Project Report

Group 6

Varun Kandukuri, Malvika Mohan, Sahana Bhat, Prathamesh Mahankal

Introduction:

Our project covers a deep dive analysis into weather forecasting and particularly determining factors that significantly affect rainfall. For our analysis we focus on the country of Australia, taking a dataset consisting of information regarding weather trends across different locations within the country for a span of ten years (2007-2017). We first perform an exploratory data analysis on our dataset, determining factors that are strongly correlated to each other. In order to gain better clarity on the weather conditions across the country, we perform a time series analysis across the various locations within Australia, determining how the weather patterns have varied throughout the years. We then go on to identifying the parameters that most significantly affect rainfall. On obtaining these factors, we train a logistic and SVM model to determine if our model can predict whether rainfall would occur or not based on these significant parameters. We go on to compare the two models to see which performs better based on several parameters. We finally conclude our project based on future scopes, implementing counterfactuals to examine the impact of global warming on the country.

Descriptive Data Analysis :

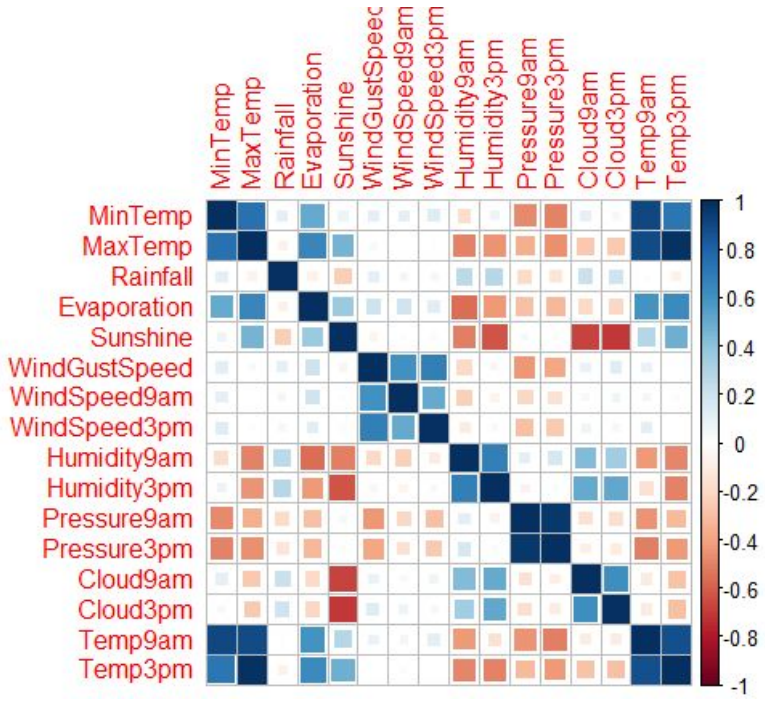
We obtained our dataset from Kaggle - Australia Rainfall Dataset (https://www.kaggle.com/jsphyg/weather-dataset-rattle-package).

The parameters we used to analyze the weather conditions in Australia are :

1. Location
2. Temperature Conditions - Minimum Temperature and Maximum Temperature (measured in Celsius)
3. Rainfall (measured in millimeters)
4. Sunshine (measured as the number of hours of sunshine per day)
5. Humidity at 9 am and 3 pm (measured as a percentage)
6. Wind Speed at 9 am and 3 pm (measured in km/hr)'
7. Cloud Cover measured at 9 am and 3 pm (measured in "oktas", which are a unit of eighths. It records how many eighths of the sky are obscured by clouds. A 0 measure indicates a completely clear sky whilst an 8 indicates that it is completely overcast.)
8. Atmospheric pressure measured at 9 am and 3 pm
9. Wind Direction at 9 am and 3 pm
10. Evaporation - Measured in millimeters in the span of 24 hours to 9 am

Exploratory Data Analysis :

On creating a correlation plot to determine the relationship between the various factors, we can see that there is a high negative correlation between cloud cover and sunshine. This is expected as from a real-world scenario we know that cloud cover reduces sunshine. Similarly, we can see there is a strong positive correlation between Evaporation and Maximum Temperature. This can also be associated with the fact that evaporation requires high temperatures to occur. Similarly, we can also see there is a strong negative correlation between humidity and Sunshine.



