BUAN 6356 - Homework 2

Group No.9 (Shubhi Kala, Spoorthi Thatipally, Hao-Yu Lin, Loc Nguyen, Tatsat Joshi)

2/24/2020

R Markdown file

Install and load necessary packages and check loading

```
if(!require("pacman")) install.packages("pacman")
pacman::p_load(caret, leaps, forecast, tidyverse, GGally, reshape2, MASS, grid, gridExtra)
search()
theme_set(theme_classic())
```

Read the data from the Airfare.csv

```
airfare.df <- read.csv("Airfares.csv")

# Removing the first 4 predictors from the analysis
airfare.df <- airfare.df[,-c(1:4)]
head(airfare.df)
```

```
##   COUPON NEW VACATION  SW      HI S_INCOME E_INCOME   S_POP   E_POP       SLOT
## 1   1.00   3       No Yes 5291.99    28637    21112 3036732  205711       Free
## 2   1.06   3       No  No 5419.16    26993    29838 3532657 7145897       Free
## 3   1.06   3       No  No 9185.28    30124    29838 5787293 7145897       Free
## 4   1.06   3       No Yes 2657.35    29260    29838 7830332 7145897 Controlled
## 5   1.06   3       No Yes 2657.35    29260    29838 7830332 7145897       Free
## 6   1.01   3       No Yes 3408.11    26046    29838 2230955 7145897       Free
##   GATE DISTANCE   PAX   FARE
## 1 Free      312  7864  64.11
## 2 Free      576  8820 174.47
## 3 Free      364  6452 207.76
## 4 Free      612 25144  85.47
## 5 Free      612 25144  85.47
## 6 Free      309 13386  56.76
```

