Diabetes Prediction model

DS II Final team

Contents

```
library(RNHANES)
library(tidyverse)
library(summarytools)
library(leaps)
library(readr)
library(caret)
library(ggplot2)
library(patchwork)
library(mgcv)
library(nlme)
library(dplyr)
library(plyr)
library(AppliedPredictiveModeling)
library(dplyr)
library(scales)
library(pROC)
#library(MASS)
#library(klaR)
library(forcats)
library(visdat)
library(glmnet)
library(mlbench)
library(pROC)
library(pdp)
library(vip)
library(rpart.plot)
library(ranger)
library(randomForest)
library(gbm)
library(e1071)
library(kernlab)
```

Load Data

```
data_files <- nhanes_load_data(file_name = "DIQ_H", year = "2013-2014")

data_files <- data_files %>%
  left_join(nhanes_load_data("HDL_H", "2013-2014"), by = "SEQN") %>%
  left_join(nhanes_load_data("INS_H", "2013-2014"), by = "SEQN") %>%
  left_join(nhanes_load_data("TRIGLY_H", "2013-2014"), by = "SEQN") %>%
  left_join(nhanes_load_data("DEMO_H", "2013-2014"), by = "SEQN") %>%
  left_join(nhanes_load_data("BMX_H", "2013-2014"), by = "SEQN") %>%
  left_join(nhanes_load_data("OGTT_H", "2013-2014"), by = "SEQN") %>%
  left_join(nhanes_load_data("BPX_H", "2013-2014"), by = "SEQN") %>%
  left_join(nhanes_load_data("PAQ_H", "2013-2014"), by = "SEQN") %>%
  left_join(nhanes_load_data("DPQ_H", "2013-2014"), by = "SEQN") %>%
  left_join(nhanes_load_data("SLQ_H", "2013-2014"), by = "SEQN")


raw_data <- data_files %>%
  select(SEQN, RIAGENDR, RIDAGEYR, RIDRETH3, BMXBMI, LBDHDD, LBDLDL, LBXTR, LBXIN, LBXGLT, BPXSY1, BPXD
```

```
raw_data <- raw_data[raw_data$DIQ010 != 3 & raw_data$DIQ010 != 7 & raw_data$DIQ010 != 9, ] %>%  mutate(
  drop_na(DIQ010)

 colnames(raw_data) <- c("ID", "gender", "age", "race", "bmi", "hdl", "ldl", "triglyceride", "insulin",

 contrasts(raw_data$diabetes)
```

```
##   2
## 1 0
## 2 1
```

```
 levels(raw_data$diabetes)[1] <- "yes"
 levels(raw_data$diabetes)[2] <- "no"
  contrasts(raw_data$diabetes)
```

```
##     no
## yes  0
## no   1
```

```
write.csv(raw_data, "final_data.csv")
```

EDA

Summary statistics

```
st_options(plain.ascii = FALSE,
           style = "rmarkdown",
           dfSummary.silent = TRUE,
           footnote = NA,
           subtitle.emphasis = FALSE)

dfSummary(raw_data[,-1], valid.col = FALSE)
```

```
## ### Data Frame Summary
## **raw_data**
## **Dimensions:** 9578 x 18
## **Duplicates:** 319
##
## ------------------------------------------------------------------------------------------------------
## No   Variable        Stats / Values               Freqs (% of Valid)     Graph                       I
## ---- --------------- ---------------------------- ---------------------- --------------------------- -
## 1    gender\         1\. 1\                       4706 (49.1%)\          IIIIIIIII \                 (
##      [factor]        2\. 2                        4872 (50.9%)           IIIIIIIII
##
## 2    age\            Mean (sd) : 32.4 (23.9)\     80 distinct values     \                           (
##      [numeric]       min < med < max:\                                   :\
##                      1 < 28 < 80\                                        : :\
##                      IQR (CV) : 41 (0.7)                                 : : .\
```

```
##                                                                                  : : : : : : : : : . .\
##                                                                                  : : : : : : : : : : :
##
## 3    race\           1\. 1\                    1616 (16.9%)\          III \                    0
##      [factor]        2\. 2\                    893 ( 9.3%)\           I \
##                      3\. 3\                    3449 (36.0%)\          IIIIIII \
##                      4\. 4\                    2148 (22.4%)\          IIII \
##                      5\. 6\                    1033 (10.8%)\          II \
##                      6\. 7                     439 ( 4.6%)
##
## 4    bmi\            Mean (sd) : 25.6 (7.9)\   436 distinct values    \                        1
##      [numeric]       min < med < max:\                                \ \ :\
##                      12.1 < 24.6 < 82.9\                              . : :\
##                      IQR (CV) : 10.4 (0.3)                            : : :\
##                                                                       : : : .\
##                                                                       : : : : .
##
## 5    hdl\            Mean (sd) : 53.2 (15.2)\  116 distinct values    \                        1
##      [numeric]       min < med < max:\                                \ \ \ \ :\
##                      10 < 51 < 173\                                   \ \ \ \ :\
##                      IQR (CV) : 19 (0.3)                              \ \ . : .\
##                                                                       \ \ : : :\
##                                                                       \ \ : : : .
##
## 6    ldl\            Mean (sd) : 106 (34.9)\   194 distinct values    \                        0
##      [numeric]       min < med < max:\                                \ \ \ \ :\
##                      14 < 103 < 375\                                  \ \ . :\
##                      IQR (CV) : 46 (0.3)                              \ \ : : .\
##                                                                       \ \ : : :\
##                                                                       . : : : .
##
## 7    triglyceride\   Mean (sd) : 111.7 (115.9)\ 344 distinct values    \                        0
##      [numeric]       min < med < max:\                                :\
##                      13 < 88 < 4233\                                  :\
##                      IQR (CV) : 73 (1)                                :\
##                                                                       :\
##                                                                       :
##
## 8    insulin\        Mean (sd) : 13.4 (18.7)\  1716 distinct values   \                        0
##      [numeric]       min < med < max:\                                :\
##                      0.1 < 9.3 < 682.5\                               :\
##                      IQR (CV) : 9.1 (1.4)                             :\
##                                                                       :\
##                                                                       :
##
## 9    glucose\        Mean (sd) : 114 (45.5)\   227 distinct values    \                        1
##      [numeric]       min < med < max:\                                \ \ :\
##                      40 < 104 < 604\                                  : :\
##                      IQR (CV) : 44 (0.4)                              : :\
##                                                                       : :\
##                                                                       : : :
##
## 10   bp_systolic\    Mean (sd) : 117.9 (18)\   71 distinct values     \                        1
##      [numeric]       min < med < max:\                                \ \ \ \ :\
```

```
##                     66 < 116 < 228\                                  \ \ \ \ : :\
##                     IQR (CV) : 20 (0.2)                              \ \ \ \ : :\
##                                                                      \ \ . : : .\
##                                                                      \ \ : : : : .
##
## 11   bp_diastolic\  Mean (sd) : 65.7 (15)\       59 distinct values   \                          1
##      [numeric]      min < med < max:\                                \ \ \ \ \ \ \ \ \ \ :\
##                     0 < 66 < 122\                                    \ \ \ \ \ \ \ \ \ \ : .\
##                     IQR (CV) : 16 (0.2)                              \ \ \ \ \ \ \ \ : : :\
##                                                                      \ \ \ \ \ \ \ \ : : :\
##                                                                      \ \ \ \ \ \ : : : : :
##
## 12   waist\         Mean (sd) : 86.9 (22.5)\     1030 distinct values \                         1
##      [numeric]      min < med < max:\                                \ \ \ \ \ \ : .\
##                     40.2 < 87.4 < 177.9\                             \ \ \ \ : : :\
##                     IQR (CV) : 31.6 (0.3)                            \ \ . : : :\
##                                                                      : : : : : :\
##                                                                      : : : : : : : .
##
## 13   lifestyle\     Mean (sd) : 478.5 (642.1)\   36 distinct values   \                          1
##      [numeric]      min < med < max:\                                :\
##                     0 < 480 < 9999\                                  :\
##                     IQR (CV) : 300 (1.3)                             :\
##                                                                      :\
##                                                                      :
##
## 14   education\     1\. 1\                       442 ( 7.9%)\        I \                        3
##      [factor]       2\. 2\                       761 (13.6%)\        II \
##                     3\. 3\                       1261 (22.6%)\       IIII \
##                     4\. 4\                       1715 (30.7%)\       IIIIII \
##                     5\. 5\                       1406 (25.1%)\       IIIII \
##                     6\. 7\                       2 ( 0.0%)\          \
##                     7\. 9                        5 ( 0.1%)
##
## 15   married\       1\. 1\                       2866 (51.3%)\       IIIIIIIIII \               3
##      [factor]       2\. 2\                       419 ( 7.5%)\        I \
##                     3\. 3\                       637 (11.4%)\        II \
##                     4\. 4\                       170 ( 3.0%)\        \
##                     5\. 5\                       1096 (19.6%)\       III \
##                     6\. 6\                       401 ( 7.2%)\        I \
##                     7\. 77\                      2 ( 0.0%)\          \
##                     8\. 99                       1 ( 0.0%)
##
## 16   depression\    Mean (sd) : 0.4 (0.8)\       0 : 3955 (75.5%)\   IIIIIIIIIIIIIII \          4
##      [numeric]      min < med < max:\            1 :  876 (16.7%)\   III \
##                     0 < 0 < 9\                   2 :  205 ( 3.9%)\   \
##                     IQR (CV) : 0 (2.1)           3 :  194 ( 3.7%)\   \
##                                                  7 :    2 ( 0.0%)\   \
##                                                  9 :    3 ( 0.1%)
##
## 17   sleep\         Mean (sd) : 7 (3.2)\         12 distinct values   \                          3
##      [numeric]      min < med < max:\                                :\
##                     2 < 7 < 99\                                      :\
##                     IQR (CV) : 2 (0.5)                               :\
```

```
##                                                                                  :\
##                                                                                  :
##
## 18   diabetes\       1\. yes\                  737 ( 7.7%)\          I \                      0
##      [factor]        2\. no                    8841 (92.3%)          IIIIIIIIIIIIIIIIII
## ----------------------------------------------------------------------------------------------------
```

```r
# Delete high missing-data covariates
raw_data <- raw_data[-c(7:10)]
dfSummary(raw_data[,-1], valid.col = FALSE)
```

