Statistical Analysis of Weather Trends using R

Vivek Chandra

2023-12-03

Goal:

My primary objective in creating this R Markdown file is to apply the concepts of inferential statistics and extract meaningful and practical insights from the data. I aim to apply the statistical techniques I have learned to draw valuable conclusions from the Telangana 2018 weather dataset.

Data Exploration

```
# Loading all the libraries used in this project
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.2     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.3     v tibble    3.2.1
## v lubridate 1.9.2     v tidyr     1.3.0
## v purrr     1.0.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(dplyr)
library(ggplot2)
library(tsibble)
```

```
## Warning: package 'tsibble' was built under R version 4.3.2
```

```
##
## Attaching package: 'tsibble'
##
## The following object is masked from 'package:lubridate':
##
##     interval
##
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, union
```

```r
library(corrplot)
```

```
## Warning: package 'corrplot' was built under R version 4.3.2
```

```
## corrplot 0.92 loaded
```

```r
library(hexbin)
```

```
## Warning: package 'hexbin' was built under R version 4.3.2
```

```r
library(viridisLite)
```

```
## Warning: package 'viridisLite' was built under R version 4.3.2
```

```r
library(MASS)
```

```
## Warning: package 'MASS' was built under R version 4.3.2
```

```
##
## Attaching package: 'MASS'
##
## The following object is masked from 'package:dplyr':
##
##     select
```

```r
# reading the csv file
my_data=read.csv("C:/Users/user/Documents/IUPUI -SEM 1/Statistics/Assignments/Telangana_2018_complete_w
```

```r
str(my_data)
```

```
## 'data.frame':    230384 obs. of  10 variables:
##  $ row_id      : int  0 1 2 3 4 5 6 7 8 9 ...
##  $ District    : chr  "Medchal-Malkajgiri" "Medchal-Malkajgiri" "Medchal-Malkajgiri" "Medchal-Malkajg
##  $ Mandal      : chr  "Uppal" "Uppal" "Uppal" "Uppal" ...
##  $ Location    : chr  "Moulali " "Moulali " "Moulali " "Moulali " ...
##  $ Date        : chr  "01-01-2018" "02-01-2018" "03-01-2018" "04-01-2018" ...
##  $ temp_min    : num  12.1 11.6 13 9.7 8.8 8.7 9.2 9.6 10.5 10.8 ...
##  $ temp_max    : num  32.6 32.6 33 31.7 31 ...
##  $ humidity_min: num  23.8 23.2 31.5 27.4 28.6 22 21.4 23.8 26.5 35.7 ...
##  $ humidity_max: num  100 100 100 100 100 100 100 100 100 100 ...
##  $ wind_speed  : num  6.6 4.7 6.3 5.2 7.1 4.5 3.7 7.1 6.1 6.4 ...
```

```r
# changing the data type of "Date" from chr to Date
my_data$Date <- as.Date(my_data$Date, format = "%d-%m-%Y")
# removing row_id column
my_data <- my_data[, -1]
```

```r
str(my_data)
```

```
## 'data.frame':    230384 obs. of  9 variables:
##  $ District    : chr  "Medchal-Malkajgiri" "Medchal-Malkajgiri" "Medchal-Malkajgiri" "Medchal-Malkaj
##  $ Mandal      : chr  "Uppal" "Uppal" "Uppal" "Uppal" ...
##  $ Location    : chr  "Moulali " "Moulali " "Moulali " "Moulali " ...
##  $ Date        : Date, format: "2018-01-01" "2018-01-02" ...
##  $ temp_min    : num  12.1 11.6 13 9.7 8.8 8.7 9.2 9.6 10.5 10.8 ...
##  $ temp_max    : num  32.6 32.6 33 31.7 31 ...
##  $ humidity_min: num  23.8 23.2 31.5 27.4 28.6 22 21.4 23.8 26.5 35.7 ...
##  $ humidity_max: num  100 100 100 100 100 100 100 100 100 100 ...
##  $ wind_speed  : num  6.6 4.7 6.3 5.2 7.1 4.5 3.7 7.1 6.1 6.4 ...
```

```r
# identifying the unique Districts from the "District" column
unique_districts= unique(my_data$District)
print(unique_districts)
```

```
##  [1] "Medchal-Malkajgiri"       "Rangareddy"
##  [3] "Hyderabad"                "Vikarabad"
##  [5] "Mahabubnagar"             "Wanaparthy"
##  [7] "Jogulamba-Gadwal"         "Nagarkurnool"
##  [9] "Yadadri-Bhongir"          "Nalgonda"
## [11] "Suryapet"                 "Siddipet"
## [13] "Jangaon"                  "Mancherial"
## [15] "Warangal (U)"             "Nirmal"
## [17] "Jayashankar-Bhupalpally"  "Sangareddy"
## [19] "Kamareddy"                "Adilabad"
## [21] "Warangal (R)"             "Mahabubabad"
## [23] "Nizamabad"                "Medak"
## [25] "Bhadradri-Kothagudem"     "Jagtial"
## [27] "Khammam"                  "Peddapalli"
## [29] "Karimnagar"               "Rajanna-Siricilla"
## [31] "Kumuram Bheem - Asifabad"
```

```r
# identifying the unique Mandals from the "Mandal" column
unique_mandals= unique(my_data$Mandal)
head(unique_mandals,60)
```

```
##  [1] "Uppal"               "Abdullapurmet"       "Shaikpet"
##  [4] "Balapur"             "Saroornagar"         "Kapra"
##  [7] "Musheerabad"         "Serilingampally"     "Chevella"
## [10] "Pargi"               "Tandur"              "Kulkacherla"
## [13] "Kodangal"            "Maddur"              "Mahabubnagar (U)"
## [16] "Marikal"             "Makthal"             "Amarchintha"
## [19] "Gadwal"              "Jadcherla"           "Undavelli"
## [22] "Midjil"              "Wanaparthy"          "Kalwakurthy"
## [25] "Nagarkurnool"        "Vangoor"             "Achampet"
## [28] "Kollapur"            "Balanagar"           "Farooqnagar"
## [31] "Kothur"              "Bibinagar"           "Bhongir"
## [34] "Alair"               "Mothkur"             "Saligouraram"
## [37] "Thungaturthi"        "Suryapet"            "Narketpalle"
## [40] "Choutuppal"          "Cheriyal"            "Jangaon"
```

```
## [43] "Raghunathpalle"       "Mancherial"           "Dharmasagar"
## [46] "Luxettipet"           "Hanamkonda"           "Chinthapalle"
## [49] "Marriguda"            "Khanapur"             "Nirmal Rural"
## [52] "Kondamallapally"      "Ghanpur (Mulug)"      "Hasanparthy"
## [55] "Pedda Adesherlapalle" "Patancheruvu"         "Kamareddy"
## [58] "Mulug"                "Kandi"                "Adilabad (Urban)"
```

Estimates of Location Estimating the location is the first basic step in understanding where the majority of the data is centered. It provides us with a central reference point, helping us grasp the "typical" value in our dataset.

```r
# Calculating mean, median, quantiles, min, and max for each numerical columns

location_estimates <- sapply(my_data[, c("temp_min", "temp_max", "humidity_min", "humidity_max", "wind_s
  c(
    Mean = mean(x, na.rm = TRUE),
    Median = median(x, na.rm = TRUE),
    Quantile = quantile(x, 0.25, na.rm = TRUE),
    Quantile = quantile(x, 0.75, na.rm = TRUE),
    Min = min(x, na.rm = TRUE),
    Max = max(x, na.rm = TRUE)
  )
})

location_estimates_df <- as.data.frame(location_estimates)

# Printing the results
print(location_estimates_df)
```

```
##              temp_min temp_max humidity_min humidity_max wind_speed
## Mean         22.50855 34.75674       41.297     81.07755   10.81777
## Median       23.50000 34.60000       36.900     83.10000   10.20000
## Quantile.25% 20.20000 31.80000       24.600     70.90000    6.80000
## Quantile.75% 25.40000 37.70000       55.500     94.70000   14.30000
## Min           5.00000 22.00000        0.100      4.00000    0.10000
## Max          34.60000 45.40000       99.900    100.00000   69.00000
```

Estimates of Variability Estimating the Variability is the next step to understand how data values are spread out or clustered

```r
# Calculating variance,standard deviation,IQR,and Range for each numerical columns

variability_estimates <- sapply(my_data[, c("temp_min", "temp_max", "humidity_min", "humidity_max", "wi
  c(
    Variance = var(x, na.rm = TRUE),
    Standard_Deviation = sd(x, na.rm = TRUE),
    IQR = IQR(x, na.rm = TRUE),
    Range = diff(range(x, na.rm = TRUE))
  )
})
```

```r
variability_estimates_df <- as.data.frame(variability_estimates)

# Printing the results
print(variability_estimates_df)
```

```
##                    temp_min  temp_max humidity_min humidity_max wind_speed
## Variance           19.327300 17.237884    440.09143    232.01556  38.389194
## Standard_Deviation  4.396283  4.151853     20.97836     15.23206   6.195901
## IQR                 5.200000  5.900000     30.90000     23.80000   7.500000
## Range              29.600000 23.400000     99.80000     96.00000  68.900000
```

Data Visualization

In this section, I would like to explore the dataset through visualization . Visualizing is a robust method for extracting insights and conveying information in a more accessible and comprehensible format.

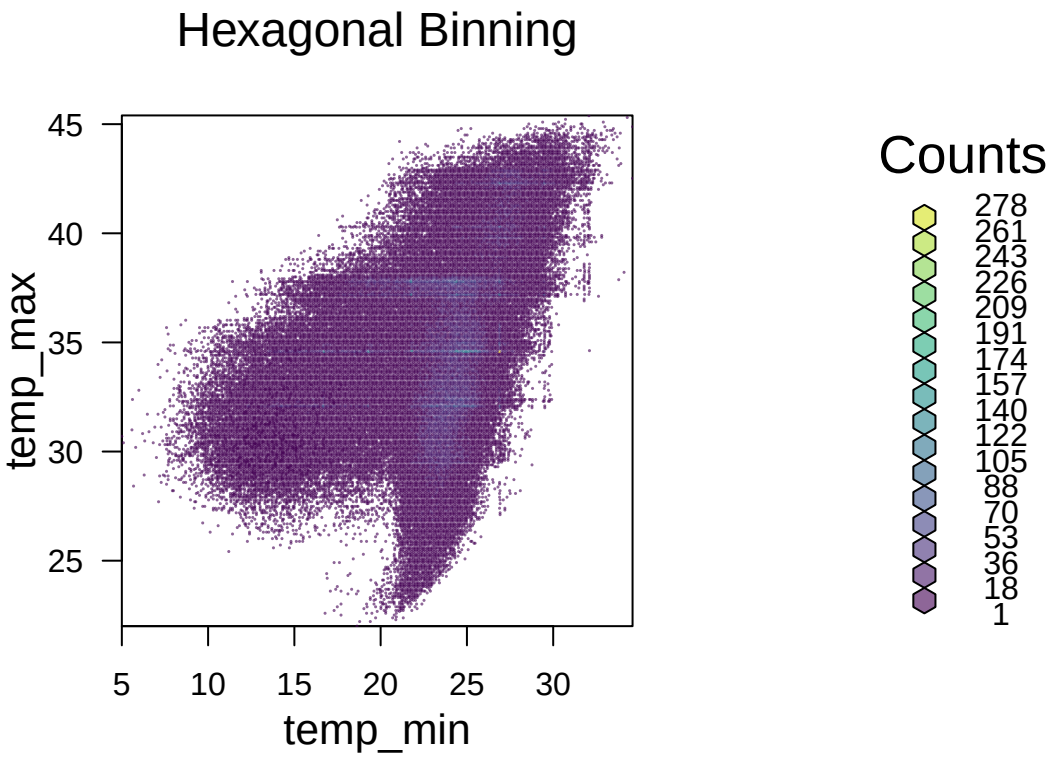
Exploring Numeric Versus Numeric Data

Hexagonal Binning In this analysis, I'm utilizing hexagonal binning to plot the relationship between temp_min and temp_max. Hexagonal binning is considered as the optimal plot for illustrating numeric versus numeric data, particularly when dealing with relatively large datasets .Its ability to mitigate over-plotting and efficiently represent data makes it a valuable tool in uncovering meaningful patterns and trends

```r
# Calculating the number of bins using the rule of thumb
num_bins <- round(sqrt(nrow(my_data)))
cat("The number of hexanoganl bins are :",num_bins)
```

```
## The number of hexanoganl bins are : 480
```

```r
# Creating a hexbin plot with the calculated number of bins
hb <- hexbin(my_data$temp_min, my_data$temp_max, xbins = num_bins)

# Plotting the hexbin
plot(hb, main = "Hexagonal Binning", xlab = "temp_min", ylab = "temp_max", colramp = function(n) viridi
```



