BUAN6356_Homework2_Group11

Jeffrey Chang,Naishal Kanubhai Thakkar,Bhavana Chowdary Kandimalla,Navya Bulusu,Rucha Nichkawade

10/1/2019

```r
if(!require("pacman")) install.packages("pacman")
pacman::p_load(data.table, leaps, MASS, GGally, caTools, forecast, plyr, dplyr, scales)
```

```r
# load file.cvs
airfares.df<-read.csv("Airfares.csv")
airfares.df<-airfares.df[,-c(1:4)]
```

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.

```r
#Run correlation table
airfares.cor.df<-airfares.df
#set up the categorical variable
airfares.cor.df$VACATION <- revalue(airfares.cor.df$VACATION, c("Yes"=1))
airfares.cor.df$VACATION <- revalue(airfares.cor.df$VACATION, c("No"=0))
airfares.cor.df$SW <- revalue(airfares.cor.df$SW , c("Yes"=1))
airfares.cor.df$SW  <- revalue(airfares.cor.df$SW , c("No"=0))
airfares.cor.df$SLOT <- revalue(airfares.cor.df$SLOT , c("Controlled"=1))
airfares.cor.df$SLOT  <- revalue(airfares.cor.df$SLOT , c("Free"=0))
airfares.cor.df$GATE <- revalue(airfares.cor.df$GATE , c("Constrained"=1))
airfares.cor.df$GATE  <- revalue(airfares.cor.df$GATE , c("Free"=0))
#set Factor into Num
airfares.cor.df$VACATION = as.numeric(airfares.cor.df$VACATION)
airfares.cor.df$SW = as.numeric(airfares.cor.df$SW)
airfares.cor.df$SLOT = as.numeric(airfares.cor.df$SLOT)
airfares.cor.df$GATE = as.numeric(airfares.cor.df$GATE)
cor.FARE<-round(cor(airfares.cor.df$FARE, airfares.cor.df),2)
cor.FARE
```

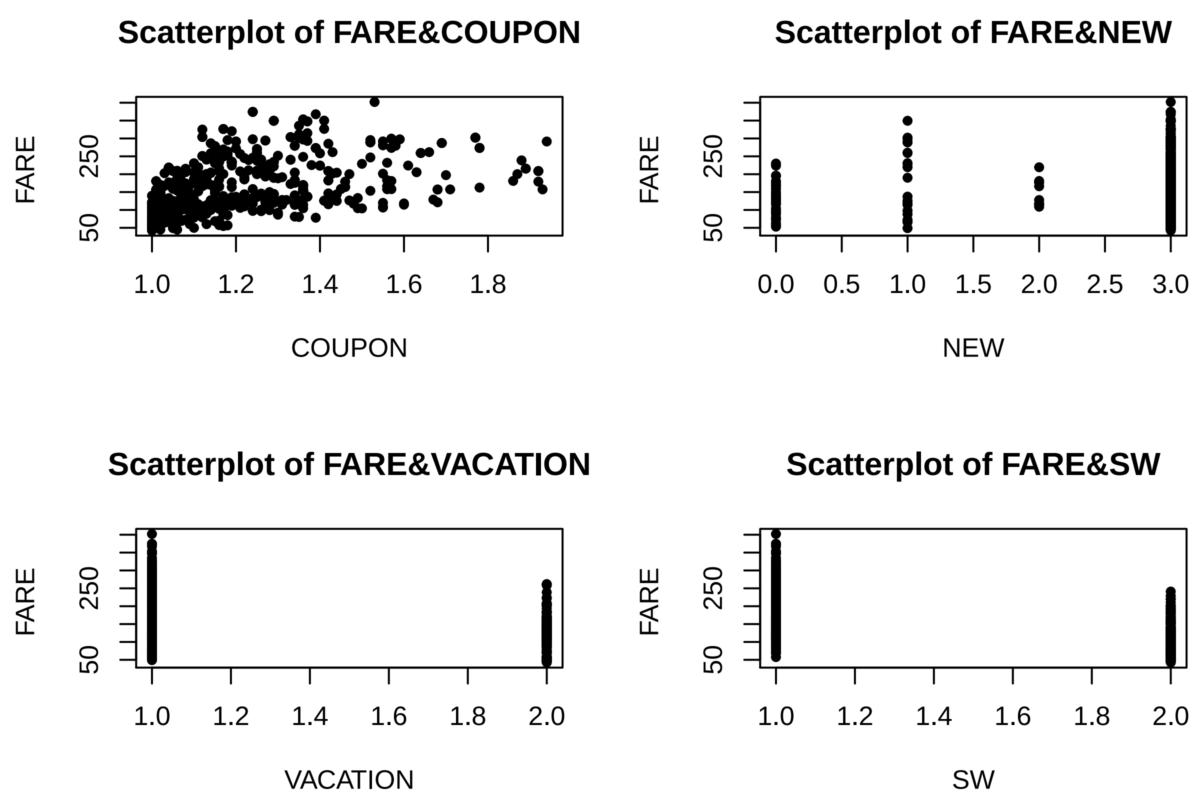
```
##      COUPON  NEW VACATION    SW   HI S_INCOME E_INCOME S_POP E_POP  SLOT
## [1,]    0.5 0.09    -0.28 -0.54 0.03     0.21     0.33  0.15  0.29 -0.21
##        GATE DISTANCE   PAX FARE
## [1,] -0.21     0.67 -0.09    1
```

