

Measuring Performance: Evaluating SMOTE

Spiro Stilianoudakis

Contents

Loading Packages	1
Setting Working directory	2
Testing SMOTE	3
Bootstrap	5
Comparing additional performance metrics across all methods	6
Comparing 100/200 SMOTE with Bootstrapped model	7

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/enetlst_sm")

#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

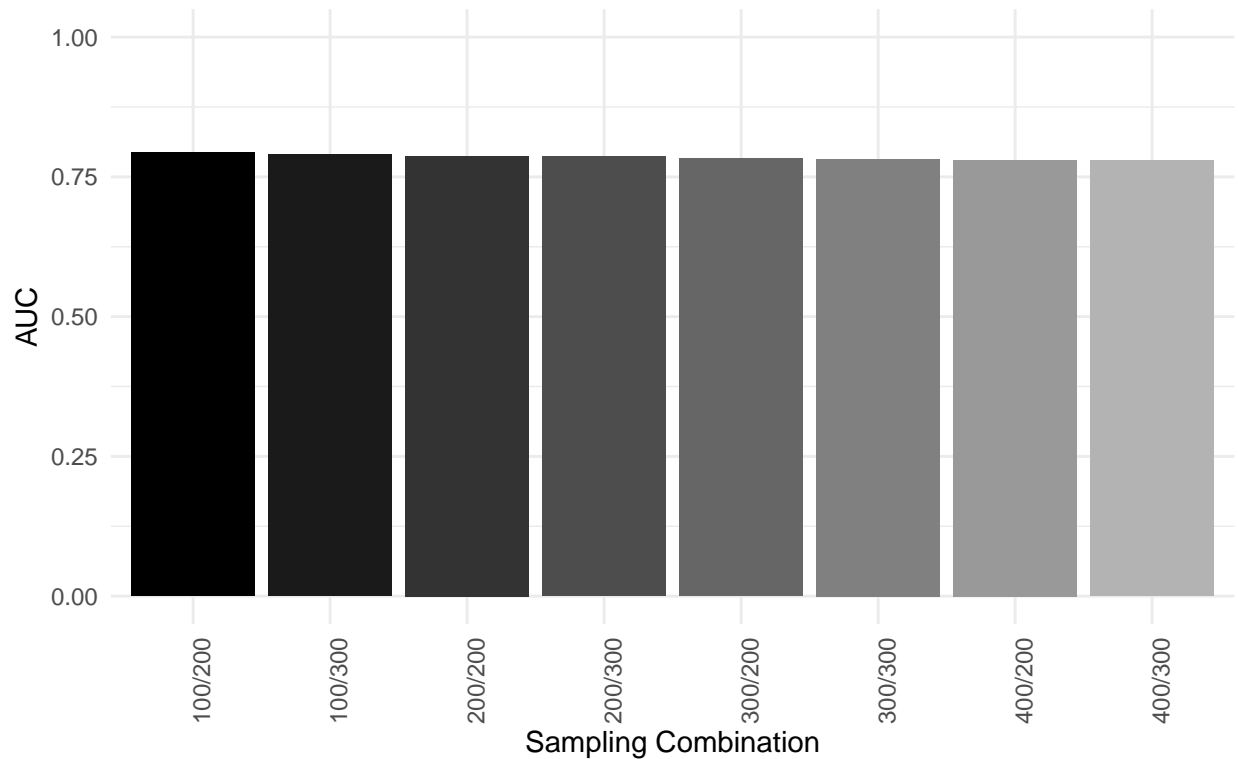
##      Combination      AUC
## 1      100/200 0.7936148
## 5      100/300 0.7898228
## 2      200/200 0.7872268
## 6      200/300 0.7864398
## 3      300/200 0.7824853
## 7      300/300 0.7820608
## 4      400/200 0.7802347
## 8      400/300 0.7790618

#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
```

Model Performance for Different Sampling Combinations using SMOTE



```
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")

