Assignment 5

Roderick Fan

2022-10-06

```
#Cleaning Environment
rm(list = ls(all=TRUE))
```

```
#Import packages
library(tidyverse)
```

```
## ── Attaching packages ──────────────────────────────────────────
─────── tidyverse 1.3.2 ──
## ✓ ggplot2 3.3.6     ✓ purrr   0.3.4
## ✓ tibble  3.1.8     ✓ dplyr   1.0.9
## ✓ tidyr   1.2.0     ✓ stringr 1.4.1
## ✓ readr   2.1.2     ✓ forcats 0.5.2
## ── Conflicts ──────────────────────────────────────────────────
───── tidyverse_conflicts() ──
## ✕ dplyr::filter() masks stats::filter()
## ✕ dplyr::lag()    masks stats::lag()
```

```
library(here)
```

```
## here() starts at D:/A_Lehigh/2022 Fall/BUAN 488 - Predictive Analytics/HW/HW5
```

```
library(gplots)
```

```
##
## 载入程辑包：'gplots'
##
## The following object is masked from 'package:stats':
##
##     lowess
```

```
library(forecast)
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```
library(leaps)
```

```
#Loading data
air <- read.csv(here("Airfares.csv"))
air2 <- air[,-c(1:4)]

#Turning into Dummy Variables
air2$VACATION <- ifelse(air2$VACATION=="Yes",1,0)
air2$SW <- ifelse(air2$SW=="Yes",1,0)
air2$SLOT <- ifelse(air2$SLOT=="Free",1,0)
air2$GATE <- ifelse(air2$GATE=="Free",1,0)
```

```
set.seed(12)
index <- sample(c(1:length(air2$FARE)), length(air2$FARE)*.6)

training <- air2[index,]
validation <- air2[-index,]
```

```
air.lm <- lm(FARE~., data = training)
summary(air.lm)
```

```
##
## Call:
## lm(formula = FARE ~ ., data = training)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -97.966 -21.928  -1.478  22.021 104.277
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.769e+01  3.704e+01  -0.747 0.455243
## COUPON       1.312e+01  1.543e+01   0.850 0.395754
## NEW         -1.809e+00  2.535e+00  -0.714 0.475826
## VACATION    -2.868e+01  4.774e+00  -6.007 4.55e-09 ***
## SW          -3.524e+01  5.093e+00  -6.919 2.03e-11 ***
## HI           6.781e-03  1.305e-03   5.198 3.35e-07 ***
## S_INCOME     1.808e-03  6.610e-04   2.736 0.006528 **
## E_INCOME     1.847e-03  5.212e-04   3.543 0.000446 ***
## S_POP        3.514e-06  8.526e-07   4.121 4.66e-05 ***
## E_POP        5.247e-06  1.033e-06   5.080 6.01e-07 ***
## SLOT        -1.300e+01  5.155e+00  -2.521 0.012115 *
## GATE        -1.908e+01  5.378e+00  -3.547 0.000440 ***
## DISTANCE     7.138e-02  4.795e-03  14.888  < 2e-16 ***
## PAX         -1.004e-03  2.018e-04  -4.974 1.01e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 35.52 on 368 degrees of freedom
## Multiple R-squared:  0.7889, Adjusted R-squared:  0.7814
## F-statistic: 105.8 on 13 and 368 DF,  p-value: < 2.2e-16
```

