Will it Rain in Seattle?

Contents

Besides coffee, grunge and technology companies, one of the things that Seattle is most famous for is how often it rains. This dataset contains complete records of daily rainfall patterns from January 1st, 1948 to December 12, 2017.

- DATE = the date of the observation
- PRCP = the amount of precipitation, in inches
- TMAX = the maximum temperature for that day, in degrees Fahrenheit
- TMIN = the minimum temperature for that day, in degrees Fahrenheit
- RAIN = TRUE if rain was observed on that day, FALSE if it was not

Here are the packages that are being used

```
library(ggplot2)
library(tidyverse) #
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.2.1 --
```

```
## v tibble  1.4.2     v purrr   0.2.4
## v tidyr   0.8.0     v dplyr   0.7.4
## v readr   1.1.1     v stringr 1.2.0
## v tibble  1.4.2     v forcats 0.2.0
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(dplyr)
library(reshape)
```

```
##
## Attaching package: 'reshape'
```

```
## The following object is masked from 'package:dplyr':
##
##     rename
```

```
## The following objects are masked from 'package:tidyr':
##
##     expand, smiths
```

```
library(caret) # classification and regression training
```

```
## Loading required package: lattice
```

```
##
## Attaching package: 'caret'
```

```
## The following object is masked from 'package:purrr':
##
##     lift
```

```
library(pROC)
```

```
## Type 'citation("pROC")' for a citation.
```

```
##
## Attaching package: 'pROC'
```

```
## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
```

Exploratory Data Analysis

Here we first read in and display the results of the file

```
data.df <- read.csv("DataRainSeattle/seattleWeather_1948-2017.csv", header = TRUE)
head(data.df)
```

```
##         DATE PRCP TMAX TMIN RAIN
## 1 1948-01-01 0.47   51   42 TRUE
## 2 1948-01-02 0.59   45   36 TRUE
## 3 1948-01-03 0.42   45   35 TRUE
## 4 1948-01-04 0.31   45   34 TRUE
## 5 1948-01-05 0.17   45   32 TRUE
## 6 1948-01-06 0.44   48   39 TRUE
```

```
tail(data.df)
```

```
##             DATE PRCP TMAX TMIN  RAIN
## 25546 2017-12-09    0   44   29 FALSE
## 25547 2017-12-10    0   49   34 FALSE
## 25548 2017-12-11    0   49   29 FALSE
## 25549 2017-12-12    0   46   32 FALSE
## 25550 2017-12-13    0   48   34 FALSE
## 25551 2017-12-14    0   50   36 FALSE
```

Missing Data

Taking care of missing data. Just removing the rows where the is no RAIN value recorded. Fixing the problem

```
which(is.na(data.df$RAIN))
```

```
## [1] 18416 18417 21068
```

```
data.df[which(is.na(data.df$RAIN)),]
```

```
##             DATE PRCP TMAX TMIN RAIN
## 18416 1998-06-02   NA   72   52   NA
## 18417 1998-06-03   NA   66   51   NA
## 21068 2005-09-05   NA   70   52   NA
```

```r
# remove those rows
data.df <- data.df[-c(18416, 18417, 21068),]

which(is.na(data.df$TMAX))
```

```
## integer(0)
```

```r
which(is.na(data.df$TMIN))
```

```
## integer(0)
```

