Modern Data Mining - HW 3

Anirudh Bajaj
Esther Shin
Matt LeBaron
Raman Chadha

Overview / Instructions

This is homework #3 of STAT 471/571/701. It will be **due on 28 October, 2018 by 11:59 PM** on Canvas. You can directly edit this file to add your answers. Submit the Rmd file, a PDF or word or HTML version with **only 1 submission** per HW team.

Note: To minimize your work and errors, we provide this Rmd file to guide you in the process of building your final report. To that end, we've included code to load the necessary data files. Make sure that the following files are in the same folder as this R Markdown file:

- `FRAMINGHAM.dat`
- `Bills.subset.csv`
- `Bills.subset.test.csv`

The data should load properly if you are working in Rstudio, *without needing to change your working directory.*

Solutions will be posted. Make sure to compare your answers to and understand the solutions.

R Markdown / Knitr tips

You should think of this R Markdown file as generating a polished report, one that you would be happy to show other people (or your boss). There shouldn't be any extraneous output; all graphs and code run should clearly have a reason to be run. That means that any output in the final file should have explanations.

A few tips:

- Keep each chunk to only output one thing! In R, if you're not doing an assignment (with the `<-` operator), it's probably going to print something.
- If you don't want to print the R code you wrote (but want to run it, and want to show the results), use a chunk declaration like this: `{r, echo=F}`
- If you don't want to show the results of the R code or the original code, use a chunk declaration like: `{r, include=F}`
- If you don't want to show the results, but show the original code, use a chunk declaration like: `{r, results='hide'}`.
- If you don't want to run the R code at all use `{r, eval = F}`.
- We show a few examples of these options in the below example code.
- For more details about these R Markdown options, see the documentation.
- Delete the instructions and this R Markdown section, since they're not part of your overall report.

Problem 0

Review the code and concepts covered during lecture, in particular, logistic regression and classification.

Problem 1

We will continue to use the Framingham Data (`Framingham.dat`) so that you are already familiar with the data and the variables. All the results are obtained through training data.

To keep our answers consistent, use a subset of the data, and exclude anyone with a missing entry. For your convenience, we've loaded it here together with a brief summary about the data.

We note that this dataset contains 311 people diagnosed with heart disease and 1095 without heart disease.

```
    0    1
 1095  311
```

After a quick cleaning up here is a summary about the data:

```
# using the comment="     ", we get rid of the ## in the output.
summary(hd_data.f)
```

```
       HD             AGE            SEX           SBP             DBP
       0:1086   Min.   :45.00   FEMALE:730   Min.   : 90.0   Min.   : 50.00
       1: 307   1st Qu.:48.00   MALE  :663   1st Qu.:130.0   1st Qu.: 80.00
                Median :52.00                Median :142.0   Median : 90.00
                Mean   :52.43                Mean   :148.1   Mean   : 90.16
                3rd Qu.:56.00                3rd Qu.:160.0   3rd Qu.: 98.00
                Max.   :62.00                Max.   :300.0   Max.   :160.00
            CHOL            FRW             CIG
       Min.   : 96.0   Min.   : 52.0   Min.   : 0.000
       1st Qu.:200.0   1st Qu.: 94.0   1st Qu.: 0.000
       Median :230.0   Median :103.0   Median : 0.000
       Mean   :234.6   Mean   :105.4   Mean   : 8.035
       3rd Qu.:264.0   3rd Qu.:114.0   3rd Qu.:20.000
       Max.   :430.0   Max.   :222.0   Max.   :60.000
```

Part 1A

Conceptual questions to understand building blocks of logistic regression. All the codes in this part should be hidden.

i. Take a random subsample of size 5 from `hd_data_f` which only includes `HD` and `SBP`. Also set `set.seed(50)`. List the three observations neatly below. No code should be shown here.

```
##      HD SBP
## 996   0 142
## 614   0 126
## 281   0 136
## 1075  0 178
## 719   0 126
```

ii. Write down the likelihood function using the five observations above.

$$\begin{aligned}\mathcal{L}\rangle\|(\beta_0,\beta_1|Data) &= \text{Prob(the outcome of the data)}\\ &= Prob((Y=0|SBP=142),(Y=0|SBP=126),(Y=0|SBP=136),(Y=0|SBP=178),(Y=0|SBP=\\ &= Prob(Y=0|SBP=142)\times Prob(Y=0|SBP=126)\times Prob(Y=0|SBP=136)\times Prob(Y=0|SBP=\\ &= \frac{1}{1+e^{\beta_0+142\beta_1}}\cdot\frac{1}{1+e^{\beta_0+126\beta_1}}\cdot\frac{1}{1+e^{\beta_0+136\beta_1}}\cdot\frac{1}{1+e^{\beta_0+178\beta_1}}\cdot\frac{1}{1+e^{\beta_0+126\beta_1}}\end{aligned}$$

iii. Find the MLE's based on this subset. Report the estimated logit function and the probability of `HD`=1. Briefly explain how the MLE's are obtained based on ii. above.

```
mle.sample <- glm(HD~SBP, sample.data, family=binomial(logit))
summary(mle.sample)
```

```
##
## Call:
## glm(formula = HD ~ SBP, family = binomial(logit), data = sample.data)
##
## Deviance Residuals:
##        996         614         281        1075         719
## -6.547e-06  -6.547e-06  -6.547e-06  -6.547e-06  -6.547e-06
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    -24.57  436053.61       0        1
## SBP              0.00    3051.55       0        1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 0.0000e+00  on 4  degrees of freedom
## Residual deviance: 2.1434e-10  on 3  degrees of freedom
## AIC: 4
##
## Number of Fisher Scoring iterations: 23
```

Probability function estimated by glm using only the above 5 sample observations:

logit = -24.57 + 0.00 SBP

$P(HD = 1|SBP) = \frac{e^{-24.57+0.00\times SBP}}{1+e^{-24.57+0.00\times SBP}}$

```
#calculated probability
exp(-24.57)/{1+exp(-24.57)}
```

```
## [1] 2.134935e-11
```

MLE is the maximul likelihood of observing the predicted data, given the input data. Mathematically, taking log both sides of the equation and differentiating it and equating to 0 leads to the logit function. GLM iterates on Beta0 and Beta1 values to maximize this probability. In the above case, the probability of HD=1 is very low. This is because the input sample data does not have any patient with heart disease. The SBP slope is 0, but also note that the intercept and SBP values are independently both not singnificant.

Part 1B

Goal: Identify important risk factors for `Heart.Disease.` through logistic regression. Start a fit with just one factor, `SBP`, and call it `fit1`. Let us add one variable to this at a time from among the rest of the variables.

```
fit1 <- glm(HD~SBP, hd_data.f, family=binomial)
summary(fit1)
fit1.1 <- glm(HD~SBP + AGE, hd_data.f, family=binomial)
summary(fit1.1)
fit1.2 <- glm(HD~SBP + SEX, hd_data.f, family=binomial)
summary(fit1.2)
fit1.3 <- glm(HD~SBP + DBP, hd_data.f, family=binomial)
```

```
summary(fit1.3)
fit1.4 <- glm(HD~SBP + CHOL, hd_data.f, family=binomial)
summary(fit1.4)
fit1.5 <- glm(HD~SBP + DBP, hd_data.f, family=binomial)
summary(fit1.5)
fit1.6 <- glm(HD~SBP + FRW, hd_data.f, family=binomial)
summary(fit1.6)
fit1.7 <- glm(HD~SBP + CIG, hd_data.f, family=binomial)
summary(fit1.7)
```

i. Which single variable would be the most important to add? Add it to your model, and call the new fit `fit2`.

```
#Based on AIC values (lowest is best), Sex is the most important variable to add. Adding it to the mode
fit2 <- glm(HD~SBP + SEX, hd_data.f, family=binomial)
summary(fit2)
```

```
##
## Call:
## glm(formula = HD ~ SBP + SEX, family = binomial, data = hd_data.f)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.6408  -0.7373  -0.5726  -0.4169   2.2452
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -4.570256   0.389727 -11.727  < 2e-16 ***
## SBP          0.018717   0.002324   8.053 8.07e-16 ***
## SEXMALE      0.903420   0.139762   6.464 1.02e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1469.3  on 1392  degrees of freedom
## Residual deviance: 1373.8  on 1390  degrees of freedom
## AIC: 1379.8
##
## Number of Fisher Scoring iterations: 4
```

We will pick up the variable either with highest $|z|$ value, or smallest p value. From all the two variable models we see that `SEX` will be the most important addition on top of the SBP. And here is the summary report.

```
## How to control the summary(fit2) output to cut some junk?
## We could use packages: xtable or broom.
library(xtable)
options(xtable.comment = FALSE)
fit2 <- glm(HD~SBP + SEX, hd_data.f, family=binomial)
xtable(fit2)
```

```
## \begin{table}[ht]
## \centering
## \begin{tabular}{rrrrr}
##   \hline
```

```
##  & Estimate & Std. Error & z value & Pr($>$$|$z$|$) \\
##   \hline
## (Intercept) & -4.5703 & 0.3897 & -11.73 & 0.0000 \\
##   SBP & 0.0187 & 0.0023 & 8.05 & 0.0000 \\
##   SEXMALE & 0.9034 & 0.1398 & 6.46 & 0.0000 \\
##    \hline
## \end{tabular}
## \end{table}
```

ii. Is the residual deviance of `fit2` always smaller than that of `fit1`? Why or why not?

