Homework 2

Group 8 Aniket Sahane Shubham Jagtap Yash Shah Mehul Sanyal Samiha Umme

2/26/2020

1. Load required packages

```
pacman::p_load(caret, corrplot, glmnet, mlbench, tidyverse, ggplot2,
goeveg, reshape, leaps, data.table,dplyr,forecast,MASS)
search()
```

```
##  [1] ".GlobalEnv"         "package:MASS"       "package:forecast"
##  [4] "package:data.table" "package:leaps"      "package:reshape"
##  [7] "package:goeveg"     "package:forcats"    "package:stringr"
## [10] "package:dplyr"      "package:purrr"      "package:readr"
## [13] "package:tidyr"      "package:tibble"     "package:tidyverse"
## [16] "package:mlbench"    "package:glmnet"     "package:Matrix"
## [19] "package:corrplot"   "package:caret"      "package:ggplot2"
## [22] "package:lattice"    "package:pacman"     "package:stats"
## [25] "package:graphics"   "package:grDevices"  "package:utils"
## [28] "package:datasets"   "package:methods"    "Autoloads"
## [31] "package:base"
```

```
theme_set(theme_classic())
```

2. Read the file 'Airfares.csv'

```
airfares.df <- read.csv("Airfares.csv")
str(airfares.df)
```

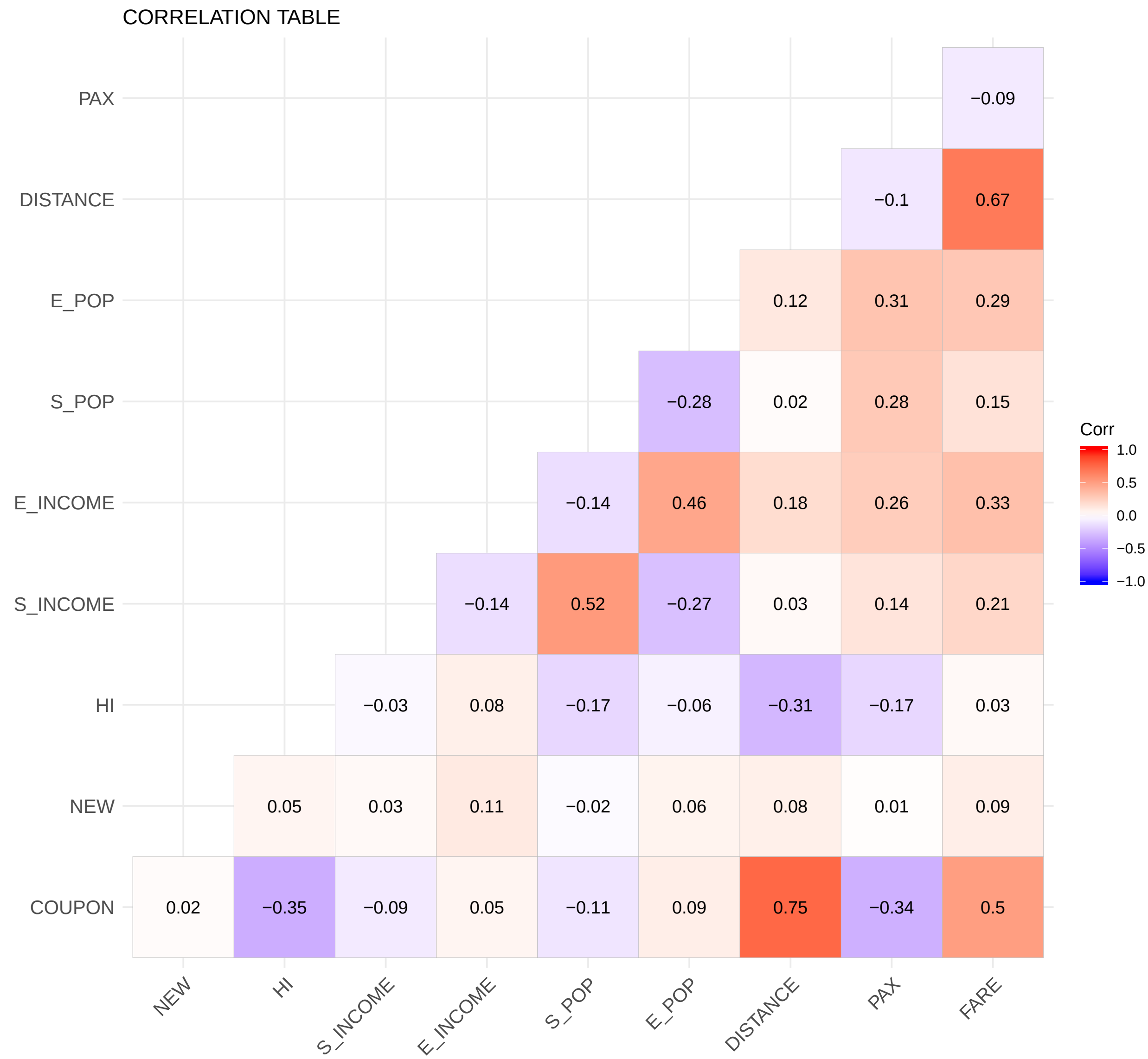
```
## 'data.frame':    638 obs. of  18 variables:
##  $ S_CODE  : Factor w/ 8 levels "*","DCA","EWR",..: 1 1 1 8 7 1 1 1 1 1 ...
##  $ S_CITY  : Factor w/ 51 levels "Albuquerque         NM",..: 14 3 7 9 9 11 14 18 23 25 ...
##  $ E_CODE  : Factor w/ 8 levels "*","DCA","EWR",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ E_CITY  : Factor w/ 68 levels "Amarillo            TX",..: 1 2 2 2 2 2 2 2 2 2 ...
##  $ COUPON  : num  1 1.06 1.06 1.06 1.06 1.01 1.28 1.15 1.33 1.6 ...
##  $ NEW     : int  3 3 3 3 3 3 3 3 3 2 ...
##  $ VACATION: Factor w/ 2 levels "No","Yes": 1 1 1 1 1 1 1 2 1 1 ...
##  $ SW      : Factor w/ 2 levels "No","Yes": 2 1 1 2 2 2 1 2 2 2 ...
##  $ HI      : num  5292 5419 9185 2657 2657 ...
##  $ S_INCOME: num  28637 26993 30124 29260 29260 ...
```

```
##  $ E_INCOME: num  21112 29838 29838 29838 29838 ...
##  $ S_POP   : int  3036732 3532657 5787293 7830332 7830332 2230955 3036732 1440377 3770125 1694803 ..
##  $ E_POP   : int  205711 7145897 7145897 7145897 7145897 7145897 7145897 7145897 7145897 7145897 ...
##  $ SLOT    : Factor w/ 2 levels "Controlled","Free": 2 2 2 1 2 2 2 2 2 2 ...
##  $ GATE    : Factor w/ 2 levels "Constrained",..: 2 2 2 2 2 2 2 2 2 2 ...
##  $ DISTANCE: int  312 576 364 612 612 309 1220 921 1249 964 ...
##  $ PAX     : int  7864 8820 6452 25144 25144 13386 4625 5512 7811 4657 ...
##  $ FARE    : num  64.1 174.5 207.8 85.5 85.5 ...
```

