ALY6000- Final Project

Exploratory Data Analytics
of
Boston Fire Department Data

Submitted by-
Jeseeka Shah
NUID-002134289

Introduction

The exploratory data analysis is for indicating the number of call that the Boston fire department had during the month of September in the city of Boston and its neighbourhood. The most frequent cause behind the fire alarm and how many of them were fake. This analysis will also point out the area that is safe and has less fire-based accidents. This analysis will considering various factors to point out the best neighbour in Boston city. The dataset was collected from the Boston government website and consisted of 4696 rows and 24 columns. After cleaning the dataset there where 19 features remaining of the 24 features.

The dataset had number accidents recorded but we were considered few of the most generic once in them all. After considering the below conditions the dataset rows got narrowed down to 916 rows with 19 columns.

Incident Type	Incident Description
111	Building fire
113	Cooking fire, confined to container
410	Combustible/flammable-gas/liquid condition, other
735	Alarm system sounded due to malfunction
745	Alarm system activation, no fire - unintentional

<u>Exploratory Data Analysis</u>

<u># Installation of packages</u>

STEP 1:
```
library("FSA")
library("FSAdata")
library("magrittr")
library("dplyr")
library("tidyr")
library("ggplot2")
install.packages("plyr")
install.packages("tidyverse")
library("plyr")
library("tidyverse")
```

<u>#Step 2: Loading the file and cleaning the dataset</u>
```
fire_safety<-read.csv("C://Users//Jessica Shah//Desktop//ALY6140//ALY6000_Project//fireincidentreportingseptember2021.csv",header = T,na.string="")
```
<u>#Step 3: Getting the Summary of the dataset before the alteration of the data frame</u>
```
summary(fire_safety)
```

```
Incident.Type     Incident.Description
Min.   :111.0     Length:4696
1st Qu.:553.0     Class :character
Median :611.0     Mode  :character
Mean   :600.8
3rd Qu.:733.0
Max.   :911.0
```

```
Neighborhood
Length:4696
Class :character
Mode  :character
```

View(head(fire_safety,5))

Incident.Type	Incident.Description	Estimated.Property.Loss	Estimated.Content.Loss	District	City.Section	Neighborhood
113	Cooking fire, confined to container	0	0	7	DO	Dorchester
735	Alarm system sounded due to malfunction	0	0	7	DO	Dorchester
611	Dispatched & cancelled en route	0	0	6	RX	Roxbury
710	Malicious, mischievous false call, Other	0	0	9	RX	Roxbury
500	Service Call, other	0	0	11	BR	Allston-Brighton

View(tail(fire_safety,5))

Alarm.Date	Alarm.Time	Incident.Type	Incident.Description	Estimated.Property.Loss	Estimated.Content.Loss	District	City.Section	Neighborhood
9/30/2021	22:23:14	553	Public service	0	0	3	BO	Boston
9/30/2021	22:44:01	552	Police matter	0	0	11	BR	Allston-Brighton
9/30/2021	22:47:31	745	Alarm system activation, no fire - unintentional	0	0	9	JP	Jamaica Plain
9/30/2021	22:58:22	736	CO detector activation due to malfunction	0	0	3	CH	Charlestown
9/30/2021	23:36:41	745	Alarm system activation, no fire - unintentional	0	0	4	BO	Boston

#Step 4:-Dropping the columns that are not required

fire_safety[, c('XStreet.Type', 'XStreet.Suffix','Address.2','XStreet.Name','XStreet.Prefix')]<- list(NULL**)**

#Step5:- Checking the structure of the value

str(fire_safety)

#Step 6 – Carrying out descriptive analysis

fire_safety$Incident.Type <- as.integer(fire_safety$Incident.Type)

```
> typeof(fire_safety$Incident.Type)
[1] "integer"
> class(fire_safety)
[1] "data.frame"
```

#Step 7-Choosing the values as described by Filtering the columns and rows

#Using the Filter function

Filtering_rows= fire_safety %>% filter(Incident.Type==111 | Incident.Type==113 | Incident.Type==410 | Incident.Type==735 | Incident.Type==745)

View(Filtering_rows)

```
> class(Filtering_rows)
[1] "data.frame"
```

This way we will be choosing the Incident type and then choosing the operation to be performed on the new data frame which will contain only 19 features for the operations and visualizations. Now, lets check the count of the unique incident type that we have chosen. The figure below shows the count of a particular incident that has occurred.