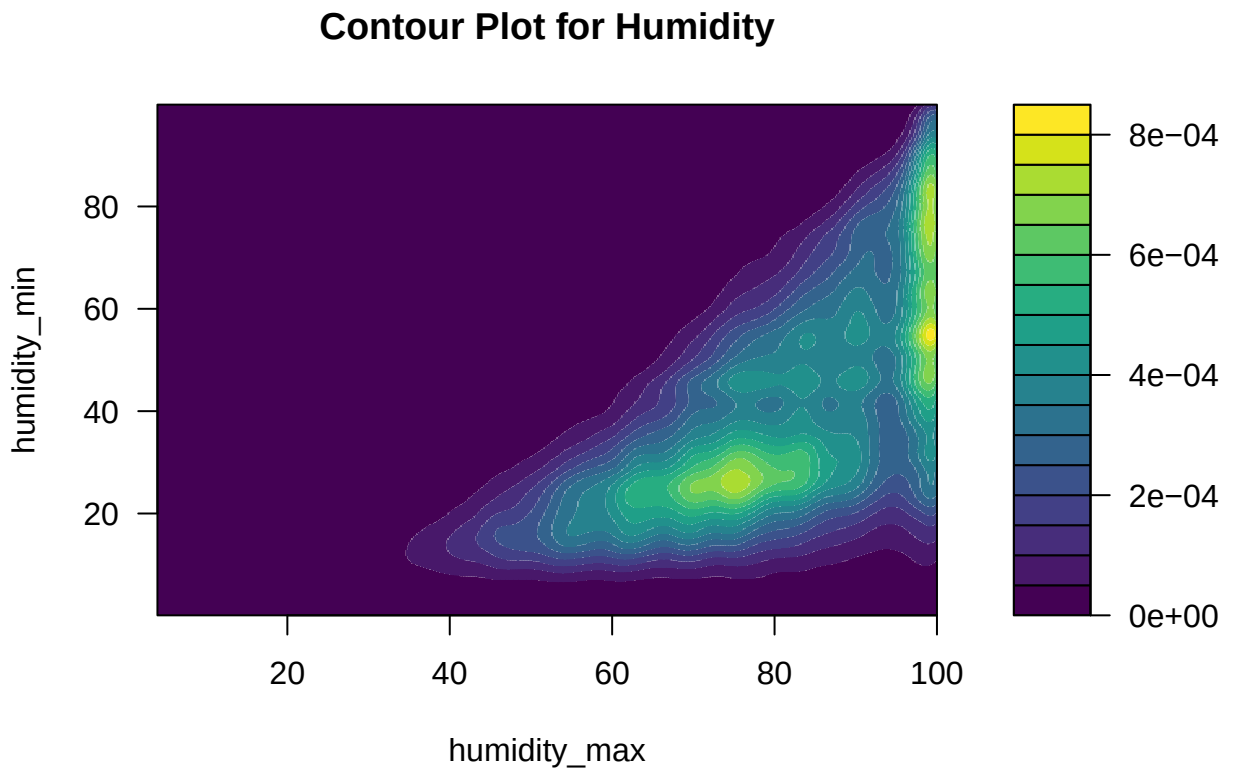
In the hexbin plot, the two dimensions on the x and y axes represent the two variables being compared, and the color of the hexagons indicates the count of observations within each hexagonal bin.

From above plot, we can see the following:

1. **Data Spread**: The data points are spread across a wide range of minimum and maximum temperatures, suggesting variability in the daily temperatures over the observed period.
2. **Density**: The color coding shows the density of observations. The lighter color (yellow-green) indicates higher counts, meaning that there are more days with temperatures that correspond to those specific temp_min and temp_max values. Conversely, the darker colors represent fewer observations.
3. **Temperature Correlation**: There is a clear positive correlation between temp_min and temp_max. As the minimum temperature increases, the maximum temperature also tends to increase, which is expected in daily weather patterns.
4. **Temperature Range and Common Values**: The most common temperature ranges, indicated by the lighter hexagons, seem to be when the temp_min is around 20 degrees and the temp_max is around 35 degrees. These might represent the most frequent temperature conditions for the region.

Contour Plot I am employing Contour plot to depict the relationship between humidity_max and humidity_min. A contour plot is also an effective choice for visualizing numeric versus numeric relationships, especially when working with large datasets.

```r
# Contour plot between humidity_max and humidity_min
data_density <- kde2d(my_data$humidity_max, my_data$humidity_min, n = 480)
filled.contour(data_density,
               main = "Contour Plot for Humidity",
               xlab = "humidity_max",
               ylab = "humidity_min",
               color.palette = viridisLite::viridis)
```



From above plot, we can observe the following:

1. **Correlation**: There is a positive correlation between humidity_min and humidity_max, meaning that days with higher minimum humidity levels also tend to have higher maximum humidity levels.
2. **Density of Observations**: The density of observations is represented by the color gradient in the contour plot. Areas with brighter colors (yellow and green) indicate higher densities of observations, while darker areas (purple and blue) indicate lower densities.
3. **Common Humidity Ranges**: The brightest area, which indicates the highest density of observations, seems to be where both humidity_min and humidity_max are relatively high, suggesting that it is common for both the minimum and maximum humidity levels to be high at the same time.
4. **Extreme Conditions**: The lower left corner of the plot, where both humidity_min and humidity_max are low, has very few observations, indicating that it is rare to have low humidity for both the minimum and maximum measurements on the same day

Exploring Categorical Versus Categorical Data

Contingency Table In this analysis, I am using a contingency table to explain the relationship between two categorical columns i.e., District and Mandal.

```
# Creating a contingency table
contingency_table <- table(my_data$District, my_data$Mandal)

view(contingency_table)

# Writing the contingency table to a CSV file
write.csv(contingency_table, file = "contingency_table.csv")
```

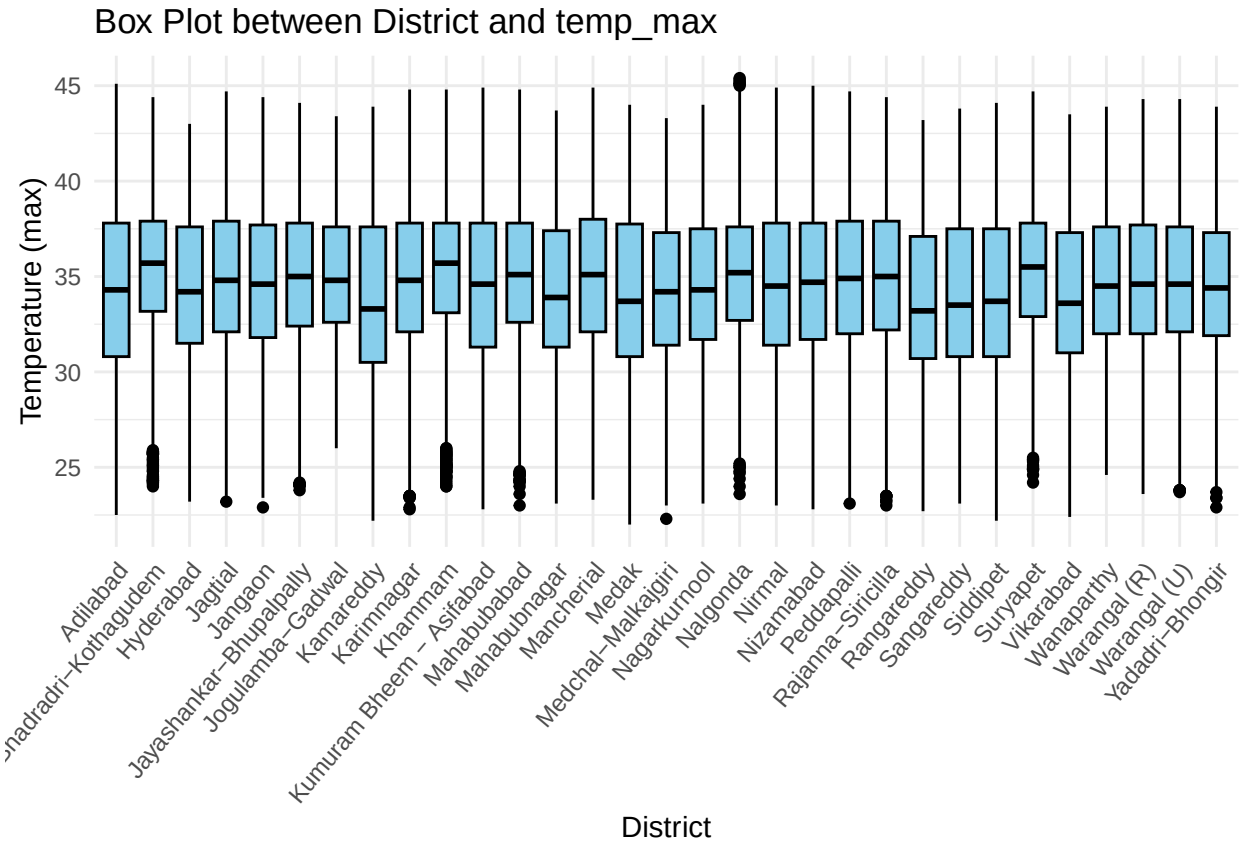
From the above Contingency table, we can infer the following

1. **Frequency Distribution:** The cells of the table display the frequency of occurrences for each combination of district and mandal, offering a comprehensive overview of the dataset's categorical composition.
2. **Association Strength:** By analyzing the values in the table, we can identify whether there are strong associations between certain districts and mandals.
3. **Data Summarization:** The contingency table serves as a concise summary of the categorical data, facilitating a clearer understanding of the distribution of districts and mandals.