In fit2, we find that SEX is a significant variable (p-value < 0.05), that is, controllig for SBP, we reject the null hypothesis that SEX has no influence on the HD value. Whenever we add an additional variable that is significant, the residual deviance will always decrease.

iii. Perform both the Wald test and the Likelihood ratio tests (Chi-Squared) to see if the added variable is significant at the .01 level. What are the p-values from each test? Are they the same?

Likelihood Ratio test

Null Deviance = 1469.3 Residual Deviance = 1373.8

```
chi_sq <- 1469.3 - 1373.8
pchisq(chi_sq, 1, lower.tail=FALSE)
```

```
## [1] 1.478913e-22
```

Wald test (installing package "survey" to directly conduct Wald test)

```
library(survey)
```

```
## Loading required package: grid
## Loading required package: Matrix
## Loading required package: survival
##
## Attaching package: 'survey'
## The following object is masked from 'package:graphics':
##
##     dotchart
```

```
regTermTest(fit2, "SEX")
```

```
## Wald test for SEX
##  in glm(formula = HD ~ SBP + SEX, family = binomial, data = hd_data.f)
## F =  41.78308  on  1  and  1390  df: p= 1.4079e-10
```

The added variable SEX is significant at the 0.01 level. p-value from Likelihood-Ratio test is 1.478913e-22 and from Wald test is 1.4079e-10. The p-values are different. Whereas Wald Test tests significance of an individual variable (in this case SEX), Likelihood Test tests significance of a set of variables (in this case SBP and SEX).

Part 1C - Model building

Start with all variables. Our goal is to fit a well-fitting model, that is still small and easy to interpret (parsimonious).

i. Use backward selection method. Only keep variables whose coefficients are significantly different from 0 at .05 level. Kick out the variable with the largest p-value first, and then re-fit the model to see if there are other variables you want to kick out.

```
fit_back1 <- glm(HD~.,hd_data.f, family=binomial)
summary(fit_back1)
```

```
##
## Call:
## glm(formula = HD ~ ., family = binomial, data = hd_data.f)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.7051  -0.7268  -0.5556  -0.3329   2.4455
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -9.334797   1.036630  -9.005  < 2e-16 ***
## AGE          0.062491   0.014995   4.167 3.08e-05 ***
## SEXMALE      0.906102   0.157639   5.748 9.03e-09 ***
## SBP          0.014838   0.003886   3.818 0.000135 ***
## DBP          0.002875   0.007620   0.377 0.705941
## CHOL         0.004459   0.001505   2.962 0.003053 **
## FRW          0.005795   0.004055   1.429 0.152957
## CIG          0.012309   0.006087   2.022 0.043150 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1469.3  on 1392  degrees of freedom
## Residual deviance: 1343.1  on 1385  degrees of freedom
## AIC: 1359.1
##
## Number of Fisher Scoring iterations: 4
```

```
fit_back2 <- glm(HD~. - DBP,hd_data.f, family=binomial)
summary(fit_back2)
```

```
##
## Call:
## glm(formula = HD ~ . - DBP, family = binomial, data = hd_data.f)
```

```
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.7066  -0.7279  -0.5517  -0.3343   2.4501
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -9.227856   0.996153  -9.263  < 2e-16 ***
## AGE          0.061529   0.014775   4.164 3.12e-05 ***
## SEXMALE      0.911274   0.157117   5.800 6.63e-09 ***
## SBP          0.015966   0.002487   6.420 1.37e-10 ***
## CHOL         0.004493   0.001503   2.990  0.00279 **
## FRW          0.006039   0.004004   1.508  0.13151
## CIG          0.012279   0.006088   2.017  0.04369 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1469.3  on 1392  degrees of freedom
## Residual deviance: 1343.3  on 1386  degrees of freedom
## AIC: 1357.3
##
## Number of Fisher Scoring iterations: 4
```

```
fit_back1 <- glm(HD~. - DBP - FRW,hd_data.f, family=binomial)
summary(fit_back1)
```

```
##
## Call:
## glm(formula = HD ~ . - DBP - FRW, family = binomial, data = hd_data.f)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.7541  -0.7288  -0.5538  -0.3438   2.4470
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -8.702282   0.926831  -9.389  < 2e-16 ***
## AGE          0.061358   0.014753   4.159 3.20e-05 ***
## SEXMALE      0.885748   0.155792   5.685 1.30e-08 ***
## SBP          0.017085   0.002372   7.203 5.88e-13 ***
## CHOL         0.004398   0.001499   2.935  0.00334 **
## CIG          0.011358   0.006058   1.875  0.06083 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1469.3  on 1392  degrees of freedom
## Residual deviance: 1345.5  on 1387  degrees of freedom
## AIC: 1357.5
##
## Number of Fisher Scoring iterations: 4
```

```
fit_back3 <- glm(HD~. - DBP - FRW - CIG, hd_data.f, family=binomial)
summary(fit_back3)
```

```
##
## Call:
## glm(formula = HD ~ . - DBP - FRW - CIG, family = binomial, data = hd_data.f)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.6069  -0.7348  -0.5523  -0.3476   2.4344
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -8.408724   0.908600  -9.255  < 2e-16 ***
## AGE          0.056639   0.014500   3.906 9.38e-05 ***
## SEXMALE      0.989870   0.145053   6.824 8.84e-12 ***
## SBP          0.016956   0.002362   7.179 7.02e-13 ***
## CHOL         0.004480   0.001495   2.996  0.00274 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1469.3  on 1392  degrees of freedom
## Residual deviance: 1349.0  on 1388  degrees of freedom
## AIC: 1359
##
## Number of Fisher Scoring iterations: 4
```

ii. Use AIC as the criterion for model selection. Find a model with small AIC through exhaustive search. Does exhaustive search guarantee that the p-values for all the remaining variables are less than .05? Is our final model here the same as the model from backwards elimination?

preparing the design matrix

```
des_mat <- model.matrix(HD ~.+0, hd_data.f)
des_mat <- data.frame(des_mat, hd_data.f$HD)
head(des_mat)
```

```
##   AGE SEXFEMALE SEXMALE SBP DBP CHOL FRW CIG hd_data.f.HD
## 1  45         0       1  90  50  216  76   5            0
## 2  49         0       1 100  64  237  97   0            0
## 3  47         0       1 100  70  215  86  50            0
## 4  48         0       1 108  70  340  93   0            0
## 5  49         0       1 108  75  149  95   0            0
## 6  47         0       1 108  68  165  88   0            0
```

```
str(des_mat)
```

```
## 'data.frame':    1393 obs. of  9 variables:
##  $ AGE         : num  45 49 47 48 49 47 48 48 46 45 ...
##  $ SEXFEMALE   : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ SEXMALE     : num  1 1 1 1 1 1 1 1 1 1 ...
##  $ SBP         : num  90 100 100 108 108 108 108 110 110 110 ...
```

```
##  $ DBP         : num  50 64 70 70 75 68 70 70 80 72 ...
##  $ CHOL        : num  216 237 215 340 149 165 196 229 204 183 ...
##  $ FRW         : num  76 97 86 93 95 88 79 85 112 93 ...
##  $ CIG         : num  5 0 50 0 0 0 20 25 0 20 ...
##  $ hd_data.f.HD: Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
```

finding best model based on AIC

```
library(bestglm)
```

```
## Loading required package: leaps
```

```
fit.all <- bestglm(des_mat, family = binomial, method = "exhaustive", IC="AIC", nvmax = 10)
```

```
## Morgan-Tatar search since family is non-gaussian.
```

```
fit.all$BestModels
```

```
##    AGE SEXFEMALE SEXMALE  SBP   DBP CHOL   FRW   CIG Criterion
## 1 TRUE     FALSE    TRUE TRUE FALSE TRUE  TRUE  TRUE  1355.285
## 2 TRUE      TRUE   FALSE TRUE FALSE TRUE  TRUE  TRUE  1355.285
## 3 TRUE     FALSE    TRUE TRUE FALSE TRUE FALSE  TRUE  1355.536
## 4 TRUE      TRUE   FALSE TRUE FALSE TRUE FALSE  TRUE  1355.536
## 5 TRUE     FALSE    TRUE TRUE FALSE TRUE FALSE FALSE  1357.011
```

```
summary(fit.all$BestModel)
```

```
##
## Call:
## glm(formula = y ~ ., family = family, data = Xi, weights = weights)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.7066  -0.7279  -0.5517  -0.3343   2.4501
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -9.227856   0.996153  -9.263  < 2e-16 ***
## AGE          0.061529   0.014775   4.164 3.12e-05 ***
## SEXMALE      0.911274   0.157117   5.800 6.63e-09 ***
## SBP          0.015966   0.002487   6.420 1.37e-10 ***
## CHOL         0.004493   0.001503   2.990  0.00279 **
## FRW          0.006039   0.004004   1.508  0.13151
## CIG          0.012279   0.006088   2.017  0.04369 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1469.3  on 1392  degrees of freedom
## Residual deviance: 1343.3  on 1386  degrees of freedom
## AIC: 1357.3
##
## Number of Fisher Scoring iterations: 4
```

No, exhaustive search does not gurantee that the p-values of all variables are less than 0.05. This is because the above algorithm finds the best model based on a different criterion (that is, lowest AIC value).

