

[TYPE THE COMPANY NAME]

20-PBD-002

Shraddha P Jain

End Semester Assignment
3803(B)-MOM

The dataset *ESE 2* is a healthcare dataset which has 14635 observations with several attributes. The objective is to build a model that predicts the presence of complications of surgery of the patient. The data description is given in sheet 2 of excel sheet.

Preprocessing codes:

```
library(xlsx)
data.2 = read.xlsx("C:\\Users\\Shraddha\\Downloads\\ESE2_002.xlsx",sheetIndex
= 1)

dim(data.2)
str(data.2)
#checking for NA
lapply(data.2,function(x) { length(which(is.na(x)))})
```

```
Console Terminal x
~/ ➔
>
> lapply(data.2,function(x) { length(which(is.na(x)))})
$bmi_002
[1] 0

$Age_002
[1] 0

$sasa_status_002
[1] 0

$sahrq_ccs_002
[1] 0

$ccsComplicationRate_002
[1] 0

$ccsMort30Rate_002
[1] 0

$complication_rsi_002
[1] 0

$dow_002
[1] 0

$gender_002
[1] 0

$hour_002
[1] 0

$month_002
[1] 0

$moonphase_002
[1] 0

$moonphase_002
[1] 0

$mort30_002
[1] 0

$mortality_rsi_002
[1] 0

$race_002
[1] 0

$complication_002
[1] 0

> |
```

```
# converting into factor
data.2$sasa_status_002 = as.factor(data.2$sasa_status_002)
data.2$gender_002 = as.factor(data.2$gender_002)
```

```
data.2$dow_002 = as.factor(data.2$dow_002)
data.2$month_002 = as.factor(data.2$month_002)
data.2$moonphase_002 = as.factor(data.2$moonphase_002)
data.2$mort30_002 = as.factor(data.2$mort30_002)
data.2$race_002 = as.factor(data.2$race_002)
data.2$complication_002 = as.factor(data.2$complication_002)
```

Answer the following questions.

- a) Obtain the proportion of patients who had complications based on asa_status. Which among these seems to have an indication of post-surgery complication according to the given data? Justify your answer.

Ans:

```
tabl.2 = table(data.2$asa_status_002, data.2$complication_002)
prop.table(tabl.2, margin = 1)
```

```
> tabl.2 = table(data.2$asa_status_002, data.2$complication_002)
> prop.table(tabl.2, margin = 1)
```

	0	1
1	0.7295716	0.2704284
2	0.7711561	0.2288439
3	0.5416667	0.4583333

Interpretation: Above is the table of proportions of people who have complications based on their asa status. Around 27% of patients who had asa status as 1 had complications, as compared to 23% of patients with asa status as 2, and 45% of patients with asa status 3.

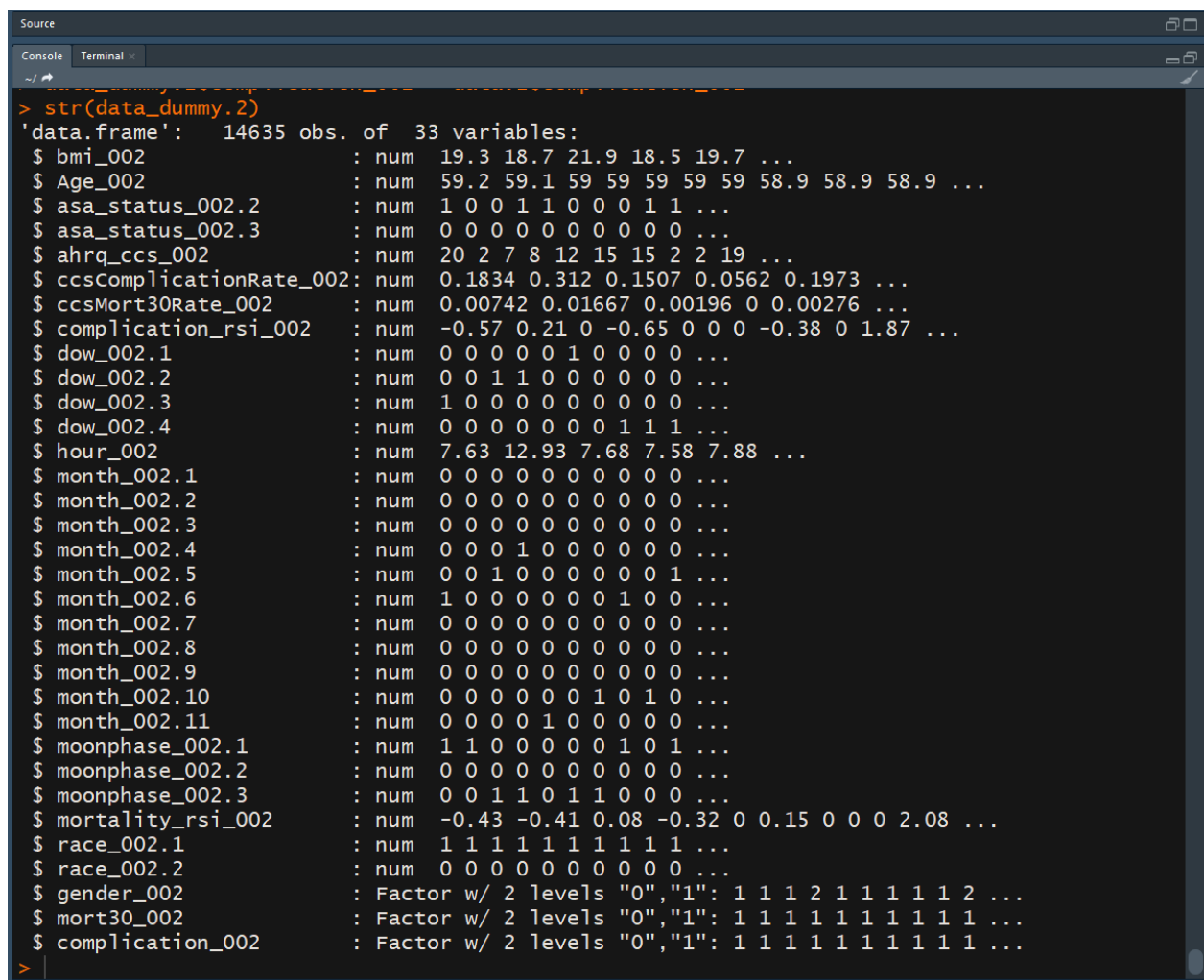
Ans: Patients with asa status 3 seem to have a higher risk of complication during surgery as compared to others. This can be justified by the above table, as they 45% of patients with asa status 3 had face complications while for patients with asa status 1 or 2, the proportion of complications in surgeries was 27% and 23% respectively

