Will It Rain in Australia?

Weather Forecasting Using Machine Learning Algorithms

Luca Bulfon and Jack Cimorelli (Team 3)

Marist College

MSIS 645: Data Mining & Predictive Analytics

Prof. Eítel Lauria

**Executive Summary**

This project will attempt to analyze multiple measurements of different meteorological parameters (temperature, wind, humidity, pressure…) during a day and predict whether it will rain tomorrow or not. The project will use data from the Australian Bureau of Meteorology (BOM) contained inside a dataset available on Kaggle.com. The motivation for this project is to help increase our understanding of weather and patterns. Understanding these patterns' importance cannot be understated. It is widely accepted that weather is easiest to predict in the near future and becomes increasingly difficult further into the future. Weather forecasting has a very vast scale of complexity. It can be performed with any level of technology or dataset (Brenner, 2015).

Before usage the data used in this project required extensive preprocessing. The dataset initially had several different data types that were not conducive to this research that required streamlining. The data also inherently has important spatial trends that must in some way be preserved. Lastly, scaling using z-scores and removing outliers occurred before use in modeling. A 70/30 training/testing split was used in this project.

This project used data mining techniques to analyze a dataset with a binary target variable: whether it will rain the next day, implementing a set of different models to explore their effectiveness on this dataset. Firstly, we will employ K-Nearest Neighbors (KNN) and Naive Bayes classifiers, known for their simplicity and efficiency in handling classification problems. Then, we will build a Random Forest classifier, which offers robustness against overfitting and can handle complex interactions between features. Finally, to investigate the influence of temporal patterns in the weather data, we included a Long Short-Term Memory (LSTM) neural network.

The project yielded encouraging results. According to the chosen evaluation metrics the most effective model was the random forest classifier. The random forest classifier yielded the highest ROC AUC, F1 measure and accuracy. The LSTM was the least effective method by way of the same metrics. The random forest classifier likely suffered from some amount of overfitting and in the future can be improved by further hyperparameter tuning. Other future improvements to the model could include a Multivariate imputation by chained equations (MICE) to attempt to fill the NA values or other imputation methods rather than excluding this data. This improvement would likely be most beneficial to the LSTM, which analyzes temporal trends and would then have more consistent daily data to work with. In the future one possible improvement to this study could be the introduction of the ArcPy library to further analyze spatial patterns in this dataset.

**Problem Description**

In a world where extreme weather events have become increasingly common, understanding weather forecasting has also become increasingly important. Data mining and predictive analytics offer us many techniques to address this issue (Brenner, 2015). The country of Australia will be the focus of this project. Over the past year Australia has had issues with rainfall. Western Australia has had some of the worst droughts and dry seasons it has seen in recorded history. Over the past 11 to 16 months Western Australia has experienced record lows in rainfall across the entire region and projections for the near future have also forecasted low expectations for rainfall. This can be contrasted with the east which has experienced some of the wettest seasons in recorded history. Despite the low rainfall in the east, rainfall averages for the entire country were 86.7% above the average March rainfall from 1961 to 1990. (Australian Government Bureau of Meteorology, 2024). These unprecedented and divergent trends indicate that further research on rain patterns is necessary. Given weather records from a singular day this model will attempt to predict if it would rain tomorrow.

# Data Description

The dataset downloaded from Kaggle.com contains 145,460 records of 20 different daily measurements of 20 parameters. The parameters are:

['Date', 'Location', 'MinTemp', 'MaxTemp', 'Rainfall', 'Evaporation',

'Sunshine', 'WindGustDir', 'WindGustSpeed', 'WindDir9am', 'WindDir3pm',

'WindSpeed9am', 'WindSpeed3pm', 'Humidity9am', 'Humidity3pm',

'Pressure9am', 'Pressure3pm', 'Cloud9am', 'Cloud3pm', 'Temp9am',

'Temp3pm', 'RainToday', 'RainTomorrow']

The data contains measurements from December 2008 to June 2017 (‘Date’) for 49 cities in Australia (‘Location’). The dataset also contains a large amount of missing data (10.25% missing values), further data preparation was necessary before model building.

