Measuring Performance: Evaluating SMOTE

Spiro Stilianoudakis

Contents

Loading Packages

```
library(caret)
```

```
## Warning: package 'caret' was built under R version 3.4.4
```

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

```
## Loading required package: ggplot2
```

```
## Warning: package 'ggplot2' was built under R version 3.4.4
```

```
#library(data.table)
library(gbm)
```

```
## Loading required package: survival
```

```
##
## Attaching package: 'survival'
```

```
## The following object is masked from 'package:caret':
##
##     cluster
```

```
## Loading required package: splines
```

```
## Loading required package: parallel
```

```
## Loaded gbm 2.1.3
```

```
library(pROC)
```

```
## Warning: package 'pROC' was built under R version 3.4.4
```

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

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

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

```
library(plyr)
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 3.4.4
```

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

```
## The following objects are masked from 'package:plyr':
##
##     arrange, count, desc, failwith, id, mutate, rename, summarise,
##     summarize
```

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

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(DMwR)
```

```
## Loading required package: grid
```

```
##
## Attaching package: 'DMwR'
```

```
## The following object is masked from 'package:plyr':
##
##     join
```

```
library(gridExtra)
```

```
##
## Attaching package: 'gridExtra'
```

```
## The following object is masked from 'package:dplyr':
##
##     combine
```

```
library(ggplot2)
library(leaps)
#library(DT)
library(knitr)
```

```
## Warning: package 'knitr' was built under R version 3.4.4
```

Setting Working directory

```
setwd("C:/Users/Spiro Stilianoudakis/Documents/TAD_data/RData/GM12878/testing_SMOTE")
```

Testing SMOTE

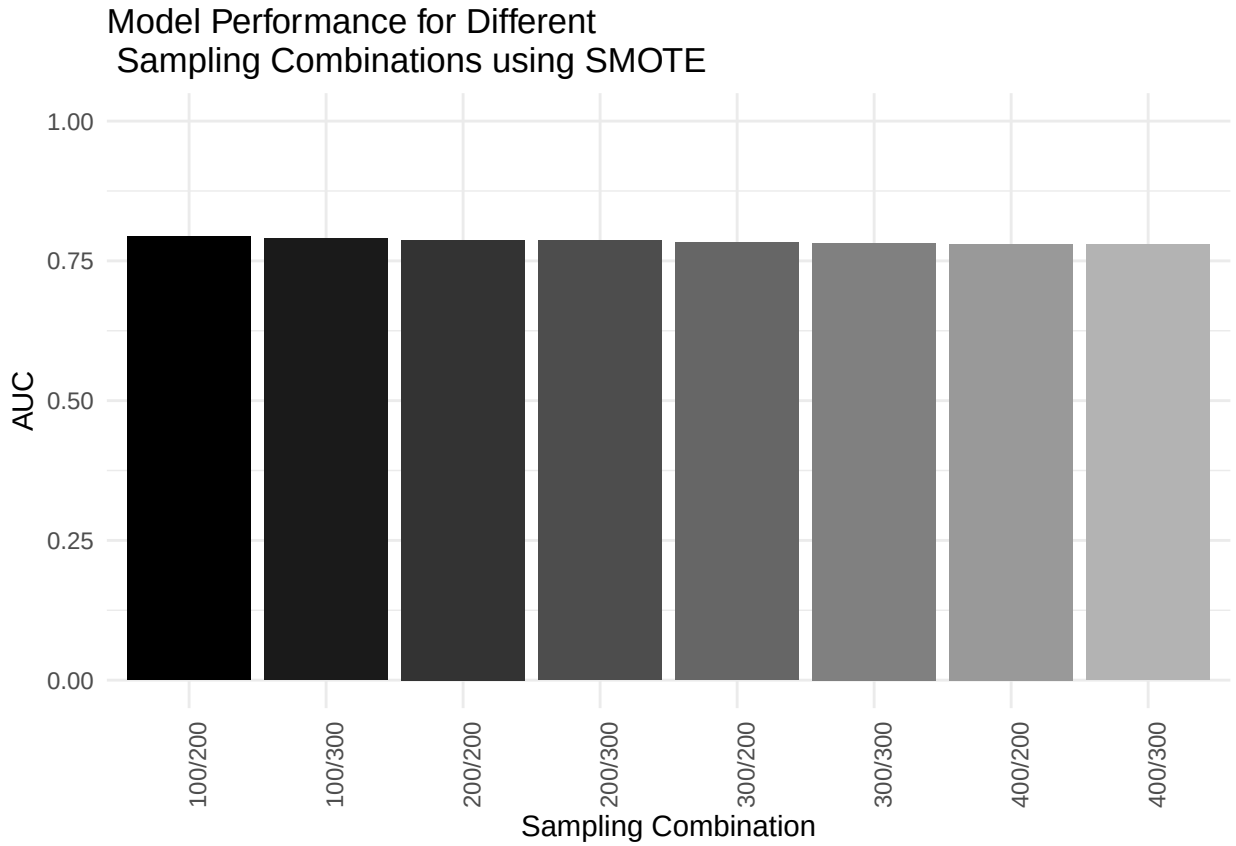
```
enetlst_sm <- readRDS("C:/Users/Spiro Stilianoudakis/Documents/TAD_data/RData/GM12878/testing_SMOTE/enet