(b) Logistic regression model

Preprocessing

Creating dummy variables from factors

```
library(caret)
str(data.2)
dummy.2 <- dummyVars(" ~ .-complication_002-gender_002-mort30_002", data =
data.2, fullRank = T)
data_dummy.2 <- data.frame(predict(dummy.2, newdata = data.2))
data_dummy.2$gender_002 = data.2$gender_002
data_dummy.2$complication_002 = data.2$complication_002
data_dummy.2$mort30_002 = data.2$mort30_002
```



```
Source
Console Terminal x
~/...
> str(data_dummy.2)
'data.frame': 14635 obs. of 33 variables:
 $ bmi_002 : num 19.3 18.7 21.9 18.5 19.7 ...
 $ Age_002 : num 59.2 59.1 59 59 59 59 59 58.9 58.9 58.9 ...
 $ asa_status_002.2 : num 1 0 0 1 1 0 0 0 1 1 ...
 $ asa_status_002.3 : num 0 0 0 0 0 0 0 0 0 0 ...
 $ ahrq_ccs_002 : num 20 2 7 8 12 15 15 2 2 19 ...
 $ ccsComplicationRate_002: num 0.1834 0.312 0.1507 0.0562 0.1973 ...
 $ ccsMort30Rate_002 : num 0.00742 0.01667 0.00196 0 0.00276 ...
 $ complication_rsi_002 : num -0.57 0.21 0 -0.65 0 0 0 -0.38 0 1.87 ...
 $ dow_002.1 : num 0 0 0 0 0 1 0 0 0 0 ...
 $ dow_002.2 : num 0 0 1 1 0 0 0 0 0 0 ...
 $ dow_002.3 : num 1 0 0 0 0 0 0 0 0 0 ...
 $ dow_002.4 : num 0 0 0 0 0 0 0 1 1 1 ...
 $ hour_002 : num 7.63 12.93 7.68 7.58 7.88 ...
 $ month_002.1 : num 0 0 0 0 0 0 0 0 0 0 ...
 $ month_002.2 : num 0 0 0 0 0 0 0 0 0 0 ...
 $ month_002.3 : num 0 0 0 0 0 0 0 0 0 0 ...
 $ month_002.4 : num 0 0 0 1 0 0 0 0 0 0 ...
 $ month_002.5 : num 0 0 1 0 0 0 0 0 0 1 ...
 $ month_002.6 : num 1 0 0 0 0 0 0 1 0 0 ...
 $ month_002.7 : num 0 0 0 0 0 0 0 0 0 0 ...
 $ month_002.8 : num 0 0 0 0 0 0 0 0 0 0 ...
 $ month_002.9 : num 0 0 0 0 0 0 0 0 0 0 ...
 $ month_002.10 : num 0 0 0 0 0 0 1 0 1 0 ...
 $ month_002.11 : num 0 0 0 0 1 0 0 0 0 0 ...
 $ moonphase_002.1 : num 1 1 0 0 0 0 0 1 0 1 ...
 $ moonphase_002.2 : num 0 0 0 0 0 0 0 0 0 0 ...
 $ moonphase_002.3 : num 0 0 1 1 0 1 1 0 0 0 ...
 $ mortality_rsi_002 : num -0.43 -0.41 0.08 -0.32 0 0.15 0 0 0 2.08 ...
 $ race_002.1 : num 1 1 1 1 1 1 1 1 1 1 ...
 $ race_002.2 : num 0 0 0 0 0 0 0 1 0 0 ...
 $ gender_002 : Factor w/ 2 levels "0","1": 1 1 1 2 1 1 1 1 1 2 ...
 $ mort30_002 : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
 $ complication_002 : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
>
```

train test split

```
smp_size.2<-floor(0.7*nrow(data_dummy.2))
```

```
set.seed(1024)
trainingdata.2 <- sample(seq_len(nrow(data_dummy.2)),size=smp_size.2)
training.2<-data_dummy.2[trainingdata.2,]
testing.2<-data_dummy.2[-trainingdata.2,]
```

- i. Build a binary logistic regression model to predict the probability of having complications of surgery of the patient based on the predictors. Comment on the overall model significance.

