

Final Project

Samira Zarandioon and Bradley Robinson

8/10/2018

Introduction

This data analysis report is based on a data set that is collected by the Global Longitudinal Study of Osteoporosis in Women (GLOW). According the official website of the research team (<http://www.outcomes-umassmed.org/glow/>), the goal of this research was to improve understanding of the risk and prevention of osteoporosis-related fractures among female residents of 10 countries who were 55 years of age and older. GLOW enrolled over 60,000 women through over 700 physicians in 10 countries, and conducted annual follow-up for up to 5 years through annual patient questionnaires.

Data Description

The dataset used in this analysis is called GLOW500, which contains 500 observations from six sites in the united states. The target (outcome/response) variable FRACTURE indicates whether the subject had any fracture in the first year of followup. It includes 13 selected potential risk factors for fracture. According to the source of data, given the fact that fracture rate was only 4% these 500 observations are over sampled from 21,000 original observation.

Description of each field in the data set is as follows:

Explanatory Variables:

SUB_ID: Id code for each subject (Categorical)
SITE_ID: Id code for the study site (Categorical)
PHY_ID: Id code for the Physician (Categorical)
PRIORFRAC: Does the subject have history of prior fracture? (1 => Yes, 0 => No) (Bineary)
AGE: Age at enrolment (Numeric)
WEIGHT: Weight at enrollment (Numeric)
HEIGHT: Height at enrollment (Numeric)
BMI: Body mass index (Numeric)
PREMENO: Did the subje had Menopause before age 45? (1 => Yes, 0 => No) (Bineary)
MOMFRAC: Has subject's mother ever had hip fracture? (1 => Yes, 0 => No) (Bineary)
ARMASSIST: Does the subject need arms to stand form a chare? (1 => Yes, 0 => No) (Bineary)
SMOKE: Is the subject a Smoker? (1 => Yes, 0 => No) (Bineary)
RATERISK: Subject's self-reported risk of fractur (1 => Less than others of the same age, 2 => Same as
FRACSCORE: Fracture risk score (Computed based on AGE, WEIGHT, ARMASSIST, SMOKE, MOMFRAC, PRIORFRAC) (1

Response Variable:

FRACTURE: Did subject have any fracture in first year? (1 => Yes, 0 => No) (Bineary)

A summary a of dataset that illustrates distribution of each filed is presented below:

##	SUB_ID	SITE_ID	PHY_ID	PRIORFRAC
##	Min. : 1.0	Min. :1.000	Min. : 1.00	Min. :0.000
##	1st Qu.:125.8	1st Qu.:2.000	1st Qu.: 57.75	1st Qu.:0.000
##	Median :250.5	Median :3.000	Median :182.50	Median :0.000
##	Mean :250.5	Mean :3.436	Mean :178.55	Mean :0.252
##	3rd Qu.:375.2	3rd Qu.:5.000	3rd Qu.:298.00	3rd Qu.:1.000

```

## Max.      :500.0    Max.      :6.000    Max.      :325.00    Max.      :1.000
##      AGE              WEIGHT              HEIGHT              BMI
## Min.      :55.00    Min.      : 39.90    Min.      :134.0    Min.      :14.88
## 1st Qu.   :61.00    1st Qu.   : 59.90    1st Qu.   :157.0    1st Qu.   :23.27
## Median    :67.00    Median    : 68.00    Median    :161.5    Median    :26.42
## Mean      :68.56    Mean      : 71.82    Mean      :161.4    Mean      :27.55
## 3rd Qu.   :76.00    3rd Qu.   : 81.30    3rd Qu.   :165.0    3rd Qu.   :30.79
## Max.      :90.00    Max.      :127.00    Max.      :199.0    Max.      :49.08
##      PREMENO          MOMFRAC          ARMASSIST          SMOKE
## Min.      :0.000    Min.      :0.00    Min.      :0.000    Min.      :0.00
## 1st Qu.   :0.000    1st Qu.   :0.00    1st Qu.   :0.000    1st Qu.   :0.00
## Median    :0.000    Median    :0.00    Median    :0.000    Median    :0.00
## Mean      :0.194    Mean      :0.13    Mean      :0.376    Mean      :0.07
## 3rd Qu.   :0.000    3rd Qu.   :0.00    3rd Qu.   :1.000    3rd Qu.   :0.00
## Max.      :1.000    Max.      :1.00    Max.      :1.000    Max.      :1.00
##      RATERISK          FRACSCORE          FRACTURE
## Min.      :1.00    Min.      : 0.000    Min.      :0.00
## 1st Qu.   :1.00    1st Qu.   : 2.000    1st Qu.   :0.00
## Median    :2.00    Median    : 3.000    Median    :0.00
## Mean      :1.96    Mean      : 3.698    Mean      :0.25
## 3rd Qu.   :3.00    3rd Qu.   : 5.000    3rd Qu.   :0.25
## Max.      :3.00    Max.      :11.000    Max.      :1.00

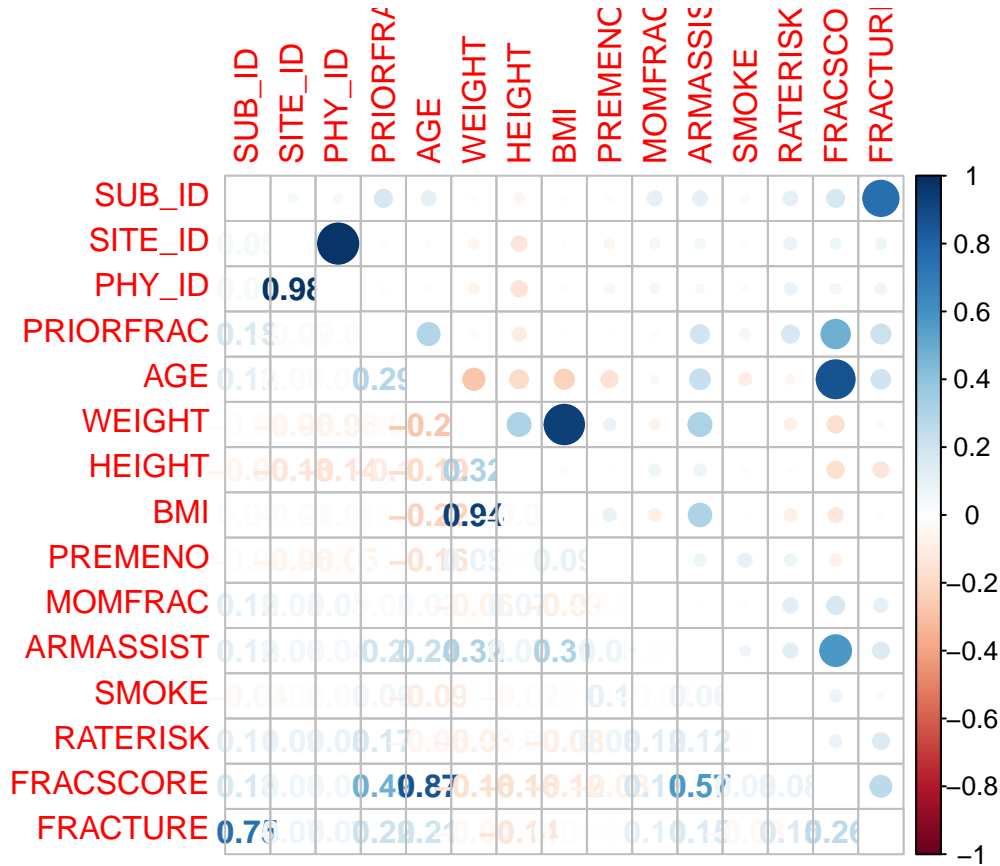
```

Exploratory Analysis

The following graphical representation of correlations between the fields of glow500 datasets, gives us a very good understanding of how the explanatory variables are related to each other and the response variable. For example, it shows high correlation of PHY_ID and SITE_ID. It also shows that BMI is highly correlated to WEIGHT, which is driven by the definition of the BMI. Moreover, it shows that FRACSCORE is highly correlated with AGE, PRIORFRAC, ..., which are used to compute the risk. Correlated variables indicate some level of redundancy in the explanatory variables.

Correlation of explanatory variables with the target variable FRACTURE is quite informative. The plot depicts that PRIORFRAC, AGE, ARMASSIST, and MOMFRAC have relatively high positive correlation to the target variable. However, HEIGHT has negative correlation to the target variable.

```
## corplot 0.84 loaded
```



Performing Principal Component Analysis (PCA), against the numeric explanatory variables also gives us some insight on redundancy and importance of explanatory variables. As the following Scree Plot illustrates only three components can explain over 99% of variance, which indicates some redundancy in these variables that can be justified due to correlation between these variables.

Addressing Objective 1:

We used step-wise parameter selection method for training a logistic regression model which produced the following model:

The corresponding regression formula is as follows:

$$\text{Logit(Probability of Fracture)} = \log(\text{Odds Ratio of Fracture}) = \log\left(\frac{\text{Probability of Fracture}}{1 - \text{Probability of Fracture}}\right) = 3.2477 - 0.0378 * \text{HEIGHT} + 0.4072 * \text{RATERISK} + 0.2244 * \text{FRACSCORE}$$

In other words:

$$\text{Probability of Fracture} = \text{Softmax}(3.2477 - 0.0378 * \text{HEIGHT} + 0.4072 * \text{RATERISK} + 0.2244 * \text{FRACSCORE})$$

The negative coefficient of HEIGHT is consistent with negative correlation that was illustrated by a pink circle in the correlation matrix in the previous section.

The following tables shows how these three variables are selected and their corresponding p-value for selection.

