Untitled

2024-04-03

```
set.seed(1007217101)
```

Load Data

```
dt <- read.csv("Toronto_clean.csv", header = TRUE)
dt <- subset(dt, select = -X)
dt$rained <- as.factor(dt$rained)
dt$snowed <- as.factor(dt $snowed)
dt$is_winter <- as.factor(dt $is_winter)

dt %>% glimpse()
```

```
## Rows: 4,048
## Columns: 31
## $ date_time                 <chr> "2009-01-01", "2009-01-02", "2009-01-03", "2~
## $ precipMM                  <dbl> 0.0, 0.4, 0.1, 0.2, 0.0, 0.2, 0.0, 0.2, 0.1,~
## $ maxtempC                  <int> -6, 0, -2, -2, 0, -2, 0, -3, -7, -6, -8, -4,~
## $ mintempC                  <int> -13, -3, -8, -9, -6, -7, -2, -8, -10, -10, -~
## $ totalSnow_cm              <dbl> 0.0, 0.4, 0.1, 0.0, 0.0, 0.2, 0.0, 0.2, 0.1,~
## $ sunHour                   <dbl> 6.9, 5.2, 8.7, 6.9, 8.7, 8.7, 3.4, 3.4, 8.2,~
## $ uvIndex                   <int> 2, 1, 2, 2, 2, 2, 1, 1, 1, 1, 2, 1, 1, 1, 1,~
## $ uvIndex.1                 <int> 2, 1, 2, 2, 2, 2, 1, 1, 1, 1, 2, 1, 1, 1, 1,~
## $ DewPointC                 <int> -11, -3, -6, -8, -5, -5, -2, -8, -10, -9, -1~
## $ FeelsLikeC                <int> -14, -7, -10, -9, -9, -7, -6, -13, -13, -14,~
## $ HeatIndexC                <int> -9, -2, -5, -5, -3, -3, -1, -5, -8, -8, -8, ~
## $ WindChillC                <int> -14, -7, -10, -9, -9, -7, -6, -13, -13, -14,~
## $ WindGustKmph              <int> 19, 34, 25, 16, 29, 19, 34, 34, 17, 29, 18, ~
## $ cloudcover                <int> 58, 76, 46, 52, 30, 33, 100, 81, 57, 81, 50,~
## $ humidity                  <int> 87, 88, 88, 82, 86, 83, 94, 85, 89, 92, 89, ~
## $ pressure                  <int> 1024, 1007, 1021, 1019, 1017, 1012, 989, 999~
## $ tempC                     <int> -6, 0, -2, -2, 0, -2, 0, -3, -7, -6, -8, -4,~
## $ visibilityKM              <int> 10, 8, 9, 9, 10, 9, 9, 9, 10, 8, 10, 10, 8, ~
## $ winddirDegree             <int> 214, 234, 282, 89, 264, 170, 160, 302, 203, ~
## $ windspeedKmph             <int> 13, 22, 17, 11, 19, 13, 23, 25, 11, 20, 12, ~
## $ moon_illumination_percent <int> 31, 38, 45, 52, 60, 67, 74, 82, 89, 96, 100,~
## $ moonrise                  <chr> "11:31 AM", "11:51 AM", "12:10 PM", "12:32 P~
## $ moonset                   <chr> "11:11 PM", "No moonset", "12:17 AM", "1:26 ~
## $ sunrise                   <chr> "8:51 AM", "8:51 AM", "8:51 AM", "8:51 AM", ~
## $ sunset                    <chr> "5:52 PM", "5:53 PM", "5:54 PM", "5:55 PM", ~
## $ rained                    <fct> 0, 1, 1, 1, 0, 1, 0, 1, 1, 1, 0, 0, 1, 1, 1,~
## $ snowed                    <fct> 0, 1, 1, 0, 0, 1, 0, 1, 1, 0, 0, 0, 1, 1, 1,~
## $ month                     <int> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,~
## $ season                    <chr> "winter", "winter", "winter", "winter", "win~
## $ is_winter                 <fct> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,~
## $ medTempC                  <dbl> -9.5, -1.5, -5.0, -5.5, -3.0, -4.5, -1.0, -5~
```

```r
dt %>% colnames()
```

```
##  [1] "date_time"                 "precipMM"                 
##  [3] "maxtempC"                  "mintempC"                 
##  [5] "totalSnow_cm"              "sunHour"                  
##  [7] "uvIndex"                   "uvIndex.1"                
##  [9] "DewPointC"                 "FeelsLikeC"               
## [11] "HeatIndexC"                "WindChillC"               
## [13] "WindGustKmph"              "cloudcover"               
## [15] "humidity"                  "pressure"                 
## [17] "tempC"                     "visibilityKM"             
## [19] "winddirDegree"             "windspeedKmph"            
## [21] "moon_illumination_percent" "moonrise"                 
## [23] "moonset"                   "sunrise"                  
## [25] "sunset"                    "rained"                   
## [27] "snowed"                    "month"                    
## [29] "season"                    "is_winter"                
## [31] "medTempC"
```

```r
dt %>% head(10) # 4048 rows * 31 cols
```

```
##     date_time precipMM maxtempC mintempC totalSnow_cm sunHour uvIndex uvIndex.1
## 1  2009-01-01      0.0       -6      -13          0.0     6.9       2         2
## 2  2009-01-02      0.4        0       -3          0.4     5.2       1         1
## 3  2009-01-03      0.1       -2       -8          0.1     8.7       2         2
## 4  2009-01-04      0.2       -2       -9          0.0     6.9       2         2
## 5  2009-01-05      0.0        0       -6          0.0     8.7       2         2
## 6  2009-01-06      0.2       -2       -7          0.2     8.7       2         2
## 7  2009-01-07      0.0        0       -2          0.0     3.4       1         1
## 8  2009-01-08      0.2       -3       -8          0.2     3.4       1         1
## 9  2009-01-09      0.1       -7      -10          0.1     8.2       1         1
## 10 2009-01-10      2.2       -6      -10          0.0     4.8       1         1
##    DewPointC FeelsLikeC HeatIndexC WindChillC WindGustKmph cloudcover humidity
## 1        -11        -14         -9        -14           19         58       87
## 2         -3         -7         -2         -7           34         76       88
## 3         -6        -10         -5        -10           25         46       88
## 4         -8         -9         -5         -9           16         52       82
## 5         -5         -9         -3         -9           29         30       86
## 6         -5         -7         -3         -7           19         33       83
## 7         -2         -6         -1         -6           34        100       94
## 8         -8        -13         -5        -13           34         81       85
## 9        -10        -13         -8        -13           17         57       89
## 10        -9        -14         -8        -14           29         81       92
##    pressure tempC visibilityKM winddirDegree windspeedKmph
## 1      1024    -6           10           214            13
## 2      1007     0            8           234            22
## 3      1021    -2            9           282            17
## 4      1019    -2            9            89            11
## 5      1017     0           10           264            19
## 6      1012    -2            9           170            13
## 7       989     0            9           160            23
## 8       999    -3            9           302            25
## 9      1019    -7           10           203            11
## 10     1021    -6            8            67            20
##    moon_illumination_percent moonrise    moonset sunrise  sunset rained snowed
## 1                          31 11:31 AM   11:11 PM 8:51 AM 5:52 PM      0      0
## 2                          38 11:51 AM No moonset 8:51 AM 5:53 PM      1      1
## 3                          45 12:10 PM   12:17 AM 8:51 AM 5:54 PM      1      1
## 4                          52 12:32 PM    1:26 AM 8:51 AM 5:55 PM      1      0
## 5                          60 12:57 PM    2:37 AM 8:51 AM 5:56 PM      0      0
## 6                          67  1:29 PM    3:53 AM 8:51 AM 5:57 PM      1      1
## 7                          74  2:10 PM    5:11 AM 8:51 AM 5:58 PM      0      0
## 8                          82  3:04 PM    6:28 AM 8:51 AM 5:59 PM      1      1
## 9                          89  4:11 PM    7:38 AM 8:50 AM 6:00 PM      1      1
## 10                         96  5:31 PM    8:35 AM 8:50 AM 6:01 PM      1      0
##    month season is_winter medTempC
## 1      1 winter         1     -9.5
## 2      1 winter         1     -1.5
## 3      1 winter         1     -5.0
## 4      1 winter         1     -5.5
## 5      1 winter         1     -3.0
## 6      1 winter         1     -4.5
## 7      1 winter         1     -1.0
## 8      1 winter         1     -5.5
## 9      1 winter         1     -8.5
## 10     1 winter         1     -8.0
```

