Bike Sharing Analysis Report

Brian Boughton and Brian Tulimero

2023-02-04

pacman::p\_load(pacman, dplyr,forecast,car,ROSE,rpart)

# Milestone 1

## Introduction

The data set used was collected by Hadi fanaee from the laboratory of Artificial Intelligence and Decision Support at the University of Porto in Portugal. The contains information around bike sharing. It consist of an entry number, date, year, season, month, day of the week, hour, and whether or not it was a weekend. It also contains information around weather like temperature, wind speed and, forecast. Lastly, it gives the amount of casual and registered bike borrows along the total number.

Our goal with this data is to make predictions of future rentals based on day information and predicted forecasts. There are a few unnecessary variables in the dataset that have been removed like the instant number and date along with the casual and registered users because we are currently only interested in the overall users.

## Data Wrangling

### Importing data and setting needed categories to factors and removing the uneeded variables.

pj.dt<-read.csv("hour.csv")  
pj.df<-read.csv("hour.csv")  
pj.df$season<- as.factor(pj.df$season)  
pj.df$yr<- as.factor(pj.df$yr)  
pj.df$mnth<- as.factor(pj.df$mnth)  
pj.df$hr<- as.factor(pj.df$hr)  
pj.df$holiday<- as.factor(pj.df$holiday)  
pj.df$weekday<- as.factor(pj.df$weekday)  
pj.df$workingday<- as.factor(pj.df$workingday)  
pj.df$weathersit<- as.factor(pj.df$weathersit)  
pj<-subset(pj.df, select = -c(instant, dteday,casual,registered))

The final data frame is called simply pj for project and there are other version of the dataset if needed. Description of each feature: \* instant: record index \* dteday : date \* season : season (1:spring, 2:summer, 3:fall, 4:winter) \* yr : year (0: 2011, 1:2012) \* mnth : month ( 1 to 12) \* hr : hour (0 to 23) \* holiday : weather day is holiday or not (extracted from <http://dchr.dc.gov/page/holiday-schedule>) \* weekday : day of the week \* workingday : if day is neither weekend nor holiday is 1, otherwise is 0. \* weathersit : \* - 1: Clear, Few clouds, Partly cloudy, Partly cloudy \* - 2: Mist + Cloudy, Mist + Broken clouds, Mist + Few clouds, Mist \* - 3: Light Snow, Light Rain + Thunderstorm + Scattered clouds, Light Rain + Scattered clouds \* - 4: Heavy Rain + Ice Pallets + Thunderstorm + Mist, Snow + Fog \* temp : Normalized temperature in Celsius. The values are divided to 41 (max) \* atemp: Normalized feeling temperature in Celsius. The values are divided to 50 (max) \* hum: Normalized humidity. The values are divided to 100 (max) \* windspeed: Normalized wind speed. The values are divided to 67 (max) \* casual: count of casual users \* registered: count of registered users \* cnt: count of total rental bikes including both casual and registered

### Getting to understand the data.

summary(pj)

## season yr mnth hr holiday weekday workingday  
## 1:4242 0:8645 5 :1488 16 : 730 0:16879 0:2502 0: 5514   
## 2:4409 1:8734 7 :1488 17 : 730 1: 500 1:2479 1:11865   
## 3:4496 12 :1483 13 : 729 2:2453   
## 4:4232 8 :1475 14 : 729 3:2475   
## 3 :1473 15 : 729 4:2471   
## 10 :1451 12 : 728 5:2487   
## (Other):8521 (Other):13004 6:2512   
## weathersit temp atemp hum windspeed   
## 1:11413 Min. :0.020 Min. :0.0000 Min. :0.0000 Min. :0.0000   
## 2: 4544 1st Qu.:0.340 1st Qu.:0.3333 1st Qu.:0.4800 1st Qu.:0.1045   
## 3: 1419 Median :0.500 Median :0.4848 Median :0.6300 Median :0.1940   
## 4: 3 Mean :0.497 Mean :0.4758 Mean :0.6272 Mean :0.1901   
## 3rd Qu.:0.660 3rd Qu.:0.6212 3rd Qu.:0.7800 3rd Qu.:0.2537   
## Max. :1.000 Max. :1.0000 Max. :1.0000 Max. :0.8507   
##   
## cnt   
## Min. : 1.0   
## 1st Qu.: 40.0   
## Median :142.0   
## Mean :189.5   
## 3rd Qu.:281.0   
## Max. :977.0   
##

