ProblemSet7.R

DavidAarhus

2020-04-12

rm(list = ls()) #removing all variables  
#a) Navigate to the UCI Machine Learning website and download the data folder for the Bike Sharing Dataset. Read the file day.csv into R using read.csv and store it as the object Bike\_DF. Read the description to get familiar with the variables.  
 Bike\_DF <- read.csv("/Users/DavidAarhus/Documents/310 R/Datasets/day.csv")  
   
  
#b) Do some basic data cleaning on Bike\_DF to ensure factor variables are recorded as factors. Then run the command sapply(Bike\_DF, is.factor) to ensure the columns in your data frame that should be factors have been converted to factors appropriately. (Hint: the predictors weather situation, season,year, month, holiday, weekday, and workingday should be factor.)  
   
library("tidyverse")

## ── Attaching packages ────────────────────────────────────────────── tidyverse 1.3.0 ──

## ✓ ggplot2 3.2.1 ✓ purrr 0.3.3  
## ✓ tibble 2.1.3 ✓ dplyr 0.8.3  
## ✓ tidyr 1.0.0 ✓ stringr 1.4.0  
## ✓ readr 1.3.1 ✓ forcats 0.5.0

## ── Conflicts ───────────────────────────────────────────────── tidyverse\_conflicts() ──  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()

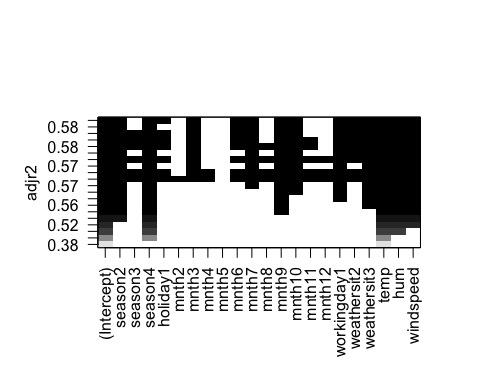
library("forcats")  
  
Bike\_DF$season <- as.factor(Bike\_DF$season)  
Bike\_DF$workingday <- as.factor(Bike\_DF$workingday)  
Bike\_DF$holiday <- as.factor(Bike\_DF$holiday)  
Bike\_DF$weathersit <- as.factor(Bike\_DF$weathersit)  
Bike\_DF$yr <- as.factor(Bike\_DF$yr)  
Bike\_DF$mnth <- as.factor(Bike\_DF$mnth)  
Bike\_DF$weekday <- as.factor(Bike\_DF$weekday)  
  
sapply(Bike\_DF, is.factor)

## instant dteday season yr mnth holiday weekday   
## FALSE TRUE TRUE TRUE TRUE TRUE TRUE   
## workingday weathersit temp atemp hum windspeed casual   
## TRUE TRUE FALSE FALSE FALSE FALSE FALSE   
## registered cnt   
## FALSE FALSE

#c) We now perform some feature transformation (creation of derived variables).   
   
Bike\_DF <- Bike\_DF %>% select(-instant)  
Bike\_DF <- Bike\_DF %>% select(-dteday)  
  
   
#Create 4 new variables as squared terms of temperature, feeling temperature, humidity, and windspeed. Refer to data description to find the corresponding predictors in the data.  
   
Bike\_DF <- Bike\_DF %>% mutate(temp\_sq = temp \* temp,  
 atemp\_sq = atemp \* atemp,  
 hum\_sq = hum \* hum,  
 windspeed\_sq = windspeed \* windspeed)  
   
  
#d) Split our data into test and training splits of size 30%/70% each. Use 310 as the seed number.  
   
set.seed(310)  
train\_idx <- sample(1:nrow(Bike\_DF),size = 0.7\*nrow(Bike\_DF),replace=FALSE)  
Bike\_train <- Bike\_DF[train\_idx,]  
Bike\_test <- Bike\_DF[-train\_idx,]  
   