Best Feature Selection

To determine what factors significantly affect rainfall, we performed a best feature selection using step AIC method, that iteratively adds and removes predictions to determine the best subset of variables that gives a model with lowest prediction error. On keeping rainfall as our response variable, the set of predictor variables that significantly influence rainfall are :

1. Humidity at 9am
2. Sunshine
3. WindSpeed at 9am
4. Temperature at 3pm
5. Maximum Temperature
6. WindGustSpeed

With this set of important variables, let us pick up a case study and see how these variables will influence certain decisions.

Case Study

Now assume there is a person who wants to visit Australia for a vacation. Disregarding the current situation with the coronavirus, his next biggest headache will be whether or not it will rain at the location he wants to visit. Can we help that person to find out whether or not it will rain at that location on a given day? What features would we need to select in order to make a prediction like this? Let's find out!

What kind of problem is this?

For prediction, we will take our 6 most significant features and trained our models to predict whether it would rain or not. Next, we have to build a few models to make a prediction and then compare these models to see which one of them performs better on prediction. **Since this is a 'Yes' and 'No' problem, we clearly have to use the classification approach here.** To predict whether it would rain at a location based on the given weather conditions or not, we decided to use two well-known classification techniques: Logistic regression and Support Vector Machines (SVM).

Why did we decide to go with these models?

Our reasoning behind picking these two algorithms was simple—the dependent variable is binary in nature i.e. It either rains or it does not. SVM and logistic regression tend to work best when there are only two factors to deal with.

Logistic regression: is the appropriate regression analysis to conduct when the dependent variable is dichotomous (binary). Like all regression analyses, logistic regression is a predictive analysis. Logistic regression is used to describe data and to explain the relationship between one dependent binary variable and one or more nominal, ordinal, interval or ratio-level independent variables.

Support Vector Machines: SVM is a supervised machine learning algorithm that can be used for classification or regression problems. It uses a technique called the kernel trick to transform your data and then based on these transformations it finds an optimal boundary between the possible outputs. Simply put, it does some extremely complex data transformations, then figures out how to separate your data based on the labels or outputs you've defined.

Now let us go ahead and build the two models:

Logistic:

```
Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.5465  -0.6394  -0.3860  -0.1327   3.2296

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)   -7.581033   0.139583 -54.312   <2e-16 ***
Humidity9am    0.070223   0.001172  59.923   <2e-16 ***
Sunshine      -0.085340   0.004281 -19.935   <2e-16 ***
WindSpeed9am   0.026261   0.002070  12.684   <2e-16 ***
Temp3pm       -0.014302   0.011922  -1.200    0.230
MaxTemp        0.016407   0.011611   1.413    0.158
WindGustSpeed  0.033600   0.001253  26.811   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 44652  on 42314  degrees of freedom
Residual deviance: 34460  on 42308  degrees of freedom
AIC: 34474

Number of Fisher Scoring iterations: 5
```

SVM:

```
Parameters:
   SVM-Type:  C-classification
 SVM-Kernel:  radial
       cost:  1

Number of Support Vectors:  16041

 ( 7873 8168 )


Number of Classes:  2

Levels:
 No Yes
```

Now that we have fit the two models, we can calculate a confusion matrix to see how our predictions compare to the actual values.

```
          Reference
Prediction     0     1
         0 10400  2007
         1   576  1122

               Accuracy : 0.8169
                 95% CI : (0.8104, 0.8232)
    No Information Rate : 0.7782
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.3659

 Mcnemar's Test P-Value : < 2.2e-16

            Sensitivity : 0.9475
            Specificity : 0.3588
         Pos Pred Value : 0.8382
         Neg Pred Value : 0.6608
             Prevalence : 0.7782
         Detection Rate : 0.7373
   Detection Prevalence : 0.8796
      Balanced Accuracy : 0.6531

       'Positive' Class : 0
```