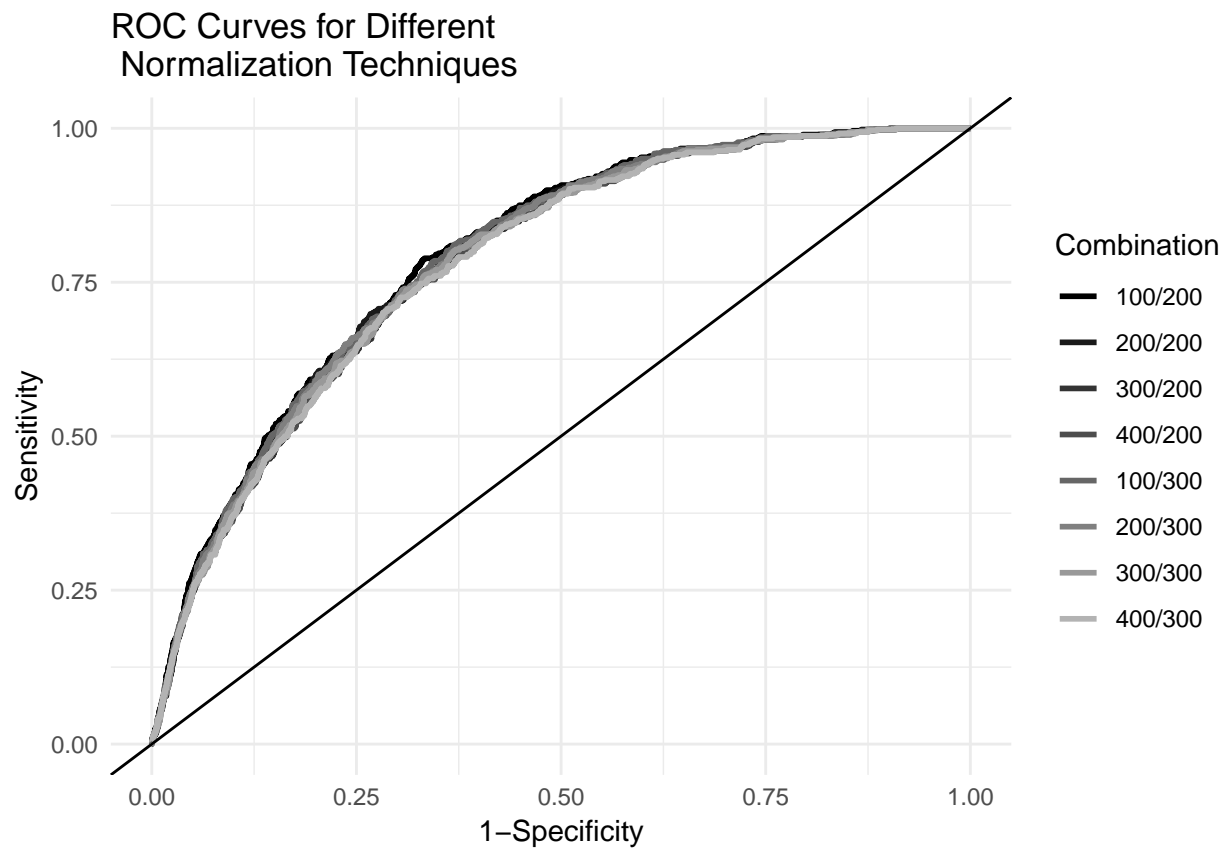
allrocdat <- rbind.data.frame(onetwo,
                             twotwo,
                             threetwo,
                             fourtwo,
                             onethree,
                             twothree,
                             threethree,
                             fourthree)

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"))
```

```

      "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/enetlst_bs")
#Mean AUC across 100 bootstrap samples
enetlst_bs[[3]]

## [1] 0.8150594 0.8014177 0.8032703 0.8138675 0.7917615

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

## [1] 0.805

```

```

#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 additional performance metrics across all methods

```

options(scipen = 999)

enetperf_sm <- readRDS("C:/Users/Spiro Stilianoudakis/Documents/TAD_data/RData/GM12878/testing_SMOTE/enetperf_sm.rds")
enetperf_b <- readRDS("C:/Users/Spiro Stilianoudakis/Documents/TAD_data/RData/GM12878/testing_SMOTE/enetperf_b.rds")

enetperf_sm <- round(enetperf_sm,2)
enetperf_b <- round(as.matrix(rowMeans(enetperf_b)),2)

