Building a Prediction Model using ML techniques: GLM, GAM, Decision Tree, Neural Networks, SVM, Random Forests, Bagging and Boosting

Professor: Dr. Tony Desmond

Student: Mohsen Selseleh

Student No:

1207477

Fall 2021

https://rpubs.com/Mohsen_Selseleh/848715

Dataset: dvisits (Doctor visits in Australia)

The data come from the Australian Health Survey of 1977-78 and consists of 5190 single adults at 19 variables.

Features	Description
sex	1 if female, 0 if male
age	Age in years divided by 100
agesq	age squared
income	Annual income in Australian dollars divided by 1000
levyplus	1 if covered by private health insurance fund for private patient in public hospital (with doctor of choice), 0 otherwise
freepoor	1 if covered by government because low income, recent immigrant, unemployed, 0 otherwise
freerepa	1 if covered free by government because of old-age or disability pension, or because invalid veteran or family of deceased veteran, 0 otherwise
illness	Number of illnesses in past 2 weeks with 5 or more coded as 5
actdays	Number of days of reduced activity in past two weeks due to illness or injury
hscore	General health questionnaire score using Goldberg's method. High score indicates bad health
chcond1	1 if chronic condition(s) but not limited in activity, 0 otherwise
chcond2	1 if chronic condition(s) and limited in activity, 0 otherwise
doctorco	Number of consultations with a doctor or specialist in the past 2 weeks
nondocco	Number of consultations with non-doctor health professionals (chemist, optician, physiotherapist, social worker, district community nurse, chiropodist or chiropractor) in the past 2 weeks
hospadmi	Number of admissions to a hospital, psychiatric hospital, nursing or convalescent home in the past 12 months (up to 5 or more admissions which is coded as 5)
hospdays	Number of nights in a hospital, etc. during most recent admission: taken, where appropriate, as the mid-point of the intervals 1, 2, 3, 4, 5, 6, 7, 8-14, 15-30, 31-60, 61-79 with 80 or more admissions coded as 80. If no admission in past 12 months then equals zero
medicine	Total number of prescribed and nonprescribed medications used in past 2 days
prescrib	Total number of prescribed medications used in past 2 days
nonpresc	Total number of nonprescribed medications used in past 2 days

Descriptve statistics

```
 sex               age               agesq             income            levyplus          freepoor
 Min.   :0.0000   Min.   :0.1900   Min.   :0.0361   Min.   :0.0000   Min.   :0.0000   Min.   :0.00000
 1st Qu.:0.0000   1st Qu.:0.2200   1st Qu.:0.0484   1st Qu.:0.2500   1st Qu.:0.0000   1st Qu.:0.00000
 Median :1.0000   Median :0.3200   Median :0.1024   Median :0.5500   Median :0.0000   Median :0.00000
 Mean   :0.5206   Mean   :0.4064   Mean   :0.2071   Mean   :0.5832   Mean   :0.4428   Mean   :0.04277
 3rd Qu.:1.0000   3rd Qu.:0.6200   3rd Qu.:0.3844   3rd Qu.:0.9000   3rd Qu.:1.0000   3rd Qu.:0.00000
 Max.   :1.0000   Max.   :0.7200   Max.   :0.5184   Max.   :1.5000   Max.   :1.0000   Max.   :1.00000
 freerepa         illness          actdays           hscore           chcond1          chcond2
 Min.   :0.0000   Min.   :0.000   Min.   : 0.0000   Min.   : 0.000   Min.   :0.0000   Min.   :0.0000
 1st Qu.:0.0000   1st Qu.:0.000   1st Qu.: 0.0000   1st Qu.: 0.000   1st Qu.:0.0000   1st Qu.:0.0000
 Median :0.0000   Median :1.000   Median : 0.0000   Median : 0.000   Median :0.0000   Median :0.00000
 Mean   :0.2102   Mean   :1.432   Mean   : 0.8619   Mean   : 1.218   Mean   :0.4031   Mean   :0.1166
 3rd Qu.:0.0000   3rd Qu.:2.000   3rd Qu.: 0.0000   3rd Qu.: 2.000   3rd Qu.:1.0000   3rd Qu.:0.0000
 Max.   :1.0000   Max.   :5.000   Max.   :14.0000   Max.   :12.000   Max.   :1.0000   Max.   :1.0000
doctorco          nondocco          hospadmi          hospdays          medicine         prescrib
 Min.   :0.0000   Min.   : 0.0000   Min.   :0.0000   Min.   : 0.000   Min.   :0.000   Min.   :0.0000
 1st Qu.:0.0000   1st Qu.: 0.0000   1st Qu.:0.0000   1st Qu.: 0.000   1st Qu.:0.000   1st Qu.:0.0000
 Median :0.0000   Median : 0.0000   Median :0.0000   Median : 0.000   Median :1.000   Median :0.0000
 Mean   :0.3017   Mean   : 0.2146   Mean   :0.1736   Mean   : 1.334   Mean   :1.218   Mean   :0.8626
 3rd Qu.:0.0000   3rd Qu.: 0.0000   3rd Qu.:0.0000   3rd Qu.: 0.000   3rd Qu.:2.000   3rd Qu.:1.0000
 Max.   :9.0000   Max.   :11.0000   Max.   :5.0000   Max.   :80.000   Max.   :8.000   Max.   :8.0000
 nonpresc
 Min.   :0.0000
 1st Qu.:0.0000
 Median :0.0000
 Mean   :0.3557
 3rd Qu.:1.0000
 Max.   :8.0000
```