#Plotting Performance
auc.sm <- data.frame(Combination=c("100/200","200/200","300/200","400/200",
                                   "100/300","200/300","300/300","400/300"),
                     AUC=c(enetlst_sm[[3]][1],enetlst_sm[[3]][2],enetlst_sm[[3]][3],
                           enetlst_sm[[3]][4],enetlst_sm[[3]][5],enetlst_sm[[3]][6],
                           enetlst_sm[[3]][7],enetlst_sm[[3]][8]))

auc.sm <- auc.sm[order(auc.sm$AUC, decreasing=TRUE),]

auc.sm$Combination <- factor(auc.sm$Combination, levels=auc.sm$Combination)

auc.sm
```

```
##   Combination       AUC
## 1     100/200 0.7936148
## 5     100/300 0.7898228
## 2     200/200 0.7872268
## 6     200/300 0.7864398
## 3     300/200 0.7824853
## 7     300/300 0.7820608
## 4     400/200 0.7802347
## 8     400/300 0.7790618
```

```
#datatable(auc.sm)
kable(auc.sm)
```

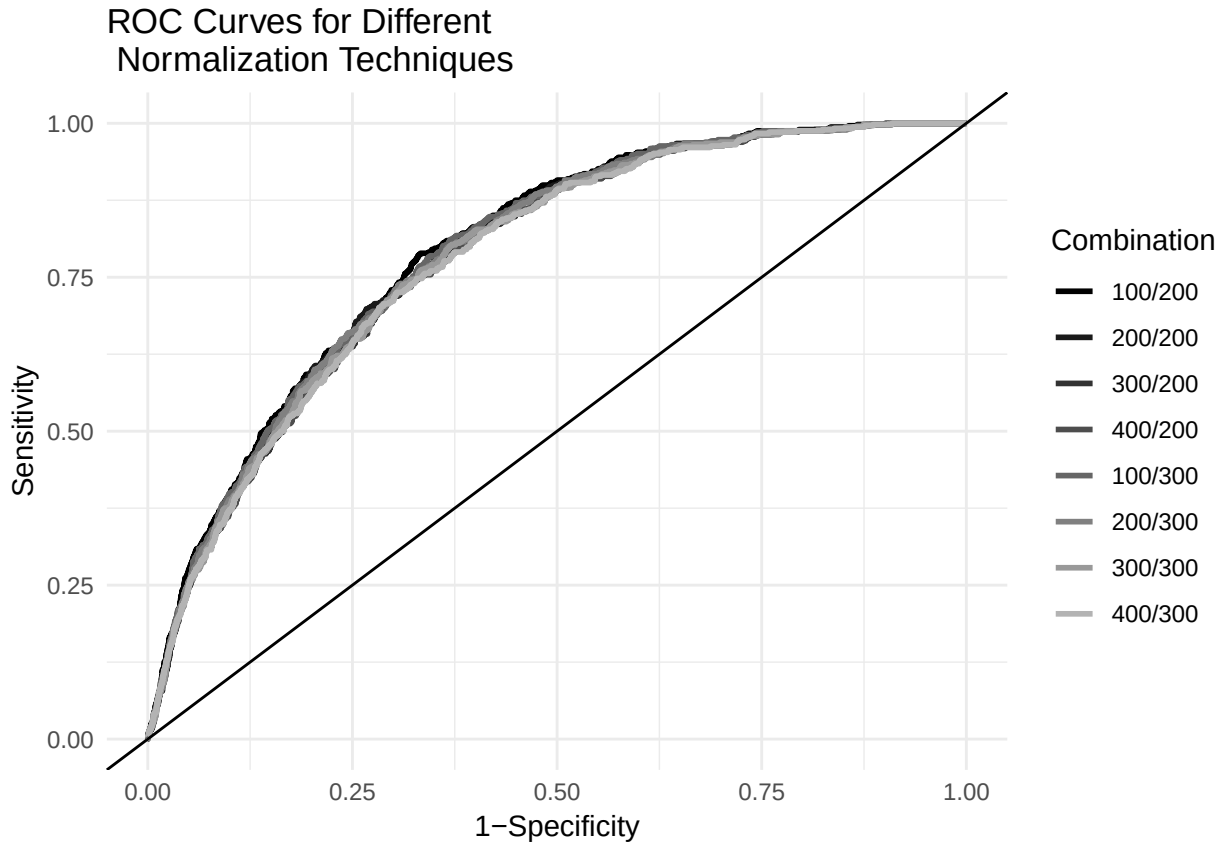
	Combination	AUC
1	100/200	0.7936148
5	100/300	0.7898228
2	200/200	0.7872268
6	200/300	0.7864398
3	300/200	0.7824853
7	300/300	0.7820608
4	400/200	0.7802347
8	400/300	0.7790618

