Statistics 452: Statistical Learning and Prediction

Review Part 3: Predicting Binary HUI

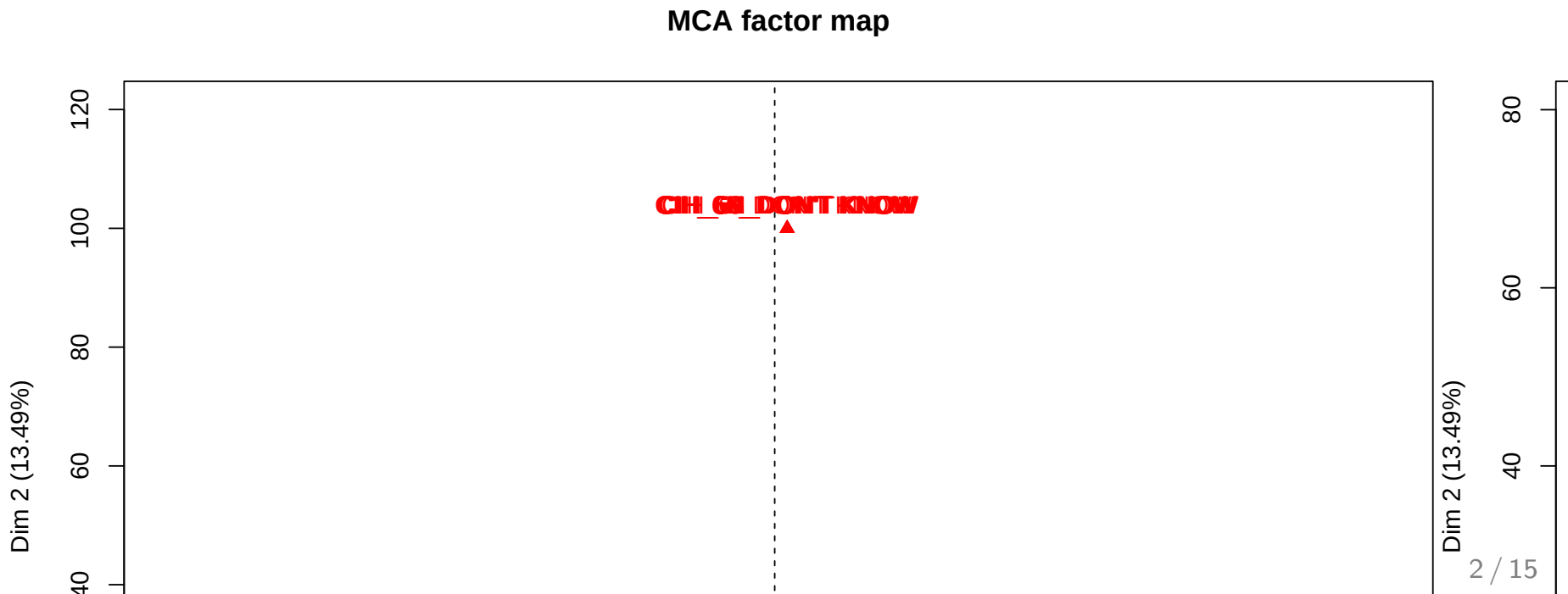
Brad McNeney

2018-11-26

Data

- Rather than analyse the quantitative HUIDHIS variable, students could break it into a binary variable that has value 0 if HUIDHIS is less than the median value of 0.905, and 1 otherwise.
- Process the data as before

```
hs <- read.csv("../../Project652/HStrain.csv")
library(dplyr)
hs <- select(hs,-starts_with("ADM"))
library(FactoMineR)
res.mca <- MCA(select(hs,starts_with("CIH")))
```



Training and test sets

```
set.seed(123)
n.train <- 7000
train <- sample(1:nrow(hs),replace=FALSE,size=n.train)
X.train <- X[train,]; Y.train <- Y[train]
X.test <- X[-train,]; Y.test <- Y[-train]
```