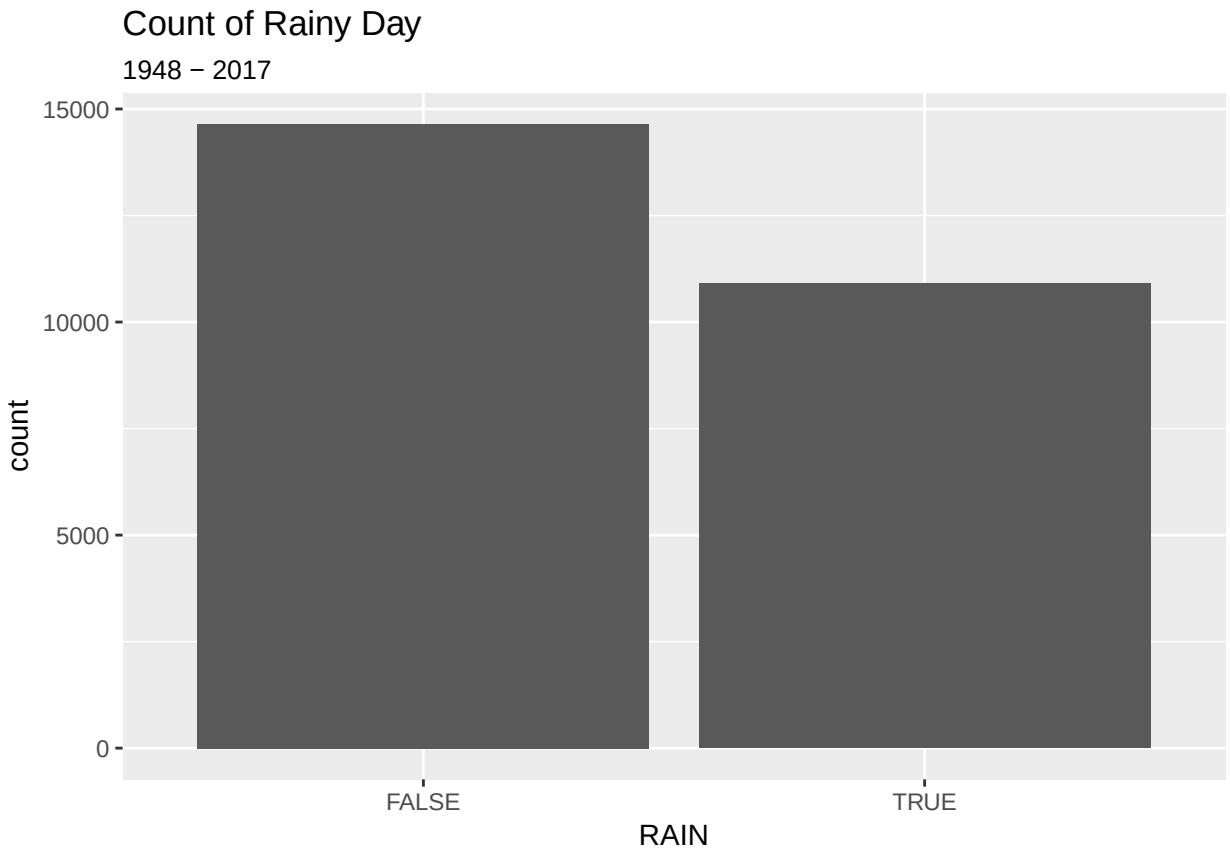
```r
data.df$DATE = as.Date(data.df$DATE)
data.df$RAIN <- as.factor(data.df$RAIN)
```