```
p<-ggplot(data=auc.sm, aes(x=Combination, y=AUC, fill=Combination)) +
  xlab("Sampling Combination") + ylab("AUC") +
  geom_bar(stat="identity") + ylim(0,1) +
  scale_fill_manual(values=gray(seq(0,.7,.1)), guide=FALSE) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
  ggtitle("Model Performance for Different \n Sampling Combinations using SMOTE")
p
```




```
onetwo <- data.frame(fpr=enetlst_sm[[2]][,1],tpr=enetlst_sm[[1]][,1], Combo = "100/200");
twotwo <- data.frame(fpr=enetlst_sm[[2]][,2],tpr=enetlst_sm[[1]][,2], Combo = "200/200");
threetwo <- data.frame(fpr=enetlst_sm[[2]][,3],tpr=enetlst_sm[[1]][,3], Combo = "300/200");
fourtwo <- data.frame(fpr=enetlst_sm[[2]][,4],tpr=enetlst_sm[[1]][,4], Combo = "400/200");
onethree <- data.frame(fpr=enetlst_sm[[2]][,5],tpr=enetlst_sm[[1]][,5], Combo = "100/300");
twothree <- data.frame(fpr=enetlst_sm[[2]][,6],tpr=enetlst_sm[[1]][,6], Combo = "200/300");
threethree <- data.frame(fpr=enetlst_sm[[2]][,7],tpr=enetlst_sm[[1]][,7], Combo = "300/300");
fourthree <- data.frame(fpr=enetlst_sm[[2]][,8],tpr=enetlst_sm[[1]][,8], Combo = "400/300")

allrocdat <- rbind.data.frame(onetwo,
                              twotwo,
                              threetwo,
                              fourtwo,
                              onethree,
                              twothree,
                              threethree,
                              fourthree)

ggplot(data=allrocdat, aes(x=fpr, y=tpr, color=Combo)) +
  geom_line(size=1) +
  scale_colour_manual(name="Combination",
    labels=c("100/200",
             "200/200",
             "300/200",
             "400/200",
             "100/300",
```