#showing number of Entries and features  
dim(pj)

## [1] 17379 13

#displaying the number of missing entries  
sum(is.na(pj))

## [1] 0

#showing numeric data features  
names(select\_if(pj, is.numeric))

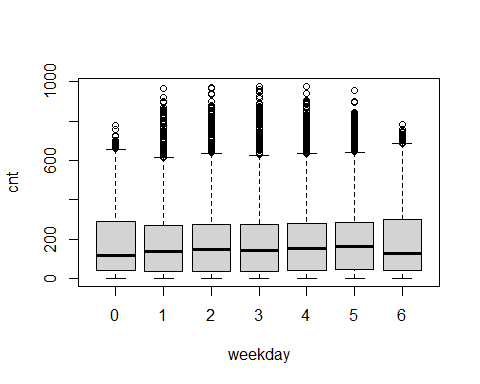
## [1] "temp" "atemp" "hum" "windspeed" "cnt"

#showing categorical data features  
names(select\_if(pj, is.factor))

## [1] "season" "yr" "mnth" "hr" "holiday"   
## [6] "weekday" "workingday" "weathersit"

We can see that the data consist of 13 different variables and 17,379 observations. There is not missing data entries found within the data. The numeric features are temp, atemp, hum, windspeed, and cnt. the categorical features are season, yr, mnth, hr, holiday, weekday, workingday, and weathersit.

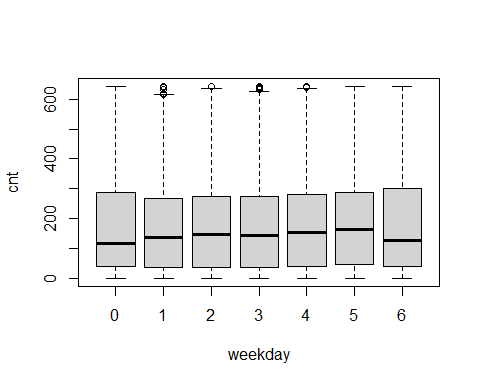
with(pj, plot(cnt~weekday))

 The above boxplot shows that there are an extreme amount of outliars

### Model development

winsor<-function(cnt){  
 #Defining the normal range  
Q3<-quantile(cnt,0.75,na.rm = TRUE)  
Q1<-quantile(cnt,0.25,na.rm = TRUE)  
upper<-Q3+1.5\*(Q3-Q1)  
lower<-Q1-1.5\*(Q3-Q1)  
#replacing values that exceed upper and lower normal ranges  
winsor.cnt<-ifelse(cnt>upper,  
 upper,   
 ifelse(cnt<lower,  
 lower,  
 cnt))  
return(winsor.cnt)  
}  
#imputing cleaned data into raw data set.  
pj$cnt<-winsor(pj$cnt)

with(pj, plot(cnt~weekday))



The number outliars now has been lowered based on the unconditional mean

### Creating training and test data frames

set.seed(1318)  
#Randomly choosing 80% of the data set  
train.rows<- sample(rownames(pj), dim(pj)[1]\*0.8)  
#Create the training set  
train<-pj[train.rows,]  
#Repeat for Test set  
valid.rows<-setdiff(rownames(pj),train.rows)  
valid<-pj[valid.rows,]

**Summaries**

summary(train)

