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)
```

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	Missing
1	gender [factor]	1. 1 2. 2	4706 (49.1%) 4872 (50.9%)	IIIIIIIII IIIIIIIIII	0 (0.0%)
2	age [numeric]	Mean (sd) : 32.4 (23.9) min < med < max: 1 < 28 < 80 IQR (CV) : 41 (0.7)	80 distinct values	: : : : : . : : : : : : : : . . : : : : : : : : : :	0 (0.0%)
3	race [factor]	1. 1 2. 2 3. 3 4. 4 5. 6 6. 7	1616 (16.9%) 893 (9.3%) 3449 (36.0%) 2148 (22.4%) 1033 (10.8%) 439 (4.6%)	III I IIIIIII IIII II	0 (0.0%)

No	Variable	Stats / Values	Freqs (% of Valid)	Graph	Missing
4	bmi [numeric]	Mean (sd) : 25.6 (7.9) min < med < max: 12.1 < 24.6 < 82.9 IQR (CV) : 10.4 (0.3)	436 distinct values	: . : : : : : : : : . : : : : .	706 (7.4%)
5	hdl [numeric]	Mean (sd) : 53.2 (15.2) min < med < max: 10 < 51 < 173 IQR (CV) : 19 (0.3)	116 distinct values	: : . : . : : : : : : .	2128 (22.2%)
6	ldl [numeric]	Mean (sd) : 106 (34.9) min < med < max: 14 < 103 < 375 IQR (CV) : 46 (0.3)	194 distinct values	: . : : : . : : : . : : : .	6553 (68.4%)
7	triglyceride [numeric]	Mean (sd) : 111.7 (115.9) min < med < max: 13 < 88 < 4233 IQR (CV) : 73 (1)	344 distinct values	: : : : :	6515 (68.0%)
8	insulin [numeric]	Mean (sd) : 13.4 (18.7) min < med < max: 0.1 < 9.3 < 682.5 IQR (CV) : 9.1 (1.4)	1716 distinct values	: : : : :	6567 (68.6%)
9	glucose [numeric]	Mean (sd) : 114 (45.5) min < med < max: 40 < 104 < 604 IQR (CV) : 44 (0.4)	227 distinct values	: : : : : : : : : :	7294 (76.2%)
10	bp_systolic [numeric]	Mean (sd) : 117.9 (18) min < med < max: 66 < 116 < 228 IQR (CV) : 20 (0.2)	71 distinct values	: : : : : . : : . : : : : .	2571 (26.8%)
11	bp_diastolic [numeric]	Mean (sd) : 65.7 (15) min < med < max: 0 < 66 < 122 IQR (CV) : 16 (0.2)	59 distinct values	: : . : : : : : : : : : : :	2571 (26.8%)
12	waist [numeric]	Mean (sd) : 86.9 (22.5) min < med < max: 40.2 < 87.4 < 177.9 IQR (CV) : 31.6 (0.3)	1030 distinct values	: . : : : . : : : : : : : : : : : : : : : : .	1091 (11.4%)
13	lifestyle [numeric]	Mean (sd) : 478.5 (642.1) min < med < max: 0 < 480 < 9999 IQR (CV) : 300 (1.3)	36 distinct values	: : : : :	2625 (27.4%)

No	Variable	Stats / Values	Freqs (% of Valid)	Graph	Missing
14	education [factor]	1. 1 2. 2 3. 3 4. 4 5. 5 6. 7 7. 9	442 (7.9%) 761 (13.6%) 1261 (22.6%) 1715 (30.7%) 1406 (25.1%) 2 (0.0%) 5 (0.1%)	I II IIII IIIIII IIIII	3986 (41.6%)
15	married [factor]	1. 1 2. 2 3. 3 4. 4 5. 5 6. 6 7. 77 8. 99	2866 (51.3%) 419 (7.5%) 637 (11.4%) 170 (3.0%) 1096 (19.6%) 401 (7.2%) 2 (0.0%) 1 (0.0%)	IIIIIIIIII I II III I	3986 (41.6%)
16	depression [numeric]	Mean (sd) : 0.4 (0.8) min < med < max: 0 < 0 < 9 IQR (CV) : 0 (2.1)	0 : 3955 (75.5%) 1 : 876 (16.7%) 2 : 205 (3.9%) 3 : 194 (3.7%) 7 : 2 (0.0%) 9 : 3 (0.1%)	IIIIIIIIIIIIIII III	4343 (45.3%)
17	sleep [numeric]	Mean (sd) : 7 (3.2) min < med < max: 2 < 7 < 99 IQR (CV) : 2 (0.5)	12 distinct values	: : : : :	3300 (34.5%)
18	diabetes [factor]	1. yes 2. no	737 (7.7%) 8841 (92.3%)	I IIIIIIIIIIIIIIIIII	0 (0.0%)