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

```
##                  ME    RMSE      MAE       MPE     MAPE
## Test set -0.4781651 36.3315 28.79205 -5.929593 20.58288
```

```
#step wise regression
stepwise <- lm(FARE~.,data=training)
step_b <- step(stepwise, direction = "both")
```

```
## Start:  AIC=2741.2
## FARE ~ COUPON + NEW + VACATION + SW + HI + S_INCOME + E_INCOME +
##     S_POP + E_POP + SLOT + GATE + DISTANCE + PAX
##
##            Df Sum of Sq    RSS    AIC
## - NEW       1       643 464822 2739.7
## - COUPON    1       912 465091 2739.9
## <none>                  464179 2741.2
## - SLOT      1      8018 472197 2745.7
## - S_INCOME  1      9439 473618 2746.9
## - E_INCOME  1     15837 480016 2752.0
## - GATE      1     15871 480050 2752.0
## - S_POP     1     21424 485603 2756.4
## - PAX       1     31206 495385 2764.1
## - E_POP     1     32552 496731 2765.1
## - HI        1     34078 498257 2766.3
## - VACATION  1     45510 509689 2774.9
## - SW        1     60378 524557 2785.9
## - DISTANCE  1    279594 743773 2919.3
##
## Step:  AIC=2739.72
## FARE ~ COUPON + VACATION + SW + HI + S_INCOME + E_INCOME + S_POP +
##     E_POP + SLOT + GATE + DISTANCE + PAX
##
##            Df Sum of Sq    RSS    AIC
## - COUPON    1      1187 466008 2738.7
## <none>                  464822 2739.7
## + NEW       1       643 464179 2741.2
## - SLOT      1      7715 472536 2744.0
## - S_INCOME  1      9651 474473 2745.6
## - GATE      1     15766 480587 2750.5
## - E_INCOME  1     15885 480707 2750.6
## - S_POP     1     22070 486892 2755.4
## - PAX       1     30749 495571 2762.2
## - E_POP     1     32426 497248 2763.5
## - HI        1     33970 498791 2764.7
## - VACATION  1     45124 509946 2773.1
## - SW        1     59735 524557 2783.9
## - DISTANCE  1    281358 746180 2918.5
##
## Step:  AIC=2738.7
## FARE ~ VACATION + SW + HI + S_INCOME + E_INCOME + S_POP + E_POP +
##     SLOT + GATE + DISTANCE + PAX
##
##            Df Sum of Sq     RSS    AIC
## <none>                   466008 2738.7
## + COUPON    1      1187  464822 2739.7
## + NEW       1       918  465091 2739.9
## - SLOT      1      9100  475108 2744.1
## - S_INCOME  1      9207  475215 2744.2
## - E_INCOME  1     15204  481212 2749.0
## - GATE      1     16129  482138 2749.7
## - S_POP     1     21173  487181 2753.7
## - HI        1     33205  499213 2763.0
## - E_POP     1     33227  499235 2763.0
## - PAX       1     40336  506345 2768.4
## - VACATION  1     45623  511632 2772.4
## - SW        1     61252  527260 2783.9
## - DISTANCE  1    594678 1060687 3050.9
```

