Laboratory Examination 2

DM 101: Data Mining 1

Ram Reniel Cañido, Jeshua Lexis Labio, C.J. Odato, Jan Iris Oiga, Virgilio Salcedo II

February 21, 2025

Loading Necessary Libraries

```
library(readxl)
library(openxlsx)
library(dplyr)
library(ggplot2)
library(zoo)
```

Loading Necessary Data

Loading three tables (Humidity, Meteorological Data, Synoptic Information) about meteorological information.

I. HUMIDITY

```
humidity <- read_excel("C:/Users/Micah/Downloads/HUMIDITY.xlsx")
print(humidity)
```

```
## # A tibble: 396 x 3
##     YEAR `SYNOPTIC STATION` HUMIDITY
##    <dbl> <chr>                 <dbl>
##  1  2013 A                      87.3
##  2  2013 A                      83.3
##  3  2013 A                      86.4
##  4  2013 A                      84.5
##  5  2013 A                      88.8
##  6  2013 A                      91.9
##  7  2013 A                      92.3
##  8  2013 A                      92.7
##  9  2013 A                      92.0
## 10  2013 A                      91.3
## # i 386 more rows
```

II. METEOROLOGICAL DATA

```r
metData <- read_excel("C:/Users/Micah/Downloads/METEOROLOGICAL DATA_1.xlsx")
print(metData)
```

```
## # A tibble: 396 x 5
##     YEAR `SYNOPTIC STATION` MONTH     RAINFALL TEMPERATURE
##    <dbl> <chr>              <chr>        <dbl>       <dbl>
##  1  2013 A                  JANUARY      0.493        18.3
##  2  2013 A                  FEBRUARY     1.08         19.8
##  3  2013 A                  MARCH        2.18         20.4
##  4  2013 A                  APRIL        2.47         21.6
##  5  2013 A                  MAY         11.1          20.9
##  6  2013 A                  JUNE         7.89         20.6
##  7  2013 A                  JULY        12.0          20.1
##  8  2013 A                  AUGUST      39.5          19.2
##  9  2013 A                  SEPTEMBER   19.8          19.7
## 10  2013 A                  OCTOBER      7.87         18.7
## # i 386 more rows
```

III. SYNOPTIC INFORMATION

```r
synInfo <- read_excel("C:/Users/Micah/Downloads/SYNOPTIC INFORMATION.xlsx")
print(synInfo)
```

```
## # A tibble: 3 x 3
##   `SYNOPTIC STATION` REGION         LOCATION
##   <chr>              <chr>          <chr>
## 1 A                  CAR            Baguio City
## 2 B                  ILOCOS         Dagupan City
## 3 C                  CAGAYAN VALLEY Basco, Batanes
```

INTEGRATION OF DATASETS

Integrating Humidity Data, Meteorological Data, and Synoptic Information into one table.

```r
mergeData <- left_join(x=humidity, y = metData, by= c("YEAR", "SYNOPTIC STATION"), relationship = "many
finalData <- left_join(x=mergeData, y=synInfo, by= "SYNOPTIC STATION")
print(finalData)
```

```
## # A tibble: 4,752 x 8
##     YEAR `SYNOPTIC STATION` HUMIDITY MONTH  RAINFALL TEMPERATURE REGION LOCATION
##    <dbl> <chr>                 <dbl> <chr>     <dbl>       <dbl> <chr>  <chr>
##  1  2013 A                      87.3 JANUA~    0.493        18.3 CAR    Baguio ~
##  2  2013 A                      87.3 FEBRU~    1.08         19.8 CAR    Baguio ~
##  3  2013 A                      87.3 MARCH     2.18         20.4 CAR    Baguio ~
##  4  2013 A                      87.3 APRIL     2.47         21.6 CAR    Baguio ~
##  5  2013 A                      87.3 MAY      11.1          20.9 CAR    Baguio ~
```

```
## 6 2013 A 87.3 JUNE 7.89 20.6 CAR Baguio ~
## 7 2013 A 87.3 JULY 12.0 20.1 CAR Baguio ~
## 8 2013 A 87.3 AUGUST 39.5 19.2 CAR Baguio ~
## 9 2013 A 87.3 SEPTE~ 19.8 19.7 CAR Baguio ~
## 10 2013 A 87.3 OCTOB~ 7.87 18.7 CAR Baguio ~
## # i 4,742 more rows
```

Saving the combined dataset as an Excel Document

```
write.xlsx(finalData, "Meteorological Information.xlsx")
```

Split Data by Synoptic Station

```
meteoInfo <- read_excel("Meteorological Information.xlsx")
stationA <- dplyr::filter(meteoInfo, `SYNOPTIC STATION` == "A")
stationB <- dplyr::filter(meteoInfo, `SYNOPTIC STATION` == "B")
stationC <- dplyr::filter(meteoInfo, `SYNOPTIC STATION` == "C")

write.xlsx(stationA, "StationA.xlsx")
write.xlsx(stationB, "StationB.xlsx")
write.xlsx(stationC, "StationC.xlsx")
```

I. SYNOPTIC STATION A

```
print(stationA)
```

```
## # A tibble: 1,584 x 8
##     YEAR `SYNOPTIC STATION` HUMIDITY MONTH  RAINFALL TEMPERATURE REGION LOCATION
##    <dbl> <chr>                 <dbl> <chr>     <dbl>       <dbl> <chr>  <chr>
##  1  2013 A                      87.3 JANUA~    0.493        18.3 CAR    Baguio ~
##  2  2013 A                      87.3 FEBRU~    1.08         19.8 CAR    Baguio ~
##  3  2013 A                      87.3 MARCH     2.18         20.4 CAR    Baguio ~
##  4  2013 A                      87.3 APRIL     2.47         21.6 CAR    Baguio ~
##  5  2013 A                      87.3 MAY      11.1          20.9 CAR    Baguio ~
##  6  2013 A                      87.3 JUNE      7.89         20.6 CAR    Baguio ~
##  7  2013 A                      87.3 JULY     12.0          20.1 CAR    Baguio ~
##  8  2013 A                      87.3 AUGUST   39.5          19.2 CAR    Baguio ~
##  9  2013 A                      87.3 SEPTE~   19.8          19.7 CAR    Baguio ~
## 10  2013 A                      87.3 OCTOB~    7.87         18.7 CAR    Baguio ~
## # i 1,574 more rows
```

II. SYNOPTIC STATION B

```
print(stationB)
```

```
## # A tibble: 1,584 x 8
##     YEAR `SYNOPTIC STATION` HUMIDITY MONTH  RAINFALL TEMPERATURE REGION LOCATION
##    <dbl> <chr>                 <dbl> <chr>     <dbl>       <dbl> <chr>  <chr>
##  1  2013 B                      82.0 JANUA~    0.484        26.4 ILOCOS Dagupan~
##  2  2013 B                      82.0 FEBRU~    0.199        27.0 ILOCOS Dagupan~
##  3  2013 B                      82.0 MARCH     3.39         28.7 ILOCOS Dagupan~
##  4  2013 B                      82.0 APRIL     3.78         30.6 ILOCOS Dagupan~
##  5  2013 B                      82.0 MAY       7.10         30.1 ILOCOS Dagupan~
##  6  2013 B                      82.0 JUNE      6.60         29.7 ILOCOS Dagupan~
##  7  2013 B                      82.0 JULY      8.24         28.8 ILOCOS Dagupan~
##  8  2013 B                      82.0 AUGUST   38.9          27.7 ILOCOS Dagupan~
##  9  2013 B                      82.0 SEPTE~   24.0          28.3 ILOCOS Dagupan~
## 10  2013 B                      82.0 OCTOB~    3.43         28.1 ILOCOS Dagupan~
## # i 1,574 more rows
```

III. SYNOPTIC STATION C

```r
print(stationC)
```

```
## # A tibble: 1,584 x 8
##     YEAR `SYNOPTIC STATION` HUMIDITY MONTH  RAINFALL TEMPERATURE REGION LOCATION
##    <dbl> <chr>                 <dbl> <chr>     <dbl>       <dbl> <chr>  <chr>
##  1  2013 C                      87.3 JANUA~    2.83         25.0 CAGAY~ Basco, ~
##  2  2013 C                      87.3 FEBRU~    1.24         26.8 CAGAY~ Basco, ~
##  3  2013 C                      87.3 MARCH     0.990        28.0 CAGAY~ Basco, ~
##  4  2013 C                      87.3 APRIL     0.969        29.2 CAGAY~ Basco, ~
##  5  2013 C                      87.3 MAY       2.67         29.6 CAGAY~ Basco, ~
##  6  2013 C                      87.3 JUNE      1.99         31.1 CAGAY~ Basco, ~
##  7  2013 C                      87.3 JULY      3.68         30.3 CAGAY~ Basco, ~
##  8  2013 C                      87.3 AUGUST    5.82         29.5 CAGAY~ Basco, ~
##  9  2013 C                      87.3 SEPTE~    9.92         29.1 CAGAY~ Basco, ~
## 10  2013 C                      87.3 OCTOB~   18.4          28.3 CAGAY~ Basco, ~
## # i 1,574 more rows
```

The purpose of dividing the new saved dataset is to conduct a separate data pre-processing for each synoptic station. Conducting a separate pre-processing method that works for one synoptic station may not be suitable for another, because each synoptic stations is located in a unique geographical area with varying weather patterns, altitude, humidity and temperature ranges.