Question 1

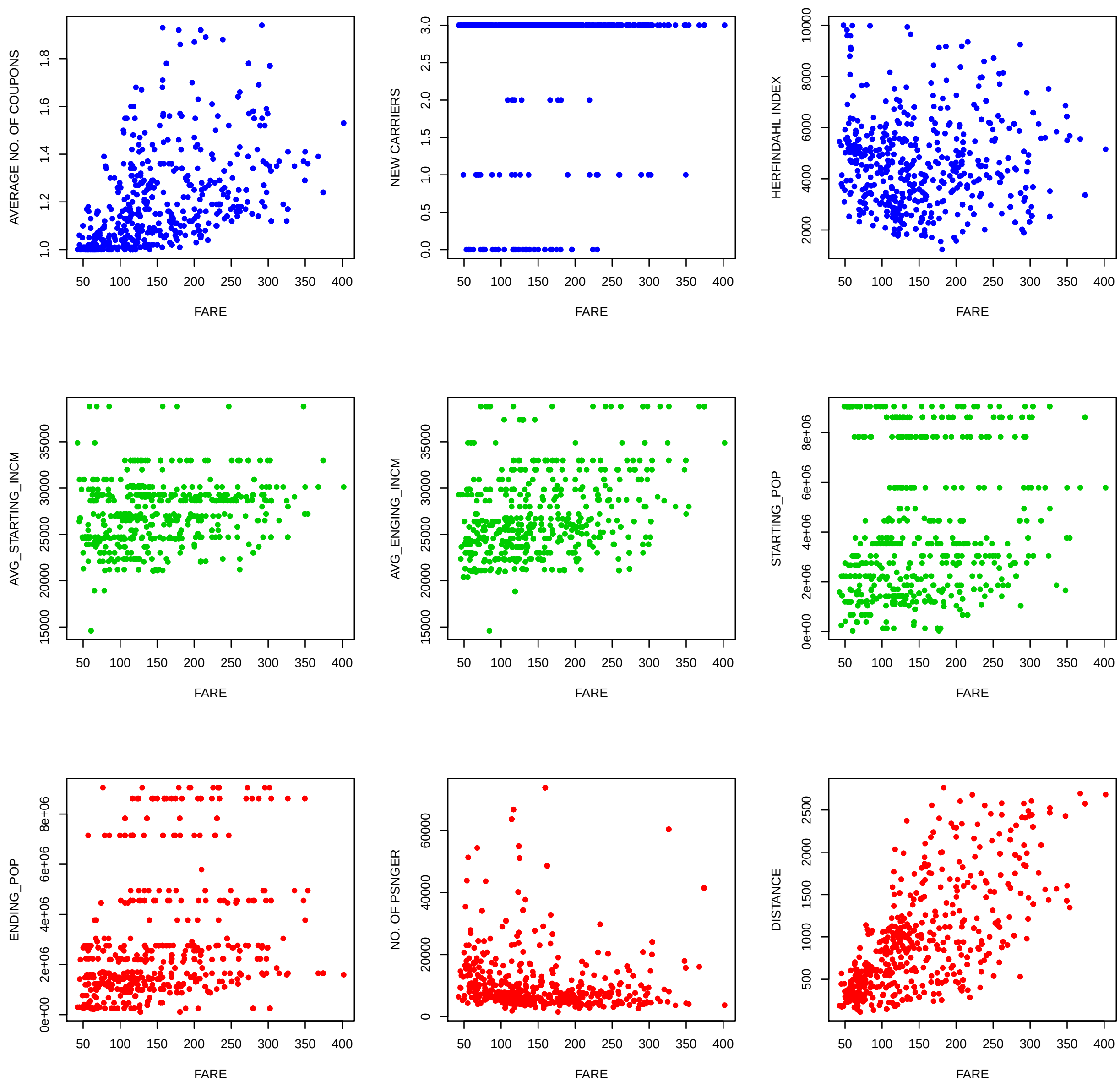
Create a correlation table and scatterplots between FARE and the predictors. What seems to be the best single predictor of FARE? Explain your answer.

```
library(ggcorrplot)
ggcorrplot(cor(airfares.df[,unlist(lapply(airfares.df, is.numeric))]), type='lower', lab=TRUE , title=
```




```r
par(mfrow = c(3,3))

plot(airfares.df$FARE,airfares.df$COUPON,pch=16,col=4, xlab="FARE",ylab="AVERAGE NO. OF COUPONS")

plot(airfares.df$FARE,airfares.df$NEW,pch=16,col=4, xlab="FARE",ylab="NEW CARRIERS")

plot(airfares.df$FARE,airfares.df$HI,pch=16,col=4, xlab="FARE",ylab="HERFINDAHL INDEX")

plot(airfares.df$FARE,airfares.df$S_INCOME,pch=16,col=3, xlab="FARE",ylab="AVG_STARTING_INCM")

plot(airfares.df$FARE,airfares.df$E_INCOME,pch=16,col=3, xlab="FARE",ylab="AVG_ENGING_INCM")

plot(airfares.df$FARE,airfares.df$S_POP,pch=16,col=3,xlab="FARE",ylab="STARTING_POP")
```

```r
plot(airfares.df$FARE,airfares.df$E_POP,pch=16,col=2,xlab="FARE",ylab="ENDING_POP")

plot(airfares.df$FARE,airfares.df$PAX,pch=16,col=2,xlab="FARE",ylab="NO. OF PSNGER")

plot(airfares.df$FARE,airfares.df$DISTANCE,pch=16,col=2,xlab="FARE",ylab="DISTANCE")
```



Explanation [1]

From the correlation table we can say that "Distance" has the highest positive correlation with Fare. This can also be observed from scatter plot of "Distance" and Fare that they have positive linear relationship. Hence we can say that "Distance" seems to be the best single predictor of FARE

Question 2

