Project Report

Analysis of Cyber Security Breaches Data

Name : Modem Praveen

Branch: Computer science and engineering

Reg No : 17BCS035

Introduction

Data Breach - A data breach comes as a result of a cyberattack that allows cybercriminals to gain unauthorized access to a computer system or network and steal the private, sensitive, or confidential personal and financial data of the customers or users contained within.

In this project we are going to analyse and visualise "Cyber Security Breaches Data" and we will conclude few solutions.

Exploring dataset

Name_of_Covered_Entity	State	Business_Associate_Involved	Individuals_Affected	Date_of_Breach	Type_of_Breach	Location_of_Breached_Information	Date_Posted_or_Updated	Summary	breach_start	breach_end	year
108	45	NA	1000	194	12	41	43	6	32	NA	2009
521	25	NA	1000	766	12	31	32	77	22	NA	2009
31	1	NA	501	179	12	37	1	NA	31	NA	2009
334	8	NA	3800	238	5	17	1	26	29	NA	2009
435	5	NA	5257	776	12	1	1	37	24	NA	2009
217	5	NA	857	776	12	1	1	34	24	NA	2009
519	5	NA	6145	776	12	1	1	36	24	NA	2009
413	5	NA	952	776	12	1	1	38	24	NA	2009
488	5	NA	5166	776	12	1	1	35	24	NA	2009
170	5	NA	5900	776	12	17	1	23	24	NA	2009
814	39	NA	943	208	12	17	1	NA	33	NA	2009
180	44	NA	6400	176	12	17	23	27	30	NA	2009
873	35	70	83000	258	11	41	1	80	40	NA	2009
426	5	NA	596	226	12	35	1	NA	36	NA	2009
424	28	177	2000	367	1	6	1	NA	47	NA	2009
232	23	NA	10000	213	12	36	1	NA	34	NA	2009
232	23	NA	646	290	12	18	1	17	43	NA	2009
880	5	NA	610	766	11	12	1	NA	22	NA	2009
215	20	NA	1860	319	12	38	1	73	50	NA	2009
492	20	NA	1076	253	12	35	1	NA	39	NA	2009
97	8	184	3400	218	12	41	43	113	35	NA	2009
97	8	135	15000	233	12	41	24	114	28	NA	2009
418	5	NA	15500	313	12	37	1	NA	45	NA	2009

Dimensions : 1055 x 14 - 1055 Rows and 14 columns

Column names

```
> colnames(x)
 [1] "X"                          "Number"                     "Name_of_Covered_Entity"
 [4] "State"                      "Business_Associate_Involved" "Individuals_Affected"
 [7] "Date_of_Breach"             "Type_of_Breach"             "Location_of_Breached_Information"
[10] "Date_Posted_or_Updated"     "Summary"                    "breach_start"
[13] "breach_end"                 "year"
```

Structure

```
> str(x)
'data.frame':   1055 obs. of  14 variables:
 $ X                          : int  1 2 3 4 5 6 7 8 9 10 ...
 $ Number                     : int  0 1 2 3 4 5 6 7 8 9 ...
 $ Name_of_Covered_Entity     : Factor w/ 967 levels " DeKalb Medical Center, Inc. d/b/a DeKalb Medical Hillandale",..: 108 521 31 554 435 237 519 413 488 170 ...
 $ State                      : Factor w/ 52 levels "AA","AL","AE",..: 45 25 1 8 5 5 5 5 5 5 ...
 $ Business_Associate_Involved: Factor w/ 232 levels "","3and Corporation",..: 1 1 1 1 1 1 1 1 1 1 ...
 $ Individuals_Affected       : int  1000 1000 501 3800 5357 667 6145 952 5166 5900 ...
 $ Date_of_Breach             : Factor w/ 600 levels "01/01/2006 - 31/12/2012",..: 134 766 179 238 776 776 776 776 776 776 ...
 $ Type_of_Breach             : Factor w/ 29 levels "Hacking/IT Incident",..: 12 12 12 5 12 12 12 12 12 12 ...
 $ Location_of_Breached_Information: Factor w/ 41 levels "Desktop Computer",..: 41 31 37 37 1 1 1 1 1 37 ...
 $ Date_Posted_or_Updated     : Factor w/ 43 levels "2014-01-23","2014-01-24",..: 43 32 1 1 1 1 1 1 1 1 ...
 $ Summary                    : Factor w/ 142 levels "","\n\nThe covered entity (CE), Medco Health Solutions, mailed letters with incorrect addresses after a program"| _
__truncated__,..: 6 77 1 26 37 34 35 38 36 23 ...
 $ breach_start               : Factor w/ 732 levels "1997-01-01","2002-05-06",..: 32 22 31 29 24 24 24 24 24 24 ...
 $ breach_end                 : Factor w/ 121 levels "2007-06-14","2011-02-28",..: NA NA NA NA NA NA NA NA NA NA ...
 $ year                       : int  2009 2008 2009 2009 2009 2008 2009 2009 2009 2008 ...
```


Above Figure shows the structure of the dataset.Type of every column is clearly mentioned.Here we can see that some of the columns have different-different levels …

Summary

```
> summary(x)
       X              Number                                                  Name_of_Covered_Entity     State          Business_Associate_Involved
 Min.   :   1.0   Min.   :   0.0   UnitedHealth Group health plan single affiliated covered entity:   7      CA     :111                          :784
 1st Qu.: 264.5   1st Qu.: 263.5   Cook County Health & Hospitals System                          :   6      TX     : 61   MedAssets              :   6
 Median : 528.0   Median : 527.0   University of California, San Francisco                        :   4      FL     : 66   StayWell Health Management, LLC :   5
 Mean   : 528.0   Mean   : 527.0   Walgreen Co.                                                   :   4      NY     : 58   Clearpoint Design, Inc.          :   4
 3rd Qu.: 791.5   3rd Qu.: 790.5   Baptist Health System                                          :   3      IL     : 49   Futurity First Insurance Group   :   4
 Max.   :1055.0   Max.   :1054.0   County of Los Angeles                                          :   3      TN     : 40   HealthPartners Administrators, Inc.:   3
                                   (Other)                                                        :1030      (Other):646   (Other)                          :249
 Individuals_Affected   Date_of_Breach                 Type_of_Breach                Location_of_Breached_Information Date_Posted_or_Updated
 Min.   :    500     1/11/2012:   7   Theft                        :516   Paper                        :227          2014-01-23:  31
 1st Qu.:   1000     8/24/2011:   7   Unauthorized Access/Disclosure:148   Laptop                       :217          2014-03-24:  48
 Median :   2300     3/10/2011:   6   Other                        : 91   Other                        :116          2014-04-21:  28
 Mean   :  30262     9/20/2013:   6   Loss                         : 85   Desktop Computer             :113          2014-04-23:  25
 3rd Qu.:   6541     9/27/2009:   6   Hacking/IT Incident           : 75   Network Server               :107          2014-02-12:  21
 Max.   :4900000     2/4/2010 :   5   Improper Disposal            : 58   Other Portable Electronic Device: 60          2014-08-18:  21
                     (Other)  :1018   (Other)                      :102   (Other)                      :215          (Other)   :221
```

```
   breach_start       breach_end        year
2012-06-15:   8   2012-10-01: 11   Min.   :1997
2011-06-24:   7   2012-10-27:  3   1st Qu.:2010
2012-01-11:   7   2012-11-15:  3   Median :2012
2009-09-27:   6   2012-04-02:  2   Mean   :2011
2011-03-10:   6   2012-09-21:  2   3rd Qu.:2013
2013-09-20:   6   (Other)    :124   Max.   :2014
(Other)   :1015   NA's       :910
```