	Var1	Freq
1	111	28
2	113	185
3	410	2
4	735	215
5	745	483

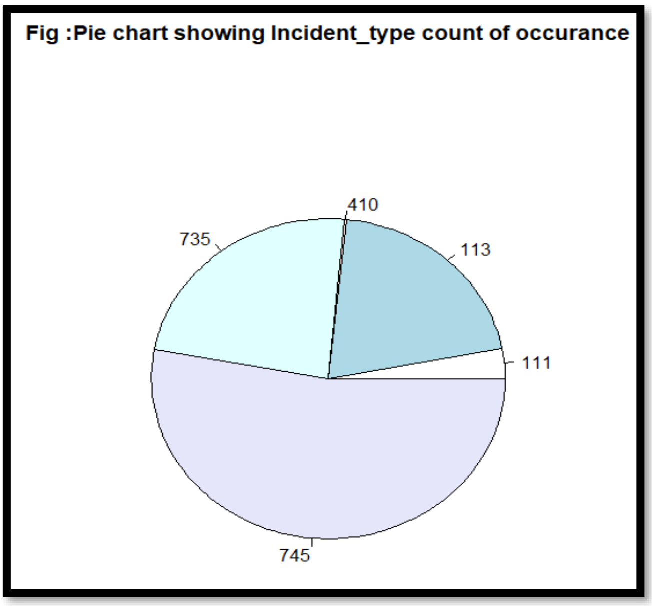


Figure: Showing the Incident_type values that are considered for the obervation

After this we will use the group by function to group the columns and to establish the relation between different values that are present in the dataframe. I have used group by initially to club Incident_type, Incident_Description and count of the occurance of these values(Figure below).

	Incident.Type	Incident.Description	count
1	111	Building fire	28
2	113	Cooking fire, confined to container	185
3	410	Combustible/flammable gas/liquid condition, other	2
4	735	Alarm system sounded due to malfunction	215
5	745	Alarm system activation, no fire - unintentional	483

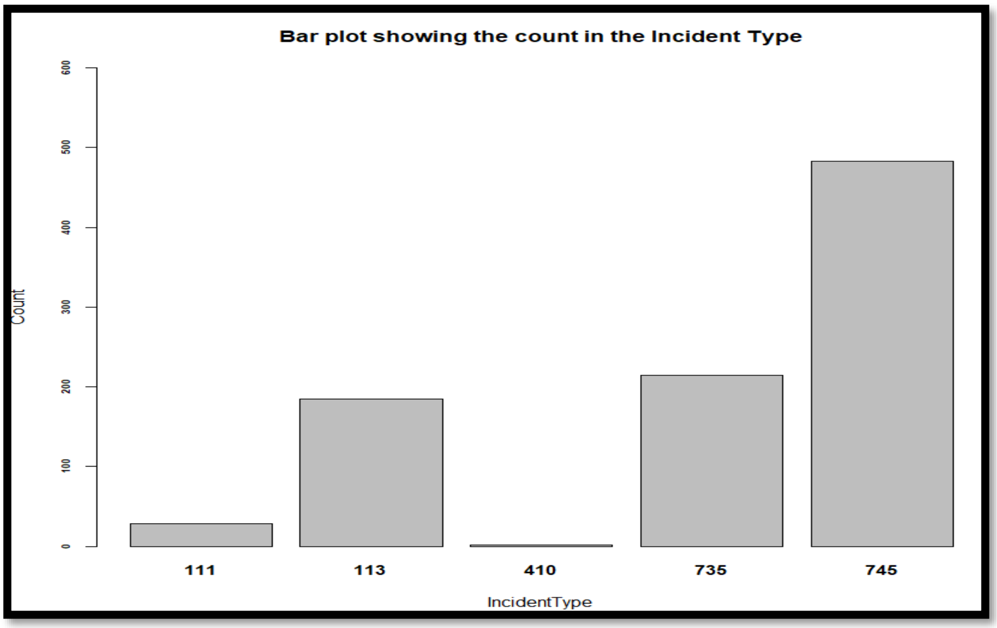
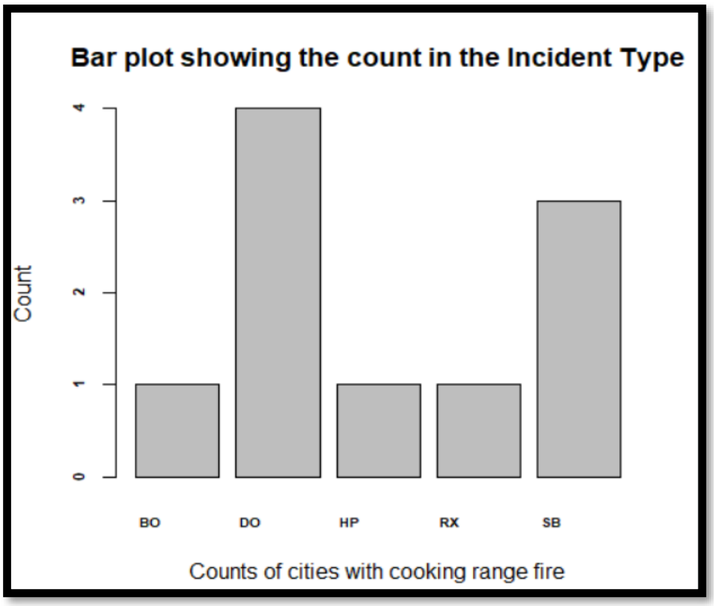


