Rain in Australia: Machine Learning

Zhen Liu
ID: 31561012
zl1r20@soton.ac.uk

Mingjun Xie
ID: 32022689
mx2n20@soton.ac.uk

Jiawei Qi
ID: 31845177
jq1u20@soton.ac.uk

Shenhui Guo
ID: 31525008
sg3m20@soton.ac.uk

Abstract

In the previous report, we have cleaned and preprocessed the data. In this report, our main purpose is to use various models to analyze, train and predict whether it will rain and how much rainfall in Australia the next day.

1. Background

As we prepare the data and complete the pre-processing of the data, we will start to select the appropriate model to train and prepare to pick a more suitable model from the selected models for analysis, prediction and comparison. Second, we will synthesize the advantages and disadvantages of various models, try to combine various models, evaluate their performance and errors, and strive to piece together the best efficient model combination. Finally, we will evaluate this report and reflect on our success and limitations.

2. Introduction

Based on the previous pre-processed data, the models we choose include random forest, XGBoost, logistic regression, Decision tree, Naive Bayes, etc. We will compare the performance of each model and use N-fold cross-validation to calculate the mean and standard deviation of each model.

The first one, we will use the random forest model. Random forest or random decision forest [3] is an ensemble learning method for classification, regression and other tasks. It operates by constructing multiple decision trees during training and outputting them as classes for classification or regression.We choose random forest because random forest does not need to do dimensionality reduction processing, and does not need to do feature fitting. The second is to use the random forest model to make it easy to distinguish each feature before, and it can only reflect the feature that has the greatest impact on the data. The third reason is that the random forest model is not easy to overfit and can balance errors.

For the second model, we will use the decision tree model. Decision tree is a supervised learning model.The advantage of this model is that it can also solve high-dimensional problems, and the selection of features will not be a problem. Secondly, decision tree model has strong generalization ability and can ignore the problem of minimum value, so we will choose it.

The third model is the XGBoost model, which is an additive model composed of a base model to optimize a distributed gradient boosting library. The reason we use this model is that this model is more accurate and more flexible. At the same time, it can prevent overfitting just like random forest. There is also a logistic regression model, which is an algorithm for solving two classifications. Its advantages are fast running speed, simple implementation, and low computational cost. The last one is Naive Bayes is one. Naive Bayes is a simple predictive modeling algorithm. Its advantage is that it runs faster, can intuitively see the attributes of each feature, and is easy to understand

3. Bayesian classifier

The Bayesian classifier[2] is a classification method based on the Bayesian principle. It builds a model of the joint probability distribution p(x,c) according to known conditions, and then calculates the size of the posterior probability according to the formula of conditional probability. The method of forming a model. (Note: P(x,c)= P(xc) P(c)). The purpose of the Bayesian model is to determine which category the X sample belongs to, so it is converted into the probability of calculating the posterior P(cx) in different categories. We calculated that the greater the posterior probability of a certain category, the greater the category is, the more likely it is that x belongs to the correct category. Therefore, our goal is to maximize the posterior probability, and then determine the category with the largest posterior probability as the category to which the sample belongs.

3.1. Feature impotence

We use BernouliNB to deal with discrete data with binary features. The maximum value of each feature probability of each class (that is, the prior condition probability $P(x_i|y)$ the five more important features as shown in the figure below.

```
feature_importance:
[-0.6892374703644464, -0.6706298766417405, -0.9418847699914377, -1.0507771767636136, -0.9533890341556308, -1.393267485712391, -0.69988161318
32974, -0.6021605861126549, -0.7407408995683369, -0.6653994822641156, -0.772696131773758, -0.6494732567772665, -0.4588559448008809, -0.30505
36270056856, -1.0016642236973183, -0.9836334715603936, -0.2756595519828764, -0.19865441322327726, -0.7687253311163893, -0.9950298333469973,
-1.20757967464116 7]

biggest_five_feature_importance:
[-0.19865441322327726, -0.2756595519828764, -0.3050536270056856, -0.4588559448008809, -0.6021605861126549]

id      feature_name
17       Cloud3pm
16       Cloud9am
13       Humidity3pm
12       Humidity9am
7        WindGustSpeed
```

