# Alex Fields

## BAN 502

### Module 3 - Assignment 1

#### Task 2 - Q&A

How many rows of data are in each set (training and testing)? **Training contains 12627 (~%70) while Test contains 5212 (~%30) of the rows**

#### Task 3 - Q&A

The R^2 value for the lm model is roughly 0.618. This value is relatively good. This is showing a lower residual error in the model.

#### Task 4 - Q&A

The Created R^2 for this is 0.6201. This values in Train dataset are very close to each other compared to the lm and the manually created R^2 value.

The prediction of the histogram seems to be normally distributed.

#### Task 5 - Q&A

The model from the Test dataset seems to be very similar to the Train dataset. Everything seems to be normally distributed. We can see that all items that are non-categorical are correlated logically. Example: Percipiation causing negative count while positive temp is showing positive count.

#### Task 6 - Q&A

We can see that our predicted R^2 value for the Test data is better fitted than the Training data. Training is showing ~0.618 while the manually calculated version was ~0.63. This shows that there is less residual errror in the test which seems to show a promising model.

#### Task 7 - Q&A

Using KFolds seems to be a better way to fit models. The R^2 value for the modCV (cross-validation) is close to perfect (~0.94). This differs since the training and test data was split into a ratio, 70-30.

options(tidyverse.quiet = TRUE)  
  
library(tidyverse)  
library(MASS)  
library(caret)  
library(GGally)

hour <- read\_csv("hour.csv")  
bike = as\_tibble(hour)  
  
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(as.character(yr))) %>%  
mutate(yr = fct\_recode(yr,  
"0" = "0",  
"1" = "1"))  
  
  
# We decided to convert yr, mnth and hr variables into factors to convert them to categorical variables.   
# These values are not logically quantitative. Even though they are numbers they represent a categorical/string value.   
# We should not be running models on "Tuesday".   
  
bike = bike %>% mutate(mnth = as\_factor(as.character(mnth))) %>%  
mutate(mnth = fct\_recode(mnth,  
"1" = "1",  
"2" = "2",  
"3" = "3",  
"4" = "4",  
"5" = "5",  
"6" = "6",  
"7" = "7",  
"8" = "8",  
"9" = "9",  
"10" = "10",  
"11" = "11",  
"12" = "12"))  
  
bike = bike %>% mutate(hr = as\_factor(as.character(hr))) %>%  
mutate(hr = fct\_recode(hr,  
"0" = "0",  
"1" = "1",  
"2" = "2",  
"3" = "3",  
"4" = "4",  
"5" = "5",  
"6" = "6",  
"7" = "7",  
"8" = "8",  
"9" = "9",  
"10" = "10",  
"11" = "11",  
"12" = "12",  
"13" = "13",  
"14" = "14",  
"15" = "15",  
"16" = "16",  
"17" = "17",  
"18" = "18",  
"19" = "19",  
"20" = "20",  
"21" = "21",  
"22" = "22",  
"23" = "23"))  
  
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,  
"NoPrecip" = "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"))  
  
bike = bike %>% drop\_na() #drops N/A's  
str(bike)

## tibble [17,379 x 17] (S3: tbl\_df/tbl/data.frame)  
## $ instant : num [1:17379] 1 2 3 4 5 6 7 8 9 10 ...  
## $ dteday : chr [1:17379] "1/1/2011" "1/1/2011" "1/1/2011" "1/1/2011" ...  
## $ season : Factor w/ 4 levels "Spring","Summer",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ yr : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...  
## $ mnth : Factor w/ 12 levels "1","2","3","4",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ hr : Factor w/ 24 levels "0","1","2","3",..: 1 2 3 4 5 6 7 8 9 10 ...  
## $ holiday : Factor w/ 2 levels "NotHoliday","Holiday": 1 1 1 1 1 1 1 1 1 1 ...  
## $ weekday : Factor w/ 7 levels "Saturday","Sunday",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ workingday: Factor w/ 2 levels "NotWorkingDay",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ weathersit: Factor w/ 4 levels "NoPrecip","Misty",..: 1 1 1 1 1 2 1 1 1 1 ...  
## $ temp : num [1:17379] 0.24 0.22 0.22 0.24 0.24 0.24 0.22 0.2 0.24 0.32 ...  
## $ atemp : num [1:17379] 0.288 0.273 0.273 0.288 0.288 ...  
## $ hum : num [1:17379] 0.81 0.8 0.8 0.75 0.75 0.75 0.8 0.86 0.75 0.76 ...  
## $ windspeed : num [1:17379] 0 0 0 0 0 0.0896 0 0 0 0 ...  
## $ casual : num [1:17379] 3 8 5 3 0 0 2 1 1 8 ...  
## $ registered: num [1:17379] 13 32 27 10 1 1 0 2 7 6 ...  
## $ count : num [1:17379] 16 40 32 13 1 1 2 3 8 14 ...

