

# Hospital Readmission sample solution

This chunk reads in the data, relevels factors, and prints a summary.

```
# Loading data
readmission <- read.csv(file="readmission.csv")

vars <- colnames(readmission)[c(2,3,5,9)] #variables to relevel
for (i in vars){
  table <- as.data.frame(table(readmission[,i]))
  max <- which.max(table[,2])
  level.name <- as.character(table[max,1])
  readmission[,i] <- relevel(readmission[,i], ref = level.name)
}
summary(readmission)
```

```
## Readmission.Status Gender          Race          ER
## Min. :0.0000 F:38011 White :56124 Min. :0.0000
## 1st Qu.:0.0000 M:28771 Black : 7099 1st Qu.:0.0000
## Median :0.0000          Hispanic: 1286 Median :0.0000
## Mean :0.1259          Others : 2273 Mean :0.5083
## 3rd Qu.:0.0000          Max. :9.0000
## Max. :1.0000
## DRG.Class          LOS          Age          HCC.Riskscore
## MED :35771 Min. : 1.000 Min. : 24.00 Min. : 0.079
## SURG :30447 1st Qu.: 3.000 1st Qu.: 67.00 1st Qu.: 1.107
## UNGROUP: 564 Median : 5.000 Median : 75.00 Median : 1.865
##          Mean : 6.693 Mean : 73.64 Mean : 2.345
##          3rd Qu.: 8.000 3rd Qu.: 83.00 3rd Qu.: 3.173
##          Max. :36.000 Max. :101.00 Max. :12.307
## DRG.Complication
## MedicalMCC.CC:18110
## MedicalNoC :12310
## Other : 9345
## SurgMCC.CC :15468
## SurgNoC :11549
##
```

## Task 1

I have elected to make a table for ER and histograms for the other three variables.

```
library(ggplot2)

## Warning: replacing previous import by 'rlang::=' when loading 'dplyr'
## Warning: replacing previous import by 'rlang::.data' when loading 'dplyr'
## Warning: replacing previous import by 'rlang::as_label' when loading
## 'dplyr'
## Warning: replacing previous import by 'rlang::as_name' when loading 'dplyr'
## Warning: replacing previous import by 'rlang::dots_n' when loading 'dplyr'
```

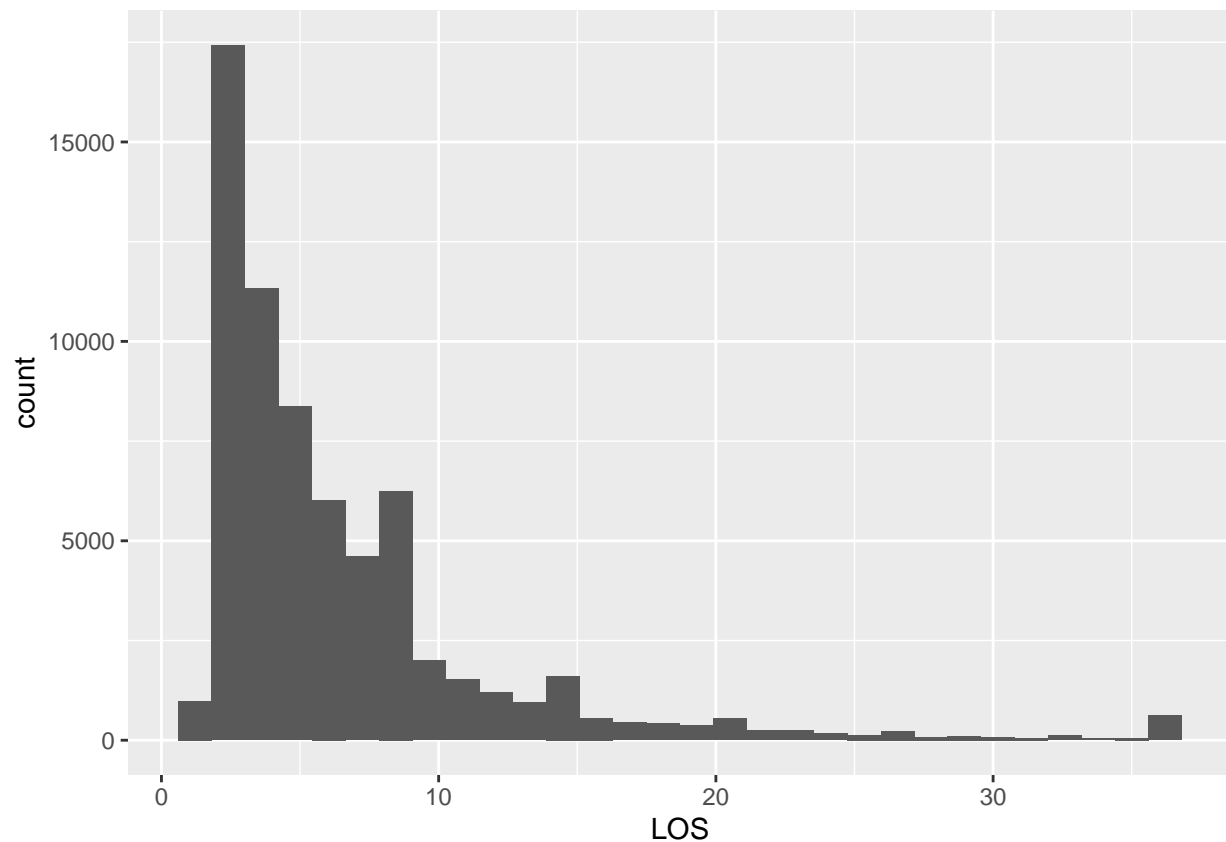
```
## Warning: replacing previous import by 'rlang::enquo' when loading 'dplyr'
## Warning: replacing previous import by 'rlang::enquos' when loading 'dplyr'
## Warning: replacing previous import by 'rlang::expr' when loading 'dplyr'
## Warning: replacing previous import by 'rlang::sym' when loading 'dplyr'
## Warning: replacing previous import by 'rlang::syms' when loading 'dplyr'
```

```
table(readmission$ER)
```

```
##
##      0      1      2      3      4      5      6      7      9
## 43086 16280  5286  1572   438   105    10     3     2
```

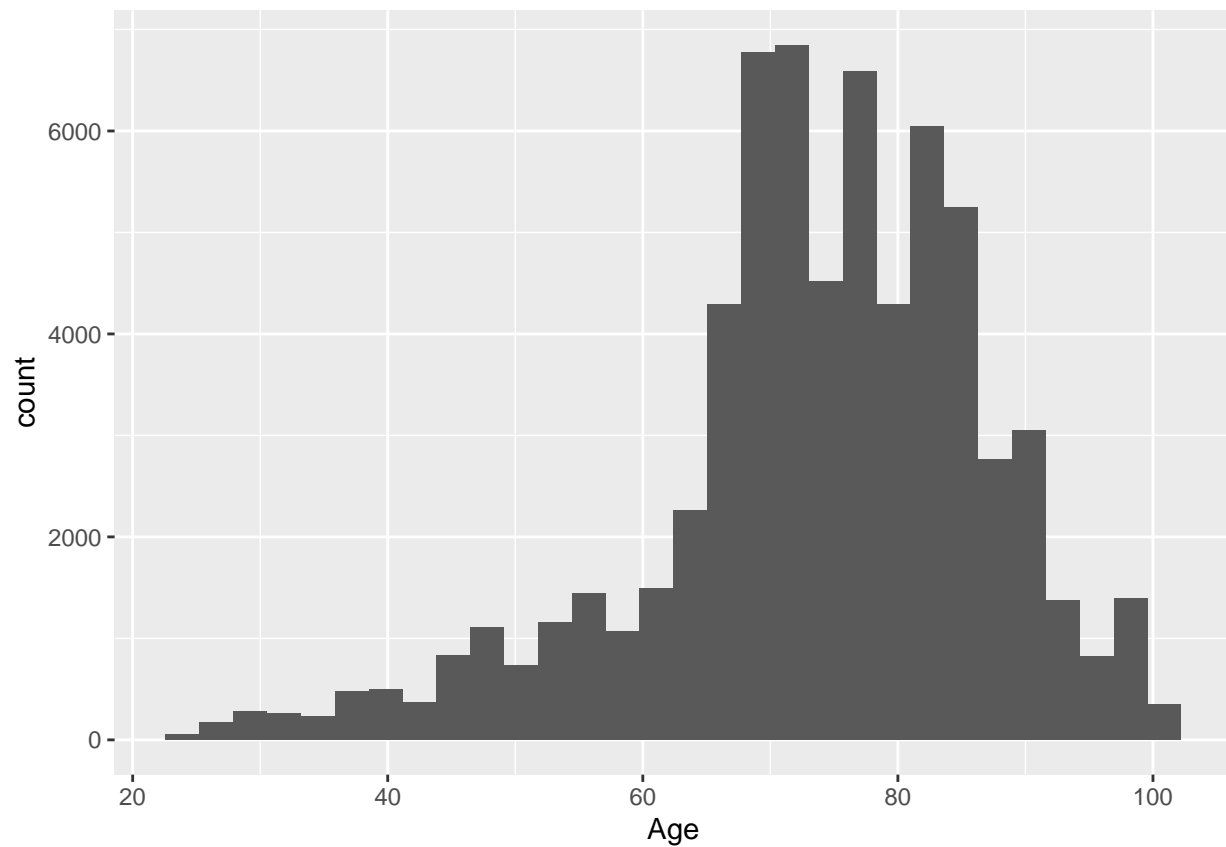
```
ggplot(readmission,aes(x=LOS))+geom_histogram()
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



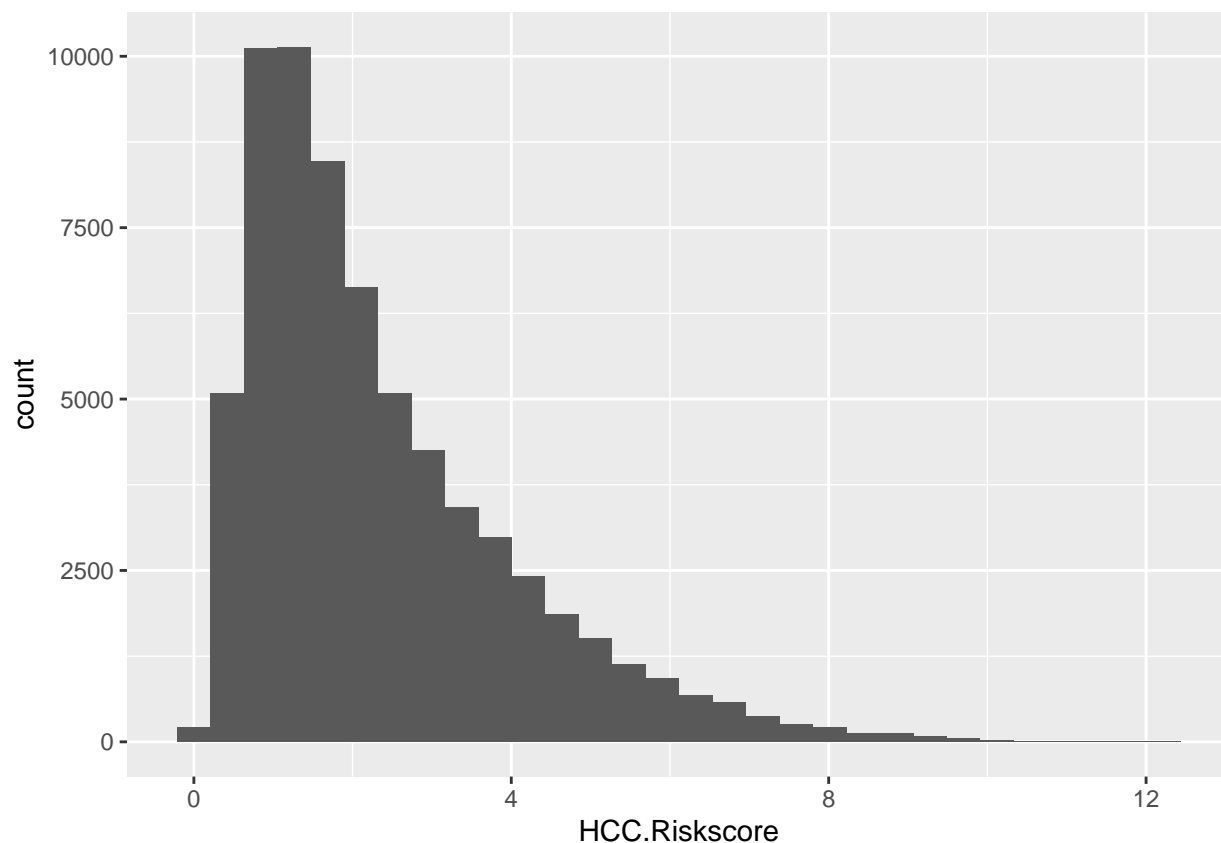
```
ggplot(readmission,aes(x=Age))+geom_histogram()
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



```
ggplot(readmission,aes(x=HCC.Riskscore))+geom_histogram()
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



I will use log transformations of the LOS and HCC.RiskScore variables. I also create an indicator variables for those under age 65. The following chunk does these.

```
#two log transforms and removal of original variables
readmission$logLOS <- log(readmission$LOS)
readmission$logRiskScore <- log(readmission$HCC.RiskScore)
readmission$LOS <- NULL
readmission$HCC.RiskScore <- NULL
readmission$Under65 <- ifelse(readmission$Age < 65, 1, 0)
summary(readmission)
```

```
## Readmission.Status Gender      Race      ER
## Min.   :0.0000    F:38011  White   :56124  Min.   :0.0000
## 1st Qu.:0.0000    M:28771  Black   : 7099  1st Qu.:0.0000
## Median :0.0000                Hispanic: 1286  Median :0.0000
## Mean   :0.1259                Others   : 2273  Mean   :0.5083
## 3rd Qu.:0.0000                                3rd Qu.:1.0000
## Max.   :1.0000                                Max.   :9.0000
## DRG.Class      Age      DRG.Complication  logLOS
## MED      :35771  Min.    : 24.00  MedicalMCC.CC:18110  Min.    :0.000
## SURG      :30447  1st Qu.: 67.00  MedicalNoC    :12310  1st Qu.:1.099
## UNGROUP: 564    Median : 75.00  Other         : 9345  Median :1.609
##                Mean   : 73.64  SurgMCC.CC    :15468  Mean   :1.653
##                3rd Qu.: 83.00  SurgNoC       :11549  3rd Qu.:2.079
##                Max.    :101.00                Max.    :3.584
## logRiskScore  Under65
## Min.    :-2.5383  Min.    :0.0000
## 1st Qu.: 0.1017  1st Qu.:0.0000
```

```
## Median : 0.6235   Median :0.0000
## Mean   : 0.5999   Mean    :0.1684
## 3rd Qu.: 1.1547   3rd Qu.:0.0000
## Max.   : 2.5102   Max.    :1.0000
```

## Task 2

This chunk creates a tabular view of the two variables.