As per the output of the correlation matrix, the correlation co - efficient is high for Fare and distance hence we can say that, the best single predictor for FARE is distance.

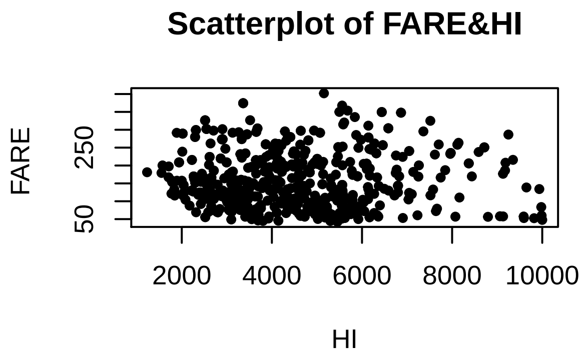
```r
cat("Scatter Plots between FARE and the Predictors")
```

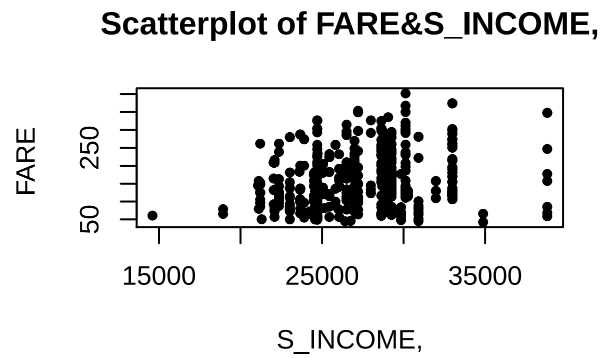
```
## Scatter Plots between FARE and the Predictors
```

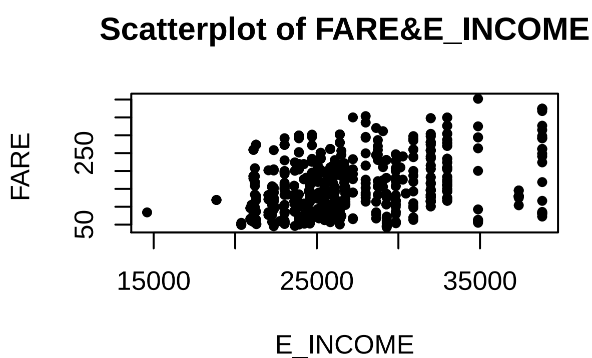
```r
par(mfrow = c(2,2))

plot(airfares.df$COUPON, airfares.df$FARE, main="Scatterplot of FARE&COUPON",
   ylab="FARE ", xlab="COUPON", pch=20)
plot(airfares.df$NEW, airfares.df$FARE, main="Scatterplot of FARE&NEW",
   ylab="FARE ", xlab="NEW", pch=20)
plot(as.numeric(airfares.df$VACATION), airfares.df$FARE, main="Scatterplot of FARE&VACATION",
   ylab="FARE ", xlab="VACATION", pch=20)
plot(as.numeric(airfares.df$SW), airfares.df$FARE, main="Scatterplot of FARE&SW",
   ylab="FARE ", xlab="SW", pch=20)
```