```

```
enetperf_b[1:5,1] <- round(enetperf_b[1:5,1],0)

perfdat <- cbind.data.frame(rownames(enetperf_b),
                             enetperf_sm,
                             enetperf_b)

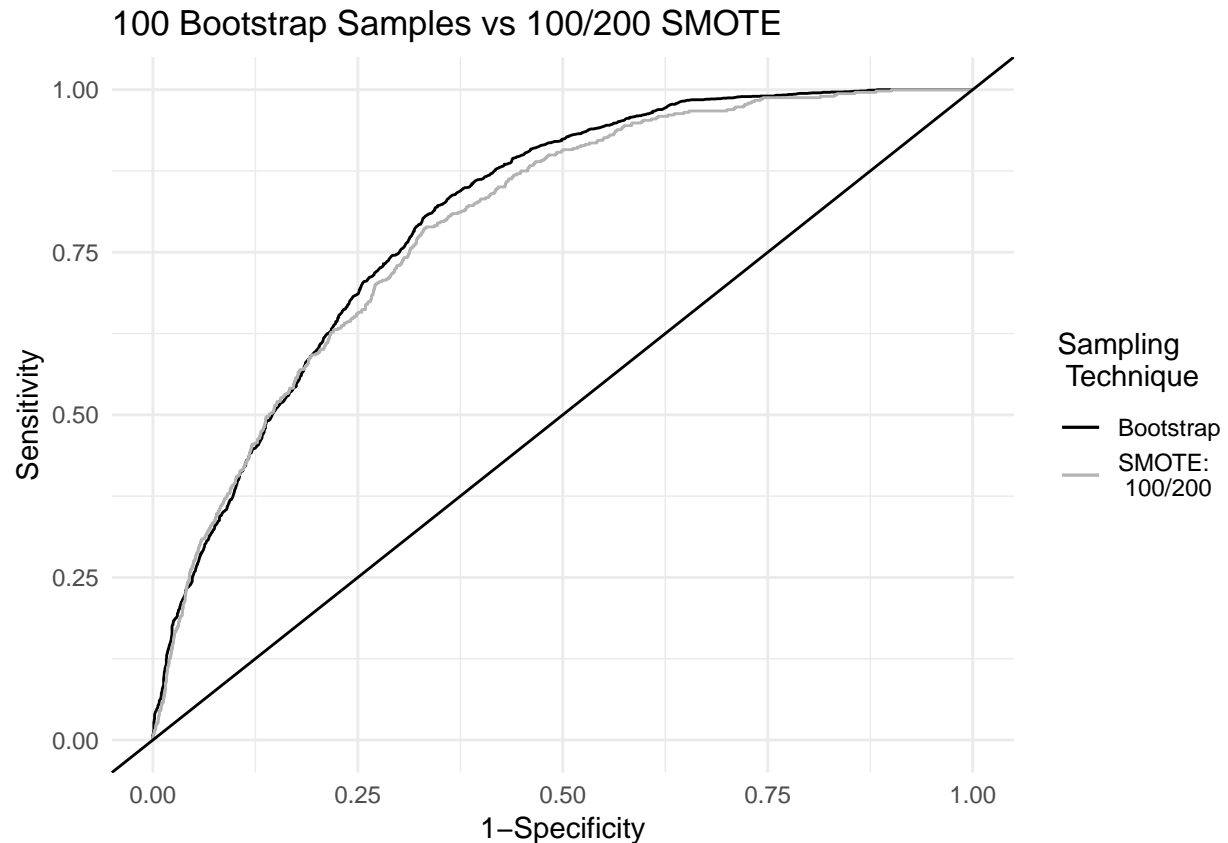
rownames(perfdat) <- NULL
colnames(perfdat) <- c("Metric", "100/200", "200/200", "200/200", "200/200",
                       "100/300", "200/300", "300/300", "400/300",
                       "Bootstraps")

kable(perfdat)
```

