# Model Validation

## Ariel Romero

Task 1

library(tidyverse)

## -- Attaching packages ------- tidyverse 1.2.1 --

## v ggplot2 3.1.0 v purrr 0.3.2   
## v tibble 2.1.1 v dplyr 0.8.0.1  
## v tidyr 0.8.3 v stringr 1.4.0   
## v readr 1.3.1 v forcats 0.4.0

## -- 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

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()  
## )

bike = bike %>% mutate(season = as\_factor(as.character(season))) %>%  
mutate(season = fct\_recode(season,  
"Spring" = "1",  
"Summer" = "2",  
"Fall" = "3",  
"Winter" = "4")) %>% mutate(yr =as\_factor(yr)) %>%   
 mutate(mnth=as\_factor(mnth)) %>% mutate(hr=as\_factor(hr)) %>% mutate(holiday = as\_factor(as.character(holiday))) %>%  
mutate(holiday = fct\_recode(holiday,  
"NotHoliday" = "0",  
"Holiday" = "1")) %>% mutate(workingday = as\_factor(as.character(workingday))) %>%  
mutate(workingday = fct\_recode(workingday,  
"NotWorkingDay" = "0",  
"WorkingDay" = "1")) %>% mutate(weathersit = as\_factor(as.character(weathersit))) %>%  
mutate(weathersit = fct\_recode(weathersit,  
"NoPrecip" = "1",  
"Misty" = "2", "LightPrecip"= "3","HeavyPrecip" = "4")) %>%  
 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"))  
  
  
set.seed(1234)  
train.rows= createDataPartition(y=bike$count, p=.7,list = FALSE)  
  
train = bike[train.rows,]  
test = bike[-train.rows,]

Task 2: In the testing dataset, there are 5,212 rows. In the training dataset, there are 12167 rows.

Task 3:

mod1 <- lm(count ~ season + mnth + hr + holiday + weekday + temp + weathersit, bike)  
summary(mod1)

##   
## Call:  
## lm(formula = count ~ season + mnth + hr + holiday + weekday +   
## temp + weathersit, data = bike)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -424.63 -62.16 -9.71 51.92 499.33   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -85.278 5.808 -14.684 < 2e-16 \*\*\*  
## seasonSummer 35.545 5.306 6.699 2.17e-11 \*\*\*  
## seasonFall 26.998 6.289 4.293 1.77e-05 \*\*\*  
## seasonWinter 65.129 5.330 12.219 < 2e-16 \*\*\*  
## mnth2 1.323 4.287 0.309 0.75768   
## mnth3 5.078 4.818 1.054 0.29187   
## mnth4 -6.014 7.152 -0.841 0.40041   
## mnth5 -5.832 7.647 -0.763 0.44566   
## mnth6 -18.097 7.850 -2.305 0.02116 \*   
## mnth7 -41.455 8.813 -4.704 2.58e-06 \*\*\*  
## mnth8 -21.251 8.572 -2.479 0.01318 \*   
## mnth9 4.316 7.622 0.566 0.57119   
## mnth10 -3.922 7.079 -0.554 0.57954   
## mnth11 -18.304 6.823 -2.683 0.00731 \*\*   
## mnth12 -15.180 5.411 -2.805 0.00503 \*\*   
## hr1 -17.912 5.849 -3.062 0.00220 \*\*   
## hr2 -26.901 5.868 -4.584 4.59e-06 \*\*\*  
## hr3 -37.809 5.909 -6.399 1.61e-10 \*\*\*  
## hr4 -41.087 5.912 -6.950 3.78e-12 \*\*\*  
## hr5 -24.888 5.873 -4.238 2.27e-05 \*\*\*  
## hr6 33.488 5.858 5.717 1.10e-08 \*\*\*  
## hr7 169.440 5.850 28.963 < 2e-16 \*\*\*  
## hr8 310.710 5.845 53.160 < 2e-16 \*\*\*  
## hr9 164.653 5.845 28.170 < 2e-16 \*\*\*  
## hr10 111.648 5.853 19.075 < 2e-16 \*\*\*  
## hr11 139.110 5.870 23.697 < 2e-16 \*\*\*  
## hr12 180.131 5.889 30.588 < 2e-16 \*\*\*  
## hr13 176.032 5.907 29.801 < 2e-16 \*\*\*  
## hr14 160.344 5.924 27.067 < 2e-16 \*\*\*  
## hr15 169.807 5.931 28.632 < 2e-16 \*\*\*  
## hr16 231.354 5.925 39.050 < 2e-16 \*\*\*  
## hr17 384.495 5.907 65.086 < 2e-16 \*\*\*  
## hr18 351.933 5.892 59.735 < 2e-16 \*\*\*  
## hr19 241.539 5.870 41.147 < 2e-16 \*\*\*  
## hr20 161.120 5.858 27.506 < 2e-16 \*\*\*  
## hr21 110.339 5.848 18.868 < 2e-16 \*\*\*  
## hr22 72.378 5.843 12.387 < 2e-16 \*\*\*  
## hr23 33.232 5.841 5.689 1.30e-08 \*\*\*  
## holidayHoliday -26.140 5.335 -4.899 9.71e-07 \*\*\*  
## weekdaySunday -15.873 3.148 -5.043 4.64e-07 \*\*\*  
## weekdayMonday -7.779 3.248 -2.395 0.01663 \*   
## weekdayTuesday -6.528 3.172 -2.058 0.03960 \*   
## weekdayWednesday -3.805 3.166 -1.202 0.22940   
## weekdayThursday -2.393 3.165 -0.756 0.44960   
## weekdayFriday 1.631 3.154 0.517 0.60515   
## temp 287.864 10.218 28.173 < 2e-16 \*\*\*  
## weathersitMisty -19.377 1.981 -9.782 < 2e-16 \*\*\*  
## weathersitLightPrecip -90.772 3.168 -28.650 < 2e-16 \*\*\*  
## weathersitHeavyPrecip -78.721 64.407 -1.222 0.22163   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 111.3 on 17330 degrees of freedom  
## Multiple R-squared: 0.6242, Adjusted R-squared: 0.6232   
## F-statistic: 599.8 on 48 and 17330 DF, p-value: < 2.2e-16

The linear regression model to predict “count”" using the variables stated appearrs to be somewhat of a decent model. The adjusted R squared value is 0.6232 which in my opinion would deem acceptable.

Task 4 & 5

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

## 1 2 3 4 5 6   
## -39.85978 -48.84889 -53.99969 -57.27775 -60.45532 11.54028

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

## 1 2 3 4 5 6   
## -16.19038 171.49170 157.46335 215.75597 203.79481 11.88948

Task: 6

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

## [1] 0.6284994

The R square value on the testing set, 0.6285, is very close to to the training model R square value of 0.6232. The model is a appears to be a consistent predictor for the data.

Task 7: In training/ testing splitting, the data is split into two subsets, typically 70% training and 30% testing. With k-fold cross validation, the data set is split in to k separate parts. The training process is then repeated k times. Each instance, one part is used as as the validation data and the rest is used for a training model.