```
table(readmission$DRG.Class,readmission$DRG.Complication)
```

```
##
##           MedicalMCC.CC MedicalNoC Other SurgMCC.CC SurgNoC
## MED                18104      12310  5357           0         0
## SURG                 6         0  3424       15468      11549
## UNGROUP              0         0   564           0         0
```

Six items will be deleted and the two existing variables recoded as a single factor variable. The following code does that.

```
readmission.new <- readmission #preserve the original data until the work can be checked
readmission.new <- readmission.new[!(readmission.new$DRG.Complication=="MedicalMCC.CC" & readmission.new$DRG.Complication=="MedicalNoC"),]
readmission.new$DRG <- ifelse(readmission.new$DRG.Complication=="MedicalMCC.CC", "Med.C",
                             ifelse(readmission.new$DRG.Complication=="MedicalNoC", "Med.NoC",
                                     ifelse(readmission.new$DRG.Complication=="SurgMCC.CC", "Surg.C",
                                             ifelse(readmission.new$DRG.Complication=="SurgNoC", "Surg.NoC",
                                                     ifelse(readmission.new$DRG.Class=="UNGROUP", "UNGROUP",
                                                             ifelse(readmission.new$DRG.Complication=="Other"&readmission.new$DRG.Class=="MED", "OtherMED", "OtherSURG")))))
readmission.new$DRG <- as.factor(readmission.new$DRG)
table(readmission.new$DRG)
```

```
##
##      Med.C  Med.NoC  OtherMED OtherSURG      Surg.C  Surg.NoC  UNGROUP
##      18104    12310    5357    3424    15468    11549    564
```

```
readmission.new$DRG.Class <- NULL
readmission.new$DRG.Complication <- NULL
```

Relevel the new variable.

```
table <- as.data.frame(table(readmission.new[, "DRG"]))
max <- which.max(table[, 2])
level.name <- as.character(table[max, 1])
readmission.new[, "DRG"] <- relevel(readmission.new[, "DRG"], ref = level.name)
table(readmission.new$DRG)
```

```
##
##      Med.C  Med.NoC  OtherMED OtherSURG      Surg.C  Surg.NoC  UNGROUP
##      18104    12310    5357    3424    15468    11549    564
```

Accept the new dataframe by renaming it back to readmission.

```
readmission <- readmission.new
readmission.new <- NULL
summary(readmission)
```

```
## Readmission.Status Gender      Race      ER
## Min.      :0.0000    F:38005  White    :56120  Min.      :0.0000
```

```
## 1st Qu.:0.0000      M:28771  Black   : 7097  1st Qu.:0.0000
## Median :0.0000      Hispanic: 1286 Median :0.0000
## Mean   :0.1259      Others   : 2273 Mean   :0.5083
## 3rd Qu.:0.0000      3rd Qu.:1.0000
## Max.    :1.0000      Max.    :9.0000
##
##      Age      logLOS      logRiskscore      Under65
## Min.    : 24.00  Min.    :0.000  Min.    :-2.5383  Min.    :0.0000
## 1st Qu.: 67.00  1st Qu.:1.099  1st Qu.: 0.1017  1st Qu.:0.0000
## Median : 75.00  Median :1.609  Median : 0.6238  Median :0.0000
## Mean   : 73.64  Mean   :1.653  Mean   : 0.6000  Mean   :0.1684
## 3rd Qu.: 83.00  3rd Qu.:2.079  3rd Qu.: 1.1547  3rd Qu.:0.0000
## Max.   :101.00  Max.   :3.584  Max.   : 2.5102  Max.   :1.0000
##
##      DRG
## Med.C      :18104
## Med.NoC    :12310
## OtherMED   : 5357
## OtherSURG  :3424
## Surg.C     :15468
## Surg.NoC   :11549
## UNGROUP    : 564
```

### Task 3

This code performs cluster analysis for from 1 to 12 clusters and constructs an elbow plot.

```
nstart.val <- 20
cluster_vars <- readmission[c('logLOS', 'Age')]
for(i in 1:ncol(cluster_vars)){
  cluster_vars[,i] <- scale(cluster_vars[,i])
}
km1 <- kmeans(cluster_vars,centers=1,nstart=nstart.val)
km2 <- kmeans(cluster_vars,centers=2,nstart=nstart.val)
km3 <- kmeans(cluster_vars,centers=3,nstart=nstart.val)
km4 <- kmeans(cluster_vars,centers=4,nstart=nstart.val)
km5 <- kmeans(cluster_vars,centers=5,nstart=nstart.val)
km6 <- kmeans(cluster_vars,centers=6,nstart=nstart.val)
km7 <- kmeans(cluster_vars,centers=7,nstart=nstart.val)
km8 <- kmeans(cluster_vars,centers=8,nstart=nstart.val)
km9 <- kmeans(cluster_vars,centers=9,nstart=nstart.val)
km10 <- kmeans(cluster_vars,centers=10,nstart=nstart.val)
km11 <- kmeans(cluster_vars,centers=11,nstart=nstart.val)
km12 <- kmeans(cluster_vars,centers=12,nstart=nstart.val)

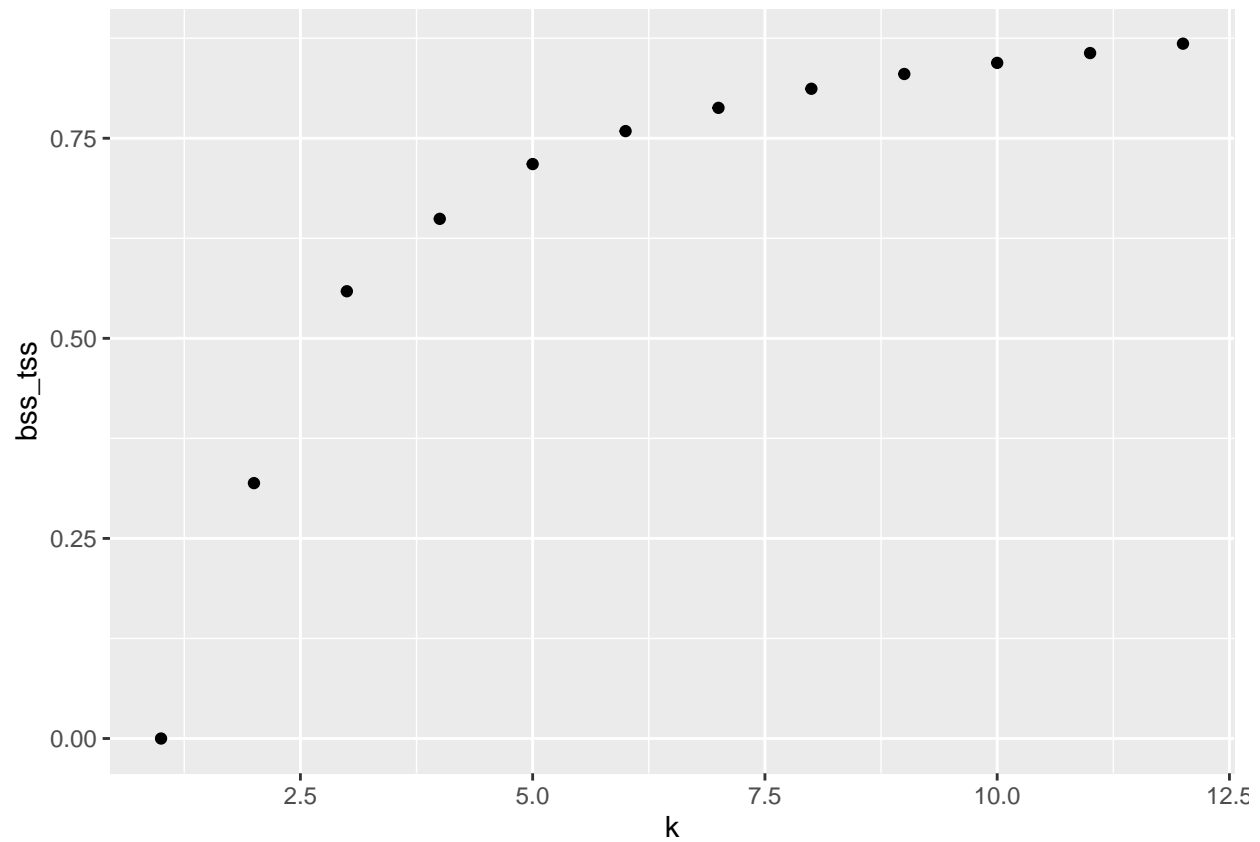
var.exp <- data.frame(k = c(1:12),
                      bss_tss = c(km1$betweenss/km1$totss,
                                   km2$betweenss/km2$totss,
                                   km3$betweenss/km3$totss,
                                   km4$betweenss/km4$totss,
                                   km5$betweenss/km5$totss,
                                   km6$betweenss/km6$totss,
                                   km7$betweenss/km7$totss,
                                   km8$betweenss/km8$totss,
```

```

km9$betweenss/km9$totss,
km10$betweenss/km10$totss,
km11$betweenss/km11$totss,
km12$betweenss/km12$totss))

ggplot(var.exp,aes(x=k,y=bss_tss))+geom_point()

```



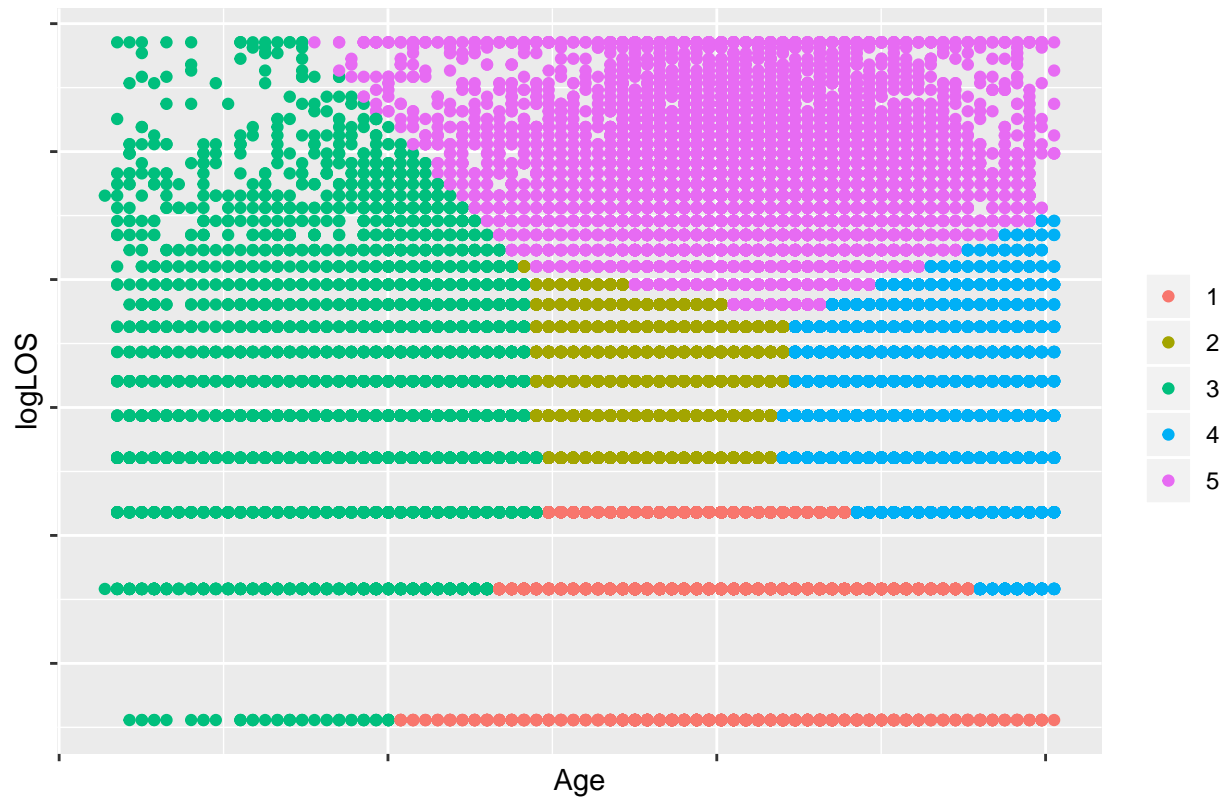
The following chunk creates the new variable based on 5 clusters.