Figure: Bar graph point at the count of Incident type that happened in the month of September

#8 Grouping the city bases on the cooking based fire alarms

Grouping_neighbour= Filter_Cooking_based_fire %>% group_by(City.Section) %>% summarise(count=n())%>% arrange(City.Section)
View(Grouping_neighbour)



	City.Section	count
1	BO	1
2	DO	4
3	HP	1
4	RX	1
5	SB	3

Figure: Bar plot showing cities with cooking incident leading to fire

#9)Cost of Property loss

Grouping_neighbour_Propertyloss= Filter_Cooking_based_fire %>% group_by(City.Section) %>% summarise(Cost=sum(Estimated.Property.Loss))%>% arrange(City.Section)
View (Grouping_neighbour_Propertyloss)
plot4 = boxplot (Grouping_neighbour_Propertyloss$Cost,col = "blue", ylab = "Cost", xlab = "Fire based on cooking practises",ylim = c(100,12000),main = "Plot4: Boxplot to identify district outlier")

	City.Section	Cost
1	BO	10
2	DO	11050
3	HP	10
4	RX	1
5	SB	85

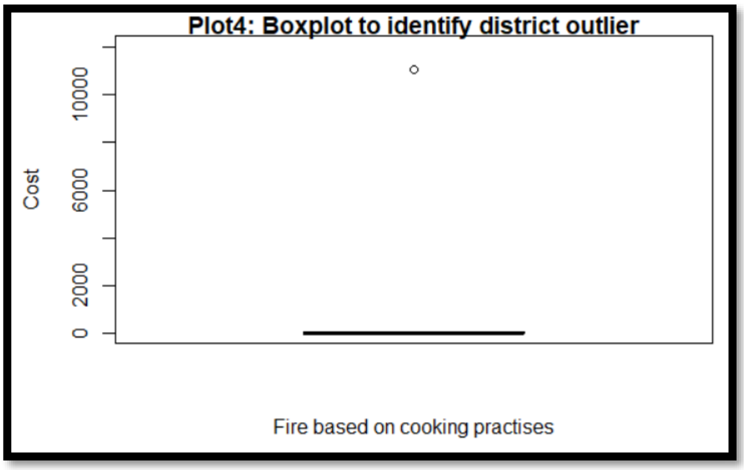
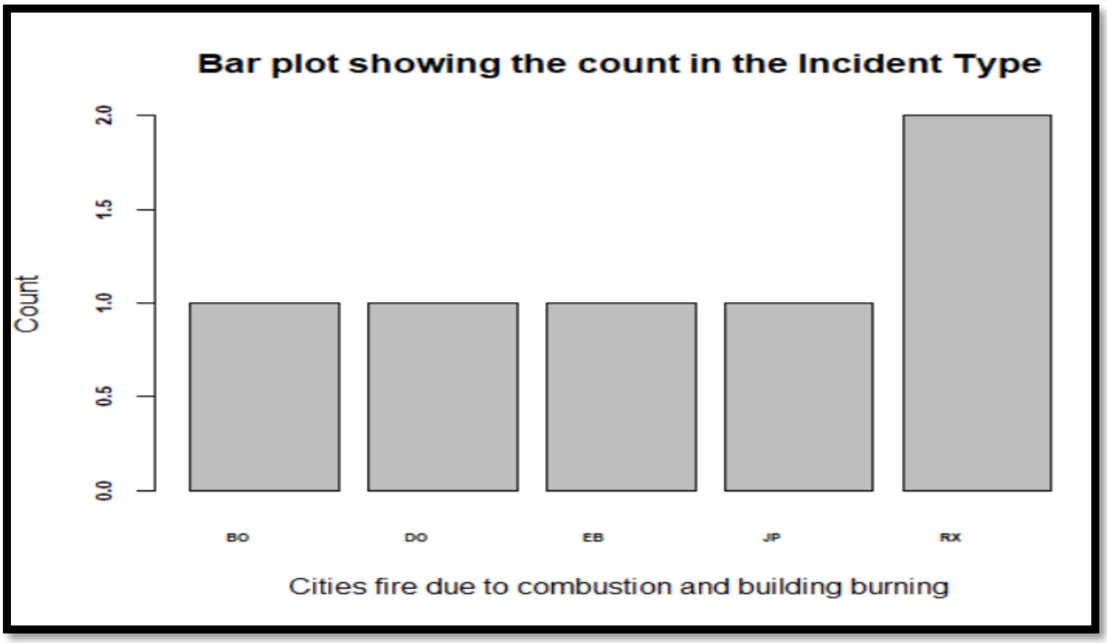