# Exploratory Analysis and Data Preparation

There were many NANs in this dataset (Fig. 1). One option to deal with this issue was to apply a MICE to this dataset documented in similar modeling procedures. We chose to omit records that contained NANs as imputation was not viable in this case. Certain cities in this dataset systematically did not record certain variables, imputing variables into these records would have compromised the integrity of these records.

The dataset was provided where the “RainTommorrow” column held a string value: yes or no. We changed the column into a binary where 0 indicates that no rain occurred the following day and 1 indicates that rain occurred the next day.

Cities such as Sydney and Melbourne had different Locations for their airport and the city itself, for the purposes of this project airports were combined with the larger city they were a part of.

The next data preparation performed was to the wind direction variable. The wind direction variable was presented in cardinal format as a string. This column was changed to an integer where 0 indicated north and 90 indicated east.

Given the size of the original data set and the spatial nature of the data, we added the ability to filter the data and select measurements from either a singular city or all cities (Fig. 3). The user can choose their option in a text box in our model. This allowed us to reduce average response times of the program while still ensuring high accuracy in the predictions. This was important as the dataset contained an average of around 1600 measurements per city and over twenty different cities. It also adds the ability to preserve spatial trends integral to the data's nature.

Another step needed before implementing the predictive models was to scale the data. The dataset was heavily unbalanced towards records where “RainTomorrow = yes” (Fig. 1). The data was undersampled randomly to balance before use.

The last step regarding data preparation was to scale the data. Without scaling, the feature with the larger scale will dominate the model, even if its underlying relationship with the target variable is weaker. StandardScaler transforms all features to have a mean of 0 and a standard deviation of 1, allowing the model to focus on the relative importance of each feature based on its variation within the data. Lastly, the ‘Date’ column was dropped.

**Technical Solution**

We will use several different machine learning algorithms as prediction tools. Three of the four algorithms used in this project are supervised binary classification tools as the independent variable in this project is either 0 (no rain) or 1 (rain). One model is a neural network. After having analyzed the dataset and selected the right features and records, we delved into the application of the before cited data mining techniques to measure their performance and evaluate which one would perform better when applied to the dataset presented. The data was partitioned into a 70/30 training/testing split.

The first data mining technique examined was the KNN (K-Nearest Neighbors) algorithm for classification. The K-Nearest Neighbors (KNN) algorithm is a powerful technique used in machine learning for both classification and regression tasks, which uses proximity to make predictions about the grouping of an individual data point. (IBM, 2024) Its core principle assumes similar data points tend to have similar characteristics or values. Data points with similar characteristics in the independent variable are more likely to be similar in the dependent variable. When predicting the result of a certain datapoint, the KNN model considers the value of the K closest datapoints and chooses the majority class label among those K closest data points.

To determine the best number of K-nearest neighbors to apply to the model, we opted to use a GridSearch method coupled with cross validation. We determined a range of K values (1 to 30), which the GridSearch method uses to train the model and provide an estimate for the best K value based on a defined performance metric, which in our case was “accuracy”.

The second model we used in this project is the Naïve Bayes classifier. The model is based upon the Bayes Theorem. Bayes Theorem describes the probability of an event using the known probability of related events. It stated as the probability of A given event B is equal to the probability of B given event A multiplied by the probability of event A all divided by the probability of event B (Lee, 2012). Naive comes from the fact that the algorithm assumes that the presence of each event is independent of others. Our model uses a Gaussian Naive Bayes which assumes that the data follows a normal distribution.