Logistic regression

- The R function `step()` can be used for stepwise model selection, but it is quite slow.
- Instead, we'll just fit a big logistic regression model and examine the coefficients.

```
hs.train <- data.frame(Y=Y.train,X.train)
lr <- glm(Y~.,data=hs.train,family="binomial")
round(summary(lr)$coef,4) # General health variables most important
```

```
##                          Estimate Std. Error z value Pr(>|z|)
## (Intercept)               -3.2496    30.2268 -0.1075   0.9144
## ADLDCLSMOD.IMPAIRMENT     -0.1309     0.0549 -2.3843   0.0171
## ADLDCLSNO.FUNC.IMPAIR      0.3293     0.0466  7.0684   0.0000
## ADLDCLSSEV.IMPAIRMENT     -1.1553    65.1561 -0.0177   0.9859
## ADLDCLSTOTAL.IMPAIRMENT   -0.0026     0.0513 -0.0515   0.9589
## ALCDTTMOCCASIO..DRINKER    0.0174     0.0379  0.4595   0.6459
## ALCDTTMREGULAR.DRINKER    -0.0107     0.0406 -0.2630   0.7925
## CAGDFAPOCC.OR.RARELY       0.0278     0.0328  0.8489   0.3959
## CAGDFAPREG.BASIS.DLY      -0.0134     0.0313 -0.4292   0.6678
## CAGDFAPREG.BASIS.LESS      0.0048     0.0320  0.1510   0.8799
## CAGDFAPREG.BASIS.MNTH      0.0130     0.0328  0.3948   0.6930
## CAGDFAPREG.BASIS.WK        0.0515     0.0320  1.6100   0.1074
## CCCF1HAS.NO.CHRON.CON      0.1240     0.0334  3.7086   0.0002
## CCCDCPDNOT.HAVE.COPD       0.0067     0.0345  0.1954   0.8451
```

Predictions

- Use the fitted model to make predictions on the "response" scale and then classify as "1" if greater than a threshold.
- Could explort different thresholds and calculate true- and false-positive rates, or AUC.
  - I'll just use a threshold of 0.5

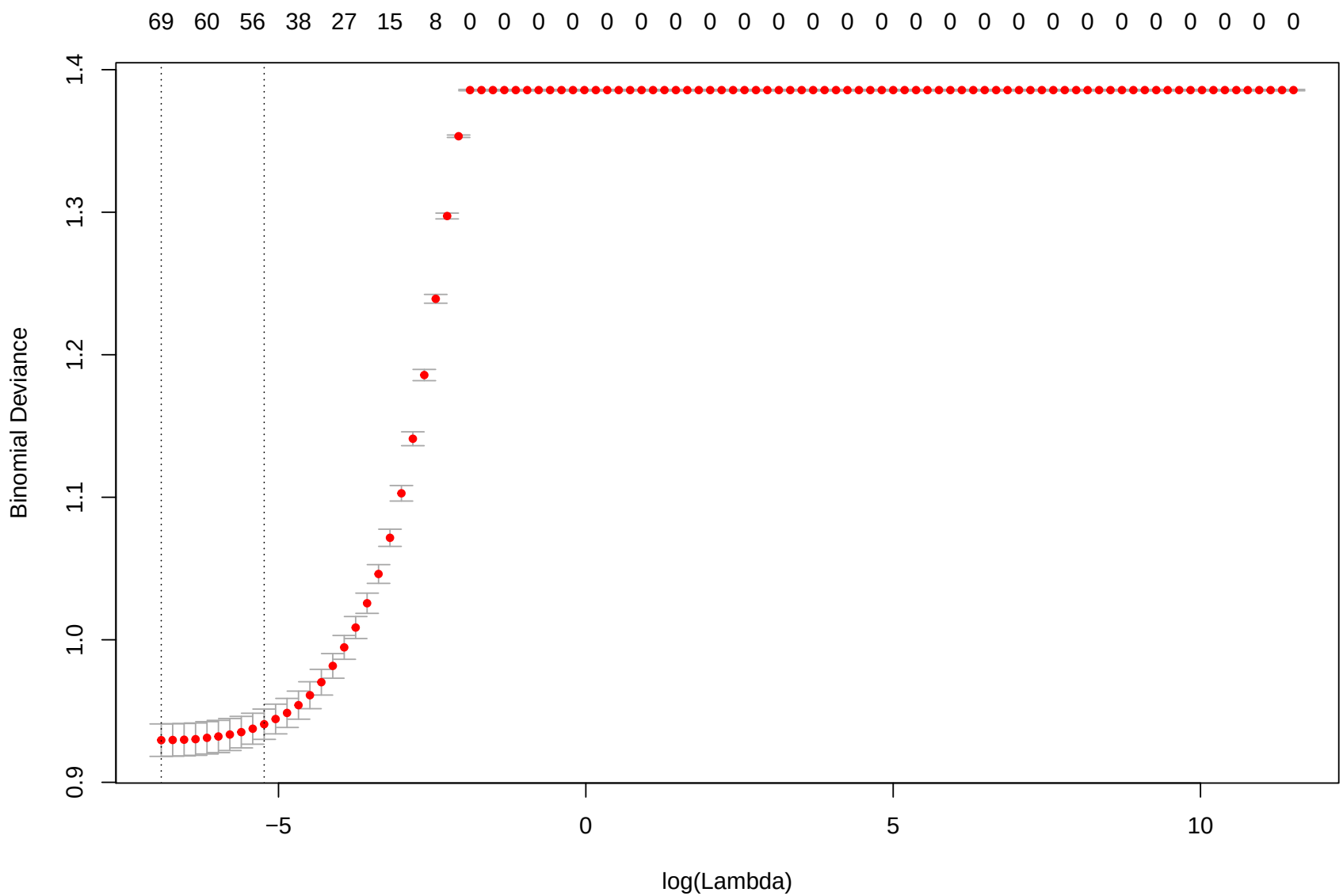
```
hs.test <- data.frame(Y=Y.test,X.test)
pred.test <- predict(lr,newdata=hs.test,type="response")
pred.test <- as.numeric(pred.test > 0.5)
mean(pred.test != Y.test)
```

```
## [1] 0.233
```