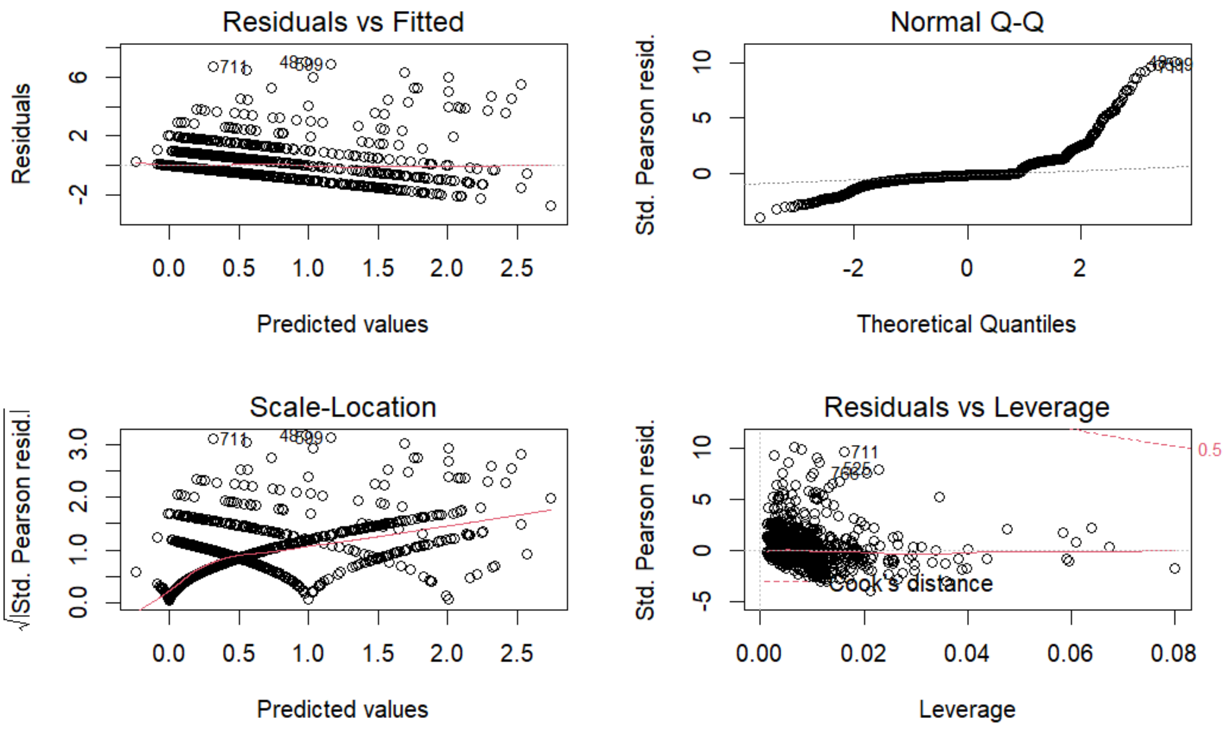
Split dataset to train and test datasets

```
#Set Seed

set.seed(10857825)

#Training and Testing Data

subset2 = sample(nrow(dvisits), nrow(dvisits) * 0.75)

dvisits.train2 = dvisits[subset2, ]

dvsits.test2 = dvisits[-subset2, ]

#GLM

set.seed(10857825)

glmmodel=glm(doctorco~., data=dvisits.train2)

summary(glmmodel)
```

```
Call:

glm(formula = doctorco ~ ., data = dvisits.train2)


Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.7455  -0.2376  -0.1116  -0.0485   7.0186
Coefficients: (1 not defined because of singularities)
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.074615   0.082395   0.906  0.36522
```

```
sex          0.017970   0.024753   0.726  0.46790
age          0.143679   0.467794   0.307  0.75875
agesq       -0.172310   0.525018  -0.328  0.74278
income      -0.038144   0.037582  -1.015  0.31019
levyplus     0.006933   0.028176   0.246  0.80566
freepoor    -0.128056   0.059937  -2.137  0.03270 *
freerepa     0.000394   0.042953   0.009  0.99268
illness      0.030791   0.009916   3.105  0.00192 **
actdays      0.090115   0.004275  21.077  < 2e-16 ***
hscore       0.013844   0.005791   2.391  0.01687 *
chcond1     -0.031700   0.027194  -1.166  0.24381
chcond2     -0.058487   0.041421  -1.412  0.15803
nondocco    -0.004674   0.012169  -0.384  0.70097
hospadmi     0.143776   0.026920   5.341 9.78e-08 ***
hospdays     0.002033   0.002219   0.916  0.35967
medicine    -0.027149   0.015925  -1.705  0.08831 .
prescrib     0.120608   0.017754   6.793 1.26e-11 ***
nonpresc           NA         NA      NA       NA
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
(Dispersion parameter for gaussian family taken to be 0.4885719)
    Null deviance: 2428.3  on 3891  degrees of freedom
Residual deviance: 1892.7  on 3874  degrees of freedom
AIC: 8277.3
Number of Fisher Scoring iterations: 2
extractAIC(glmmodel)
18.000 8277.258
```

```
extractAIC(glmmodel, k=log(nrow(dvisits.train2)))
18.000 8390.058
#Prediction with training data
pi1_IS = predict(object = glmmodel, newdata = dvisits.train2)
mean((pi1_IS-dvisits.train2$doctorco)^2)
0.4863123
#Prediction with testing data
pi1_OS = predict(object = glmmodel, newdata = dvisits.test2)
mean((pi1_OS-dvisits.test2$doctorco)^2)
0.4864957
library(modelr)
data.frame(
  R2 = rsquare(glmmodel, data = dvisits),
  RMSE = rmse(glmmodel, data = dvisits),
  MAE = mae(glmmodel, data = dvisits)
)
R2 :    0.2363743
RMSE :  0.6973938
MAE :    0.380663

#Residual Plots
par(mfrow=c(2,2))
plot(glmmodel)
```