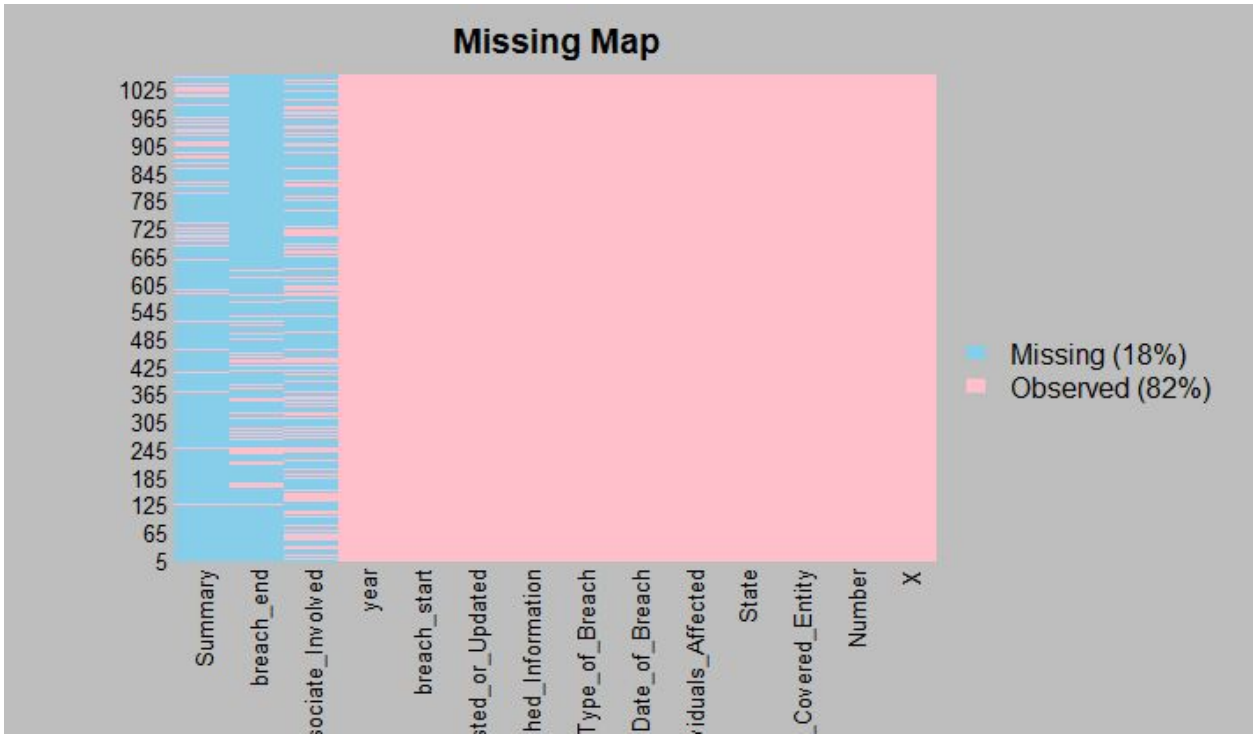
In the above figure the whole summary of the dataset is clearly mentioned.

Data Preprocessing

If we open the dataset ,we can clearly see that there are some missing values and NA values.So we have 2 ways here to clean that dataset. One way is filling those missing values manually(If very less values are missed in a column) and another alternative is removing that particular column(If more values are missed in a column).

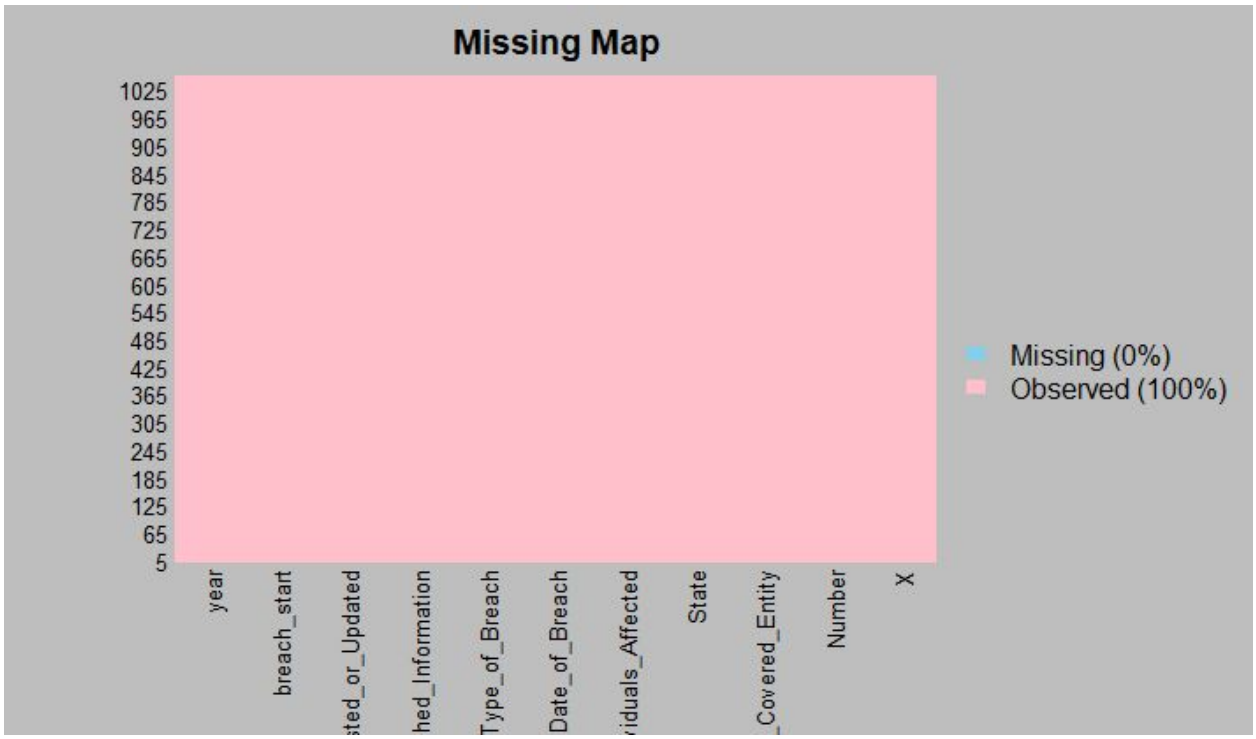
Firstly we are converting empty cells in the Csv file to NA.

