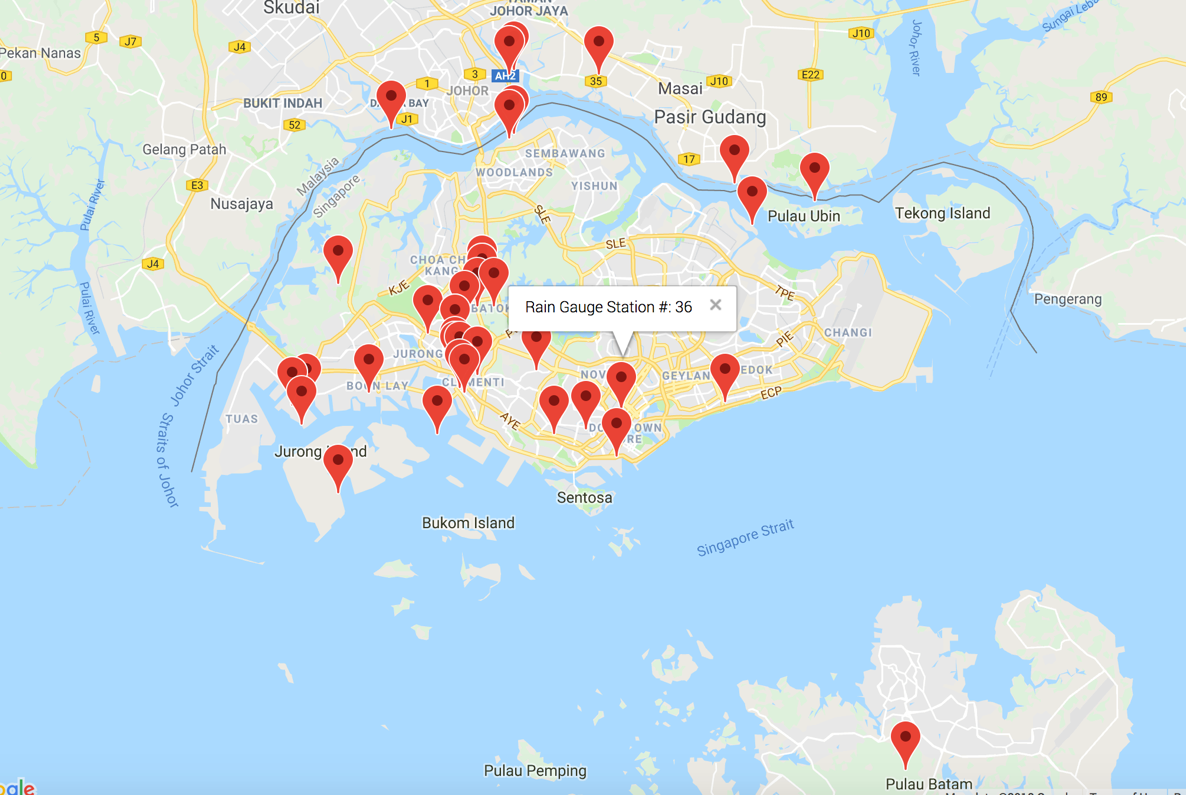
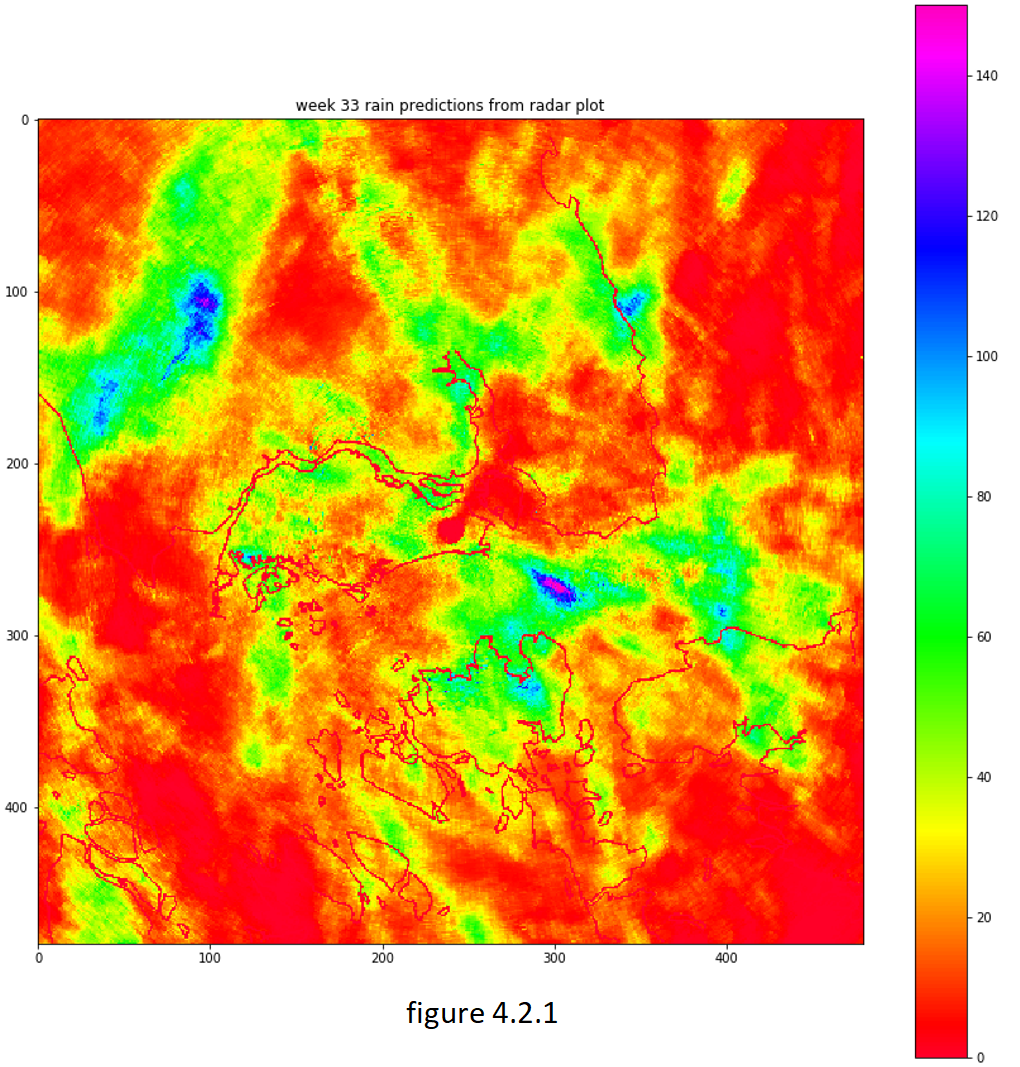