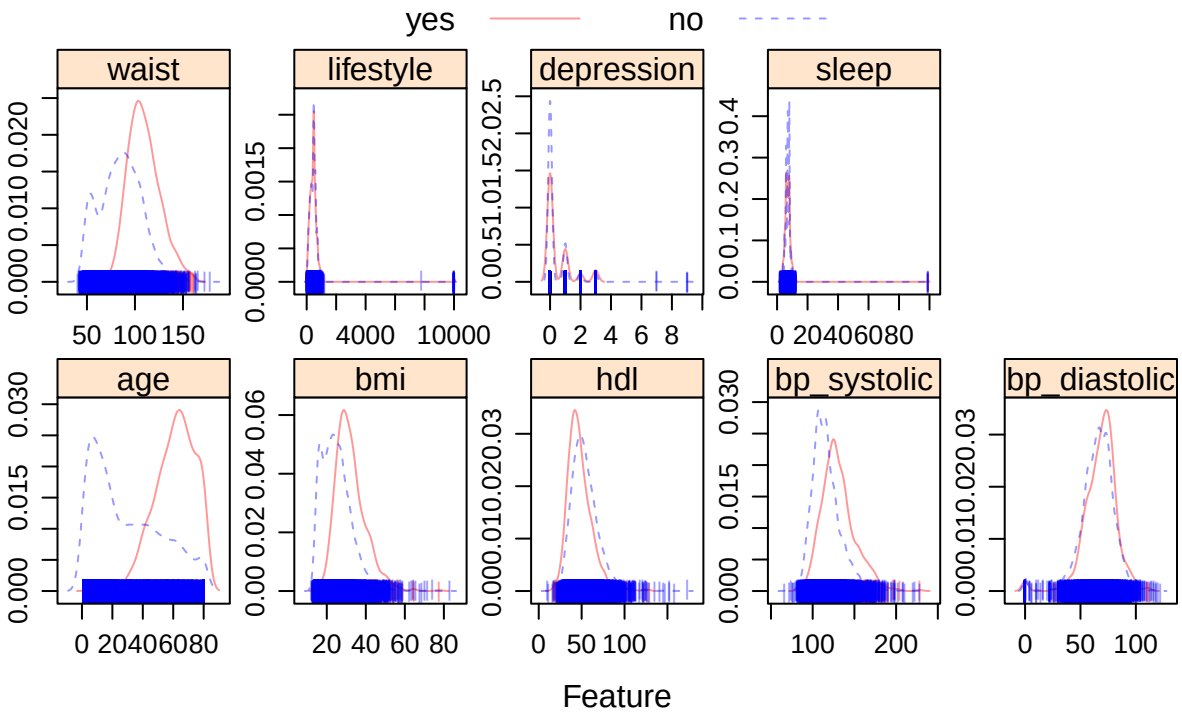
```
## ### Data Frame Summary
## **raw_data**
## **Dimensions:** 9578 x 14
## **Duplicates:** 319
##
## ----------------------------------------------------------------------------------------------------
## No   Variable        Stats / Values              Freqs (% of Valid)     Graph                     I
## ---- -------------- --------------------------- ---------------------- ------------------------- -
## 1    gender\         1\. 1\                      4706 (49.1%)\          IIIIIIIII \               0
##      [factor]        2\. 2                       4872 (50.9%)           IIIIIIIII
##
## 2    age\            Mean (sd) : 32.4 (23.9)\    80 distinct values     \                         0
##      [numeric]       min < med < max:\                                  :\
##                      1 < 28 < 80\                                       : :\
##                      IQR (CV) : 41 (0.7)                                : : .\
##                                                                         : : : : : : : : . .\
##                                                                         : : : : : : : : : : :
##
## 3    race\           1\. 1\                      1616 (16.9%)\          III \                     0
##      [factor]        2\. 2\                      893 ( 9.3%)\           I \
##                      3\. 3\                      3449 (36.0%)\          IIIIIII \
##                      4\. 4\                      2148 (22.4%)\          IIII \
##                      5\. 6\                      1033 (10.8%)\          II \
##                      6\. 7                       439 ( 4.6%)
##
## 4    bmi\            Mean (sd) : 25.6 (7.9)\     436 distinct values    \                         1
##      [numeric]       min < med < max:\                                  \ \ :\
##                      12.1 < 24.6 < 82.9\                                . : :\
##                      IQR (CV) : 10.4 (0.3)                              : : :\
##                                                                         : : : .\
##                                                                         : : : : .
##
## 5    hdl\            Mean (sd) : 53.2 (15.2)\    116 distinct values    \                         1
##      [numeric]       min < med < max:\                                  \ \ \ \ :\
##                      10 < 51 < 173\                                     \ \ \ \ :\
##                      IQR (CV) : 19 (0.3)                                \ \ . : .\
##                                                                         \ \ : : :\
##                                                                         \ \ : : : .
##
## 6    bp_systolic\    Mean (sd) : 117.9 (18)\     71 distinct values     \                         1
##      [numeric]       min < med < max:\                                  \ \ \ \ :\
##                      66 < 116 < 228\                                    \ \ \ \ : :\
##                      IQR (CV) : 20 (0.2)                                \ \ \ \ : :\
```

```
##                                                                    \ \ . : : .\
##                                                                    \ \ : : : : .
##
## 7    bp_diastolic\   Mean (sd) : 65.7 (15)\      59 distinct values    \                        1
##      [numeric]       min < med < max:\                                 \ \ \ \ \ \ \ \ \ \ :\
##                      0 < 66 < 122\                                     \ \ \ \ \ \ \ \ \ \ : .\
##                      IQR (CV) : 16 (0.2)                               \ \ \ \ \ \ \ \ : : :\
##                                                                        \ \ \ \ \ \ \ \ : : :\
##                                                                        \ \ \ \ \ \ : : : : :
##
## 8    waist\          Mean (sd) : 86.9 (22.5)\    1030 distinct values  \                        1
##      [numeric]       min < med < max:\                                 \ \ \ \ \ \ : .\
##                      40.2 < 87.4 < 177.9\                              \ \ \ \ : : :\
##                      IQR (CV) : 31.6 (0.3)                             \ \ . : : :\
##                                                                        : : : : : :\
##                                                                        : : : : : : : : .
##
## 9    lifestyle\      Mean (sd) : 478.5 (642.1)\  36 distinct values    \                        1
##      [numeric]       min < med < max:\                                 :\
##                      0 < 480 < 9999\                                   :\
##                      IQR (CV) : 300 (1.3)                              :\
##                                                                        :\
##                                                                        :
##
## 10   education\      1\. 1\                      442 ( 7.9%)\          I \                      3
##      [factor]        2\. 2\                      761 (13.6%)\          II \
##                      3\. 3\                      1261 (22.6%)\         IIII \
##                      4\. 4\                      1715 (30.7%)\         IIIIII \
##                      5\. 5\                      1406 (25.1%)\         IIIII \
##                      6\. 7\                      2 ( 0.0%)\            \
##                      7\. 9                       5 ( 0.1%)
##
## 11   married\        1\. 1\                      2866 (51.3%)\         IIIIIIIII \              3
##      [factor]        2\. 2\                      419 ( 7.5%)\          I \
##                      3\. 3\                      637 (11.4%)\          II \
##                      4\. 4\                      170 ( 3.0%)\          \
##                      5\. 5\                      1096 (19.6%)\         III \
##                      6\. 6\                      401 ( 7.2%)\          I \
##                      7\. 77\                     2 ( 0.0%)\            \
##                      8\. 99                      1 ( 0.0%)
##
## 12   depression\     Mean (sd) : 0.4 (0.8)\      0 : 3955 (75.5%)\     IIIIIIIIIIIIII \         4
##      [numeric]       min < med < max:\           1 :  876 (16.7%)\     III \
##                      0 < 0 < 9\                  2 :  205 ( 3.9%)\     \
##                      IQR (CV) : 0 (2.1)          3 :  194 ( 3.7%)\     \
##                                                  7 :    2 ( 0.0%)\     \
##                                                  9 :    3 ( 0.1%)
##
## 13   sleep\          Mean (sd) : 7 (3.2)\        12 distinct values    \                        3
##      [numeric]       min < med < max:\                                 :\
##                      2 < 7 < 99\                                       :\
##                      IQR (CV) : 2 (0.5)                                :\
##                                                                        :\
##                                                                        :
```

```
##
## 14   diabetes\       1\. yes\                    737 ( 7.7%)\          I \                    (
##      [factor]        2\. no                      8841 (92.3%)          IIIIIIIIIIIIIIIIII
## ------------------------------------------------------------------------------------------------
```

