Prediction of Rainfall in Australia Using Logistic Regression and Random Forest Algorithms

Doruntina Bekolli [*1]

Abstract

Rainfall is a source of hydration for plants and therefore prevents dryness, it plays a major role in the ecosystem and other fields. Rainfall greatly impacts outdoor activities, its prediction is an especially important and useful issue.This report aims to predict as accurately as possible using previous day's data, if it will rain the next day. We will compare the logistic regression and the random forest algorithm, both useful for binary classification.

1. Introduction

It is interesting to know if it is possible to predict rain with data collected the day before. In the field of aquatic sciences, precipitation plays a big role in the different biological and hydrological cycles. For example a farmer will not water his fields if it is going to rain the next day. The application of fertilizers can also be anticipated in relation to the rain, if fertilizers are applied during rains there is a greater risk of leaching and thus of polluted groundwater.

The Australian climate can be extreme, switching from incessant rainfall(1) to drought, so it is important to study these phenomena. Moreover, it is interesting to try with a model such as in Australia and use it for other cases. Our goal is to predict whether or not it will rain the next day, so we use a binary classification, so we chose two algorithms used for which the features importances can be interpreted*(3).

1.1. Random forest

In recent times, random forests have gained popularity as a method for performing statistical classification. Random forest techniques generate a panel of decision trees. Decision trees give the advantage of being interpretable and drawtaken. The built of the multitude decision trees is done during the training time via bagging or pasting method(4).

The particularity of the random forest is that it introduces randomness when growing trees. It wont find the best feature among the splitting node but among the random subset of features. It has a higher bias and a lower variance. We can measure the importance of each feature with Scikit-Learn.

The hyperparameter of random forest(2):

- N_estimators: The number of decision trees being built in the forest. it is correlated to the size of data.
- Criterion: Function used to measure the quality of splits in a decision tree (Classification Problem).
- Max_depth: The maximum levels allowed in a decision tree. If it is not set, the split will end when the purity is reached.
- Max_features: Maximum number of features used for a node split process. Types: sqrt, log2.
- Bootstrap: used when building decision trees.
- Min_samples_split: This parameter decides the minimum number of samples required to split an internal node. Default value =2.
- Min_sample_leaf: This parameter sets the minimum number of data point requirements in a node of the decision tree.

1.2. Logistic regression

It is a regression algorithm that can be used for classification. it is used to predict the probability for a input to belongs to a particular class.

The function will enter the variable and will use the sigmoids function, the result will be a value that is between 0 and 1(4), the value represents the probability , if it is greater than 50% then the model predicts that it belongs to the class and if it is smaller than 50% it means it does not belong to the class. It is a binary classification.

the logistic function is defined as follows :

[1]GSE, University of Lausane, Lausanne, Canada. Correspondence to: Doruntina Bekolli <doruntina.bekolli@unil.ch>.

Proceedings of the 38^{th} International Conference on Machine Learning, PMLR 139, 2021.

$$\sigma(t) = \frac{1}{1 + exp(-t)} \quad (1)$$

With scikit-learn we can define some hyperparameter for the model, the hyperparameters are:

- solver : Algorithm used for optimization problem
- penalty: penalty to the logistic model if it has too many features
- regularization strength (C): Inverse of regularization strength

1.3. GridsearchCV

With GridSearchCV, hyperparameters are tuned to determine the optimal value for a given model based on the characteristics of the data(4). In order for a model to perform optimally, it is important to consider the values of hyperparameters. In order to know the optimal values, we need to try all the possible values. The process of manually tuning hyperparameters would take a considerable amount of time and resources, which is why we use GridSearchCV to automate the process of tuning hyperparameters.

2. Dataset

This dataset includes approximately 10 years of daily weather records from many different sites in Australia. It contains 23 columns and 145'460 rows. The target variable is "rain tomorrow", the value "yes" for this variable means that the value of rain for that day was 1mm or more. The set of variables contains ; ; location, date, wind direction and speed, pressure, humidity, sunshine, evaporation, cloud, temperature. The values of the target variable are not balanced, that means that we have much value for when it does not rain than when it rains. This can be a problem for the training and to assess correctly the accuracy. We will oversample later the Training set so the data are balanced.