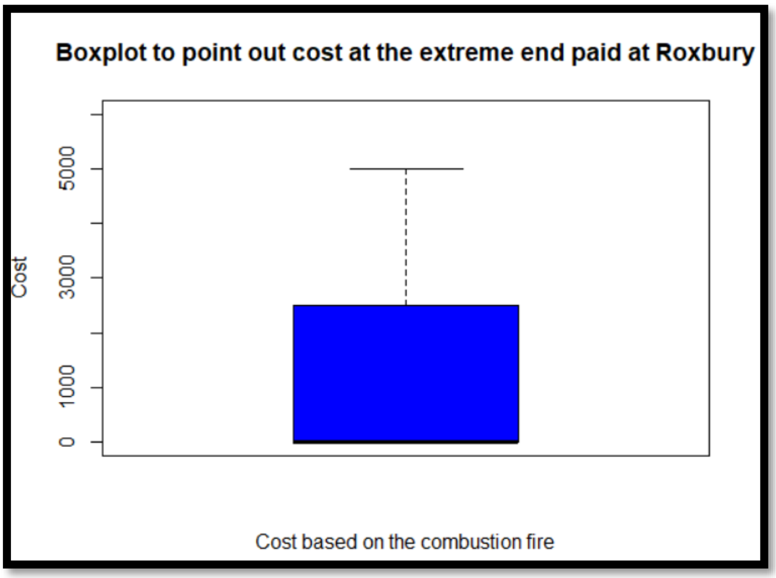
Figure: Showing the cost at Dorchester as the outlier based on the property loss by cooking based fire

From the above table, we can see that property loss due to cooking based fire is more at the **Dorchester** neighbourhood rather **Roxbury** area as the frequency of occurrence is also more.

#10)Grouping Based on the Burning and Combustible Flame

Considering that the risk is combustion caused and building fire in the month of September 2021, the analysis showed that Roxbury area has the maximum time fire alarm approaching as compared to other places. While plotting this graph I considered the Estimated Content Loss as the factor with respect to the cities.





	City.Section	Cost
1	BO	0
2	DO	5000
3	EB	0
4	JP	0
5	RX	2500

Figure : Above shows the contest loss estimation cost by the combustion and building fire

#11)Considering the Alarm system activation, no fire - unintentional (Code=745)

Here, I have considered the values based on the district of boston that had generated the false alarm when there was no fire. The values are as below and the graphs shows the same point that these system needs to be checked and repaired.

	District	Count
1	3	69
2	4	112
3	6	41

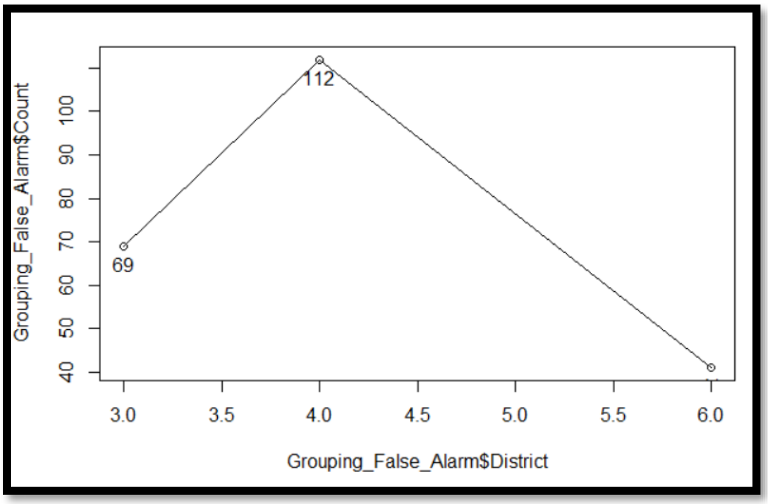


Figure: Shows the False alarm caused by the district 4 in Boston is more compared to others