Exploring Categorical Versus Numerical Data

Box Plot In this analysis, I am using box plot to illustrate the relationship between District and temp_max columns.Box plots provides a visual comparison of the distribution of maximum temperatures across different districts.

```
# Creating a box plot
ggplot(my_data, aes(x = District, y = temp_max)) +
  geom_boxplot(fill = "skyblue", color = "black", width = 0.7) +
  labs(title = "Box Plot between District and temp_max",
       x = "District",
       y = "Temperature (max)") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 50, hjust = 1))
```



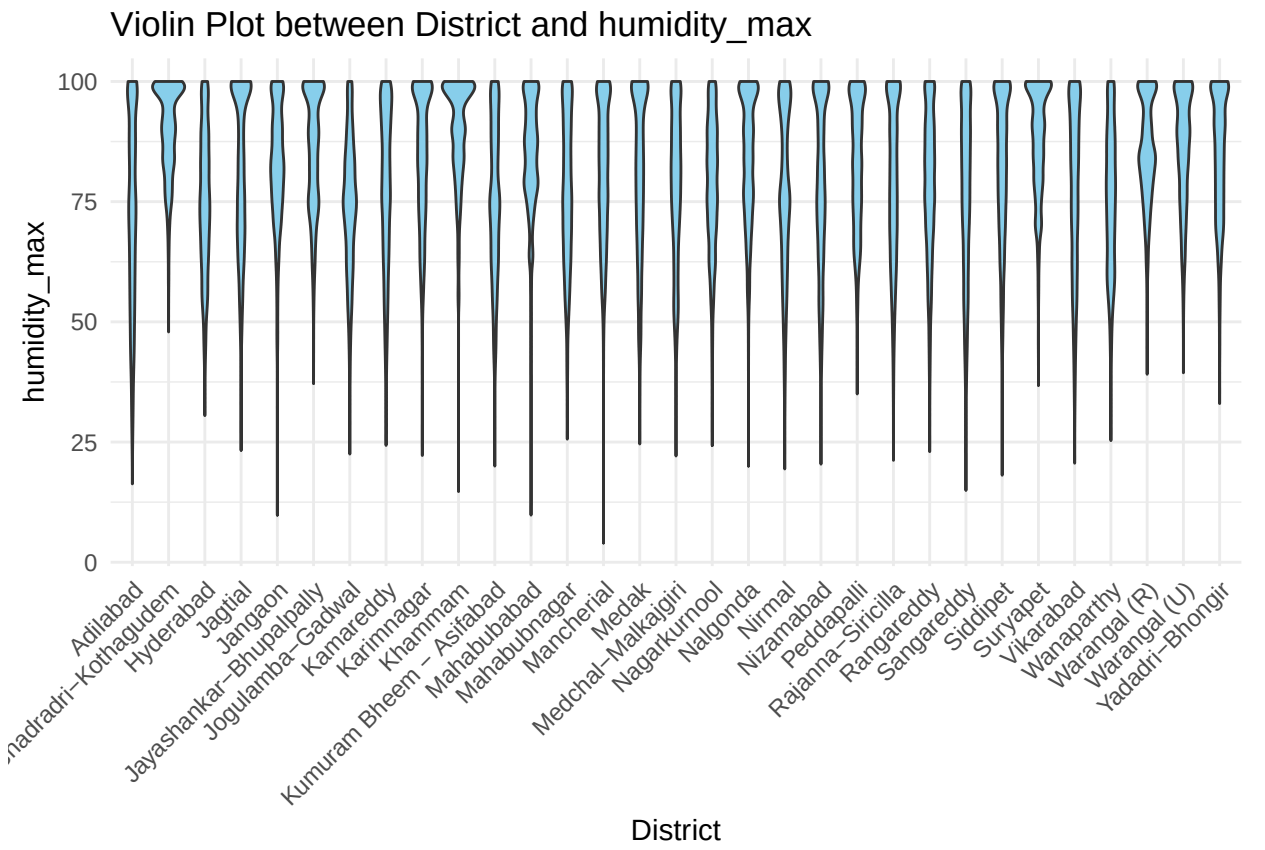
From the above Box Plot , we can see the following

1. **Central Line in Boxes**: The line that divides each box into two parts represents the median maximum temperature for each district.
2. **Box Length**: The length of the box, which represents the interquartile range (IQR), shows the middle 50% of the data. A longer box indicates greater variability in maximum temperatures within that district.
3. **Outliers**: Outlier points are shown as individual dots outside the whiskers. These represent districts with maximum temperature values that are unusually high or low compared to the rest of the data for that district.
4. **Uniformity**: The relatively consistent box sizes and median lines across the districts suggest that the climate in terms of maximum temperature is fairly uniform across these areas, with some exceptions as indicated by outliers.