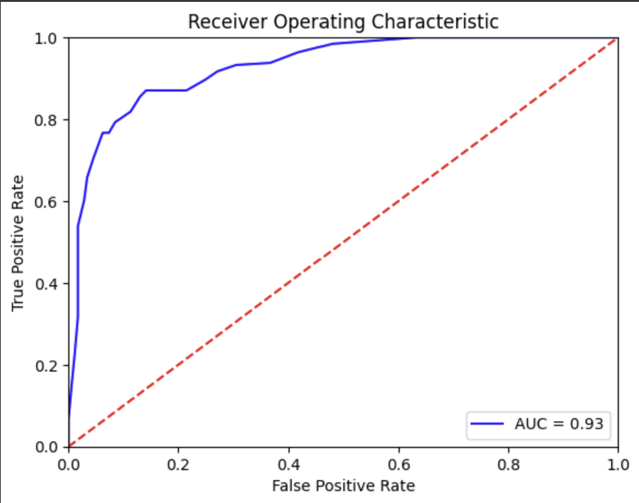
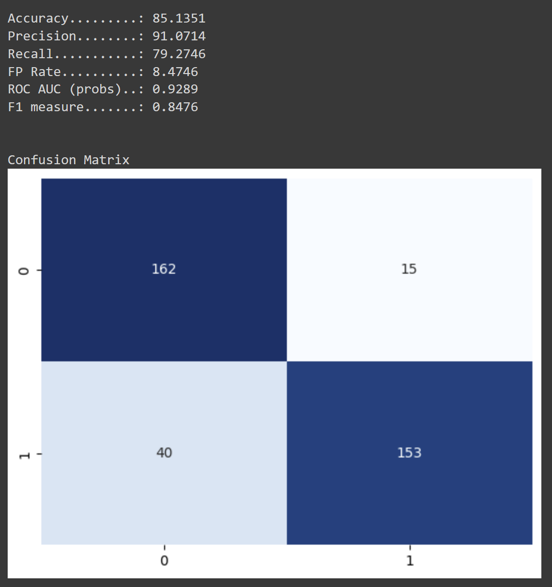
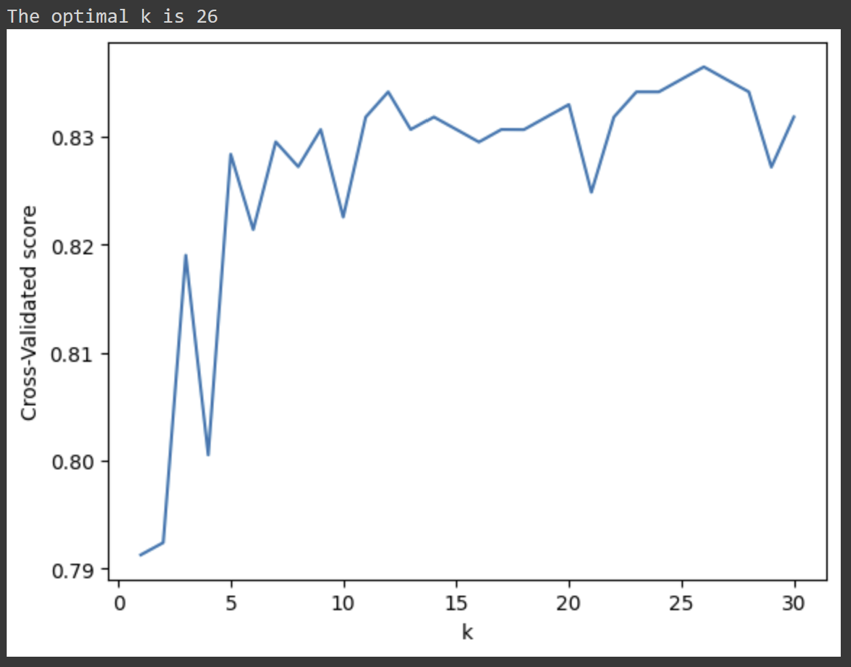
The third model used in this project was a Random Forest Classifier. Random Forest classifiers are an ensemble learning method. Ensemble learning methods use individual classifiers predictions to improve the model's robustness and accuracy. Random forests are built upon decision trees which break data down into different nodes until certain hyperparameters or constraints are met. Random forests differ from decision trees in that they introduce increased randomness (Kam Ho, 1995). Randomness came first from bootstrap aggregation and then at each split a random subset of features being considered. We chose not to use certain hyperparameters because of hardware restrictions.

