Unsupervised Anomaly Detection – Independent Study

**** All the coding in this study has been done on R software.**

Research Question:

How does anomaly detection accuracy vary among the three unsupervised methods, i.e., Kernel Density Estimation, One-Class SVM, and Isolation Forests?

Abstract:

The burgeoning of technology and social media has led to a significant increase in data worldwide. An anomaly within data is an event that differs from the rest, such as fraud detection in financial transactions, intrusion detection, and ecological disturbances. Information on such peculiar circumstances trains the applications for unexpected events and allows them to be modeled effectively. Unsupervised anomaly detection is a similar phenomenon applied to unlabeled datasets. As tagging new data daily is rendered impossible, the advent of this field has substituted the need to study only labeled datasets. The Unsupervised techniques used in this study are Kernel Density Estimation, One-Class SVM, and Isolation Forests. The results imply that the degree of anomalies across these unsupervised techniques is in the following ascending order: One-Class SVM, Kernel Density Estimation, and Isolation Forests.

Introduction:

The data set for this study, "**USArrests: Violent Crime Rates by US State**," is a package dataset taken from the R-data Statistics page. This data set contains four continuous variables **assault, murder, rape,** and the **population percentage** for each of the 50 US states in 1973. The data set had no missing values or unnecessary features, so data cleaning was insignificant.

After loading the entire data frame, we use the R to find the summary, box plot, and standard deviation for each column variable in our data set. In the boxplots **(Fig 1-4)**, we could see that a few of our training points were potential outliers. These outliers were not extracted from the original data set as in real-world sampled data; we can have an outlier test data point, and our model should be trained on outliers to handle such cases. It is also possible that the outliers may be potential anomalies, and thus, they must exist. This improves the efficiency and feasibility of results by giving the minimum error rate in prediction.

```
summary(USArrests_data['Murder'])

boxplot(USArrests_data['Murder'], ylab= 'Murder')

sd(as.matrix(USArrests_data['Murder']))
```

Finally, a correlation plot was done amongst all four variables to check for perfect collinearity. **Fig 5**. Indicates that none of the predictor variables is perfectly collinear, but the elements of multi-collinearity are present. As normalizing the variables does not impact the model's deviation or predictions, none of

the variables were normalized in this study. Since all the basic necessary steps of pre-processing and evaluating the data set are ticked, the method for data analysis is extensively elaborated upon, assumptions are explained, results are analyzed, and three disparate unsupervised models are made, concluding with the predicted anomalies in the data set.

Methods:

The three unsupervised methods used for anomaly detection are:

i) Kernel Density Estimation
ii) One-Class SVM
iii) Isolation Forests

A kernel density function is a mathematical approach to identifying the peculiar points from the normal/standard data points. Inside R, a function **density** is used to compute the densities of features in a neighborhood around those features.

```
USArrests_den <- density(USArrests_mat)
```

The density is a reflective measure of the dispersion from one point to another. This implies that an anomaly is a point whose density measure is extreme compared to the mean of the densities. The mathematical formula for measuring that extreme is the following. Anything less than 20% of the ratio of minimum density to mean density (aka center of radius) is considered an anomaly.

$$anomaly < 0.20 * \frac{\min density}{mean\ density}$$

One-Class SVM is considered the most popular and practical approach to detecting anomalies. A One-Class SVM aims to divide the whole data into two classes, the positive ones as a class of normal points and the negative ones considered as a class of outliers. In R, the **SVM** function is used to serve this task. The values of gamma and nu are set to standard values of 0.05. A radial kernel is a popular kernel used in various kernelized algorithms.