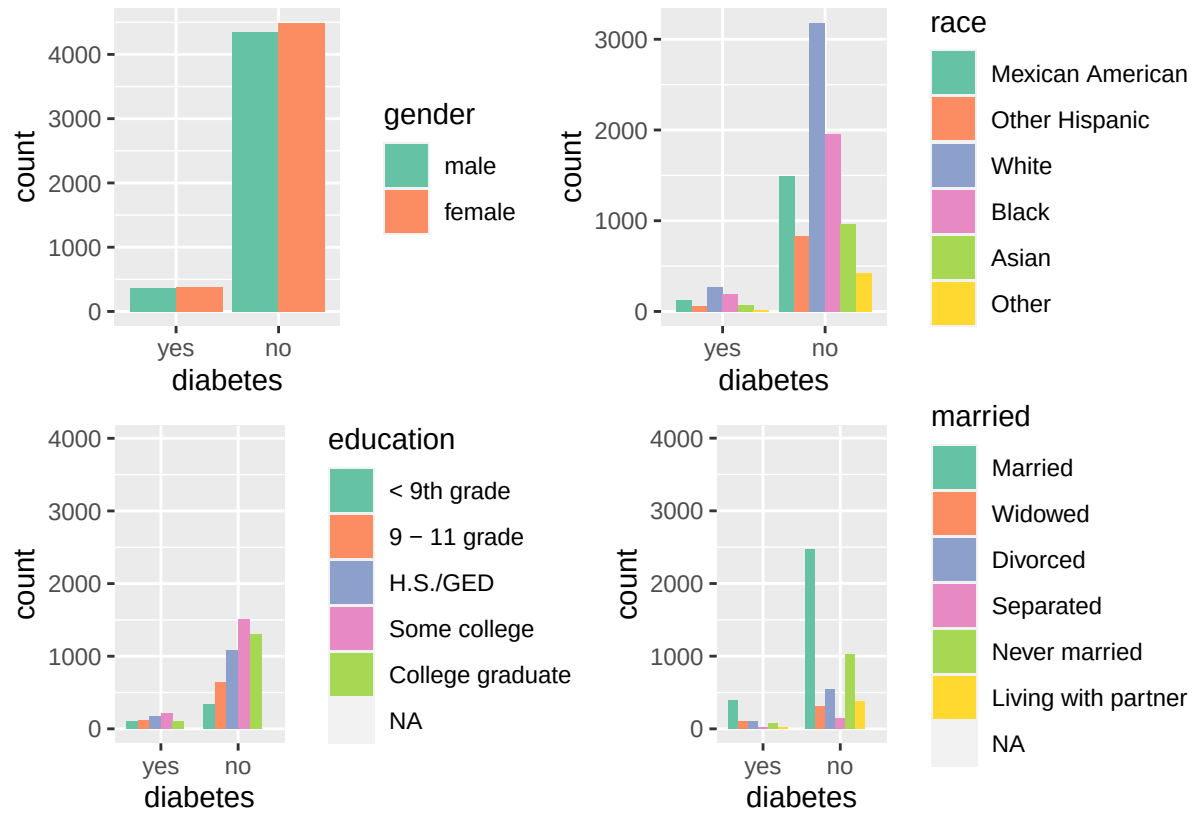
Density plots (numerical covariates)

```
theme1 <- transparentTheme(trans = .4)
trellis.par.set(theme1)

raw_data <- raw_data %>%
  select(married, everything()) %>%
  select(education, everything()) %>%
  select(race, everything()) %>%
  select(gender, everything()) %>%
  select(ID, everything())

featurePlot(x = raw_data[, 6:14],
            y = raw_data$diabetes,
            scales = list(x = list(relation = "free"),
                          y = list(relation = "free")),
            plot = "density", pch = "|",
            auto.key = list(columns = 2))
```



