Part2 Hypoglycemic Classification

Kim Tiller

April 19, 2021

Load Data, prepare training and validation, performance function

```
#Load Data and split into train and validate
drug = read.csv("hypoglycemic.csv")
drug$hypoglycemic <- as.factor(drug$hypoglycemic)
drug$asthma <- as.factor(drug$asthma)
drug$cad <- as.factor(drug$cad)
drug$chf <- as.factor(drug$chf)
drug$copd <- as.factor(drug$copd)
drug$cardio_respiratory_arrest <- as.factor(drug$cardio_respiratory_arres)
drug$cerebro_vascular <- as.factor(drug$cerebro_vascular)
drug$decubitus_ulcer <- as.factor(drug$decubitus_ulcer)
drug$delirium <- as.factor(drug$delirium)
#drug$developmental_disability <- as.factor(drug$developmental_disabilit)
drug$mental_health <- as.factor(drug$mental_health)
#drug$pregnancy <- as.factor(drug$pregnancy)
drug$renal <- as.factor(drug$renal)
drug$substance_abuse <- as.factor(drug$substance_abuse)
drug$vascular_disease <- as.factor(drug$vascular_disease)

#Set Training data to contain 70% of records
set.seed(123)
getSamp <- sample(nrow(drug), .7*nrow(drug),replace=F)
train <- drug[getSamp,]
valid = drug[-getSamp,]

#function for evaluating trees
performance <- function(table, n=2){
tn <- table[1,1]
fp <- table[1,2]
fn <- table[2,1]
tp <- table[2,2]
sensitivity <- tp/(tp+fn)
specificity <- tn/(tn+fp)
ppv <- tp/(tp+fp)
npv <- tn/(tn+fn)
acc <- (tp+tn)/(tp+tn+fp+fn)

result <- paste("Sensitivity (True Postive Rate)= ", round(sensitivity, n),
                "\nSpecificity (True Negative Rate) = ", round(specificity, n),
```

```
                 "\nFalse Negative Rate = ", round(1-sensitivity,n),
                 "\nPositives Predictive Value (odds of positive if postive prediction) = ", round(ppv, n
                 "\nNegative Predictive value (odds of negative if negative prediction) = ", round(npv, n
                 "\nAccuracy = ", round(acc, n), "\n", sep="")
cat(result)
}

#How many hypoglycemic members?
summary(drug$hypoglycemic)
```

```
##    0    1
## 1718  403
```

Explore training options with oversampling, undersampling, and synthetic data

(SKIP this in Final Output: Use Train instead of bal_train for all models Note: Evaluation was performed using several over and undersampling techniques as well as synthetic data using the ROSE package. These techniques produced very low sensitivity and high rate of false negatives. RandomForest mode performed best with undersampling. However, for the purposes of this exercise, the original slightly undersampled (30% of diabetes members without hypoglycemia) was used to compare models. It is noted that these models may include overfitting.

```
#bal_train <- ovun.sample(hypoglycemic ~ .-mem_key, data = train, method = "over", N= 2000, seed = 1)$d
#summary(bal_train$hypoglycemic)
```

```
#syn_train <- ROSE(hypoglycemic ~ .-mem_key, data = train, seed=1)$data
#summary(syn_train$hypoglycemic)
```

Decision Tree using tree library

```
#Decision Tree using tree library
#Train the tree
library(tree)
set.seed(123)
dtree = tree(hypoglycemic ~ . -mem_key, data = train)
summary(dtree)
```

```
##
## Classification tree:
## tree(formula = hypoglycemic ~ . - mem_key, data = train)
## Variables actually used in tree construction:
## [1] "a_ins_rapid"    "op_visits"      "Sup_INS"        "big"
## [5] "AnitInfect"     "raw_risk_score" "a_ins_long"
## Number of terminal nodes:  9
## Residual mean deviance:  0.748 = 1103 / 1475
## Misclassification error rate: 0.1509 = 224 / 1484
```

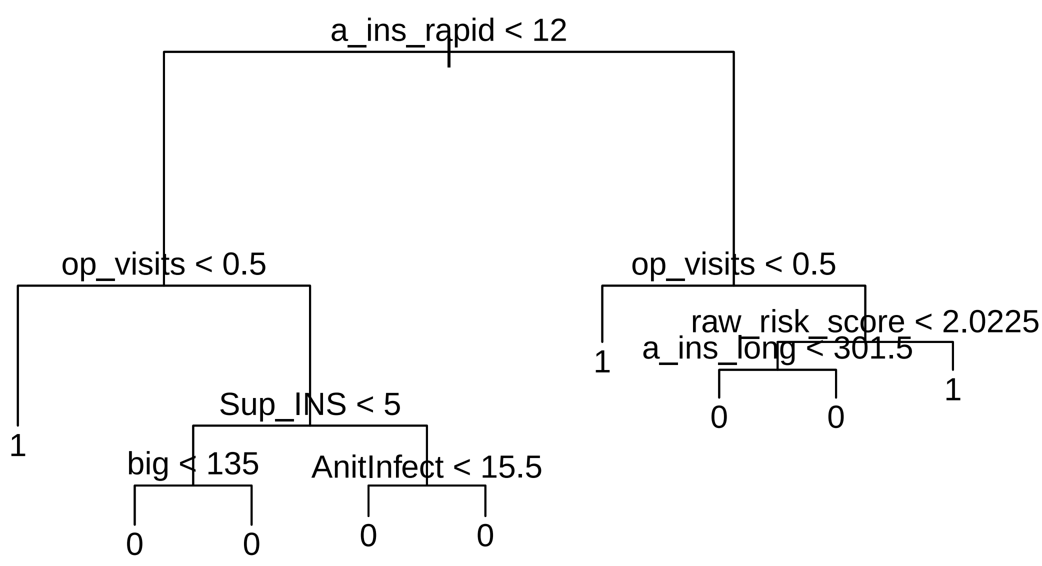
```r
#predict using validation set
dt.pred=predict(dtree,valid,type="class")
dt.perf <- table(dt.pred,valid$hypoglycemic)
dt.perf
```

```
##
## dt.pred   0   1
##       0 491  88
##       1  23  35
```

```r
#validate
performance(dt.perf)
```

```
## Sensitivity (True Postive Rate)= 0.6
## Specificity (True Negative Rate) = 0.85
## False Negative Rate = 0.4
## Positives Predictive Value (odds of positive if postive prediction) = 0.28
## Negative Predictive value (odds of negative if negative prediction) = 0.96
## Accuracy = 0.83
```

```r
#plot the tree for better understanding
plot(dtree)
text(dtree,pretty=0) #label nodes with text
```



```r
print(dtree)
```

```
## node), split, n, deviance, yval, (yprob)
##       * denotes terminal node
##
##  1) root 1484 1437.00 0 ( 0.81132 0.18868 )
##    2) a_ins_rapid < 12 1264 1006.00 0 ( 0.86392 0.13608 )
##      4) op_visits < 0.5 35   37.63 1 ( 0.22857 0.77143 ) *
##      5) op_visits > 0.5 1229  892.00 0 ( 0.88202 0.11798 )
##       10) Sup_INS < 5 1039  644.70 0 ( 0.90664 0.09336 )
##         20) big < 135 314  275.30 0 ( 0.84076 0.15924 ) *
##         21) big > 135 725  348.10 0 ( 0.93517 0.06483 ) *
##       11) Sup_INS > 5 190  214.80 0 ( 0.74737 0.25263 )
##         22) AnitInfect < 15.5 136  123.60 0 ( 0.83088 0.16912 ) *
##         23) AnitInfect > 15.5 54   74.56 0 ( 0.53704 0.46296 ) *
##    3) a_ins_rapid > 12 220  304.90 0 ( 0.50909 0.49091 )
##      6) op_visits < 0.5 20    0.00 1 ( 0.00000 1.00000 ) *
##      7) op_visits > 0.5 200  274.40 0 ( 0.56000 0.44000 )
##       14) raw_risk_score < 2.0225 119  150.50 0 ( 0.67227 0.32773 )
##         28) a_ins_long < 301.5 76  104.50 0 ( 0.55263 0.44737 ) *
##         29) a_ins_long > 301.5 43   30.91 0 ( 0.88372 0.11628 ) *
##       15) raw_risk_score > 2.0225 81  108.70 1 ( 0.39506 0.60494 ) *
```