Before:



Above plot shows that there are a lot of NA values in Summary,breach_end and Bussiness_Associate_Involved columns.Pink colour shows that no NA values. So we have to remove these columns.

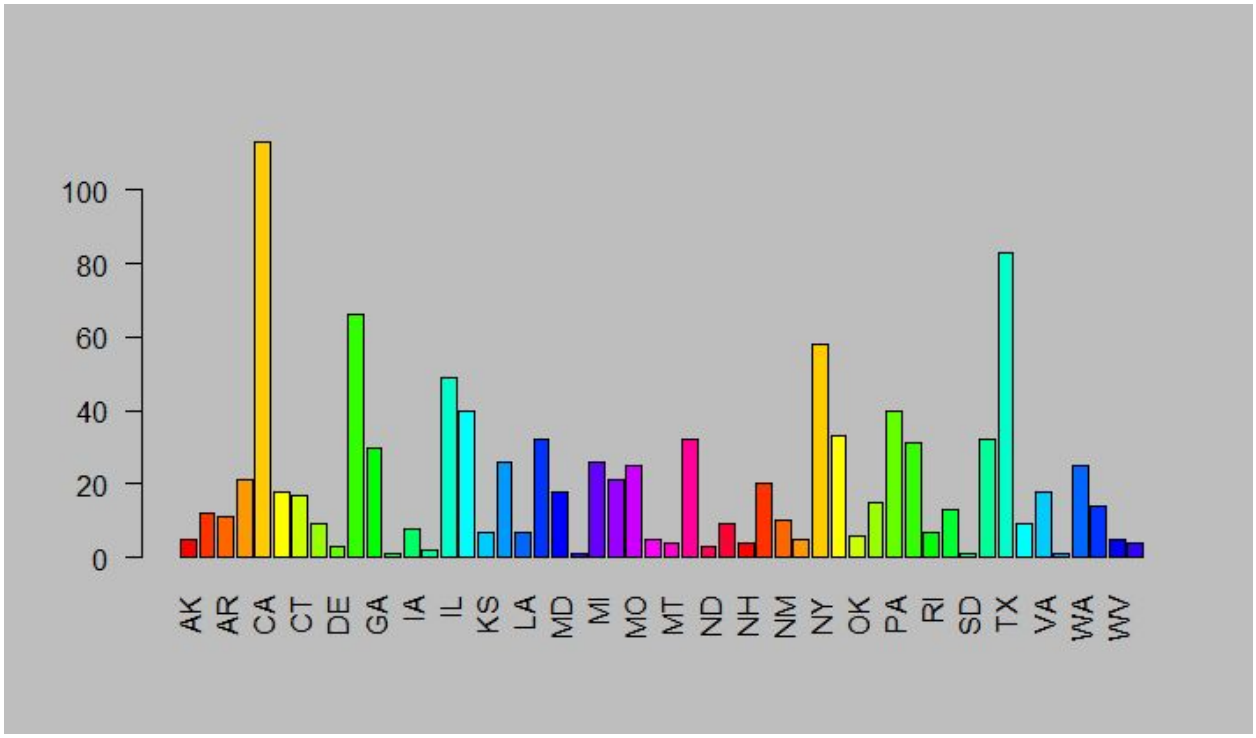
After:



After removing those columns here we can see that there is 0% of missing values

Data Visualisation

Bar Plots:

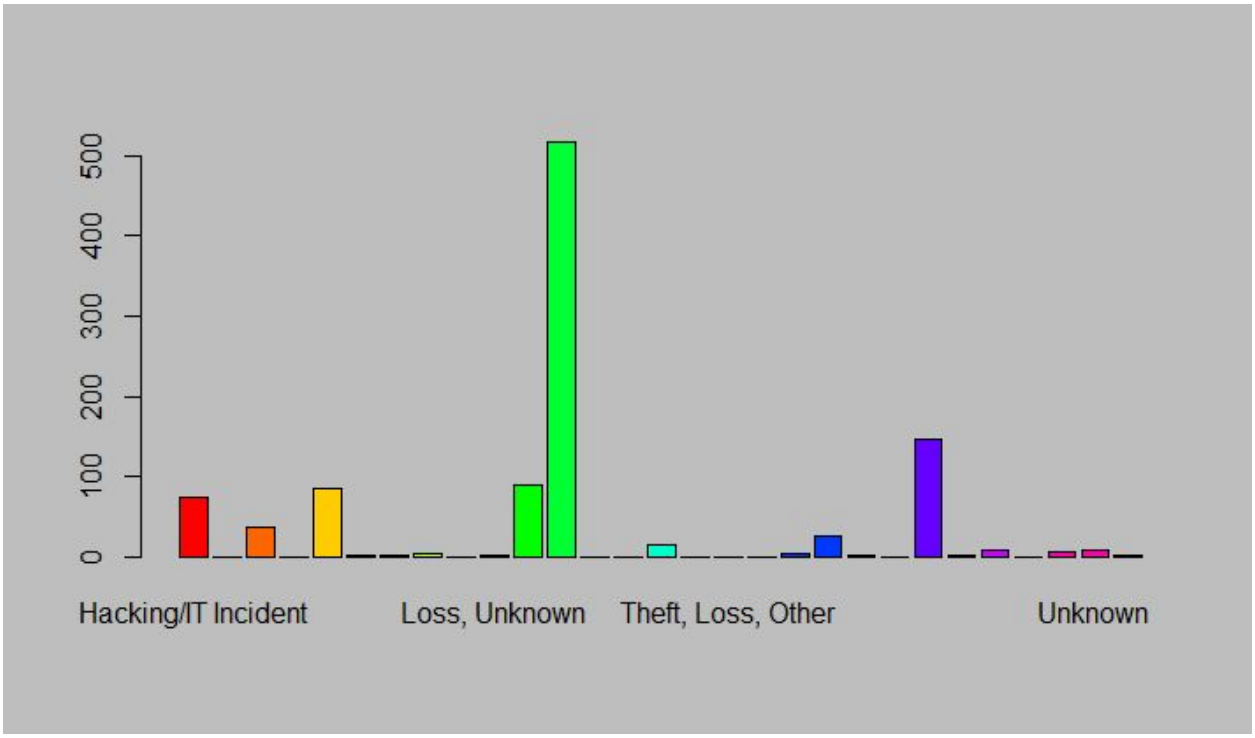


Above plots show the count of all states present in the dataset .

```
> count<-table(dataset1$State)
> count

 AK  AL  AR  AZ  CA  CO  CT  DC  DE  FL  GA  HI  IA  ID  IL  IN  KS  KY  LA  MA  MD  ME  MI  MN  MO  MS  MT  NC  ND  NE  NH  NJ  NM  NV  NY  OH  OK  OR  PA  PR  RI  SC  SD
  5  12  11  21 113  16  17   9   3  66  30   1   8   2  49  40   7  26   7  32  18   1  26  21  25   5   4  32   3   9   4  20  10   5  58  33   9  13  36   1   1   6   1
TN  TX  UT  VA  VT  WA  WI  WV  WY
32  83   9  16   1  25  14   5   4
```

Count



Above plots show the count of all “Type_of_breach” present in the dataset .

