Table of Contents:

COMSATS University Islamabad
Department of Computer Science
Data Science Fundamentals (DSC293) – BDS-2A
Project (CLO-5)

Introduction:

This project documentation focuses on our analysis of the "Weather Australia Dataset" obtained from a reliable source. Our goal was to extract valuable insights and predictions from the dataset to gain a deeper understanding of weather patterns in Australia.

To achieve this, we followed a systematic approach, starting with a thorough exploration of the dataset. We examined the variables, their distributions, and any missing values. Next, we performed data wrangling operations to clean and transform the data into a tidy format, ensuring it was suitable for analysis.

Afterwards, we identified a specific predictive problem related to weather conditions and selected a suitable predictive algorithm. We applied linear regression to predict rainfall based on humidity levels, and logistic regression to forecast the likelihood of rainfall tomorrow based on maximum temperature. We evaluated the performance of these models and calculated relevant metrics such as mean squared error and accuracy.

To enhance the visual representation of our findings, we employed various visualization techniques. Scatter plots were used to illustrate the relationship between humidity and rainfall, and line plots showcased the overall rainfall trends. We also utilized bar graphs to display the frequency distribution of specific weather conditions.

Additionally, we explored clustering using the k-means algorithm to identify distinct weather patterns within the dataset. We performed clustering analysis based on multiple weather variables, creating clusters and visualizing them on scatter plots. This allowed us to gain insights into different weather clusters and their characteristics.

To compare the results of our predictive algorithms and clustering analysis, we employed box plots. These plots provided a visual representation of the stability and consistency of the outcomes produced by each algorithm, aiding in the selection of the most reliable method.

Finally, we developed an interactive interface that showcases the entire data analysis process, including data wrangling, predictive modeling, comparison, and visualization. This interface enables users to explore the dataset, interact with the models, and gain a comprehensive understanding of the weather patterns and predictions.

Through this project, our aim was to demonstrate the power of data analysis and predictive modeling in uncovering insights and making informed decisions using the Weather Australia Dataset. By following a systematic approach and employing various techniques, we were able to extract valuable information and showcase the potential applications of this dataset in understanding and predicting weather conditions in Australia.

Information About Dataset:

The Weather Australia dataset is a comprehensive collection of weather-related observations recorded across various locations in Australia. It provides valuable information about meteorological conditions such as temperature, rainfall, humidity, wind speed, and atmospheric pressure. This dataset is widely used by researchers, weather forecasters, and data analysts to study climate patterns, analyze weather trends, and develop predictive models.

The dataset encompasses a substantial time span, typically spanning several years, which enables the exploration of seasonal variations and long-term climate patterns. It includes a diverse range of variables, allowing for in-depth analysis of how different weather factors interact and influence each other.

One of the key advantages of the Weather Australia dataset is its spatial coverage. It contains observations from multiple locations across the country, including major cities, regional areas, and remote stations. This geographical diversity facilitates the examination of weather patterns on both a local and regional scale, and it offers insights into the unique climate characteristics of different regions within Australia.

Researchers and data analysts can leverage this dataset to explore various research questions and hypotheses. It enables the investigation of factors influencing rainfall patterns, the relationship between temperature and humidity, the impact of wind on weather conditions, and much more. The dataset also serves as a valuable resource for training and evaluating predictive models that aim to forecast future weather events or assess the likelihood of specific weather phenomena.

Libraries & Functions used:

We used the following libraries & packages in R-script to perform the analysis:

```
library(tidyr)
library(dplyr)
library(ggplot2)
library(lubridate)
library(scales)
library(ggthemes)
library(randomForest)
library(mdsr)
library(tidyverse)
library(tidytext)
library(DT)
library(ggfortify)
```

Snapshots of code along with their output:

Exploring Dataset:

	ID	Exercise	Calories.Burn	Dream.Weight	Actual.Weight	Age	Gender	Duration	Heart.Rate	BMI	Weather.Conditions	Exercise.Intensity
1	1	Exercise 2	286.9599	91.89253	96.30112	45	Male	37	170	29.42627	Rainy	5
2	2	Exercise 7	343.4530	64.16510	61.10467	25	Male	43	142	21.28635	Rainy	5
3	3	Exercise 4	261.2235	70.84622	71.76672	20	Male	20	148	27.89959	Cloudy	4
4	4	Exercise 5	127.1839	79.47701	82.98446	33	Male	39	170	33.72955	Sunny	10
5	5	Exercise 10	416.3184	89.96023	85.64317	29	Female	34	118	23.28611	Cloudy	3
6	6	Exercise 1	479.7227	78.88758	NA	60	Female	41	169	34.71934	Rainy	10
7	7	Exercise 9	457.6314	65.68113	61.81539	18	Male	53	103	34.59464	Cloudy	10
8	8	Exercise 4	272.9570	64.92956	62.80649	42	Male	25	104	22.05010	Cloudy	2
9	9	Exercise 10	195.0325	52.73107	54.53769	49	Male	37	161	30.94885	Sunny	1
10	10	Exercise 8	259.5311	95.16410	97.43683	NA	Male	55	103	31.22404	Cloudy	10
11	11	Exercise 5	248.5361	56.82978	54.14440	41	Male	52	151	34.01757	Cloudy	3

```r
# 2: EXPLORING THE DATASET

# Displaying the dataset
str(Weather)

# to display few rows
head(Weather)
view(Weather)

#  To see overview of the dataset along with the first few values of each variable
glimpse(Weather)

# for the Summary statistics of our dataset
summary(Weather)

# Check the column names
colnames(Weather)
```

Output:

Str:

```
> str(Weather)
tibble [99,516 x 23] (S3: tbl_df/tbl/data.frame)
 $ row ID       : chr [1:99516] "Row0" "Row1" "Row2" "Row3" ...
 $ Location     : chr [1:99516] "Albury" "Albury" "Albury" "Albury" ...
 $ MinTemp      : num [1:99516] 13.4 7.4 17.5 14.6 7.7 13.1 13.4 15.9 12.6 9.8 ...
 $ MaxTemp      : num [1:99516] 22.9 25.1 32.3 29.7 26.7 30.1 30.4 21.7 21 27.7 ...
 $ Rainfall     : num [1:99516] 0.6 0 1 0.2 0 1.4 0 2.2 3.6 NA ...
 $ Evaporation  : logi [1:99516] NA NA NA NA NA NA ...
 $ Sunshine     : logi [1:99516] NA NA NA NA NA NA ...
 $ WindGustDir  : chr [1:99516] "W" "WNW" "W" "WNW" ...
 $ WindGustSpeed: num [1:99516] 44 44 41 56 35 28 30 31 44 50 ...
 $ WindDir9am   : chr [1:99516] "W" "NNW" "ENE" "W" ...
 $ WindDir3pm   : chr [1:99516] "WNW" "WSW" "NW" "W" ...
 $ WindSpeed9am : num [1:99516] 20 4 7 19 6 15 17 15 24 NA ...
 $ WindSpeed3pm : num [1:99516] 24 22 20 24 17 11 6 13 20 22 ...
 $ Humidity9am  : num [1:99516] 71 44 82 55 48 58 48 89 65 50 ...
 $ Humidity3pm  : num [1:99516] 22 25 33 23 19 27 22 91 43 28 ...
 $ Pressure9am  : num [1:99516] 1008 1011 1011 1009 1013 ...
 $ Pressure3pm  : num [1:99516] 1007 1008 1006 1005 1010 ...
 $ Cloud9am     : num [1:99516] 8 NA 7 NA NA NA NA 8 NA 0 ...
 $ Cloud3pm     : num [1:99516] NA NA 8 NA NA NA NA 8 7 NA ...
 $ Temp9am      : num [1:99516] 16.9 17.2 17.8 20.6 16.3 20.1 20.4 15.9 15.8 17.3 ...
 $ Temp3pm      : num [1:99516] 21.8 24.3 29.7 28.9 25.5 28.2 28.8 17 19.8 26.2 ...
 $ RainToday    : chr [1:99516] "No" "No" "No" "No" ...
 $ RainTomorrow : num [1:99516] 0 0 0 0 0 0 1 1 0 0 ...
```