```

LOS_Age_Clust <- as.factor(km5$cluster) #This creates a new variable based on having 5 clusters.
cluster_vars$LOS_Age_Clust <- LOS_Age_Clust
ggplot(data = cluster_vars, aes(x = Age, y = logLOS, col = LOS_Age_Clust)) + geom_point() + theme(axis.

```

### Clustering with 5 groups

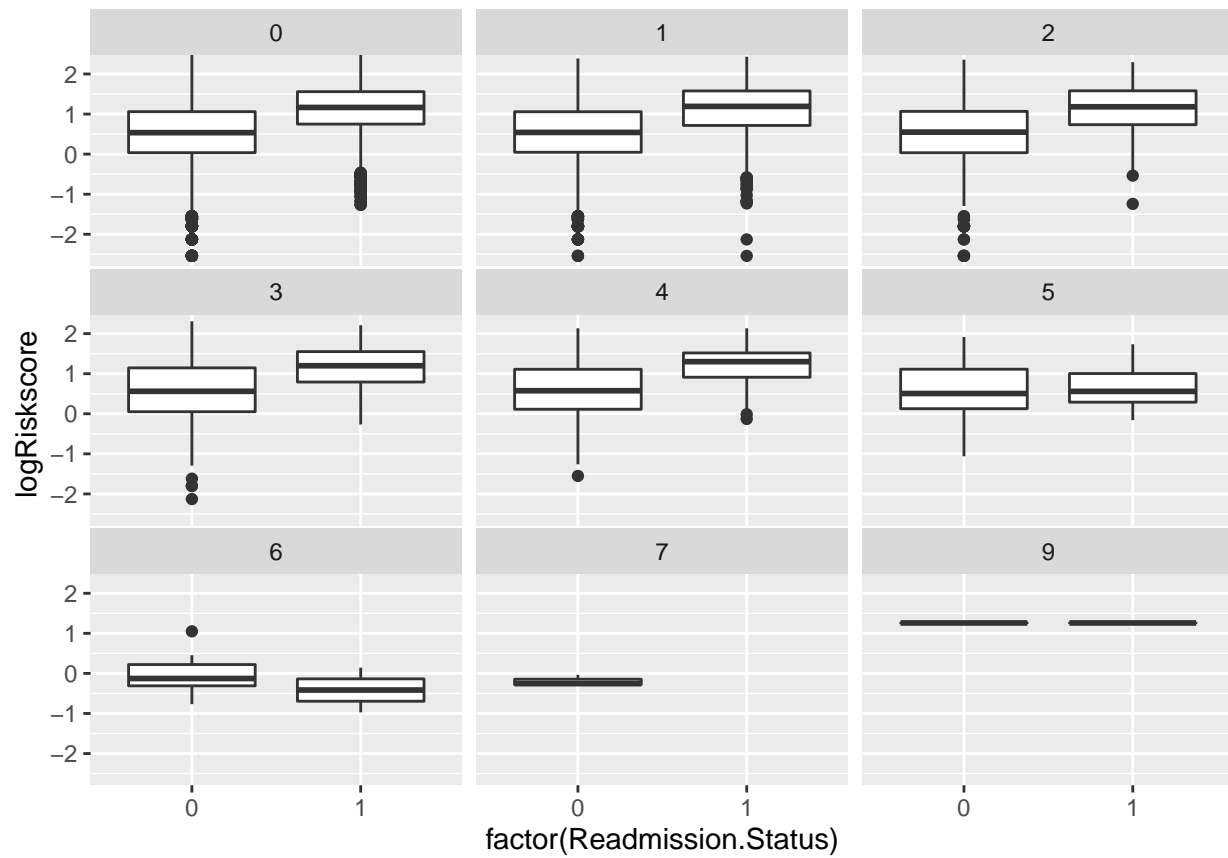


### Task 4

My first look is at ER and logRiskScore.

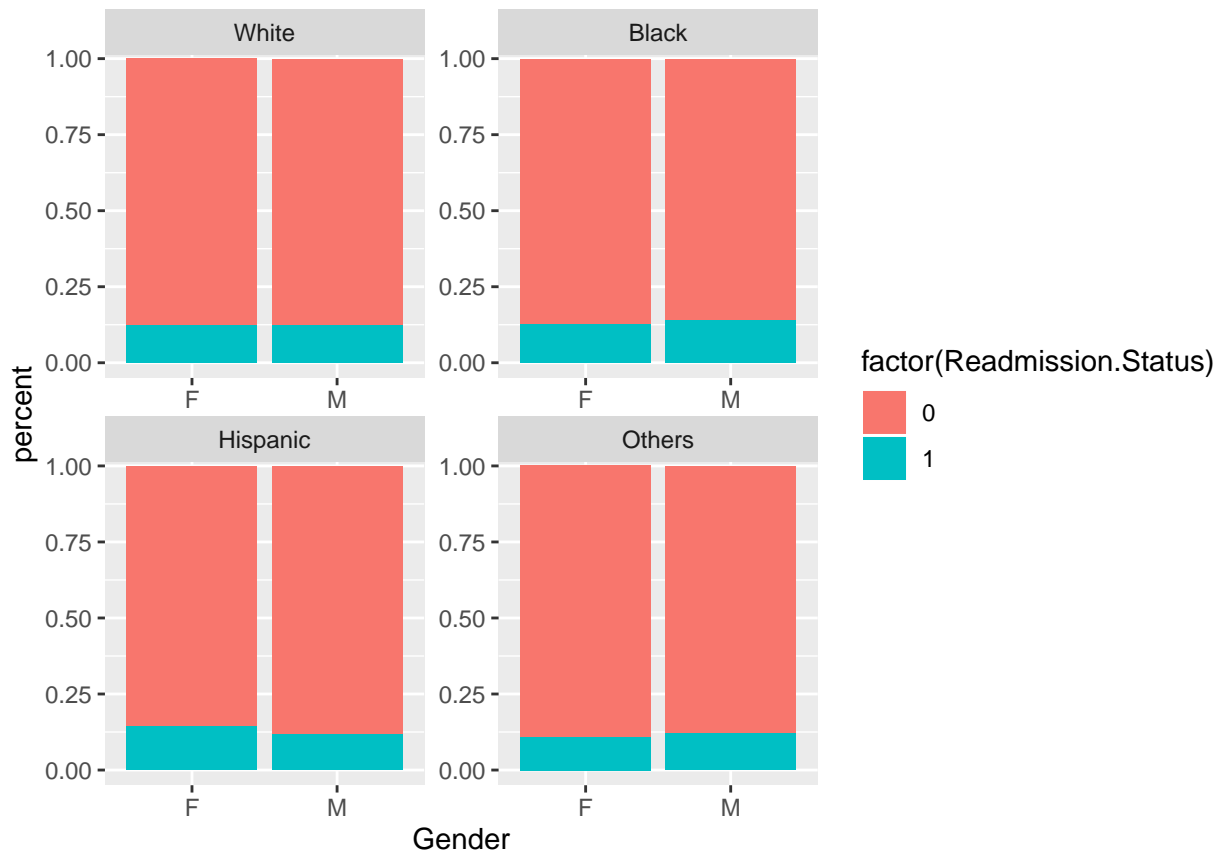
```
ggplot(readmission,aes(x=factor(Readmission.Status),y=logRiskScore)) + geom_boxplot() + facet_wrap(~fact
```





Trying again with Race and Gender.

```
ggplot(readmission,aes(Gender,fill=factor(Readmission.Status))) + geom_bar(position = "fill") +
  facet_wrap(~Race,ncol=2,scales="free")+scale_y_continuous()+ylab("percent")
```



## Task 5

Before fitting a GLM, I will split the data into training and test sets.

```
#Create train and test sets
```

```
library(caret)
```

```
## Loading required package: lattice
```

```
## Warning: replacing previous import by 'rlang::!!' when loading 'recipes'
```

```
## Warning: replacing previous import by 'rlang::as_character' when loading  
## 'recipes'
```

```
## Warning: replacing previous import by 'rlang::call2' when loading 'recipes'
```

```
## Warning: replacing previous import by 'rlang::exec' when loading 'recipes'
```

```
## Warning: replacing previous import by 'rlang::expr' when loading 'recipes'
```

```
## Warning: replacing previous import by 'rlang::f_lhs' when loading 'recipes'
```

```
## Warning: replacing previous import by 'rlang::f_rhs' when loading 'recipes'
```

```
## Warning: replacing previous import by 'rlang::is_empty' when loading
```

```
## 'recipes'
```

```
## Warning: replacing previous import by 'rlang::is_quosure' when loading
```

```
## 'recipes'
```

```
## Warning: replacing previous import by 'rlang::na_dbl' when loading
```

```

## 'recipes'
## Warning: replacing previous import by 'rlang::names2' when loading
## 'recipes'
## Warning: replacing previous import by 'rlang::quo' when loading 'recipes'
## Warning: replacing previous import by 'rlang::quo_get_expr' when loading
## 'recipes'
## Warning: replacing previous import by 'rlang::quo_squash' when loading
## 'recipes'
## Warning: replacing previous import by 'rlang::quo_text' when loading
## 'recipes'
## Warning: replacing previous import by 'rlang::quos' when loading 'recipes'
## Warning: replacing previous import by 'rlang::sym' when loading 'recipes'
## Warning: replacing previous import by 'rlang::syms' when loading 'recipes'
## Warning: replacing previous import by 'tibble::tibble' when loading
## 'recipes'
## Warning: replacing previous import by 'plyr::ddply' when loading 'caret'
## Warning: replacing previous import by 'recipes::all_outcomes' when loading
## 'caret'
## Warning: replacing previous import by 'recipes::all_predictors' when
## loading 'caret'
## Warning: replacing previous import by 'recipes::bake' when loading 'caret'
## Warning: replacing previous import by 'recipes::has_role' when loading
## 'caret'
## Warning: replacing previous import by 'recipes::juice' when loading 'caret'
## Warning: replacing previous import by 'recipes::prep' when loading 'caret'
set.seed(4321)
partition <- createDataPartition(readmission[,1], list = FALSE, p = .75) #The partition will stratify u
train <- readmission[partition, ]
test <- readmission[-partition, ]

print("TRAIN")

## [1] "TRAIN"
mean(train$Readmission.Status)

## [1] 0.1269518
print("TEST")

## [1] "TEST"
mean(test$Readmission.Status)

## [1] 0.1228585

```

I will now run a glm using the binomial distribution and logit link and add the desired interaction.

```

library(pROC)

## Type 'citation("pROC")' for a citation.
##
## Attaching package: 'pROC'
##
## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
glmlogit <- glm(Readmission.Status ~ . + Gender*Race, data=train, family = binomial(link="logit"))
summary(glmlogit)

##
## Call:
## glm(formula = Readmission.Status ~ . + Gender * Race, family = binomial(link = "logit"),
##     data = train)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.3460  -0.5643  -0.3911  -0.2544   3.0541
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    -2.603997   0.136923  -19.018 < 2e-16 ***
## GenderM         -0.021739   0.031186   -0.697  0.48575
## RaceBlack        0.050490   0.059038    0.855  0.39244
## RaceHispanic     0.142477   0.129136    1.103  0.26989
## RaceOthers      -0.127669   0.113611   -1.124  0.26112
## ER              -0.005826   0.017218   -0.338  0.73508
## Age             -0.007438   0.001617   -4.598 4.26e-06 ***
## logLOS          0.065528   0.020390    3.214  0.00131 **
## logRiskscore     1.330490   0.023717   56.100 < 2e-16 ***
## Under65         -0.064840   0.057945   -1.119  0.26314
## DRGMed.NoC      -0.040421   0.042587   -0.949  0.34254
## DRGOtherMED      0.125416   0.054738    2.291  0.02195 *
## DRGOtherSURG     0.103201   0.065984    1.564  0.11781
## DRGSurg.C        0.007461   0.039946    0.187  0.85184
## DRGSurg.NoC     -0.010876   0.043732   -0.249  0.80360
## DRGUNGROUP       0.134489   0.139704    0.963  0.33571
## GenderM:RaceBlack 0.003736   0.090470    0.041  0.96706
## GenderM:RaceHispanic -0.321541  0.213049   -1.509  0.13124
## GenderM:RaceOthers 0.237410   0.158452    1.498  0.13405
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 38117  on 50081  degrees of freedom
## Residual deviance: 33955  on 50063  degrees of freedom
## AIC: 33993
##
## Number of Fisher Scoring iterations: 5

```

```

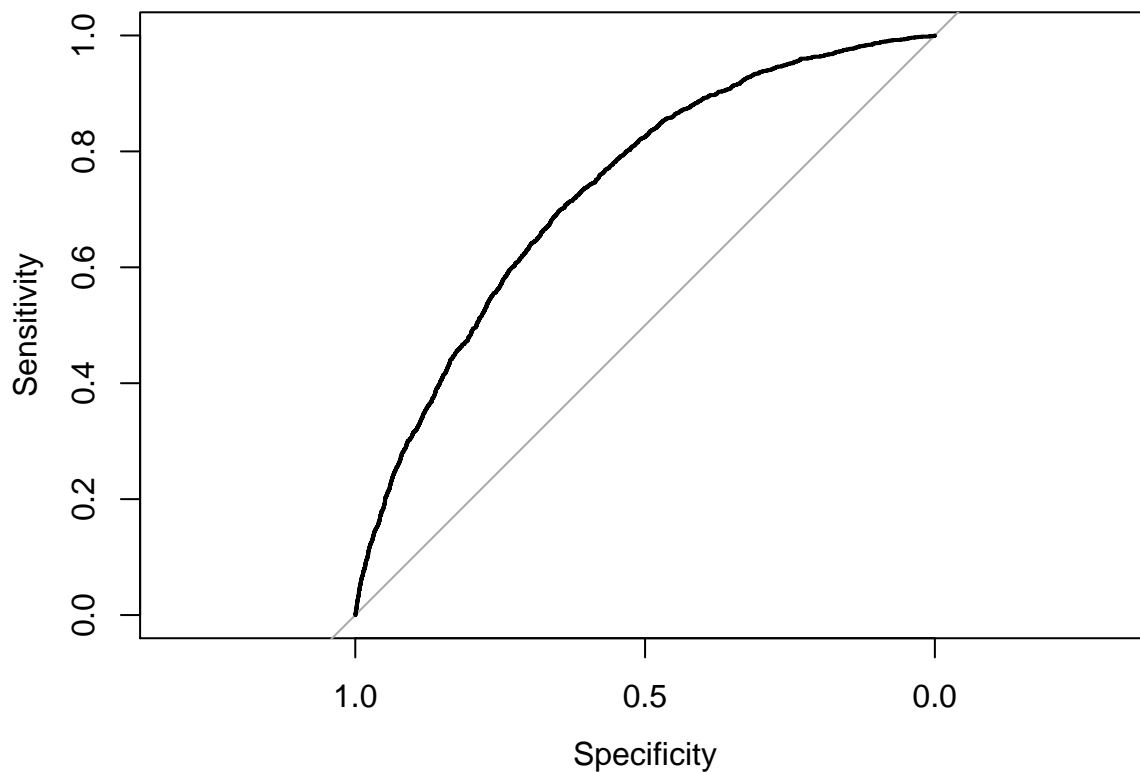
predslogit <- predict(glmlogit,newdat=test,type="response")

roclogit <- roc(test$Readmission.Status,predslogit)

## Setting levels: control = 0, case = 1
## Setting direction: controls < cases
confusionMatrix(factor(1*(predslogit>.5)),factor(test$Readmission.Status))

## Confusion Matrix and Statistics
##
##           Reference
## Prediction      0      1
##           0 14630  2042
##           1     13     9
##
##           Accuracy : 0.8769
##           95% CI : (0.8718, 0.8818)
##       No Information Rate : 0.8771
##       P-Value [Acc > NIR] : 0.5434
##
##           Kappa : 0.0061
##
##  Mcnemar's Test P-Value : <2e-16
##
##           Sensitivity : 0.999112
##           Specificity : 0.004388
##       Pos Pred Value : 0.877519
##       Neg Pred Value : 0.409091
##           Prevalence : 0.877141
##       Detection Rate : 0.876363
##   Detection Prevalence : 0.998682
##       Balanced Accuracy : 0.501750
##
##       'Positive' Class : 0
##
plot(roclogit)

```



```
auc(roclogit)
```

```
## Area under the curve: 0.7324
```

The same code is now run with the probit link.