Ans:

```
model.2 = glm(complication_002~.,data = training.2,family = 'binomial')
summary(model)
```

```
> summary(model.2)

Call:
glm(formula = complication_002 ~ ., family = "binomial", data = training.2)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.9069  -0.6875  -0.3404   0.1825   2.8170

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)   -2.033370   0.257092  -7.909 2.59e-15 ***
bmi_002        -0.042194   0.003698 -11.409 < 2e-16 ***
Age_002         0.018374   0.002138   8.592 < 2e-16 ***
asa_status_002.2 0.257168   0.062273   4.130 3.63e-05 ***
asa_status_002.3 0.490136   0.157070   3.120 0.001805 **
ahrq_ccs_002    -0.009026   0.004870  -1.854 0.063808 .
ccsComplicationRate_002 8.461334   0.510662  16.569 < 2e-16 ***
ccsMort30Rate_002 -38.765141   9.087470  -4.266 1.99e-05 ***
complication_rsi_002 0.345559   0.035328   9.781 < 2e-16 ***
dow_002.1       0.354814   0.084488   4.200 2.67e-05 ***
dow_002.2       0.314098   0.089677   3.503 0.000461 ***
dow_002.3       0.255033   0.085792   2.973 0.002952 **
dow_002.4       0.357683   0.085715   4.173 3.01e-05 ***
hour_002        -0.010160   0.009588  -1.060 0.289279 .
month_002.1      0.224008   0.125450   1.786 0.074158 .
month_002.2      0.285296   0.145473   1.961 0.049861 *
month_002.3      0.212453   0.131319   1.618 0.105698
month_002.4      0.360886   0.124434   2.900 0.003729 **
month_002.5     -0.020953   0.135196  -0.155 0.876833
month_002.6      0.169563   0.124292   1.364 0.172496
month_002.7     -0.032883   0.133106  -0.247 0.804877
month_002.8     -0.873545   0.129832  -6.728 1.72e-11 ***
month_002.9      0.005057   0.139181   0.036 0.971018
month_002.10     0.331476   0.135334   2.449 0.014313 *
```

```

Console Terminal
~/
dow_002.1      0.354814    0.084488    4.200 2.67e-05 ***
dow_002.2      0.314098    0.089677    3.503 0.000461 ***
dow_002.3      0.255033    0.085792    2.973 0.002952 **
dow_002.4      0.357683    0.085715    4.173 3.01e-05 ***
hour_002       -0.010160    0.009588   -1.060 0.289279
month_002.1     0.224008    0.125450    1.786 0.074158 .
month_002.2     0.285296    0.145473    1.961 0.049861 *
month_002.3     0.212453    0.131319    1.618 0.105698
month_002.4     0.360886    0.124434    2.900 0.003729 **
month_002.5    -0.020953    0.135196   -0.155 0.876833
month_002.6     0.169563    0.124292    1.364 0.172496
month_002.7    -0.032883    0.133106   -0.247 0.804877
month_002.8    -0.873545    0.129832   -6.728 1.72e-11 ***
month_002.9     0.005057    0.139181    0.036 0.971018
month_002.10    0.331476    0.135334    2.449 0.014313 *
month_002.11    0.150129    0.130313    1.152 0.249296
moonphase_002.1 0.499557    0.076991    6.489 8.67e-11 ***
moonphase_002.2 0.346942    0.078200    4.437 9.14e-06 ***
moonphase_002.3 0.448983    0.077991    5.757 8.57e-09 ***
mortality_rsi_002 0.237648    0.038768    6.130 8.79e-10 ***
race_002.1     -0.119049    0.083829   -1.420 0.155565
race_002.2      0.074441    0.158586    0.469 0.638780
gender_0021     0.051754    0.055851    0.927 0.354110
mort30_0021    -0.963737    0.443096   -2.175 0.029630 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 11554.0  on 10243  degrees of freedom
Residual deviance:  8831.9  on 10211  degrees of freedom
AIC: 8897.9

Number of Fisher scoring iterations: 5

```

```
with(model.2,pchisq(null.deviance-deviance,df.null-df.residual,lower.tail = F))
```

```
> with(model.2,pchisq(null.deviance-deviance,df.null-df.residual,lower.tail = F))
[1] 0
```

Comment: Above I have built a logistic regression model to predict the probability of having complication in surgery using all available predictors.

