Crimes in Chicago - A Predictive Approach

Milestone 2 - Data Analysis I

By Rushiraj Nenuji,Nischitha Krishna, Chandan Ramanna, Santhosh Kumar Ravindran and Yin Yang

March 5, 2017

# Regression models

## Loading data set

crimedf <- read.csv(file = "Crimes\_in\_Chicago.csv", stringsAsFactors = FALSE)

## Cleansing data

crimedf <- na.omit(crimedf)

## Convert Domestic into Dummy 1 for true and 0 for false

crimedf$domestic\_n <- ifelse (crimedf$Domestic == 'true',1,0)

# Regression models

## Convert Domestic into Dummy 1 for true and 0 for false

crimedf$domestic\_n <- ifelse (crimedf$Domestic == 'true',1,0)

## Filtering 1000 records for past2 years.

crimedf1 <- subset(crimedf,Year == 2017)  
  
library(magrittr)  
library(dplyr)

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

Crimes\_Type <- crimedf1 %>% group\_by(Primary.Type) %>% dplyr:: summarise(arrest\_count = n())   
  
Crimes\_Type <- data.frame (Crimes\_Type)  
Crimes\_Type<-Crimes\_Type[order(-Crimes\_Type$arrest\_count),]  
Crimes\_Type <- head(Crimes\_Type,5)  
top10.crimes <- c(Crimes\_Type$Primary.Type)  
  
crimedf2 <- subset(crimedf1, Primary.Type %in% top10.crimes)  
  
crimedf.new <- crimedf2[c(6:10,14,18)]  
crimedf\_filtered <- head(crimedf.new,1000)

## Logistic regression model

## Train a logistic regression model with 10-fold cross-validation

library(caret)

## Loading required package: lattice

## Loading required package: ggplot2

fitControl <- trainControl(method = "cv",number = 15)  
library(caret)  
set.seed(123)  
logit\_fit <- train(Arrest ~ ., data = crimedf\_filtered[1:7],  
 trControl = fitControl,  
 method="glm", family=binomial(link='logit'))

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type =  
## ifelse(type == : prediction from a rank-deficient fit may be misleading

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type =  
## ifelse(type == : prediction from a rank-deficient fit may be misleading

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type =  
## ifelse(type == : prediction from a rank-deficient fit may be misleading

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type =  
## ifelse(type == : prediction from a rank-deficient fit may be misleading

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type =  
## ifelse(type == : prediction from a rank-deficient fit may be misleading

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type =  
## ifelse(type == : prediction from a rank-deficient fit may be misleading

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type =  
## ifelse(type == : prediction from a rank-deficient fit may be misleading

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type =  
## ifelse(type == : prediction from a rank-deficient fit may be misleading

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type =  
## ifelse(type == : prediction from a rank-deficient fit may be misleading

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type =  
## ifelse(type == : prediction from a rank-deficient fit may be misleading

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type =  
## ifelse(type == : prediction from a rank-deficient fit may be misleading

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type =  
## ifelse(type == : prediction from a rank-deficient fit may be misleading

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type =  
## ifelse(type == : prediction from a rank-deficient fit may be misleading

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type =  
## ifelse(type == : prediction from a rank-deficient fit may be misleading

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

## Warning in predict.lm(object, newdata, se.fit, scale = 1, type =  
## ifelse(type == : prediction from a rank-deficient fit may be misleading

## Warning: glm.fit: algorithm did not converge

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

print(logit\_fit)

## Generalized Linear Model   
##   
## 1000 samples  
## 6 predictor  
## 2 classes: 'false', 'true'   
##   
## No pre-processing  
## Resampling: Cross-Validated (15 fold)   
## Summary of sample sizes: 933, 933, 933, 934, 933, 933, ...   
## Resampling results:  
##   
## Accuracy Kappa   
## 0.8589628 0.2307191  
##   
##

confusionMatrix(logit\_fit)

## Cross-Validated (15 fold) Confusion Matrix   
##   
## (entries are percentual average cell counts across resamples)  
##   
## Reference  
## Prediction false true  
## false 83.0 11.3  
## true 2.8 2.9  
##   
## Accuracy (average) : 0.859

## Train Support Vector Machine (Radial Basis Function Kernel) with 2-fold Cross-Validation

set.seed(123)  
svmRadial\_fit <- train(Arrest ~ ., data = crimedf\_filtered[1:7],  
 trControl = fitControl, method = "svmRadial",  
 verbose=FALSE)

## Loading required package: kernlab

##   
## Attaching package: 'kernlab'

## The following object is masked from 'package:ggplot2':  
##   
## alpha

## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.  
  