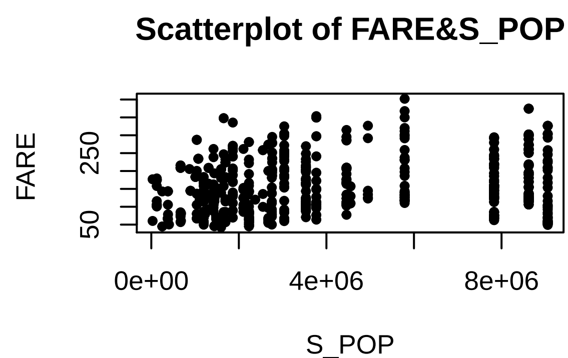



```r
plot(airfares.df$HI, airfares.df$FARE, main="Scatterplot of FARE&HI",
   ylab="FARE ", xlab="HI", pch=20)
plot(airfares.df$S_INCOME, airfares.df$FARE, main="Scatterplot of FARE&S_INCOME,",
   ylab="FARE ", xlab="S_INCOME,", pch=20)
plot(airfares.df$E_INCOME, airfares.df$FARE, main="Scatterplot of FARE&E_INCOME",
   ylab="FARE ", xlab="E_INCOME", pch=20)
plot(airfares.df$S_POP, airfares.df$FARE, main="Scatterplot of FARE&S_POP",
   ylab="FARE ", xlab="S_POP", pch=20)
```