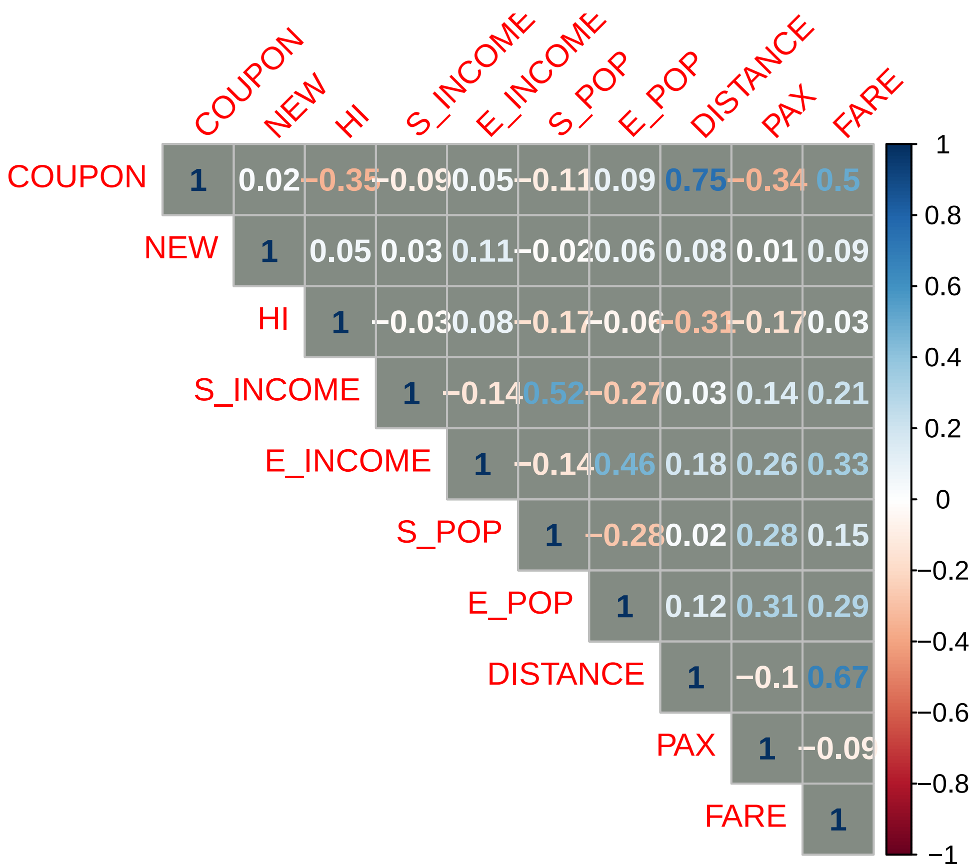
Question 1: Correlation table and scatter plots between FARE and other predictors

```
cor.mat <- round(cor(airfare.df[,-c(3,4,10,11)]),2)  # rounded correlation matrix

# Correlation Table between numeric variables
cor.mat
```

```
##          COUPON   NEW    HI S_INCOME E_INCOME S_POP E_POP DISTANCE   PAX  FARE
## COUPON     1.00  0.02 -0.35    -0.09     0.05 -0.11  0.09     0.75 -0.34  0.50
## NEW        0.02  1.00  0.05     0.03     0.11 -0.02  0.06     0.08  0.01  0.09
## HI        -0.35  0.05  1.00    -0.03     0.08 -0.17 -0.06    -0.31 -0.17  0.03
## S_INCOME  -0.09  0.03 -0.03     1.00    -0.14  0.52 -0.27     0.03  0.14  0.21
## E_INCOME   0.05  0.11  0.08    -0.14     1.00 -0.14  0.46     0.18  0.26  0.33
## S_POP     -0.11 -0.02 -0.17     0.52    -0.14  1.00 -0.28     0.02  0.28  0.15
## E_POP      0.09  0.06 -0.06    -0.27     0.46 -0.28  1.00     0.12  0.31  0.29
## DISTANCE   0.75  0.08 -0.31     0.03     0.18  0.02  0.12     1.00 -0.10  0.67
## PAX       -0.34  0.01 -0.17     0.14     0.26  0.28  0.31    -0.10  1.00 -0.09
## FARE       0.50  0.09  0.03     0.21     0.33  0.15  0.29     0.67 -0.09  1.00
```

```r
# Check correlation between numeric variables
corrplot::corrplot(cor.mat, method = "number",type = "upper",tl.srt = 45, bg = "honeydew4")
```

