DSII Midterm Project

Yiru Gong, yg2832

2022-03-27

Contents

```
library(tidyverse)
library(summarytools)
library(caret)
library(MASS)
library(mlbench)
library(pROC) #ROCR
library(pdp)
library(vip)
library(AppliedPredictiveModeling) #for transparentTheme function
```

Data Input

```
data = read.csv('Covid19_vacc_predict_handout.csv')
data = data %>%
  na.omit() %>%
  dplyr::select(-id) %>%
  mutate(
    atlas_type_2015_mining_no = factor(atlas_type_2015_mining_no),
    covid_vaccination = factor(covid_vaccination),
    hum_region = factor(hum_region),
    sex_cd = factor(sex_cd),
    race_cd = factor(race_cd),
    lang_spoken_cd = factor(lang_spoken_cd),
    atlas_low_education_2015_update = factor(atlas_low_education_2015_update)
    )
# summary(data)
# by(data[,c(5,7,8,10,11,17,18)], data$covid_vaccination, summary)
dfSummary(data[,c(5,7,8,10,11,17,18)])
```

Data Frame Summary
Dimensions: 8308 x 7
Duplicates: 7802

No	Variable	Stats / Values	Freqs (% of Valid)	Graph	Valid	Missing
1	atlas_type_2015_mining_no [factor]	1. 0 2. 1	8177 (98.4%) 131 (1.6%)	IIIIIIIIIIIIIIIII	8308 (100.0%)	0 (0.0%)
2	covid_vaccination [factor]	1. no_vacc 2. vacc	6682 (80.4%) 1626 (19.6%)	IIIIIIIIIIIIIII III	8308 (100.0%)	0 (0.0%)

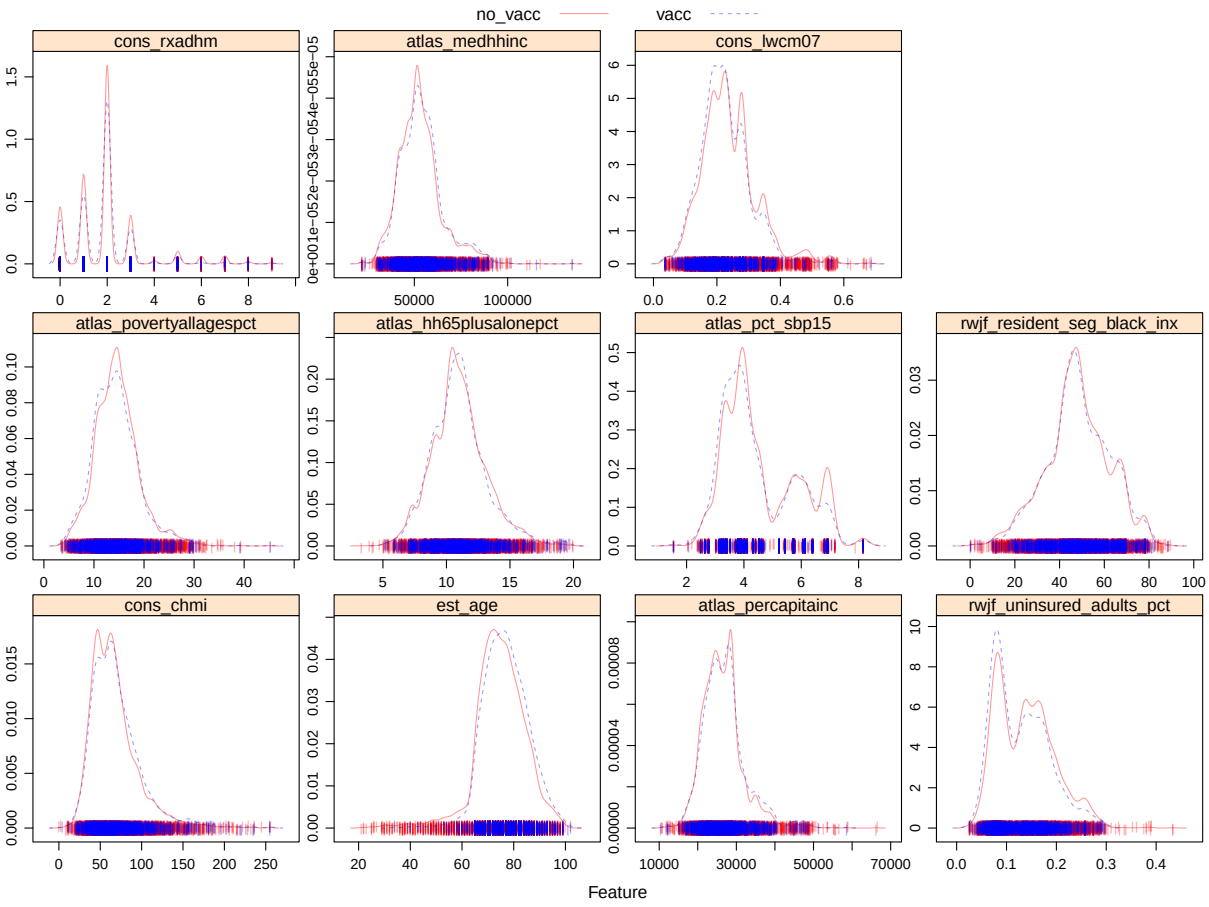
No	Variable	Stats / Values	Freqs (% of Valid)	Graph	Valid	Missing
3	hum_region [factor]	1. CALIFORNIA/NEVADA 2. CENTRAL 3. CENTRAL WEST 4. EAST 5. EAST CENTRAL 6. FLORIDA 7. GREAT LAKES/CENTRAL NORTH 8. GULF STATES 9. INTERMOUNTAIN 10. MID-ATLANTIC/NORTH CAROLI [5 others]	299 (3.6%) 551 (6.6%) 238 (2.9%) 491 (5.9%) 1370 (16.5%) 607 (7.3%) 1111 (13.4%) 454 (5.5%) 220 (2.6%) 845 (10.2%) 2122 (25.5%)	I	8308 (100.0%)	0 (0.0%)
4	sex_cd [factor]	1. F 2. M	4527 (54.5%) 3781 (45.5%)	IIIIIIIII IIIIIIII	8308 (100.0%)	0 (0.0%)
5	lang_spoken_cd [factor]	1. * 2. CHI 3. CRE 4. ENG 5. KOR 6. OTH 7. SPA 8. VIE	10 (0.1%) 13 (0.2%) 4 (0.0%) 7957 (95.8%) 7 (0.1%) 34 (0.4%) 276 (3.3%) 7 (0.1%)		8308 (100.0%)	0 (0.0%)
6	atlas_low_education_2015_update [factor]	1. 0 2. 1	7769 (93.5%) 539 (6.5%)	IIIIIIIIIIIIIIIII	8308 (100.0%)	0 (0.0%)
7	race_cd [factor]	1. 0 2. 1 3. 2 4. 3 5. 4 6. 5 7. 6	160 (1.9%) 7317 (88.1%) 558 (6.7%) 80 (1.0%) 56 (0.7%) 129 (1.6%) 8 (0.1%)	IIIIIIIIIIIIIII	8308 (100.0%)	0 (0.0%)

```
# cat_sum = NULL
# for (n in c(5,8,10,11,17,18)){
#   cat = data[,c(n,7)]
#   name = colnames(cat)[1]
#   cat2 = cat %>%
#     group_by(covid_vaccination,cat[,1]) %>%
#     count() %>%
#     rename(cat=`cat[, 1]`) %>%
#     pivot_wider(
#       names_from = covid_vaccination,
#       values_from = n
#     ) %>%
#     mutate(variable = name) %>%
#     relocate(variable,everything())
#   cat_sum = rbind(cat_sum,cat2)
# }
```

```
# knitr::kable(cat_sum)

# cat_sum %>%
#   pivot_longer(
#     c("no_vacc","vacc"),
#     names_to = 'covid_vaccination',
#     values_to = 'count'
#   ) %>%
#   ggplot(aes(variable,count,group=covid_vaccination,fill=cat))+geom_bar(stat = 'identity')

data2 = model.matrix(covid_vaccination ~ ., data)[ ,-1]
```