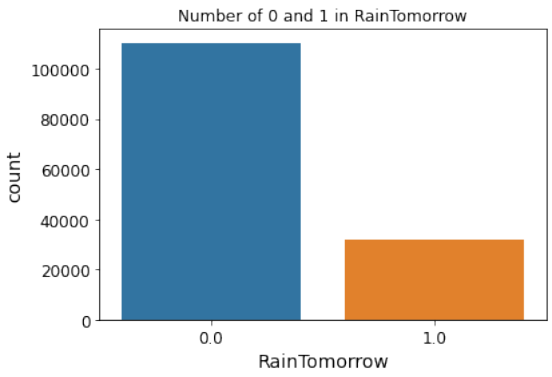


Figure 1. Ratio

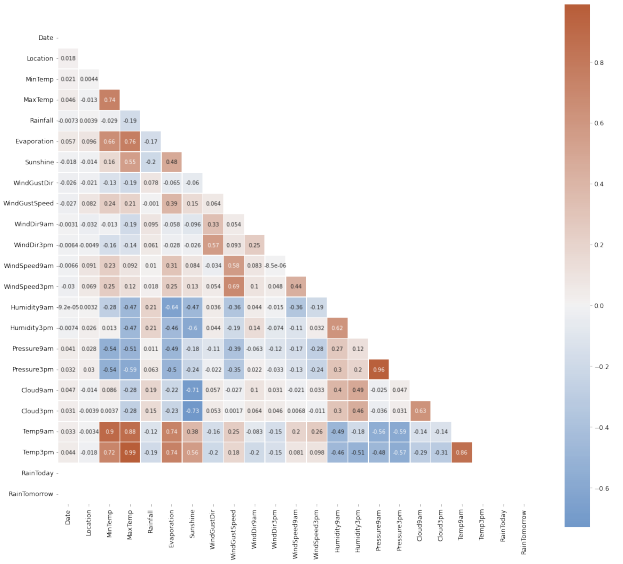


Figure 2. correlation between features

3. Methods

The dataset is downloaded and converted to DataFrame with panda. The computing and program is coded with python on colab.

3.1. Data preprocessing

3.1.1. NAN VALUE

The database contains a lot of NAN values, these values prevent the the machine learning algorithms to perform and must be replaced or deleted. In our case they will be replaced by the most common value for the variable, using the mode() method. We remove outlier in our data to improve the efficiency.

3.1.2. CONVERT VALUE

The data has some issue that interfere with the training and the prediction. Some variable are object dtype and it needs to be encoded into int or float type. We will transform those with the method fit.transform of LabelEncoder.

3.1.3. VALIDATION AND TRAINING SPLIT SET

Now that i the data is cleaned and converted into float types, we have to standardize the data and then separate the features from the target. We will separate with a ration of 75% and 25%

3.1.4. OVERSAMPLE

Since our database is not homogeneous I have increased the number of y assigned to 1 so that the ration between 0 and

1 is identical, there is also undersampling which consists in decreasing the number of data for the majority category however it is less efficient than oversampling (5). I will do the oversampling only for the train data in order not to corrupt the evaluation of the precision with the validation test.

3.2. Model fit

We created the model and fit the data. then we did a grid-search to find the optimal hyperparameter to improve the accuracy of the model. we did this for logistic regression and for random forest. we evaluated the accuracy with the ROC curve, the f1 score, the confusion matrix and extract the importance feature.

4. Results

4.1. logistic regression

The train accuracy is around 0.85, it is a bit lower than the validation accuracy.