```
#Step wise Regression with Exhaustive Search
search <- regsubsets(FARE ~ .,
                     data = air2,
                     nbest = 1,
                     nvmax = dim(training)[2],
                     method = "exhaustive")

sum <- summary(search)
sum$which
```

```
##    (Intercept) COUPON   NEW VACATION    SW    HI S_INCOME E_INCOME S_POP E_POP
## 1         TRUE  FALSE FALSE    FALSE FALSE FALSE    FALSE    FALSE FALSE FALSE
## 2         TRUE  FALSE FALSE    FALSE  TRUE FALSE    FALSE    FALSE FALSE FALSE
## 3         TRUE  FALSE FALSE     TRUE  TRUE FALSE    FALSE    FALSE FALSE FALSE
## 4         TRUE  FALSE FALSE     TRUE  TRUE  TRUE    FALSE    FALSE FALSE FALSE
## 5         TRUE  FALSE FALSE     TRUE  TRUE  TRUE    FALSE    FALSE FALSE FALSE
## 6         TRUE  FALSE FALSE     TRUE  TRUE  TRUE    FALSE    FALSE FALSE FALSE
## 7         TRUE  FALSE FALSE     TRUE  TRUE  TRUE    FALSE    FALSE  TRUE  TRUE
## 8         TRUE  FALSE FALSE     TRUE  TRUE  TRUE    FALSE    FALSE  TRUE  TRUE
## 9         TRUE  FALSE FALSE     TRUE  TRUE  TRUE    FALSE    FALSE  TRUE  TRUE
## 10        TRUE  FALSE FALSE     TRUE  TRUE  TRUE    FALSE     TRUE  TRUE  TRUE
## 11        TRUE  FALSE FALSE     TRUE  TRUE  TRUE     TRUE     TRUE  TRUE  TRUE
## 12        TRUE  FALSE  TRUE     TRUE  TRUE  TRUE     TRUE     TRUE  TRUE  TRUE
## 13        TRUE   TRUE  TRUE     TRUE  TRUE  TRUE     TRUE     TRUE  TRUE  TRUE
##     SLOT  GATE DISTANCE   PAX
## 1  FALSE FALSE     TRUE FALSE
## 2  FALSE FALSE     TRUE FALSE
## 3  FALSE FALSE     TRUE FALSE
## 4  FALSE FALSE     TRUE FALSE
## 5   TRUE FALSE     TRUE FALSE
## 6   TRUE  TRUE     TRUE FALSE
## 7  FALSE FALSE     TRUE  TRUE
## 8  FALSE  TRUE     TRUE  TRUE
## 9   TRUE  TRUE     TRUE  TRUE
## 10  TRUE  TRUE     TRUE  TRUE
## 11  TRUE  TRUE     TRUE  TRUE
## 12  TRUE  TRUE     TRUE  TRUE
## 13  TRUE  TRUE     TRUE  TRUE
```

```
rsq <- sum$rsq
adjr2 <- sum$adjr2
cp <- sum$cp
bic <- sum$bic

cbind(rsq,adjr2,cp,bic)
```

```
##              rsq     adjr2        cp       bic
##  [1,] 0.4489214 0.4480550 978.68967 -367.2534
##  [2,] 0.6043580 0.6031119 525.81627 -572.2077
##  [3,] 0.7078166 0.7064340 225.05268 -759.1449
##  [4,] 0.7335202 0.7318363 151.83292 -811.4358
##  [5,] 0.7457416 0.7437300 118.06801 -834.9298
##  [6,] 0.7624809 0.7602224  71.08175 -871.9212
##  [7,] 0.7666590 0.7640663  60.85490 -876.7855
##  [8,] 0.7719799 0.7690798  47.28349 -885.0441
##  [9,] 0.7806697 0.7775265  23.85336 -903.3753
## [10,] 0.7843770 0.7809381  15.00424 -907.7932
## [11,] 0.7861670 0.7824096  11.76598 -906.6533
## [12,] 0.7867381 0.7826435  12.09482 -901.9011
## [13,] 0.7867705 0.7823282  14.00000 -895.5397
```