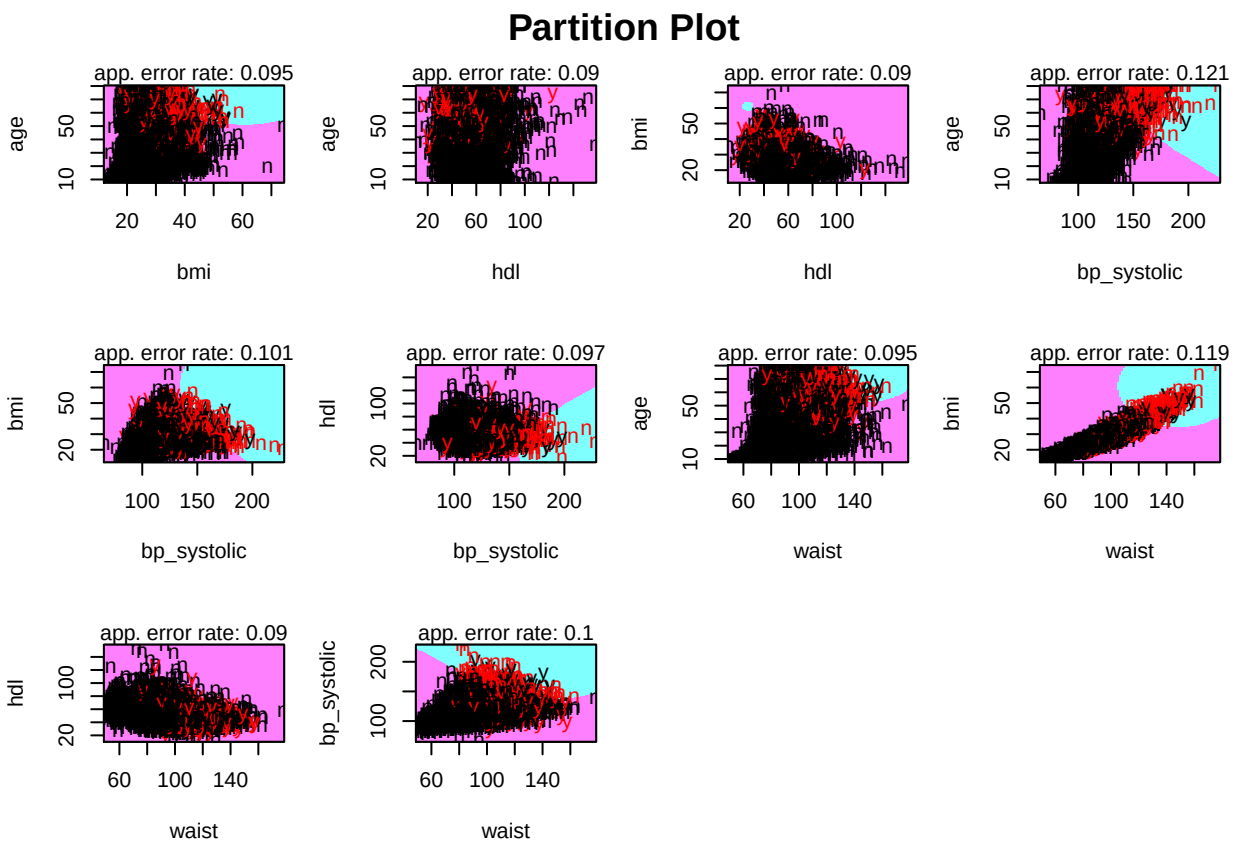
Bar plots (categorical covariates)

```
diabetes_gender = ggplot(raw_data,
       aes(x = diabetes,
           fill = factor(gender,
                         levels = c("1", "2"),
                         labels = c("male", "female")))) +
  geom_bar(position = position_dodge(preserve = "single")) +
   scale_fill_brewer(palette = "Set2") +
  labs(fill = "gender")

diabetes_race = ggplot(raw_data,
       aes(x = diabetes,
           fill = factor(race,
                         levels = c("1", "2", "3", "4", "6", "7"),
                         labels = c("Mexican American", "Other Hispanic", "White", "Black", "Asian", "O
  geom_bar(position = position_dodge(preserve = "single")) +
   scale_fill_brewer(palette = "Set2") +
   labs(fill = "race")

diabetes_education = ggplot(raw_data,
       aes(x = diabetes,
           fill = factor(education,
                         levels = c("1", "2", "3", "4", "5"),
                         labels = c("< 9th grade", "9 - 11 grade", "H.S./GED", "Some college", "College
  geom_bar(position = position_dodge(preserve = "single")) +
   scale_fill_brewer(palette = "Set2") +
   labs(fill = "education")

diabetes_married = ggplot(raw_data,
       aes(x = diabetes,
           fill = factor(married,
                         levels = c("1", "2", "3", "4", "5", "6"),
                         labels = c("Married", "Widowed", "Divorced", "Separated", "Never married", "Li
  geom_bar(position = position_dodge(preserve = "single")) +
   scale_fill_brewer(palette = "Set2") +
    labs(fill = "married")

(diabetes_gender + diabetes_race)  / (diabetes_education + diabetes_married)
```



Partition-plots

```
set.seed(1)
rowTrain <- createDataPartition(y = raw_data$diabetes,
                                p = 0.7,
                                list = FALSE)

# Exploratory analysis: LDA/QDA/NB based on every combination of two variables

# klaR::partimat(diabetes ~ age + bmi +  hdl + bp_systolic + waist,
#        data = raw_data, subset = rowTrain, method = "lda")

# klaR::partimat(diabetes ~ age + bmi + hdl  + bp_systolic + waist,
#         data = raw_data, subset = rowTrain, method = "qda")

klaR::partimat(diabetes ~ age + bmi + hdl  + bp_systolic + waist,
          data = raw_data, subset = rowTrain, method = "naiveBayes")
```



Models

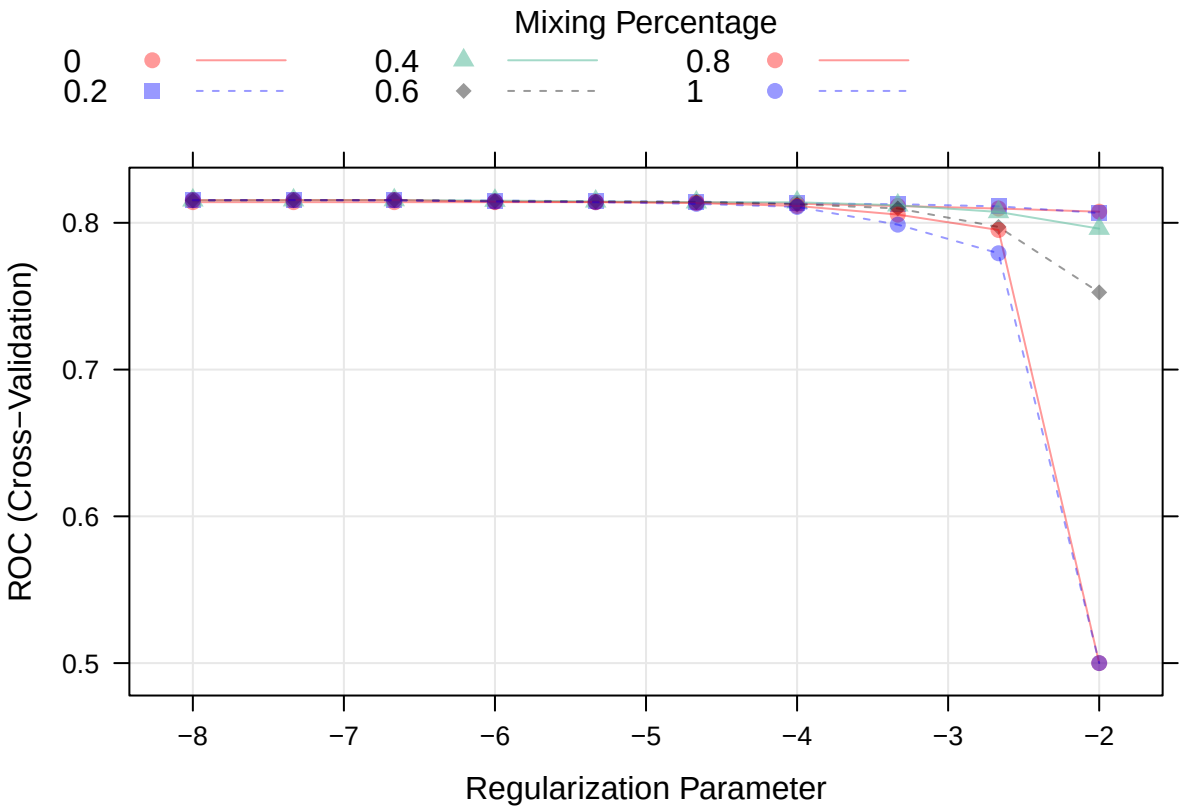
Prep/partition data

```
# Omit Missing data
diabetes_data <- na.omit(raw_data)

# Omit low-count subcategories
diabetes_data <- na.omit(diabetes_data) %>%
  filter(married != "77") %>%
  filter(education != "7") %>%
  filter(education != "9") %>%
  droplevels()

set.seed(1)
trainRows <- createDataPartition(diabetes_data$diabetes, p = 0.8, list = FALSE)


# training data
x <- diabetes_data[trainRows ,-c(1, 15)]
y <- diabetes_data$diabetes[trainRows]

# test data
```

```
x2 <- diabetes_data[-trainRows ,-c(1, 15)]
y2 <- diabetes_data$diabetes[-trainRows]

# Setup CV method
ctrl <- trainControl(method = "cv",
                     summaryFunction = twoClassSummary,
                     classProbs = TRUE)
```

Linear models

```
# glm
set.seed(1)

model.glm <- train(x = x,
                   y = y,
                   method = "glm",
                   metric = "ROC",
                   trControl = ctrl)

# glm.pred <- predict(model.glm, newdata = x2, type = "prob")[,2]
# roc.glm <- roc(y2, glm.pred)
# plot(roc.glm, legacy.axes = TRUE, print.auc = TRUE)
# plot(smooth(roc.glm), col = 4, add = TRUE)

# Penalized Logistic regression
glmnGrid <- expand.grid(.alpha = seq(0, 1, length = 6),
                        .lambda = exp(seq(-8, -2, length = 10)))
set.seed(1)

model.glmn <- train(x = data.matrix(x),
                    y = y,
                    method = "glmnet",
                    tuneGrid = glmnGrid,
                    metric = "ROC",
                    trControl = ctrl)

plot(model.glmn, xTrans = function(x)log(x))
```



