Analysis of Accidental Drug related Deaths in Connecticut

By

Srujan Kumar Goregey (G01247878)

Submitted to:

Harry Foxwell,

George Mason University,

AIT-580 : Analytics : Big Data to Information

Date: 05/10/2020

Abstract:

Significant rates of fatalities are correlated with illness and injuries induced by the worldwide use of drugs, narcotics. This study is a trend analysis on Connecticut's drug-related deaths from 2012-2018 and it also determines main contributing factors, draw assumptions that may both raise health understanding and encourage responsive entities to take proactive measures. Data consists of reports of toxicity, death certificate and scene of investigation which is derived from an investigation by the Office of Chief Medical Examiner.

The dataset has 5105 records with 41 attributes. By doing analysis of this dataset, I would like to know if there is an increase in drug related deaths over year, if there is seasonal impact on drug related deaths. Which age group, sex, race has more drug related deaths. Which combination of drugs causes more deaths and which city in Connecticut has more cases. This analysis also predicts the number of deaths due to drug overdose for next 12 months i.e. for year 2019. For predictive analysis, Linear regression is used.

Data cleaning is required to remove the NA values, duplicate values, rows which are not required for analysis. Out of 41 attributes we only need 25 attributes(ID', 'Date', 'Age', 'Sex', 'Race', 'DeathCity', 'DeathCityGeo', 'COD',' Heroin', 'Cocaine', 'Fentanyl', 'FentanylAnalogue', 'Oxycodone', 'Oxymorphone', 'Ethanol', 'Hydrocodone', 'Benzodiazepine', 'Methadone', 'Amphet', 'Tramad', 'Morphine_NotHeroin', 'Hydromorphone', 'Other', 'OpiateNOS', 'AnyOpioid') for analysis. So, we can narrow down the data set . The NA values in different drug attributes can be filled with 'N'. Since age, sex, race is necessary for our analysis we can drop the rows without these values.

After preprocessing the data, we can do analysis by plotting appropriate graphs. The above assumptions can be useful to some agency to decrease drug related deaths. Agencies could target the people of right age group, race , sex and city and take proactive measures to decrease the deaths due to drugs.

Introduction:

On average two people die every day in Connecticut from a drug overdose. Many people suffer from drugs here than from vehicle crashes or by shooting. There were about 2,000 deaths from drug overdose between 2012 and 2015 in Connecticut. There were 723 deaths registered in the year 2018, more than double the amount three years earlier. Nearly half a million people died from drug overdoses in the United States from 2000 to 2014, according to the Centers for Disease Control and Prevention. Since 2013 Connecticut has exceeded the national mortality rate for drug overdoses. Drug-induced mortality in Connecticut was the leading cause of death by accident in adults.

The main tasks in this analysis include data preparation, data analysis, selection and usage of significant features to forecast mortality levels for next year. Data analysis involves plots with scatter graphs, box plots, and other plots required to explain the results.

The data set was taken from data.gov[1] website which has 5105 records with 41 attributes.

Objectives

The main objectives of this analysis are following :

1. See if there is an increase in drug related deaths over year
2. See if there is seasonal impact on drug related deaths
3. Which age group, sex, race has more drug related deaths
4. What drugs causes more deaths and which city in Connecticut has more cases.
5. Predict number of deaths for 12 months after 2018.

Literature review:

Who is dying in Connecticut's opioid overdose crisis?

This article[2] gives an in-depth analysis of Connecticut's drug overdose issue. It compares cause of deaths by Drug overdose, car accidents and firearms for years 2012 to 2015. The study indicates that more people suffer from opioid overuse than car accidents in Connecticut. Drug-related fatalities have been close in amount to motor vehicle incidents since 1999. There was an increase in 2006 and 2007 but in 2010, the figure fell to a low of 357 fatalities

This article also compares drug overdose death rate by each state in US. According to the Centers for Disease Control and Prevention, nearly half a million individuals suffered from opioid abuse in the United States from 2000 to 2014. Since 2013 Connecticut has surpassed the national mortality threshold for alcohol and opioid deaths.

This study also gives good analysis about place of death. It says, there are more fatalities in homes but less in other areas such as the hotel house, parking lots. The study shows trend of Opioid prescriptions dispensed over time. Opioid medications are important for pain and disease treatment, but they can also contribute to dependency, violence, addiction, and eventually overdose. More than 60 percent of the 723 overdose fatalities in 2015 included opioids. Opioids are artificial medications developed to act in a similar manner to opiates. They include drugs such as oxycodone, methadone, hydrocodone, fentanyl and hydromorphone.

If there is tighter regulation of opioid pain medication then it pushes certain patients who misuse drugs to turn to cheaper, much more available, heroin. Heroin usage is through again, according to the report, just though drug misuse is dropping off. This study also shows us the increase in use of Heroin, Morphine, Fentanyl which are cheaper drugs than opioid from 2012 to 2015.

Gaps and conclusion:

This article has given good analysis on comparing type of death, most used drugs and comparing accident deaths but it does not give analysis on impact of gender, race, age group, seasonal impact.

Materials and Methods:

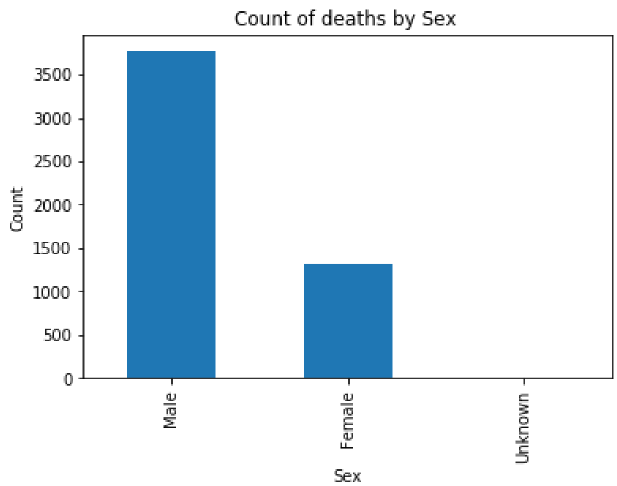
For this analysis I have used python for data cleaning, data exploration, cross analysis of different attributes. Using R, I have built a Linear model and done prediction of number of deaths of next 12 months after 2018. I have imported the dataset into SQL database and did few queries.

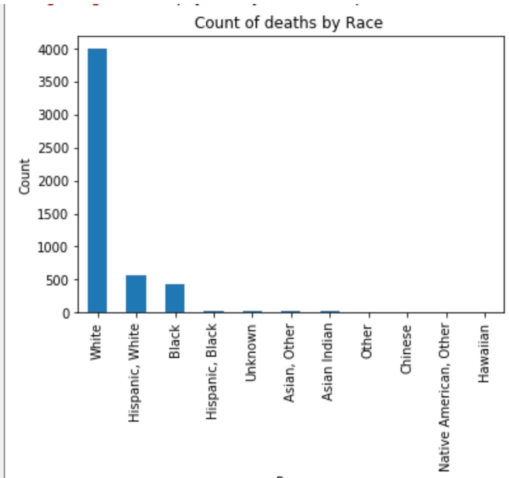
Data Description: Dataset has 5088 records with 41 attributes

