Problem xy (use separate files for each problem)

10111

11 januar, 2023

```
install.packages("knitr")
install.packages("MASS")
install.packages("caret")
install.packages("pls")
install.packages("glmnet")
install.packages("gam")
install.packages("gbm")
install.packages("randomForest")
install.packages("ggfortify")
install.packages("leaps")
install.packages("pROC")
install.packages("sfsmisc")
```

```
id <- "1HM1ytt-x9QkTHQu7bMvhBJSJWihzpZJ2" # google file ID
d.heart <- read.csv(sprintf("https://docs.google.com/uc?id=%s&export=download", id))
d.heart$HeartDisease <- as.factor(d.heart$HeartDisease)



# 70% of the sample size for training set
training_set_size <- floor(0.70 * nrow(d.heart))

set.seed(4268)
train_ind <- sample(seq_len(nrow(d.heart)), size = training_set_size)

train <- d.heart[train_ind, ]
test <- d.heart[-train_ind, ]
```

a)

```
r.glm <- glm(HeartDisease ~ BMI + Smoking + AlcoholDrinking + Sex + AgeCategory + Smoking:Sex + Alcohol
summary(r.glm)
```

```
##
## Call:
## glm(formula = HeartDisease ~ BMI + Smoking + AlcoholDrinking +
##     Sex + AgeCategory + Smoking:Sex + AlcoholDrinking:Sex, family = "binomial",
##     data = train)
```

```
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.6768  -0.4691  -0.3061  -0.1491   3.5543
##
## Coefficients:
##                              Estimate Std. Error z value Pr(>|z|)
## (Intercept)                 -7.035768   0.523698 -13.435  < 2e-16 ***
## BMI                          0.042950   0.004948   8.681  < 2e-16 ***
## SmokingYes                   0.574912   0.097360   5.905 3.53e-09 ***
## AlcoholDrinkingYes          -0.314465   0.234249  -1.342  0.17945
## SexMale                      0.563997   0.097182   5.803 6.49e-09 ***
## AgeCategory25-29            -0.766922   0.868092  -0.883  0.37699
## AgeCategory30-34             1.161927   0.565991   2.053  0.04008 *
## AgeCategory35-39             0.995359   0.570593   1.744  0.08108 .
## AgeCategory40-44             1.669477   0.537230   3.108  0.00189 **
## AgeCategory45-49             1.651726   0.537050   3.076  0.00210 **
## AgeCategory50-54             2.409280   0.517575   4.655 3.24e-06 ***
## AgeCategory55-59             2.626340   0.513470   5.115 3.14e-07 ***
## AgeCategory60-64             2.880835   0.510277   5.646 1.65e-08 ***
## AgeCategory65-69             3.034768   0.509109   5.961 2.51e-09 ***
## AgeCategory70-74             3.532333   0.507697   6.958 3.46e-12 ***
## AgeCategory75-79             3.788152   0.508763   7.446 9.64e-14 ***
## AgeCategory80 or older       4.185637   0.507548   8.247  < 2e-16 ***
## SmokingYes:SexMale           0.141130   0.130081   1.085  0.27795
## AlcoholDrinkingYes:SexMale   0.001067   0.300323   0.004  0.99717
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 8265.8  on 13999  degrees of freedom
## Residual deviance: 7024.4  on 13981  degrees of freedom
## AIC: 7062.4
##
## Number of Fisher Scoring iterations: 8
```

There is 12 age categories in this dataset

b)

The purpose of this model can be both inference and prediction. We can use it to determine what variables that have a higher chance to cause Heart Disease, but we can also use it to predict the chance that a person given the variables will develop Heart Disease

c)

```
linear <- lda(HeartDisease ~ ., train)
summary(linear)
```

```
##         Length Class  Mode
```

```
## prior    2     -none- numeric
## counts   2     -none- numeric
## means   68     -none- numeric
## scaling 34     -none- numeric
## lev      2     -none- character
## svd      1     -none- numeric
## N        1     -none- numeric
## call     3     -none- call
## terms    3     terms  call
## xlevels 12     -none- list
```

```
quad <- qda(HeartDisease ~ ., train)
summary(quad)
```

```
##         Length Class  Mode
## prior      2   -none- numeric
## counts     2   -none- numeric
## means     68   -none- numeric
## scaling 2312   -none- numeric
## ldet       2   -none- numeric
## lev        2   -none- character
## N          1   -none- numeric
## call       3   -none- call
## terms      3   terms  call
## xlevels   12   -none- list
```

```
predictedlin <- predict(linear, test, type="response")
#auc(test$HeartDisease, predictedlin)

predictedqua <- predict(quad, test, type="response")
#auc(test$HeartDisease, predictedqua)
```

KNN classification probably wont work well for this task because KNN doesn't work well with large datasets and it doesn't work well with a high number of dimensions. The training dataset have 14000 observations with 17 variables. That is a LOT of data.

d)

Not enought time to train trandomforest to find optimal number of tree, so i will use 1000 trees. I choose $mtry = p/3$ number of trees where p is the number of predictors because that is the default for regression trees.

```
bag.HeartDisease <- randomForest(HeartDisease ~ ., data = train, mtry = 4, ntree = 1000)
bag.HeartDisease
```

```
##
## Call:
##  randomForest(formula = HeartDisease ~ ., data = train, mtry = 4,      ntree = 1000)
##                Type of random forest: classification
##                      Number of trees: 1000
## No. of variables tried at each split: 4
```

```
##
##         OOB estimate of  error rate: 8.67%
## Confusion matrix:
##        No Yes class.error
## No  12698  86 0.006727159
## Yes  1128  88 0.927631579
```

```
randompred <- predict(bag.HeartDisease, newdata = test)
# randompred
```