Head:

```
> head(Weather)
# A tibble: 6 x 23
  `row ID` Location MinTemp MaxTemp Rainfall Evaporation Sunshine WindGustDir WindGustSpeed WindDir9am
  <chr>    <chr>      <dbl>   <dbl>    <dbl> <lgl>       <lgl>    <chr>               <dbl> <chr>
1 Row0     Albury      13.4    22.9      0.6 NA          NA       W                      44 W
2 Row1     Albury       7.4    25.1      0   NA          NA       WNW                    44 NNW
3 Row2     Albury      17.5    32.3      1   NA          NA       W                      41 ENE
4 Row3     Albury      14.6    29.7      0.2 NA          NA       WNW                    56 W
5 Row4     Albury       7.7    26.7      0   NA          NA       W                      35 SSE
6 Row5     Albury      13.1    30.1      1.4 NA          NA       W                      28 S
# i 13 more variables: WindDir3pm <chr>, WindSpeed9am <dbl>, WindSpeed3pm <dbl>, Humidity9am <dbl>,
#   Humidity3pm <dbl>, Pressure9am <dbl>, Pressure3pm <dbl>, Cloud9am <dbl>, Cloud3pm <dbl>, Temp9am <dbl>,
#   Temp3pm <dbl>, RainToday <chr>, RainTomorrow <dbl>
```

glimpse:

```
> glimpse(Weather)
Rows: 99,516
Columns: 23
$ `row ID`       <chr> "Row0", "Row1", "Row2", "Row3", "Row4", "Row5", "Row6", "Row7", "Row8", "Row9", "Row10",~
$ Location       <chr> "Albury", "Albury", "Albury", "Albury", "Albury", "Albury", "Albury", "Albury", "Albury"~
$ MinTemp        <dbl> 13.4, 7.4, 17.5, 14.6, 7.7, 13.1, 13.4, 15.9, 12.6, 9.8, 14.1, 13.5, 11.2, 9.8, 17.1, 20~
$ MaxTemp        <dbl> 22.9, 25.1, 32.3, 29.7, 26.7, 30.1, 30.4, 21.7, 21.0, 27.7, 20.9, 22.9, 22.5, 25.6, 33.0~
$ Rainfall       <dbl> 0.6, 0.0, 1.0, 0.2, 0.0, 1.4, 0.0, 2.2, 3.6, NA, 0.0, 16.8, 10.6, 0.0, 0.0, 0.0, 0.0, 0.~
$ Evaporation    <lgl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, ~
$ Sunshine       <lgl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, ~
$ WindGustDir    <chr> "W", "WNW", "W", "WNW", "W", "W", "N", "NNE", "SW", "WNW", "ENE", "W", "SSE", "SSE", "NE~
$ WindGustSpeed  <dbl> 44, 44, 41, 56, 35, 28, 30, 31, 44, 50, 22, 63, 43, 26, 43, 41, 33, 43, 57, 48, 46, 50, ~
$ WindDir9am     <chr> "W", "NNW", "ENE", "W", "SSE", "S", "SSE", "NE", "W", "NA", "SSW", "N", "WSW", "SE", "NE~
$ WindDir3pm     <chr> "WNW", "WSW", "NW", "W", "W", "SSE", "ESE", "ENE", "SSW", "WNW", "E", "WNW", "SW", "NNW"~
$ WindSpeed9am   <dbl> 20, 4, 7, 19, 6, 15, 17, 15, 24, NA, 11, 6, 24, 17, 17, 19, 6, 4, 0, 13, 19, 11, 19, 11,~
$ WindSpeed3pm   <dbl> 24, 22, 20, 24, 17, 11, 6, 13, 20, 22, 9, 20, 17, 6, 22, 20, 13, 19, 26, 30, 30, 22, 11,~
$ Humidity9am    <dbl> 71, 44, 82, 55, 48, 58, 48, 89, 65, 50, 69, 80, 47, 45, 38, 54, 55, 49, 41, 56, 49, 78, ~
$ Humidity3pm    <dbl> 22, 25, 33, 23, 19, 27, 22, 91, 43, 28, 82, 65, 32, 26, 28, 24, 23, 17, 28, 15, 22, 70, ~
$ Pressure9am    <dbl> 1007.7, 1010.6, 1010.8, 1009.2, 1013.4, 1007.0, 1011.8, 1010.5, 1001.2, 1013.4, 1012.2, ~
$ Pressure3pm    <dbl> 1007.1, 1007.8, 1006.0, 1005.4, 1010.1, 1005.7, 1008.7, 1004.2, 1001.8, 1010.3, 1010.4, ~
$ Cloud9am       <dbl> 8, NA, 7, NA, NA, NA, NA, 8, NA, 0, 8, 8, NA, NA, NA, NA, 5, NA, NA, NA, NA, 8, NA, NA, ~
$ Cloud3pm       <dbl> NA, NA, 8, NA, NA, NA, NA, 8, 7, NA, 1, 1, 2, NA, 1, NA, NA, NA, 1, NA, NA, 8, NA, NA, N~
$ Temp9am        <dbl> 16.9, 17.2, 17.8, 20.6, 16.3, 20.1, 20.4, 15.9, 15.8, 17.3, 17.2, 18.0, 15.5, 15.8, 24.5~
$ Temp3pm        <dbl> 21.8, 24.3, 29.7, 28.9, 25.5, 28.2, 28.8, 17.0, 19.8, 26.2, 18.1, 21.5, 21.0, 23.2, 31.6~
$ RainToday      <chr> "No", "No", "No", "No", "No", "Yes", "No", "Yes", "Yes", "NA", "No", "Yes", "Yes", "No",~
$ RainTomorrow   <dbl> 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0~
```

Summary:

```
> summary(weather)
    row ID              Location            MinTemp          MaxTemp          Rainfall         Evaporation    
 Length:99516       Length:99516       Min.   :-8.50   Min.   :-4.10   Min.   :  0.000   Mode :logical  
 Class :character   Class :character   1st Qu.: 7.60   1st Qu.:17.90   1st Qu.:  0.000   FALSE:2851     
 Mode  :character   Mode  :character   Median :12.00   Median :22.60   Median :  0.000   TRUE :54134    
                                       Mean   :12.18   Mean   :23.22   Mean   :  2.353   NA's :42531    
                                       3rd Qu.:16.80   3rd Qu.:28.20   3rd Qu.:  0.800                  
                                       Max.   :33.90   Max.   :48.10   Max.   :371.000                  
                                       NA's   :443     NA's   :230     NA's   :979                      
  Sunshine        WindGustDir        WindGustSpeed     WindDir9am         WindDir3pm         WindSpeed9am
 Mode :logical   Length:99516       Min.   :  6.00   Length:99516       Length:99516       Min.   :  0  
 FALSE:3973      Class :character   1st Qu.: 31.00   Class :character   Class :character   1st Qu.:  7  
 TRUE :48226     Mode  :character   Median : 39.00   Mode  :character   Mode  :character   Median : 13  
 NA's :47317                        Mean   : 39.98                                         Mean   : 14  
                                    3rd Qu.: 48.00                                         3rd Qu.: 19  
                                    Max.   :135.00                                         Max.   :130  
                                    NA's   :6480                                           NA's   :935  
  WindSpeed3pm    Humidity9am      Humidity3pm       Pressure9am      Pressure3pm        Cloud9am    
 Min.   : 0.00   Min.   :  0.00   Min.   :  0.00   Min.   : 980.5   Min.   : 978.2   Min.   :0.00  
 1st Qu.:13.00   1st Qu.: 57.00   1st Qu.: 37.00   1st Qu.:1013.0   1st Qu.:1010.5   1st Qu.:1.00  
 Median :19.00   Median : 70.00   Median : 52.00   Median :1017.7   Median :1015.3   Median :5.00  
 Mean   :18.65   Mean   : 68.87   Mean   : 51.43   Mean   :1017.7   Mean   :1015.3   Mean   :4.45  
 3rd Qu.:24.00   3rd Qu.: 83.00   3rd Qu.: 65.00   3rd Qu.:1022.4   3rd Qu.:1020.0   3rd Qu.:7.00  
 Max.   :87.00   Max.   :100.00   Max.   :100.00   Max.   :1041.0   Max.   :1039.6   Max.   :9.00  
 NA's   :1835    NA's   :1233     NA's   :2506     NA's   :9748     NA's   :9736     NA's   :37572 
    Cloud3pm        Temp9am          Temp3pm         RainToday          RainTomorrow   
 Min.   :0.00   Min.   :-7.00   Min.   :-5.10   Length:99516       Min.   :0.0000  
 1st Qu.:2.00   1st Qu.:12.30   1st Qu.:16.60   Class :character   1st Qu.:0.0000  
 Median :5.00   Median :16.70   Median :21.10   Mode  :character   Median :0.0000  
 Mean   :4.52   Mean   :16.97   Mean   :21.68                      Mean   :0.2247  
 3rd Qu.:7.00   3rd Qu.:21.50   3rd Qu.:26.40                      3rd Qu.:0.0000  
```

colnames:

```
> # Check the column names
> colnames(Weather)
 [1] "row ID"        "Location"      "MinTemp"       "MaxTemp"       "Rainfall"      "Evaporation"
 [7] "Sunshine"      "WindGustDir"   "WindGustSpeed" "WindDir9am"    "WindDir3pm"    "WindSpeed9am"
[13] "WindSpeed3pm"  "Humidity9am"   "Humidity3pm"   "Pressure9am"   "Pressure3pm"   "Cloud9am"
[19] "Cloud3pm"      "Temp9am"       "Temp3pm"       "RainToday"     "RainTomorrow"
```

Data Wrangling:

```
# 3: WRANGLING

#Filtering Rows: Select only the Bendigo's data.
Weather_w <- filter(Weather, Location == "Bendigo")
Weather_w

# Delete two columns
Weather_w <- subset(Weather_w, select = -c(Evaporation, Sunshine))
Weather_w

# Finding the missing values
missing_values <- sum(is.na(Weather_w))
missing_values

# Removing Missing Values
Weather_w <- na.omit(Weather_w)
View(Weather_w)
```

Output:

Filter rows:

```
> Weather_w <- filter(Weather, Location == "Bendigo")
> Weather_w
# A tibble: 2,110 x 23
   `row ID` Location MinTemp MaxTemp Rainfall Evaporation Sunshine WindGustDir WindGustSpeed WindDir9am
   <chr>    <chr>      <dbl>   <dbl>    <dbl> <lgl>       <lgl>    <chr>               <dbl> <chr>
 1 Row40730 Bendigo      9.1    21.7      0   TRUE        NA       W                      44 WNW
 2 Row40731 Bendigo     10.8    23.7      0   TRUE        NA       WSW                    44 SW
 3 Row40732 Bendigo      8.2    24.6      0   TRUE        NA       NNE                    41 SE
 4 Row40733 Bendigo     15.1    30.3      0.2 TRUE        NA       NW                     54 NW
 5 Row40734 Bendigo      9.9    27.2      0.2 TRUE        NA       WNW                    54 WNW
 6 Row40735 Bendigo      7.8    25.5      0   TRUE        NA       W                      35 S
 7 Row40736 Bendigo      8.7    28.7      0.2 TRUE        NA       SSE                    43 SSE
 8 Row40737 Bendigo     16.5    19.9      0   TRUE        NA       ENE                    44 NE
 9 Row40738 Bendigo     14.1    20.2     30.4 FALSE       NA       SW                     70 WNW
10 Row40739 Bendigo     13.5    22        1.2 TRUE        NA       N                      24 SSE
# i 2,100 more rows
# i 13 more variables: WindDir3pm <chr>, WindSpeed9am <dbl>, WindSpeed3pm <dbl>, Humidity9am <dbl>,
```

Deleting two columns:

```
# i Use `print(n = ...)` to see more rows
> Weather_w <- subset(Weather_w, select = -c(Evaporation, Sunshine))
> Weather_w
# A tibble: 2,110 x 21
   `row ID` Location MinTemp MaxTemp Rainfall WindGustDir WindGustSpeed WindDir9am WindDir3pm WindSpeed9am
   <chr>    <chr>      <dbl>   <dbl>    <dbl> <chr>               <dbl> <chr>      <chr>             <dbl>
 1 Row40730 Bendigo      9.1    21.7      0   W                      44 WNW        WSW                  20
 2 Row40731 Bendigo     10.8    23.7      0   WSW                    44 SW         W                    17
 3 Row40732 Bendigo      8.2    24.6      0   NNE                    41 SE         NNE                  17
 4 Row40733 Bendigo     15.1    30.3      0.2 NW                     54 NW         W                    19
 5 Row40734 Bendigo      9.9    27.2      0.2 WNW                    54 WNW        NW                    9
 6 Row40735 Bendigo      7.8    25.5      0   W                      35 S          WSW                  11
 7 Row40736 Bendigo      8.7    28.7      0.2 SSE                    43 SSE        ESE                  22
 8 Row40737 Bendigo     16.5    19.9      0   ENE                    44 NE         ENE                  13
 9 Row40738 Bendigo     14.1    20.2     30.4 SW                     70 WNW        SW                   13
10 Row40739 Bendigo     13.5    22        1.2 N                      24 SSE        NNE                   4
# i 2,100 more rows
```

Find missing values:

```
# i Use `print(n = ...)` to see more rows
> missing_values <- sum(is.na(Weather_w))
> missing_values
[1] 1232
>
```

Tidy data :

Output

Renaming:

```
# A tibble: 1,323 x 21
   `row ID` Location MinTemp MaxTemp Rainfall WindGustDirection WindGustSpeed WindDir9am WindDir3pm WindSpeed9am
   <chr>    <chr>      <dbl>   <dbl>    <dbl> <chr>                     <dbl> <chr>      <chr>             <dbl>
 1 Row40730 Bendigo      9.1    21.7      0   W                            44 WNW        WSW                  20
 2 Row40731 Bendigo     10.8    23.7      0   WSW                          44 SW         W                    17
 3 Row40732 Bendigo      8.2    24.6      0   NNE                          41 SE         NNE                  17
 4 Row40733 Bendigo     15.1    30.3      0.2 NW                           54 NW         W                    19
 5 Row40734 Bendigo      9.9    27.2      0.2 WNW                          54 WNW        NW                    9
 6 Row40735 Bendigo      7.8    25.5      0   W                            35 S          WSW                  11
 7 Row40736 Bendigo      8.7    28.7      0.2 SSE                          43 SSE        ESE                  22
 8 Row40737 Bendigo     16.5    19.9      0   ENE                          44 NE         ENE                  13
 9 Row40738 Bendigo     14.1    20.2     30.4 SW                           70 WNW        SW                   13
10 Row40739 Bendigo     13.5    22        1.2 N                            24 SSE        NNE                   4
# i 1,313 more rows
```

Choose a predictive algorithm to solve your problem:

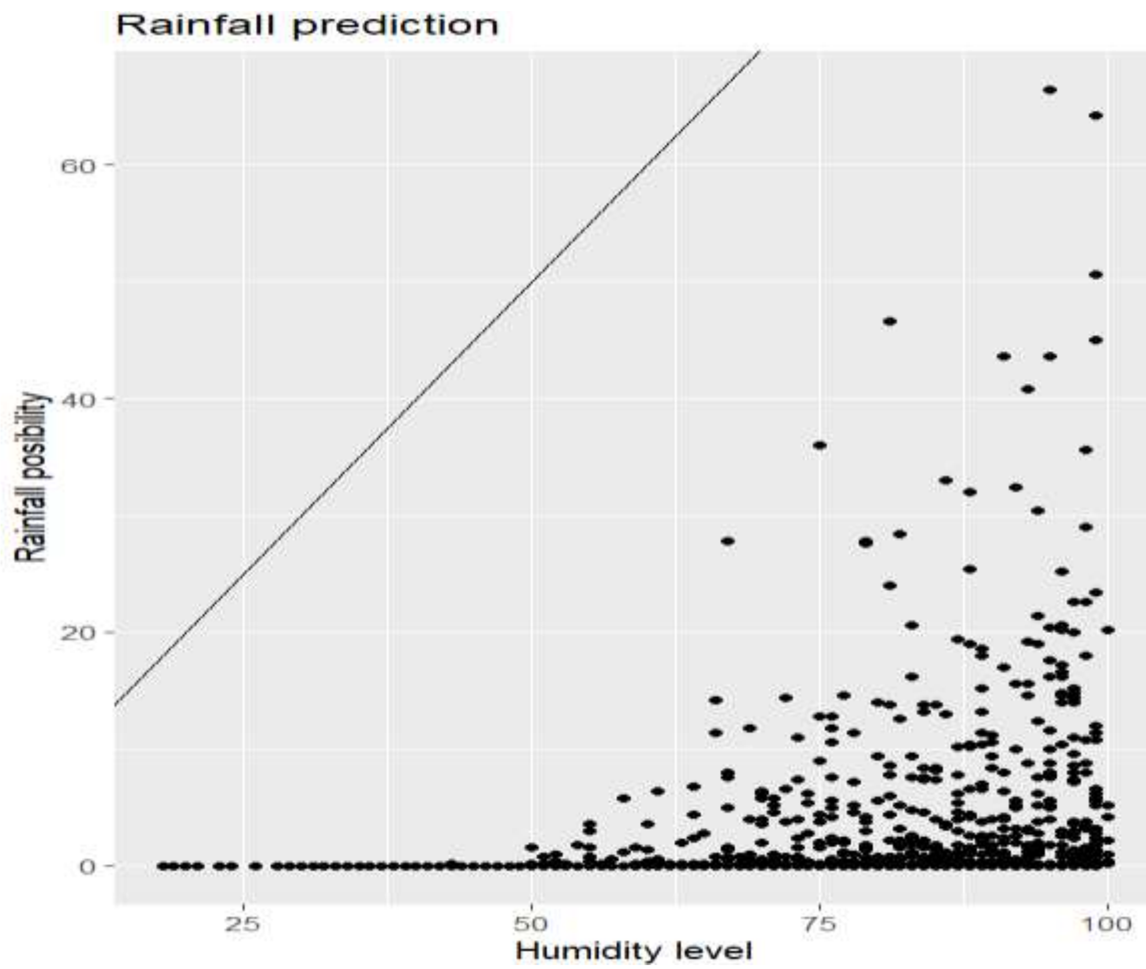
Output

Linear model:

```
> linerModel <- lm(Rainfall ~ Humidity9am, data = Weather_w)
> summary(linerModel)

Call:
lm(formula = Rainfall ~ Humidity9am, data = Weather_w)

Residuals:
   Min     1Q Median     3Q    Max 
-4.909 -2.634 -1.225  0.387 61.925 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept) -5.817520   0.705824  -8.242 4.03e-16 ***
Humidity9am  0.108348   0.009309  11.639  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 5.78 on 1321 degrees of freedom
Multiple R-squared:  0.09301,	Adjusted R-squared:  0.09233 
F-statistic: 135.5 on 1 and 1321 DF,  p-value: < 2.2e-16
```