4.1.1. Val Accuracy

The initial model and the model with hyperparameter founds by grindsearch present the same results. Here is the results

Initial hyper parameter				
Accuracy = 0.8227796884483107				
ROC Area under Curve = 0.7797764590618101				
Cohen's Kappa = 0.5169829536702616				
	precision	recall	f1-score	support
0.0	0.91285	0.85556	0.88328	27118
1.0	0.57354	0.70400	0.63211	7483
accuracy			0.82278	34601
macro avg	0.74320	0.77978	0.75769	34601
weighted avg	0.83947	0.82278	0.82896	34601

Table 1. metrics for the default hyper parameters

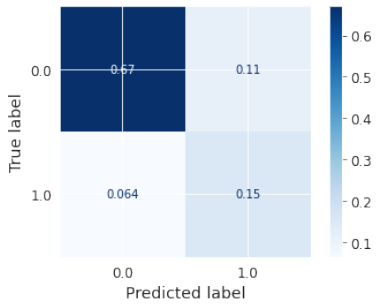


Figure 3. Matrix confusion of the logistic regression

4.2. Random forest

The accuracy of the training set is around 1.00

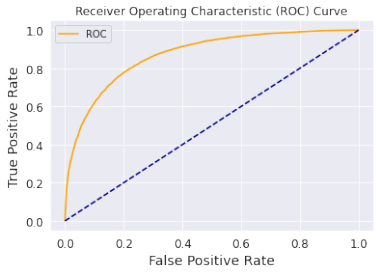


Figure 4. ROC curve logistic regression

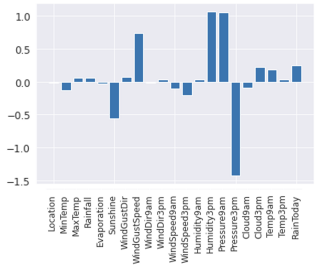


Figure 5. feature importance of the logistic regression

Metrics default hyperparameters validation set				
Accuracy = 0.8518250917603537				
ROC Area under Curve = 0.7673916496045312				
Cohen's Kappa = 0.5502502201519581				
	precision	recall	f1-score	support
0.0	0.89696	0.91618	0.90647	27118
1.0	0.67068	0.61860	0.64359	7483
accuracy			0.85183	34601
macro avg	0.78382	0.76739	0.77503	34601
weighted avg	0.84803	0.85183	0.84962	34601

Table 2. Validation metrics of Randomforest with default hyper parameters

Metrics for improved hyperparameters validation set				
Accuracy = 0.8538192537787925				
ROC Area under Curve = 0.7700668926317309				
Cohen's Kappa = 0.5560599359448546				
	precision	recall	f1-score	support
0.0	0.89805	0.91766	0.90775	27118
1.0	0.67595	0.62248	0.64811	7483
accuracy	0.85382	34601	0.85183	34601
macro avg	0.78700	0.77007	0.77793	34601
weighted avg	0.85002	0.85382	0.85160	34601

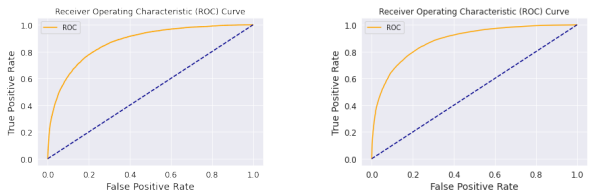
Table 3. Validation metrics of Randomforest with gridsearchCV hyperparameters

4.2.1. ROC curve

4.2.2. Matrix confusion

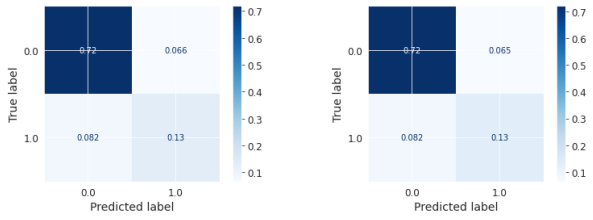
5. Discussion

We can see that the accuracy is around 80% and the f1 score is different for the label 1 and 0. The Random forest algorithms shows better accuracy than logisistic regression. The problem in this dataset is the non-equity of the number



((a)) Default ((b)) GridsearchCV

Figure 6. ROC curve random forest of validation set



((a)) Default ((b)) GridSearch

Figure 7. Matrix confusion Random Forest of validation set

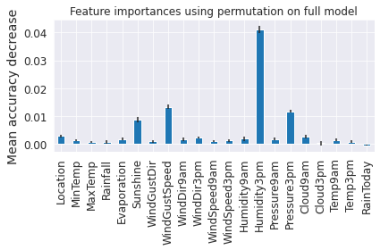


Figure 8. feature importance of the logistic regression

of data in the two categories, this is problematic for two reasons. The model will have difficulty predicting for the limiting factor. One must be careful with oversampling because it can lead to an overfitting. Removing the outliers generally improves the accuracy of the model but it also reduces the amount of data available. Replacing nan values by common values allows to not greatly impact the calculation of the importance of the features and to keep a large amount of data which has a positive impact on the accuracy.