```
library(pROC)
```

```
glmprobit <- glm(Readmission.Status ~ . + Gender*Race, data=train, family = binomial(link="probit"))
```

```
summary(glmprobit)
```

```
##
```

```
## Call:
```

```
## glm(formula = Readmission.Status ~ . + Gender * Race, family = binomial(link = "probit"),  
##      data = train)
```

```
##
```

```
## Deviance Residuals:
```

```
##      Min       1Q   Median       3Q      Max  
## -1.2689  -0.5756  -0.3944  -0.2384   3.2478
```

```
##
```

```
## Coefficients:
```

```
##              Estimate Std. Error z value Pr(>|z|)  
## (Intercept)    -1.4371865   0.0741325  -19.387  < 2e-16 ***  
## GenderM        -0.0138744   0.0168726   -0.822   0.41090  
## RaceBlack       0.0237364   0.0321619    0.738   0.46050  
## RaceHispanic    0.0763654   0.0711823    1.073   0.28335  
## RaceOthers     -0.0712955   0.0606013   -1.176   0.23941  
## ER             -0.0025486   0.0093214   -0.273   0.78454  
## Age            -0.0045131   0.0008786   -5.137  2.79e-07 ***  
## logLOS          0.0343572   0.0112270    3.060   0.00221 **  
## logRiskscore    0.7121243   0.0124659   57.126  < 2e-16 ***
```

```

## Under65          -0.0408896  0.0314227  -1.301  0.19316
## DRGMed.NoC       -0.0216338  0.0230263  -0.940  0.34746
## DRGOtherMED      0.0679331  0.0298657   2.275  0.02293 *
## DRGOtherSURG     0.0540437  0.0362048   1.493  0.13551
## DRGSurg.C        0.0086575  0.0215849   0.401  0.68835
## DRGSurg.NoC      -0.0045170  0.0236140  -0.191  0.84830
## DRGUNGROUP       0.0704562  0.0778915   0.905  0.36571
## GenderM:RaceBlack 0.0159955  0.0493390   0.324  0.74579
## GenderM:RaceHispanic -0.1665655  0.1147644  -1.451  0.14668
## GenderM:RaceOthers 0.1305341  0.0854834   1.527  0.12676
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
## Null deviance: 38117 on 50081 degrees of freedom
## Residual deviance: 33928 on 50063 degrees of freedom
## AIC: 33966
##
## Number of Fisher Scoring iterations: 5
predsprobit <- predict(glmprobit,newdat=test,type="response")
rocprobit <- roc(test$Readmission.Status,predsprobit)

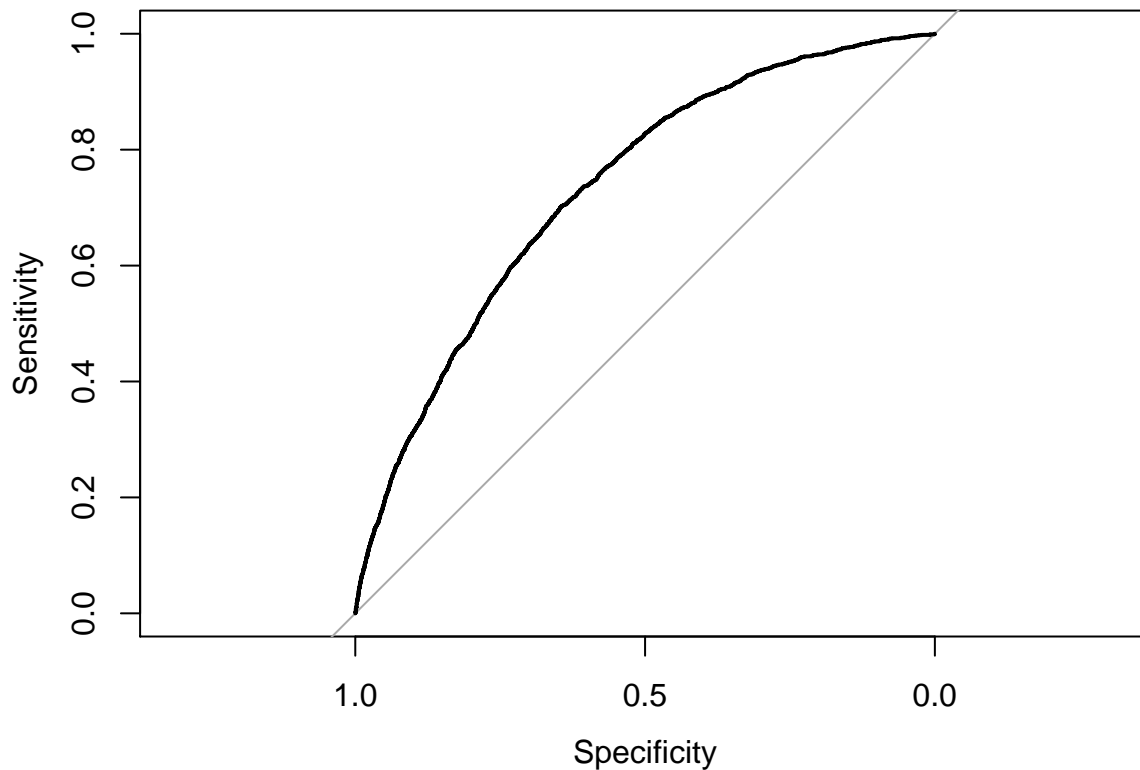
## Setting levels: control = 0, case = 1
## Setting direction: controls < cases
confusionMatrix(factor(1*(predsprobit>.5)),factor(test$Readmission.Status))

## Confusion Matrix and Statistics
##
##           Reference
## Prediction    0    1
##           0 14638 2048
##           1     5    3
##
##               Accuracy : 0.877
##               95% CI : (0.8719, 0.882)
##           No Information Rate : 0.8771
##           P-Value [Acc > NIR] : 0.5247
##
##               Kappa : 0.002
##
## Mcnemar's Test P-Value : <2e-16
##
##           Sensitivity : 0.999659
##           Specificity : 0.001463
##           Pos Pred Value : 0.877262
##           Neg Pred Value : 0.375000
##           Prevalence : 0.877141
##           Detection Rate : 0.876842
##           Detection Prevalence : 0.999521
##           Balanced Accuracy : 0.500561
##

```

```
##      'Positive' Class : 0
##
```

```
plot(rocprobit)
```



```
auc(rocprobit)
```

```
## Area under the curve: 0.7324
```

## Task 6

A new dataframe is created that drops logLOS and Age and adds LOS\_Age\_Clust. It must then be partitioned.

```
readmission.cluster <- readmission
readmission.cluster$Age <- NULL
readmission.cluster$logLOS <- NULL
readmission.cluster$LOS_Age_Clust <- LOS_Age_Clust
table(readmission.cluster$LOS_Age_Clust)
```

```
##
##      1      2      3      4      5
## 14231 18346  7642 15798 10759
```

```
train.cluster <- readmission.cluster[partition, ]
test.cluster <- readmission.cluster[-partition, ]
```

This code reruns the probit regression using the cluster variable.

```
library(pROC)
glmprobit.clust <- glm(Readmission.Status ~ . + Gender*Race, data=train.cluster, family = binomial(link=
```



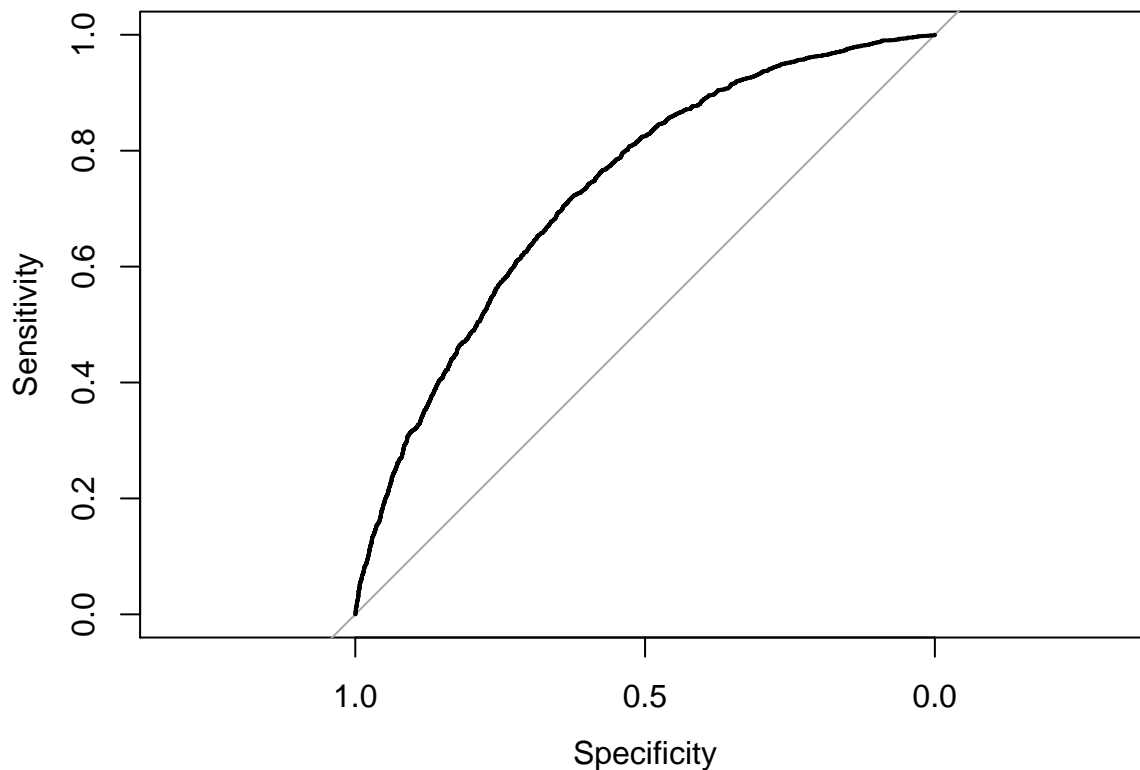
```
summary(glmprobit.clust)
```

```
##
## Call:
## glm(formula = Readmission.Status ~ . + Gender * Race, family = binomial(link = "probit"),
##      data = train.cluster)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.2356  -0.5765  -0.3957  -0.2391   3.2764
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    -1.765042   0.025669  -68.761 < 2e-16 ***
## GenderM         -0.009304   0.016857   -0.552  0.58098
## RaceBlack        0.029914   0.032122    0.931  0.35172
## RaceHispanic     0.081600   0.071135    1.147  0.25133
## RaceOthers      -0.064829   0.060541   -1.071  0.28425
## ER              -0.002689   0.009317   -0.289  0.77289
## logRiskscore     0.708474   0.012368   57.281 < 2e-16 ***
## Under65          0.078926   0.033482    2.357  0.01841 *
## DRGMed.NoC       -0.021876   0.023018   -0.950  0.34192
## DRGOtherMED       0.069645   0.029842    2.334  0.01960 *
## DRGOtherSURG      0.056327   0.036179    1.557  0.11950
## DRGSurg.C         0.007811   0.021578    0.362  0.71735
## DRGSurg.NoC      -0.004139   0.023606   -0.175  0.86083
## DRGUNGROUP        0.072636   0.077946    0.932  0.35140
## LOS_Age_Clust2    0.051873   0.023269    2.229  0.02579 *
## LOS_Age_Clust3    0.029991   0.042493    0.706  0.48031
## LOS_Age_Clust4    0.003972   0.023923    0.166  0.86813
## LOS_Age_Clust5    0.081151   0.025264    3.212  0.00132 **
## GenderM:RaceBlack  0.014201   0.049321    0.288  0.77340
## GenderM:RaceHispanic -0.164464  0.114648   -1.435  0.15143
## GenderM:RaceOthers 0.128279   0.085418    1.502  0.13315
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 38117  on 50081  degrees of freedom
## Residual deviance: 33948  on 50061  degrees of freedom
## AIC: 33990
##
## Number of Fisher Scoring iterations: 5
predsprobit.clust <- predict(glmprobit.clust,newdat=test.cluster,type="response")
rocprobit.clust <- roc(test.cluster$Readmission.Status,predsprobit.clust)

## Setting levels: control = 0, case = 1
## Setting direction: controls < cases
confusionMatrix(factor(1*(predsprobit.clust>.5)),factor(test.cluster$Readmission.Status))
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    0    1
##           0 14639 2049
##           1     4    2
##
##           Accuracy : 0.877
##           95% CI : (0.8719, 0.882)
##       No Information Rate : 0.8771
##       P-Value [Acc > NIR] : 0.5247
##
##           Kappa : 0.0012
##
##  McNemar's Test P-Value : <2e-16
##
##           Sensitivity : 0.9997268
##           Specificity : 0.0009751
##       Pos Pred Value : 0.8772172
##       Neg Pred Value : 0.3333333
##           Prevalence : 0.8771415
##       Detection Rate : 0.8769019
##   Detection Prevalence : 0.9996406
##       Balanced Accuracy : 0.5003510
##
##       'Positive' Class : 0
##
```

```
plot(rocprobit.clust)
```



```
auc(rocprobit.clust)
```

```
## Area under the curve: 0.7321
```

## Task 7

The first step is to binarize the factor levels that have more than two levels. `fullRank` is set to `FALSE` so that all levels get binarized. The interaction variable is also a factor variable and must be binarized as well. In each case the base level will then need to be deleted. I do it this way because if `fullRank` is `TRUE` I can't be sure about which level becomes the base.

```
#Add the interaction variable to the data frame
```

```
readmission$RaceGender <- factor(paste0(readmission$Race,readmission$Gender))
summary(readmission$RaceGender)
```

```
##      BlackF      BlackM HispanicF HispanicM      OthersF      OthersM      WhiteF
##      4157      2940      751      535      1206      1067      31891
##      WhiteM
##      24229
```

```
factor_names <- c("Race","DRG","RaceGender")
factor_vars <- readmission[,factor_names]
for (var in factor_names) {
  factor_vars[, var] <- as.character(factor_vars[, var])
}
```

```
binarizer <- caret::dummyVars(paste("~", paste(factor_names, collapse = "+")) , data = factor_vars, fullRank = FALSE)
binarized_vars <- data.frame(predict(binarizer, newdata = factor_vars))
head(binarized_vars)
```

```
##      RaceBlack RaceHispanic RaceOthers RaceWhite DRGMed.C DRGMed.NoC
## 1           0           0           0           1           0           0
## 2           0           0           0           1           0           0
## 3           0           0           0           1           0           0
## 4           0           0           0           1           0           1
## 5           0           0           0           1           0           1
## 6           0           0           0           1           0           0
##      DRGOtherMED DRGOtherSURG DRGSurg.C DRGSurg.NoC DRGUNGROUP
## 1             1             0             0             0             0
## 2             0             0             0             1             0
## 3             0             0             0             1             0
## 4             0             0             0             0             0
## 5             0             0             0             0             0
## 6             0             0             1             0             0
##      RaceGenderBlackF RaceGenderBlackM RaceGenderHispanicF
## 1                   0                   0                   0
## 2                   0                   0                   0
## 3                   0                   0                   0
## 4                   0                   0                   0
## 5                   0                   0                   0
## 6                   0                   0                   0
##      RaceGenderHispanicM RaceGenderOthersF RaceGenderOthersM RaceGenderWhiteF
## 1                   0                   0                   0                   0
```

```
## 2      0      0      0      0
## 3      0      0      0      0
## 4      0      0      0      0
## 5      0      0      0      0
## 6      0      0      0      0
## RaceGenderWhiteM
## 1      1
## 2      1
## 3      1
## 4      1
## 5      1
## 6      1
```

Now delete the three base variables.