Explore the categorical predictors by computing the percentage of flights in each category. Create a pivot table with the average fare in each category. Which categorical predictor seems best for predicting FARE? Explain your answer

```
vacation <- transform(as.data.frame(table(airfares.df$VACATION)),
Percentage=Freq/nrow(airfares.df)*100)
pivot_vacation <- airfares.df %>%
group_by(VACATION) %>% summarize(AVG_FARE=mean(FARE))
print(pivot_vacation)
```

```
## # A tibble: 2 x 2
##   VACATION AVG_FARE
##   <fct>       <dbl>
## 1 No           174.
## 2 Yes          126.
```

```
sw <- transform(as.data.frame(table(airfares.df$SW)),
Percentage=Freq/nrow(airfares.df)*100)
pivot_sw <- airfares.df %>%
group_by(SW) %>% summarize(AVG_FARE=mean(FARE))
print(pivot_sw)
```

```
## # A tibble: 2 x 2
##   SW    AVG_FARE
##   <fct>    <dbl>
## 1 No       188.
## 2 Yes       98.4
```

```
slot <- transform(as.data.frame(table(airfares.df$SLOT)),
Percentage=Freq/nrow(airfares.df)*100)
pivot_slot <- airfares.df %>%
group_by(SLOT) %>% summarize(AVG_FARE=mean(FARE))
pivot_slot
```

```
## # A tibble: 2 x 2
##   SLOT       AVG_FARE
##   <fct>         <dbl>
## 1 Controlled     186.
## 2 Free           151.
```

```
gate <- transform(as.data.frame(table(airfares.df$GATE)),
Percentage=Freq/nrow(airfares.df)*100)
pivot_gate <- airfares.df %>%
group_by(GATE) %>% summarize(AVG_FARE=mean(FARE))
pivot_gate
```

```
## # A tibble: 2 x 2
##   GATE        AVG_FARE
##   <fct>          <dbl>
## 1 Constrained     193.
## 2 Free            153.
```