Logistic regression:

```
    No Information Rate : 0.7782
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.367

 Mcnemar's Test P-Value : < 2.2e-16

            Sensitivity : 0.9643
            Specificity : 0.3214
         Pos Pred Value : 0.8350
         Neg Pred Value : 0.7257
             Prevalence : 0.7782
         Detection Rate : 0.7504
   Detection Prevalence : 0.8987
      Balanced Accuracy : 0.6479

       'Positive' Class : No
```

SVM:

Model Comparison:

More often than not, we judge the performance of a particular model by the accuracy with which that model makes correct predictions. This measure of performance is perfectly valid when it comes to training data that had a balanced number of observations for each class. However, in our model, there is a severe class imbalance between the 'Yes' and the 'No' class. Thus, accuracy will not be a good metric to judge the performance of a model which is built on such an imbalanced set of training data.

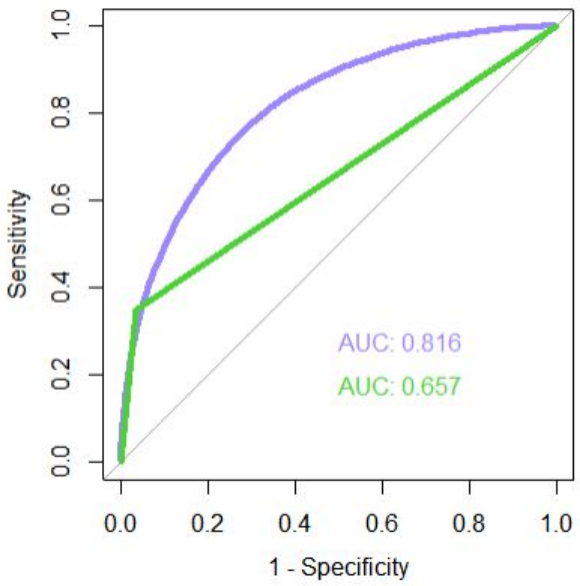
To overcome this issue, we can perform upsampling or downsampling on our training data so that we get an equal number of observations for each class. If we do not want to tinker our training data, we can use some model performance metrics that take this imbalance into account while judging model performance for classification.

Thus, to judge which model performed better in our case, we have employed the following metrics:-

1) **Sensitivity**

Since even the slightest inkling of rain can ruin a short vacation, we use sensitivity i.e. our models will err on the side of rain even if the weather conditions are mildly harsh.
However, in the above case we manually set the threshold for classification at 0.5 for logistic regression. We wanted to explore which model performs better at varying thresholds.