The fourth and last model we implemented in this project is a Long Short-Term Memory (LSTM) Neural Network. LSTM models utilize gated cell units, consisting of a forget, an input, and an output gate. These gates act as sophisticated filters, regulating the flow of information. The forget gate decides what information from the previous cell state to discard, the input gate determines what new information to include, and the output gate controls what portion of the current cell state is exposed to the network. This gating mechanism allows the LSTM to selectively remember relevant information over long time intervals, effectively addressing the vanishing gradient problem, which characterizes Recurrent Neural Networks (RNN). LSTMs are effective at processing and analyzing sequential data like speech, text, and time series of data, this is reason behind our choice of adding a LSTM to this project.

# Conclusions

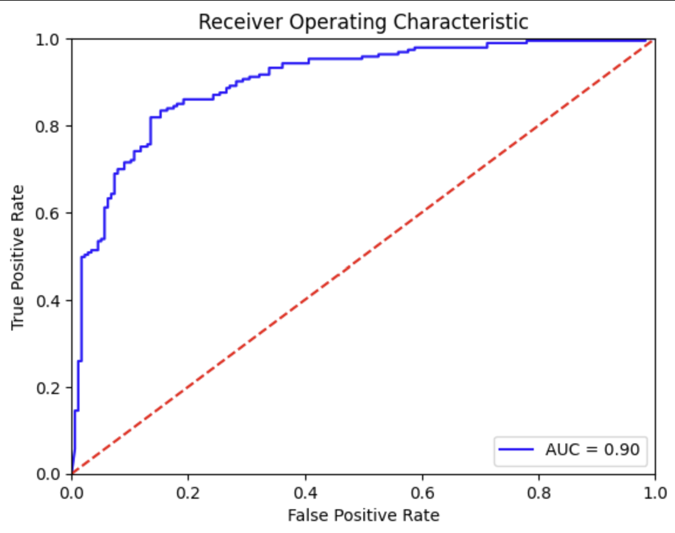
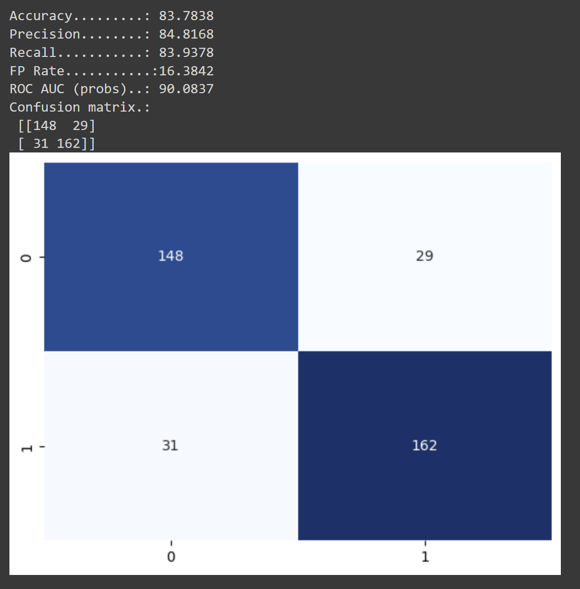
*KNN*

The results of the project were tested on the city of Perth. Perth was randomly selected as the test city for the results. After finding an optimal k value of 26, the KNN model yielded a 92 percent ROC AUC and an 84% F1 measure.