Conclusion

The exploratory data analysis showed us that the fire at **Roxbury** was more when it come about combustion and building fire and the cost estimation was also the highest. The cooking based fire alarm that was raised by the city **Dorchester** was the at the peak in the month of September as compared to Roxbury (as it was the least). Furthermore, the Boston district 4 has the highest false followed by district 3 and 6.

Reference: - Of the dataset used in the Exploratory Data Analytics

Fire Incident Reporting - Fire Incident Reporting September 2021 - Analyze Boston. (n.d.). Boston Fire Department. https://data.boston.gov/dataset/fire-incident-reporting/resource/ffdced8b-d453-4f8b-a38a-82b72b5c2ba5

Appendix

STEP 1B: Importing the packages.

```
library("FSA")
library("FSAdata")
library("magrittr")
library("dplyr")
library("tidyr")
library("ggplot2")
install.packages("plyr")
install.packages("tidyverse")
library("plyr")
library("tidyverse")
install.packages("RColorBrewer")
install.packages("lessR")
library(lessR)
library(RColorBrewer)
```

#Step 2: Loading the file and cleaning the dataset

```
fire_safety<-read.csv("C://Users//Jessica Shah//Desktop//ALY6140//ALY6000_Project//fireincidentreportingseptember2021.csv",header = T,na.string="")
```

#Getting the Summary of the dataset before the alteration of the dataframe

summary(fire_safety)

```
View(head(fire_safety,5))
View(tail(fire_safety,5))
```

#Dropping the columns that are not required

```
fire_safety[ , c('XStreet.Type', 'XStreet.Suffix','Address.2','XStreet.Name','XStreet.Prefix')] <- list(NULL)
```

#Checking the Structure of the values

```
str(fire_safety)
```

#Step 7:Carrying out the descriptive statistics

#:Changing the type

```
fire_safety$Incident.Type <- as.integer(fire_safety$Incident.Type)
#Incident_type=count(fire_safety$Incident.Type)
typeof(fire_safety$Incident.Type)
class(fire_safety)
```

#Filtering the rows and columns as per the Incident_Type

```
Filtering_rows= fire_safety %>% filter( Incident.Type==111 | Incident.Type==113 | Incident.Type==410 | Incident.Type==735 | Incident.Type==745)
View(Filtering_rows)
class(Filtering_rows)
```

#Creating a table with Incident_type

```
Incidenttype_count = table(Filtering_rows$Incident.Type)
View(Incidenttype_count)
class(Incidenttype_count)
```