  
#e) Fit a forward stepwise linear model on the training data with cnt as your outcome variable, and season, holiday, month, workingday, weathersit, temp, atemp, hum, windspeed, and the four squared terms, as your predcitor varaibles. Set max number of variables to 20. Note we end up with more variables because of factor variables. Save this as an object fwd\_fit.  
   
library("leaps")  
fwd\_fit <- regsubsets(cnt ~ season + holiday + mnth + workingday + weathersit + temp + hum + windspeed,  
 data = Bike\_train,  
 method = "forward",  
 nvmax = 20)  
   
#f) Run the summary command over fwd\_fit. What are the first five variables selected? Use the plot command to show variables selected. Hint: use scale="adjr2" to sort the models based on the adjusterd R2.  
   
summary(fwd\_fit)

## Subset selection object  
## Call: regsubsets.formula(cnt ~ season + holiday + mnth + workingday +   
## weathersit + temp + hum + windspeed, data = Bike\_train, method = "forward",   
## nvmax = 20)  
## 21 Variables (and intercept)  
## Forced in Forced out  
## season2 FALSE FALSE  
## season3 FALSE FALSE  
## season4 FALSE FALSE  
## holiday1 FALSE FALSE  
## mnth2 FALSE FALSE  
## mnth3 FALSE FALSE  
## mnth4 FALSE FALSE  
## mnth5 FALSE FALSE  
## mnth6 FALSE FALSE  
## mnth7 FALSE FALSE  
## mnth8 FALSE FALSE  
## mnth9 FALSE FALSE  
## mnth10 FALSE FALSE  
## mnth11 FALSE FALSE  
## mnth12 FALSE FALSE  
## workingday1 FALSE FALSE  
## weathersit2 FALSE FALSE  
## weathersit3 FALSE FALSE  
## temp FALSE FALSE  
## hum FALSE FALSE  
## windspeed FALSE FALSE  
## 1 subsets of each size up to 20  
## Selection Algorithm: forward  
## season2 season3 season4 holiday1 mnth2 mnth3 mnth4 mnth5 mnth6 mnth7  
## 1 ( 1 ) " " " " " " " " " " " " " " " " " " " "   
## 2 ( 1 ) " " " " "\*" " " " " " " " " " " " " " "   
## 3 ( 1 ) " " " " "\*" " " " " " " " " " " " " " "   
## 4 ( 1 ) " " " " "\*" " " " " " " " " " " " " " "   
## 5 ( 1 ) "\*" " " "\*" " " " " " " " " " " " " " "   
## 6 ( 1 ) "\*" " " "\*" " " " " " " " " " " " " " "   
## 7 ( 1 ) "\*" " " "\*" " " " " " " " " " " " " " "   
## 8 ( 1 ) "\*" " " "\*" " " " " " " " " " " " " " "   
## 9 ( 1 ) "\*" " " "\*" " " " " " " " " " " " " " "   
## 10 ( 1 ) "\*" " " "\*" " " " " " " " " " " " " "\*"   
## 11 ( 1 ) "\*" " " "\*" " " " " "\*" " " " " " " "\*"   
## 12 ( 1 ) "\*" " " "\*" " " " " "\*" " " " " " " "\*"   
## 13 ( 1 ) "\*" " " "\*" " " " " "\*" " " " " "\*" "\*"   
## 14 ( 1 ) "\*" " " "\*" "\*" " " "\*" " " " " "\*" "\*"   
## 15 ( 1 ) "\*" "\*" "\*" "\*" " " "\*" " " " " "\*" "\*"   
## 16 ( 1 ) "\*" "\*" "\*" "\*" " " "\*" " " " " "\*" "\*"   
## 17 ( 1 ) "\*" "\*" "\*" "\*" " " "\*" " " " " "\*" "\*"   
## 18 ( 1 ) "\*" "\*" "\*" "\*" " " "\*" " " " " "\*" "\*"   
## 19 ( 1 ) "\*" "\*" "\*" "\*" " " "\*" "\*" " " "\*" "\*"   
## 20 ( 1 ) "\*" "\*" "\*" "\*" "\*" "\*" "\*" " " "\*" "\*"   
## mnth8 mnth9 mnth10 mnth11 mnth12 workingday1 weathersit2 weathersit3  
## 1 ( 1 ) " " " " " " " " " " " " " " " "   
## 2 ( 1 ) " " " " " " " " " " " " " " " "   
## 3 ( 1 ) " " " " " " " " " " " " " " " "   
## 4 ( 1 ) " " " " " " " " " " " " " " " "   
## 5 ( 1 ) " " " " " " " " " " " " " " " "   
## 6 ( 1 ) " " "\*" " " " " " " " " " " " "   
## 7 ( 1 ) " " "\*" " " " " " " " " " " "\*"   
## 8 ( 1 ) " " "\*" " " " " " " "\*" " " "\*"   
## 9 ( 1 ) " " "\*" "\*" " " " " "\*" " " "\*"   
## 10 ( 1 ) " " "\*" "\*" " " " " "\*" " " "\*"   
## 11 ( 1 ) " " "\*" "\*" " " " " "\*" " " "\*"   
## 12 ( 1 ) " " "\*" "\*" " " " " "\*" "\*" "\*"   
## 13 ( 1 ) " " "\*" "\*" " " " " "\*" "\*" "\*"   
## 14 ( 1 ) " " "\*" "\*" " " " " "\*" "\*" "\*"   
## 15 ( 1 ) " " "\*" "\*" " " " " "\*" "\*" "\*"   
## 16 ( 1 ) " " "\*" "\*" "\*" " " "\*" "\*" "\*"   
## 17 ( 1 ) "\*" "\*" "\*" "\*" " " "\*" "\*" "\*"   
## 18 ( 1 ) "\*" "\*" "\*" "\*" "\*" "\*" "\*" "\*"   
## 19 ( 1 ) "\*" "\*" "\*" "\*" "\*" "\*" "\*" "\*"   
## 20 ( 1 ) "\*" "\*" "\*" "\*" "\*" "\*" "\*" "\*"   
## temp hum windspeed  
## 1 ( 1 ) "\*" " " " "   
## 2 ( 1 ) "\*" " " " "   
## 3 ( 1 ) "\*" "\*" " "   
## 4 ( 1 ) "\*" "\*" "\*"   
## 5 ( 1 ) "\*" "\*" "\*"   
## 6 ( 1 ) "\*" "\*" "\*"   
## 7 ( 1 ) "\*" "\*" "\*"   
## 8 ( 1 ) "\*" "\*" "\*"   
## 9 ( 1 ) "\*" "\*" "\*"   
## 10 ( 1 ) "\*" "\*" "\*"   
## 11 ( 1 ) "\*" "\*" "\*"   
## 12 ( 1 ) "\*" "\*" "\*"   
## 13 ( 1 ) "\*" "\*" "\*"   
## 14 ( 1 ) "\*" "\*" "\*"   
## 15 ( 1 ) "\*" "\*" "\*"   
## 16 ( 1 ) "\*" "\*" "\*"   
## 17 ( 1 ) "\*" "\*" "\*"   
## 18 ( 1 ) "\*" "\*" "\*"   
## 19 ( 1 ) "\*" "\*" "\*"   
## 20 ( 1 ) "\*" "\*" "\*"