DATA CLEANING FOR SYNOPTIC STATION A

Estimating the missing values for synoptic station A

```r
stationA <- read_excel("stationA.xlsx")
missingValA <- colMeans(is.na(stationA)) * 100
print(missingValA)
```

```
##             YEAR SYNOPTIC STATION          HUMIDITY             MONTH
##         0.000000         0.000000         3.030303         0.000000
##         RAINFALL      TEMPERATURE           REGION          LOCATION
##         0.000000         2.272727         0.000000         0.000000
```

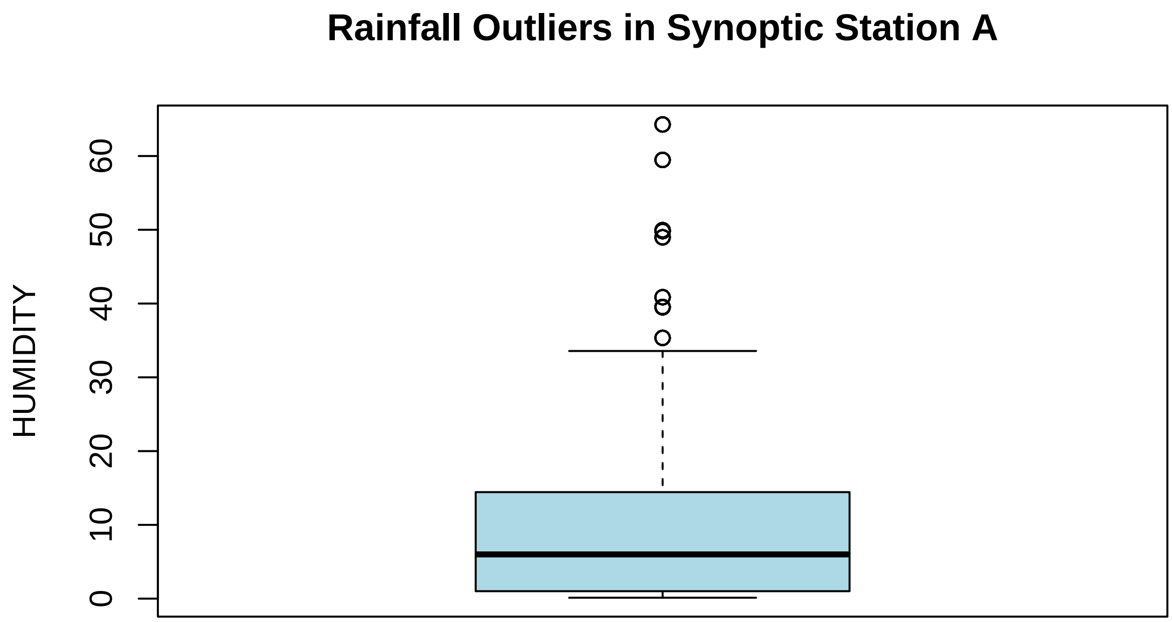
We will going to use Forward Fill approach to supply missing data, since weather data is time-dependent, the last known value is likely a better estimate:

```
stationA$HUMIDITY <- na.locf(stationA$HUMIDITY)
stationA$TEMPERATURE <- na.locf(stationA$TEMPERATURE)
View(stationA)
print(stationA)
```

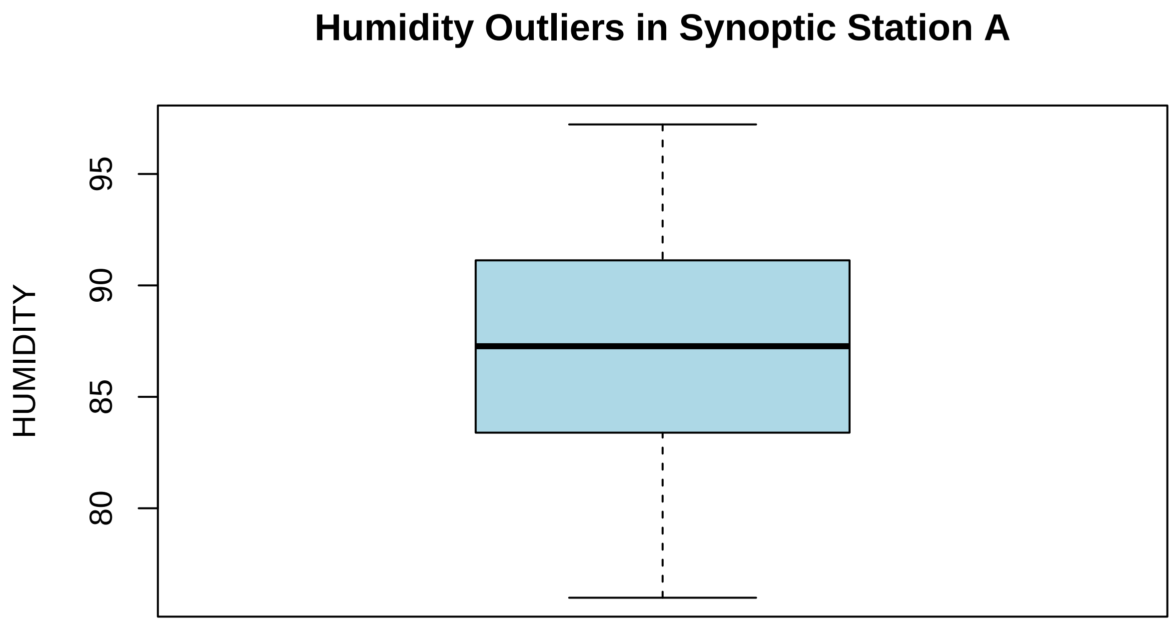
```
## # A tibble: 1,584 x 8
##     YEAR `SYNOPTIC STATION` HUMIDITY MONTH  RAINFALL TEMPERATURE REGION LOCATION
##    <dbl> <chr>                 <dbl> <chr>     <dbl>       <dbl> <chr>  <chr>
##  1  2013 A                      87.3 JANUA~    0.493        18.3 CAR    Baguio ~
##  2  2013 A                      87.3 FEBRU~    1.08         19.8 CAR    Baguio ~
##  3  2013 A                      87.3 MARCH     2.18         20.4 CAR    Baguio ~
##  4  2013 A                      87.3 APRIL     2.47         21.6 CAR    Baguio ~
##  5  2013 A                      87.3 MAY      11.1          20.9 CAR    Baguio ~
##  6  2013 A                      87.3 JUNE      7.89         20.6 CAR    Baguio ~
##  7  2013 A                      87.3 JULY     12.0          20.1 CAR    Baguio ~
##  8  2013 A                      87.3 AUGUST   39.5          19.2 CAR    Baguio ~
##  9  2013 A                      87.3 SEPTE~   19.8          19.7 CAR    Baguio ~
## 10  2013 A                      87.3 OCTOB~    7.87         18.7 CAR    Baguio ~
## # i 1,574 more rows
```

Detecting outliers for synoptic station A.

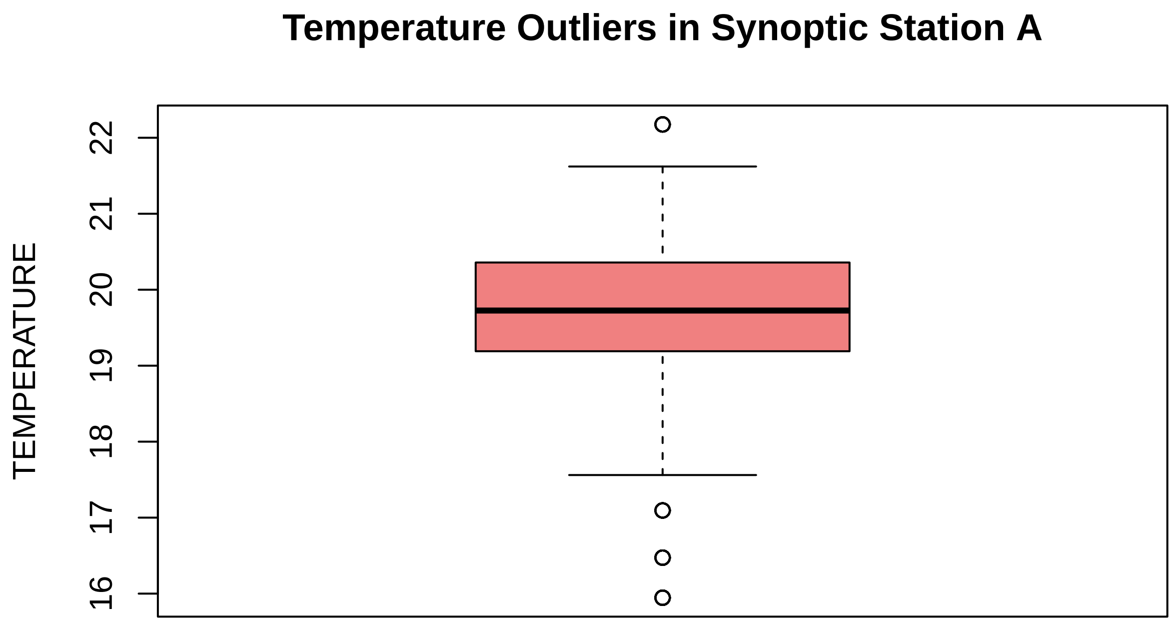
```
stationAoutliers <- function(data, col){
q1 <- quantile(data[[col]], 0.25, na.rm = TRUE)
q3 <- quantile(data[[col]], 0.75, na.rm = TRUE)
iqr <- q3-q1
lowbound <- q1 - 1.5 * iqr
upbound <- q3 + 1.5 * iqr
outliers <- data %>% filter((.data[[col]] < lowbound) | (.data[[col]] > upbound))
return(outliers)
}
rainOutliers <- stationAoutliers(stationA, "RAINFALL")
humOutliers <- stationAoutliers(stationA, "HUMIDITY")
tempOutliers <- stationAoutliers(stationA, "TEMPERATURE")

boxplot(stationA$RAINFALL, main="Rainfall Outliers in Synoptic Station A",
        ylab="HUMIDITY", col="lightblue")
```




```r
boxplot(stationA$HUMIDITY, main="Humidity Outliers in Synoptic Station A",
        ylab="HUMIDITY", col="lightblue")
```




```
boxplot(stationA$TEMPERATURE, main="Temperature Outliers in Synoptic Station A",
        ylab="TEMPERATURE", col="lightcoral")
```



DATA CLEANING FOR SYNOPTIC STATION B

Estimating the missing values for synoptic station B

```
stationB <- read_excel("stationB.xlsx")
missingValB <- colMeans(is.na(stationB)) * 100
print(missingValB)
```

```
##              YEAR SYNOPTIC STATION         HUMIDITY            MONTH
##          0.000000         0.000000         3.030303         0.000000
##          RAINFALL      TEMPERATURE           REGION         LOCATION
##          1.515152         3.030303         0.000000         0.000000
```