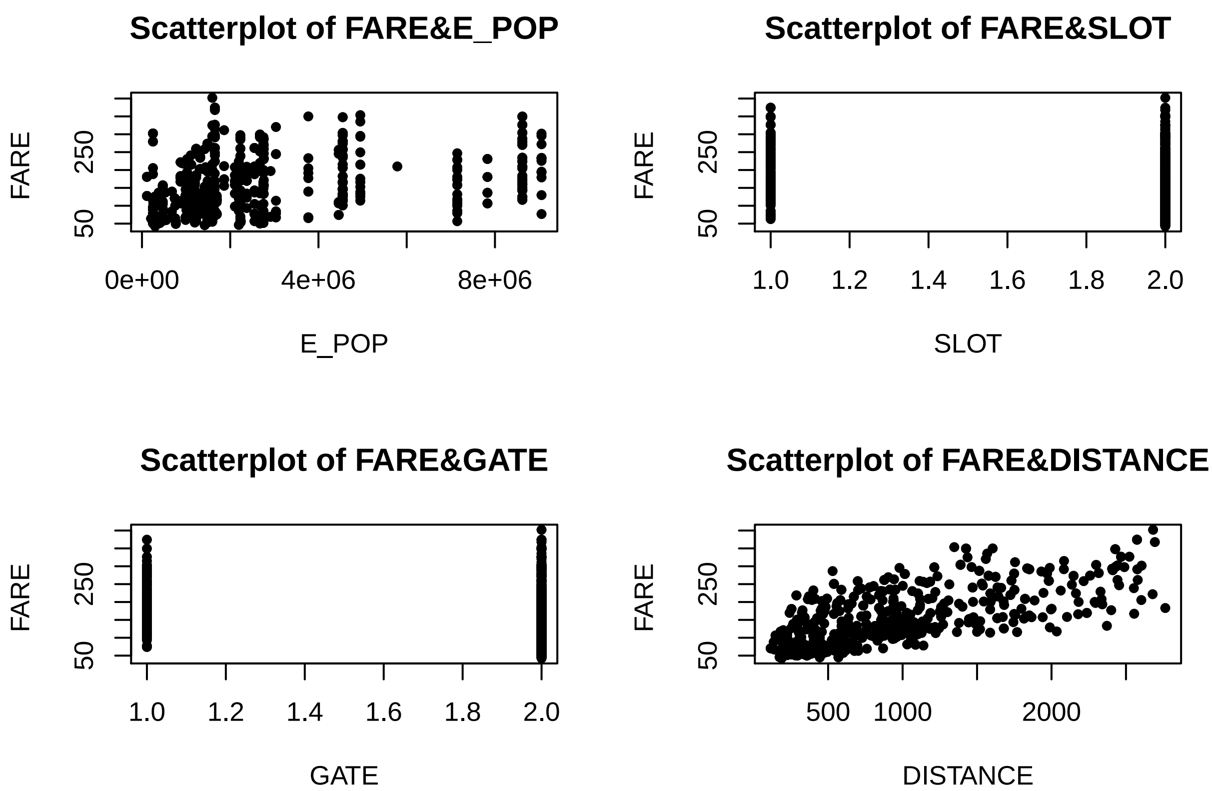




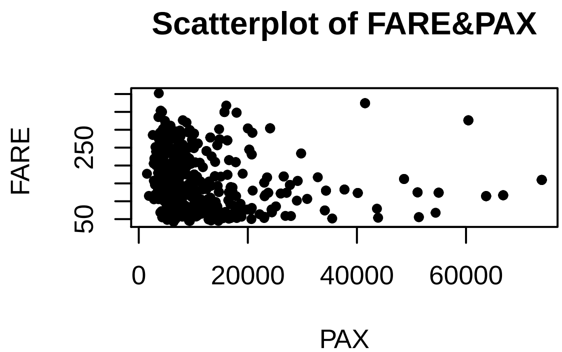





```r
plot(airfares.df$E_POP, airfares.df$FARE, main="Scatterplot of FARE&E_POP",
   ylab="FARE ", xlab="E_POP", pch=20)
plot(as.numeric(airfares.df$SLOT), airfares.df$FARE, main="Scatterplot of FARE&SLOT",
   ylab="FARE ", xlab="SLOT", pch=20)
plot(as.numeric(airfares.df$GATE), airfares.df$FARE, main="Scatterplot of FARE&GATE",
   ylab="FARE ", xlab="GATE", pch=20)
plot(airfares.df$DISTANCE, airfares.df$FARE, main="Scatterplot of FARE&DISTANCE",
   ylab="FARE ", xlab="DISTANCE", pch=20)
```




```r
plot(airfares.df$PAX, airfares.df$FARE, main="Scatterplot of FARE&PAX",
   ylab="FARE ", xlab="PAX", pch=20)
```



The above represents the scatterplot for FARE vs all the variables. When we plot a scatterplot of FARE vs Categorical Variable(VACATION,SW,SLOT,GATE) there is no trend observed. when we plot a scatterplot of FARE vs numerical variables(COUPON,NEW,HI,S_INCOME,E_INCOME,S_POP,E_POP,DISTANCE,PAX) maximum trend/co-linearity is observed between FARE and distance which again proves that distance is the single best 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.

```
airfares<-airfares.df
mean(airfares$FARE)
```

```
## [1] 160.8767
```

```
#VACATION
v1<-airfares%>%group_by(VACATION)%>%summarize(Mean_fare=mean(FARE))
v2<-airfares%>%group_by(VACATION)%>%count(VACATION)%>%summarize(Percentage_flights=n*100/nrow(airfares))
cbind(v1,v2[,-1])
```

```
##   VACATION Mean_fare Percentage_flights
## 1       No  173.5525           73.35423
## 2      Yes  125.9809           26.64577
```

```
#SW
s1<-airfares%>%group_by(SW)%>%summarize(Mean_fare=mean(FARE))
s2<-airfares%>%group_by(SW)%>%count(SW)%>%summarize(Percentage_flights=n*100/nrow(airfares))
cbind(s1,s2[,-1])
```

```
##    SW Mean_fare Percentage_flights
## 1  No 188.18279           69.59248
## 2 Yes  98.38227           30.40752
```

```
#SLOT
sl1<-airfares%>%group_by(SLOT)%>%summarize(Mean_fare=mean(FARE))
sl2<-airfares%>%group_by(SLOT)%>%count(SLOT)%>%summarize(Percentage_flights=n*100/nrow(airfares))
cbind(sl1,sl2[,-1])
```

```
##         SLOT Mean_fare Percentage_flights
## 1 Controlled  186.0594           28.52665
## 2       Free  150.8257           71.47335
```

```
#GATE
g1<-airfares%>%group_by(GATE)%>%summarize(Mean_fare=mean(FARE))
g2<-airfares%>%group_by(GATE)%>%count(GATE)%>%summarize(Percentage_flights=n*100/nrow(airfares))
cbind(g1,g2[,-1])
```

```
##          GATE Mean_fare Percentage_flights
## 1 Constrained   193.129           19.43574
## 2        Free   153.096           80.56426
```

```
aov_SW <- aov(FARE ~ SW, data = airfares.df)
summary(aov_SW)
```

```
##               Df  Sum Sq Mean Sq F value Pr(>F)    
## SW             1 1088734 1088734   267.1 <2e-16 ***
## Residuals    636 2592751    4077                   
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
aov_VACATION <- aov(FARE ~ VACATION, data = airfares.df)
summary(aov_VACATION)
```

```
##               Df  Sum Sq Mean Sq F value   Pr(>F)    
## VACATION       1  282208  282208    52.8 1.09e-12 ***
## Residuals    636 3399276    5345                     
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
aov_SLOT <- aov(FARE ~ SLOT, data = airfares.df)
summary(aov_SLOT)
```

```
##               Df  Sum Sq Mean Sq F value   Pr(>F)    
## SLOT           1  161485  161485   29.18 9.34e-08 ***
## Residuals    636 3520000    5535                     
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
aov_GATE <- aov(FARE ~ GATE, data = airfares.df)
summary(aov_GATE)
```

```
##               Df  Sum Sq Mean Sq F value   Pr(>F)    
## GATE           1  160104  160104   28.92 1.06e-07 ***
## Residuals    636 3521381    5537                     
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

1.The best predictor for FARE is the categorical variable which produces the largest difference in the average FARE between two categories, so the average between two categories is highest for SW hence we can say that SW is the best categorical predictor.

2.From to analysis of variance(ANOVA), the P value of SW is least(2*e^-16) we can say that the relationship between FARE and SW is significant compared to other categorical variables.

3.As the correlation co-efficient of SW and FARE is the highest amongst the categorical variables, we can say that SW seems to be the best predictor of FARE.

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.