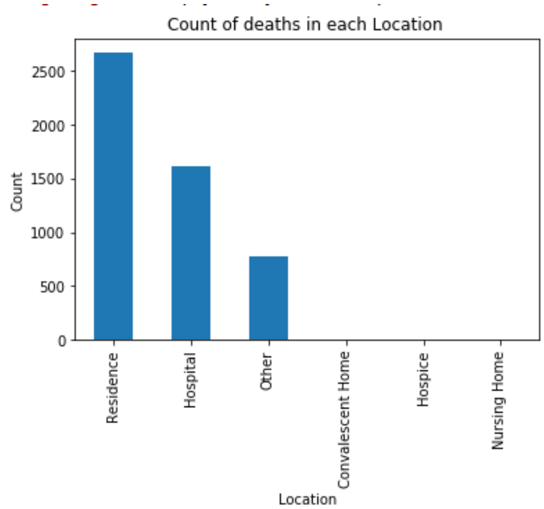
```
In [58]: mydata.shape
Out[58]: (5088, 41)
```

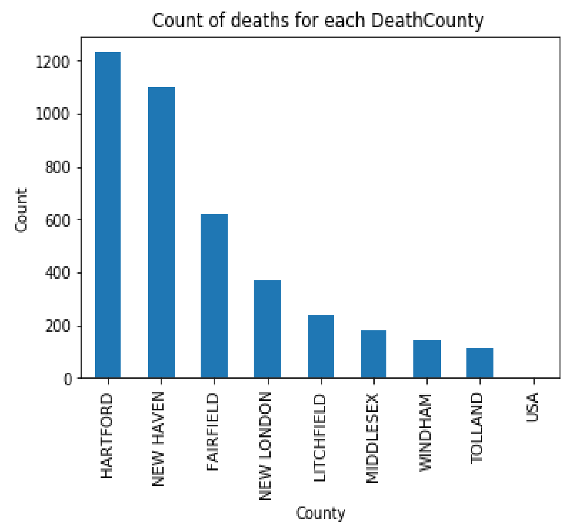
Below are the columns from dataset.

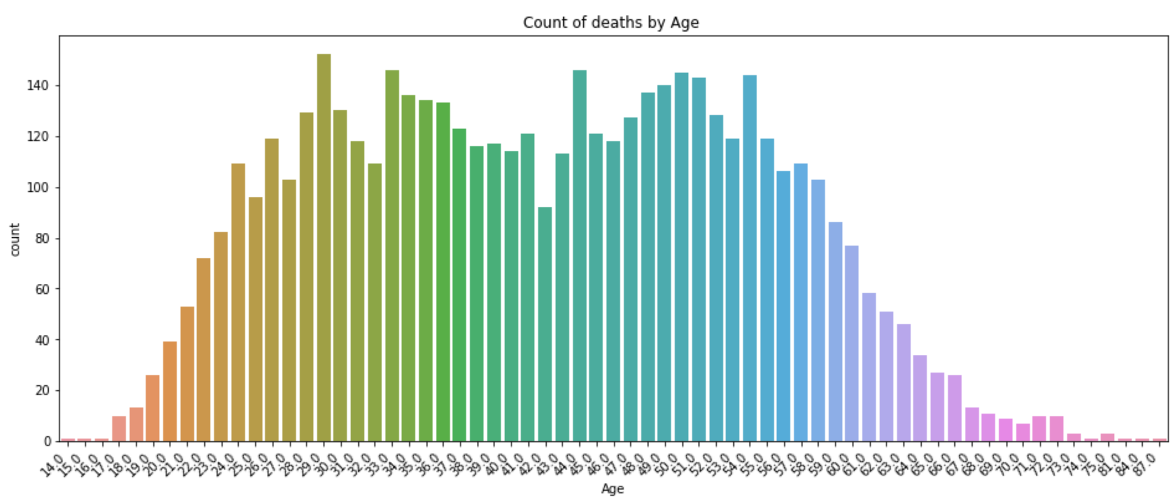
```
In [80]: mydata.columns
Out[80]:
Index(['ID', 'Date', 'DateType', 'Age', 'Sex', 'Race', 'ResidenceCity',
       'ResidenceCounty', 'ResidenceState', 'DeathCity', 'DeathCounty',
       'Location', 'LocationifOther', 'DescriptionofInjury', 'InjuryPlace',
       'InjuryCity', 'InjuryCounty', 'InjuryState', 'COD', 'OtherSignifican',
       'Heroin', 'Cocaine', 'Fentanyl', 'FentanylAnalogue', 'Oxycodone',
       'Oxymorphone', 'Ethanol', 'Hydrocodone', 'Benzodiazepine', 'Methadone',
       'Amphet', 'Tramad', 'Morphine_NotHeroin', 'Hydromorphone', 'Other',
       'OpiateNOS', 'AnyOpioid', 'MannerofDeath', 'DeathCityGeo',
       'ResidenceCityGeo', 'InjuryCityGeo'],
      dtype='object')
```


Count of deaths by Sex
Count
3500
3000
2500
2000
1500
1000
500
0
Male
Female
Unknown
Sex


Count of deaths by Race
Count
4000
3500
3000
2500
2000
1500
1000
500
0
White
Hispanic, White
Black
Hispanic, Black
Unknown
Asian, Other
Asian Indian
Other
Chinese
Native American, Other
Hawaiian
Race


Count of deaths in each Location
Count
2500
2000
1500
1000
500
0
Residence
Hospital
Other
Convalescent Home
Hospice
Nursing Home
Location


Count of deaths for each DeathCounty
Count
1200
1000
800
600
400
200
0
HARTFORD
NEW HAVEN
FAIRFIELD
NEW LONDON
LITCHFIELD
MIDDLESEX
WINDHAM
TOLLAND
USA
County


Count of deaths by Age
count
140
120
100
80
60
40
20
0
Age

The dataset comprises of toxicity records, death certificates and accident scene which are obtained from an examination by the Chief Medical Examiner Office. If person has died due to overdose of specific drug, then the column of drug has ‘Y’ value.

U	V	W	X	Y	Z	AA	AB	
Heroin	Cocaine	Fentanyl	FentanylAı	Oxycodon	Oxymorph	Ethanol	Hydrocodc	Be
		Y					Y	Y
	Y							
Y	Y							
Y		Y						
		Y						
Y								
Y								
	Y							

The following are attributes and count for non-null records for each attribute:

```
mydata.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5105 entries, 0 to 5104
Data columns (total 41 columns):
ID                5105 non-null object
Date              5103 non-null object
DateType          5103 non-null object
Age               5102 non-null float64
Sex               5099 non-null object
Race              5092 non-null object
ResidenceCity     4932 non-null object
ResidenceCounty   4308 non-null object
ResidenceState    3556 non-null object
DeathCity         5100 non-null object
DeathCounty       4005 non-null object
Location          5081 non-null object
LocationifOther   590 non-null object
DescriptionofInjury   4325 non-null object
InjuryPlace       5039 non-null object
InjuryCity        3349 non-null object
InjuryCounty      2364 non-null object
InjuryState       1424 non-null object
COD               5105 non-null object
```

```
OtherSignifican      169 non-null object
Heroin               2529 non-null object
Cocaine              1521 non-null object
Fentanyl             2232 non-null object
FentanylAnalogue     389 non-null object
Oxycodone            607 non-null object
Oxymorphone          108 non-null object
Ethanol              1247 non-null object
Hydrocodone          118 non-null object
Benzodiazepine       1343 non-null object
Methadone            474 non-null object
Amphet               159 non-null object
Tramad               130 non-null object
Morphine_NotHeroin   42 non-null object
Hydromorphone        25 non-null object
Other                435 non-null object
OpiateNOS            88 non-null object
AnyOpioid            2466 non-null object
MannerofDeath        5095 non-null object
DeathCityGeo         5105 non-null object
ResidenceCityGeo     5012 non-null object
InjuryCityGeo        5027 non-null object
dtypes: float64(1), object(40)
memory usage: 1.6+ MB
```

