# Module 3 Assignment 1 - Model Validation Assignment

## Brian Libby

## Answer to Questions (See relevant sections below for analysis)

Task 2:  
1. How many rows of data are in each set (training and testing)?  
**training = 12,167**  
**testing = 5,212**

Task 3:  
1. Comment on the quality of the model. Be sure to note the Adjusted R-squared value.  
**The resulting model appears to be good. Most of the variables have a p-value less than 0.05. This, however, can be misleading. This is a large data set. Larger data sets are more likely to produce significant p-values. The adjusted R-squared value of 0.5805 looks ok.**

Task 4:  
1. Comment on the predictions.  
**There are multiple negative predictions made by the current model. This indicates the model is not as good as the initial statistics showed. It is not possible to have negative bike rentals.**

Task 5:  
1. Comment on the predictions.  
**The predictions from the testing data set are very similar to those from the training data set. The summary statisitics (min, max, median, mean, and quartile values) are very similar between the two data sets. This indicates that the training/testing split was effective for the bike data set.**

Task 6:  
1. Comment on how this value compares to the model’s performance on the training set.  
**The adjusted R-squared value of the training data set was 0.5805. If the training data set is representative of the entire data set, the calculated R-squared value from the testing data set should be very similar. As the results show, the calculated R-squared value for the testing data set is 0.5944. The predicted values paired with the calculated R-squared value shows the model performs the same on both the training and the testing data sets.**

Task 7:  
1. Describe how k-fold cross-validation differs from model validation via a training/testing split.  
**K-fold cross-validation divides a data set into folds/partitions, usually 3, 5, or 10 folds/partitions. The model is then run k times, leaving one of the folds/partitions out each time the model is run. K-fold cross-validation is appropriate for large data sets. Training/testing split model validation splits the data set into two parts. The training set usually contains 70%-80% of the data while the testing set contains the remaining 20%-30% of the data. The model is built and “perfected” on the training set and is then used on the testing set. The training/testing validation is appropriate for very large data sets.**

### Load Libraries

library(tidyverse)  
library(MASS)  
library(caret)

### Load Data and Convert Data Types

bike <- read\_csv("hour.csv")

## Parsed with column specification:  
## cols(  
## instant = col\_double(),  
## dteday = col\_date(format = ""),  
## season = col\_double(),  
## yr = col\_double(),  
## mnth = col\_double(),  
## hr = col\_double(),  
## holiday = col\_double(),  
## weekday = col\_double(),  
## workingday = col\_double(),  
## weathersit = col\_double(),  
## temp = col\_double(),  
## atemp = col\_double(),  
## hum = col\_double(),  
## windspeed = col\_double(),  
## casual = col\_double(),  
## registered = col\_double(),  
## count = col\_double()  
## )

str(bike) #view structure of dataframe

## Classes 'spec\_tbl\_df', 'tbl\_df', 'tbl' and 'data.frame': 17379 obs. of 17 variables:  
## $ instant : num 1 2 3 4 5 6 7 8 9 10 ...  
## $ dteday : Date, format: "2011-01-01" "2011-01-01" ...  
## $ season : num 1 1 1 1 1 1 1 1 1 1 ...  
## $ yr : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ mnth : num 1 1 1 1 1 1 1 1 1 1 ...  
## $ hr : num 0 1 2 3 4 5 6 7 8 9 ...  
## $ holiday : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ weekday : num 6 6 6 6 6 6 6 6 6 6 ...  
## $ workingday: num 0 0 0 0 0 0 0 0 0 0 ...  
## $ weathersit: num 1 1 1 1 1 2 1 1 1 1 ...  
## $ temp : num 0.24 0.22 0.22 0.24 0.24 0.24 0.22 0.2 0.24 0.32 ...  
## $ atemp : num 0.288 0.273 0.273 0.288 0.288 ...  
## $ hum : num 0.81 0.8 0.8 0.75 0.75 0.75 0.8 0.86 0.75 0.76 ...  
## $ windspeed : num 0 0 0 0 0 0.0896 0 0 0 0 ...  
## $ casual : num 3 8 5 3 0 0 2 1 1 8 ...  
## $ registered: num 13 32 27 10 1 1 0 2 7 6 ...  
## $ count : num 16 40 32 13 1 1 2 3 8 14 ...  
## - attr(\*, "spec")=  
## .. cols(  
## .. instant = col\_double(),  
## .. dteday = col\_date(format = ""),  
## .. season = col\_double(),  
## .. yr = col\_double(),  
## .. mnth = col\_double(),  
## .. hr = col\_double(),  
## .. holiday = col\_double(),  
## .. weekday = col\_double(),  
## .. workingday = col\_double(),  
## .. weathersit = col\_double(),  
## .. temp = col\_double(),  
## .. atemp = col\_double(),  
## .. hum = col\_double(),  
## .. windspeed = col\_double(),  
## .. casual = col\_double(),  
## .. registered = col\_double(),  
## .. count = col\_double()  
## .. )