We will going to use Forward Fill approach to supply missing data, since weather data is time-dependent, the last known value is likely a better estimate:

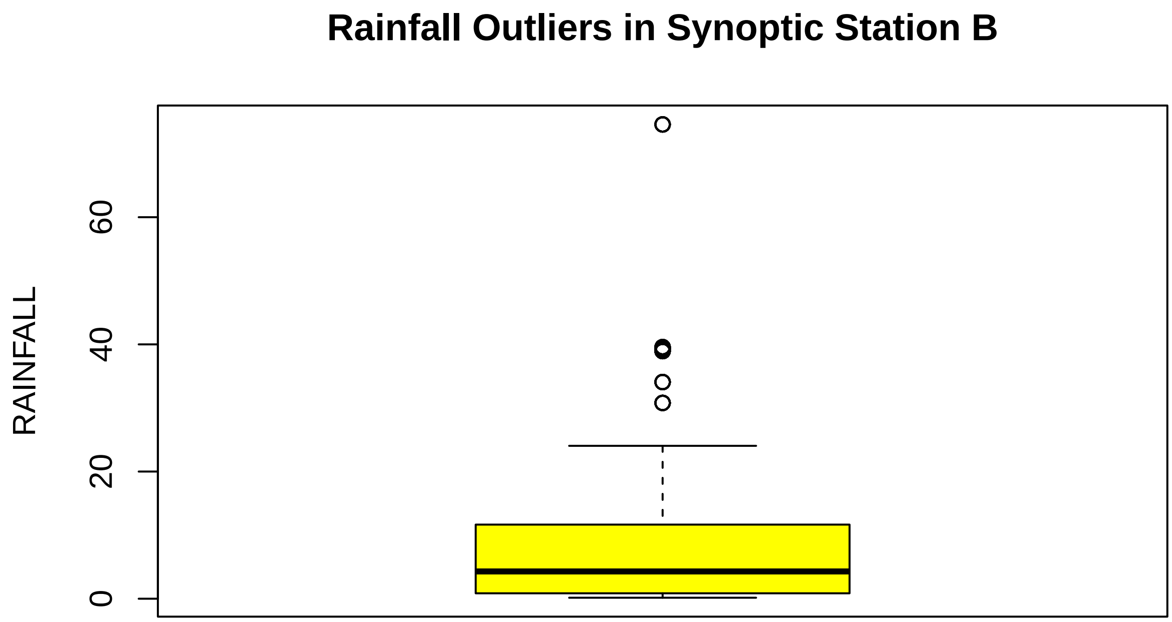
```
stationB$RAINFALL <- na.locf(stationB$RAINFALL)
stationB$HUMIDITY <- na.locf(stationB$HUMIDITY)
stationB$TEMPERATURE <- na.locf(stationB$TEMPERATURE)
View(stationB)
print(stationB)
```

```
## # A tibble: 1,584 x 8
```

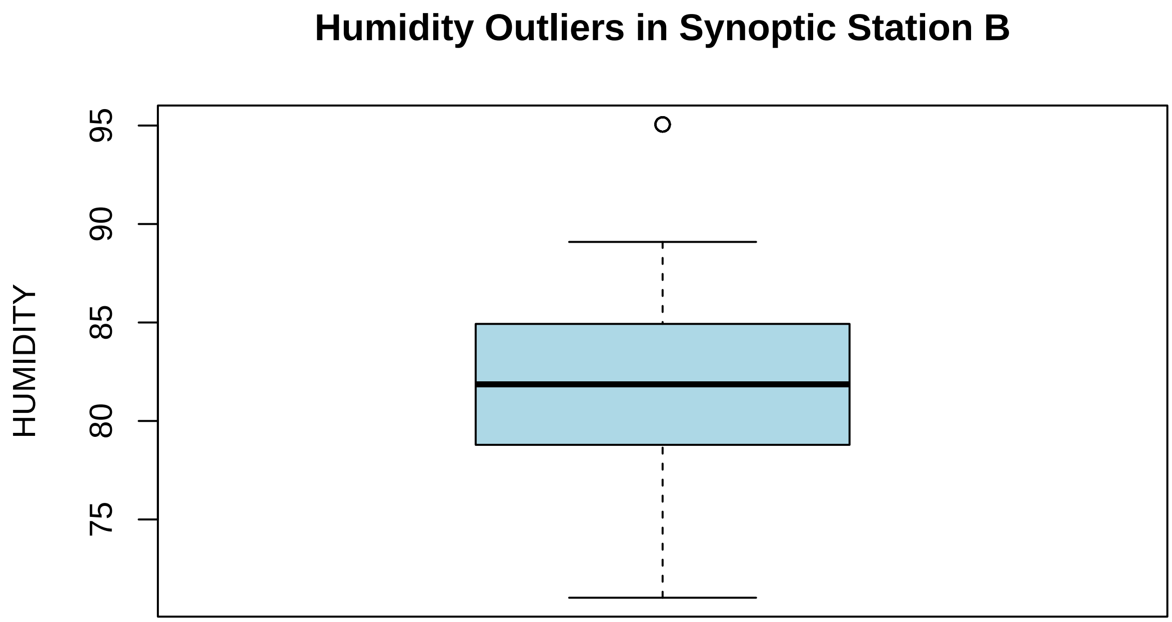
```
##     YEAR `SYNOPTIC STATION` HUMIDITY MONTH  RAINFALL TEMPERATURE REGION LOCATION
##    <dbl> <chr>                <dbl> <chr>     <dbl>       <dbl> <chr>  <chr>
##  1  2013 B                     82.0 JANUA~    0.484        26.4 ILOCOS Dagupan~
##  2  2013 B                     82.0 FEBRU~    0.199        27.0 ILOCOS Dagupan~
##  3  2013 B                     82.0 MARCH     3.39         28.7 ILOCOS Dagupan~
##  4  2013 B                     82.0 APRIL     3.78         30.6 ILOCOS Dagupan~
##  5  2013 B                     82.0 MAY       7.10         30.1 ILOCOS Dagupan~
##  6  2013 B                     82.0 JUNE      6.60         29.7 ILOCOS Dagupan~
##  7  2013 B                     82.0 JULY      8.24         28.8 ILOCOS Dagupan~
##  8  2013 B                     82.0 AUGUST   38.9          27.7 ILOCOS Dagupan~
##  9  2013 B                     82.0 SEPTE~   24.0          28.3 ILOCOS Dagupan~
## 10  2013 B                     82.0 OCTOB~    3.43         28.1 ILOCOS Dagupan~
## # i 1,574 more rows
```

Detecting outliers for synoptic station B.

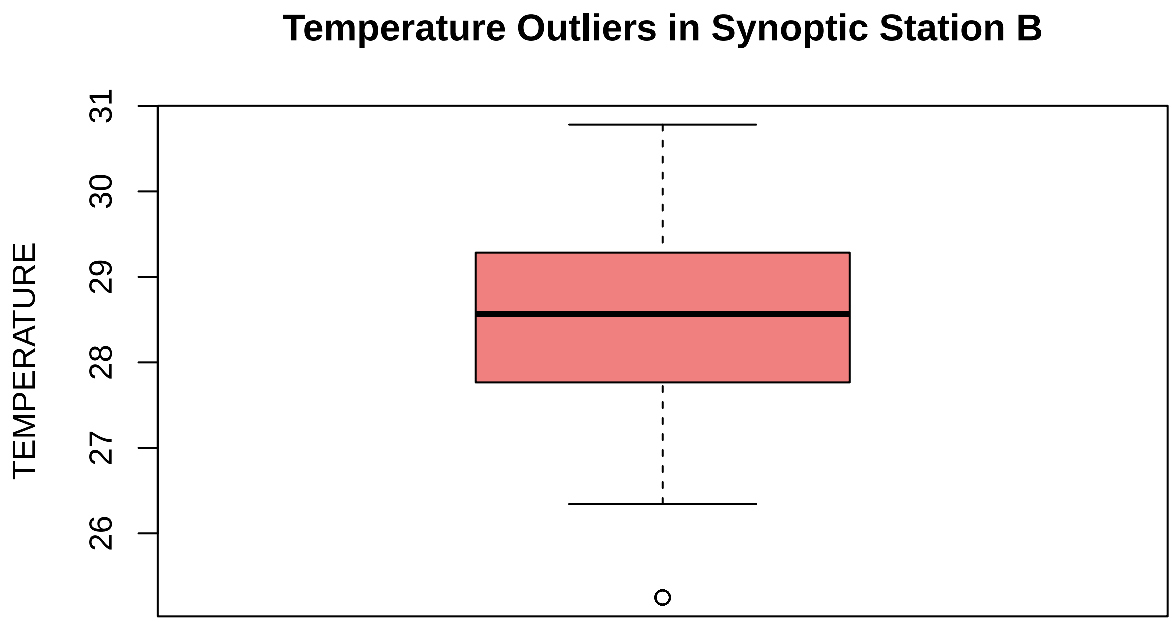
```r
stationBoutliers <- function(data, col){
q1 <- quantile(data[[col]], 0.25, na.rm = TRUE)
q3 <- quantile(data[[col]], 0.75, na.rm = TRUE)
iqr <- q3-q1
lowbound <- q1 - 1.5 * iqr
upbound <- q3 + 1.5 * iqr
outliersB <- data %>% filter((.data[[col]] < lowbound) | (.data[[col]] > upbound))
return(outliersB)
}
rainOutliersB <- stationBoutliers(stationB, "RAINFALL")
humOutliersB <- stationBoutliers(stationB, "HUMIDITY")
tempOutliersB <- stationBoutliers(stationB, "TEMPERATURE")

boxplot(stationB$RAINFALL, main="Rainfall Outliers in Synoptic Station B",
        ylab="RAINFALL", col="yellow")
```




```r
boxplot(stationB$HUMIDITY, main="Humidity Outliers in Synoptic Station B",
        ylab="HUMIDITY", col="lightblue")
```




```r
boxplot(stationB$TEMPERATURE, main="Temperature Outliers in Synoptic Station B",
        ylab="TEMPERATURE", col="lightcoral")
```