Note: There are many types are there in our dataset ,so because of limited space it didn’t plot labels properly . So everything is mentioned below in detail. But from the above plot we can see that some types are in high frequency and some are in very low frequency.

Count of Every breach type :

Hacking/IT Incident - 75

Hacking/IT Incident, Other - 1

Improper Disposal - 38

Improper Disposal, Unauthorized Access/Disclosure - 1

Loss - 85

Loss, Improper Disposal - 3

Loss, Other -2

Loss, Unauthorized Access/Disclosure - 5

Loss, Unauthorized Access/Disclosure, Unknown - 1

Loss, Unknown - 2

Other - 91

Theft - 516

Theft, Hacking/IT Incident - 1

Theft, Improper Disposal, Unauthorized Access/Disclosure - 1

Theft, Loss - 15

Theft, Loss, Improper Disposal - 1

Theft, Loss, Other - 1

Theft, Loss, Unauthorized Access/Disclosure, Unknown - 1

Theft, Other - 5

Theft, Unauthorized Access/Disclosure - 26

Theft, Unauthorized Access/Disclosure, Hacking/IT Incident -3

Theft, Unauthorized Access/Disclosure, Other - 1

Unauthorized Access/Disclosure - 148

Unauthorized Access/Disclosure - 2

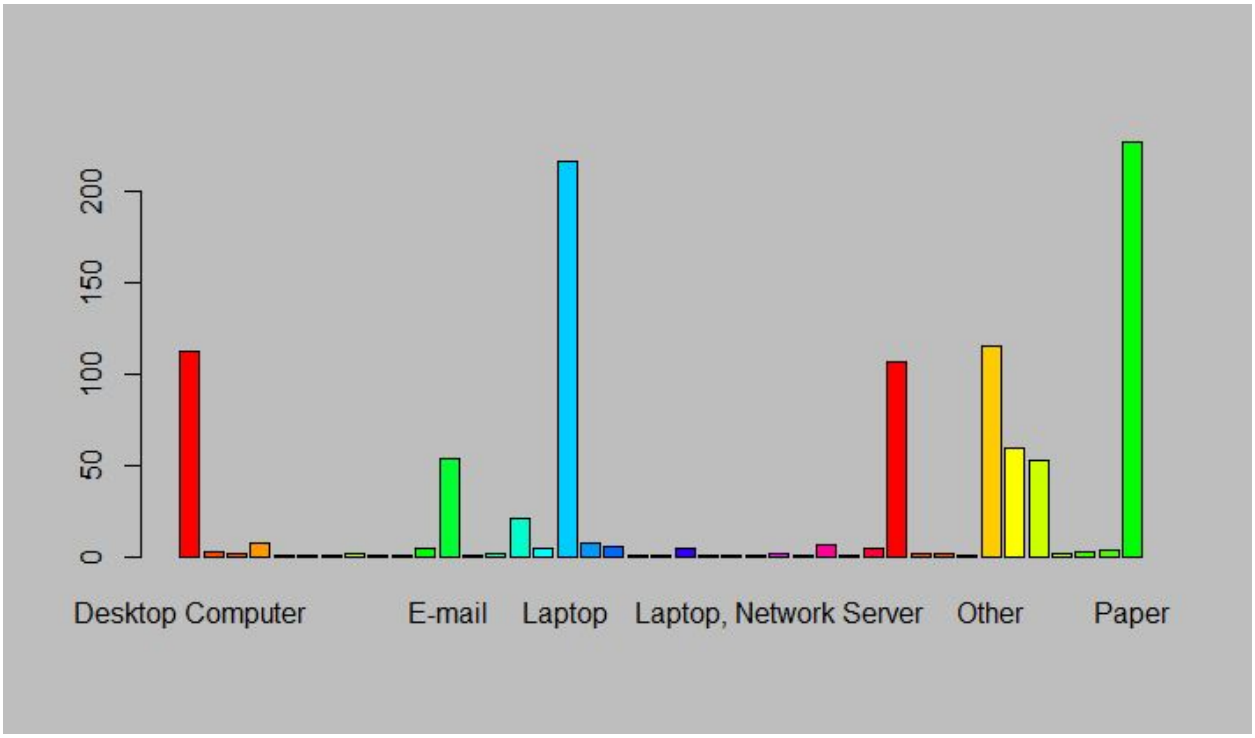
Unauthorized Access/Disclosure, Hacking/IT Incident - 9

Unauthorized Access/Disclosure, Hacking/IT Incident, Other - 1

Unauthorized Access/Disclosure, Other - 8

Unknown -10

Unknown, Other - 2



Above plots show the count of all “Location of breached information” present in the dataset .

Note: There are many locations are there in our dataset ,so because of limited space it didn’t plot labels properly . So everything is mentioned below in detail. But from the above plot we can see that some types are in high frequency and some are in very low frequency.

Count :

Desktop Computer - 113

Desktop Computer, E-mail - 3

Desktop Computer, Electronic Medical Record -2

Desktop Computer, Network Server- 8

Desktop Computer, Network Server, E-mail, Electronic Medical Record, Paper - 1

Desktop Computer, Network Server, Electronic Medical Record - 1

Desktop Computer, Network Server, Other Portable Electronic Device, Other - 1

Desktop Computer, Other - 2

Desktop Computer, Other Portable Electronic Device -1

Desktop Computer, Other Portable Electronic Device, Other - 1

Desktop Computer, Paper - 5

E-mail -54

E-mail, Other - 1

E-mail, Other Portable Electronic Device - 2

Electronic Medical Record - 21

Electronic Medical Record, Paper - 5

Laptop - 217

Laptop, Desktop Computer - 8

Laptop, Desktop Computer, Network Server, E-mail - 6

Laptop, Desktop Computer, Network Server, E-mail, Other Portable Electronic Device, Other, Electronic Medical Record - 1

