TU Clausthal

# Analyse Project


Autralia Weachter 2020

Informatik M.Sc.

Shu Huichen: 515681

## 1.1 The Distribution of Features

Before analyzing, we should check the attributes of the data set, namely Location, MinTemp, MaxTemp, Rainfall, Evaporation, Sunshine,WindGustSpeed, WindSpeed9am, WindSpeed3pm, Humidity9am, Humidity3pm, Pressure9am, Pressure3pm, Cloud9am, Cloud3pm, Temp9am, Temp3pm, latitude, and longitude. With these 19 attributes, we can predict whether it rain today or tomorrow or not. In this report, the author has used these 17 attributes without Latitude and Longitude to predict rain today or not.

## 1.2 The Preprocesse of Given Dataset

### 1.2.1 Preprocesse of Date attribute

The DATE attribute describes the collected data, which starts from Jan 1, 2013, to Jul 20, 2018, with 2036 days in total. Only samples from 2013 to 2015 contains the 12 moths records. Observations from 2016 are rare, as for from 2017 or from 2018 are without from 2 or 3 months. During analyzing, text values from the date attribute can raise errors, so that the author has transformed string to timestamp.

### 1.2.2 Preprocesse of String attribute

The LOCATION attribute describes the collected places, which come mainly from Perth with 1592 samples. Because string value can not be analyzed in python, the following dictionary is the presentation of the original 6 locations, namely "{'WaggaWagga':1, 'AliceSprings':2, 'Sydney':3, 'Melbourne':4, 'Darwin':5, 'Perth':6, 'Portland':7, 'Brisbane':8} :".

### 1.2.3 Others

Other attributes about parameters of temperature, wind, pressure, humidity, and cloud at different times of a day, are all continuous value. For example, the maximum temperature varies from different places. The distribution of all attributes is shown in figure 1.

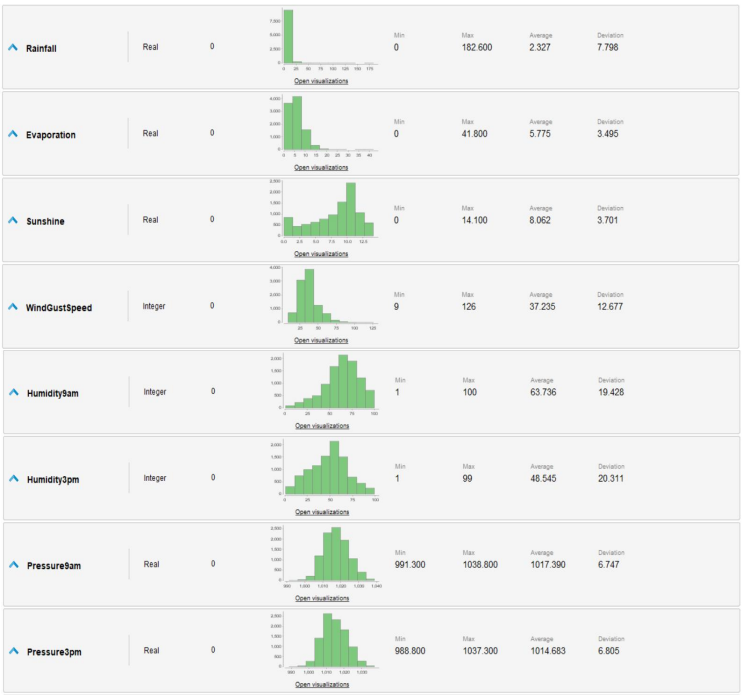


Figure 1(a)


Figure 1(b)

The distribution of maximum and minimum temperatures, the one of humidity and pressure at the same clock, and of cloud all day, are all similar. According to the rough description of features, we can already see some correlations between features. Whether the characters of features show certain forms? We need more further preprocess.

The reason why the author does not use attributes of LATITUDE and LONGITUDE is that the location contains already similar information. In another world, it is acceptable to use only one of them, namely only use LATITUDE and LONGITUDE or only LOCATION. In this report, only the LOCATION is used.The attribute about the date is usually a symbol for time series issues, in which data change with the time so that standard prediction methods may perform poorly. Consequently, it is necessary to check the assumption that this is a time series issue.

## 1.2.4 Correlations of Features

A significant requirement for time series issues is that there are high correlations between features. So through "corr()" and "heatmap()" functions in python the correlations are in figure 2.