Exploratory analysis

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

#figure 1
featurePlot(x = data[,-c(5,7,8,10,11,17,18)],
            y = data$covid_vaccination,
            scales = list(x = list(relation = "free"),
                          y = list(relation = "free")),
            plot = "density", pch = "|",
            auto.key = list(columns = 2))
```



Data split

```
set.seed(1)
rowTrain <- createDataPartition(y = data$covid_vaccination,
                                p = 0.7,
                                list = FALSE)
x = data2[rowTrain,]
y = data$covid_vaccination[rowTrain]
x2 = data2[-rowTrain,]
y2 = data$covid_vaccination[-rowTrain]
```

Model fitting

GLM

```
ctrl <- trainControl(method = "repeatedcv",
                     summaryFunction = twoClassSummary,
                     classProbs = TRUE)
```

```
set.seed(1)
model.glm <- train(x,y,
                   method = "glm",
                   metric = "ROC",
                   trControl = ctrl)
summary(model.glm)
```

```
##
## Call:
## NULL
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.0312  -0.7067  -0.6123  -0.4717   2.4000
##
## Coefficients:
##                                          Estimate Std. Error z value Pr(>|z|)
## (Intercept)                            -3.500e+00  1.309e+00  -2.674   0.0075
## cons_chmi                               1.781e-03  1.252e-03   1.423   0.1549
## est_age                                 2.222e-02  4.148e-03   5.357 8.48e-08
## atlas_percapitainc                     -4.337e-06  7.893e-06  -0.550   0.5827
## rwjf_uninsured_adults_pct              -1.880e+00  9.177e-01  -2.049   0.0405
## atlas_type_2015_mining_no1             -1.429e-01  3.298e-01  -0.433   0.6648
## atlas_povertyallagespct                 4.796e-03  1.001e-02   0.479   0.6320
## hum_regionCENTRAL                      -3.391e-01  2.235e-01  -1.517   0.1293
## `hum_regionCENTRAL WEST`               -5.710e-01  2.739e-01  -2.085   0.0371
## hum_regionEAST                         -2.745e-01  2.306e-01  -1.191   0.2338
## `hum_regionEAST CENTRAL`               -3.866e-01  1.968e-01  -1.964   0.0495
## hum_regionFLORIDA                      -5.517e-01  2.301e-01  -2.398   0.0165
## `hum_regionGREAT LAKES/CENTRAL NORTH`  -1.103e-01  1.985e-01  -0.556   0.5785
## `hum_regionGULF STATES`                 1.823e-01  2.361e-01   0.772   0.4401
## hum_regionINTERMOUNTAIN                -3.571e-01  2.734e-01  -1.306   0.1915
## `hum_regionMID-ATLANTIC/NORTH CAROLINA` -2.551e-01  2.040e-01  -1.251   0.2111
## `hum_regionMID-SOUTH`                  -6.210e-01  2.460e-01  -2.525   0.0116
## hum_regionNORTHEAST                    -3.916e-01  2.343e-01  -1.671   0.0947
## hum_regionPACIFIC                      -7.853e-01  1.097e+00  -0.716   0.4741
## hum_regionSOUTHEAST                    -3.564e-02  2.344e-01  -0.152   0.8792
## hum_regionTEXAS                        -5.237e-01  2.653e-01  -1.974   0.0484
## atlas_hh65plusalonepct                  1.389e-02  1.798e-02   0.773   0.4398
## sex_cdM                                -3.688e-02  7.162e-02  -0.515   0.6066
## lang_spoken_cdCHI                      -4.650e-01  1.531e+00  -0.304   0.7613
## lang_spoken_cdCRE                      -1.293e+01  6.220e+02  -0.021   0.9834
## lang_spoken_cdENG                       2.357e-01  1.098e+00   0.215   0.8299
## lang_spoken_cdKOR                      -1.277e+01  3.292e+02  -0.039   0.9690
## lang_spoken_cdOTH                       5.733e-01  1.212e+00   0.473   0.6361
## lang_spoken_cdSPA                       7.368e-02  1.121e+00   0.066   0.9476
## lang_spoken_cdVIE                      -1.251e+01  3.516e+02  -0.036   0.9716
## atlas_pct_sbp15                        -9.983e-03  4.995e-02  -0.200   0.8416
## rwjf_resident_seg_black_inx             1.025e-03  2.753e-03   0.372   0.7096
## cons_rxadhm                             2.659e-02  3.228e-02   0.824   0.4101
## atlas_medhhinc                          5.727e-06  4.164e-06   1.375   0.1690
## cons_lwcm07                            -1.092e+00  5.285e-01  -2.065   0.0389
## atlas_low_education_2015_update1       -3.694e-02  1.615e-01  -0.229   0.8191
```

