Predicting weather in Australia using Machine Learning

Siraj Us Salekin

Data Mining

Abstract

The dataset we have worked with is of 10 years of weather observation across many locations in Australia and is taken from Kaggle. Based on the attributes of the weather, the problem requires us to predict if it will rain in Australia in the future. In our project, we pre-processed the data and used several classifiers for prediction. We used different machine learning models of which XGBoost proved to provide better results than other classifiers. Results obtained from the classifiers that were used in the project were then compared. In addition, we also did a time-series analysis to predict the future temperature of Australia.

1 Introduction

Accurate rainfall forecasting continues to be a key goal, with understanding the intricate physical processes that cause rainfall remaining a significant issue with substantial effects on securing and producing food as well as providing enough water for major population centers while reducing flood hazards.

The dataset obtained from Kaggle provides 22 attributes for every day's weather and has 145,460 entries having both categorical and numeric data. The problem at hand was to predict if it will rain the next day by assessing today's weather attributes using different machine learning classifiers. We have tried to use several classifiers for prediction.

In addition to the rain prediction, we have also applied time-series analysis to predict future temperature of Australia based on the available data. We have predicted up until the year 2039 by using data of 2007-2015 period.

1.1 Our Contribution

In this project, we have performed the following:

- Processed the data.
- Divided data into training and test sets.

- Tried to find the correlation between variables.
- Used Logistic Regression.
- Used Decision Tree.
- Used Random Forest.
- Used XGBoost.
- Used Adaboost.
- Compared results obtained from different classifiers.
- Performed time-series analysis to predict temperature.

2 Background

During World War II in the late 1930s, the British developed an ingenious radar that could not only be used to track enemy aircraft but also to pick up echoes from raindrops at precise wavelengths (5–10 cm) [1]. This method can be applied to follow and analyze a single shower as well as to study the structure of a bigger storm's precipitation. It is evident that this can only predict rainfall in a short period of time, but it is an early application of scientific and technology techniques by contemporary scientists to predict the possibility of future precipitation. Scientists are now able to make decisions based on a vast amount of research data and geophysical knowledge thanks to the quick growth of technical scientific data and technology. Utilizing machine learning for artificial intelligence is the first option. In January of last year, Google engineer Jason Hickey unveiled a machine learning method for analyzing weather radar charts. The goal is to transform the radar chart into an automated "computer vision" problem. He eventually employed U-Net in artificial neural networks after trying to use a lot of data to teach robots how to learn physics principles from algorithms (CNNs). In terms of ultra-short-term changes, his machine learning technique outperforms the three traditions of HRRR numerical forecasting, optical flow method, and persistent modeling (persistent model) [2]. Some data processing techniques and models are required when the experiment employs decision trees to forecast rain in Australia.

3 Related Works

Ye Zhao et al's paper [3] focused on using ensemble methods to predict rain using the same dataset as well as using classifiers such as Logistic Regression, KNN, Bagging, AdaBoost. But, their finding was that the trained models did not perform well enough and the highest accuracy they could achieve was 82&. Ensemble method was only slightly better than some weak learners. Their takeaway was that in order to achieve better results, they need to enhance the models further. They also acknowledged that predicting rain is a difficult task based on some attributes only.

4 Material and Methods

While doing this project, we have followed the standard approaches of solving a machine learning classification problem.

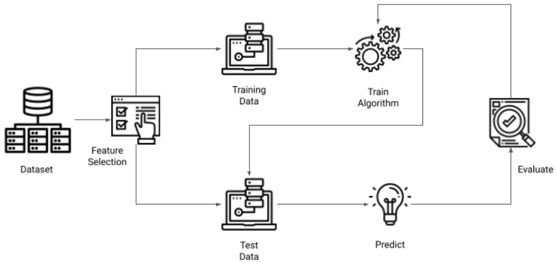


Figure 1: Workflow In This Project

First, the dataset was evaluated and anomalies were taken care of. We dropped the columns with the highest null values and all rows containing the null values. The categorical columns were then transformed into numeric. After that, we divided the whole set into training and test data (80-20). The chosen algorithms were trained using the training set and then used to make predictions. After evaluating the accuracy of the models, we compared the results.

4.1 Algorithms Used

We have used the following algorithms for the prediction problem:

- Logistic Regression: Logistic regression is a process of modeling the probability of a discrete outcome given an input variable. The most common logistic regression model a binary outcome; something that can take two values such as true/false, yes/no, and so on. Multinomial logistic regression can model scenarios where there are more than two possible discrete outcomes.