Violin Plot I am employing Violin plot to depict the relationship between District and humidity_max. A Violin Plot is also a good choice for visualizing categorical versus numeric relationships.

```
# Creating a violin plot
ggplot(my_data, aes(x = District, y = humidity_max)) +
  geom_violin(fill = "skyBlue") +
  labs(title = "Violin Plot between District and humidity_max",
       x = "District",
       y = "humidity_max") +
```

```
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



From the above Violin Plot, we can infer the following

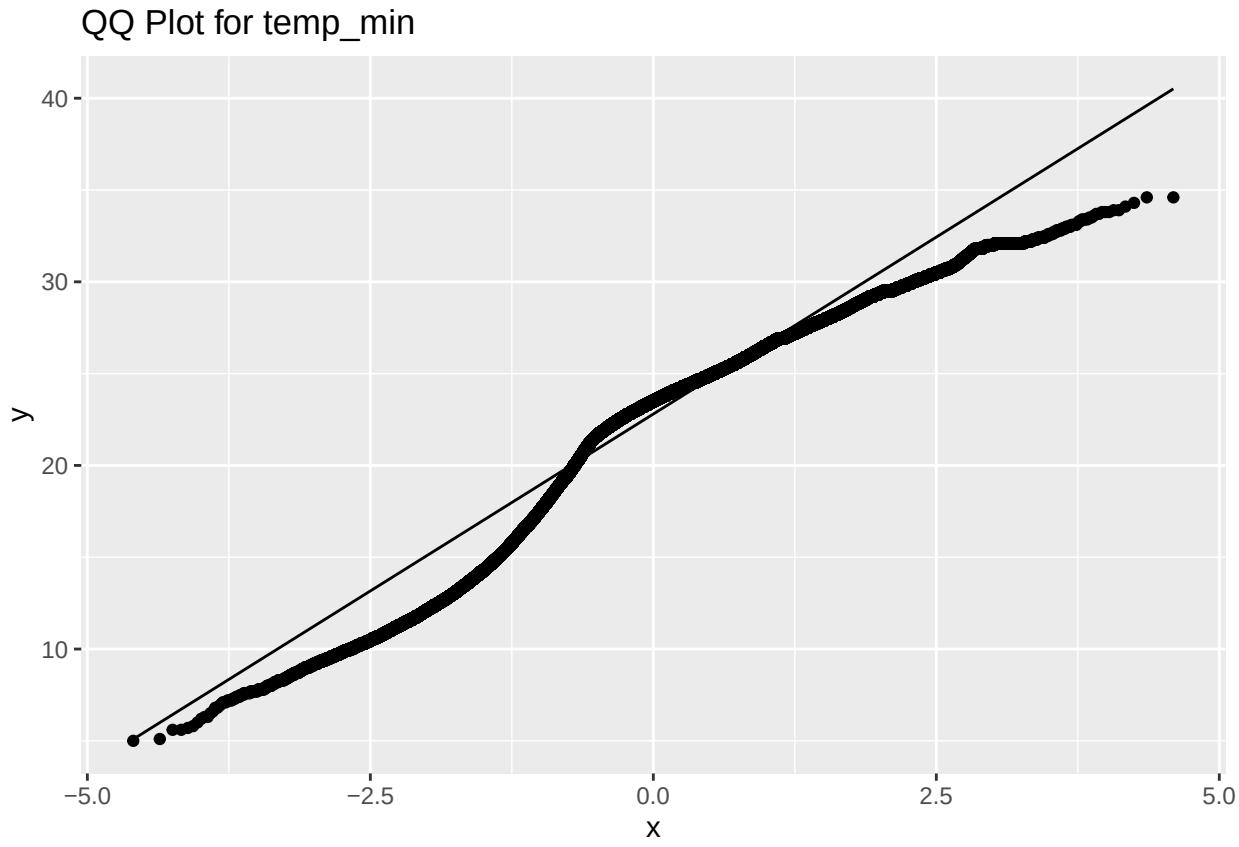
1. **Shape of Violins**: Each 'violin' represents the distribution of maximum humidity for a district. The width of the violin at different levels indicates the density of data points at those humidity values. Wider sections mean more days had humidity values near that level.
2. **Box Plot Inside the Violins**: Inside each violin, there's typically a miniature box plot. The thick bar in the center of the box indicates the interquartile range, and the thin lines (whiskers) extending from the box show the range of the data excluding outliers.
3. **Density of Data Points**: The 'thickness' of the violins at the top is quite pronounced in all districts, which means that high maximum humidity values are very common. The pointed tops of the violins suggest that the distribution of maximum humidity values tails off as it approaches 100%.

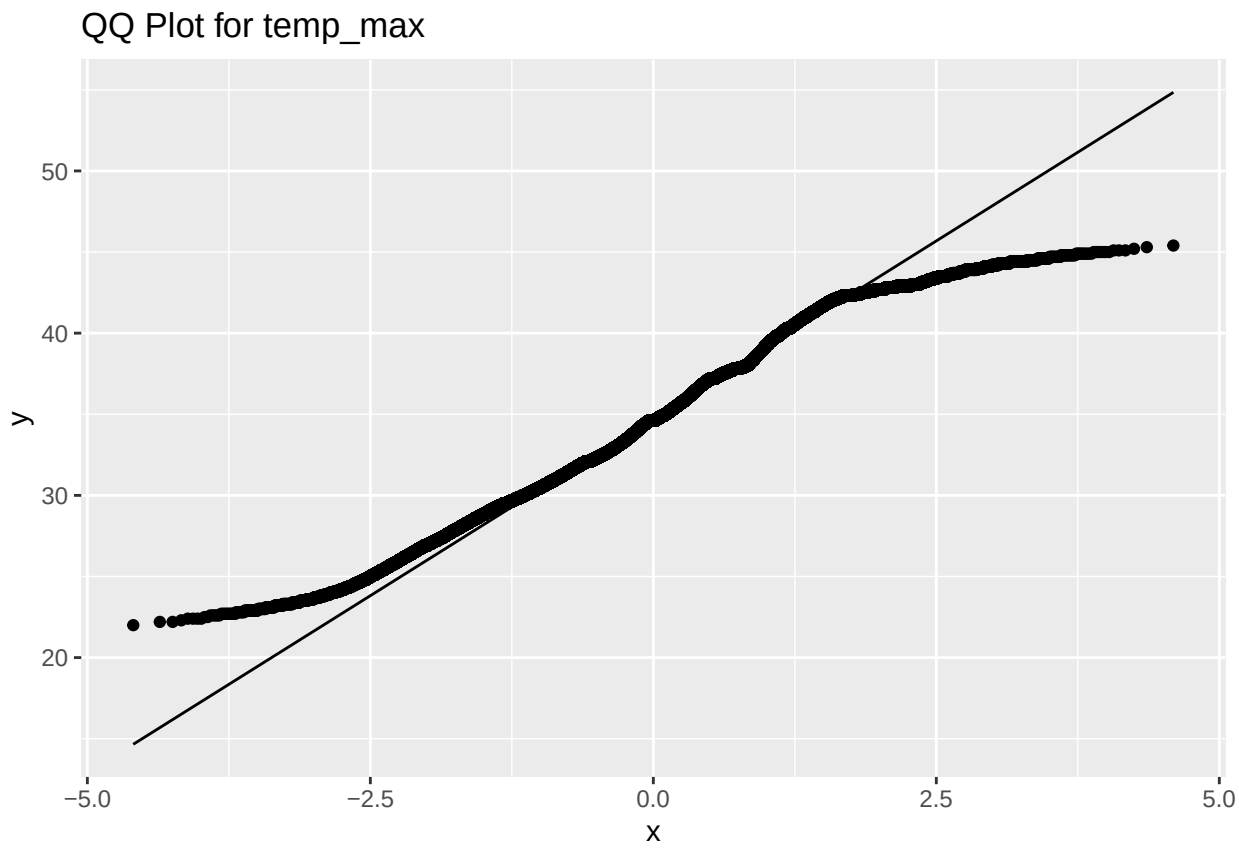
Analyzing Data Distribution

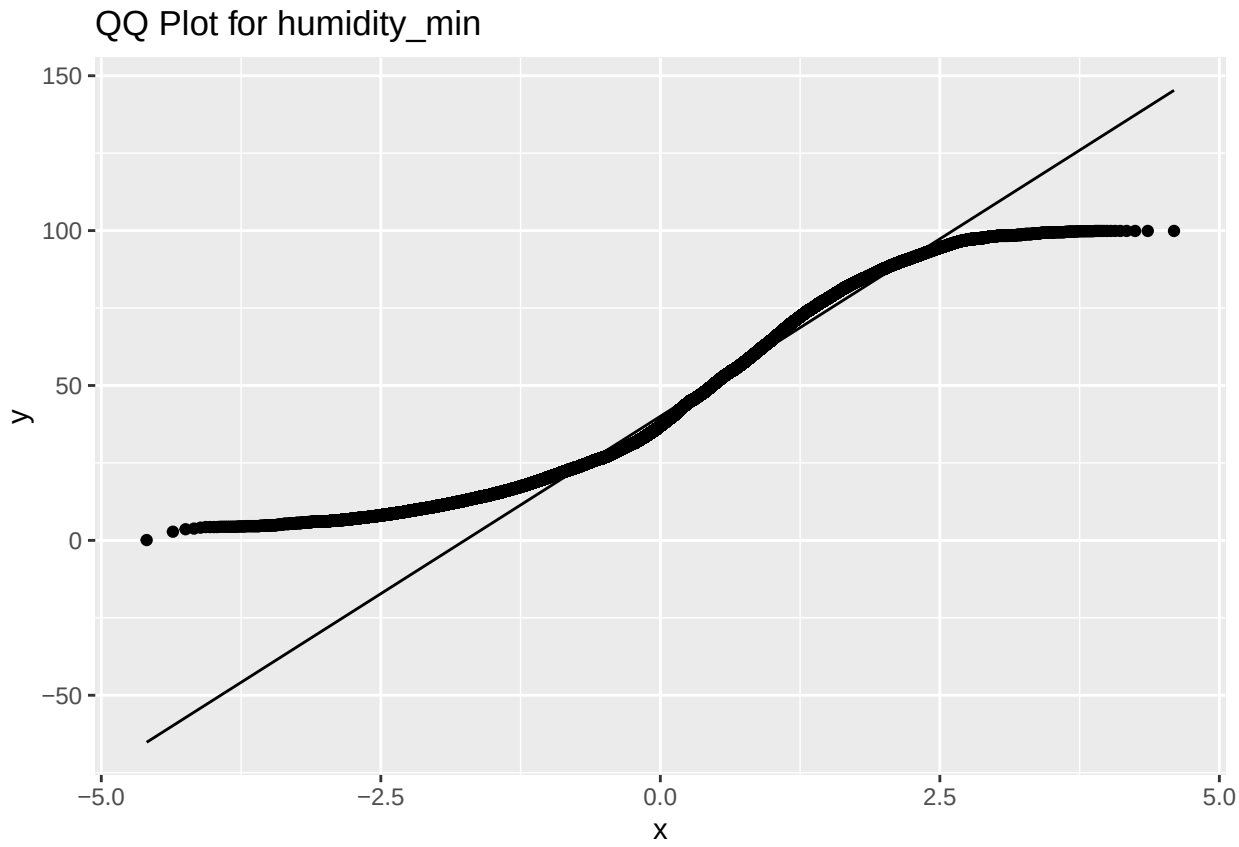
QQ Plot I am using Quantile-Quantile (QQ) plot to analyze the distribution of data. Basically, QQ plot is a graphical tool used to assess whether a given dataset follows a particular theoretical distribution, such as a normal distribution.

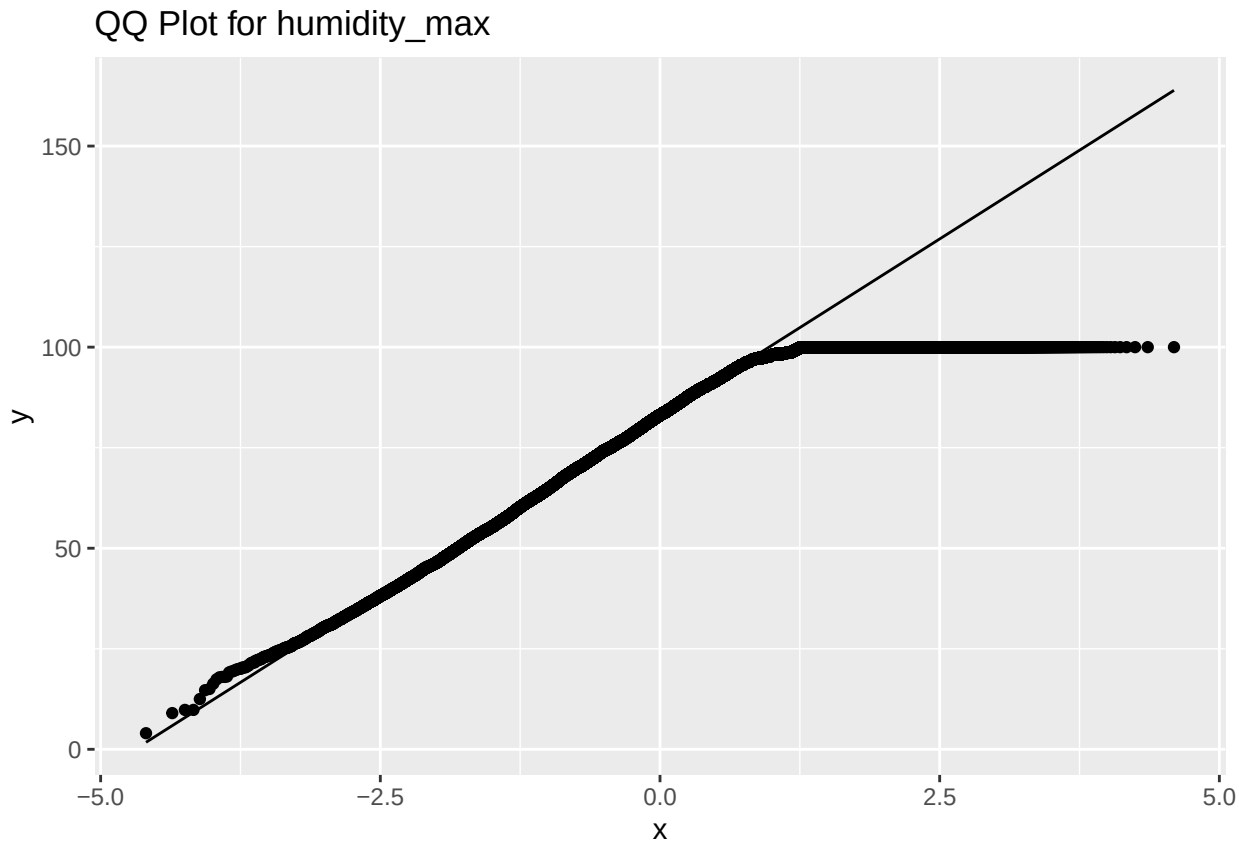
```
# Creating a function to generate QQ plots
plot_qq <- function(column_name) {
```

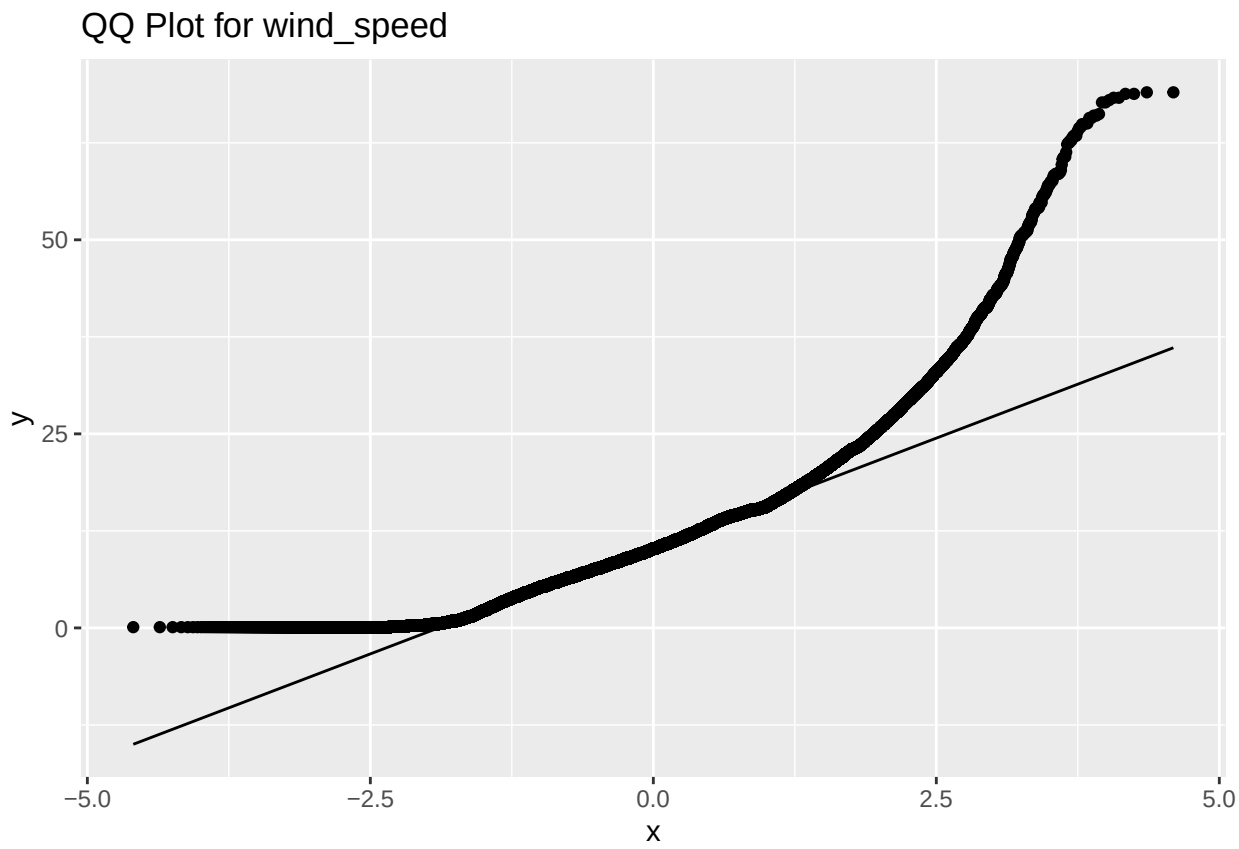
```r
  ggplot(my_data, aes(sample = .data[[column_name]])) +
    geom_qq() +
    geom_qq_line() +
    labs(title = paste("QQ Plot for", column_name))
}