```
## race_cd1                                4.168e-01  2.922e-01   1.426   0.1538
## race_cd2                                3.727e-01  3.216e-01   1.159   0.2465
## race_cd3                                2.405e-01  4.590e-01   0.524   0.6003
## race_cd4                                2.875e-01  5.173e-01   0.556   0.5784
## race_cd5                               -1.550e-01  4.928e-01  -0.314   0.7532
## race_cd6                               -1.227e+01  4.404e+02  -0.028   0.9778
##
## (Intercept)                            **
## cons_chmi
## est_age                                ***
## atlas_percapitainc
## rwjf_uninsured_adults_pct              *
## atlas_type_2015_mining_no1
## atlas_povertyallagespct
## hum_regionCENTRAL
## `hum_regionCENTRAL WEST`               *
## hum_regionEAST
## `hum_regionEAST CENTRAL`               *
## hum_regionFLORIDA                      *
## `hum_regionGREAT LAKES/CENTRAL NORTH`
## `hum_regionGULF STATES`
## hum_regionINTERMOUNTAIN
## `hum_regionMID-ATLANTIC/NORTH CAROLINA`
## `hum_regionMID-SOUTH`                  *
## hum_regionNORTHEAST                    .
## hum_regionPACIFIC
## hum_regionSOUTHEAST
## hum_regionTEXAS                        *
## atlas_hh65plusalonepct
## sex_cdM
## lang_spoken_cdCHI
## lang_spoken_cdCRE
## lang_spoken_cdENG
## lang_spoken_cdKOR
## lang_spoken_cdOTH
## lang_spoken_cdSPA
## lang_spoken_cdVIE
## atlas_pct_sbp15
## rwjf_resident_seg_black_inx
## cons_rxadhm
## atlas_medhhinc
## cons_lwcm07                            *
## atlas_low_education_2015_update1
## race_cd1
## race_cd2
## race_cd3
## race_cd4
## race_cd5
## race_cd6
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
```

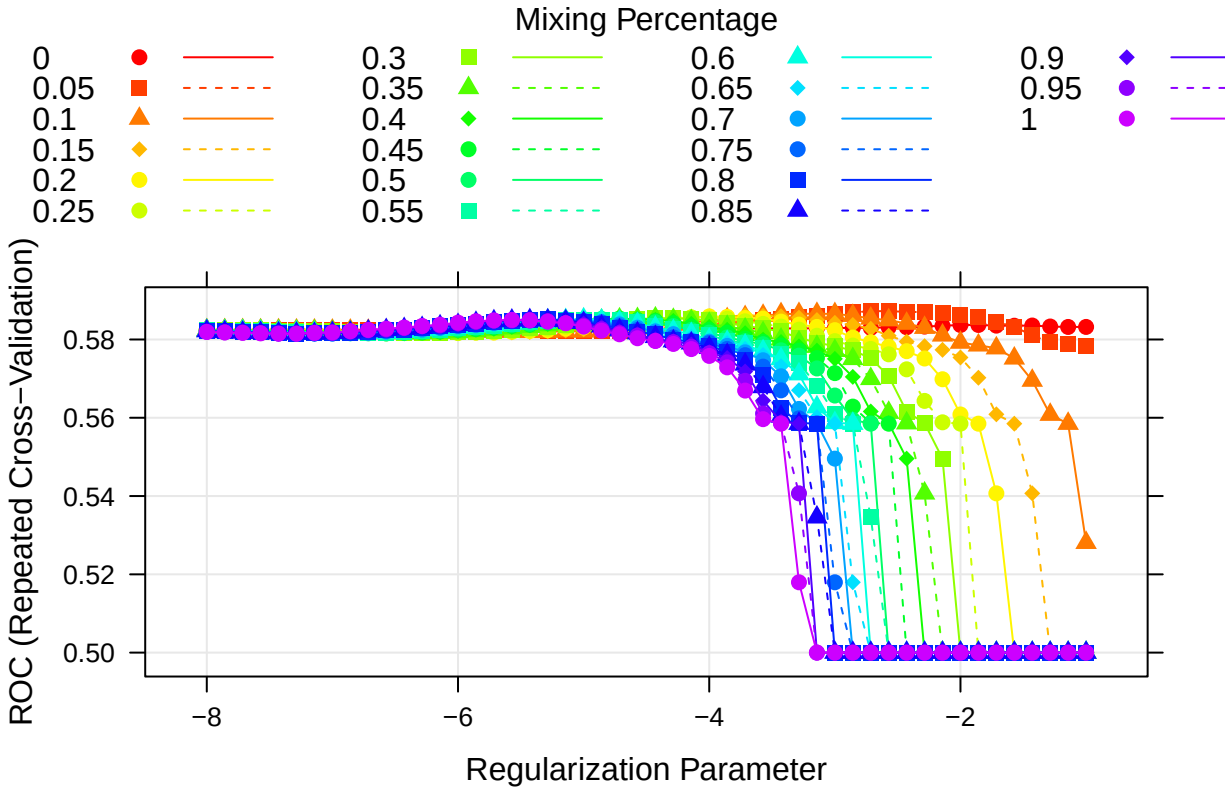
```
##     Null deviance: 5753.4  on 5816  degrees of freedom
## Residual deviance: 5608.2  on 5775  degrees of freedom
## AIC: 5692.2
##
## Number of Fisher Scoring iterations: 13
```

Penalized logistic regression

```
glmnGrid <- expand.grid(.alpha = seq(0, 1, length = 21),
                        .lambda = exp(seq(-8, -1, length = 50)))
set.seed(1)
model.glmn <- train(x, y,
                    method = "glmnet",
                    tuneGrid = glmnGrid,
                    metric = "ROC",
                    trControl = ctrl)

model.glmn$bestTune
```

```
##    alpha     lambda
## 89  0.05 0.07642629
```

```
myCol<- rainbow(25)
myPar <- list(superpose.symbol = list(col = myCol),
              superpose.line = list(col = myCol))