Mutates

```
dt <- dt %>% mutate(log_visibility = log(visibilityKM))
# p<-ggplot(as_tibble(dt), aes(x=log_visibility, fill=rained)) +
#       geom_histogram(position="dodge", binwidth=1) +
#       labs(x="Visibility (km)", y="Count of Rainy Days")
# p
```

Split Data Train Test 8:2. But this step is not necessary in this study as we will be using lrm() to validate, which automatically conducts k-fold cross-validation. At this stage, we would just define variable spaces.

```
# train_indexes <- sample(1:nrow(dt), size = 0.8 * nrow(dt)) # 80% for training
# dt_train <- dt[train_indexes, ]
# dt_test <- dt[-train_indexes, ]
#
# # dt_train %>% glimpse() #3238
# # dt_test %>% glimpse() #810



full_predictors <- c('medTempC' , "humidity",
                      'cloudcover', 'windspeedKmph' , 'log_visibility' , 'pressure' ,
                      'DewPointC' , 'sunHour' , 'uvIndex' , 'WindGustKmph' ,
                      'winddirDegree' ,'moon_illumination_percent' ,'FeelsLikeC')
# full_predictors <- c(full_predictors,'rained')
full_predictors
```

```
##  [1] "medTempC"                  "humidity"
##  [3] "cloudcover"                "windspeedKmph"
##  [5] "log_visibility"            "pressure"
##  [7] "DewPointC"                 "sunHour"
##  [9] "uvIndex"                   "WindGustKmph"
## [11] "winddirDegree"             "moon_illumination_percent"
## [13] "FeelsLikeC"
```

```
# dt['rained'] is the response variable.
```

Correlation Matrix to first select candidate featrues (Not included)

```
# numer = c('medTempC' , "humidity" ,
#                         'cloudcover', 'windspeedKmph' , 'visibilityKM' , 'pressure' ,
#                         'DewPointC' , 'sunHour' , 'uvIndex' , 'WindGustKmph' ,
#                         'winddirDegree' ,'moon_illumination_percent' ,'FeelsLikeC')
#
# cmatrix <- cor(dt[numer])
# dt[numer]
# #col <- colorRampPalette(c())
# corrplot(cmatrix,addCoef.col="grey",number.cex=0.5,tl.cex=0.6)
```

Full model)

```
model1 <- glm(rained ~ .,
              family = binomial(link = logit),
              data = dt[c(full_predictors,'rained')])

model1 %>% summary()
```

```
##
## Call:
## glm(formula = rained ~ ., family = binomial(link = logit), data = dt[c(full_predictors,
##     "rained")])
##
## Coefficients:
##                             Estimate Std. Error z value Pr(>|z|)
## (Intercept)               23.3099429  7.8430023   2.972  0.00296 **
## medTempC                   0.1189599  0.0509051   2.337  0.01944 *
## humidity                   0.0221876  0.0143781   1.543  0.12279
## cloudcover                 0.0510064  0.0029572  17.248  < 2e-16 ***
## windspeedKmph             -0.0450805  0.0221528  -2.035  0.04185 *
## log_visibility            -5.3255504  0.5393381  -9.874  < 2e-16 ***
## pressure                  -0.0166182  0.0072993  -2.277  0.02280 *
## DewPointC                 -0.0546610  0.0648344  -0.843  0.39918
## sunHour                    0.0062425  0.0210027   0.297  0.76630
## uvIndex                    0.3244435  0.0667044   4.864 1.15e-06 ***
## WindGustKmph               0.0302502  0.0128800   2.349  0.01884 *
## winddirDegree              0.0001195  0.0005531   0.216  0.82890
## moon_illumination_percent  0.0006697  0.0012483   0.536  0.59161
## FeelsLikeC                -0.0229457  0.0515653  -0.445  0.65633
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
```

```
##
##     Null deviance: 5592.3  on 4047  degrees of freedom
## Residual deviance: 4034.6  on 4034  degrees of freedom
## AIC: 4062.6
##
## Number of Fisher Scoring iterations: 5
```

Perform AIC based stepwise selection

```
## Stepwise elimination based on AIC ##
sel.var.aic <- step(model1, trace = 0, k = 2, direction = "both")
select_var_aic<-attr(terms(sel.var.aic), "term.labels")
select_var_aic
```

```
## [1] "medTempC"       "humidity"       "cloudcover"     "windspeedKmph"
## [5] "log_visibility" "pressure"       "DewPointC"      "uvIndex"
## [9] "WindGustKmph"
```

Now perform BIC based selection

```
## Stepwise elimination based on AIC ##
sel.var.bic <- step(model1, trace = 0, k = log(nrow(dt)), direction = "both")
select_var_bic<-attr(terms(sel.var.bic), "term.labels")
select_var_bic
```

```
## [1] "medTempC"       "cloudcover"     "windspeedKmph"  "log_visibility"
## [5] "pressure"       "uvIndex"        "WindGustKmph"
```