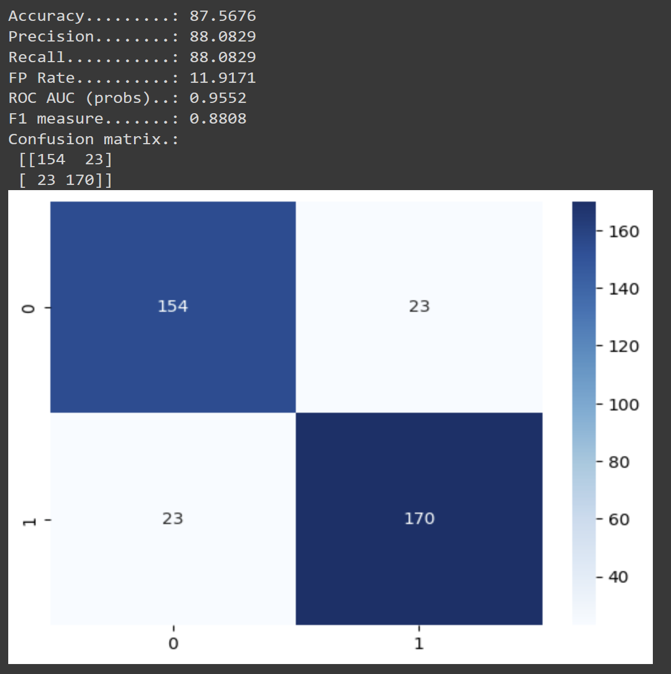
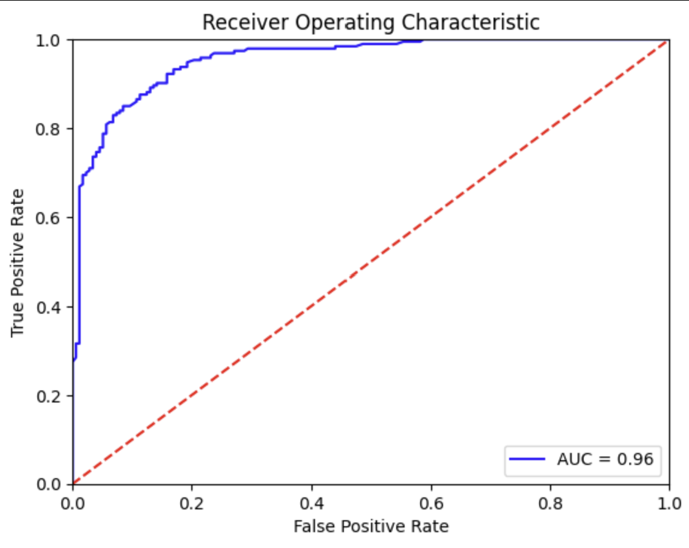
*Gaussian Naive Bayes*

Naive Bayes is commonly most effective with categorical data, however yielded good results in this study. It registered a ROC AUC of 90% and an accuracy of 83%. It was less effective than KNN and as will be seen the random forests but more effective than the LSTM.

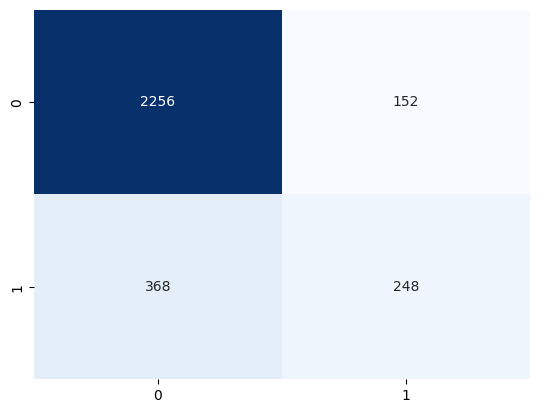
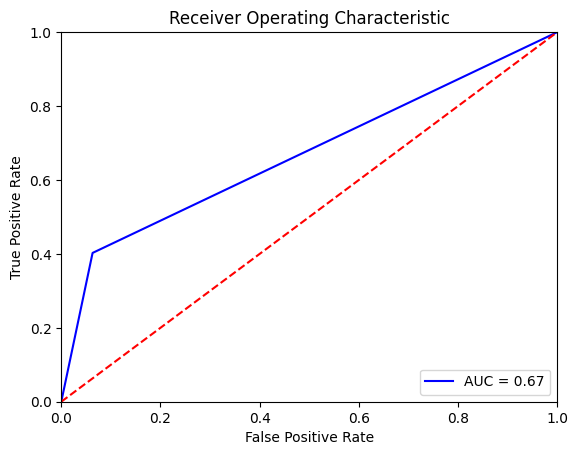
 

*Random Forests Classifier*

For the random forest classifier, the number of estimators was set to 500 and the random state was set to 1. The accuracy was 87%, the ROC AUC 95% and F1 measure was 88%. The random forest classifier had the highest ROC AUC, and F1 measure of all models. The random forest classifier did the best job at successfully predicting whether it would rain or not the next day of all models.