DATA CLEANING FOR SYNOPTIC STATION C

Estimating the missing values for synoptic station C

```
stationC <- read_excel("stationC.xlsx")
missingValC <- colMeans(is.na(stationC)) * 100
print(missingValC)
```

```
##              YEAR SYNOPTIC STATION         HUMIDITY            MONTH
##          0.000000         0.000000         3.787879         0.000000
##          RAINFALL      TEMPERATURE           REGION         LOCATION
##          3.787879         5.303030         0.000000         0.000000
```

We will going to use Forward Fill approach to supply missing data, since weather data is time-dependent, the last known value is likely a better estimate:

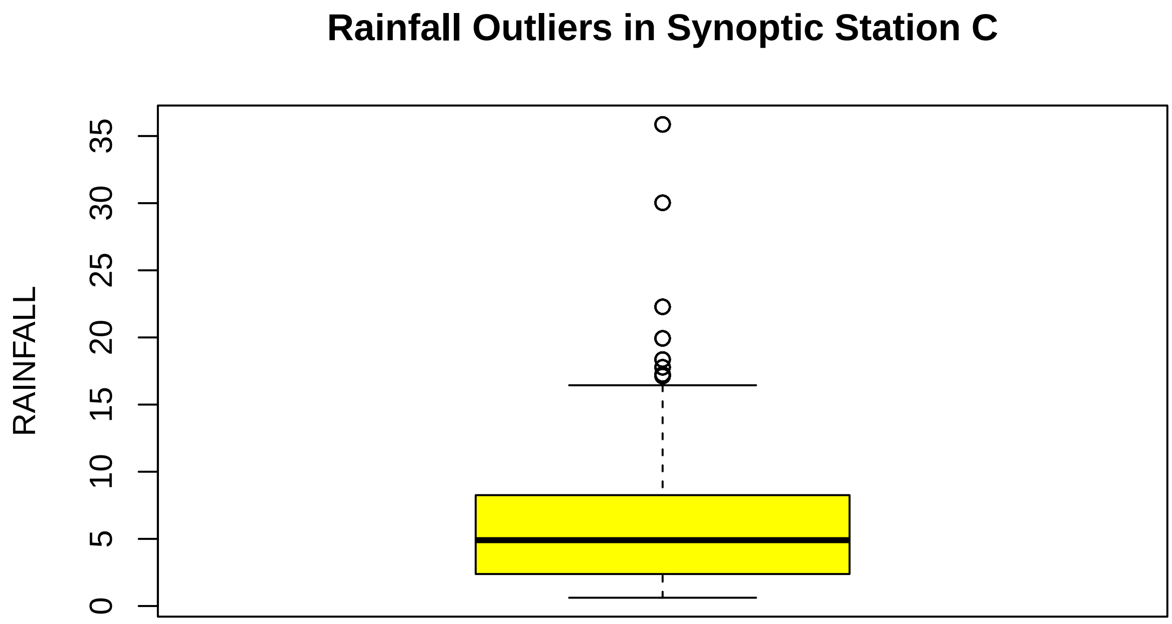
```
stationC$RAINFALL <- na.locf(stationC$RAINFALL)
stationC$HUMIDITY <- na.locf(stationC$HUMIDITY)
stationC$TEMPERATURE <- na.locf(stationC$TEMPERATURE)
View(stationC)
print(stationC)
```

```
## # A tibble: 1,584 x 8
```

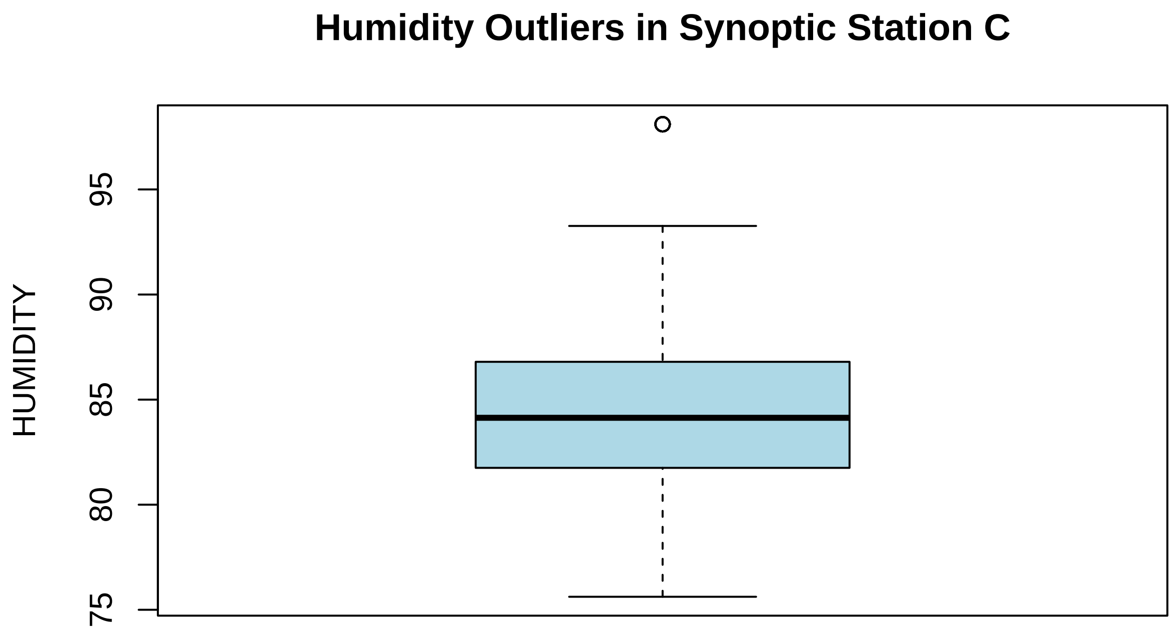
```
##     YEAR `SYNOPTIC STATION` HUMIDITY MONTH  RAINFALL TEMPERATURE REGION LOCATION
##    <dbl> <chr>                 <dbl> <chr>     <dbl>       <dbl> <chr>  <chr>
##  1  2013 C                      87.3 JANUA~    2.83         25.0 CAGAY~ Basco, ~
##  2  2013 C                      87.3 FEBRU~    1.24         26.8 CAGAY~ Basco, ~
##  3  2013 C                      87.3 MARCH     0.990        28.0 CAGAY~ Basco, ~
##  4  2013 C                      87.3 APRIL     0.969        29.2 CAGAY~ Basco, ~
##  5  2013 C                      87.3 MAY       2.67         29.6 CAGAY~ Basco, ~
##  6  2013 C                      87.3 JUNE      1.99         31.1 CAGAY~ Basco, ~
##  7  2013 C                      87.3 JULY      3.68         30.3 CAGAY~ Basco, ~
##  8  2013 C                      87.3 AUGUST    5.82         29.5 CAGAY~ Basco, ~
##  9  2013 C                      87.3 SEPTE~    9.92         29.1 CAGAY~ Basco, ~
## 10  2013 C                      87.3 OCTOB~   18.4          28.3 CAGAY~ Basco, ~
## # i 1,574 more rows
```

Detecting outliers for synoptic station C.

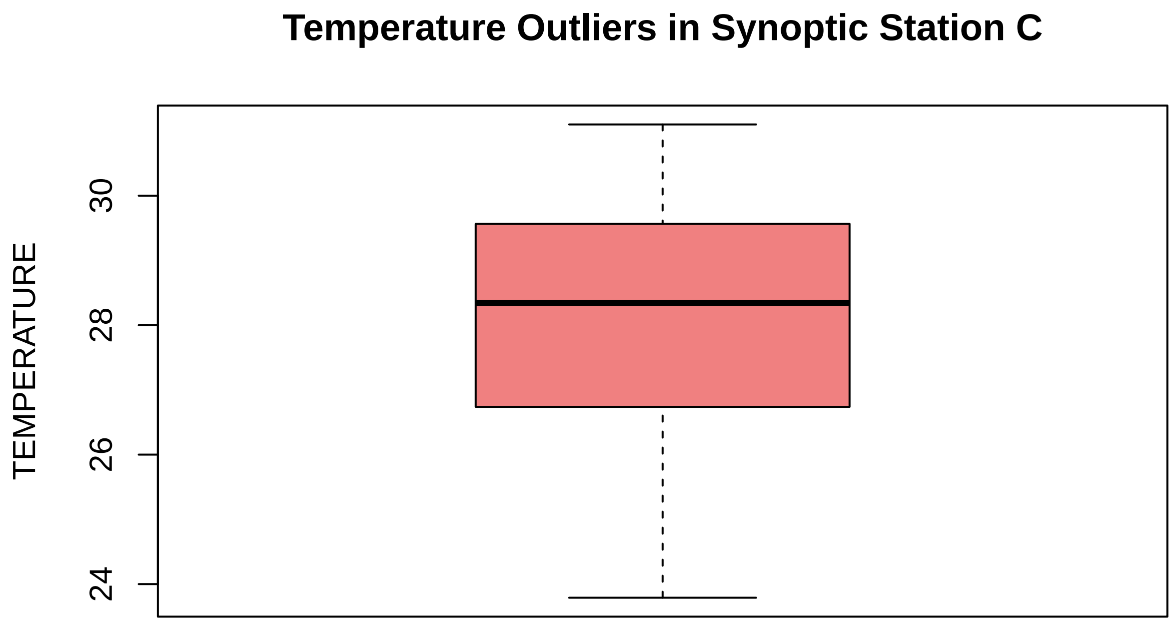
```
stationCoutliers <- function(data, col){
q1 <- quantile(data[[col]], 0.25, na.rm = TRUE)
q3 <- quantile(data[[col]], 0.75, na.rm = TRUE)
iqr <- q3-q1
lowbound <- q1 - 1.5 * iqr
upbound <- q3 + 1.5 * iqr
outliersC <- data %>% filter((.data[[col]] < lowbound) | (.data[[col]] > upbound))
return(outliersB)
}
rainOutliersC <- stationBoutliers(stationC, "RAINFALL")
humOutliersC <- stationBoutliers(stationC, "HUMIDITY")
tempOutliersC <- stationBoutliers(stationC, "TEMPERATURE")

boxplot(stationC$RAINFALL, main="Rainfall Outliers in Synoptic Station C",
        ylab="RAINFALL", col="yellow")
```




```r
boxplot(stationC$HUMIDITY, main="Humidity Outliers in Synoptic Station C",
        ylab="HUMIDITY", col="lightblue")
```