scatter plot for linear regression:

```
> predictions1 <- predict(linerModel, newdata = weather_w)
> predictions1
          1           2           3           4           5           6           7           8           9 
-0.72518630 -0.50849124 -0.40014371  1.55011183  0.68333159 -0.72518630 -1.15857642  0.35828900  4.36714761 
         10          11          12          13          14          15          16          17          18 
 3.50036737  4.47549514  0.46663653  0.03324641  0.68333159 -0.50849124 -0.50849124  1.55011183 -0.29179618 
         19          20          21          22          23          24          25          26          27 
-0.07510112 -0.50849124 -2.78378937 -1.05022889  0.14159394 -0.94188136  0.24994147  0.03324641 -0.29179618 
         28          29          30          31          32          33          34          35          36 
-1.05022889 -3.86726467  0.35828900 -0.18344865 -0.40014371 -2.56709431 -3.00048443  1.44176430 -0.40014371 
         37          38          39          40          41          42          43          44          45 
-1.05022889 -3.32552702 -2.45874678 -1.26692395 -0.29179618  0.68333159 -2.02535666 -0.94188136 -0.50849124 
         46          47          48          49          50          51          52          53          54 
-1.48361901 -1.26692395  1.00837418 -0.29179618 -1.48361901 -2.67544184  2.20019701  0.24994147  1.11672171 
         55          56          57          58          59          60          61          62          63 
-1.26692395  0.68333159  1.33341677 -1.48361901  0.03324641  1.00837418  1.65845936  1.87515442 -0.07510112 
         64          65          66          67          68          69          70          71          72 
 4.69219020  3.39201984  2.63358713  2.74193466  1.00837418  2.63358713 -0.50849124 -1.48361901 -2.13370419 
         73          74          75          76          77          78          79          80          81 
 2.41689207  1.65845936  0.14159394  3.71706243  0.57498406  2.74193466 -1.26692395  0.90002665  0.35828900 
         82          83          84          85          86          87          88          89          90 
 2.30854454  0.57498406 -0.61683877  0.03324641  1.22506924  1.11672171 -0.50849124  1.98350195 -1.70031407 
         91          92          93          94          95          96          97          98          99 
 0.79167912  0.35828900  1.98350195  1.00837418  1.33341677  0.46663653 -0.40014371  3.93375749  2.95862972 
        100         101         102         103         104         105         106         107         108 
 2.52523960  1.33341677  0.79167912  1.44176430  2.63358713  2.20019701  1.87515442  1.87515442  2.20019701 
        109         110         111         112         113         114         115         116         117 
 1.98350195  2.74193466  2.20019701  1.98350195  3.17532478  4.15045255  1.98350195  2.41689207  0.90002665 
        118         119         120         121         122         123         124         125         126 
 0.46663653  1.00837418 -0.40014371  0.90002665  0.79167912  3.17532478  3.60871490  2.74193466  2.85028219 
        127         128         129         130         131         132         133         134         135 
```



Calculate the mean squared error for linear regression

```
# Calculate the mean squared error for linear regression
mean_sqrd_error <- mean((Weather_w$Rainfall - predictions1)^2)
mean_sqrd_error
1] 33.3584
```

logistic regression:

```
Call:
glm(formula = RainTomorrow ~ MaxTemp, family = "binomial", data = Weather_w)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-0.9762  -0.8091  -0.6720  -0.4165   2.1767

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  0.02187    0.20026   0.109    0.913
MaxTemp     -0.06065    0.01027  -5.905 3.53e-09 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 1463.8  on 1322  degrees of freedom
Residual deviance: 1425.3  on 1321  degrees of freedom
AIC: 1429.3

Number of Fisher Scoring iterations: 4
```

Convert probabilities to class labels:

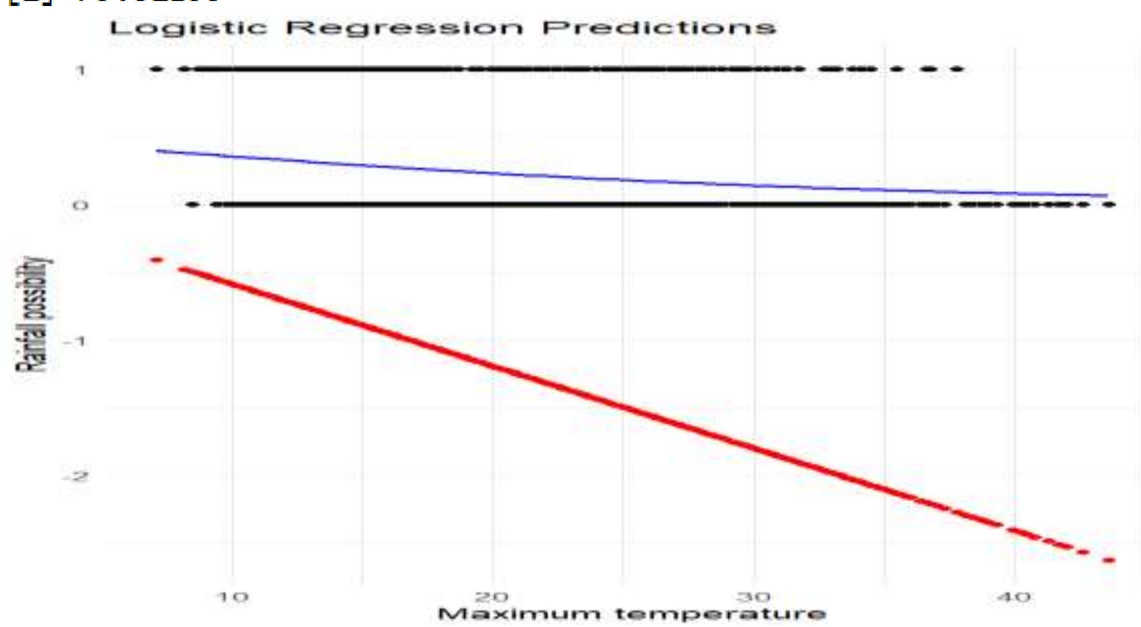
```
-0.7544805 -0.8030023 -0.7848067 -0.6998935 -0.7544805 -0.7726762 -0.6816979 -0.6998935 -0.7120240 -0.8818502
       861        862        863        864        865        866        867        868        869        870
-0.6938283 -0.6938283 -0.8030023 -0.8818502 -0.7180892 -0.8757850 -0.8636545 -0.8090675 -0.5725239 -0.8211980
       871        872        873        874        875        876        877        878        879        880
-0.7120240 -0.7605458 -0.7544805 -0.8030023 -0.6816979 -0.6998935 -0.7969371 -0.8333284 -0.8454589 -0.7241544
       881        882        883        884        885        886        887        888        889        890
-0.7787414 -0.7362849 -0.6210457 -0.6210457 -0.6149804 -0.9546329 -1.0456112 -0.9061111 -0.7180892 -0.6574370
       891        892        893        894        895        896        897        898        899        900
-0.7787414 -1.0334808 -0.6816979 -1.1123287 -0.9728285 -0.8211980 -0.9728285 -0.7787414 -0.7848067 -0.9061111
       901        902        903        904        905        906        907        908        909        910
-0.8211980 -0.8454589 -0.8575893 -1.0213503 -0.6574370 -0.9121763 -1.0880678 -1.2700244 -1.3428071 -1.0516764
       911        912        913        914        915        916        917        918        919        920
-1.1790461 -1.6642639 -1.4883724 -1.3003505 -1.6278725 -1.3913289 -1.1851113 -1.3124810 -1.4095245 -1.5672203
       921        922        923        924        925        926        927        928        929        930
-1.1911765 -1.1790461 -1.2154374 -1.6885248 -1.5732855 -1.1426548 -1.0820025 -1.1365895 -1.2578940 -0.8575893
       931        932        933        934        935        936        937        938        939        940
-0.7848067 -0.9364372 -0.8697198 -0.8090675 -0.8636545 -0.6938283 -1.0274155 -0.8151328 -0.9849590 -0.8818502
       941        942        943        944        945        946        947        948        949        950
-0.8636545 -0.7666110 -0.8090675 -0.8090675 -0.8575893 -0.9485677 -0.9425024 -0.8939807 -0.8333284 -0.7969371
       951        952        953        954        955        956        957        958        959        960
-0.7726762 -0.7666110 -0.9788938 -0.9485677 -1.1608504 -0.8151328 -0.5179369 -0.5785891 -0.8211980 -0.9910242
       961        962        963        964        965        966        967        968        969        970
-0.8333284 -0.6695674 -0.7908719 -0.6635022 -0.5603934 -0.8636545 -0.9364372 -0.8333284 -0.6392413 -0.6453065
       971        972        973        974        975        976        977        978        979        980
-0.9606981 -0.9121763 -0.7908719 -0.9485677 -0.6998935 -1.0759373 -0.8030023 -0.9485677 -0.9243068 -0.9121763
       981        982        983        984        985        986        987        988        989        990
-1.0213503 -0.9182415 -0.9606981 -0.9000459 -0.8575893 -1.0759373 -0.9485677 -1.1608504 -1.1183939 -1.1608504
       991        992        993        994        995        996        997        998        999       1000
-1.1244591 -1.0880678 -0.8272632 -1.0092199 -1.1729809 -1.3064158 -1.1426548 -1.3428071 -1.3610028 -1.7006552
 [ reached getOption("max.print") -- omitted 323 entries ]
> predicted_classes <- ifelse(predictions2 > 0.5, 1, 0
```

Create the confusion matrix:

```
> # Print the confusion matrix
> confusion_matrix
   predicted_classes
       0
  0 1003
  1  320
> 
```

Finding the accuracy of logistic model and regression:

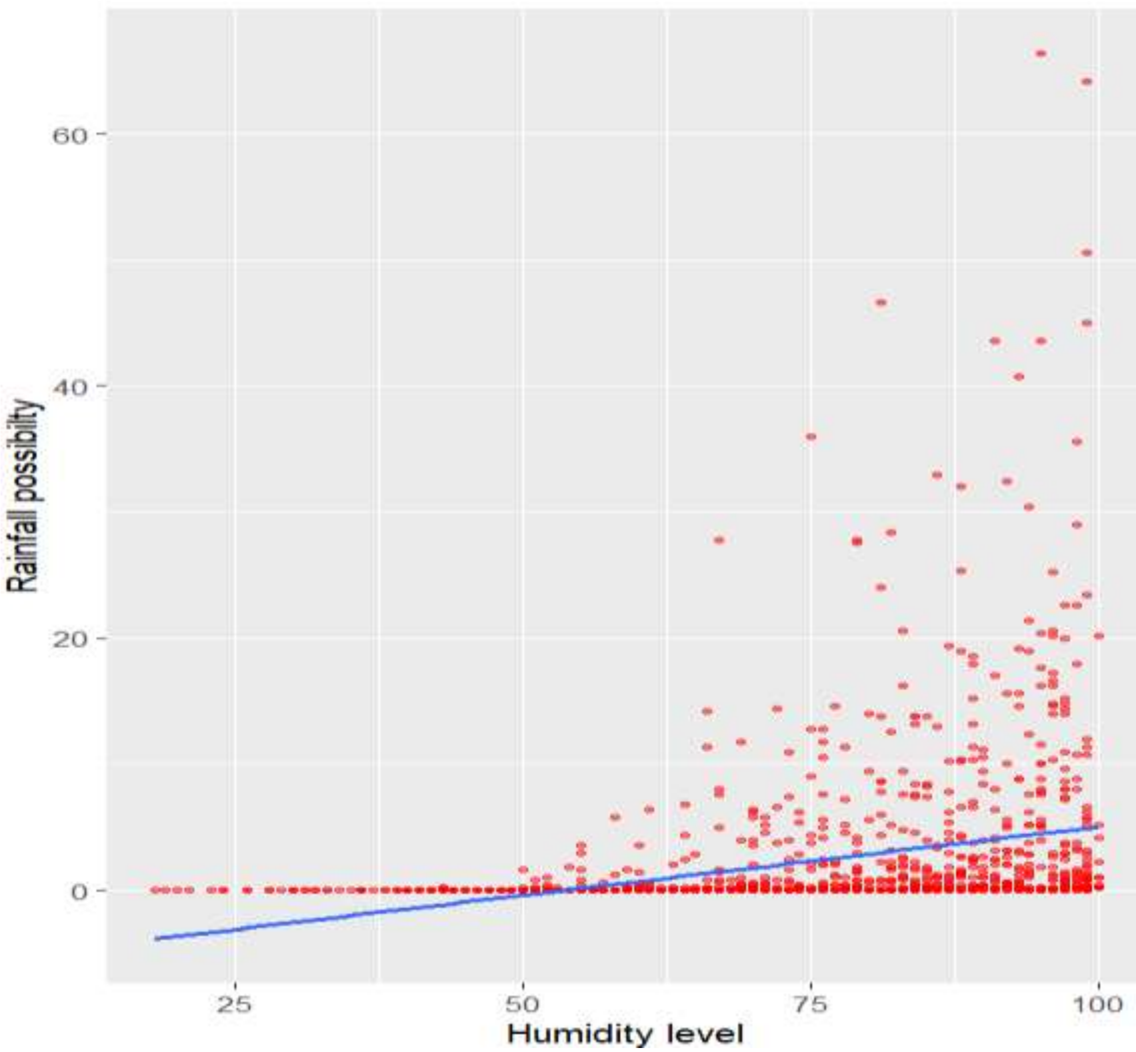
```
> # Finding the accuracy of logistic model
> accuracy <- sum(diag(confusion_matrix)) / sum(confusion_matrix)
> accuracy<- accuracy*100
> accuracy
[1] 75.81255
```



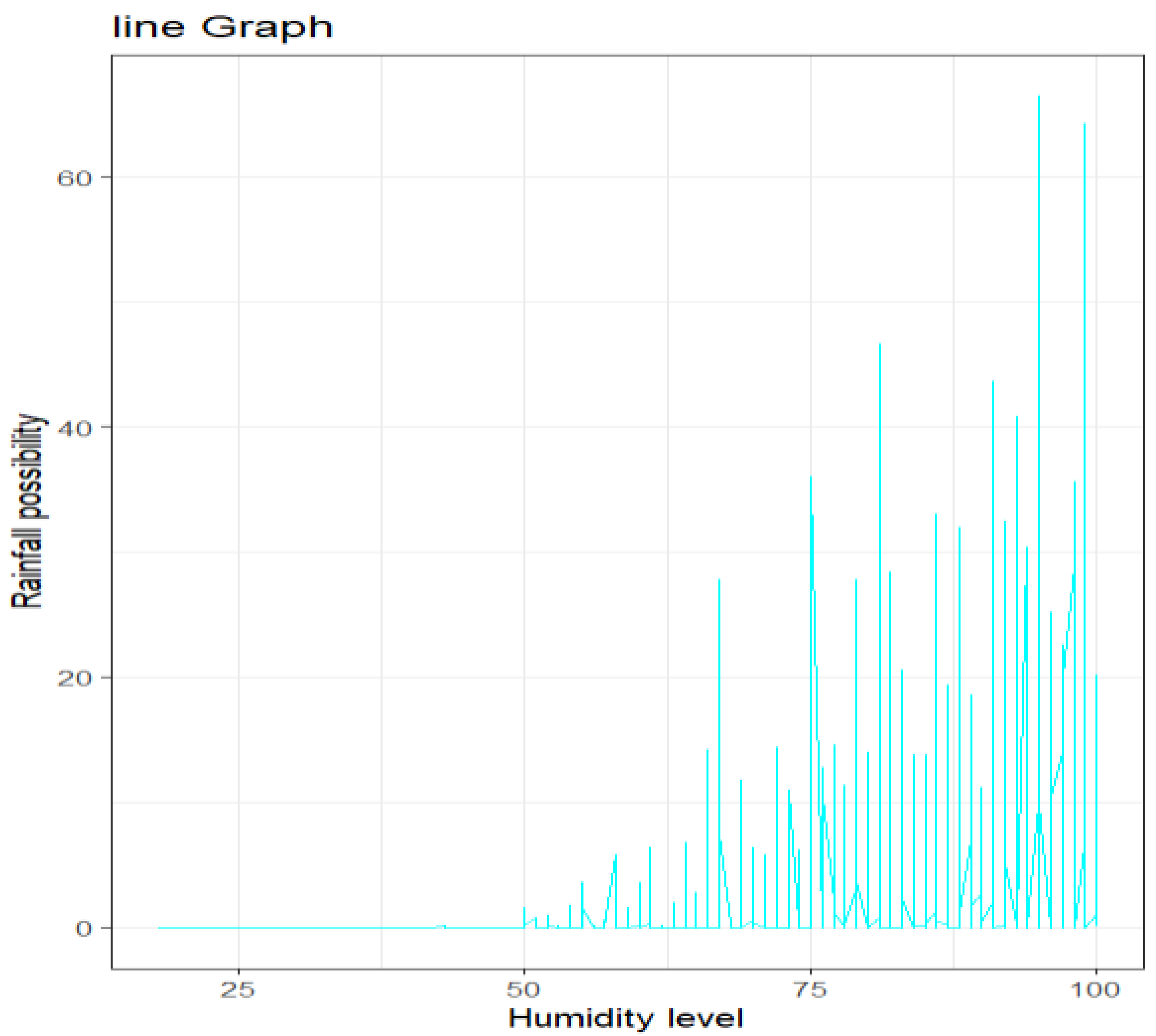
Visualize the predictions in multiple ways:

output:

scatterplot:



Line plot:

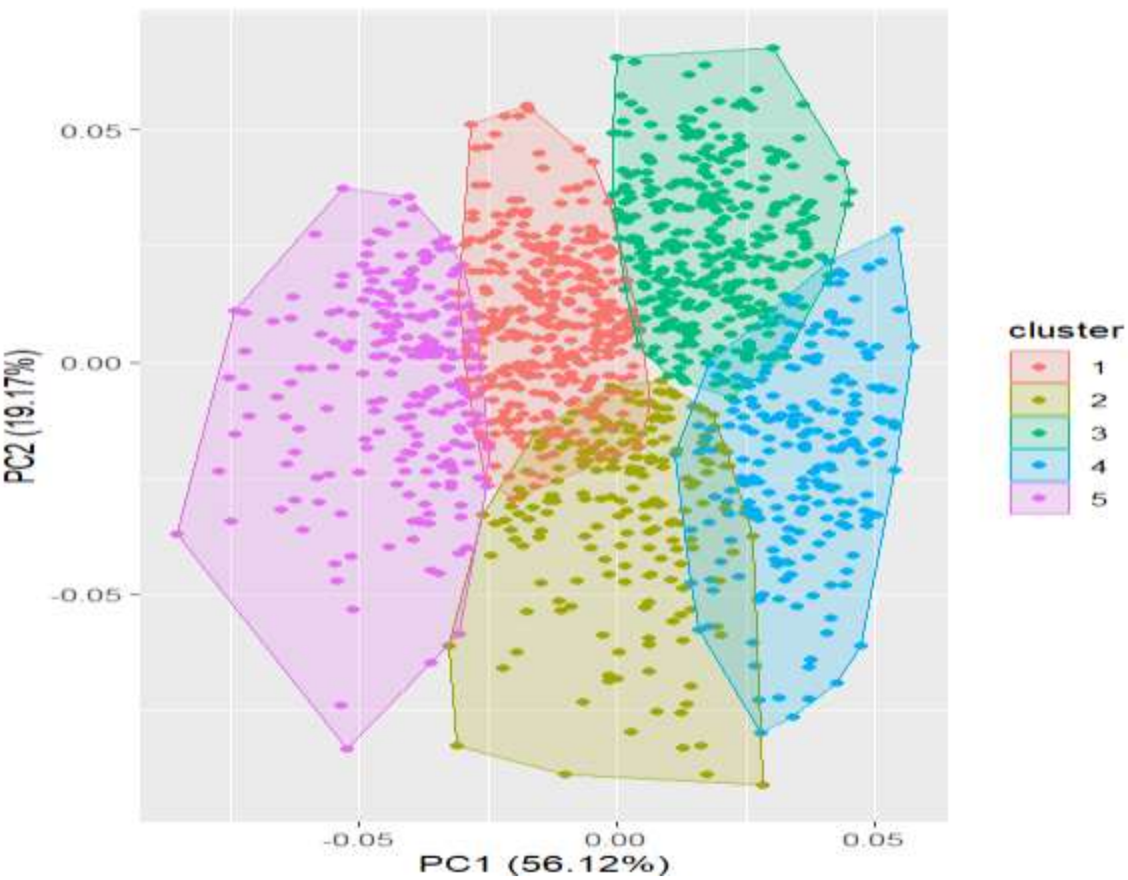

line Graph
Rainfall possibility
0
20
40
60
25
50
75
100
Humidity level

Applying k-means for two clustering pairs:

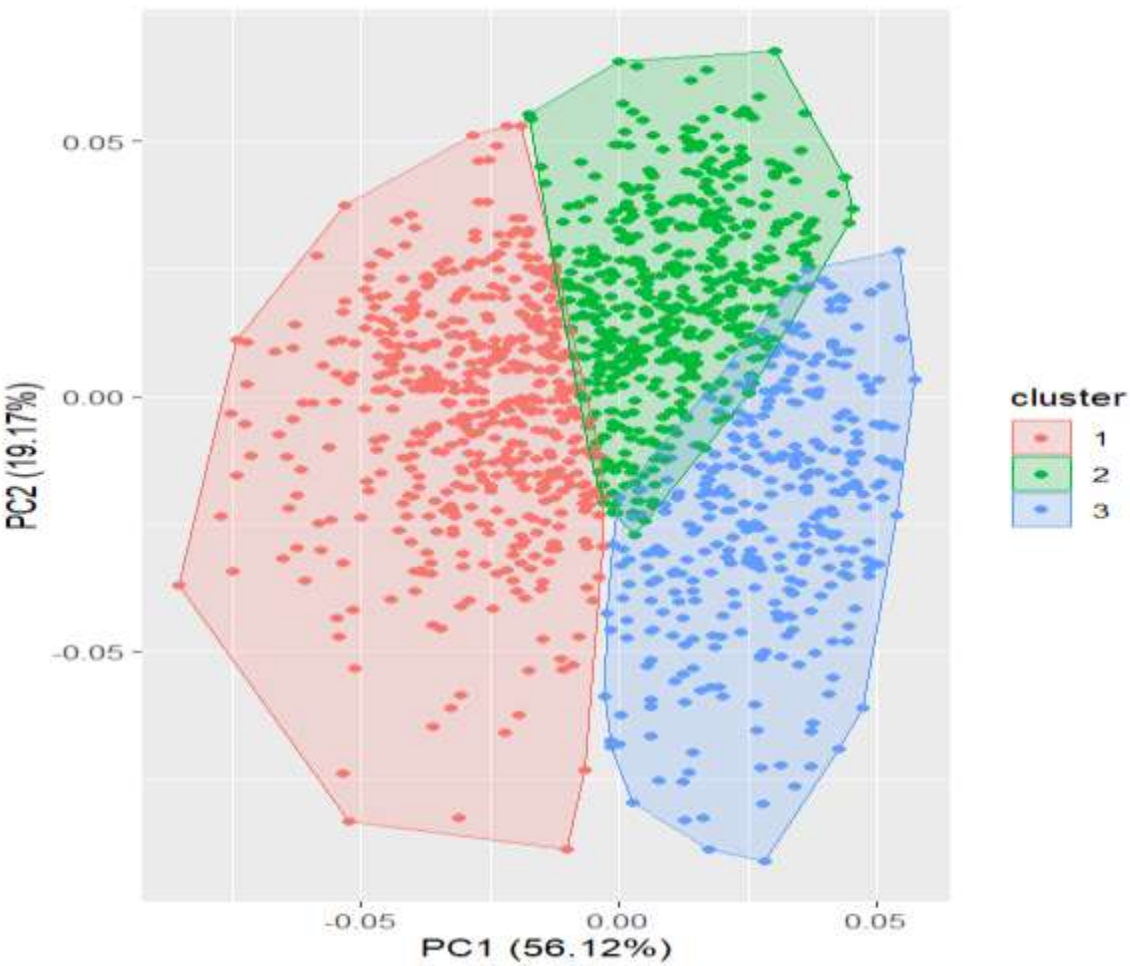
```
Clustering vector:
   [1] 2 2 2 2 2 2 2 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
  [54] 2 2 2 2 2 2 2 2 2 2 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 1 2 2 2 1 1 1 1
 [107] 1 1 2 1 2 1 1 1 2 1 2 2 2 2 2 1 1 1 2 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 [160] 1 2 2 1 1 1 1 1 1 1 1 1 1 1 2 2 1 1 2 1 1 1 2 2 1 1 1 2 2 1 2 1 1 1 2 2 1 2 2 1 2 1 1 2 2 1 2 2 1 1 1 2 2
 [213] 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2
 [266] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 1 1 2 2 2 2 2 2 2 2 1 2 1 2 2 2 1 1 1 2 1
 [319] 2 2 2 1 2 1 2 1 2 1 1 1 1 1 2 2 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 [372] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1
 [425] 1 2 2 2 1 1 2 2 2 2 2 2 2 2 1 2 1 2 2 1 1 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 1 2 2 2 1 1 1 1 1 1
 [478] 2 2 1 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 1 1 1 2 2 2 2 2 2 2 1 2 2 2 2 2 2 1 1 2 2 1 2 2 2 1 1 2
 [531] 2 2 2 1 1 2 2 2 1 1 1 2 2 2 2 2 1 1 1 2 2 1 2 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 [584] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1
 [637] 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 1 1 1 2 2 2 1 1 1 2 1 2 2 2 2 2 2 1 1 2 2 2 2 2 2 2 2 2 2 1 2
 [690] 2 2 2 2 2 2 1 2 2 2 2 2 1 2 2 2 2 2 2 2 2 2 2 2 1 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
 [743] 2 2 2 2 2 2 2 2 2 2 2 2 2 1 2 2 2 2 1 1 2 1 2 1 1 1 2 2 2 2 1 2 1 2 2 2 2 2 1 2 2 1 2 2 2 2 2 2 2 2 2 2 2
 [796] 2 2 2 2 2 1 1 1 1 2 2 2 2 2 2 1 2 2 1 1 1 1 2 2 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 [849] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 1 1 1 2 1 2 1 1 1 1 1 1 1
 [902] 2 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 1 1 2 1 1 1 2 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 [955] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 1 1 2 1 1 2 2 2 2 2 1 1 2 2 1 1 2 2
 [ reached getOption("max.print") -- omitted 323 entries ]

Within cluster sum of squares by cluster:
[1] 633320.1 437009.2
 (between_SS / total_SS =  38.6 %)

Available components:

[1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss" "betweenss"    "size"
[8] "iter"         "ifault"
```

cluster plot for 2 pair:

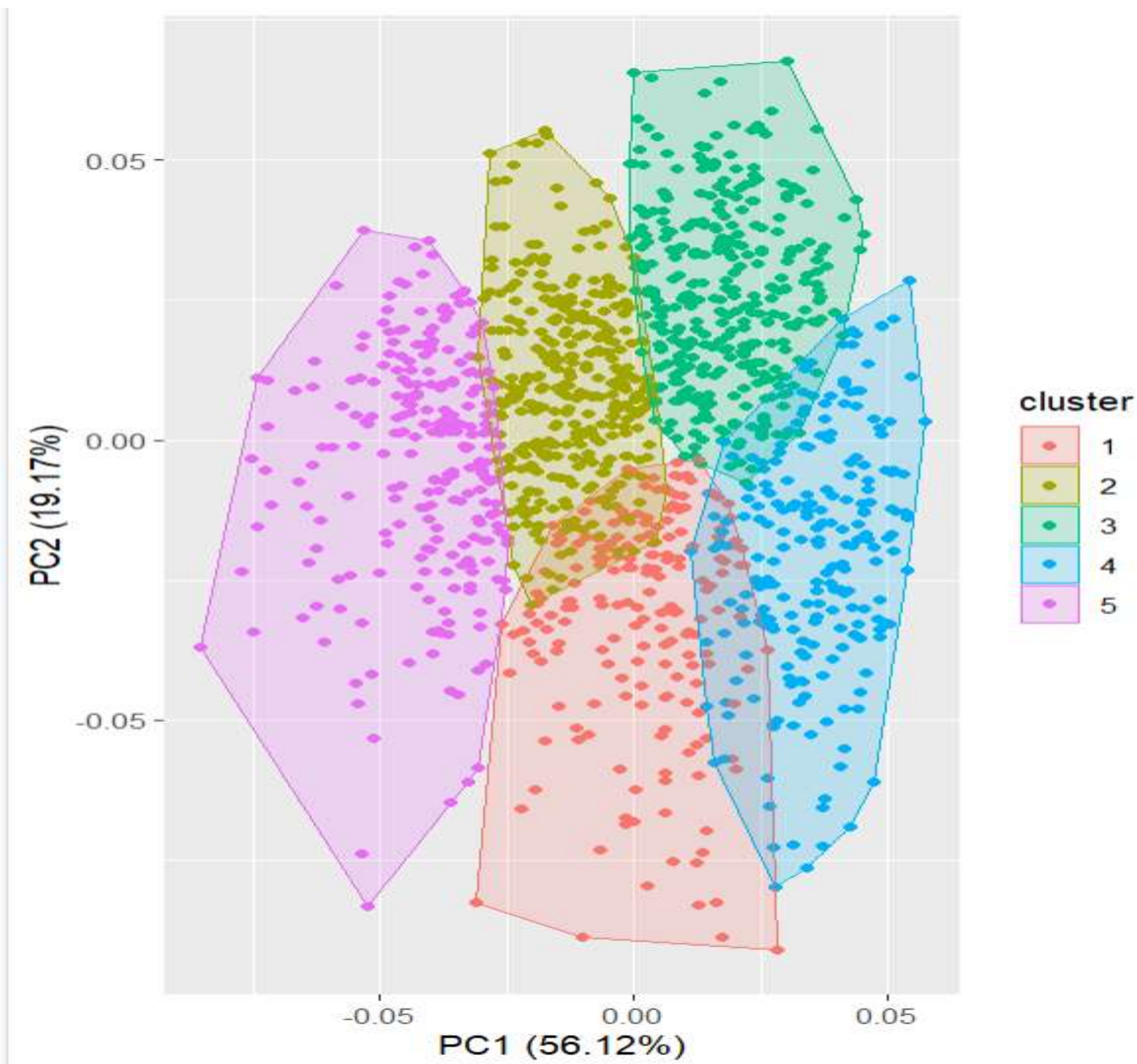
COMSATS University Islamabad
Department of Computer Science
Data Science Fundamentals (DSC293) – BDS-2A
Project (CLO-5)