The model is overall significant, but there are many variables that are not significant

- ii. Find out and report which variables are statistically significant in the logistic regression model built in (i).

```

> anova(model.2,test = 'chisq')
Analysis of Deviance Table

Model: binomial, link: logit

Response: complication_002

Terms added sequentially (first to last)


```

	Df	Deviance	Resid. Df	Resid. Dev	Pr(>Chi)	
NULL			10243	11554.0		
bmi_002	1	270.02	10242	11283.9	< 2.2e-16	***
Age_002	1	157.66	10241	11126.3	< 2.2e-16	***
asa_status_002.2	1	13.52	10240	11112.8	0.0002359	***
asa_status_002.3	1	127.59	10239	10985.2	< 2.2e-16	***
ahrq_ccs_002	1	76.55	10238	10908.6	< 2.2e-16	***
ccsComplicationRate_002	1	1179.28	10237	9729.3	< 2.2e-16	***
ccsMort30Rate_002	1	8.47	10236	9720.9	0.0036076	**
complication_rsi_002	1	486.35	10235	9234.5	< 2.2e-16	***
dow_002.1	1	12.40	10234	9222.1	0.0004295	***
dow_002.2	1	8.59	10233	9213.5	0.0033788	**
dow_002.3	1	11.42	10232	9202.1	0.0007267	***
dow_002.4	1	60.93	10231	9141.2	5.905e-15	***
hour_002	1	0.00	10230	9141.2	0.9508225	
month_002.1	1	5.40	10229	9135.8	0.0200807	*
month_002.2	1	5.15	10228	9130.6	0.0232690	*
month_002.3	1	6.88	10227	9123.7	0.0086986	**
month_002.4	1	23.12	10226	9100.6	1.524e-06	***
month_002.5	1	0.70	10225	9099.9	0.4040287	
month_002.6	1	12.09	10224	9087.8	0.0005074	***
month_002.7	1	2.12	10223	9085.7	0.1457842	
month_002.8	1	144.01	10222	8941.7	< 2.2e-16	***

month_002.6	1	12.09	10224	9087.8	0.0005074	***
month_002.7	1	2.12	10223	9085.7	0.1457842	
month_002.8	1	144.01	10222	8941.7	< 2.2e-16	***
month_002.9	1	1.29	10221	8940.4	0.2552987	
month_002.10	1	3.83	10220	8936.6	0.0504522	.
month_002.11	1	1.43	10219	8935.2	0.2313182	
moonphase_002.1	1	14.14	10218	8921.0	0.0001695	***
moonphase_002.2	1	4.01	10217	8917.0	0.0453531	*
moonphase_002.3	1	36.19	10216	8880.8	1.787e-09	***
mortality_rsi_002	1	40.08	10215	8840.7	2.444e-10	***
race_002.1	1	3.17	10214	8837.6	0.0750708	.
race_002.2	1	0.21	10213	8837.4	0.6473235	
gender_002	1	0.83	10212	8836.5	0.3609002	
mort30_002	1	4.61	10211	8831.9	0.0318381	*

```

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
>

```

Interpretation: Above, we can see the significance of all the variables used. All the variables with p value of chi square less than .05 are significant predictors. These include:

"bmi_002"	"Age_002"	"asa_status_002.2"
"asa_status_002.3"	"ahrq_ccs_002"	"ccsComplicationRate_002"
"ccsMort30Rate_002"	"complication_rsi_002"	"dow_002.1"
"dow_002.2"	"dow_002.3"	"dow_002.4"
"month_002.1"	"month_002.2"	"month_002.3"
"month_002.4"	"month_002.6"	"month_002.8"
"month_002.10"	"moonphase_002.1"	"moonphase_002.2"
"moonphase_002.3"	"mortality_rsi_002"	"race_002.1"
"mort30_002"	"complication_002"	

- iii. Build a new logistic regression model using only significant features. Report the model diagnostics followed to build this model.