- Decision Tree: One of the most popular and also easiest classification algorithms. It is widely used as it is very understand and interpret the results. In Decision Trees, for predicting a class label for a record we start from the root of the tree. We compare the values of the root attribute

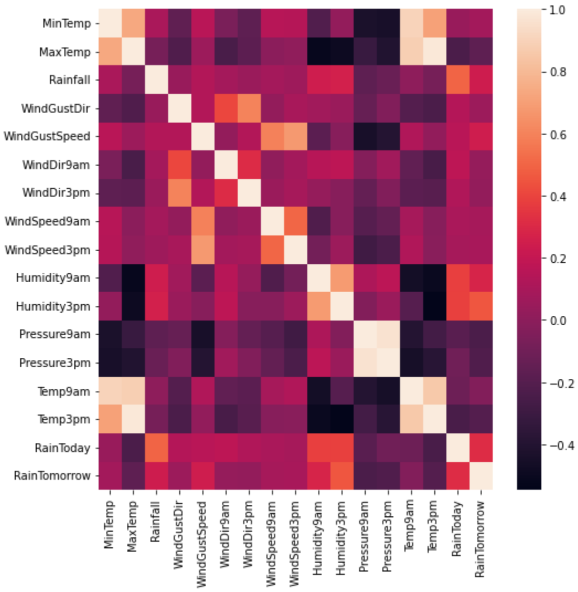


Figure 2: Correlation Heatmap

with the record's attribute. On the basis of comparison, we follow the branch corresponding to that value and jump to the next node.

- Random Forest: One of the common problem of using Decision Tree algorithm is that they tend to over-fit a lot on the training data. To overcome this problem, we have used Random Forest - an ensemble method which is a collection of numerous Decision Trees.

- AdaBoost: This algorithm improves the weak classifiers and punishes outliers in the dataset. This might help us overcome the limitations of the above algorithms.

- XGBoost: XGBoost is an implementation of Gradient Boosted decision trees. XGBoost models majorly dominate in many Kaggle Competitions. In this algorithm, decision trees are created in sequential form. Weights play an important role in XGBoost. Weights are assigned to all the independent variables which are then fed into the decision tree which predicts results. The weight of variables predicted wrong by the tree is increased and these variables are then fed to the second decision tree. These individual classifiers/predictors then ensemble to give a strong and more precise model. It can work on regression, classification, ranking, and user-defined prediction problems.

4.1.1 Logistic Regression

This powerful algorithm, especially in the case of binary classification, yielded us a prediction with an accuracy of 84%.

4.1.2 Decision Tree

While using the Decision Tree algorithm, we have used Entropy as the criterion as opposed to Information Gain, Gini Index, or Gain Ratio. Entropy is chosen as it provides as it always lies between 0 to 1 and provides better performance.

The accuracy achieved from this algorithm is 79%.

4.1.3 Random Forest

While using the Random Forest algorithm, we have used Entropy as the criterion as well. The accuracy achieved from this algorithm is 85%.

4.1.4 AdaBoost

This algorithm provided one of the best results, achieving an accuracy of 84%. The n estimators was set to 50, meaning that 50 weak learners will be trained in each iteration.

4.1.5 XGBoost

This algorithm provided the best results, achieving an accuracy of 86%.

4.1.6 Temperature Prediction Using Time-series Analysis

In addition to the prediction for rain, we have also predicted the temperature of Melbourne city by using time-series analysis. We have kept only the rows containing the data of Melbourne. After that, we plotted the data to review the data availability for Melbourne and found that data is missing from 2015-2016. So, we decided to keep data only until 2015 for our prediction. As the next step, we selected Date & temp3pm as the parameter and trained the model with the data.

After training, the model predicted the temperature of Melbourne from 2015 to 2039. It is observed that the temperature will rise greatly in the upcoming years in the city.