Explanation[2]

From the above pivot tables, it is clear that average fare of SW is 98.38 (SW=YES) whereas if it's not SW the average price is much higher i.e. 188.18, thus SW affects the price fare the most. It is also clear that "Southwest airlines" seems to be the best predictor for predicting fare.We can observe that the average FARE of SW is spread.

Question 3

Create data partition by assigning 80% of the records to the training dataset. Use rounding if 80% of the index generates a fraction. Also, set the seed at 42.

```
airf.df<-airfares.df[ ,-c(1:4)]
set.seed(42)
train.index <- sample(1:nrow(airf.df), 0.8 *round(nrow(airf.df)))
train.df <- airf.df[train.index, ]
test.df <- airf.df[-train.index, ]
```

Question 4

Using leaps package, run stepwise regression to reduce the number of predictors. Discuss the results from this model

```
library(leaps)
airfares.lm <- lm(FARE ~ ., data = train.df)
airfares.step <-  regsubsets(FARE ~ ., data = train.df, nbest = 1, nvmax = dim(train.df)[2], method = "
summary(airfares.step)$which
```

```
##    (Intercept) COUPON   NEW VACATIONYes SWYes    HI S_INCOME E_INCOME S_POP
## 1         TRUE  FALSE FALSE       FALSE FALSE FALSE    FALSE    FALSE FALSE
## 2         TRUE  FALSE FALSE       FALSE  TRUE FALSE    FALSE    FALSE FALSE
## 3         TRUE  FALSE FALSE        TRUE  TRUE FALSE    FALSE    FALSE FALSE
## 4         TRUE  FALSE FALSE        TRUE  TRUE  TRUE    FALSE    FALSE FALSE
## 5         TRUE  FALSE FALSE        TRUE  TRUE  TRUE    FALSE    FALSE FALSE
## 6         TRUE  FALSE FALSE        TRUE  TRUE  TRUE    FALSE    FALSE FALSE
## 7         TRUE  FALSE FALSE        TRUE  TRUE  TRUE    FALSE     TRUE FALSE
## 8         TRUE  FALSE FALSE        TRUE  TRUE  TRUE    FALSE     TRUE  TRUE
## 9         TRUE  FALSE FALSE        TRUE  TRUE  TRUE    FALSE    FALSE  TRUE
## 10        TRUE   TRUE  TRUE        TRUE  TRUE  TRUE     TRUE     TRUE  TRUE
## 11        TRUE  FALSE  TRUE        TRUE  TRUE  TRUE    FALSE     TRUE  TRUE
## 12        TRUE  FALSE  TRUE        TRUE  TRUE  TRUE     TRUE     TRUE  TRUE
## 13        TRUE   TRUE  TRUE        TRUE  TRUE  TRUE     TRUE     TRUE  TRUE
##    E_POP SLOTFree GATEFree DISTANCE   PAX
## 1  FALSE    FALSE    FALSE     TRUE FALSE
## 2  FALSE    FALSE    FALSE     TRUE FALSE
## 3  FALSE    FALSE    FALSE     TRUE FALSE
## 4  FALSE    FALSE    FALSE     TRUE FALSE
## 5  FALSE     TRUE    FALSE     TRUE FALSE
## 6  FALSE     TRUE     TRUE     TRUE FALSE
## 7  FALSE     TRUE     TRUE     TRUE FALSE
```

```
## 8   TRUE    FALSE    FALSE      TRUE  TRUE
## 9   TRUE     TRUE     TRUE      TRUE  TRUE
## 10  TRUE     TRUE    FALSE     FALSE FALSE
## 11  TRUE     TRUE     TRUE      TRUE  TRUE
## 12  TRUE     TRUE     TRUE      TRUE  TRUE
## 13  TRUE     TRUE     TRUE      TRUE  TRUE
```

```r
print("The R-squared Values:")
```

```
## [1] "The R-squared Values:"
```

```r
summary(airfares.step)$rsq
```

```
##  [1] 0.4168069 0.5793894 0.6966218 0.7232479 0.7366555 0.7565835 0.7604199
##  [8] 0.7674947 0.7748171 0.6303171 0.7809073 0.7813501 0.7816700
```

```r
print("The Adjusted R-squared Values:")
```

```
## [1] "The Adjusted R-squared Values:"
```

```r
summary(airfares.step)$adjr2
```

```
##  [1] 0.4156589 0.5777302 0.6948231 0.7210558 0.7340429 0.7536799 0.7570792
##  [8] 0.7637820 0.7707638 0.6229086 0.7760679 0.7760708 0.7759476
```

```r
print("The Cp Values:")
```

```
## [1] "The Cp Values:"
```

```r
summary(airfares.step)$cp
```

```
##  [1] 818.89220 451.53899 187.21153 128.72255 100.26346  56.99127  50.27558
##  [8]  36.20326  21.56831 351.84190  11.73270  12.72670  14.00000
```