Hence, the model that the above algorithm produces is different as that given by backwards elimination.

iii. Use the model chosen from part ii. as the final model. Write a brief summary to describe important factors relating to Heart Diseases (i.e. the relationships between those variables in the model and heart disease). Give a definition of "important factors".

For the purpose of this analysis, important factors are defined as those factors that collectively best predict the probability of a heart disease in a patient.Based on the best model above (by AIC rule), the important prdictors of heard disease are as follows:

a) AGE - As age increases, probability of heart disease increases. Controlling for other variables, increasing age by 1 year leads to an increase of log(odds of getting heart disease) by ~0.06.

b) SEX - Being a male increases the probability of having a heart disease. Controlling for other variables, being a male increases log(odds of getting heart disease) by ~0.9.

c) SBP - As SBP increases, probability of heart disease increases. Controlling for other variables, increasing SBP by 1 unit leads to an increase of log(odds of getting heart disease) by ~0.02.

d) CHOL - As cholesterol increases, probability of heart disease increases. Controlling for other variables, increasing cholesterol by 1 unit leads to an increase of log(odds of getting heart disease) by ~0.004.

e) FRW - As weight increases, probability of heart disease increases. Controlling for other variables, increasing weight by 1 pound leads to an increase of log(odds of getting heart disease) by ~0.006.

f) CIG - As number of cigarettes smoked by a patient increases, probability of heart disease increases. Controlling for other variables, increasing number of cigarettes by 1 leads to an increase of log(odds of getting heart disease) by ~0.01.

Part 1D - Prediction

Liz is a patient with the following readings: `AGE=50, GENDER=FEMALE, SBP=110, DBP=80, CHOL=180, FRW=105, CIG=0`. What is the probability that she will have heart disease, according to our final model?

```
prob_HD <- exp(-9.227856+(0.061529*50)+(0.015966*110)+(0.004493*180)+(0.006039*105)+(0.012279*0))/(1+exp
prob_HD
```

```
## [1] 0.0496274
```

The probability that Liz will have a heard disease is 0.0496274.

Part 2 - Classification analysis

a. Display the ROC curve using `fit1`. Explain what ROC reports and how to use the graph. Specify the classifier such that the False Positive rate is less than .1 and the True Positive rate is as high as possible.

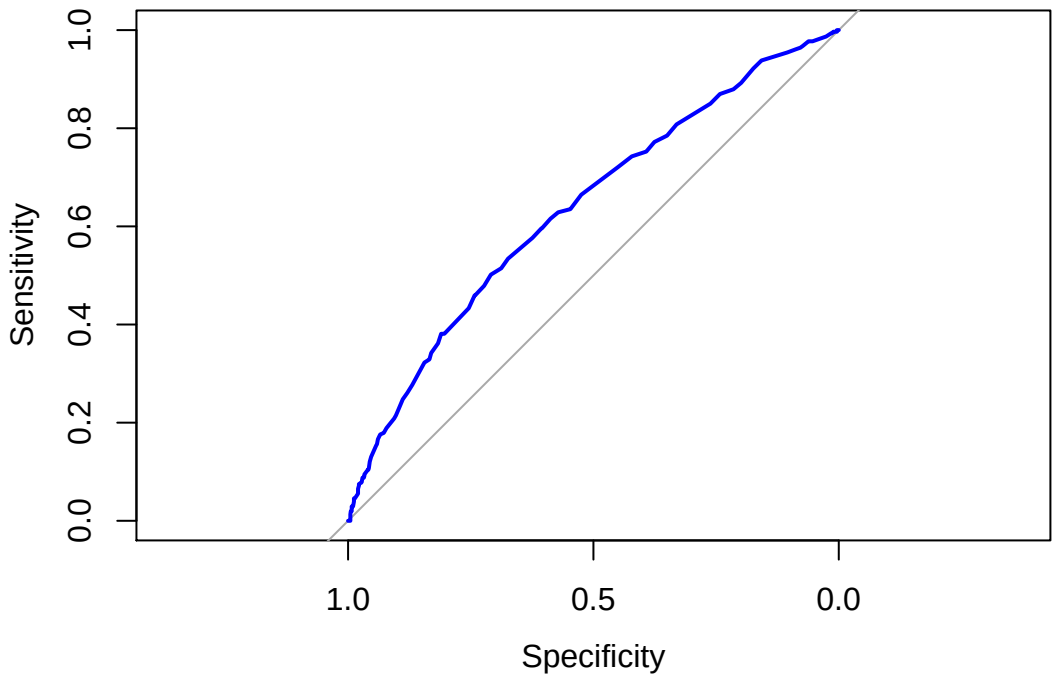
```
library(pROC)
```

```
## Type 'citation("pROC")' for a citation.
```

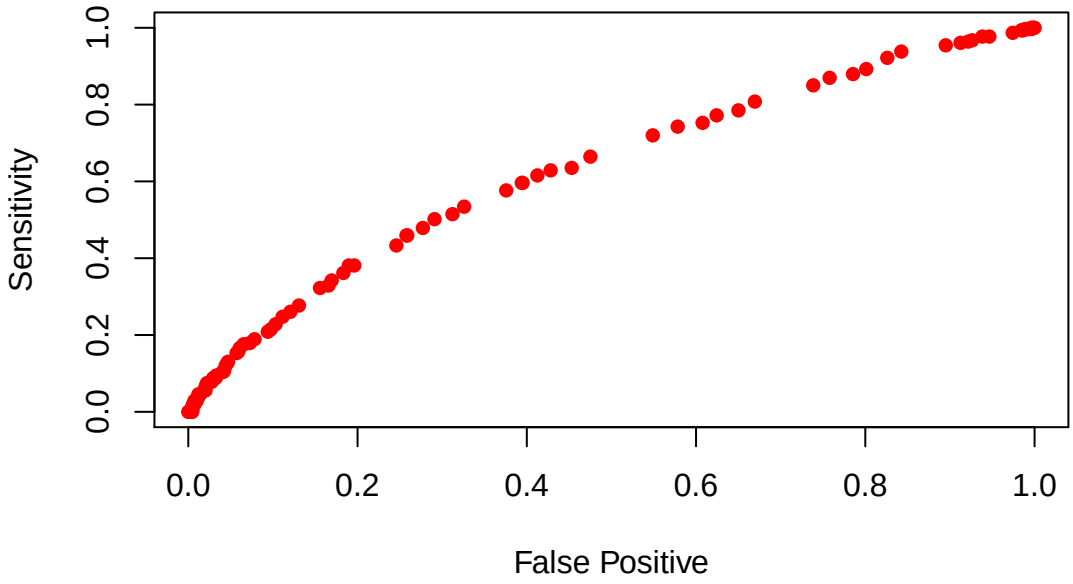
```
##
## Attaching package: 'pROC'
```

```
## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
```

```
fit1.roc<- roc(hd_data.f$HD, fit1$fitted, plot=T, col="blue")
```




```
plot(1-fit1.roc$specificities, fit1.roc$sensitivities, col="red", pch=16,
     xlab="False Positive",
     ylab="Sensitivity")
```



Let us figure out the classifier such that False Positive Rate is less than 0.1 and True Positive Rate is as high as possible

```
#coords(fit1.roc, "best", ret = "threshold") This gives the best threshold value (will not be used here
coord_list <- list()
coord_list[[1]] <- coords(fit1.roc, x = "all")
coord_list[[1]]
```