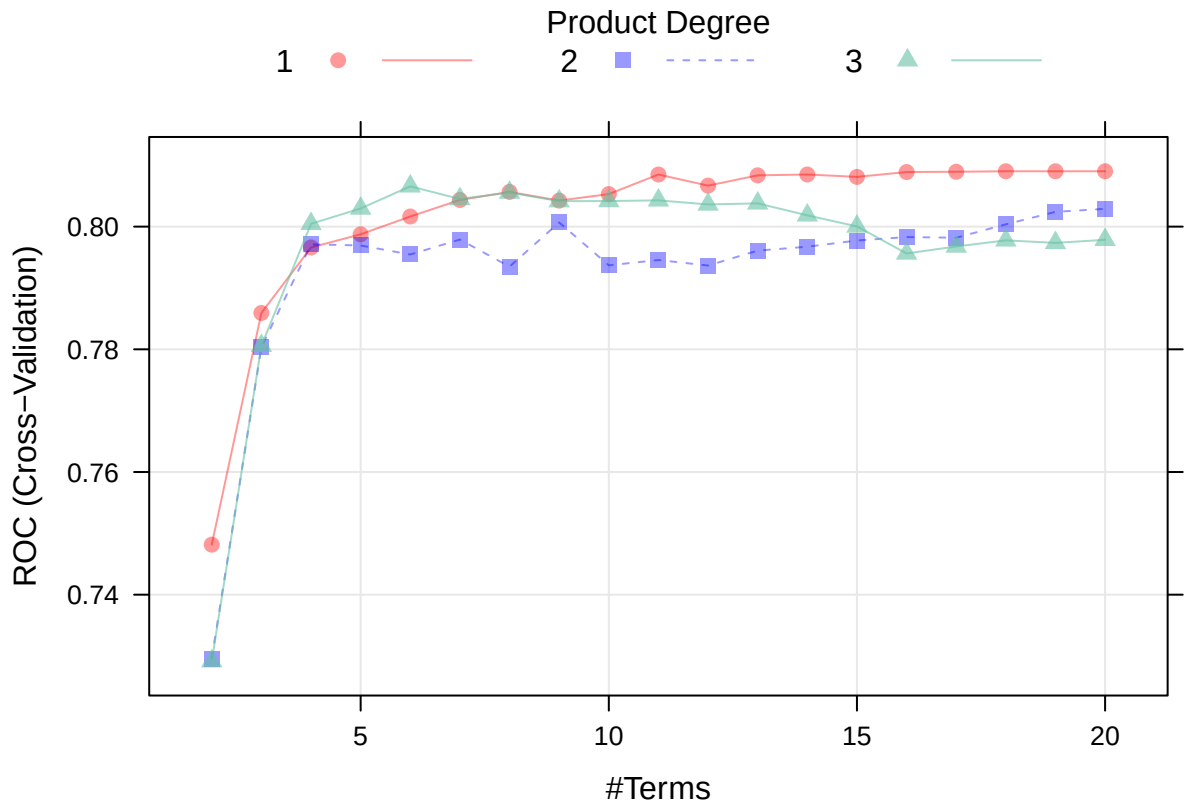

```
model.glmn$bestTune
```

```
##    alpha      lambda
## 33   0.6 0.001272634
```

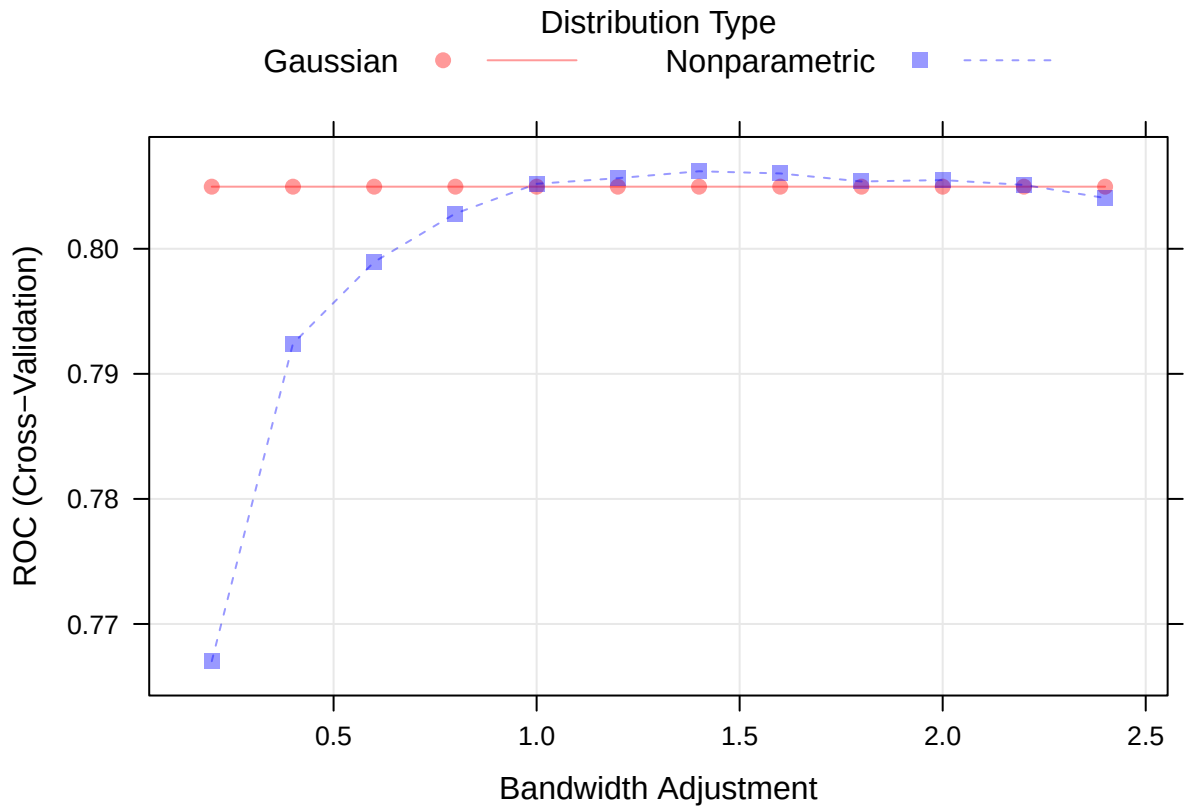
```
# glmn.pred <- predict(model.glmn, newdata = data.matrix(x2), type = "prob")[,2]
# roc.glmn <- roc(y2, glmn.pred)
# plot(roc.glmn, legacy.axes = TRUE, print.auc = TRUE)
# plot(smooth(roc.glmn), col = 4, add = TRUE)

# LDA
# set.seed(1)

# model.lda <- train(x = data.matrix(x),
#                    y = y,
#                    method = "lda",
#                    metric = "ROC",
#                    trControl = ctrl)

# lda.pred <- predict(model.lda, newdata = data.matrix(x2), type = "prob") [,2]

# roc.lda <- roc(y2, lda.pred)
# plot(roc.lda, legacy.axes = TRUE, print.auc = TRUE)
# plot(smooth(roc.lda), col = 4, add = TRUE)
```

Nonlinear models

```
## Non-linear Logistic regression: GAM, MARS
# GAM
#set.seed(1)
#model.gam <- train(x = x,
#                   y = y,
#                   method = "gam",
#                   metric = "ROC",
#                   trControl = ctrl)

#model.gam$finalModel

# MARS
set.seed(1)

model.mars <- train(x = x,
                    y = y,
                    method = "earth",
                    tuneGrid = expand.grid(degree = 1:3,
                                           nprune = 2:20),
                    metric = "ROC",
                    trControl = ctrl)

plot(model.mars)
```



```r
#coef(model.mars$finalModel)

## Non-linear Discriminant analysis: QDA, Naive Bayes (NB)
# QDA = for continuous features
#set.seed(1)
#model.qda <- train(x = x,
#                   y = y,
#                   method = "qda",
#                   metric = "ROC",
#                   trControl = ctrl)

# NB
set.seed(1)

nbGrid <- expand.grid(usekernel = c(FALSE,TRUE),
                      fL = 1,
                      adjust = seq(.2, 2.5, by = .2))

model.nb <- train(x = x,
                  y = y,
                  method = "nb",
                  tuneGrid = nbGrid,
                  metric = "ROC",
                  trControl = ctrl)

plot(model.nb)
```