```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

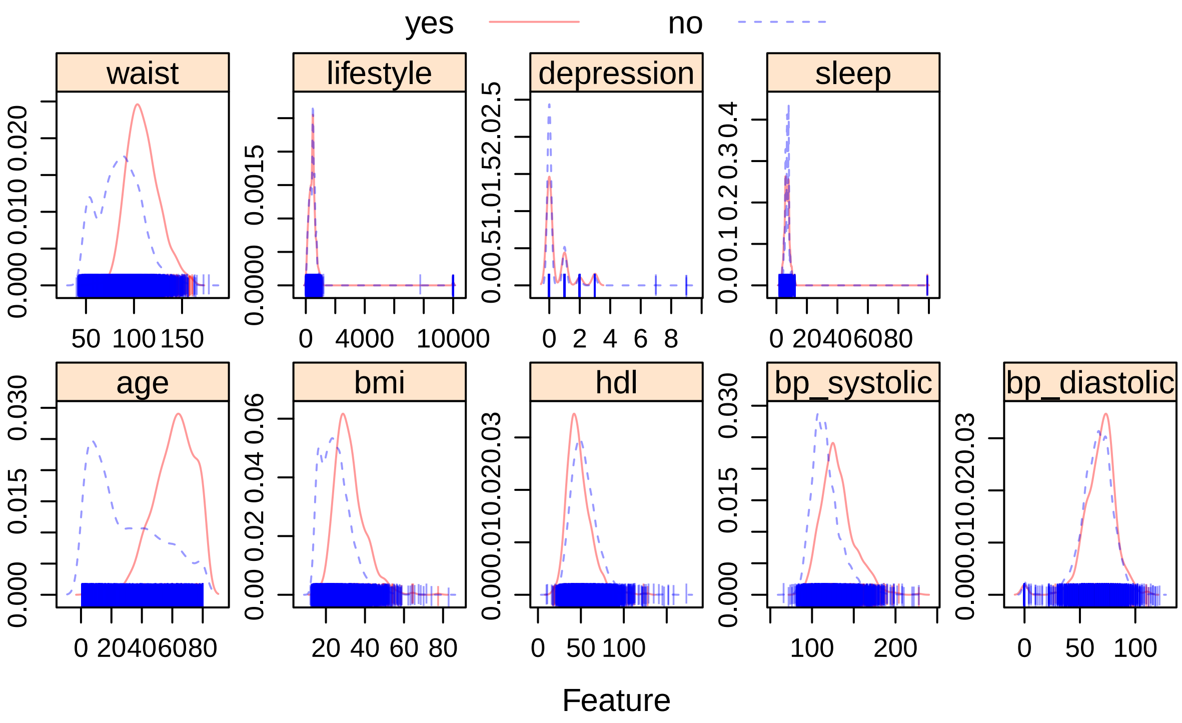
No	Variable	Stats / Values	Freqs (% of Valid)	Graph	Missing
1	gender [factor]	1. 1 2. 2	4706 (49.1%) 4872 (50.9%)	IIIIIIIII IIIIIIIIII	0 (0.0%)
2	age [numeric]	Mean (sd) : 32.4 (23.9) min < med < max: 1 < 28 < 80 IQR (CV) : 41 (0.7)	80 distinct values	: : : : : . : : : : : : : : . . : : : : : : : : : :	0 (0.0%)