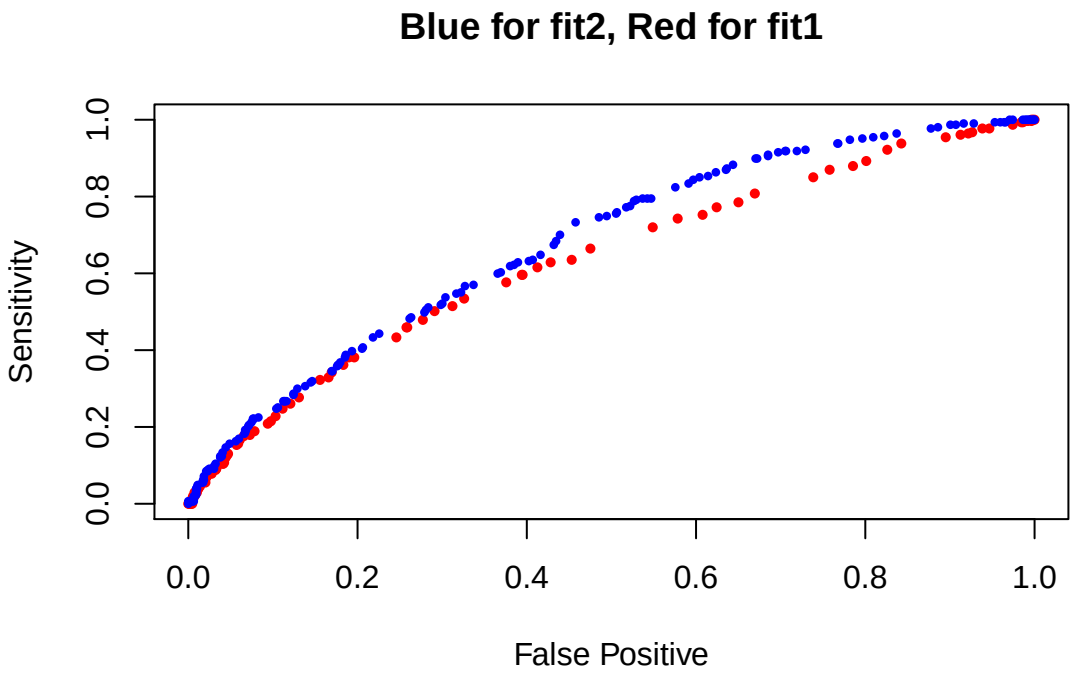
```
##              all         all         all         all         all
## threshold   -Inf 0.101257641 0.107084595 0.110146435 0.113284712
## specificity    0 0.001841621 0.003683241 0.003683241 0.008287293
## sensitivity    1 1.000000000 1.000000000 0.996742671 0.996742671
##                     all        all        all        all        all
## threshold   0.116500696 0.11895694 0.12062431 0.12317078 0.12662733
## specificity 0.009208103 0.01104972 0.01381215 0.01565378 0.02578269
## sensitivity 0.996742671 0.99674267 0.99348534 0.99348534 0.98697068
##                    all        all        all        all        all
## threshold   0.1301665 0.13378939 0.13655419 0.13842945 0.14129099
## specificity 0.0534070 0.06169429 0.07366483 0.07826888 0.08747698
## sensitivity 0.9771987 0.97719870 0.96742671 0.96416938 0.96091205
##                    all       all       all       all       all       all
## threshold   0.1451718 0.1491408 0.1531988 0.1562928 0.1583894 0.1615856
## specificity 0.1049724 0.1574586 0.1740331 0.1988950 0.2145488 0.2421731
## sensitivity 0.9543974 0.9381107 0.9218241 0.8925081 0.8794788 0.8697068
##                    all       all       all       all       all       all
## threshold   0.1659162 0.1703392 0.1748555 0.1782954 0.1806239 0.1841700
## specificity 0.2615101 0.3305709 0.3499079 0.3756906 0.3922652 0.4217311
```

```
## sensitivity 0.8501629 0.8078176 0.7850163 0.7719870 0.7524430 0.7426710
##                   all       all       all       all       all       all
## threshold   0.1889694 0.1938642 0.1988546 0.2026514 0.2052186 0.2078099
## specificity 0.4511971 0.5248619 0.5469613 0.5718232 0.5874770 0.6049724
## sensitivity 0.7198697 0.6644951 0.6351792 0.6286645 0.6156352 0.5960912
##                   all       all       all       all       all       all
## threshold   0.2104252 0.2144023 0.2197773 0.2252484 0.2294057 0.2322131
## specificity 0.6058932 0.6243094 0.6740331 0.6878453 0.7090239 0.7228361
## sensitivity 0.5960912 0.5765472 0.5342020 0.5146580 0.5016287 0.4788274
##                   all       all       all       all       all       all
## threshold   0.2350444 0.2378996 0.2422357 0.2480881 0.2540345 0.2585468
## specificity 0.7412523 0.7421731 0.7541436 0.8038674 0.8103131 0.8167587
## sensitivity 0.4592834 0.4592834 0.4332248 0.3811075 0.3811075 0.3615635
##                   all       all       all       all       all       all
## threshold   0.2615898 0.2662060 0.2724292 0.2787427 0.2851452 0.2899965
## specificity 0.8305709 0.8342541 0.8443831 0.8692449 0.8793738 0.8885820
## sensitivity 0.3420195 0.3289902 0.3224756 0.2768730 0.2605863 0.2475570
##                   all       all       all       all       all       all
## threshold   0.2932632 0.2982114 0.3048720 0.3116153 0.3184394 0.3236021
## specificity 0.8968692 0.9023941 0.9060773 0.9217311 0.9272560 0.9346225
## sensitivity 0.2280130 0.2149837 0.2084691 0.1889251 0.1791531 0.1758958
##                   all       all       all       all       all       all
## threshold   0.3270730 0.3323220 0.3393760 0.3465020 0.3536976 0.3591322
## specificity 0.9392265 0.9410681 0.9429098 0.9530387 0.9539595 0.9558011
## sensitivity 0.1661238 0.1563518 0.1530945 0.1302932 0.1270358 0.1205212
##                   all       all       all        all        all        all
## threshold   0.3627798 0.3682866 0.3756744 0.38312062 0.39062204 0.39627780
## specificity 0.9576427 0.9585635 0.9594843 0.96685083 0.96685083 0.96777164
## sensitivity 0.1074919 0.1042345 0.1042345 0.09446254 0.09120521 0.08794788
##                   all        all       all        all        all
## threshold   0.40006716 0.40577782 0.4134256 0.42111526 0.42884344
## specificity 0.96961326 0.97053407 0.9723757 0.97790055 0.97790055
## sensitivity 0.08794788 0.08794788 0.0781759 0.07491857 0.07166124
##                   all        all        all        all        all
## threshold   0.43465963 0.43854940 0.44831790 0.46597603 0.47778528
## specificity 0.97882136 0.97974217 0.97974217 0.98618785 0.98802947
## sensitivity 0.06840391 0.06514658 0.05537459 0.04560261 0.04560261
##                   all        all        all        all        all
## threshold   0.48370720 0.49160854 0.50346653 0.51532192 0.52715437
## specificity 0.98802947 0.98802947 0.98987109 0.99079190 0.99263352
## sensitivity 0.04234528 0.03908795 0.02931596 0.02931596 0.02931596
##                   all        all        all         all         all
## threshold   0.54679130 0.56631246 0.59314953 0.630625886 0.659680208
## specificity 0.99263352 0.99447514 0.99539595 0.995395948 0.995395948
## sensitivity 0.01954397 0.01954397 0.01302932 0.006514658 0.003257329
##                   all       all       all       all       all all
## threshold   0.6772407 0.6942211 0.7107897 0.7236149 0.7390850 Inf
## specificity 0.9953959 0.9963168 0.9972376 0.9981584 0.9990792   1
## sensitivity 0.0000000 0.0000000 0.0000000 0.0000000 0.0000000   0
```

For False Positive Rate to be less than 0.1, Specificity has to be more than 0.9. Based on the above table, we can see that for a Specificity of 0.9, the classifier has a threhold of 0.298. Any classifier chosen greater than this will increase specificity, but decrease sensitivity.

b. Overlay two ROC curves: one from `fit1`, the other from `fit2`. Does one curve always contain the other curve? Is the AUC of one curve always larger than the AUC of the other one? Why or why not?

```
fit2.roc<- roc(hd_data.f$HD, fit2$fitted)
plot(1-fit1.roc$specificities, fit1.roc$sensitivities, col="red", pch=16, cex=.7,
     xlab="False Positive",
     ylab="Sensitivity")
points(1-fit2.roc$specificities, fit2.roc$sensitivities, col="blue", pch=16, cex=.6)
title("Blue for fit2, Red for fit1")
```



fit2 curve (blue) contains the fit1 curve (red). Hence AUC for fit2 curve is more than AUC for fit1. This is because for any threshold value, fit2 offers a better predictive power than fit1 (as we observed previously through AIC values and other tests). Thus, for any given value of False Positive, fit2 always gives a higher True Positive rate.

c. Estimate the Positive Prediction Values and Negative Prediction Values for `fit1` and `fit2` using .5 as a threshold. Which model is more desirable if we prioritize the Positive Prediction values?

Preparing the Confusion Matrix for fit1 and fit2:

```
fit1.pred <- ifelse(fit1$fitted > 1/2, "1", "0")
fit2.pred <- ifelse(fit2$fitted > 1/2, "1", "0")
fit1.confmat <- table(fit1.pred, hd_data.f$HD)
fit2.confmat <- table(fit2.pred, hd_data.f$HD)
fit1.confmat
```

```
##
## fit1.pred    0    1
##         0 1075  298
##         1   11    9
```

```
fit2.confmat
```

```
##
## fit2.pred    0    1
##         0 1067  290
##         1   19   17
```

```
fit1.pospre <- fit1.confmat[2,2]/(fit1.confmat[2,1]+fit1.confmat[2,2])
fit1.negpre <- fit1.confmat[1,1]/(fit1.confmat[1,1]+fit1.confmat[1,2])
fit2.pospre <- fit2.confmat[2,2]/(fit2.confmat[2,1]+fit2.confmat[2,2])
fit2.negpre <- fit2.confmat[1,1]/(fit2.confmat[1,1]+fit2.confmat[1,2])
fit1.pospre
```

```
## [1] 0.45
```

```
fit1.negpre
```

```
## [1] 0.782957
```

```
fit2.pospre
```

```
## [1] 0.4722222
```

```
fit2.negpre
```

```
## [1] 0.7862933
```

As can be seen from above, fit2 is more desirable if we prioritize positive prediction values since Positive Prediction rate for fit2 is 0.47, which is superior than the Positive Prediction rate of fit1, which is 0.45.

d. (Optional/extra credit) For `fit1`: overlay two curves, but put the threshold over the probability function as the x-axis and positive prediction values and the negative prediction values as the y-axis. Overlay the same plot for `fit2`. Which model would you choose if the set of positive and negative prediction values are the concerns? If you can find an R package to do so, you may use it directly.