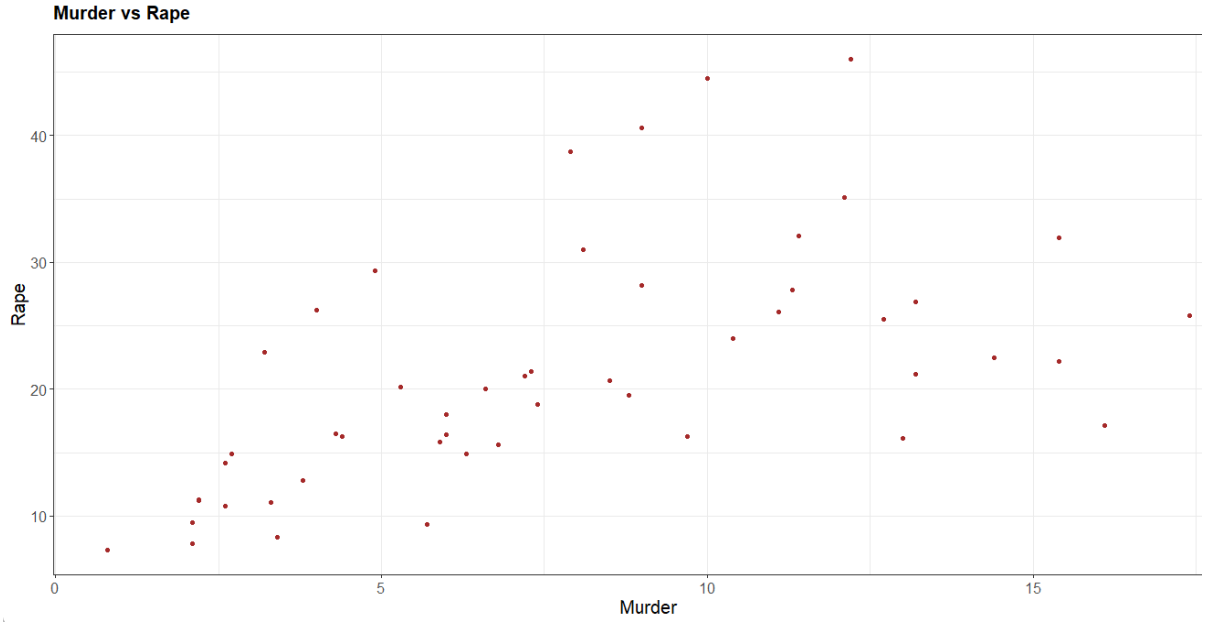
```
model_oneclasssvm <- svm(USArrests_data,type='one-classification',kernel =
"radial",gamma=0.05,nu=0.05)
```

Isolation Forest is an algorithm designed specifically for outlier detection. The algorithm's general gist is to split the data's sub-samples according to some feature at random. The rarer an observation, the more likely it will end up on an isolated sample with all the standard values appearing in another sample. In R, the **Isolation.Forest** function is used to run the algorithm. The ntrees and nthreads are set to default values of 10 and 1. The ntrees

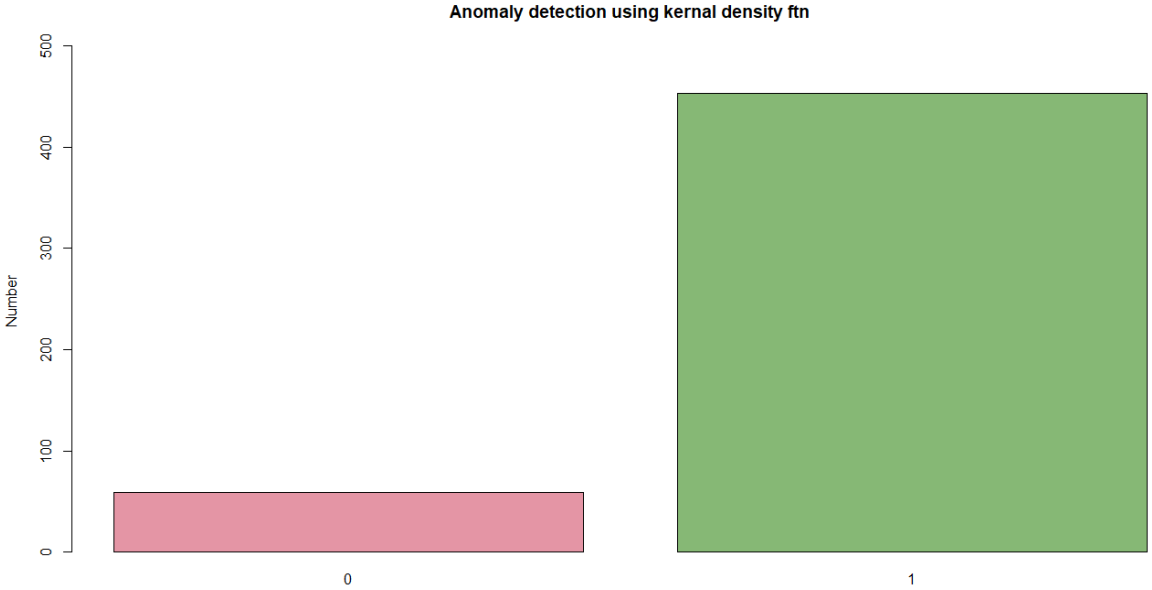
```
iforest <- isolation.forest(USArrests_data, ntrees = 10, nthreads = 1)
```