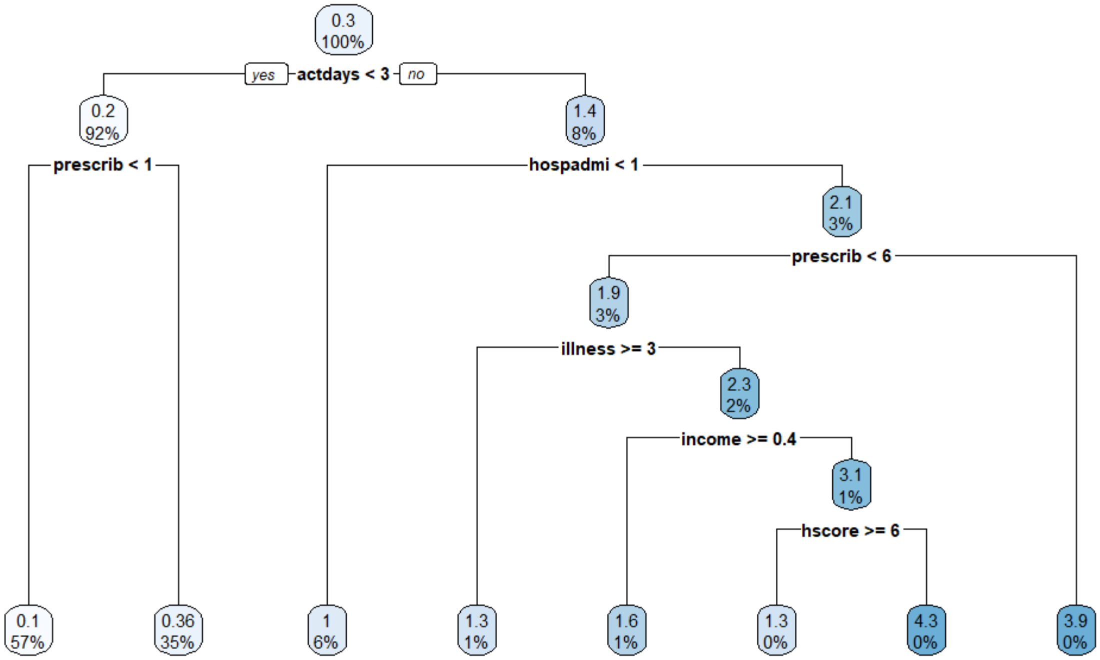


```
# cross-validation

set.seed(10857825)

library(boot)

fullmodel = glm(doctorco ~ ., data=dvisits)

cvmodel2<-cv.glm(data = dvisits,glmfit = fullmodel, K = 3)

cvmodel2$delta[2]

0.4931256
```