Laptop, Desktop Computer, Network Server, E-mail, Other Portable Electronic Device, Other, Electronic Medical Record, Paper - 1

Laptop, Desktop Computer, Other Portable Electronic Device - 5

Laptop, Desktop Computer, Other Portable Electronic Device, Other - 1
Laptop, E-mail, Other Portable Electronic Device - 1

Laptop, Electronic Medical Record -1

Laptop, Network Server - 2

Laptop, Network Server, E-mail -1

Laptop, Other Portable Electronic Device - 7

Laptop, Other Portable Electronic Device, Paper - 1

Laptop, Paper - 5

Network Server - 107

Network Server, E-mail - 2

Network Server, Electronic Medical Record - 2

Network Server, Other - 1

Other - 116

Other Portable Electronic Device - 60

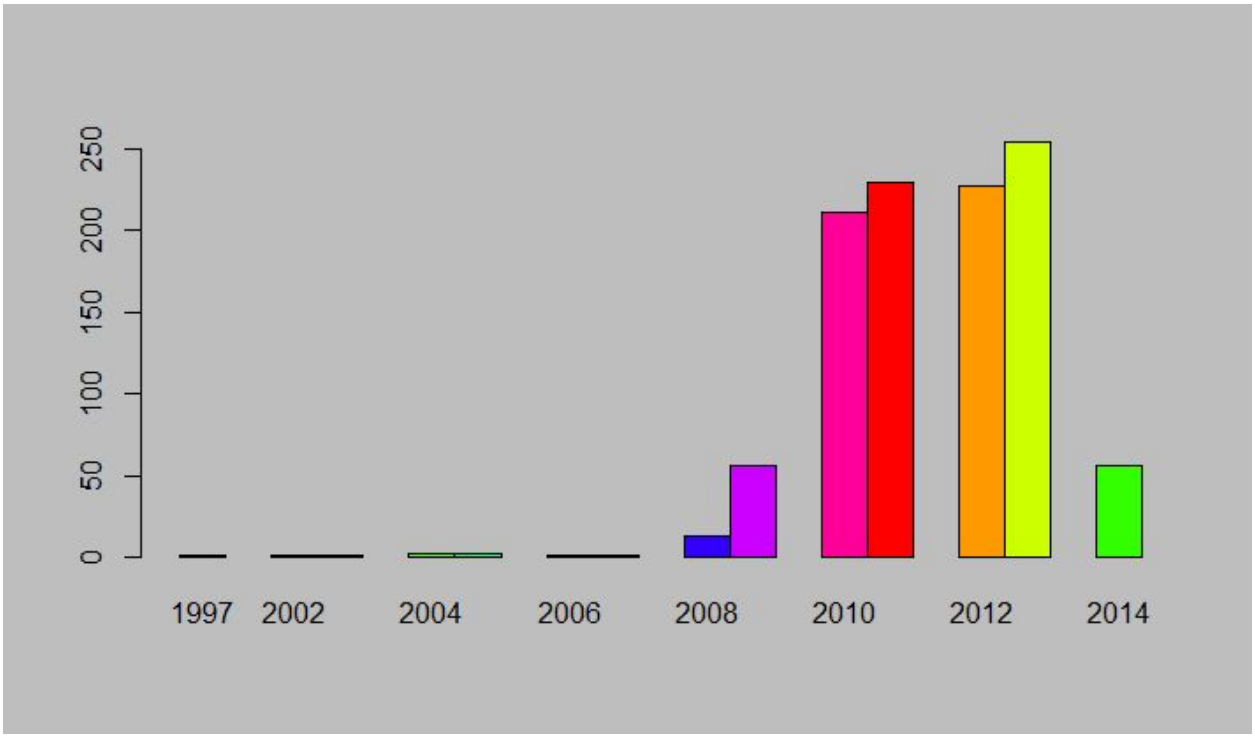
Other Portable Electronic Device, Other - 53

Other Portable Electronic Device, Other, Electronic Medical Record - 2

Other, Electronic Medical Record - 3

Other, Paper - 4

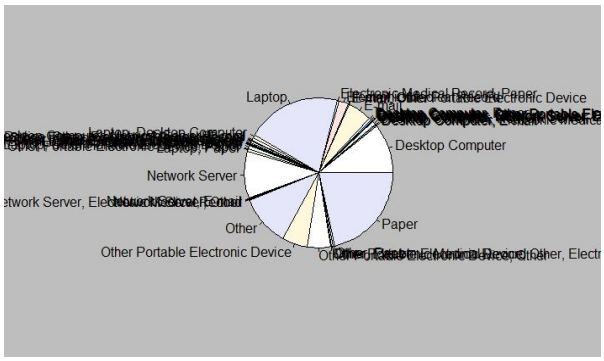
Paper - 227



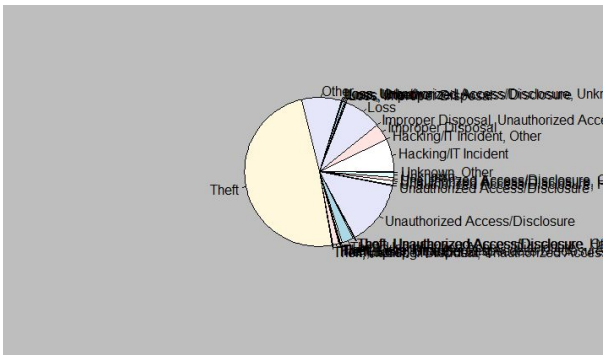
This plot shows the count of all “years” present in the dataset .