Prune dtree to reduce complexity

```r
#use 10-fold CV to choose optimal # of leaves
set.seed(123)
dtree.cv = cv.tree(dtree, FUN = prune.misclass)
dtree.cv
```

```
## $size
## [1] 9 5 1
##
## $dev
## [1] 256 254 275
##
## $k
## [1] -Inf    0   14
##
## $method
## [1] "misclass"
##
## attr(,"class")
## [1] "prune"         "tree.sequence"
```

```r
#plot(dtree.cv)

#Extract optimal number of leaves
min(dtree.cv$dev)  #min deviance
```

```
## [1] 254
```

```r
which(dtree.cv$dev == min(dtree.cv$dev)) #which records are equal to minimum
```

```
## [1] 2
```

```r
dtree.cv$size[ which(dtree.cv$dev == min(dtree.cv$dev))] #what size corresponds to min error
```

```
## [1] 5
```

Prune dtree and evaluate

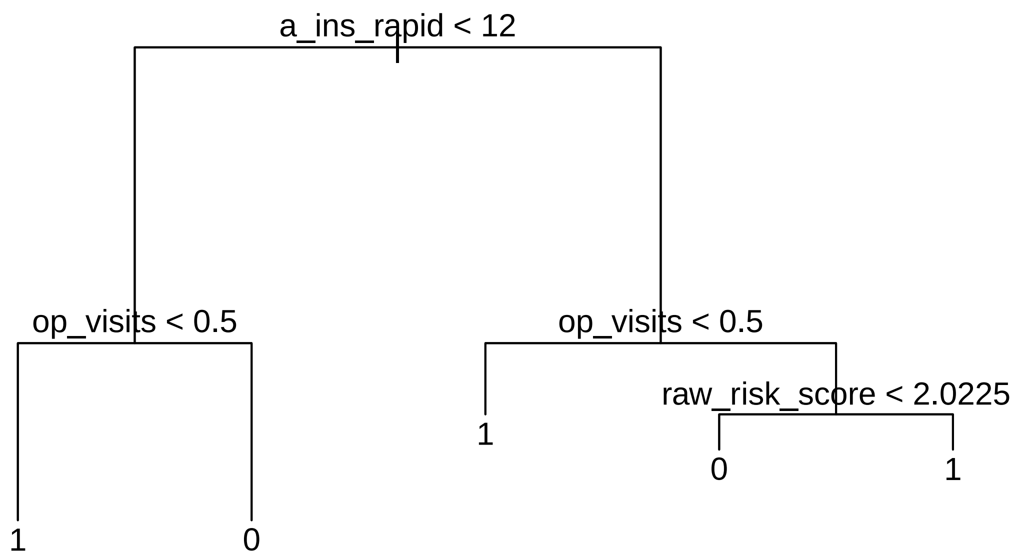
```r
#Prune down to the optimal leaves
set.seed(123)
prune.dtree = prune.misclass(dtree,best=5)

#Predict using Pruned tree
dt.pred2=predict(prune.dtree,valid,type="class")
dt.perf2 <- table(dt.pred2,valid$hypoglycemic)
dt.perf2
```

```
##
## dt.pred2   0   1
##        0 491  88
##        1  23  35
```

```r
#Results
performance(dt.perf2)
```

```
## Sensitivity (True Postive Rate)= 0.6
## Specificity (True Negative Rate) = 0.85
## False Negative Rate = 0.4
## Positives Predictive Value (odds of positive if postive prediction) = 0.28
## Negative Predictive value (odds of negative if negative prediction) = 0.96
## Accuracy = 0.83
```

```r
plot(prune.dtree)
text(prune.dtree,pretty=0)
```




```r
summary(prune.dtree)
```

```
##
## Classification tree:
## snip.tree(tree = dtree, nodes = c(5L, 14L))
## Variables actually used in tree construction:
## [1] "a_ins_rapid"    "op_visits"      "raw_risk_score"
## Number of terminal nodes:  5
## Residual mean deviance:  0.8038 = 1189 / 1479
## Misclassification error rate: 0.1509 = 224 / 1484
```

```r
print(prune.dtree)
```

```
## node), split, n, deviance, yval, (yprob)
##       * denotes terminal node
##
##  1) root 1484 1437.00 0 ( 0.8113 0.1887 )
##    2) a_ins_rapid < 12 1264 1006.00 0 ( 0.8639 0.1361 )
##      4) op_visits < 0.5 35   37.63 1 ( 0.2286 0.7714 ) *
##      5) op_visits > 0.5 1229  892.00 0 ( 0.8820 0.1180 ) *
##    3) a_ins_rapid > 12 220  304.90 0 ( 0.5091 0.4909 )
##      6) op_visits < 0.5 20    0.00 1 ( 0.0000 1.0000 ) *
##      7) op_visits > 0.5 200  274.40 0 ( 0.5600 0.4400 )
##       14) raw_risk_score < 2.0225 119  150.50 0 ( 0.6723 0.3277 ) *
##       15) raw_risk_score > 2.0225 81  108.70 1 ( 0.3951 0.6049 ) *
```

Performance is the same with 5 terminal nodes

Try another tree using rpart library

```
#Decision Tree Using rpart and prepare to prune
library(rpart)
set.seed(123)
rtree <- rpart(hypoglycemic ~ . -mem_key, data = train, method="class")

#Predict
rt.pred <- predict(rtree, valid, type="class")
rt.perf <- table(rt.pred,valid$hypoglycemic)
rt.perf
```