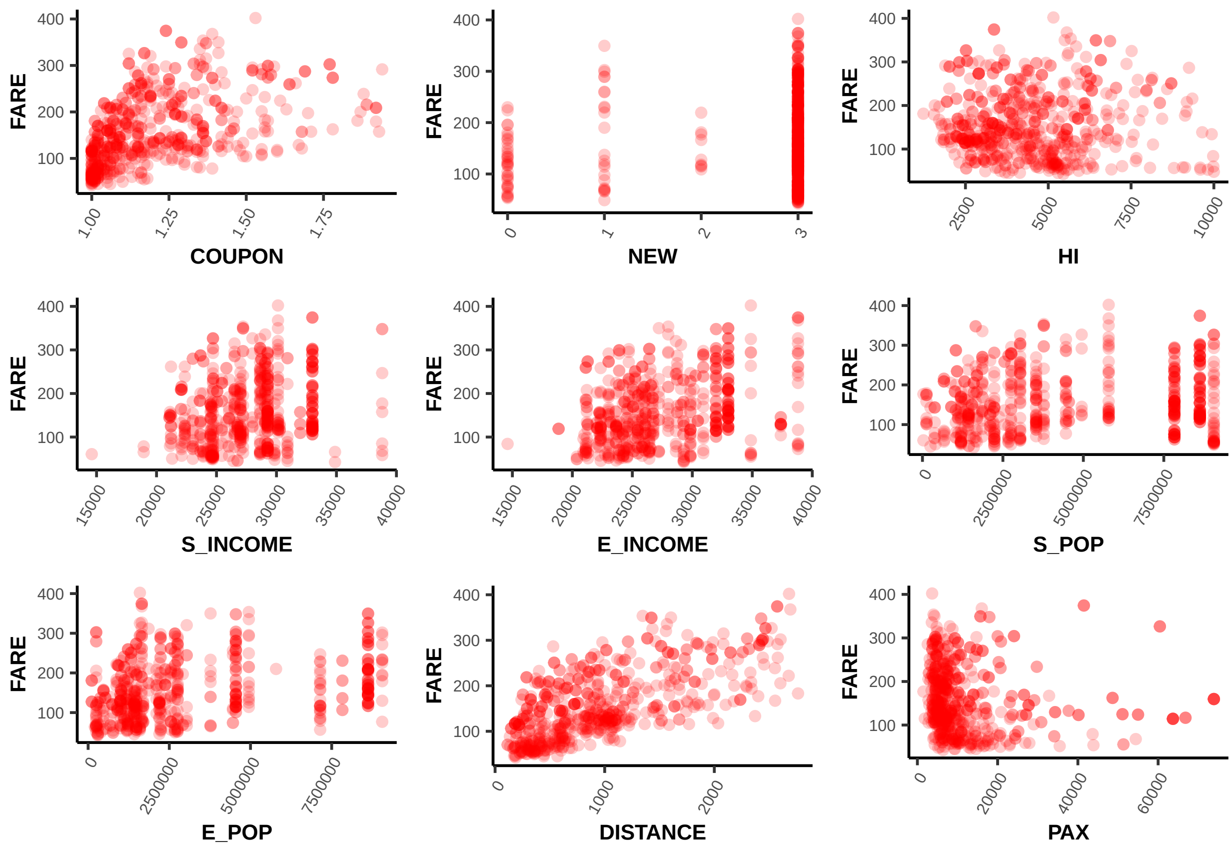



```r
# Scatter plots between FARE and other predictors
x <- ggplot(airfare.df) + theme(axis.text.x = element_text(angle = 60, hjust = 1),
                                axis.text=element_text(size=6),
                                axis.title=element_text(size=8,face="bold"))

coupon.plot <- x + geom_point(color = "red", aes(COUPON, FARE), alpha = 0.2)
new.plot <- x + geom_point(color = "red", aes(NEW, FARE), alpha = 0.2)
hi.plot <- x + geom_point(color = "red", aes(HI, FARE), alpha = 0.2)
s_income.plot <-x + geom_point(color = "red", aes(S_INCOME, FARE), alpha = 0.2)
e_income.plot <- x + geom_point(color = "red", aes(E_INCOME, FARE), alpha = 0.2)
```

```r
s_pop.plot <- x + geom_point(color = "red", aes(S_POP, FARE), alpha = 0.2)
e_pop.plot <- x + geom_point(color = "red", aes(E_POP, FARE), alpha = 0.2)
distance.plot <- x + geom_point(color = "red", aes(DISTANCE, FARE), alpha = 0.2)
pax.plot <- x + geom_point(color = "red", aes(PAX, FARE), alpha = 0.2)

grid.arrange(coupon.plot, new.plot, hi.plot, s_income.plot, e_income.plot, s_pop.plot,
             e_pop.plot, distance.plot, pax.plot, nrow = 3)
```



Answer 1: From the above correlation table and correlation plot, we can find that the best single predictor of FARE is DISTANCE because their correlation coefficient is 0.67 which is highest absolute value as compared to other predictors.

From the plot we find that FARE and DISTANCE have strong positive correlation and are linearly coorelated which means With the increase in the distance between the two endpoint airports the average fare along that route increases.