## Warning in .local(x, ...): Variable(s) `' constant. Cannot scale data.

print(svmRadial\_fit)

## Support Vector Machines with Radial Basis Function Kernel   
##   
## 1000 samples  
## 6 predictor  
## 2 classes: 'false', 'true'   
##   
## No pre-processing  
## Resampling: Cross-Validated (15 fold)   
## Summary of sample sizes: 933, 933, 933, 934, 933, 933, ...   
## Resampling results across tuning parameters:  
##   
## C Accuracy Kappa  
## 0.25 0.858028 0   
## 0.50 0.858028 0   
## 1.00 0.858028 0   
##   
## Tuning parameter 'sigma' was held constant at a value of 0.02397997  
## Accuracy was used to select the optimal model using the largest value.  
## The final values used for the model were sigma = 0.02397997 and C = 0.25.

confusionMatrix(svmRadial\_fit)

## Cross-Validated (15 fold) Confusion Matrix   
##   
## (entries are percentual average cell counts across resamples)  
##   
## Reference  
## Prediction false true  
## false 85.8 14.2  
## true 0.0 0.0  
##   
## Accuracy (average) : 0.858

## Train Gradient Boosted Machine (GBM)

library(plyr)

## -------------------------------------------------------------------------

## You have loaded plyr after dplyr - this is likely to cause problems.  
## If you need functions from both plyr and dplyr, please load plyr first, then dplyr:  
## library(plyr); library(dplyr)

## -------------------------------------------------------------------------

##   
## Attaching package: 'plyr'

## The following objects are masked from 'package:dplyr':  
##   
## arrange, count, desc, failwith, id, mutate, rename, summarise,  
## summarize

set.seed(123)  
gbm\_fit <-train(Arrest ~ ., data = crimedf\_filtered[1:7],  
 trControl = fitControl, method = "gbm",  
 verbose=FALSE)

## Loading required package: gbm

## Loading required package: survival

##   
## Attaching package: 'survival'

## The following object is masked from 'package:caret':  
##   
## cluster

## Loading required package: splines

## Loading required package: parallel

## Loaded gbm 2.1.1

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 101: Year has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 101: Year has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 101: Year has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 96: Location.DescriptionVEHICLE - DELIVERY TRUCK has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 101: Year has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 96: Location.DescriptionVEHICLE - DELIVERY TRUCK has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 101: Year has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 96: Location.DescriptionVEHICLE - DELIVERY TRUCK has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 101: Year has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 12: DescriptionAGGRAVATED PO: OTHER DANG WEAP has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 35: DescriptionPROBATION VIOLATION has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0,  
## 0, : variable 73: Location.DescriptionGOVERNMENT BUILDING/PROPERTY has no  
## variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 77: Location.DescriptionMOVIE HOUSE/THEATER has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 101: Year has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 12: DescriptionAGGRAVATED PO: OTHER DANG WEAP has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 35: DescriptionPROBATION VIOLATION has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0,  
## 0, : variable 73: Location.DescriptionGOVERNMENT BUILDING/PROPERTY has no  
## variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 77: Location.DescriptionMOVIE HOUSE/THEATER has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 101: Year has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 12: DescriptionAGGRAVATED PO: OTHER DANG WEAP has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 35: DescriptionPROBATION VIOLATION has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0,  
## 0, : variable 73: Location.DescriptionGOVERNMENT BUILDING/PROPERTY has no  
## variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 77: Location.DescriptionMOVIE HOUSE/THEATER has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 101: Year has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 101: Year has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 101: Year has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 101: Year has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 93: Location.DescriptionTAVERN/LIQUOR STORE has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 101: Year has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 93: Location.DescriptionTAVERN/LIQUOR STORE has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 101: Year has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 93: Location.DescriptionTAVERN/LIQUOR STORE has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 101: Year has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, :  
## variable 6: DescriptionAGG PRO.EMP:KNIFE/CUTTING INST has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, :  
## variable 101: Year has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, :  
## variable 6: DescriptionAGG PRO.EMP:KNIFE/CUTTING INST has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, :  
## variable 101: Year has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, :  
## variable 6: DescriptionAGG PRO.EMP:KNIFE/CUTTING INST has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, :  
## variable 101: Year has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 48: Location.DescriptionAIRPORT EXTERIOR - NON-SECURE AREA has no  
## variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 49: Location.DescriptionAIRPORT TERMINAL LOWER LEVEL - SECURE AREA  
## has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 101: Year has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 48: Location.DescriptionAIRPORT EXTERIOR - NON-SECURE AREA has no  
## variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 49: Location.DescriptionAIRPORT TERMINAL LOWER LEVEL - SECURE AREA  
## has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 101: Year has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 48: Location.DescriptionAIRPORT EXTERIOR - NON-SECURE AREA has no  
## variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 49: Location.DescriptionAIRPORT TERMINAL LOWER LEVEL - SECURE AREA  
## has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 101: Year has no variation.

