Bike Analysis Variable Selection

2023-11-11

Overview

Objective: Perform variable selection on

The response variable is:
Y (Cnt): Total bikes rented by both casual & registered users together

The predicting variables are:
X_1 (Instant): Record index
X_2 (Dteday): Day on which the observation is made
X_3 (Season): Season which the observation is made (1 = Winter, 2 = Spring, 3 = Summer, 4 = Fall)
X_4 (Yr): Year on which the observation is made
X_5 (Mnth): Month on which the observation is made
X_6 (Hr): Day on which the observation is made (0 through 23)
X_7 (Holiday): Indictor of a public holiday or not (1 = public holiday, 0 = not a public holiday)
X_8 (Weekday): Day of week (0 through 6)
X_9 (Working day): Indicator of a working day (1 = working day, 0 = not a working day)
X_{10} (Weathersit): Weather condition (1 = Clear, Few clouds, Partly cloudy, Partly cloudy, 2 = Mist & Cloudy, Mist & Broken clouds, Mist & Few clouds, Mist, 3 = Light Snow, Light Rain, Thunderstorm & Scattered clouds, Light Rain & Scattered clouds, 4 = Heavy Rain, Ice Pallets, Thunderstorm & Mist, Snow & Fog)
X_{11} (Temp): Normalized temperature in Celsius
X_{12} (Atemp): Normalized feeling temperature in Celsius
X_{13} (Hum): Normalized humidity
X_{14} (Windspeed): Normalized wind speed
X_{15} (Casual): Bikes rented by casual users in that hour
X_{16} (Registered): Bikes rented by registered users in that hour

```r
# Bike Sharing DC
# We have analyzed in Model 2 with Mult Regr Model and Model 3 with Poission, w/ Poisson showing better
gtblue = rgb(0, 48, 87, maxColorValue = 255)
techgold = rgb(179, 163, 105, maxColorValue = 255)
buzzgold = rgb(234, 170, 0, maxColorValue = 255)
bobbyjones = rgb(55, 113, 23, maxColorValue = 255)
# Read the data using read.csv or Import Manually
data = read.csv("Bikes.csv")
# Show the number of observations

obs = nrow(data)
cat("There are", obs, "observations in the data")
```

```
## There are 17379 observations in the data
```

```r
## Preparing the data

# Set a seed for reproducibility
set.seed(9)

# Remove the irrelevant columns
clean_data = data[-c(1,2,9,15,16)]

# Convert the numerical categorical variables to predictors
clean_data$season = as.factor(clean_data$season)
clean_data$yr = as.factor(clean_data$yr)
clean_data$mnth = as.factor(clean_data$mnth)
clean_data$hr = as.factor(clean_data$hr)
clean_data$holiday = as.factor(clean_data$holiday)
clean_data$weekday = as.factor(clean_data$weekday)
clean_data$weathersit = as.factor(clean_data$weathersit)



model_bikes = glm(cnt~., data=clean_data, family = "poisson")
summary(model_bikes)
```

```
##
## Call:
## glm(formula = cnt ~ ., family = "poisson", data = clean_data)
##
## Coefficients:
##              Estimate Std. Error  z value Pr(>|z|)
## (Intercept)  2.917366   0.006790  429.679   <2e-16 ***
## season2      0.274074   0.003680   74.486   <2e-16 ***
## season3      0.267229   0.004211   63.457   <2e-16 ***
## season4      0.457991   0.004081  112.212   <2e-16 ***
## yr1          0.468568   0.001151  407.084   <2e-16 ***
## mnth2        0.113477   0.003775   30.063   <2e-16 ***
## mnth3        0.223629   0.003935   56.827   <2e-16 ***
## mnth4        0.181256   0.005234   34.628   <2e-16 ***
## mnth5        0.244622   0.005476   44.669   <2e-16 ***
## mnth6        0.196331   0.005584   35.157   <2e-16 ***
## mnth7        0.098776   0.006063   16.291   <2e-16 ***
## mnth8        0.195068   0.005898   33.076   <2e-16 ***
## mnth9        0.270833   0.005426   49.916   <2e-16 ***
## mnth10       0.187673   0.005394   34.794   <2e-16 ***
## mnth11       0.061080   0.005302   11.519   <2e-16 ***
## mnth12       0.045320   0.004675    9.694   <2e-16 ***
## hr1         -0.466686   0.008182  -57.037   <2e-16 ***
## hr2         -0.839682   0.009313  -90.161   <2e-16 ***
## hr3         -1.507858   0.012163 -123.968   <2e-16 ***
## hr4         -2.110449   0.015858 -133.084   <2e-16 ***
## hr5         -0.956563   0.009787  -97.738   <2e-16 ***
## hr6          0.400500   0.006619   60.509   <2e-16 ***
## hr7          1.422873   0.005666  251.117   <2e-16 ***
## hr8          1.916567   0.005423  353.411   <2e-16 ***
## hr9          1.391884   0.005648  246.430   <2e-16 ***
```