Question 2: Explore categorical predictors and create pivot table with average fare in each category

```r
vacation<-factor(airfare.df$VACATION)
vacation_table<-table(vacation)
round(prop.table(vacation_table),digits=2)
```

```
## vacation
```

```
##   No  Yes
## 0.73 0.27
```

```r
prop_vac<-round(100*prop.table(vacation_table),digits=0)

sw<-factor(airfare.df$SW)
sw_table<-table(sw)
round(prop.table(sw_table),digits=2)
```

```
## sw
##  No Yes
## 0.7 0.3
```

```r
prop_sw<-round(100*prop.table(sw_table),digits=0)

slot<-factor(airfare.df$SLOT)
slot_table<-table(slot)
round(prop.table(slot_table),digits=2)
```

```
## slot
## Controlled       Free
##       0.29       0.71
```

```r
prop_slot<-round(100*prop.table(slot_table),digits=0)

gate<-factor(airfare.df$GATE)
gate_table<-table(gate)
round(prop.table(gate_table),digits=2)
```

```
## gate
## Constrained        Free
##        0.19        0.81
```

```r
prop_gate<-round(100*prop.table(gate_table),digits=0)

data.frame(prop_vac,prop_sw,prop_slot,prop_gate)
```

```
##   vacation Freq  sw Freq.1       slot Freq.2        gate Freq.3
## 1       No   73  No     70 Controlled     29 Constrained     19
## 2      Yes   27 Yes     30       Free     71        Free     81
```

```r
print("Percentage of flights in each category")
```

```
## [1] "Percentage of flights in each category"
```

```r
airfares_melt <- melt(airfare.df, id = c(3,4,10,11), measure.vars = "FARE")
airfares_castvac <- dcast(airfares_melt, VACATION~ variable, mean)
airfares_castsw <- dcast(airfares_melt, SW~ variable, mean)
airfares_castslot <- dcast(airfares_melt, SLOT~ variable, mean)
airfares_castgate <- dcast(airfares_melt, GATE~ variable, mean)
airfares_cast.df <- data.frame(airfares_castvac, airfares_castsw,
                               airfares_castslot , airfares_castgate)
print("Average fare in each category")
```

```
## [1] "Average fare in each category"
```

```
airfares_cast.df
```

```
##   VACATION     FARE  SW    FARE.1       SLOT   FARE.2        GATE  FARE.3
## 1       No 173.5525  No 188.18279 Controlled 186.0594 Constrained 193.129
## 2      Yes 125.9809 Yes  98.38227       Free 150.8257        Free 153.096
```

Answer 2: From the pivote table of mean FARE of different categorical variable, it is observed that there is a drastic diffrence in fares when SouthWest is serving on routes. While SouthWest is serving the fares are 3/7 times lower than FARE when SouthWest is not serving. We can therefore infere that SW is the best for predicting fares as compared to the effect of other variables.

Question 3: Data partition by assigning 80% to training dataset and 20% to the test dataset.

```
set.seed(42)
sample_size = round(0.80*nrow(airfare.df))
train.index <- sample(nrow(airfare.df), sample_size)
train.df <- airfare.df[train.index,]
valid.df <- airfare.df[-train.index,]
```

Answer 3: While creating a predictive model, we don't use the complete data set to train the model but create a training set which is 80% of the data set in this case. On the other hand, the rest 20% of the data is called validation set which is used in evaluating the performance of the model.