# Trees

```
#Regression Trees

install.packages("rpart")

library(rpart.plot)

library(rpart) #Tree modeling

dvisits.rpart <- rpart(formula = doctorco ~ ., data = dvisits.train2)
```

```
rpart.plot(dvisits.rpart)
```



```
#MSE FOR TREE
dvisits.train.pred.tree = predict(dvisits.rpart, dvisits.train2)
mean((dvisits.train.pred.tree - dvisits.train2$doctorco)^2)
0.4417439
library(modelr)
data.frame(
  R2 = rsquare(dvisits.rpart, data = dvisits),
  RMSE = rmse(dvisits.rpart, data = dvisits),
  MAE = mae(dvisits.rpart, data = dvisits))
R2: 0.2666172
RMSE: 0.6834386
MAE: 0.3811555
```

```
#Cross-Validation for TREE

library(caret)

b <- train(doctorco ~ sex + age + agesq + income + levyplus + freepoor + freerepa
+ illness + actdays + hscore + chcond1 + chcond2,data=dvisits,

        method = "gam",

        trControl = trainControl(method = "cv", number = 10),

        tuneGrid = data.frame(method = "GCV.Cp", select = FALSE)

)

mse<-b$results[3]^2

mse

RMSE

0.5757068

summary(b$finalModel)

Family: gaussian

Link function: identity

Formula:

.outcome ~ sex + levyplus + freerepa + chcond1 + chcond2 + illness +

    s(age) + s(agesq) + s(hscore) + s(income)

Parametric coefficients:

            Estimate Std. Error t value Pr(>|t|)

(Intercept) 0.121119   0.025639   4.724 2.37e-06 ***

sex         0.035171   0.023204   1.516    0.130

levyplus    0.037330   0.026097   1.430    0.153

freerepa    0.033630   0.040284   0.835    0.404

chcond1     0.003365   0.025506   0.132    0.895
```

```
chcond2     0.179939   0.038154   4.716 2.47e-06 ***
illness     0.081268   0.008942   9.088  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
Approximate significance of smooth terms:
          edf Ref.df      F p-value
s(age)      1      1  0.013   0.909
s(agesq)    1      1  0.210   0.646
s(hscore)   1      1 74.539  <2e-16 ***
s(income)   1      1  0.710   0.400
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
R-sq.(adj) =  0.0765   Deviance explained = 7.83%
GCV = 0.58952  Scale est. = 0.58827   n = 5190
```

```
#GAM
require(gam)
doctor_gam1 <- gam(doctorco ~ sex + age  + income + freepoor + freerepa +
illness + actdays + hscore , family=poisson, data = dvisits)
doctor_gam1
summary(doctor_gam1)
AIC(doctor_gam1)
```

```
Call: gam(formula = doctorco ~ sex + age + income + freepoor + freerepa +
    illness + actdays + hscore, family = poisson, data = dvisits)
Deviance Residuals:
    Min      1Q  Median      3Q     Max
-2.9423 -0.6818 -0.5757 -0.4942  5.6966
```

```
(Dispersion Parameter for poisson family taken to be 1)
    Null Deviance: 5634.821 on 5189 degrees of freedom
Residual Deviance: 4388.056 on 5181 degrees of freedom
AIC: 6737.623
Number of Local Scoring Iterations: 5
```