```
## hr10         1.123196   0.005806  193.439   <2e-16 ***
## hr11         1.269600   0.005717  222.072   <2e-16 ***
## hr12         1.447488   0.005642  256.546   <2e-16 ***
## hr13         1.427095   0.005663  251.992   <2e-16 ***
## hr14         1.364778   0.005707  239.122   <2e-16 ***
## hr15         1.405028   0.005693  246.796   <2e-16 ***
## hr16         1.628131   0.005592  291.130   <2e-16 ***
## hr17         2.036237   0.005445  373.973   <2e-16 ***
## hr18         1.970314   0.005442  362.032   <2e-16 ***
## hr19         1.674867   0.005518  303.541   <2e-16 ***
## hr20         1.377558   0.005648  243.883   <2e-16 ***
## hr21         1.121996   0.005800  193.434   <2e-16 ***
## hr22         0.864956   0.006005  144.039   <2e-16 ***
## hr23         0.483910   0.006419   75.382   <2e-16 ***
## holiday1    -0.160986   0.003797  -42.401   <2e-16 ***
## weekday1     0.051215   0.002167   23.632   <2e-16 ***
## weekday2     0.060927   0.002103   28.971   <2e-16 ***
## weekday3     0.066412   0.002103   31.584   <2e-16 ***
## weekday4     0.067340   0.002089   32.229   <2e-16 ***
## weekday5     0.093582   0.002089   44.798   <2e-16 ***
## weekday6     0.079610   0.002091   38.064   <2e-16 ***
## weathersit2 -0.064258   0.001422  -45.177   <2e-16 ***
## weathersit3 -0.492933   0.002863 -172.188   <2e-16 ***
## temp         0.164379   0.019469    8.443   <2e-16 ***
## atemp        0.946853   0.020326   46.584   <2e-16 ***
## hum         -0.205704   0.004129  -49.823   <2e-16 ***
## windspeed   -0.109968   0.004869  -22.583   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 2891591  on 17378  degrees of freedom
## Residual deviance:  572011  on 17327  degrees of freedom
## AIC: 683016
##
## Number of Fisher Scoring iterations: 5
```

```r
# ALL predicting variables are statistically significant...LOL WOW!
# We should perform variable selection
# When we analyzed this example, we saw the implications of applying regression to a large sample size
# When we use subsampling approach, some pred vars were identified as not statistically significant
```

```r
# Now we do a second model which will be reduced (exclude temp)
model_bikes2 = glm(cnt~-temp, data = clean_data, family = "poisson")

n = nrow(clean_data)

# Full model
c(AIC(model_bikes), AIC(model_bikes, k = log(n)))
```

```
## [1] 683016.2 683419.9
```

```r
# Reduced model without temp & compare values
c(AIC(model_bikes2), AIC(model_bikes2, k=log(n)))
```

```
## [1] 3002494 3002502
```

```r
# only use likelihood based criteria for logs so only AIC and BIC!

# Based on these two criteria, Full model is better than Reduced b/c values are smaller
```

```r
# Now stepwise forward regression
null_model = glm(formula = cnt ~ 1, data = clean_data, family = "poisson") # Null model with no variabl
full_model = glm(formula = cnt ~ ., data = clean_data, family = "poisson")
n = nrow(clean_data)

# With AIC
AIC <- step(null_model, scope = list(lower=null_model, upper = full_model), direction = "forward")
```

```
## Start:  AIC=3002494
## cnt ~ 1
##
##               Df Deviance     AIC
## + hr          23  1139526 1250475
## + temp         1  2390840 2501745
## + atemp        1  2395528 2506433
## + hum          1  2578465 2689370
## + mnth        11  2646585 2757510
## + season       3  2672290 2783199
## + yr           1  2700213 2811117
## + weathersit   2  2819029 2929936
## + windspeed    1  2865801 2976706
## + weekday      6  2887929 2998844
## + holiday      1  2888529 2999434
## <none>            2891591 3002494
##
## Step:  AIC=1250475
## cnt ~ hr
##
##               Df Deviance     AIC
## + atemp        1   881226  992177
## + mnth        11   884522  995493
## + temp         1   886326  997277
## + season       3   910332 1021286
## + yr           1   942887 1053838
## + weathersit   2  1059541 1170493
## + hum          1  1115105 1226056
## + windspeed    1  1132140 1243091
## + weekday      6  1135697 1246658
## + holiday      1  1136385 1247336
## <none>            1139526 1250475
##
## Step:  AIC=992176.9
## cnt ~ hr + atemp
```

```