Question 4: Running stepwise regression to reduce the number of predictors.

```
set.seed(42)

# Running stepwise regression to reduce the number of predictors
af.stepwise <- regsubsets(FARE~ ., data = train.df, nbest = 1,
                                   nvmax = dim(train)[2], method = "seqrep")
af.stepwise
```

```
## Subset selection object
## Call: regsubsets.formula(FARE ~ ., data = train.df, nbest = 1, nvmax = dim(train)[2],
##     method = "seqrep")
## 13 Variables  (and intercept)
##             Forced in Forced out
## COUPON          FALSE      FALSE
## NEW             FALSE      FALSE
## VACATIONYes     FALSE      FALSE
## SWYes           FALSE      FALSE
## HI              FALSE      FALSE
## S_INCOME        FALSE      FALSE
## E_INCOME        FALSE      FALSE
## S_POP           FALSE      FALSE
## E_POP           FALSE      FALSE
## SLOTFree        FALSE      FALSE
```

```
## GATEFree        FALSE      FALSE
## DISTANCE        FALSE      FALSE
## PAX             FALSE      FALSE
## 1 subsets of each size up to 13
## Selection Algorithm: 'sequential replacement'
```

```r
sum <- summary(af.stepwise)

# show models
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     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
# show metrics
sum$rsq  #gives r square for this model
```

```
##  [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
sum$adjr2 # gives adjusted r square of the model
```

```
##  [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
sum$cp #gives the value of Mallow 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
```

```r
step.lm <- lm(FARE ~ VACATION + SW + HI + E_INCOME + S_POP + E_POP + SLOT +
              GATE + DISTANCE + PAX, data = train.df)

step.lm.pred <- predict(step.lm, valid.df)
accuracy(step.lm.pred, valid.df$FARE)
```

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

Answer 4: Interpretation of the model 1. Stepwise regression consists of iteratively adding and removing predictors, in order to find the subset of variables in the data set resulting in the best performing model.

2. It starts with forward selection and also consider dropping the non significant predictors at each step.
3. regsubsets() method from 'leaps' package is used, it has a tuning parameter 'nvmax' specifying maximum number of predictors to incorporate in the model.
4. regsubsets has the option 'method' which takes values 'exhaustive', 'backward', 'forward' and 'seqrep'(combination of bakward and forward selections) for selections. Here we are using seqrep option.
5. r square, adjusted rsquare and Mallow cp are the values of the chosen model statistic for each model.
6. R-square: This value explains the variation of the variable FARE (dependent variable) with the other thirteen variables in the model. The higher the R square, the better the model. We can infer that the value of R square is increasing with the addition of each predictor. Hence, this is not the best statistic to find the model of best fit.
7. Adjusted R-square: On the other hand, the adjusted R square value whose value is dependent upon the number of variables in the model and the value with highest Adjusted R square indicates the best model without including the unneccesary variables. So here the model with 12 variables would be considered the best as its adj r square value is 0.7760708 which is the maximum.
8. Mallow cp: The value of Mallow cp decreases with the increase in the variables in the model. The model with the minimum value of Mallow cp can be considered the best. Here the model 10 has the minimum value(number of variables + 1) therefore we consider the model with variables VACATION, SW, HI, E_INCOME, S_POP, E_POP, SLOT, GATE, DISTANCE, PAX in the final model.
9. Finding: As we are searching for the best model based on the cp and adjusted R-squared of each model, we realized there is an abnormal occurrence. DISTANCE has been consistently chosen as a varible for the best models from 1 to 9 variable. However, in the model with 10 variables, it's suddenly dropped. This result may be caused by the choosing varibles technique of stepwise, which consists of iteratively adding and removing predictors, in order to find the subset of variables in the data set resulting in the best performing model. This technique doesn't apply for backward and forward methods.

Question 5: Using exhaustive search to reduce the number of predictors

```r
#nbest = number of the best subsets of each size to keep in the results
#Period notation regresses Fare against all the other variables

airfare.lm.exhautive <- regsubsets(FARE~ ., data = train.df, nbest = 1,
                                   nvmax = dim(train)[2], method = "exhaustive")
airfare.lm.exhautive
```

```
## Subset selection object
## Call: regsubsets.formula(FARE ~ ., data = train.df, nbest = 1, nvmax = dim(train)[2],
##     method = "exhaustive")
## 13 Variables  (and intercept)
##              Forced in Forced out
## COUPON           FALSE      FALSE
## NEW              FALSE      FALSE
## VACATIONYes      FALSE      FALSE
## SWYes            FALSE      FALSE
## HI               FALSE      FALSE
## S_INCOME         FALSE      FALSE
## E_INCOME         FALSE      FALSE
## S_POP            FALSE      FALSE
## E_POP            FALSE      FALSE
## SLOTFree         FALSE      FALSE
## GATEFree         FALSE      FALSE
## DISTANCE         FALSE      FALSE
## PAX              FALSE      FALSE
## 1 subsets of each size up to 13
## Selection Algorithm: exhaustive
```

```r
sum <- summary(airfare.lm.exhautive)

# show models
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
```

```
# show metrics
sum$rsq  #gives r square for this model
```

```
##  [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
```

```
# show adjusted r sq.
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
```

```
# Show Mallow cp
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
```