No	Variable	Stats / Values	Freqs (% of Valid)	Graph	Missing
3	race [factor]	1. 1	1616 (16.9%)	III	0 (0.0%)
		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 [numeric]	Mean (sd) : 25.6 (7.9) min < med < max: 12.1 < 24.6 < 82.9 IQR (CV) : 10.4 (0.3)	436 distinct values	: . : : : : : : : : . : : : : .	706 (7.4%)
5	hdl [numeric]	Mean (sd) : 53.2 (15.2) min < med < max: 10 < 51 < 173 IQR (CV) : 19 (0.3)	116 distinct values	: : . : . : : : : : : .	2128 (22.2%)
6	bp_systolic [numeric]	Mean (sd) : 117.9 (18) min < med < max: 66 < 116 < 228 IQR (CV) : 20 (0.2)	71 distinct values	: : : : : . : : . : : : : .	2571 (26.8%)
7	bp_diastolic [numeric]	Mean (sd) : 65.7 (15) min < med < max: 0 < 66 < 122 IQR (CV) : 16 (0.2)	59 distinct values	: : . : : : : : : : : : : :	2571 (26.8%)
8	waist [numeric]	Mean (sd) : 86.9 (22.5) min < med < max: 40.2 < 87.4 < 177.9 IQR (CV) : 31.6 (0.3)	1030 distinct values	: . : : : . : : : : : : : : : : : : : : : : .	1091 (11.4%)
9	lifestyle [numeric]	Mean (sd) : 478.5 (642.1) min < med < max: 0 < 480 < 9999 IQR (CV) : 300 (1.3)	36 distinct values	: : : : :	2625 (27.4%)
10	education [factor]	1. 1	442 (7.9%)	I	3986 (41.6%)
		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 [factor]	1. 1	2866 (51.3%)	IIIIIIIIII	3986 (41.6%)
		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%)		

No	Variable	Stats / Values	Freqs (% of Valid)	Graph	Missing
12	depression [numeric]	Mean (sd) : 0.4 (0.8) min < med < max: 0 < 0 < 9 IQR (CV) : 0 (2.1)	0 : 3955 (75.5%) 1 : 876 (16.7%) 2 : 205 (3.9%) 3 : 194 (3.7%) 7 : 2 (0.0%) 9 : 3 (0.1%)	IIIIIIIIIIIIIII III	4343 (45.3%)
13	sleep [numeric]	Mean (sd) : 7 (3.2) min < med < max: 2 < 7 < 99 IQR (CV) : 2 (0.5)	12 distinct values	: : : : :	3300 (34.5%)
14	diabetes [factor]	1. yes 2. no	737 (7.7%) 8841 (92.3%)	I IIIIIIIIIIIIIIIIII	0 (0.0%)

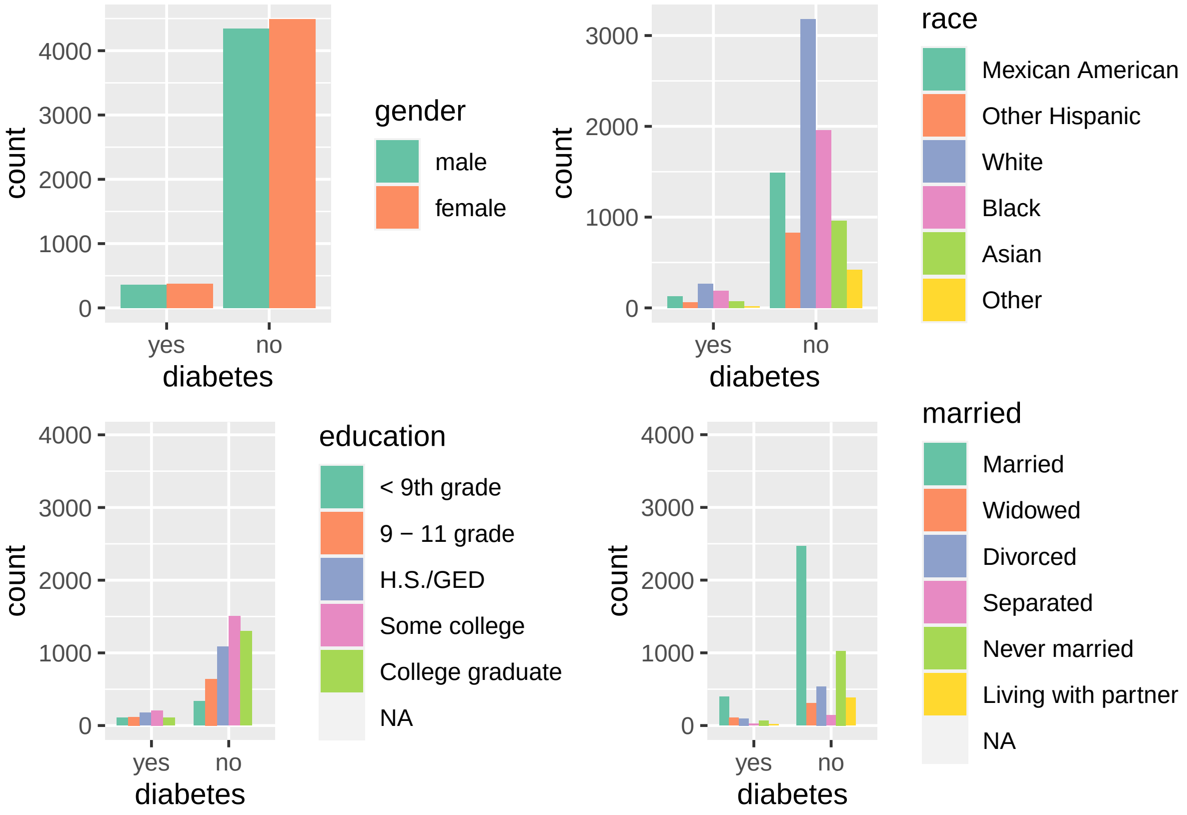
Density plots (numerical covariates)

```
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))
```



