MiniProject

2023-12-02

## Team :

* Rohith Ganni
* Pranthi Cavuturu
* Akhil Addepalli

## Given Information:

Dataset Source : <https://archive.ics.uci.edu/ml/datasets/Bias+correction+of+numerical+> prediction+model+temperature+forecast

Target Variable to Predict : Next\_Tmax

#### **(a, b, c) Preprocessing, Splitting, Initial Model**

Summary of the dataset is as follows:

library(readr)

## Warning: package 'readr' was built under R version 4.3.2

data <- read\_csv("Bias\_correction\_ucl.csv")

## Rows: 7752 Columns: 25  
## ── Column specification ────────────────────────────────────────────────────────  
## Delimiter: ","  
## dbl (24): station, Present\_Tmax, Present\_Tmin, LDAPS\_RHmin, LDAPS\_RHmax, LD...  
## date (1): Date  
##   
## ℹ Use `spec()` to retrieve the full column specification for this data.  
## ℹ Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

summary(data)

## station Date Present\_Tmax Present\_Tmin   
## Min. : 1 Min. :2013-06-30 Min. :20.00 Min. :11.30   
## 1st Qu.: 7 1st Qu.:2014-07-15 1st Qu.:27.80 1st Qu.:21.70   
## Median :13 Median :2015-07-30 Median :29.90 Median :23.40   
## Mean :13 Mean :2015-07-30 Mean :29.77 Mean :23.23   
## 3rd Qu.:19 3rd Qu.:2016-08-15 3rd Qu.:32.00 3rd Qu.:24.90   
## Max. :25 Max. :2017-08-30 Max. :37.60 Max. :29.90   
## NA's :2 NA's :2 NA's :70 NA's :70   
## LDAPS\_RHmin LDAPS\_RHmax LDAPS\_Tmax\_lapse LDAPS\_Tmin\_lapse  
## Min. :19.79 Min. : 58.94 Min. :17.62 Min. :14.27   
## 1st Qu.:45.96 1st Qu.: 84.22 1st Qu.:27.67 1st Qu.:22.09   
## Median :55.04 Median : 89.79 Median :29.70 Median :23.76   
## Mean :56.76 Mean : 88.37 Mean :29.61 Mean :23.51   
## 3rd Qu.:67.19 3rd Qu.: 93.74 3rd Qu.:31.71 3rd Qu.:25.15   
## Max. :98.52 Max. :100.00 Max. :38.54 Max. :29.62   
## NA's :75 NA's :75 NA's :75 NA's :75   
## LDAPS\_WS LDAPS\_LH LDAPS\_CC1 LDAPS\_CC2   
## Min. : 2.883 Min. :-13.60 Min. :0.0000 Min. :0.0000   
## 1st Qu.: 5.679 1st Qu.: 37.27 1st Qu.:0.1467 1st Qu.:0.1406   
## Median : 6.547 Median : 56.87 Median :0.3157 Median :0.3124   
## Mean : 7.098 Mean : 62.51 Mean :0.3688 Mean :0.3561   
## 3rd Qu.: 8.032 3rd Qu.: 84.22 3rd Qu.:0.5755 3rd Qu.:0.5587   
## Max. :21.858 Max. :213.41 Max. :0.9673 Max. :0.9684   
## NA's :75 NA's :75 NA's :75 NA's :75   
## LDAPS\_CC3 LDAPS\_CC4 LDAPS\_PPT1 LDAPS\_PPT2   
## Min. :0.0000 Min. :0.00000 Min. : 0.00000 Min. : 0.00000   
## 1st Qu.:0.1014 1st Qu.:0.08153 1st Qu.: 0.00000 1st Qu.: 0.00000   
## Median :0.2626 Median :0.22766 Median : 0.00000 Median : 0.00000   
## Mean :0.3184 Mean :0.29919 Mean : 0.59199 Mean : 0.48500   
## 3rd Qu.:0.4967 3rd Qu.:0.49949 3rd Qu.: 0.05252 3rd Qu.: 0.01836   
## Max. :0.9838 Max. :0.97471 Max. :23.70154 Max. :21.62166   
## NA's :75 NA's :75 NA's :75 NA's :75   
## LDAPS\_PPT3 LDAPS\_PPT4 lat lon   
## Min. : 0.0000 Min. : 0.00000 Min. :37.46 Min. :126.8   
## 1st Qu.: 0.0000 1st Qu.: 0.00000 1st Qu.:37.51 1st Qu.:126.9   
## Median : 0.0000 Median : 0.00000 Median :37.55 Median :127.0   
## Mean : 0.2782 Mean : 0.26941 Mean :37.54 Mean :127.0   
## 3rd Qu.: 0.0079 3rd Qu.: 0.00004 3rd Qu.:37.58 3rd Qu.:127.0   
## Max. :15.8412 Max. :16.65547 Max. :37.65 Max. :127.1   
## NA's :75 NA's :75   
## DEM Slope Solar radiation Next\_Tmax   
## Min. : 12.37 Min. :0.09847 Min. :4330 Min. :17.40   
## 1st Qu.: 28.70 1st Qu.:0.27130 1st Qu.:4999 1st Qu.:28.20   
## Median : 45.72 Median :0.61800 Median :5436 Median :30.50   
## Mean : 61.87 Mean :1.25705 Mean :5342 Mean :30.27   
## 3rd Qu.: 59.83 3rd Qu.:1.76780 3rd Qu.:5728 3rd Qu.:32.60   
## Max. :212.34 Max. :5.17823 Max. :5993 Max. :38.90   
## NA's :27   
## Next\_Tmin   
## Min. :11.30   
## 1st Qu.:21.30   
## Median :23.10   
## Mean :22.93   
## 3rd Qu.:24.60   
## Max. :29.80   
## NA's :27

Moving on with three different approaches of handling NULL Values:

1. Imputing the missing values with the mean values of the variables
2. Imputing the missing values with the median values of the variables
3. Omitting the Null Values

Common step in all the above methods is to remove the NULL values in the ‘date’ column

library(dplyr)

## Warning: package 'dplyr' was built under R version 4.3.2

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

initial\_structure <- dim(data)  
  
data <- data %>% filter(!is.na(Date))  
  
cleaned\_structure <- dim(data)  
  
print(initial\_structure)

## [1] 7752 25

print(cleaned\_structure)

## [1] 7750 25

data\_med <- data  
data\_mean <- data

#### Approach 1 - Imputing with mean values

# Imputing missing values with mean for numeric columns  
numeric\_columns <- sapply(data\_mean, is.numeric)  
data\_mean[numeric\_columns] <- lapply(data\_mean[numeric\_columns], function(x) ifelse(is.na(x), mean(x, na.rm = TRUE), x))  
  
data\_mean$station <- as.factor(data\_mean$station)  
  
str(data\_mean)