Results and analysis:

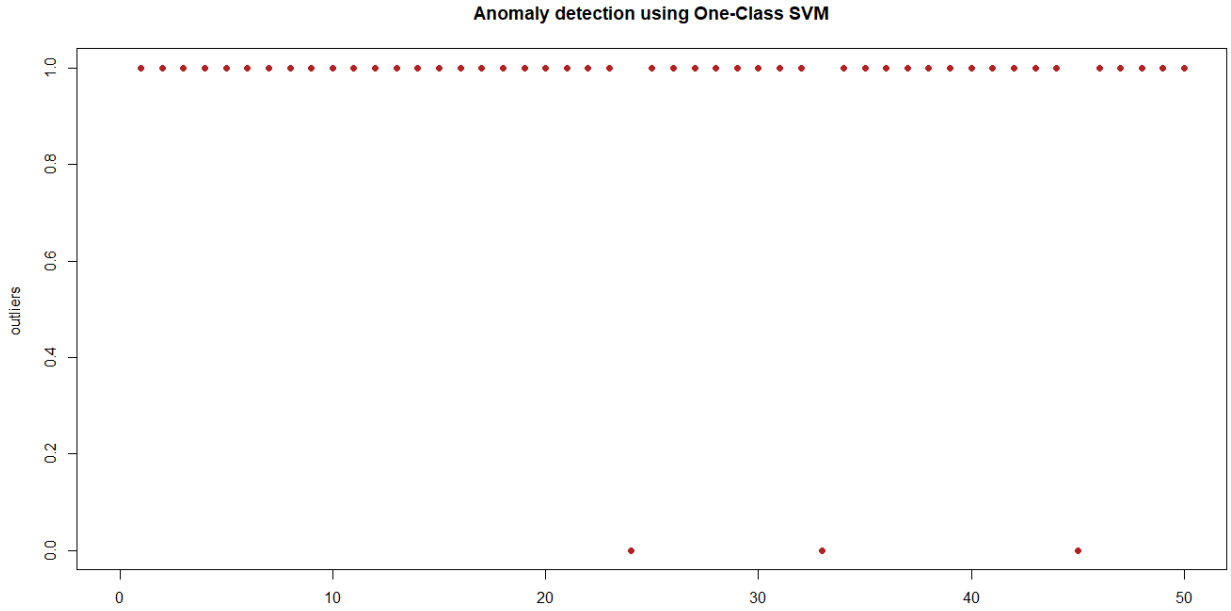
We start our analysis by observing the scatterplot, correlation, and covariance of each independent predictor variable with others (Fig, 6). Based on figs 5 and 6, we can see a significant positive correlation among predictor variables, with the highest correlation (0.80) amongst the feature murder against assault, as shown below.