The below shows the sum of null values of each attribute

```
In [44]: mydata.isna().sum()
Out[44]:
ID                     0
Date                   2
DateType               2
Age                    3
Sex                    6
Race                  13
ResidenceCity        173
ResidenceCounty      797
ResidenceState      1549
DeathCity              5
DeathCounty         1100
Location              24
LocationifOther     4515
DescriptionofInjury  780
InjuryPlace           66
InjuryCity          1756
InjuryCounty        2741
InjuryState         3681
COD                    0
OtherSignifican     4936
Heroin              2576
Cocaine             3584
Fentanyl            2873
FentanylAnalogue    4716
Oxycodone           4498
```

Because age, sex, race is important to our study, we can drop with null values in these 4 attributes.

```
In [45]: mydata.dropna(subset=['Date', 'Age','Sex','Race'], inplace=True)
```

The Date column in in form of “6/28/2014 12:00:00 AM”. To see the count of deaths per year, month and weekday, I have created a new column and extracted year, month and day of week from Date column using insert function. The four columns are placed at 2,3, 4 and 5th position in mydata data frame.

```
In [68]: Date_df = pd.to_datetime(mydata['Date'])
    ...: mydata.insert(loc=2, column='new_date', value=Date_df)
    ...: # Creating new column for Year by extracting Year from old Date column
    ...: mydata.insert(loc=3,column='Year',value=mydata['new_date'].dt.year)
    ...: #Creating Month Column and adding it our data frame mydata
    ...: mydata.insert(loc=4,column='Month',value=mydata['new_date'].dt.month)

In [69]:
```

The mydata[‘new_date’].dt.month function extracts month in numerical format. To see the month in word format I have created a dictionary with key as 1:12 and values as January: December. Using apply and lambda function I have replaced each numerical value in ‘Month’ column with word from month_dict(dictionary).

```
In [70]: month_dict={1:'January',2:'Febuary',3:'March',4:'April',5:'May',6:'June',
7:'July',8:'August',9:'September',10:'October',11:'November',12:'December'}
    ...: #replacing the numerical month with words from above dictionary using apply
function
    ...: mydata['Month'] = mydata['Month'].apply(lambda x: month_dict[x])
```

Followed similar process for converting numerical day-of-week to words

```
In [71]:
mydata.insert(loc=5,column='day_of_week',value=mydata['new_date'].dt.dayofweek)
    ...: days_dict = {0:'Monday',1:'Tuesday',2:'Wednesday',3:'Thurday',4:'Friday',
5:'Saturday',6:'Sunday'}
    ...: mydata['day_of_week'] = mydata['day_of_week'].apply(lambda x: days_dict[x])
```

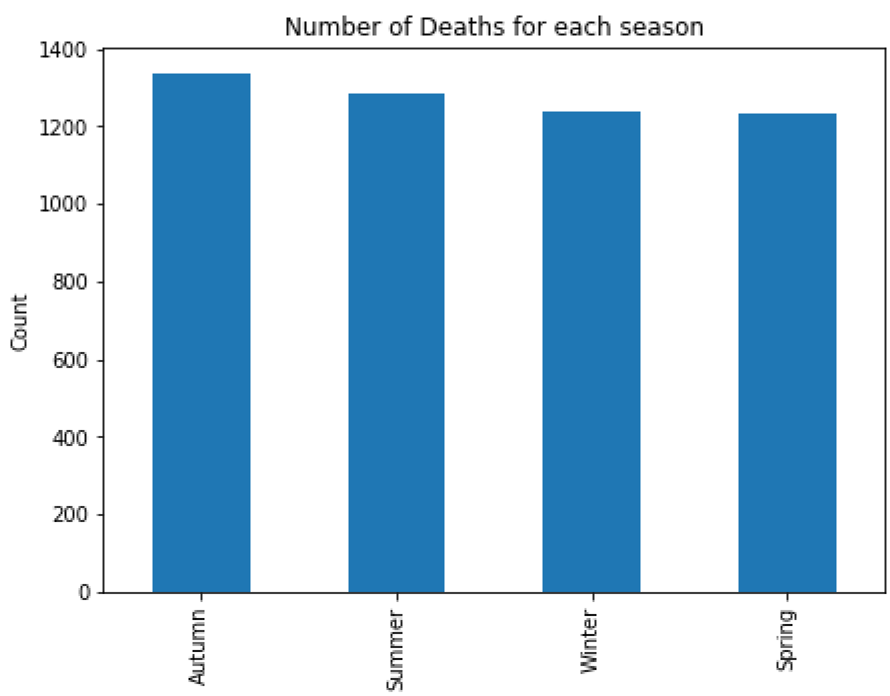
Below are the new columns created.

```
In [73]: mydata.iloc[:,[2,3,4,5]].head()
Out[73]:
     new_date  Year    Month day_of_week
1  2013-03-21  2013    March     Thurday
2  2016-03-13  2016    March      Sunday
3  2016-03-31  2016    March     Thurday
4  2013-02-13  2013  Febuary   Wednesday
5  2014-06-29  2014     June      Sunday
```

Created a new column for season

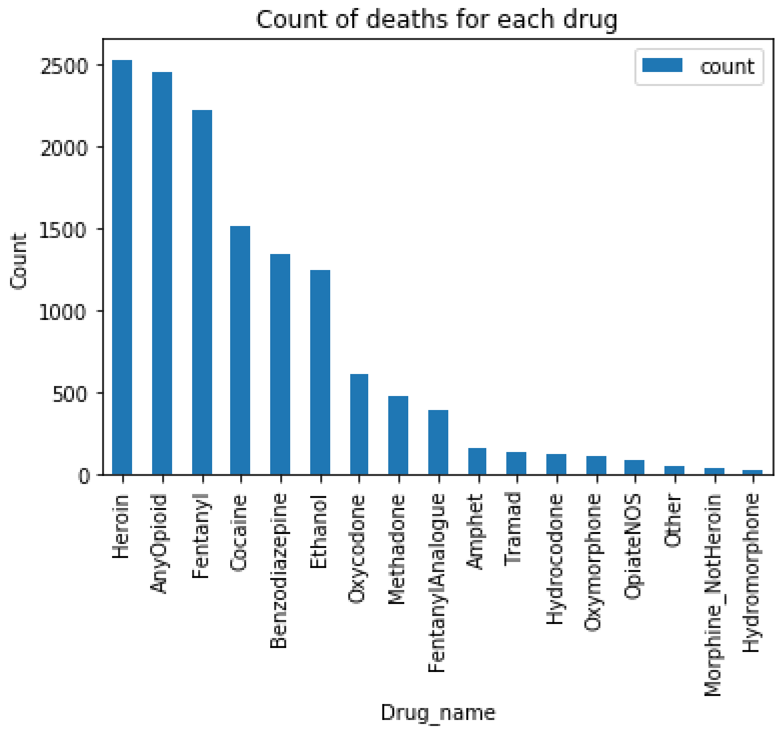
```
In [63]: lookup={11:'Autumn',12:'Winter',1:'Winter',2:'Winter',3:'Spring',4:'Spring',5:'Spring',
6:'Summer',7:'Summer',8:'Summer',9:'Autumn',10:'Autumn'}
    ...: mydata['Season']=mydata['new_date'].dt.month.apply(lambda x: lookup[x])
    ...: mydata.loc[:,'multiple_drugs_used':'Season'].head()
    ...:
    ...: mydata['Season'].value_counts()
Out[63]:
Autumn    1337
Summer    1283
Winter    1237
Spring    1231
Name: Season, dtype: int64
```

```
Out[64]: Text(0.5, 1.0, 'Number of Deaths for each season')
```