## spc\_tbl\_ [7,750 × 25] (S3: spec\_tbl\_df/tbl\_df/tbl/data.frame)  
## $ station : Factor w/ 25 levels "1","2","3","4",..: 1 2 3 4 5 6 7 8 9 10 ...  
## $ Date : Date[1:7750], format: "2013-06-30" "2013-06-30" ...  
## $ Present\_Tmax : num [1:7750] 28.7 31.9 31.6 32 31.4 31.9 31.4 32.1 31.4 31.6 ...  
## $ Present\_Tmin : num [1:7750] 21.4 21.6 23.3 23.4 21.9 23.5 24.4 23.6 22 20.5 ...  
## $ LDAPS\_RHmin : num [1:7750] 58.3 52.3 48.7 58.2 56.2 ...  
## $ LDAPS\_RHmax : num [1:7750] 91.1 90.6 84 96.5 90.2 ...  
## $ LDAPS\_Tmax\_lapse: num [1:7750] 28.1 29.9 30.1 29.7 29.1 ...  
## $ LDAPS\_Tmin\_lapse: num [1:7750] 23 24 24.6 23.3 23.5 ...  
## $ LDAPS\_WS : num [1:7750] 6.82 5.69 6.14 5.65 5.74 ...  
## $ LDAPS\_LH : num [1:7750] 69.5 51.9 20.6 65.7 108 ...  
## $ LDAPS\_CC1 : num [1:7750] 0.234 0.226 0.209 0.216 0.151 ...  
## $ LDAPS\_CC2 : num [1:7750] 0.204 0.252 0.257 0.226 0.25 ...  
## $ LDAPS\_CC3 : num [1:7750] 0.162 0.159 0.204 0.161 0.179 ...  
## $ LDAPS\_CC4 : num [1:7750] 0.131 0.128 0.142 0.134 0.17 ...  
## $ LDAPS\_PPT1 : num [1:7750] 0 0 0 0 0 0 0 0 0 0 ...  
## $ LDAPS\_PPT2 : num [1:7750] 0 0 0 0 0 0 0 0 0 0 ...  
## $ LDAPS\_PPT3 : num [1:7750] 0 0 0 0 0 0 0 0 0 0 ...  
## $ LDAPS\_PPT4 : num [1:7750] 0 0 0 0 0 0 0 0 0 0 ...  
## $ lat : num [1:7750] 37.6 37.6 37.6 37.6 37.6 ...  
## $ lon : num [1:7750] 127 127 127 127 127 ...  
## $ DEM : num [1:7750] 212.3 44.8 33.3 45.7 35 ...  
## $ Slope : num [1:7750] 2.785 0.514 0.266 2.535 0.505 ...  
## $ Solar radiation : num [1:7750] 5993 5869 5864 5857 5860 ...  
## $ Next\_Tmax : num [1:7750] 29.1 30.5 31.1 31.7 31.2 31.5 30.9 31.1 31.3 30.5 ...  
## $ Next\_Tmin : num [1:7750] 21.2 22.5 23.9 24.3 22.5 24 23.4 22.9 21.6 21 ...  
## - attr(\*, "spec")=  
## .. cols(  
## .. station = col\_double(),  
## .. Date = col\_date(format = ""),  
## .. Present\_Tmax = col\_double(),  
## .. Present\_Tmin = col\_double(),  
## .. LDAPS\_RHmin = col\_double(),  
## .. LDAPS\_RHmax = col\_double(),  
## .. LDAPS\_Tmax\_lapse = col\_double(),  
## .. LDAPS\_Tmin\_lapse = col\_double(),  
## .. LDAPS\_WS = col\_double(),  
## .. LDAPS\_LH = col\_double(),  
## .. LDAPS\_CC1 = col\_double(),  
## .. LDAPS\_CC2 = col\_double(),  
## .. LDAPS\_CC3 = col\_double(),  
## .. LDAPS\_CC4 = col\_double(),  
## .. LDAPS\_PPT1 = col\_double(),  
## .. LDAPS\_PPT2 = col\_double(),  
## .. LDAPS\_PPT3 = col\_double(),  
## .. LDAPS\_PPT4 = col\_double(),  
## .. lat = col\_double(),  
## .. lon = col\_double(),  
## .. DEM = col\_double(),  
## .. Slope = col\_double(),  
## .. `Solar radiation` = col\_double(),  
## .. Next\_Tmax = col\_double(),  
## .. Next\_Tmin = col\_double()  
## .. )  
## - attr(\*, "problems")=<externalptr>

After Imputing the Values with mean, we split this data set into train, validation and test data sets. Following is the number of rows for the entire data set, train, vaidation & test data sets respectively

data\_mean <- data\_mean[order(data\_mean$Date), ]  
  
train\_size\_mean <- round(nrow(data\_mean) \* 0.60)  
valid\_size\_mean <- round(nrow(data\_mean) \* 0.80)  
  
train\_data\_mean <- data\_mean[1:train\_size\_mean, ]  
valid\_data\_mean <- data\_mean[(train\_size\_mean + 1):valid\_size\_mean, ]  
test\_data\_mean <- data\_mean[(valid\_size\_mean + 1):nrow(data\_mean), ]  
  
nrow(data\_mean)

## [1] 7750

nrow(train\_data\_mean)

## [1] 4650

nrow(valid\_data\_mean)

## [1] 1550

nrow(test\_data\_mean)

## [1] 1550

Now we apply regression model and train it with the train dataset for mean and check it’s Evaluation Metric (Root Mean Squared Error [RMSE]) value for our Approach - 1

predictors <- setdiff(names(train\_data\_mean), c('Next\_Tmax', 'Date', 'station', 'Next\_Tmin'))  
train\_data\_subset\_mean <- train\_data\_mean[, c('Next\_Tmax', predictors)]  
  
model\_mean <- lm(Next\_Tmax ~ ., data = train\_data\_subset\_mean)  
valid\_data\_subset\_mean <- valid\_data\_mean[, predictors]  
predictions\_mean <- predict(model\_mean, newdata = valid\_data\_subset\_mean)  
  
rmse <- sqrt(mean((valid\_data\_mean$Next\_Tmax - predictions\_mean)^2))  
rmse

## [1] 1.644064

#### Approach 2 - Imputing with median values

Structure of the data set when imputed with median values is as follows:

# Imputing missing values with median for numeric columns  
numeric\_columns <- sapply(data\_med, is.numeric)  
data\_med[numeric\_columns] <- lapply(data\_med[numeric\_columns], function(x) ifelse(is.na(x), median(x, na.rm = TRUE), x))  
  
data\_med$station <- as.factor(data\_med$station)  
  
str(data\_med)