Explantaion[4]

We can interpret this model by taking into consideration the Adjusted R-square and Mallow's Cp values. As seen from above Adjusted R-square values there is no significant increase in adjusted r-square after considering 11 variables (0.7760).The Mallow's Cp value for 11 variables in our model is 11.7320 which is closest to the ideal value of 12 according to the formula (p+1). Therefore according to stepwise search the best variables for predicting FARE are NEW, VACATION, SW, HI, E_INCOME, S_POP, E_POP, SLOT, GATE, DISTANCE, PAX.

Question 5

Repeat the process in (4) using exhaustive search instead of stepwise regression.Compare the resulting best model to the one you obtained in (4) in terms of thepredictors included in the final model.

```r
library(leaps)
airfares.exhaust <- regsubsets(FARE ~., data = train.df, nbest = 1, nvmax = dim(train.df)[2], method =

sum <- summary(airfares.exhaust)
sum$which
```

```
##    (Intercept) COUPON   NEW VACATIONYes SWYes    HI S_INCOME E_INCOME S_POP
## 1         TRUE  FALSE FALSE       FALSE FALSE FALSE    FALSE    FALSE FALSE
## 2         TRUE  FALSE FALSE       FALSE  TRUE FALSE    FALSE    FALSE FALSE
## 3         TRUE  FALSE FALSE        TRUE  TRUE FALSE    FALSE    FALSE FALSE
## 4         TRUE  FALSE FALSE        TRUE  TRUE  TRUE    FALSE    FALSE FALSE
## 5         TRUE  FALSE FALSE        TRUE  TRUE  TRUE    FALSE    FALSE FALSE
## 6         TRUE  FALSE FALSE        TRUE  TRUE  TRUE    FALSE    FALSE FALSE
## 7         TRUE  FALSE FALSE        TRUE  TRUE  TRUE    FALSE    FALSE  TRUE
## 8         TRUE  FALSE FALSE        TRUE  TRUE  TRUE    FALSE     TRUE  TRUE
## 9         TRUE  FALSE FALSE        TRUE  TRUE  TRUE    FALSE    FALSE  TRUE
## 10        TRUE  FALSE FALSE        TRUE  TRUE  TRUE    FALSE     TRUE  TRUE
## 11        TRUE  FALSE  TRUE        TRUE  TRUE  TRUE    FALSE     TRUE  TRUE
## 12        TRUE  FALSE  TRUE        TRUE  TRUE  TRUE     TRUE     TRUE  TRUE
## 13        TRUE   TRUE  TRUE        TRUE  TRUE  TRUE     TRUE     TRUE  TRUE
##    E_POP SLOTFree GATEFree DISTANCE   PAX
## 1  FALSE    FALSE    FALSE     TRUE FALSE
## 2  FALSE    FALSE    FALSE     TRUE FALSE
## 3  FALSE    FALSE    FALSE     TRUE FALSE
## 4  FALSE    FALSE    FALSE     TRUE FALSE
## 5  FALSE     TRUE    FALSE     TRUE FALSE
## 6  FALSE     TRUE     TRUE     TRUE FALSE
## 7   TRUE    FALSE    FALSE     TRUE  TRUE
## 8   TRUE    FALSE    FALSE     TRUE  TRUE
## 9   TRUE     TRUE     TRUE     TRUE  TRUE
## 10  TRUE     TRUE     TRUE     TRUE  TRUE
## 11  TRUE     TRUE     TRUE     TRUE  TRUE
## 12  TRUE     TRUE     TRUE     TRUE  TRUE
## 13  TRUE     TRUE     TRUE     TRUE  TRUE
```

```r
sum$rsq
```

```
##  [1] 0.4168069 0.5793894 0.6966218 0.7232479 0.7366555 0.7565835 0.7607777
##  [8] 0.7674947 0.7748171 0.7803115 0.7809073 0.7813501 0.7816700
```

```r
sum$adjr2
```

```
##  [1] 0.4156589 0.5777302 0.6948231 0.7210558 0.7340429 0.7536799 0.7574419
##  [8] 0.7637820 0.7707638 0.7759090 0.7760679 0.7760708 0.7759476
```

```r
sum$cp
```

```
##  [1] 818.89220 451.53899 187.21153 128.72255 100.26346  56.99127  49.46286
##  [8]  36.20326  21.56831  11.08605  11.73270  12.72670  14.00000
```

Explanation[5]:

We can interpret this model by taking into consideration the Adjusted R-square and Mallow's Cp values. As seen from above Adjusted R-square values there is no significant increase in adjusted r-square after considering 10 variables (0.7759) . The Mallow's Cp value for 10 variables in our model is 11.08605 which is closest to the ideal value of 11 according to the formula (p+1). Therefore according to stepwise search the best variables for predicting FARE are VACATION, SW, HI, E_INCOME, S_POP, E_POP, SLOT, GATE, DISTANCE, PAX.