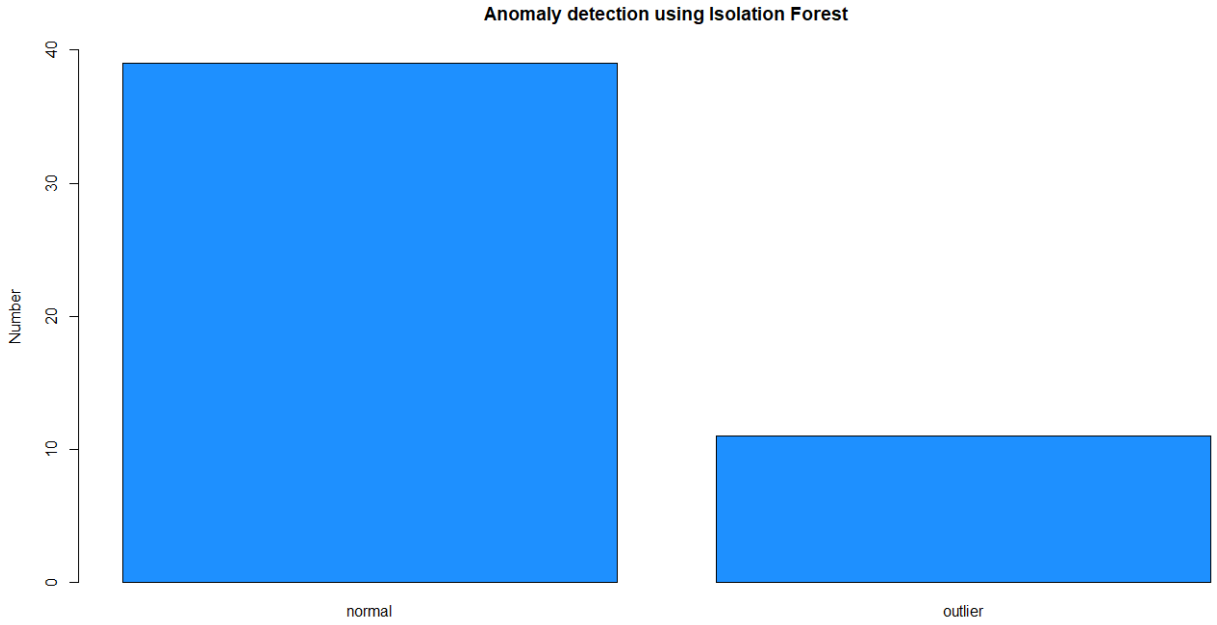
The following results are divided according to each of the methods above. The first is a figure implicating the results obtained from the kernel density estimation. Class 0 represents the outlier (59), and class 1 represents the normal points (453) in the data set. The y-axis represents the total number of densities computed in the data set. The summary statistics show that the anomaly percentage is **13.4**%.



The second is a figure depicting the results achieved from the one-class SVM. The y-axis represents the outliers computed in the data set. The X-axis represents the 50 states. The summary statistics show that the anomaly percentage is **6.38**%, with the states (of **Mississippi, North Carolina, and Vermont**) as the only three anomalies in this data set of 50 observations.



The last figure replicates the results achieved from the Isolation Forests. The y-axis represents the states given in the data set. The X-axis represents the two levels: outliers (11) and normal (39). The summary statistics show that the anomaly percentage is **28.2**%, with the states **(Alaska, Arizona, California, Florida, Nevada, New York, Rhode Island, South Carolina, West Virginia, Mississippi, and North Carolina)** as the eleven anomalies in this data set of 50 observations.



A cross-study of these results shows that the degree of anomalies across these unsupervised techniques is in the following ascending order: One-Class SVM, Kernel Density Estimation, and Isolation Forests. This implication depends on the anomaly percentage results, which ultimately depend on how these models are set up. Any changes in the model could vary the nature of the results.

Lastly, the results signify that the states attributed as anomalies may be where crime rates are significantly high or low. This result is crucial as it could call for strict policy measures to account for such high crime rates. Furthermore, it may also suggest which of the 50 states is better at controlling the crimes of rape, murder, and assault.

Conclusion:

The unsupervised anomaly detection techniques in this report are used to find anomalies in the data set. The data set had no missing values, and the boxplot outliers were not excluded from the study. The correlation matrix signifies that there is no perfect collinearity amongst any of the predictor variables. Once these unsupervised techniques were applied, the results imply that the degree of anomalies across these unsupervised techniques is in the following ascending order One-Class SVM, Kernel Density Estimation, and Isolation Forests. The model's results could have been further improved by pruning the model for better estimation methods that give better results. The significant correlation among predictor variables has also impacted the results. Future studies may look over other unsupervised techniques to see their efficacy in achieving a similar effect.

Figures:

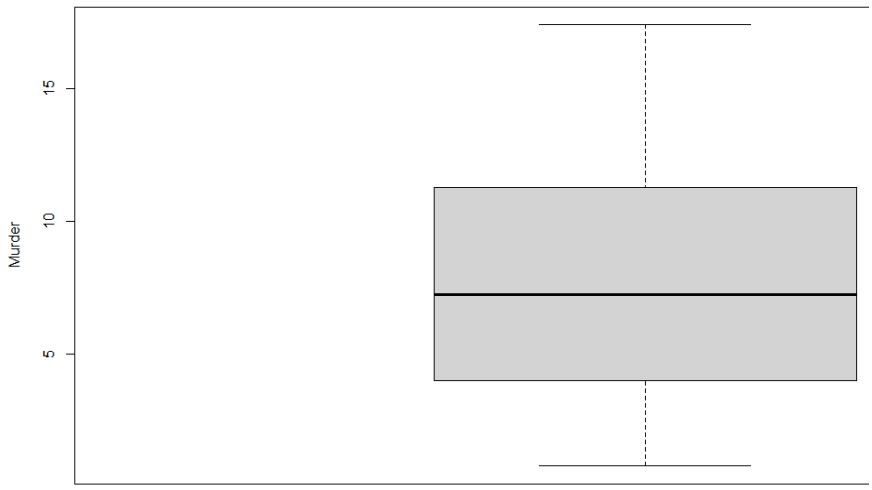


Fig 1: Boxplot of Murder

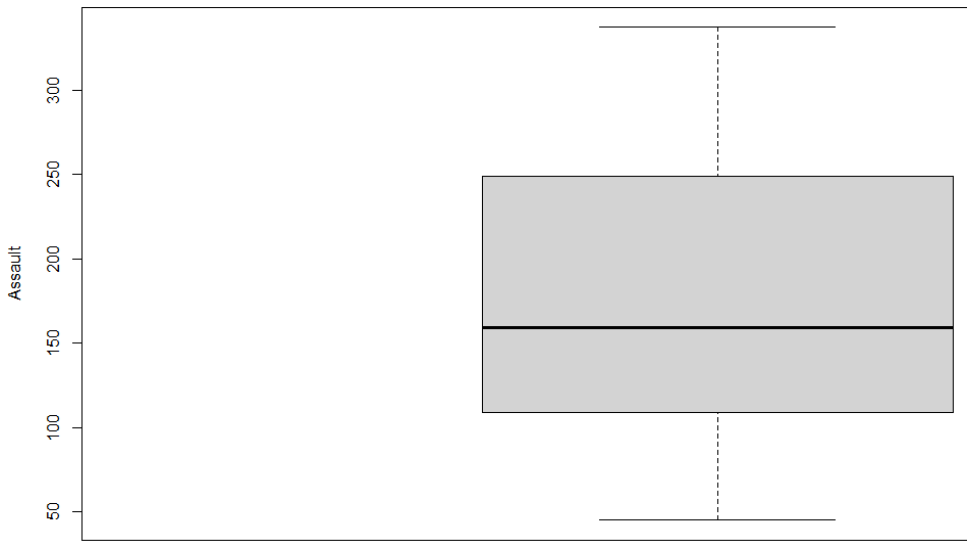


Fig 2: Boxplot of Assault

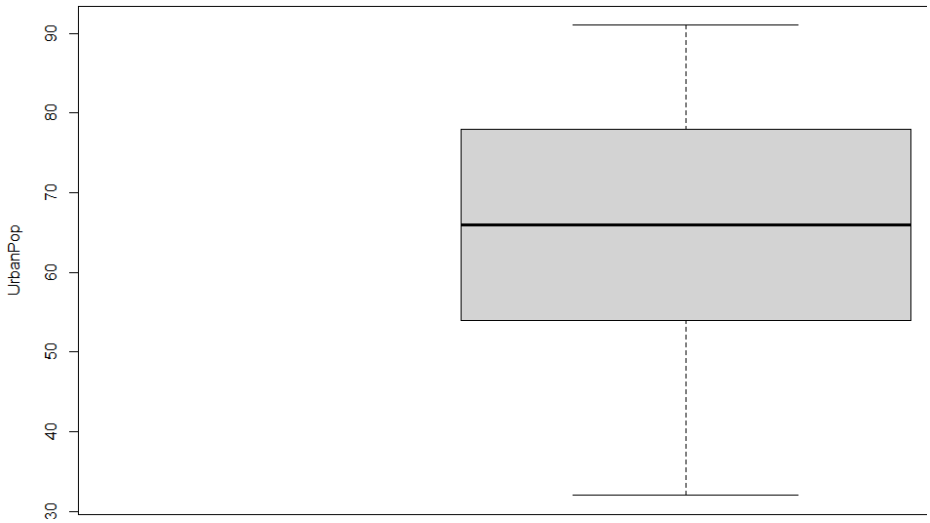


Fig 3: Boxplot of UrbanPop

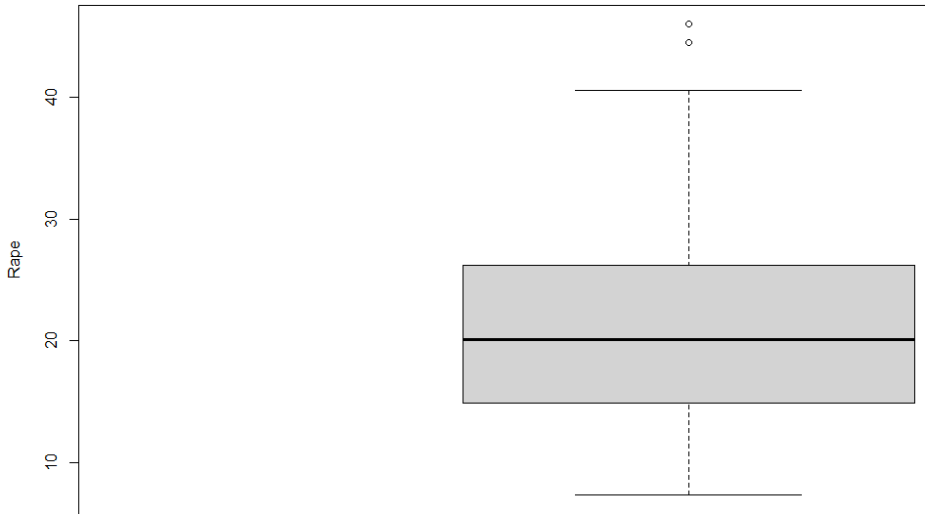


Fig 4: Boxplot of Rape

Fig 5: Correlation matrix

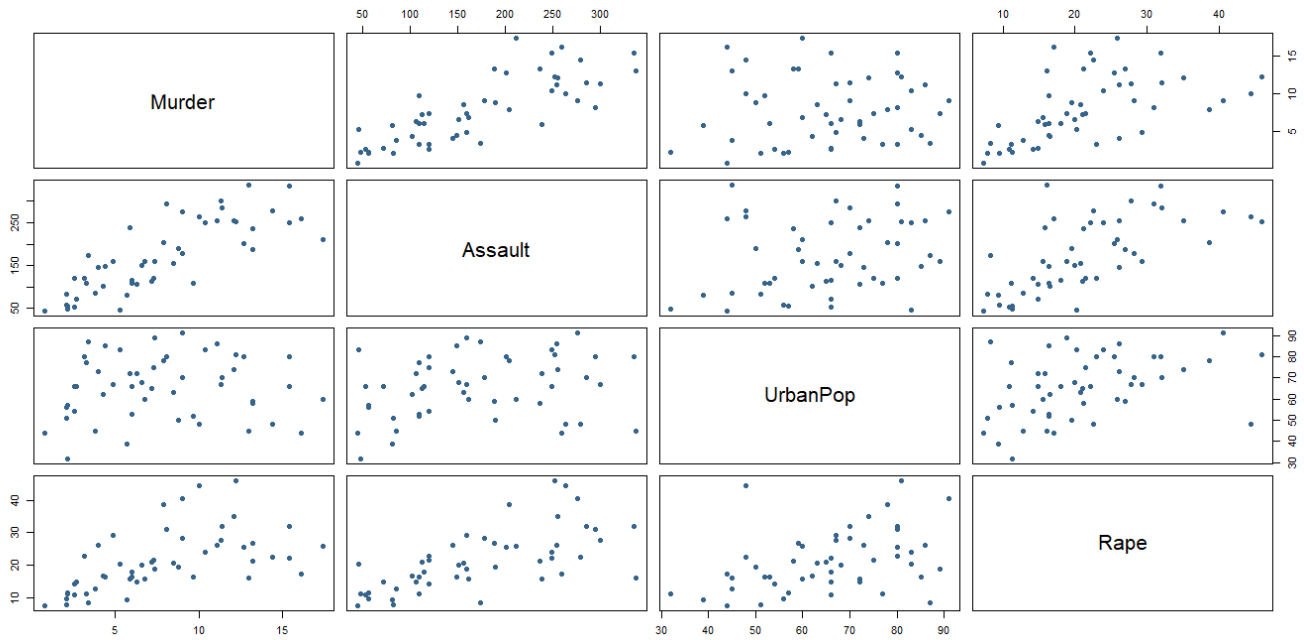


Fig 6: Scatterplot of predictor variables

References:

https://www.rdocumentation.org/packages/stats/versions/3.6.2/topics/density

https://pro.arcgis.com/en/pro-app/2.9/tool-reference/spatial-analyst/how-kernel-density-works.htm

https://www.rdocumentation.org/packages/e1071/versions/1.7-11/topics/svm

https://www.rdocumentation.org/packages/isotree/versions/0.5.15/topics/isolation.forest