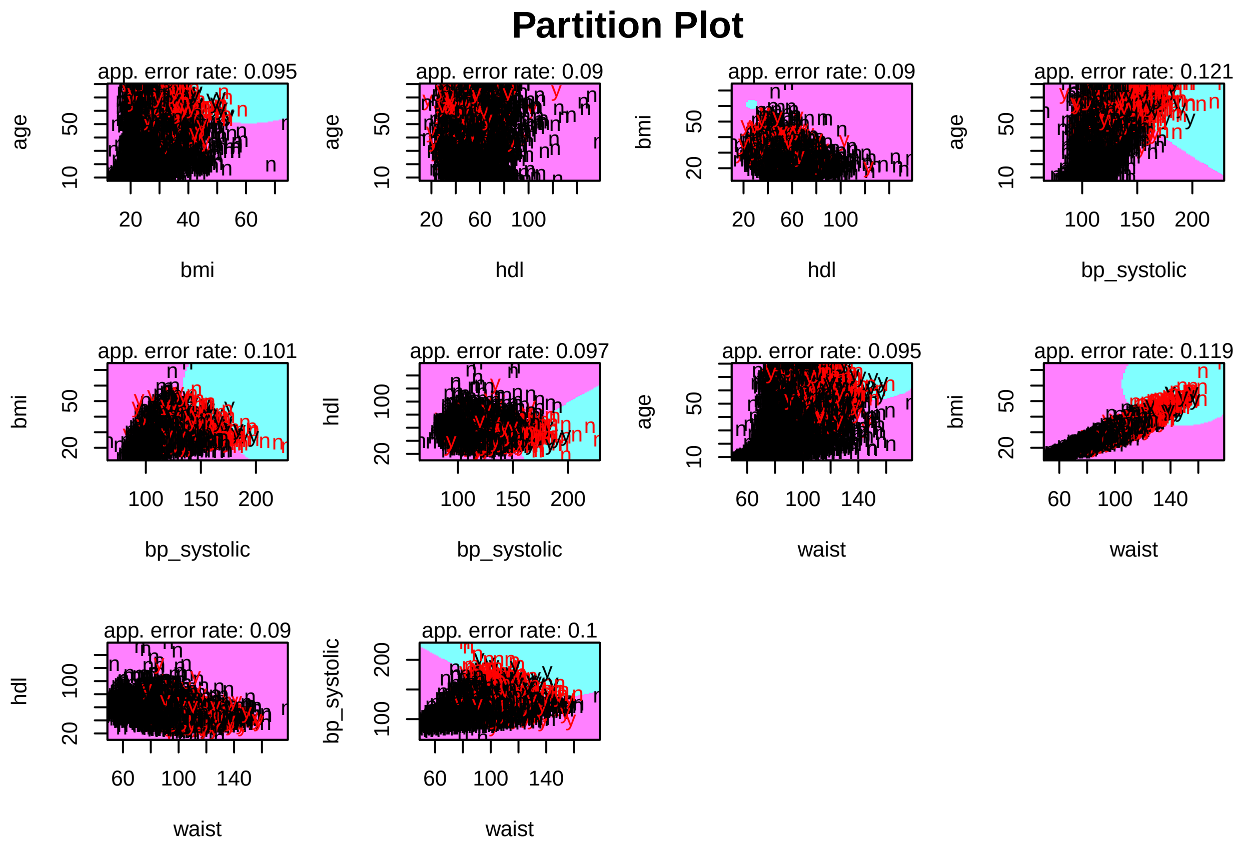
Bar plots (categorical covariates)

```