## Warning in gbm.fit(x = structure(c(1, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 23: DescriptionGUN OFFENDER: ANNUAL REGISTRATION has no variation.

## Warning in gbm.fit(x = structure(c(1, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0,  
## 0, : variable 47: DescriptionVIOLENT OFFENDER: DUTY TO REGISTER has no  
## variation.

## Warning in gbm.fit(x = structure(c(1, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 80: Location.DescriptionOTHER RAILROAD PROP / TRAIN DEPOT has no  
## variation.

## Warning in gbm.fit(x = structure(c(1, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 101: Year has no variation.

## Warning in gbm.fit(x = structure(c(1, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 23: DescriptionGUN OFFENDER: ANNUAL REGISTRATION has no variation.

## Warning in gbm.fit(x = structure(c(1, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0,  
## 0, : variable 47: DescriptionVIOLENT OFFENDER: DUTY TO REGISTER has no  
## variation.

## Warning in gbm.fit(x = structure(c(1, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 80: Location.DescriptionOTHER RAILROAD PROP / TRAIN DEPOT has no  
## variation.

## Warning in gbm.fit(x = structure(c(1, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 101: Year has no variation.

## Warning in gbm.fit(x = structure(c(1, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 23: DescriptionGUN OFFENDER: ANNUAL REGISTRATION has no variation.

## Warning in gbm.fit(x = structure(c(1, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0,  
## 0, : variable 47: DescriptionVIOLENT OFFENDER: DUTY TO REGISTER has no  
## variation.

## Warning in gbm.fit(x = structure(c(1, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 80: Location.DescriptionOTHER RAILROAD PROP / TRAIN DEPOT has no  
## variation.

## Warning in gbm.fit(x = structure(c(1, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 101: Year has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 101: Year has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 101: Year has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 101: Year has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 101: Year has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 101: Year has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 101: Year has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 81: Location.DescriptionPARK PROPERTY has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 101: Year has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 81: Location.DescriptionPARK PROPERTY has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 101: Year has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 81: Location.DescriptionPARK PROPERTY has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 101: Year has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 30: DescriptionOTHER WEAPONS VIOLATION has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 50: Location.DescriptionAIRPORT TERMINAL UPPER LEVEL - SECURE AREA  
## has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 54: Location.DescriptionBANK has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 62: Location.DescriptionCONSTRUCTION SITE has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 65: Location.DescriptionCTA BUS STOP has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 101: Year has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 30: DescriptionOTHER WEAPONS VIOLATION has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 50: Location.DescriptionAIRPORT TERMINAL UPPER LEVEL - SECURE AREA  
## has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 54: Location.DescriptionBANK has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 62: Location.DescriptionCONSTRUCTION SITE has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 65: Location.DescriptionCTA BUS STOP has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 101: Year has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 30: DescriptionOTHER WEAPONS VIOLATION has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 50: Location.DescriptionAIRPORT TERMINAL UPPER LEVEL - SECURE AREA  
## has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 54: Location.DescriptionBANK has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 62: Location.DescriptionCONSTRUCTION SITE has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 65: Location.DescriptionCTA BUS STOP has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 101: Year has no variation.

## Warning in gbm.fit(x = structure(c(0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, :  
## variable 20: DescriptionEAVESDROPPING has no variation.

## Warning in gbm.fit(x = structure(c(0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, :  
## variable 56: Location.DescriptionBARBERSHOP has no variation.

## Warning in gbm.fit(x = structure(c(0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, :  
## variable 101: Year has no variation.

## Warning in gbm.fit(x = structure(c(0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, :  
## variable 20: DescriptionEAVESDROPPING has no variation.

## Warning in gbm.fit(x = structure(c(0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, :  
## variable 56: Location.DescriptionBARBERSHOP has no variation.

## Warning in gbm.fit(x = structure(c(0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, :  
## variable 101: Year has no variation.