Eigenvectors					
	Prin1	Prin2	Prin3	Prin4	Prin5
AGE	-.494722	0.467421	0.152466	0.716546	-.009160
WEIGHT	0.527303	0.465788	0.088410	0.032402	-.704363
HEIGHT	0.234577	-.081961	0.938232	0.018856	0.240042
BMI	0.474103	0.516152	-.245337	0.051374	0.667821
FRACSCORE	-.444298	0.539841	0.168723	-.694635	0.013601

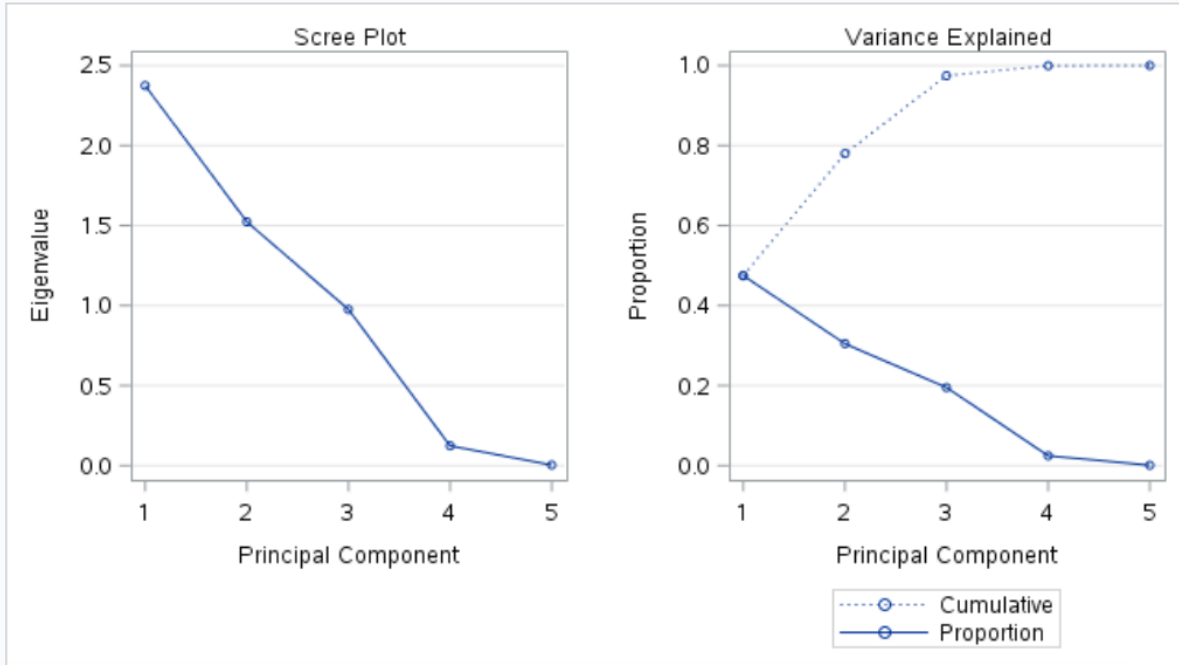


Figure 1: Principal Component Analysis On Numerical Fields

Analysis of Maximum Likelihood Estimates					
Parameter	DF	Estimate	Standard Error	Wald Chi-Square	Pr > ChiSq
Intercept	1	3.2477	2.8517	1.2970	0.2548
HEIGHT	1	-0.0378	0.0175	4.6608	0.0309
RATERISK	1	0.4072	0.1392	8.5576	0.0034
FRACSCORE	1	0.2244	0.0440	25.9729	<.0001

Figure 2:

Summary of Stepwise Selection							
Step	Effect		DF	Number In	Score Chi-Square	Wald Chi-Square	Pr > ChiSq
	Entered	Removed					
1	FRACSCORE		1	1	34.9747		<.0001
2	RATERISK		1	2	8.7541		0.0031
3	HEIGHT		1	3	4.7043		0.0301

Effect of each variable on the odds ratio (point estimate and 95% confidence interval) is presented in the

Odds Ratio Estimates			
Effect	Point Estimate	95% Wald Confidence Limits	
HEIGHT	0.963	0.930	0.997
RATERISK	1.503	1.144	1.974
FRACSCORE	1.252	1.148	1.364

following table:

Model Selection Required

Type of Selection

Any or all: , RIDGE, ELASTIC NET,
Step-wise, Forward, Backward
Manual / Intuition

Checking Assumptions Required

Lack of fit test
Influential point analysis (Cook's D and Leverage)
Optional Residual Plots

Parameter Interpretation

Interpretation Required
Confidence Intervals Required

Final conclusions from the analyses of Objective 1 Required

Addressing Objective 2

Make sure it is clear how many models were created to compete against the one in Objective 1. Make note of any tuning parameters that were used and how you came up with them (knn and random forest logistics)
Required

Main Analysis Content Required Overall report of the error metrics on a test set or CV run. Also if the two best models have error rates of .05 and .045, can we really say that one model is outperforming the other? What other tools that we learned in the second half of this class that could help us get at that?