```
binarized_vars$RaceWhite <- NULL
binarized_vars$DRGMed.C <- NULL
binarized_vars$RaceGenderWhiteF <- NULL
head(binarized_vars)
```

```
## RaceBlack RaceHispanic RaceOthers DRGMed.NoC DRGOtherMED DRGOtherSURG
## 1      0      0      0      0      1      0
## 2      0      0      0      0      0      0
## 3      0      0      0      0      0      0
## 4      0      0      0      1      0      0
## 5      0      0      0      1      0      0
## 6      0      0      0      0      0      0
## DRGSurg.C DRGSurg.NoC DRGUNGROUP RaceGenderBlackF RaceGenderBlackM
## 1      0      0      0      0      0
## 2      0      1      0      0      0
## 3      0      1      0      0      0
## 4      0      0      0      0      0
## 5      0      0      0      0      0
## 6      1      0      0      0      0
## RaceGenderHispanicF RaceGenderHispanicM RaceGenderOthersF
## 1      0      0      0
## 2      0      0      0
## 3      0      0      0
## 4      0      0      0
## 5      0      0      0
## 6      0      0      0
## RaceGenderOthersM RaceGenderWhiteM
## 1      0      1
## 2      0      1
## 3      0      1
## 4      0      1
## 5      0      1
## 6      0      1
```

I now attach the binarized variables and remove the three original factor variables. A new dataframe is created so the old one is preserved.

```
readmission.bin <- cbind(readmission,binarized_vars)
readmission.bin$DRG <- NULL
readmission.bin$Race <- NULL
readmission.bin$RaceGender <- NULL
```

```
summary(readmission.bin)
```

```
## Readmission.Status Gender          ER          Age
## Min.   :0.0000    F:38005   Min.   :0.0000   Min.   : 24.00
## 1st Qu.:0.0000    M:28771   1st Qu.:0.0000   1st Qu.: 67.00
## Median :0.0000                Median :0.0000   Median : 75.00
## Mean   :0.1259                Mean   :0.5083   Mean   : 73.64
## 3rd Qu.:0.0000                3rd Qu.:1.0000   3rd Qu.: 83.00
## Max.   :1.0000                Max.   :9.0000   Max.   :101.00
##      logLOS      logRiskScore      Under65      RaceBlack
## Min.   :0.000   Min.   :-2.5383   Min.   :0.0000   Min.   :0.0000
## 1st Qu.:1.099   1st Qu.: 0.1017   1st Qu.:0.0000   1st Qu.:0.0000
## Median :1.609   Median : 0.6238   Median :0.0000   Median :0.0000
## Mean   :1.653   Mean   : 0.6000   Mean   :0.1684   Mean   :0.1063
## 3rd Qu.:2.079   3rd Qu.: 1.1547   3rd Qu.:0.0000   3rd Qu.:0.0000
## Max.   :3.584   Max.   : 2.5102   Max.   :1.0000   Max.   :1.0000
##      RaceHispanic      RaceOthers      DRGMed.NoC      DRGOtherMED
## Min.   :0.00000   Min.   :0.00000   Min.   :0.0000   Min.   :0.00000
## 1st Qu.:0.00000   1st Qu.:0.00000   1st Qu.:0.0000   1st Qu.:0.00000
## Median :0.00000   Median :0.00000   Median :0.0000   Median :0.00000
## Mean   :0.01926   Mean   :0.03404   Mean   :0.1843   Mean   :0.08022
## 3rd Qu.:0.00000   3rd Qu.:0.00000   3rd Qu.:0.0000   3rd Qu.:0.00000
## Max.   :1.00000   Max.   :1.00000   Max.   :1.0000   Max.   :1.00000
##      DRGOtherSURG      DRGSurg.C      DRGSurg.NoC      DRGUNGROUP
## Min.   :0.00000   Min.   :0.0000   Min.   :0.000   Min.   :0.000000
## 1st Qu.:0.00000   1st Qu.:0.0000   1st Qu.:0.000   1st Qu.:0.000000
## Median :0.00000   Median :0.0000   Median :0.000   Median :0.000000
## Mean   :0.05128   Mean   :0.2316   Mean   :0.173   Mean   :0.008446
## 3rd Qu.:0.00000   3rd Qu.:0.0000   3rd Qu.:0.000   3rd Qu.:0.000000
## Max.   :1.00000   Max.   :1.0000   Max.   :1.000   Max.   :1.000000
##      RaceGenderBlackF      RaceGenderBlackM      RaceGenderHispanicF
## Min.   :0.00000   Min.   :0.00000   Min.   :0.00000
## 1st Qu.:0.00000   1st Qu.:0.00000   1st Qu.:0.00000
## Median :0.00000   Median :0.00000   Median :0.00000
## Mean   :0.06225   Mean   :0.04403   Mean   :0.01125
## 3rd Qu.:0.00000   3rd Qu.:0.00000   3rd Qu.:0.00000
## Max.   :1.00000   Max.   :1.00000   Max.   :1.00000
##      RaceGenderHispanicM      RaceGenderOthersF      RaceGenderOthersM      RaceGenderWhiteM
## Min.   :0.000000   Min.   :0.00000   Min.   :0.00000   Min.   :0.00000
## 1st Qu.:0.000000   1st Qu.:0.00000   1st Qu.:0.00000   1st Qu.:0.00000
## Median :0.000000   Median :0.00000   Median :0.00000   Median :0.00000
## Mean   :0.008012   Mean   :0.01806   Mean   :0.01598   Mean   :0.3628
## 3rd Qu.:0.000000   3rd Qu.:0.00000   3rd Qu.:0.00000   3rd Qu.:1.0000
## Max.   :1.000000   Max.   :1.00000   Max.   :1.00000   Max.   :1.0000
```

I need to again split the data into train and test sets.

```
train.bin <- readmission.bin[partition, ]
test.bin <- readmission.bin[-partition, ]
```

I next run the GLM on the binarized data. In doing so, I recognized that the interaction variable created combinations that were redundant. I need to remove all the interactions with male and then re-partition the data.

```
readmission.bin$RaceGenderOthersM <- NULL
readmission.bin$RaceGenderWhiteM <- NULL
readmission.bin$RaceGenderHispanicM <- NULL
readmission.bin$RaceGenderBlackM <- NULL
summary(readmission.bin)
```

```
## Readmission.Status Gender ER Age
## Min. :0.0000 F:38005 Min. :0.0000 Min. : 24.00
## 1st Qu.:0.0000 M:28771 1st Qu.:0.0000 1st Qu.: 67.00
## Median :0.0000 Median :0.0000 Median : 75.00
## Mean :0.1259 Mean :0.5083 Mean : 73.64
## 3rd Qu.:0.0000 3rd Qu.:1.0000 3rd Qu.: 83.00
## Max. :1.0000 Max. :9.0000 Max. :101.00
## logLOS logRiskscore Under65 RaceBlack
## Min. :0.000 Min. : -2.5383 Min. :0.0000 Min. : 0.0000
## 1st Qu.:1.099 1st Qu.: 0.1017 1st Qu.:0.0000 1st Qu.:0.0000
## Median :1.609 Median : 0.6238 Median :0.0000 Median :0.0000
## Mean :1.653 Mean : 0.6000 Mean :0.1684 Mean :0.1063
## 3rd Qu.:2.079 3rd Qu.: 1.1547 3rd Qu.:0.0000 3rd Qu.:0.0000
## Max. :3.584 Max. : 2.5102 Max. :1.0000 Max. :1.0000
## RaceHispanic RaceOthers DRGMed.NoC DRGOtherMED
## Min. :0.00000 Min. :0.00000 Min. :0.0000 Min. :0.00000
## 1st Qu.:0.00000 1st Qu.:0.00000 1st Qu.:0.0000 1st Qu.:0.00000
## Median :0.00000 Median :0.00000 Median :0.0000 Median :0.00000
## Mean :0.01926 Mean :0.03404 Mean :0.1843 Mean :0.08022
## 3rd Qu.:0.00000 3rd Qu.:0.00000 3rd Qu.:0.0000 3rd Qu.:0.00000
## Max. :1.00000 Max. :1.00000 Max. :1.0000 Max. :1.00000
## DRGOtherSURG DRGSurg.C DRGSurg.NoC DRGUNGROUP
## Min. :0.00000 Min. :0.0000 Min. :0.000 Min. :0.000000
## 1st Qu.:0.00000 1st Qu.:0.0000 1st Qu.:0.000 1st Qu.:0.000000
## Median :0.00000 Median :0.0000 Median :0.000 Median :0.000000
## Mean :0.05128 Mean :0.2316 Mean :0.173 Mean :0.008446
## 3rd Qu.:0.00000 3rd Qu.:0.0000 3rd Qu.:0.000 3rd Qu.:0.000000
## Max. :1.00000 Max. :1.0000 Max. :1.000 Max. :1.000000
## RaceGenderBlackF RaceGenderHispanicF RaceGenderOthersF
## Min. :0.00000 Min. :0.00000 Min. :0.00000
## 1st Qu.:0.00000 1st Qu.:0.00000 1st Qu.:0.00000
## Median :0.00000 Median :0.00000 Median :0.00000
## Mean :0.06225 Mean :0.01125 Mean :0.01806
## 3rd Qu.:0.00000 3rd Qu.:0.00000 3rd Qu.:0.00000
## Max. :1.00000 Max. :1.00000 Max. :1.00000
```

```
train.bin <- readmission.bin[partition, ]
test.bin <- readmission.bin[-partition, ]
```

```
glmprobit <- glm(Readmission.Status ~ . , data=train.bin, family = binomial(link="probit"))
summary(glmprobit)
```

```
##
## Call:
## glm(formula = Readmission.Status ~ ., family = binomial(link = "probit"),
## data = train.bin)
##
## Deviance Residuals:
```

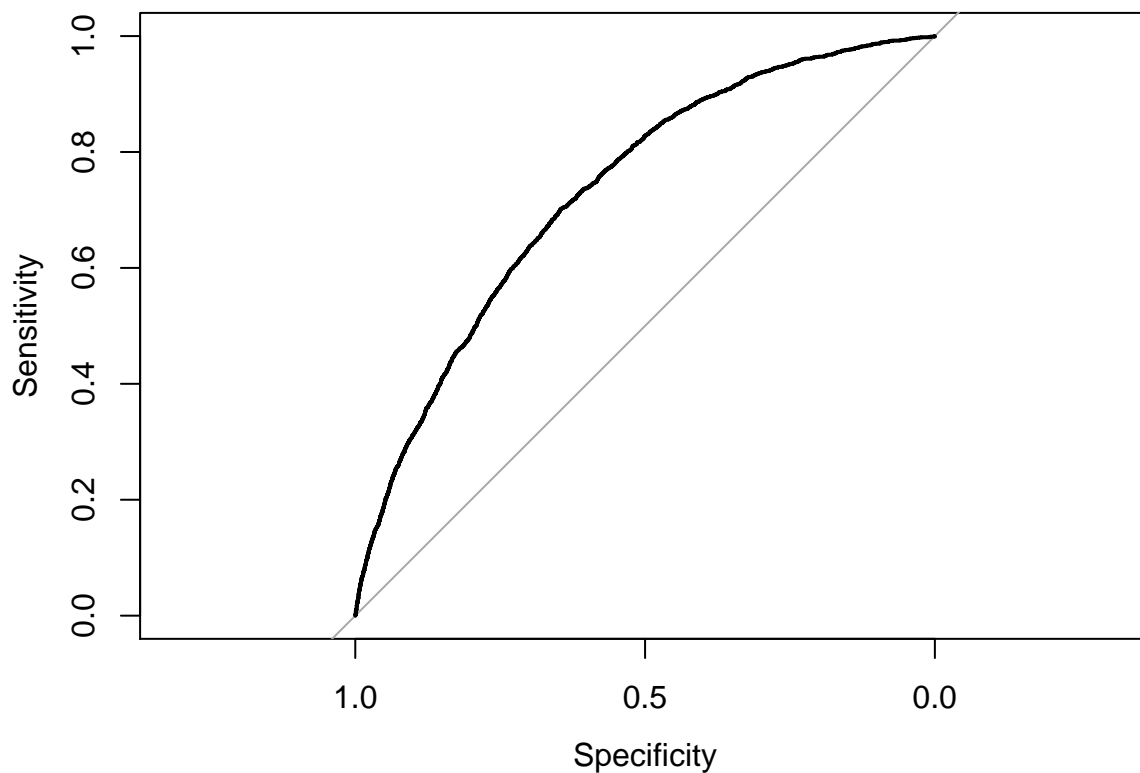
```
##      Min      1Q   Median      3Q      Max
## -1.2689 -0.5756 -0.3944 -0.2384  3.2478
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    -1.4371865   0.0741325  -19.387 < 2e-16 ***
## GenderM        -0.0138744   0.0168726   -0.822  0.41090
## ER            -0.0025486   0.0093214   -0.273  0.78454
## Age           -0.0045131   0.0008786   -5.137 2.79e-07 ***
## logLOS         0.0343572   0.0112270    3.060  0.00221 **
## logRiskscore   0.7121243   0.0124659   57.126 < 2e-16 ***
## Under65       -0.0408896   0.0314227   -1.301  0.19316
## RaceBlack      0.0397319   0.0378371    1.050  0.29368
## RaceHispanic  -0.0902001   0.0901389   -1.001  0.31698
## RaceOthers     0.0592385   0.0603545    0.982  0.32634
## DRGMed.NoC    -0.0216338   0.0230263   -0.940  0.34746
## DRGOtherMED    0.0679331   0.0298657    2.275  0.02293 *
## DRGOtherSURG   0.0540437   0.0362048    1.493  0.13551
## DRGSurg.C      0.0086575   0.0215849    0.401  0.68835
## DRGSurg.NoC   -0.0045170   0.0236140   -0.191  0.84830
## DRGUNGROUP     0.0704562   0.0778915    0.905  0.36571
## RaceGenderBlackF -0.0159955  0.0493390   -0.324  0.74579
## RaceGenderHispanicF 0.1665655  0.1147644    1.451  0.14668
## RaceGenderOthersF -0.1305341  0.0854834   -1.527  0.12676
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 38117  on 50081  degrees of freedom
## Residual deviance: 33928  on 50063  degrees of freedom
## AIC: 33966
##
## Number of Fisher Scoring iterations: 5
predsprobit <- predict(glmprobit,newdat=test.bin,type="response")
rocprobit <- roc(test.bin$Readmission.Status,predsprobit)

## Setting levels: control = 0, case = 1
## Setting direction: controls < cases
confusionMatrix(factor(1*(predsprobit>.5)),factor(test.bin$Readmission.Status))

## Confusion Matrix and Statistics
##
##              Reference
## Prediction    0      1
##      0 14638  2048
##      1      5      3
##
##              Accuracy : 0.877
##              95% CI : (0.8719, 0.882)
##      No Information Rate : 0.8771
##      P-Value [Acc > NIR] : 0.5247
```

```
##
##           Kappa : 0.002
##
## Mcnemar's Test P-Value : <2e-16
##
##           Sensitivity : 0.999659
##           Specificity : 0.001463
##           Pos Pred Value : 0.877262
##           Neg Pred Value : 0.375000
##           Prevalence : 0.877141
##           Detection Rate : 0.876842
##           Detection Prevalence : 0.999521
##           Balanced Accuracy : 0.500561
##
##           'Positive' Class : 0
##
```

```
plot(rocprobit)
```



```
auc(rocprobit)
```

```
## Area under the curve: 0.7324
```

The next step is to run stepAIC on this model.