We have death records associated with consumption of one or many combinations of drugs. Below we can see the deaths associated by each drug

```
In [92]: Drugs_used=pd.DataFrame()
    ...: Drugs_used['Drug_name']=0
    ...: Drugs_used['count']=0
    ...:
    ...: for column in mydata.loc[:, 'Heroin':'AnyOpioid']:
    ...:
Drugs_used=Drugs_used.append({'Drug_name':column,'count':mydata[column].value_counts()
[0]},ignore_index=True)
    ...:
    ...:
    ...:
    ...: # plot for number of deaths associated by each drug
    ...: Drugs_used.sort_values('count')
Out[92]:
            Drug_name  count
13      Hydromorphone     25
12  Morphine_NotHeroin     38
14              Other     47
15          OpiateNOS     88
5         Oxymorphone    107
7         Hydrocodone    116
11             Tramad    130
10             Amphet    159
3     FentanylAnalogue    388
9           Methadone    472
4           Oxycodone    605
6             Ethanol   1242
8        Benzodiazepine   1338
1             Cocaine   1514
2            Fentanyl   2221
16          AnyOpioid   2452
0              Heroin   2525
```



In below, a new column has been created to track combination of drugs used by each record.

```
In [93]: mydata['multiple_drugs_used'] = mydata.loc[:,
'Heroin':'AnyOpioid'].apply(lambda value: ', '.join(value[value.notnull()]), axis = 1)

In [94]: mydata['multiple_drugs_used']
Out[94]:
1                                                 COCAINE
2                              HEROIN, COCAINE, ANYOPIOID
3                             HEROIN, FENTANYL, ANYOPIOID
4                                                FENTANYL
5                                                  HEROIN
                              ...
5100                              ETHANOL, BENZODIAZEPINE
5101                               HEROIN, BENZODIAZEPINE
5102    HEROIN, FENTANYL, FENTANYLANALOGUE, TRAMAD, AN...
5103                                             FENTANYL
5104                                    HEROIN, ANYOPIOID
Name: multiple_drugs_used, Length: 5088, dtype: object
```

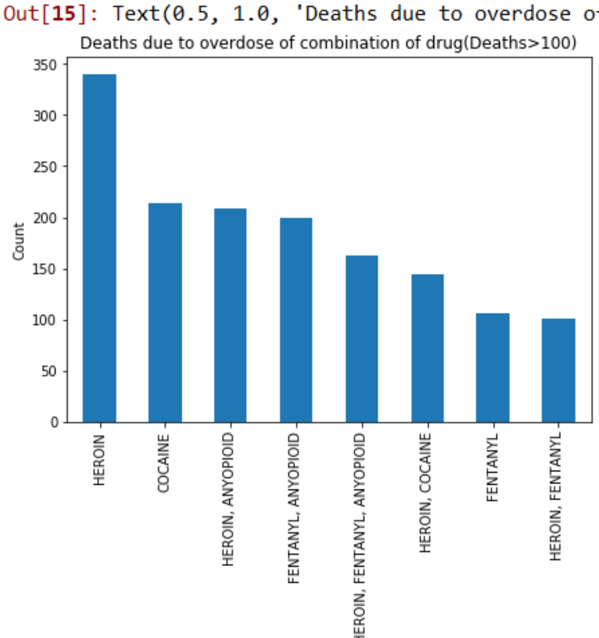
A new column was created to track count of drugs used for each record

```
In [104]: mydata['drugs_count'] = mydata['multiple_drugs_used'].apply(lambda drugs:
len(drugs.split(', ')))

In [105]: mydata['drugs_count']
Out[105]:
1       1
2       3
3       3
4       1
5       1
       ..
5100    2
5101    2
5102    5
5103    1
5104    2
Name: drugs_count, Length: 5088, dtype: int64
```

Below we can see the deaths associated with combination of drugs where count is more than 100

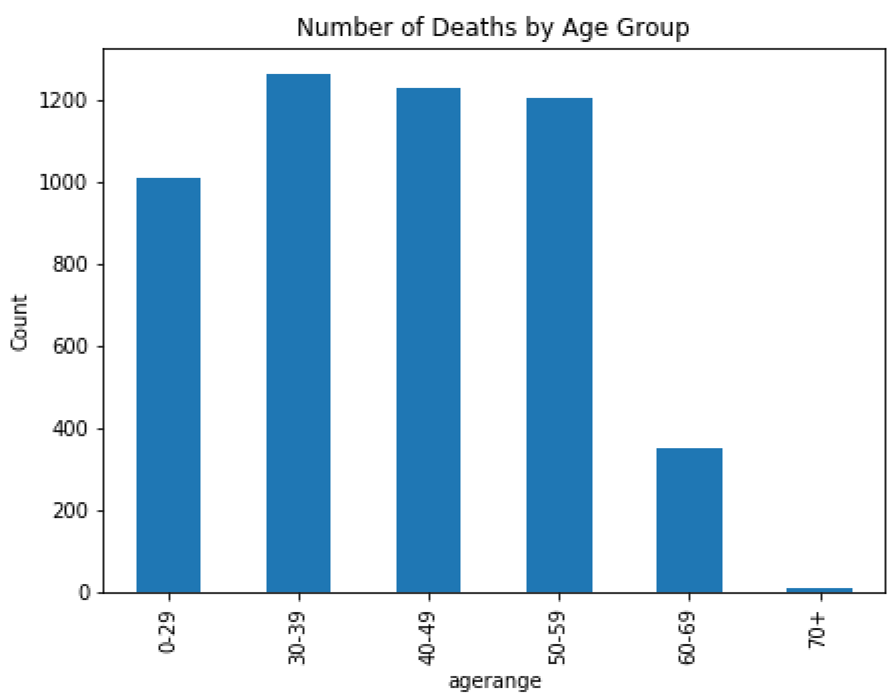
```
In [107]: multiple_drugs_count=mydata['multiple_drugs_used'].value_counts()
     ...: multiple_drugs_count.loc[mydata['multiple_drugs_used'].value_counts()>100]
Out[107]:
HEROIN                          340
COCAINE                         214
HEROIN, ANYOPIOID               209
FENTANYL, ANYOPIOID             200
HEROIN, FENTANYL, ANYOPIOID     162
HEROIN, COCAINE                 144
FENTANYL                        106
HEROIN, FENTANYL                101
Name: multiple_drugs_used, dtype: int64
```

```
Out[15]: Text(0.5, 1.0, 'Deaths due to overdose of combination of drug(Deaths>100)')
```

