# Model Validation

## May 25, 2020

### Stephen Kiser

Libraries

library(tidyverse)

## -- Attaching packages ------------------------------- tidyverse 1.3.0 --

## v ggplot2 3.3.0 v purrr 0.3.3  
## v tibble 2.1.3 v dplyr 0.8.5  
## v tidyr 1.0.2 v stringr 1.4.0  
## v readr 1.3.1 v forcats 0.5.0

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

library(MASS)

##   
## Attaching package: 'MASS'

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

library(caret)

## Loading required package: lattice

##   
## Attaching package: 'caret'

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

Read-in Dataset

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()  
## )

Convert variables

bike <- bike %>% mutate(season = as\_factor(as.character(season))) %>%   
 mutate(season = fct\_recode(season,  
 "Spring" = "1",  
 "Summer" = "2",  
 "Fall" = "3",  
 "Winter" = "4"))  
  
bike <- bike %>% mutate(yr = as\_factor(yr), mnth = as\_factor(mnth), hr = as\_factor(hr))  
  
bike <- bike %>% mutate(holiday = as\_factor(as.character(holiday))) %>%  
 mutate(holiday = fct\_recode(holiday,   
 "NotHoliday" = "0",  
 "Holiday" = "1"))  
  
bike <- bike %>% mutate(workingday = as\_factor(as.character(workingday))) %>%  
 mutate(workingday = fct\_recode(workingday,  
 "NotWorkingDay" = "0",  
 "WorkingDay" = "1"))  
  
bike <- bike %>% mutate(weathersit = as\_factor(as.character(weathersit))) %>%  
 mutate(weathersit = fct\_recode(weathersit,   
 "No Precip" = "1",  
 "Misty" = "2",  
 "LightPrecip" = "3",  
 "HeavyPrecip" = "4"))  
  
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"))  
  
glimpse(bike)

## 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> No Precip, No Precip, No Precip, No Precip, No Precip, M...  
## $ 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, ...

### Task 1

set.seed(1234)  
  
train.rows = createDataPartition(y=bike$count, p=0.7, list = FALSE)  
  
train = slice(bike, train.rows)  
test = slice(bike, -train.rows)

### Task 2

The number of rows in the training set is 12167 observations, and in the test set there is 5212 observations.

### Task 3

mod1 = lm(count ~ season + mnth + hr + holiday + weekday + temp + weathersit, train)  
summary(mod1)

##   
## Call:  
## lm(formula = count ~ season + mnth + hr + holiday + weekday +   
## temp + weathersit, data = train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -419.31 -61.93 -9.98 52.57 504.24   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -81.2946 6.9356 -11.721 < 2e-16 \*\*\*  
## seasonSummer 28.8486 6.4074 4.502 6.78e-06 \*\*\*  
## seasonFall 19.7865 7.6029 2.602 0.009266 \*\*   
## seasonWinter 62.0339 6.4333 9.643 < 2e-16 \*\*\*  
## mnth2 -0.8013 5.1396 -0.156 0.876114   
## mnth3 2.5584 5.7973 0.441 0.659003   
## mnth4 -1.2250 8.6334 -0.142 0.887166   
## mnth5 -1.5879 9.2279 -0.172 0.863382   
## mnth6 -15.3992 9.4846 -1.624 0.104485   
## mnth7 -38.8277 10.6085 -3.660 0.000253 \*\*\*  
## mnth8 -16.8557 10.3542 -1.628 0.103569   
## mnth9 5.4060 9.2152 0.587 0.557459   
## mnth10 -2.7341 8.5079 -0.321 0.747943   
## mnth11 -12.8043 8.2169 -1.558 0.119193   
## mnth12 -15.3615 6.5409 -2.349 0.018864 \*   
## hr1 -19.7855 6.9722 -2.838 0.004550 \*\*   
## hr2 -28.2440 6.9696 -4.052 5.10e-05 \*\*\*  
## hr3 -40.3146 7.0910 -5.685 1.34e-08 \*\*\*  
## hr4 -40.5469 7.0249 -5.772 8.03e-09 \*\*\*  
## hr5 -26.7454 6.9592 -3.843 0.000122 \*\*\*  
## hr6 32.8518 7.0435 4.664 3.13e-06 \*\*\*  
## hr7 161.3872 6.9925 23.080 < 2e-16 \*\*\*  
## hr8 312.2263 6.9502 44.923 < 2e-16 \*\*\*  
## hr9 164.2556 7.0163 23.411 < 2e-16 \*\*\*  
## hr10 107.1856 6.9552 15.411 < 2e-16 \*\*\*  
## hr11 139.6256 7.0057 19.930 < 2e-16 \*\*\*  
## hr12 179.7448 6.9778 25.760 < 2e-16 \*\*\*  
## hr13 178.6812 7.0201 25.453 < 2e-16 \*\*\*  
## hr14 156.2811 7.0628 22.127 < 2e-16 \*\*\*  
## hr15 168.7543 7.0939 23.788 < 2e-16 \*\*\*  
## hr16 228.1106 7.0881 32.182 < 2e-16 \*\*\*  
## hr17 377.6085 7.0185 53.802 < 2e-16 \*\*\*  
## hr18 347.7287 6.9806 49.813 < 2e-16 \*\*\*  
## hr19 238.7339 7.0128 34.043 < 2e-16 \*\*\*  
## hr20 159.7394 7.0231 22.745 < 2e-16 \*\*\*  
## hr21 108.1070 6.9494 15.556 < 2e-16 \*\*\*  
## hr22 72.3808 6.9874 10.359 < 2e-16 \*\*\*  
## hr23 32.5734 6.9996 4.654 3.30e-06 \*\*\*  
## holidayHoliday -29.0249 6.4088 -4.529 5.98e-06 \*\*\*  
## weekdaySunday -14.0349 3.7638 -3.729 0.000193 \*\*\*  
## weekdayMonday -6.5302 3.8944 -1.677 0.093604 .   
## weekdayTuesday -7.2790 3.8319 -1.900 0.057509 .   
## weekdayWednesday -3.2707 3.7984 -0.861 0.389212   
## weekdayThursday -1.7267 3.8053 -0.454 0.650004   
## weekdayFriday 1.3251 3.7744 0.351 0.725539   
## temp 288.1743 12.1860 23.648 < 2e-16 \*\*\*  
## weathersitMisty -19.6696 2.3717 -8.293 < 2e-16 \*\*\*  
## weathersitLightPrecip -94.1331 3.8166 -24.664 < 2e-16 \*\*\*  
## weathersitHeavyPrecip -80.2490 64.7672 -1.239 0.215356   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 111.9 on 12118 degrees of freedom  
## Multiple R-squared: 0.6217, Adjusted R-squared: 0.6202   
## F-statistic: 414.8 on 48 and 12118 DF, p-value: < 2.2e-16