```
            "200/300",
            "300/300",
            "400/300"),
   values=gray(seq(0,.7,.1))) +
 xlab("1-Specificity") +
 ylab("Sensitivity") +
 xlim(0, 1) +
 ylim(0, 1) +
 geom_abline(intercept=0, slope=1) +
 theme_minimal() +
 ggtitle("ROC Curves for Different \n Normalization Techniques")
```



Bootstrap

```
enetlst_bs <- readRDS("C:/Users/Spiro Stilianoudakis/Documents/TAD_data/RData/GM12878/testing_SMOTE/enet

#Mean AUC across 100 bootstrap samples
enetlst_bs[[3]]
```

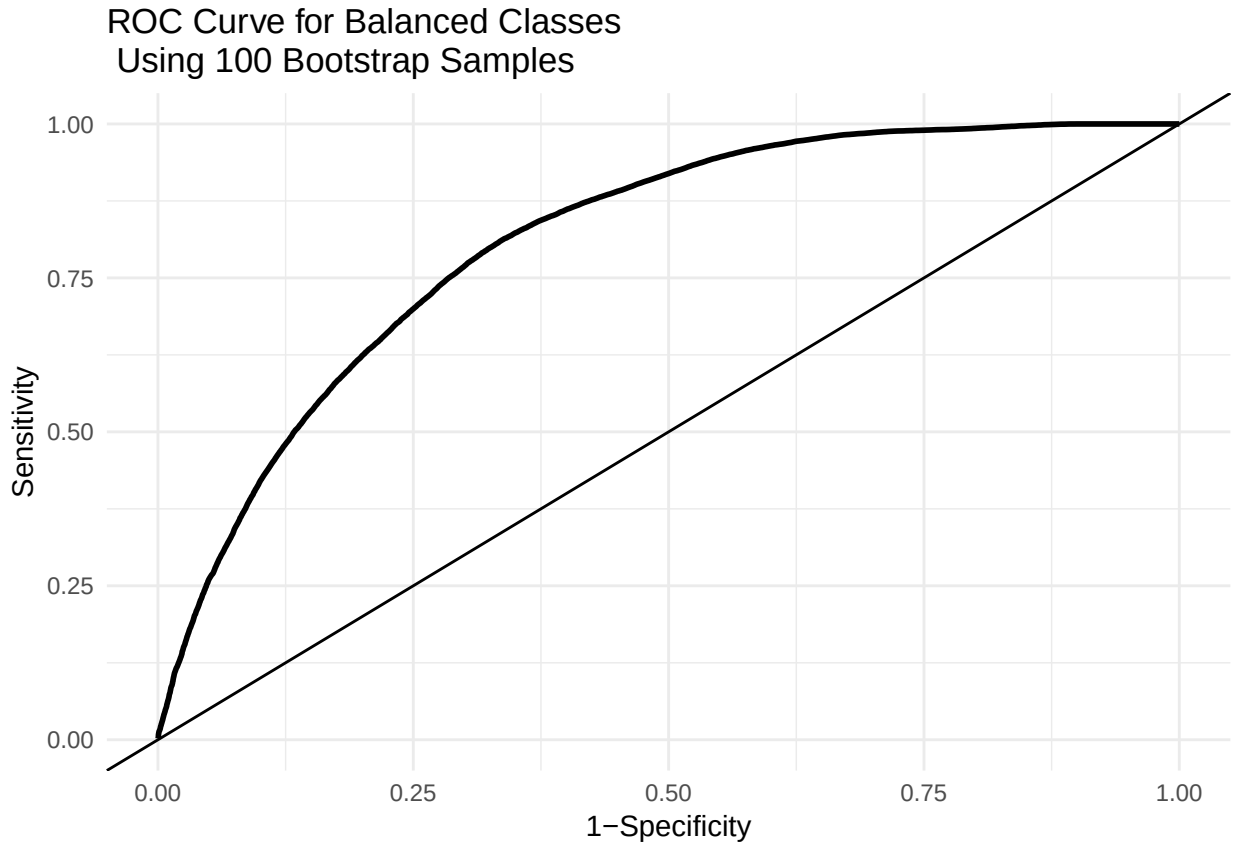
```
##   [1] 0.8178990 0.8037889 0.8082051 0.8145617 0.7967255 0.7963031 0.8069379
##   [8] 0.7884660 0.7931457 0.8075987 0.8188859 0.8066243 0.7964453 0.7947809
##  [15] 0.8081842 0.8216209 0.8056332 0.8248662 0.7772499 0.8016895 0.7996822
##  [22] 0.8139302 0.8035714 0.7896495 0.8346144 0.8260999 0.8171295 0.8032034
##  [29] 0.8277643 0.8142732 0.7823310 0.8095642 0.8184384 0.8112036 0.8189696
```