## spc\_tbl\_ [7,750 × 25] (S3: spec\_tbl\_df/tbl\_df/tbl/data.frame)  
## $ station : Factor w/ 25 levels "1","2","3","4",..: 1 2 3 4 5 6 7 8 9 10 ...  
## $ Date : Date[1:7750], format: "2013-06-30" "2013-06-30" ...  
## $ Present\_Tmax : num [1:7750] 28.7 31.9 31.6 32 31.4 31.9 31.4 32.1 31.4 31.6 ...  
## $ Present\_Tmin : num [1:7750] 21.4 21.6 23.3 23.4 21.9 23.5 24.4 23.6 22 20.5 ...  
## $ LDAPS\_RHmin : num [1:7750] 58.3 52.3 48.7 58.2 56.2 ...  
## $ LDAPS\_RHmax : num [1:7750] 91.1 90.6 84 96.5 90.2 ...  
## $ LDAPS\_Tmax\_lapse: num [1:7750] 28.1 29.9 30.1 29.7 29.1 ...  
## $ LDAPS\_Tmin\_lapse: num [1:7750] 23 24 24.6 23.3 23.5 ...  
## $ LDAPS\_WS : num [1:7750] 6.82 5.69 6.14 5.65 5.74 ...  
## $ LDAPS\_LH : num [1:7750] 69.5 51.9 20.6 65.7 108 ...  
## $ LDAPS\_CC1 : num [1:7750] 0.234 0.226 0.209 0.216 0.151 ...  
## $ LDAPS\_CC2 : num [1:7750] 0.204 0.252 0.257 0.226 0.25 ...  
## $ LDAPS\_CC3 : num [1:7750] 0.162 0.159 0.204 0.161 0.179 ...  
## $ LDAPS\_CC4 : num [1:7750] 0.131 0.128 0.142 0.134 0.17 ...  
## $ LDAPS\_PPT1 : num [1:7750] 0 0 0 0 0 0 0 0 0 0 ...  
## $ LDAPS\_PPT2 : num [1:7750] 0 0 0 0 0 0 0 0 0 0 ...  
## $ LDAPS\_PPT3 : num [1:7750] 0 0 0 0 0 0 0 0 0 0 ...  
## $ LDAPS\_PPT4 : num [1:7750] 0 0 0 0 0 0 0 0 0 0 ...  
## $ lat : num [1:7750] 37.6 37.6 37.6 37.6 37.6 ...  
## $ lon : num [1:7750] 127 127 127 127 127 ...  
## $ DEM : num [1:7750] 212.3 44.8 33.3 45.7 35 ...  
## $ Slope : num [1:7750] 2.785 0.514 0.266 2.535 0.505 ...  
## $ Solar radiation : num [1:7750] 5993 5869 5864 5857 5860 ...  
## $ Next\_Tmax : num [1:7750] 29.1 30.5 31.1 31.7 31.2 31.5 30.9 31.1 31.3 30.5 ...  
## $ Next\_Tmin : num [1:7750] 21.2 22.5 23.9 24.3 22.5 24 23.4 22.9 21.6 21 ...  
## - attr(\*, "spec")=  
## .. cols(  
## .. station = col\_double(),  
## .. Date = col\_date(format = ""),  
## .. Present\_Tmax = col\_double(),  
## .. Present\_Tmin = col\_double(),  
## .. LDAPS\_RHmin = col\_double(),  
## .. LDAPS\_RHmax = col\_double(),  
## .. LDAPS\_Tmax\_lapse = col\_double(),  
## .. LDAPS\_Tmin\_lapse = col\_double(),  
## .. LDAPS\_WS = col\_double(),  
## .. LDAPS\_LH = col\_double(),  
## .. LDAPS\_CC1 = col\_double(),  
## .. LDAPS\_CC2 = col\_double(),  
## .. LDAPS\_CC3 = col\_double(),  
## .. LDAPS\_CC4 = col\_double(),  
## .. LDAPS\_PPT1 = col\_double(),  
## .. LDAPS\_PPT2 = col\_double(),  
## .. LDAPS\_PPT3 = col\_double(),  
## .. LDAPS\_PPT4 = col\_double(),  
## .. lat = col\_double(),  
## .. lon = col\_double(),  
## .. DEM = col\_double(),  
## .. Slope = col\_double(),  
## .. `Solar radiation` = col\_double(),  
## .. Next\_Tmax = col\_double(),  
## .. Next\_Tmin = col\_double()  
## .. )  
## - attr(\*, "problems")=<externalptr>

After Imputing the Values with median, we split this data set into train, validation and test data sets. Following is the number of rows for the entire data set, train, vaidation & test data sets respectively

data\_med <- data\_med[order(data\_med$Date), ]  
  
train\_size\_med <- round(nrow(data\_med) \* 0.60)  
valid\_size\_med <- round(nrow(data\_med) \* 0.80)  
  
train\_data\_med <- data\_med[1:train\_size\_med, ]  
valid\_data\_med <- data\_med[(train\_size\_med + 1):valid\_size\_med, ]  
test\_data\_med <- data\_med[(valid\_size\_med + 1):nrow(data\_med), ]  
  
nrow(data\_med)

## [1] 7750

nrow(train\_data\_med)

## [1] 4650

nrow(valid\_data\_med)

## [1] 1550

nrow(test\_data\_med)

## [1] 1550

Now we apply regression model and train it with the train dataset for median and check it’s Evaluation Metric (Root Mean Squared Error [RMSE]) value for our Approach - 2

predictors <- setdiff(names(train\_data\_med), c('Next\_Tmax', 'Date', 'station', 'Next\_Tmin'))  
train\_data\_subset\_med <- train\_data\_med[, c('Next\_Tmax', predictors)]  
  
model\_med <- lm(Next\_Tmax ~ ., data = train\_data\_subset\_med)  
valid\_data\_subset\_med <- valid\_data\_med[, predictors]  
predictions\_med <- predict(model\_med, newdata = valid\_data\_subset\_med)  
  
rmse <- sqrt(mean((valid\_data\_med$Next\_Tmax - predictions\_med)^2))  
rmse

## [1] 1.629672

#### Approach 3 - Omitting NULL values

data <- na.omit(data)  
data$station <- as.factor(data$station)  
sum(is.na(data))

## [1] 0

nrow(data)

## [1] 7588

After removing the NULL values, we split this data set into train, validation and test data sets. Following is the number of rows for the entire data set, train, vaidation & test data sets respectively

data <- data[order(data$Date), ]  
  
train\_size <- round(nrow(data) \* 0.60)  
valid\_size <- round(nrow(data) \* 0.80)  
  
train\_data <- data[1:train\_size, ]  
valid\_data <- data[(train\_size + 1):valid\_size, ]  
test\_data <- data[(valid\_size + 1):nrow(data), ]  
  
nrow(data)

## [1] 7588

nrow(train\_data)

## [1] 4553

nrow(valid\_data)

## [1] 1517

nrow(test\_data)

## [1] 1518

Now we apply regression model and train it with the train dataset and check it’s Evaluation Metric (Root Mean Squared Error [RMSE]) value for our Approach - 3

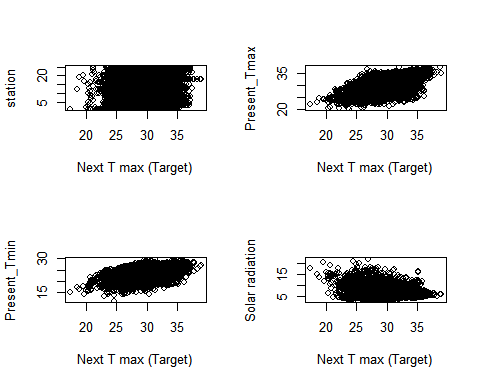
predictors <- setdiff(names(train\_data), c('Next\_Tmax', 'Date', 'station', 'Next\_Tmin'))  
train\_data\_subset <- train\_data[, c('Next\_Tmax', predictors)]  
  
model <- lm(Next\_Tmax ~ ., data = train\_data\_subset)  
valid\_data\_subset <- valid\_data[, predictors]  
predictions <- predict(model, newdata = valid\_data\_subset)  
  
rmse <- sqrt(mean((valid\_data$Next\_Tmax - predictions)^2))  
rmse

## [1] 1.513085

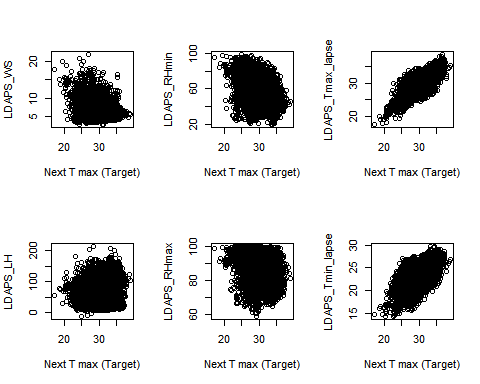
From the above three approaches, it is clearly evident that the Approach - 3 which omits null values is the best method of handling null values for the given data set with the least RMSE Value of 1.513. Thus, this approach is selected for further improvements