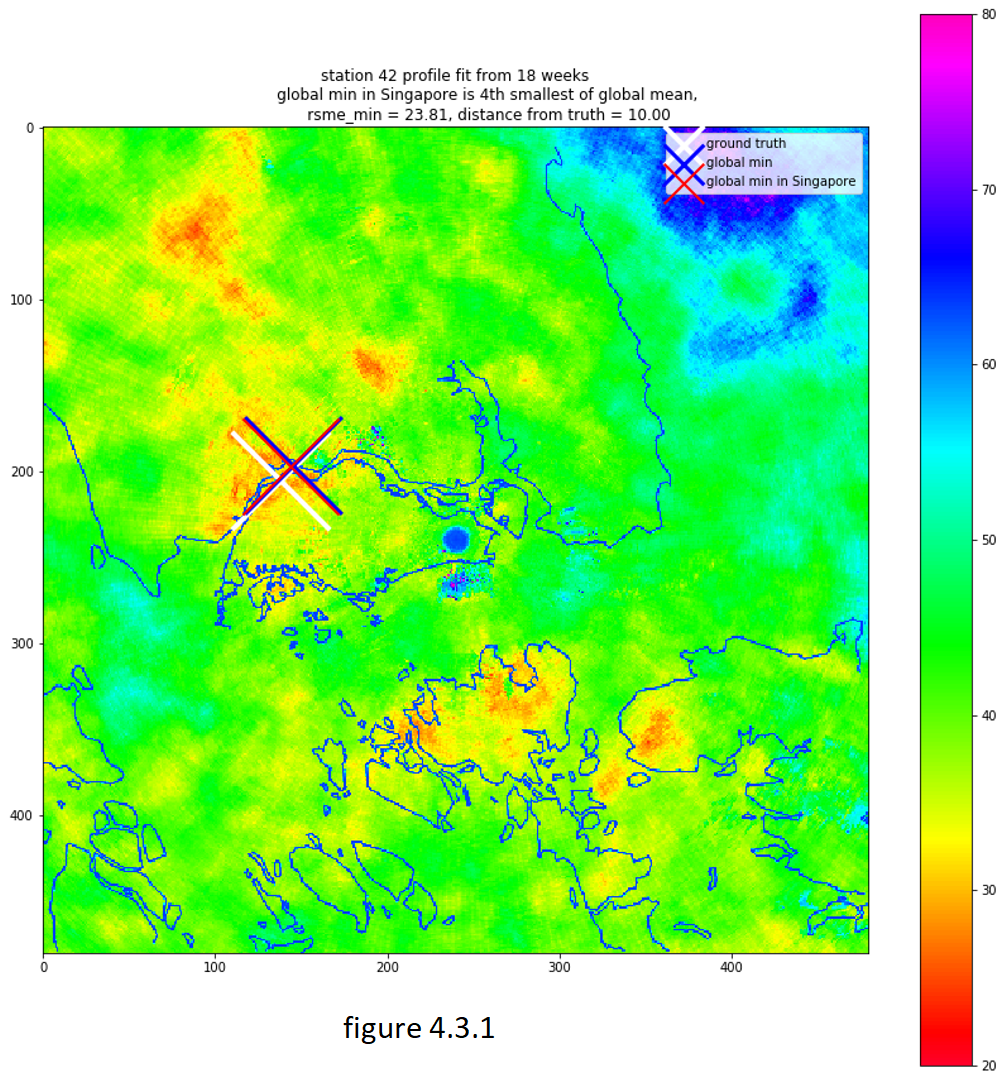