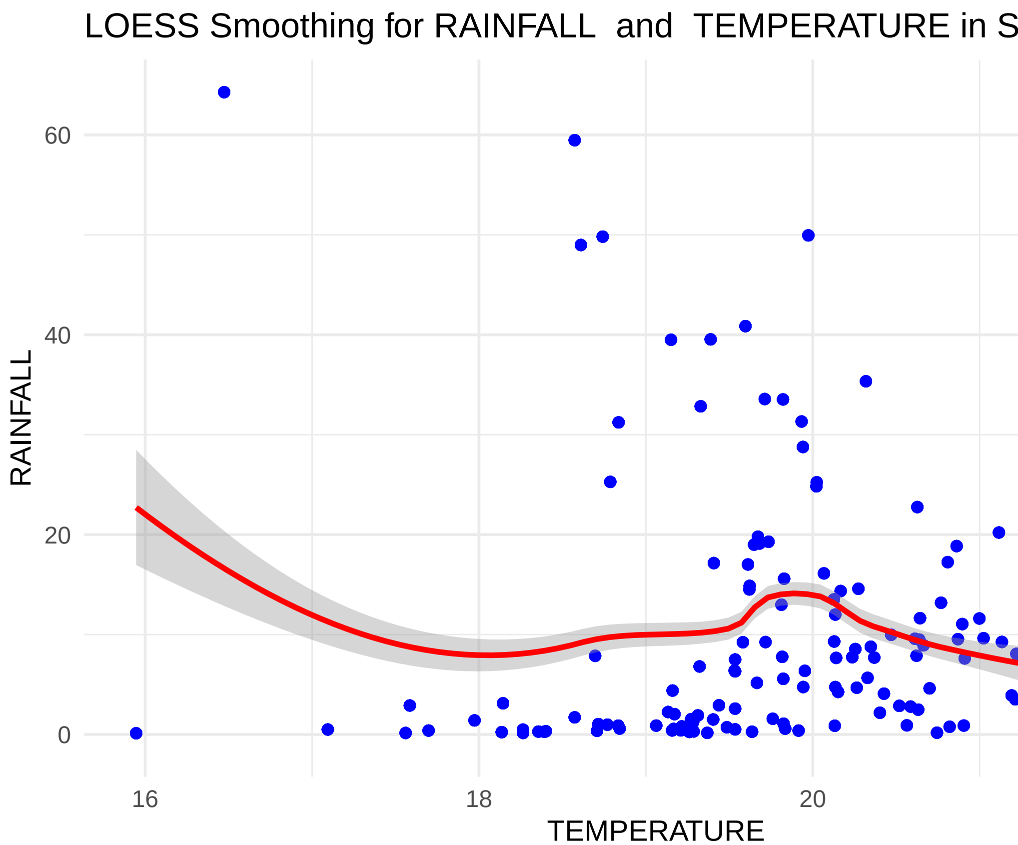

```r
boxplot(stationC$TEMPERATURE, main="Temperature Outliers in Synoptic Station C",
        ylab="TEMPERATURE", col="lightcoral")
```



Is it justifiable to completely remove the detected outliers [if there are any]? Can your group suggest a better approach aside from simply removing these outliers[if there are any]?

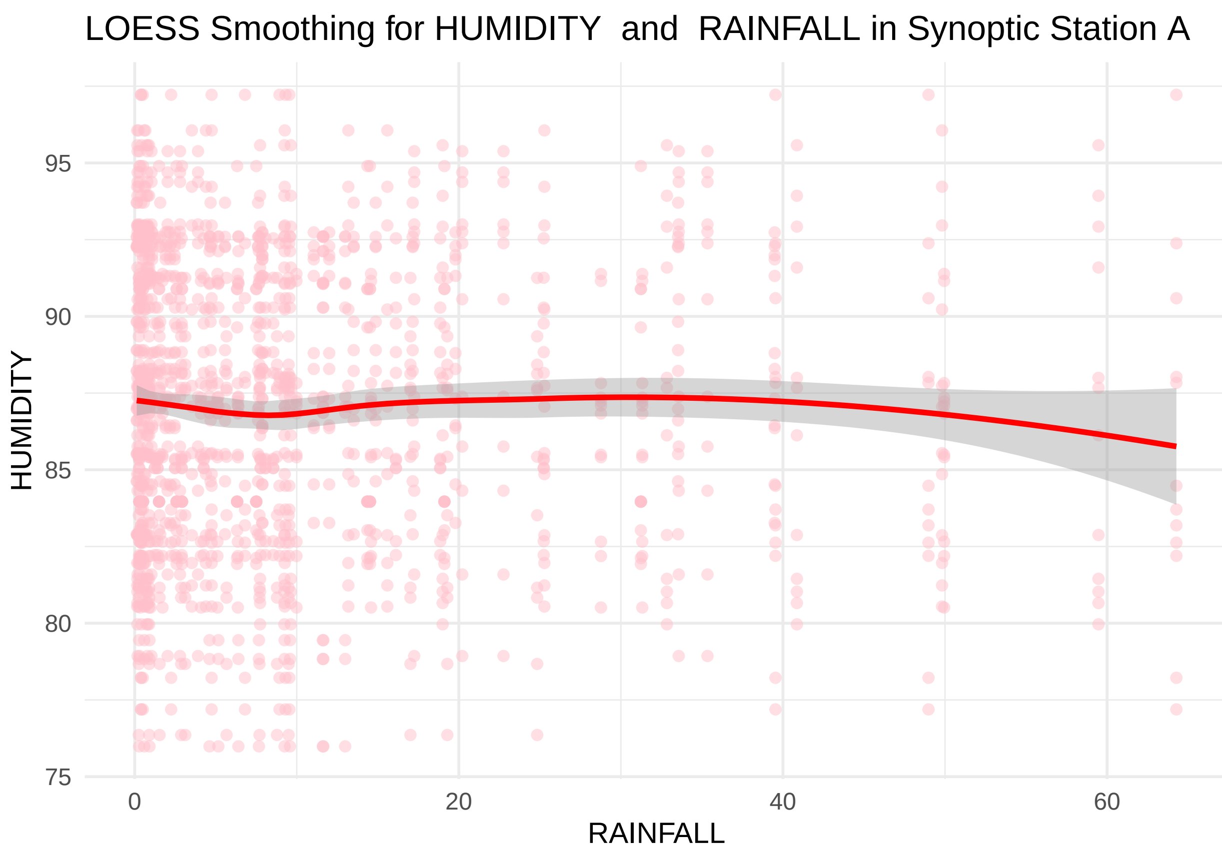
Answer: Best approach depends on why the outliers exist. If the outliers are valid weather events, we can keep them. If the outliers are due to sensor errors, we can replace them with imputed values or remove them. If using for modeling, we can apply transformations instead of outright deletion.