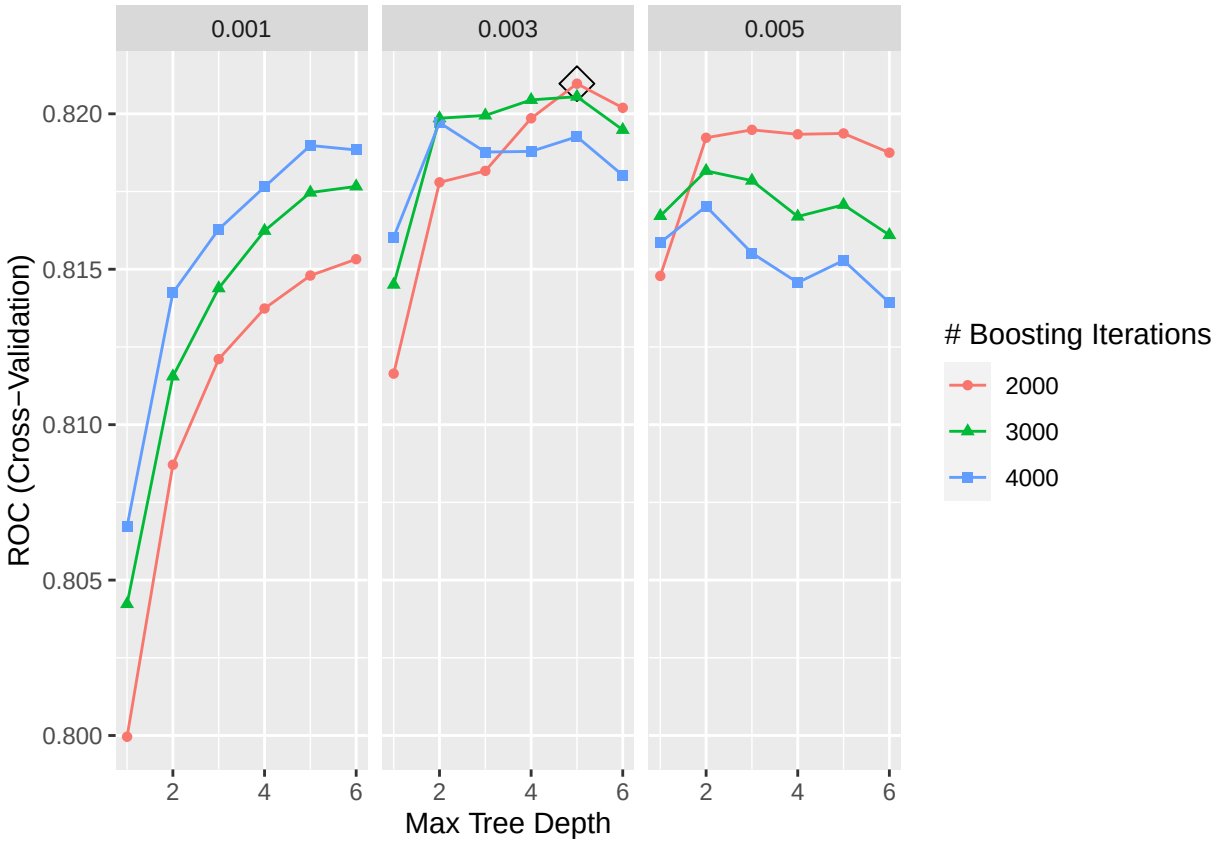


trees/ SVM

```
## single tree. very useless
#set.seed(1)
#rpart.fit <- train(diabetes ~ . ,
#                   diabetes_data[trainRows,-1],
#                   method = "rpart",
#                   tuneGrid = data.frame(cp = exp(seq(-1,10, length = 500))),
#                   trControl = ctrl)
#ggplot(rpart.fit, highlight = TRUE)
#rpart.plot(rpart.fit$finalModel)

## random forest in caret

#rf_grid = expand.grid(mtry = 1:13,
#                      splitrule = "gini",
#                      min.node.size = seq(from = 2, to = 10, by = 2))

#set.seed(1)
#rf.fit = train(diabetes ~ . ,
#               diabetes_data[trainRows,-1],
#               method = "ranger",
#               tuneGrid = rf_grid,
#               metric = "ROC",
```

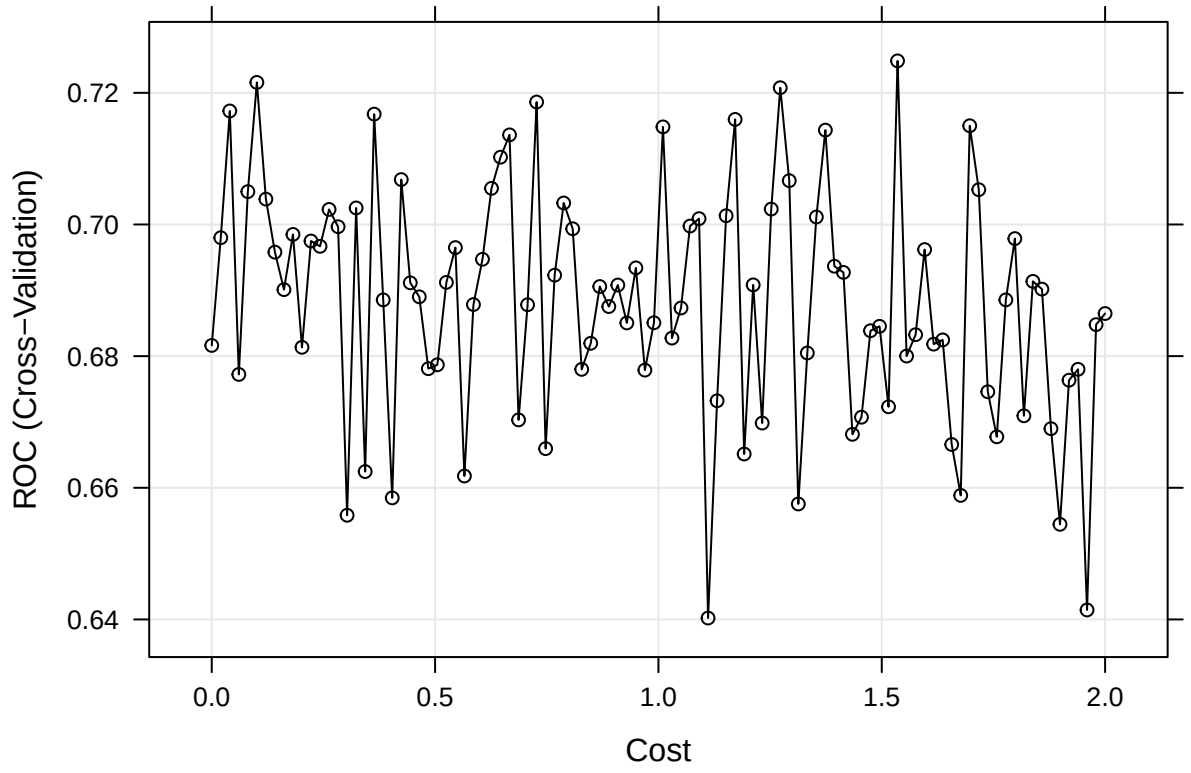
```
#               trControl = ctrl)

#ggplot(rf.fit, highlight = TRUE)

#set.seed(1)
#rf_final = ranger(diabetes ~ . ,
#                  diabetes_data[trainRows,-1],
#                  mtry = rf_fit$bestTune[[1]],
#                  min.node.size = rf_fit$bestTune[[3]],
#                  importance = "permutation",
#                  scale.permutation.importance = TRUE)
#rf_table=rf_final$variable.importance
#rf_final$prediction.error
#rfclass_pred = predict(rf_final, data = diabetes_data[-trainRows,-1], type = "response")$predictions
#rfconf = confusionMatrix(data = as.factor(rfclass_pred),
#                reference = y2,
#                 positive = "yes")

#rf_err = (rfconf$table[1,2]+rfconf$table[2,1])/(rfconf$table[1,1]+rfconf$table[1,2]+rfconf$table[2,1]+

### gbm/gbma
#gbm_grid = expand.grid(n.trees = c(0,1000,2000,3000,4000,5000,6000),
#                       interaction.depth = 1:4,
#                       shrinkage = c(0.001,0.003,0.005),
#                       n.minobsinnode = c(1,10))

#set.seed(1)
#gbm_fit = train(diabetes ~ . ,
#                diabetes_data[trainRows,-1],
#                method = "gbm",
#                tuneGrid = gbm_grid,
#                trControl = ctrl,
#                verbose = FALSE)
#ggplot(gbm_fit, highlight = TRUE)
#summary(gbm_fit$finalModel)
#gbm_pred <- predict(gbm_fit, newdata = diabetes_data[-trainRows,], type = "prob")[,1]
#gbm_test_pred = rep("no", length(gbm_pred))
#gbm_test_pred[gbm_pred>0.5] = "yes"
#gbmconf = confusionMatrix(data = as.factor(gbm_test_pred),
#                reference = diabetes_data$diabetes[-trainRows],
#                positive = "yes")
#gbmconf$table
#gbm_err = (gbmconf$table[1,2]+gbmconf$table[2,1])/(gbmconf$table[1,1]+gbmconf$table[1,2]+gbmconf$table

gbmA_grid <- expand.grid(n.trees = c(2000,3000,4000),
                         interaction.depth = 1:6,
                         shrinkage = c(0.001,0.003,0.005),
                         n.minobsinnode = 1)

set.seed(1)
gbmA.fit <- train(diabetes ~ . ,
                  diabetes_data,
                  subset = trainRows,
```

```
                    tuneGrid = gbmA_grid,
                    trControl = ctrl,
                    method = "gbm",
                    distribution = "adaboost",
                    metric = "ROC",
                    verbose = FALSE)
ggplot(gbmA.fit, highlight = TRUE)
```




```
#gbmA_pred <- predict(gbmA_fit, newdata = diabetes_data[-trainRows,], type = "prob")[,1]
#gbmA_test_pred = rep("no", length(gbmA_pred))
#gbmA_test_pred[gbmA_pred>0.5] = "yes"
#gbmAconf = confusionMatrix(data = as.factor(gbmA_test_pred),
#                 reference = diabetes_data$diabetes[-trainRows],
#                 positive = "yes")
#gbmAconf$table
#gbmA_err = (gbmAconf$table[1,2]+gbmAconf$table[2,1])/(gbmAconf$table[1,1]+gbmAconf$table[1,2]+gbmAconf

# Comparing Ensemble methods
#res <- resamples(list(rf = rf.fit,
#                      gbm = gbm_fit,
#                      gbmA = gbmA.fit ))

#summary(res)
#bwplot(res, metric = "ROC")
```

```
## SVML/R
# e1071
set.seed(1)
svml.fit <- train(diabetes ~ . ,
                  data = diabetes_data[trainRows,],
                  method = "svmLinear2",
                  tuneGrid = data.frame(cost = exp(seq(0,2,len = 100))),
                  metric = "ROC",
                  trControl = ctrl)
plot(svml.fit, highlight = TRUE, xTrans = log)
```




```
#pred.svml <- predict(svml.fit, newdata = diabetes_data[-trainRows,])
#confusionMatrix(data = pred.svml,
#                reference = diabetes_data$diabetes[-trainRows])

## radial
#svmr.grid <- expand.grid(C = exp(seq(-1,3,len = 10)),
#                         sigma = exp(seq(-4,0,len = 10)))

# tunes over both cost and sigma
#set.seed(1)
#svmr.fit <- train(diabetes ~ . ,
#                  diabetes_data,
#                  subset = trainRows,
#                  method = "svmRadialSigma",
```

```
#                   preProcess = c("center", "scale"),
#                   tuneGrid = svmr.grid,
#                   trControl = ctrl)
#plot(svmr.fit, highlight = TRUE)
#pred.svmr <- predict(svmr.fit, newdata = diabetes_data[-trainRows,])
#confusionMatrix(data = pred.svmr,
#                reference = diabetes_data$diabetes[-trainRows])

# Comparing svm methods
#res <- resamples(list(svml = svml.fit,
#                      svmr = svmr.fit))

#summary(res)
#bwplot(res, metric = "ROC")
```