Figure 4.1.4

## Results - Rainfall Prediction



## Results - Location Prediction



Minimum loss is merely only one-fifth of the maximum loss typically. This could be further improved.

Refined model

## Data Preparation

Since radar reading data does not contain null value. We can easily create a table that concatenate all weeks into one complete set. We removed unsure data in gauge measurements, which left us 47 out of 50 into the modelling. We split the data into 7:1:2 (training, validation, testing). The split actually is not rounded from the closest division value as we would like to split the test data according to the station. By isolating a few stations from other, we can prevent potential boost up of model brought by data leakage.

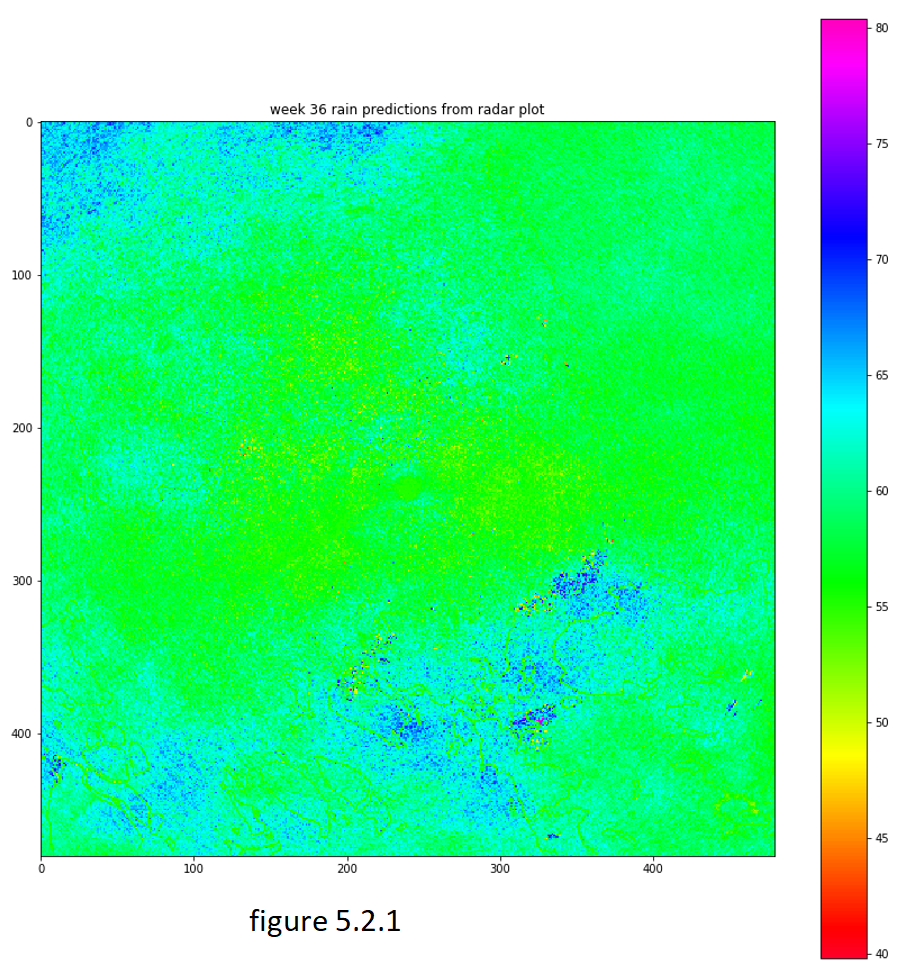
We use NEA’s calibrated data as our ground truth to optimize a model that maps the k-bin array to predicted rainfall. This is the rain prediction data map from the plot of the week 33.