Ans: `model_step = step(glm(complication_002~.,data = dat_transformed,family = 'binomial'))`
`summary(model_step)`

```
> summary(model_step)

Call:
glm(formula = complication_002 ~ bmi_002 + Age_002 + asa_status_002.2 +
  asa_status_002.3 + ahrq_ccs_002 + ccsComplicationRate_002 +
  ccsMort30Rate_002 + complication_rsi_002 + dow_002.1 + dow_002.2 +
  dow_002.3 + dow_002.4 + month_002.1 + month_002.2 + month_002.3 +
  month_002.4 + month_002.6 + month_002.8 + month_002.10 +
  month_002.11 + moonphase_002.1 + moonphase_002.2 + moonphase_002.3 +
  mortality_rsi_002 + race_002.1 + mort30_002, family = "binomial",
  data = training.2)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.9232  -0.6885  -0.3436   0.1816   2.8108

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)   -2.110686   0.219370  -9.622  < 2e-16 ***
bmi_002        -0.042767   0.003663 -11.676  < 2e-16 ***
Age_002         0.018635   0.002126   8.764  < 2e-16 ***
asa_status_002.2  0.257535   0.062139   4.144 3.41e-05 ***
asa_status_002.3  0.492806   0.156822   3.142 0.001675 **
ahrq_ccs_002    -0.008598   0.004853  -1.772 0.076455 .
ccsComplicationRate_002  8.476753   0.510498  16.605  < 2e-16 ***
ccsMort30Rate_002 -38.903459   9.078289  -4.285 1.82e-05 ***
complication_rsi_002  0.345240   0.035289   9.783  < 2e-16 ***
dow_002.1       0.349187   0.084305   4.142 3.44e-05 ***
dow_002.2       0.311692   0.089585   3.479 0.000503 ***
dow_002.3       0.252100   0.085682   2.942 0.003258 **
dow_002.4       0.346854   0.085199   4.071 4.68e-05 ***
month_002.1     0.233528   0.098919   2.361 0.018236 *
month_002.2     0.222175   0.122127   1.819 0.071312 .
month_002.3     0.222175   0.122127   1.819 0.071312 .
month_002.4     0.222175   0.122127   1.819 0.071312 .
month_002.6     0.222175   0.122127   1.819 0.071312 .
month_002.8     0.222175   0.122127   1.819 0.071312 .
month_002.10    0.222175   0.122127   1.819 0.071312 .
month_002.11    0.222175   0.122127   1.819 0.071312 .
moonphase_002.1  0.222175   0.122127   1.819 0.071312 .
moonphase_002.2  0.222175   0.122127   1.819 0.071312 .
moonphase_002.3  0.222175   0.122127   1.819 0.071312 .
mortality_rsi_002 0.222175   0.122127   1.819 0.071312 .
race_002.1      0.222175   0.122127   1.819 0.071312 .
mort30_002      0.222175   0.122127   1.819 0.071312 .
```

```

dow_002.3      0.232100    0.083682    2.942 0.003238 ***
dow_002.4      0.346854    0.085199    4.071 4.68e-05 ***
month_002.1     0.233528    0.098919    2.361 0.018236 *
month_002.2     0.293475    0.123197    2.382 0.017212 *
month_002.3     0.227498    0.106278    2.141 0.032307 *
month_002.4     0.373481    0.097635    3.825 0.000131 ***
month_002.6     0.183399    0.097486    1.881 0.059933 .
month_002.8    -0.854846    0.104594   -8.173 3.01e-16 ***
month_002.10    0.340094    0.111042    3.063 0.002193 **
month_002.11    0.158346    0.104869    1.510 0.131057
moonphase_002.1 0.494006    0.076830    6.430 1.28e-10 ***
moonphase_002.2 0.341194    0.078010    4.374 1.22e-05 ***
moonphase_002.3 0.445326    0.077776    5.726 1.03e-08 ***
mortality_rsi_002 0.231919    0.038572    6.013 1.83e-09 ***
race_002.1     -0.132170    0.074275   -1.779 0.075164 .
mort30_0021    -0.961468    0.442614   -2.172 0.029837 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 11554.0  on 10243  degrees of freedom
Residual deviance:  8834.3  on 10217  degrees of freedom
AIC: 8888.3

Number of Fisher Scoring iterations: 5
> |

```

The model diagnostic used for this is AIC.

iv. Write an estimated logistic regression model obtained in (iii)

Ans: The estimated logistic regression model is:

$$p = \frac{1}{1 + e^{-2.11 - .0427 * bmi + 0.186 * Age + 0.25 * asa_Status1}}$$

(similarly all the variables will be added)

v. Use Youden's index to find the most optimal cut-off probability value for the best model chosen in (iii).

```
p = predict(model_step,type = 'response')
```

```
p
```

```
actual = training.2$complication_002
```

```
acc <- function(mod, pp, p,actual) {
```

```

out = c()
## Classification table
pred <- ifelse(p<pp,0,1)
# pred_test <- ifelse(p_test<pp,0,1)
tab<- table(pred,actual = actual)
out$sumtab<- addmargins(tab,FUN=sum)

TAP <- sum(tab[,2]) #Total actual positives
TAN <- sum(tab[,1]) # Total actual negatives

TP <- out$sumtab[2,2]
TN <- out$sumtab[1,1]
FP <- out$sumtab[2,1]
FN <- out$sumtab[1,2]
out$TPR = TP/TAP # Sensitivity or recall ## ability to correctly classify
out$FPR = FP/TAN
out$TNR = TN/TAN # Specificity
out$FNR = FN/TAP
out$accuracy = (TP+TN)/(TAN+TAP)
out$miss_classification_error = 1-out$accuracy

out$precision = TP/(TP+FP)
# conditional probability of being positive when predicted positive
out$specificity <- TN/TAN
out$f_score = TP/(TP+0.5*(FP+FN))
out$cut_off = pp
out$youden = out$TPR+out$TNR-1
return(out)
}
acc(model_step,0.5, p,actual)$youden
acc(model_step,0.4, p,actual)$youden
acc(model_step,0.3, p,actual)$youden
acc(model_step,0.2, p,actual)$youden
acc(model_step,0.1, p,actual)$youden
acc(model_step,0.25, p,actual)$youden
acc(model_step,0.24, p,actual)$youden
acc(model_step,0.238, p,actual)$youden

```

```

[1] 0.4899408
> acc(model_step,0.1, p,actual)$youden
Margins computed over dimensions
in the following order:
1: pred
2: actual
[1] 0.3718541
> acc(model_step,0.25, p,actual)$youden
Margins computed over dimensions
in the following order:
1: pred
2: actual
[1] 0.4821578
> acc(model_step,0.24, p,actual)$youden
Margins computed over dimensions
in the following order:
1: pred
2: actual
[1] 0.4821728
> acc(model_step,0.238, p,actual)$youden
Margins computed over dimensions
in the following order:
1: pred
2: actual
[1] 0.4814965

```

The optimal youden index was found to be at cut off 0.24.