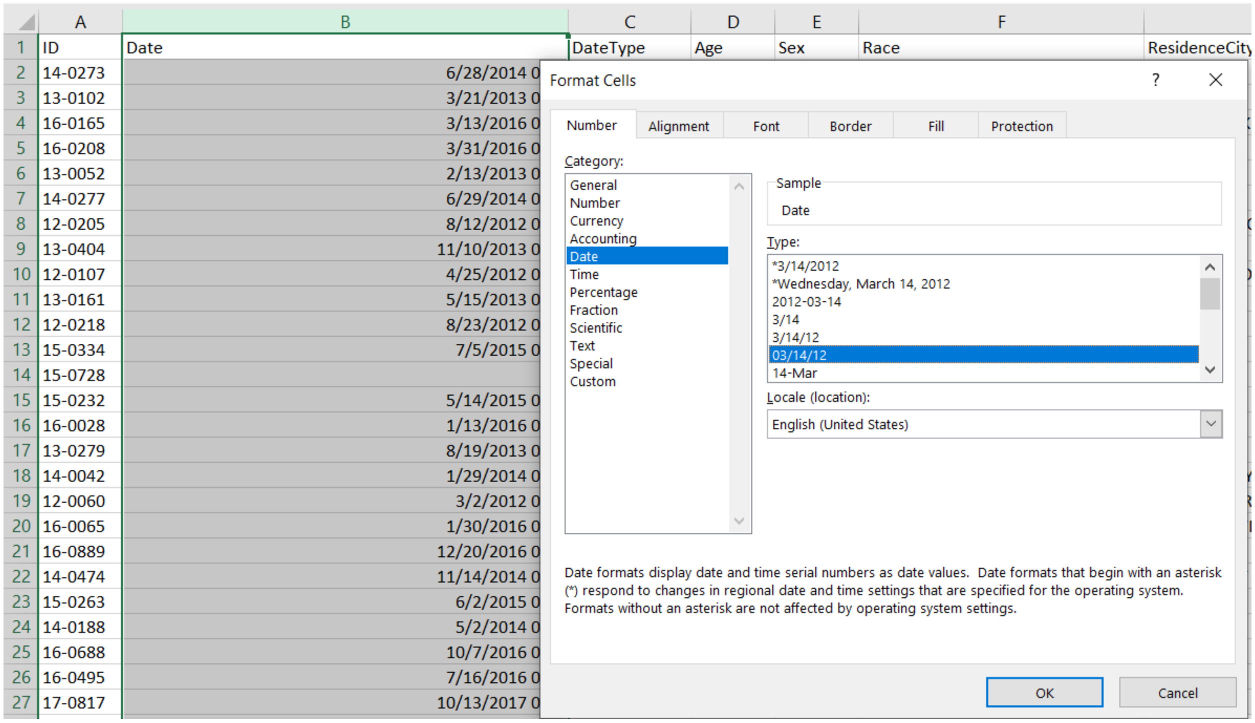


A new column is created for age range for each record

```
In [109]: bins = [0, 29, 39, 49,59,69, 70]
     ...: labels = ['0-29', '30-39', '40-49', '50-59', '60-69', '70+']
     ...: mydata['agerange'] = pd.cut(mydata.Age, bins, labels = labels,include_lowest
= True)
     ...:
     ...: mydata.groupby('agerange')['agerange'].count()
Out[109]:
agerange
0-29     1006
30-39    1262
40-49    1229
50-59    1202
60-69     352
70+         7
Name: agerange, dtype: int64
```



Using R programming, prediction of number of cases for next 12 months has been made using linear regression. In excel file I have formatted the Date column and converted it to simple Date format : 'MM/DD/YYYY'.



A new variable 'tab' is created to track number of deaths in each month from 2012-2018

```
> library('ggplot2')
> mydata=read.csv("C:/Users/sruja/Desktop/Courses/AIT 580/Final Project/Drug-related-deaths2.csv")
> mydata$Date <- as.Date(mydata$Date, format= "%m/%d/%y")
> tab <- table(cut(mydata$Date, 'month'))
> tab

2012-01-01 2012-02-01 2012-03-01 2012-04-01 2012-05-01 2012-06-01 2012-07-01 2012-08-01 2012-09-01 2012-10-01 2012-11-01 2012-12-01 2013-01-01
        31         27         24         30         28         28         29         31         27         36         27         37         35
2013-02-01 2013-03-01 2013-04-01 2013-05-01 2013-06-01 2013-07-01 2013-08-01 2013-09-01 2013-10-01 2013-11-01 2013-12-01 2014-01-01 2014-02-01
        38         42         33         27         44         37         37         39         48         54         56         47         50
2014-03-01 2014-04-01 2014-05-01 2014-06-01 2014-07-01 2014-08-01 2014-09-01 2014-10-01 2014-11-01 2014-12-01 2015-01-01 2015-02-01 2015-03-01
        47         42         46         47         32         42         46         54         52         53         51         53         50
2015-04-01 2015-05-01 2015-06-01 2015-07-01 2015-08-01 2015-09-01 2015-10-01 2015-11-01 2015-12-01 2016-01-01 2016-02-01 2016-03-01 2016-04-01
        54         50         62         76         51         66         86         76         52         70         69         69         79
2016-05-01 2016-06-01 2016-07-01 2016-08-01 2016-09-01 2016-10-01 2016-11-01 2016-12-01 2017-01-01 2017-02-01 2017-03-01 2017-04-01 2017-05-01
        78         79         76         76         73         68         99         81         85         88         97         77         98
2017-06-01 2017-07-01 2017-08-01 2017-09-01 2017-10-01 2017-11-01 2017-12-01 2018-01-01 2018-02-01 2018-03-01 2018-04-01 2018-05-01 2018-06-01
        95         95         81         67         77         81         97         63         81         97         80         89        105
2018-07-01 2018-08-01 2018-09-01 2018-10-01 2018-11-01 2018-12-01
        92         73         92         83         89         74
```

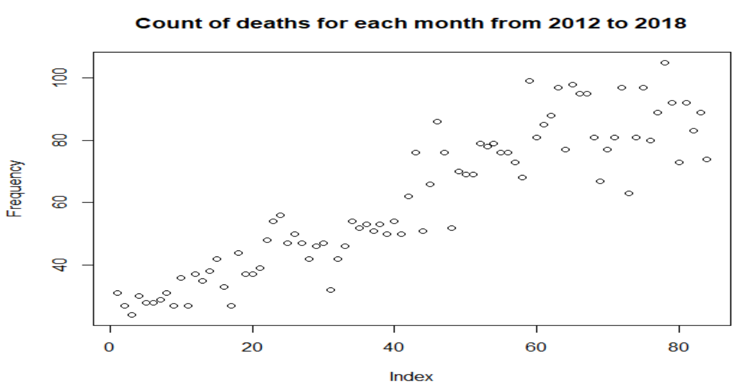
To see if there is Linear relation among month and number of deaths, months from 2012-2018 are indexed from 1 to 84. If the index is 1, it represents the record of the first month in 2012. If the index is 13, it represents the record for first month of 2013.

```
84 12/2018          74     84
> final_df=data.frame(Date=format(as.Date(names(tab)), '%m/%Y'
+             Frequency=as.vector(tab))
>
> final_df['Index']=1:84
> final_df
      Date Frequency Index
1  01/2012        31     1
2  02/2012        27     2
3  03/2012        24     3
4  04/2012        30     4
5  05/2012        28     5
6  06/2012        28     6
7  07/2012        29     7
8  08/2012        31     8
9  09/2012        27     9
10 10/2012        36    10
11 11/2012        27    11
12 12/2012        37    12
13 01/2013        35    13
14 02/2013        38    14
15 03/2013        42    15
16 04/2013        33    16
```