# Applying the function to each numeric column in the dataset
numeric_columns <- sapply(my_data, is.numeric)
numeric_column_names <- names(numeric_columns)[numeric_columns]

# Plot QQ plots for each numeric column
for (column_name in numeric_column_names) {
  print(plot_qq(column_name))
}
```




QQ Plot for temp_max
50
40
30
20
y
−5.0
−2.5
0.0
2.5
5.0
x


QQ Plot for humidity_min
150
100
50
0
−50
y
−5.0
−2.5
0.0
2.5
5.0
x


QQ Plot for humidity_max
150
100
50
0
y
−5.0
−2.5
0.0
2.5
5.0
x



From the above QQ Plots, we can conclude the following

1. **QQ Plot for temp_min:** The points follow the line closely in the central range but deviate at both ends, especially at the lower tail. This indicates that the distribution of 'temp_min' is roughly similar to a normal distribution.

2. **QQ Plot for temp_max:** The points follow the line closely in the middle but deviate at the ends. The lower tail deviates less than the 'temp_min', while the upper tail shows a sharp upward deviation, indicating possible outliers in the higher temperature range.

3. **QQ Plot for humidity_min:** The plot shows a strong deviation from the line in both tails, with the lower tail indicating that the minimum humidity values are much higher than what would be expected in a normal distribution. The upper tail also suggests outliers on the higher end.

4. **QQ Plot for humidity_max:** This plot is similar to that of 'humidity_min', with deviations at both ends. The points in the middle follow the line quite well, suggesting that a portion of the data is normally distributed. However, the tails, especially the upper tail, indicate the presence of outliers or an extreme range of maximum humidity values.

5. **QQ Plot for wind_speed:** The data points for wind speed show a strong deviation from the normal distribution line at the upper tail, suggesting that there are more high wind speed values than would be expected in a normal distribution. The middle part of the data follows the line closely, but the lower tail shows some deviation, indicating less frequent low wind speed values.

Bivariate Analysis

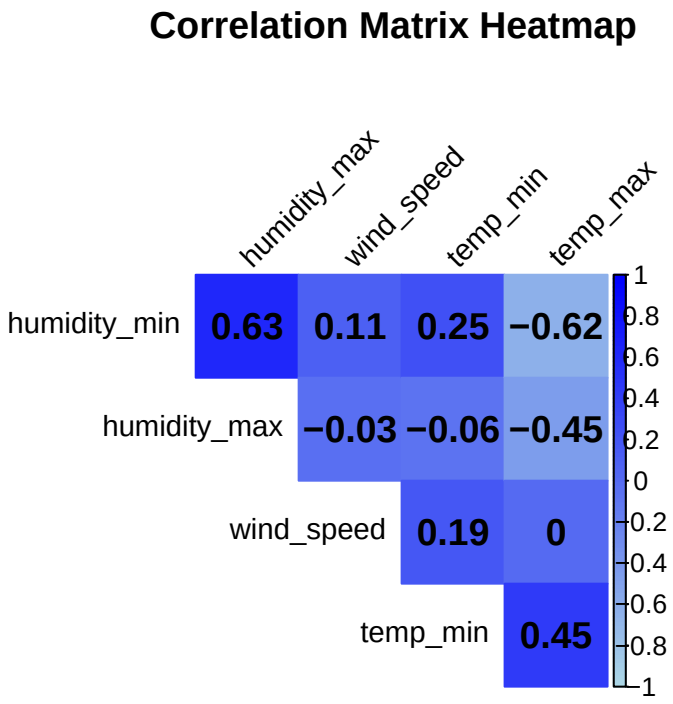
Bivariate Analysis can be defined as a method that compares two variables, such as Correlation.

```r
numeric_columns <- my_data[, sapply(my_data, is.numeric)]

# Computing Pearson correlation coefficients
cor_matrix <- cor(numeric_columns)

# Print the correlation matrix
print(cor_matrix)
```

Correlation

```
##                temp_min     temp_max humidity_min humidity_max   wind_speed
## temp_min      1.0000000  0.453317351    0.2469399  -0.06292710  0.193527173
## temp_max      0.4533174  1.000000000   -0.6245801  -0.45078959  0.004387721
## humidity_min  0.2469399 -0.624580119    1.0000000   0.62811284  0.113855909
## humidity_max -0.0629271 -0.450789586    0.6281128   1.00000000 -0.027225906
## wind_speed    0.1935272  0.004387721    0.1138559  -0.02722591  1.000000000
```

```r
# Plotting the correlation matrix
corrplot(
  cor_matrix,
  method = "color",
  type = "upper",
  order = "hclust",
  tl.col = "black",
  tl.srt = 45,
  diag = FALSE,
  addCoef.col = "black",
  tl.cex = 0.9,
  number.cex = 1.2,
  col = colorRampPalette(c("lightblue", "blue"))(100),
  title = "Correlation Matrix Heatmap",
  mar = c(2, 2, 2, 10)
)
```



The above correlation matrix heatmap displays the Pearson correlation coefficients between pairs of variables, which measure the linear relationship between them.

The interpretation for each pair:

humidity_min and humidity_max (0.63): A moderate positive correlation suggests that days with higher minimum humidity levels tend to have higher maximum humidity levels as well.

humidity_min and wind_speed (0.11): A very weak positive correlation suggests that there's almost no linear relationship between minimum humidity and wind speed.

humidity_min and temp_min (-0.62): A moderate negative correlation implies that higher minimum humidity levels are associated with lower minimum temperatures.

humidity_min and temp_max (0.45): A moderate positive correlation suggests that higher minimum humidity levels tend to occur on days with higher maximum temperatures.

humidity_max and wind_speed (-0.06): A very weak negative correlation, indicating no significant linear relationship between maximum humidity and wind speed.

humidity_max and temp_min (-0.45): A moderate negative correlation suggests that days with higher maximum humidity levels tend to have lower minimum temperatures.

humidity_max and temp_max (0): No correlation indicates that there is no linear relationship between maximum humidity and maximum temperature.

wind_speed and temp_min (0.19): A weak positive correlation suggests that higher wind speeds are slightly associated with higher minimum temperatures.

wind_speed and temp_max (0): No correlation, suggesting no linear relationship between wind speed and maximum temperature.

temp_min and temp_max (0.45): A moderate positive correlation indicates that higher minimum temperatures on a given day are associated with higher maximum temperatures.

The colors on the heatmap correspond to the correlation values, with darker shades typically indicating stronger correlations (either positive or negative). Lighter shades or white represent weaker correlations.

Hypothesis Tests

A Hypothesis test can be simply defined as the processing of testing whether a claim is valid or not.

Implementing T-test statistic A t-test is a statistical test that compares the means of two groups to determine whether there is statistical significance or not

Context:

In the city of Hyderabad, the onset of summer often brings significant heat, with temperatures rising markedly during the month of May. This period is crucial for various sectors including public health, agriculture, and energy supply, as they must prepare for the potential impact of high temperatures. These People are interested in understanding whether the minimum daily temperatures exceed a comfortable threshold, which might necessitate heat advisories or the implementation of heat action plans.

Question :

In the month of May, is there evidence to suggest that the average minimum daily temperature in Hyderabad exceeds 20°C, which is considered a threshold for initiating preventive measures against heat-related issues?

Null Hypothesis (H0): The mean of temp_min in Hyderabad for May is less than or equal to 20°C.

Alternative Hypothesis (H1): The mean of temp_min in Hyderabad for May is greater than 20°C.

```
# Extracting the data for Hyderabad in May
hyderabad_may_data <- my_data %>%
  filter(District == "Hyderabad", format(Date, "%Y-%m") == "2018-05")


# Performing one-sample t-test
t_test_result <- t.test(hyderabad_may_data$temp_min, mu = 20, alternative = "greater")


alpha <- 0.05

# Printing  the results
cat("T-Test Results:\n")
```

```
## T-Test Results:
```

```
cat("p-value:", t_test_result$p.value, "\n")
```

```
## p-value: 7.456806e-306
```

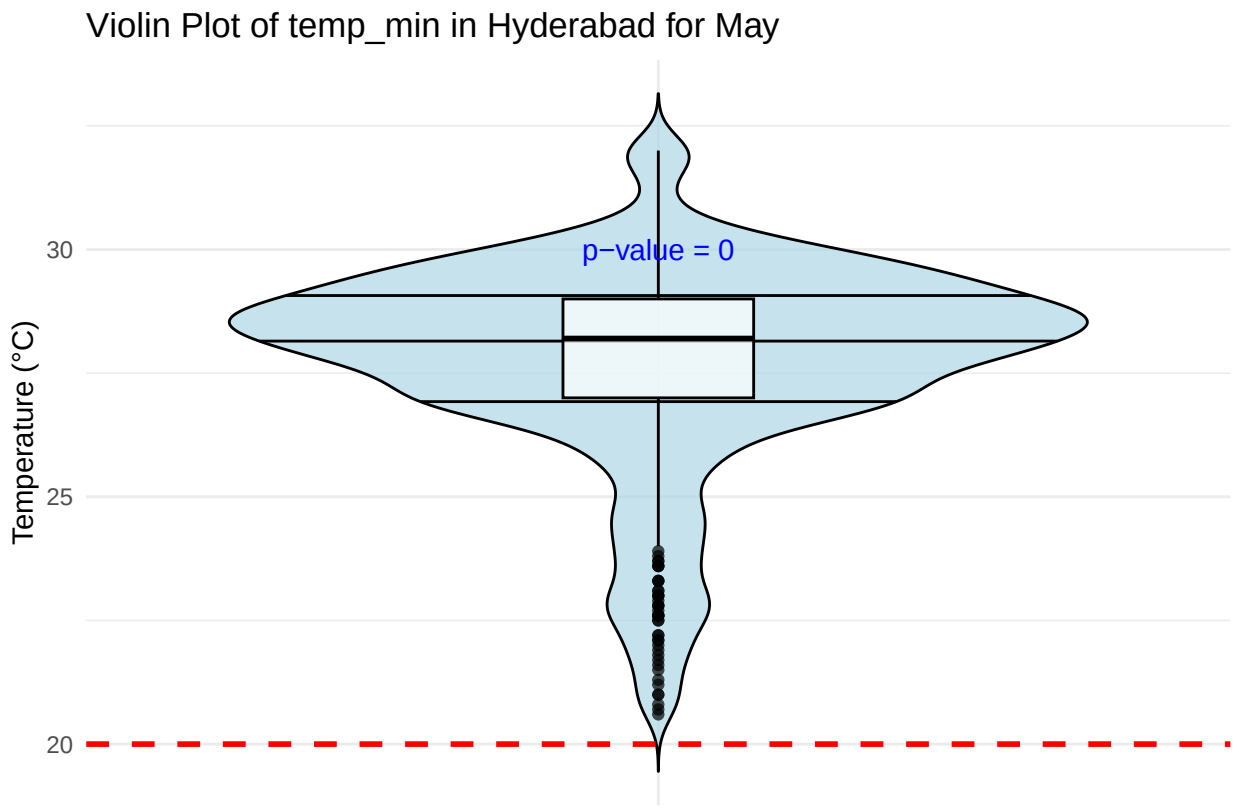
```
cat("Mean temperature (temp_min) for Hyderabad in May:", mean(hyderabad_may_data$temp_min), "°C\n")
```