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

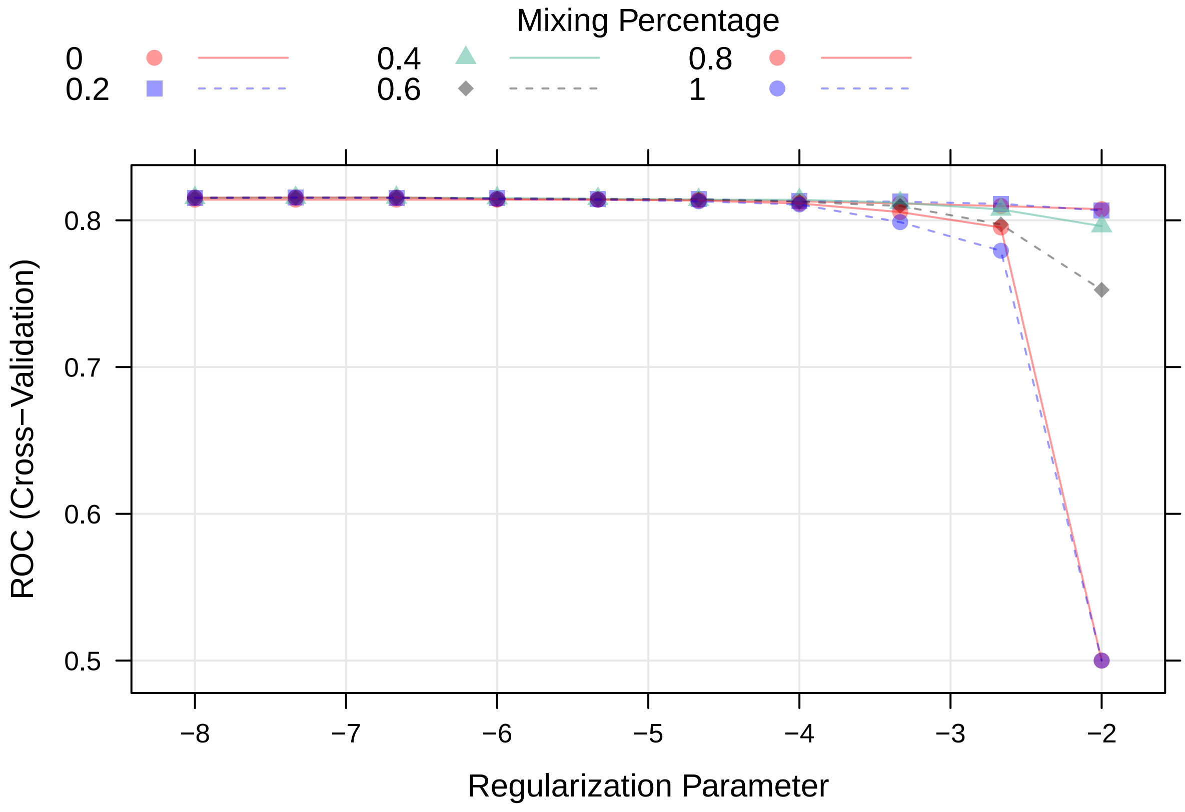
Prep/partition data