Question 6

Compare the predictive accuracy of both models—stepwise regression and exhaustive search—using measures such as RMSE.

```r
print("Stepwise Search")
```

```
## [1] "Stepwise Search"
```

```r
stepwise.lm<-lm(formula = FARE ~ NEW+ VACATION + SW + HI + E_INCOME + S_POP + E_POP +SLOT + GATE + DISTA
stepwise.lm.pred <- predict(stepwise.lm,test.df)
accuracy(stepwise.lm.pred,test.df$FARE)
```

```
##                ME     RMSE      MAE       MPE     MAPE
## Test set 3.166677 36.82363 27.57897 -5.812025 21.44043
```

```r
print("Exhaustive Search")
```

```
## [1] "Exhaustive Search"
```

```r
exhaustive.lm<-lm(formula = FARE ~ VACATION + SW + HI + E_INCOME + S_POP + E_POP + SLOT + GATE + DISTAN
exhaustive.lm.pred <- predict(exhaustive.lm,test.df)
accuracy(exhaustive.lm.pred,test.df$FARE)
```

```
##               ME    RMSE      MAE       MPE     MAPE
## Test set 3.06081 36.8617 27.70568 -5.938062 21.62142
```

Explanation[6]

RMSE is a measure of how spread out the residuals are, therfore lower the RMSE value signifies a better fit. As seen from above comparison it is evident that stepwise search has slightly low RMSE (36.823) than RMSE value of exhaustive search (36.861). This can also be observed MAE values. Hence stepwise model is a better fit.

Question 7

Using the exhaustive search model, predict the average fare on a route with the following characteristics: COUPON = 1.202, NEW = 3, VACATION = No, SW = No, HI = 4442.141, S_INCOME = $28,760, E_INCOME = $27,664, S_POP = 4,557,004, E_POP = 3,195,503, SLOT = Free, GATE = Free, PAX = 12,782, DISTANCE = 1976 miles.

```
validation.df <- data.frame('COUPON' = 1.202, 'NEW' = 3, 'VACATION' = 'No',
                        'SW' ='No', 'HI' = 4442.141, 'S_INCOME' = 28760,
                        'E_INCOME' = 27664, 'S_POP' =4557004,
                        'E_POP' = 3195503, 'SLOT' = 'Free', 'GATE' = 'Free',
                        'PAX' = 12782,'DISTANCE' = 1976)

exhaustive.lm<-lm(formula = FARE ~ VACATION + SW + HI + E_INCOME + S_POP + E_POP +
                    SLOT + GATE + DISTANCE + PAX, data = train.df)
exhaustive.lm.pred <- predict(exhaustive.lm,validation.df)
exhaustive.lm.pred
```

```
##       1
## 247.684
```

Explanation[7] :

With the given test value of variables the average is 247.684

Question 8

Predict the reduction in average fare on the route in question (7.), if Southwest decides to cover this route using the exhaustive search model above.

```
validation_sw.df <- data.frame('COUPON' = 1.202, 'NEW' = 3, 'VACATION' = 'No', 'SW' =
                          'Yes', 'HI' = 4442.141, 'S_INCOME' = 28760, 'E_INCOME' = 27664,
                          'S_POP'= 4557004, 'E_POP' = 3195503, 'SLOT' = 'Free',
                          'GATE' = 'Free', 'PAX' = 12782,
                          'DISTANCE' = 1976)

exhaustive.lm.pred <- predict(exhaustive.lm,validation_sw.df)
exhaustive.lm.pred
```

```
##        1
## 207.1558
```

Explanation[8]:

Southwest beign the best airlines if it decides to cover the route there is a significant drop in the average price from 247.684 to 207.1558. Hence we can safely say that there is a reduction in average fare.