Conclusion/Discussion Required

The conclusion should reprise the questions and conclusions of objective 2 with recommendations of

Appendix Required

Well commented SAS/R Code Required

Graphics and summary tables (Can be placed in the appendix or in the written report itself.)

References

- Section 1.6.3: [https://ia800206.us.archive.org/7/items/WileySeriesInProbabilityAndStatisticsDavidW.HosmerStanleyLemeshowRodneyX.Sturdiv/\(Wiley%20Series%20in%20Probability%20and%20Statistics\)%20David%20W.%20Hosmer,%20Stanley%20Lemeshow,%20Rodney%20X.%20Sturdivant%20\(auth.\),%20Walter%20A.%20Shewhart,%20Samuel%20S.%20Wilks%20\(eds.\)-Applied%20Logistic%20Regression-Wiley%20\(2013\).pdf](https://ia800206.us.archive.org/7/items/WileySeriesInProbabilityAndStatisticsDavidW.HosmerStanleyLemeshowRodneyX.Sturdiv/(Wiley%20Series%20in%20Probability%20and%20Statistics)%20David%20W.%20Hosmer,%20Stanley%20Lemeshow,%20Rodney%20X.%20Sturdivant%20(auth.),%20Walter%20A.%20Shewhart,%20Samuel%20S.%20Wilks%20(eds.)-Applied%20Logistic%20Regression-Wiley%20(2013).pdf)

```
library(MASS)           # provides LDA & QDA model functions

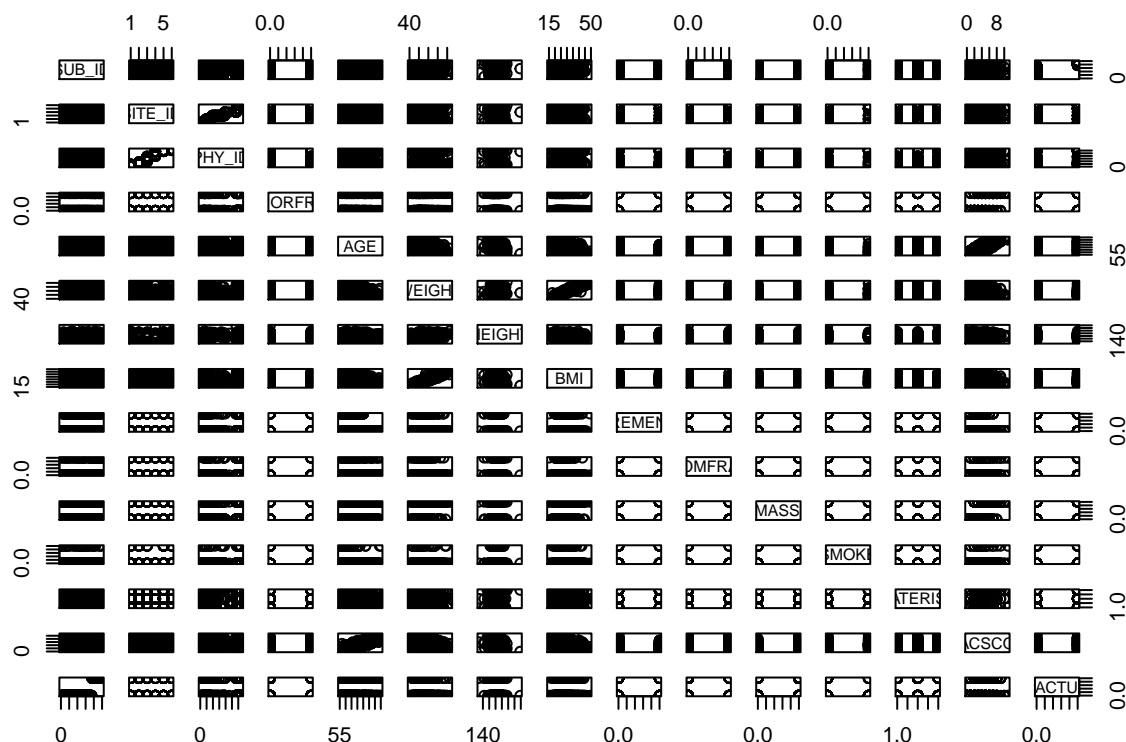
glow500 <- read.csv(file="glow500.csv",head=TRUE,sep=",")

set.seed(123)
sample <- sample(c(TRUE, FALSE), nrow(glow500), replace = T, prob = c(0.6,0.4))
train <- glow500[sample, ]
test <- glow500[!sample, ]

(lda.m1 <- lda(FRACTURE ~ RATERISK + FRACSCORE + HEIGHT, data = train))

## Call:
## lda(FRACTURE ~ RATERISK + FRACSCORE + HEIGHT, data = train)
##
## Prior probabilities of groups:
##      0      1
## 0.7508197 0.2491803
##
## Group means:
##      RATERISK FRACSCORE  HEIGHT
## 0 1.882096   3.410480 161.9389
## 1 2.118421   4.815789 160.0789
##
## Coefficients of linear discriminants:
##              LD1
## RATERISK    0.5461345
## FRACSCORE   0.3285060
## HEIGHT     -0.0516470

pairs(glow500)
```