summary(bike) #view summary of dataframe

## instant dteday season yr   
## Min. : 1 Min. :2011-01-01 Min. :1.000 Min. :0.0000   
## 1st Qu.: 4346 1st Qu.:2011-07-04 1st Qu.:2.000 1st Qu.:0.0000   
## Median : 8690 Median :2012-01-02 Median :3.000 Median :1.0000   
## Mean : 8690 Mean :2012-01-02 Mean :2.502 Mean :0.5026   
## 3rd Qu.:13034 3rd Qu.:2012-07-02 3rd Qu.:3.000 3rd Qu.:1.0000   
## Max. :17379 Max. :2012-12-31 Max. :4.000 Max. :1.0000   
## mnth hr holiday weekday   
## Min. : 1.000 Min. : 0.00 Min. :0.00000 Min. :0.000   
## 1st Qu.: 4.000 1st Qu.: 6.00 1st Qu.:0.00000 1st Qu.:1.000   
## Median : 7.000 Median :12.00 Median :0.00000 Median :3.000   
## Mean : 6.538 Mean :11.55 Mean :0.02877 Mean :3.004   
## 3rd Qu.:10.000 3rd Qu.:18.00 3rd Qu.:0.00000 3rd Qu.:5.000   
## Max. :12.000 Max. :23.00 Max. :1.00000 Max. :6.000   
## workingday weathersit temp atemp   
## Min. :0.0000 Min. :1.000 Min. :0.020 Min. :0.0000   
## 1st Qu.:0.0000 1st Qu.:1.000 1st Qu.:0.340 1st Qu.:0.3333   
## Median :1.0000 Median :1.000 Median :0.500 Median :0.4848   
## Mean :0.6827 Mean :1.425 Mean :0.497 Mean :0.4758   
## 3rd Qu.:1.0000 3rd Qu.:2.000 3rd Qu.:0.660 3rd Qu.:0.6212   
## Max. :1.0000 Max. :4.000 Max. :1.000 Max. :1.0000   
## hum windspeed casual registered   
## Min. :0.0000 Min. :0.0000 Min. : 0.00 Min. : 0.0   
## 1st Qu.:0.4800 1st Qu.:0.1045 1st Qu.: 4.00 1st Qu.: 34.0   
## Median :0.6300 Median :0.1940 Median : 17.00 Median :115.0   
## Mean :0.6272 Mean :0.1901 Mean : 35.68 Mean :153.8   
## 3rd Qu.:0.7800 3rd Qu.:0.2537 3rd Qu.: 48.00 3rd Qu.:220.0   
## Max. :1.0000 Max. :0.8507 Max. :367.00 Max. :886.0   
## count   
## Min. : 1.0   
## 1st Qu.: 40.0   
## Median :142.0   
## Mean :189.5   
## 3rd Qu.:281.0   
## Max. :977.0

#convert season to named factor  
bike <- bike %>%   
 mutate(season = as\_factor(as.character(season))) %>%  
 mutate(season = fct\_recode(season,  
 "Spring" = "1",  
 "Summer" = "2",  
 "Fall" = "3",  
 "Winter" = "4")  
 )  
  
#Convert “yr”, “mnth”, and “hr” to factors. Do NOT need to recode (rename) the factors  
bike <- bike %>%   
 mutate(yr = as\_factor(as.character(yr)), mnth = as\_factor(as.character(mnth)), hr = as\_factor(as.character(hr)))  
  