Exploratory Analysis

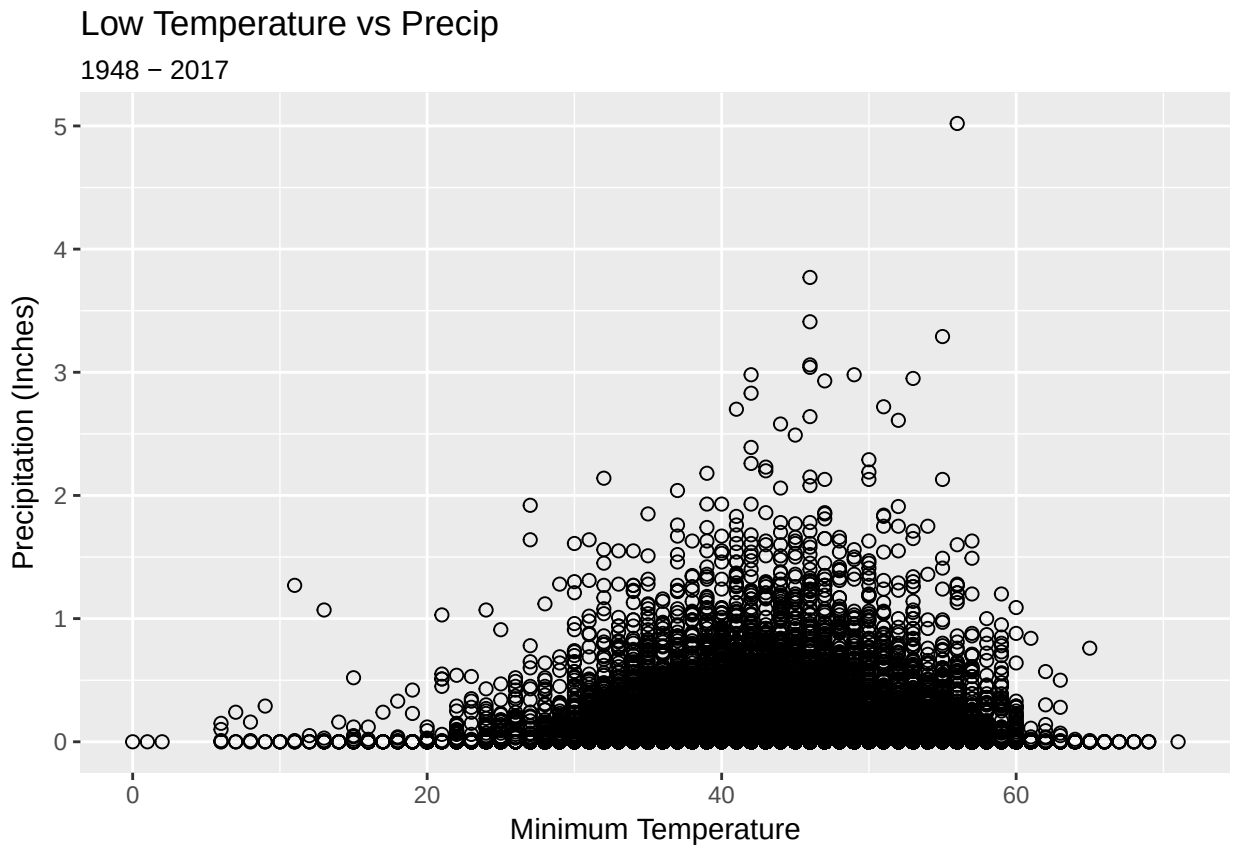
```r
summary(data.df)
```

```
##       DATE                PRCP             TMAX             TMIN
##  Min.   :1948-01-01   Min.   :0.0000   Min.   :  4.00   Min.   : 0.00
##  1st Qu.:1965-06-26   1st Qu.:0.0000   1st Qu.: 50.00   1st Qu.:38.00
##  Median :1982-12-21   Median :0.0000   Median : 58.00   Median :45.00
##  Mean   :1982-12-22   Mean   :0.1062   Mean   : 59.54   Mean   :44.51
##  3rd Qu.:2000-06-18   3rd Qu.:0.1000   3rd Qu.: 69.00   3rd Qu.:52.00
##  Max.   :2017-12-14   Max.   :5.0200   Max.   :103.00   Max.   :71.00
##     RAIN
##  FALSE:14648
##  TRUE :10900
##
##
##
##
```