#The cor() function produces a matrix that contains all of the pairwise correlations among the predictors
`cor(glow500)`

```
##          SUB_ID    SITE_ID    PHY_ID    PRIORFRAC    AGE
## SUB_ID    1.00000000  0.04844619  0.04469372  0.179556091  0.11702656
## SITE_ID    0.04844619  1.00000000  0.97516015 -0.022477436  0.02644718
## PHY_ID     0.04469372  0.97516015  1.00000000 -0.010830605  0.02271392
## PRIORFRAC  0.17955609 -0.02247744 -0.01083061  1.000000000  0.29145290
## AGE        0.11702656  0.02644718  0.02271392  0.291452898  1.00000000
## WEIGHT     -0.01351774 -0.05952453 -0.06222351 -0.023993946 -0.27159637
## HEIGHT     -0.05944021 -0.13009606 -0.14411659 -0.102203188 -0.19264861
## BMI         0.01157865 -0.01431135 -0.01286993  0.003314938 -0.22125651
## PREMENO    -0.02486197 -0.05604168 -0.05414796  0.006477171 -0.15911055
## MOMFRAC     0.11991894  0.05410014  0.05338689  0.022190273  0.03474619
## ARMASSIST  0.11717237  0.05643701  0.04206567  0.196139746  0.23831932
## SMOKE       -0.03505559  0.01600795  0.02312727  0.057413702 -0.09048779
## RATERISK    0.11323889  0.08516374  0.08315208  0.174844904 -0.04889352
## FRACSCORE   0.17637715  0.06300991  0.05670781  0.486079265  0.86991650
## FRACTURE    0.75000150  0.06935643  0.06745920  0.218088192  0.20765352
##          WEIGHT    HEIGHT    BMI    PREMENO    MOMFRAC
## SUB_ID    -0.013517738 -0.059440206  0.011578645 -0.024861968  0.11991894
## SITE_ID    -0.059524527 -0.130096059 -0.014311354 -0.056041680  0.05410014
## PHY_ID     -0.062223509 -0.144116593 -0.012869934 -0.054147961  0.05338689
## PRIORFRAC  -0.023993946 -0.102203188  0.003314938  0.006477171  0.02219027
## AGE        -0.271596372 -0.192648608 -0.221256511 -0.159110550  0.03474619
## WEIGHT     1.000000000  0.315969149  0.937336030  0.080381676 -0.06124937
## HEIGHT     0.315969149  1.000000000 -0.024376893 -0.009008094  0.06963166
## BMI         0.937336030 -0.024376893  1.000000000  0.094600200 -0.08804359
## PREMENO    0.080381676 -0.009008094  0.094600200  1.000000000 -0.00917403
## MOMFRAC    -0.061249375  0.069631660 -0.088043587 -0.009174030  1.00000000
## ARMASSIST  0.319197889  0.070604294  0.308034603  0.078605874  0.00687544
```

```
## SMOKE      0.002906384 -0.024370933  0.008832892  0.103278242 -0.01281949
## RATERISK   -0.082881514 -0.016604810 -0.084304949  0.075919297  0.12473287
## FRACSCORE  -0.161375361 -0.161995249 -0.120347231 -0.078528348  0.17564672
## FRACTURE   -0.036259440 -0.136400553  0.014985061  0.008760366  0.10643875
##           ARMASSIST      SMOKE      RATERISK      FRACSCORE      FRACTURE
## SUB_ID     0.11717237 -0.035055586  0.113238894  0.17637715  0.750001500
## SITE_ID    0.05643701  0.016007953  0.085163740  0.06300991  0.069356431
## PHY_ID     0.04206567  0.023127267  0.083152085  0.05670781  0.067459202
## PRIORFRAC  0.19613975  0.057413702  0.174844904  0.48607927  0.218088192
## AGE        0.23831932 -0.090487793 -0.048893522  0.86991650  0.207653516
## WEIGHT     0.31919789  0.002906384 -0.082881514 -0.16137536 -0.036259440
## HEIGHT     0.07060429 -0.024370933 -0.016604810 -0.16199525 -0.136400553
## BMI        0.30803460  0.008832892 -0.084304949 -0.12034723  0.014985061
## PREMENO    0.07860587  0.103278242  0.075919297 -0.07852835  0.008760366
## MOMFRAC    0.00687544 -0.012819494  0.124732868  0.17564672  0.106438749
## ARMASSIST  1.00000000  0.062141897  0.122703011  0.57269737  0.152567885
## SMOKE      0.06214190  1.000000000  0.003961627  0.07725582 -0.031679398
## RATERISK   0.12270301  0.003961627  1.000000000  0.08206561  0.151731878
## FRACSCORE  0.57269737  0.077255822  0.082065606  1.00000000  0.264479514
## FRACTURE   0.15256788 -0.031679398  0.151731878  0.26447951  1.000000000
```