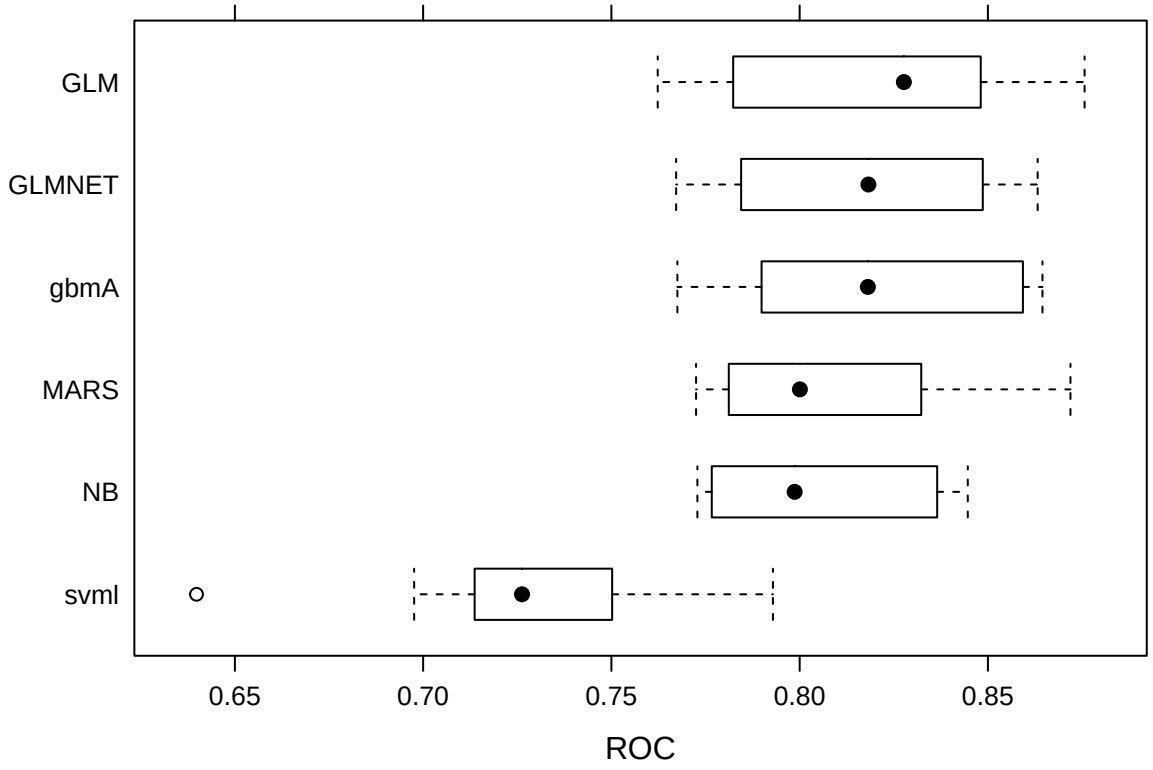
Model comparison

```
res <- resamples(list(GLM = model.glm,
                      GLMNET = model.glmn,
                      MARS = model.mars,
                      NB = model.nb,
                      gbmA = gbmA.fit,
                      svml = svml.fit))

summary(res)
```

```
##
## Call:
## summary.resamples(object = res)
##
## Models: GLM, GLMNET, MARS, NB, gbmA, svml
## Number of resamples: 10
##
## ROC
##             Min.   1st Qu.    Median      Mean   3rd Qu.      Max. NA's
## GLM    0.7623056 0.7861968 0.8276816 0.8196973 0.8465606 0.8756791    0
## GLMNET 0.7671958 0.7860153 0.8182620 0.8155202 0.8457133 0.8632151    0
## MARS   0.7724868 0.7818502 0.8000641 0.8090250 0.8308415 0.8719239    0
## NB     0.7728447 0.7796216 0.7986685 0.8061999 0.8354096 0.8446480    0
## gbmA   0.7675199 0.7941118 0.8181100 0.8209701 0.8562155 0.8644934    0
## svml   0.6398108 0.7139657 0.7262604 0.7248513 0.7459954 0.7929051    0
##
## Sens
##              Min.    1st Qu.     Median       Mean   3rd Qu.      Max. NA's
## GLM    0.07142857 0.11904762 0.12929125 0.14147287 0.1607143 0.2325581    0
## GLMNET 0.04761905 0.07142857 0.10575858 0.09905869 0.1183555 0.1627907    0
## MARS   0.04761905 0.11904762 0.14285714 0.13676633 0.1578073 0.2093023    0
## NB     0.16666667 0.22245293 0.30952381 0.28117386 0.3313953 0.3720930    0
## gbmA   0.04761905 0.07142857 0.09413068 0.08732004 0.1110188 0.1190476    0
## svml   0.00000000 0.00000000 0.00000000 0.00000000 0.0000000 0.0000000    0
##
```

```
## Spec
##              Min.   1st Qu.    Median      Mean   3rd Qu.      Max. NA's
## GLM    0.9730640 0.9772727 0.9815154 0.9831842 0.9898990 0.9966330    0
## GLMNET 0.9697987 0.9806906 0.9898990 0.9858778 0.9898990 0.9932886    0
## MARS   0.9697987 0.9747475 0.9815154 0.9825142 0.9882296 1.0000000    0
## NB     0.9259259 0.9351852 0.9478114 0.9471855 0.9587542 0.9664430    0
## gbmA   0.9764310 0.9898990 0.9932660 0.9902459 0.9932829 0.9966443    0
## svml   1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000    0
```

```r
bwplot(res, metric = "ROC")
```