Part 3 - Bayes Rule

Bayes rules with risk ratio $\frac{a_{10}}{a_{01}} = 10$ or $\frac{a_{10}}{a_{01}} = 1$. Use your final model obtained from 1 B) to build a class of linear classifiers.

a. Write down the linear boundary for the Bayes classifier if the risk ratio of $a_{10}/a_{01} = 10$.

```r
#Copying final model from 1B for reference
fit2 <- glm(HD~SBP + SEX, hd_data.f, family=binomial)
summary(fit2)
```

```
##
## Call:
## glm(formula = HD ~ SBP + SEX, family = binomial, data = hd_data.f)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.6408  -0.7373  -0.5726  -0.4169   2.2452
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -4.570256   0.389727 -11.727  < 2e-16 ***
## SBP          0.018717   0.002324   8.053 8.07e-16 ***
## SEXMALE      0.903420   0.139762   6.464 1.02e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1469.3  on 1392  degrees of freedom
## Residual deviance: 1373.8  on 1390  degrees of freedom
## AIC: 1379.8
##
## Number of Fisher Scoring iterations: 4
```

In this case, $\frac{a_{0,1}}{a_{1,0}} = \frac{1}{10} = 0.1$, therefore $prob(Y = 1|x) > \frac{0.1}{(1+0.1)} = 0.09$ and $logit > \log(\frac{0.09}{0.91}) = -2.31$

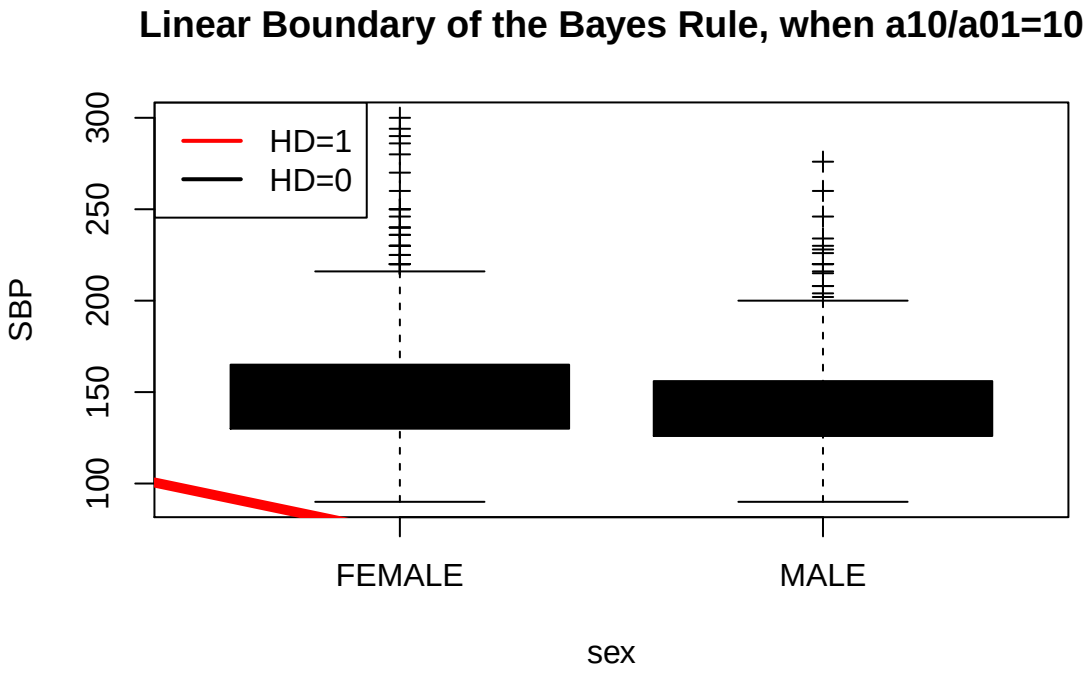
For our fit model, $-4.570 + 0.0187SBP + .9034Sex \geq -2.31$

$0.0187SBP + .9034Sex \geq -2.31 + 4.570$

$SBP \geq -48.31Sex + 120.855$

This is the linear boundary

```r
#drawing the linear boundary
plot(hd_data.f$SEX, hd_data.f$SBP, col=hd_data.f$HD,
     pch=as.numeric(hd_data.f$HD)+2,
     xlab="sex", ylab="SBP")
legend("topleft", legend=c("HD=1", "HD=0"),
       lty=c(1,1), lwd=c(2,2), col=c("red", "black"))
abline(a=120.85, b=-48.31, lwd=5, col="red")
title("Linear Boundary of the Bayes Rule, when a10/a01=10")
```



b. What is your estimated weighted misclassification error for this given risk ratio?

```
fit2.pred.bayes <- rep("0", 1406)
fit2.pred.bayes[fit2$fitted > .09] = "1"
fit2.pred.bayes <- as.factor(ifelse(fit2$fitted > .09, "1", "0"))
MCE.bayes=(sum(5*(fit2.pred.bayes[hd_data.f$HD == "1"] != "1"))
           + sum(fit2.pred.bayes[hd_data.f$HD == "0"] != "0"))/length(hd_data.f$HD)
MCE.bayes
```

```
## [1] 0.7343862
```

c. Recall Liz, our patient from part 1. How would you classify her under this classifier?

First we need to calculate what will be the predicted value for Liz under the fit2 model, then use the classifier to classify her accordingly

```
fit2.liz <- -4.570256 + (0.018717*110) + (0.903420*0)
fit2.liz <- as.factor(ifelse(fit2.liz > .09, "1", "0"))
  fit2.liz
```

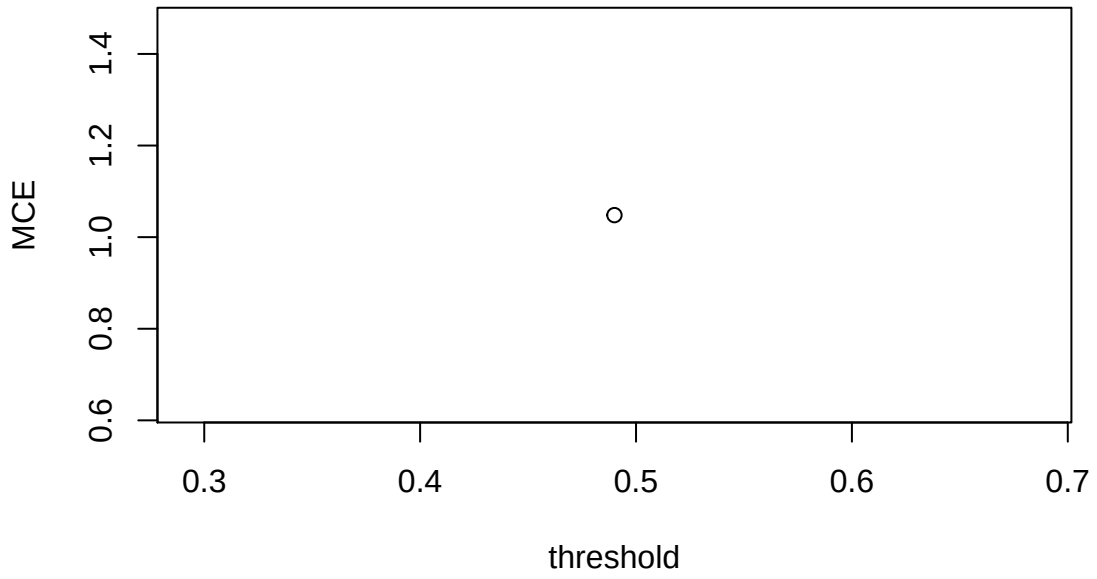
```
## [1] 0
## Levels: 0
```

She would be classified as not at risk (level 0) under this classifier

Now, draw two estimated curves where x = posterior threshold, and y = misclassification errors, corresponding to the thresholding rule given in x-axis.