Figure 1. Feature impotence

```
Mean of Accuracy = 0.7537338252942363
Mean of Error = 0.24626617470576367
Standard deviation of Error = 8.542544962973906e-06
```

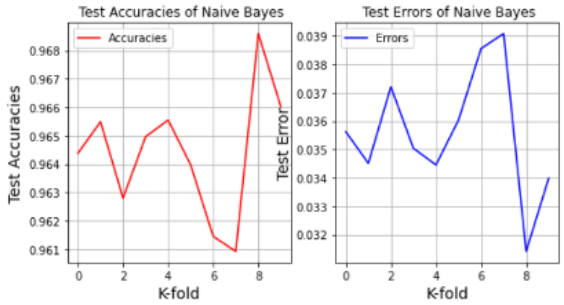

Figure 2. Bayesian

4. Logistic regression

The principles of Logistic Regression and Linear Regression are similar. First look for a prediction function. This function is the classification function we are looking for. It is used to predict the judgment result of the input data. Then construct a loss function (Cost function) that represents the deviation between the predicted output and the true category of the training data. What we have to do is to continuously optimize the loss function[6], the smaller the loss function, the higher the accuracy we get. Logistic Regression sometimes uses gradient descent (Gradient Descent).

4.1. Feature impotence

In logistic regression, we directly use laso for feature selection. Laso can change the weight of some unimportant feature values to 0. The picture below is the 5 more important special evidence we got using laso.

4.2. Results predicted using Logistic regression

The accuracy of logistic regression in the test set prediction is slightly better than that of Bayesian, but the performance of the Random forest and XGboost models is much worse. We found that its standard deviation is the smallest, so the logistic regression of these four models has the best stability for rain prediction in Australia.

id	feature_name
5	Sunshine
7	WindGustSpeed
13	Humidity3pm
15	Pressure3pm
17	Cloud3pm

Figure 3. Feature impotence

3.2. Results predicted using Bayesian model

Naive Bayes is the smallest test accuracy of our 4 models and its loss contrast is also very large, which means that this model is very unstable, and sometimes the results will be very poor (this point we can also easily find the following The figure shows that the model has a lot of volatility).

```
Mean of Accuracy = 0.7952234975561225
Mean of Error = 0.20477650244387746
Standard deviation of Error = 3.4588929127924753e-06
```

Figure 4. Logistic regression

5. Random forest

Random forest [4] belongs to bagging algorithm in Ensemble Learning, which combines Decision Tree with Bagging method. Random Forest has two random ideas, one is to randomly select samples, and the other is to randomly select features. We first use the bootstrap method to generate N random training sets, and use each training set to build a decision tree model. At present, there are three main decision tree algorithms: ID3, ID4.5 and CART. We hope that the samples contained in the branch nodes of the decision tree belong to the same category as possible, that is, the higher the 'purity' of the node, the better. Each trained decision tree is predicted on the test set, and the average value is calculated.

5.1. Feature impotence

In the random deep forest, the importance of attributes is obtained by calculating and sorting each attribute in the data set. In this decision tree, the importance of attributes is calculated by the amount of improvement in performance metrics at each attribute split point, and the node is responsible for weighting and counting times. The closer to the root node, the greater the weight. You can use Gini to cal-

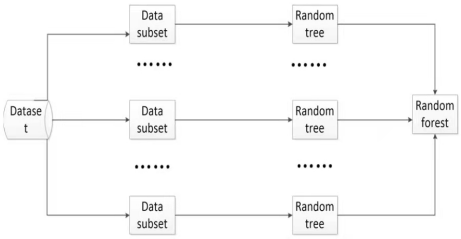

Figure 5. Flow chart of random forests[3]

culate performance metrics.The picture below is the 5 most important features we selected.

```
feature_importance:
[0.026398128226455 48, 0.029137259976981793, 0.029424331418740 57, 0.030068736609686 48, 0.027626237140884496, 0.167104527923 1007, 0.017929836290099706, 0.041945410466781 86, 0.018335425668730503, 0.019317273353991447, 0.018871598391043676, 0.020803024385 1963, 0.028492952190451 82, 0.125582076336448 6, 0.046869553352060306, 0.063422642557202 23, 0.055326771042003 51, 0.160162097438865 4, 0.028387124727297496, 0.031382206532 8632, 0.013208075444267322]

biggest five feature importance:
[0.1671045279231007, 0.1601662097438865 4, 0.125582076336448 6, 0.063422642557202 23, 0.055326771042003 51]

id      feature_name
5       Sunshine
17      Cloud3pm
13      Humidity3pm
15      Pressure3pm
16      Cloud9am
```