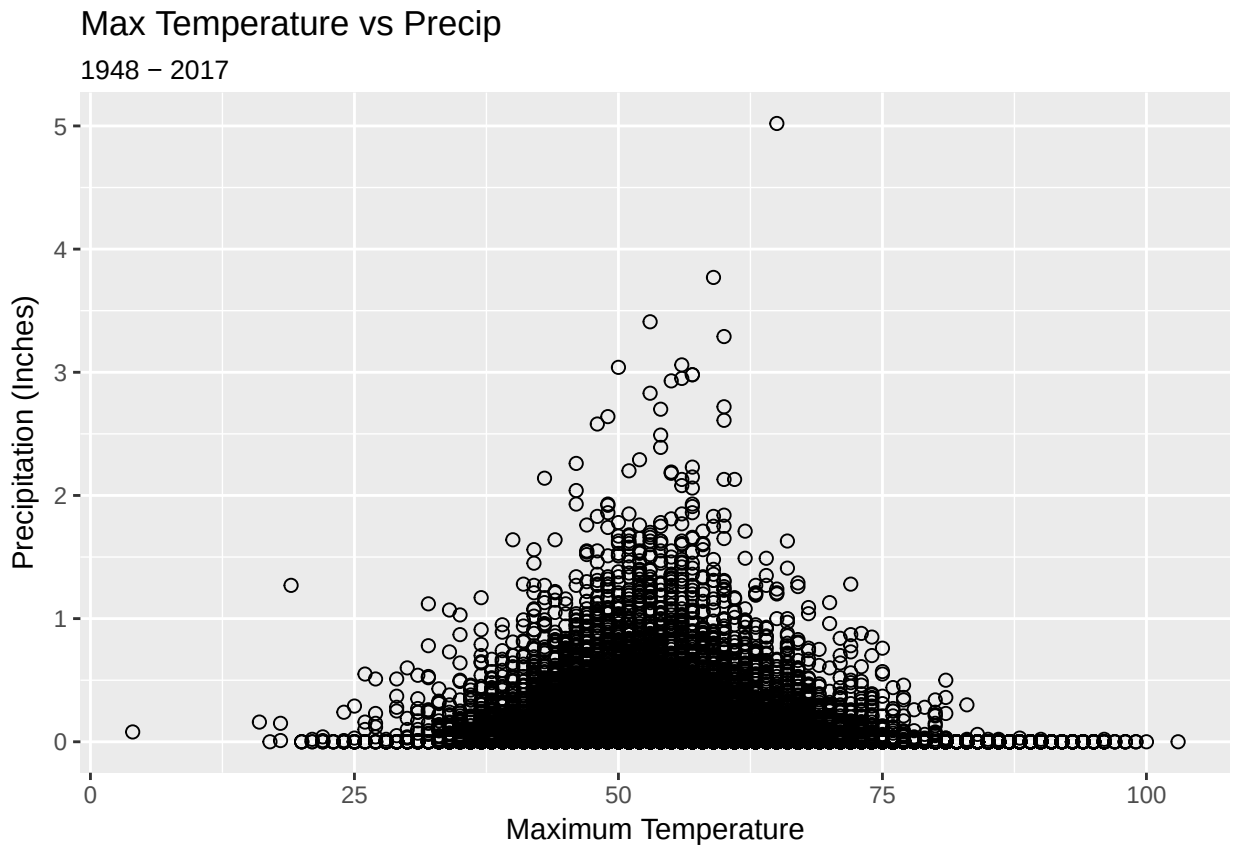
```r
# plot of Rainy vs Non-Rainy days
ggplot(data.df, aes(RAIN)) + geom_bar() + labs(title="Count of Rainy Day",  subtitle="1948 - 2017")
```




```
ggplot(data.df, aes(x=TMIN, y=PRCP)) + geom_point(size=2, shape=1) + xlab("Minimum Temperature")  + ylab
```