2) **ROC curves**



Purple: Logistic Regression
Green: SVM

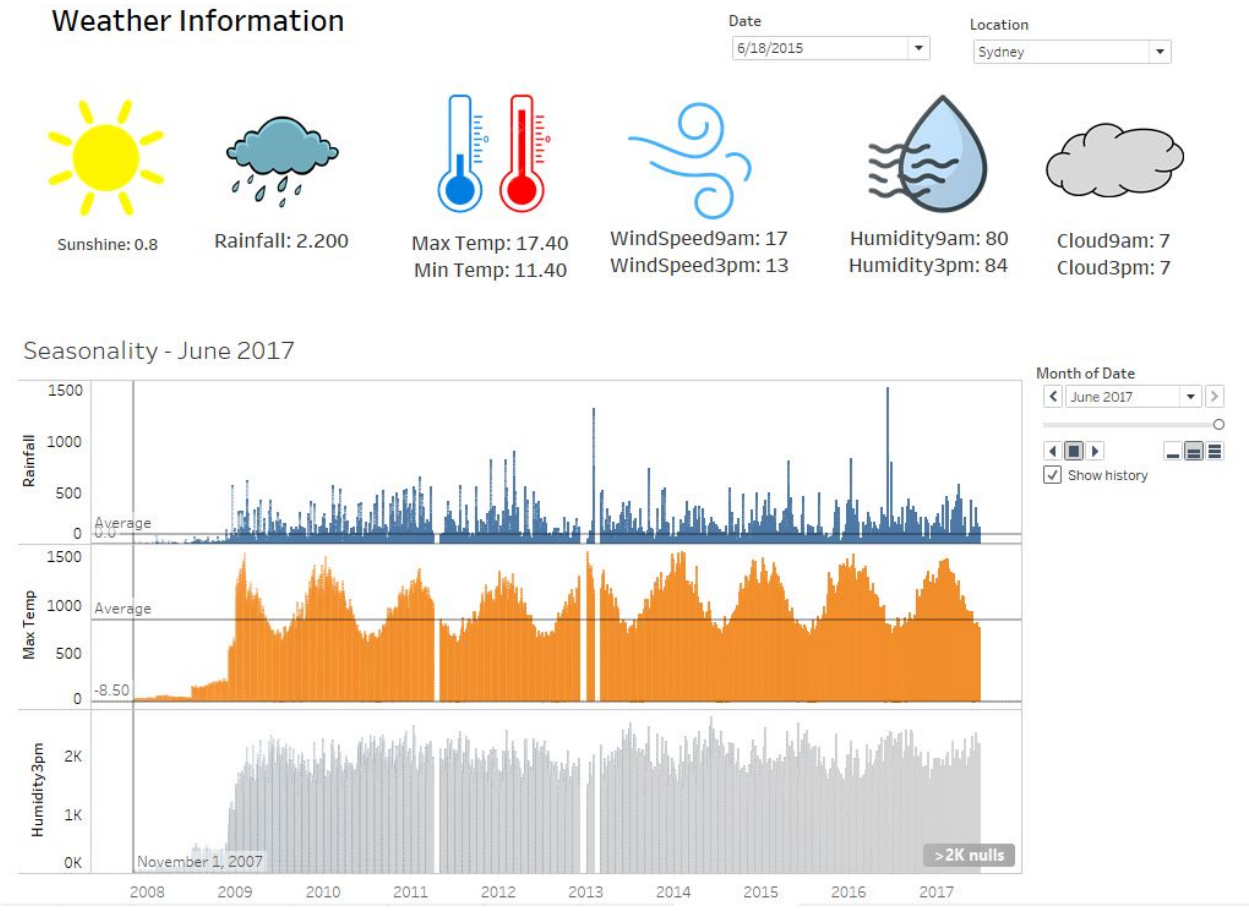
Judging by the area under the curve, Logistic Regression clearly has better performance for the given data.

Visualizing trends and patterns in Weather conditions

Visualizing the dataset helps us identify the trends and patterns in the weather conditions over the time across various locations in Australia. It can help us answer the questions such as,

- Which locations receive the highest amount of Rainfall ?
- What is the lowest and highest Temperature in a particular location?
- What is the similarity in trends of Rainfall and Max Temp?

A Tableau dashboard was built to visualize the most significant variables identified using step AIC to obtain the best fit model. The variables represented are: Sunshine, Rainfall, Max & Min Temperature, WindSpeed at 9am & 3pm, Humidity at 9am & 3pm, Cloud cover at 9am & 3pm. The dashboard has filters to select a particular Location and Date. The measure for each variable is updated based on the selected values. Apart from this, to identify the trends and seasonality in the selected variables, we have built bar charts representing the measures of the variables along the time period. We also have a play button that can be used to graphically animate the growth of the variable across. The dashboard has been published to Tableau Public.(https://public.tableau.com/views/IMT574_Group6_WeatherDashboard/Dashboard3?:display_count=y&publish=yes&:origin=viz_share_link)