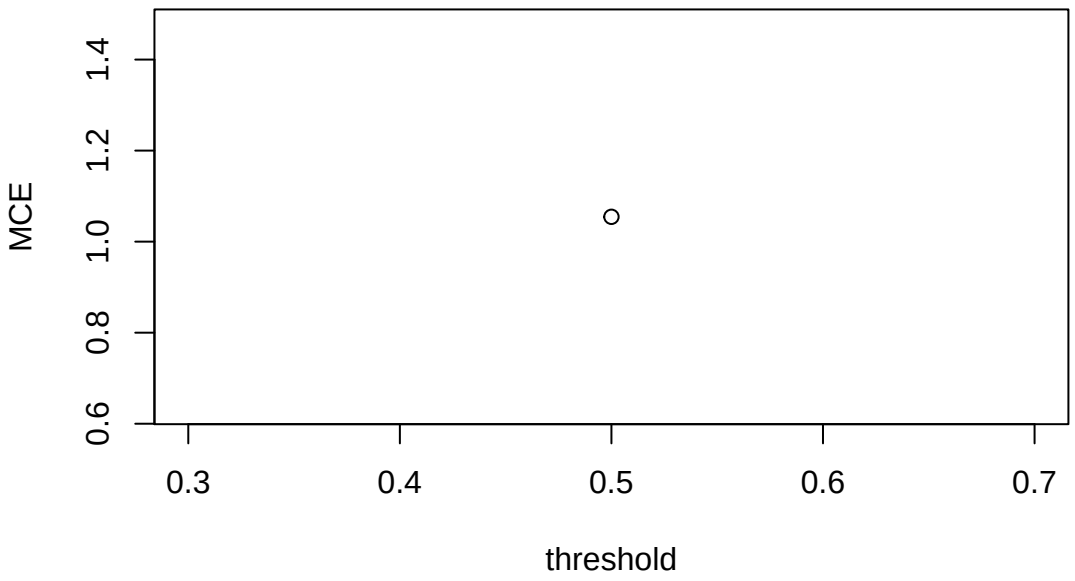
d. Use weighted misclassification error, and set $a_{10}/a_{01} = 10$. How well does the Bayes rule classifier perform?

```
i <- 0.09
while (i <.5) {

fit2.pred.bayes <- rep("0", 1406)
fit2.pred.bayes[fit2$fitted > i] = "1"
fit2.pred.bayes <- as.factor(ifelse(fit2$fitted > i, "1", "0"))
MCE.bayes<-(sum(5*(fit2.pred.bayes[hd_data.f$HD == "1"] != "1"))
           + sum(fit2.pred.bayes[hd_data.f$HD == "0"] != "0"))/length(hd_data.f$HD)
MCE.i <- i
i <- i+0.01
}

plot(MCE.bayes ~ MCE.i,xlab="threshold", ylab="MCE" )
```



e. Use weighted misclassification error, and set $a_{10}/a_{01} = 1$. How well does the Bayes rule classifier perform?

```
fit2.pred.bayes <- rep("0", 1406)
fit2.pred.bayes[fit2$fitted > .5] = "1"
fit2.pred.bayes <- as.factor(ifelse(fit2$fitted > .5, "1", "0"))
MCE.bayes<-(sum(5*(fit2.pred.bayes[hd_data.f$HD == "1"] != "1"))
           + sum(fit2.pred.bayes[hd_data.f$HD == "0"] != "0"))/length(hd_data.f$HD)

plot(0.5,MCE.bayes,xlab="threshold", ylab="MCE" )
```



Problem 2

How well can we predict whether a bill will be passed by the legislature?

Hundreds to thousands of bills are written each year in Pennsylvania. Some are long, others are short. Most of the bills do not even get to be voted on ("sent to the floor"). The chamber meets for 2-year sessions. Bills that are not voted on before the end of the session (or which are voted on but lose the vote) are declared dead. Most bills die. In this study we examine about 8000 bills proposed since 2009, with the goal of building a classifier which has decent power to forecast which bills are likely to be passed.

We have available some information about 8011 bills pertaining to legislation introduced into the Pennsylvania House of Representatives. The goal is to predict which proposals will pass the House. Here is some information about the data:

The response is the variable called `status`. `Bill:passed` means that the bill passed the House; `governor:signed` means that the bill passed both chambers (including the House) and was enacted into law; `governor:received` means that the bill has passed both chambers and was placed before the governor for consideration. All three of these statuses signify a success or a PASS (Meaning that the legislature passed the bill. This does not require it becoming law). All other outcomes are failures.

Here are the rest of the columns:

- `Session` : in which legislative session was the bill introduced
- `Sponsor_party` : the party of the legislator who sponsored the bill (every bill has a sponsor)
- `Bill_id` : of the form HB-[bill number]-[session], e.g., `HB-2661-2013-2014` for the 2661st House Bill introduced in the 2013-2014 session.
- `Num_cosponsors` : how many legislators cosponsored the bill
- `Num_d_cosponsors` : how many Democrats cosponsored the bill
- `Num_r_cosponsors` : how many Republicans cosponsored the bill
- `Title_word_count` : how many words are in the bill's title
- `Originating_committee` : most bills are sent ("referred") to a committee of jurisdiction (like the

transportation committee, banking & insurance committee, agriculture & rural affairs committee) where they are discussed and amended. The originating committee is the committee to which a bill is referred.
- `Day_of_week_introduced` : on what day the bill was introduced in the House (1 is Monday)
- `Num_amendments` : how many amendments the bill has
- `Is_sponsor_in_leadership` : does the sponsor of the bill hold a position inside the House (such as speaker, majority leader, etc.)
- `num_originating_committee_cosponsors` : how many cosponsors sit on the committee to which the bill is referred
- `num_originating_committee_cosponsors_r` : how many Republican cosponsors sit on the committee to which the bill is referred
- `num_originating_committee_cosponsors_d` - how many Democratic cosponsors sit on the committee to which the bill is referred

The data you can use to build the classifier is called `Bills.subset`. It contains 7011 records from the full data set. I took a random sample of 1000 bills from the 2013-2014 session as testing data set in order to test the quality of your classifier, it is called `Bills.subset.test.`

Your job is to choose a best set of classifiers such that

- The testing ROC curve pushes to the upper left corner the most, and has a competitive AUC value.
- Propose a reasonable loss function, and report the Bayes rule together with its weighted MIC.
- You may also create some sensible variables based on the predictors or make other transformations to improve the performance of your classifier.

Here is what you need to report:

1. Write a summary about the goal of the project. Give some background information. If desired, you may go online to find out more information.

The objective of this project is to create a model which can predict a legislative bill's likelihood of passing. This information would be valuable to have (a) at the time the bill is introduced, to see how optomistic we should be about its success, or (b) before the bill is even introduced, to try to create a more effective environment within which the bill could be introduced. For example, if our model indicates that introducing a bill on a Tuesday is more likely to lead to success than introducing the bill on a Monday, then the necessary preparations can be made for a Tuesday introduction.

The model's implications could even be valuable for bills already under consideration. For example, if bipartisan support for a bill makes it much more likely to pass, then the bill sponsors could look for ways to attract support from the opposing party.

This project should be of interest not just to legislators, but to involved citizens across the country. Applying analytics to the legislative process will yield interesting insights which could give one party or cause a leg up if they are able to apply the model implications to real-world issues (e.g. via lobbying).