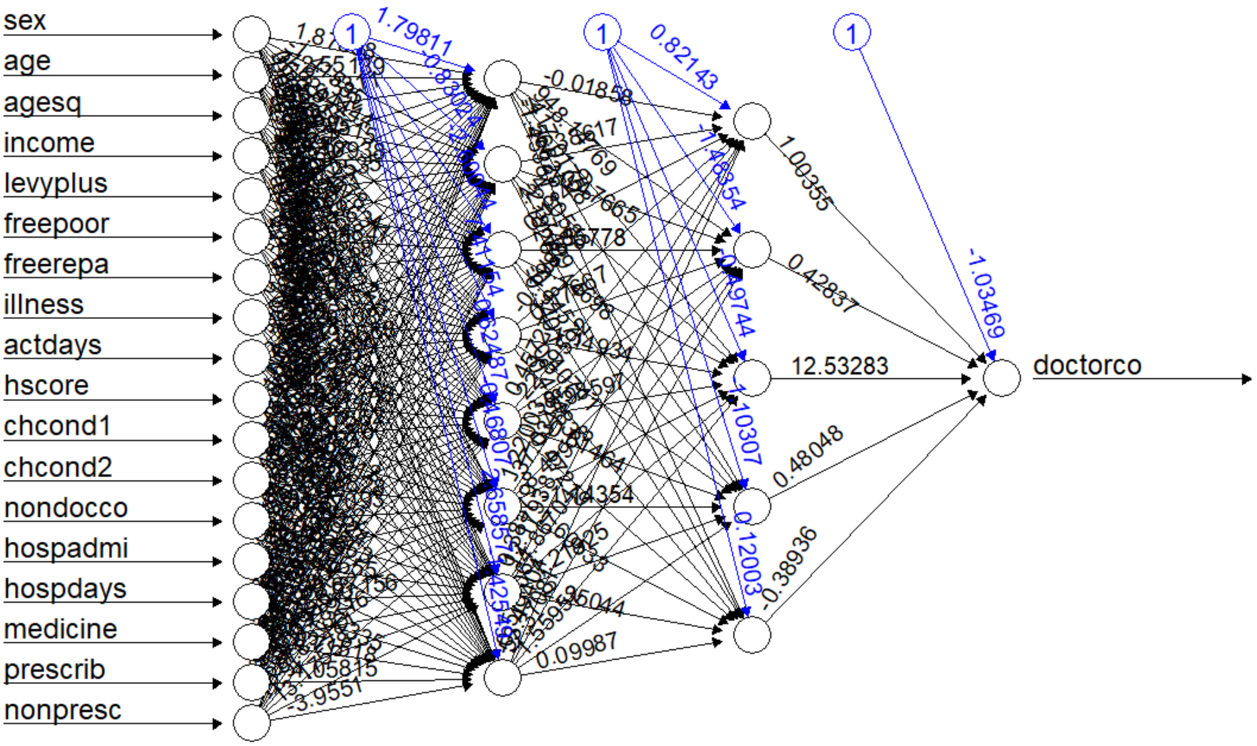
```
library(modelr)
data.frame(
  R2 = rsquare(doctor_gam1, data = dvisits),
  RMSE = rmse(doctor_gam1, data = dvisits),
  MAE = mae(doctor_gam1, data = dvisits)
)
```

```
R2: 0.06919112
RMSE: 1.936503
MAE: 1.793583
```

```
Anova for Parametric Effects
            Df Sum Sq Mean Sq  F value    Pr(>F)
sex          1   80.4   80.37  60.8437  7.42e-15 ***
age          1  224.0  223.99 169.5757 < 2.2e-16 ***
income       1   18.0   18.03  13.6492 0.0002226 ***
freepoor     1    8.9    8.88   6.7237 0.0095409 **
freerepa     1    6.9    6.90   5.2232 0.0223274 *
illness      1  469.1  469.11 355.1524 < 2.2e-16 ***
actdays      1  857.1  857.10 648.8860 < 2.2e-16 ***
hscore       1   10.7   10.69   8.0939 0.0044588 **
```

```
Residuals 5181 6843.4    1.32
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
[1] 6737.624
BIC(doctor_gam1)
6796.614
#In-sample performance
mean(residuals(doctor_gam1)^2) #In-Sample
0.8454828
pi1 = predict(object = doctor_gam1, newdata = dvisits.test2)
#Out of Sample performance
mean((pi1-dvisits.test2$doctorco)^2) #Out of Sample
3.804385
#Neural Networks
#Scaling Inputs- To get a range from 0-1
require(neuralnet)
maxs <- apply(dvisits, 2, max)
mins <- apply(dvisits, 2, min)
scaled <- as.data.frame(scale(dvisits, center = mins, scale = maxs - mins))
set.seed(10857825)
subset2 = sample(nrow(dvisits), nrow(dvisits) * 0.75)
dvisits.train2 = scaled[subset2, ]
dvisits.test2 = scaled[-subset2, ]
n <- names(dvisits.train2)
f <- as.formula(paste("doctorco ~", paste(n[!n %in% "doctorco"], collapse = " + ")))
```

~~~
nnetmodel <- neuralnet(f, data=dvisits.train2,hidden=c(8,5), linear.output = TRUE)

plot(nnetmodel)

#Out of Sample

pr.nn<- neuralnet::compute(nnetmodel, dvisits.test2[,1:13])

#Scaling back to get a prediction

pr.nn_ <- pr.nn$net.result*(max(dvisits$doctorco)-
min(dvisits$doctorco))+min(dvisits$doctorco)

test.r <- (dvisits.test2$doctorco)*(max(dvisits$doctorco)-
min(dvisits$doctorco))+min(dvisits$doctorco)

sum((test.r - pr.nn_)^2)/nrow(dvisits.test2)

```
~~~



~~~
## Prediction using neural network

Predict=compute(nnetmodel,dvisits.test2)