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



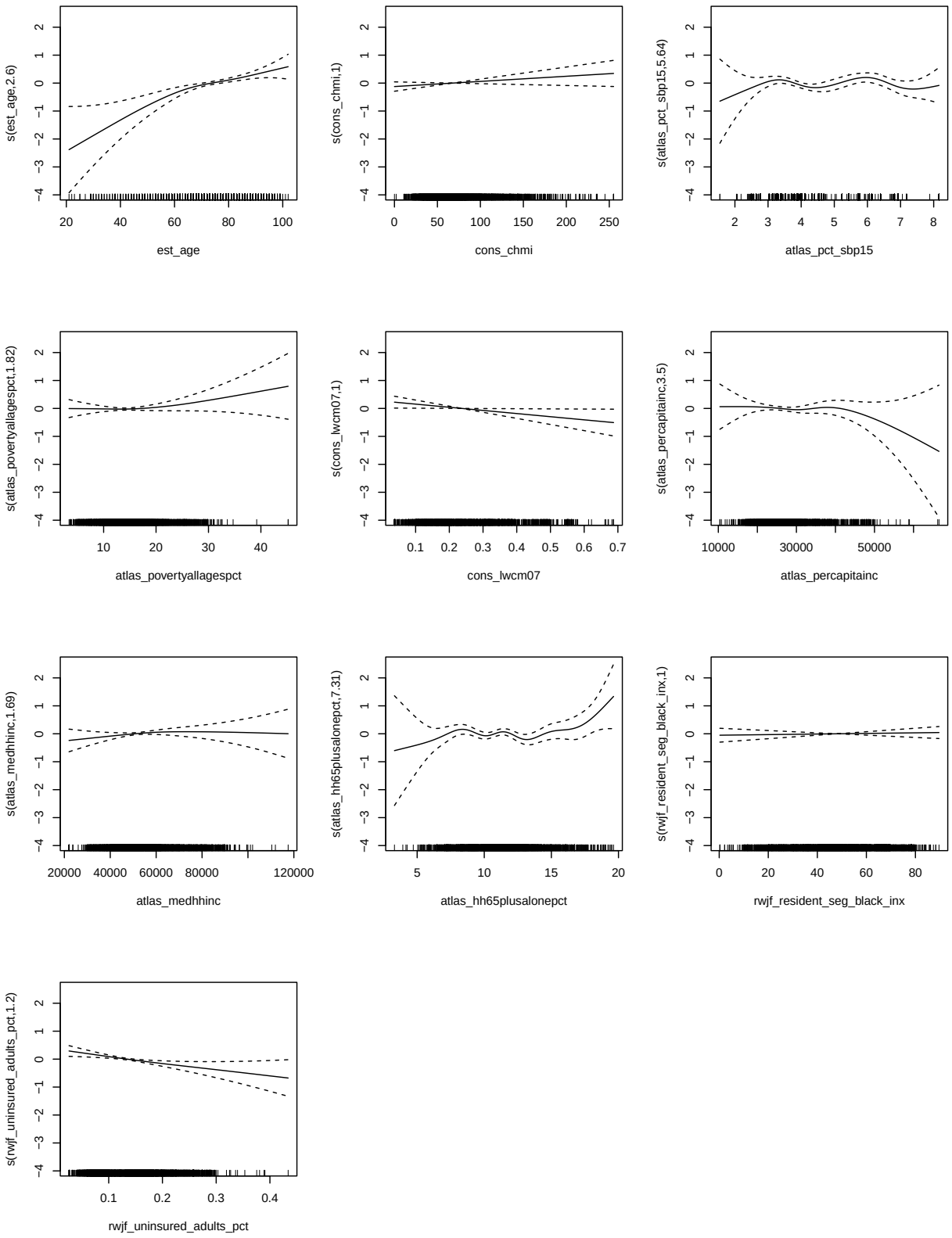
GAM

```
set.seed(1)
model.gam <- train(data[rowTrain,-c(7:8)], y,
                   method = "gam",
                   metric = "ROC",
                   trControl = ctrl)
### row 8: hum_region report error

model.gam$finalModel
```

```
##
## Family: binomial
## Link function: logit
##
## Formula:
## .outcome ~ sex_cd + atlas_low_education_2015_update + race_cd +
##     cons_rxadhm + s(est_age) + s(cons_chmi) + s(atlas_pct_sbp15) +
##     s(atlas_povertyallagespct) + s(cons_lwcm07) + s(atlas_percapitainc) +
##     s(atlas_medhhinc) + s(atlas_hh65plusalonepct) + s(rwjf_resident_seg_black_inx) +
##     s(rwjf_uninsured_adults_pct)
##
## Estimated degrees of freedom:
```

```
## 2.60 1.00 5.64 1.82 1.00 3.50 1.69
## 7.31 1.00 1.20  total = 36.76
##
## UBRE score: -0.02449249
```