```
##  [36] 0.8305997 0.8044204 0.7910547 0.8203036 0.8082302 0.8016728 0.7911132
##  [43] 0.8332929 0.8131608 0.8021119 0.7978128 0.8007862 0.8264386 0.8119982
##  [50] 0.7998453 0.7913642 0.8055830 0.8084727 0.8333431 0.8341502 0.8001756
##  [57] 0.8320132 0.7959225 0.7894781 0.8086986 0.7999875 0.8010915 0.8203120
##  [64] 0.7969430 0.8049222 0.7933255 0.8216251 0.8245651 0.8163934 0.8235865
##  [71] 0.8051062 0.7971270 0.8107394 0.7769823 0.7978212 0.8132988 0.8222064
##  [78] 0.8113625 0.7988709 0.7956340 0.8339453 0.7983690 0.8020994 0.8151054
##  [85] 0.8183883 0.8063399 0.8142857 0.8291736 0.8347148 0.8022708 0.8177191
##  [92] 0.8102919 0.8128889 0.7857687 0.8208431 0.8158414 0.8010497 0.8257653
##  [99] 0.8115632 0.8185388
```

```
auc.bs <- round(mean(enetlst_bs[[3]]),3)
auc.bs
```

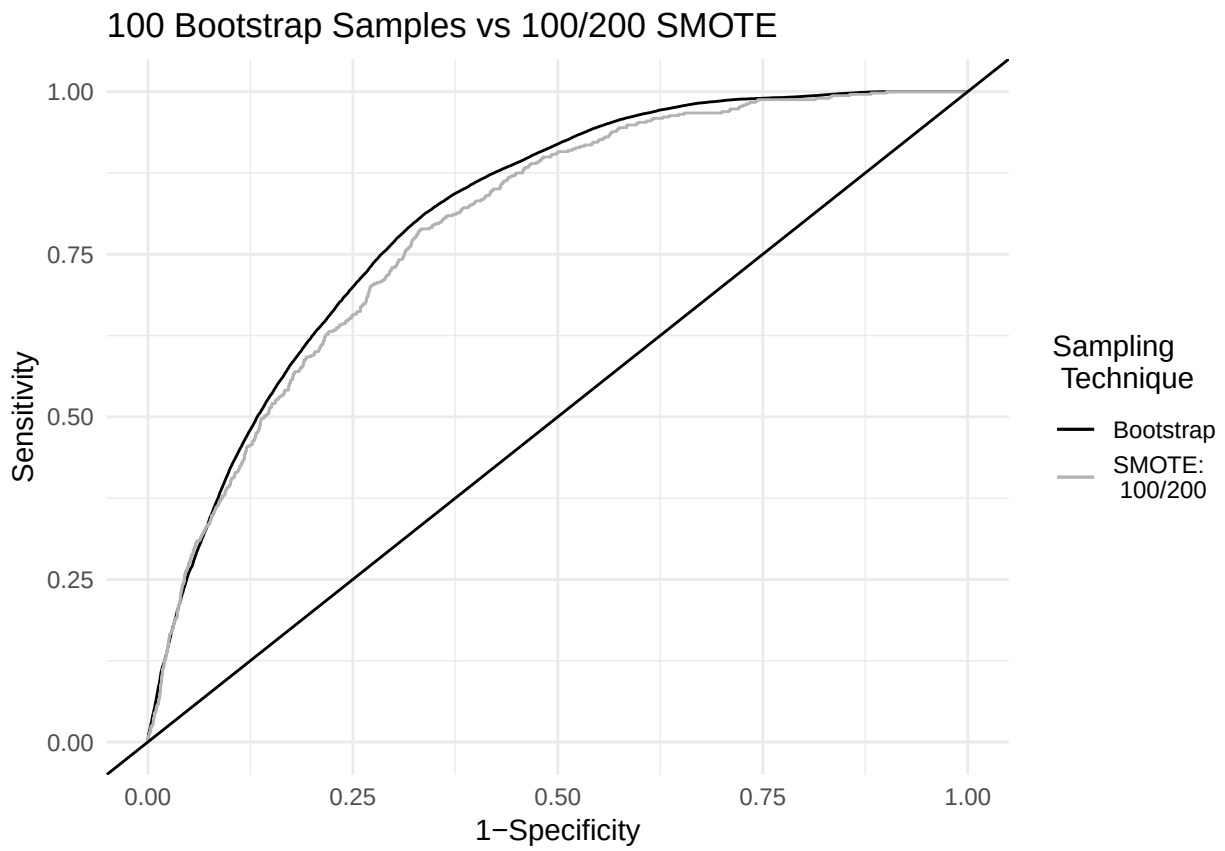
```
## [1] 0.809
```