## season yr mnth hr holiday weekday workingday  
## 1:3399 0:6898 7 :1203 15 : 597 0:13511 0:1993 0:4374   
## 2:3492 1:7005 8 :1194 17 : 596 1: 392 1:1964 1:9529   
## 3:3647 3 :1188 21 : 595 2:1986   
## 4:3365 12 :1178 23 : 594 3:1970   
## 5 :1172 1 : 592 4:2008   
## 9 :1157 19 : 591 5:1993   
## (Other):6811 (Other):10338 6:1989   
## weathersit temp atemp hum windspeed   
## 1:9153 Min. :0.0200 Min. :0.0000 Min. :0.0000 Min. :0.0000   
## 2:3634 1st Qu.:0.3400 1st Qu.:0.3333 1st Qu.:0.4800 1st Qu.:0.1045   
## 3:1113 Median :0.5000 Median :0.4848 Median :0.6300 Median :0.1940   
## 4: 3 Mean :0.4972 Mean :0.4759 Mean :0.6266 Mean :0.1897   
## 3rd Qu.:0.6600 3rd Qu.:0.6212 3rd Qu.:0.7800 3rd Qu.:0.2537   
## Max. :1.0000 Max. :1.0000 Max. :1.0000 Max. :0.8060   
##   
## cnt   
## Min. : 1.0   
## 1st Qu.: 40.0   
## Median :143.0   
## Mean :186.9   
## 3rd Qu.:282.0   
## Max. :642.5   
##

# dimensions of train data set  
dim(train)

## [1] 13903 13

summary(valid)

## season yr mnth hr holiday weekday workingday  
## 1:843 0:1747 5 : 316 0 : 159 0:3368 0:509 0:1140   
## 2:917 1:1729 12 : 305 7 : 159 1: 108 1:515 1:2336   
## 3:849 4 : 297 10 : 159 2:467   
## 4:867 10 : 297 14 : 156 3:505   
## 6 : 292 18 : 156 4:463   
## 3 : 285 6 : 152 5:494   
## (Other):1684 (Other):2535 6:523   
## weathersit temp atemp hum windspeed   
## 1:2260 Min. :0.0200 Min. :0.0303 Min. :0.0000 Min. :0.0000   
## 2: 910 1st Qu.:0.3400 1st Qu.:0.3333 1st Qu.:0.4700 1st Qu.:0.1045   
## 3: 306 Median :0.5000 Median :0.4848 Median :0.6400 Median :0.1940   
## 4: 0 Mean :0.4961 Mean :0.4754 Mean :0.6297 Mean :0.1917   
## 3rd Qu.:0.6400 3rd Qu.:0.6212 3rd Qu.:0.7900 3rd Qu.:0.2537   
## Max. :0.9600 Max. :0.9242 Max. :1.0000 Max. :0.8507   
##   
## cnt   
## Min. : 1.0   
## 1st Qu.: 38.0   
## Median :138.5   
## Mean :184.3   
## 3rd Qu.:275.0   
## Max. :642.5   
##

# dimensions of valid data set  
dim(valid)

## [1] 3476 13

### Creating a linear regression

# Creating null model  
null<-lm(cnt~1,data=train)  
#Creating Full model  
full<-lm(cnt~.,data=train)  
#Using stepwise function to find the optimal variable configuration for the model  
opti<-step(  
 full,  
 scope = list(upper = full, lower = null),  
 direction = "both",  
 trace = FALSE  
 )  
summary(opti)