Lasso Selection

Similar process: first we fit model with differnt lambdas

```
X <- as.matrix(dt[full_predictors])
Y <-dt$rained

# grid = 10^seq(10,-2,length = 100)

cv.out <- cv.glmnet(X,Y,alpha=1, family= 'binomial') # 10 fold cross validation
bestlam <- cv.out$lambda.min
bestlam
```

```
## [1] 0.0007576844
```

```
#Then we look at the best model i.e. who has the least lambda. And extract its variables.


lasso.mod <- glmnet(X,Y,alpha=1,lambda=bestlam,family= 'binomial') #get the model under the best lambda
coefs <-coef(lasso.mod)[,1]
coefs<-coefs[coefs != 0]
a <-coefs %>% as.data.frame()
select_var_lasso =a %>% rownames()
select_var_lasso=select_var_lasso[!select_var_lasso %in% c("(Intercept)")]
select_var_lasso
```

```
##  [1] "medTempC"                  "humidity"
##  [3] "cloudcover"                "windspeedKmph"
##  [5] "log_visibility"            "pressure"
##  [7] "sunHour"                   "uvIndex"
##  [9] "WindGustKmph"              "winddirDegree"
## [11] "moon_illumination_percent"
```

A Helper Function for df beta:

```
Draw_dfbetas <- function(Features, y='rained'){
  modeltemp <- glm(rained ~ .,
              family = binomial(link = logit), data = dt[Features])
  # modeltemp %>% summary
  df.final <- dfbetas(modeltemp)

  for (feature in Features[!Features %in% 'rained']) {

    # df.final %>% head

    par(family = 'serif')
    plot(dt[,feature], df.final[,feature], xlab=feature,
         ylab='dfbeta')
    lines(lowess(dt[,feature], df.final[,feature] ), lwd=2, col='blue')
    abline(h=0, lty='dotted')
    abline(h=-2/sqrt(nrow(df.final)), lty='dotted')
    abline(h=2/sqrt(nrow(df.final)), lty='dotted')
  }
}

# Draw_dfbetas(Features = candidates_6)
```

Helper Function for Deviance residuals

```
# ## Plot the deviance residuals ##
# res.dev <- residuals(model2, type = "deviance")
# par(family = 'serif')
# plot(dt[,'WindGustKmph'], res.dev, xlab='WindGustKmph',
#      ylab='Deviance Residuals')
# lines(lowess(dt[,'WindGustKmph'], res.dev), lwd=2, col='blue')
# abline(h=0, lty='dotted')
#




Draw_devianceResidual <- function(Features, y='rained'){
  modeltemp <- glm(rained ~ .,
              family = binomial(link = logit), data = dt[Features])
  # modeltemp %>% summary
  df.final <- dfbetas(modeltemp)

  for (feature in Features[!Features %in% 'rained']) {

    res.dev <- residuals(modeltemp, type = "deviance")
    par(family = 'serif')
    plot(dt[,feature], res.dev, xlab=feature,
         ylab='Deviance Residuals')
    lines(lowess(dt[,feature], res.dev), lwd=2, col='blue')
    abline(h=0, lty='dotted')
  }
}
# Draw_devianceResidual(candidates_6)
```

Define Variable Space for the final model

```
candidates_ab <- intersect(select_var_aic, select_var_bic)
candidates_abl <- intersect(candidates_ab, select_var_lasso)
candidates_bl <- intersect(select_var_bic, select_var_lasso)

candidates_1 <- candidates_abl[!candidates_abl %in% c('uvIndex','WindGustKmph')]
candidates_1 = c(candidates_1,'humidity' )

candidates_2 = candidates_1 = c(candidates_1,'is_winter','snowed' )
candidates_2 <- candidates_2[!candidates_2 %in% c('windspeedKmph','snowed')]
# dt$windspeedKmph
# candidates_2 <- sel.var.b2[!sel.var.b2 %in% c('WindGustKmph')]
# candidates_2 = c(candidates_2,'humidity' )

# 'WindGustKmph','uvIndex'
# # candidates_6 = c(candidates_4,'log_visibility' )
# # candidates_6= candidates_6[!candidates_6 %in% c("visibilityKM")]
#
# candidates_7 = c(full_predictors,'rained','log_visibility')
# candidates_7 = candidates_7[!candidates_7 %in% c('FeelsLikeC',"DewPointC",'moon_illumination_percent'
```

Plot the dfbetas and deviance residuals

Fit the initial final model, and plot deviance betas

```
ft = candidates_2
ftr = c(ft,'rained')

modelF <- glm(rained ~ .,family = binomial(link = logit), data = dt[ftr])
# Draw_dfbetas(Features = ftr)
# Draw_devianceResidual(ftr)

modelF %>% summary()
```

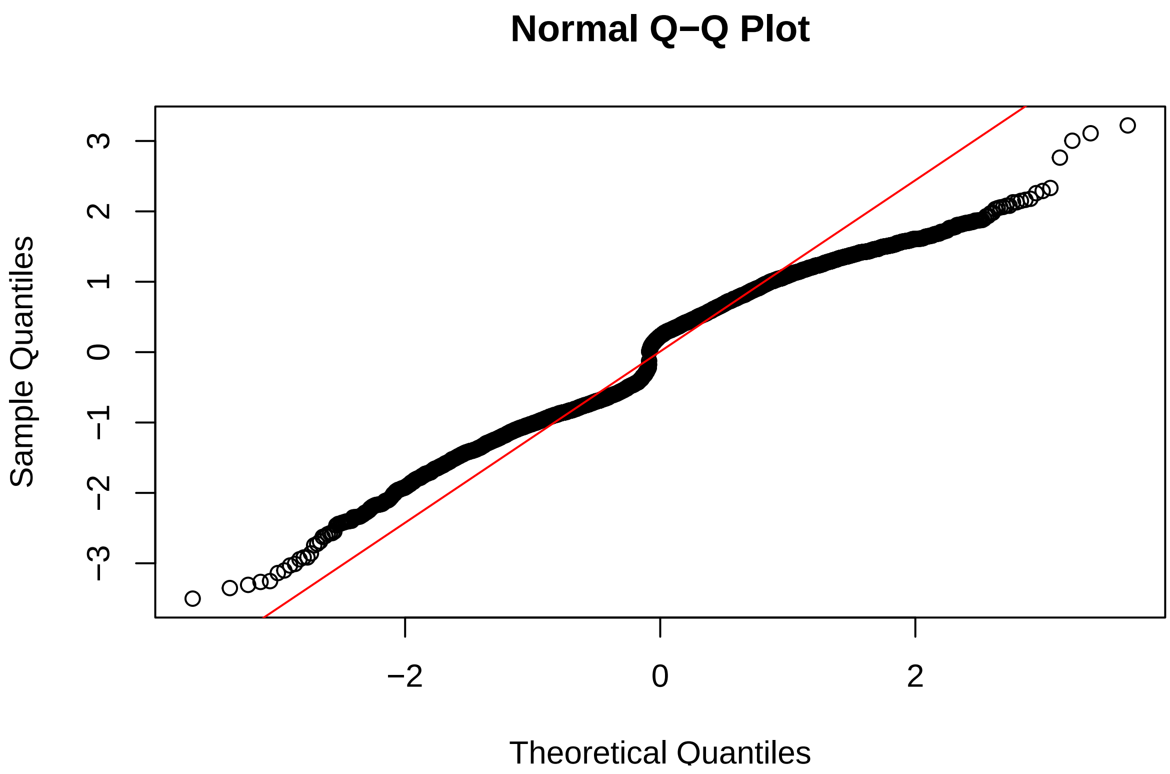
```
##
## Call:
## glm(formula = rained ~ ., family = binomial(link = logit), data = dt[ftr])
##
## Coefficients:
##                 Estimate Std. Error z value Pr(>|z|)
## (Intercept)    26.627925   7.009882   3.799 0.000146 ***
## medTempC        0.076947   0.006865  11.208  < 2e-16 ***
## cloudcover      0.045117   0.002249  20.061  < 2e-16 ***
## log_visibility -5.273507   0.541511  -9.739  < 2e-16 ***
## pressure       -0.018329   0.006592  -2.780 0.005431 **
## humidity        0.019354   0.004780   4.048 5.15e-05 ***
## is_winter1     -0.287755   0.132951  -2.164 0.030437 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 5592.3  on 4047  degrees of freedom
## Residual deviance: 4063.0  on 4041  degrees of freedom
## AIC: 4077
##
## Number of Fisher Scoring iterations: 5
```

```
modelF%>% vif()
```