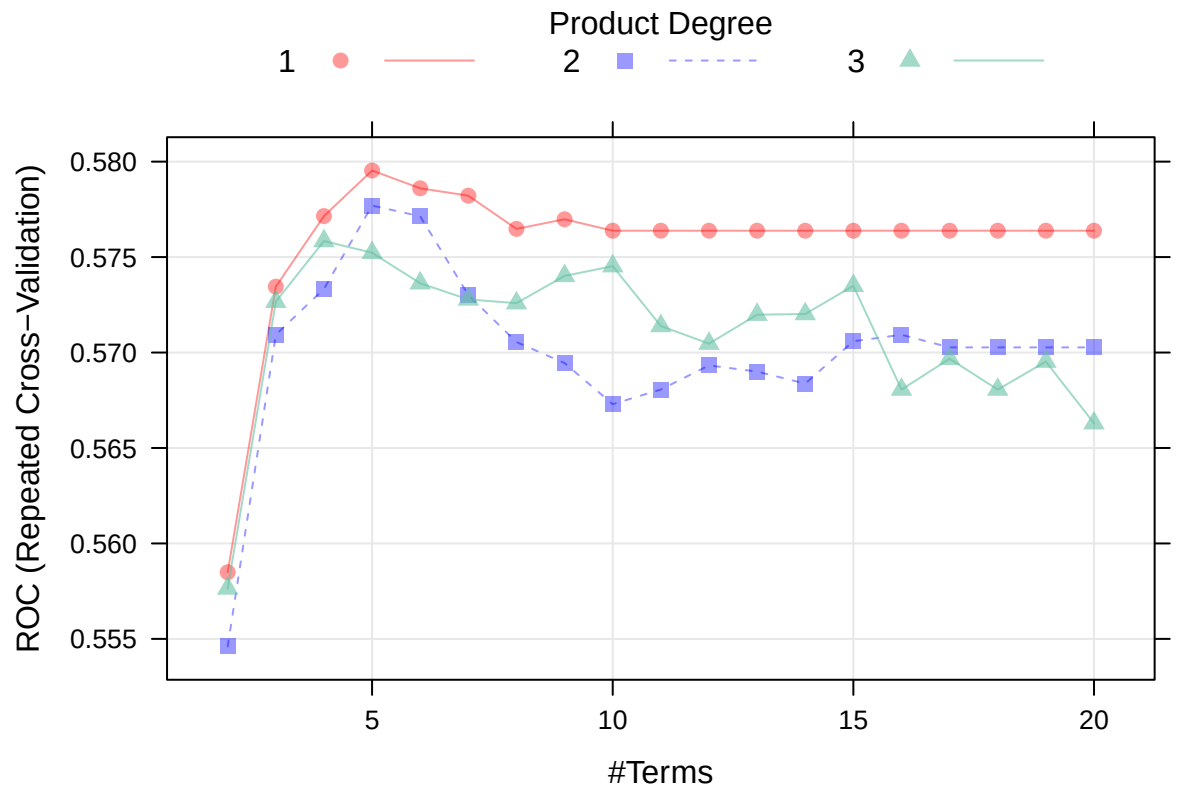
```
# fig 2
par(mfrow=c(4,3))
plot(model.gam$finalModel)
```


s(est_age,2.6)
est_age
s(cons_chmi,1)
cons_chmi
s(atlas_pct_sbp15,5.64)
atlas_pct_sbp15
s(atlas_povertyallagespct,1.82)
atlas_povertyallagespct
s(cons_lwcm07,1)
cons_lwcm07
s(atlas_percapitainc,3.5)
atlas_percapitainc
s(atlas_medhhinc,1.69)
atlas_medhhinc
s(atlas_hh65plusalonepct,7.31)
atlas_hh65plusalonepct
s(rwjf_resident_seg_black_inx,1)
rwjf_resident_seg_black_inx
s(rwjf_uninsured_adults_pct,1.2)
rwjf_uninsured_adults_pct

```
# ## add data preprocessing
# model.gam.proc <- train(data[rowTrain,-c(7:8)], y,
#                         preProcess = c("zv"),
#                      method = "gam",
#                      metric = "ROC",
#                      trControl = ctrl)
# plot(model.gam.proc$finalModel, select = 3)
```