Figure 6. Feature impotence

5.2. Results predicted using Random forest

Since the random deep forest uses multiple models for predictive voting, in the experimental results, we found that the model constructed by the random deep forest performs very well on the test set. The accuracy of Random Deep Forest predicting whether it will rain enough for the next day on the test set is as high as 93.4%, which is much higher than Bayesian and logistic regression.

```
Mean of Accuracy = 0.9340829427394233
Mean of Error = 0.0659170572605765 8
Standard deviation of Error = 4.827668835735026e-06
```

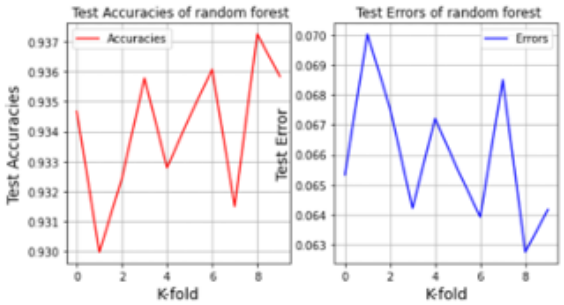

Figure 7. RandomForest

6. XGBOOST

In this section we use XGBOOST model to train and test the data. EXtreme Gradient Boosting [1], is an algorithm of ensemble learning. XGBoost makes a second-order Taylor expansion of the loss function, and adds a regular term to the objective function to find the optimal solution of the whole, so as to balance the decline of the objective function and the complexity of the model which can avoid over-fitting.

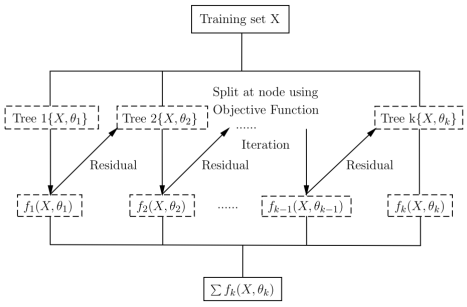

Figure 8. Flow chart of extreme gradient boosting[5]

6.1. Feature impotence

The most significant features for XGBOOST are: Cloud3pm; Humidity3pm; Pressure3pm; RainToday; Sunshine.

```
feature_importance:
[0.030862150713801384, 0.024812454357743263, 0.027056213468313217, 0.033240234968404 77, 0.024116283282637596, 0.054505426436662674, 0.026482262637061653, 0.043243885040283 2, 0.026718746870756 15, 0.028240157291293144, 0.025040315464138985, 0.026136822998523712, 0.024149008095264435, 0.079667814075946 81, 0.025742949917912483, 0.060947313904762 27, 0.048622239381074905, 0.276094675064086 9, 0.024711951613426 21, 0.029518578201532364, 0.060090463608503 34]

biggest five feature importance:
[0.2760946750640869, 0.0796678140759468 1, 0.060947313904762 27, 0.060090463608503 34, 0.054505426436662674]

id      feature_name
17      Cloud3pm
13      Humidity3pm
15      Pressure3pm
20      RainToday
5       Sunshine
```

Figure 9. Feature impotence

```
Mean of Accuracy = 0.9644106522194729
Mean of Error = 0.035589347780527136
Standard deviation of Error = 4.6630124143935565e-06
```

