library("arules")

library("e1071")

library("rpart")

# Apriori and SVM

#Identifying important crime factors

colnames(dataABC)

table(dataABC$Arson)

#Cleaning the data

data <- dataABC[,c(67:74,32)]

colnames(data)

data <- na.omit(data)

sum([is.na](http://is.na/)(data))

colnames(data)

data$Aggravated.assault <- as.factor(data$Aggravated.assault)

data$Arson <- as.factor(data$Arson)

data$Burglary <- as.factor(data$Burglary)

data$Larceny <- as.factor(data$Larceny)

data$Murder <- as.factor(data$Murder)

data$Robbery <- as.factor(data$Robbery)

data$Vehicle.theft <- as.factor(data$Vehicle.theft)

data$Total <- as.factor(data$Total)

data$VacantBuilding <- as.factor(data$VacantBuilding)

#Applying apriori to identify important factors contributing to VacantBuilding feature

rules <- apriori(data,

parameter = list(supp = 0.5, conf = 0.9,target = "rules"))

#Analyzing the rules and finding out most important factors which contribute to "No" in VacantBuilding column

inspect(rules)

top.support <- sort(rules, decreasing = TRUE, na.last = NA, by = "lift")

inspect( subset( top.support, subset = rhs %pin% "VacantBuilding=N" ) )

# From the below result, we can infer that when there are no aggravated assault, no arson, no robbery and no Vehicle theft then there is a good chance the buildings in the blocks will not be empty

# lhs rhs support confidence lift count

#[1] {Aggravated.assault=0,

#Arson=0,

#Robbery=0,

#Vehicle.theft=0} => {VacantBuilding=N} 0.5704553 0.9411002 1.0091244 8708

#SVM model

#The SVM model is built with these four features - Aggravted assault, Arson, Robbery and Vehicle theft to predict vacant buildings

#Dividing the dataset to train and test data

colnames(data)

#Selecting only the features - (Aggravated assault, Arson, Robbery, Vehicle theft) and the prediction-(VacantBuilding) from the dataset

colnames(data)

data1 <- data[,c(1,2,6,7,9)]

table(data1$Arson)

#Randomizing the rows

data1 <- data1[sample(nrow(data1)),]

# Taking 80% of data as training data and 20% as testing

data\_train <- data1[1:12212,]

data\_test <- data1[12213:15265,]

# Building the SVM model

svm.model <- svm(VacantBuilding ~ ., data = data\_train, cost = 100, gamma = 100,kernel="radial")

#Using the test data to predict the model

svm.pred <- predict(svm.model, data\_test)

svm.pred

#Comparing the testing values and predicted values

comparison <- data.frame(data\_test$VacantBuilding,svm.pred)

comparison