```
## Mean temperature (temp_min) for Hyderabad in May: 27.7259 °C
```

```r
# Checking if the p-value is less than the significance level
if (t_test_result$p.value < alpha) {
  cat("Result: Reject the null hypothesis. There is enough evidence to suggest that the mean temp_min i
} else {
  cat("Result: Fail to reject the null hypothesis. There is not enough evidence to suggest that the mea
}
```

```
## Result: Reject the null hypothesis. There is enough evidence to suggest that the mean temp_min in Hy
```

```r
# Creating a violin plot of the data
ggplot(hyderabad_may_data, aes(x = "", y = temp_min, fill = factor(District))) +
  geom_violin(trim = FALSE, draw_quantiles = c(0.25, 0.5, 0.75), fill = "lightblue", color = "black", a
  geom_boxplot(width = 0.2, fill = "white", color = "black", alpha = 0.7) +

  # Adding a horizontal line at the hypothesized mean
  geom_hline(yintercept = 20, linetype = "dashed", color = "red", linewidth = 1) +

  # Adding labels and title
  labs(
    title = "Violin Plot of temp_min in Hyderabad for May",
    x = "",
    y = "Temperature (°C)"
  ) +

  theme_minimal() +
  theme(
    axis.text.x = element_blank(),
    legend.position = "none"
  ) +

  # Printing the p-value on the plot
  annotate(
    "text",
    x = 1,
    y = max(hyderabad_may_data$temp_min) - 2,
    label = paste("p-value =", round(t_test_result$p.value, 4)),
    color = "blue"
  )
```



From the above Hypothesis test, the low p-value leads to rejecting Null hypothesis.This suggests an evidence that the mean temperature for May in Hyderabad is 27.73°C, significantly surpassing the 20°C threshold.

Implementing ANOVA ANOVA ,which stands for Analysis of Variance is a statistical test that compares the variances of group means to determine whether there is a statistical significance or not.

Context

In Telangana, the diverse geography leads to varying humidity levels across districts, influenced by factors like altitude, water bodies, urbanization, and vegetation. These differences can impact climate, agriculture, health, and the economy. For instance, districts with higher humidity might be more prone to certain agricultural pests or diseases, while those with lower humidity might experience different agricultural or health challenges.

Question

Is there a significant variation in the average maximum humidity among the different districts in Telangana?

Null Hypothesis (H0): There is no difference in average maximum humidity among all the districts.

Alternative Hypothesis (H1): At least one district has a different average maximum humidity.

```
# Performing ANOVA
anova_result <- aov(humidity_max ~ District, data = my_data)

# Display ANOVA summary
cat("ANOVA Results:\n")
```

```
## ANOVA Results:
```

```r
print(summary(anova_result))
```

```
##                 Df   Sum Sq Mean Sq F value Pr(>F)
## District        30  4112839  137095   640.1 <2e-16 ***
## Residuals   230353 49339602     214
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
# Performing Tukey's test for multiple comparisons
tukey_result <- TukeyHSD(anova_result)

# Extract p-values for pairwise comparisons
p_values <- tukey_result$District[, 4]

alpha <- 0.05

# Couning the number of significant comparisons
significant_comparisons <- sum(p_values < alpha)

# Print the results
if (significant_comparisons > 0) {
  cat("Reject the null hypothesis. At least one district has a significantly different mean 'humidity_ma
} else {
  cat("Fail to reject the null hypothesis. There is no significant difference in mean 'humidity_max' am
}
```

```
## Reject the null hypothesis. At least one district has a significantly different mean 'humidity_max'.
```

From the above Hypothesis test, the low p-value ($p < 0.05$) from ANOVA indicates rejecting the null hypothesis. This provides evidence that the average maximum humidity differs significantly among districts.

Implementing Chi-Square Test Statistic Chi-Square test is a statistical test that is primarily used to analyze categorical data. It compares the difference between observed data and expected data to determine statistical significance.

Context

Telangana's climate can vary considerably from one district to another due to regional differences in geography and urbanization. This variation could potentially manifest in the occurrence of hot days. Identifying if certain districts are more prone to hot days can be critical for public health planning, energy management, and agricultural practices.

Question

Is the likelihood of experiencing a hot day dependent on the district in Telangana?

Null Hypothesis (H0): There is no association between being a hot day and the district.

Alternative Hypothesis (H1): There is a significant association between being a hot day and the district.

```r
# creating a new column
my_new_data <- my_data %>%
mutate(temp_average = (temp_min + temp_max) / 2)
```

```r
# converting a column to binary column
my_new_data$hot_day <- ifelse(my_new_data$temp_average > 28.50, 1, 0)

contingency_table <- table(my_new_data$District, my_new_data$hot_day)

view(contingency_table)

# performing Chi-square test
chi_squared_test <- chisq.test(contingency_table)

p_value <- chi_squared_test$p.value

alpha <- 0.05

# Comparing p-value with significance level
if (p_value < alpha) {
  cat("Reject the null hypothesis. There is a significant association between being a hot day and the di
} else {
  cat("Fail to reject the null hypothesis. There is no significant association between being a hot day a
}
```

```
## Reject the null hypothesis. There is a significant association between being a hot day and the distri
```

From the above Hypothesis Test, the Chi-Square test results in a low p-value ($p < 0.05$), leading to the rejection of the null hypothesis. This indicates a significant association between being a hot day and the district in Telangana.

Temporal Trends for Agriculture Insights

In this section, I am trying to find the temporal trends of three distinct periods—January to March, April to June, and July to September—in Telangana's districts. My primary focus is to calculate the mean temperatures during these temporal windows and recommend suitable crops based on the temperatures .Moreover, the analysis of these temporal trends holds significance for crop planning, providing benefits to farmers, agro-industries, and government agencies

```r
# creating a new average temperature column
my_temp_data <- my_data %>%
mutate(temp_avg = (temp_min + temp_max) / 2)

# Checking for duplicated rows based on the "Date" column
duplicated_rows <- duplicated(my_temp_data$Date)

# Removing duplicated rows
my_temp_data <- my_temp_data[!duplicated_rows, ]

# Convert to tsibble
ts_temp_avg <- as_tsibble(my_temp_data, index = Date)
```

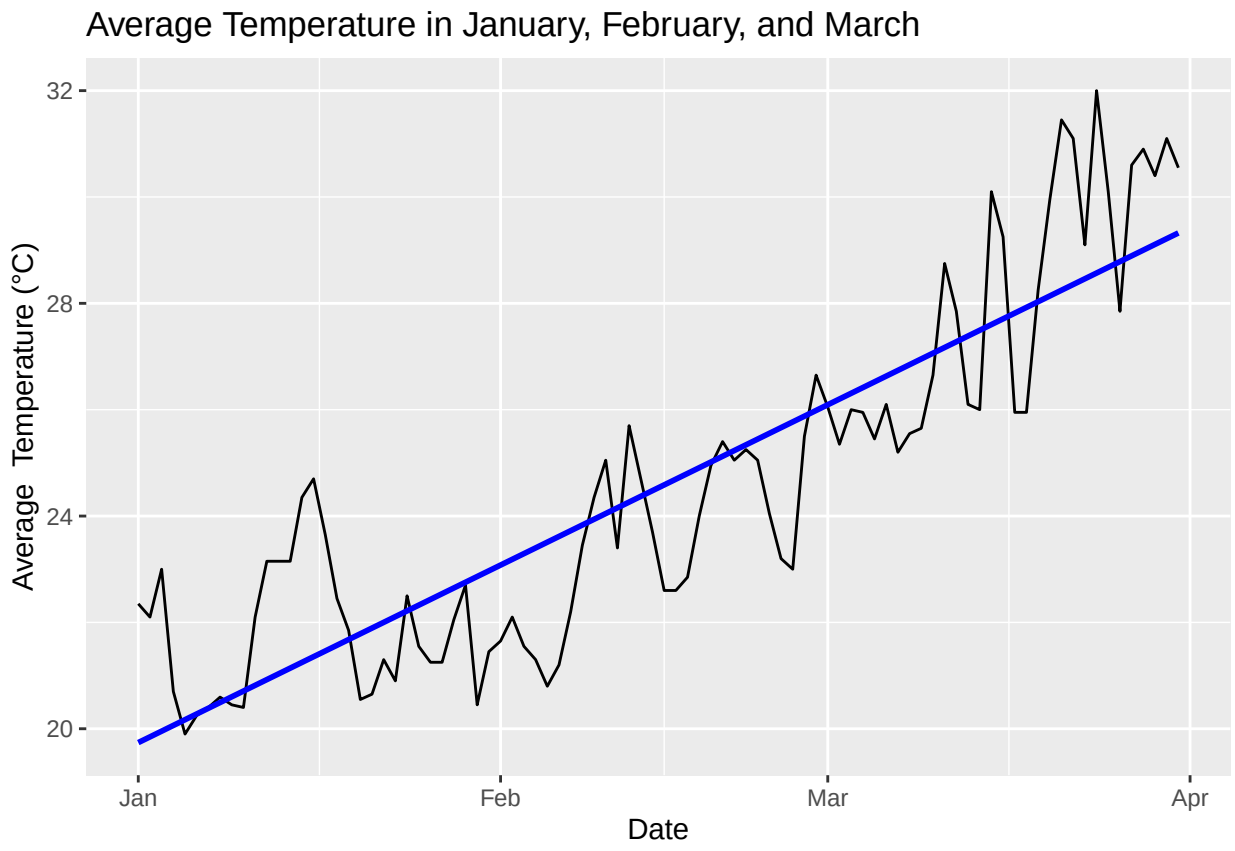
```r
filtered_data <-ts_temp_avg %>%
  filter(month(Date) %in% c(1, 2, 3))
```

```r
ts_data <- as_tsibble(filtered_data, index = Date)
```

```r
ggplot(ts_data, aes(x = Date, y = temp_avg)) +
  geom_line() +
  geom_smooth(method = 'lm', color = 'blue', se=FALSE) +
  labs(title = "Average Temperature in January, February, and March", x = "Date", y = "Average  Tempera
```

Temporal Trends From January to March