(c)Decision tree classifier model

```

# normalizing the variables
attach(data_dummy.2)
Age_002 = (Age_002 - mean(Age_002))/sd(Age_002)
bmi_002= (bmi_002 - mean(bmi_002))/sd(bmi_002)
ccsComplicationRate_002= ( ccsComplicationRate_002-
mean(ccsComplicationRate_002))/sd(ccsComplicationRate_002)
ccsMort30Rate_002= (ccsMort30Rate_002 -
mean(ccsMort30Rate_002))/sd(ccsMort30Rate_002)
complication_rsi_002= ( complication_rsi_002-
mean(complication_rsi_002))/sd(complication_rsi_002)
detach(data_dummy.2)

# Test Train split
smp_size.2<-floor(0.7*nrow(data_dummy.2))
set.seed(1024)

```

```
trainingdata.2 <- sample(seq_len(nrow(data_dummy.2)),size=smp_size.2)
training.2<-data_dummy.2[trainingdata.2,]
testing.2<-data_dummy.2[-trainingdata.2,]
```

- i. Build Random Forest decision tree and clearly identify and report predictor which is classifying the patient having complications of post-surgery.

```
Ranfor= randomForest(complication_002~., data = training.2)
summary(Ranfor)
print(Ranfor)
```

```
> Randomforest_002= randomForest(complication_002~., data = training.2)
> print(Randomforest_002)

Call:
randomForest(formula = complication_002 ~ ., data = training.2)
      Type of random forest: classification
      Number of trees: 500
No. of variables tried at each split: 5

      OOB estimate of  error rate: 13.98%
Confusion matrix:
      0      1 class.error
0 7347  321  0.04186228
1 1111 1465  0.43128882
```

Above is the random forest model.

Below are the variables used for the decision tree

```
> importance(Randomforest_002)
```

	MeanDecreaseGini
bmi_002	394.894808
Age_002	781.092919
asa_status_002.2	49.143322
asa_status_002.3	16.065613
ahrq_ccs_002	145.570857
ccsComplicationRate_002	298.188442
ccsMort30Rate_002	169.666706
complication_rsi_002	472.678302
dow_002.1	36.267476
dow_002.2	32.733013
dow_002.3	35.530217
dow_002.4	35.369240
hour_002	282.145923
month_002.1	28.377854
month_002.2	25.256290
month_002.3	25.136139
month_002.4	30.089250
month_002.5	23.310302
month_002.6	29.334994
month_002.7	25.858633
month_002.8	53.760218
month_002.9	23.218827
month_002.10	23.159694
month_002.11	26.244929
moonphase_002.1	42.066888
moonphase_002.2	38.229533
moonphase_002.3	39.839525
mortality_rsi_002	447.707413
race_002.1	34.279643
race_002.2	15.291180
gender_002	39.974885
mort30_002	2.800705

- ii. Also plot that decision tree and report the most important splitting criteria (rule).

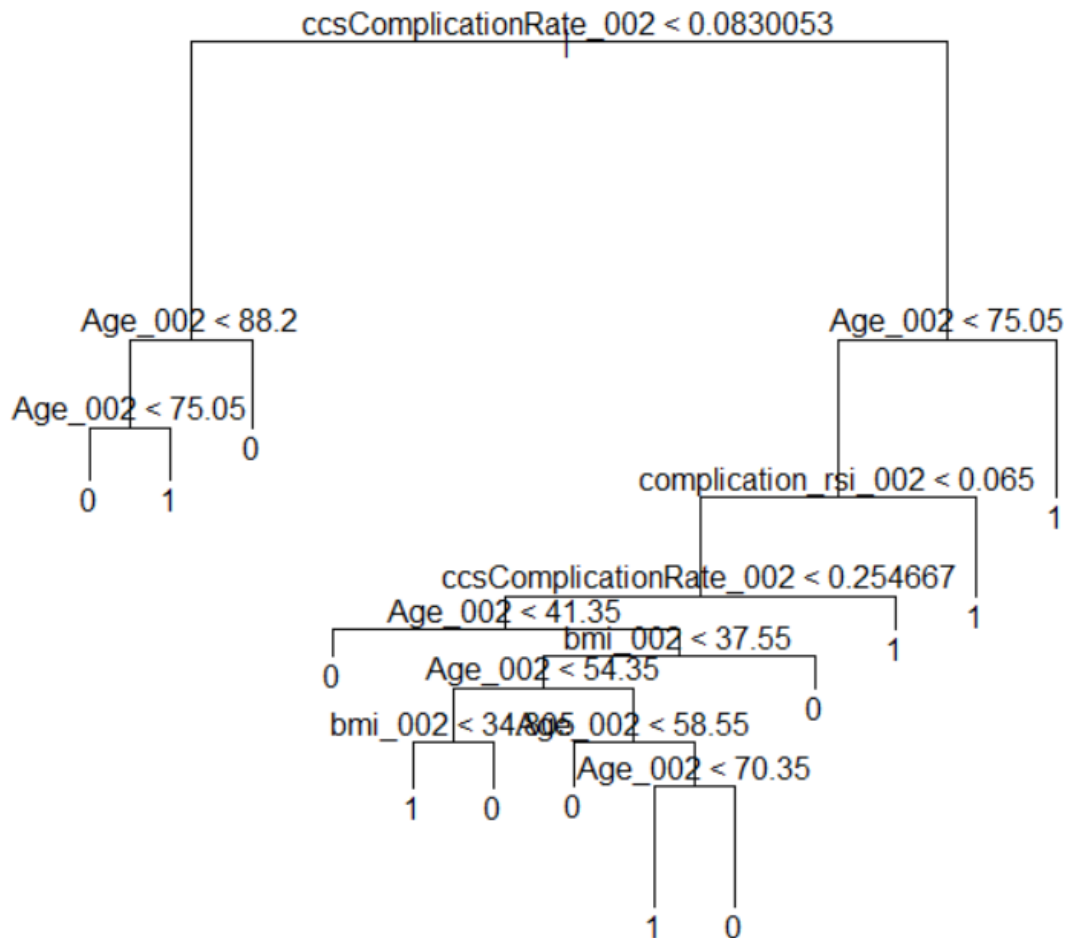
```
treemod.2= tree(training.2$complication_002~., data=training.2)
```

```
summary(treemod.2)
```

```
plot (treemod.2)
```

```
text(treemod.2,pretty=0)
```