5.1. Logistic regression

We can observe that logistic regression is better at predicting when it will not rain . The precision for label 0 is 0.91 but for label 1 it is 0.57 which underlines the problem of the lack of balance in the data between 1 and 0.

However, wwe have to be careful using this metric because the data are not balanced and it would be better to focus on the f1 score. We can observe the same trend as for the ROC and the accuracy. f1 score is higher for the prediction of label 0 than for label 1. The model as a better a accuracy for label 0 than for label 1. The f1 score for label 0 is 88% and 63% for label 1.

In the confusion matrix, the same tendency is shown, label 0 is more easily predicted than label 1. We can see that for the prediction of label 1 does not have a very high ratio between the good and bad prediction. The prediction for label 0 is however high and this can be explained by the fact that the validation set has a much higher proportion of 0's than 1's and that it did not have an oversampling.

The gridsearch algorithm did not find more efficient parameters for the algorithm, so there was no improvement in accuracy for logistic regression.

The Features sunshine ,wind gust speed, humidity, pressure 9 am and 3 pm seem to have a greater importance for the prediction. The accuracy for training set is a bit greater than the validation one, it can be understand that there is no overfitting. The oversample improves the label 1 prediction.

5.2. Random forest

The random forest in comparison with logistic regression for accuracy presents two interesting points. The first one is that it predicts less correctly the value 0 than the logistic regression but it is more efficient to predict the value 1 which remains however low, 0.67%. This still leads to a higher overall accuracy than logistic regression. However, as said before, we have to be careful with the precision as well as with the ROC because our data are not balanced. However, the f1 score remains higher with the use of Random Forest in comparison with Logistic Regression. The ROC is lower than for the logistic regression, we can observe that the curve increases less quickly than for the logistic regression. But our data being unbalanced, it is more interesting to use the f1 score.

For matrix confusion the same observations are made as for the logistic regression except that the ration for label 1 is better than for the logistic regression. GridsearchCV found parameters that improved the accuracy but not significantly. There is not a significant difference compared to other

The accuracy of the train for random forest is really high with a score of 1.00. The hyperparameters chosen are ideal for the validation test but it shows overfitting on the training set. If we decrease the number of max depth the accuracy for the validation set decrease too. The oversample improves the label 1 prediction.

6. Conclusion

The prediction of rain using this model remains not very conclusive given the accuracy for label 1, when it rains. The gridsearch allows us to find other more efficient parameters,

but the parameters do not seem to greatly impact the prediction's accuracy. Random forest seems to be a more efficient algorithm than logistic regression mainly for the prediction of label 1. The importance features are not the same for both algorithms even if they share many similarities. It is therefore useful to find other solutions to improve the accuracy and focus on a random forest algorithm because it shows better accuracy.

7. Software and Data

Here is the github link
https://github.com/dorunbek/2022_ML_EES/blob/4b84bc861c1e6da70b881bc019f87f98f50f279e/Project/Rain_AUS_Pred.ipynb
Here is the link to the dataset.
https://www.kaggle.com/datasets/jsphyg/weather-dataset-rattle-package

References

[1] 'rain bursts' over sydney have intensified 40% over last two decades, research finds, Nov 2022.

[2] Random forest hyperparameter tuning: Processes explained with coding, Sep 2022.

[3] Antonio Cabezuelo. Prediction of rainfall in australia using machine learning. *Information*, 13:163, 03 2022.

[4] Aurélien Géron. *Hands-on machine learning with Scikit-Learn and TensorFlow : concepts, tools, and techniques to build intelligent systems*. O'Reilly Media, Sebastopol, CA, 2017.

[5] et Al. Roweida Mohammed. *Machine Learning with Oversampling and Undersampling Techniques: Overview Study and Experimental Results*. PhD thesis, Jordan University of Science and Technology, 2020.