#### Target Features vs Different Variables in the Dataset

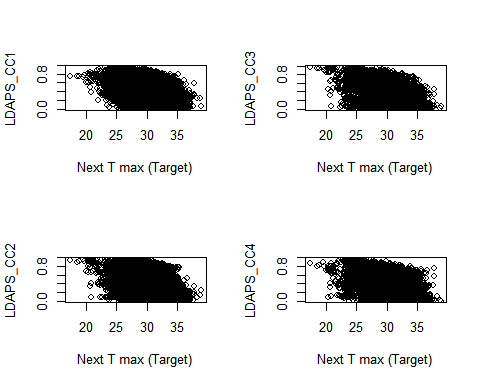
layout(matrix(1:4,2,2))  
 plot(data$Next\_Tmax,data$station , ylab="station", xlab="Next T max (Target)")  
 plot(data$Next\_Tmax,data$Present\_Tmin , ylab="Present\_Tmin", xlab="Next T max (Target)")  
 plot(data$Next\_Tmax,data$Present\_Tmax , ylab="Present\_Tmax", xlab="Next T max (Target)")  
 plot(data$Next\_Tmax,data$LDAPS\_WS , ylab="Solar radiation", xlab="Next T max (Target)")



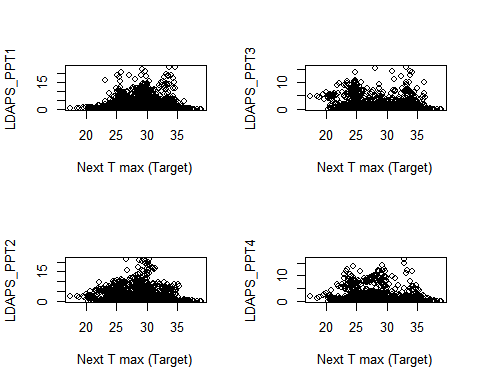
layout(matrix(1:6,2,3))  
 plot(data$Next\_Tmax,data$LDAPS\_WS , ylab="LDAPS\_WS", xlab="Next T max (Target)")  
 plot(data$Next\_Tmax,data$LDAPS\_LH , ylab="LDAPS\_LH", xlab="Next T max (Target)")  
 plot(data$Next\_Tmax,data$LDAPS\_RHmin , ylab="LDAPS\_RHmin", xlab="Next T max (Target)")  
 plot(data$Next\_Tmax,data$LDAPS\_RHmax , ylab="LDAPS\_RHmax", xlab="Next T max (Target)")  
 plot(data$Next\_Tmax,data$LDAPS\_Tmax\_lapse , ylab="LDAPS\_Tmax\_lapse", xlab="Next T max (Target)")  
 plot(data$Next\_Tmax,data$LDAPS\_Tmin\_lapse , ylab="LDAPS\_Tmin\_lapse", xlab="Next T max (Target)")



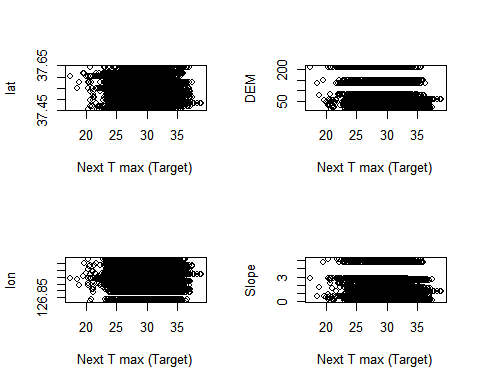
layout(matrix(1:4,2,2))  
 plot(data$Next\_Tmax,data$LDAPS\_CC1 , ylab="LDAPS\_CC1", xlab="Next T max (Target)")  
 plot(data$Next\_Tmax,data$LDAPS\_CC2 , ylab="LDAPS\_CC2", xlab="Next T max (Target)")  
 plot(data$Next\_Tmax,data$LDAPS\_CC3 , ylab="LDAPS\_CC3", xlab="Next T max (Target)")  
 plot(data$Next\_Tmax,data$LDAPS\_CC4 , ylab="LDAPS\_CC4", xlab="Next T max (Target)")



layout(matrix(1:4,2,2))  
 plot(data$Next\_Tmax,data$LDAPS\_PPT1 , ylab="LDAPS\_PPT1", xlab="Next T max (Target)")  
 plot(data$Next\_Tmax,data$LDAPS\_PPT2 , ylab="LDAPS\_PPT2", xlab="Next T max (Target)")  
 plot(data$Next\_Tmax,data$LDAPS\_PPT3 , ylab="LDAPS\_PPT3", xlab="Next T max (Target)")  
 plot(data$Next\_Tmax,data$LDAPS\_PPT4 , ylab="LDAPS\_PPT4", xlab="Next T max (Target)")



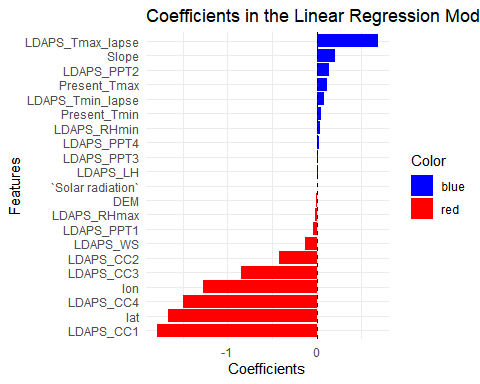
layout(matrix(1:4,2,2))  
 plot(data$Next\_Tmax,data$lat , ylab="lat", xlab="Next T max (Target)")  
 plot(data$Next\_Tmax,data$lon , ylab="lon", xlab="Next T max (Target)")  
 plot(data$Next\_Tmax,data$DEM , ylab="DEM", xlab="Next T max (Target)")  
 plot(data$Next\_Tmax,data$Slope , ylab="Slope", xlab="Next T max (Target)")



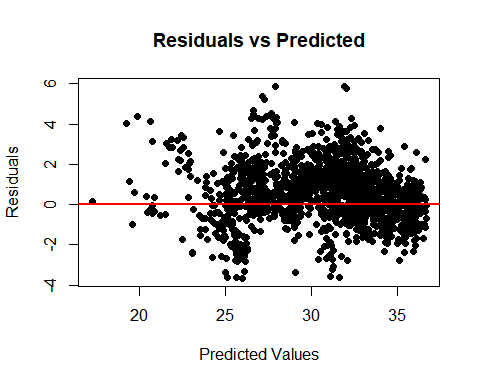
coefficients <- coef(model)[-1] # Exclude the intercept  
  
coefficients\_df <- data.frame(Feature = names(coefficients), Coefficient = coefficients)  
coefficients\_df$Color <- ifelse(coefficients\_df$Coefficient > 0, "blue", "red")  
  
library(ggplot2)

## Warning: package 'ggplot2' was built under R version 4.3.2

ggplot(coefficients\_df, aes(x = reorder(Feature, Coefficient), y = Coefficient, fill = Color)) +  
 geom\_bar(stat = "identity") +  
 coord\_flip() +  
 theme\_minimal() +  
 labs(x = "Features", y = "Coefficients", title = "Coefficients in the Linear Regression Model") +  
 geom\_hline(yintercept = 0, linetype = "dashed", color = "black") +  
 scale\_fill\_manual(values = c("blue", "red"))



# Residuals Plot  
residuals <- valid\_data$Next\_Tmax - predictions  
  
plot(predictions, residuals,  
 xlab = "Predicted Values",  
 ylab = "Residuals",  
 main = "Residuals vs Predicted",  
 pch = 19,  
 col = "black")  
abline(h = 0, col = "red", lwd = 2)



#### **(d) Improved Model**

### Model 1

We calculate the P-value every variable in the dataset to know what are the “non-significant” variables for analysis. By “non-significant”, here we mean which variables doesn’t contribute much to predict the target variable

P-Value > 0.05 is taken into consideration to consider variables to be non-significant under the following conditions for Hypothesis