Question 9

Using leaps package, run backward selection regression to reduce the number of predictors. Discuss the results from this model

```
airfares.back <- regsubsets(FARE ~ ., data = train.df, nbest = 1, nvmax = dim(airfares.df)[2],method =
backward <- summary(airfares.back)
backward$which
```

```
##    (Intercept) COUPON   NEW VACATIONYes SWYes    HI S_INCOME E_INCOME S_POP
## 1         TRUE  FALSE FALSE       FALSE FALSE FALSE    FALSE    FALSE FALSE
## 2         TRUE  FALSE FALSE       FALSE  TRUE FALSE    FALSE    FALSE FALSE
## 3         TRUE  FALSE FALSE        TRUE  TRUE FALSE    FALSE    FALSE FALSE
## 4         TRUE  FALSE FALSE        TRUE  TRUE  TRUE    FALSE    FALSE FALSE
## 5         TRUE  FALSE FALSE        TRUE  TRUE  TRUE    FALSE    FALSE FALSE
## 6         TRUE  FALSE FALSE        TRUE  TRUE  TRUE    FALSE    FALSE  TRUE
## 7         TRUE  FALSE FALSE        TRUE  TRUE  TRUE    FALSE    FALSE  TRUE
## 8         TRUE  FALSE FALSE        TRUE  TRUE  TRUE    FALSE    FALSE  TRUE
## 9         TRUE  FALSE FALSE        TRUE  TRUE  TRUE    FALSE    FALSE  TRUE
## 10        TRUE  FALSE FALSE        TRUE  TRUE  TRUE    FALSE     TRUE  TRUE
## 11        TRUE  FALSE  TRUE        TRUE  TRUE  TRUE    FALSE     TRUE  TRUE
## 12        TRUE  FALSE  TRUE        TRUE  TRUE  TRUE     TRUE     TRUE  TRUE
## 13        TRUE   TRUE  TRUE        TRUE  TRUE  TRUE     TRUE     TRUE  TRUE
##    E_POP SLOTFree GATEFree DISTANCE   PAX
## 1  FALSE    FALSE    FALSE     TRUE FALSE
## 2  FALSE    FALSE    FALSE     TRUE FALSE
## 3  FALSE    FALSE    FALSE     TRUE FALSE
## 4  FALSE    FALSE    FALSE     TRUE FALSE
## 5   TRUE    FALSE    FALSE     TRUE FALSE
## 6   TRUE    FALSE    FALSE     TRUE FALSE
## 7   TRUE    FALSE    FALSE     TRUE  TRUE
## 8   TRUE    FALSE     TRUE     TRUE  TRUE
## 9   TRUE     TRUE     TRUE     TRUE  TRUE
## 10  TRUE     TRUE     TRUE     TRUE  TRUE
## 11  TRUE     TRUE     TRUE     TRUE  TRUE
## 12  TRUE     TRUE     TRUE     TRUE  TRUE
## 13  TRUE     TRUE     TRUE     TRUE  TRUE
```

```
backward$rsq
```

```
##  [1] 0.4168069 0.5793894 0.6966218 0.7232479 0.7322282 0.7509946 0.7607777
##  [8] 0.7663728 0.7748171 0.7803115 0.7809073 0.7813501 0.7816700
```

```
backward$adjr2
```

```
##  [1] 0.4156589 0.5777302 0.6948231 0.7210558 0.7295718 0.7480243 0.7574419
##  [8] 0.7626422 0.7707638 0.7759090 0.7760679 0.7760708 0.7759476
```

```
backward$cp
```

```
##  [1] 818.89220 451.53899 187.21153 128.72255 110.32120  69.68802  49.46286
##  [8]  38.75199  21.56831  11.08605  11.73270  12.72670  14.00000
```

Explanation[9]

We can interpret this backward search model by taking into consideration the Adjusted R-square and Mallow's Cp values. As seen from above Adjusted R-square values there is no significant increase in adjusted r-square after considering 10 variables (0.7759) .Whereas the adjusted r-square of 12 variable is higher than the other variables. The Mallow's Cp value for 10 variables in our model is 11.08605 which is closest to the ideal value of 11 according to the formula (p+1).

VACATION, SW, HI, E_INCOME, S_POP,E_POP, SLOT, GATE, DISTANCE, PAX according to stepwise search are the best variables for predicting FARE.However backward search model in not reccomended when the number of predictor variables is high, as its computation is expensive.