```
#roc curve
fpr.bs <- rowMeans(enetlst_bs[[2]])
tpr.bs <- rowMeans(enetlst_bs[[1]])
rocdat.bs <- data.frame(fpr=fpr.bs, tpr=tpr.bs)
ggplot(rocdat.bs, aes(x=fpr, y=tpr)) +
  geom_line(size=1, color="black") +
  xlab("1-Specificity") +
  ylab("Sensitivity") +
  xlim(0, 1) +
  ylim(0, 1) +
  geom_abline(intercept=0, slope=1) +
  theme_minimal() +
  ggtitle("ROC Curve for Balanced Classes \n Using 100 Bootstrap Samples")
```

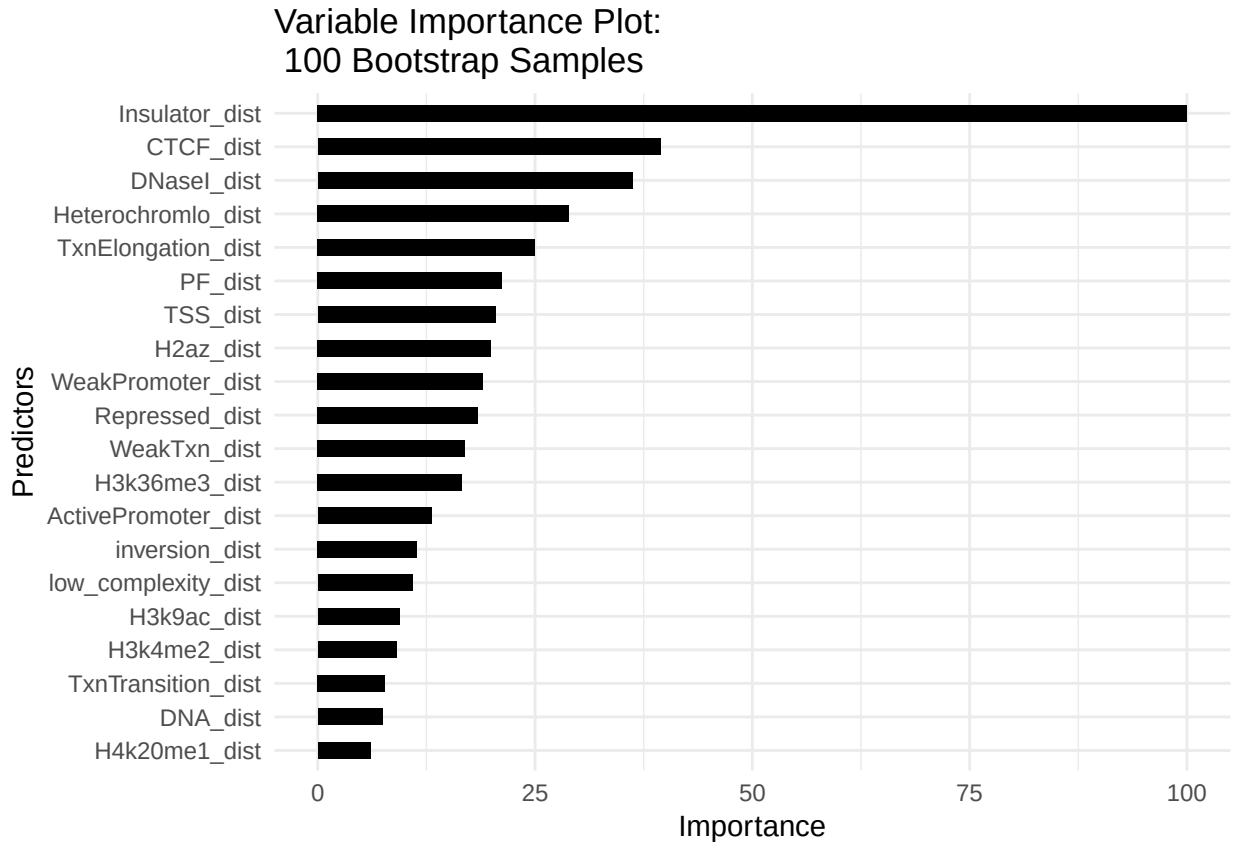


Comparing 100/200 SMOTE with Bootstrapped model