```r
set.seed(42)
sample = sample.split(airfares.df$FARE, SplitRatio = .80)
train.df= subset(airfares.df, sample == TRUE)
valid.df= subset(airfares.df, sample == FALSE)
head(train.df)
```

```
##   COUPON NEW VACATION  SW      HI S_INCOME E_INCOME   S_POP   E_POP SLOT
## 3   1.06   3       No  No 9185.28    30124    29838 5787293 7145897 Free
## 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
## 7   1.28   3       No  No 6754.48    28637    29838 3036732 7145897 Free
## 8   1.15   3      Yes Yes 5584.00    26752    29838 1440377 7145897 Free
## 9   1.33   3       No Yes 4662.44    27211    29838 3770125 7145897 Free
##   GATE DISTANCE   PAX   FARE
## 3 Free      364  6452 207.76
## 5 Free      612 25144  85.47
## 6 Free      309 13386  56.76
## 7 Free     1220  4625 228.00
## 8 Free      921  5512 116.54
## 9 Free     1249  7811 172.63
```

Question 4:

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

```
stepwise.regression = regsubsets(FARE ~ ., data = train.df, nvmax = 14, method = "seqrep")
reg.summary = summary(stepwise.regression)

print("summary$which")
```

```
## [1] "summary$which"
```

```
reg.summary$which
```

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

```
print("summary$rsq")
```

```
## [1] "summary$rsq"
```

```
reg.summary$rsq
```

```
##  [1] 0.4454418 0.6044197 0.7058560 0.7350201 0.7463320 0.7602025 0.7638719
##  [8] 0.7681274 0.7745708 0.7799948 0.6330714 0.7839333 0.7839352
```

```r
print("summary$adjr2")
```

```
## [1] "summary$adjr2"
```

```r
reg.summary$adjr2
```

```
##  [1] 0.4443501 0.6028592 0.7041121 0.7329213 0.7438154 0.7573421 0.7605793
##  [8] 0.7644249 0.7705130 0.7755859 0.6249665 0.7787164 0.7782722
```

```r
print("summary$Cp")
```

```
## [1] "summary$Cp"
```

```r
reg.summary$cp
```

```
##  [1] 767.04804 404.09709 173.23918 108.28976  84.32224  54.48090  48.05749
##  [8]  40.28841  27.49709  17.04564 356.32413  12.00425  14.00000
```

After running the stepwise regression, we observed that the ajusted r2 is high for model 12 and the corresponding cp value is low which are the main factors for determining the best possible model after regression. Hence we choose model which has 12 explainatory variables i.e NEW,VACATION,SWYES,HI,S_INCOME,E_INCOME,S_POP,E_ and PAX as the best possible model, and eliminate one of the predictors i.e COUPON.

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 the predictors included in the final model.