##   
## Call:  
## lm(formula = cnt ~ season + yr + mnth + hr + holiday + weekday +   
## weathersit + temp + atemp + hum + windspeed, data = train)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -367.49 -58.05 -6.65 51.61 387.99   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -78.673 6.894 -11.411 < 2e-16 \*\*\*  
## season2 40.402 5.039 8.017 1.17e-15 \*\*\*  
## season3 34.259 5.997 5.713 1.14e-08 \*\*\*  
## season4 66.694 5.091 13.102 < 2e-16 \*\*\*  
## yr1 79.205 1.622 48.819 < 2e-16 \*\*\*  
## mnth2 5.711 4.065 1.405 0.160084   
## mnth3 12.349 4.553 2.712 0.006693 \*\*   
## mnth4 3.799 6.788 0.560 0.575715   
## mnth5 18.343 7.273 2.522 0.011671 \*   
## mnth6 1.754 7.463 0.235 0.814180   
## mnth7 -19.232 8.378 -2.295 0.021721 \*   
## mnth8 -1.253 8.180 -0.153 0.878255   
## mnth9 23.482 7.281 3.225 0.001263 \*\*   
## mnth10 9.256 6.734 1.374 0.169324   
## mnth11 -8.848 6.478 -1.366 0.172006   
## mnth12 -3.823 5.164 -0.740 0.459142   
## hr1 -20.157 5.544 -3.636 0.000278 \*\*\*  
## hr2 -28.345 5.593 -5.068 4.08e-07 \*\*\*  
## hr3 -39.588 5.604 -7.064 1.69e-12 \*\*\*  
## hr4 -40.362 5.660 -7.131 1.05e-12 \*\*\*  
## hr5 -25.009 5.612 -4.456 8.40e-06 \*\*\*  
## hr6 35.363 5.604 6.311 2.86e-10 \*\*\*  
## hr7 170.546 5.609 30.405 < 2e-16 \*\*\*  
## hr8 299.802 5.565 53.875 < 2e-16 \*\*\*  
## hr9 164.404 5.579 29.471 < 2e-16 \*\*\*  
## hr10 109.626 5.635 19.454 < 2e-16 \*\*\*  
## hr11 132.981 5.637 23.592 < 2e-16 \*\*\*  
## hr12 170.755 5.691 30.007 < 2e-16 \*\*\*  
## hr13 164.795 5.754 28.642 < 2e-16 \*\*\*  
## hr14 152.093 5.800 26.224 < 2e-16 \*\*\*  
## hr15 164.052 5.745 28.556 < 2e-16 \*\*\*  
## hr16 221.235 5.769 38.351 < 2e-16 \*\*\*  
## hr17 344.472 5.702 60.414 < 2e-16 \*\*\*  
## hr18 322.626 5.717 56.432 < 2e-16 \*\*\*  
## hr19 236.707 5.623 42.098 < 2e-16 \*\*\*  
## hr20 154.979 5.621 27.573 < 2e-16 \*\*\*  
## hr21 106.067 5.557 19.088 < 2e-16 \*\*\*  
## hr22 69.909 5.562 12.570 < 2e-16 \*\*\*  
## hr23 30.890 5.541 5.575 2.53e-08 \*\*\*  
## holiday1 -18.337 5.097 -3.598 0.000322 \*\*\*  
## weekday1 4.098 3.091 1.326 0.184971   
## weekday2 5.858 3.005 1.949 0.051285 .   
## weekday3 7.056 3.013 2.342 0.019211 \*   
## weekday4 9.082 2.999 3.028 0.002467 \*\*   
## weekday5 14.650 3.002 4.879 1.08e-06 \*\*\*  
## weekday6 15.944 2.994 5.325 1.03e-07 \*\*\*  
## weathersit2 -9.852 1.992 -4.946 7.67e-07 \*\*\*  
## weathersit3 -65.771 3.378 -19.469 < 2e-16 \*\*\*  
## weathersit4 -55.962 54.628 -1.024 0.305656   
## temp 132.843 29.820 4.455 8.46e-06 \*\*\*  
## atemp 110.392 30.841 3.579 0.000346 \*\*\*  
## hum -75.707 5.748 -13.170 < 2e-16 \*\*\*  
## windspeed -29.366 7.316 -4.014 6.01e-05 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 94.31 on 13850 degrees of freedom  
## Multiple R-squared: 0.7003, Adjusted R-squared: 0.6992   
## F-statistic: 622.5 on 52 and 13850 DF, p-value: < 2.2e-16

accuracy(opti$fitted.values, train$cnt)

## ME RMSE MAE MPE MAPE  
## Test set 4.786584e-15 94.12853 71.40409 60.16987 277.3954

pred <- predict(opti, newdata = valid)  
accuracy(pred, valid$cnt)

## ME RMSE MAE MPE MAPE  
## Test set -1.27783 96.02195 72.22612 55.66991 268.9787

### Creating a logistic Model

#Creating binary values for cnt.  
pj.2<-pj  
#Finding the mean of the cnt data  
mean<-mean(pj.2$cnt)  
#replaceing values above or equal to mean with '1' and all below with '0'  
pj.2$cnt<-with(pj.2, ifelse(cnt>=mean,1,0))  
  
#getting total value of entries with value 1 and seeing the percentage.  
sum(pj.2$cnt) # 7045 values were equal to or above the mean