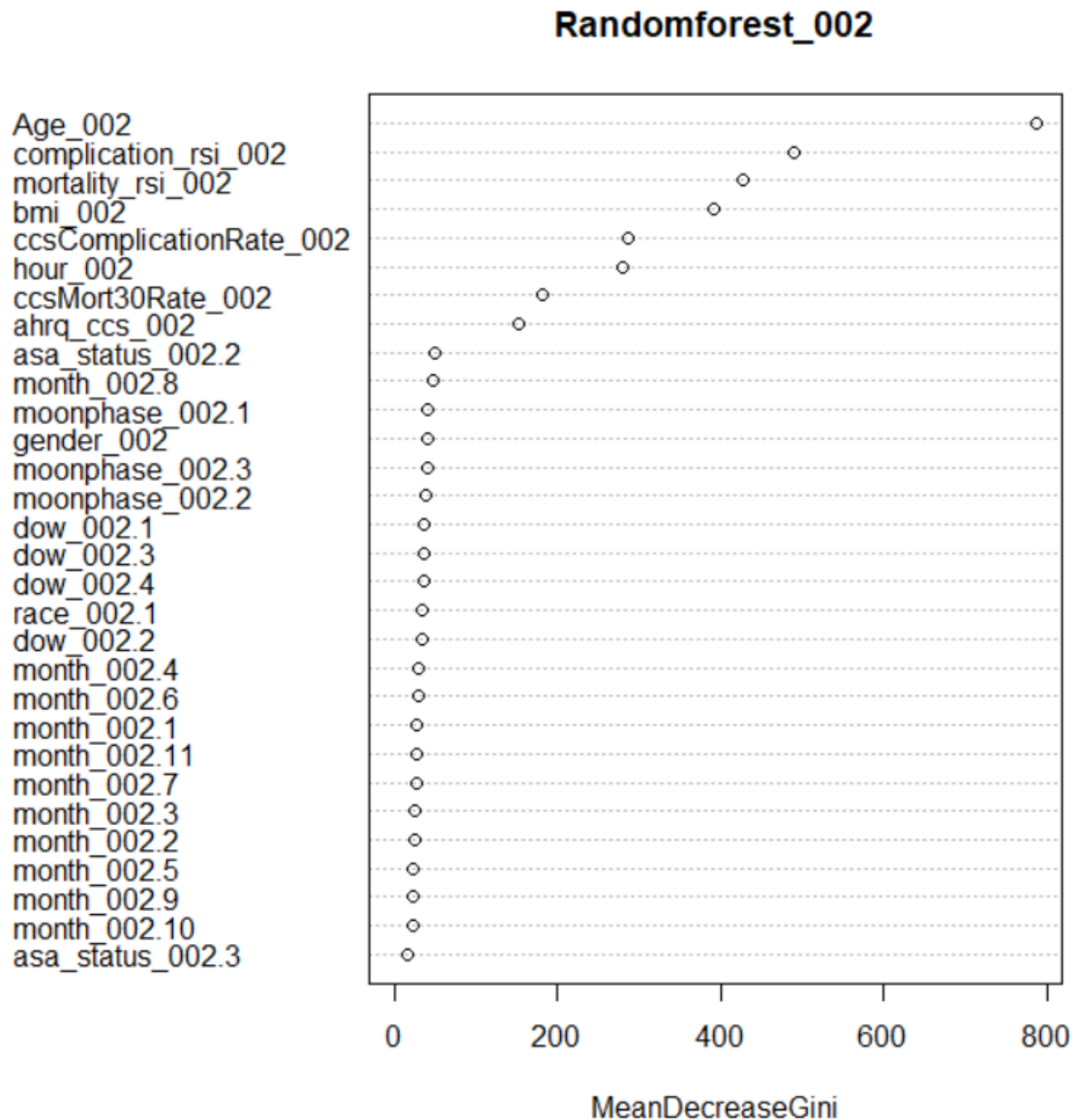
```
Classification tree:
tree(formula = training.2$complication_002 ~ ., data = training.2)
Variables actually used in tree construction:
[1] "ccsComplicationRate_002" "Age_002"                "complication_rsi_002"
[4] "bmi_002"
Number of terminal nodes: 13
Residual mean deviance: 0.6515 = 6666 / 10230
Misclassification error rate: 0.1393 = 1427 / 10244
```



The most important splitting criteria is whether the ccsCompilation rate is < 0.083003 or not.