##
##              Df Deviance    AIC
## + yr          1   700568 811521
## + season      3   826772 937729
## + weathersit  2   831984 942939
## + mnth       11   832426 943399
## + hum         1   861591 972544
## + weekday     6   877927 988889
## + holiday     1   879279 990232
## + windspeed   1   879395 990348
## + temp        1   881099 992052
## <none>            881226 992177
##
## Step:  AIC=811521.3
## cnt ~ hr + atemp + yr
##
##              Df Deviance    AIC
## + season      3   641460 752419
## + mnth       11   647074 758048
## + weathersit  2   655955 766912
## + hum         1   691115 802070
## + weekday     6   697625 808589
## + holiday     1   698292 809246
## + windspeed   1   699219 810174
## + temp        1   700525 811480
## <none>            700568 811521
##
## Step:  AIC=752419.1
## cnt ~ hr + atemp + yr + season
##
##              Df Deviance    AIC
## + weathersit  2   592460 703423
## + hum         1   623576 734537
## + mnth       11   631589 742570
## + weekday     6   638180 749151
## + holiday     1   639255 750216
## + temp        1   640837 751798
## + windspeed   1   641172 752133
## <none>            641460 752419
##
## Step:  AIC=703422.7
## cnt ~ hr + atemp + yr + season + weathersit
##
##             Df Deviance    AIC
## + mnth      11   579853 690838
## + weekday    6   589150 700125
## + holiday    1   589675 700640
## + hum        1   590745 701709
## + temp       1   591999 702964
## + windspeed  1   592360 703325
## <none>           592460 703423
##
## Step:  AIC=690837.7
## cnt ~ hr + atemp + yr + season + weathersit + mnth
```

```
##
##              Df Deviance    AIC
## + weekday    6   576736 687733
## + hum        1   577084 688071
## + holiday    1   577729 688716
## + temp       1   579664 690651
## + windspeed  1   579692 690679
## <none>           579853 690838
##
## Step:  AIC=687732.8
## cnt ~ hr + atemp + yr + season + weathersit + mnth + weekday
##
##              Df Deviance    AIC
## + hum        1   574373 685372
## + holiday    1   574875 685874
## + windspeed  1   576570 687569
## + temp       1   576641 687640
## <none>           576736 687733
##
## Step:  AIC=685372.1
## cnt ~ hr + atemp + yr + season + weathersit + mnth + weekday +
##     hum
##
##              Df Deviance    AIC
## + holiday    1   572558 683559
## + windspeed  1   573919 684920
## + temp       1   574361 685362
## <none>           574373 685372
##
## Step:  AIC=683559.1
## cnt ~ hr + atemp + yr + season + weathersit + mnth + weekday +
##     hum + holiday
##
##              Df Deviance    AIC
## + windspeed  1   572082 683085
## + temp       1   572522 683525
## <none>           572558 683559
##
## Step:  AIC=683084.9
## cnt ~ hr + atemp + yr + season + weathersit + mnth + weekday +
##     hum + holiday + windspeed
##
##         Df Deviance    AIC
## + temp  1   572011 683016
## <none>      572082 683085
##
## Step:  AIC=683016.2
## cnt ~ hr + atemp + yr + season + weathersit + mnth + weekday +
##     hum + holiday + windspeed + temp
```

```r
# With BIC
BIC <- step(null_model, scope=list(lower=null_model, upper = full_model), direction = "forward", k=log(
```

```
## Start:  AIC=3002502
```

```
## cnt ~ 1
##
##              Df Deviance     AIC
## + hr         23  1139526 1250661
## + temp        1  2390840 2501760
## + atemp       1  2395528 2506449
## + hum         1  2578465 2689386
## + mnth       11  2646585 2757603
## + season      3  2672290 2783230
## + yr          1  2700213 2811133
## + weathersit  2  2819029 2929959
## + windspeed   1  2865801 2976721
## + weekday     6  2887929 2998898
## + holiday     1  2888529 2999449
## <none>           2891591 3002502
##
## Step:  AIC=1250661
## cnt ~ hr
##
##              Df Deviance     AIC
## + atemp       1   881226  992371
## + mnth       11   884522  995764
## + temp        1   886326  997471
## + season      3   910332 1021496
## + yr          1   942887 1054032
## + weathersit  2  1059541 1170695
## + hum         1  1115105 1226250
## + windspeed   1  1132140 1243285
## + weekday     6  1135697 1246891
## + holiday     1  1136385 1247530
## <none>           1139526 1250661
##
## Step:  AIC=992370.9
## cnt ~ hr + atemp
##
##              Df Deviance    AIC
## + yr          1   700568 811723
## + season      3   826772 937946
## + weathersit  2   831984 943149
## + mnth       11   832426 943678
## + hum         1   861591 972746
## + weekday     6   877927 989130
## + holiday     1   879279 990434
## + windspeed   1   879395 990549
## + temp        1   881099 992254
## <none>            881226 992371
##
## Step:  AIC=811723.1
## cnt ~ hr + atemp + yr
##
##              Df Deviance    AIC
## + season      3   641460 752644
## + mnth       11   647074 758336
## + weathersit  2   655955 767129
```