Here we can see that we have less 1997,2002,2004,2006 data

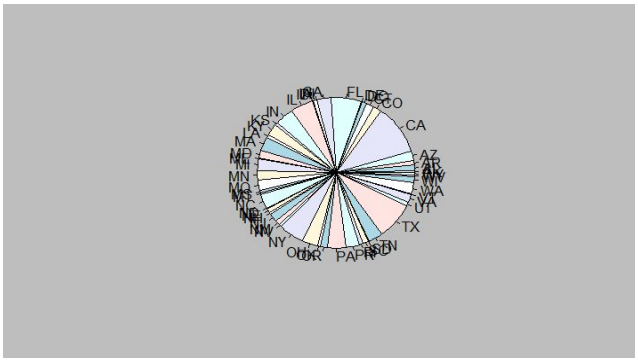
Pie chart



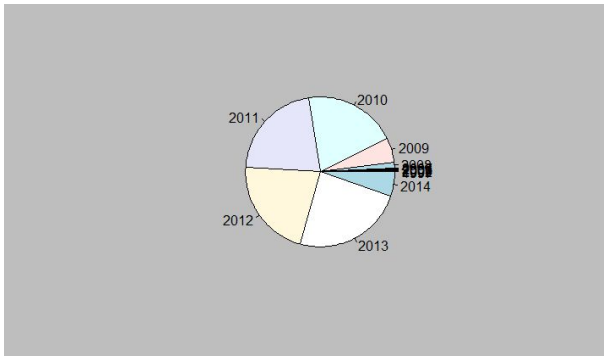
Location of Breached Information



Type of Breach



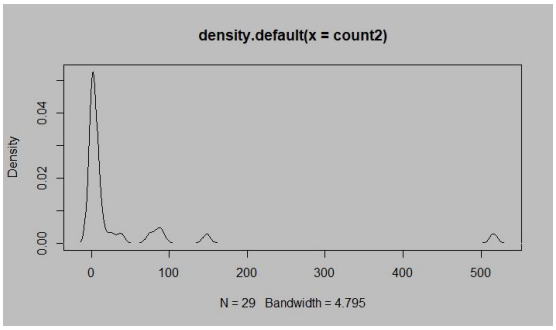
States



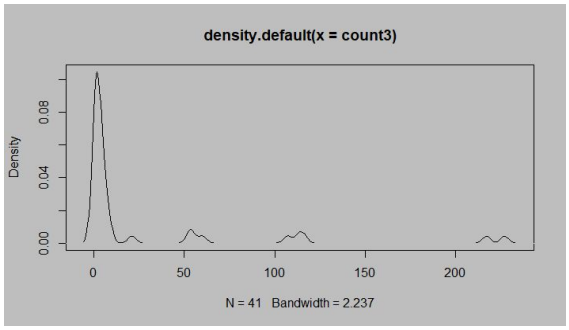
Year

Above plotted bar plots are plotted as Pie charts

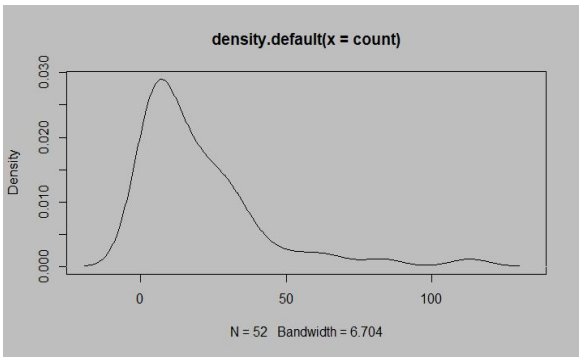
Density plots



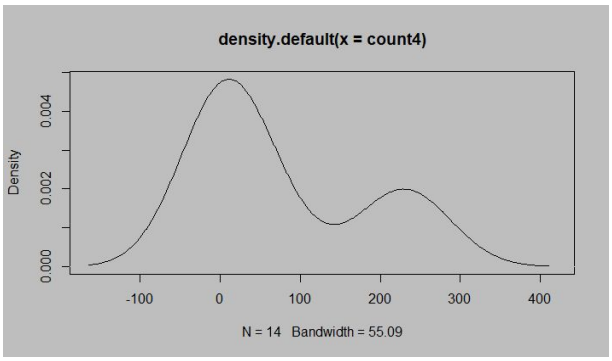
Type_of_Breach



Location_of_Breached_Information



State



Year

Above plotted bar plots are plotted as density plots

Applying Machine Learning

Now here we are going to apply some machine learning algorithms to Predict Type_of_breach and State using other columns.

Multinomial Logistic Regression

Firstly we should split the data in to train and test. For this there is a package called caTools.

```
library(caTools)
split <- sample.split(data, SplitRatio = 0.8)
train <- subset(data, split== "TRUE")
test <- subset(data, split== "FALSE")
```

Here we splitted data with 0.8 ratio

For Type of breach:

Now we should write a Logistic Regression model and have to train it with train data using the nnet package.

```
mymodel<-multinom(Type_of_Breach~.,data=train)
```

```
# weights:  312 (275 variable)
initial  value 2498.960045
iter  10 value 2282.797197
iter  20 value 1679.410580
iter  30 value 1643.052650
iter  40 value 1565.355437
iter  50 value 1530.626434
iter  60 value 1491.172184
iter  70 value 1470.323725
iter  80 value 1455.119522
iter  90 value 1444.846770
iter 100 value 1442.707979
final  value 1442.707979
stopped after 100 iterations
```

Here we took Type_of_Breach as Y and remaining data as X and trained the model with train data. Means we are going to predict Y with X.

Now the next step is predicting test data using our trained model.

```
predict(mymodel,test)
```

```
 [1] 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 12 5  3  12 12 12 5  12 12 12
[29] 12 3  12 3  1  12 12 12 12 3  12 12 3  3  3  3  12 12 12 12 12 12 3  12 12 3  3  3
[57] 12 12 12 12 12 12 3  12 12 12 12 12 12 12 12 12 12 1  23 12 12 12 12 12 12 12 12 5
[85] 12 5  1  12 12 23 12 23 12 5  12 12 5  12 3  5  5  12 12 12 12 12 5  3  12 3  12 12
[113] 1  12 5  12 12 12 12 12 5  5  12 12 12 12 12 12 12 12 5  23 5  23 15 5  5  12 12 12
[141] 12 12 12 23 12 12 12 5  5  12 12 12 5  12 1  12 5  5  23 12 12 12 12 12 5  12 12 12
[169] 12 12 12 23 23 12 12 12 12 12 23 12 12 12 12 12 12 12 12 12 12 12 23 5  5  12 12
[197] 12 12 23 5  5  12 12 23 12 12 12 12 1  12 5  23 12 12 12 23 23 23 19 12 23 12 12 12
[225] 12 23 23 12 12 12 12 12 12 1  12 12 23 12 12 12 12 12 12 5  23 5  12 12 12 12 12 23
[253] 23 12 12 5  12 5  23 23 12 11 23 12 23 23 8  12 12 12 23 12 12 12 12 12 12 12 12 12
[281] 23 23 12 21 23 12 12 12
.evels: 1 2 3 5 6 7 8 10 11 12 13 14 15 16 17 19 20 21 22 23 24 25 26 27 28 29
```

Above mentioned are predicted values

Predicted vs actual values table

```
table(predict(mymodel,test),test$Type_of_Breach)
print(cm)
```

	1	3	4	5	6	8	9	11	12	15	18	19	20	23	25	27	28	29
1	0	0	0	1	0	0	0	1	3	0	0	0	0	1	0	0	1	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	1	3	0	1	1	0	0	2	4	1	0	0	0	2	0	0	1	0
5	1	1	0	6	1	0	1	3	11	0	0	0	0	4	0	1	1	1
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	12	5	0	6	0	0	0	14	128	2	1	2	5	15	1	2	3	0
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
21	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
22	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
23	4	0	1	1	0	0	0	7	8	0	0	0	1	11	0	0	0	0
24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
25	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
26	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
27	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
28	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

From Above matrix we can see how many correct and wrong predictions.