```
# 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))
```



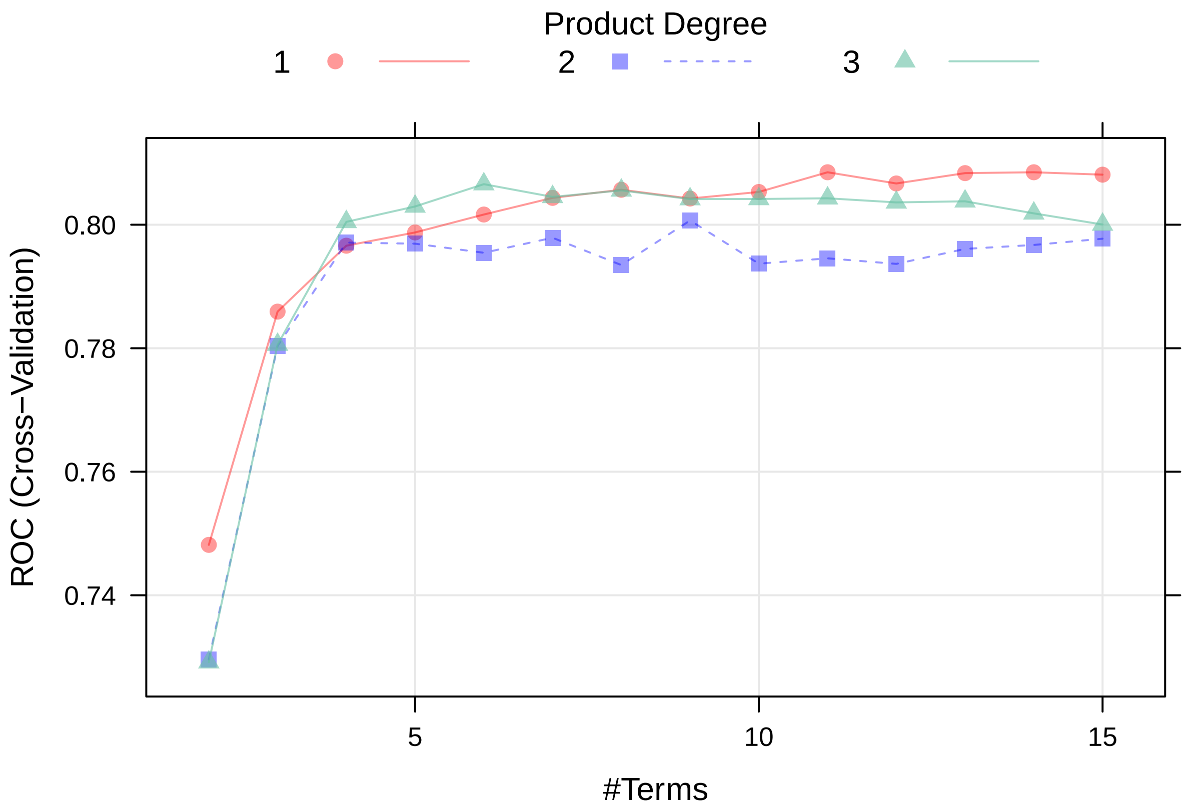
```
model.glmn$bestTune
```

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

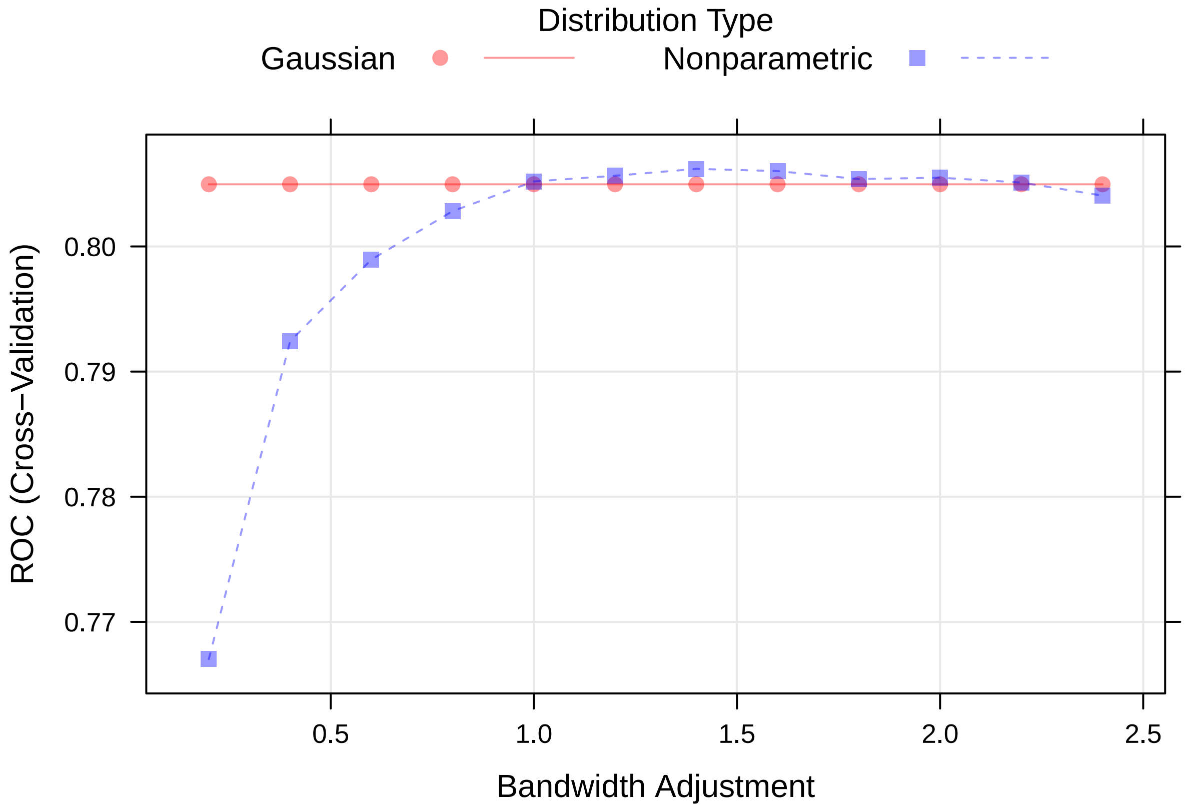
```
## Non-linear Logistic regression: GAM, MARS
# GAM
#set.seed(1)
#model.gam <- train(x = x,
#                    y = y,
#                    method = "gam",
#                    metric = "ROC",
#                    trControl = ctrl)