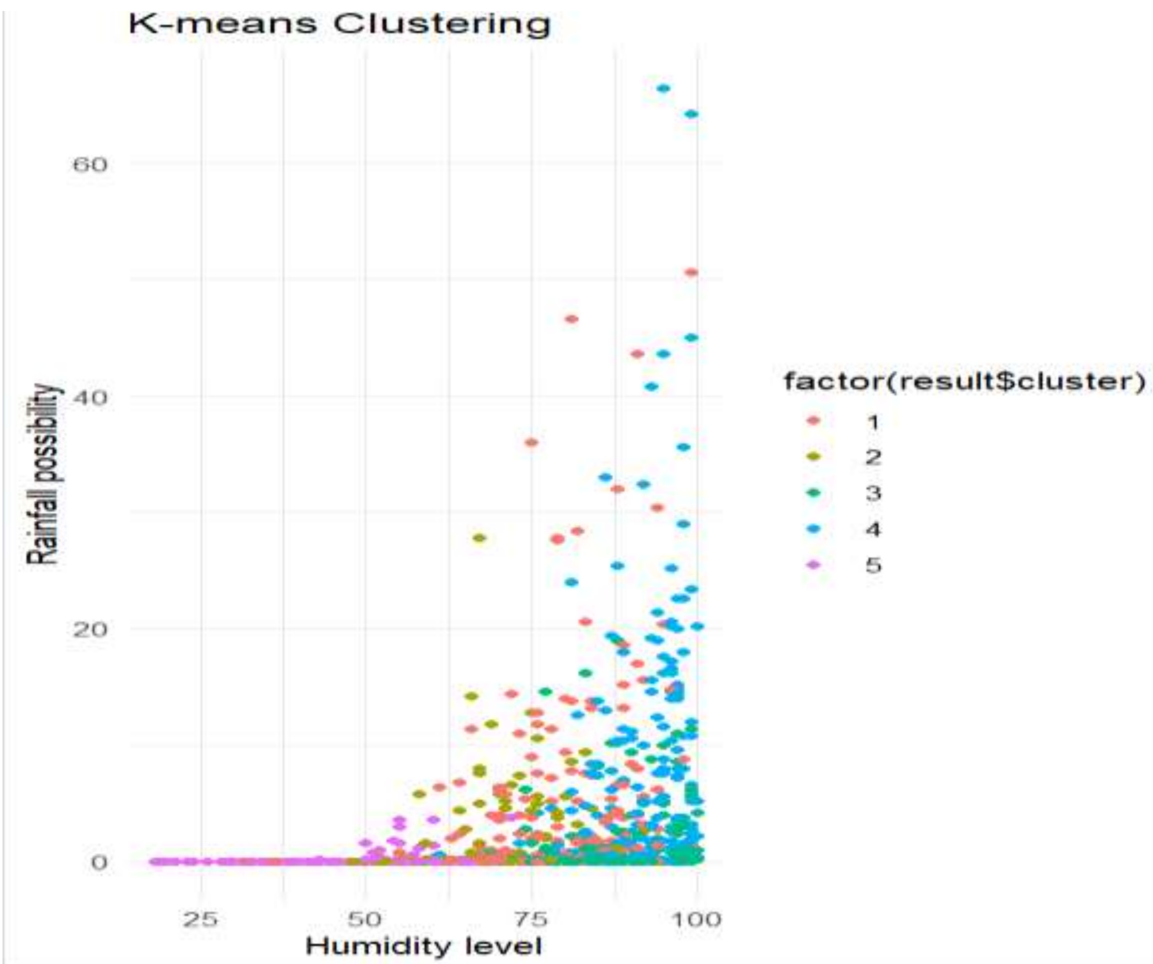
cluster plot for 3 pairs



cluster plot for 4 pairs:



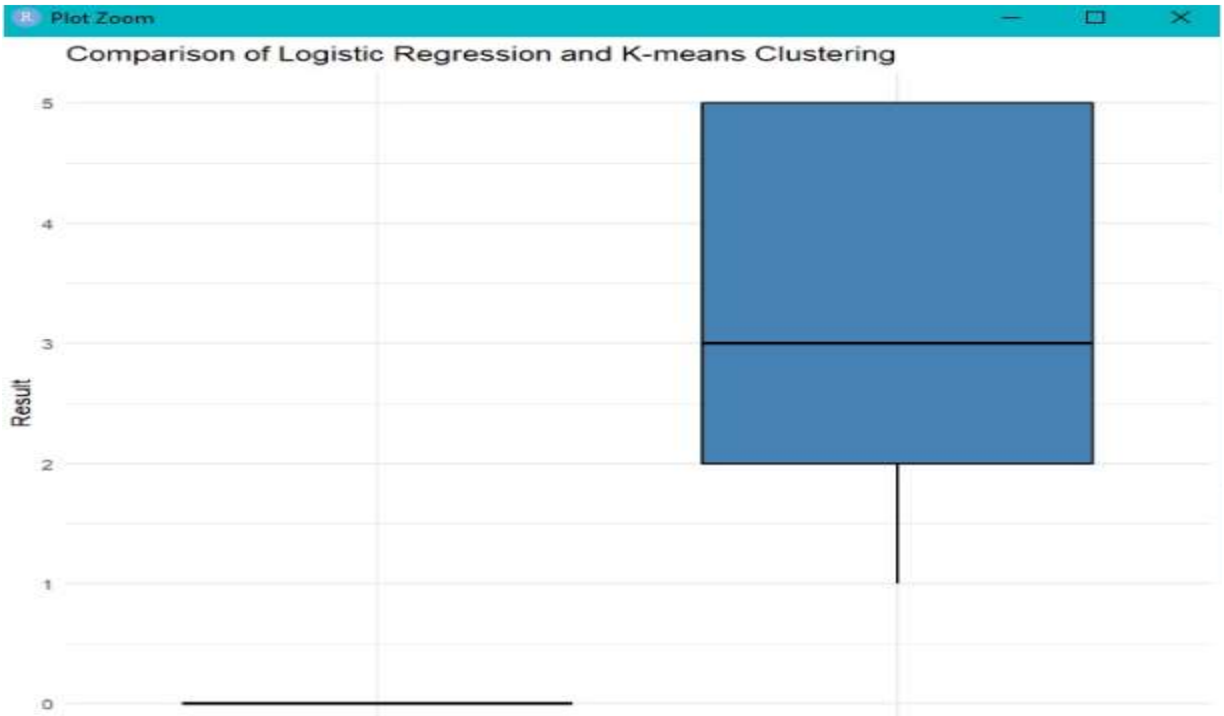
Ggplot of 4 pairs:



Result centres

```
> result$centers
    MinTemp  MaxTemp  Rainfall WindGustSpeed WindSpeed9am WindSpeed3pm Humidity9am Humidity3pm Pressure9am
1  9.418848 18.07644 4.6345550      54.20942    18.853403     25.18325    77.37173    54.57068    1010.898
2  8.775449 22.08892 0.6538922      37.33234    13.814371     16.73952    66.25150    41.79341    1019.227
3  3.559880 14.85240 0.9047904      28.49701     7.026946     13.61677    86.27844    58.02695    1023.952
4  9.059193 15.20045 6.5434978      38.49327    12.775785     15.50224    89.85650    83.88789    1012.841
5 13.324066 30.10456 0.1170124      46.06224    15.224066     20.46473    49.70124    23.70954    1014.549
  Pressure3pm Cloud9am Cloud3pm   Temp9am  Temp3pm
1    1010.372 5.769634 5.696335 12.629843 16.51361
2    1017.110 2.880240 3.985030 14.320958 20.73263
3    1021.893 4.883234 5.209581  8.476048 13.97275
4    1010.645 7.484305 7.349776 11.436771 13.73004
5    1012.054 2.211618 3.124481 19.829461 28.42739
```

Comparison of logistic regression and k-means:



Comparison of dataframe

```
> comparison_df <- data.frame(Logistic_Regression = predicted_classes, KMeans_Cluster = result$cluster)
> # Plotting a box plot to compare the results
> comparison_melted <- melt(comparison_df)
No id variables; using all as measure variables
> boxplot_plot <- ggplot(comparison_melted, aes(x = variable, y = value)) +
+   geom_boxplot(fill = "steelblue", color = "black") +
+   labs(title = "Comparison of Logistic Regression and K-means Clustering",
+        x = "Algorithm", y = "Result") +
+   theme_minimal()
> boxplot_plot
```

Conclusion:

In this predictive analysis, we explored multiple models, including Linear Regression, and K-means Clustering, to understand their performance on the given dataset. After evaluating the models based on various metrics, we found that the K-means Clustering model exhibited the best fit for this dataset. Here are 7 key points summarizing the process and conclusion:

1. Required Packages: The code begins by installing and loading necessary packages, including tidyr, dplyr, ggplot2, randomForest, and others.