Final step is finding overall accuracy .

```
acc=mean(predict(mymodel,test) != test$Type_of_Breach)
acc
```

```
> acc
[1] 0.4826389
```

We got 48.283% accuracy . As we know it is very low accuracy .There will be 2 reasons behind it . one is having unnecessary columns in X which don't depend on Y and another not having a perfect dataset .

In summary(model) we can see which column is giving more error and not dependent on Y all those things.

But in this case 0.482 is best accuracy.If we remove some columns in X then accuracy will be reduced

For State :

Till now we took Type_of_Breach as Y and remaining dataset as X . But now we are considering State as Y and we are going to predict it.

```
mymodel<-multinom(State~.,data=train)
accu=mean(predict(mymodel,test) != data$State)
accu
```

```
> accu
[1] 0.9090047
```

We found State using other columns . Here we can see that we got 90.09% accuracy.

Naive Bayes

It is implemented using the e1071 package. Here we are going to apply Naive Bayes to Predict Type_of_breach and State using other columns

For Type of breach :

Firstly we should train our model .

```
model<-naiveBayes(x=c(train$State+train$Individuals_Affected+
                      train$ Location_of_Breached_Information+
                      train$year),y=train$Type_of_Breach)
```

Above code shows we are training our model by considering Type_of Breach as Y and State, year , Individuals_Affected , Location_of_Breached_Information as X.

Next step is we should predict and find accuracy:

```
a=predict(model,newdata = train$Type_of_Breach)
```

```
acc=mean(train$Type_of_Breach != a)
acc
```

```
> acc
[1] 0.5084746
```

From the above image we can see that we got 50.8% accuracy.

For State :

Now we are going to consider State as Y and other columns as X and also before we predicted with test data ,but we can predict with insample data too.

So here we are going to predict with insample data.

```
train$State=as.factor(train$State)
test$State=as.factor(test$State)
model<-naiveBayes(x=c(train$Type_of_Breach+train$Individuals_Affected+
                      train$Location_of_Breached_Information+
                      train$year),y=train$State)
```

Here we trained our data with X to predict Y(state).

```
a=predict(model,newdata = train$State)
```

```
acc=mean(train$State != a)
acc
```

```
> acc
[1] 0.9335072
```

We can see that we got 93.33% accuracy using this model

Decision tree

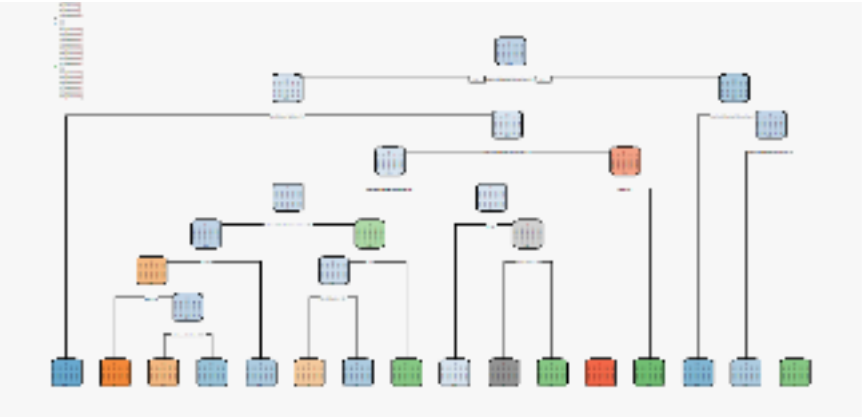
Here we are going to apply Decision tree to Predict Type_of_breach and State using other columns

For Type of breach :

Firstly we should train our dataset to Decision tree model.

```
DC<-rpart(Type_of_Breach ~ State+year+Individuals_Affected+
            Location_of_Breached_Information,
          data=train,method="class")
```

Decision tree model.Here Y is Type_of_breach



This above picture is plot of our trained model

Now we should predict Y value and find accuracy

```
fitval<-predict(DC,newdata = test,type="class")
```

```
acc<-mean(fitval != test$Type_of_Breach)
acc
```

```
> acc
[1] 0.4791667
```

For State :

```
DC<-rpart(State~Type_of_Breach+year+Individuals_Affected+
            Location_of_Breached_Information,
          data=train,method="class")
```

Here we wrote a Decision tree model and trained with train data.

```
fitval<-predict(DC,newdata = test,type="class")
```

Here we predicted test data using trained model

```
acc<-mean(fitval != test$State)
acc
```

```
> acc
[1] 0.9166667
```

We got 91.7% accuracy using the decision tree model.

K Means clustering

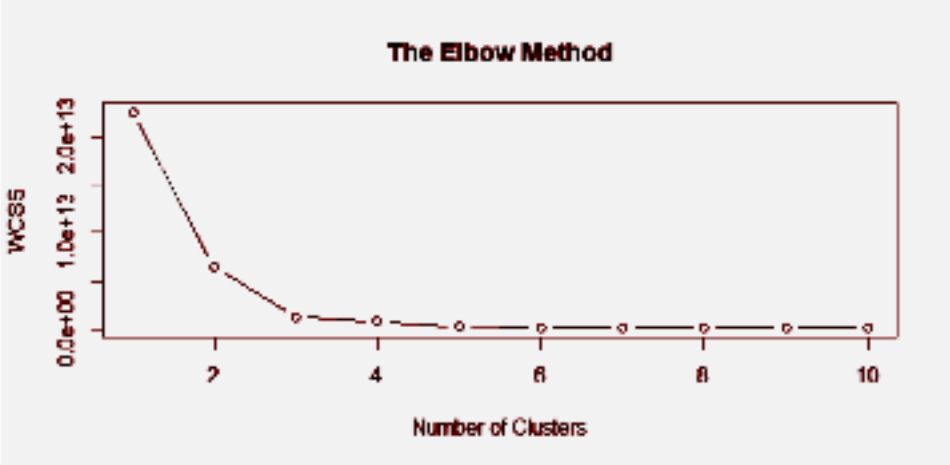
Here we are going to apply K Means Clustering Model to find clusters

Firstly we should train our dataset to the K Means Clustering model.

```
kmeans.result <- kmeans(test, centers = 3, nstart = 20)
kmeans.result$cluster
```

K Means clustering model

Here we are using the ELBOW method to set the number of clusters.