```
exhaustive.search<-regsubsets(FARE ~ ., data=train.df, nbest = 1, nvmax =14 , method = "exhaustive")
sum<-summary(exhaustive.search)

#show model
print("sum$which")
```

```
## [1] "sum$which"
```

```
sum$which
```

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

```
#show metrics
print("sum$rsq")
```

```
## [1] "sum$rsq"
```

```
sum$rsq
```

```
##  [1] 0.4454418 0.6044197 0.7058560 0.7350201 0.7463320 0.7602025 0.7638719
##  [8] 0.7681274 0.7745708 0.7799948 0.7836397 0.7839333 0.7839352
```

```r
print("sum$adjr2")
```

```
## [1] "sum$adjr2"
```

```r
sum$adjr2
```

```
##  [1] 0.4443501 0.6028592 0.7041121 0.7329213 0.7438154 0.7573421 0.7605793
##  [8] 0.7644249 0.7705130 0.7755859 0.7788606 0.7787164 0.7782722
```

```r
print("sum$Cp")
```

```
## [1] "sum$Cp"
```

```r
sum$cp
```

```
##  [1] 767.04804 404.09709 173.23918 108.28976  84.32224  54.48090  48.05749
##  [8]  40.28841  27.49709  17.04564  10.67839  12.00425  14.00000
```

After running the regression model, we observed that the ajusted r2 is high for model 11 and the corresponding cp value is low which are the main factors for determining the best posssible model after regression. Hence we choose model which has 11 explainatory variables i.e VACATION,SWYES,HI,S_INCOME,E_INCOME,S_POP,E_POP,SLOTFREE,GATEFREE,DISTANCE and PAX as the best possible model, and eliminate two of the predictors i.e COUPON and NEW.

Question 6:

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

```r
print("stepwise regression")
```

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

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

```
##                  ME     RMSE      MAE       MPE     MAPE
## Test set -5.117483 32.52078 25.54853 -8.124715 20.19226
```

```r
print("exhaustive search")
```

```
## [1] "exhaustive search"
```

```r
airfares.es.lm<-lm(formula = FARE ~ VACATION+SW+HI+S_INCOME+E_INCOME+S_POP+E_POP+SLOT+GATE+DISTANCE+PAX
airfares.exhaustive.search.pred <- predict(airfares.es.lm, valid.df)
accuracy(airfares.exhaustive.search.pred, valid.df$FARE)
```

```
##                  ME     RMSE      MAE       MPE     MAPE
## Test set -5.078597 32.65557 25.73471 -8.103627 20.38168
```

The RMSE values for step and exhaustive are 32.52078 and 32.65557 respectively.The RMSE value is stepwise regression is comparitively less than the RMSE value of exhaustive regression.

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.

```
#No Coupon and New

new_1.df <- data.frame(COUPON = 1.202, NEW = 3,VACATION = "No", SW="No", HI= 4442.141, S_INCOME = 28760

model_1<-lm(FARE ~ VACATION + SW + HI + S_INCOME + E_INCOME + S_POP + E_POP + SLOT + GATE + PAX + DISTAN
            data = train.df)

predict(model_1, newdata = new_1.df)
```

```
##        1
## 252.9115
```

From the Exhaustive search, we eliminate the predictors COUPON and NEW we do not include those predictors while running the model.So the average fare on this route is 252.9115.

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].

```
new_2.df <- data.frame(COUPON = 1.202, NEW = 3,VACATION = "No", SW="Yes", HI= 4442.141, S_INCOME = 2876

model_2<-lm(FARE ~ VACATION + SW + HI + S_INCOME + E_INCOME + S_POP + E_POP + SLOT + GATE + PAX + DISTAN
            data = train.df)

predict(model_2, newdata = new_2.df)
```

```
##        1
## 213.4966
```

As the southwest airlines decides to cover this route we change the SW predictor to 'YES'.The average fare on this route is 213.4966. So the redution after the southwest airlines decides to cover this route will be 252.9115 - 213.4966 = 39.414

Question 9:

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

```
stepwise.regression.b = regsubsets(FARE ~ ., data = train.df, nvmax = 14, method = "backward")
summary.b = summary(stepwise.regression.b)

print("summary$which")
```

```
## [1] "summary$which"
```

```
summary.b$which
```

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

```
print("summary$rsq")
```

```
## [1] "summary$rsq"
```

```
summary.b$rsq
```

```
##  [1] 0.4454418 0.6044197 0.7058560 0.7350201 0.7404052 0.7509459 0.7601661
##  [8] 0.7670480 0.7745708 0.7799948 0.7836397 0.7839333 0.7839352
```