```
## + hum          1   691115 802280
## + weekday      6   697625 808838
## + holiday      1   698292 809456
## + windspeed    1   699219 810384
## + temp         1   700525 811690
## <none>             700568 811723
##
## Step:  AIC=752644.2
## cnt ~ hr + atemp + yr + season
##
##               Df Deviance    AIC
## + weathersit  2   592460 703663
## + hum          1   623576 734770
## + mnth        11   631589 742880
## + weekday      6   638180 749422
## + holiday      1   639255 750449
## + temp         1   640837 752031
## + windspeed    1   641172 752365
## <none>             641460 752644
##
## Step:  AIC=703663.4
## cnt ~ hr + atemp + yr + season + weathersit
##
##              Df Deviance    AIC
## + mnth       11   579853 691164
## + weekday     6   589150 700412
## + holiday     1   589675 700889
## + hum         1   590745 701958
## + temp        1   591999 703213
## + windspeed   1   592360 703574
## <none>            592460 703663
##
## Step:  AIC=691163.7
## cnt ~ hr + atemp + yr + season + weathersit + mnth
##
##              Df Deviance    AIC
## + weekday     6   576736 688105
## + hum         1   577084 688405
## + holiday     1   577729 689050
## + temp        1   579664 690985
## + windspeed   1   579692 691013
## <none>            579853 691164
##
## Step:  AIC=688105.4
## cnt ~ hr + atemp + yr + season + weathersit + mnth + weekday
##
##              Df Deviance    AIC
## + hum         1   574373 685753
## + holiday     1   574875 686254
## + windspeed   1   576570 687949
## + temp        1   576641 688020
## <none>            576736 688105
##
## Step:  AIC=685752.5
```

```
## cnt ~ hr + atemp + yr + season + weathersit + mnth + weekday +
##     hum
##
##             Df Deviance    AIC
## + holiday    1   572558 683947
## + windspeed  1   573919 685308
## + temp       1   574361 685750
## <none>           574373 685753
##
## Step:  AIC=683947.3
## cnt ~ hr + atemp + yr + season + weathersit + mnth + weekday +
##     hum + holiday
##
##             Df Deviance    AIC
## + windspeed  1   572082 683481
## + temp       1   572522 683921
## <none>           572558 683947
##
## Step:  AIC=683480.8
## cnt ~ hr + atemp + yr + season + weathersit + mnth + weekday +
##     hum + holiday + windspeed
##
##        Df Deviance    AIC
## + temp  1   572011 683420
## <none>      572082 683481
##
## Step:  AIC=683419.9
## cnt ~ hr + atemp + yr + season + weathersit + mnth + weekday +
##     hum + holiday + windspeed + temp
```

```r
# only difference between AIC is the log(n) addition

# Predictors selected are same for AIC and BIC
# BIC output still shows AIC rather than BIC

# if we did backwards stepwise, we would select all predictors using both AIC or BIC
```

```r
BIC_back <- step(full_model, scope=list(lower=null_model, upper = full_model), direction = "backward", l
```

```
## Start:  AIC=683419.9
## cnt ~ season + yr + mnth + hr + holiday + weekday + weathersit +
##     temp + atemp + hum + windspeed
##
##              Df Deviance     AIC
## <none>              572011  683420
## - temp        1    572082  683481
## - windspeed   1    572522  683921
## - holiday     1    573886  685285
## - atemp       1    574323  685721
## - weekday     6    574413  685763
## - hum         1    574496  685894
## - mnth       11    584820  696121
## - season      3    587160  698539
```

```
## - weathersit  2   604754  716143
## - yr          1   742504  853903
## - hr         23  1843652 1954836
```

```r
# This is the output from the lecture notes too! All predictors selected
```