plot(fwd\_fit, scale="adjr2")



#g) Fit a Ridge model against the bike\_train dataset. Call the plot function against the fitted model to see how MSE varies as we move λ.  
   
library("glmnet")

## Loading required package: Matrix

##   
## Attaching package: 'Matrix'

## The following objects are masked from 'package:tidyr':  
##   
## expand, pack, unpack

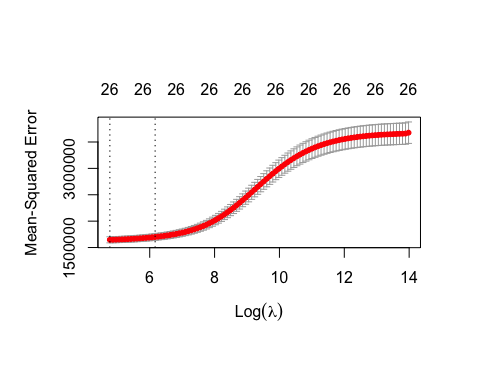
## Loaded glmnet 3.0-2

library("glmnetUtils")

##   
## Attaching package: 'glmnetUtils'

## The following objects are masked from 'package:glmnet':  
##   
## cv.glmnet, glmnet

ridge\_fit <- cv.glmnet(cnt ~ season + holiday + mnth + workingday + weathersit + temp + hum + windspeed,  
 data = Bike\_train,  
 alpha = 0,  
 nfolds = 10)  
plot(ridge\_fit)



#h) What are the values for lambda.min and lambda.1se? What is the meaning of each of these lambdas?  
   
 #lambda.min = 118.0217  
 #lambda.1se = 476.4557  
   
 #These lambdas decide how much we care about bias vs variance. Because our lambdas are pretty big, overall we are choosing to increase our bias a little bit in exchange for less variance.  
   
   
#i) Print the value of the coefficients at lambda.min and lambda.1se. What do you notice about the differenecs between the coefficients. (Note: you will need to type as.matrix(coef(ridge\_fit, s = "lambda.min")) to convert the coefficient vector from a sparse data matrix to a matrix.  
   
lasso\_mod <- cv.glmnet(cnt ~ season + holiday + mnth + workingday + weathersit + temp + hum + windspeed,  
 data = Bike\_train,  
 alpha = 1,  
 nfolds = 10)  
   
#k) How many variables are selected by the lambda.min and lambda.1se versions of the model? Print the coefficient vectors for each.  
   
coef(lasso\_mod, lasso\_mod$lambda.1se)

## 27 x 1 sparse Matrix of class "dgCMatrix"  
## 1  
## (Intercept) 3937.5222  
## season1 -942.2750  
## season2 .   
## season3 .   
## season4 180.0135  
## holiday0 .   
## holiday1 .   
## mnth1 -157.8390  
## mnth2 .   
## mnth3 .   
## mnth4 .   
## mnth5 .   
## mnth6 .   
## mnth7 -226.6174  
## mnth8 .   
## mnth9 200.6087  
## mnth10 394.6708  
## mnth11 .   
## mnth12 .   
## workingday0 -133.6080  
## workingday1 .   
## weathersit1 293.4725  
## weathersit2 .   
## weathersit3 -1057.3753  
## temp 4507.1560  
## hum -1924.0604  
## windspeed -2189.7929

coef\_mat\_1se <- data.frame(rownames = rownames(coef(lasso\_mod)) %>% data.frame(),coef\_1se <- as.matrix(coef(lasso\_mod, lasso\_mod$lambda.1se)) %>% round(3)) %>% remove\_rownames() %>% rename(rownames = 1,coef\_1se = 2)  
  
coef(lasso\_mod, lasso\_mod$lambda.min)

## 27 x 1 sparse Matrix of class "dgCMatrix"  
## 1  
## (Intercept) 3562.8667932  
## season1 -411.7612165  
## season2 343.4790537  
## season3 -0.9794722  
## season4 909.5018810  
## holiday0 226.3281168  
## holiday1 .   
## mnth1 -88.7986078  
## mnth2 .   
## mnth3 407.5649685  
## mnth4 217.8892837  
## mnth5 218.0005249  
## mnth6 -12.3560774  
## mnth7 -480.6579882  
## mnth8 .   
## mnth9 617.7800335  
## mnth10 483.4944878  
## mnth11 -159.6718645  
## mnth12 .   
## workingday0 -263.6005051  
## workingday1 .   
## weathersit1 257.3603690  
## weathersit2 .   
## weathersit3 -1103.1563481  
## temp 6140.2092550  
## hum -3191.6129044  
## windspeed -3598.1932373

coef\_mat\_min <- data.frame(rownames = rownames(coef(lasso\_mod)) %>% data.frame(),coef\_1se <- as.matrix(coef(lasso\_mod, lasso\_mod$lambda.min)) %>% round(3)) %>% remove\_rownames() %>% rename(rownames = 1,coef\_1se = 2)