: Coefficient of Variable = 0 : Coefficient of Variable 0

Thus if P-Value > 0.05, it means that the coefficient of that particular variable in the multiple linear regression equation is 0 i.e. predicted value is not dependent on this variable

For our Approach - 3, let’s move on to find such “non-significant” variables

summary(model)$coefficients[, "Pr(>|t|)"]

## (Intercept) Present\_Tmax Present\_Tmin LDAPS\_RHmin   
## 6.060376e-11 9.644519e-21 7.675173e-03 8.812238e-25   
## LDAPS\_RHmax LDAPS\_Tmax\_lapse LDAPS\_Tmin\_lapse LDAPS\_WS   
## 4.527696e-08 2.250477e-231 1.764642e-03 9.290196e-36   
## LDAPS\_LH LDAPS\_CC1 LDAPS\_CC2 LDAPS\_CC3   
## 3.173148e-27 3.062570e-31 2.677528e-02 7.754436e-06   
## LDAPS\_CC4 LDAPS\_PPT1 LDAPS\_PPT2 LDAPS\_PPT3   
## 7.492563e-24 6.134319e-05 2.698886e-24 5.557203e-01   
## LDAPS\_PPT4 lat lon DEM   
## 2.598408e-01 2.266374e-04 6.829667e-06 1.716810e-12   
## Slope `Solar radiation`   
## 2.223459e-15 1.353044e-02

non\_significant\_vars <- summary(model)$coefficients[, "Pr(>|t|)"] > 0.05  
names(non\_significant\_vars[non\_significant\_vars])

## [1] "LDAPS\_PPT3" "LDAPS\_PPT4"

From the above result, “LDAPS\_PPT3” “LDAPS\_PPT4” are the “non-significant” variables. Thus, we remove them and apply the model again to test the RMSE value. Along with the above variables, we also removed redundant variables (Eg: Station code can be used instead of lat, lon, DEM, Slope)

# Remove the non-significant columns and Next\_Tmin from the dataset  
data$LDAPS\_PPT3 <- NULL  
data$LDAPS\_PPT4 <- NULL  
data$DEM <- NULL  
data$Next\_Tmin <- NULL  
  
train\_df1 <- subset(train\_data, select = -c(Next\_Tmin, Date, lon, lat, Slope, DEM, LDAPS\_PPT3, LDAPS\_PPT4))  
valid\_df1 <- subset(valid\_data, select = -c(Next\_Tmax, Next\_Tmin, Date, lon, lat, Slope, DEM, LDAPS\_PPT3, LDAPS\_PPT4))  
  
model1 <- lm(Next\_Tmax ~ ., data = train\_df1)  
  
summary(model1)

##   
## Call:  
## lm(formula = Next\_Tmax ~ ., data = train\_df1)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -5.5277 -0.8230 0.0042 0.8172 5.2315   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 5.036e+00 5.784e-01 8.708 < 2e-16 \*\*\*  
## station2 8.512e-01 1.472e-01 5.782 7.86e-09 \*\*\*  
## station3 7.122e-01 1.642e-01 4.338 1.47e-05 \*\*\*  
## station4 1.704e+00 1.499e-01 11.363 < 2e-16 \*\*\*  
## station5 9.943e-01 1.495e-01 6.652 3.23e-11 \*\*\*  
## station6 1.171e+00 1.537e-01 7.622 3.02e-14 \*\*\*  
## station7 1.125e+00 1.536e-01 7.324 2.83e-13 \*\*\*  
## station8 1.219e+00 1.516e-01 8.042 1.12e-15 \*\*\*  
## station9 1.656e+00 1.463e-01 11.322 < 2e-16 \*\*\*  
## station10 1.460e+00 1.448e-01 10.084 < 2e-16 \*\*\*  
## station11 1.192e+00 1.524e-01 7.819 6.58e-15 \*\*\*  
## station12 1.260e+00 1.552e-01 8.113 6.30e-16 \*\*\*  
## station13 1.109e+00 1.590e-01 6.972 3.58e-12 \*\*\*  
## station14 1.195e+00 1.624e-01 7.361 2.16e-13 \*\*\*  
## station15 9.804e-01 1.562e-01 6.277 3.77e-10 \*\*\*  
## station16 6.100e-01 1.447e-01 4.215 2.55e-05 \*\*\*  
## station17 8.210e-01 1.500e-01 5.475 4.62e-08 \*\*\*  
## station18 2.368e+00 1.504e-01 15.741 < 2e-16 \*\*\*  
## station19 1.121e+00 1.508e-01 7.431 1.28e-13 \*\*\*  
## station20 2.240e+00 1.469e-01 15.244 < 2e-16 \*\*\*  
## station21 4.741e-01 1.633e-01 2.904 0.003703 \*\*   
## station22 1.234e+00 1.491e-01 8.274 < 2e-16 \*\*\*  
## station23 1.832e+00 1.494e-01 12.264 < 2e-16 \*\*\*  
## station24 1.411e+00 1.543e-01 9.146 < 2e-16 \*\*\*  
## station25 1.322e+00 1.627e-01 8.121 5.90e-16 \*\*\*  
## Present\_Tmax 7.358e-02 1.183e-02 6.222 5.37e-10 \*\*\*  
## Present\_Tmin -3.623e-03 1.605e-02 -0.226 0.821469   
## LDAPS\_RHmin 3.110e-02 3.672e-03 8.470 < 2e-16 \*\*\*  
## LDAPS\_RHmax -2.499e-02 4.143e-03 -6.032 1.75e-09 \*\*\*  
## LDAPS\_Tmax\_lapse 6.777e-01 1.892e-02 35.819 < 2e-16 \*\*\*  
## LDAPS\_Tmin\_lapse 1.483e-01 2.384e-02 6.219 5.46e-10 \*\*\*  
## LDAPS\_WS -1.236e-01 1.067e-02 -11.577 < 2e-16 \*\*\*  
## LDAPS\_LH 7.268e-03 1.209e-03 6.010 2.01e-09 \*\*\*  
## LDAPS\_CC1 -1.847e+00 1.488e-01 -12.408 < 2e-16 \*\*\*  
## LDAPS\_CC2 -3.618e-01 1.800e-01 -2.010 0.044468 \*   
## LDAPS\_CC3 -6.629e-01 1.800e-01 -3.684 0.000232 \*\*\*  
## LDAPS\_CC4 -1.476e+00 1.368e-01 -10.787 < 2e-16 \*\*\*  
## LDAPS\_PPT1 -3.985e-02 1.134e-02 -3.515 0.000445 \*\*\*  
## LDAPS\_PPT2 1.499e-01 1.302e-02 11.520 < 2e-16 \*\*\*  
## `Solar radiation` 1.033e-04 5.303e-05 1.947 0.051592 .   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 1.348 on 4513 degrees of freedom  
## Multiple R-squared: 0.7613, Adjusted R-squared: 0.7592   
## F-statistic: 369.1 on 39 and 4513 DF, p-value: < 2.2e-16

predictions1 <- predict(model1, newdata = valid\_df1)  
  
# Calculate new RMSE  
rmse1 <- sqrt(mean((valid\_data$Next\_Tmax - predictions1)^2))  
rmse1

## [1] 1.465021

anova(model1, model)

## Analysis of Variance Table  
##   
## Model 1: Next\_Tmax ~ station + Present\_Tmax + Present\_Tmin + LDAPS\_RHmin +   
## LDAPS\_RHmax + LDAPS\_Tmax\_lapse + LDAPS\_Tmin\_lapse + LDAPS\_WS +   
## LDAPS\_LH + LDAPS\_CC1 + LDAPS\_CC2 + LDAPS\_CC3 + LDAPS\_CC4 +   
## LDAPS\_PPT1 + LDAPS\_PPT2 + `Solar radiation`  
## Model 2: Next\_Tmax ~ Present\_Tmax + Present\_Tmin + LDAPS\_RHmin + LDAPS\_RHmax +   
## LDAPS\_Tmax\_lapse + LDAPS\_Tmin\_lapse + LDAPS\_WS + LDAPS\_LH +   
## LDAPS\_CC1 + LDAPS\_CC2 + LDAPS\_CC3 + LDAPS\_CC4 + LDAPS\_PPT1 +   
## LDAPS\_PPT2 + LDAPS\_PPT3 + LDAPS\_PPT4 + lat + lon + DEM +   
## Slope + `Solar radiation`  
## Res.Df RSS Df Sum of Sq F Pr(>F)   
## 1 4513 8205.3   
## 2 4531 8984.2 -18 -778.98 23.803 < 2.2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

From Anova above we can conclude that RSS value has decreased. Hence, Model1 is better fit for this data than Model.