*Long Term Short Memory Neural Network*

****

For the LSTM network we did not randomly split the data into training and testing as it is important to feed the model with data in a time series format. The results were not as good as expected, the model needs some fine tuning. Due to time constraints and lack of experience, we were not able to enhance the performance of the LSTM to match the other models in this project. As can be seen from the picture of the ROC curve, with an AUC of 0.67, the model is not able to effectively distinguish between true positives and false positives.

**Appendix**

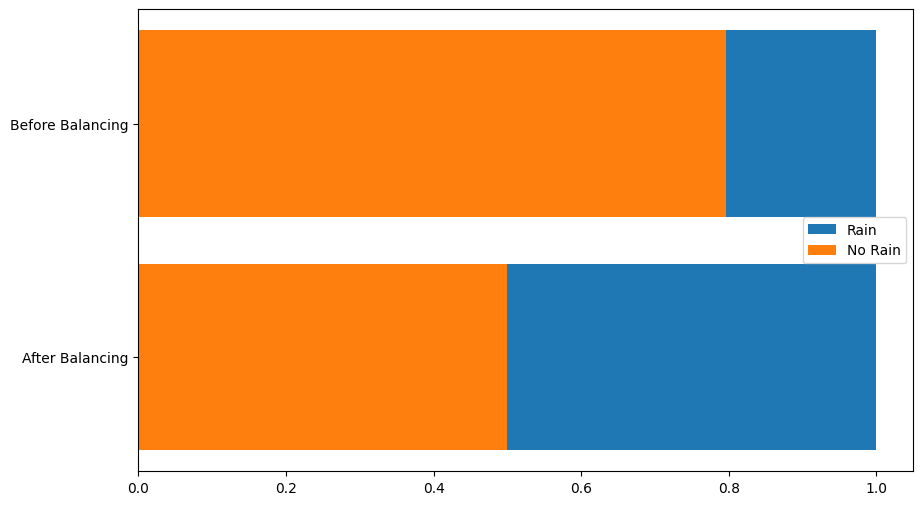


Figure 1. Distribution of “RainTomorrow” in the Dataset

A close-up of a number

Description automatically generated

Figure 2 Missing Measurements in the Dataset before preprocessing

A screenshot of a computer

Description automatically generated

Figure 3. Text Box to Choose a City.

Works Cited and Corresponding Code

Australian Bureau of Meteorology. (2024) Drought Statement March 2024 and outlook.

Retrieved From:

http://www.bom.gov.au/climate/data

Brenner, Steve. (2015). Re: What are the WRF - ARW weather model hardware and software requirements? Retrieved from: https://www.researchgate.net/post/What\_are\_the\_WRFARW\_weather\_model\_hardware\_and\_software\_requirements/5575da7a5e9d97487a8b442/citaion/download.

Kam Ho, Tin (1995) Random Decision Forests, AT&T Laboratories, https://web.archive.org/web/20160417030218/http://ect.bell-labs.com/who/tkh/publications/papers/odt.pdf

IBM (2024). “What is the K-nearest neighbors algorithm?” https://www.ibm.com/topics/knn#:~:text=The%20k%2Dnearest%20neighbors%20(KNN,used%20in%20machine%20learning%20today.

Lee, Peter M (2012), *Bayesian Statistics: An Introduction*, 4th edition. Wiley.

Link to the ipynb file used for this analysis:

https://colab.research.google.com/drive/1RmsL2enDSX44zgP2lHvu8M2UccyxTNPb?usp=sharing