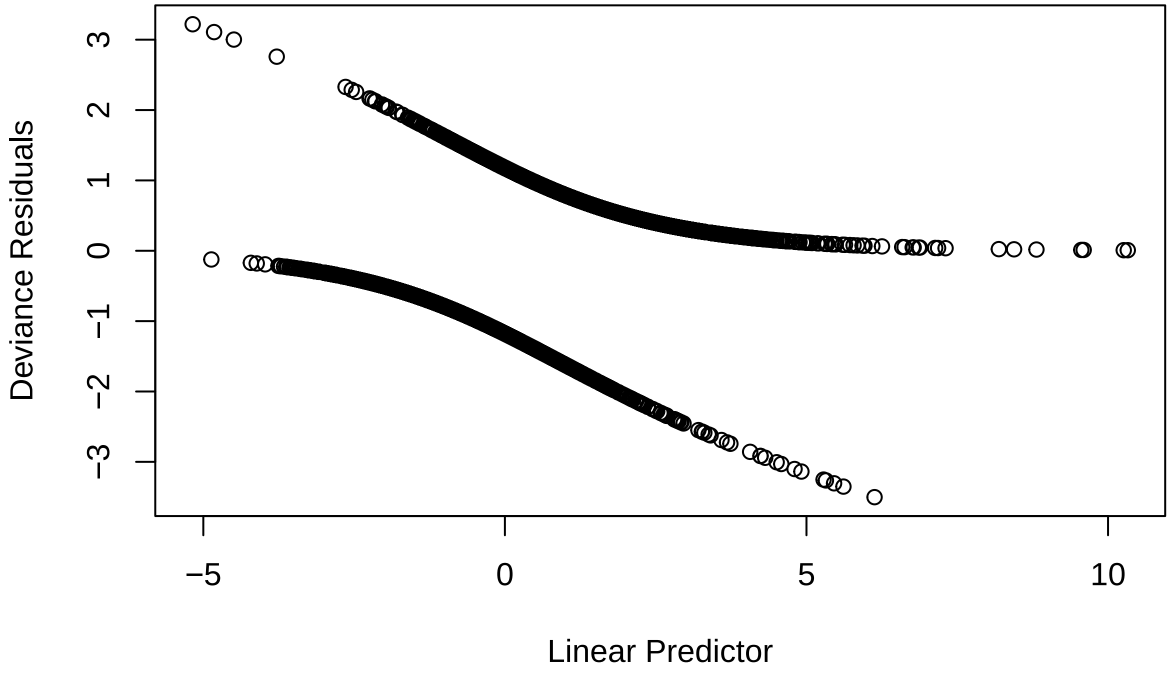
```
##       medTempC     cloudcover log_visibility       pressure       humidity
##      19.222823      16.138246      24.794427       9.484808       9.494927
##     is_winter1
##      15.986083
```