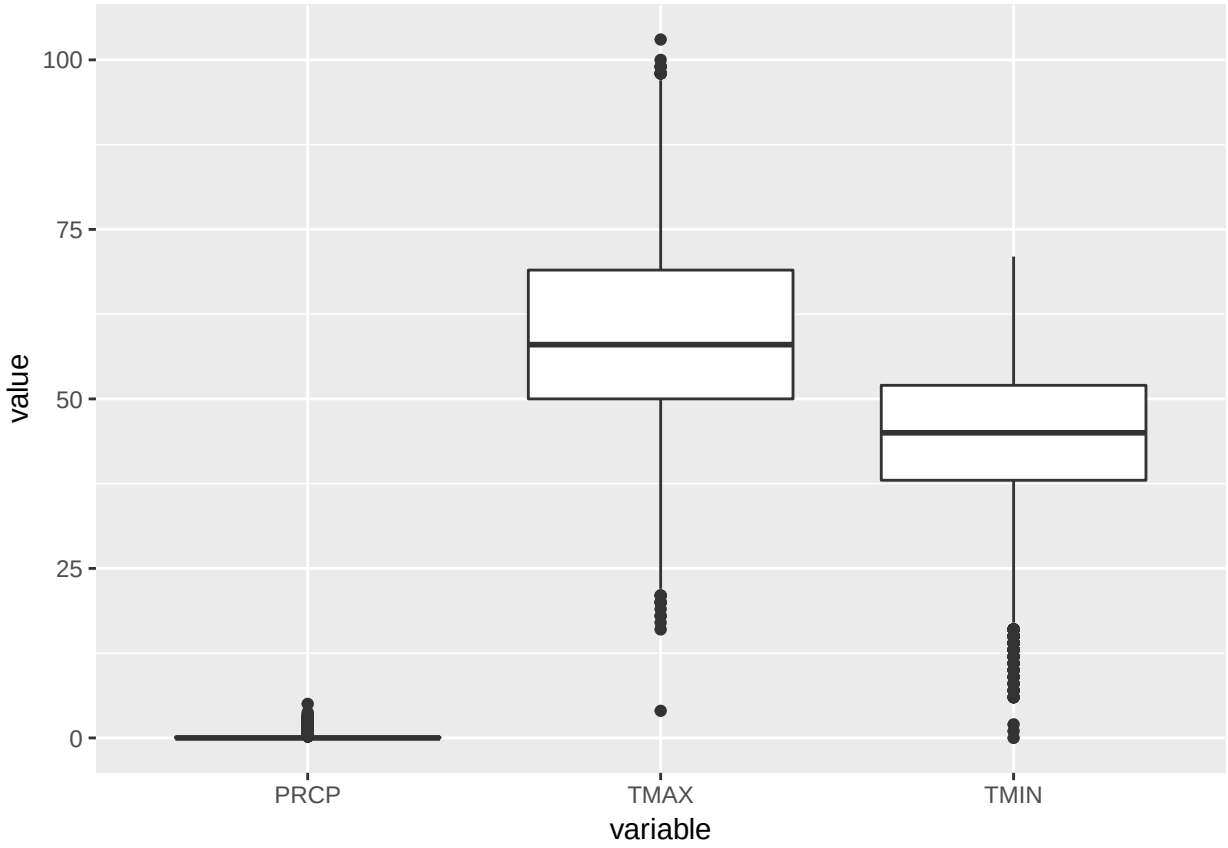

```
ggplot(data.df, aes(x=TMAX, y=PRCP)) + geom_point(size=2, shape=1) + xlab("Maximum Temperature")  + ylab
```




```
data.df.melted = melt(data.df[,-5],id.vars = c("DATE"))

# grouped histogram plot

ggplot(data.df.melted, aes(x=variable, y=value, fill=value)) + geom_boxplot()
```



remove the outliners

```
TMIN_todrop = which(data.df$TMIN < 17)
TMAX_todrop = which(data.df$TMAX > 97.5 | data.df$TMAX < 21.5)
PRECIP_todrop = which(data.df$PRECIP > 0.25 | data.df$PRECIP < -0.15)

index_todrop = union(TMIN_todrop, TMAX_todrop)
index_todrop = union(index_todrop, PRECIP_todrop)

data.df.filtered = data.df[-index_todrop,]
```

```
#Function createDataPartition to create train and test dataset (0.8: 0.2)
index <- createDataPartition(data.df.filtered$RAIN, p = 0.8, list = FALSE)

# Training
train.df <- data.df.filtered[index,]

# Testing
test.df <- data.df.filtered[-index,]
```