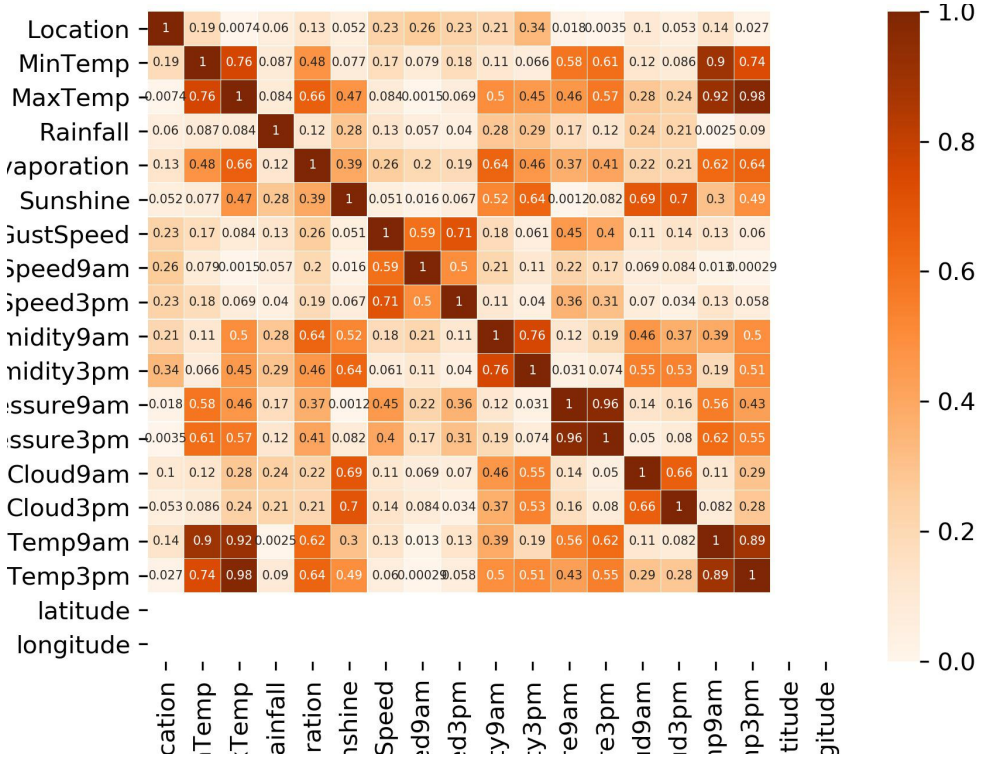


Figure 2 Correlations Heatmap

As the distribution of correlations in figure 1, we can see that nearly all the attributes have a

higher level with each other. Pearson correlation coefficient between HUMIDITY9AM and PRESSURE9AM, between HUMIDITY3PM and PRESSURE3PM, between CLOUD9AM and CLOUD3PM, TEMP9AM with TEMP3PM and SUNSHINE are all higher than 0.3 but lower than 0.8. Namely, they have weak correlations with each other. Further more, the Pearson between MAXTEMP and TEMP3PM, between MINTEMP and TEMP9AM, between MAXTEMP and MINTEMP, have strong correlations with each other. Based on the common sense of life and certain physical foundations, we can know that this result is taken for granted. So we need more evidences to approve the assumption that features do have some relationships with each other.

### 1.2.5 Dimensionality Reduction Algorithm: Isomap

To support that further, the visualization of attributes' distribution with time is essential. Therefore, the reduce dimensional method Isomap is used in python, like in figure 3.

```
from sklearn.manifold import Isomap
embedding = Isomap(n_components=3)
X_transformed = embedding.fit_transform(prob)
```

Figure 3

Traditional methods, such as PCA and MDS, are not very efficient. Because the original data space is a two-dimensionally distributed plane, the distance between points on the famous body in three-dimensional space cannot be calculated using the distance of the 2-norm, namely traditional European space—the actual distance of the points, but much completer norm. The norm of this space, namely the distance of this space is count on the function.

$$\|x\|_\infty = (|x_1|^\infty + |x_2|^\infty + \ldots + |x_n|^\infty)^{1/\infty}$$

A concrete example for actual distance is the geodesic distance. We construct a connected graph, where each point is directly connected to the only k points closest to this point, and not directly connected to other points. In this way, we can construct the adjacency matrix, and then find the shortest path of any two points in the figure instead of the geodesic distance.

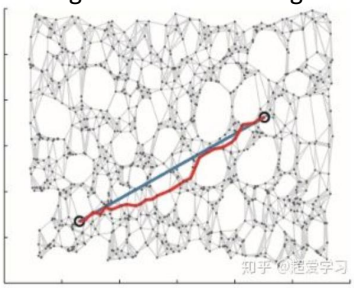

Figure 4

In Figure 4, the blue point represents the geodesic distance between the two points, and the red line represents the shortest path between the two points in the figure. The distance between the two is similar, so we use the latter instead of the former. When we get the distances of a point to x-, y-, and z-axis, then we got the distribution of this issue in three-dimensional axes, like in figure 5.