Check and remove outliers

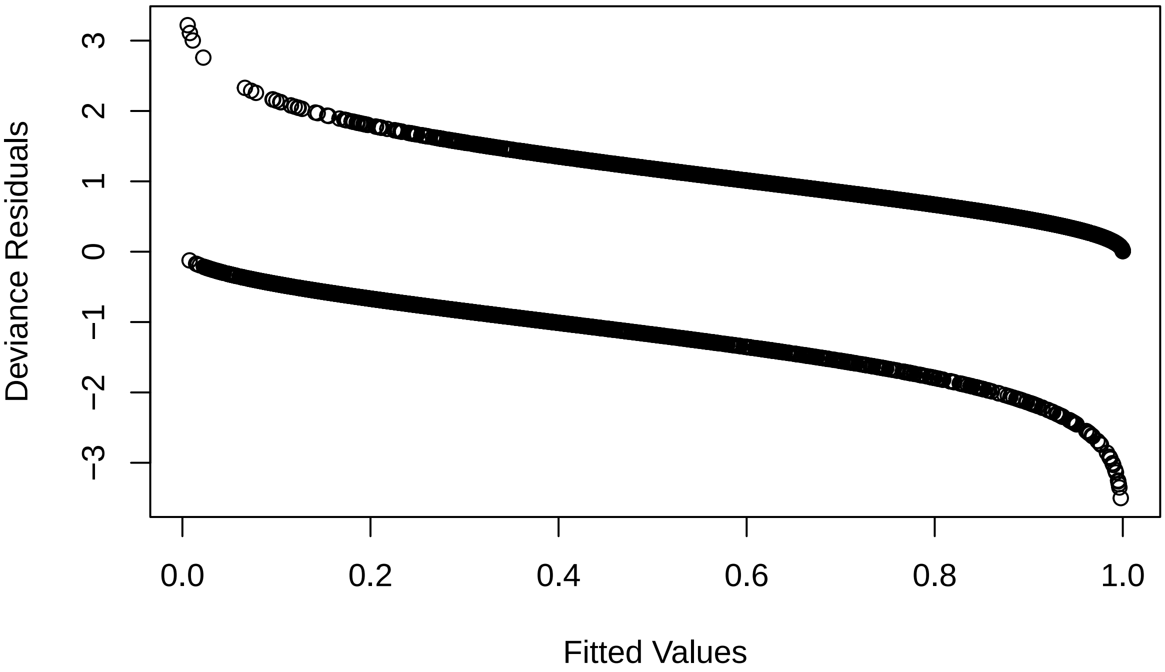
```
# Extract linear predictors (eta)
eta <- predict(modelF, type = "link")

# Calculate standardized residuals
residuals_standardized <- rstandard(modelF)

# Create QQ plot
qqnorm(residuals_standardized)
qqline(residuals_standardized, col = "red")
```



```
######
plot(eta, resid(modelF, type = "deviance"),
     xlab = "Linear Predictor", ylab = "Deviance Residuals", main = "")
```



```
# Deviance Residuals vs. Fitted Values
plot(fitted(modelF), resid(modelF, type = "deviance"),
     xlab = "Fitted Values", ylab = "Deviance Residuals", main = "")
```



```r
#identify potential outliers with absolute standardized residuals greater than 2
potential_outliers <- which(abs(residuals_standardized) > 2)

#Find  potential outliers
outlier_data <- modelF$data[potential_outliers, ]
outlier_data
```

```
##      medTempC cloudcover log_visibility pressure humidity is_winter rained
## 7        -1.0        100       2.197225      989       94         1      0
## 44       -4.0         21       2.302585     1018       70         1      1
## 56       -1.5         24       2.302585     1024       81         1      1
## 85        6.0         82       1.609438     1011       95         1      0
## 94        2.0         87       2.302585     1004       89         0      0
## 97       -1.0         99       2.302585      998       78         0      0
## 129      11.0         83       1.609438     1003       92         0      0
## 160      12.0         40       1.945910     1011       85         0      0
## 184      16.5         62       2.079442     1012       86         0      0
## 241      18.5         57       2.197225     1005       86         0      0
## 280      12.0         76       2.302585     1002       82         0      0
## 302      10.0         62       1.791759     1024       91         0      0
## 319       8.5         46       1.945910     1015       87         0      0
## 337       5.5         86       2.302585     1000       88         1      0
## 390       2.5         93       1.791759      991       93         1      0
## 419      -0.5         91       1.609438     1008       95         1      0
## 426      -0.5         85       2.079442     1016       94         1      0
## 433       4.5         42       1.386294     1015       95         1      0
## 463       8.0         90       2.197225     1001       95         0      0
## 499      11.0         55       1.945910     1016       83         0      0
## 508      15.0         51       1.945910     1020       88         0      0
## 560      22.0         40       1.945910     1016       84         0      0
## 565      21.0         42       2.079442     1009       84         0      0
## 630      17.0         41       2.079442     1015       89         0      0
## 700       4.5         82       2.197225     1006       91         1      0
## 768       -3.5         83       1.945910     1012       94         1      0
## 824        7.0         91       2.302585      994       98         0      0
## 830        9.5         64       1.791759     1010       92         0      0
## 853        4.5         88       2.197225     1020       92         0      0
## 866        7.0         86       2.197225     1012       89         0      0
## 877       10.0         87       2.079442     1014       94         0      0
## 906       17.5         78       2.302585     1008       88         0      0
## 1095       2.0         77       2.079442     1011       92         1      0
## 1107       3.5         92       2.079442      996       95         1      0
## 1161      -1.5         32       2.302585     1029       67         1      1
## 1175      11.5         54       1.386294     1022       95         1      0
## 1240      17.0         40       1.945910     1014       88         0      0
## 1393      11.0         77       2.302585     1018       90         0      0
## 1473       6.5         88       2.302585     1013       98         1      0
## 1520      -1.0         88       1.791759     1003       95         1      0
## 1554      -1.5         20       2.302585     1026       67         0      1
## 1581      12.0         65       1.945910     1019       91         0      0
## 1592       9.5         68       2.197225     1007       85         0      0
## 1619      13.0         84       2.079442     1011       91         0      0
## 1623      16.5         68       2.079442     1007       89         0      0
## 1766      12.0         77       2.302585      995       82         0      0
## 1892      -4.0         50       1.609438     1018       94         1      0
## 1895       0.5         84       2.079442     1003       95         1      0
## 1920       2.5         91       2.302585     1004       96         0      0
## 1988      17.5         73       1.945910     1011       91         0      0
## 1989      19.5         58       1.791759     1008       91         0      0
## 2002      18.5         73       1.945910     1012       92         0      0
## 2060      20.0         57       2.197225     1015       87         0      0
## 2062      20.5         35       1.945910     1020       84         0      0
## 2134      11.5         82       2.302585     1013       86         0      0
## 2160       8.0         80       2.302585     1008       95         0      0
## 2195       1.5         91       2.197225     1005       91         1      0
## 2357      17.5         85       1.791759     1013       94         0      0
## 2370      15.0         99       2.302585     1007       91         0      0
## 2506      10.0         60       2.079442     1014       86         0      0
## 2508       7.5         86       2.302585     1003       73         0      0
## 2526       5.5         88       2.079442     1014       90         1      0
## 2552       3.0         87       2.079442     1018       84         1      0
## 2596       0.5        100       2.302585     1000       87         1      0
## 2603      -1.5         94       2.197225     1008       89         1      0
## 2609      -3.5         38       2.302585     1025       63         1      1
## 2630       5.5        100       2.302585     1008       96         1      0
## 2643       3.5         13       2.302585     1019       84         1      1
## 2654       2.5         89       2.197225      996       87         0      0
## 2785      22.5         69       2.302585     1014       84         0      0
## 2851       9.5        100       2.302585     1013       85         0      0
## 2940       2.0         92       2.197225     1015       93         1      0
## 2948       2.5         96       2.197225      998       88         1      0
## 2950      -1.0         99       2.197225     1002       77         1      0
## 2985      -8.5         36       2.197225     1039       58         1      1
## 3012       2.5        100       2.302585     1009       95         1      0
## 3033       7.0         75       2.079442     1012       85         0      0
## 3047       6.0        100       2.302585     1004       96         0      0
## 3068      12.5         92       1.945910     1007       92         0      0
## 3242      6.0         94       2.197225     1010       74         0      0
## 3310      0.5         85       1.945910     1002       87         1      0
## 3567     15.0         91       2.302585     1024       80         0      0
## 3649      7.5         90       2.302585     1008       90         1      0
## 3674    -10.0         44       2.302585     1028       76         1      1
## 3692     -9.0         61       2.890372     1039       55         1      1
## 3693     -7.0         54       2.484907     1035       67         1      1
## 3699     -5.0         54       2.639057     1012       74         1      1
## 3700     -8.0         73       2.639057     1018       73         1      1
## 3706     -1.5         35       2.995732     1023       81         1      1
## 3729      0.5         68       2.944439     1026       57         1      1
## 3739      4.0         81       2.890372     1018       70         1      1
## 3740      7.0         32       2.890372     1019       57         1      1
## 3817     16.5         48       2.772589     1011       58         0      1
## 3964     -1.0         24       2.302585     1026       60         0      1
```

```r
#Remove them:
dt2 <- modelF$data[-potential_outliers, ]
```