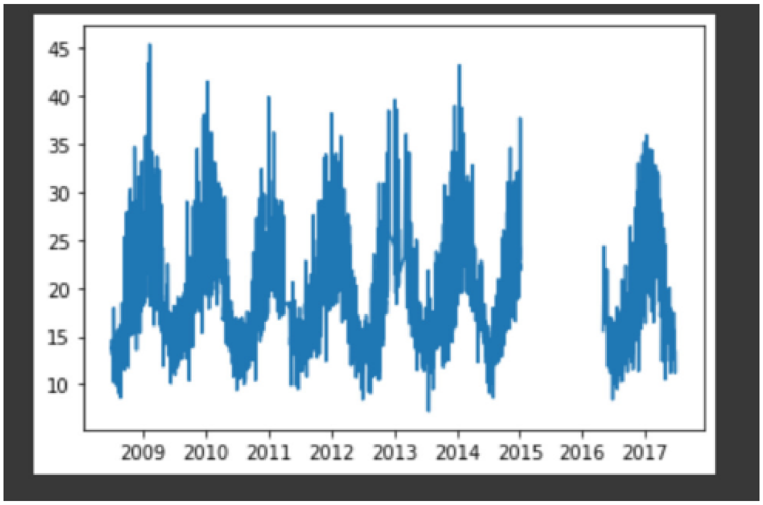


Figure 3: Data unavailable for Melbourne after 2015

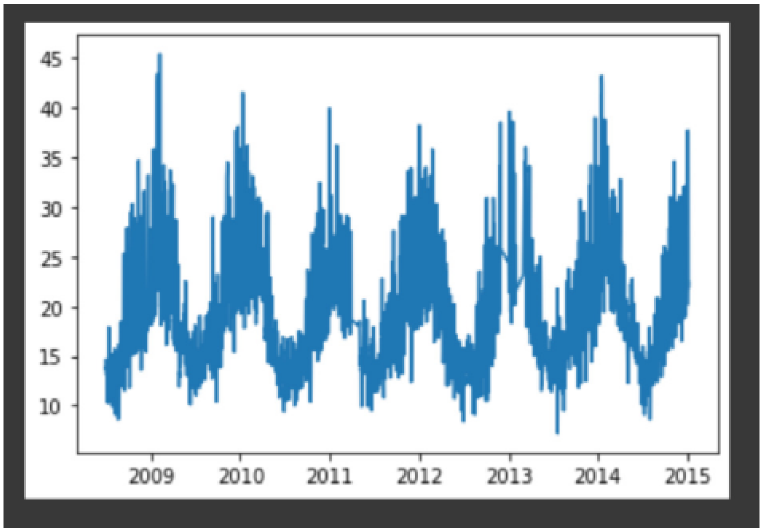
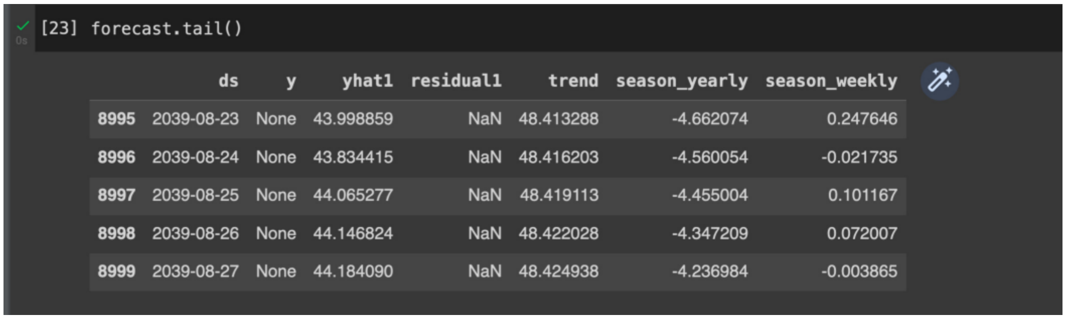


Figure 4: Data removed after 2015



```
[23] forecast.tail()

              ds     y     yhat1  residual1      trend  season_yearly  season_weekly
8995  2039-08-23  None  43.998859        NaN  48.413288      -4.662074       0.247646
8996  2039-08-24  None  43.834415        NaN  48.416203      -4.560054      -0.021735
8997  2039-08-25  None  44.065277        NaN  48.419113      -4.455004       0.101167
8998  2039-08-26  None  44.146824        NaN  48.422028      -4.347209       0.072007
8999  2039-08-27  None  44.184090        NaN  48.424938      -4.236984      -0.003865
```

Figure 5: Temperature prediction of Melbourne

5 Experimental Results & Analysis

Evaluation of Algorithms					
Score Type	Logistic Regression	Decision Tree	Random Forest	AdaBoost	XGBoost
Precision	84%	80%	85%	84%	85%
Recall	85%	79%	86%	85%	86%
F1 Score	84%	79%	85%	84%	85%

5.1 ROC Curves

After analyzing the ROC curves for all of the algorithms that we've used, it can be clearly seen that XGBoost is performing the best. The area under the ROC is greater for XGBoost and Random Forest, AdaBoost & Logistic Regression are coming close to it.

5.2 Code

The code for this project can be found here-

- Github: https://github.com/insert link

6 Conclusion & Future Works

The goal of the project was to accurately predict the possibility of rain in Australia. Using different machine learning models, specially XGBoost provided the best result alongside Random Forest, Logistic Regression, and AdaBoost. However, the accuracy is still not up to the mark and can be improved.

The following steps can be taken to improve the accuracy in the future:

- Using ensemble methods.

- Balancing data by creating a portion of synthetic data.
- Further algorithm tuning.
- Careful feature engineering.

References

[1] John P. Rafferty. Numerical weather prediction (nwp) models. 2021.

[2] Jason Hickey. Using machine learning to 'nowcast' precipitation in high resolution. 2021.

[3] Yifei Ma Mengyan He Ye Zhao, Hanqi Shi. Rain prediction based on machine learning. 2022.