mean((Predict$net.result-dvisits.test2$doctorco)^2)
~~~

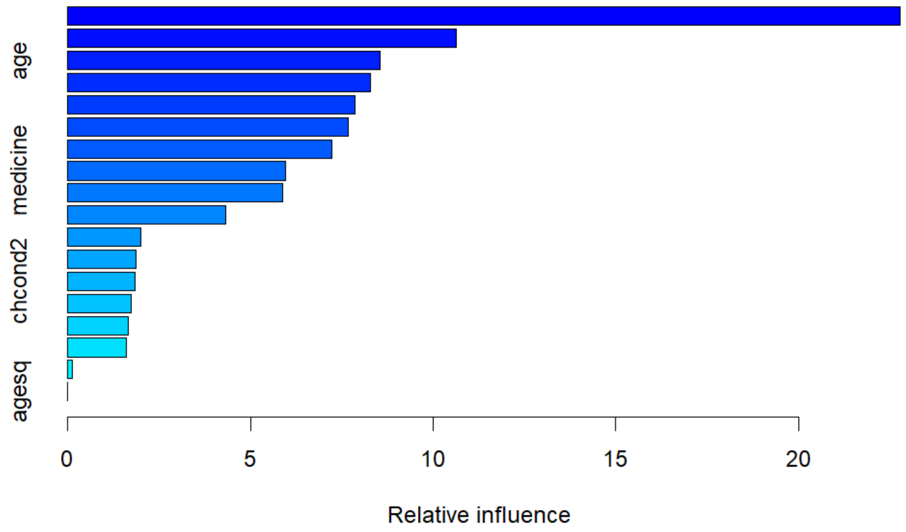
```
0.1624712
library(modelr)
data.frame(
  R2 = rsquare(nnetmodel, data = dvisits),
  RMSE = rmse(nnetmodel, data = dvisits),
  MAE = mae(nnetmodel, data = dvisits)
)
R2: -1.352226
RMSE: 1.227864
MAE: 0.4271425
#Support Vector Machines
library(e1071)
svmmodel<-svm(doctorco~., dvisits.train2)
#MSE- In-Sample
mean(residuals(svmmodel)^2)
predsvm<- predict(svmmodel, dvisits.test2)
#MSE- Out-of-Sample
mean((predsvm-dvisits.test2$doctorco)^2)
[1] 0.005431535
[1] 0.007400845
library(modelr)
data.frame(
  R2 = rsquare(svmmodel, data = dvisits),
  RMSE = rmse(svmmodel, data = dvisits),
  MAE = mae(svmmodel, data = dvisits)
```

```
)
R2: 0.007738408
RMSE: 0.8358721
MAE: 0.3242822
#Bagging
library(randomForest)
set.seed(10857825)
subset2 = sample(nrow(dvisits), nrow(dvisits) * 0.75)
dvisits.train2 = dvisits[subset2, ]
dvisits.test2 = dvisits[-subset2, ]
set.seed(10857825)
bag.dvisits=randomForest(doctorco~.,data=dvisits.train2,mtry=13,importance=TRUE)
bag.dvisits
Call:
 randomForest(formula = doctorco ~ ., data = dvisits.train2, mtry = 13,
importance = TRUE)
               Type of random forest: regression
                     Number of trees: 500
No. of variables tried at each split: 13

          Mean of squared residuals: 0.5346554
                    % Var explained: 14.31
library(modelr)
data.frame(
  R2 = rsquare(bag.dvisits, data = dvisits),
  RMSE = rmse(bag.dvisits, data = dvisits),
  MAE = mae(bag.dvisits, data = dvisits)
)
R2: 0.6401244
RMSE: 0.4790284
MAE: 0.2515578
#MSE FOR SVM
pred.bag = predict(bag.dvisits,newdata=dvisits.test2)
mean((pred.bag-dvisits.test2$doctorco)^2)
0.504
```

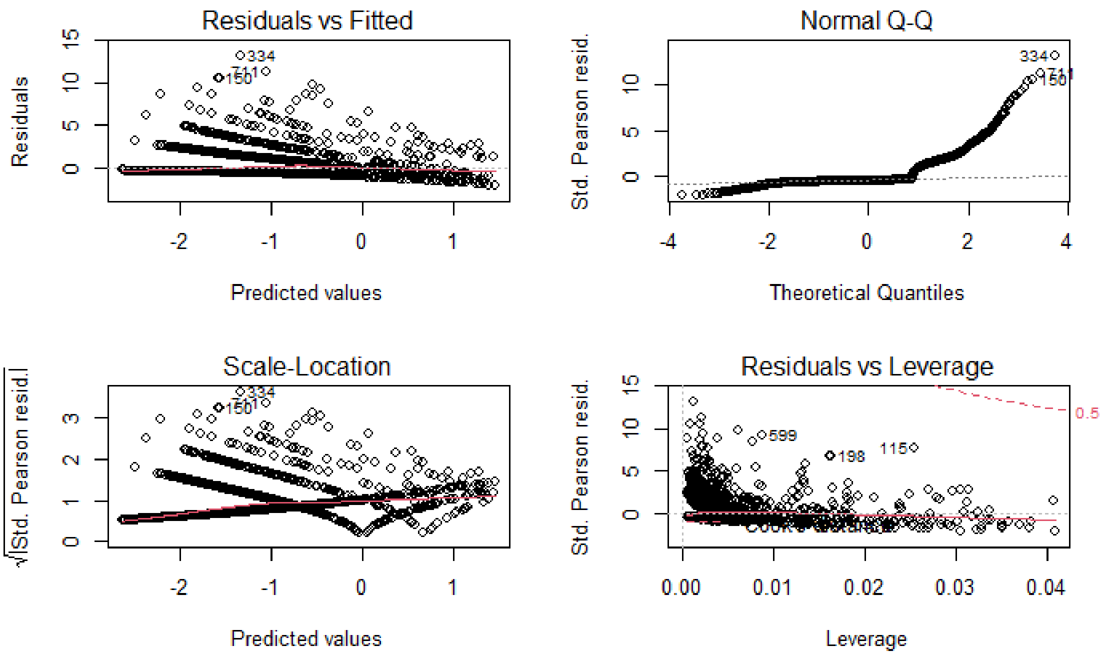
#Random Forests