## Rainfall predictions

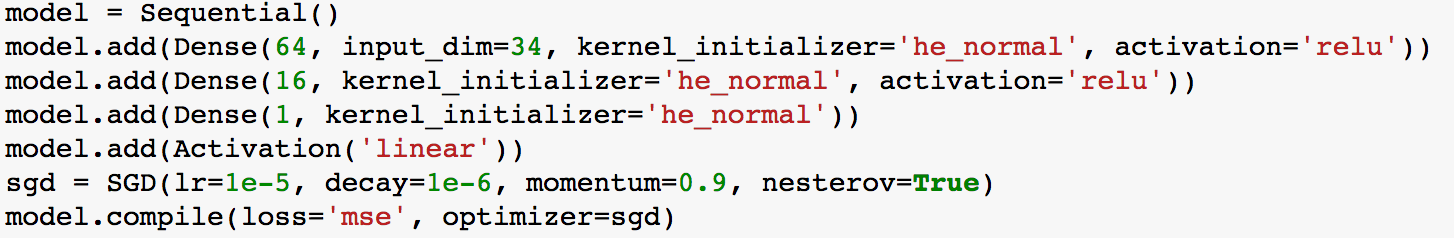
We devised an optimization function adopting a deep learning based approach. We implemented a multi-layer MLP model. The model takes a vector of dimension 34 as input. It contains two fully connected hidden layers with dimensions 64 and 16 respectively. The weights are initialized using HE normalization and the activation function is ReLU. Since we are solving a multi-variant regression problem. We used linear activation function before the singular output layer. The optimization function is using stochastic gradient decent, with learning rate of 1e-5, decay 1e-6, momentum 0.9 and enables Nesterov acceleration. To ensure the consistency we used the same mean square error as the model's loss function.

## Training

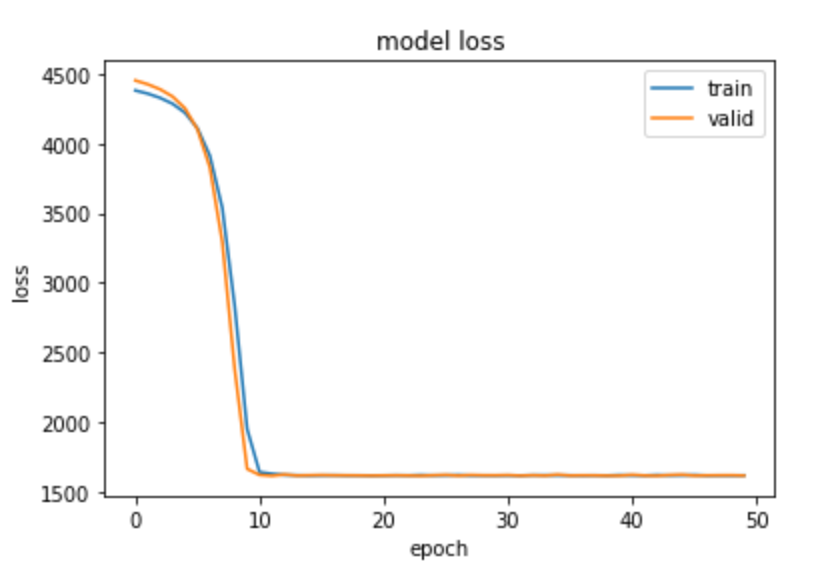
We used 50 epoches with 64 batches as training step.