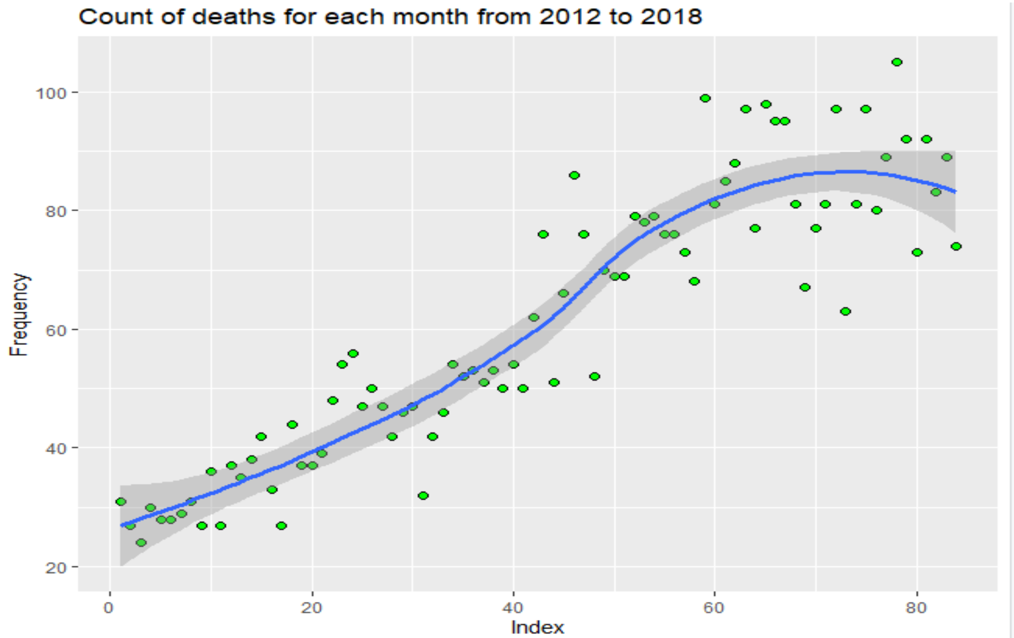
After indexing the months, we can see the below plot which shows us the count of deaths from 2012 to 2018

Linear Regression

Linear regression attempts to model the relationship between two variables by fitting the observed data with a linear equation. One variable is an independent variable and the other variable is dependent variable. A linear regression line has an equation of the form Y = a + bX, where X is the independent variable and Y is the dependent variable , b is the slope of the line and a is the intercept. In our data the independent variable(X) is Index of month and Y is the frequency of deaths.



Smoothing Plot for index of months and count of deaths shows us the strong linear relation.



cor.test() function is used to check the association between two variables.

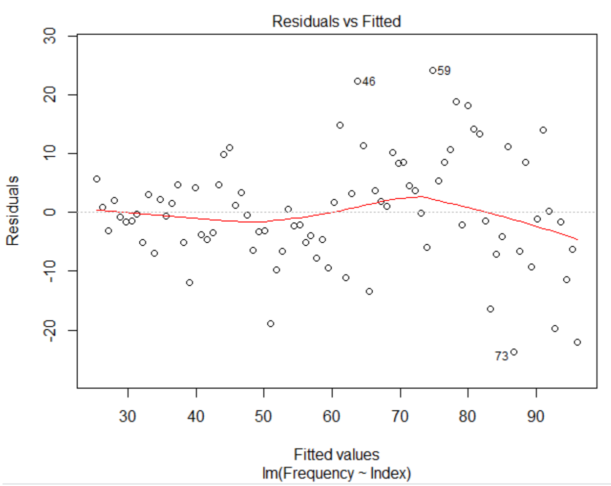
```
> cor.test(final_df$Index,final_df$Frequency, method=c("pearson"))

        Pearson's product-moment correlation

data:  final_df$Index and final_df$Frequency
t = 20.127, df = 82, p-value < 2.2e-16
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.8670873 0.9421410
sample estimates:
      cor
0.9119513
```

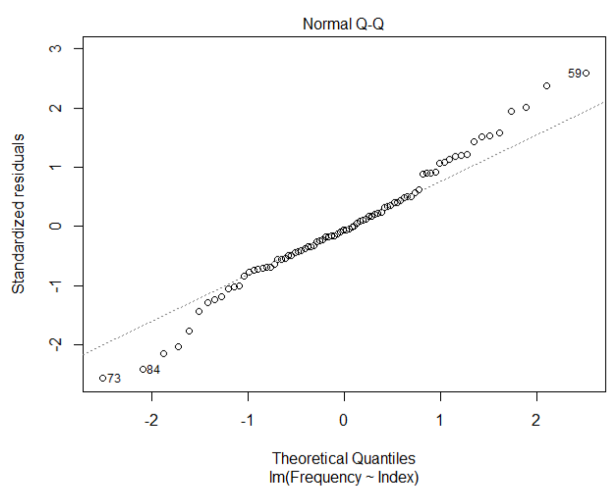
The p-value of the test is 2.2^{-16}, which is less than the significance level alpha = 0.05. We can conclude that Frequency and index of month are significantly correlated with a correlation coefficient of 0.911 and p-value of 2.2^{-16} .

The cor value is 0.911 which means that there is strong correlation between index of month and frequency . This means that Frequency of Deaths increase with index of month.

Now we can fit a linear model .

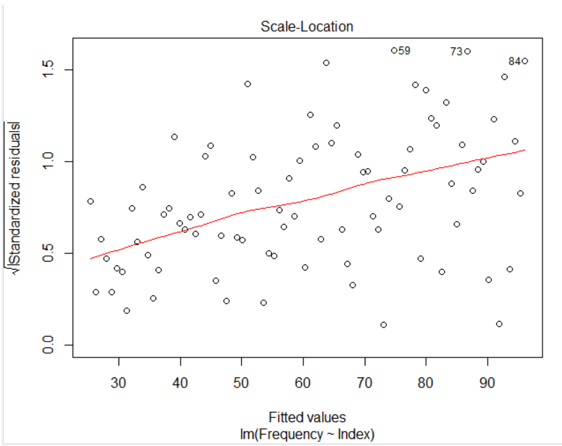
```
> lm.fit <- lm(Frequency~Index,data=final_df)
> summary(lm.fit)

Call:
lm(formula = Frequency ~ Index, data = final_df)

Residuals:
     Min       1Q   Median       3Q      Max 
-23.7542  -5.1641  -0.5666   4.7044  24.1822 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 24.51463    2.07273   11.83   <2e-16 ***
Index        0.85260    0.04236   20.13   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 9.414 on 82 degrees of freedom
Multiple R-squared:  0.8317,	Adjusted R-squared:  0.8296 
F-statistic: 405.1 on 1 and 82 DF,  p-value: < 2.2e-16
```

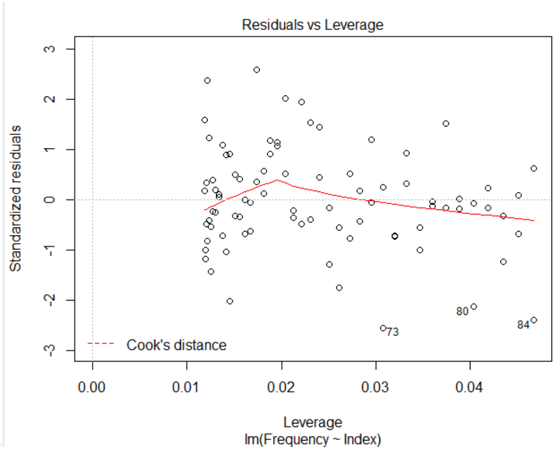




In the Residuals vs fitted plot, we can check the linear relationship between x and y. The red line must be along 0. Residuals are normally distributed with mean of 0 and constant variance.

The normal Q-Q plot is used to check normality. To compare the quantiles of standardized residuals to those of a standard normal . if the residuals are normally distributed the points should align linearly along the black line which we can see in above right-side plot.