Let's look at the summary of the results. We see that the mean values are aproximately the same, as well as the temperature TMAX and TMIN values.

```
head(train.df)
```

```
##         DATE PRCP TMAX TMIN RAIN
## 2 1948-01-02 0.59   45   36 TRUE
## 4 1948-01-04 0.31   45   34 TRUE
```

```
## 5 1948-01-05 0.17   45   32 TRUE
## 6 1948-01-06 0.44   48   39 TRUE
## 7 1948-01-07 0.41   50   40 TRUE
## 8 1948-01-08 0.04   48   35 TRUE
```

```
head(test.df)
```

```
##          DATE PRCP TMAX TMIN  RAIN
## 1  1948-01-01 0.47   51   42  TRUE
## 3  1948-01-03 0.42   45   35  TRUE
## 13 1948-01-13 0.00   45   29 FALSE
## 23 1948-01-23 0.00   47   43 FALSE
## 28 1948-01-28 0.00   53   25 FALSE
## 29 1948-01-29 0.22   42   34  TRUE
```

```
summary(train.df)
```

```
##       DATE                 PRCP             TMAX            TMIN
##  Min.   :1948-01-02   Min.   :0.0000   Min.   :24.00   Min.   :17.00
##  1st Qu.:1965-08-21   1st Qu.:0.0000   1st Qu.:50.00   1st Qu.:38.00
##  Median :1983-03-21   Median :0.0000   Median :58.00   Median :45.00
##  Mean   :1983-02-26   Mean   :0.1061   Mean   :59.63   Mean   :44.61
##  3rd Qu.:2000-09-15   3rd Qu.:0.1000   3rd Qu.:69.00   3rd Qu.:52.00
##  Max.   :2017-12-13   Max.   :5.0200   Max.   :97.00   Max.   :69.00
##     RAIN
##  FALSE:11653
##  TRUE : 8703
##
##
##
##
```

```
summary(test.df)
```

```
##       DATE                 PRCP             TMAX            TMIN
##  Min.   :1948-01-01   Min.   :0.0000   Min.   :27.00   Min.   :17.00
##  1st Qu.:1965-04-14   1st Qu.:0.0000   1st Qu.:50.00   1st Qu.:38.00
##  Median :1982-05-28   Median :0.0000   Median :58.00   Median :45.00
##  Mean   :1982-07-14   Mean   :0.1082   Mean   :59.72   Mean   :44.68
##  3rd Qu.:1999-09-27   3rd Qu.:0.1100   3rd Qu.:69.00   3rd Qu.:52.00
##  Max.   :2017-12-14   Max.   :2.9800   Max.   :96.00   Max.   :68.00
##     RAIN
##  FALSE:2913
##  TRUE :2175
##
##
##
##
```

Training Model

```
# glm logistic regression
# rain is being predicted as a function of the temperatures
model <- glm(RAIN ~ TMAX + TMIN, data = train.df, family = binomial)
```

```r
summary(model)
```

```
##
## Call:
## glm(formula = RAIN ~ TMAX + TMIN, family = binomial, data = train.df)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.4858  -0.8024  -0.2508   0.8361   3.3364
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  2.881682   0.094537   30.48   <2e-16 ***
## TMAX        -0.253818   0.003896  -65.15   <2e-16 ***
## TMIN         0.262120   0.004813   54.46   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 27790  on 20355  degrees of freedom
## Residual deviance: 20236  on 20353  degrees of freedom
## AIC: 20242
##
## Number of Fisher Scoring iterations: 5
```

```r
predicted_values <- predict(model, test.df[,-5], type = "response")
head(predicted_values)
```

```
##          1          3         13         23         28         29
## 0.72031212 0.65341365 0.28117528 0.90233127 0.01767696 0.75646793
```

Validating Model

```r
# table of the test set
table(test.df$RAIN)
```

```
##
## FALSE  TRUE
##  2913  2175
```

```r
nrows_prediction<-nrow(test.df)

prediction <- data.frame(c(1:nrows_prediction))
colnames(prediction) <- c("RAIN")

prediction$RAIN <- as.character(prediction$RAIN)
prediction$RAIN <- "TRUE"


prediction$RAIN[ predicted_values < 0.5] <- "FALSE"
prediction$RAIN <- as.factor(prediction$RAIN)
```