2. Exploring the Dataset: The code displays the structure of the dataset, showcasing the variable types and dimensions. It also presents the first few rows, an overview of the dataset, and summary statistics.

3. Data Wrangling: The code filters the dataset to include only data from Bendigo and removes two columns, "Evaporation" and "Sunshine." Missing values are identified and removed from the dataset.

4. Tidying the Dataset: The code renames the column "WindGustDir" to "WindGustDirection" for clarity and consistency.

5. Predictive Modeling: Two predictive algorithms are implemented: linear regression and logistic regression. The linear regression model predicts rainfall based on humidity levels, while the logistic regression model predicts rain tomorrow based on the maximum temperature. Model summaries, predictions, and accuracy are calculated and presented.

6. Visualizing Predictions: Various plots are created to visualize the predictions and analyze the relationship between variables. Scatter plots, line plots, histograms, and bar graphs are generated using the ggplot2 package. K-means clustering is applied to the numerical variables, and cluster plots are produced.

7. Conclusion: The code concludes by comparing the results of logistic regression and k-means clustering using a box plot. This allows for an assessment of the performance and alignment of the two algorithms.

SOURCE CODE

```
# Installing the required packages


install.packages("tidyr")
install.packages("dplyr")
install.packages("stats")
install.packages("DT")
install.packages("tidytext")
install.packages("tidyverse")
install.packages("ggplot2")
install.packages("ggthemes")
install.packages("lubridate")
install.packages("scales")
install.packages("ggthemes")
install.packages("randomForest")
install.packages("mdsr")
install.packages("ggfortify")
install.packages("ROCR")
install.packages("pROC")
install.packages("caret")
install.packages("reshape2")
#Loading the libraries
```

```
library(reshape2)
library(ROCR)
library(pROC)
library(caret)
library(tidyr)
library(dplyr)
library(ggplot2)
library(lubridate)
library(scales)
library(ggthemes)
library(randomForest)
library(mdsr)
library(tidyverse)
library(tidytext)
library(DT)
library(ggfortify)

# Loading the dataset
Weather <- Weather_Training_Data
Weather

# 2: EXPLORING THE DATASET
```

```
# Displaying the dataset
str(Weather)

# to display few rows
head(Weather)
view(Weather)

#  To see overview of the dataset along with the first few values of each variable
glimpse(Weather)

# for the Summary statistics of our dataset
summary(Weather)

# Check the column names
colnames(Weather)


# 3: WRANGLING

#Filtering Rows: Select only the Bendigo's data.
Weather_w <- filter(Weather, Location == "Bendigo")
Weather_w
```

```
# Delete two columns
Weather_w <- subset(Weather_w, select = -c(Evaporation, Sunshine))
Weather_w

# Finding the missing values
missing_values <- sum(is.na(Weather_w))
missing_values

# Removing Missing Values
Weather_w <- na.omit(Weather_w)
View(Weather_w)

# 4: TIDY YOUR DATASET

# Re-nameing the column "WindGustDir" to a more comprehensive name

Weather_w <- rename(Weather_w, "WindGustDirection" = WindGustDir )
Weather_w

# 5: Choose a predictive algorithm to solve your problem
```

```
linerModel <- lm(Rainfall ~ Humidity9am, data = Weather_w)

summary(linerModel)


# scatter plot for linear regression

ggplot(Weather_w, aes(x = Humidity9am , y = Rainfall )) +

  geom_point() +

labs(x = "Humidity level", y = "Rainfall posibility", title = "Rainfall prediction")


ggplot(Weather_w, aes(x = Humidity9am , y = Rainfall )) +

  geom_point() +

  geom_line() +

labs(x = "Humidity level", y = "Rainfall posibility", title = "Rainfall prediction")


ggplot(Weather_w, aes(x = Humidity9am , y = Rainfall )) +

  geom_point() +

  geom_abline() +

  labs(x = "Humidity level", y = "Rainfall posibility", title = "Rainfall prediction")



predictions1 <- predict(linerModel, newdata = Weather_w)

predictions1
```

```
# Calculate the mean squared error for linear regression
mean_sqrd_error <- mean((Weather_w$Rainfall - predictions1)^2)
mean_sqrd_error


#logistic regression
logistic_model <- glm(RainTomorrow ~ MaxTemp, data = Weather_w, family = "binomial")
summary(logistic_model)


predictions2 <- predict(logistic_model, newdata = Weather_w)
predictions2

# Convert probabilities to class labels
predicted_classes <- ifelse(predictions2 > 0.5, 1, 0)

# Create the confusion matrix
confusion_matrix <- table(Weather_w$RainTomorrow, predicted_classes)

# Print the confusion matrix
confusion_matrix

# Finding the accuracy of logistic model
```

```
accuracy <- sum(diag(confusion_matrix)) / sum(confusion_matrix)

accuracy<- accuracy*100

accuracy



# scatter plot for logistic regression

#ggplot(Weather_w, aes(x = MaxTemp, y = RainTomorrow)) +

 # geom_point() +

  #geom_smooth(method = "glm", se = FALSE, color = "blue", method.args = list(family = "binomial")) +

  #geom_point(aes(y = predictions2), color = "red") +

  #labs(x = "Maximum temperature", y = "Rainfall possibility", title = "Logistic Regression Predictions") +

  #theme_minimal()


# 6: Visualize the predictions in multiple ways


#scatterplot

plot_scatter <- ggplot(Weather_w, aes(x= Humidity9am, y = Rainfall))+

  labs(x = "Humidity level", y = "Rainfall possibilty") +
```

```
  geom_point(size= 1, alpha= 0.5, color = "red") +
  geom_smooth(method = "lm", se = FALSE)
plot_scatter


#bar graph
#plot_bar <- ggplot(Weather_w, aes(x = Humidity9am, y = Rainfall)) + #, fill = categories
#geom_bar(stat= "identity", fill = "steelblue", color= "red") +
#theme(legend.position = "none")
#labs(x = "Humidity level", y = "Rainfall possibility", title = "Bar Graph") +
#theme_minimal()
#plot_bar

# Line plot
plot_line <- ggplot(data = Weather_w, aes(x = Humidity9am, y = Rainfall)) +
  geom_line(color = "cyan") +
  labs(x = "Humidity level", y = "Rainfall possibility", title = "line Graph") +
  theme_bw()
plot_line

#predicted values vs actual values
```

```
# Histogram

#plot_histogram <- ggplot(data = Weather_w, aes(x = Humidity9am)) +

#geom_histogram(binwidth = 1) +

#labs(x = "Humidity level", y = "Rainfall possibility", title = "Histogram Graph") +

#theme_bw()

#plot_histogram


# Clustering with k-means

# We need to use numerical values, as K-means algorithm uses only numerical values

numeric_values<- Weather_w[ c("MinTemp", "MaxTemp", "Rainfall", "WindGustSpeed",
"WindSpeed9am", "WindSpeed3pm", "Humidity9am", "Humidity3pm", "Pressure9am", "Pressure3pm",
"Cloud9am", "Cloud3pm", "Temp9am", "Temp3pm")]


# Applying k-means for two clustering pairs

result<-kmeans(numeric_values,2)

result


# cluster plot for 2 pair

autoplot(result,numeric_values,frame=TRUE)


# Applying k-means for three clustering pairs

result<-kmeans(numeric_values,3)

result
```

```
# cluster plot for 3 pairs

autoplot(result,numeric_values,frame=TRUE)

# Applying k-means for four clustering pairs

result<-kmeans(numeric_values,5)

result

# cluster plot for 4 pairs

autoplot(result,numeric_values,frame=TRUE)

ggplot(Weather_w, aes(x = Humidity9am, y = Rainfall, color = factor(result$cluster))) +
  geom_point() +
  labs(x = "Humidity level", y = "Rainfall possibility", title = "K-means Clustering", resolution(12000)) +
  theme_minimal()

result$centers
```

```
view(Weather_w)



#box plot


comparison_df <- data.frame(Logistic_Regression = predicted_classes, KMeans_Cluster = result$cluster)


# Plotting a box plot to compare the results
comparison_melted <- melt(comparison_df)


boxplot_plot <- ggplot(comparison_melted, aes(x = variable, y = value)) +
  geom_boxplot(fill = "steelblue", color = "black") +
  labs(title = "Comparison of Logistic Regression and K-means Clustering",
       x = "Algorithm", y = "Result") +
  theme_minimal()
boxplot_plot
```