MARS

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

plot(model.mars)
```

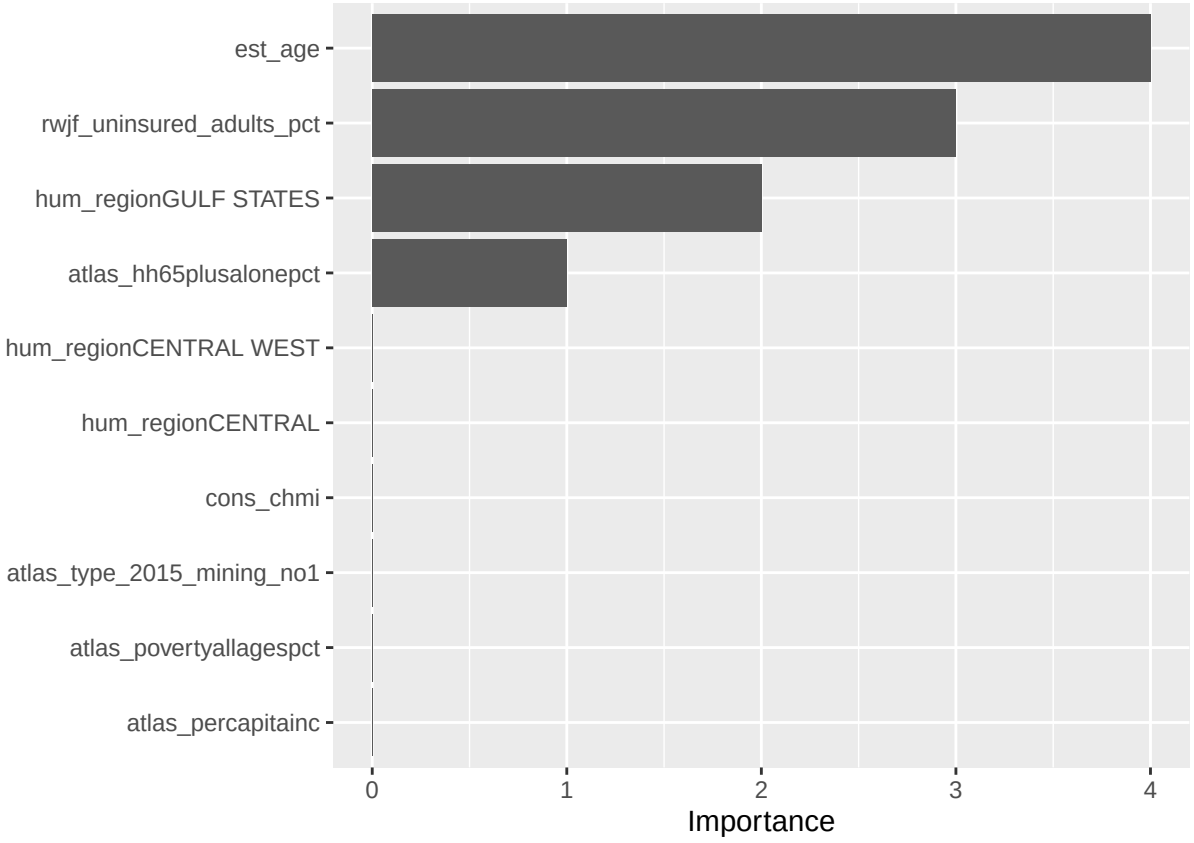



```
model.mars$bestTune
```

```
##   nprune degree
```

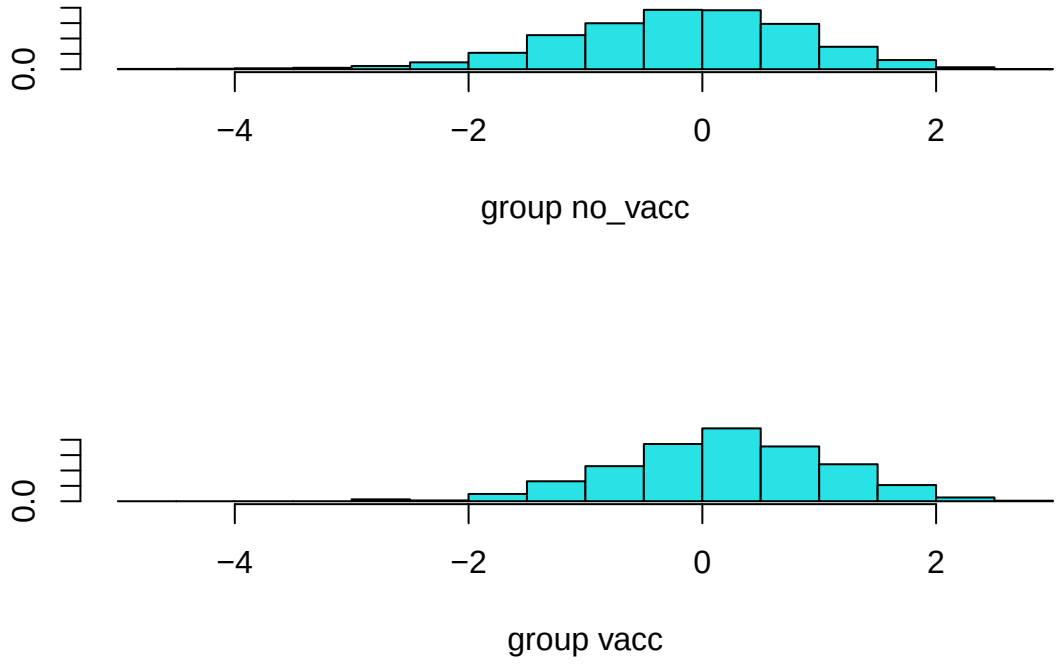
```
## 4      5      1
```

```
# fig 3
vip(model.mars$finalModel)
```



LDA

```
lda.fit <- lda(y~x)
plot(lda.fit)
```



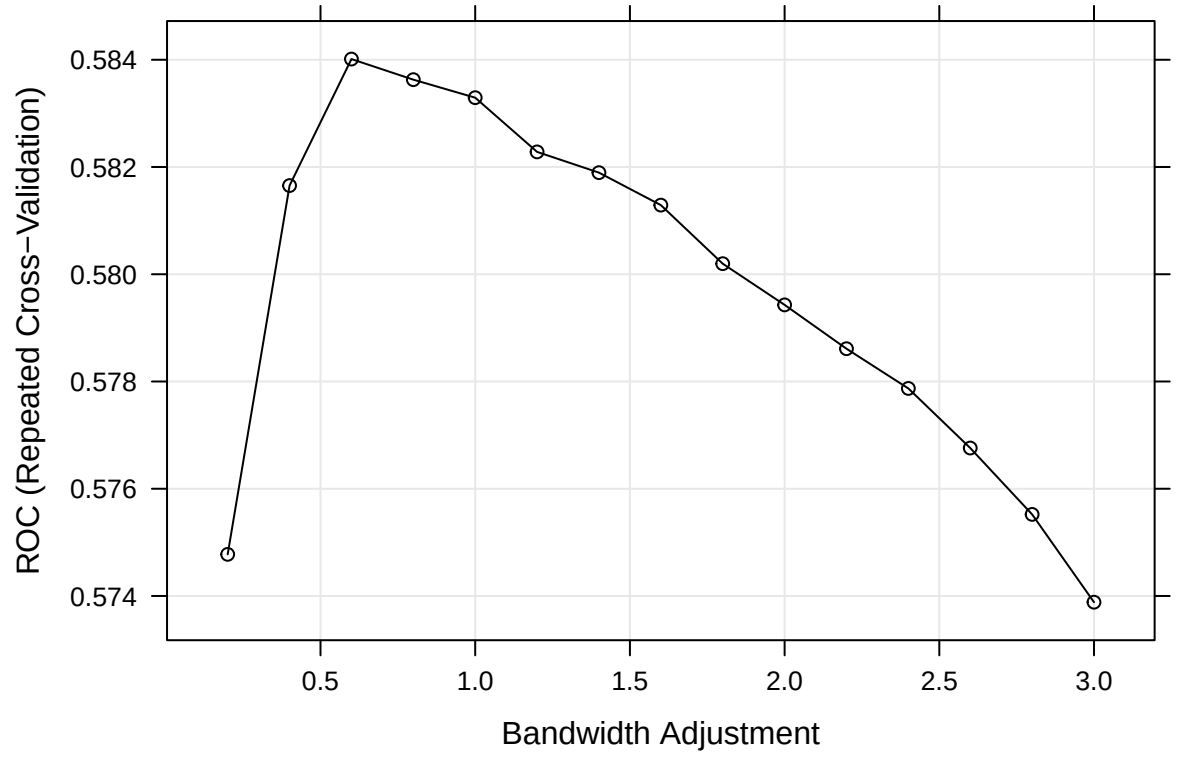

```
set.seed(1)
model.lda <- train(x, y,
                   method = "lda",
                   metric = "ROC",
                   trControl = ctrl)
```

QDA

```
data_limit = data[,-c(8,11,18)]
data2_limit = model.matrix(covid_vaccination~., data_limit)[,-1]
x_limit = data2_limit[rowTrain,]
x2_limit = data2_limit[-rowTrain,]


set.seed(1)
model.qda <- train(x_limit, y,
                   method = "qda",
                   metric = "ROC",
                   trControl = ctrl)
```

Naive Bayes (NB)

```
nbGrid <- expand.grid(usekernel = TRUE, #FALSE
                      fL = 1,
                      adjust = seq(.2, 3, by = .2))

set.seed(1)
model.nb <- train(data[rowTrain,-7], y,
                  method = "nb",
                  tuneGrid = nbGrid,
                  metric = "ROC",
                  trControl = ctrl)

plot(model.nb)
```




```
model.nb$bestTune
```

```
##   fL usekernel adjust
## 3  1      TRUE    0.6
```

KNN

```
set.seed(1)
model.knn <- train(x, y,
                   method = "knn",
                   metric = "ROC",
                   trControl = ctrl)
model.knn$bestTune
```