```
library(MASS)
stepAIC(glmprobit)
```

```
## Start:  AIC=33965.61
## Readmission.Status ~ Gender + ER + Age + logLOS + logRiskscore +
##      Under65 + RaceBlack + RaceHispanic + RaceOthers + DRGMed.NoC +
```



```

##      DRGOtherMED + DRGOtherSURG + DRGSurg.C + DRGSurg.NoC + DRGUNGROUP +
##      RaceGenderBlackF + RaceGenderHispanicF + RaceGenderOthersF
##
##              Df Deviance   AIC
## - DRGSurg.NoC      1    33928 33964
## - ER                1    33928 33964
## - RaceGenderBlackF  1    33928 33964
## - DRGSurg.C        1    33928 33964
## - Gender           1    33928 33964
## - DRGUNGROUP       1    33928 33964
## - DRGMed.NoC       1    33928 33964
## - RaceOthers       1    33929 33965
## - RaceHispanic     1    33929 33965
## - RaceBlack        1    33929 33965
## - Under65          1    33929 33965
## <none>              33928 33966
## - RaceGenderHispanicF 1    33930 33966
## - DRGOtherSURG     1    33930 33966
## - RaceGenderOthersF 1    33930 33966
## - DRGOtherMED      1    33933 33969
## - logLOS           1    33937 33973
## - Age              1    33954 33990
## - logRiskscore     1    37741 37777
##
## Step:  AIC=33963.65
## Readmission.Status ~ Gender + ER + Age + logLOS + logRiskscore +
##      Under65 + RaceBlack + RaceHispanic + RaceOthers + DRGMed.NoC +
##      DRGOtherMED + DRGOtherSURG + DRGSurg.C + DRGUNGROUP + RaceGenderBlackF +
##      RaceGenderHispanicF + RaceGenderOthersF
##
##              Df Deviance   AIC
## - ER                1    33928 33962
## - RaceGenderBlackF  1    33928 33962
## - DRGSurg.C        1    33928 33962
## - Gender           1    33928 33962
## - DRGUNGROUP       1    33929 33963
## - DRGMed.NoC       1    33929 33963
## - RaceOthers       1    33929 33963
## - RaceHispanic     1    33929 33963
## - RaceBlack        1    33929 33963
## - Under65          1    33929 33963
## <none>              33928 33964
## - RaceGenderHispanicF 1    33930 33964
## - RaceGenderOthersF  1    33930 33964
## - DRGOtherSURG     1    33930 33964
## - DRGOtherMED      1    33934 33968
## - logLOS           1    33937 33971
## - Age              1    33954 33988
## - logRiskscore     1    37742 37776
##
## Step:  AIC=33961.73
## Readmission.Status ~ Gender + Age + logLOS + logRiskscore + Under65 +
##      RaceBlack + RaceHispanic + RaceOthers + DRGMed.NoC + DRGOtherMED +
##      DRGOtherSURG + DRGSurg.C + DRGUNGROUP + RaceGenderBlackF +

```

```

##      RaceGenderHispanicF + RaceGenderOthersF
##
##      Df Deviance   AIC
## - RaceGenderBlackF      1    33928 33960
## - DRGSurg.C              1    33928 33960
## - Gender                 1    33928 33960
## - DRGUNGROUP            1    33929 33961
## - DRGMed.NoC            1    33929 33961
## - RaceOthers            1    33929 33961
## - RaceHispanic          1    33929 33961
## - RaceBlack             1    33929 33961
## - Under65               1    33929 33961
## <none>                   33928 33962
## - RaceGenderHispanicF   1    33930 33962
## - RaceGenderOthersF     1    33930 33962
## - DRGOtherSURG          1    33930 33962
## - DRGOtherMED           1    33934 33966
## - logLOS                 1    33937 33969
## - Age                    1    33954 33986
## - logRiskscore          1    37742 37774
##
## Step: AIC=33959.83
## Readmission.Status ~ Gender + Age + logLOS + logRiskscore + Under65 +
##      RaceBlack + RaceHispanic + RaceOthers + DRGMed.NoC + DRGOtherMED +
##      DRGOtherSURG + DRGSurg.C + DRGUNGROUP + RaceGenderHispanicF +
##      RaceGenderOthersF
##
##      Df Deviance   AIC
## - DRGSurg.C              1    33928 33958
## - Gender                 1    33928 33958
## - DRGUNGROUP            1    33929 33959
## - DRGMed.NoC            1    33929 33959
## - RaceOthers            1    33929 33959
## - RaceHispanic          1    33929 33959
## - RaceBlack             1    33929 33959
## - Under65               1    33930 33960
## <none>                   33928 33960
## - RaceGenderHispanicF   1    33930 33960
## - RaceGenderOthersF     1    33930 33960
## - DRGOtherSURG          1    33930 33960
## - DRGOtherMED           1    33934 33964
## - logLOS                 1    33937 33967
## - Age                    1    33954 33984
## - logRiskscore          1    37742 37772
##
## Step: AIC=33958.11
## Readmission.Status ~ Gender + Age + logLOS + logRiskscore + Under65 +
##      RaceBlack + RaceHispanic + RaceOthers + DRGMed.NoC + DRGOtherMED +
##      DRGOtherSURG + DRGUNGROUP + RaceGenderHispanicF + RaceGenderOthersF
##
##      Df Deviance   AIC
## - Gender                 1    33929 33957
## - DRGUNGROUP            1    33929 33957
## - RaceOthers            1    33929 33957

```

```

## - RaceHispanic          1    33929 33957
## - DRGMed.NoC            1    33929 33957
## - RaceBlack             1    33930 33958
## - Under65               1    33930 33958
## <none>                  33928 33958
## - RaceGenderHispanicF   1    33930 33958
## - RaceGenderOthersF     1    33930 33958
## - DRGOtherSURG          1    33930 33958
## - DRGOtherMED           1    33934 33962
## - logLOS                1    33937 33965
## - Age                   1    33954 33982
## - logRiskscore          1    37742 37770
##
## Step:  AIC=33956.68
## Readmission.Status ~ Age + logLOS + logRiskscore + Under65 +
##      RaceBlack + RaceHispanic + RaceOthers + DRGMed.NoC + DRGOtherMED +
##      DRGOtherSURG + DRGUNGROUP + RaceGenderHispanicF + RaceGenderOthersF
##
##              Df Deviance   AIC
## - RaceOthers          1    33929 33955
## - DRGUNGROUP           1    33929 33955
## - RaceHispanic         1    33930 33956
## - DRGMed.NoC           1    33930 33956
## - RaceBlack            1    33930 33956
## - Under65              1    33930 33956
## - RaceGenderOthersF    1    33931 33957
## <none>                 33929 33957
## - DRGOtherSURG         1    33931 33957
## - RaceGenderHispanicF  1    33931 33957
## - DRGOtherMED          1    33934 33960
## - logLOS               1    33938 33964
## - Age                  1    33955 33981
## - logRiskscore         1    37743 37769
##
## Step:  AIC=33955.42
## Readmission.Status ~ Age + logLOS + logRiskscore + Under65 +
##      RaceBlack + RaceHispanic + DRGMed.NoC + DRGOtherMED + DRGOtherSURG +
##      DRGUNGROUP + RaceGenderHispanicF + RaceGenderOthersF
##
##              Df Deviance   AIC
## - DRGUNGROUP           1    33930 33954
## - RaceGenderOthersF     1    33931 33955
## - RaceHispanic          1    33931 33955
## - DRGMed.NoC            1    33931 33955
## - RaceBlack             1    33931 33955
## - Under65               1    33931 33955
## <none>                  33929 33955
## - DRGOtherSURG         1    33932 33956
## - RaceGenderHispanicF  1    33932 33956
## - DRGOtherMED           1    33935 33959
## - logLOS                1    33939 33963
## - Age                   1    33956 33980
## - logRiskscore          1    37743 37767
##

```

```

## Step: AIC=33954.19
## Readmission.Status ~ Age + logLOS + logRiskscore + Under65 +
##     RaceBlack + RaceHispanic + DRGMed.NoC + DRGOtherMED + DRGOtherSURG +
##     RaceGenderHispanicF + RaceGenderOthersF
##
##           Df Deviance   AIC
## - RaceGenderOthersF    1    33931 33953
## - RaceHispanic         1    33931 33953
## - RaceBlack            1    33932 33954
## - DRGMed.NoC           1    33932 33954
## - Under65              1    33932 33954
## <none>                  1    33930 33954
## - DRGOtherSURG         1    33932 33954
## - RaceGenderHispanicF  1    33933 33955
## - DRGOtherMED          1    33936 33958
## - logLOS               1    33940 33962
## - Age                  1    33956 33978
## - logRiskscore         1    37747 37769
##
## Step: AIC=33953.39
## Readmission.Status ~ Age + logLOS + logRiskscore + Under65 +
##     RaceBlack + RaceHispanic + DRGMed.NoC + DRGOtherMED + DRGOtherSURG +
##     RaceGenderHispanicF
##
##           Df Deviance   AIC
## - RaceHispanic         1    33933 33953
## - DRGMed.NoC           1    33933 33953
## - RaceBlack            1    33933 33953
## - Under65              1    33933 33953
## <none>                  1    33931 33953
## - DRGOtherSURG         1    33934 33954
## - RaceGenderHispanicF  1    33934 33954
## - DRGOtherMED          1    33937 33957
## - logLOS               1    33941 33961
## - Age                  1    33957 33977
## - logRiskscore         1    37749 37769
##
## Step: AIC=33952.59
## Readmission.Status ~ Age + logLOS + logRiskscore + Under65 +
##     RaceBlack + DRGMed.NoC + DRGOtherMED + DRGOtherSURG + RaceGenderHispanicF
##
##           Df Deviance   AIC
## - RaceGenderHispanicF  1    33934 33952
## - DRGMed.NoC           1    33934 33952
## - Under65              1    33934 33952
## - RaceBlack            1    33934 33952
## <none>                  1    33933 33953
## - DRGOtherSURG         1    33935 33953
## - DRGOtherMED          1    33938 33956
## - logLOS               1    33942 33960
## - Age                  1    33958 33976
## - logRiskscore         1    37751 37769
##
## Step: AIC=33951.95

```

```

## Readmission.Status ~ Age + logLOS + logRiskscore + Under65 +
##   RaceBlack + DRGMed.NoC + DRGOtherMED + DRGOtherSURG
##
##           Df Deviance   AIC
## - DRGMed.NoC      1    33935 33951
## - RaceBlack       1    33936 33952
## - Under65         1    33936 33952
## <none>              33934 33952
## - DRGOtherSURG    1    33936 33952
## - DRGOtherMED     1    33940 33956
## - logLOS          1    33943 33959
## - Age             1    33960 33976
## - logRiskscore    1    37754 37770
##
## Step:  AIC=33951.46
## Readmission.Status ~ Age + logLOS + logRiskscore + Under65 +
##   RaceBlack + DRGOtherMED + DRGOtherSURG
##
##           Df Deviance   AIC
## - RaceBlack       1    33937 33951
## - Under65         1    33937 33951
## <none>              33935 33951
## - DRGOtherSURG    1    33938 33952
## - DRGOtherMED     1    33942 33956
## - logLOS          1    33945 33959
## - Age             1    33961 33975
## - logRiskscore    1    37754 37768
##
## Step:  AIC=33951.02
## Readmission.Status ~ Age + logLOS + logRiskscore + Under65 +
##   DRGOtherMED + DRGOtherSURG
##
##           Df Deviance   AIC
## - Under65         1    33938 33950
## <none>              33937 33951
## - DRGOtherSURG    1    33940 33952
## - DRGOtherMED     1    33944 33956
## - logLOS          1    33946 33958
## - Age             1    33964 33976
## - logRiskscore    1    37759 37771
##
## Step:  AIC=33950.44
## Readmission.Status ~ Age + logLOS + logRiskscore + DRGOtherMED +
##   DRGOtherSURG
##
##           Df Deviance   AIC
## <none>              33938 33950
## - DRGOtherSURG    1    33941 33951
## - DRGOtherMED     1    33945 33955
## - logLOS          1    33948 33958
## - Age             1    33981 33991
## - logRiskscore    1    37765 37775
##

```

```
## Call: glm(formula = Readmission.Status ~ Age + logLOS + logRiskscore +
##   DRGOtherMED + DRGOtherSURG, family = binomial(link = "probit"),
##   data = train.bin)
##
## Coefficients:
## (Intercept)      Age      logLOS logRiskscore  DRGOtherMED
##   -1.50930    -0.00371    0.03436    0.71158    0.07107
## DRGOtherSURG
##    0.05586
##
## Degrees of Freedom: 50081 Total (i.e. Null); 50076 Residual
## Null Deviance:      38120
## Residual Deviance: 33940    AIC: 33950
```

The probit model is now run using the five surviving variables.