```
rf.dvisits=randomForest(doctorco~.,data=dvisits.train2,mtry=6,importance=TRUE)
rf.dvisits
Call:
 randomForest(formula = doctorco ~ ., data = dvisits.train2, mtry = 6, 
importance = TRUE)
               Type of random forest: regression
                     Number of trees: 500
No. of variables tried at each split: 6

          Mean of squared residuals: 0.512795
                    % Var explained: 17.81
#MSE Random Forest
pred.rf = predict(rf.dvisits,newdata=dvisits.test2)
mean((pred.rf-dvisits.test2$doctorco)^2)
0.4976903
```

```
library(modelr)
data.frame(
  R2 = rsquare(rf.dvisits, data = dvisits),
  RMSE = rmse(rf.dvisits, data = dvisits),
  MAE = mae(rf.dvisits, data = dvisits)
)
R2: 0.6276496
RMSE: 0.4870887
MAE: 0.2589066
#Boosting
install.packages("gbm")
library(gbm)
set.seed(10857825)
boost.dvisits=gbm(doctorco~.,data=dvisits.train2,distribution="gaussian",n.trees=5000,interaction.depth=4)
summary(boost.dvisits)
```



```
#MSE BOOSTING
pred.boost=predict(boost.dvisits,newdata=dvisits.test2,n.trees=5000)
mean((pred.boost-dvisits.test2$doctorco)^2)
0.6289

library(modelr)
data.frame(
  R2 = rsquare(boost.dvisits, data = dvisits),
  RMSE = rmse(boost.dvisits, data = dvisits),
  MAE = mae(boost.dvisits, data = dvisits)
)
R2: 0.6136155
RMSE: 0.4961005
MAE: 0.2677312
#GLM WITH POISSON
doctor_poisson <- glm(doctorco ~ sex + age + agesq + income + levyplus +
freepoor + freerepa + illness + actdays + hscore + chcond1 + chcond2,
family=poisson, data = dvisits)
doctor_poisson
summary(doctor_poisson)
par(mfrow=c(2,2))
plot(doctor_poisson)
step(doctor_poisson, direction="backward")
predict(doctor_poisson, dvisits[5190,], type="response")
print(paste0("Probability of 0 doctor's visits: ", dpois(0, lambda = 0.153)))
print(paste0("Probability of 1 doctor's visits: ", dpois(1, lambda = 0.153)))
print(paste0("Probability of 2 doctor's visits: ", dpois(2, lambda = 0.153)))

print(paste0("Probability of 3 doctor's visits: ", dpois(3, lambda = 0.153)))

print(paste0("Probability of 4 doctor's visits: ", dpois(4, lambda = 0.153)))

print(paste0("Probability of 5 doctor's visits: ", dpois(5, lambda = 0.153)))

round(dpois(0:5, 0.153),3)

doctor_lm <- lm(doctorco ~ sex + age + agesq + income + levyplus + freepoor +
freerepa + illness + actdays + hscore + chcond1 + chcond2, data=dvisits)

doctor_lm

summary(doctor_lm)

predict(doctor_lm, dvisits[5190,])
```



```
# GAM WITH POISSON

#requiring the Package

require(gam)
```

```
doctor_gam <- gam(doctorco ~ sex + age + agesq + income + levyplus + freepoor
+ freerepa + illness + actdays + hscore + chcond1 + chcond2, family=poisson, data
= dvisits)

doctor_gam

summary(doctor_gam)

doctor_gam1 <- gam(doctorco ~ sex + age  + income + freepoor + freerepa +
illness + actdays + hscore , family=poisson, data = dvisits)

doctor_gam1

summary(doctor_gam1)
```

all:

```
glm(formula = doctorco ~ sex + age + agesq + income + levyplus +
    freepoor + freerepa + illness + actdays + hscore + chcond1 +
    chcond2, data = dvisits)


Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.1352  -0.2588  -0.1435  -0.0433   7.0327


Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  0.027632   0.072220   0.383  0.70202
sex          0.033811   0.021604   1.565  0.11764
age          0.203201   0.410016   0.496  0.62020
agesq       -0.062103   0.458716  -0.135  0.89231
income      -0.057323   0.033089  -1.732  0.08326 .
levyplus     0.035179   0.024882   1.414  0.15748
```