```r
print("summary$adjr2")
```

```
## [1] "summary$adjr2"
```

```r
summary.b$adjr2
```

```
##  [1] 0.4443501 0.6028592 0.7041121 0.7329213 0.7378299 0.7479751 0.7568218
##  [8] 0.7633282 0.7705130 0.7755859 0.7788606 0.7787164 0.7782722
```

```r
print("summary$Cp")
```

```
## [1] "summary$Cp"
```

```r
summary.b$cp
```

```
##  [1] 767.04804 404.09709 173.23918 108.28976  97.92768  75.73047  56.56455
##  [8]  42.76639  27.49709  17.04564  10.67839  12.00425  14.00000
```

After running the regression model, we observed that the ajusted r2 is high for model 11 and the corresponding cp value is low which are the main factors for determining the best posssible model after regression. Hence we choose model which has 11 explainatory variables i.e VACATION,SWYES,HI,S_INCOME,E_INCOME,S_POP,E_POP,SLOTFREE,GATEFREE,DISTANCE and PAX as the best possible model, and eliminate two of the predictors i.e COUPON and NEW.

Question 10:

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

```r
library(MASS)
airfares_stepAIC<- lm(FARE ~ ., data = train.df)
airfares_stepAIC_results <- stepAIC(airfares_stepAIC, direction = 'backward')
```

```
## Start:  AIC=3679.29
## 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         6  655928 3677.3
## - NEW       1       897  656819 3678.0
## <none>                   655922 3679.3
## - SLOT      1     11248  667170 3686.0
## - S_INCOME  1     11881  667803 3686.4
## - E_INCOME  1     22277  678199 3694.3
## - S_POP     1     23478  679400 3695.2
## - E_POP     1     26869  682791 3697.8
## - GATE      1     29811  685733 3700.0
## - PAX       1     37732  693654 3705.8
## - HI        1     67715  723637 3727.4
## - VACATION  1    103256  759178 3751.8
## - SW        1    111897  767819 3757.6
## - DISTANCE  1    432800 1088722 3935.7
##
## Step:  AIC=3677.29
## FARE ~ NEW + VACATION + SW + HI + S_INCOME + E_INCOME + S_POP +
##     E_POP + SLOT + GATE + DISTANCE + PAX
##
##            Df Sum of Sq     RSS    AIC
## - NEW       1       892  656819 3676.0
## <none>                   655928 3677.3
## - SLOT      1     11353  667280 3684.0
## - S_INCOME  1     12050  667977 3684.6
## - E_INCOME  1     22521  678449 3692.5
## - S_POP     1     23772  679700 3693.4
## - E_POP     1     26875  682803 3695.8
## - GATE      1     29812  685740 3698.0
## - PAX       1     43080  699007 3707.7
## - HI        1     74827  730755 3730.4
## - VACATION  1    103936  759863 3750.3
## - SW        1    112476  768404 3756.0
## - DISTANCE  1    856858 1512786 4101.5
##
## Step:  AIC=3675.98
## FARE ~ VACATION + SW + HI + S_INCOME + E_INCOME + S_POP + E_POP +
##     SLOT + GATE + DISTANCE + PAX
##
##            Df Sum of Sq     RSS    AIC
## <none>                   656819 3676.0
```

```
## - SLOT      1     11065  667884 3682.5
## - S_INCOME  1     11775  668594 3683.0
## - E_INCOME  1     22112  678931 3690.9
## - S_POP     1     23883  680702 3692.2
## - E_POP     1     26682  683501 3694.3
## - GATE      1     30262  687081 3697.0
## - PAX       1     42905  699724 3706.3
## - HI        1     74119  730938 3728.5
## - VACATION  1    103465  760284 3748.6
## - SW        1    112121  768941 3754.4
## - DISTANCE  1    856931 1513750 4099.8
```

In the backward Regression, in the first step we consider all the predictors so the AIC value is 3679.29. We can see from the model that on removing the COUPON predictor the AIC will decrease to 3677.3. Further on removing the NEW predictor the AIC will decrease to 3676.0. Now we can see that AIC is already the least amoung all the predictors so on removing any other variable the AIC would not decrease. Hence we get the model which comprises of 11 predictors.