Recurrent Neural Networks(RNN) to forecast weather conditions

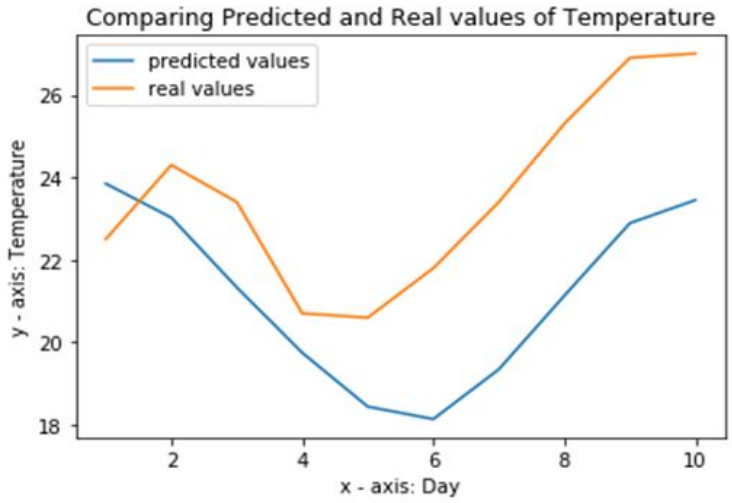
RNN is a deep learning model that is used for Time-series prediction. Unlike traditional neural networks, recurrent networks use their memory (also called states) to predict sequence outputs. Such controlled states are part of Long Short-Term Memory networks (LSTMs). Bi-directional RNNs use a finite sequence to predict or label each element of the sequence based on the element's past and future contexts. This is done by concatenating the outputs of two RNNs, one processing the sequence from left to right, the other one from right to left. In simple words, RNN is used when we want to predict a future outcome based on the previous sequential inputs.

The RNN model is built to forecast values of Max Temperature for the next 10 days using the values from the previous 30 days as an input. The model is built using Keras library in Python. A keras sequential model with keras layers of LSTM, Dense and dropout is trained. A sequential model is created by adding layers sequentially. The first layer is a Bidirectional LSTM with 30 memory units. With Bidirectional LSTM the output layer gets feedback from past(forward) as well as future(backward) states simultaneously. We add 3 hidden layers and an output layer with a linear activation function that outputs 10 days temperature. The RNN model is fit using the training data which consists of a univariate dataset of maximum temperature i.e Max Temp values subset from the complete dataset. The summary of the model trained is as below:

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
bidirectional_1 (Bidirection (None, 30, 60)            7680
_________________________________________________________________
dropout_1 (Dropout)          (None, 30, 60)            0
_________________________________________________________________
lstm_2 (LSTM)                (None, 30, 30)            10920
_________________________________________________________________
dropout_2 (Dropout)          (None, 30, 30)            0
_________________________________________________________________
lstm_3 (LSTM)                (None, 30, 30)            7320
_________________________________________________________________
dropout_3 (Dropout)          (None, 30, 30)            0
_________________________________________________________________
lstm_4 (LSTM)                (None, 30)                7320
_________________________________________________________________
dropout_4 (Dropout)          (None, 30)                0
_________________________________________________________________
dense_1 (Dense)              (None, 10)                310
=================================================================
Total params: 33,550
Trainable params: 33,550
Non-trainable params: 0
```

The last 10 observations from the original dataset are used as set as real temperature values to compare with the predicted outputs. The next 30 observations above these are set as testing

dataset and are provided as input to the model. The output results are as tabulated below. The graph obtained when comparing with real values of temperature recorded is as shown.



Real values as measured	**Predicted values** from the model
22.5	21.89
24.3	21.65
23.4	21.07
20.7	20.53
20.6	20.52
21.8	20.48
23.4	20.70
25.3	21.29
26.9	21.98
27.0	22.56

Inference: The calculated Mean Absolute Error is 2.725 and Root Mean Square Error RMSE is 2.986. The model has performed well for the number of observations used to train the model which was just 2000 and the number of epochs set to 500.

Improving the RNN model

As Recurrent Neural Networks take up a lot of memory for processing, to increase the accuracy of the mode, we can try to:

- Increase or decrease the number of epochs.
- Use a large dataset and train the model, but this might take a longer time.

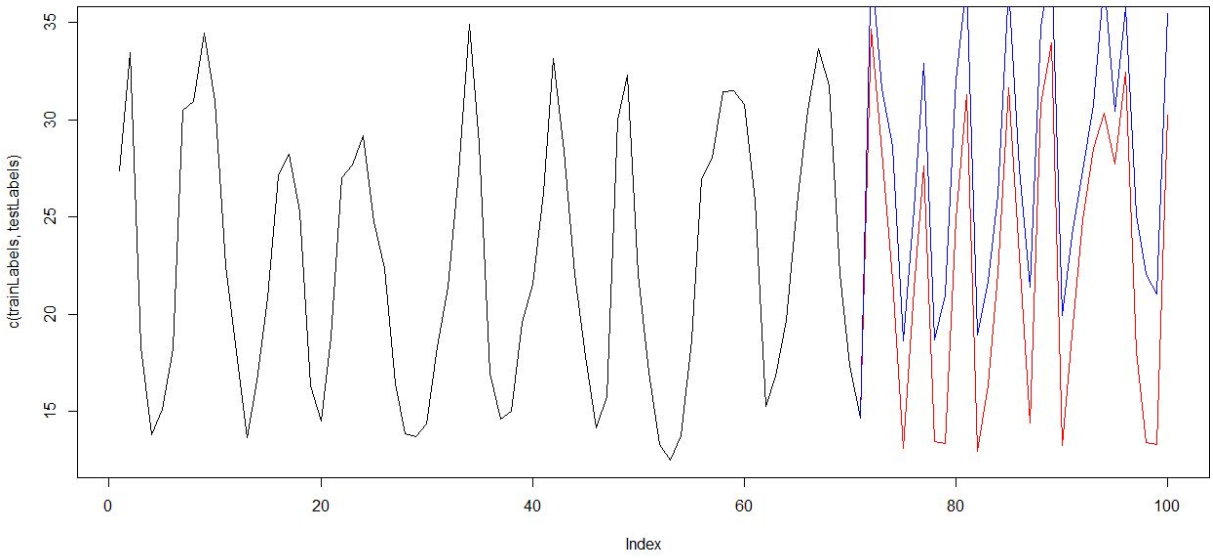
Reinforcement Learning

Reinforcement learning (RL) is an area of machine learning which tells us how particular agents ought to take actions in an environment in order to maximize some notion of cumulative reward. As mentioned by Prof. Shah in class, Supervised Learning is a task-driven process while Unsupervised Learning is a data-driven process. Reinforcement Learning, on the other hand, is an environment-driven process.

Consider that you wish to know ‘What If’ a particular change takes place; what would the effects of that change and how would these effects compare to the effects in case the change was not made. This is where Counterfactual Learning comes into the picture. A counterfactual explanation describes a causal situation in the form: “If X had not occurred, Y would not have

occurred". For example: "If I hadn't taken a sip of this hot coffee, I wouldn't have burned my tongue". Event Y is that I burned my tongue; cause X is that I had a hot coffee. Thinking in counterfactuals requires imagining a hypothetical reality that contradicts the observed facts (e.g. a world in which I have not drunk the hot coffee), hence the name "counterfactual". **In summary, a counterfactual explanation of a prediction describes the smallest change to the feature values that changes the prediction to a predefined output.**

In our case, we ran a counterfactual analysis to see what is the difference between the effects that have been observed in the Maximum Temperature recorded, and what would the effects be in case the changes were not made in the first place. As you can see in the graph below, the temperature values in red were the ones that were expected, and the values in blue are the ones that are observed. **This confirms that the maximum temperature in Australia has actually increased and that global warming is real.**



Conclusion:

Due to its drastic variations throughout the year, month or even day, analyzing the climate of a particular region is always a challenge. This is exactly what makes the life of the people from the weather forecast department so difficult. However, one thing is for sure that using all the various methods we have mentioned above, the weather for a particular day can be estimated as closely as one can.

GitHub Link for the project:
https://github.com/prathameshmahankal/Predicting-Rainfall-Using-Machine-Learning

References:

SVM: https://community.alteryx.com/t5/Data-Science-Blog/Why-use-SVM/ba-p/138440
Logistic regression: https://www.statisticssolutions.com/what-is-logistic-regression/
Recurrent Neural Networks:
https://medium.com/analytics-vidhya/weather-forecasting-with-recurrent-neural-networks-1eaa057d70c3
Counterfactuals in Reinforcement Learning:
https://christophm.github.io/interpretable-ml-book/counterfactual.html