## Warning in gbm.fit(x = structure(c(0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, :  
## variable 20: DescriptionEAVESDROPPING has no variation.

## Warning in gbm.fit(x = structure(c(0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, :  
## variable 56: Location.DescriptionBARBERSHOP has no variation.

## Warning in gbm.fit(x = structure(c(0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, :  
## variable 101: Year has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 83: Location.DescriptionPOOL ROOM has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 101: Year has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 83: Location.DescriptionPOOL ROOM has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 101: Year has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 83: Location.DescriptionPOOL ROOM has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 101: Year has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 11: DescriptionAGGRAVATED OF A UNBORN CHILD has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 39: DescriptionSEX OFFENDER: FAIL TO REGISTER has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 101: Year has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 11: DescriptionAGGRAVATED OF A UNBORN CHILD has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 39: DescriptionSEX OFFENDER: FAIL TO REGISTER has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 101: Year has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 11: DescriptionAGGRAVATED OF A UNBORN CHILD has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 39: DescriptionSEX OFFENDER: FAIL TO REGISTER has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 101: Year has no variation.

## Warning in gbm.fit(x = structure(c(1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, :  
## variable 101: Year has no variation.

print(gbm\_fit)

## Stochastic Gradient Boosting   
##   
## 1000 samples  
## 6 predictor  
## 2 classes: 'false', 'true'   
##   
## No pre-processing  
## Resampling: Cross-Validated (15 fold)   
## Summary of sample sizes: 933, 933, 933, 934, 933, 933, ...   
## Resampling results across tuning parameters:  
##   
## interaction.depth n.trees Accuracy Kappa   
## 1 50 0.8580280 0.000000000  
## 1 100 0.8580280 0.000000000  
## 1 150 0.8580280 0.000000000  
## 2 50 0.8580280 0.000000000  
## 2 100 0.8560229 -0.003728759  
## 2 150 0.8500226 -0.014330791  
## 3 50 0.8580280 0.000000000  
## 3 100 0.8510176 0.004421006  
## 3 150 0.8479723 0.030392489  
##   
## Tuning parameter 'shrinkage' was held constant at a value of 0.1  
##   
## Tuning parameter 'n.minobsinnode' was held constant at a value of 10  
## Accuracy was used to select the optimal model using the largest value.  
## The final values used for the model were n.trees = 50, interaction.depth  
## = 1, shrinkage = 0.1 and n.minobsinnode = 10.

confusionMatrix(gbm\_fit)

## Cross-Validated (15 fold) Confusion Matrix   
##   
## (entries are percentual average cell counts across resamples)  
##   
## Reference  
## Prediction false true  
## false 85.8 14.2  
## true 0.0 0.0  
##   
## Accuracy (average) : 0.858

## Compare Different Predictive Models

## Data Partitioning

crimedf\_filtered$THEFT <- ifelse(crimedf\_filtered$Primary.Type == 'THEFT', 1,0)  
crimedf\_filtered$BATTERY <- ifelse(crimedf\_filtered$Primary.Type == 'BATTERY', 1,0)  
crimedf\_filtered$ASSAULT <- ifelse(crimedf\_filtered$Primary.Type == 'ASSAULT', 1,0)  
crimedf\_filtered$CRIMINAL.DAMAGE <- ifelse(crimedf\_filtered$Primary.Type == 'CRIMINAL DAMAGE', 1,0)  
crimedf\_filtered$OTHER.OFFENSE <- ifelse(crimedf\_filtered$Primary.Type == 'OTHER OFFENSE', 1,0)  
crimedf\_filtered$Arrest1 <- ifelse(crimedf\_filtered$Arrest == 'true', 1,0)  
crimedf\_filtered$Domestic1 <- ifelse(crimedf\_filtered$Domestic == 'true', 1,0)  
crimedf\_filtered$Year <- as.numeric(crimedf\_filtered$Year)  
crimeneuraldf <- crimedf\_filtered[c(7:14)]  
crimeneuraldf$Year <- ifelse(crimedf\_filtered$Year == 2017,1,0)

## Structure

str(crimeneuraldf)

## 'data.frame': 1000 obs. of 8 variables:  
## $ Year : num 1 1 1 1 1 1 1 1 1 1 ...  
## $ THEFT : num 0 0 0 0 0 0 1 0 0 0 ...  
## $ BATTERY : num 1 0 1 0 1 0 0 0 0 0 ...  
## $ ASSAULT : num 0 0 0 0 0 0 0 1 1 0 ...  
## $ CRIMINAL.DAMAGE: num 0 1 0 1 0 1 0 0 0 1 ...  
## $ OTHER.OFFENSE : num 0 0 0 0 0 0 0 0 0 0 ...  
## $ Arrest1 : num 1 0 0 1 0 0 0 0 0 1 ...  
## $ Domestic1 : num 0 0 1 0 0 0 0 0 1 0 ...