ESTIMATED LOESS GRAPH

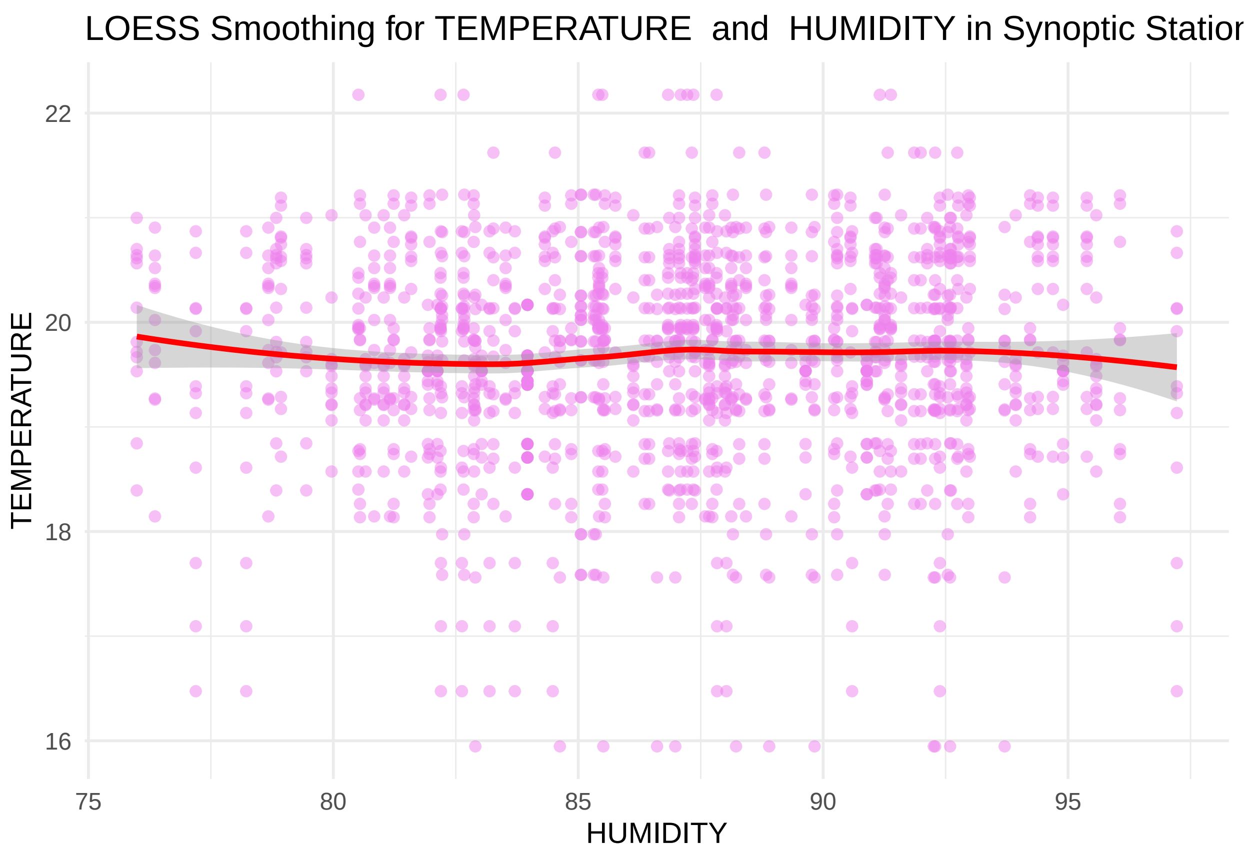


I. Synoptic Station A LOESS Graph

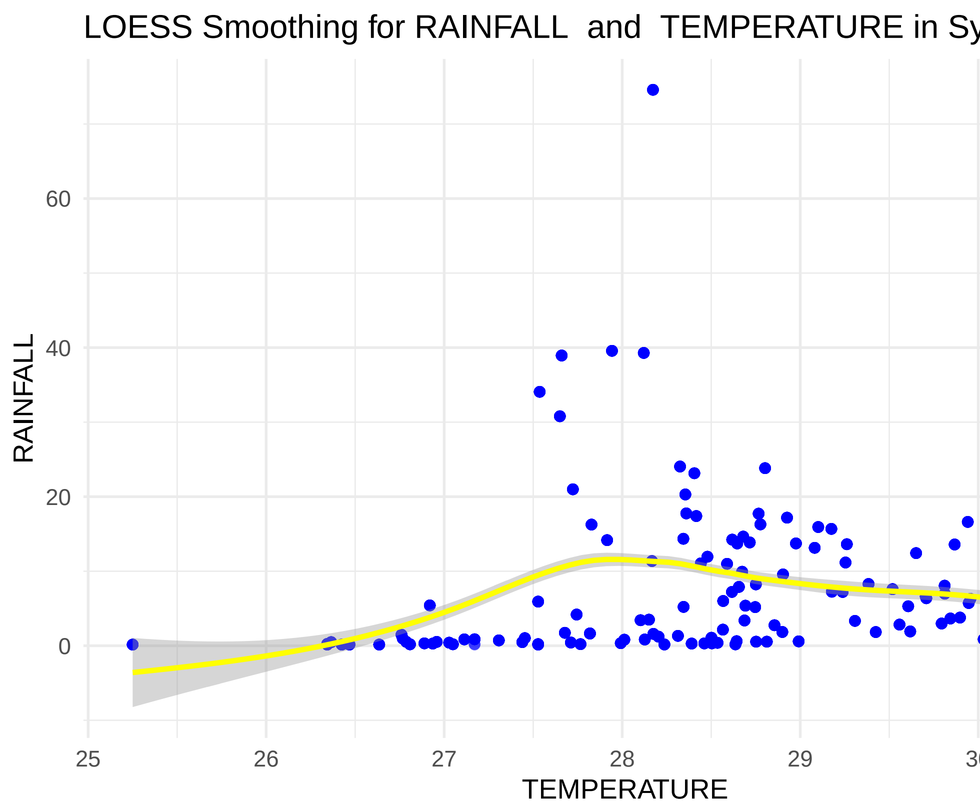
```
##
## The LOESS smoothing plot shows the relationship between temperature and rainfall, with a non-linear t
```




```
##
## The LOESS graph shows the relationship between humidity and rainfall in Synoptic Station A, where hum
```

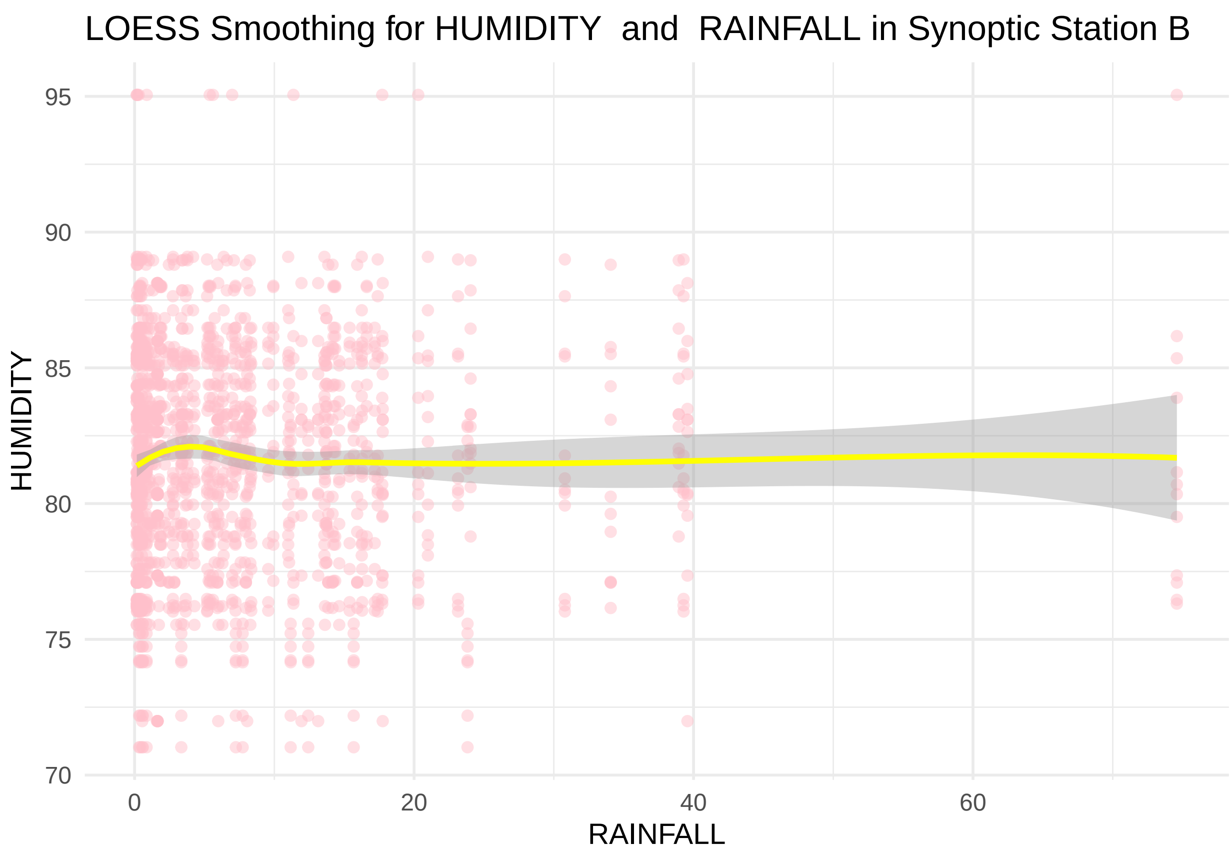



```
##
## The LOESS graph shows a weak relationship between temperature and humidity in Synoptic Station A, wi
```