```
##   k
## 3 9
```

Model Comparison

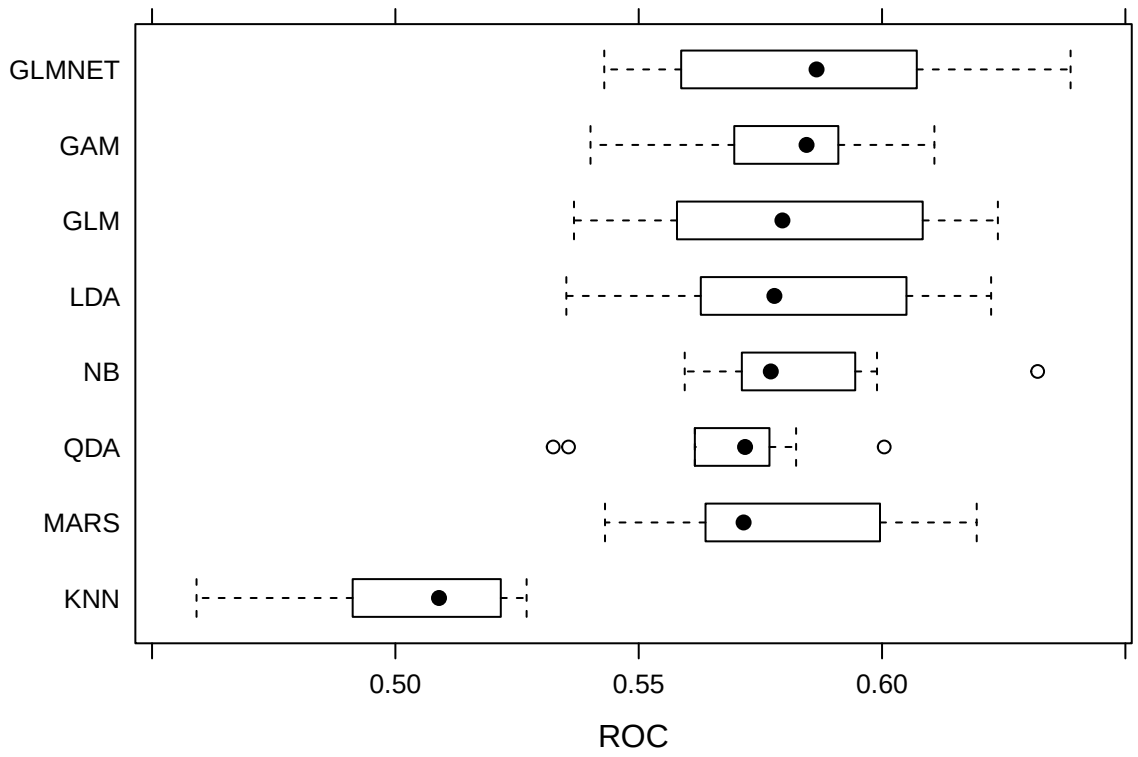
CV Compare

```
res <- resamples(list(GLM = model.glm,
                      GLMNET = model.glmn,
                      GAM = model.gam,
                      MARS = model.mars,
                      LDA = model.lda,
                      QDA = model.qda,
                      NB = model.nb,
                      KNN = model.knn))