Answer 5: 1. The exhaustive search model runs a linear regression model for each combination of variables, giving us predictions for each regression subset. Each regression iteration returns either a TRUE or FALSE value against the set of predictors, indicating their inclusion into the model.

2. r square, adjusted rsquare and Mallow cp are the values of the chosen model statistic for each model.
3. From above we can infer that Intercept is TRUE for every model. The first model will have one predictor true i.e. DISTANCE Then in the second model we have 2 predictors true which are DISTANCE and SW. Similarly, the model 3 has three predictors TRUE which are DISTANCE, SW and VACATION. This is how the most significant variables keeps on adding to the model.
4. To find the best model we have to consider the values of adjusted r square and mallow cp.
5. The model with the maximum adjusted r square will be taken as the best one. The value of adj r square will decrease after that indicating the addition of unneccesary vairables. Here the model with 12 variables can be considered the best.
6. Another statistic for finding the best model is Mallow cp, the value of Mallow cp decrease with the addition of predictors. The model with the minimum cp can be chosen. Here we have chosen the model with 11 predictors as it's cp value is 11.73270, which is the least. Considering the value of cp to choose the best model as it gives the model of good fit with less number of predictors.
7. We reject two variables; COUPON and S_INCOME because they show max FALSE values (not a good fit) while running exhaustive search on the subset variables.

Question 6: Comparing the predictive accuracy

```
ex.lm <- lm(FARE ~ VACATION + SW + NEW + HI + E_INCOME + S_POP + E_POP + SLOT +
              GATE + DISTANCE + PAX, data = train.df)

airfare.lm.exhautive.pred <- predict(ex.lm, valid.df)
airfare.lm.step.predicted <- predict(step.lm, valid.df)

# Finding the accuracy of exhaustive and stepwise regression
print("Accuracy of Exhaustive regression")
```

```
## [1] "Accuracy of Exhaustive regression"
```

```
accuracy(airfare.lm.exhautive.pred, valid.df$FARE)
```

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

```
print("Accuracy of stepwise regression")
```

```
## [1] "Accuracy of stepwise regression"
```

```
accuracy(airfare.lm.step.predicted, valid.df$FARE)
```

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

Answer 6: RMSE is the standard deviation of the residuals (prediction errors). The lower the RMSE (root mean squared error) for a model, the better is its accuracy.

We observe that the model of 11 predictors created using exhaustive search have RMSE value 36.82363 which is smaller as compared to the RMSE value 36.8617 of stepwise regression for 10 predictors. When we take more predictors the RMSE value decreases.

Question 7: Using the exhaustive search model to predict the average fare on a route for the test dataset

```
predict.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)

print("Average Fare on the route when SW decided not to cover the route")
```

```
## [1] "Average Fare on the route when SW decided not to cover the route"
```

```r
estimated.fare <- predict(ex.lm, predict.df)
estimated.fare
```

```
##        1
## 247.2198
```

Question 8: The reduction in average fare on the route if SW decides to serve the route

```r
predict2.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)
estimated.fare.sw <- predict(ex.lm,predict2.df)

print("Average Fare on the route when SW decided to cover the route")
```

```
## [1] "Average Fare on the route when SW decided to cover the route"
```

```r
estimated.fare.sw
```

```
##        1
## 206.6483
```

```r
print("Reduction in average fare if SW decides to cover the route")
```

```
## [1] "Reduction in average fare if SW decides to cover the route"
```

```r
reduction <- estimated.fare - estimated.fare.sw
reduction
```

```
##        1
## 40.57159
```

Answer 8:If southwest covers the same route then the Fare reduces to $207.1558 by 40.57159, instead of the previous 247.684 dollars.