```
## `geom_smooth()` using formula = 'y ~ x'
```




```r
cat("The mean temperature for jan,feb,march is:",mean(ts_data$temp_avg))
```

```
## The mean temperature for jan,feb,march is: 24.53161
```

From the above analysis, it can be concluded that the average temperature from January to March is 24.5 degrees Celsius.

Here are some crops that typically thrive in a temperature range around 24.5 degrees Celsius:

Maize (Corn): Maize is a warm-season crop that thrives in temperatures between 21-27 degrees Celsius.

Rice: Rice is a staple crop that grows well in warm climates, and the temperature range of 24.5 degrees Celsius is suitable for its cultivation.

Tomatoes: Tomatoes are warm-season vegetables that prefer temperatures between 20-26 degrees Celsius.

Beans (Green Beans, Snap Beans): Beans are generally warm-weather crops and can be cultivated in temperatures around 18-24 degrees Celsius.

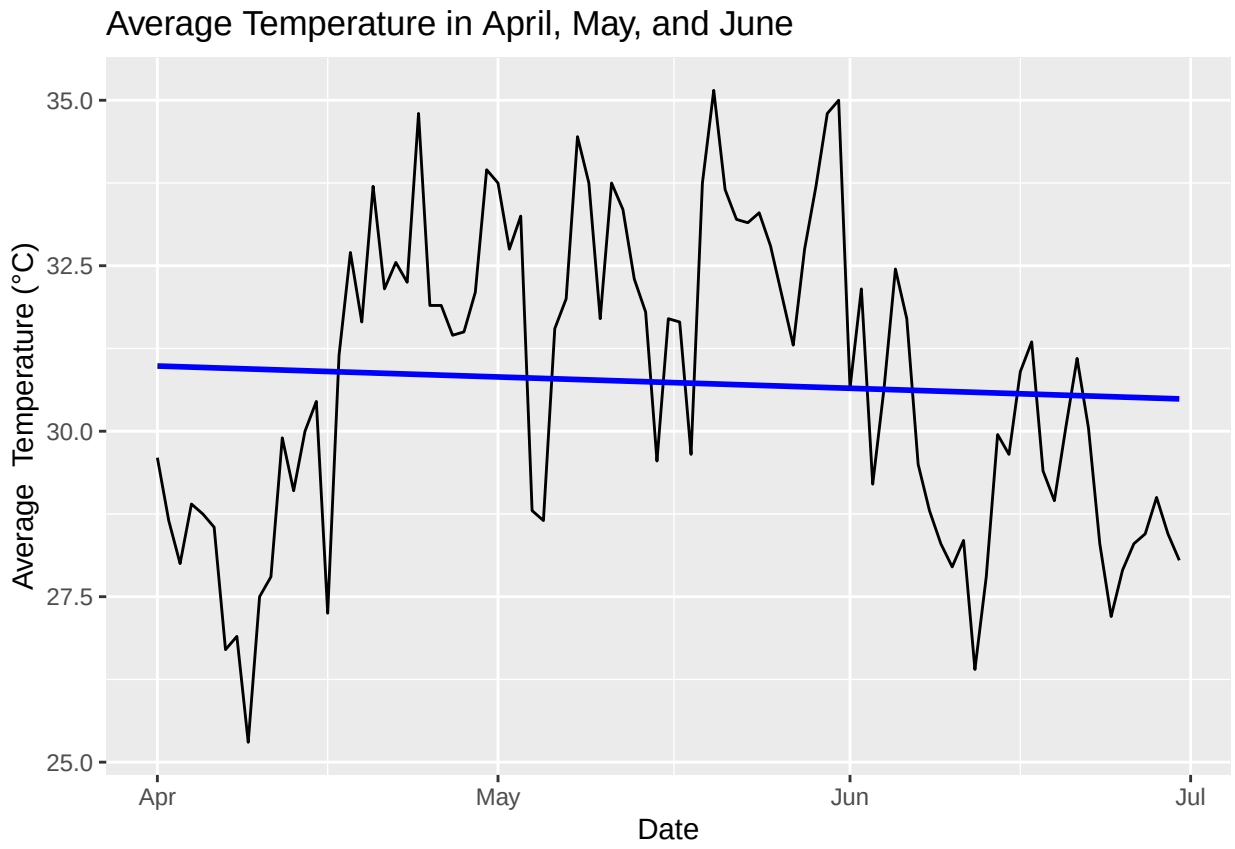
Cotton: Cotton is a warm-season crop that thrives in temperatures between 20-30 degrees Celsius.

Soybeans: Soybeans are warm-weather legumes that prefer temperatures between 20-26 degrees Celsius.

```
filtered_data <-ts_temp_avg %>%
  filter(month(Date) %in% c(4, 5, 6))


ts_data <- as_tsibble(filtered_data, index = Date)



ggplot(ts_data, aes(x = Date, y = temp_avg)) +
  geom_line() +
  geom_smooth(method = 'lm', color = 'blue', se=FALSE) +
  labs(title = "Average Temperature in April, May, and June", x = "Date", y = "Average  Temperature (°C)
```

Temporal Trends From April to June

```
## `geom_smooth()` using formula = 'y ~ x'
```




```
cat("The mean temperature for April,may,june is:",mean(ts_data$temp_avg))
```

```
## The mean temperature for April,may,june is: 30.73681
```

From the above analysis, we can infer that the average temperature from April to June is 31 degrees Celsius.

Here are some crops that are generally well-suited for high temperatures around 31 degrees Celsius:

Chilies and Peppers: Hot peppers, such as chili peppers and bell peppers, generally thrive in warm to hot temperatures.

Eggplant (Brinjal): Eggplants are warm-season vegetables that do well in high temperatures.

Okra: Okra, also known as ladyfinger, is a heat-loving vegetable and grows well in hot climates.

Sorghum: Sorghum is a cereal grain that is well-adapted to high-temperature conditions.

Millet: Certain types of millet, such as pearl millet, are drought-tolerant and can withstand high temperatures.

Sunflowers: Sunflowers are heat-tolerant and can be grown in hot climates.

Cowpeas (Black-eyed Peas): Cowpeas are heat-loving legumes that can be grown in warm conditions.

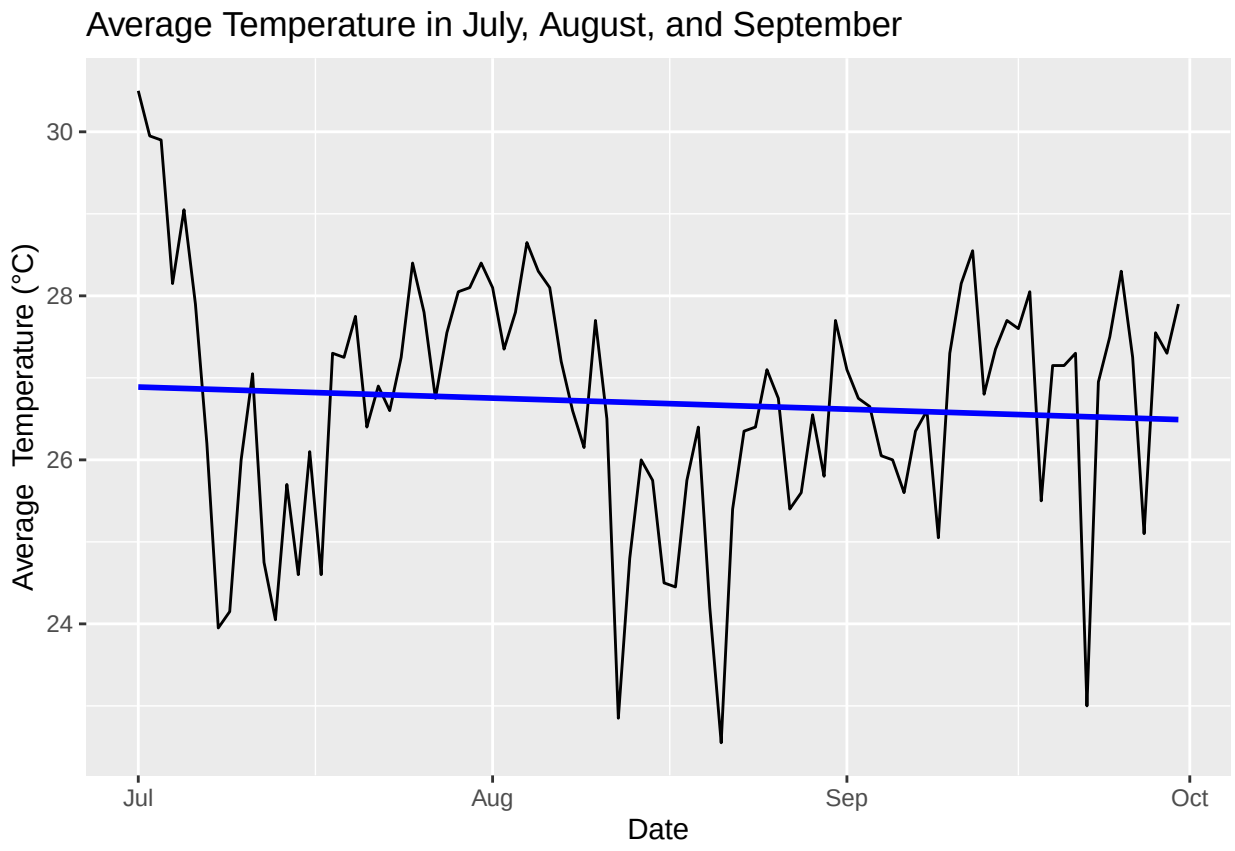
Tropical Fruits (e.g., Mangoes, Pineapples, Bananas): Many tropical fruits thrive in hot temperatures, and mangoes, pineapples, and bananas are examples of such crops.

Cucumber and Watermelon: These are heat-tolerant fruits that can do well in warm to hot climates.