#Converting it to dataframe from a table

```
Incidenttype_count_df = data.frame(Incidenttype_count)
```

```
typeof(Incidenttype_count_df$Var1)
#Plotting the Incident_type count
par(mar=c(10, 7, 1, 1))
pie(Incidenttype_count_df$Freq,labels = Incidenttype_count_df$Var1,values = "%", axes=F,space=0.15,ylim = c(0,100),main = "Fig :Pie chart showing Incident_type count of occurance")
#plot1 = PieChart(Incidenttype_count_df$Freq, hole = 0, values = "%", data = finaldata,
          # col = brewer.pal(6,"Set1"), main = "Plot6: Pie chart dipicting Gender wise share")
#Using the grouping function
#Using Group by Function
Grouping_values= Filtering_rows %>% group_by(Incident.Type,Incident.Description) %>% summarise(count=n())%>% arrange(Incident.Type)
View(Grouping_values)
class(Grouping_values)
typeof(Filtering_rows$Incident.Description)
View(Filtering_rows)
#Plotting Figure 2
plot<-barplot(Grouping_values$count,xlab = "IncidentType",ylab ="Count",main="Bar plot showing the count in the Incident Type",cex.axis=0.6,font=2,ylim=c(0,600),names.arg =Grouping_values$Incident.Type)
#text(plot, Grouping_values$count, col = "black", cex = 0.8)
#Plot 3 showing the Fire alarm based on Incident_type as Cooking
#Now grouping based on the incidents and checking the street that have more issues recalled
#Filtering the Filter data set to derive insights
Filter_Cooking_based_fire=Filtering_rows %>% filter( Incident.Type==113 ,Estimated.Property.Loss >0 )
View(Filter_Cooking_based_fire)
#Grouping By Neighbourhood
Grouping_neighbour_Propertyloss= Filter_Cooking_based_fire %>% group_by(City.Section) %>% summarise(Cost=sum(Estimated.Property.Loss))%>% arrange(City.Section)
Grouping_neighbour= Filter_Cooking_based_fire %>% group_by(City.Section) %>% summarise(count=n())%>% arrange(City.Section)
View(Grouping_neighbour)
View(Grouping_neighbour_Propertyloss)
par(mar=c(10, 7, 1, 1))
#Plotting the cities and their count based on cooking accidents
plot<-barplot(Grouping_neighbour$count,xlab = "Counts of cities with cooking range fire",ylab = "Count", main="Bar plot showing the count in the Incident Type",cex.axis=0.7,font=2,cex.names = 0.6,names.arg =Grouping_neighbour$City.Section)
#plot1<-barplot(Grouping_neighbour_Propertyloss$Cost,xlab = "Estimated property loss in cities with cooking range fire",ylab = "Count", main="Bar plot showing the count in the Incident Type",cex.axis=0.7,font=2,cex.names = 0.6,names.arg =Grouping_neighbour_Propertyloss$Cost)
plot4 = boxplot(Grouping_neighbour_Propertyloss$Cost,col = "blue", ylab = "Cost", xlab = "Fire based on cooking practises",ylim = c(100,12000 ),main = "Plot4: Boxplot to identify district outlier")
```

#Grouping the function with the least and server 410 ANd 111 --> Building Fire and Combustion

#Filtering the Filter data set to derive insights

```
Burning_Combustible_Flame=Filtering_rows %>% filter( Incident.Type==410 | Incident.Type==111,Estimated.Property.Loss==0)
 View(Burning_Combustible_Flame)
Grouping_Burning_Combustion= Burning_Combustible_Flame %>% group_by(City.Section) %>% summarise(Cost=sum(Estimated.Content.Loss))%>% arrange(City.Section)
View(Grouping_Burning_Combustion)
Grouping_Filter_Combustible_Flame= Burning_Combustible_Flame %>% group_by(City.Section) %>% summarise(count=n()) %>% arrange(City.Section)
View(Grouping_Filter_Combustible_Flame)
par(mar=c(10, 7, 1, 1))
#z=plot(as.vector(x=Grouping_Filter_Combustible_Flame$count),y=Grouping_Filter_Combustible_Flame$Neighborhood,type="o",pos=1)
plot<-barplot(Grouping_Filter_Combustible_Flame$count,xlab = "Cities fire due to combustion and building burning",ylab = "Count", main="Bar plot showing the count in the Incident Type",cex.axis=0.7,font=2,cex.names = 0.5,names.arg =Grouping_Filter_Combustible_Flame$City.Section)
plot4 = boxplot(Grouping_Burning_Combustion$Cost,col = "blue", ylab = "Cost", xlab = "Cost based on the combustion fire",ylim = c(0,6000),main = "Boxplot to point out cost at the extreme end paid at Roxbury")
#pie(Grouping_Filter_Combustible_Flame$count,labels =Grouping_Filter_Combustible_Flame$Neighborhood,values = "%", axes=F,space=0.15,ylim =c(0,100),main = "Fig :Pie chart showing Incident_type count of occurance")
```

 # Grouping Based on the Alarm system activation, no fire - unintentional code 745

```
False_Alarm_based_fire=Filtering_rows %>% filter( Incident.Type==745, District==4 | District==3 | District==6)
View(False_Alarm_based_fire)
```

#**View(cITY_BOSTON)**

```
Grouping_False_Alarm= False_Alarm_based_fire %>% group_by(District) %>% summarise(Count=n())%>% arrange(District)
View(Grouping_False_Alarm)
False_Alarm=plot(x=Grouping_False_Alarm$District,y=Grouping_False_Alarm$Count, type="o")
text(x =Grouping_False_Alarm$District , y =Grouping_False_Alarm$Count, labels = Grouping_False_Alarm$Count, pos = 1, col = "black")
```