Lasso

```
library(glmnet)
lambdas <- 10^{seq(from=-3,to=5,length=100)}
cv.lafit <- cv.glmnet(as.matrix(X.train),Y.train,
                      family="binomial",alpha=1,lambda=lambdas)
```

```
plot(cv.lafit)
```




```
la.best.lam <- cv.lafit$lambda.1se
```

Lasso coefficients

```
ll <- glmnet(as.matrix(X.train),Y.train,alpha=1,lambda=la.best.lam)
coef(ll) # General health, pain levels, sleep, satisfaction with life variables important
```

```
## 81 x 1 sparse Matrix of class "dgCMatrix"
##                                      s0
## (Intercept)                0.4862202958
## ADLDCLSMOD IMPAIRMENT     -0.0025533782
## ADLDCLSNO FUNC IMPAIR      0.0465507390
## ADLDCLSSEV IMPAIRMENT      .
## ADLDCLSTOTAL IMPAIRMENT    .
## ALCDTTMOCCASIO. DRINKER    .
## ALCDTTMREGULAR DRINKER     .
## CAGDFAPOCC OR RARELY       0.0002132181
## CAGDFAPREG BASIS DLY      -0.0014915438
## CAGDFAPREG BASIS LESS      .
## CAGDFAPREG BASIS MNTH      .
## CAGDFAPREG BASIS WK        0.0036274983
## CCCF1HAS NO CHRON CON      0.0298670924
## CCCDCPDNOT HAVE COPD       .
## CR1FRHCREC FORMAL H C      .
## CR2DTHCDID NOT REC H C     0.0159246930
## CR2DTHCFORMAL H C ONLY     .
## CR2DTHCINFORMAL H C ONL    .
## CR2DFAROCC OR RARELY       .
## CR2DFARREG BASIS DLY      -0.0027353294
## CR2DFARREG BASIS LESS     -0.0051206596
## CR2DFARREG BASIS MNTH      0.0034721492
## CR2DFARREG BASIS WK        .
## DPSDSF                    -0.0016910299
## EDUDR04OTHER POST-SEC.     .
## EDUDR04POST-SEC. GRAD.     0.0113671582
## EDUDR04SECONDARY GRAD.     .
## FALG022                    .
## FALG023                   -0.0016570794
```