Moving on to LASSO Regression

```r
library(glmnet)
```

```
## Loading required package: Matrix
```

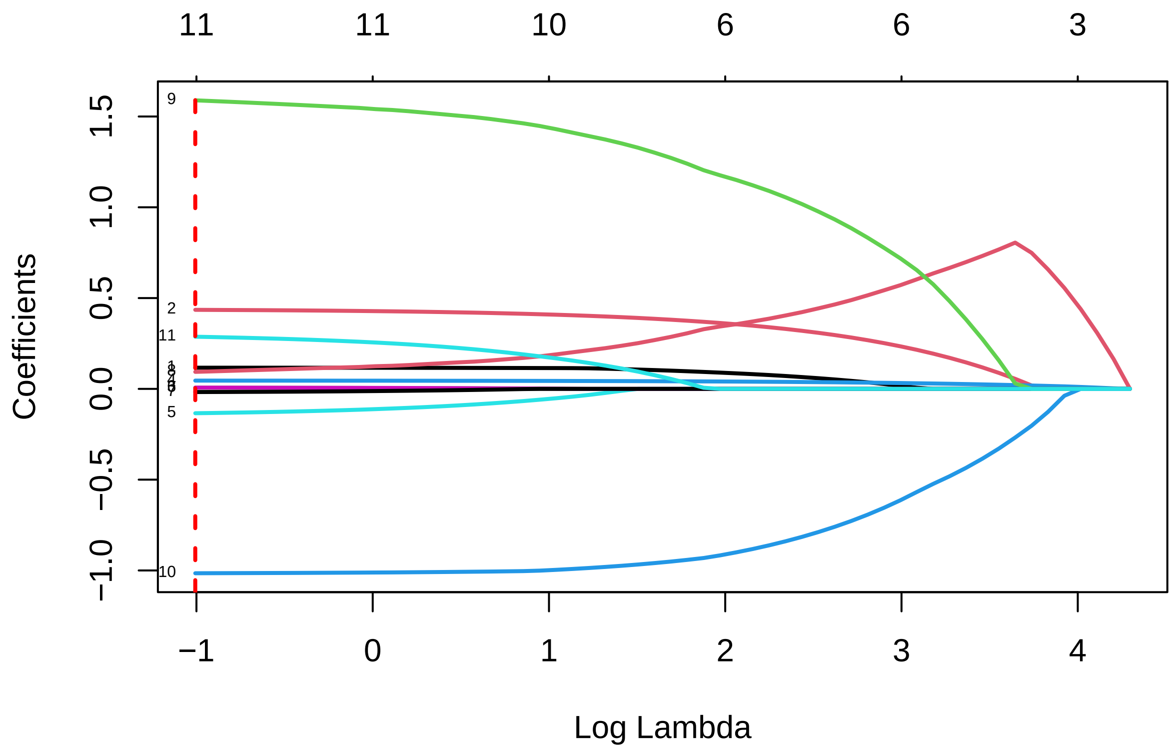
```
## Loaded glmnet 4.1-8
```

```r
x_pred = cbind(data$season, data$yr, data$mnth, data$hr, data$holiday, data$weekday, data$weathersit, da

# 10fold CV to find optimal lambda
bike_model.cv = cv.glmnet(x_pred, data$cnt, family = c("poisson"), alpha = 1, nfolds = 10)

# fit lasso model with 100 values for lambda:
bike_model = glmnet(x_pred, data$cnt, family = c("poisson"), alpha = 1, nlambda = 100)

# Extract coefficients at the optimal lambda:
coef(bike_model, s=bike_model.cv$lambda.min)
```

```
## 12 x 1 sparse Matrix of class "dgCMatrix"
##                       s1
## (Intercept)  3.766615288
## V1           0.116597053
## V2           0.435241032
## V3           0.006157263
## V4           0.045448937
## V5          -0.134120336
## V6           0.006824934
## V7          -0.017524449
## V8           0.093703648
## V9           1.588886247
## V10         -1.015351159
## V11          0.287170182
```

```r
# can see what these variables align with. V1 = season, V2 = year, V3 = month, V4 = HR
```

```r
# plot the lasso coef  path
plot(bike_model, xvar = "lambda", label = TRUE, lwd = 2)
abline(v=log(bike_model.cv$lambda.min),col='red',lty = 2,lwd=2)
```