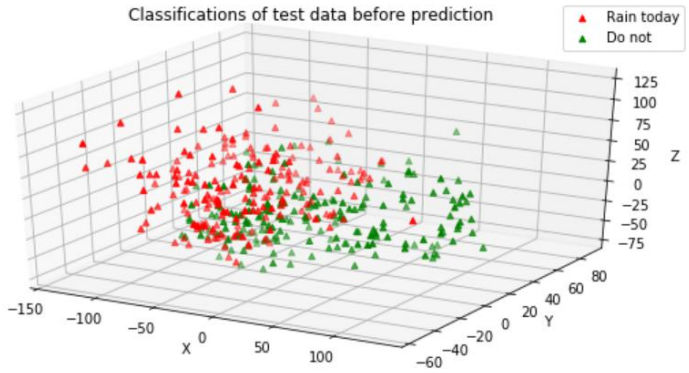


Figure 5

As the visualization with a small part of the original dataset shows that there are some differences between Rain Today or Not Rain Today.

### 1.2.6 Time Series Issues

Therefore, the distribution of features may depend on some requirements, for instance, changing with time. Therefore "qplot()" function in R with Date attribute and WindGustSPeed attribute as axis produces the following figure 6.


Figure 6

Although the attribution of RainToday label is not precisely separated and missing data from 2016, 2017, and 2018, there are do some potential forms; for example, the lowest value of wind speed throughout the year presents a triangular vertex. Therefore, the assumption fits a time series issue. The proper models for time series issues are linear.

## 1.3 Implementation and Interpretation

### 1.3.1 K Nearest neighbor algorithm

According to the distribution of 2 classes in figure 5, there are do two clusters, although the boundary is not clear. Therefore, the simple distance-based method, namely the k nearest neighbor algorithm, should be useful, but with lower performance. In order to test the hypothesis, the results of four models with different parameters are shown in Figure 7.

```
k = 1, score 0.8609972677595629
k = 2, score 0.8666325136612022
k = 7, score 0.8837090163934426
k = 17, score 0.8794398907103825
```

Figure 7

The average distances of observations to their nearest neighbors are a crucial measurement for classification. The results for all four models used the same confidence. However, the differences between the four models are, unfortunately, small, although the 15-KNN model performs the best.

It is not particularly surprising to get this result, as shown in Figure 5, many red dots have not only red dots but also green dots in the neighborhood. Furthermore, there is a red dot in the middle of a bunch of green dots. Therefore, the KNN model cannot entirely distinguish two clusters. However, because of the large amount of data, the density of data points in the original data space is large, so the overall performance of the simple model is also not bad.

### 1.3.2 Logistic Regression Model

Logistic regression is both a linear model and a binary classification model. In this process, it indicates the probability that this data belongs to a specific class, namely the probability to rain today. The accuracy of the model depends not only on the appropriate k value but also on the appropriate confidence.

Because sometimes, using python or r to implement the model has individual deviations, so the author uses two programming languages at the same time to compare the results. One of the advantages of r language is that it can intuitively return a result equation. The following equations were obtained by inputting 17 features for calculation.