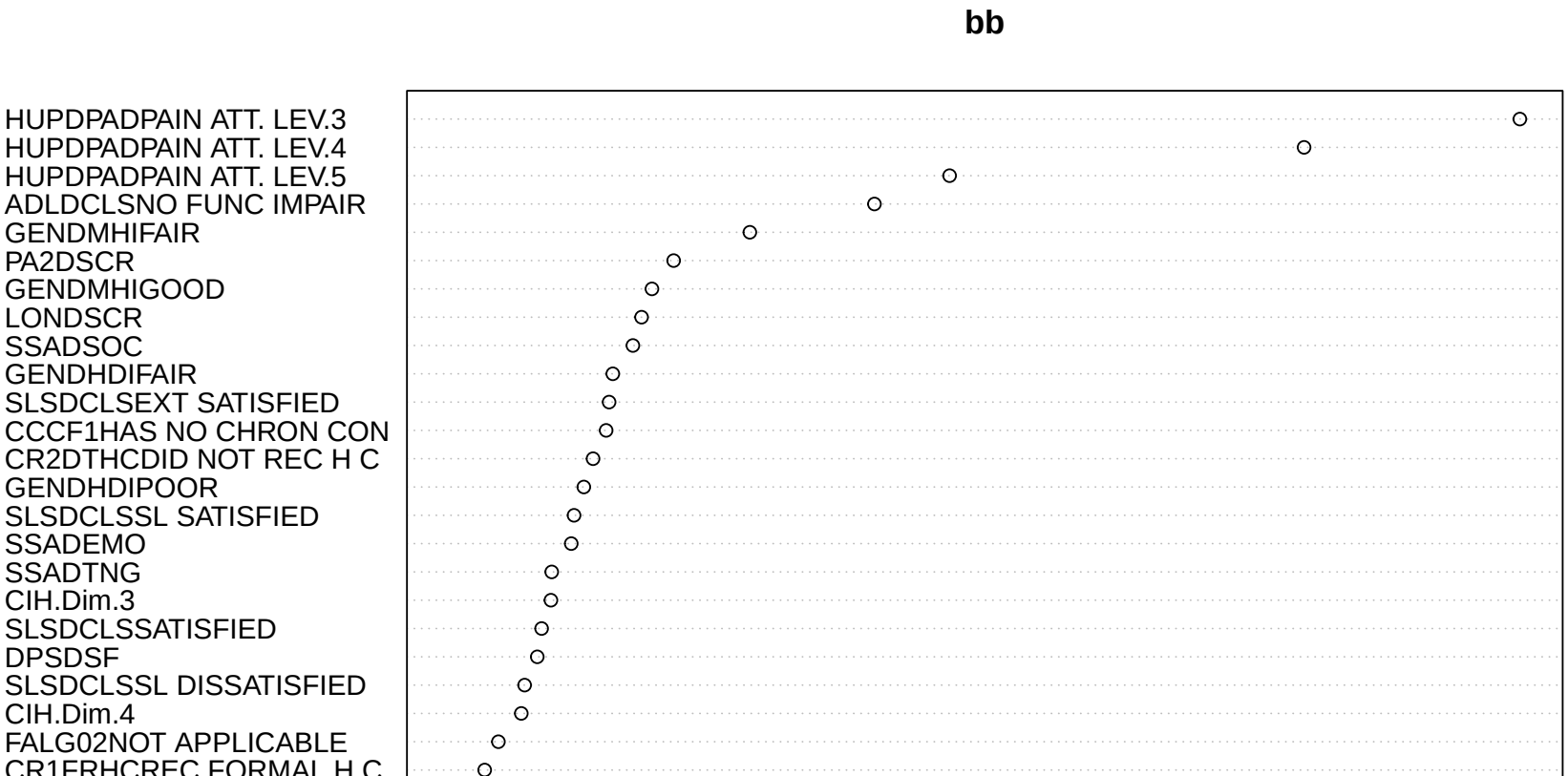
```r
pred.test <- predict(ll,as.matrix(X.test),type="response")
pred.test <- as.numeric(pred.test > 0.5)
mean((Y.test-pred.test)^2)
```

```
## [1] 0.2373333
```

Random Forest

- The response should be a factor to build classification trees.

```
library(randomForest)
set.seed(1)
Y.tr.fact <- as.factor(Y.train); Y.te.fact <- as.factor(Y.test)
bb <- randomForest(X.train,y=Y.tr.fact,xtest=X.test,
          ytest=Y.te.fact,ntree=200,
          mtry=sqrt(ncol(X.train)),importance=TRUE)
varImpPlot(bb,type=1)
```