```
glmprobit <- glm(Readmission.Status ~ logLOS + Age + logRiskscore + DRGOtherSURG + DRGOtherMED, data=train.bin)
summary(glmprobit)
```

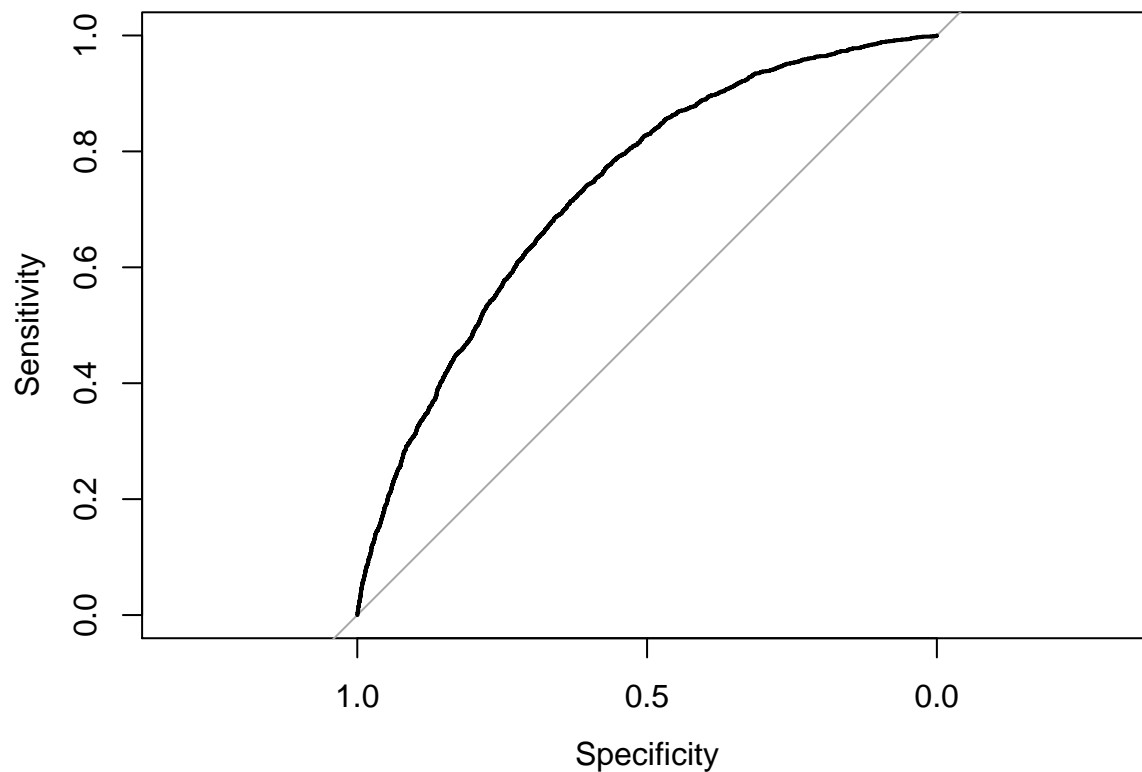
```
##
## Call:
## glm(formula = Readmission.Status ~ logLOS + Age + logRiskscore +
##   DRGOtherSURG + DRGOtherMED, family = binomial(link = "probit"),
##   data = train.bin)
##
## Deviance Residuals:
##   Min       1Q   Median       3Q      Max
## -1.2840  -0.5759  -0.3948  -0.2391   3.2705
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -1.5092975   0.0469985  -32.114 < 2e-16 ***
## logLOS        0.0343635   0.0112216   3.062  0.0022 **
## Age          -0.0037101   0.0005667  -6.547 5.88e-11 ***
## logRiskscore  0.7115770   0.0124458  57.174 < 2e-16 ***
## DRGOtherSURG  0.0558636   0.0341427   1.636  0.1018
## DRGOtherMED   0.0710677   0.0273260   2.601  0.0093 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##   Null deviance: 38117  on 50081  degrees of freedom
## Residual deviance: 33938  on 50076  degrees of freedom
## AIC: 33950
##
## Number of Fisher Scoring iterations: 5
predsprobit <- predict(glmprobit,newdat=test.bin,type="response")
rocprobit <- roc(test.bin$Readmission.Status,predsprobit)

## Setting levels: control = 0, case = 1
## Setting direction: controls < cases
```

```
confusionMatrix(factor(1*(predsprobit>.5)),factor(test.bin$Readmission.Status))
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction      0      1
##           0 14639  2047
##           1      4      4
##
##           Accuracy : 0.8771
##           95% CI : (0.8721, 0.8821)
##           No Information Rate : 0.8771
##           P-Value [Acc > NIR] : 0.5059
##
##           Kappa : 0.0029
##
##  Mcnemar's Test P-Value : <2e-16
##
##           Sensitivity : 0.99973
##           Specificity : 0.00195
##           Pos Pred Value : 0.87732
##           Neg Pred Value : 0.50000
##           Prevalence : 0.87714
##           Detection Rate : 0.87690
##           Detection Prevalence : 0.99952
##           Balanced Accuracy : 0.50084
##
##           'Positive' Class : 0
##
```

```
plot(rocprobit)
```



```
auc(rocprobit)
```

```
## Area under the curve: 0.7334
```

## Task 8

I first run the model on the full dataset.

```
glmprobit <- glm(Readmission.Status ~ logLOS + Age + logRiskscore + DRGOtherSURG + DRGOtherMED, data=readmission.bin)
```

```
summary(glmprobit)
```

```
##
## Call:
## glm(formula = Readmission.Status ~ logLOS + Age + logRiskscore +
##      DRGOtherSURG + DRGOtherMED, family = binomial(link = "probit"),
##      data = readmission.bin)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.2680  -0.5747  -0.3952  -0.2421   3.9745
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -1.5026235  0.0407530 -36.871  < 2e-16 ***
## logLOS        0.0329948  0.0097365   3.389  0.000702 ***
## Age          -0.0036999  0.0004916  -7.526  5.25e-14 ***
## logRiskscore  0.7008229  0.0107283  65.325  < 2e-16 ***
## DRGOtherSURG  0.0560715  0.0293707   1.909  0.056250 .
```



```
## DRGOtherMED    0.0691546  0.0237996   2.906 0.003664 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 50559  on 66775  degrees of freedom
## Residual deviance: 45116  on 66770  degrees of freedom
## AIC: 45128
##
## Number of Fisher Scoring iterations: 5
```

I now create some arbitrary patients to learn how changing the values affects the predicted probability of readmission.

```
new.data <- data.frame("logLOS" = c(log(5),log(6),log(5),log(5),log(5),log(5)), "Age" = c(75,75,80,75,75,75),
new.data
```

```
##      logLOS Age logRiskscore DRGOtherSURG DRGOtherMED
## 1 1.609438  75    0.6237971           0           0
## 2 1.791759  75    0.6237971           0           0
## 3 1.609438  80    0.6237971           0           0
## 4 1.609438  75    0.7193021           0           0
## 5 1.609438  75    0.6237971           1           0
## 6 1.609438  75    0.6237971           0           1
```

```
predict(glmprobit, newdat = new.data, type = "response")
```

```
##           1           2           3           4           5           6
## 0.09855331 0.09960192 0.09537932 0.11068247 0.10864481 0.11110281
```

## Task 9

I rerun the model on the full dataset.

```
glmprobit <- glm(Readmission.Status ~ logLOS + Age + logRiskscore + DRGOtherSURG + DRGOtherMED, data=readmission.bin)
summary(glmprobit)
```

```
##
## Call:
## glm(formula = Readmission.Status ~ logLOS + Age + logRiskscore +
##      DRGOtherSURG + DRGOtherMED, family = binomial(link = "probit"),
##      data = readmission.bin)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.2680  -0.5747  -0.3952  -0.2421   3.9745
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.5026235  0.0407530 -36.871  < 2e-16 ***
## logLOS       0.0329948  0.0097365   3.389 0.000702 ***
## Age         -0.0036999  0.0004916  -7.526 5.25e-14 ***
## logRiskscore 0.7008229  0.0107283  65.325  < 2e-16 ***
## DRGOtherSURG 0.0560715  0.0293707   1.909 0.056250 .
```

```
## DRGOtherMED    0.0691546  0.0237996   2.906 0.003664 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 50559  on 66775  degrees of freedom
## Residual deviance: 45116  on 66770  degrees of freedom
## AIC: 45128
##
## Number of Fisher Scoring iterations: 5

predsprobit <- predict(glmprobit,newdat=readmission.bin,type="response")

confusionMatrix(factor(1*(predsprobit>.5)),factor(readmission.bin$Readmission.Status))

## Confusion Matrix and Statistics
##
##              Reference
## Prediction      0      1
##              0 58356  8399
##              1   11    10
##
##              Accuracy : 0.8741
##              95% CI : (0.8715, 0.8766)
##      No Information Rate : 0.8741
##      P-Value [Acc > NIR] : 0.5076
##
##              Kappa : 0.0017
##
##  Mcnemar's Test P-Value : <2e-16
##
##              Sensitivity : 0.999812
##              Specificity : 0.001189
##              Pos Pred Value : 0.874182
##              Neg Pred Value : 0.476190
##              Prevalence : 0.874072
##              Detection Rate : 0.873907
##      Detection Prevalence : 0.999686
##              Balanced Accuracy : 0.500500
##
##              'Positive' Class : 0
##
```

This code calculates the cost for different cutoff values. I ran it at different values and am pretesting the final choice of 0.08.

```
cutoff <- 0.08
pred_readmit <- 1*(predsprobit > cutoff)
cm <- confusionMatrix(factor(pred_readmit),factor(readmission.bin$Readmission.Status))

no_intervention_cost <- 25*sum(readmission.bin$Readmission.Status == 1)
full_intervention_cost <- 2*nrow(readmission.bin)
modified_cost <- cm$table[2,1]*2+cm$table[2,2]*2+cm$table[1,2]*25
no_intervention_cost
```

```
## [1] 210225
```

```
full_intervention_cost
```

```
## [1] 133552
```

```
modified_cost
```

```
## [1] 106002
```

The final step is to get the confusion matrix.

```
cm
```

```
## Confusion Matrix and Statistics
```

```
##
```

```
##           Reference
```

```
## Prediction      0      1
```

```
##           0 26379 1096
```

```
##           1 31988 7313
```

```
##
```

```
##           Accuracy : 0.5046
```

```
##           95% CI : (0.5008, 0.5084)
```

```
## No Information Rate : 0.8741
```

```
## P-Value [Acc > NIR] : 1
```

```
##
```

```
##           Kappa : 0.125
```

```
##
```

```
## Mcnemar's Test P-Value : <2e-16
```

```
##
```

```
##           Sensitivity : 0.4520
```

```
##           Specificity : 0.8697
```

```
## Pos Pred Value : 0.9601
```

```
## Neg Pred Value : 0.1861
```

```
##           Prevalence : 0.8741
```

```
## Detection Rate : 0.3950
```

```
## Detection Prevalence : 0.4115
```

```
## Balanced Accuracy : 0.6608
```

```
##
```

```
## 'Positive' Class : 0
```

```
##
```

## ALTERNATIVE FEATURE SELECTION

This section presents an alternative approach. It is based on using hypothesis tests to sequentially remove features. It begins by re-running the probit model on the training set using all the available features.

```
glmprobit <- glm(Readmission.Status ~ . + Race*Gender, data=train, family = binomial(link="probit"))
```

```
summary(glmprobit)
```

```
##
```

```
## Call:
```

```
## glm(formula = Readmission.Status ~ . + Race * Gender, family = binomial(link = "probit"),
```

```
## data = train)
```

```
##
```

```
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.2689  -0.5756  -0.3944  -0.2384   3.2478
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    -1.4371865   0.0741325  -19.387 < 2e-16 ***
## GenderM        -0.0138744   0.0168726   -0.822  0.41090
## RaceBlack       0.0237364   0.0321619    0.738  0.46050
## RaceHispanic    0.0763654   0.0711823    1.073  0.28335
## RaceOthers     -0.0712955   0.0606013   -1.176  0.23941
## ER             -0.0025486   0.0093214   -0.273  0.78454
## Age            -0.0045131   0.0008786   -5.137 2.79e-07 ***
## logLOS          0.0343572   0.0112270    3.060  0.00221 **
## logRiskscore    0.7121243   0.0124659   57.126 < 2e-16 ***
## Under65        -0.0408896   0.0314227   -1.301  0.19316
## DRGMed.NoC     -0.0216338   0.0230263   -0.940  0.34746
## DRGOtherMED     0.0679331   0.0298657    2.275  0.02293 *
## DRGOtherSURG    0.0540437   0.0362048    1.493  0.13551
## DRGSurg.C       0.0086575   0.0215849    0.401  0.68835
## DRGSurg.NoC    -0.0045170   0.0236140   -0.191  0.84830
## DRGUNGROUP      0.0704562   0.0778915    0.905  0.36571
## GenderM:RaceBlack 0.0159955   0.0493390    0.324  0.74579
## GenderM:RaceHispanic -0.1665655  0.1147644   -1.451  0.14668
## GenderM:RaceOthers 0.1305341   0.0854834    1.527  0.12676
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 38117  on 50081  degrees of freedom
## Residual deviance: 33928  on 50063  degrees of freedom
## AIC: 33966
##
## Number of Fisher Scoring iterations: 5
```

It is difficult to deal with a categorical interaction variable if the goal is to remove levels. With the interaction, there are actually 8 levels in play and they would need to be created as a new variable in order to merge some of them. Given that none of the three interaction terms appear to add value, I'll save time and go ahead and remove them.

```
glmprobit <- glm(Readmission.Status ~ ., data=train, family = binomial(link="probit"))
summary(glmprobit)
```

```
##
## Call:
## glm(formula = Readmission.Status ~ ., family = binomial(link = "probit"),
##      data = train)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.2706  -0.5757  -0.3945  -0.2387   3.2499
##
## Coefficients:
```

```
##           Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.4392073  0.0740620 -19.432 < 2e-16 ***
## GenderM     -0.0107974  0.0154547  -0.699  0.48477
## RaceBlack    0.0304827  0.0246611   1.236  0.21644
## RaceHispanic 0.0102060  0.0557975   0.183  0.85487
## RaceOthers  -0.0067658  0.0427270  -0.158  0.87418
## ER          -0.0025921  0.0093202  -0.278  0.78092
## Age         -0.0045021  0.0008785  -5.125 2.98e-07 ***
## logLOS       0.0343042  0.0112250   3.056  0.00224 **
## logRiskscore 0.7122625  0.0124642  57.145 < 2e-16 ***
## Under65     -0.0405129  0.0314153  -1.290  0.19719
## DRGMed.NoC   -0.0220462  0.0230238  -0.958  0.33830
## DRGOtherMED   0.0683215  0.0298592   2.288  0.02213 *
## DRGOtherSURG  0.0531373  0.0362010   1.468  0.14215
## DRGSurg.C     0.0083802  0.0215833   0.388  0.69782
## DRGSurg.NoC  -0.0045720  0.0236071  -0.194  0.84643
## DRGUNGROUP    0.0678532  0.0778955   0.871  0.38371
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 38117  on 50081  degrees of freedom
## Residual deviance: 33932  on 50066  degrees of freedom
## AIC: 33964
##
## Number of Fisher Scoring iterations: 5
```

Normally the next step would be combine Race-Others with Race-White (the base) but given the similarity to Race-Hispanic, both will be combined with Race-White. This is mostly done to save time.

```
readmission$RaceGender <- NULL #Need to remove this variable that was created earlier.
```

```
readmission2<-readmission
```

```
library(plyr)
```

```
var <- "Race"
```

```
var.levels <- levels(readmission2[,var])
```

```
readmission2[,var] <- mapvalues(readmission2[,var],var.levels,c("NonBlack","Black","NonBlack","NonBlack")
#Relevel
```

```
table <- as.data.frame(table(readmission2[,var]))
```

```
max <- which.max(table[,2])
```

```
level.name <- as.character(table[max,1])
```

```
readmission2[,var] <- relevel(readmission2[,var], ref = level.name)
```

```
table(readmission2[,var])
```

```
##
```

```
## NonBlack    Black
```

```
##      59679      7097
```

Running the model again, remembering to create new train and test sets. The same partition continues to be used to keep results consistent.

```
readmission <- readmission2
```

```
train <- readmission[partition,]
```

```
test <- readmission[-partition,]
glmprobit <- glm(Readmission.Status ~ ., data=train, family = binomial(link="probit"))

summary(glmprobit)
```

```
##
## Call:
## glm(formula = Readmission.Status ~ ., family = binomial(link = "probit"),
##      data = train)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.2707  -0.5759  -0.3945  -0.2387   3.2495
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -1.439477   0.073950 -19.465  < 2e-16 ***
## GenderM      -0.010864   0.015453  -0.703  0.48201
## RaceBlack     0.030502   0.024555   1.242  0.21416
## ER           -0.002584   0.009320  -0.277  0.78156
## Age          -0.004499   0.000878  -5.124 2.98e-07 ***
## logLOS        0.034325   0.011224   3.058  0.00223 **
## logRiskscore  0.712279   0.012464  57.147 < 2e-16 ***
## Under65      -0.040346   0.031403  -1.285  0.19886
## DRGMed.NoC   -0.022075   0.023023  -0.959  0.33763
## DRGOtherMED   0.068353   0.029858   2.289  0.02206 *
## DRGOtherSURG  0.053060   0.036199   1.466  0.14271
## DRGSurg.C     0.008356   0.021583   0.387  0.69863
## DRGSurg.NoC  -0.004598   0.023606  -0.195  0.84556
## DRGUNGROUP    0.067814   0.077894   0.871  0.38397
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 38117  on 50081  degrees of freedom
## Residual deviance: 33932  on 50068  degrees of freedom
## AIC: 33960
##
## Number of Fisher Scoring iterations: 5
```

For similar reasons, we simultaneously merge the two insignificant DRG levels into the base.

```
readmission2<-readmission

var <- "DRG"
var.levels <- levels(readmission2[,var])
readmission2[,var] <- mapvalues(readmission2[,var],var.levels,c("DRGbase","Med.NoC","OtherMED","OtherSURG"))
#Relevel
table <- as.data.frame(table(readmission2[,var]))
max <- which.max(table[,2])
level.name <- as.character(table[max,1])
readmission2[,var] <- relevel(readmission2[,var], ref = level.name)

table(readmission2[,var])
```

```
##
##   DRGbase   Med.NoC   OtherMED   OtherSURG   UNGROUP
##   45121     12310     5357       3424       564
readmission <- readmission2
train <- readmission[partition,]
test <- readmission[-partition,]
glmprobit <- glm(Readmission.Status ~ ., data=train, family = binomial(link="probit"))
summary(glmprobit)
```

```
##
## Call:
## glm(formula = Readmission.Status ~ ., family = binomial(link = "probit"),
##      data = train)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.2706  -0.5759  -0.3946  -0.2387   3.2554
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -1.438216   0.073069 -19.683  < 2e-16 ***
## GenderM      -0.010833   0.015452  -0.701  0.48325
## RaceBlack     0.030546   0.024555   1.244  0.21350
## ER           -0.002595   0.009320  -0.278  0.78065
## Age          -0.004493   0.000878  -5.118 3.09e-07 ***
## logLOS        0.034320   0.011223   3.058  0.00223 **
## logRiskscore  0.712242   0.012463  57.147  < 2e-16 ***
## Under65      -0.040227   0.031401  -1.281  0.20017
## DRGMed.NoC   -0.023769   0.020119  -1.181  0.23744
## DRGOtherMED   0.066662   0.027689   2.408  0.01606 *
## DRGOtherSURG  0.051371   0.034428   1.492  0.13567
## DRGUNGROUP    0.066127   0.077090   0.858  0.39101
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 38117  on 50081  degrees of freedom
## Residual deviance: 33933  on 50070  degrees of freedom
## AIC: 33957
##
## Number of Fisher Scoring iterations: 5
```

Now remove ER.

```
glmprobit <- glm(Readmission.Status ~ Gender + Race + Age + logLOS + logRiskscore + Under65 + DRG, data=
summary(glmprobit)
```

```
##
## Call:
## glm(formula = Readmission.Status ~ Gender + Race + Age + logLOS +
```

```
##      logRiskscore + Under65 + DRG, family = binomial(link = "probit"),
##      data = train)
##
## Deviance Residuals:
##      Min        1Q      Median        3Q        Max
## -1.2714   -0.5761   -0.3946   -0.2387    3.2566
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -1.439469    0.072930 -19.738 < 2e-16 ***
## GenderM      -0.010808    0.015452  -0.699  0.48426
## RaceBlack     0.030535    0.024555   1.244  0.21367
## Age          -0.004495    0.000878  -5.119 3.07e-07 ***
## logLOS        0.034349    0.011223   3.061  0.00221 **
## logRiskscore  0.712211    0.012463  57.148 < 2e-16 ***
## Under65      -0.040278    0.031400  -1.283  0.19959
## DRGMed.NoC    -0.023733    0.020119  -1.180  0.23815
## DRGOtherMED   0.066616    0.027688   2.406  0.01613 *
## DRGOtherSURG  0.051364    0.034428   1.492  0.13572
## DRGUNGROUP    0.066078    0.077092   0.857  0.39137
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 38117  on 50081  degrees of freedom
## Residual deviance: 33933  on 50071  degrees of freedom
## AIC: 33955
##
## Number of Fisher Scoring iterations: 5
```

Now remove Gender

```
glmprobit <- glm(Readmission.Status ~ Race + Age + logLOS + logRiskscore + Under65 + DRG, data=train,
summary(glmprobit)
```

```
##
## Call:
## glm(formula = Readmission.Status ~ Race + Age + logLOS + logRiskscore +
##      Under65 + DRG, family = binomial(link = "probit"), data = train)
##
## Deviance Residuals:
##      Min        1Q      Median        3Q        Max
## -1.2749   -0.5758   -0.3947   -0.2389    3.2608
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -1.4479936    0.0719116 -20.136 < 2e-16 ***
## RaceBlack     0.0308596    0.0245506   1.257  0.20876
## Age          -0.0044419    0.0008748  -5.078 3.82e-07 ***
## logLOS        0.0343108    0.0112223   3.057  0.00223 **
## logRiskscore  0.7120724    0.0124610  57.144 < 2e-16 ***
## Under65      -0.0395810    0.0313830  -1.261  0.20723
## DRGMed.NoC    -0.0237511    0.0201186  -1.181  0.23778
```



```
## DRGOtherMED    0.0667391  0.0276865   2.411  0.01593 *
## DRGOtherSURG   0.0515353  0.0344275   1.497  0.13441
## DRGUNGROUP    0.0658603  0.0770924   0.854  0.39294
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 38117  on 50081  degrees of freedom
## Residual deviance: 33933  on 50072  degrees of freedom
## AIC: 33953
##
## Number of Fisher Scoring iterations: 5
```

Rather than remove DRGUNGROUP at this stage, it will be combined with DRGOtherSURG. The difference in significance is likely due to sample sizes.

```
readmission2<-readmission

var <- "DRG"
var.levels <- levels(readmission2[,var])
readmission2[,var] <- mapvalues(readmission2[,var],var.levels,c("DRGbase","Med.NoC","OtherMED","OSUngror
#Relevel
table <- as.data.frame(table(readmission2[,var]))
max <- which.max(table[,2])
level.name <- as.character(table[max,1])
readmission2[,var] <- relevel(readmission2[,var], ref = level.name)

table(readmission2[,var])
```

```
##
##      DRGbase   Med.NoC   OtherMED   OSUngror
##      45121      12310      5357      3988
```

```
readmission <- readmission2
train <- readmission[partition,]
test <- readmission[-partition,]
glmprobit <- glm(Readmission.Status ~ Race + Age + logLOS + logRiskscore + Under65 + DRG, data=train, f
summary(glmprobit)
```

```
##
## Call:
## glm(formula = Readmission.Status ~ Race + Age + logLOS + logRiskscore +
##      Under65 + DRG, family = binomial(link = "probit"), data = train)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.2749  -0.5758  -0.3947  -0.2390   3.2609
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.4480241  0.0719121 -20.136  < 2e-16 ***
## RaceBlack    0.0308837  0.0245501   1.258  0.20840
## Age         -0.0044420  0.0008748  -5.078 3.82e-07 ***
## logLOS       0.0343218  0.0112225   3.058  0.00223 **
```

```
## logRiskscore 0.7120944 0.0124602 57.149 < 2e-16 ***
## Under65      -0.0395813 0.0313827 -1.261 0.20722
## DRGMed.NoC   -0.0237522 0.0201187 -1.181 0.23776
## DRGOtherMED  0.0667366 0.0276866 2.410 0.01593 *
## DRGOSUngroup 0.0537795 0.0318155 1.690 0.09096 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
## Null deviance: 38117 on 50081 degrees of freedom
## Residual deviance: 33933 on 50073 degrees of freedom
## AIC: 33951
##
## Number of Fisher Scoring iterations: 5
```

Next to go is Med.NoC with the highest p-value.

```
readmission2<-readmission

var <- "DRG"
var.levels <- levels(readmission2[,var])
readmission2[,var] <- mapvalues(readmission2[,var],var.levels,c("DRGbase","DRGbase","OtherMED","OSUngroup")
#Relevel
table <- as.data.frame(table(readmission2[,var]))
max <- which.max(table[,2])
level.name <- as.character(table[max,1])
readmission2[,var] <- relevel(readmission2[,var], ref = level.name)

table(readmission2[,var])

##
## DRGbase OtherMED OSUngroup
## 57431 5357 3988

readmission <- readmission2
train <- readmission[partition,]
test <- readmission[-partition,]
glmprobit <- glm(Readmission.Status ~ Race + Age + logLOS + logRiskscore + Under65 + DRG, data=train, family=binomial)

summary(glmprobit)

##
## Call:
## glm(formula = Readmission.Status ~ Race + Age + logLOS + logRiskscore +
## Under65 + DRG, family = binomial(link = "probit"), data = train)
##
## Deviance Residuals:
## Min 1Q Median 3Q Max
## -1.2749 -0.5759 -0.3944 -0.2391 3.2651
##
## Coefficients:
## Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.4524752 0.0718246 -20.223 < 2e-16 ***
## RaceBlack 0.0305938 0.0245494 1.246 0.21269
## Age -0.0044443 0.0008748 -5.081 3.76e-07 ***
```

```
## logLOS      0.0341318  0.0112210   3.042  0.00235 **
## logRiskscore 0.7119330  0.0124591  57.142 < 2e-16 ***
## Under65     -0.0394441  0.0313816  -1.257  0.20878
## DRGOtherMED  0.0718764  0.0273458   2.628  0.00858 **
## DRGOSUngroup 0.0589066  0.0315200   1.869  0.06164 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 38117  on 50081  degrees of freedom
## Residual deviance: 33935  on 50074  degrees of freedom
## AIC: 33951
##
## Number of Fisher Scoring iterations: 5
```

Now remove Race.

```
readmission <- readmission2
train <- readmission[partition,]
test <- readmission[-partition,]
glmprobit <- glm(Readmission.Status ~ Age + logLOS + logRiskscore + Under65 + DRG, data=train, family =
summary(glmprobit)
```

```
##
## Call:
## glm(formula = Readmission.Status ~ Age + logLOS + logRiskscore +
##      Under65 + DRG, family = binomial(link = "probit"), data = train)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.2801  -0.5760  -0.3945  -0.2391   3.2626
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.4456943   0.0716320 -20.182 < 2e-16 ***
## Age         -0.0044962   0.0008739  -5.145 2.68e-07 ***
## logLOS       0.0340328   0.0112205   3.033  0.00242 **
## logRiskscore  0.7121407   0.0124577  57.165 < 2e-16 ***
## Under65     -0.0371728   0.0313192  -1.187  0.23527
## DRGOtherMED  0.0721155   0.0273450   2.637  0.00836 **
## DRGOSUngroup 0.0588470   0.0315197   1.867  0.06190 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 38117  on 50081  degrees of freedom
## Residual deviance: 33936  on 50075  degrees of freedom
## AIC: 33950
##
## Number of Fisher Scoring iterations: 5
```

Now remove Under65.

```
readmission <- readmission2
train <- readmission[partition,]
test <- readmission[-partition,]
glmprobit <- glm(Readmission.Status ~ Age + logLOS + logRiskscore + DRG, data=train, family = binomial())
summary(glmprobit)
```

```
##
## Call:
## glm(formula = Readmission.Status ~ Age + logLOS + logRiskscore +
##      DRG, family = binomial(link = "probit"), data = train)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.2838  -0.5759  -0.3948  -0.2391   3.2709
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -1.5099622   0.0470000  -32.127  < 2e-16 ***
## Age          -0.0037063   0.0005667   -6.540 6.15e-11 ***
## logLOS         0.0341991   0.0112199    3.048  0.00230 **
## logRiskscore  0.7114348   0.0124460   57.162  < 2e-16 ***
## DRGOtherMED   0.0719061   0.0273415    2.630  0.00854 **
## DRGOSUngroup  0.0590267   0.0315181    1.873  0.06110 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 38117  on 50081  degrees of freedom
## Residual deviance: 33938  on 50076  degrees of freedom
## AIC: 33950
##
## Number of Fisher Scoring iterations: 5
```

Everything is now significant at the 10% level and we stop. If we wanted a lower significance level for decision making, the process would continue.