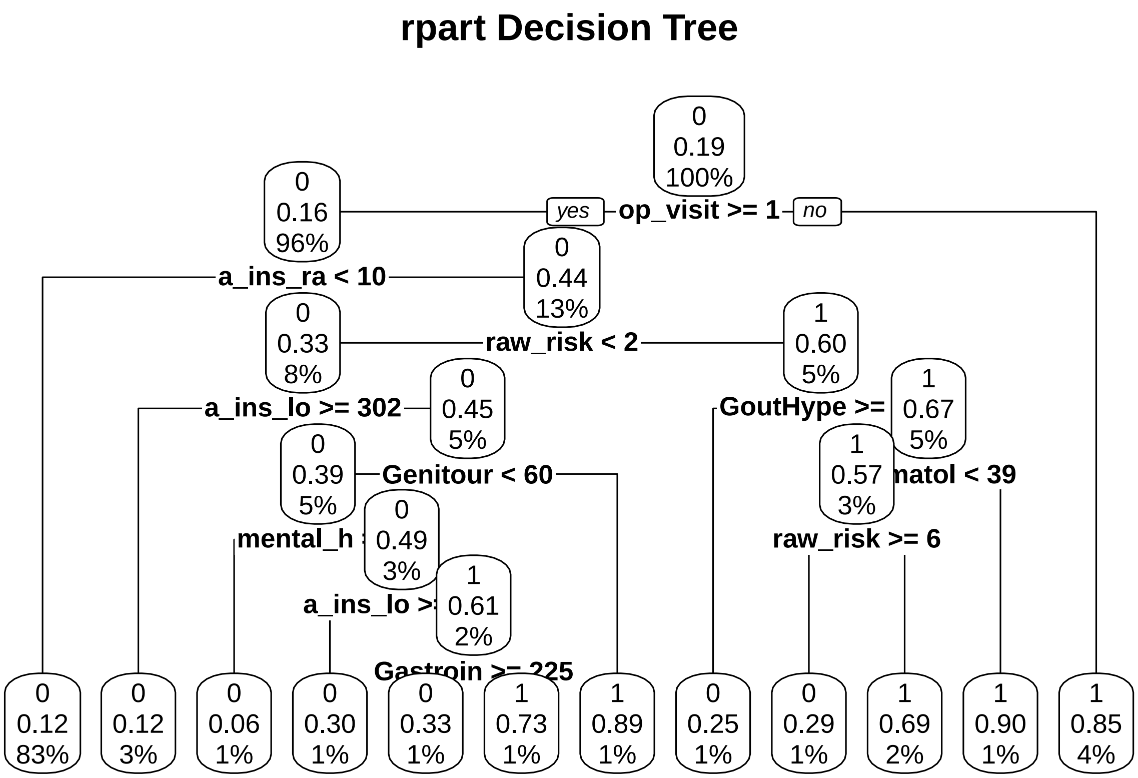
```
##
## rt.pred   0   1
##       0 494  85
##       1  20  38
```

```
#Performance
performance(rt.perf)
```

```
## Sensitivity (True Postive Rate)= 0.66
## Specificity (True Negative Rate) = 0.85
## False Negative Rate = 0.34
## Positives Predictive Value (odds of positive if postive prediction) = 0.31
## Negative Predictive value (odds of negative if negative prediction) = 0.96
## Accuracy = 0.84
```

I am interested in High Sensitivity and low False Negatives. This tree is sliglty better than dtree with default parameters.

```
#Plot
library(rpart.plot)
prp(rtree, type=2, extra = "auto", fallen.leaves = TRUE, cex = .8, uniform = TRUE,compress = TRUE, main
```




```r
print(rtree)
```

```
## n= 1484
##
## node), split, n, loss, yval, (yprob)
##       * denotes terminal node
##
##   1) root 1484 280 0 (0.8113208 0.1886792)
##     2) op_visits>=0.5 1429 233 0 (0.8369489 0.1630511)
##       4) a_ins_rapid< 10 1229 145 0 (0.8820179 0.1179821) *
##       5) a_ins_rapid>=10 200  88 0 (0.5600000 0.4400000)
##        10) raw_risk_score< 2.0225 119  39 0 (0.6722689 0.3277311)
##          20) a_ins_long>=301.5 43   5 0 (0.8837209 0.1162791) *
##          21) a_ins_long< 301.5 76  34 0 (0.5526316 0.4473684)
##            42) Genitour< 59.5 67  26 0 (0.6119403 0.3880597)
##              84) mental_health=1 16   1 0 (0.9375000 0.0625000) *
##              85) mental_health=0 51  25 0 (0.5098039 0.4901961)
##               170) a_ins_long>=186 20   6 0 (0.7000000 0.3000000) *
##               171) a_ins_long< 186 31  12 1 (0.3870968 0.6129032)
##                 342) Gastroint>=225 9   3 0 (0.6666667 0.3333333) *
##                 343) Gastroint< 225 22   6 1 (0.2727273 0.7272727) *
##            43) Genitour>=59.5 9   1 1 (0.1111111 0.8888889) *
##        11) raw_risk_score>=2.0225 81  32 1 (0.3950617 0.6049383)
##          22) GoutHyper>=45 12   3 0 (0.7500000 0.2500000) *
##          23) GoutHyper< 45 69  23 1 (0.3333333 0.6666667)
```

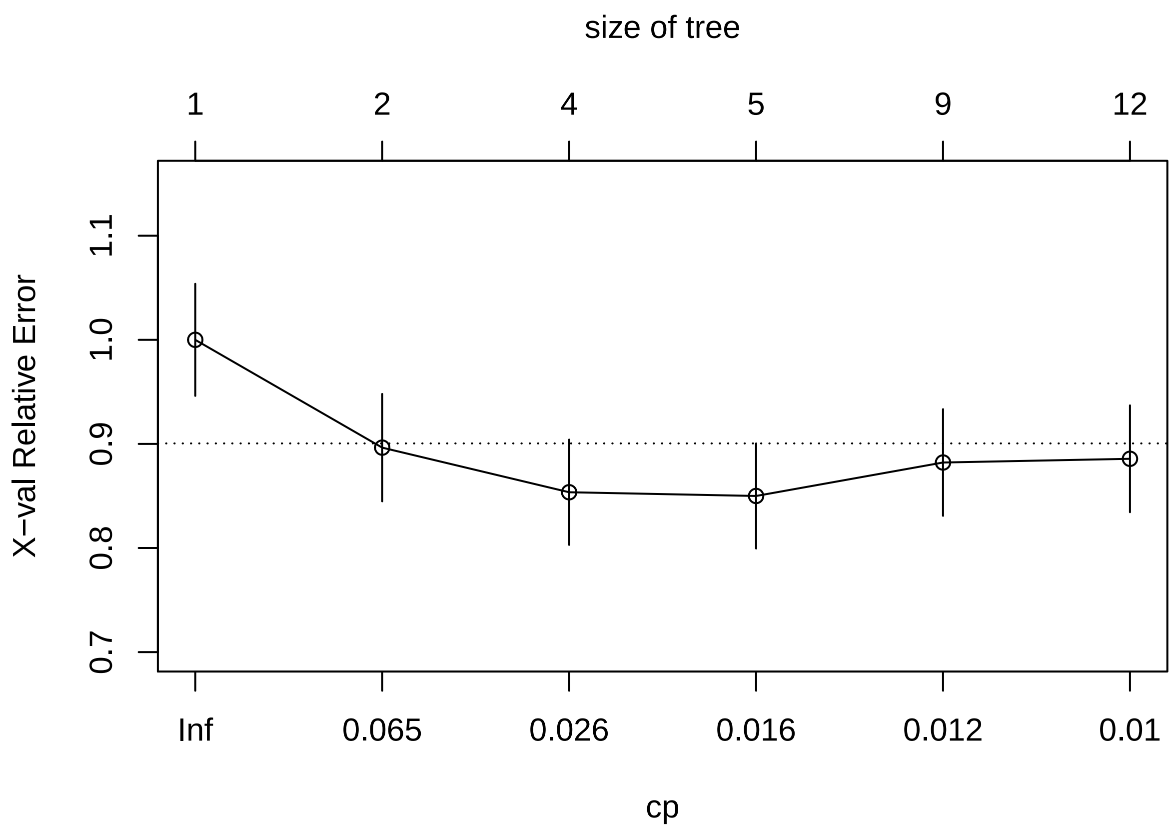
```
##            46) Dermatol< 38.5 49  21 1 (0.4285714 0.5714286)
##              92) raw_risk_score>=5.972 14   4 0 (0.7142857 0.2857143) *
##              93) raw_risk_score< 5.972 35  11 1 (0.3142857 0.6857143) *
##            47) Dermatol>=38.5 20   2 1 (0.1000000 0.9000000) *
##     3) op_visits< 0.5 55   8 1 (0.1454545 0.8545455) *
```

Rpart tree is more complex than dtree. ##Prepare to Prune rtree

```
#prepare to prune rtree
set.seed(123)
rtree$cptable
```

```
##           CP nsplit rel error    xerror       xstd
## 1 0.13928571      0 1.0000000 1.0000000 0.05382912
## 2 0.03035714      1 0.8607143 0.8964286 0.05157550
## 3 0.02142857      3 0.8000000 0.8535714 0.05057183
## 4 0.01250000      4 0.7785714 0.8500000 0.05048618
## 5 0.01071429      8 0.7285714 0.8821429 0.05124581
## 6 0.01000000     11 0.6964286 0.8857143 0.05132868
```

```
plotcp(rtree)
```