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

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

train\_pred = predict(mod1, newdata = train)  
  
SSE = sum((train$count - train\_pred)^2)  
SST = sum((train$count - mean(train$count))^2)  
1 - SSE/SST

## [1] 0.6216738

summary(train\_pred)

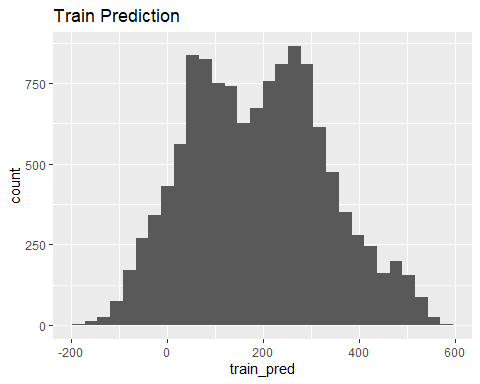
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## -183.90 75.04 188.25 189.33 289.19 584.44

head(train\_pred)

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

ggplot(train,aes(x=train\_pred)) + geom\_histogram() + labs(title = "Train Prediction")

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

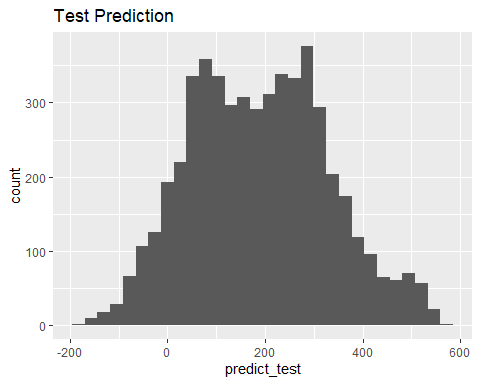


predict\_test = predict(mod1, newdata = test)  
summary(predict\_test)

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

ggplot(test, aes(x=predict\_test)) + geom\_histogram() + labs(title = "Test Prediction")

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

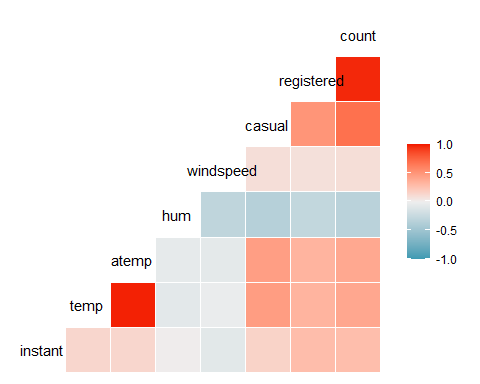


SSE = sum((test$count - predict\_test)^2)  
SST = sum((test$count - mean(test$count))^2)  
1 - SSE/SST

## [1] 0.6289223

ggcorr(bike)#Determining what variables are most correlated with Count

## Warning in ggcorr(bike): data in column(s) 'dteday', 'season', 'yr', 'mnth',  
## 'hr', 'holiday', 'weekday', 'workingday', 'weathersit' are not numeric and were  
## ignored



ctrl = trainControl(method = "cv", number = 10)  
  
set.seed(123)  
modCV = train(count ~ registered, bike, method = "lm", trControl = ctrl, metric = "Rsquared")  
summary(modCV)

##   
## Call:  
## lm(formula = .outcome ~ ., data = dat)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -118.031 -16.264 -9.957 4.730 304.223   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 10.296458 0.459718 22.4 <2e-16 \*\*\*  
## registered 1.165032 0.002131 546.8 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 42.51 on 17377 degrees of freedom  
## Multiple R-squared: 0.9451, Adjusted R-squared: 0.9451   
## F-statistic: 2.99e+05 on 1 and 17377 DF, p-value: < 2.2e-16