#Convert the “holiday” variable to a factor and recode the levels from 0 to “NotHoliday” and 1 to “Holiday”  
bike <- bike %>%   
 mutate(holiday = as\_factor(as.character(holiday))) %>%  
 mutate(holiday = fct\_recode(holiday,  
 "NotHoliday" = "0",  
 "Holiday" = "1")  
 )   
  
#Convert “workingday” to a factor and recode the levels from 0 to “NotWorkingDay” and 1 to “WorkingDay”  
bike <- bike %>%   
 mutate(workingday = as\_factor(as.character(workingday))) %>%  
 mutate(workingday = fct\_recode(workingday,  
 "NotWorkingDay" = "0",  
 "WorkingDay" = "1")  
 )   
#Convert “weathersit” to a factor and recode the levels. Level 1 should be “NoPrecip”, 2 should become  
#“Misty”, 3 should become “LightPrecip”, and 4 should become “HeavyPrecip”  
bike <- bike %>%   
 mutate(weathersit = as\_factor(as.character(weathersit))) %>%  
 mutate(weathersit = fct\_recode(weathersit,  
 "NoPrecip" = "1",  
 "Misty" = "2",  
 "LightPrecip" = "3",  
 "HeavyPrecip" = "4")  
 )  
  
#Convert the “weekday” variable to a factor and recode the levels. Note that 6 is “Saturday” and 0 is “Sunday”  
bike <- bike %>%   
 mutate(weekday = as\_factor(as.character(weekday))) %>%  
 mutate(weekday = fct\_recode(weekday,  
 "Sunday" = "0",  
 "Monday" = "1",  
 "Tuesday" = "2",  
 "Wednesday" = "3",  
 "Thursday" = "4",  
 "Friday" = "5",  
 "Saturday" = "6")  
 )  
  
#Verify structure of data  
glimpse(bike) #view structure of dataframe

## Observations: 17,379  
## Variables: 17  
## $ instant <dbl> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 1...  
## $ dteday <date> 2011-01-01, 2011-01-01, 2011-01-01, 2011-01-01, 2011-01...  
## $ season <fct> Spring, Spring, Spring, Spring, Spring, Spring, Spring, ...  
## $ yr <fct> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...  
## $ mnth <fct> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...  
## $ hr <fct> 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16...  
## $ holiday <fct> NotHoliday, NotHoliday, NotHoliday, NotHoliday, NotHolid...  
## $ weekday <fct> Saturday, Saturday, Saturday, Saturday, Saturday, Saturd...  
## $ workingday <fct> NotWorkingDay, NotWorkingDay, NotWorkingDay, NotWorkingD...  
## $ weathersit <fct> NoPrecip, NoPrecip, NoPrecip, NoPrecip, NoPrecip, Misty,...  
## $ temp <dbl> 0.24, 0.22, 0.22, 0.24, 0.24, 0.24, 0.22, 0.20, 0.24, 0....  
## $ atemp <dbl> 0.2879, 0.2727, 0.2727, 0.2879, 0.2879, 0.2576, 0.2727, ...  
## $ hum <dbl> 0.81, 0.80, 0.80, 0.75, 0.75, 0.75, 0.80, 0.86, 0.75, 0....  
## $ windspeed <dbl> 0.0000, 0.0000, 0.0000, 0.0000, 0.0000, 0.0896, 0.0000, ...  
## $ casual <dbl> 3, 8, 5, 3, 0, 0, 2, 1, 1, 8, 12, 26, 29, 47, 35, 40, 41...  
## $ registered <dbl> 13, 32, 27, 10, 1, 1, 0, 2, 7, 6, 24, 30, 55, 47, 71, 70...  
## $ count <dbl> 16, 40, 32, 13, 1, 1, 2, 3, 8, 14, 36, 56, 84, 94, 106, ...