#KNN
summary(res)
```

```
##
## Call:
## summary.resamples(object = res)
##
## Models: GLM, GLMNET, GAM, MARS, LDA, QDA, NB, KNN
## Number of resamples: 10
##
## ROC
##             Min.   1st Qu.    Median      Mean   3rd Qu.      Max. NA's
## GLM    0.5367031 0.5601992 0.5795309 0.5819863 0.6057686 0.6238004    0
## GLMNET 0.5429205 0.5624063 0.5865405 0.5872412 0.6059043 0.6387389    0
## GAM    0.5400922 0.5726839 0.5844861 0.5816581 0.5903860 0.6107550    0
## MARS   0.5430707 0.5641508 0.5715484 0.5795346 0.5993159 0.6194519    0
## LDA    0.5351253 0.5645383 0.5778804 0.5822242 0.6033940 0.6224134    0
## QDA    0.5324205 0.5625984 0.5718530 0.5674057 0.5764948 0.6004463    0
## NB     0.5594500 0.5722232 0.5771383 0.5840130 0.5940546 0.6319726    0
## KNN    0.4591393 0.4928278 0.5089623 0.5025626 0.5205611 0.5269624    0
##
## Sens
##             Min.   1st Qu.    Median      Mean   3rd Qu.      Max. NA's
## GLM    1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000    0
## GLMNET 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000    0
## GAM    0.9978632 1.0000000 1.0000000 0.9995726 1.0000000 1.0000000    0
```

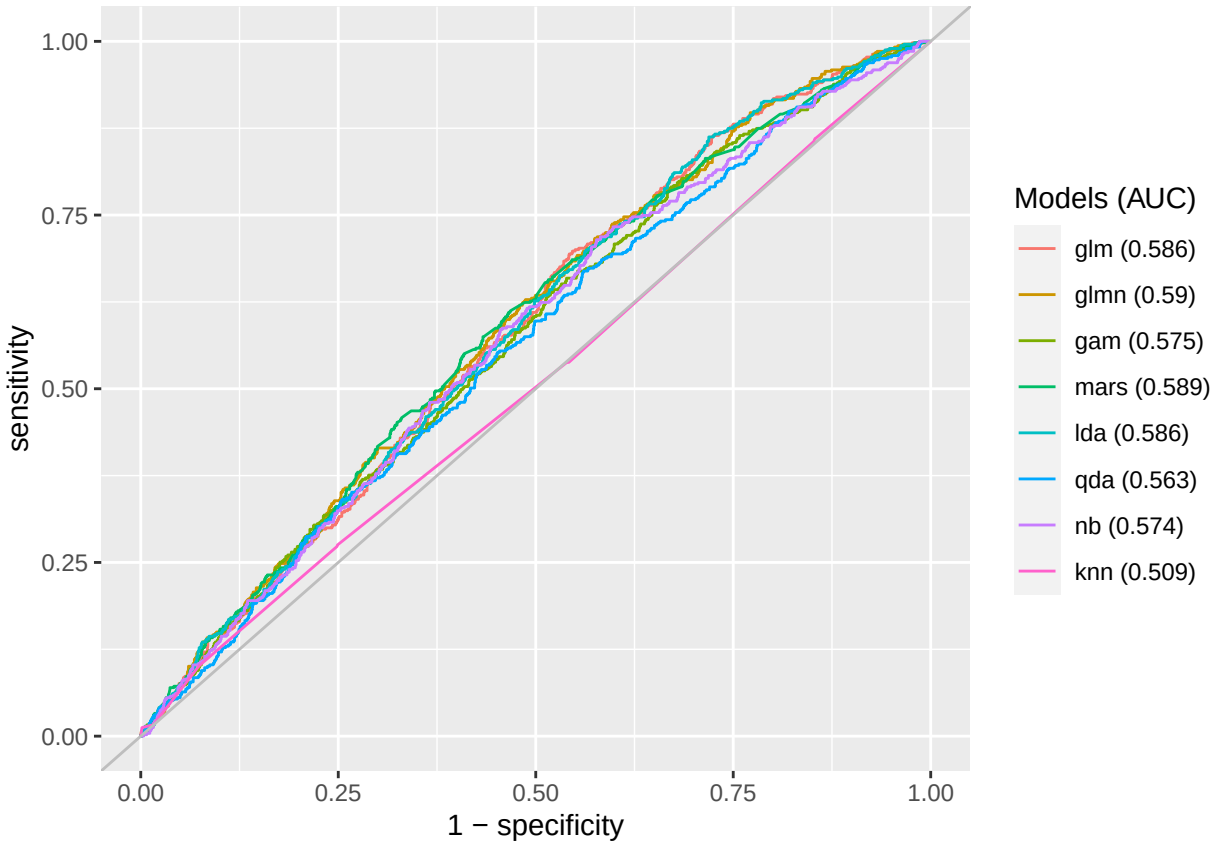
```
## MARS   0.9978632 1.0000000 1.0000000 0.9995726 1.0000000 1.0000000    0
## LDA    1.0000000 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000    0
## QDA    0.9550321 0.9642094 0.9668803 0.9683578 0.9738248 0.9807692    0
## NB     0.9658120 0.9732460 0.9775641 0.9781951 0.9839641 0.9914530    0
## KNN    0.9700855 0.9791564 0.9839744 0.9837552 0.9887821 0.9957265    0
##
## Spec
##              Min.    1st Qu.     Median        Mean    3rd Qu.       Max. NA's
## GLM    0.00000000 0.00000000 0.00000000 0.000000000 0.00000000 0.00000000    0
## GLMNET 0.00000000 0.00000000 0.00000000 0.000000000 0.00000000 0.00000000    0
## GAM    0.00000000 0.00000000 0.00000000 0.000877193 0.00000000 0.00877193    0
## MARS   0.00000000 0.00000000 0.00000000 0.001754386 0.00000000 0.00877193    0
## LDA    0.00000000 0.00000000 0.00000000 0.000000000 0.00000000 0.00000000    0
## QDA    0.00877193 0.02631579 0.03947368 0.034249340 0.04385965 0.04424779    0
## NB     0.00000000 0.02631579 0.03081820 0.028093464 0.03508772 0.04385965    0
## KNN    0.00000000 0.00877193 0.01315789 0.014927806 0.02416162 0.02631579    0
```

```
# figure 4
bwplot(res, metric = "ROC")
```



Test data performance

```
glm.pred <- predict(model.glm, newdata = x2, type = "prob")[,2]
glmn.pred <- predict(model.glmn, newdata = x2, type = "prob")[,2]
gam.pred <- predict(model.gam, newdata = data[-rowTrain,-c(7:8)], type = "prob")[,2]
mars.pred <- predict(model.mars, newdata = x2, type = "prob")[,2]
lda.pred <- predict(model.lda, newdata = x2, type = "prob")[,2]
qda.pred <- predict(model.qda, newdata = x2_limit, type = "prob")[,2]
nb.pred <- predict(model.nb, newdata = data[-rowTrain,-7], type = "prob")[,2]
knn.pred <- predict(model.knn, newdata = x2, type = "prob")[,2]

roc.glm <- roc(y2, glm.pred)
roc.glmn <- roc(y2, glmn.pred)
roc.gam <- roc(y2, gam.pred)
roc.mars <- roc(y2, mars.pred)
roc.lda <- roc(y2, lda.pred)
roc.qda <- roc(y2, qda.pred)
roc.nb <- roc(y2, nb.pred)
roc.knn <- roc(y2, knn.pred)

auc <- c(roc.glm$auc[1], roc.glmn$auc[1], roc.gam$auc[1], roc.mars$auc[1], roc.lda$auc[1], roc.qda$auc[

modelNames <- c("glm","glmn","gam","mars","lda","qda","nb","knn")

# fig 5
ggroc(list(roc.glm, roc.glmn, roc.gam, roc.mars, roc.lda, roc.qda, roc.nb, roc.knn), legacy.axes = TRUE
  scale_color_discrete(labels = paste0(modelNames, " (", round(auc,3),")"),
                       name = "Models (AUC)") +
  geom_abline(intercept = 0, slope = 1, color = "grey")
```



model evaluation

```
gam.pred <- predict(model.gam, newdata = data[-rowTrain,-c(7:8)], type = "prob")[,2]
test.pred <- rep("no_vacc", length(gam.pred))
test.pred[gam.pred>0.5] <- "vacc"

cm = confusionMatrix(data = factor(test.pred),
                reference = y2,
                positive = "vacc")
cm
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction no_vacc vacc
##    no_vacc    2003  487
##    vacc          1    0
##
##                Accuracy : 0.8041
##                  95% CI : (0.788, 0.8195)
##     No Information Rate : 0.8045
##     P-Value [Acc > NIR] : 0.5322
##
##                   Kappa : -8e-04
```

```
##
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.0000000
##             Specificity : 0.9995010
##          Pos Pred Value : 0.0000000
##          Neg Pred Value : 0.8044177
##              Prevalence : 0.1955038
##          Detection Rate : 0.0000000
##    Detection Prevalence : 0.0004014
##       Balanced Accuracy : 0.4997505
##
##        'Positive' Class : vacc
##
```