Fit again, as the final Model:

```r
ft = candidates_2
ftr = c(ft,'rained')
ftr
```

```
## [1] "medTempC"       "cloudcover"     "log_visibility" "pressure"
## [5] "humidity"       "is_winter"      "rained"
```

```r
modelFF <- glm(rained ~ .,family = binomial(link = logit), data = dt2[ftr])
modelFF %>% summary()
```

```
##
## Call:
## glm(formula = rained ~ ., family = binomial(link = logit), data = dt2[ftr])
##
## Coefficients:
##                 Estimate Std. Error z value Pr(>|z|)
## (Intercept)     62.815242   8.296131   7.572 3.69e-14 ***
## medTempC         0.094736   0.007788  12.165  < 2e-16 ***
## cloudcover       0.052145   0.002646  19.709  < 2e-16 ***
## log_visibility -14.858063   0.985487 -15.077  < 2e-16 ***
## pressure        -0.033469   0.007555  -4.430 9.42e-06 ***
## humidity         0.029245   0.005432   5.384 7.27e-08 ***
## is_winter1      -0.601526   0.153177  -3.927 8.60e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 5452.5  on 3953  degrees of freedom
## Residual deviance: 3320.5  on 3947  degrees of freedom
## AIC: 3334.5
##
## Number of Fisher Scoring iterations: 6
```

```
modelFF%>% vif()
```

```
##        medTempC     cloudcover log_visibility       pressure       humidity
##        24.36311       21.54563       73.50999       11.86848       11.79543
##      is_winter1
##        20.63976
```

Model calibration with cross-validation and bootstrap.

Plus the QQ-Plot and Deviance Residual plot, which will be cobined to display in the report.

```
library(rms)
```

```
## Loading required package: Hmisc
```

```
##
## Attaching package: 'Hmisc'
```

```
## The following objects are masked from 'package:dplyr':
##
##     src, summarize
```

```
## The following objects are masked from 'package:base':
##
##     format.pval, units
```

```
##
## Attaching package: 'rms'
```

```
## The following object is masked from 'package:faraway':
##
##     vif
```

```
## The following objects are masked from 'package:car':
##
##     Predict, vif
```

```
par(mfrow = c(2,2))

## Fit the model with lrm from rms package ##
lrm.final <- lrm(rained ~ ., data = dt2[ftr], x =TRUE, y = TRUE, model= T)
# cross.calib <- calibrate(lrm.final, method="crossvalidation", B=10) # model calibration
cross.calib <- calibrate(lrm.final, method="crossvalidation", B=10) # model calibration
plot(cross.calib, las=1, xlab = "Predicted Probability")
```

```
##
## n=3954   Mean absolute error=0.021   Mean squared error=0.00057
## 0.9 Quantile of absolute error=0.039
```

```
## Discrimination with ROC curve

# library(pROC)
p <- predict(lrm.final, type = "fitted")

roc_logit <- roc(dt2$rained ~ p)
```

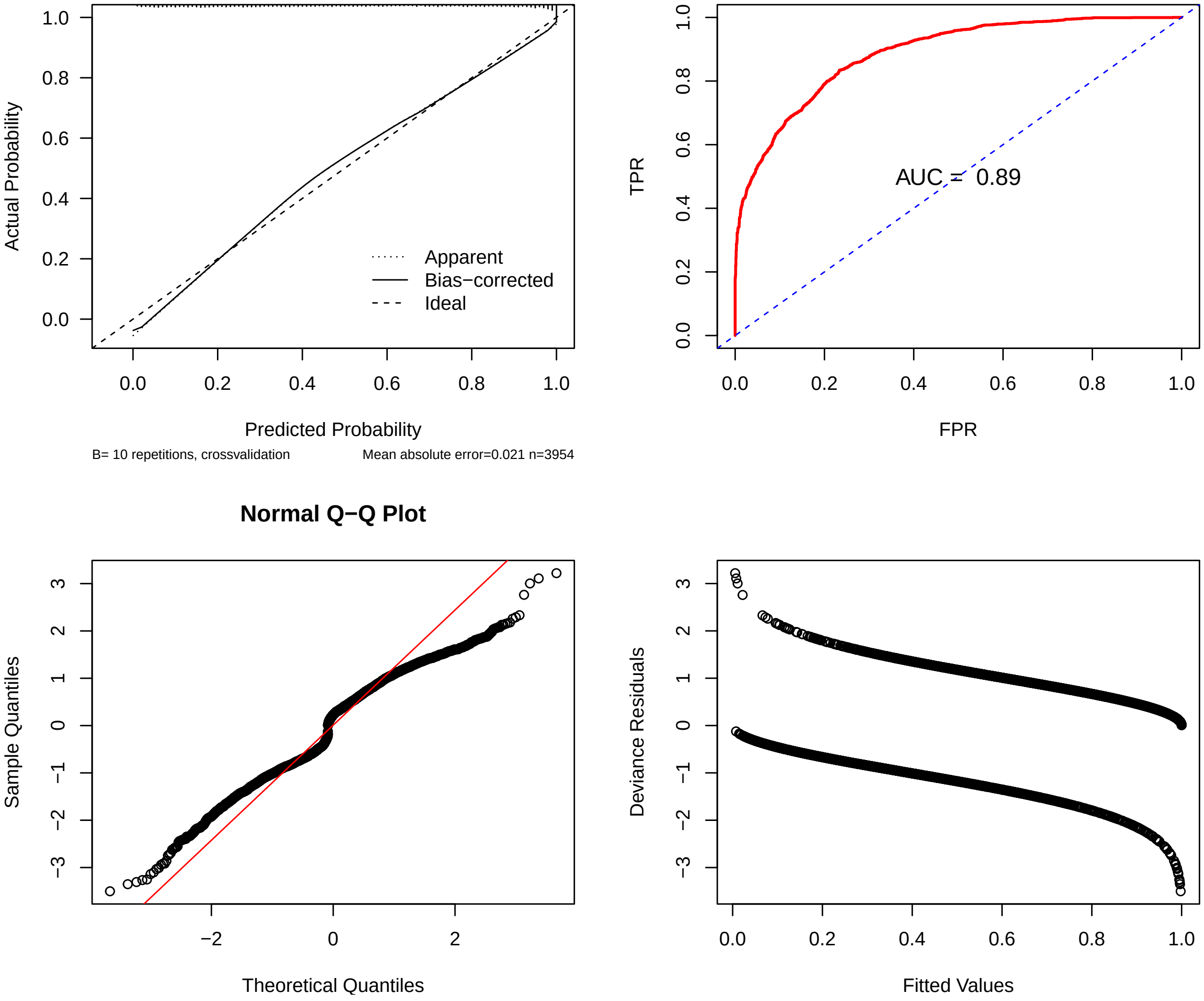
```
## Setting levels: control = 0, case = 1
```

```
## Setting direction: controls < cases
```

```r
## The True Positive Rate ##
TPR <- roc_logit$sensitivities
## The False Positive Rate ##
FPR <- 1 - roc_logit$specificities

plot(FPR, TPR, xlim = c(0,1), ylim = c(0,1), type = 'l', lty = 1, lwd = 2,col = 'red')
abline(a = 0, b = 1, lty = 2, col = 'blue')
text(cex = 1.2,0.5,0.5,label = paste("AUC = ", round(auc(roc_logit),2)))

auc(roc_logit)
```

```
## Area under the curve: 0.8862
```

```r
qqnorm(residuals_standardized)
qqline(residuals_standardized, col = "red")

# Deviance Residuals vs. Fitted Values
plot(fitted(modelF), resid(modelF, type = "deviance"),
     xlab = "Fitted Values", ylab = "Deviance Residuals", main = "")
```