## [1] 7045

sum(pj.2$cnt)/length(pj.2$cnt) # These values make up about 40% of the data

## [1] 0.4053743

set.seed(1318)  
#Randomly choosing 60% of the data set  
train.rows<- sample(rownames(pj.2), dim(pj.2)[1]\*0.6)  
#Create the training set  
train.2<-pj.2[train.rows,]  
#Repeat for Test set  
valid.rows<-setdiff(rownames(pj.2),train.rows)  
valid.2<-pj.2[valid.rows,]

# Creating null model  
null2<-glm(cnt~1,data=train.2)  
#Creating Full model  
full2<-glm(cnt~.,data=train.2)  
#Using stepwise function to find the optimal variable configuration for the model  
opti2<-step(  
 full2,  
 scope = list(upper = full2, lower = null2),  
 direction = "both",  
 trace = FALSE, family = binomial  
 )  
summary(opti2)

##   
## Call:  
## glm(formula = cnt ~ season + yr + mnth + hr + holiday + weekday +   
## weathersit + temp + atemp + hum + windspeed, data = train.2)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -0.98627 -0.22041 0.02726 0.24274 0.94837   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -0.380240 0.027706 -13.724 < 2e-16 \*\*\*  
## season2 0.083034 0.020175 4.116 3.89e-05 \*\*\*  
## season3 0.097162 0.023991 4.050 5.16e-05 \*\*\*  
## season4 0.189590 0.020666 9.174 < 2e-16 \*\*\*  
## yr1 0.188008 0.006526 28.809 < 2e-16 \*\*\*  
## mnth2 0.030906 0.016312 1.895 0.058162 .   
## mnth3 0.068787 0.018421 3.734 0.000189 \*\*\*  
## mnth4 0.056157 0.027204 2.064 0.039014 \*   
## mnth5 0.095322 0.029159 3.269 0.001083 \*\*   
## mnth6 0.038130 0.029858 1.277 0.201621   
## mnth7 -0.010150 0.033601 -0.302 0.762606   
## mnth8 0.026287 0.032764 0.802 0.422398   
## mnth9 0.059710 0.029265 2.040 0.041345 \*   
## mnth10 0.020421 0.027254 0.749 0.453699   
## mnth11 -0.003029 0.026235 -0.115 0.908088   
## mnth12 0.020496 0.020890 0.981 0.326555   
## hr1 -0.009459 0.022258 -0.425 0.670865   
## hr2 -0.003180 0.022659 -0.140 0.888383   
## hr3 -0.003228 0.022568 -0.143 0.886260   
## hr4 0.007166 0.022528 0.318 0.750414   
## hr5 0.019253 0.022658 0.850 0.395505   
## hr6 0.054653 0.022642 2.414 0.015807 \*   
## hr7 0.551320 0.022658 24.332 < 2e-16 \*\*\*  
## hr8 0.667253 0.022496 29.661 < 2e-16 \*\*\*  
## hr9 0.629819 0.022182 28.394 < 2e-16 \*\*\*  
## hr10 0.310019 0.022755 13.624 < 2e-16 \*\*\*  
## hr11 0.411440 0.022630 18.181 < 2e-16 \*\*\*  
## hr12 0.582380 0.023100 25.211 < 2e-16 \*\*\*  
## hr13 0.554109 0.023240 23.842 < 2e-16 \*\*\*  
## hr14 0.487737 0.023170 21.050 < 2e-16 \*\*\*  
## hr15 0.520468 0.023060 22.570 < 2e-16 \*\*\*  
## hr16 0.680536 0.023036 29.543 < 2e-16 \*\*\*  
## hr17 0.761166 0.022878 33.270 < 2e-16 \*\*\*  
## hr18 0.746114 0.022936 32.530 < 2e-16 \*\*\*  
## hr19 0.691522 0.022835 30.284 < 2e-16 \*\*\*  
## hr20 0.559135 0.022669 24.666 < 2e-16 \*\*\*  
## hr21 0.415275 0.022363 18.570 < 2e-16 \*\*\*  
## hr22 0.161786 0.022561 7.171 7.94e-13 \*\*\*  
## hr23 0.026679 0.022305 1.196 0.231676   
## holiday1 -0.032043 0.020280 -1.580 0.114130   
## weekday1 0.025853 0.012357 2.092 0.036445 \*   
## weekday2 0.041380 0.012074 3.427 0.000612 \*\*\*  
## weekday3 0.034532 0.012079 2.859 0.004260 \*\*   
## weekday4 0.041010 0.012048 3.404 0.000667 \*\*\*  
## weekday5 0.086703 0.012089 7.172 7.90e-13 \*\*\*  
## weekday6 0.042384 0.012080 3.509 0.000452 \*\*\*  
## weathersit2 -0.022655 0.007983 -2.838 0.004549 \*\*   
## weathersit3 -0.170717 0.013542 -12.607 < 2e-16 \*\*\*  
## weathersit4 -0.184571 0.190479 -0.969 0.332575   
## temp 0.191049 0.120734 1.582 0.113590   
## atemp 0.470505 0.125169 3.759 0.000172 \*\*\*  
## hum -0.197121 0.023077 -8.542 < 2e-16 \*\*\*  
## windspeed -0.081654 0.029387 -2.779 0.005470 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for gaussian family taken to be 0.1079108)  
##   
## Null deviance: 2519.6 on 10426 degrees of freedom  
## Residual deviance: 1119.5 on 10374 degrees of freedom  
## AIC: 6430.2  
##   
## Number of Fisher Scoring iterations: 2