Metric	100/200	200/200	200/200	200/200	100/300	200/300	300/300	400/300	Bootstraps
TN	57722.00	62345.00	64076.00	65046.00	62720.00	65803.00	66564.00	67192.00	333.00
FN	190.00	246.00	271.00	289.00	239.00	287.00	305.00	314.00	109.00
FP	16078.00	11455.00	9724.00	8754.00	11080.00	7997.00	7236.00	6608.00	155.00
TP	298.00	242.00	217.00	199.00	249.00	201.00	183.00	174.00	381.00
Total	74288.00	74288.00	74288.00	74288.00	74288.00	74288.00	74288.00	74288.00	978.00
Sensitivity	0.61	0.50	0.44	0.41	0.51	0.41	0.38	0.36	0.78
Specificity	0.78	0.84	0.87	0.88	0.85	0.89	0.90	0.91	0.68
Kappa	0.02	0.03	0.03	0.03	0.03	0.03	0.03	0.04	0.46
Accuracy	0.78	0.84	0.87	0.88	0.85	0.89	0.90	0.91	0.73
Precision	0.02	0.02	0.02	0.02	0.02	0.02	0.02	0.03	0.71
FPR	0.22	0.16	0.13	0.12	0.15	0.11	0.10	0.09	0.32
FNR	0.39	0.50	0.56	0.59	0.49	0.59	0.62	0.64	0.22
FOR	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.25
NPV	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	0.75
MCC	0.08	0.08	0.07	0.07	0.08	0.08	0.07	0.07	0.46
F1	0.76	0.66	0.62	0.58	0.68	0.58	0.55	0.53	0.87

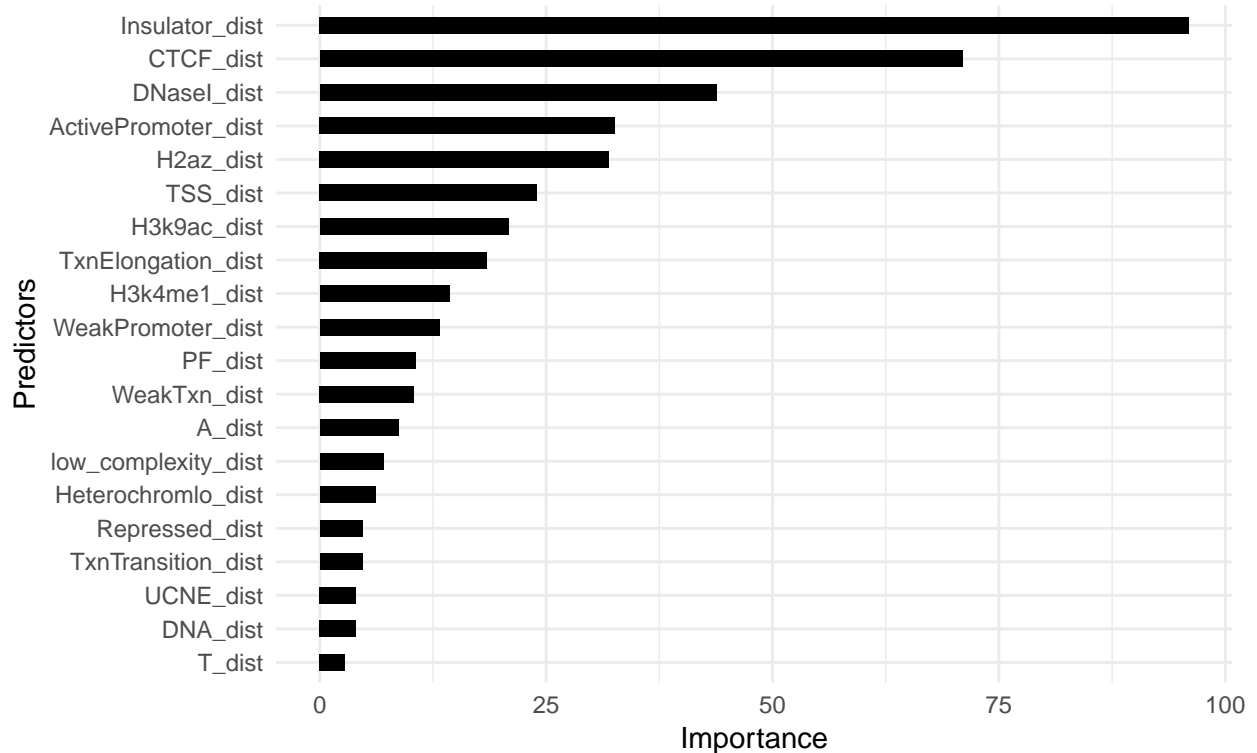
Comparing 100/200 SMOTE with Bootstrapped model

```
ggplot() +
  geom_line(aes(fpr, tpr, colour=gray(.7)[1]), rocdata.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
```


Variable Importance Plot:
100 Bootstrap Samples



```
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