```r
filtered_data <-ts_temp_avg %>%
  filter(month(Date) %in% c(7, 8, 9))


ts_data <- as_tsibble(filtered_data, index = Date)



ggplot(ts_data, aes(x = Date, y = temp_avg)) +
  geom_line() +
  geom_smooth(method = 'lm', color = 'blue', se=FALSE) +
  labs(title = "Average Temperature in July, August, and September", x = "Date", y = "Average  Temperatu
```

Temporal Trends From July to September

```
## `geom_smooth()` using formula = 'y ~ x'
```




```r
cat("The mean temperature for july,august,september is:",mean(ts_data$temp_avg))
```

```
## The mean temperature for july,august,september is: 26.68967
```

From the above analysis, we can conclude that the average temperature from July to September is 27 degrees Celsius.

Here are some crops that are commonly grown in conditions around 27 degrees Celsius:

Rice: Rice is a staple crop that thrives in warm temperatures, and an average of 27 degrees Celsius is suitable for its cultivation.

Maize (Corn): Maize is a warm-season crop that does well in temperatures between 21-27 degrees Celsius.

Tomatoes: Tomatoes are warm-season vegetables that prefer temperatures between 20-26 degrees Celsius.

Peppers (Bell Peppers, Chili Peppers): Peppers are warm-season crops that typically grow well in temperatures around 25-29 degrees Celsius.

Potatoes: Potatoes are cool-season crops but can tolerate moderately warm temperatures. An average of 27 degrees Celsius is generally suitable for potato cultivation.

Lettuce: Lettuce is a cool-season crop, but certain varieties can be grown in slightly warmer temperatures, especially if provided with some shade.

Cucumbers: Cucumbers are warm-season vegetables that can thrive in temperatures around 27 degrees Celsius.

Cabbage: Cabbage is a cool-season crop, but certain varieties can be grown in warmer temperatures.

Soybeans: Soybeans are warm-weather legumes that prefer temperatures between 20-26 degrees Celsius.

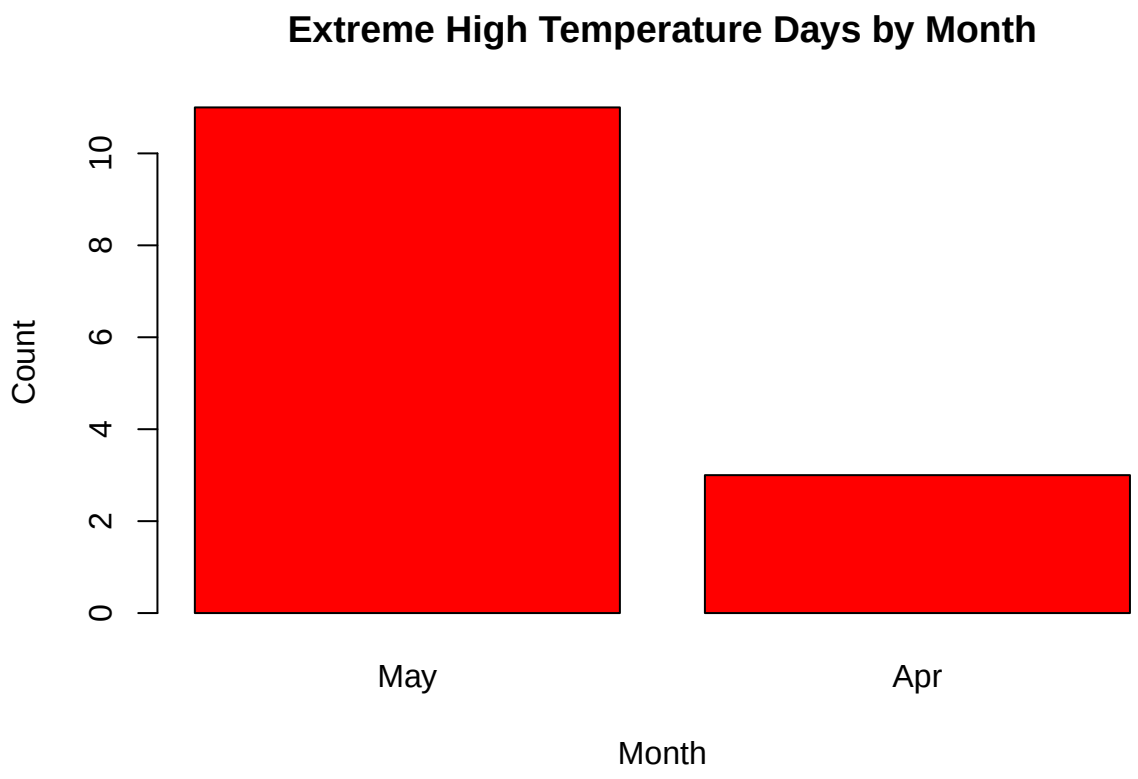
Insights of Extreme weather Events

In this section, I aim to identify the months characterized by extreme temperatures. This outcome holds significance for government and weather forecasting agencies, providing valuable insights into the periods with exceptional temperature conditions

Extreme High Temperatures I want to figure out the month in Telangana that experiences extreme high temperatures

```
# Extracting  month from the date and convert it to abbreviated month name
my_temp_data$month <- format(my_temp_data$Date, "%b")

# Checking  if there are extreme high temperature events
if (nrow(my_temp_data) > 0) {
  # Identifying  extreme high temperature events
  high_temp_threshold <- quantile(my_temp_data$temp_avg, 0.95)
  extreme_high_temps <- my_temp_data[my_temp_data$temp_avg > high_temp_threshold, ]

  # Counting the occurrences of extreme high temperature days for each month
  extreme_high_temp_counts <- table(extreme_high_temps$month)

  # Converting the result to a data frame for easy plotting
  extreme_high_temp_counts_df <- data.frame(month = names(extreme_high_temp_counts), count = as.numeric

  # Sorting the data frame by count in descending order
  extreme_high_temp_counts_df <- extreme_high_temp_counts_df[order(-extreme_high_temp_counts_df$count),

  # Plot the bar graph
  barplot(extreme_high_temp_counts_df$count, names.arg = extreme_high_temp_counts_df$month,
          col = "red", main = "Extreme High Temperature Days by Month",
          xlab = "Month", ylab = "Count")
} else {
```

```
  print("No extreme high temperature events found.")
}
```

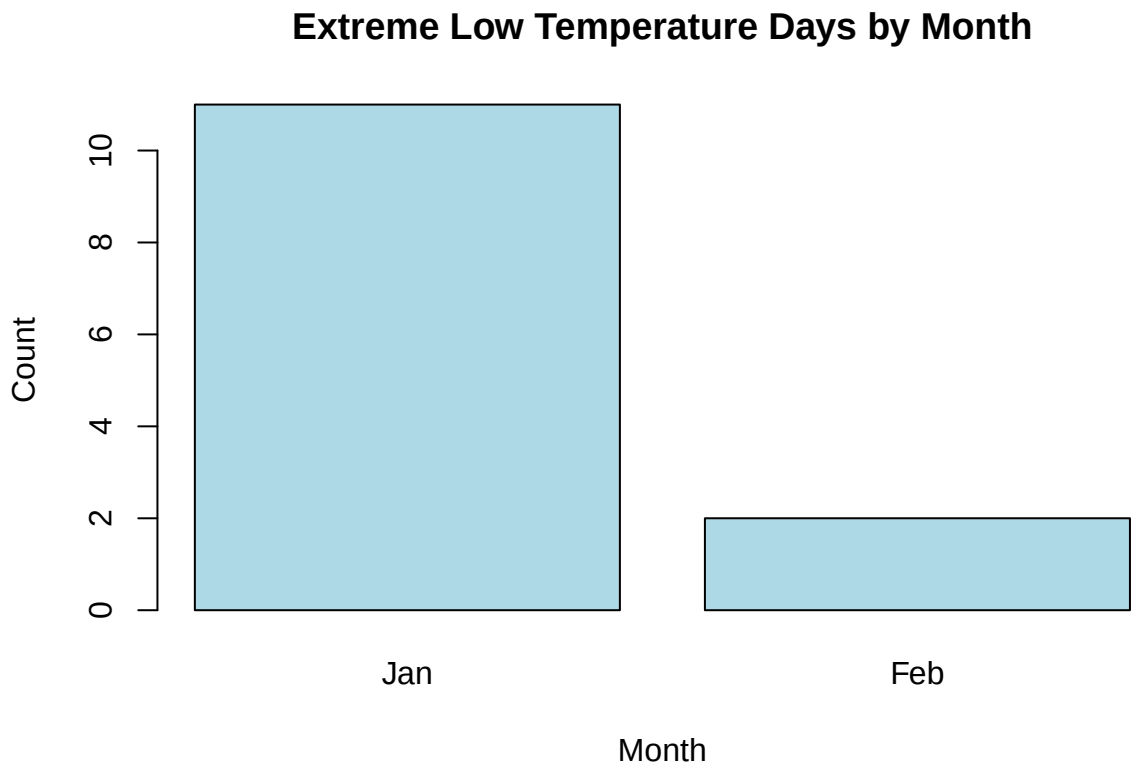


From the above bar graph , we can conclude that May is the hottest month in Telangana, experiencing extreme high temperatures

Extreme Low Temperatures I aim to identify the month in Telangana that experiences extreme low temperatures

```
# Extracting month from the date and convert it to abbreviated month name
my_temp_data$month <- format(my_temp_data$Date, "%b")

# Identifying extreme low temperature events
low_temp_threshold <- quantile(my_temp_data$temp_avg, 0.05)
extreme_low_temps <- my_temp_data[my_temp_data$temp_avg < low_temp_threshold, ]

# Checking if there are extreme low temperature events
if (nrow(extreme_low_temps) > 0) {
  # Counting the occurrences of extreme low temperature days for each month
  extreme_low_temp_counts <- table(extreme_low_temps$month)

  # Converting the result to a data frame for easy plotting
  extreme_low_temp_counts_df <- data.frame(month = names(extreme_low_temp_counts),count = as.numeric(ex

  # Sorting the data frame by count in descending order
```

```r
  extreme_low_temp_counts_df <- extreme_low_temp_counts_df[order(-extreme_low_temp_counts_df$count),]

  # Plot the bar graph for extreme low temperature days
  barplot(extreme_low_temp_counts_df$count, names.arg = extreme_low_temp_counts_df$month,
          col = "lightblue", main = "Extreme Low Temperature Days by Month",
          xlab = "Month", ylab = "Count")
} else {
  print("No extreme low temperature events found.")
}
```



From the above bar graph , we can conclude that January is the coldest month in Telangana, experiencing extreme low temperatures.

Conclusion

In conclusion, this analysis successfully extracted valuable insights from the weather data, primarily benefiting agriculture and government organizations. The initial stages of the analysis involved understanding the data by estimating location and variability. Subsequently, we delved into exploring the relationships between variables through the utilization of correlation matrices and various visualizations.The comprehensive examination led us to the analysis of temporal trends and extreme temperatures . These findings are particularly valuable for farmers, agro-industries, and government agencies, enabling them to make informed decisions and undertake proactive measures in response to climatic variations. The amalgamation of statistical analyses and visualizations throughout this project has contributed to a nuanced understanding of the weather dataset, emphasizing its potential impact on key stakeholders.

Stakeholders Farmers:

- **Interest:** Farmers would be highly interested in the temporal trends and recommended crops. They can use this information for crop planning, selection, and managing agricultural activities based on seasonal variations.
- **Impact:** Improved decision-making regarding crop choices and planting times, leading to potentially higher yields and better resource management.

Agro-Industries and Suppliers:

- **Interest:** Companies involved in the agricultural supply chain, such as seed and fertilizer suppliers, would be interested in the recommended crops for each season.
- **Impact:** Tailoring product offerings and marketing strategies based on the identified crops suitable for specific temperature ranges.

Government Agricultural Departments:

- **Interest:** Agricultural departments at the local and regional levels would be interested in understanding climate patterns to develop policies, provide guidance to farmers, and allocate resources effectively.
- **Impact:** Informed policymaking, targeted support programs, and resource allocation based on seasonal variations and climate trends.

Weather Forecasting Agencies:

- **Interest:** Organizations providing weather forecasts would be interested in understanding extreme weather events like hottest or coldest months.
- **Impact:** Improved accuracy and specificity in weather predictions for the region, aiding in better preparedness for extreme conditions.