Question 10

Now run a backward selection model using stepAIC() function. Discuss theresults from this model, including the role of AIC in this model.

```
library(MASS)
air.lm<-lm(FARE ~ .,data = train.df)
air.lm<- stepAIC(air.lm,direction = "backward")
```

```
## Start:  AIC=3652.06
## FARE ~ COUPON + NEW + VACATION + SW + HI + S_INCOME + E_INCOME +
##     S_POP + E_POP + SLOT + GATE + DISTANCE + PAX
##
##            Df Sum of Sq     RSS    AIC
## - COUPON    1       911  622732 3650.8
## - NEW       1      1459  623280 3651.3
## - S_INCOME  1      1460  623281 3651.3
## <none>                   621821 3652.1
## - E_INCOME  1     17499  639320 3664.2
## - SLOT      1     17769  639590 3664.4
## - PAX       1     24441  646263 3669.7
## - E_POP     1     28296  650118 3672.8
## - GATE      1     28881  650702 3673.2
## - S_POP     1     36680  658501 3679.3
## - HI        1     76469  698290 3709.2
## - SW        1    105205  727026 3729.8
## - VACATION  1    113382  735204 3735.5
## - DISTANCE  1    417379 1039200 3912.0
##
## Step:  AIC=3650.81
## FARE ~ NEW + VACATION + SW + HI + S_INCOME + E_INCOME + S_POP +
##     E_POP + SLOT + GATE + DISTANCE + PAX
##
##            Df Sum of Sq     RSS    AIC
## - S_INCOME  1      1261  623994 3649.8
## - NEW       1      1678  624410 3650.2
## <none>                   622732 3650.8
## - E_INCOME  1     17126  639859 3662.6
## - SLOT      1     18407  641139 3663.7
## - GATE      1     29285  652018 3672.2
```

```
## - E_POP     1     29484  652217 3672.4
## - PAX       1     34128  656860 3676.0
## - S_POP     1     36089  658821 3677.5
## - HI        1     78594  701326 3709.4
## - SW        1    107735  730468 3730.2
## - VACATION  1    114276  737009 3734.7
## - DISTANCE  1    824468 1447200 4078.9
##
## Step:  AIC=3649.84
## FARE ~ NEW + VACATION + SW + HI + E_INCOME + S_POP + E_POP +
##     SLOT + GATE + DISTANCE + PAX
##
##            Df Sum of Sq     RSS    AIC
## - NEW       1      1697  625690 3649.2
## <none>                   623994 3649.8
## - E_INCOME  1     16167  640161 3660.9
## - SLOT      1     20012  644006 3663.9
## - E_POP     1     28559  652552 3670.7
## - GATE      1     29766  653759 3671.6
## - PAX       1     32869  656863 3674.0
## - S_POP     1     41722  665715 3680.8
## - HI        1     79501  703495 3709.0
## - SW        1    126837  750831 3742.2
## - VACATION  1    128080  752073 3743.1
## - DISTANCE  1    826967 1450960 4078.2
##
## Step:  AIC=3649.22
## FARE ~ VACATION + SW + HI + E_INCOME + S_POP + E_POP + SLOT +
##     GATE + DISTANCE + PAX
##
##            Df Sum of Sq     RSS    AIC
## <none>                   625690 3649.2
## - E_INCOME  1     15649  641339 3659.8
## - SLOT      1     19217  644907 3662.6
## - E_POP     1     28766  654456 3670.1
## - GATE      1     29165  654856 3670.5
## - PAX       1     32706  658396 3673.2
## - S_POP     1     42648  668338 3680.9
## - HI        1     78891  704581 3707.8
## - SW        1    126577  752267 3741.2
## - VACATION  1    127066  752756 3741.5
## - DISTANCE  1    825966 1451656 4076.4
```

```
summary(air.lm)
```

```
##
## Call:
## lm(formula = FARE ~ VACATION + SW + HI + E_INCOME + S_POP + E_POP +
##     SLOT + GATE + DISTANCE + PAX, data = train.df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -99.148 -22.077  -2.028  21.491 107.744
##
```

```
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  4.208e+01  1.476e+01   2.851 0.004534 **
## VACATIONYes -3.876e+01  3.850e+00 -10.067  < 2e-16 ***
## SWYes       -4.053e+01  4.034e+00 -10.047  < 2e-16 ***
## HI           8.268e-03  1.042e-03   7.932 1.43e-14 ***
## E_INCOME     1.445e-03  4.089e-04   3.533 0.000450 ***
## S_POP        4.185e-06  7.176e-07   5.832 9.85e-09 ***
## E_POP        3.779e-06  7.890e-07   4.790 2.21e-06 ***
## SLOTFree    -1.685e+01  4.305e+00  -3.915 0.000103 ***
## GATEFree    -2.122e+01  4.399e+00  -4.823 1.88e-06 ***
## DISTANCE     7.367e-02  2.870e-03  25.666  < 2e-16 ***
## PAX         -7.619e-04  1.492e-04  -5.107 4.66e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 35.41 on 499 degrees of freedom
## Multiple R-squared:  0.7803, Adjusted R-squared:  0.7759
## F-statistic: 177.2 on 10 and 499 DF,  p-value: < 2.2e-16
```

```r
air.lm.pred <- predict(air.lm, train.df)
accuracy(air.lm.pred, train.df$FARE)
```

```
##                     ME     RMSE      MAE       MPE     MAPE
## Test set -9.660619e-14 35.02633 27.75874 -4.446173 20.94924
```

Explanation[10]

Before using stepAIC we had 13 predictors and the start AIC=3652.06.AIC quantifies how much information is lost due to simplification and penalizes the model for including too many predictors. Thus, the preferable model will be the one with the lowest AIC. By running backward seection using step AIC function, we get the best model with 10 predictors which are VACATION, SW, HI, E_INCOME, S_POP, E_POP, SLOT,GATE, DISTANCE and PAX. In first step we eliminated COUPON, in the second we eliminated S_INCOME and in the third step we eliminated NEW predictor.