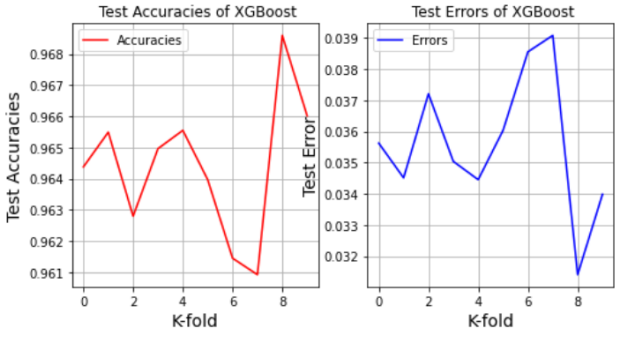

Figure 10. XGBOOST

6.2. Results predicted using XGBOOST

The objective function of logistic regression problem is $obj = \sum_{i=1}^{n} l(y_i, \hat{y}_i^t) + \sum_{i=1}^{t} \Omega(f_i)$. Set The total number of iterations to be 500 and the depth of the tree to be 16. The accuracy of the XGBOOST model is 0.96 and the mean squared error is 0.0356, which is better then all the above method. Then, we use 10-fold cross validation to divided

the data set X into 10 packages. One of the packages is used as the test set and the remaining 9 packages are used as the training set for training. The result is shown in Fig.10.

7. Conclusion

Through the training and prediction of these models, we have obtained a series of accuracy and loss values. We will analyze and evaluate each model. After training and testing, XGBoost has the highest accuracy rate, about 96 %, and the loss rate is about 18 %. The accuracy and loss of random forest and Logistic regression models are about the same, the accuracy is about 86 %, the accuracy of logistic regression is the lowest, only about 72 %, and the loss is about 20 %. After analysis, we feel that XGBoost has the highest accuracy rate among the models we use, which shows that the model we use is very successful, and the preprocessing of data is also very successful, so the accuracy of prediction and analysis is very high. This shows that the XGBoost model is a more suitable model for our data set, and it is also a model with higher accuracy in predicting the next day's rain and rainfall in Australia.

In the process of completing this graduation project, I have a deeper understanding of the concepts of deep learning and neural networks, and the charm of deep learning, which will be of great benefit to my future studies in deep learning. At the same time, our team members also actively cooperate and work hard to complete their respective tasks; we will try to solve problems by ourselves when encountering difficulties, and the same team members will also help each other; if there are different understandings and suggestions, the team members They can also communicate calmly and give their own suggestions, so that the entire task can be completed smoothly.

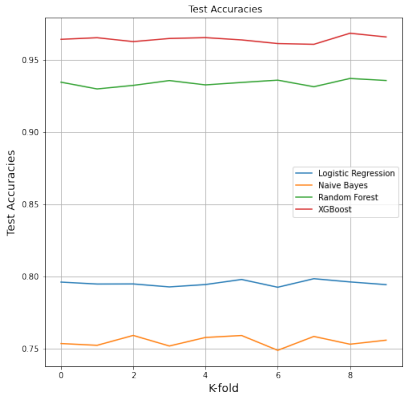

Figure 11. The accuracy of various models on the test set

References

[1] Zhuo Chen, Fu Jiang, Yijun Cheng, Xin Gu, Weirong Liu, and Jun Peng. Xgboost classifier for ddos attack detection and analysis in sdn-based cloud. In *2018 IEEE International Conference on Big Data and Smart Computing (BigComp)*, pages 251–256, 2018.

[2] D P He, Z L He, and C Liu. Recommendation algorithm combining tag data and naive bayes classification. In *2020 3rd International Conference on Electron Device and Mechanical Engineering (ICEDME)*, pages 662–666, 2020.

[3] Hua Lan and Yun Pan. A crowdsourcing quality prediction model based on random forests. In *2019 IEEE/ACIS 18th International Conference on Computer and Information Science (ICIS)*, pages 315–319, 2019.

[4] J.J. Rodriguez, L.I. Kuncheva, and C.J. Alonso. Rotation forest: A new classifier ensemble method. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(10):1619–1630, 2006.

[5] Dahai Zhang, Liyang Qian, Baijin Mao, Can Huang, Bin Huang, and Yulin Si. A data-driven design for fault detection of wind turbines using random forests and xgboost. *IEEE Access*, 6:21020–21031, 2018.

[6] Xiaonan Zou, Yong Hu, Zhewen Tian, and Kaiyuan Shen. Logistic regression model optimization and case analysis. In *2019 IEEE 7th International Conference on Computer Science and Network Technology (ICCSNT)*, pages 135–139, 2019.