Train randomforest:

```
library(randomForest)
```

```
## randomForest 4.6-12
```

```
## Type rfNews() to see new features/changes/bug fixes.
```

```
train$FRACTURE_F <- as.factor(train$FRACTURE)
rf.m <- randomForest(FRACTURE_F ~ RATERISK + FRACSCORE + HEIGHT, data=train, maxnodes=4, ntree=30)
test.predicted.rf <- predict(rf.m, newdata = test, type="response")
```

Linear Discriminant Analysis

```
(lda.m1 <- lda(FRACTURE ~ RATERISK + FRACSCORE + HEIGHT, data = train))
```

```
## Call:
```

```
## lda(FRACTURE ~ RATERISK + FRACSCORE + HEIGHT, data = train)
```

```
##
```

```
## Prior probabilities of groups:
```

```
##      0      1
```

```
## 0.7508197 0.2491803
```

```
##
```

```
## Group means:
```

```
##   RATERISK FRACSCORE   HEIGHT
```

```
## 0 1.882096  3.410480 161.9389
```

```
## 1 2.118421  4.815789 160.0789
```

```
##
```

```
## Coefficients of linear discriminants:
```

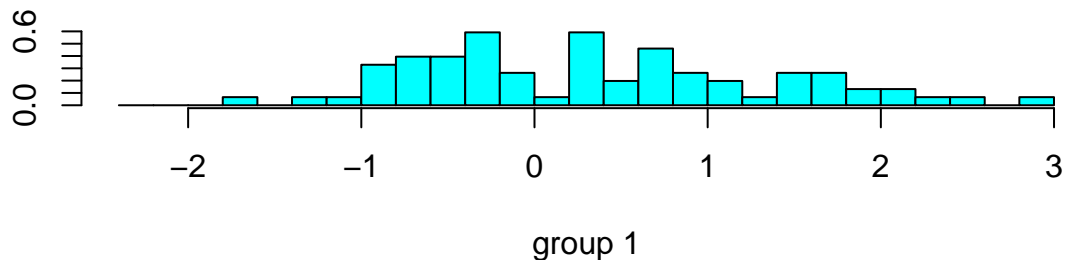
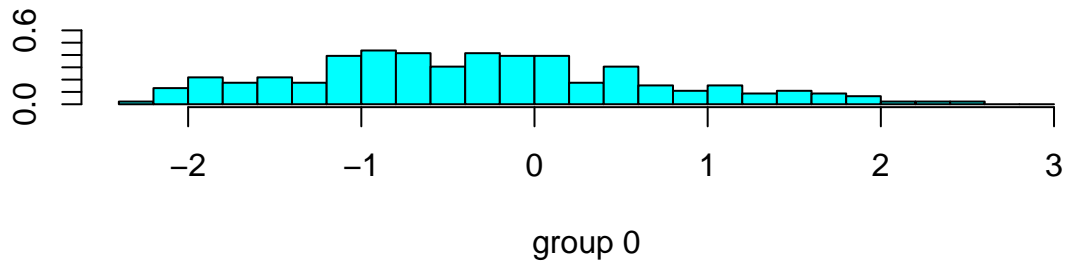
```
##           LD1
```

```
## RATERISK   0.5461345
```

```
## FRACSCORE  0.3285060
```

```
## HEIGHT    -0.0516470
```

```
plot(lda.m1)
```

```
(qda.m1 <- qda(FRACTURE ~ RATERISK + FRACSCORE + HEIGHT, data = train))
```

```
## Call:
## qda(FRACTURE ~ RATERISK + FRACSCORE + HEIGHT, data = train)
##
## Prior probabilities of groups:
##      0      1
## 0.7508197 0.2491803
##
## Group means:
##   RATERISK FRACSCORE  HEIGHT
## 0  1.882096  3.410480 161.9389
## 1  2.118421  4.815789 160.0789
```

Train linear regression model:

```
(glm.fit <- glm(FRACTURE ~ RATERISK + FRACSCORE + HEIGHT, data = train))
```

```
##
## Call:  glm(formula = FRACTURE ~ RATERISK + FRACSCORE + HEIGHT, data = train)
##
## Coefficients:
## (Intercept)    RATERISK    FRACSCORE    HEIGHT
##   0.967155    0.064873    0.039022   -0.006135
##
## Degrees of Freedom: 304 Total (i.e. Null);  301 Residual
## Null Deviance:      57.06
## Residual Deviance: 52.41    AIC: 338.4
```

Evaluate random forest:

```
library(randomForest)
train$FRACTURE <- as.factor(train$FRACTURE)
rf.m <- randomForest(FRACTURE ~ RATERISK + FRACSCORE + HEIGHT, data=train, maxnodes=5, ntree=5)
```

```
test.predicted.rf <- predict(rf.m, newdata = test, type="response")

test.predicted.rf <- predict(rf.m, newdata = test, type="response")
# confusion matrix
table(test$FRACTURE, test.predicted.rf)
```

```
##      test.predicted.rf
##      0      1
## 0 146      0
## 1   49      0
```

```
# accuracy rate
mean(test.predicted.rf == test$FRACTURE)
```

```
## [1] 0.7487179
```