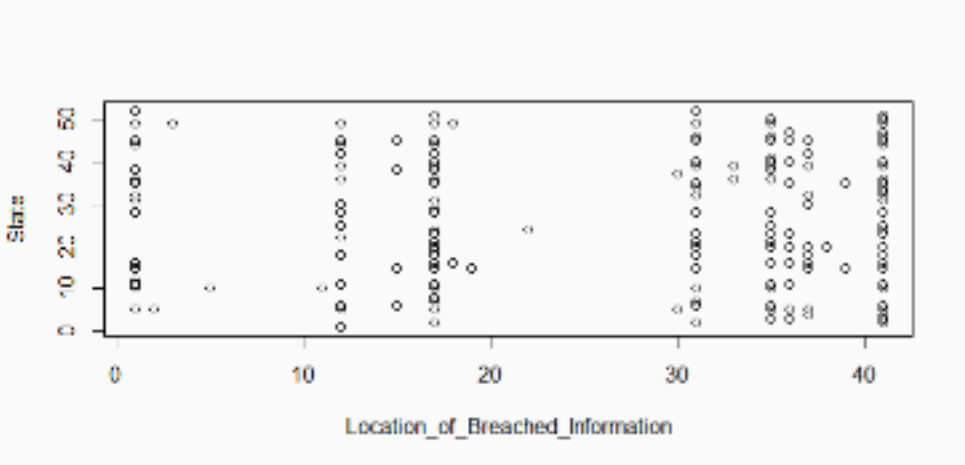


From the above figure we can see 3 is the best value. After 3 there is less fall of curve.

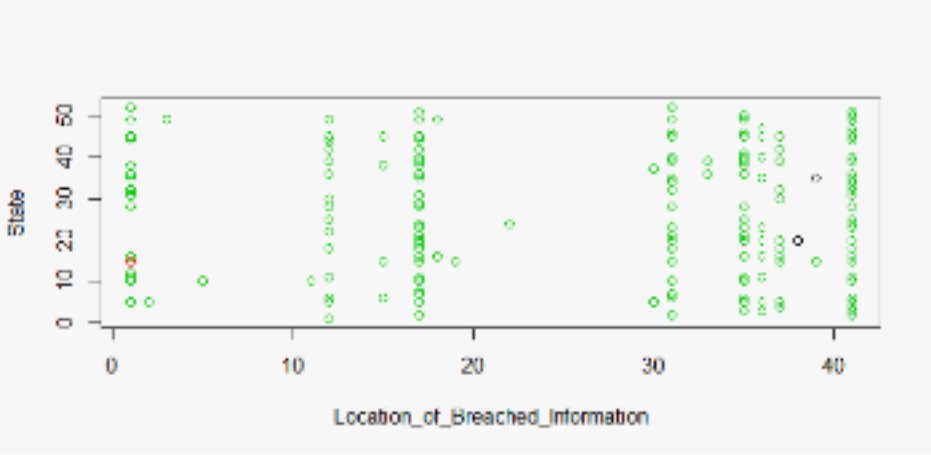
Plots of Clusters:

This below plot is Location of breached information VS State and plotted by taking number of clusters=3

Before:

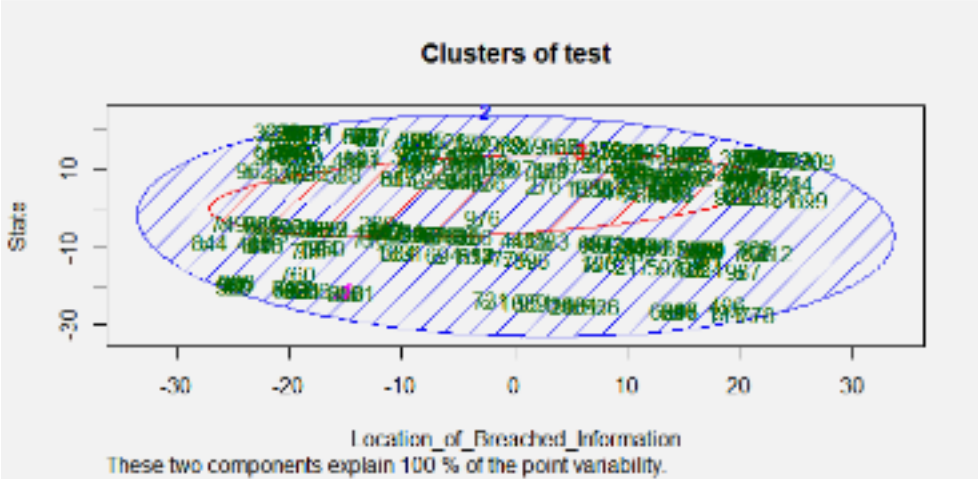


After:

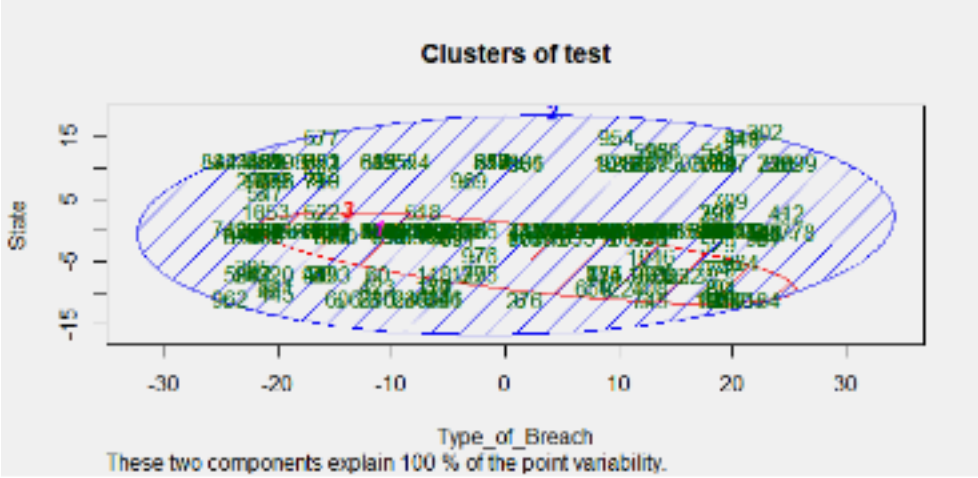


In the above plot we can see that almost all points are in the green cluster and left side 1 point in red and right side a few points in black.

So it is very difficult to find proper clusters with this dataset.



Location of Breached Information VS State



Type of breach VS State

Conclusion :

From this analysis and Model predictions ,I am concluding that If we get Type of breach,Individuals affected,year ,Location of breached information data then we can predict the state easily with approx 90% accuracy.

And also we can predict Type of breach using ,Individuals affected , year ,Location of breached information,State data ,but only with 50% accuracy(our models and data is not suitable to predict Type of breach properly).

In the below url they mentioned about every column . Through my analysis and predictions .We can reduce breaches .

For example : Location_of_Breached_Information - Desktop Computer

Type of breach - theft

So from the above information I can predict the state easily .CA is that state in the US region . So those particular states can focus on those types of breaches .And they can use their technology to prevent theft in that CA.

Reference

https://perspectives.ahima.org/cyber-analytics-identifying-discriminants-of-data-breaches/