Question 9: Using leaps package, run backward selection regression to reduce the number of predictors.

```r
airfare.back.lm <- regsubsets(FARE~., train.df, nbest =1, nvmax = dim(airfare.df)[2],
                              method = "backward")
sum.back <- summary(airfare.back.lm)
sum.back$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
```

```r
sum.back$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
```

```r
sum.back$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
```

```r
ex.lm.backward <- lm(FARE ~ VACATION + SW + HI + E_INCOME + S_POP + E_POP + SLOT +
              GATE + DISTANCE + PAX, data = train.df)

af.lm.backward.pred <- predict(ex.lm.backward, valid.df)
accuracy(af.lm.backward.pred, valid.df$FARE)
```

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

Answer 9: Backward selection starts with all predictors in the model (full model), iteratively removes the least contributive predictors, and stops when you have a model where all predictors are statistically significant.

When the backward selection regression is performed in the first iteration least significant predictor i.e. COUPON will be removed followed by S_INCOME and NEW. The value of mallow cp till variable 10 is decreasing and from variable 11 the cp value is increasing. So we just include 11 predictors in the model which are: NEW, VACATIONYes, SWYes,HI, E_INCOME, S_POP, E_POP, SLOTFree, GATEFree, DISTANCE, PAX.

Question 10: Backward selection model using stepAIC() function

```r
library(MASS) ### stepAIC is in the mass package


airfare.lm <- lm(FARE ~ ., data = train.df)
airfare.back.AIC <- stepAIC(airfare.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
```

```r
summary(airfare.back.AIC)
```

```
##
## 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
```

Answer 10: 1. AIC function is used to optimize the regression search for the final set of predictors. It takes into account the amount of information loss due to the simplification during regression iterations. AIC also penalizes the model for adding extra variables.

2. Initial AIC Value of the model is 3652.06 when all the predictors are included in the model. The predictor with the lowest AIC value is dropped untill the AIC of the model decreases, when AIC starts increasing the regression is stopped and includes the predictors at that step.
3. In the first step if backward selection regression, AIC =3652.06 and the Predictor with lowest AIC is COUPON = 3650.8
4. In step 2, COUPON is dropped and FARE is regressed against 12 other predictors, AIC value of the model in step 2 is 3650.81 and the predictor with lowest AIC is S_INCOME(3649.8) which is dropped in the next step.
5. In step 3, when S_INCOME is dropped and FARE is regressed against 11 other predictors the AIC value of the model in step 3 is 3649.84 and the predictor with lowest AIC is S_INCOME(3649.8) which is dropped in the next step.
6. In step 3, the lowest AIC value is for NEW and which is dropped in step 4, the AIC value of the model here is 3649.22. We notce that there is drop in AIC of the model. At this point the regression is stopped and model includes all the predictors contributed at this step of regreession.(VACATION,SW,HI,E_INCOME,S_POP,E_POP,SLOT,GATE,DISTANCE, PAX)
7. The Multiple R-squared IS 0.7803, the model explains 78.03% of variablility and is 78.03% efficient.
8. The p-value for the variables indicates whether the predictor is meaningful or not for the model. The p-value of DISTANCE, VACATION or SW are significantly small hence they are the excellent addition to the model. On the other hand, p-value for SLOT and Ending average personal income is large hence the slot and ending income has no significant effect on the Fare.
9. The value of Estimate for SW is -40.52. The model predicts that the value of FARE decreases by 40.52 if the Southwest Airline serves the route.
10. Similarly, if it's a vacation route then the average fare along the route decreases by 38.7.
11. It is important to note that once the model is found, stepAIC doesn't take into account the p value for significance levels.