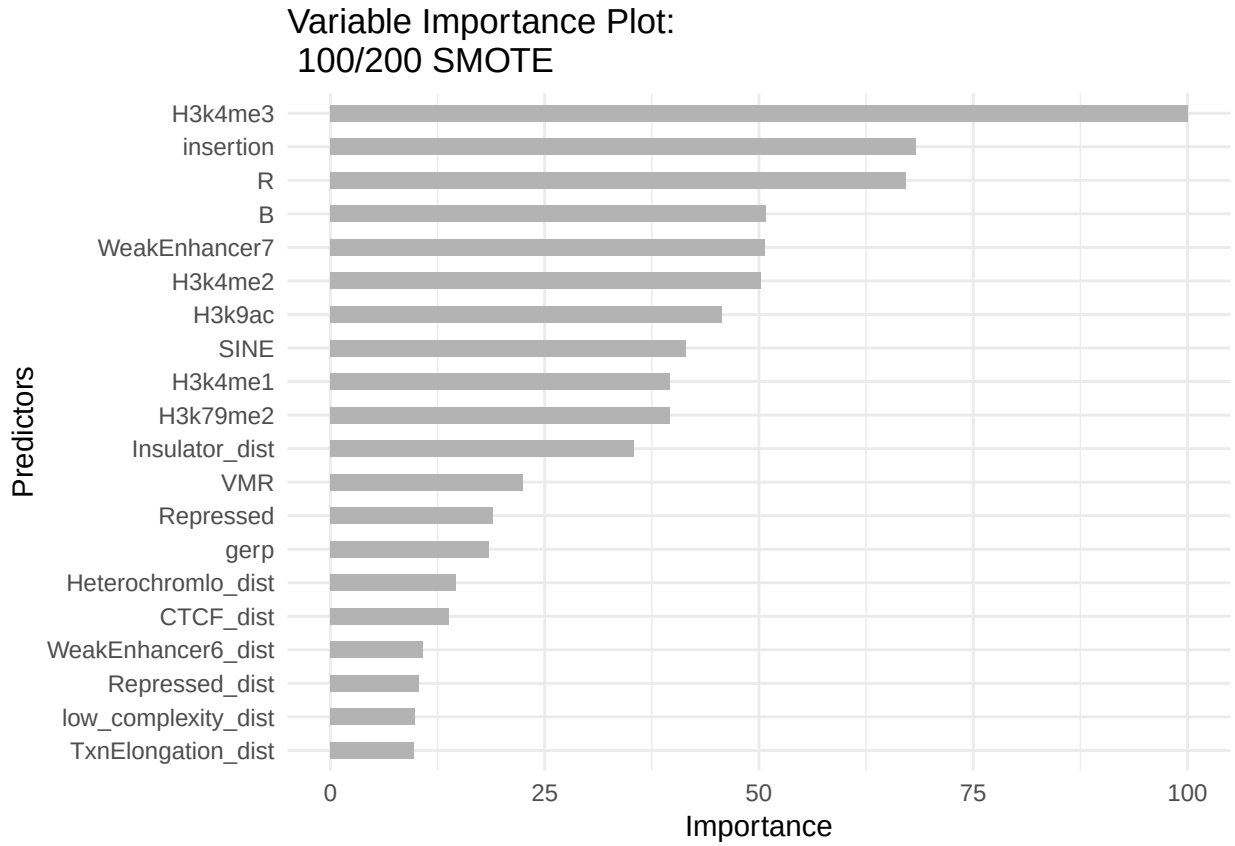
```
ggplot() +
  geom_line(aes(fpr, tpr, colour=gray(.7)[1]), rocdat.bs) +
  geom_line(aes(fpr, tpr, colour="black"), onetwo) +
  scale_colour_manual(name="Sampling \n Technique",
    labels=c("Bootstrap","SMOTE: \n 100/200"),
    values=c("black",gray(.7))) +
  xlab("1-Specificity") +
  ylab("Sensitivity") +
  xlim(0, 1) +
  ylim(0, 1) +
  geom_abline(intercept=0, slope=1) +
  theme_minimal() +
  ggtitle("100 Bootstrap Samples vs 100/200 SMOTE")
```