R code:

```
# libraries
library(MASS)
library(corrplot)
library(ggplot2)
library(e1071)
library(isotree)
```

```
#install.packages('corrplot')

# exploring the data set
USArrests_data <- USArrests
View(USArrests_data)
str(USArrests_data)

nrow(USArrests_data)
head(USArrests_data)
na.omit(USArrests_data)

# summary, boxplot, and sd of each predictor variable
par(mar=c(3,4,4,3.5))
summary(USArrests_data['Murder'])
boxplot(USArrests_data['Murder'], ylab= 'Murder')
sd(as.matrix(USArrests_data['Murder']))

summary(USArrests_data['Assault'])
boxplot(USArrests_data['Assault'], ylab= 'Assault')
sd(as.matrix(USArrests_data['Assault']))

summary(USArrests_data['UrbanPop'])
boxplot(USArrests_data['UrbanPop'], ylab= 'UrbanPop')
sd(as.matrix(USArrests_data['UrbanPop']))

summary(USArrests_data['Rape'])
boxplot(USArrests_data['Rape'], ylab= 'Rape')
sd(as.matrix(USArrests_data['Rape']))
```

```
# correlation matrix

corrplot(cor(as.matrix(USArrests_data)), method = 'number')


# plots of each predictor variables

g <- ggplot()

g <- g + theme_bw()

g <- g + geom_point(data = USArrests_data, aes(Murder, Rape), color='brown')

g <- g+ggtitle('Murder vs Rape')+theme(plot.title = element_text(size=15,
face="bold", margin = margin(10, 0, 10, 0)))

g <- g + labs(x = "Murder", y="Rape") +theme(text =element_text(size=15))

g


par(mar =c(1,1,1,1))

plot(USArrests_data, col='steelblue4',pch=19)


# kernel density function

USArrests_mat <- as.matrix(USArrests_data[,1:4])

USArrests_mat

USArrests_den <- density(USArrests_mat)

USArrests_den


df_density <- as.data.frame(USArrests_den$y)

df_density


min_density <- min(df_density$`USArrests_den$y`)

mean_density <- mean(df_density$`USArrests_den$y`)

bench <- 0.20*min_density/mean_density

bench
```

```
df_density$outlier <- ifelse(df_density$`USArrests_den$y`< bench,0,1)

densities <- df_density

densities$outlier <- as.factor(densities$outlier)


s <- summary(densities$outlier)

s

par(mar=c(3.5,4,4,3.5))

plot(densities$outlier, main="Anomaly detection using kernal density ftn",

    xlab="Levels", ylab="Number ", pch=19, col=hcl(c(0, 120, 240), 50, 70), ylim=c(0,500))


anomaly_percent <- 100*s['0']/s['1']

anomaly_percent


# One-Class SVM

model_oneclasssvm <- svm(USArrests_data,type='one-classification',kernel =
"radial",gamma=0.05,nu=0.05)

model_oneclasssvm


pred_oneclasssvm <- predict(model_oneclasssvm,USArrests_data)

pred_oneclasssvm

s1 <- summary(pred_oneclasssvm)

s1


plot(pred_oneclasssvm, main="Anomaly detection using One-Class SVM",

    xlab="Cities", ylab="outliers ", pch=19, col='firebrick', xlim=c(0,50))

s2 <- as.integer(s1)

anomaly_percent_1 <- 100*s2[2]/s2[3]

anomaly_percent_1
```

```
# Isolation Forest

iforest <- isolation.forest(USArrests_data, ntrees = 10, nthreads = 1)


USArrests_data$pred <- predict(iforest, USArrests_data, type = "score")

USArrests_data$outlier <- as.factor(ifelse(USArrests_data$pred >=0.50, "outlier",
"normal"))

USArrests_data$outlier


s3 <- summary(USArrests_data$outlier)

s3

plot(USArrests_data$outlier, main="Anomaly detection using Isolation Forest",

    xlab="levels", ylab="Number", pch=19, col='dodgerblue', ylim=c(0,40))


s4 <- as.integer(s3)

s4

anomaly_percent_2 <- 100*s4[2]/s4[1]

anomaly_percent_2
```