### Model 2

In Model2 to avoid over fitting the model with Cloud coverage and Precipitation values, we simplify the data by adding new columns “Night\_cloud\_cover”, “Day\_Cloud\_Cover”, “Precipitation” and removed respective old columns.

train\_df2 <- within(train\_df1, {  
 Night\_Cloud\_Cover = (LDAPS\_CC1 + LDAPS\_CC4) / 2  
 Day\_Cloud\_Cover = (LDAPS\_CC2 + LDAPS\_CC3) / 2  
 Precipitation = (LDAPS\_PPT1 + LDAPS\_PPT2) / 2  
   
 # Remove the original columns  
 LDAPS\_CC1 = NULL  
 LDAPS\_CC2 = NULL  
 LDAPS\_CC3 = NULL  
 LDAPS\_CC4 = NULL  
 LDAPS\_PPT1 = NULL  
 LDAPS\_PPT2 = NULL  
})  
  
head(train\_df2)

## # A tibble: 6 × 14  
## station Present\_Tmax Present\_Tmin LDAPS\_RHmin LDAPS\_RHmax LDAPS\_Tmax\_lapse  
## <fct> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 1 28.7 21.4 58.3 91.1 28.1  
## 2 2 31.9 21.6 52.3 90.6 29.9  
## 3 3 31.6 23.3 48.7 84.0 30.1  
## 4 4 32 23.4 58.2 96.5 29.7  
## 5 5 31.4 21.9 56.2 90.2 29.1  
## 6 6 31.9 23.5 52.4 85.3 29.2  
## # ℹ 8 more variables: LDAPS\_Tmin\_lapse <dbl>, LDAPS\_WS <dbl>, LDAPS\_LH <dbl>,  
## # `Solar radiation` <dbl>, Next\_Tmax <dbl>, Precipitation <dbl>,  
## # Day\_Cloud\_Cover <dbl>, Night\_Cloud\_Cover <dbl>

valid\_df2 <- within(valid\_df1, {  
 Night\_Cloud\_Cover = (LDAPS\_CC1 + LDAPS\_CC4) / 2  
 Day\_Cloud\_Cover = (LDAPS\_CC2 + LDAPS\_CC3) / 2  
 Precipitation = LDAPS\_PPT1 + LDAPS\_PPT2 / 2  
   
 # Remove the original columns  
 LDAPS\_CC1 = NULL  
 LDAPS\_CC2 = NULL  
 LDAPS\_CC3 = NULL  
 LDAPS\_CC4 = NULL  
 LDAPS\_PPT1 = NULL  
 LDAPS\_PPT2 = NULL  
})  
head(valid\_df2)

## # A tibble: 6 × 13  
## station Present\_Tmax Present\_Tmin LDAPS\_RHmin LDAPS\_RHmax LDAPS\_Tmax\_lapse  
## <fct> <dbl> <dbl> <dbl> <dbl> <dbl>  
## 1 14 30.6 19.8 33.8 76.5 30.6  
## 2 15 30.4 19.9 35.8 77.7 30.3  
## 3 16 28.8 18 37.4 90.0 29.4  
## 4 17 28.9 16.5 39.8 94.2 29.9  
## 5 18 30.6 19.9 36.4 90.3 29.8  
## 6 19 30.7 19.4 35.5 79.6 29.9  
## # ℹ 7 more variables: LDAPS\_Tmin\_lapse <dbl>, LDAPS\_WS <dbl>, LDAPS\_LH <dbl>,  
## # `Solar radiation` <dbl>, Precipitation <dbl>, Day\_Cloud\_Cover <dbl>,  
## # Night\_Cloud\_Cover <dbl>

model2 <- lm(Next\_Tmax ~ ., data = train\_df2)  
  
predictions2 <- predict(model2, newdata = valid\_df2)  
  
rmse <- sqrt(mean((valid\_data$Next\_Tmax - predictions2)^2))  
  
print(rmse)

## [1] 1.431952

summary(model2)

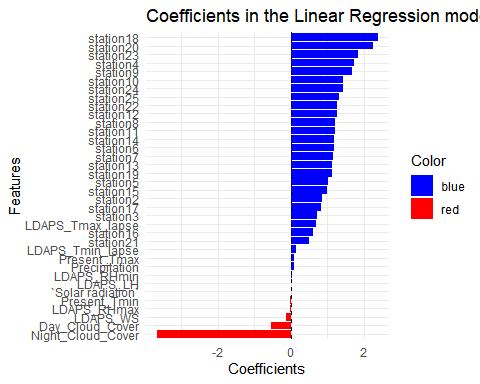
##   
## Call:  
## lm(formula = Next\_Tmax ~ ., data = train\_df2)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -5.4626 -0.8074 0.0237 0.8218 5.2902   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 5.358e+00 5.784e-01 9.264 < 2e-16 \*\*\*  
## station2 8.619e-01 1.486e-01 5.800 7.08e-09 \*\*\*  
## station3 7.296e-01 1.656e-01 4.407 1.07e-05 \*\*\*  
## station4 1.726e+00 1.516e-01 11.387 < 2e-16 \*\*\*  
## station5 1.016e+00 1.512e-01 6.718 2.07e-11 \*\*\*  
## station6 1.188e+00 1.553e-01 7.654 2.37e-14 \*\*\*  
## station7 1.147e+00 1.553e-01 7.384 1.82e-13 \*\*\*  
## station8 1.222e+00 1.532e-01 7.974 1.93e-15 \*\*\*  
## station9 1.667e+00 1.478e-01 11.275 < 2e-16 \*\*\*  
## station10 1.445e+00 1.464e-01 9.874 < 2e-16 \*\*\*  
## station11 1.214e+00 1.540e-01 7.884 3.93e-15 \*\*\*  
## station12 1.257e+00 1.567e-01 8.027 1.26e-15 \*\*\*  
## station13 1.130e+00 1.605e-01 7.043 2.17e-12 \*\*\*  
## station14 1.189e+00 1.638e-01 7.255 4.70e-13 \*\*\*  
## station15 9.908e-01 1.577e-01 6.283 3.64e-10 \*\*\*  
## station16 6.012e-01 1.464e-01 4.107 4.09e-05 \*\*\*  
## station17 8.241e-01 1.516e-01 5.436 5.73e-08 \*\*\*  
## station18 2.380e+00 1.520e-01 15.653 < 2e-16 \*\*\*  
## station19 1.119e+00 1.523e-01 7.351 2.32e-13 \*\*\*  
## station20 2.250e+00 1.486e-01 15.139 < 2e-16 \*\*\*  
## station21 5.014e-01 1.648e-01 3.043 0.00236 \*\*   
## station22 1.259e+00 1.508e-01 8.347 < 2e-16 \*\*\*  
## station23 1.849e+00 1.510e-01 12.246 < 2e-16 \*\*\*  
## station24 1.426e+00 1.559e-01 9.145 < 2e-16 \*\*\*  
## station25 1.332e+00 1.642e-01 8.111 6.43e-16 \*\*\*  
## Present\_Tmax 8.439e-02 1.151e-02 7.332 2.67e-13 \*\*\*  
## Present\_Tmin -2.217e-02 1.603e-02 -1.383 0.16685   
## LDAPS\_RHmin 3.409e-02 3.654e-03 9.329 < 2e-16 \*\*\*  
## LDAPS\_RHmax -2.934e-02 4.046e-03 -7.251 4.83e-13 \*\*\*  
## LDAPS\_Tmax\_lapse 6.835e-01 1.888e-02 36.201 < 2e-16 \*\*\*  
## LDAPS\_Tmin\_lapse 1.367e-01 2.346e-02 5.828 6.02e-09 \*\*\*  
## LDAPS\_WS -1.224e-01 1.079e-02 -11.347 < 2e-16 \*\*\*  
## LDAPS\_LH 7.178e-03 1.214e-03 5.912 3.62e-09 \*\*\*  
## `Solar radiation` 1.086e-04 5.299e-05 2.050 0.04044 \*   
## Precipitation 8.416e-02 1.453e-02 5.793 7.39e-09 \*\*\*  
## Day\_Cloud\_Cover -5.374e-01 2.232e-01 -2.408 0.01608 \*   
## Night\_Cloud\_Cover -3.656e+00 2.038e-01 -17.935 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 1.365 on 4516 degrees of freedom  
## Multiple R-squared: 0.7554, Adjusted R-squared: 0.7534   
## F-statistic: 387.4 on 36 and 4516 DF, p-value: < 2.2e-16