## We use a single 80/20% split.

set.seed(1234)  
library(caret)  
trainIndex <- createDataPartition(crimeneuraldf$Arrest1, p = .8, list = FALSE)  
head(trainIndex)

## Resample1  
## [1,] 1  
## [2,] 2  
## [3,] 3  
## [4,] 4  
## [5,] 6  
## [6,] 7

train\_data <- crimeneuraldf[ trainIndex,]  
test\_data <- crimeneuraldf[-trainIndex,]

## Fit ANN Model on the Training Dataset

We use the neuralnet R package for neural network modeling. For a list of prediction and classification models, refer to <http://topepo.github.io/caret/available-models.html>

# Load the neuralnet package  
library('neuralnet')

##   
## Attaching package: 'neuralnet'

## The following object is masked from 'package:dplyr':  
##   
## compute

After we load the neuralnet package, we can use the neuralnet() method to train the neural network model.

The 'hidden' is used to set a vector of integers specifying the number of hidden neurons (vertices) in each layer.

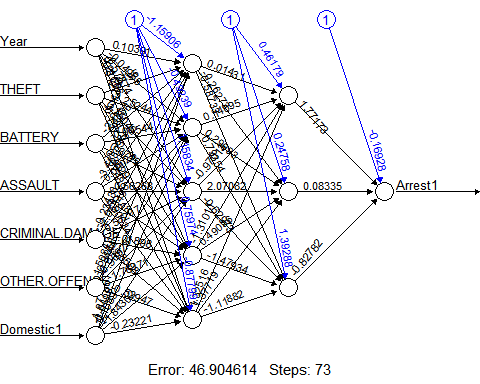
f <- as.formula(Arrest1 ~ Year+THEFT+BATTERY+ASSAULT+CRIMINAL.DAMAGE+OTHER.OFFENSE+Domestic1)

# Fit a neural network model with 2 hidden layers

nn\_fit\_2 <- neuralnet(f, data = train\_data, hidden = c(5,3), linear.output=TRUE)

Now, let's show the structure of the neural network trained.

plot(nn\_fit\_2,rep="best", cex=0.8)



# Show results

summary(nn\_fit\_2)

## Length Class Mode   
## call 5 -none- call   
## response 800 -none- numeric   
## covariate 5600 -none- numeric   
## model.list 2 -none- list   
## err.fct 1 -none- function  
## act.fct 1 -none- function  
## linear.output 1 -none- logical   
## data 8 data.frame list   
## net.result 1 -none- list   
## weights 1 -none- list   
## startweights 1 -none- list   
## generalized.weights 1 -none- list   
## result.matrix 65 -none- numeric

# Evaluate Predictive Performance of the ANN Model

## Computes the outputs of the ANN model on the test dataset

Notice that the prediction result of the neuralnet is the probability of belonging to a class rather than the class.

Hint:

* Use the compute() method to calculate prediction;
* In the compute() method, you need to NOT include the 'Personal\_Loan' column, or else it will pop up an error of non-conformable arguments.

pred2\_norm <- compute(nn\_fit\_2, test\_data[-1])  
pred2\_norm <- pred2\_norm$net.result

test\_data$pred <- ifelse(pred2\_norm >= 0.2, 1, 0 )

## Create a confusion matrix to show the performance of the ANN model on the test set.

confusionMatrix(test\_data$Arrest1,test\_data$pred)

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction 0 1  
## 0 125 49  
## 1 9 17  
##   
## Accuracy : 0.71   
## 95% CI : (0.6418125, 0.771843)  
## No Information Rate : 0.67   
## P-Value [Acc > NIR] : 0.1291374   
##   
## Kappa : 0.2250134   
## Mcnemar's Test P-Value : 0.0000003039988   
##   
## Sensitivity : 0.9328358   
## Specificity : 0.2575758   
## Pos Pred Value : 0.7183908   
## Neg Pred Value : 0.6538462   
## Prevalence : 0.6700000   
## Detection Rate : 0.6250000   
## Detection Prevalence : 0.8700000   
## Balanced Accuracy : 0.5952058   
##   
## 'Positive' Class : 0   
##