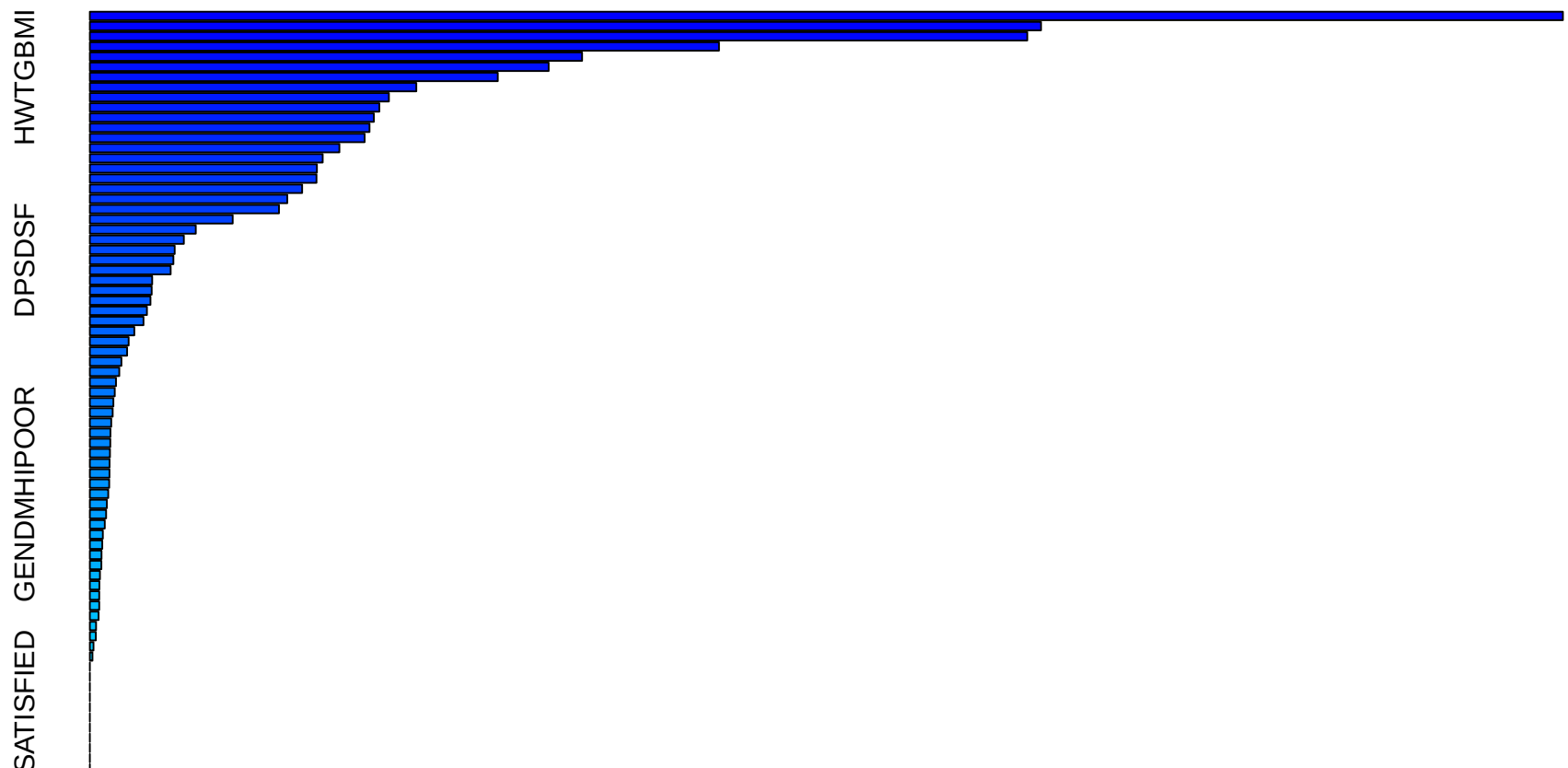
```r
pred.test <- bb$test$predicted
mean((Y.te.fact != pred.test))
```

```
## [1] 0.2413333
```

Boosting

- I thought a factor response would cause `gbm` to build classification trees, but I'm getting errors (`NaN` predictions).
- Instead, revert to 0/1 `Y`.

```
library(gbm)
hs.train <- data.frame(Y=Y.train,X.train)
hboost <- gbm(Y ~ ., data=hs.train,interaction.depth=2,
              n.trees=500,distribution="bernoulli")
summary(hboost)
```



```r
library(gbm)
hs.test <- data.frame(HUIDHSI=Y.test,X.test)
pred.test <- predict(hboost,newdata=hs.test, n.trees=200,type="response")
pred.test <- as.numeric(pred.test > 0.5)
mean((Y.test!=pred.test))
```

```
## [1] 0.2293333
```

SVM

- We discussed several kernels.
- I'll use radial without any tuning.
  - Computation: with $n = 7000$, the pairwise distance matrix is 7000×7000.
  - Recall `tune()` function if you want to tune.

```
library(e1071)
set.seed(123)
ss <- svm(Y~.,data=hs.train,type="C-classification",
          cost=1,kernel="radial",gamma=1)
preds <- predict(ss,newdata=hs.test)
mean(preds != Y.test)
```

```
## [1] 0.473
```

The winner

- The winner is boosting with interaction depth 2.
- Fit the winner to all data – this is my $\hat{f}$ to predict the hold-out test data on Monday.

```
hs <- data.frame(HUIDHSI=Y,X)
winner <- gbm(HUIDHSI ~ ., data=hs,
              n.trees=200,interaction.depth=2,
              distribution="bernoulli")
```