```
t(t(sum$adjr2))
```

```
##            [,1]
##  [1,] 0.4480550
##  [2,] 0.6031119
##  [3,] 0.7064340
##  [4,] 0.7318363
##  [5,] 0.7437300
##  [6,] 0.7602224
##  [7,] 0.7640663
##  [8,] 0.7690798
##  [9,] 0.7775265
## [10,] 0.7809381
## [11,] 0.7824096
## [12,] 0.7826435
## [13,] 0.7823282
```

```
# top 3 models
models <-  order(sum$adjr2, decreasing = T)[1:3]
models
```

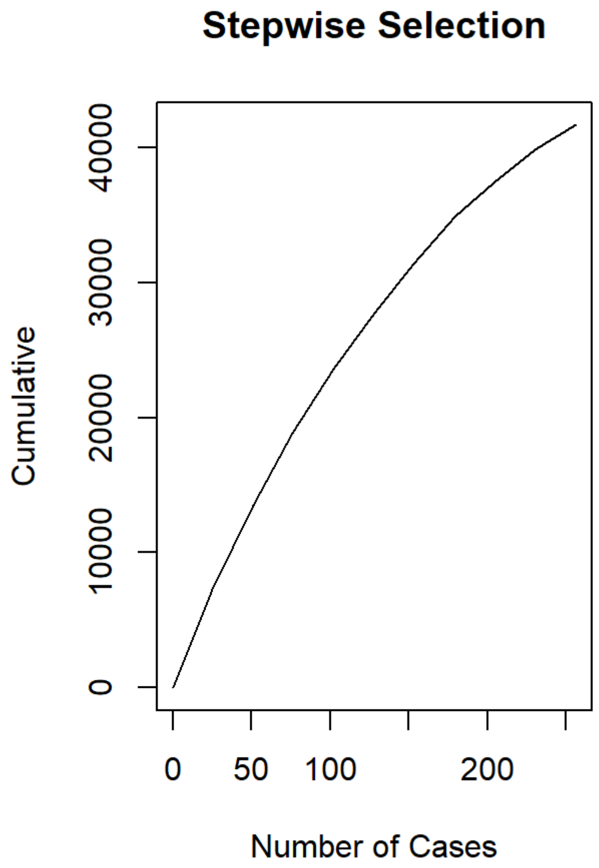
```
## [1] 12 11 13
```

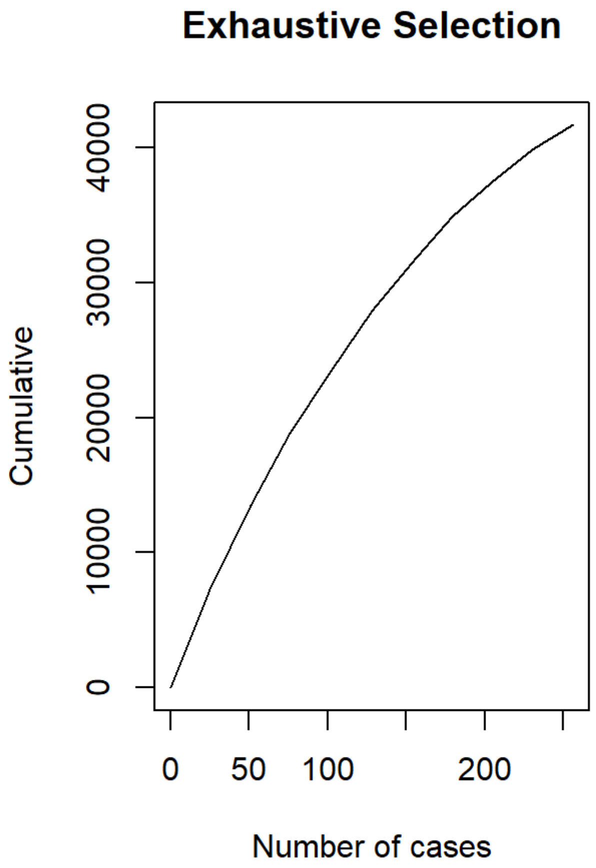
According to step wise (FARE ~ VACATION + SW + HI + S_INCOME + E_INCOME + S_POP + E_POP + SLOT + GATE + DISTANCE + PAX) According to Exhaustive (FARE ~ NEW + VACATION + SW + HI + S_INCOME + E_INCOME + S_POP + E_POP + SLOT + GATE + DISTANCE + PAX)

```
#Compare results given by step wise regression and exhaustive research
air.step <- lm(FARE~ VACATION + SW + HI + S_INCOME + E_INCOME + S_POP + E_POP +
               SLOT + GATE + DISTANCE + PAX, data = training)

pred.step <- predict(air.step, validation)
acc.step <- accuracy(pred.step, validation$FARE)

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

pred.ex <- predict(air.ex, validation)
acc.ex <- accuracy(pred.ex, validation$FARE)


t(cbind(acc.step,acc.ex))
```

```
##        Test set
## ME   -0.8685256
## RMSE 36.3404550
## MAE  28.8566889
## MPE  -6.2311947
## MAPE 20.7747199
## ME   -0.9211285
## RMSE 36.2631126
## MAE  28.7306902
## MPE  -6.2180862
## MAPE 20.6031619
```

```
library(gains)
par(mfcol=c(1,2))
gain1 <- gains(validation$FARE, pred.step)
plot(c(0, gain1$cume.pct.of.total*sum(validation$FARE)) ~ c(0, gain1$cume.obs), xlab="Number of
Cases", ylab="Cumulative", main = "Stepwise Selection", type="l")
gain2 <- gains(validation$FARE, pred.ex)
plot(c(0, gain2$cume.pct.of.total*sum(validation$FARE)) ~ c(0, gain2$cume.obs), xlab="Number of
cases", ylab="Cumulative", main = "Exhaustive Selection",type="l")
```





According to the comparison between two models' accuracy (ME, RMSE, MAE, MPE, and MAPE)and Lift Charts, we can see that optimal model given by exhaustive research, predictive model based on 12 predictors, (NEW + VACATION + SW + HI + S_INCOME + E_INCOME + S_POP + E_POP + SLOT + GATE + DISTANCE + PAX). However, since the predictive model based on 11 predictors(VACATION + SW + HI + S_INCOME + E_INCOME + S_POP + E_POP + SLOT + GATE + DISTANCE + PAX) given by step wise regression also has a close accuracy, we can also choose it as the final predictive model since it's less complex.