anova(model2, model)

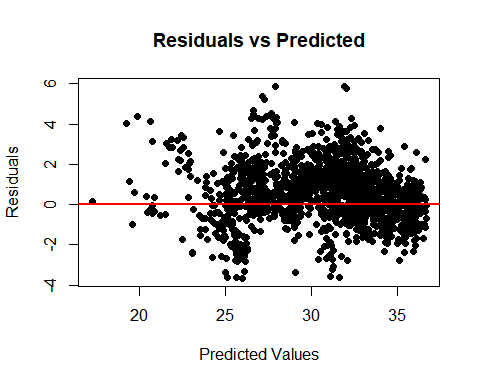
## Analysis of Variance Table  
##   
## Model 1: Next\_Tmax ~ station + Present\_Tmax + Present\_Tmin + LDAPS\_RHmin +   
## LDAPS\_RHmax + LDAPS\_Tmax\_lapse + LDAPS\_Tmin\_lapse + LDAPS\_WS +   
## LDAPS\_LH + `Solar radiation` + Precipitation + Day\_Cloud\_Cover +   
## Night\_Cloud\_Cover  
## Model 2: Next\_Tmax ~ Present\_Tmax + Present\_Tmin + LDAPS\_RHmin + LDAPS\_RHmax +   
## LDAPS\_Tmax\_lapse + LDAPS\_Tmin\_lapse + LDAPS\_WS + LDAPS\_LH +   
## LDAPS\_CC1 + LDAPS\_CC2 + LDAPS\_CC3 + LDAPS\_CC4 + LDAPS\_PPT1 +   
## LDAPS\_PPT2 + LDAPS\_PPT3 + LDAPS\_PPT4 + lat + lon + DEM +   
## Slope + `Solar radiation`  
## Res.Df RSS Df Sum of Sq F Pr(>F)   
## 1 4516 8408.7   
## 2 4531 8984.2 -15 -575.55 20.607 < 2.2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

From above, we can conclude that RMSE value has decreased. Hence, Model2 is better fit for this data than Model1, which shows better predictive capability of model2.

coefficients <- coef(model2)[-1]  
  
coefficients\_df <- data.frame(Feature = names(coefficients), Coefficient = coefficients)  
coefficients\_df$Color <- ifelse(coefficients\_df$Coefficient > 0, "blue", "red")  
library(ggplot2)  
ggplot(coefficients\_df, aes(x = reorder(Feature, Coefficient), y = Coefficient, fill = Color)) +  
 geom\_bar(stat = "identity") +  
 coord\_flip() +  
 theme\_minimal() +  
 labs(x = "Features", y = "Coefficients", title = "Coefficients in the Linear Regression model2") +  
 geom\_hline(yintercept = 0, linetype = "dashed", color = "black") +  
 scale\_fill\_manual(values = c("blue", "red"))



residuals <- valid\_data$Next\_Tmax - predictions  
  
plot(predictions, residuals,  
 xlab = "Predicted Values",  
 ylab = "Residuals",  
 main = "Residuals vs Predicted",  
 pch = 19,  
 col = "black")  
abline(h = 0, col = "red", lwd = 2)



### Model 3

Now we again calculate p -values with newly added columns, and remove “non-significant” variables for analysis.

summary(model2)$coefficients[, "Pr(>|t|)"]

## (Intercept) station2 station3 station4   
## 2.966230e-20 7.080116e-09 1.072493e-05 1.230780e-29   
## station5 station6 station7 station8   
## 2.070290e-11 2.374584e-14 1.823000e-13 1.927438e-15   
## station9 station10 station11 station12   
## 4.242386e-29 9.207510e-23 3.934371e-15 1.261330e-15   
## station13 station14 station15 station16   
## 2.168611e-12 4.698410e-13 3.638506e-10 4.087312e-05   
## station17 station18 station19 station20   
## 5.728521e-08 8.063978e-54 2.322741e-13 1.523495e-50   
## station21 station22 station23 station24   
## 2.357054e-03 9.193862e-17 6.063895e-34 8.825244e-20   
## station25 Present\_Tmax Present\_Tmin LDAPS\_RHmin   
## 6.429237e-16 2.674240e-13 1.668512e-01 1.631955e-20   
## LDAPS\_RHmax LDAPS\_Tmax\_lapse LDAPS\_Tmin\_lapse LDAPS\_WS   
## 4.826490e-13 3.494525e-252 6.015211e-09 1.912460e-29   
## LDAPS\_LH `Solar radiation` Precipitation Day\_Cloud\_Cover   
## 3.624385e-09 4.044390e-02 7.386235e-09 1.608306e-02   
## Night\_Cloud\_Cover   
## 1.531597e-69

non\_significant\_vars <- summary(model2)$coefficients[, "Pr(>|t|)"] > 0.05  
names(non\_significant\_vars[non\_significant\_vars])

## [1] "Present\_Tmin"

From the above result, “Present\_Tmin” are the “non-significant” variables. Thus, we remove them and apply the model again to test the RMSE value.

train\_df3 = train\_df2  
valid\_df3 = valid\_df2

train\_df3$Present\_Tmin <- NULL  
valid\_df3$Present\_Tmin <- NULL  
  
model3 <- lm(Next\_Tmax ~ ., data = train\_df3)  
  
predictions3 <- predict(model3, newdata = valid\_df3)  
  
rmse <- sqrt(mean((valid\_data$Next\_Tmax - predictions3)^2))  
  
print(rmse)

## [1] 1.427989

summary(model3)$coefficients[, "Pr(>|t|)"]