summary(bike) #view summary of dataframe

## instant dteday season yr mnth   
## Min. : 1 Min. :2011-01-01 Spring:4242 0:8645 5 :1488   
## 1st Qu.: 4346 1st Qu.:2011-07-04 Summer:4409 1:8734 7 :1488   
## Median : 8690 Median :2012-01-02 Fall :4496 12 :1483   
## Mean : 8690 Mean :2012-01-02 Winter:4232 8 :1475   
## 3rd Qu.:13034 3rd Qu.:2012-07-02 3 :1473   
## Max. :17379 Max. :2012-12-31 10 :1451   
## (Other):8521   
## hr holiday weekday workingday   
## 16 : 730 NotHoliday:16879 Saturday :2512 NotWorkingDay: 5514   
## 17 : 730 Holiday : 500 Sunday :2502 WorkingDay :11865   
## 13 : 729 Monday :2479   
## 14 : 729 Tuesday :2453   
## 15 : 729 Wednesday:2475   
## 12 : 728 Thursday :2471   
## (Other):13004 Friday :2487   
## weathersit temp atemp hum   
## NoPrecip :11413 Min. :0.020 Min. :0.0000 Min. :0.0000   
## Misty : 4544 1st Qu.:0.340 1st Qu.:0.3333 1st Qu.:0.4800   
## LightPrecip: 1419 Median :0.500 Median :0.4848 Median :0.6300   
## HeavyPrecip: 3 Mean :0.497 Mean :0.4758 Mean :0.6272   
## 3rd Qu.:0.660 3rd Qu.:0.6212 3rd Qu.:0.7800   
## Max. :1.000 Max. :1.0000 Max. :1.0000   
##   
## windspeed casual registered count   
## Min. :0.0000 Min. : 0.00 Min. : 0.0 Min. : 1.0   
## 1st Qu.:0.1045 1st Qu.: 4.00 1st Qu.: 34.0 1st Qu.: 40.0   
## Median :0.1940 Median : 17.00 Median :115.0 Median :142.0   
## Mean :0.1901 Mean : 35.68 Mean :153.8 Mean :189.5   
## 3rd Qu.:0.2537 3rd Qu.: 48.00 3rd Qu.:220.0 3rd Qu.:281.0   
## Max. :0.8507 Max. :367.00 Max. :886.0 Max. :977.0   
##

### Task 1 - Split the data into training and testing sets.

set.seed(1234)  
training.rows <- createDataPartition(y = bike$count, p = 0.7, list = FALSE) #70% in training dataframe  
training <- bike[training.rows,]  
testing <- bike[-training.rows,]

### Task 2 - Number of rows in training and testing data frames.

nrow(training) #number rows in Training data frame

## [1] 12167

nrow(testing) #number rows in Testing data frame

## [1] 5212

### Task 3 - Build a linear regression model (.

model1 <- lm(count ~ season + mnth + hr + holiday + weekday, data = training)  
summary(model1) #examine the model