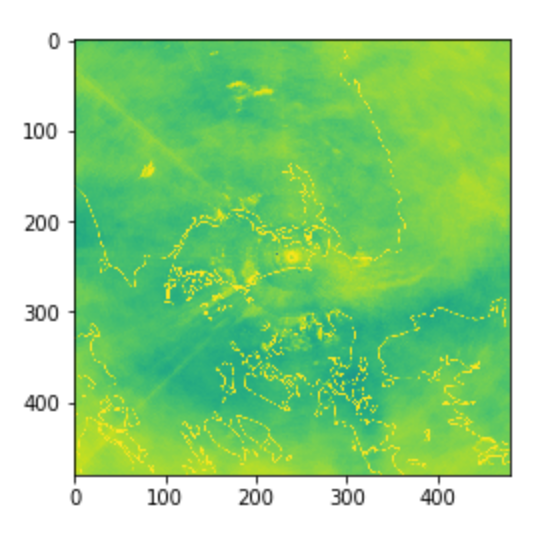
# Results

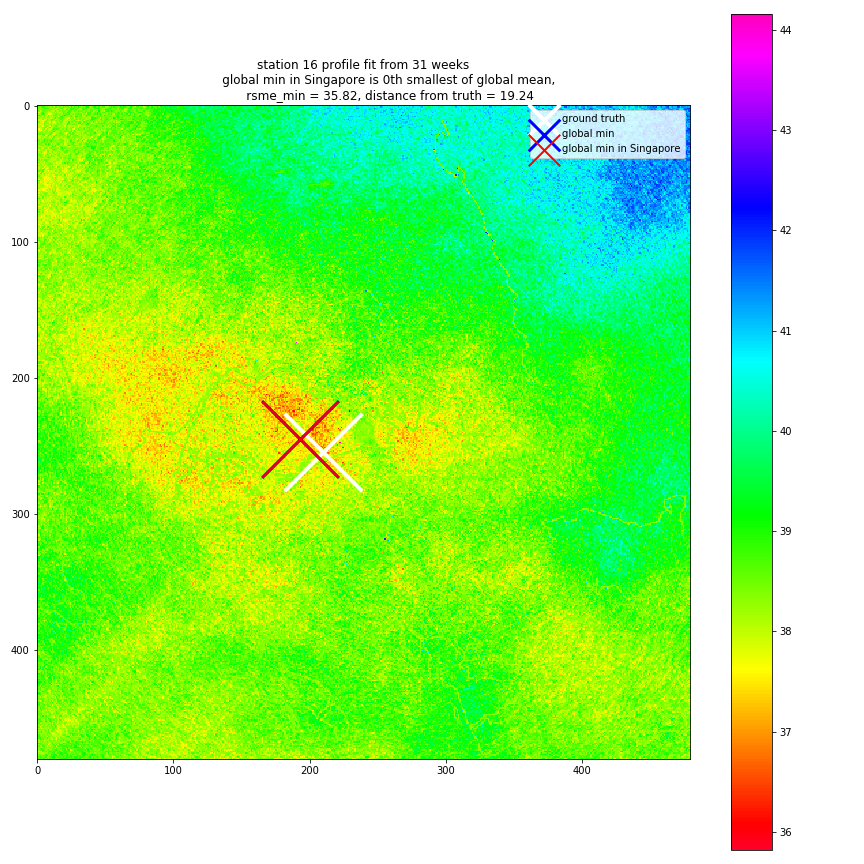
  
Figure 6.1

The training takes very fast as the size of training and testing dataset are small. We obtained a smooth decrease of training and validation loss. We decreased the loss from 2348 MSE down to 1617 for training data, 1615.98 validation data and 1350 for isolated testing data.

  
Figure 6.2

The predicted full-slide image can be sketch the boundaries of islands and oceans correctly.

  
Figure 6.3

  
Figure 6.4

Conclusion

In this work, we determined the geolocations of nearly all stations (47 out of 50) with the help from external data source such as NEA and official Singapore government data portal. We evaluated the performance of provided linear mapping function and interpreted it using visualization tools. Furthermore, we proposed a deep neural network-based optimization approach and implemented using modern deep learning framework Keras. After achieving promising result, we compared various of options and highlighted our contribution to the given task set.

Appendix

Coordinates of all stations:

{

0: (222, 248),

1: (123, 249),

2: (218, 255),

3: (238, 252),

4: (214, 237),

5: (159, 245),

6: (245, 231),

7: (171, 198),

8: (113, 248),

9: (190, 257),

10: (130, 274),

11: (183, 254),

12: (217, 218),

13: (201, 257),

14: (190, 253),

15: (167, 244),

16: (210, 254),

17: (187, 208),

18: (186, 276),

19: (186, 230),

20: (146, 229),

21: (178, 230),

22: (202, 234),

23: (234, 232),

24: (161, 226),

25: (180, 241),

26: (135, 229),

27: (183, 250),

28: (169, 260),

29: (201, 243),

30: (198, 249),

31: (143, 238),

32: (171, 263),

33: (141, 254),

34: (209, 242),

35: (227, 224),

36: (193, 257),

37: (149, 227),

38: (231, 240),

39: (165, 230),

40: (202, 265),

41: (194, 228),

42: (138, 205),

43: (185, 213),

44: (171, 198),

45: (158, 264),

46: (106, 259),

47: (188, 241),

48: (239, 213),

49: (163, 299)

}