## (Intercept) station2 station3 station4   
## 3.581151e-20 8.594860e-09 2.160638e-05 2.697849e-29   
## station5 station6 station7 station8   
## 4.033351e-11 6.193104e-14 4.504811e-13 5.049505e-15   
## station9 station10 station11 station12   
## 1.076020e-28 1.920073e-22 1.031099e-14 2.148432e-15   
## station13 station14 station15 station16   
## 5.713142e-12 1.163389e-12 7.797537e-10 4.532234e-05   
## station17 station18 station19 station20   
## 5.767960e-08 7.621044e-54 4.573071e-13 2.818758e-50   
## station21 station22 station23 station24   
## 4.452705e-03 2.360320e-16 1.155000e-33 2.068781e-19   
## station25 Present\_Tmax LDAPS\_RHmin LDAPS\_RHmax   
## 1.676716e-15 6.580759e-13 2.214436e-20 3.348754e-14   
## LDAPS\_Tmax\_lapse LDAPS\_Tmin\_lapse LDAPS\_WS LDAPS\_LH   
## 8.432877e-253 8.531739e-09 1.839901e-30 6.035723e-09   
## `Solar radiation` Precipitation Day\_Cloud\_Cover Night\_Cloud\_Cover   
## 2.627857e-02 1.810371e-08 2.034394e-02 1.499942e-70

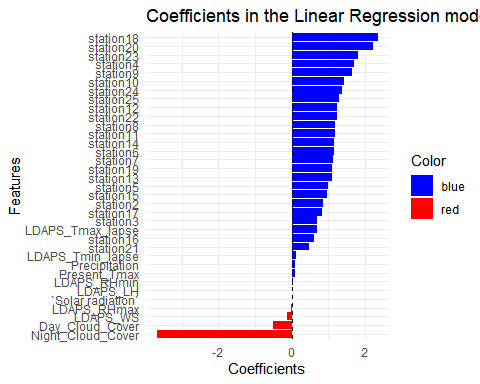
non\_significant\_vars <- summary(model3)$coefficients[, "Pr(>|t|)"] > 0.05  
names(non\_significant\_vars[non\_significant\_vars])

## character(0)

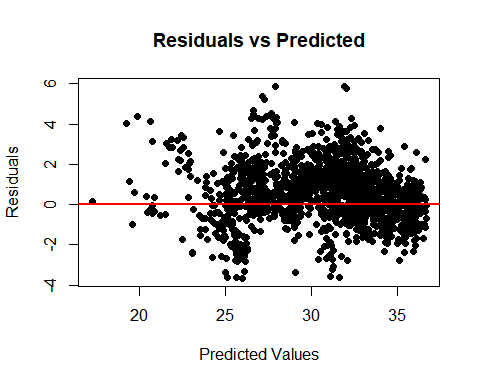
anova(model3, model)

## Analysis of Variance Table  
##   
## Model 1: Next\_Tmax ~ station + Present\_Tmax + LDAPS\_RHmin + LDAPS\_RHmax +   
## LDAPS\_Tmax\_lapse + LDAPS\_Tmin\_lapse + LDAPS\_WS + LDAPS\_LH +   
## `Solar radiation` + Precipitation + Day\_Cloud\_Cover + Night\_Cloud\_Cover  
## Model 2: Next\_Tmax ~ Present\_Tmax + Present\_Tmin + LDAPS\_RHmin + LDAPS\_RHmax +   
## LDAPS\_Tmax\_lapse + LDAPS\_Tmin\_lapse + LDAPS\_WS + LDAPS\_LH +   
## LDAPS\_CC1 + LDAPS\_CC2 + LDAPS\_CC3 + LDAPS\_CC4 + LDAPS\_PPT1 +   
## LDAPS\_PPT2 + LDAPS\_PPT3 + LDAPS\_PPT4 + lat + lon + DEM +   
## Slope + `Solar radiation`  
## Res.Df RSS Df Sum of Sq F Pr(>F)   
## 1 4517 8412.2   
## 2 4531 8984.2 -14 -571.99 21.938 < 2.2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

coefficients <- coef(model3)[-1]  
  
coefficients\_df <- data.frame(Feature = names(coefficients), Coefficient = coefficients)  
coefficients\_df$Color <- ifelse(coefficients\_df$Coefficient > 0, "blue", "red")  
library(ggplot2)  
ggplot(coefficients\_df, aes(x = reorder(Feature, Coefficient), y = Coefficient, fill = Color)) +  
 geom\_bar(stat = "identity") +  
 coord\_flip() +  
 theme\_minimal() +  
 labs(x = "Features", y = "Coefficients", title = "Coefficients in the Linear Regression model2") +  
 geom\_hline(yintercept = 0, linetype = "dashed", color = "black") +  
 scale\_fill\_manual(values = c("blue", "red"))



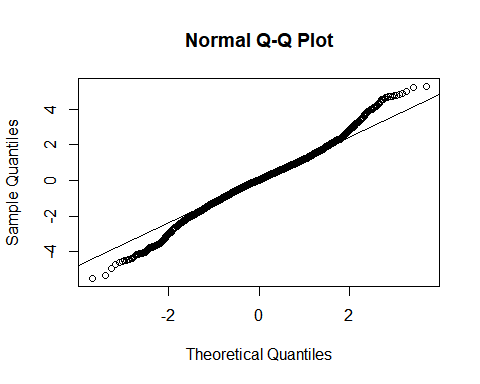
# Residuals Plot  
residuals <- valid\_data$Next\_Tmax - predictions  
  
plot(predictions, residuals,  
 xlab = "Predicted Values",  
 ylab = "Residuals",  
 main = "Residuals vs Predicted",  
 pch = 19,  
 col = "black")  
abline(h = 0, col = "red", lwd = 2)

 From above analysis we dont have any P-values that are > 0.05, but from above coefficients histogram we can tell that Next\_Tmax is least dependent on LDAPS\_LH, Solar Radiation, LDAPS\_RHmax

shapiro.test(model3$residuals)

##   
## Shapiro-Wilk normality test  
##   
## data: model3$residuals  
## W = 0.98899, p-value < 2.2e-16

{  
 qqnorm(model3$residuals)  
 qqline(model3$residuals)  
}



The middle portion of the plot, where points conform more closely to a straight line, suggests that the data distribution is approximately normal.

### Results from Test Data

test\_data\_subset <- test\_data[, predictors]  
test\_predictions <- predict(model, newdata = test\_data\_subset)  
test\_rmse1 <- sqrt(mean((test\_data$Next\_Tmax - test\_predictions)^2))  
test\_rmse1

## [1] 1.647331

test\_df2 <- within(test\_data, {  
 Night\_Cloud\_Cover = (LDAPS\_CC1 + LDAPS\_CC4) / 2  
 Day\_Cloud\_Cover = (LDAPS\_CC2 + LDAPS\_CC3) / 2  
 Precipitation = (LDAPS\_PPT1 + LDAPS\_PPT2) / 2  
   
 LDAPS\_CC1 = NULL  
 LDAPS\_CC2 = NULL  
 LDAPS\_CC3 = NULL  
 LDAPS\_CC4 = NULL  
 LDAPS\_PPT1 = NULL  
 LDAPS\_PPT2 = NULL  
 Next\_Tmin = NULL  
 Date = NULL  
 lon = NULL  
 lat = NULL  
 Slope = NULL   
 DEM = NULL   
 LDAPS\_PPT3 = NULL   
 LDAPS\_PPT4 = NULL  
})  
  
test\_predictions2 <- predict(model3, newdata = test\_df2)  
test\_rmse2 <- sqrt(mean((test\_data$Next\_Tmax - test\_predictions2)^2))  
test\_rmse2

## [1] 1.591924

Based on the RMSE values comparision, it clearly shows that our Improved model can predict with better accuracy than the initial model.