```
freepoor    -0.103314   0.052471  -1.969  0.04901 *
freerepa     0.033241   0.038157   0.871  0.38371
illness      0.059946   0.008357   7.173 8.39e-13 ***
actdays      0.103192   0.003657  28.216  < 2e-16 ***
hscore       0.016976   0.005190   3.271  0.00108 **
chcond1      0.004384   0.023740   0.185  0.85349
chcond2      0.041617   0.035863   1.160  0.24592
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
(Dispersion parameter for gaussian family taken to be 0.5096272)
    Null deviance: 3305.5  on 5189  degrees of freedom
Residual deviance: 2638.3  on 5177  degrees of freedom
AIC: 11245
Number of Fisher Scoring iterations: 2
predict(doctor_gam1, dvisits[5190,], type="response")
print(paste0("Probability of 0 doctor's visits: ", dpois(0, lambda = 0.167)))
print(paste0("Probability of 1 doctor's visits: ", dpois(1, lambda = 0.167)))
13.00 11326.58
print(paste0("Probability of 2 doctor's visits: ", dpois(2, lambda = 0.167)))
print(paste0("Probability of 3 doctor's visits: ", dpois(3, lambda = 0.167)))
print(paste0("Probability of 4 doctor's visits: ", dpois(4, lambda = 0.167)))
print(paste0("Probability of 5 doctor's visits: ", dpois(5, lambda = 0.167)))
13.00 11326.58
round(dpois(0:5, 0.153),3)
Family: poisson
```

```
Link function: log
Formula:
doctorco ~ sex + age + agesq + income + levyplus + freepoor +
    freerepa + illness + actdays + hscore + chcond1 + chcond2
Total model degrees of freedom 13
UBRE score: -0.1511532
Family: poisson
Link function: log
Formula:
doctorco ~ sex + age + agesq + income + levyplus + freepoor +
    freerepa + illness + actdays + hscore + chcond1 + chcond2
Parametric coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -2.223848   0.189816 -11.716   <2e-16 ***
sex          0.156882   0.056137   2.795   0.0052 **
age          1.056299   1.000780   1.055   0.2912
agesq       -0.848704   1.077784  -0.787   0.4310
income      -0.205321   0.088379  -2.323   0.0202 *
levyplus     0.123185   0.071640   1.720   0.0855 .
freepoor    -0.440061   0.179811  -2.447   0.0144 *
freerepa     0.079798   0.092060   0.867   0.3860
illness      0.186948   0.018281  10.227   <2e-16 ***
actdays      0.126846   0.005034  25.198   <2e-16 ***
hscore       0.030081   0.010099   2.979   0.0029 **
chcond1      0.114085   0.066640   1.712   0.0869 .
```

```
chcond2      0.141158   0.083145   1.698   0.0896 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1



R-sq.(adj) =  0.155   Deviance explained = 22.3%
UBRE = -0.15115  Scale est. = 1         n = 5190
Family: poisson
Link function: log
Formula:
doctorco ~ sex + age + income + freepoor + freerepa + illness +
    actdays + hscore
Total model degrees of freedom 9
UBRE score: -0.151049
Family: poisson
Link function: log
Formula:
doctorco ~ sex + age + income + freepoor + freerepa + illness +
    actdays + hscore
Parametric coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -2.052299   0.099492 -20.628  < 2e-16 ***
sex          0.175856   0.055451   3.171  0.00152 **
age          0.448185   0.153571   2.918  0.00352 **
income      -0.176135   0.085364  -2.063  0.03908 *
freepoor    -0.499016   0.175758  -2.839  0.00452 **
```

```
freerepa    -0.014633   0.069268  -0.211  0.83270
illness      0.196105   0.017592  11.147  < 2e-16 ***
actdays      0.127813   0.004899  26.087  < 2e-16 ***
hscore       0.032566   0.009961   3.269  0.00108 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
R-sq.(adj) =  0.151   Deviance explained = 22.1%
UBRE = -0.15105  Scale est. = 1         n = 5190
```

The model GAM is best then the GLM becaues the GAM results hows high probability as compared to the other models

Conclusion

Models	R-Squared	RMSE	MAE
GLM	24	0.69	0.38
GAM	6.9	1.93	1.79
NN	NA	1.23	0.43
SVM	0.0077	0.83	0.32
BAGGING	0.64	0.47	0.25
RF	0.62	0.48	0.26
BOOSTING	0.61	0.5	.27
DT	26.67	.68	.38

Whereas R-squared is a relative measure of fit, RMSE is an absolute measure of fit. Lower values of RMSE indicate better fit. RMSE is a good measure of how accurately the model predicts the response, and it is the most important criterion for fit if the main purpose of the model is prediction. Meanwhile MAE, Lower values are better.

I see that all of them had good performance. But BAGGING, RF, BOOSTING outperformed than other models and they have lower RMSE and MAE. Therefore, I could result that ensemble models were better than other classification models in divisits dataset.