Evaluate Linear Discriminant Analysis Model:

```
# predictions
test.predicted.lda <- predict(lda.m1, newdata = test)

# confusion matrix
table(test$FRACTURE, test.predicted.lda$class)
```

```
##
##      0      1
## 0 140      6
## 1   41      8
```

```
# accuracy rate
mean(test.predicted.lda$class == test$FRACTURE)
```

```
## [1] 0.7589744
```

```
# predictions
test.predicted.qda <- predict(qda.m1, newdata = test)

# confusion matrix
table(test$FRACTURE, test.predicted.qda$class)
```

```
##
##      0      1
## 0 140      6
## 1   45      4
```

```
# accuracy rate
mean(test.predicted.qda$class == test$FRACTURE)
```

```
## [1] 0.7384615
```

Evaluate Linear Regression Model:

Confusion Matrix:

```
# predictions
glm.probs <- predict(glm.fit, test, type = "response")

# confusion matrix
table(test$FRACTURE, ifelse(glm.probs < 0.5, 0, 1))
```

```
##
##      0   1
##    0 142   4
##    1  47   2
```

Accuracy:

```
# accuracy rate
mean(ifelse(glm.probs > 0.5, 1 , 0) == test$FRACTURE)
```

```
## [1] 0.7384615
```

```
# ROC curves
library(ROCR)
```

```
## Loading required package: gplots
```

```
##
```

```
## Attaching package: 'gplots'
```

```
## The following object is masked from 'package:stats':
```

```
##
```

```
##      lowess
```

```
library(dplyr)
```

```
##
```

```
## Attaching package: 'dplyr'
```

```
## The following object is masked from 'package:randomForest':
```

```
##
```

```
##      combine
```

```
## The following object is masked from 'package:MASS':
```

```
##
```

```
##      select
```

```
## The following objects are masked from 'package:stats':
```

```
##
```

```
##      filter, lag
```

```
## The following objects are masked from 'package:base':
```

```
##
```

```
##      intersect, setdiff, setequal, union
```

```
library(randomForest)
train$FRACTURE <- as.factor(train$FRACTURE)
rf.m <- randomForest(FRACTURE ~ RATERISK + FRACSCORE + HEIGHT, data=train, maxnodes=5, ntree= 1000)
test.predicted.rf <- predict(rf.m, newdata = test, type="response")
```

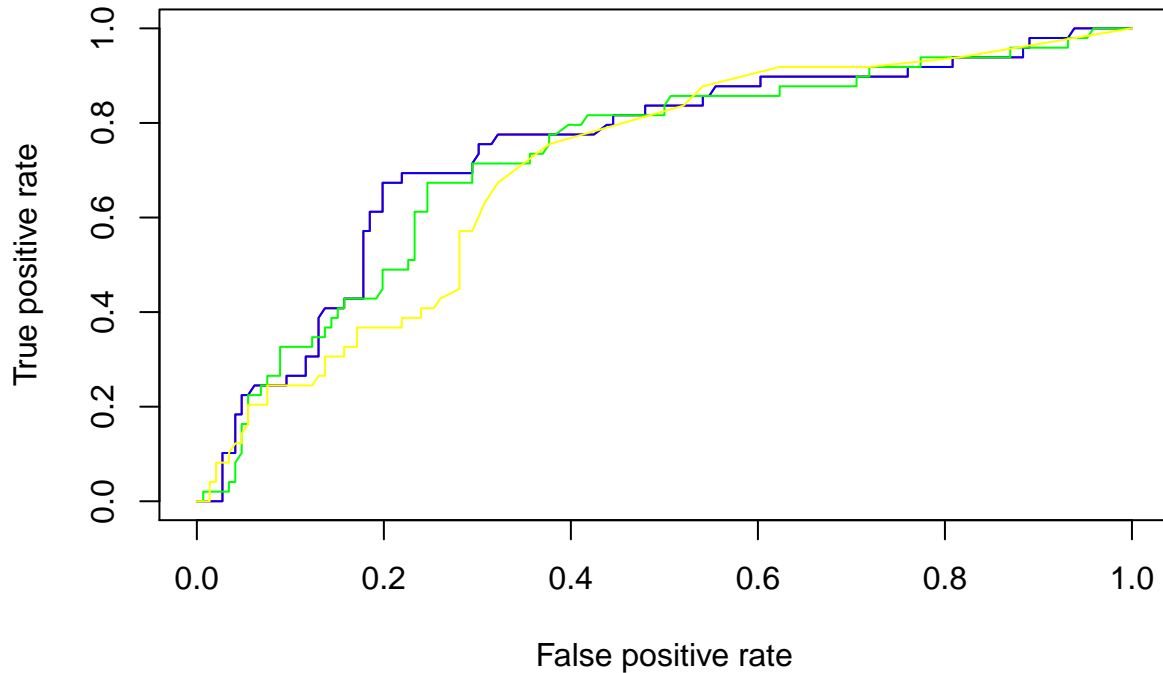
```
logistic <- prediction(glm.probs, test$FRACTURE) %>%
  performance(measure = "tpr", x.measure = "fpr")
```

```
lda <- prediction(test.predicted.lda$posterior[,2], test$FRACTURE) %>%
  performance(measure = "tpr", x.measure = "fpr")
```

```
qda <- prediction(test.predicted.qda$posterior[,2], test$FRACTURE) %>%
  performance(measure = "tpr", x.measure = "fpr")
```

```
test.predicted.rf.prob <- predict(rf.m, newdata = test, type="prob")
rf <- prediction(test.predicted.rf.prob[,2], test$FRACTURE) %>%
  performance(measure = "tpr", x.measure = "fpr")

plot(logistic, col = "red")
plot(lda, add = TRUE, col = "blue")
plot(qda, add = TRUE, col = "green")
plot(rf, add = TRUE, col = "yellow")
```



```
# Logistic regression AUC
prediction(glm.probs, test$FRACTURE) %>%
  performance(measure = "auc") %>%
  .@y.values
```

```
## [[1]]
## [1] 0.7435001
```

```
# LDA AUC
prediction(test.predicted.lda$posterior[,2], test$FRACTURE) %>%
  performance(measure = "auc") %>%
  .@y.values
```

```
## [[1]]
## [1] 0.7435001
```

```
# QDA AUC
prediction(test.predicted.qda$posterior[,2], test$FRACTURE) %>%
  performance(measure = "auc") %>%
  .@y.values
```

```
## [[1]]
## [1] 0.7264467
```

```
# RandomForest AUC
prediction(test.predicted.rf.prob[,2], test$FRACTURE) %>%
  performance(measure = "auc") %>%
  .@y.values
```

```
## [[1]]
## [1] 0.7038021
```