Scale Location plot : this is transformed version of first plot . We take absolute value of standard residuals and make square root.

Residuals vs leverage: The points that have high leverage and residuals can be considered as outliers. Any points that extend the cooks distance have high leverage and high residuals. There are no outliers in our data.

By this we can say that our data satisfies all the assumptions of Linear model.

For implementing SQL queries, I have choose only below columns which are useful for interpreting results.

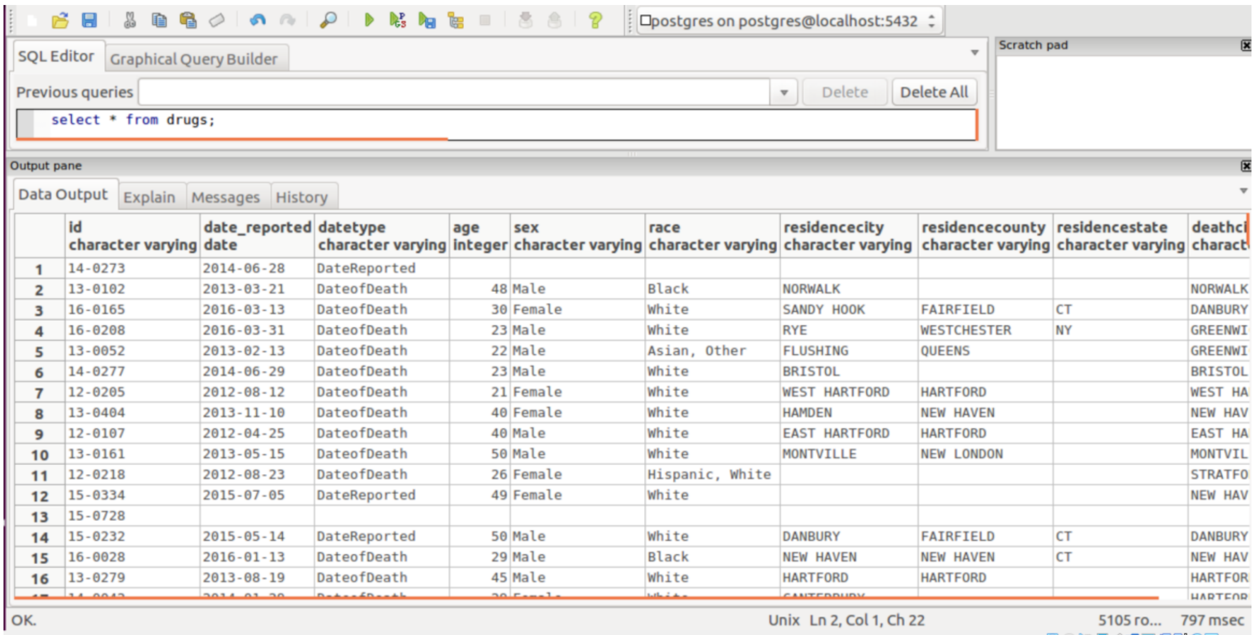
ID	Date_reported	DateType	Age	Sex	Race	ResidenceCity	ResidenceCounty	ResidenceState	DeathCity

Create table query:

```
CREATE TABLE public.drugs
(
 id character varying,
 date_reported date,
 datetype character varying,
 age integer,
 sex character varying,
 race character varying,
 residencecity character varying,
```

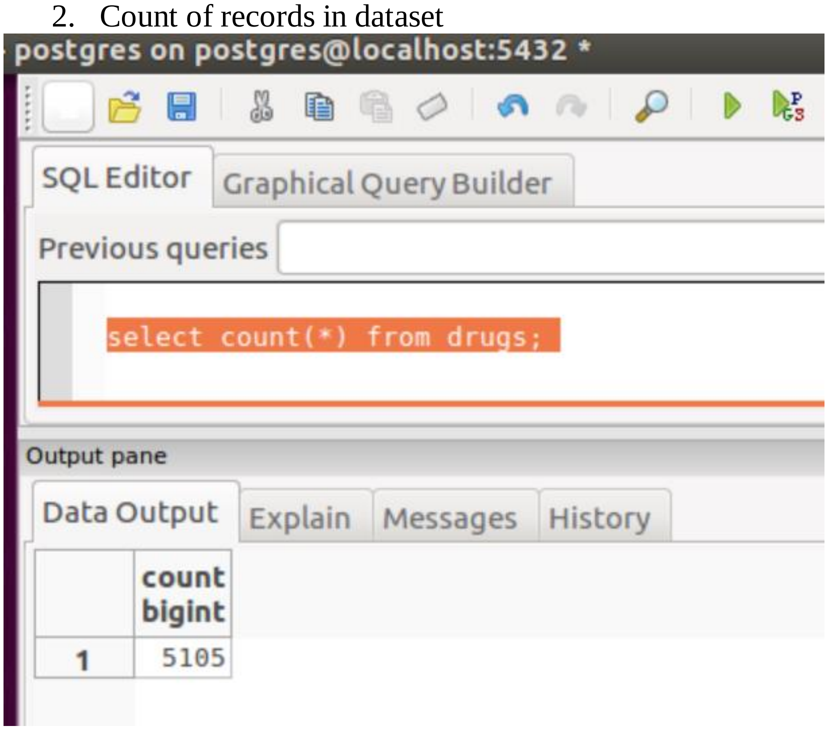
residencecounty character varying,
residencestate character varying,
deathcity character varying
)