The adjusted R-squared value is 0.6202. This R-squared value is high, but the closer we can get to 1 the better the model would fit. This linear regression has several months and weekdays that are not significant to the count of bike rentals. We see that weathersit, tep, holiday, hr, season are significant variables in the effect of bike rentals.

### Task 4

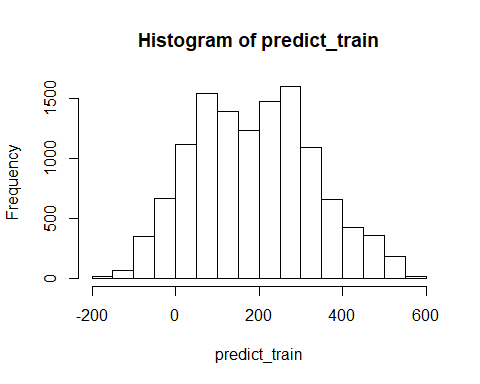
predict\_train <- predict(mod1, newdata = train)  
head(predict\_train)

## 1 2 3 4 5 6   
## -37.68169 -46.14026 -52.44730 -52.67962 -58.54772 14.95557

summary(predict\_train)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## -183.90 75.04 188.25 189.33 289.19 584.44

hist(predict\_train)



The predictions are negative for the first 5 rows. This must mean that the variables had a negative effect on the chance of renting a bike. As we can see in the histogram our predictions go from -200 to 600 as values. We have the majority of our bike rentals in the training set 50 to 350 of our bell curve. The distribution of predictions seems reasonable for this data.

### Task 5

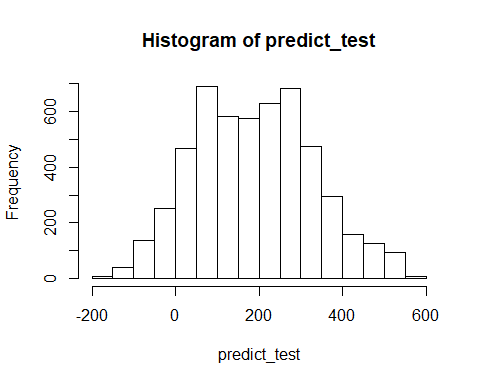
predict\_test <- predict(mod1, newdata = test)  
head(predict\_test)

## 1 2 3 4 5 6   
## -12.13272 137.72755 174.04493 17.56108 -22.20993 168.48847

summary(predict\_test)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## -187.00 78.66 188.34 189.41 288.72 567.39

hist(predict\_test)



As we can see in our head function the top 6 predictions of our data set, test. The testing set compared to our training set shows off higher predicted values of bike rentals. However, our values for median and mean are about the same. There is only a 0.08 difference in the mean values. When we look at the histogram the majority of our bell curve is around the 50 to 350 bike rentals. The distribution for these predictions is reasonable.

### Task 6

SSE = sum((test$count - predict\_test)^2)  
SST = sum((test$count - mean(test$count))^2)  
1 - SSE/SST

## [1] 0.6289223

This value is very close to the R-squared value that our training set model gives us. Our model gives us 0.6217 as the R-squared value in the training set.

### Task 7

K-fold cross validation is different from training/testing splits because it has k-amount of models that it takes data from. The typical k-values are 3,5,and 10. It gathers the data for the training set by grabbing the first 4 partitions of the first model and then the last partition for the testing set. Then for the second model it grabs the first 3 and last partition for the training set then adds the the 4th partition to the testing set. It keeps doing this until we reach the k-amount of folds. The testing/training split takes 70-80% of the data for the training set then the last 20-30% for the testing set.