2. Give a preliminary summary of the data.

```
# Read in the data
bills_train <- read.csv("Bills.subset.csv")
bills_test <- read.csv("Bills.subset.test.csv")

# View variable names
names(bills_train)
```

```
##  [1] "bill_id"
##  [2] "sponsor_party"
##  [3] "session"
##  [4] "num_cosponsors"
##  [5] "num_d_cosponsors"
##  [6] "num_r_cosponsors"
```

```
##  [7] "title_word_count"
##  [8] "originating_committee"
##  [9] "day.of.week.introduced"
## [10] "num_amendments"
## [11] "status"
## [12] "is_sponsor_in_leadership"
## [13] "num_originating_committee_cosponsors"
## [14] "num_originating_committee_cosponsors_r"
## [15] "num_originating_committee_cosponsors_d"
```

```
# Simple overview of variable components
str(bills_train)
```

```
## 'data.frame':    7011 obs. of  15 variables:
##  $ bill_id                              : Factor w/ 7011 levels "HB-1-2009-2010",..: 5591 3720 4699
##  $ sponsor_party                        : Factor w/ 3 levels "","Democratic",..: 3 2 2 2 2 2 3 2 3 3
##  $ session                              : Factor w/ 4 levels "2009-2010","2009-2010 Special Session
##  $ num_cosponsors                       : int  32 0 9 30 4 0 34 4 9 61 ...
##  $ num_d_cosponsors                     : int  0 0 6 24 4 0 14 3 2 22 ...
##  $ num_r_cosponsors                     : int  32 0 3 6 0 0 20 1 7 39 ...
##  $ title_word_count                     : int  27 50 20 25 29 30 32 25 20 46 ...
##  $ originating_committee                : Factor w/ 26 levels "","PAC000001",..: 1 6 3 11 3 8 1 3 1
##  $ day.of.week.introduced               : int  3 5 1 1 1 3 5 2 1 4 ...
##  $ num_amendments                       : int  0 0 0 1 0 0 2 0 0 1 ...
##  $ status                               : Factor w/ 10 levels "","amendment:passed",..: 8 10 8 8 8 8
##  $ is_sponsor_in_leadership             : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ num_originating_committee_cosponsors : int  0 0 3 3 0 0 0 0 0 0 ...
##  $ num_originating_committee_cosponsors_r: int  0 0 1 1 0 0 0 0 0 0 ...
##  $ num_originating_committee_cosponsors_d: int  0 0 2 2 0 0 0 0 0 0 ...
```

```
# Summary of variables
summary(bills_train)
```

```
##                                                bill_id
##  HB-1-2009-2010                                    :   1
##  HB-1-2009-2010 Special Session #1 (Transportation) :   1
##  HB-1-2011-2012                                    :   1
##  HB-10-2009-2010                                   :   1
##  HB-10-2009-2010 Special Session #1 (Transportation):   1
##  HB-10-2011-2012                                   :   1
##  (Other)                                           :7005
##     sponsor_party                                          session
##            : 102   2009-2010                                  :2787
##  Democratic:3213   2009-2010 Special Session #1 (Transportation):  23
##  Republican:3696   2011-2012                                  :2709
##                    2013-2014                                  :1492
##
##
##
##  num_cosponsors  num_d_cosponsors num_r_cosponsors title_word_count
##  Min.   :  0.0   Min.   : 0.00    Min.   : 0.00    Min.   :  6.00
##  1st Qu.: 11.0   1st Qu.: 3.00    1st Qu.: 3.00    1st Qu.: 22.00
##  Median : 20.0   Median : 8.00    Median : 8.00    Median : 27.00
##  Mean   : 23.5   Mean   :10.43    Mean   :13.08    Mean   : 33.95
##  3rd Qu.: 32.0   3rd Qu.:15.00    3rd Qu.:19.00    3rd Qu.: 34.00
##  Max.   :165.0   Max.   :90.00    Max.   :99.00    Max.   :751.00
```

```
##
##  originating_committee day.of.week.introduced num_amendments
##  PAC000004:1047        Min.   :1.000          Min.   :0.0000
##  PAC000012: 810        1st Qu.:2.000          1st Qu.:0.0000
##  PAC000001: 677        Median :3.000          Median :0.0000
##  PAC000016: 622        Mean   :2.724          Mean   :0.1774
##  PAC000017: 526        3rd Qu.:4.000          3rd Qu.:0.0000
##  PAC000007: 329        Max.   :6.000          Max.   :8.0000
##  (Other)  :3000        NA's   :5
##                 status     is_sponsor_in_leadership
##  committee:referred:6113   Min.   :0.0000
##  governor:signed   : 428   1st Qu.:0.0000
##  bill:reading:1    : 234   Median :1.0000
##  committee:passed  : 106   Mean   :0.5904
##  bill:reading:2    :  73   3rd Qu.:1.0000
##  bill:passed       :  31   Max.   :1.0000
##  (Other)           :  26
##  num_originating_committee_cosponsors
##  Min.   : 0.00
##  1st Qu.: 0.00
##  Median : 1.00
##  Mean   : 2.05
##  3rd Qu.: 3.00
##  Max.   :19.00
##
##  num_originating_committee_cosponsors_r
##  Min.   : 0.000
##  1st Qu.: 0.000
##  Median : 1.000
##  Mean   : 1.342
##  3rd Qu.: 2.000
##  Max.   :14.000
##
##  num_originating_committee_cosponsors_d
##  Min.   :0.0000
##  1st Qu.:0.0000
##  Median :0.0000
##  Mean   :0.7082
##  3rd Qu.:1.0000
##  Max.   :8.0000
##
```

```
# Create the target variable -- pass or fail
bills_train$pass <- 0
bills_train$pass[bills_train$status == 'bill:passed'] <- 1
bills_train$pass[bills_train$status == 'governor:signed'] <- 1
bills_train$pass[bills_train$status == 'governor:received'] <- 1
summary(bills_train$pass)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## 0.00000 0.00000 0.00000 0.06604 0.00000 1.00000
```

```
bills_test$pass <- 0
bills_test$pass[bills_test$status == 'bill:passed'] <- 1
bills_test$pass[bills_test$status == 'governor:signed'] <- 1
```

```
bills_test$pass[bills_test$status == 'governor:received'] <- 1
summary(bills_test$pass)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   0.000   0.000   0.068   0.000   1.000
```

By looking at the mean of the target "pass" variable that we have created, we see that only 6.6% of the bills which are introduced end up passing.