##   
## Call:  
## lm(formula = count ~ season + mnth + hr + holiday + weekday,   
## data = training)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -491.51 -61.55 -10.01 54.90 514.70   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -34.2889 6.7577 -5.074 3.95e-07 \*\*\*  
## seasonSummer 21.6330 6.7298 3.215 0.001310 \*\*   
## seasonFall 38.1088 7.9524 4.792 1.67e-06 \*\*\*  
## seasonWinter 74.4276 6.7434 11.037 < 2e-16 \*\*\*  
## mnth2 15.2055 5.3395 2.848 0.004410 \*\*   
## mnth3 48.3364 5.7518 8.404 < 2e-16 \*\*\*  
## mnth4 70.8624 8.5336 8.304 < 2e-16 \*\*\*  
## mnth5 107.6548 8.4950 12.673 < 2e-16 \*\*\*  
## mnth6 116.3789 8.2962 14.028 < 2e-16 \*\*\*  
## mnth7 98.3342 9.4897 10.362 < 2e-16 \*\*\*  
## mnth8 104.8969 9.5204 11.018 < 2e-16 \*\*\*  
## mnth9 95.3349 8.7437 10.903 < 2e-16 \*\*\*  
## mnth10 53.5768 8.5214 6.287 3.34e-10 \*\*\*  
## mnth11 12.7934 8.5556 1.495 0.134859   
## mnth12 -0.6776 6.8211 -0.099 0.920875   
## hr1 -22.9612 7.3252 -3.135 0.001725 \*\*   
## hr2 -34.3298 7.3216 -4.689 2.78e-06 \*\*\*  
## hr3 -47.8838 7.4456 -6.431 1.31e-10 \*\*\*  
## hr4 -51.5771 7.3713 -6.997 2.75e-12 \*\*\*  
## hr5 -38.1144 7.2968 -5.223 1.79e-07 \*\*\*  
## hr6 21.7730 7.3862 2.948 0.003207 \*\*   
## hr7 148.9485 7.3365 20.302 < 2e-16 \*\*\*  
## hr8 306.3794 7.2999 41.970 < 2e-16 \*\*\*  
## hr9 165.6835 7.3716 22.476 < 2e-16 \*\*\*  
## hr10 114.8402 7.2968 15.738 < 2e-16 \*\*\*  
## hr11 153.1223 7.3331 20.881 < 2e-16 \*\*\*  
## hr12 198.0786 7.2790 27.212 < 2e-16 \*\*\*  
## hr13 201.7389 7.3000 27.635 < 2e-16 \*\*\*  
## hr14 182.2426 7.3236 24.884 < 2e-16 \*\*\*  
## hr15 194.8233 7.3529 26.496 < 2e-16 \*\*\*  
## hr16 254.3963 7.3458 34.632 < 2e-16 \*\*\*  
## hr17 398.7885 7.3000 54.629 < 2e-16 \*\*\*  
## hr18 365.6865 7.2790 50.238 < 2e-16 \*\*\*  
## hr19 252.5569 7.3407 34.405 < 2e-16 \*\*\*  
## hr20 171.4685 7.3591 23.300 < 2e-16 \*\*\*  
## hr21 115.5656 7.2971 15.837 < 2e-16 \*\*\*  
## hr22 77.2345 7.3412 10.521 < 2e-16 \*\*\*  
## hr23 31.2198 7.3549 4.245 2.20e-05 \*\*\*  
## holidayHoliday -15.5263 6.7216 -2.310 0.020910 \*   
## weekdaySunday -13.2719 3.9549 -3.356 0.000794 \*\*\*  
## weekdayMonday -5.7093 4.0892 -1.396 0.162680   
## weekdayTuesday -4.2730 4.0168 -1.064 0.287449   
## weekdayWednesday -2.5402 3.9794 -0.638 0.523275   
## weekdayThursday 4.1389 3.9915 1.037 0.299790   
## weekdayFriday 3.1967 3.9652 0.806 0.420145   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 117.6 on 12122 degrees of freedom  
## Multiple R-squared: 0.582, Adjusted R-squared: 0.5805   
## F-statistic: 383.6 on 44 and 12122 DF, p-value: < 2.2e-16

### Task 4: Make predictions (using model from Task 3) on the training set.

predict\_training <- predict(model1, newdata = training) #develop predictions  
head(predict\_training, 6) #display first 6 predictions

## 1 2 3 4 5 6   
## -57.25008 -68.61871 -82.17268 -85.86603 -72.40327 -12.51588

summary(predict\_training)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## -107.10 74.74 197.29 189.33 285.54 538.40

### Task 5: Make predictions (using model from Task 3) on the testing set.

predict\_testing <- predict(model1, newdata = testing) #develop predictions  
head(predict\_testing, 6) #display first 6 predictions

## 1 2 3 4 5 6   
## -34.2889 114.6596 137.1796 -47.5608 -81.8906 150.5178

summary(predict\_testing)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## -107.8 77.0 196.0 189.6 282.8 538.4

### Task 6 - Manually calculate the R squared value on the testing set.

SSE <- sum((testing$count - predict\_testing)^2) #sum of squared residuals of model  
SST <- sum((testing$count - mean(testing$count))^2) #sum of squared residuals from a naive model  
rsquared <- 1 - SSE/SST #definition of R-squared  
  
rsquared

## [1] 0.5943977

### Task 7 - K-Fold vs Trianing/Testing Split.

ctrl = trainControl(method = "cv",number = 10) #set up caret 10 fold cross validation  
  
set.seed(1234) #set random number seed for cross validation  
modelCV = train(count ~ season + mnth + hr + holiday + weekday, data = bike, method = "lm", trControl = ctrl, metric="Rsquared")  
  
summary(modelCV)