Smallest xerror = .8500 with xerror between .8 and .9, all of the xerrors fall within this range Try cp = 0.0214 or .0125 or .01071 ##Prune rtree

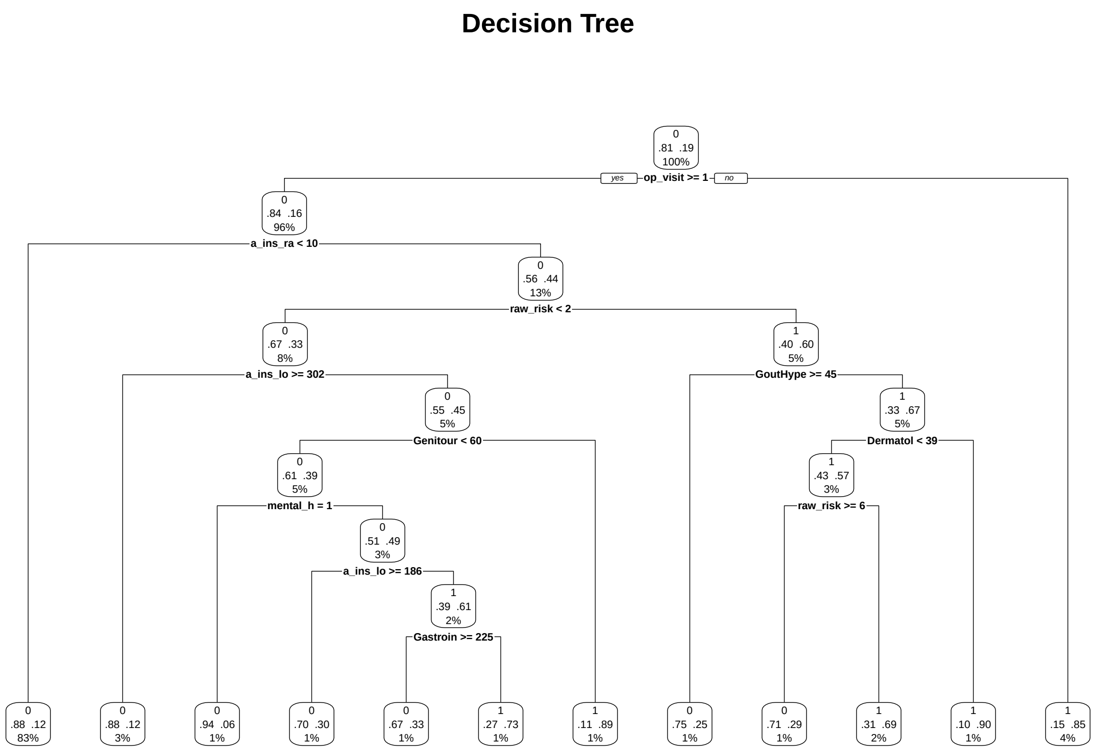
```r
#Prune the rpart tree and validate
rtree.pruned <-prune(rtree,cp=.01071) #better performance at cp=.01071
rtree.pred2 <- predict(rtree.pruned, valid, type="class")
rtree.perf2 <- table(valid$hypoglycemic, rtree.pred2, dnn=c("Actual", "predicted"))

#Performance
rtree.perf2
```

```
##       predicted
## Actual   0   1
##      0 494  20
##      1  85  38
```

```r
performance(rtree.perf2)
```

```
## Sensitivity (True Postive Rate)= 0.31
## Specificity (True Negative Rate) = 0.96
## False Negative Rate = 0.69
## Positives Predictive Value (odds of positive if postive prediction) = 0.66
## Negative Predictive value (odds of negative if negative prediction) = 0.85
## Accuracy = 0.84
```

```r
#Plot
library(rpart.plot)
prp(rtree.pruned, type=2, extra = 104, fallen.leaves = TRUE, main="Decision Tree")
```



Pruning rpart tree lowered sensitivity and increased false negative rate.

RandomForest

** rftree **

```
library(randomForest)
```

```
## randomForest 4.6-14
```

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

```
rftree = randomForest(hypoglycemic~. -mem_key, data=train, mtry=11,ntree=1000, importance=T, Xtest = val
rftree
```

```
##
## Call:
##  randomForest(formula = hypoglycemic ~ . - mem_key, data = train,      mtry = 11, ntree = 1000, impor
##                Type of random forest: classification
##                      Number of trees: 1000
## No. of variables tried at each split: 11
##
##         OOB estimate of  error rate: 16.11%
## Confusion matrix:
##      0  1 class.error
## 0 1183 21  0.01744186
## 1  218 62  0.77857143
```

```
performance(rftree$confusion)
```

```
## Sensitivity (True Postive Rate)= 0.22
## Specificity (True Negative Rate) = 0.98
## False Negative Rate = 0.78
## Positives Predictive Value (odds of positive if postive prediction) = 0.75
## Negative Predictive value (odds of negative if negative prediction) = 0.84
## Accuracy = 0.84
```

```
print(rftree)
```


```
##
## Call:
##  randomForest(formula = hypoglycemic ~ . - mem_key, data = train,      mtry = 11, ntree = 1000, impor
##                Type of random forest: classification
##                      Number of trees: 1000
## No. of variables tried at each split: 11
##
##         OOB estimate of  error rate: 16.11%
## Confusion matrix:
##      0  1 class.error
## 0 1183 21  0.01744186
## 1  218 62  0.77857143
```