#model.gam$finalModel

# MARS
set.seed(1)

model.mars <- train(x = x,
                    y = y,
                    method = "earth",
                    tuneGrid = expand.grid(degree = 1:3,
                                           nprune = 2:15),
                    metric = "ROC",
                    trControl = ctrl)

plot(model.mars)
```



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

```
   (Intercept)          h(age-76)          h(76-age)        h(waist-88)
   1.658285895        0.122437959        0.062661766       -0.055430289
     h(74-hdl) h(86-bp_diastolic)              race3        h(bmi-38.9)
  -0.027315168       -0.012127252        0.549041828        0.150319736
      married2     h(waist-104.2)        h(bmi-48.4)         education5
  -0.214623872        0.005005836       -0.186737365        0.695318548
    education3         education4
   0.440916855        0.367828136
```

```r
## 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

Model comparison

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

summary(res)
```

Call: summary.resamples(object = res)

Models: GLM, GLMNET, MARS, NB 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 0.7704024 0.7818502 0.7989819 0.8084961 0.8296717 0.8719239 0 NB 0.7728447 0.7796216 0.7986685 0.8061999 0.8354096 0.8446480 0

Sens Min. 1st Qu. Median Mean 3rd Qu. Max. NA's GLM 0.07142857 0.11904762 0.1292913 0.14147287 0.1607143 0.2325581 0 GLMNET 0.04761905 0.07142857 0.1057586 0.09905869 0.1183555 0.1627907 0 MARS 0.04761905 0.11904762 0.1428571 0.13909192 0.1578073 0.2325581 0 NB 0.16666667 0.22245293 0.3095238 0.28117386 0.3313953 0.3720930 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 0.9747644 0.9831650 0.9828509 0.9882296 1.0000000 0 NB 0.9259259 0.9351852 0.9478114 0.9471855 0.9587542 0.9664430 0