##   
## Call:  
## lm(formula = .outcome ~ ., data = dat)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -497.65 -61.50 -9.66 55.72 507.05   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -39.8155 5.6728 -7.019 2.32e-12 \*\*\*  
## seasonSummer 31.2462 5.5647 5.615 1.99e-08 \*\*\*  
## seasonFall 46.0328 6.5610 7.016 2.36e-12 \*\*\*  
## seasonWinter 78.7081 5.5757 14.116 < 2e-16 \*\*\*  
## mnth2 18.1036 4.4445 4.073 4.66e-05 \*\*\*  
## mnth3 50.4133 4.7762 10.555 < 2e-16 \*\*\*  
## mnth4 64.5009 7.0723 9.120 < 2e-16 \*\*\*  
## mnth5 100.5891 7.0485 14.271 < 2e-16 \*\*\*  
## mnth6 112.3618 6.8486 16.407 < 2e-16 \*\*\*  
## mnth7 94.8390 7.8587 12.068 < 2e-16 \*\*\*  
## mnth8 99.5632 7.8631 12.662 < 2e-16 \*\*\*  
## mnth9 94.3764 7.2128 13.085 < 2e-16 \*\*\*  
## mnth10 51.8409 7.0700 7.332 2.36e-13 \*\*\*  
## mnth11 7.3305 7.0896 1.034 0.301160   
## mnth12 -0.5872 5.6354 -0.104 0.917017   
## hr1 -20.4207 6.1367 -3.328 0.000878 \*\*\*  
## hr2 -31.7342 6.1560 -5.155 2.56e-07 \*\*\*  
## hr3 -44.8673 6.1962 -7.241 4.64e-13 \*\*\*  
## hr4 -50.3097 6.1962 -8.119 5.00e-16 \*\*\*  
## hr5 -34.8743 6.1517 -5.669 1.46e-08 \*\*\*  
## hr6 22.1481 6.1345 3.610 0.000307 \*\*\*  
## hr7 158.2666 6.1303 25.817 < 2e-16 \*\*\*  
## hr8 305.2130 6.1303 49.787 < 2e-16 \*\*\*  
## hr9 165.5115 6.1303 26.999 < 2e-16 \*\*\*  
## hr10 119.8705 6.1303 19.554 < 2e-16 \*\*\*  
## hr11 154.3451 6.1303 25.177 < 2e-16 \*\*\*  
## hr12 199.6516 6.1282 32.579 < 2e-16 \*\*\*  
## hr13 199.9511 6.1261 32.639 < 2e-16 \*\*\*  
## hr14 187.2392 6.1261 30.564 < 2e-16 \*\*\*  
## hr15 197.5231 6.1261 32.243 < 2e-16 \*\*\*  
## hr16 258.3975 6.1240 42.194 < 2e-16 \*\*\*  
## hr17 407.8660 6.1240 66.601 < 2e-16 \*\*\*  
## hr18 371.8649 6.1282 60.681 < 2e-16 \*\*\*  
## hr19 257.8772 6.1282 42.080 < 2e-16 \*\*\*  
## hr20 172.3841 6.1282 28.130 < 2e-16 \*\*\*  
## hr21 118.6684 6.1282 19.364 < 2e-16 \*\*\*  
## hr22 77.6890 6.1282 12.677 < 2e-16 \*\*\*  
## hr23 34.1849 6.1282 5.578 2.47e-08 \*\*\*  
## holidayHoliday -14.6620 5.5903 -2.623 0.008730 \*\*   
## weekdaySunday -14.7091 3.3023 -4.454 8.47e-06 \*\*\*  
## weekdayMonday -6.2270 3.4064 -1.828 0.067566 .   
## weekdayTuesday -3.7064 3.3201 -1.116 0.264280   
## weekdayWednesday -2.5092 3.3124 -0.758 0.448755   
## weekdayThursday 2.9937 3.3148 0.903 0.366478   
## weekdayFriday 3.8107 3.3081 1.152 0.249359   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 116.8 on 17334 degrees of freedom  
## Multiple R-squared: 0.5861, Adjusted R-squared: 0.5851   
## F-statistic: 557.9 on 44 and 17334 DF, p-value: < 2.2e-16

set.seed(1234) #set random number seed for cross validation  
modelCV2 = train(count ~ season + mnth + hr + holiday + weekday, data = training, method = "lm", trControl = ctrl, metric="Rsquared")  
  
summary(modelCV)