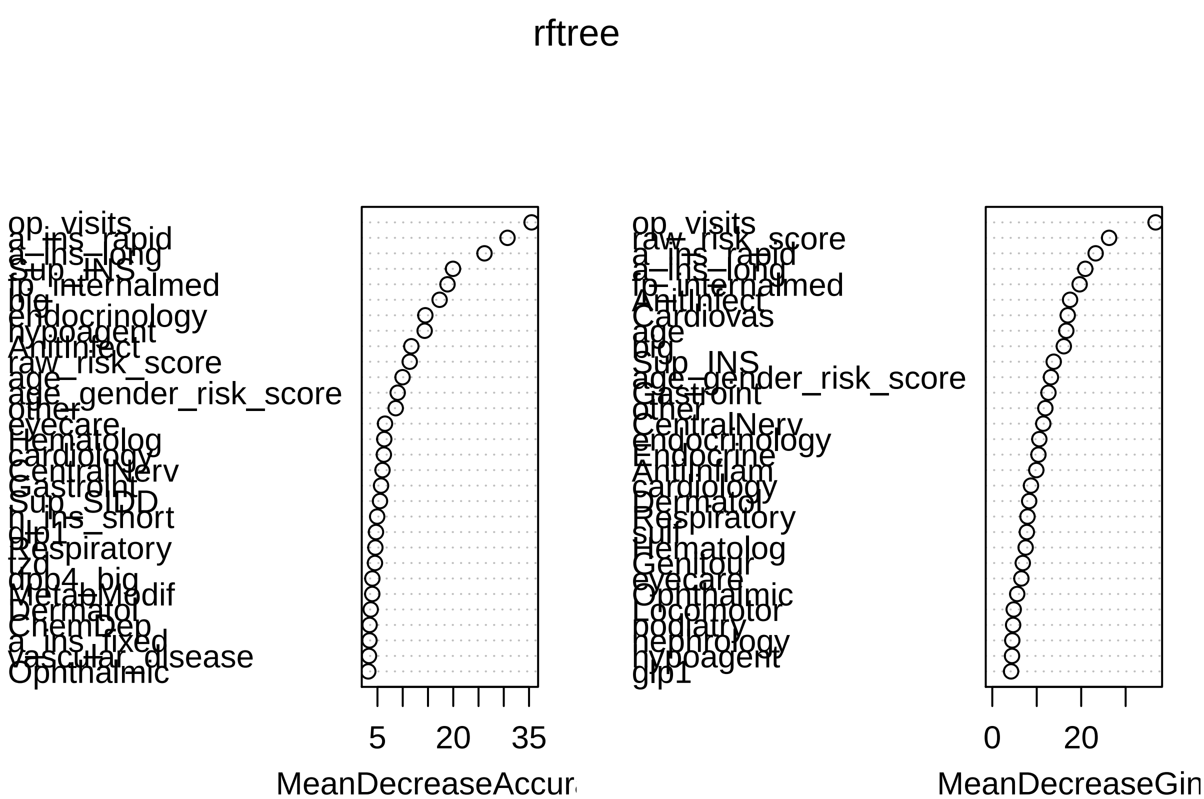
```r
#Evaluate
importance(rftree, type=2) #node impurity
```

```
##                              MeanDecreaseGini
## age                               16.644312690
## grb                                0.000000000
## dpp4_tzd                           0.000000000
## sglt2                              3.723822517
## sglt2_big                          0.764211101
## dpp4_big                           0.992248664
## h_ins_fixed                        0.138122518
## h_ins_interm                       0.898809801
## h_ins_rapid                        0.000000000
## h_ins_short                        1.534512136
## a_ins_fixed                        1.990839407
## a_ins_long                        20.920074284
## a_ins_rapid                       23.248173556
## aat                                0.000000000
## glp1                               4.230202011
## dpp4                               3.696734456
## meg                                0.718404608
## sulf                               7.778192522
## big                               16.090816114
## alphagi                            0.398189743
## tzd                                2.786992650
## tzd_sulf                           0.000000000
## meg_big                            0.000000000
## sulf_big                           0.819434929
## tzd_big                            0.008842777
## dra                                0.000000000
## slgt2_dpp4                         0.030905005
## ins_glp1                           0.505229989
## slgt2_dpp_big                      0.000000000
## hypoagent                          4.439099401
## Sup_BGT                            0.000000000
## Sup_BGKCTS                         0.000000000
## Sup_GMTS                           0.000000000
## Sup_INS                           13.820045901
## Sup_SIDD                           0.883386353
## Sup_SIP                            0.000000000
## Sup_UGT                            0.000000000
## Sup_UGACT                          0.000000000
## Sup_UKT                            0.000000000
## Sup_GMI                            0.000000000
## Alt_ther                           0.000000000
## AntiInflam                         9.905254605
## Anesthet                           0.000000000
## Anorect                            0.460135976
## Antidotes                          0.759533274
## AnitInfect                        17.499735349
## Antineopl                          1.445912281
## Antisept                           0.261442477
## Biologic                           2.024325694
```

```
## Cardiovas                    16.971124114
## CentralNerv                  11.472856428
## ChemDep                       1.110089223
## ChemPharm                     0.000000000
## Cognitive                     1.167943135
## Contracept                    0.000000000
## Dermatol                      8.305986431
## Diagnostic                    0.000000000
## ErectileDys                   0.012324660
## EatingDis                     0.036712542
## Electrolyte                   3.330596947
## Endocrine                    10.379560860
## Enzymes                       0.000000000
## Gastroint                    12.632838729
## Genitour                      6.863145637
## GoutHyper                     2.809973892
## Hematolog                     7.502554555
## Hepatobil                     0.000000000
## Histamine                     0.000000000
## Immunosup                     0.551914734
## Locomotor                     4.826308884
## OthrMedSup                    0.092169623
## MetabDisEnzyme                0.000000000
## MetabModif                    0.434994070
## MouthThrDen                   2.403077019
## MultSclerosis                 0.082892677
## Ophthalmic                    5.606562871
## OrganPresSol                  0.000000000
## Otic                          1.175700531
## RenalRepl                     0.000000000
## Respiratory                   7.916600210
## SepsisSynd                    0.000000000
## Vaginal                       1.517312616
## op_visits                    36.729377086
## cardiology                    8.706024407
## dermatology                   4.181666268
## endocrinology                10.583165064
## fp_internalmed               19.650052383
## mentalhealth                  1.171690969
## eyecare                       6.544349113
## urology                       3.444688155
## vascularsurg                  0.727408834
## rheumatology                  0.878764614
## podiatry                      4.689763672
## osteopathic                   0.000000000
## nephrology                    4.498501178
## orthopedics                   3.224926689
## other                        11.937641655
## age_gender_risk_score        13.189937443
## raw_risk_score               26.278626584
## asthma                        1.850676498
## cad                           2.668976028
## chf                           2.284381244
## copd                          1.934750894
```

```
## cardio_respiratory_arrest      1.588646329
## cerebro_vascular               2.272360193
## decubitus_ulcer                0.551213792
## delirium                       0.468692082
## developmental_disability       0.000000000
## mental_health                  2.591759429
## pregnancy                      0.000000000
## renal                          2.720557416
## substance_abuse                1.426596931
## vascular_disease               2.353812675
```

```r
varImpPlot(rftree)
```



Test trees on hyperglycemic2 data set from different customer

```r
#Load 2nd set of data from different customer
drug2 = read.csv("hypoglycemic2.csv")
drug2$hypoglycemic <- as.factor(drug2$hypoglycemic)
drug2$asthma <- as.factor(drug2$asthma)
drug2$cad <- as.factor(drug2$cad)
drug2$chf <- as.factor(drug2$chf)
drug2$copd <- as.factor(drug2$copd)
drug2$cardio_respiratory_arrest <- as.factor(drug2$cardio_respiratory_arres)
drug2$cerebro_vascular <- as.factor(drug2$cerebro_vascular)
drug2$decubitus_ulcer <- as.factor(drug2$decubitus_ulcer)
drug2$delirium <- as.factor(drug2$delirium)
```

```r
drug2$mental_health <- as.factor(drug2$mental_health)
drug2$renal <- as.factor(drug2$renal)
drug2$substance_abuse <- as.factor(drug2$substance_abuse)
drug2$vascular_disease <- as.factor(drug2$vascular_disease)

summary(drug2$hypoglycemic)
```

```
##    0    1
## 1734  819
```

Dtree using Customer 2

```r
#Predict using Pruned dtree
dt2.pred2=predict(prune.dtree,drug2,type="class")
dt2.perf2 <- table(dt2.pred2,drug2$hypoglycemic)

#Results
print("Tree Performance on Customer 2 Data")
```

```
## [1] "Tree Performance on Customer 2 Data"
```

```r
dt2.perf2
```

```
##
## dt2.pred2    0    1
##         0 1588  514
##         1  146  305
```

```r
performance(dt2.perf2)
```

```
## Sensitivity (True Postive Rate)= 0.68
## Specificity (True Negative Rate) = 0.76
## False Negative Rate = 0.32
## Positives Predictive Value (odds of positive if postive prediction) = 0.37
## Negative Predictive value (odds of negative if negative prediction) = 0.92
## Accuracy = 0.74
```

RPart Using Customer 2

```r
#Predict Rpart using Customer 2 Data
rt2.pred2 <- predict(rtree, drug2, type="class")
rt2.perf2 <- table(rt2.pred2, drug2$hypoglycemic)

#Performance
print("Rpart Performance on Customer 2 Data")
```

```
## [1] "Rpart Performance on Customer 2 Data"
```

```
rt2.perf2
```

```
##
## rt2.pred2    0    1
##         0 1597  526
##         1  137  293
```

```
performance(rt2.perf2)
```

```
## Sensitivity (True Postive Rate)= 0.68
## Specificity (True Negative Rate) = 0.75
## False Negative Rate = 0.32
## Positives Predictive Value (odds of positive if postive prediction) = 0.36
## Negative Predictive value (odds of negative if negative prediction) = 0.92
## Accuracy = 0.74
```