```
varimp.bs <- as.vector(rowMeans(enetlst_bs[[4]]))
Labels <- rownames(enetlst_bs[[4]])
Labels[grep("Gm12878_", Labels)] <- gsub("Gm12878_","",Labels[grep("Gm12878_", Labels)])
varimp.bs.df <- data.frame(Feature=Labels,
                           Importance=varimp.bs)
varimp.bs.df <- varimp.bs.df[order(varimp.bs.df$Importance),]
varimp.bs.df <- varimp.bs.df[(dim(varimp.bs.df)[1]-19):dim(varimp.bs.df)[1],]
varimp.bs.df$Feature <- factor(varimp.bs.df$Feature,
                               levels=varimp.bs.df$Feature)
p.bs <- ggplot(varimp.bs.df, aes(x=Feature, y=Importance)) +
  xlab("Predictors") +
  ylab("Importance") +
  #ggtitle("Importance Plot for Gradient Boosting Machine") +
  geom_bar(stat="identity",
           width=.5,
           position="dodge",
           fill="black") +
  coord_flip() +
  theme_minimal() +
  ggtitle("Variable Importance Plot: \n 100 Bootstrap Samples")
p.bs
```




```
varimp.sm <- as.vector(enetlst_sm[[4]][,1])
Labels <- names(enetlst_sm[[4]][,1])
Labels[grep("Gm12878_", Labels)] <- gsub("Gm12878_","",Labels[grep("Gm12878_", Labels)])
varimp.sm.df <- data.frame(Feature=Labels,
                           Importance=varimp.sm)
varimp.sm.df <- varimp.sm.df[order(varimp.sm.df$Importance),]
varimp.sm.df <- varimp.sm.df[(dim(varimp.sm.df)[1]-19):dim(varimp.sm.df)[1],]
varimp.sm.df$Feature <- factor(varimp.sm.df$Feature,
                               levels=varimp.sm.df$Feature)
p.sm <- ggplot(varimp.sm.df, aes(x=Feature, y=Importance)) +
  xlab("Predictors") +
  ylab("Importance") +
  #ggtitle("Importance Plot for Gradient Boosting Machine") +
  geom_bar(stat="identity",
           width=.5,
           position="dodge",
           fill=gray(.7)) +
  coord_flip() +
  theme_minimal() +
  ggtitle("Variable Importance Plot: \n 100/200 SMOTE")
p.sm
```


Variable Importance Plot:
100/200 SMOTE
Predictors
H3k4me3
insertion
R
B
WeakEnhancer7
H3k4me2
H3k9ac
SINE
H3k4me1
H3k79me2
Insulator_dist
VMR
Repressed
gerp
Heterochromlo_dist
CTCF_dist
WeakEnhancer6_dist
Repressed_dist
low_complexity_dist
TxnElongation_dist
0
25
50
75
100
Importance