- iii. Use variance importance plot, report the variables which are classifying the surgery complication.

```
varImpPlot(Randomforest_002)
```



The important variables for classifying are age, complication_rsi, mortality_rsi, bmi, ccsCompilationRate, hour, ccsMort30Rate and ahrq_ccs.

(d) Compare the logistic regression model and decision tree classifier performance using confusion matrix with specific accuracy measures or ROC and AUC?

```
acc <- function(mod, pp, p,actual) {  
  out = c()  
  ## Classification table  
  pred <- ifelse(p<pp,0,1)  
  # pred_test <- ifelse(p_test<pp,0,1)  
  tab<- table(pred,actual = actual)  
  out$sumtab<- addmargins(tab,FUN=sum)  
  
  TAP <- sum(tab[,2]) #Total actual positives  
  TAN <- sum(tab[,1]) # Total actual negatives  
  
  TP <- out$sumtab[2,2]  
  TN <- out$sumtab[1,1]  
  FP <- out$sumtab[2,1]  
  FN <- out$sumtab[1,2]  
  out$TPR = TP/TAP # Sensitivity or recall ## ability to correctly classify  
  out$FPR = FP/TAN  
  out$TNR = TN/TAN # Specificity  
  out$FNR = FN/TAP  
  out$accuracy = (TP+TN)/(TAN+TAP)  
  out$miss_classification_error = 1-out$accuracy  
  
  out$precision = TP/(TP+FP)  
  # conditional probability of being positive when predicted positive  
  out$specificity <- TN/TAN  
  out$f_score = TP/(TP+0.5*(FP+FN))  
  out$cut_off = pp  
  out$youden = out$TPR+out$TNR-1  
  return(out)  
}  
# Accuracy of logistic regression, with cut off probability = 0.24  
acc(model_step,0.24, p,actual)
```

```
p = predict(model_step,type = 'response',newdata = testing.2)
p
```

```
actual = testing.2$complication_002
```

```
> acc(model_step,0.24, p,actual)
```

```
Margins computed over dimensions  
in the following order:
```

```
1: pred
```

```
2: actual
```

```
$sumtab
```

	actual		
pred	0	1	sum
0	2292	275	2567
1	985	839	1824
sum	3277	1114	4391

```
$TPR
```

```
[1] 0.7531418
```

```
$FPR
```

```
[1] 0.3005798
```

```
$TNR
```

```
[1] 0.6994202
```

```
$FNR
```

```
[1] 0.2468582
```

```
$accuracy
```

```
[1] 0.7130494
```

```
$miss_classification_error
```

```
[1] 0.2869506
```

```
$precision
```

```
[1] 0.4599781
```

```
$specificity
```

```
[1] 0.6994202
```

```

$accuracy
[1] 0.7130494

$miss_classification_error
[1] 0.2869506

$precision
[1] 0.4599781

$specificity
[1] 0.6994202

$f_score
[1] 0.5711368

$cut_off
[1] 0.24

$youden
[1] 0.452562

```

```

# accuracy of random forest
p_random = predict(Randomforest_002,type = 'response',newdata = testing.2)
actual = testing.2$complication_002
acc <- function(mod, p,actual) {
  out = c()
  ## Classification table
  #pred <- ifelse(p<pp,0,1)
  pred = p
# pred_test <- ifelse(p_test<pp,0,1)
  tab<- table(pred,actual = actual)
  out$sumtab<- addmargins(tab,FUN=sum)

  TAP <- sum(tab[,2]) #Total actual positives
  TAN <- sum(tab[,1]) # Total actual negatives

  TP <- out$sumtab[2,2]
  TN <- out$sumtab[1,1]
  FP <- out$sumtab[2,1]
  FN <- out$sumtab[1,2]
  out$TPR = TP/TAP # Sensitivity or recall ## ability to correctly classify

```

```
out$FPR = FP/TAN
out$TNR = TN/TAN # Specificity
out$FNR = FN/TAP
out$accuracy = (TP+TN)/(TAN+TAP)
out$miss_classification_error = 1-out$accuracy

out$precision = TP/(TP+FP)
# conditional probability of being positive when predicted positive
out$specificity <- TN/TAN
out$f_score = TP/(TP+0.5*(FP+FN))
#out$cut_off = pp
out$youden = out$TPR+out$TNR-1
return(out)
}
```

```
> acc(Randomforest_002,p= p_random,actual)
```

```
Margins computed over dimensions  
in the following order:
```

```
1: pred
```

```
2: actual
```

```
$sumtab
```

	actual		
pred	0	1	sum
0	3103	486	3589
1	174	628	802
sum	3277	1114	4391

```
$TPR
```

```
[1] 0.5637343
```

```
$FPR
```

```
[1] 0.05309735
```

```
$TNR
```

```
[1] 0.9469027
```

```
$FNR
```

```
[1] 0.4362657
```

```
$accuracy
```

```
[1] 0.8496926
```

```
$miss_classification_error
```

```
[1] 0.1503074
```

```
$precision
```

```
[1] 0.7830424
```

```
$specificity
```

```
[1] 0.9469027
```

```
$miss_classification_error  
[1] 0.1503074
```

```
$precision  
[1] 0.7830424
```

```
$specificity  
[1] 0.9469027
```

```
$f_score  
[1] 0.6555324
```

```
$youden  
[1] 0.5106369
```

```
acc(Randomforest_002,p= p_random,actual)
```