RandomForest using Customer2

Pretty good results using medicare population for different customer.

```
#Random Forest on different set
rf2.pred2 <- predict(rftree, newdata=drug2, type="response")
rf2.perf2 <- table(rf2.pred2, drug2$hypoglycemic)
print("Random Forest on Customer 2 Data")
```

```
## [1] "Random Forest on Customer 2 Data"
```

```
rf2.perf2
```

```
##
## rf2.pred2    0    1
##         0 1646  546
##         1   88  273
```

```
performance(rf2.perf2)
```

```
## Sensitivity (True Postive Rate)= 0.76
## Specificity (True Negative Rate) = 0.75
## False Negative Rate = 0.24
## Positives Predictive Value (odds of positive if postive prediction) = 0.33
## Negative Predictive value (odds of negative if negative prediction) = 0.95
## Accuracy = 0.75
```

Boosting

```
#Try Boosting since there are a large number of variables
#reload to remove factors, use drug3 but same training data as previous models
library(gbm)
```

```
## Loaded gbm 2.1.8
```

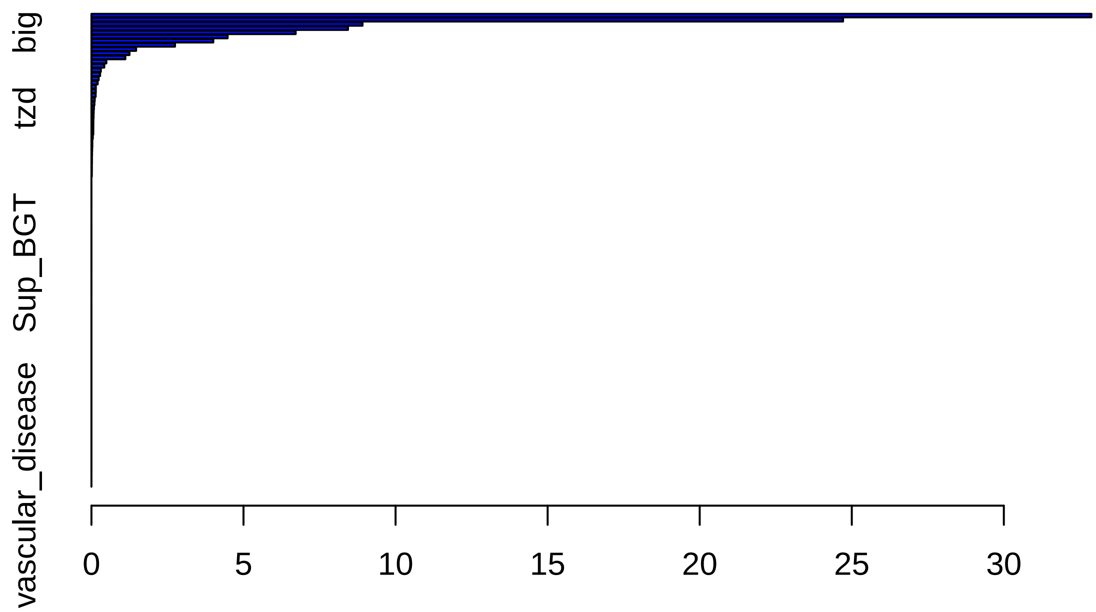
```r
drug3 = read.csv("hypoglycemic.csv")

#Set Training data to contain 70% of records
set.seed(123)
getSamp <- sample(nrow(drug3), .7*nrow(drug3),replace=F)
train3 <- drug3[getSamp,]
valid3 = drug3[-getSamp,]
```

```r
#undersampling (Excluded from final run)
#bal_train <- ovun.sample(hypoglycemic ~ .-mem_key, data = train, method = "under", N=800, seed = 1)$da
#summary(as.factor(bal_train$hypoglycemic))
```

```r
#Boosting
boost = gbm(hypoglycemic~. -mem_key, data=train3, distribution = "bernoulli", n.trees=1000
              , shrinkage=.001, interaction.depth = 3)

#Information
boost
```

```
## gbm(formula = hypoglycemic ~ . - mem_key, distribution = "bernoulli",
##     data = train3, n.trees = 1000, interaction.depth = 3, shrinkage = 0.001)
## A gradient boosted model with bernoulli loss function.
## 1000 iterations were performed.
## There were 113 predictors of which 39 had non-zero influence.
```

```r
summary(boost)
```




```
##                                           var     rel.inf
## op_visits                           op_visits 32.87755646
## a_ins_rapid                       a_ins_rapid 24.71332432
## a_ins_long                         a_ins_long  8.91239955
## raw_risk_score                 raw_risk_score  8.43574456
## big                                       big  6.71516669
## AnitInfect                         AnitInfect  4.47808051
## fp_internalmed                 fp_internalmed  4.00621232
## Sup_INS                               Sup_INS  2.74758365
## Gastroint                           Gastroint  1.46892044
## endocrinology                   endocrinology  1.25147634
## age                                       age  1.11516520
## age_gender_risk_score   age_gender_risk_score  0.48986409
## hypoagent                           hypoagent  0.42136800
## other                                   other  0.30920995
## cardiology                         cardiology  0.28467064
## Dermatol                             Dermatol  0.23812799
## eyecare                               eyecare  0.20491042
## Ophthalmic                         Ophthalmic  0.14506870
## nephrology                         nephrology  0.14379651
## Cardiovas                           Cardiovas  0.13889906
## sulf                                     sulf  0.10832107
## dpp4                                     dpp4  0.09840493
## tzd                                       tzd  0.08198914
## AntiInflam                         AntiInflam  0.07444106
## Locomotor                           Locomotor  0.07046610
```

```
## Hematolog                                         Hematolog  0.06437849
## Respiratory                                     Respiratory  0.06191838
## dpp4_big                                          dpp4_big  0.06102122
## Endocrine                                         Endocrine  0.06074463
## CentralNerv                                     CentralNerv  0.04601020
## Genitour                                           Genitour  0.02935176
## renal                                                 renal  0.02881643
## glp1                                                   glp1  0.02337151
## mental_health                                 mental_health  0.01973538
## cardio_respiratory_arrest cardio_respiratory_arrest  0.01697872
## orthopedics                                     orthopedics  0.01634990
## asthma                                               asthma  0.01399428
## dermatology                                     dermatology  0.01384474
## podiatry                                           podiatry  0.01231665
## grb                                                     grb  0.00000000
## dpp4_tzd                                           dpp4_tzd  0.00000000
## sglt2                                                 sglt2  0.00000000
## sglt2_big                                         sglt2_big  0.00000000
## h_ins_fixed                                     h_ins_fixed  0.00000000
## h_ins_interm                                   h_ins_interm  0.00000000
## h_ins_rapid                                     h_ins_rapid  0.00000000
## h_ins_short                                     h_ins_short  0.00000000
## a_ins_fixed                                     a_ins_fixed  0.00000000
## aat                                                     aat  0.00000000
## meg                                                     meg  0.00000000
## alphagi                                             alphagi  0.00000000
## tzd_sulf                                           tzd_sulf  0.00000000
## meg_big                                             meg_big  0.00000000
## sulf_big                                           sulf_big  0.00000000
## tzd_big                                             tzd_big  0.00000000
## dra                                                     dra  0.00000000
## slgt2_dpp4                                       slgt2_dpp4  0.00000000
## ins_glp1                                           ins_glp1  0.00000000
## slgt2_dpp_big                                 slgt2_dpp_big  0.00000000
## Sup_BGT                                             Sup_BGT  0.00000000
## Sup_BGKCTS                                       Sup_BGKCTS  0.00000000
## Sup_GMTS                                           Sup_GMTS  0.00000000
## Sup_SIDD                                           Sup_SIDD  0.00000000
## Sup_SIP                                             Sup_SIP  0.00000000
## Sup_UGT                                             Sup_UGT  0.00000000
## Sup_UGACT                                         Sup_UGACT  0.00000000
## Sup_UKT                                             Sup_UKT  0.00000000
## Sup_GMI                                             Sup_GMI  0.00000000
## Alt_ther                                           Alt_ther  0.00000000
## Anesthet                                           Anesthet  0.00000000
## Anorect                                             Anorect  0.00000000
## Antidotes                                         Antidotes  0.00000000
## Antineopl                                         Antineopl  0.00000000
## Antisept                                           Antisept  0.00000000
## Biologic                                           Biologic  0.00000000
## ChemDep                                             ChemDep  0.00000000
## ChemPharm                                         ChemPharm  0.00000000
## Cognitive                                         Cognitive  0.00000000
## Contracept                                       Contracept  0.00000000
```

```
## Diagnostic                                    Diagnostic  0.00000000
## ErectileDys                                  ErectileDys  0.00000000
## EatingDis                                      EatingDis  0.00000000
## Electrolyte                                  Electrolyte  0.00000000
## Enzymes                                          Enzymes  0.00000000
## GoutHyper                                      GoutHyper  0.00000000
## Hepatobil                                      Hepatobil  0.00000000
## Histamine                                      Histamine  0.00000000
## Immunosup                                      Immunosup  0.00000000
## OthrMedSup                                    OthrMedSup  0.00000000
## MetabDisEnzyme                            MetabDisEnzyme  0.00000000
## MetabModif                                    MetabModif  0.00000000
## MouthThrDen                                  MouthThrDen  0.00000000
## MultSclerosis                              MultSclerosis  0.00000000
## OrganPresSol                                OrganPresSol  0.00000000
## Otic                                                Otic  0.00000000
## RenalRepl                                      RenalRepl  0.00000000
## SepsisSynd                                    SepsisSynd  0.00000000
## Vaginal                                          Vaginal  0.00000000
## mentalhealth                                mentalhealth  0.00000000
## urology                                          urology  0.00000000
## vascularsurg                                vascularsurg  0.00000000
## rheumatology                                rheumatology  0.00000000
## osteopathic                                  osteopathic  0.00000000
## cad                                                  cad  0.00000000
## chf                                                  chf  0.00000000
## copd                                                copd  0.00000000
## cerebro_vascular                        cerebro_vascular  0.00000000
## decubitus_ulcer                          decubitus_ulcer  0.00000000
## delirium                                        delirium  0.00000000
## developmental_disability   developmental_disability  0.00000000
## pregnancy                                      pregnancy  0.00000000
## substance_abuse                          substance_abuse  0.00000000
## vascular_disease                        vascular_disease  0.00000000
```

```
#Performance of GBM Model
boost.pred = predict(boost, newdata=valid3, n.trees=1000)
boost.results = table(boost.pred >.5, (valid3$hypoglycemic))
print("GBM performance on Customer 1 Data")
```

```
## [1] "GBM performance on Customer 1 Data"
```

```
performance(boost.results)
```

```
## Sensitivity (True Postive Rate)= 0.5
## Specificity (True Negative Rate) = 0.81
## False Negative Rate = 0.5
## Positives Predictive Value (odds of positive if postive prediction) = 0.01
## Negative Predictive value (odds of negative if negative prediction) = 1
## Accuracy = 0.81
```