Generate summary table for final data. Exported and pasted into report document.

```
# install.packages("psych")
library(psych)
```

```
##
## Attaching package: 'psych'

## The following object is masked from 'package:Hmisc':
##
##     describe

## The following object is masked from 'package:faraway':
##
##     logit

## The following object is masked from 'package:car':
##
##     logit

## The following objects are masked from 'package:ggplot2':
##
##     %+%, alpha
```

```
# write.csv(describe(dt2), file = "summary_table.csv", row.names = T)

mean(dt2$is_winter==1)
```

```
## [1] 0.3340921
```

```
# modelFF$coefficients
```

Summary table for all candidate models

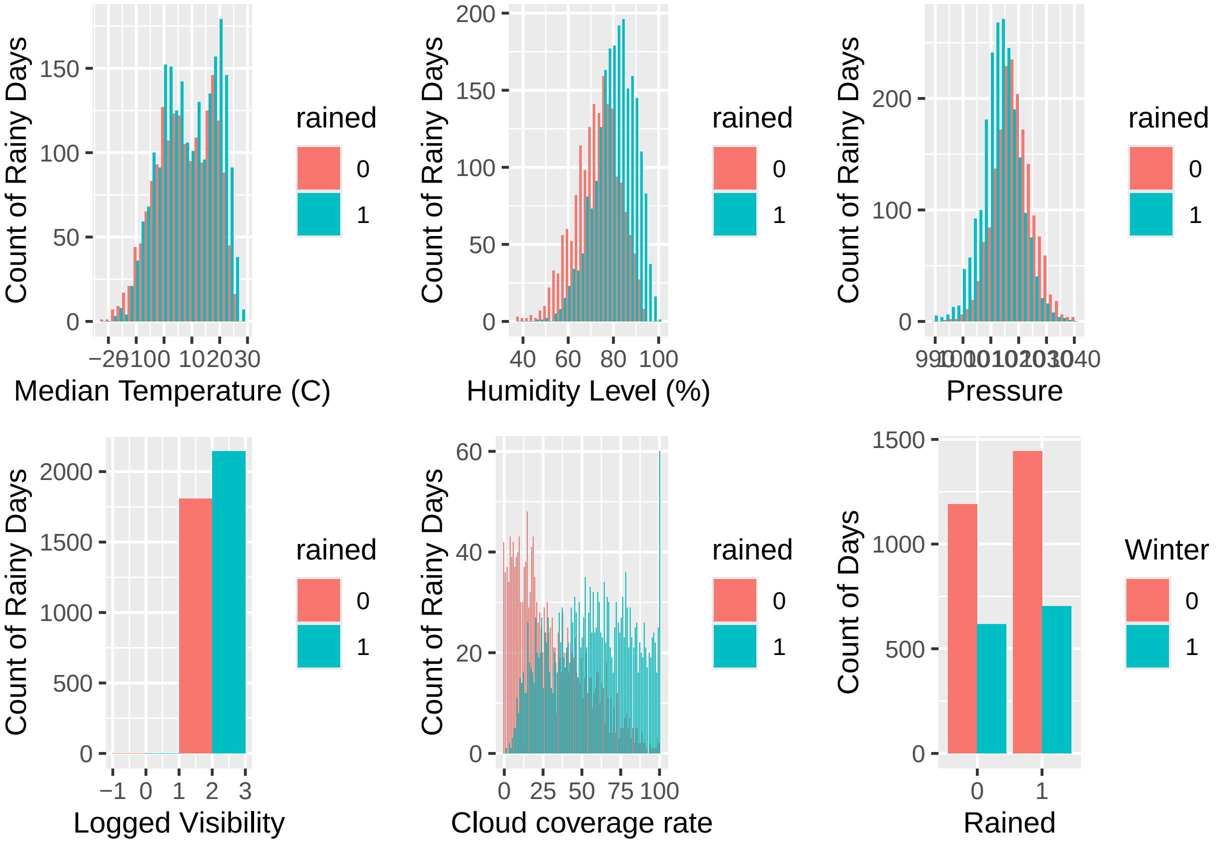
```
suppressWarnings(suppressMessages({
  modelA<-glm(rained ~ .,family = binomial(link = logit), data = dt[c(select_var_aic,'rained')])
  modelB<-glm(rained ~ .,family = binomial(link = logit), data = dt[c(select_var_bic,'rained')])
  modelL<-glm(rained ~ .,family = binomial(link = logit), data = dt[c(select_var_lasso,'rained')])

  # modelsn <- glm(rained ~ .,family = binomial(link = logit), data = dt[c(candidates_2,'rained')])

  # install.packages('stargazer')
  library(stargazer)

  stargazer(modelFF, modelA, modelB, modelL, type='text', digits = 4, title = 'Table 2: Summary of All 
}))
```