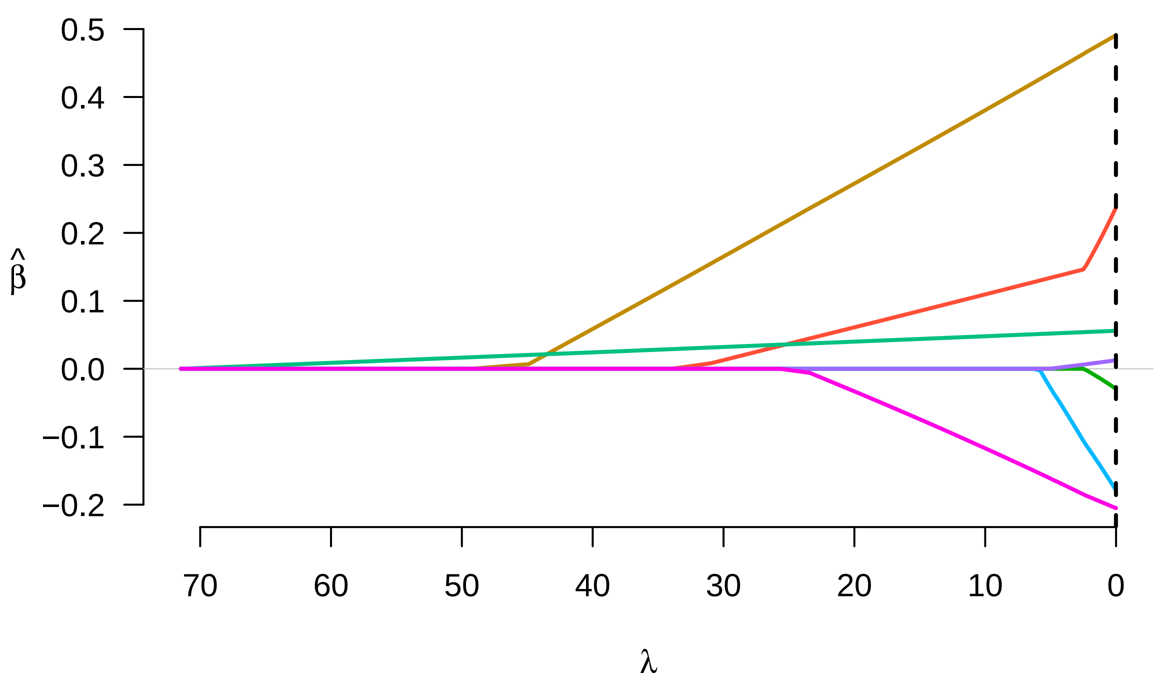



```
# if we compared to Elastic net, we'd see a similar output for coef path but elastic net would be smoot
# some coef paths for Elastic net are closer to the 0 line, indicating lower contribution to explanator
```

```
# GROUP LASSO:
library(grpreg)
library(scales)
library(caret)
# we have multiple qualitative and multiple dummy variables
# month of the year adds 11 dummy variables to the model
num_var <- cbind(data$temp, data$atemp, data$hum, data$windspeed)
num_var_scale <- sapply(num_var, rescale)
dv <- dummyVars("~ season + yr + mnth + hr + holiday + weekday + weathersit", data = data)

num_var_scale_matrix <- matrix(num_var_scale, nrow = nrow(num_var), byrow = FALSE)

# Create the dummy variables dataframe
x_dummy <- predict(dv, newdata = data)

x_pred_scale <- cbind(x_dummy, as.matrix(num_var_scale_matrix))

# set up the groups of variables here:
group = c(1,1,1,2,3,3,3,3,3,3,3,3,3,3,3,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,5,6,6,6,6,6,6,7,7
# 6 groups for each qualitative variable and 4 variables that are not part of a group (corresponding to
```

```r
# group lasso CV command to find optimal lambda and implement group lambda for the optimal lambda, then

# 10 fold CV to get optimal lambda:

group = 1:ncol(x_pred_scale)  # Assign each predictor its own group
grouplasso.cv = cv.grpreg(x_pred_scale, data$cnt, group = group, family = "poisson", nfolds = 10)

# fit model for 100 values for lambda:
grouplasso = grpreg(x_pred_scale, data$cnt, group = group, penalty = "grLasso", family = "poisson")

# Get the minimum lambda value from cross-validation
min_lambda <- grouplasso.cv$lambda.min

# Extract coefficients at the optimal lambda
coefficients_at_min_lambda <- coef(grouplasso, s = min_lambda)

# Display the coefficients
#coefficients_at_min_lambda
```

```r
# path of coeffs from Lasso regression:
plot(grouplasso, lwd=2)
abline(v=grouplasso.cv$lambda.min, col = 'black', lty = 2, lwd =2)
```



```r
# coef paths are plotted from largest to the smallest lambda
# most of the regr coeffs get selected for large lambda values
```

```r
library(dplyr)
```