SAS Codes:

```
LIBNAME MYSASLIB '/home/szarandioon0/';
DATA GLOW500_ORIG;
INFILE '/home/szarandioon0/statistics2/Project2/glow500.csv' DLM = ',' FIRSTOBS = 2;
INPUT SUB_ID SITE_ID PHY_ID PRIORFRAC AGE WEIGHT HEIGHT BMI PREMENO MOMFRAC ARMASSIST SMOKE RATERISK FRACSCORE;
RUN;

DATA GLOW500(DROP = SUB_ID);
SET GLOW500_ORIG;
RUN;

proc factor data=GLOW500 simple corr;
run;

ods graphics on;
proc princomp data=GLOW500 plots(ncomp=3)=all n=5;
run;

proc candisc data=GLOW500 out=discrim_out ;
  class FRACTURE;
  var SITE_ID PHY_ID PRIORFRAC AGE WEIGHT HEIGHT BMI PREMENO MOMFRAC ARMASSIST SMOKE RATERISK FRACSCORE;
run;

  title 'Stepwise Regression on Global Longitudinal Study of Osteoporosis in Women (GLOW) Dataset';
proc logistic data=GLOW500 outest=betas covout;
model FRACTURE(event='1')=SITE_ID PHY_ID PRIORFRAC AGE WEIGHT HEIGHT BMI PREMENO MOMFRAC ARMASSIST SMOKE RATERISK FRACSCORE
  / selection=stepwise;
output out=pred p=phat lower=lcl upper=ucl predprob=(individual crossvalidate);
run;

data train test;
set GLOW500;
if rand('uniform') <= 0.3
then output test;
else output train;
run;

ods graphics on;
proc logistic data=train;
model FRACTURE(event="1") = RATERISK FRACSCORE HEIGHT / outroc=troc;
score data=test out=valpred outroc=vroc;
roc; roccontrast;
run;

proc logistic data=train plots(only)=roc;
model FRACTURE(event="1") = RATERISK FRACSCORE HEIGHT;
```

```

run;

proc logistic data=train rocoptions(crossvalidate) plots(only)=roc;
model FRACTURE(event="1") = RATERISK FRACSCORE HEIGHT;
run;

proc discrim data=train testdata=test canonical;
class FRACTURE;
var SITE_ID PHY_ID PRIORFRAC AGE WEIGHT HEIGHT BMI PREMENO MOMFRAC ARMASSIST SMOKE RATERISK FRACSCORE;
run;

proc hpforest data=train;
target FRACTURE/level=nominal;
input PRIORFRAC PREMENO MOMFRAC ARMASSIST SMOKE/level=nominal;
input SITE_ID PHY_ID AGE WEIGHT HEIGHT BMI RATERISK FRACSCORE/level=interval;
run;

```

Including Plots

You can also embed plots, for example:

