The combination of Artificial Neural network + Feature selection

R Markdown

```r
#Kdd3 is the kidney disease dataset after replacing outliers by interuitile

setwd("C:/Users/Administrator/Documents/archive (3)")
ktd <- read.csv("kdd3.csv",header = TRUE )


#ktd<-na.omit(ktd)
#ktd
nrow(ktd)
```

```
## [1] 397
```

```r
#Ktd dataset is used as input for Elsatic net
ktd<-na.omit(ktd)


#dataset for neural network
kdn<-ktd

#view(kdn)
```

```r
set.seed(123)
training.samples <- ktd$classification %>%createDataPartition(p = 0.75, list = FALSE)
data.train <- ktd[training.samples, ]
data.test<- ktd[-training.samples, ]

#view(data.test)

#nrow(ktd)
```

```r
# Create the matrix of predictors for glmnet function
x <- model.matrix(classification~., data.train)[,-25]

# Convert the outcome (class) to a numerical variable


y<-data.train[25] <- ifelse(data.train$classification == "ckd", 1, 0)
```

```r
#summary(data.train)
```

```
#cross validation to find optimal lambda and alpha for Elastic net

set.seed(123)
cv_10 = trainControl(method = "cv", number = 10)
# Train the model


elasticm <- train(as.factor(classification) ~ .,
             method     = "glmnet",

             trControl  = cv_10,

             data= data.train,
            # family= binomial()
            tuneLength=10

            )

elasticm$bestTune
```

```
##   alpha    lambda
## 5   0.1 0.0120223
```

```
coef(elasticm$finalModel, elasticm$bestTune$lambda)
```

```
## 35 x 1 sparse Matrix of class "dgCMatrix"
##                          s1
## (Intercept)    1.113149e+02
## age            1.015628e-02
## bp             2.477300e-02
## sg            -1.019520e+02
## al             3.926642e-01
## su             1.524578e-01
## rbcabnormal    6.925580e-01
## rbcnormal     -2.144371e+00
## pcabnormal     1.123432e+00
## pcnormal      -4.946454e-01
## pccnotpresent  .
## pccpresent     4.528610e-01
## banotpresent   7.044754e-02
## bapresent      .
## bgr            1.075798e-02
## bu             1.485262e-03
## sc             3.104330e-01
## sod           -3.021896e-02
## pot           -1.043897e-01
## hemo          -2.040754e-01
## pcv           -5.838881e-02
## wc             1.611172e-04
## rc            -4.125646e-01
## htnno         -5.558346e-01
## htnyes         7.247176e-01
## dmno          -8.639834e-01
## dmyes          1.048595e+00
## cadno          .
## cadyes         .
## appetgood      .
## appetpoor      .
## peno           .
## peyes          .
## aneno         -1.030100e-01
## aneyes         2.529575e-01
```

```
#the result shows that cadno ,cadyes,appetgood, appetpoor,peno, peyes is removed and we remove them fro
```

```
#bemodel <- glmnet(x, y, alpha = 0.1, lambda = 0.0120223    ,family="binomial")
```

```
setwd("C:/Users/Administrator/Documents/archive (3)")
kdn <- read.csv("kd-cor-el.csv",header = TRUE )

"kdn includes variables that elastic net model has selected "
```

```
## [1] "kdn includes variables that elastic net model has selected "
```

```
kdn<-na.omit(kdn)
#view(kdn)
#nrow(kdn)
```

```
#prepare dataset for applying neuralnetwork

kdn[9]<- ifelse(kdn$ba == "notpresent", 1, 0)
kdn[19]<- ifelse(kdn$dm == "yes", 1, 0)
#kdn[21] <- ifelse(kdn$cad == "yes", 1, 0)

kdn[6] <- ifelse(kdn$rbc == "normal", 1, 0)
kdn[7] <- ifelse(kdn$pc == "normal", 0, 1)

kdn[8] <- ifelse(kdn$pcc == "present", 1, 0)

#kdn[22] <- ifelse(kdn$appet == "good", 1, 0)
#kdn[23] <- ifelse(kdn$pe== "yes", 1, 0)
kdn[20] <- ifelse(kdn$ane == "yes", 1, 0)
kdn[18] <- ifelse(kdn$htn == "yes", 0, 1)


kdn[21] <- ifelse(kdn$classification == "ckd", 1, 0)
#view(kdn)

# Scaling data for the NN
maxs <- apply(kdn, 2, max)
mins <- apply(kdn, 2, min)
scaled <- as.data.frame(scale(kdn, center = mins, scale = maxs - mins))
#view(scaled)
# Train-test split
train_ <- scaled[184,]
```

```
test_ <- scaled[-184,]
```

```
view(test_)
```

```
set.seed(123)

#applying neural network after doing feature selection with Elastic net

library(neuralnet) # library to fit neural network

n <- names(train_)
f <- as.formula(paste("classification ~", paste(n[!n %in% "classification"], collapse = " + ")))
nn <- neuralnet(f,data=train_,hidden=c(8,8), act.fct = "logistic", linear.output=T)
plot(nn)
```

```
nn$result.matrix
```

```
##                                   [,1]
## error                     1.404661e-05
## reached.threshold         5.300303e-03
## steps                     1.400000e+01
## Intercept.to.1layhid1    -6.299756e-01
## age.to.1layhid1          -2.996775e-01
## bp.to.1layhid1            1.489208e+00
## sg.to.1layhid1            1.008391e-03
## al.to.1layhid1            1.292877e-01
## su.to.1layhid1            1.715065e+00
## rbc.to.1layhid1           3.914162e-01
## pc.to.1layhid1           -1.265061e+00
## pcc.to.1layhid1          -6.868529e-01
## ba.to.1layhid1           -5.151620e-01
## bgr.to.1layhid1           1.154582e+00
## bu.to.1layhid1            2.903138e-01
## sc.to.1layhid1            3.312715e-01
## sod.to.1layhid1           4.118272e-02
## pot.to.1layhid1          -6.253411e-01
## hemo.to.1layhid1          1.717413e+00
## wc.to.1layhid1            4.283505e-01
## rc.to.1layhid1           -2.036117e+00
## htn.to.1layhid1           6.318559e-01
## dm.to.1layhid1           -4.727914e-01
## ane.to.1layhid1          -1.067824e+00
## Intercept.to.1layhid2    -2.874749e-01
## age.to.1layhid2          -1.095504e+00
## bp.to.1layhid2           -7.983912e-01
## sg.to.1layhid2           -6.945393e-01
## al.to.1layhid2           -1.686693e+00
## su.to.1layhid2            8.377870e-01
## rbc.to.1layhid2           8.387312e-02
## pc.to.1layhid2           -1.138137e+00
## pcc.to.1layhid2           1.253815e+00
## ba.to.1layhid2            3.569642e-01
## bgr.to.1layhid2             -3.645715e-01
## bu.to.1layhid2               8.256257e-01
## sc.to.1layhid2               8.086335e-01
## sod.to.1layhid2              7.520811e-01
## pot.to.1layhid2              6.191403e-01
## hemo.to.1layhid2             4.844177e-01
## wc.to.1layhid2              -1.314117e-01
## rc.to.1layhid2              -3.754627e-01
## htn.to.1layhid2             -4.499710e-01
## dm.to.1layhid2              -6.947070e-01
## ane.to.1layhid2             -2.079173e-01
## Intercept.to.1layhid3       -1.334896e+00
## age.to.1layhid3              2.099456e+00
## bp.to.1layhid3               1.138462e+00
## sg.to.1layhid3              -1.192609e+00
## al.to.1layhid3              -4.028848e-01
## su.to.1layhid3              -4.666554e-01
## rbc.to.1layhid3              7.104651e-01
## pc.to.1layhid3              -8.336907e-02
## pcc.to.1layhid3              2.533185e-01
## ba.to.1layhid3              -9.804676e-02
## bgr.to.1layhid3             -1.123705e-01
## bu.to.1layhid3               1.299102e+00
## sc.to.1layhid3              -2.952710e-01
## sod.to.1layhid3              1.446971e+00
## pot.to.1layhid3             -1.618253e+00
## hemo.to.1layhid3             5.151137e-01
## wc.to.1layhid3               5.435424e-02
## rc.to.1layhid3               1.464416e-01
## htn.to.1layhid3              3.101395e-01
## dm.to.1layhid3              -5.023235e-01
## ane.to.1layhid3             -3.332074e-01
## Intercept.to.1layhid4       -9.490754e-01
## age.to.1layhid4             -1.002291e+00
## bp.to.1layhid4               3.730286e-01
## sg.to.1layhid4               5.177098e-01
## al.to.1layhid4               5.300423e-02
## su.to.1layhid4               9.222675e-01
## rbc.to.1layhid4              2.119585e+00
## pc.to.1layhid4              -4.910312e-01
## pcc.to.1layhid4             -2.309169e+00
## ba.to.1layhid4               1.075239e+00
## bgr.to.1layhid4             -6.397008e-01
## bu.to.1layhid4              -6.185086e-01
## sc.to.1layhid4               1.095071e+00
## sod.to.1layhid4             -2.152730e-01
## pot.to.1layhid4             -1.151218e+00
## hemo.to.1layhid4             2.508035e-01
## wc.to.1layhid4              -6.939136e-02
## rc.to.1layhid4               7.526419e-02
## htn.to.1layhid4              4.547804e-01
## dm.to.1layhid4              -3.706600e-01
## ane.to.1layhid4              6.443765e-01
## Intercept.to.1layhid5       -5.529866e-01
## age.to.1layhid5             -7.180361e-04
## bp.to.1layhid5               7.643390e-01
## sg.to.1layhid5               1.026815e-01
## al.to.1layhid5              -3.259316e-01
## su.to.1layhid5               1.148808e+00
## rbc.to.1layhid5              6.610039e-01
## pc.to.1layhid5               5.483970e-01
## pcc.to.1layhid5              2.387317e-01
## ba.to.1layhid5              -9.604061e-01
## bgr.to.1layhid5              1.028152e+00
## bu.to.1layhid5              -9.327596e-01
## sc.to.1layhid5               1.854833e+00
## sod.to.1layhid5              1.200111e+00
## pot.to.1layhid5             -5.682004e-01
## hemo.to.1layhid5            -1.358921e+00
## wc.to.1layhid5              -1.042907e+00
## rc.to.1layhid5              -7.561629e-02
## htn.to.1layhid5             -5.791919e-01
## dm.to.1layhid5              -3.475426e-01
## ane.to.1layhid5             -9.516186e-01
## Intercept.to.1layhid6       -1.145277e-01
## age.to.1layhid6             -8.544045e-01
## bp.to.1layhid6              -1.737442e+00
## sg.to.1layhid6              -4.497265e-01
## al.to.1layhid6               9.189966e-01
## su.to.1layhid6              -5.753470e-01
## rbc.to.1layhid6              5.384643e-01
## pc.to.1layhid6              -1.617883e+00
## pcc.to.1layhid6             -5.556197e-02
## ba.to.1layhid6               4.499072e-01
## bgr.to.1layhid6              2.316534e-01
## bu.to.1layhid6               3.617619e-02
## sc.to.1layhid6              -7.102060e-01
## sod.to.1layhid6             -9.192043e-01
## pot.to.1layhid6             -1.093629e+00
## hemo.to.1layhid6             4.814660e-02
## wc.to.1layhid6              -1.016975e+00
## rc.to.1layhid6              -5.600574e-01
## htn.to.1layhid6             -3.255922e-01
## dm.to.1layhid6               1.843862e+00
## ane.to.1layhid6             -6.519499e-01
## Intercept.to.1layhid7        3.048866e-01
## age.to.1layhid7              1.474608e-01
## bp.to.1layhid7              -8.923566e-01
## sg.to.1layhid7              -1.808086e-03
## al.to.1layhid7               1.444551e+00
## su.to.1layhid7               4.515041e-01
## rbc.to.1layhid7              1.107329e-01
## pc.to.1layhid7              -4.224968e-01
## pcc.to.1layhid7             -2.053247e+00
## ba.to.1layhid7               1.200837e+00
## bgr.to.1layhid7             -1.391140e+00
## bu.to.1layhid7               8.094475e-01
## sc.to.1layhid7               1.978604e+00
## sod.to.1layhid7             -1.374393e+00
## pot.to.1layhid7              7.712843e-01
## hemo.to.1layhid7            -1.926975e-01
## wc.to.1layhid7              -1.502644e+00
## rc.to.1layhid7              -1.445168e+00
## htn.to.1layhid7             -1.532036e+00
## dm.to.1layhid7              -5.309065e-01
## ane.to.1layhid7             -1.461756e+00
## Intercept.to.1layhid8        7.574168e-01
## age.to.1layhid8              2.169609e+00
## bp.to.1layhid8              -1.217530e+00
## sg.to.1layhid8               8.572388e-01
## al.to.1layhid8               7.690422e-01
## su.to.1layhid8               3.322026e-01
## rbc.to.1layhid8             -9.388766e-01
## pc.to.1layhid8              -1.194526e-01
## pcc.to.1layhid8             -2.803953e-01
## ba.to.1layhid8               6.324895e-01
## bgr.to.1layhid8             -3.029388e-01
## bu.to.1layhid8               1.046473e+00
## sc.to.1layhid8              -3.050809e-01
## sod.to.1layhid8              1.122211e+00
## pot.to.1layhid8             -9.796770e-01
## hemo.to.1layhid8            -1.190655e+00
## wc.to.1layhid8               3.310540e+00
## rc.to.1layhid8              -3.473576e-01
## htn.to.1layhid8              3.677276e-01
## dm.to.1layhid8               6.365697e-01
## ane.to.1layhid8             -4.837806e-01
## Intercept.to.2layhid1        4.473620e-01
## 1layhid1.to.2layhid1         2.994645e-01
## 1layhid2.to.2layhid1        -2.848805e-01
## 1layhid3.to.2layhid1        -4.206966e-03
## 1layhid4.to.2layhid1        -1.035673e-01
## 1layhid5.to.2layhid1         2.058952e+00
## 1layhid6.to.2layhid1        -8.108361e-01
## 1layhid7.to.2layhid1        -1.165496e+00
## 1layhid8.to.2layhid1        -3.171160e-02
## Intercept.to.2layhid2        3.799807e-01
## 1layhid1.to.2layhid2         5.060235e-01
## 1layhid2.to.2layhid2        -3.888653e-01
## 1layhid3.to.2layhid2        -9.938261e-01
## 1layhid4.to.2layhid2         1.332685e+00
## 1layhid5.to.2layhid2        -2.801504e-01
## 1layhid6.to.2layhid2        -7.960129e-01
## 1layhid7.to.2layhid2        -1.667796e-01
## 1layhid8.to.2layhid2        -1.276759e-01
## Intercept.to.2layhid3        1.040420e+00
## 1layhid1.to.2layhid3         1.523729e-02
## 1layhid2.to.2layhid3         6.845538e-01
## 1layhid3.to.2layhid3        -5.687920e-01
## 1layhid4.to.2layhid3         1.449453e-01
## 1layhid5.to.2layhid3        -3.941859e-01
## 1layhid6.to.2layhid3         2.508353e-02
## 1layhid7.to.2layhid3        -9.648634e-01
## 1layhid8.to.2layhid3        -1.380302e+00
## Intercept.to.2layhid4        1.927713e+00
## 1layhid1.to.2layhid4         5.312088e-01
## 1layhid2.to.2layhid4        -1.320771e+00
## 1layhid3.to.2layhid4        -6.806659e-01
## 1layhid4.to.2layhid4        -1.254980e+00
## 1layhid5.to.2layhid4         2.129310e+00
## 1layhid6.to.2layhid4         1.242913e+00
## 1layhid7.to.2layhid4        -3.346451e-01
## 1layhid8.to.2layhid4         4.736941e-01
## Intercept.to.2layhid5       -3.448399e-01
## 1layhid1.to.2layhid5        -4.067469e-01
## 1layhid2.to.2layhid5        -7.191028e-01
## 1layhid3.to.2layhid5        -5.251173e-01
## 1layhid4.to.2layhid5         1.720407e+00
## 1layhid5.to.2layhid5         1.547187e-02
## 1layhid6.to.2layhid5         1.887452e-01
## 1layhid7.to.2layhid5         3.131874e-01
## 1layhid8.to.2layhid5         1.301976e+00
## Intercept.to.2layhid6       -7.542238e-01
## 1layhid1.to.2layhid6        -1.230667e+00
## 1layhid2.to.2layhid6         1.437537e+00
## 1layhid3.to.2layhid6        -6.793232e-01
## 1layhid4.to.2layhid6        -9.612260e-01
## 1layhid5.to.2layhid6        -1.474433e+00
## 1layhid6.to.2layhid6        -1.522876e+00
## 1layhid7.to.2layhid6        -8.121335e-01
## 1layhid8.to.2layhid6         3.798258e-01
## Intercept.to.2layhid7        1.040348e+00
## 1layhid1.to.2layhid7         6.380884e-01
## 1layhid2.to.2layhid7        -4.331573e-01
## 1layhid3.to.2layhid7        -9.750063e-03
## 1layhid4.to.2layhid7        -7.740965e-01
## 1layhid5.to.2layhid7        -7.867182e-01
## 1layhid6.to.2layhid7         8.151505e-01
## 1layhid7.to.2layhid7        -1.085093e+00
## 1layhid8.to.2layhid7         1.885794e+00
## Intercept.to.2layhid8       -1.598196e-01
## 1layhid1.to.2layhid8         1.450388e-01
## 1layhid2.to.2layhid8        -8.080277e-01
## 1layhid3.to.2layhid8        -6.438887e-01
## 1layhid4.to.2layhid8        -1.386516e+00
## 1layhid5.to.2layhid8        -2.524254e-01
## 1layhid6.to.2layhid8         3.494824e-01
## 1layhid7.to.2layhid8         2.548043e-01
## 1layhid8.to.2layhid8        -8.510365e-01
## Intercept.to.classification -7.191220e-01
## 2layhid1.to.classification  -4.326987e-01
## 2layhid2.to.classification   1.565561e+00
## 2layhid3.to.classification  -1.067804e+00
## 2layhid4.to.classification  -1.095516e-01
## 2layhid5.to.classification   1.971862e+00
## 2layhid6.to.classification  -3.147489e-02
## 2layhid7.to.classification  -1.290341e+00
## 2layhid8.to.classification  -5.952694e-01
```

```
nn.results <- compute(nn,test_[,1:21])

results <- data.frame(actual = test_$classification, prediction = nn.results$net.result)

results
```

```
##     actual    prediction
## 4        1 -0.1925852961
## 6        1 -0.4230598256
## 10       1 -0.6607037997
## 12       1 -0.9444979999
## 13       1 -1.0626837006
## 15       1 -0.9631680213
## 16       1  0.2464493494
## 19       1 -0.4331987476
## 21       1 -0.4190381477
## 23       1  0.5816776212
## 26       1 -0.4281237748
## 27       1 -0.7215561002
## 28       1  0.0542333962
## 32       1 -0.9986116132
## 33       1 -0.6678105429
## 40       1  0.0454524273
## 43       1 -0.5894453810
## 44       1 -0.1894830531
## 47       1 -0.5319119614
## 49       1 -0.1496790499
## 59       1 -0.4773301805
## 63       1 -0.1475320931
## 70       1 -0.6253690288
## 71       1 -0.2166970689
## 74       1 -0.2521761423
## 76       1 -0.6907647488
## 77       1 -0.4183497746
## 80       1 -0.7605141471
## 84       1  0.4929564785
## 90       1 -0.2129408149
## 91       1 -0.2837965235
## 92       1 -1.1045974970
## 93       1 -1.0741154687
## 94       1 -0.7476123893
## 97       1 -0.3739219222
## 101      1 -1.1832498669
## 105      1 -0.2227114322
## 107      1 -0.0162011824
## 108      1 -0.4578826539
## 110      1 -0.4001009092
## 111      1 -0.2469307004
## 126      1 -0.6227725472
## 127      1  0.2058331392
## 128      1  0.4254782330
## 130         1 -0.9044116454
## 133         1  0.3442001839
## 144         1 -0.4086815776
## 147         1 -0.6217745163
## 153         1 -0.4191169425
## 157         1  0.1943250638
## 159         1 -0.6014481140
## 162         1 -0.5038460239
## 169         1 -0.3463977104
## 170         1  0.5260286270
## 171         1 -0.5632771440
## 175         1 -0.5595671078
## 180         1 -0.5102596937
## 183         1 -0.3352889082
## 187         1 -0.5313179948
## 188         1 -0.6083526862
## 194         1 -0.0630543274
## 196         1  0.2020646235
## 197         1 -0.5682726069
## 198         1 -0.5506675494
## 208         1  0.2900601929
## 210         1  0.3037537727
## 211         1 -1.1064644141
## 215         1 -0.5709811733
## 221         1 -0.7615991490
## 223         1 -0.7938132096
## 224         1 -0.5823582580
## 227         1  0.1109094815
## 239         1 -0.5941559170
## 240         1 -0.8350465816
## 241         1 -0.3213023057
## 242         1 -0.9097098665
## 244         1 -0.7324275808
## 246         1 -0.7148460508
## 247         1 -0.5868644642
## 248         0  0.0693210798
## 249         0 -0.0053708097
## 250         0 -0.0250278027
## 251         0  0.0268209960
## 252         0  0.0688491389
## 253         0  0.0281194568
## 254         0 -0.0040598187
## 255         0  0.0320035281
## 256         0  0.0127809821
## 257         0 -0.0885899063
## 258         0 -0.0605064944
## 259         0 -0.0727904501
## 260         0 -0.0015437953
## 261         0 -0.0578029829
## 262         0 -0.0882921536
## 263         0 -0.0755336263
## 264         0 -0.0177730349
## 265         0 -0.0058609597
## 267         0 -0.0020440927
## 268      0 -0.0071527494
## 269      0  0.0312757275
## 270      0 -0.0005976957
## 273      0 -0.0942694386
## 275      0  0.0190457375
## 276      0 -0.0536468026
## 277      0 -0.0398489138
## 279      0 -0.0756514015
## 280      0 -0.0552581067
## 282      0 -0.0125910812
## 283      0  0.0052005082
## 284      0 -0.0216333364
## 286      0  0.0677950731
## 287      0 -0.0450963562
## 288      0 -0.0070656185
## 289      0 -0.0307521947
## 290      0 -0.0373216753
## 292      0 -0.0887201385
## 294      0 -0.1106872578
## 295      0 -0.0100252253
## 296      0  0.0423021191
## 297      0 -0.1466684812
## 298      0 -0.0169535244
## 299      0 -0.0069911398
## 302      0  0.0101836308
## 303      0 -0.0437878308
## 304      0 -0.1045245442
## 305      0 -0.0710561089
## 306      0  0.0468127214
## 307      0 -0.0407273299
## 308      0 -0.0106798177
## 309      0 -0.0040987608
## 311      0 -0.0540549052
## 312      0 -0.0618406898
## 315      0 -0.0321941681
## 316      0  0.0128048733
## 318      0 -0.1191127626
## 319      0 -0.0519862429
## 321      0 -0.0261567871
## 323      0 -0.0238014056
## 324      0 -0.0130456011
## 325      0 -0.0004210785
## 327      0  0.0530326951
## 329      0 -0.0709571887
## 331      0 -0.1528857608
## 333      0 -0.1139677391
## 335      0  0.0292349021
## 336      0 -0.1165804981
## 337      0  0.0084515470
## 338      0  0.0381526224
## 339      0 -0.0879797772
## 340      0 -0.0318096552
## 341      0  0.0353735683
## 342      0 -0.0636337189
## 343      0  0.1133293092
## 345      0  0.0052475631
## 346      0 -0.0180144124
## 347      0  0.0130329074
## 348      0  0.0427263624
## 349      0 -0.0181572524
## 350      0 -0.0472629198
## 351      0 -0.0685726681
## 352      0  0.0248216672
## 353      0 -0.0214970420
## 354      0 -0.0090585009
## 355      0  0.0570401487
## 356      0  0.0090866823
## 357      0 -0.0928128556
## 358      0  0.0416309044
## 359      0  0.0062951024
## 360      0  0.0531546676
## 362      0  0.0034829918
## 364      0  0.0640795049
## 365      0 -0.0168023704
## 366      0  0.0013664169
## 367      0 -0.0668150876
## 368      0 -0.0006440036
## 369      0 -0.0195453357
## 370      0 -0.1053969145
## 371      0 -0.0112600723
## 372      0 -0.0207406864
## 373      0 -0.0398621110
## 374      0  0.0344555559
## 375      0 -0.0548618853
## 377      0  0.0252428253
## 379      0 -0.0246807503
## 380      0  0.0327280083
## 381      0 -0.0717443058
## 382      0 -0.0830628643
## 383      0 -0.0892677058
## 384      0 -0.0184964305
## 385      0  0.0033545381
## 386      0  0.0130847356
## 387      0 -0.0470485362
## 388      0  0.0468155794
## 389      0  0.0223047350
## 390      0 -0.1179505321
## 391      0  0.0004813954
## 392      0 -0.0904139708
## 393      0 -0.1930669830
## 394      0  0.0540635098
## 395      0 -0.0689727460
## 396      0  0.0521634093
## 397      0  0.0089135859
```

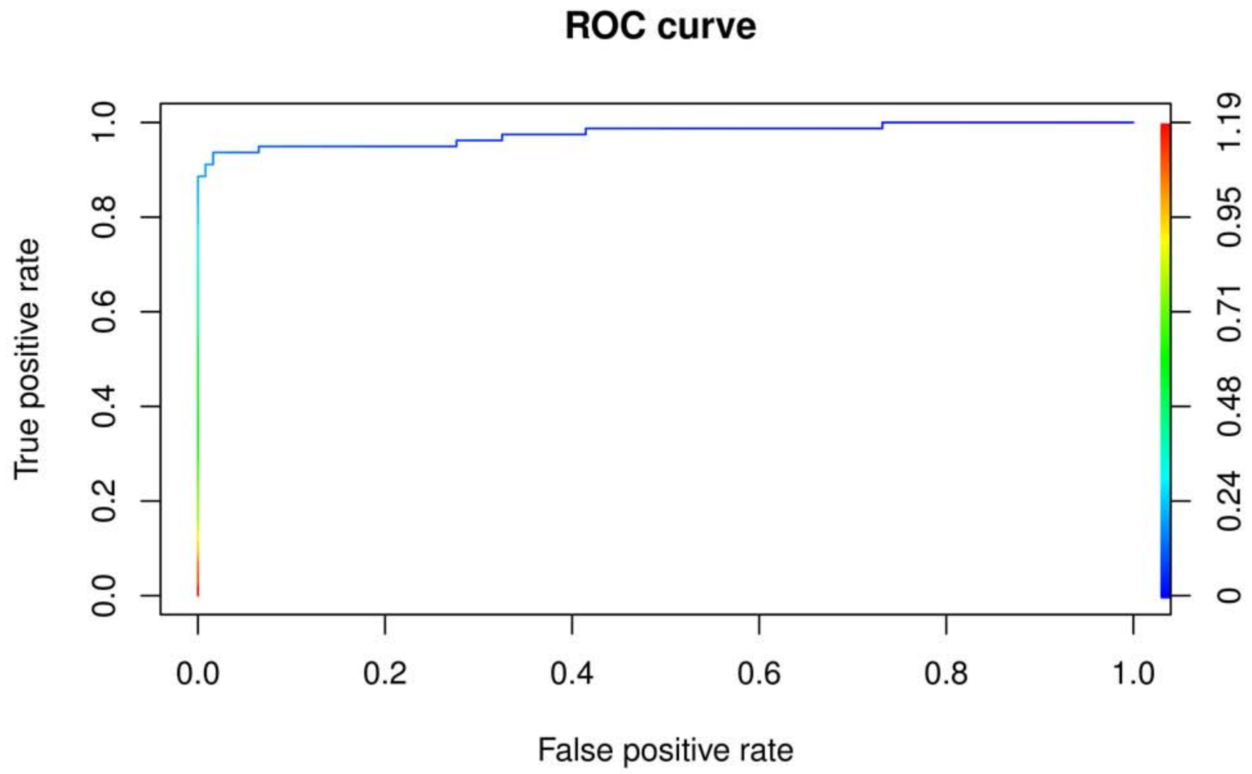
```
roundedresults<-sapply(results,round,digits=0)
roundedresultsdf=data.frame(roundedresults)
attach(roundedresultsdf)
```

```
#table(actual,prediction)
```

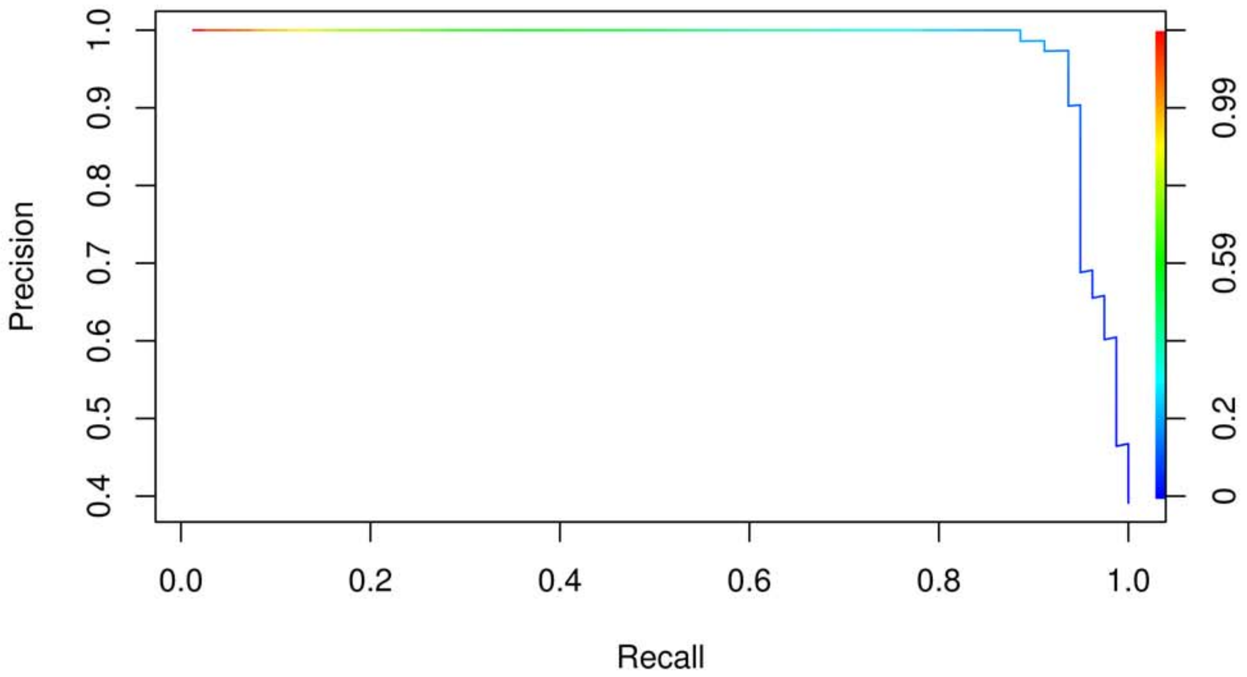
```
#84 percetnt ACCuracey   123+45 divide over 203
detach(package:neuralnet,unload = T)
#dplyr::select(neuralnet)

pr<-nn.results$net.result
pr<-abs(pr)


library(ROCR)
```

```
## Warning: package 'ROCR' was built under R version 4.2.3
```

```
nn.pred = prediction(pr, test_$classification)

pref <- performance(nn.pred, "tpr", "fpr")


pref2 <- performance(nn.pred, "prec", "rec")
plot(pref,colorize=T,main="ROC curve ")
```



```
plot(pref2,colorize=T)
```



```
auc(test_$classification, pr)
```

```
## Setting levels: control = 0, case = 1
```

```
## Warning in roc.default(response, predictor, auc = TRUE, ...): Deprecated use a
## matrix as predictor. Unexpected results may be produced, please pass a numeric
## vector.
```

```
## Setting direction: controls < cases
```

```
## Area under the curve: 0.9764
```

```
roc_score = roc(test_[, 21], pr)  # AUC score
```

```
## Setting levels: control = 0, case = 1
```

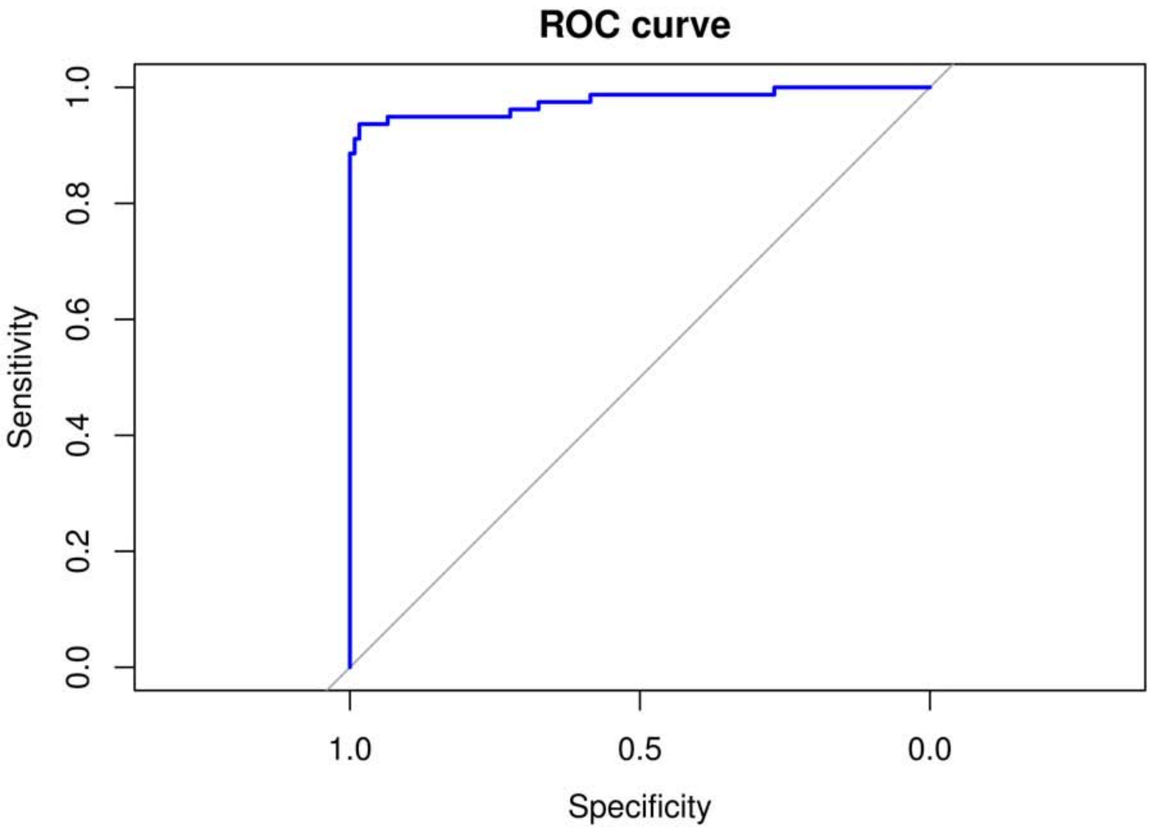
```
## Warning in roc.default(test_[, 21], pr): Deprecated use a matrix as predictor.
## Unexpected results may be produced, please pass a numeric vector.
```

```
## Setting direction: controls < cases
```

```
roc_score
```

```
##
## Call:
## roc.default(response = test_[, 21], predictor = pr)
##
## Data: pr in 123 controls (test_[, 21] 0) < 79 cases (test_[, 21] 1).
## Area under the curve: 0.9764
```

```
plot(roc_score,col="blue", main="ROC curve ")
```



```
#confusion matrix
Result<-confusionMatrix(test_$classification, pr)
Result
```

```
##       [,1] [,2]
## [1,]   123   38
## [2,]     0   41
```

```
#ACCuracy
```

Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.