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

##
## Attaching package: 'dplyr'

## The following object is masked from 'package:grpreg':
##
##     select

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

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

```r
cnt <- data$cnt
n = length(cnt)
x_pred = cbind(data$season, data$yr, data$mnth, data$hr, data$holiday, data$weekday, data$weathersit, da
colnames(x_pred) <- c("season", "yr", "mnth", "hr", "holiday", "weekday", "weathersit", "temp", "atemp"

# Sample 50% of the dataset:
perc = 0.5
var_count <- data.frame("var" = colnames(x_pred), "count" = 0) # initial count

for (i in 1:100) {
  subsample = sample(n, floor(n * perc), replace = FALSE)
  sub_x = x_pred[subsample, ]
  sub_cnt = cnt[subsample]
  # Find optimal lambda using 5-fold CV
  sub_model.cv = cv.glmnet(sub_x, sub_cnt, family = "poisson", alpha = 1, nfolds = 5)
  # Fit lasso model with 100 values for lambda
  sub_model = glmnet(sub_x, sub_cnt, family = "poisson", alpha = 1, nlambda = 100)
  # Extract coefficients at optimal lambda
  var_temp = as.matrix(coef(sub_model, s = sub_model.cv$lambda.min))

  # Remove the intercept and convert to a data frame
  var_temp_df = as.data.frame(var_temp[-1, , drop = FALSE])

  # Increment 'count' for non-zero coefficients
  var_count$count = var_count$count + (var_temp_df != 0)
}
```