3. Based on the data available to you, you need to build a classifier. Provide the following information:
   - The process of building your classifier
   - Methods explored, and why you chose your final model
   - Did you use a training and test set to build your classifier using the training data? If so, describe the process including information about the size of your training and test sets.
   - What is the criterion being used to build your classifier?
   - How do you estimate the quality of your classifier?

```
# fit cosponsor variables
fit1 <- glm(pass ~ num_cosponsors + num_d_cosponsors + num_r_cosponsors + num_originating_committee_cosp
summary(fit1) # show results
```

```
##
## Call:
## glm(formula = pass ~ num_cosponsors + num_d_cosponsors + num_r_cosponsors +
##     num_originating_committee_cosponsors + num_originating_committee_cosponsors_r +
##     num_originating_committee_cosponsors_d, family = binomial(),
##     data = bills_train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.0246  -0.3809  -0.3343  -0.3070   2.7151
##
## Coefficients: (2 not defined because of singularities)
##                                        Estimate Std. Error z value
## (Intercept)                           -3.031245   0.082399 -36.787
## num_cosponsors                         0.011941   0.004484   2.663
## num_d_cosponsors                      -0.023225   0.008222  -2.825
## num_r_cosponsors                             NA         NA      NA
## num_originating_committee_cosponsors   0.167777   0.050472   3.324
## num_originating_committee_cosponsors_r -0.054925   0.065438  -0.839
## num_originating_committee_cosponsors_d       NA         NA      NA
##                                       Pr(>|z|)
## (Intercept)                            < 2e-16 ***
## num_cosponsors                        0.007738 **
## num_d_cosponsors                      0.004730 **
## num_r_cosponsors                            NA
## num_originating_committee_cosponsors  0.000887 ***
## num_originating_committee_cosponsors_r 0.401281
## num_originating_committee_cosponsors_d       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 3411.1  on 7010  degrees of freedom
```

```
## Residual deviance: 3322.5  on 7006  degrees of freedom
## AIC: 3332.5
##
## Number of Fisher Scoring iterations: 5
```

```
# fit non-cosponsor variables
fit2 <- glm(pass ~ sponsor_party + session + title_word_count + originating_committee + day.of.week.int
summary(fit2) # show results
```

```
##
## Call:
## glm(formula = pass ~ sponsor_party + session + title_word_count +
##     originating_committee + day.of.week.introduced + num_amendments +
##     is_sponsor_in_leadership, family = binomial(), data = bills_train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.1040  -0.2773  -0.1813  -0.1219   3.1777
##
## Coefficients:
##                                                    Estimate Std. Error
## (Intercept)                                       -6.553496   1.170822
## sponsor_partyDemocratic                            1.141101   1.070369
## sponsor_partyRepublican                            1.955559   1.066506
## session2009-2010 Special Session #1 (Transportation) -11.349491 298.079519
## session2011-2012                                   1.035794   0.424721
## session2013-2014                                   1.038678   0.434490
## title_word_count                                   0.004720   0.001169
## originating_committeePAC000001                     1.887603   0.493695
## originating_committeePAC000004                     0.607766   0.501670
## originating_committeePAC000005                     1.245614   0.554099
## originating_committeePAC000007                     1.536742   0.513295
## originating_committeePAC000008                     2.993151   0.508659
## originating_committeePAC000010                     0.029338   0.622444
## originating_committeePAC000012                     0.115505   0.532820
## originating_committeePAC000015                     1.660406   0.590676
## originating_committeePAC000016                    -0.550298   0.607834
## originating_committeePAC000017                    -0.574121   0.617079
## originating_committeePAC000019                     1.117554   0.598886
## originating_committeePAC000035                     0.788729   0.659192
## originating_committeePAC000040                    -1.397454   0.851893
## originating_committeePAC000099                     0.827584   0.660587
## originating_committeePAC000129                     1.124096   0.547253
## originating_committeePAC000141                     0.090606   0.634912
## originating_committeePAC000165                   -11.179088 414.857019
## originating_committeePAC000190                     1.306243   0.607048
## originating_committeePAC000215                     2.135535   0.685683
## originating_committeePAC000229                     1.733595   0.550091
## originating_committeePAC000244                    -0.842565   0.876238
## originating_committeePAC000246                     0.889633   0.622370
## originating_committeePAC000263                     0.415435   0.649253
## originating_committeePAC000264                     0.353089   0.595957
## originating_committeePAC000265                    -0.665699   1.107935
## day.of.week.introduced                             0.014022   0.044391
## num_amendments                                     1.845034   0.083089
```

```
## is_sponsor_in_leadership                                -0.519720   0.411673
##                                                        z value Pr(>|z|)
## (Intercept)                                             -5.597 2.18e-08 ***
## sponsor_partyDemocratic                                  1.066 0.286386
## sponsor_partyRepublican                                  1.834 0.066711 .
## session2009-2010 Special Session #1 (Transportation)    -0.038 0.969628
## session2011-2012                                         2.439 0.014738 *
## session2013-2014                                         2.391 0.016822 *
## title_word_count                                         4.037 5.40e-05 ***
## originating_committeePAC000001                           3.823 0.000132 ***
## originating_committeePAC000004                           1.211 0.225710
## originating_committeePAC000005                           2.248 0.024576 *
## originating_committeePAC000007                           2.994 0.002755 **
## originating_committeePAC000008                           5.884 4.00e-09 ***
## originating_committeePAC000010                           0.047 0.962407
## originating_committeePAC000012                           0.217 0.828379
## originating_committeePAC000015                           2.811 0.004938 **
## originating_committeePAC000016                          -0.905 0.365284
## originating_committeePAC000017                          -0.930 0.352171
## originating_committeePAC000019                           1.866 0.062034 .
## originating_committeePAC000035                           1.197 0.231498
## originating_committeePAC000040                          -1.640 0.100920
## originating_committeePAC000099                           1.253 0.210278
## originating_committeePAC000129                           2.054 0.039969 *
## originating_committeePAC000141                           0.143 0.886523
## originating_committeePAC000165                          -0.027 0.978502
## originating_committeePAC000190                           2.152 0.031413 *
## originating_committeePAC000215                           3.114 0.001843 **
## originating_committeePAC000229                           3.151 0.001625 **
## originating_committeePAC000244                          -0.962 0.336265
## originating_committeePAC000246                           1.429 0.152881
## originating_committeePAC000263                           0.640 0.522260
## originating_committeePAC000264                           0.592 0.553533
## originating_committeePAC000265                          -0.601 0.547942
## day.of.week.introduced                                   0.316 0.752096
## num_amendments                                          22.206  < 2e-16 ***
## is_sponsor_in_leadership                                -1.262 0.206783
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 3410.5  on 7005  degrees of freedom
## Residual deviance: 2227.9  on 6971  degrees of freedom
##   (5 observations deleted due to missingness)
## AIC: 2297.9
##
## Number of Fisher Scoring iterations: 14
```

```r
# fit all variables
fit3 <- glm(pass ~ sponsor_party + session + num_cosponsors + num_d_cosponsors + num_r_cosponsors + titl
summary(fit3) # show results
```

```
##
## Call:
```

```
## glm(formula = pass ~ sponsor_party + session + num_cosponsors +
##     num_d_cosponsors + num_r_cosponsors + title_word_count +
##     originating_committee + day.of.week.introduced + num_amendments +
##     is_sponsor_in_leadership + num_originating_committee_cosponsors +
##     num_originating_committee_cosponsors_r + num_originating_committee_cosponsors_d,
##     family = binomial(), data = bills_train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.1085  -0.2793  -0.1854  -0.1237   3.2259
##
## Coefficients: (2 not defined because of singularities)
##                                                    Estimate Std. Error
## (Intercept)                                       -6.737e+00  1.184e+00
## sponsor_partyDemocratic                            1.098e+00  1.075e+00
## sponsor_partyRepublican                            1.968e+00  1.067e+00
## session2009-2010 Special Session #1 (Transportation) -1.121e+01  2.975e+02
## session2011-2012                                   9.717e-01  4.273e-01
## session2013-2014                                   9.738e-01  4.388e-01
## num_cosponsors                                    -9.591e-04  6.500e-03
## num_d_cosponsors                                   1.208e-02  1.189e-02
## num_r_cosponsors                                          NA         NA
## title_word_count                                   4.787e-03  1.172e-03
## originating_committeePAC000001                     1.903e+00  5.089e-01
## originating_committeePAC000004                     6.235e-01  5.255e-01
## originating_committeePAC000005                     1.133e+00  5.787e-01
## originating_committeePAC000007                     1.633e+00  5.286e-01
## originating_committeePAC000008                     3.089e+00  5.395e-01
## originating_committeePAC000010                     4.311e-02  6.345e-01
## originating_committeePAC000012                     1.799e-01  5.493e-01
## originating_committeePAC000015                     1.632e+00  6.198e-01
## originating_committeePAC000016                    -5.563e-01  6.270e-01
## originating_committeePAC000017                    -5.834e-01  6.319e-01
## originating_committeePAC000019                     1.107e+00  6.328e-01
## originating_committeePAC000035                     7.666e-01  6.650e-01
## originating_committeePAC000040                    -1.299e+00  8.596e-01
## originating_committeePAC000099                     8.623e-01  6.795e-01
## originating_committeePAC000129                     1.091e+00  5.822e-01
## originating_committeePAC000141                     1.083e-01  6.470e-01
## originating_committeePAC000165                    -1.107e+01  4.147e+02
## originating_committeePAC000190                     1.366e+00  6.216e-01
## originating_committeePAC000215                     2.108e+00  7.121e-01
## originating_committeePAC000229                     1.698e+00  5.851e-01
## originating_committeePAC000244                    -8.255e-01  8.938e-01
## originating_committeePAC000246                     9.554e-01  6.319e-01
## originating_committeePAC000263                     3.793e-01  6.664e-01
## originating_committeePAC000264                     3.554e-01  6.210e-01
## originating_committeePAC000265                    -7.953e-01  1.114e+00
## day.of.week.introduced                             1.505e-02  4.451e-02
## num_amendments                                     1.808e+00  8.427e-02
## is_sponsor_in_leadership                          -4.544e-01  4.137e-01
## num_originating_committee_cosponsors               2.651e-02  7.042e-02
## num_originating_committee_cosponsors_r             8.896e-03  8.818e-02
## num_originating_committee_cosponsors_d                    NA         NA
```

```
##                                                  z value Pr(>|z|)
## (Intercept)                                       -5.688 1.29e-08 ***
## sponsor_partyDemocratic                            1.021 0.307180
## sponsor_partyRepublican                            1.844 0.065133 .
## session2009-2010 Special Session #1 (Transportation)  -0.038 0.969931
## session2011-2012                                   2.274 0.022970 *
## session2013-2014                                   2.219 0.026464 *
## num_cosponsors                                    -0.148 0.882687
## num_d_cosponsors                                   1.016 0.309493
## num_r_cosponsors                                      NA       NA
## title_word_count                                   4.086 4.40e-05 ***
## originating_committeePAC000001                     3.740 0.000184 ***
## originating_committeePAC000004                     1.187 0.235419
## originating_committeePAC000005                     1.958 0.050226 .
## originating_committeePAC000007                     3.090 0.002002 **
## originating_committeePAC000008                     5.727 1.02e-08 ***
## originating_committeePAC000010                     0.068 0.945835
## originating_committeePAC000012                     0.327 0.743332
## originating_committeePAC000015                     2.633 0.008475 **
## originating_committeePAC000016                    -0.887 0.375012
## originating_committeePAC000017                    -0.923 0.355878
## originating_committeePAC000019                     1.749 0.080269 .
## originating_committeePAC000035                     1.153 0.248950
## originating_committeePAC000040                    -1.511 0.130708
## originating_committeePAC000099                     1.269 0.204470
## originating_committeePAC000129                     1.875 0.060854 .
## originating_committeePAC000141                     0.167 0.867071
## originating_committeePAC000165                    -0.027 0.978700
## originating_committeePAC000190                     2.198 0.027984 *
## originating_committeePAC000215                     2.960 0.003080 **
## originating_committeePAC000229                     2.903 0.003698 **
## originating_committeePAC000244                    -0.924 0.355724
## originating_committeePAC000246                     1.512 0.130565
## originating_committeePAC000263                     0.569 0.569239
## originating_committeePAC000264                     0.572 0.567103
## originating_committeePAC000265                    -0.714 0.475452
## day.of.week.introduced                             0.338 0.735210
## num_amendments                                    21.451  < 2e-16 ***
## is_sponsor_in_leadership                          -1.098 0.272074
## num_originating_committee_cosponsors               0.377 0.706545
## num_originating_committee_cosponsors_r             0.101 0.919642
## num_originating_committee_cosponsors_d                NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 3410.5  on 7005  degrees of freedom
## Residual deviance: 2221.4  on 6967  degrees of freedom
##   (5 observations deleted due to missingness)
## AIC: 2299.4
##
## Number of Fisher Scoring iterations: 14
```

To build the classifier, we started by building several glm models using different input variables. We looked at several versions of the model and compared them using AIC. Models which only looked at cosponsor variables had higher AIC than models with additional variables, so we chose to include other variables as well.

The training set was used to build the model, which is then tested on the test data set. Although the test dataset only contains 1000 observations (compared to 7011 in the training dataset), it gives us plenty of information about how our model holds up and that we are not overfitting on the training dataset.

The quality of our classifier is relatively good (e.g. compared to the cosponser model "fit1") as measured by AIC. However, AIC is better as it gets closer to 0, and ours is still fairly high, implying that we could improve the model further. One way to approach this would be to come up with new interaction terms from combinations of variables (e.g. a variable which considers both the day of the week and the number of sponsors).

4. Suggestions you may have: what important features should have been collected which would have helped us to improve the quality of the classifiers.

The model is sound in determining whether the bill will pass, but it does not give us more detailed sense on why certain bills pass, and the probability of certain bills passing. The following additions may help the model become more nuanced:

Incorporate party-specific stance on certain political issues: Certain political issues are more Democrat-leaning than Republican, and vice versa (e.g. Democrats and Republicans fundamentally differ in their views on corporate taxation). Depending on which party was more influential in a given year, and what kind of political issues were in debate in that year, it is possible that the data may provide a skewed view on the number of bills passed. Knowing this correlation and factoring it into the model may help us understand why, and potentially go further in predicting party-specific behavior for similar issues in the future.

Longer time horizon: For above reason, it is important for a well-performing model to capture long enough timeline to cover multitude of scenarios.

Final notes: The data is graciously lent from a friend. It is only meant for you to use in this class. All other uses are prohibited without permission.