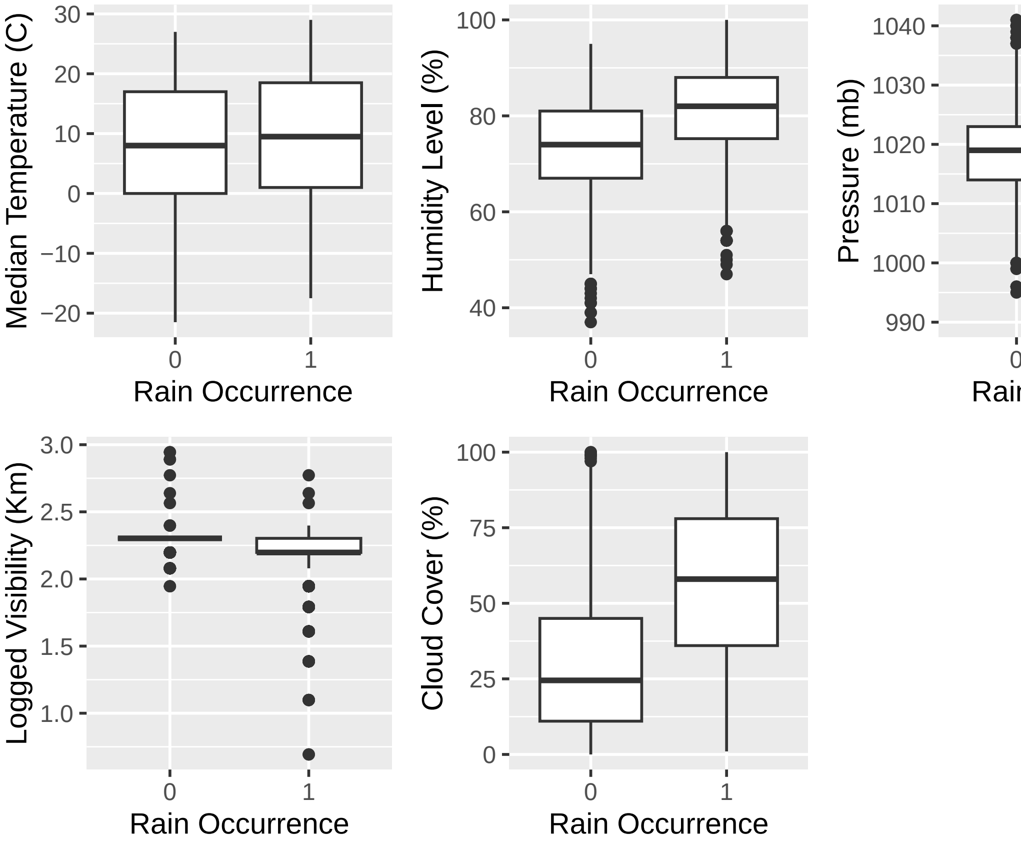
```
##
## Table 2: Summary of All Candidate Models
## ==============================================================================
##                                                  Models
##                           Final Model AIC Selected BIC Selected Lasso Selected
##                               (1)         (2)          (3)           (4)
## ------------------------------------------------------------------------------
## medTempC                   0.0947***    0.1083**    0.0322***     0.0381***
##                            (0.0078)     (0.0466)     (0.0105)      (0.0110)
```

```
##
## cloudcover                 0.0521***   0.0506***    0.0531***     0.0518***
##                            (0.0026)     (0.0027)     (0.0025)      (0.0029)
##
## windspeedKmph                          -0.0419**    -0.0598***    -0.0409**
##                                         (0.0192)     (0.0170)      (0.0198)
##
## log_visibility            -14.8581***  -5.3166***   -5.5069***    -5.3299***
##                            (0.9855)     (0.5388)     (0.5323)      (0.5398)
##
## pressure                  -0.0335***   -0.0173**    -0.0206***    -0.0166**
##                            (0.0076)     (0.0071)     (0.0069)      (0.0073)
##
## DewPointC                               -0.0735
##                                         (0.0470)
##
## sunHour                                                              0.0066
##                                                                    (0.0209)
##
## uvIndex                                0.3319***    0.3558***     0.3284***
##                                         (0.0646)     (0.0638)      (0.0667)
##
## WindGustKmph                            0.0300**    0.0419***      0.0309**
##                                         (0.0124)     (0.0113)      (0.0126)
##
## winddirDegree                                                       0.0002
##                                                                    (0.0006)
##
## moon_illumination_percent                                           0.0005
##                                                                    (0.0012)
##
## humidity                   0.0292***    0.0258**                   0.0110**
##                            (0.0054)     (0.0110)                   (0.0054)
##
## is_winter1                -0.6015***
##                            (0.1532)
##
## Constant                  62.8152***   23.7562***   29.6681***    24.2718***
##                            (8.2961)     (7.6060)     (7.2180)      (7.7808)
##
## N                            3,954       4,048        4,048         4,048
## Log Likelihood            -1,660.2630 -2,017.5950 -2,020.7940   -2,018.5990
## Akaike Inf. Crit.         3,334.5260   4,055.1910   4,057.5880    4,061.1980
## ==============================================================================
## Notes:                                  ***Significant at the 1 percent level.
##                                          **Significant at the 5 percent level.
##                                          *Significant at the 10 percent level.
```

The following are EDA plots, which will be displayed in appendix of the report.

```
suppressWarnings(suppressMessages({

p1 <- ggplot(as_tibble(dt2), aes(x=medTempC, fill=rained)) +
      geom_histogram(position="dodge", binwidth=2) +
      labs(x="Median Temperature (C)", y="Count of Rainy Days")

p2 <- ggplot(as_tibble(dt2), aes(x=humidity, fill=rained)) +
      geom_histogram(position="dodge", binwidth=2) +
      labs(x="Humidity Level (%)", y="Count of Rainy Days")

p3 <- ggplot(as_tibble(dt2), aes(x=pressure, fill=rained)) +
      geom_histogram(position="dodge", binwidth=2) +
      labs(x="Pressure", y="Count of Rainy Days")

p4 <- ggplot(as_tibble(dt2), aes(x=log_visibility, fill=rained)) +
      geom_histogram(position="dodge", binwidth=2) +
      labs(x="Logged Visibility", y="Count of Rainy Days")

p5 <- ggplot(as_tibble(dt2), aes(x=cloudcover , fill=rained)) +
      geom_histogram(position="dodge", binwidth=1) +
      labs(x="Cloud coverage rate", y="Count of Rainy Days")

bar1 <- ggplot(dt2, aes(x=rained, fill=is_winter)) +
    geom_bar(position="dodge", binwidth=5)+
    labs(x="Rained", y="Count of Days", fill="Winter")

grid.arrange(p1, p2, p3, p4, p5, bar1, nrow=2)
}))
```





### Boxplots for numeric variables

```
# grid.arrange(p1, p2, p3, p4, p5, bar1,
#                pb1, pb2, pb3, pb4, pb5, nrow=2)
```