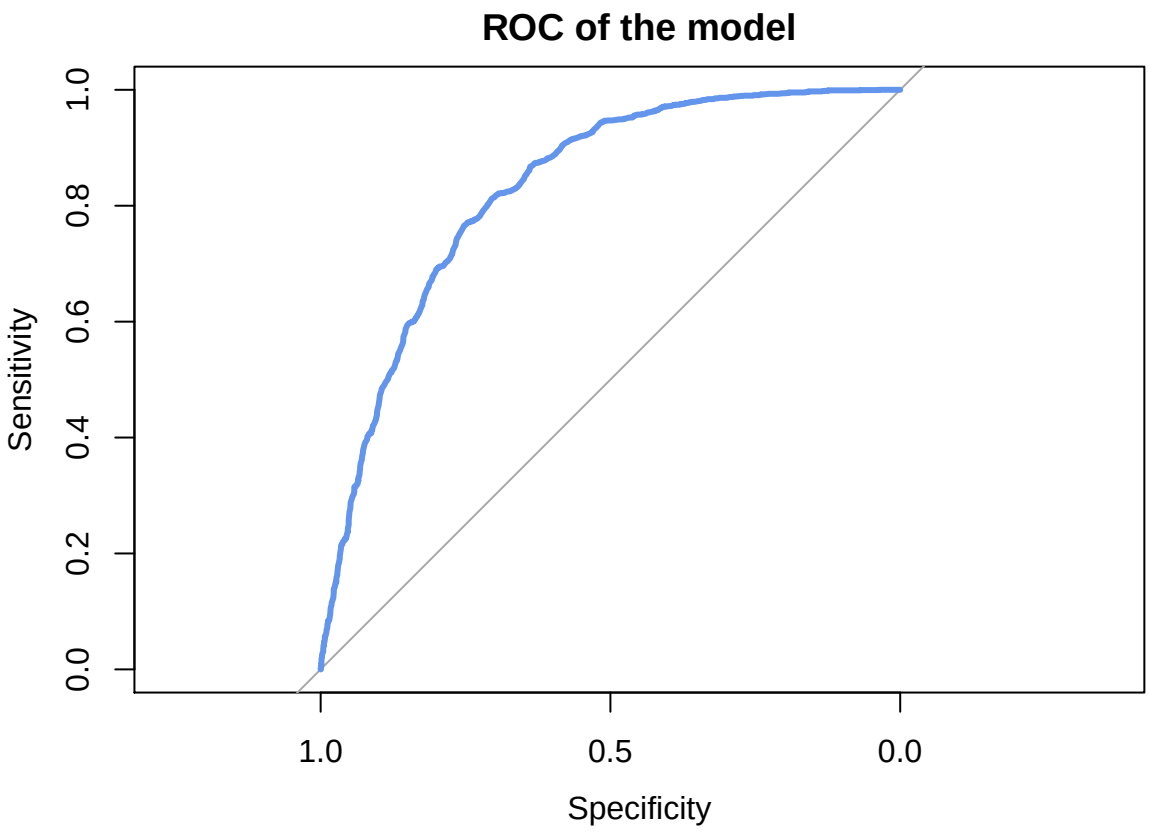
```r
table(prediction$RAIN, test.df$RAIN)
```

```
##
##         FALSE TRUE
##   FALSE  2271  640
##   TRUE    642 1535
```

```r
confusionMatrix(prediction$RAIN, test.df$RAIN)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction FALSE TRUE
##      FALSE  2271  640
##      TRUE    642 1535
##
##                Accuracy : 0.748
##                  95% CI : (0.7359, 0.7599)
##     No Information Rate : 0.5725
##     P-Value [Acc > NIR] : <2e-16
##
##                   Kappa : 0.4853
##  Mcnemar's Test P-Value : 0.9777
##
##             Sensitivity : 0.7796
##             Specificity : 0.7057
##          Pos Pred Value : 0.7801
##          Neg Pred Value : 0.7051
##              Prevalence : 0.5725
##          Detection Rate : 0.4463
##    Detection Prevalence : 0.5721
##       Balanced Accuracy : 0.7427
##
##        'Positive' Class : FALSE
##
```

```r
plot(roc(test.df$RAIN, predicted_values, direction="<"),
     col="cornflowerblue", lwd=3, main="ROC of the model", xlim=c(0.9,0),ylim=c(0,1.0))
```




```
# Call:
# roc.default(response = test.df$RAIN, predictor = predicted_values,     direction = "<")
# Data: predicted_values in 4394 controls (test.df$RAIN FALSE) < 3270 cases (test.df$RAIN TRUE).
#Area under the curve: 0.8282
```