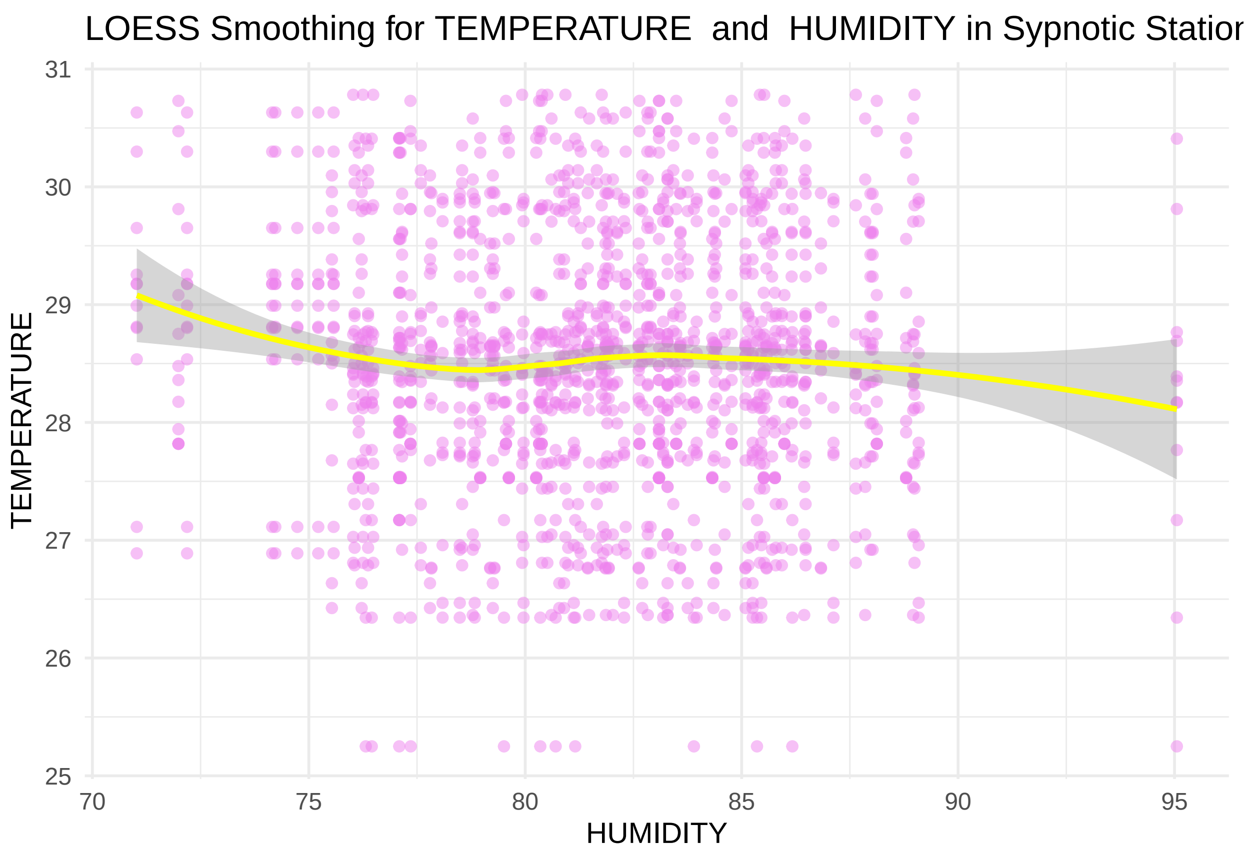


II. Synoptic Station B LOESS Graph

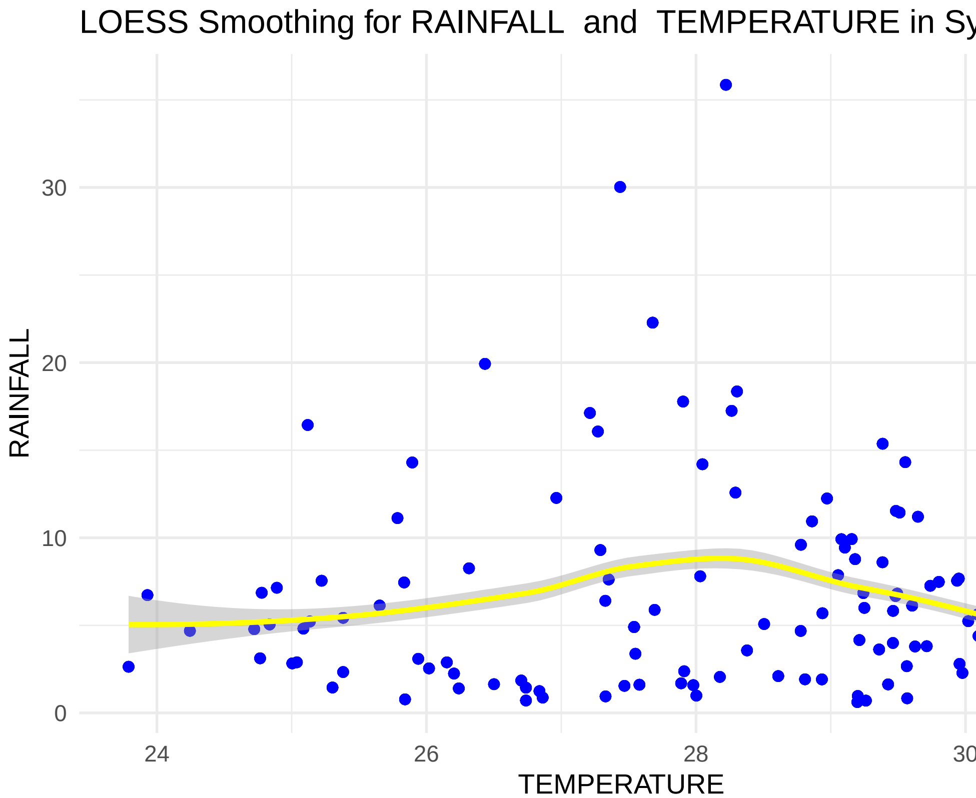
```
##
## The LOESS graph illustrates the relationship between rainfall and temperature in Synoptic Station B,
```




```
##
## The LOESS graph shows the relationship between humidity and rainfall in Synoptic Station B, indicati
```

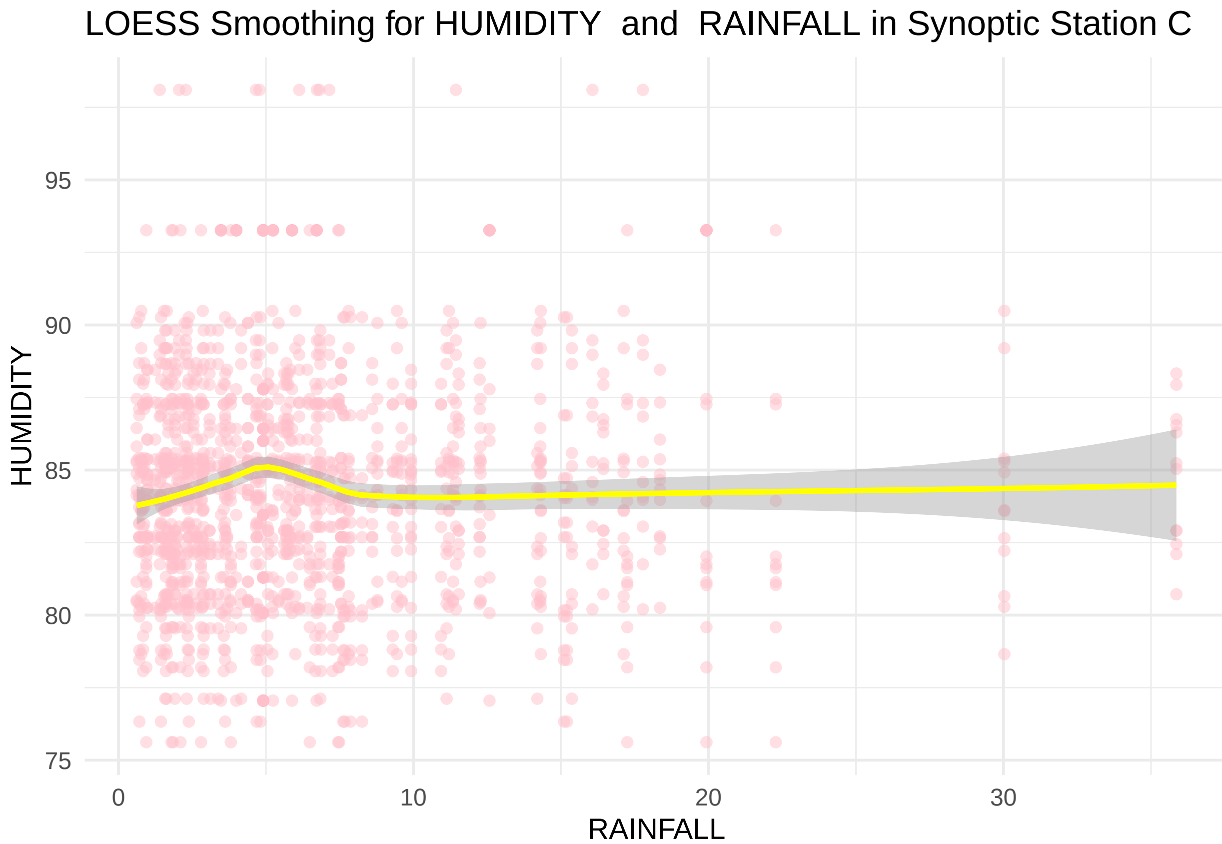



```
##
## The LOESS graph illustrates the relationship between temperature and humidity, showing a slight down
```

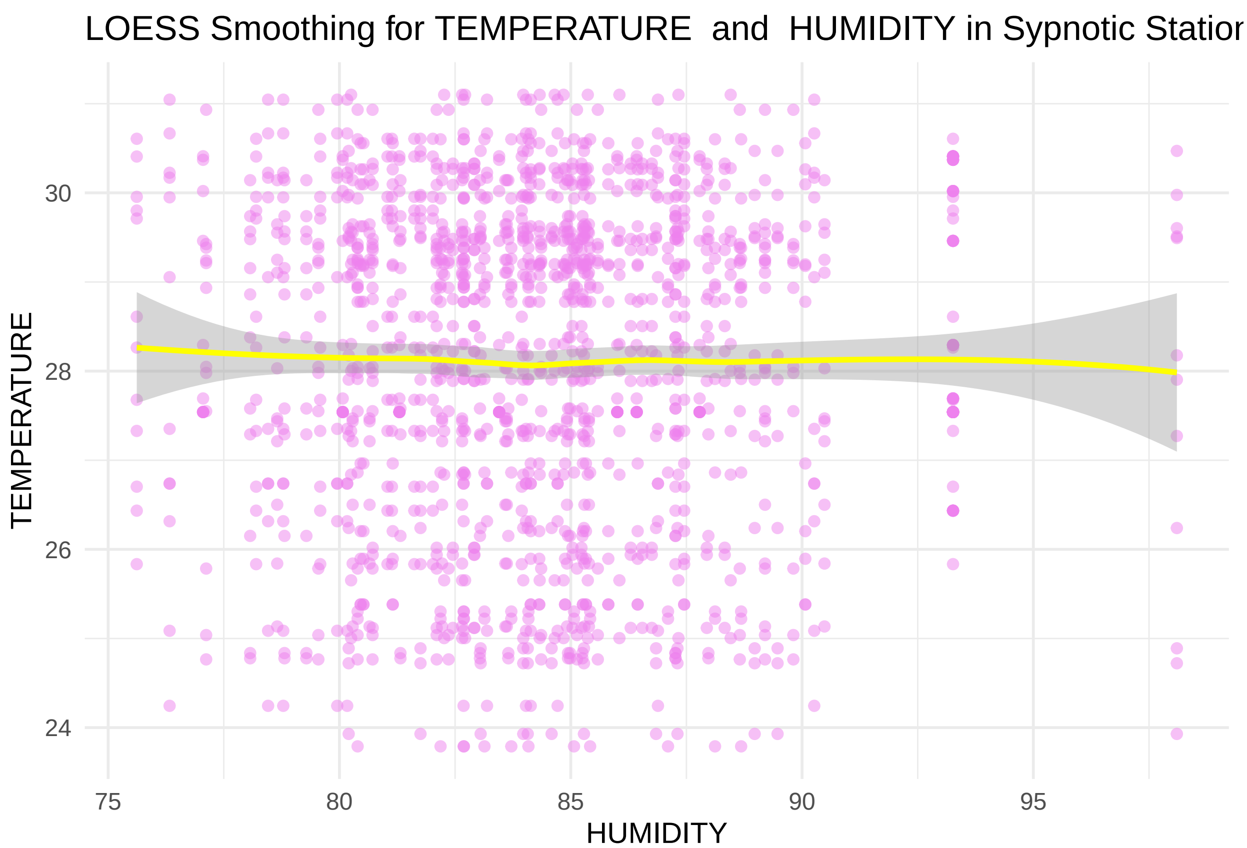


III. Synoptic Station C LOESS Graph

```
##
## The LOESS graph visualizes the relationship between rainfall and temperature, showing an initial inc
```




```
##
## This LOESS smoothing plot illustrates the relationship between humidity and rainfall in Synoptic Sta
```




```
##
## This LOESS smoothing plot visualizes the relationship between temperature and humidity in a synoptic
```