##   
## Call:  
## lm(formula = .outcome ~ ., data = dat)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -497.65 -61.50 -9.66 55.72 507.05   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -39.8155 5.6728 -7.019 2.32e-12 \*\*\*  
## seasonSummer 31.2462 5.5647 5.615 1.99e-08 \*\*\*  
## seasonFall 46.0328 6.5610 7.016 2.36e-12 \*\*\*  
## seasonWinter 78.7081 5.5757 14.116 < 2e-16 \*\*\*  
## mnth2 18.1036 4.4445 4.073 4.66e-05 \*\*\*  
## mnth3 50.4133 4.7762 10.555 < 2e-16 \*\*\*  
## mnth4 64.5009 7.0723 9.120 < 2e-16 \*\*\*  
## mnth5 100.5891 7.0485 14.271 < 2e-16 \*\*\*  
## mnth6 112.3618 6.8486 16.407 < 2e-16 \*\*\*  
## mnth7 94.8390 7.8587 12.068 < 2e-16 \*\*\*  
## mnth8 99.5632 7.8631 12.662 < 2e-16 \*\*\*  
## mnth9 94.3764 7.2128 13.085 < 2e-16 \*\*\*  
## mnth10 51.8409 7.0700 7.332 2.36e-13 \*\*\*  
## mnth11 7.3305 7.0896 1.034 0.301160   
## mnth12 -0.5872 5.6354 -0.104 0.917017   
## hr1 -20.4207 6.1367 -3.328 0.000878 \*\*\*  
## hr2 -31.7342 6.1560 -5.155 2.56e-07 \*\*\*  
## hr3 -44.8673 6.1962 -7.241 4.64e-13 \*\*\*  
## hr4 -50.3097 6.1962 -8.119 5.00e-16 \*\*\*  
## hr5 -34.8743 6.1517 -5.669 1.46e-08 \*\*\*  
## hr6 22.1481 6.1345 3.610 0.000307 \*\*\*  
## hr7 158.2666 6.1303 25.817 < 2e-16 \*\*\*  
## hr8 305.2130 6.1303 49.787 < 2e-16 \*\*\*  
## hr9 165.5115 6.1303 26.999 < 2e-16 \*\*\*  
## hr10 119.8705 6.1303 19.554 < 2e-16 \*\*\*  
## hr11 154.3451 6.1303 25.177 < 2e-16 \*\*\*  
## hr12 199.6516 6.1282 32.579 < 2e-16 \*\*\*  
## hr13 199.9511 6.1261 32.639 < 2e-16 \*\*\*  
## hr14 187.2392 6.1261 30.564 < 2e-16 \*\*\*  
## hr15 197.5231 6.1261 32.243 < 2e-16 \*\*\*  
## hr16 258.3975 6.1240 42.194 < 2e-16 \*\*\*  
## hr17 407.8660 6.1240 66.601 < 2e-16 \*\*\*  
## hr18 371.8649 6.1282 60.681 < 2e-16 \*\*\*  
## hr19 257.8772 6.1282 42.080 < 2e-16 \*\*\*  
## hr20 172.3841 6.1282 28.130 < 2e-16 \*\*\*  
## hr21 118.6684 6.1282 19.364 < 2e-16 \*\*\*  
## hr22 77.6890 6.1282 12.677 < 2e-16 \*\*\*  
## hr23 34.1849 6.1282 5.578 2.47e-08 \*\*\*  
## holidayHoliday -14.6620 5.5903 -2.623 0.008730 \*\*   
## weekdaySunday -14.7091 3.3023 -4.454 8.47e-06 \*\*\*  
## weekdayMonday -6.2270 3.4064 -1.828 0.067566 .   
## weekdayTuesday -3.7064 3.3201 -1.116 0.264280   
## weekdayWednesday -2.5092 3.3124 -0.758 0.448755   
## weekdayThursday 2.9937 3.3148 0.903 0.366478   
## weekdayFriday 3.8107 3.3081 1.152 0.249359   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 116.8 on 17334 degrees of freedom  
## Multiple R-squared: 0.5861, Adjusted R-squared: 0.5851   
## F-statistic: 557.9 on 44 and 17334 DF, p-value: < 2.2e-16