$$y = a_1 x_1 + a_2 x_2 + \ldots + a_{17} x_{17}$$

Each x corresponds to a feature in figure 8, and each represents the estimated value of the first column in the figure. Although the contribution value of each eigenvalue is relatively small from the estimation result, through the z test, we can intuitively see the correlation between the eigenvalue and the final result in figure 8.

```
Coefficients:
               Estimate Std. Error z value Pr(>|z|)
(Intercept)   34.8545701  5.5971061   6.227 4.75e-10 ***
Date           0.0001280  0.0000471   2.719  0.00655 **
MinTemp       -0.0311863  0.0173914  -1.793  0.07294 .
MaxTemp       -0.0037797  0.0231858  -0.163  0.87050
Rainfall       0.0105055  0.0033332   3.152  0.00162 **
Evaporation   -0.0269277  0.0125650  -2.143  0.03211 *
Sunshine      -0.1031141  0.0125560  -8.212  < 2e-16 ***
WindGustSpeed  0.0498861  0.0032442  15.377  < 2e-16 ***
WindSpeed9am  -0.0158036  0.0048433  -3.263  0.00110 **
WindSpeed3pm  -0.0280571  0.0051668  -5.430 5.63e-08 ***
Humidity9am    0.0008864  0.0035186   0.252  0.80111
Humidity3pm    0.0435060  0.0037678  11.547  < 2e-16 ***
Pressure9am    0.0547198  0.0172737   3.168  0.00154 **
Pressure3pm   -0.0962751  0.0171948  -5.599 2.15e-08 ***
Cloud9am       0.0126691  0.0165537   0.765  0.44407
Cloud3pm       0.0932375  0.0176734   5.276 1.32e-07 ***
Temp9am        0.0349982  0.0251028   1.394  0.16326
Temp3pm        0.0015489  0.0258532   0.060  0.95223
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Figure 8

It can be seen from Figure 8 that whether it rains today depends mainly on Sunshine, Wind Gust Speed, Wind Speed at 3 pm, Humidity at 3 pm, and Pressure at 3 pm. The features Date, Rainfall, Evaporation, Wind Speed at 9 am, and Pressure at 9 am play also a role in this weather forecast. Surprisingly, although the temperature affects pressure, humidity, and evaporation to some extent, the correlation between the two temperatures and the response is not high.

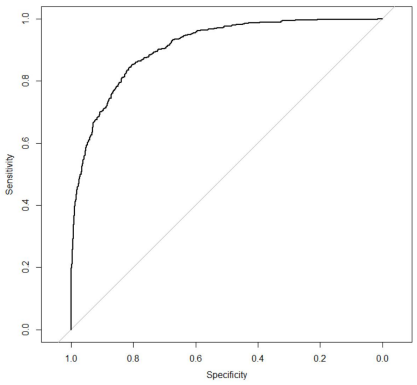


Figure 9

With the "roc()" function from the "pROC" library, the performance of Logistic Regression is excellent, like in figure 9. The area under the roc curve, namely the AUC, is 0.9084, outstanding. Though using the same input python model results even better, it is incredibly close to 1, like in figure 10.

```
accurate of train model: 0.9992313604919293
accurate of test model: 0.9974385245901639
auc of test model: 0.9999845932316882
precision and recall of test model:
              precision    recall  f1-score   support

          No       1.00      1.00      1.00      4561
         Yes       1.00      0.99      0.99      1295

    accuracy                           1.00      5856
   macro avg       1.00      0.99      1.00      5856
weighted avg       1.00      1.00      1.00      5856
```

Figure 10

According to the author's not too short analysis experience, this result is almost perfect. Not only the accuracy of the two classes but also the recall are both ridiculous. To be more comprehensive, the visualization with the predicted labels from the test model is nearly the same. Although the model established by r language is already perfect, the model implemented by python is perfect.

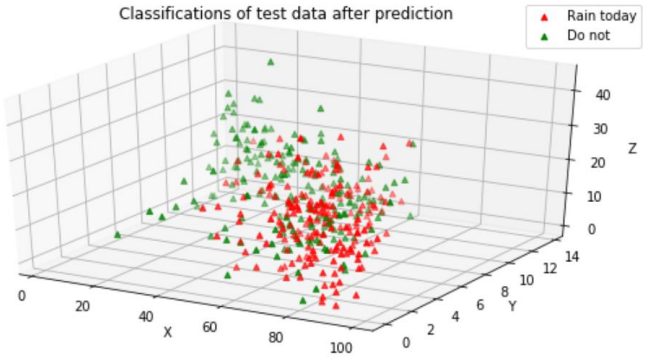


Figure 11

It may be because the R language needs to set manually, and the author has not adjusted the parameters. In the python model, all parameters have default values. Therefore, the reason why the python model performs better than the R may be lying on the differences between the default parameter values of the r and the python.

### 1.3.3 Multiplayer Perception Classifier

Although the logistic regression model performs very well, does a simple neural network model achieve the same brilliant result?

The multilayer perceptron is a simple neural network model that automatically adjusts the weight of each node through back feedback. Based on two input-output models, this model uses hidden layers to approach the original linear model gradually. Time and computational costs are critical considerations in neural network models. Although there is no need to waste time training the model when the ordinary model can meet the needs, its applicability is much higher than the ordinary model.

Using the "adam" activation function, the results of the neural network model of about 5000 layers are currently relatively good. But it takes almost a minute to calculate, as in figure 12.

```
Start time: 18:51:42.771203

AUC: 0.9999437906914337
              precision    recall  f1-score   support

          No       0.98      1.00      0.99      4561
         Yes       1.00      0.93      0.97      1295

    accuracy                           0.99      5856
   macro avg       0.99      0.97      0.98      5856
weighted avg       0.99      0.99      0.98      5856

End time: 18:52:28.651574
```

Figure 12

Although the results of the multilayer perceptron are also excellent, compared with the fast and accurate performance of logistic regression, neural networks are not the most suitable model at present.For each different project, it is most important to obtain the model that meets the requirements with minimal calculation and time cost.