DATA TRANSFORMATION

I. Synoptic Station A

```r
zscorerain <- (stationA$RAINFALL - mean(stationA$RAINFALL))/sd(stationA$RAINFALL)
zscorehum <- (stationA$HUMIDITY - mean(stationA$HUMIDITY))/sd(stationA$HUMIDITY)
zscoretemp <- (stationA$TEMPERATURE - mean(stationA$TEMPERATURE))/sd(stationA$TEMPERATURE)

stationA_zscore <- data.frame(RAINFALL_Z = zscorerain, HUMIDITY_Z = zscorehum, TEMPERATURE_Z = zscoreter
stationA_transformed <- bind_cols(stationA, stationA_zscore)
print(stationA_transformed)
```

```
## # A tibble: 1,584 x 11
##     YEAR ‘SYNOPTIC STATION‘ HUMIDITY MONTH  RAINFALL TEMPERATURE REGION LOCATION
##    <dbl> <chr>                 <dbl> <chr>     <dbl>       <dbl> <chr>  <chr>
##  1  2013 A                      87.3 JANUA~    0.493        18.3 CAR    Baguio ~
##  2  2013 A                      87.3 FEBRU~    1.08         19.8 CAR    Baguio ~
##  3  2013 A                      87.3 MARCH     2.18         20.4 CAR    Baguio ~
##  4  2013 A                      87.3 APRIL     2.47         21.6 CAR    Baguio ~
##  5  2013 A                      87.3 MAY      11.1          20.9 CAR    Baguio ~
##  6  2013 A                      87.3 JUNE      7.89         20.6 CAR    Baguio ~
```

```
## 7   2013 A                       87.3 JULY     12.0          20.1 CAR    Baguio ~
## 8   2013 A                       87.3 AUGUST   39.5          19.2 CAR    Baguio ~
## 9   2013 A                       87.3 SEPTE~   19.8          19.7 CAR    Baguio ~
## 10  2013 A                       87.3 OCTOB~    7.87         18.7 CAR    Baguio ~
## # i 1,574 more rows
## # i 3 more variables: RAINFALL_Z <dbl>, HUMIDITY_Z <dbl>, TEMPERATURE_Z <dbl>
```

II. Synoptic Station B

```
zscorerainB <- (stationB$RAINFALL - mean(stationB$RAINFALL))/sd(stationB$RAINFALL)
zscorehumB <- (stationB$HUMIDITY - mean(stationB$HUMIDITY))/sd(stationB$HUMIDITY)
zscoretempB <- (stationB$TEMPERATURE - mean(stationB$TEMPERATURE))/sd(stationB$TEMPERATURE)

stationB_zscore <- data.frame(RAINFALL_Z = zscorerainB, HUMIDITY_Z = zscorehumB, TEMPERATURE_Z = zscore
stationB_transformed <- bind_cols(stationB, stationB_zscore)
View(stationB_transformed)
print(stationB_transformed)
```

```
## # A tibble: 1,584 x 11
##      YEAR `SYNOPTIC STATION` HUMIDITY MONTH  RAINFALL TEMPERATURE REGION LOCATION
##     <dbl> <chr>                 <dbl> <chr>     <dbl>       <dbl> <chr>  <chr>
## 1   2013 B                       82.0 JANUA~    0.484        26.4 ILOCOS Dagupan~
## 2   2013 B                       82.0 FEBRU~    0.199        27.0 ILOCOS Dagupan~
## 3   2013 B                       82.0 MARCH     3.39         28.7 ILOCOS Dagupan~
## 4   2013 B                       82.0 APRIL     3.78         30.6 ILOCOS Dagupan~
## 5   2013 B                       82.0 MAY       7.10         30.1 ILOCOS Dagupan~
## 6   2013 B                       82.0 JUNE      6.60         29.7 ILOCOS Dagupan~
## 7   2013 B                       82.0 JULY      8.24         28.8 ILOCOS Dagupan~
## 8   2013 B                       82.0 AUGUST   38.9          27.7 ILOCOS Dagupan~
## 9   2013 B                       82.0 SEPTE~   24.0          28.3 ILOCOS Dagupan~
## 10  2013 B                       82.0 OCTOB~    3.43         28.1 ILOCOS Dagupan~
## # i 1,574 more rows
## # i 3 more variables: RAINFALL_Z <dbl>, HUMIDITY_Z <dbl>, TEMPERATURE_Z <dbl>
```

III. Synoptic Station C

```
zscorerainC <- (stationC$RAINFALL - mean(stationC$RAINFALL))/sd(stationC$RAINFALL)
zscorehumC <- (stationC$HUMIDITY - mean(stationC$HUMIDITY))/sd(stationC$HUMIDITY)
zscoretempC <- (stationC$TEMPERATURE - mean(stationC$TEMPERATURE))/sd(stationC$TEMPERATURE)

stationC_zscore <- data.frame(RAINFALL_Z = zscorerainC, HUMIDITY_Z = zscorehumC, TEMPERATURE_Z = zscore
stationC_transformed <- bind_cols(stationC, stationC_zscore)
View(stationC_transformed)
print(stationC_transformed)
```

```
## # A tibble: 1,584 x 11
##      YEAR `SYNOPTIC STATION` HUMIDITY MONTH  RAINFALL TEMPERATURE REGION LOCATION
##     <dbl> <chr>                 <dbl> <chr>     <dbl>       <dbl> <chr>  <chr>
## 1   2013 C                       87.3 JANUA~    2.83         25.0 CAGAY~ Basco, ~
## 2   2013 C                       87.3 FEBRU~    1.24         26.8 CAGAY~ Basco, ~
## 3   2013 C                       87.3 MARCH     0.990        28.0 CAGAY~ Basco, ~
## 4   2013 C                       87.3 APRIL     0.969        29.2 CAGAY~ Basco, ~
```

```
## 5  2013 C                        87.3 MAY       2.67         29.6 CAGAY~ Basco, ~
## 6  2013 C                        87.3 JUNE      1.99         31.1 CAGAY~ Basco, ~
## 7  2013 C                        87.3 JULY      3.68         30.3 CAGAY~ Basco, ~
## 8  2013 C                        87.3 AUGUST    5.82         29.5 CAGAY~ Basco, ~
## 9  2013 C                        87.3 SEPTE~    9.92         29.1 CAGAY~ Basco, ~
## 10 2013 C                        87.3 OCTOB~   18.4          28.3 CAGAY~ Basco, ~
## # i 1,574 more rows
## # i 3 more variables: RAINFALL_Z <dbl>, HUMIDITY_Z <dbl>, TEMPERATURE_Z <dbl>
```

We use Z-score standardization because it effectively handles different measurement scales and minimizes outlier influence.We used dataframe to bind the original and transformed data.

##BINNING

I. Binning for Sypnotic Station A

RAINFALL

```
bins <- 3
binrainA <- stationA$RAINFALL <- cut(stationA$RAINFALL, bins, include.lowest = TRUE, labels =c("Low","Me
#print(binrainA)
```

HUMIDITY

```
binhumA <- stationA$HUMIDITY <- cut(stationA$HUMIDITY, bins, include.lowest = TRUE, labels =c("Low","Med
#print(binhumA)
```

TEMPERATURE

```
bintempA <- stationA$TEMPERATURE <- cut(stationA$TEMPERATURE, bins, include.lowest = TRUE, labels =c("Lo
#print(bintempA)
```

II. Binning for Sypnotic Station B

RAINFALL

```
bins <- 3
binrainB <- stationB$RAINFALL <- cut(stationB$RAINFALL, bins, include.lowest = TRUE, labels =c("Low","Me
#print(binrainB)
```

HUMIDITY

```
binhumB <- stationB$HUMIDITY <- cut(stationB$HUMIDITY, bins, include.lowest = TRUE, labels =c("Low","Med
#print(binhumB)
```

TEMPERATURE

```
bintempB <- stationB$TEMPERATURE <- cut(stationB$TEMPERATURE, bins, include.lowest = TRUE, labels =c("Lo
#print(bintempB)
```

III. Binning for Sypnotic Station C

RAINFALL

```
bins <- 3
binrainC <- stationC$RAINFALL <- cut(stationC$RAINFALL, bins, include.lowest = TRUE, labels =c("Low","Me
#print(binrainC)
```

HUMIDITY

```
binhumC <- stationC$HUMIDITY <- cut(stationC$HUMIDITY, bins, include.lowest = TRUE, labels =c("Low","Med
#print(binhumC)
```

TEMPERATURE

```
bintempC <- stationC$TEMPERATURE <- cut(stationC$TEMPERATURE, bins, include.lowest = TRUE, labels =c("Lo
#print(bintempC)
```

Final preprocessed dataset for each synoptic station.

```
write.xlsx(stationA_transformed, "FinalStationA.xlsx")
write.xlsx(stationB_transformed, "FinalStationB.xlsx")
write.xlsx(stationC_transformed, "FinalStationc.xlsx")
```

Significant contributions of each group member

C.J. ODATO

LEADER Oversees the entire project, ensuring that every task meets all the requirements and completed. Coordinates team meetings and assigns responsibilities. Ensures smooth communication between programmers and researchers. Reviews final outputs before submission.

Ram Reniel Canido | Jan Iris Oiga

PROGRAMMERS

Develop and implement the code for data processing, cleaning, and analysis. Write scripts for data transformation, outlier detection, and visualization. Debug and optimize code for efficiency.

Jeshua Lexis Labio | Virgilio Salcedo II

RESEARCHERS

Analyze and interpret relevant literature, methodologies, and datasets. Conduct background research on data cleaning, outlier detection, and transformation techniques. Justify the chosen methodologies