actual.train<-train.2$cnt  
pred.train<-predict(opti2,train.2,type='response')  
actual.valid<-valid.2$cnt  
pred.valid<-predict(opti2,valid.2,type='response')  
(conf.matrix1<-table(actual.train, pred.train>.5))

##   
## actual.train FALSE TRUE  
## 0 5239 928  
## 1 484 3776

(conf.matrix2<-table(actual.valid, pred.valid>.5))

##   
## actual.valid FALSE TRUE  
## 0 3520 647  
## 1 332 2453

Accuracy, sensitivity, and specificity for training data

#Accuracy  
sum(diag(conf.matrix1))/nrow(train.2)

## [1] 0.8645823

#sensitivity  
conf.matrix1[2,2]/ sum(conf.matrix1[2,])

## [1] 0.886385

#specificity  
conf.matrix1[1,1]/sum(conf.matrix1[1,])

## [1] 0.8495216

Accuracy, sensitivity, and specificity for test data

#Accuracy  
sum(diag(conf.matrix2))/nrow(valid.2)

## [1] 0.8591772

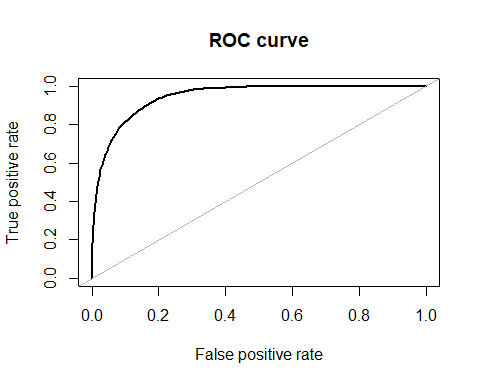
#sensitivity  
conf.matrix2[2,2]/ sum(conf.matrix2[2,])

## [1] 0.8807899

#specificity  
conf.matrix2[1,1]/sum(conf.matrix2[1,])

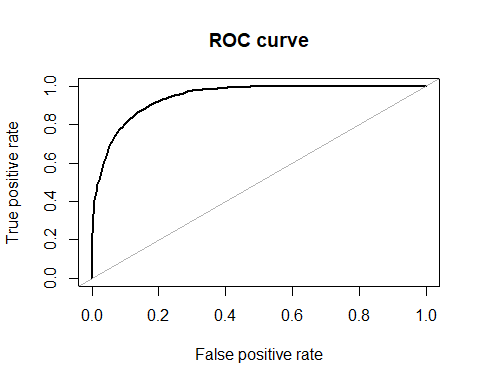
## [1] 0.8447324

#Training AUC  
roc.curve(response =actual.train, predicted = pred.train)



## Area under the curve (AUC): 0.946

#Valid AUC  
roc.curve(response =actual.valid, predicted = pred.valid)



## Area under the curve (AUC): 0.944