Evaluate boosted model with Customer2 data

```
drug4 = read.csv("hypoglycemic2.csv")  #reload to remove factors
drug4$hypoglycemic <- as.factor(drug4$hypoglycemic)
boost2.pred2 = predict(boost, newdata=drug4, n.trees=1000, type = "response")
boost2.results2 = table(boost2.pred2>.5, drug4$hypoglycemic)
print("GBM on Customer 2 Data")
```

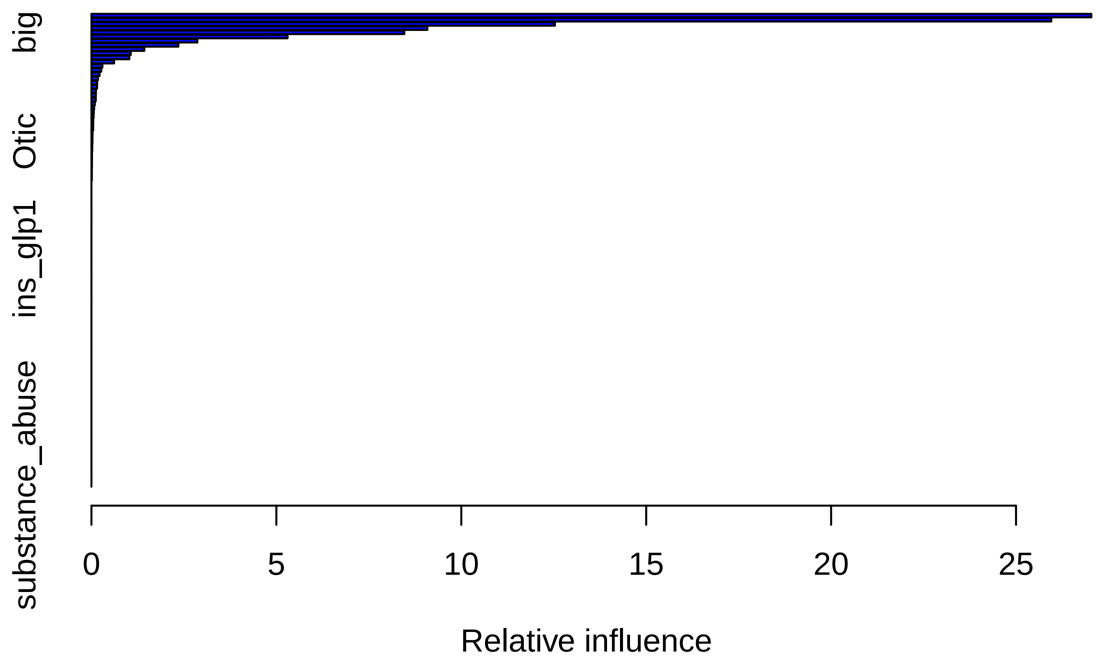
```
## [1] "GBM on Customer 2 Data"
```

```
performance(boost2.results2)
```

```
## Sensitivity (True Postive Rate)= 0.92
## Specificity (True Negative Rate) = 0.71
## False Negative Rate = 0.08
## Positives Predictive Value (odds of positive if postive prediction) = 0.16
## Negative Predictive value (odds of negative if negative prediction) = 0.99
## Accuracy = 0.73
```