Final model

```r
# glm
set.seed(1)

model.glm <- train(x = x,
                   y = y,
                   method = "glm",
                   metric = "ROC",
                   trControl = ctrl)

summary(model.glm)
```

```
##
## Call:
## NULL
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.1772   0.1424   0.2918   0.5104   2.1264
##
## Coefficients:
##                Estimate Std. Error z value Pr(>|z|)
## (Intercept)   7.926e+00  7.603e-01  10.424  < 2e-16 ***
## gender2      -3.396e-01  1.398e-01  -2.429 0.015130 *
## race2        -1.555e-01  2.396e-01  -0.649 0.516436
## race3         4.706e-01  1.927e-01   2.442 0.014606 *
## race4        -2.126e-01  2.071e-01  -1.026 0.304758
## race6        -4.240e-01  2.655e-01  -1.597 0.110279
## race7         1.025e-01  4.537e-01   0.226 0.821237
## education2    1.569e-01  2.263e-01   0.693 0.488051
## education3    5.478e-01  2.189e-01   2.502 0.012340 *
## education4    4.838e-01  2.148e-01   2.252 0.024335 *
## education5    8.664e-01  2.343e-01   3.697 0.000218 ***
## married2      5.346e-02  1.989e-01   0.269 0.788132
## married3      1.703e-01  1.800e-01   0.947 0.343893
## married4     -1.123e-01  3.191e-01  -0.352 0.724982
## married5      3.894e-01  2.096e-01   1.858 0.063144 .
## married6      8.015e-01  3.735e-01   2.146 0.031886 *
## age          -4.899e-02  4.998e-03  -9.803  < 2e-16 ***
## bmi           6.495e-02  2.242e-02   2.896 0.003774 **
## hdl           2.708e-02  4.772e-03   5.675 1.39e-08 ***
## bp_systolic  -6.210e-03  3.465e-03  -1.792 0.073077 .
## bp_diastolic  1.140e-02  4.399e-03   2.592 0.009551 **
## waist        -6.666e-02  9.927e-03  -6.714 1.89e-11 ***
## lifestyle     6.979e-05  8.360e-05   0.835 0.403802
## depression   -1.151e-01  6.910e-02  -1.666 0.095755 .
## sleep        -3.465e-02  1.668e-02  -2.078 0.037736 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2553.2  on 3395  degrees of freedom
## Residual deviance: 1993.5  on 3371  degrees of freedom
## AIC: 2043.5
##
## Number of Fisher Scoring iterations: 6
```

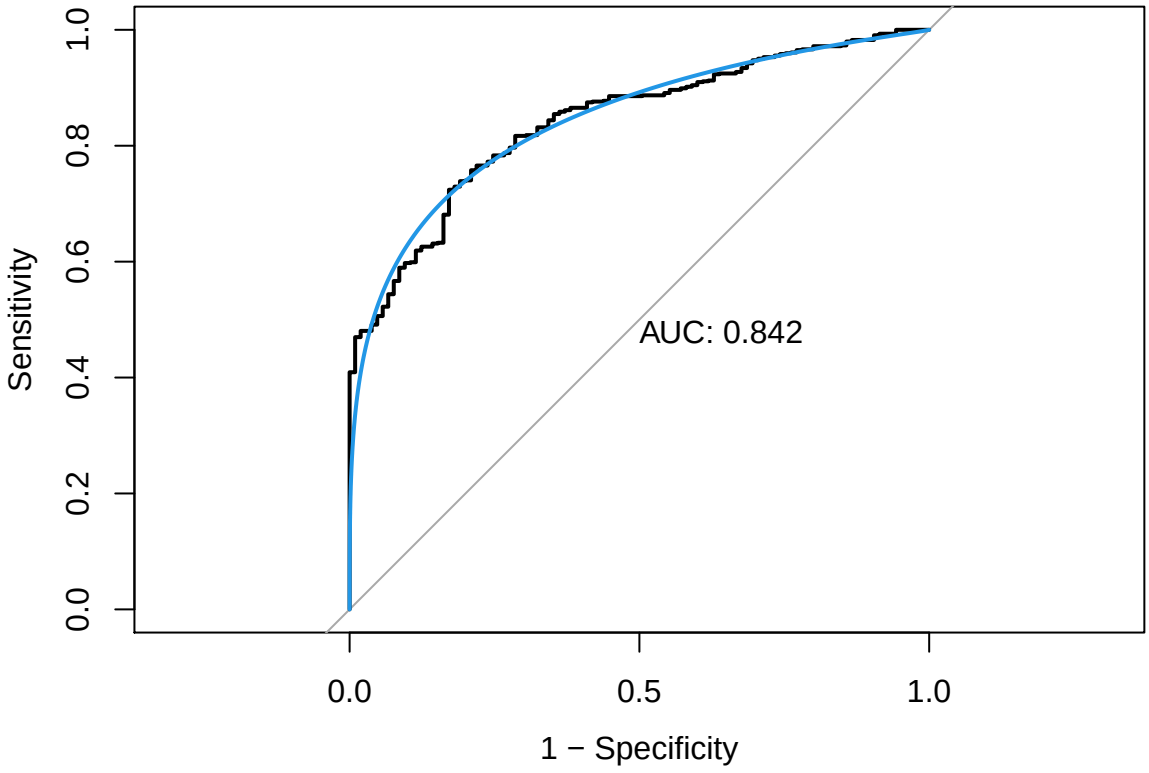
Model prediction performance

```
# glm
set.seed(1)

model.glm <- train(x = x,
```

```
                  y = y,
                  method = "glm",
                  metric = "ROC",
                  trControl = ctrl)

## Test data classification performance: confusion matrix at 0.5 cut-off
test.pred.prob <- predict(model.glm, newdata = x2,
                          type = "prob")[,2]
test.pred <- rep("yes", length(test.pred.prob))
test.pred[test.pred.prob > 0.5] <- "no"

confusionMatrix(data = as.factor(test.pred),
                reference = diabetes_data$diabetes[-trainRows],
                positive = "no")
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction yes  no
##        yes  15  15
##        no   90 728
##
##                Accuracy : 0.8762
##                  95% CI : (0.8521, 0.8976)
##     No Information Rate : 0.8762
##     P-Value [Acc > NIR] : 0.526
##
##                   Kappa : 0.1769
##
##  Mcnemar's Test P-Value : 5.136e-13
##
##             Sensitivity : 0.9798
##             Specificity : 0.1429
##          Pos Pred Value : 0.8900
##          Neg Pred Value : 0.5000
##              Prevalence : 0.8762
##          Detection Rate : 0.8585
##    Detection Prevalence : 0.9646
##       Balanced Accuracy : 0.5613
##
##        'Positive' Class : no
##
```

```
## Test data performance: ROC curve
glm.pred <- predict(model.glm, newdata = x2, type = "prob")[,2]
roc.glm <- roc(y2, glm.pred)
plot(roc.glm, legacy.axes = TRUE, print.auc = TRUE)
plot(smooth(roc.glm), col = 4, add = TRUE)
```


Sensitivity
0.0
0.2
0.4
0.6
0.8
1.0
AUC: 0.842
0.0
0.5
1.0
1 − Specificity