Cross validation to reduce overfitting

```
#Estimate error rate with 10-fold cv
n=2121
k=10
groups = c(rep(1:k,floor(n/k)),1:(n-floor(n/k)*k))
set.seed(123)
cvgroups = sample(groups,n)
boostcv.predict = rep(-1,n)

for(i in 1:k){
  groupi = (cvgroups==i)
  boostcv = gbm(hypoglycemic~. -mem_key, data=drug3[!groupi,], distribution = "bernoulli", n.trees=1000
              , shrinkage=.001, interaction.depth = 3)
  boostcv.predict[groupi] = predict(boostcv, newdata=drug3[groupi,], n.trees=1000, type = "response")
}

summary(boostcv)
```




```
##                                               var     rel.inf
## op_visits                               op_visits 27.03655773
## a_ins_rapid                           a_ins_rapid 25.95368021
## a_ins_long                             a_ins_long 12.53214127
## fp_internalmed                     fp_internalmed  9.08318521
## big                                           big  8.45879426
## raw_risk_score                     raw_risk_score  5.30404489
## Sup_INS                                   Sup_INS  2.86464248
## endocrinology                       endocrinology  2.35099652
## AnitInfect                             AnitInfect  1.43010594
## hypoagent                               hypoagent  1.06320681
## cardiology                             cardiology  1.02424049
## Gastroint                               Gastroint  0.61253625
## Hematolog                               Hematolog  0.29640499
## age                                           age  0.26430117
## Cardiovas                               Cardiovas  0.22150098
## AntiInflam                             AntiInflam  0.17551184
## other                                       other  0.15551943
## Locomotor                               Locomotor  0.15290061
## Dermatol                                 Dermatol  0.12292691
## dpp4                                         dpp4  0.11948248
## eyecare                                   eyecare  0.11835798
## nephrology                             nephrology  0.09354165
## Endocrine                               Endocrine  0.07340719
## Ophthalmic                             Ophthalmic  0.06415351
## sulf                                         sulf  0.05999486
```

```
## dermatology                             dermatology 0.04870409
## age_gender_risk_score         age_gender_risk_score 0.04794337
## dpp4_big                                   dpp4_big 0.04678597
## chf                                             chf 0.03175095
## decubitus_ulcer                     decubitus_ulcer 0.02926055
## Otic                                           Otic 0.02838897
## vascular_disease                   vascular_disease 0.02203159
## sglt2                                         sglt2 0.02100363
## cerebro_vascular                   cerebro_vascular 0.01492090
## Electrolyte                             Electrolyte 0.01481989
## CentralNerv                             CentralNerv 0.01456275
## Genitour                                   Genitour 0.01456192
## Respiratory                             Respiratory 0.01132775
## glp1                                           glp1 0.01111164
## cad                                             cad 0.01069039
## grb                                             grb 0.00000000
## dpp4_tzd                                   dpp4_tzd 0.00000000
## sglt2_big                                 sglt2_big 0.00000000
## h_ins_fixed                             h_ins_fixed 0.00000000
## h_ins_interm                           h_ins_interm 0.00000000
## h_ins_rapid                             h_ins_rapid 0.00000000
## h_ins_short                             h_ins_short 0.00000000
## a_ins_fixed                             a_ins_fixed 0.00000000
## aat                                             aat 0.00000000
## meg                                             meg 0.00000000
## alphagi                                     alphagi 0.00000000
## tzd                                             tzd 0.00000000
## tzd_sulf                                   tzd_sulf 0.00000000
## meg_big                                     meg_big 0.00000000
## sulf_big                                   sulf_big 0.00000000
## tzd_big                                     tzd_big 0.00000000
## dra                                             dra 0.00000000
## slgt2_dpp4                               slgt2_dpp4 0.00000000
## ins_glp1                                   ins_glp1 0.00000000
## slgt2_dpp_big                         slgt2_dpp_big 0.00000000
## Sup_BGT                                     Sup_BGT 0.00000000
## Sup_BGKCTS                               Sup_BGKCTS 0.00000000
## Sup_GMTS                                   Sup_GMTS 0.00000000
## Sup_SIDD                                   Sup_SIDD 0.00000000
## Sup_SIP                                     Sup_SIP 0.00000000
## Sup_UGT                                     Sup_UGT 0.00000000
## Sup_UGACT                                 Sup_UGACT 0.00000000
## Sup_UKT                                     Sup_UKT 0.00000000
## Sup_GMI                                     Sup_GMI 0.00000000
## Alt_ther                                   Alt_ther 0.00000000
## Anesthet                                   Anesthet 0.00000000
## Anorect                                     Anorect 0.00000000
## Antidotes                                 Antidotes 0.00000000
## Antineopl                                 Antineopl 0.00000000
## Antisept                                   Antisept 0.00000000
## Biologic                                   Biologic 0.00000000
## ChemDep                                     ChemDep 0.00000000
## ChemPharm                                 ChemPharm 0.00000000
## Cognitive                                 Cognitive 0.00000000
```

```
## Contracept                                      Contracept  0.00000000
## Diagnostic                                      Diagnostic  0.00000000
## ErectileDys                                    ErectileDys  0.00000000
## EatingDis                                        EatingDis  0.00000000
## Enzymes                                            Enzymes  0.00000000
## GoutHyper                                        GoutHyper  0.00000000
## Hepatobil                                        Hepatobil  0.00000000
## Histamine                                        Histamine  0.00000000
## Immunosup                                        Immunosup  0.00000000
## OthrMedSup                                      OthrMedSup  0.00000000
## MetabDisEnzyme                              MetabDisEnzyme  0.00000000
## MetabModif                                      MetabModif  0.00000000
## MouthThrDen                                    MouthThrDen  0.00000000
## MultSclerosis                                MultSclerosis  0.00000000
## OrganPresSol                                  OrganPresSol  0.00000000
## RenalRepl                                        RenalRepl  0.00000000
## SepsisSynd                                      SepsisSynd  0.00000000
## Vaginal                                            Vaginal  0.00000000
## mentalhealth                                  mentalhealth  0.00000000
## urology                                            urology  0.00000000
## vascularsurg                                  vascularsurg  0.00000000
## rheumatology                                  rheumatology  0.00000000
## podiatry                                          podiatry  0.00000000
## osteopathic                                    osteopathic  0.00000000
## orthopedics                                    orthopedics  0.00000000
## asthma                                              asthma  0.00000000
## copd                                                  copd  0.00000000
## cardio_respiratory_arrest cardio_respiratory_arrest  0.00000000
## delirium                                          delirium  0.00000000
## developmental_disability   developmental_disability  0.00000000
## mental_health                                mental_health  0.00000000
## pregnancy                                        pregnancy  0.00000000
## renal                                                renal  0.00000000
## substance_abuse                            substance_abuse  0.00000000
```

Results of boosted prediction

```
boostcv.predict[1:10]
```

```
##  [1] 0.3631763 0.1491091 0.1921426 0.3622738 0.1466219 0.1257971 0.1270122
##  [8] 0.1382299 0.1442351 0.1410024
```

```
bcvresults = table(boostcv.predict>.5, drug3$hypoglycemic)
print("GBM with CV on Customer1 using CV")
```

```
## [1] "GBM with CV on Customer1 using CV"
```

```
performance(bcvresults)
```

```
## Sensitivity (True Postive Rate)= 0.85
```

```
## Specificity (True Negative Rate) = 0.83
## False Negative Rate = 0.15
## Positives Predictive Value (odds of positive if postive prediction) = 0.11
## Negative Predictive value (odds of negative if negative prediction) = 1
## Accuracy = 0.83
```

CV Boosted model on Customer 2 data

```r
bcv2.pred2 = predict(boostcv, newdata = drug4, n.trees = 1000, type = "response")
bcv2.perf2 = table(bcv2.pred2 >.5, drug4$hypoglycemic)
print("GBM with CV on Customer2 using CV")
```

```
## [1] "GBM with CV on Customer2 using CV"
```

```r
performance(bcv2.perf2)
```

```
## Sensitivity (True Postive Rate)= 0.93
## Specificity (True Negative Rate) = 0.71
## False Negative Rate = 0.07
## Positives Predictive Value (odds of positive if postive prediction) = 0.12
## Negative Predictive value (odds of negative if negative prediction) = 1
## Accuracy = 0.72
```