1. Displaying all records in data

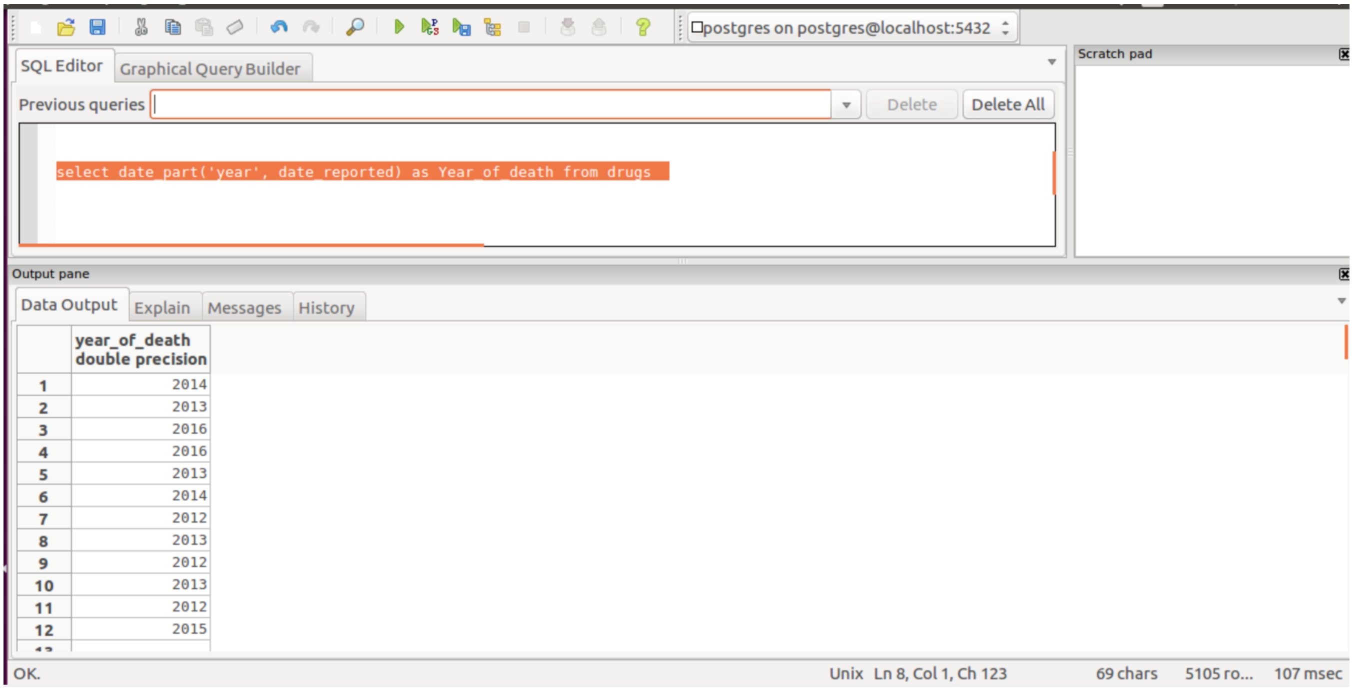


	id character varying	date_reported date	datetype character varying	age integer	sex character varying	race character varying	residencecity character varying	residencecounty character varying	residencestate character varying	deathci charact
1	14-0273	2014-06-28	DateReported							
2	13-0102	2013-03-21	DateofDeath	48	Male	Black	NORWALK			NORWALK
3	16-0165	2016-03-13	DateofDeath	30	Female	White	SANDY HOOK	FAIRFIELD	CT	DANBURY
4	16-0208	2016-03-31	DateofDeath	23	Male	White	RYE	WESTCHESTER	NY	GREENWI
5	13-0052	2013-02-13	DateofDeath	22	Male	Asian, Other	FLUSHING	QUEENS		GREENWI
6	14-0277	2014-06-29	DateofDeath	23	Male	White	BRISTOL			BRISTOL
7	12-0205	2012-08-12	DateofDeath	21	Female	White	WEST HARTFORD	HARTFORD		WEST HA
8	13-0404	2013-11-10	DateofDeath	40	Female	White	HAMDEN	NEW HAVEN		NEW HAV
9	12-0107	2012-04-25	DateofDeath	40	Male	White	EAST HARTFORD	HARTFORD		EAST HA
10	13-0161	2013-05-15	DateofDeath	50	Male	White	MONTVILLE	NEW LONDON		MONTVIL
11	12-0218	2012-08-23	DateofDeath	26	Female	Hispanic, White				STRATFO
12	15-0334	2015-07-05	DateReported	49	Female	White				NEW HAV
13	15-0728									
14	15-0232	2015-05-14	DateReported	50	Male	White	DANBURY	FAIRFIELD	CT	DANBURY
15	16-0028	2016-01-13	DateofDeath	29	Male	Black	NEW HAVEN	NEW HAVEN	CT	NEW HAV
16	13-0279	2013-08-19	DateofDeath	45	Male	White	HARTFORD	HARTFORD		HARTFOR

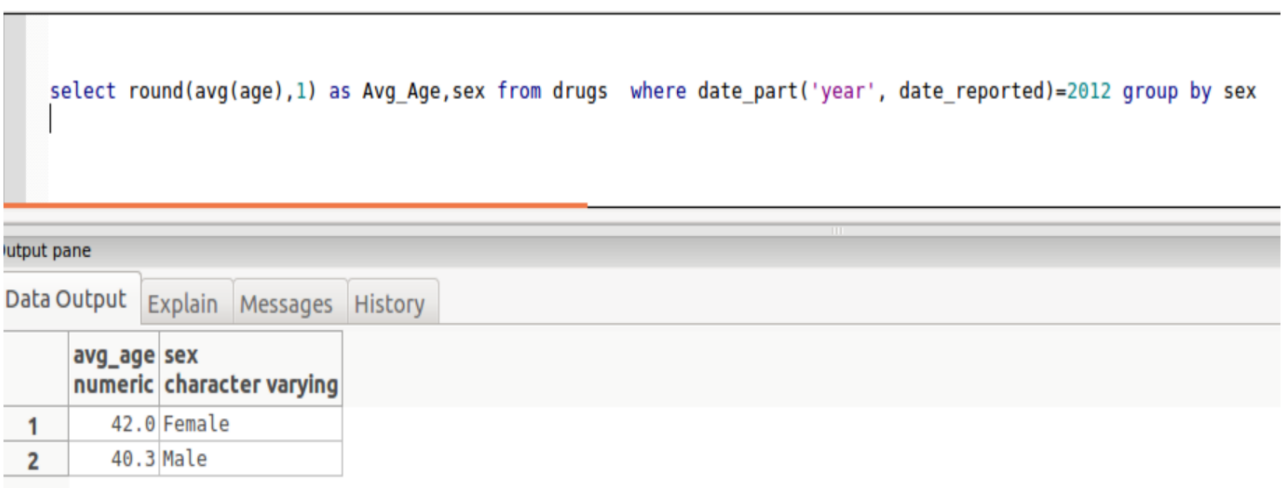
2. Count of records in dataset



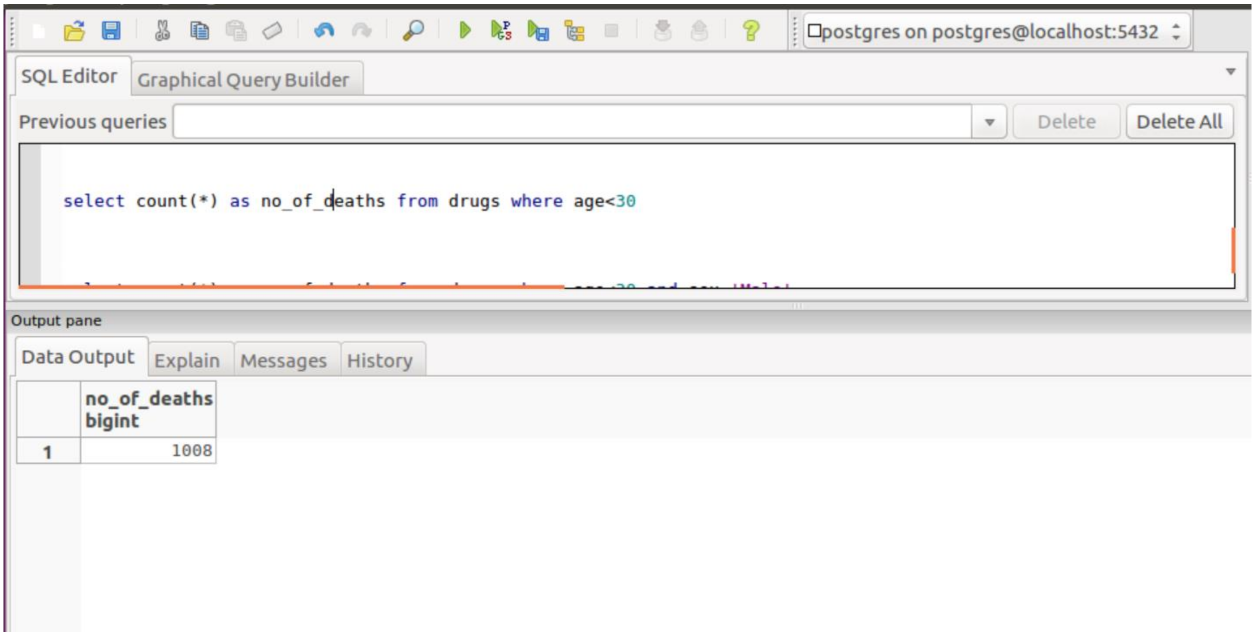
3. Extracting Year from date column



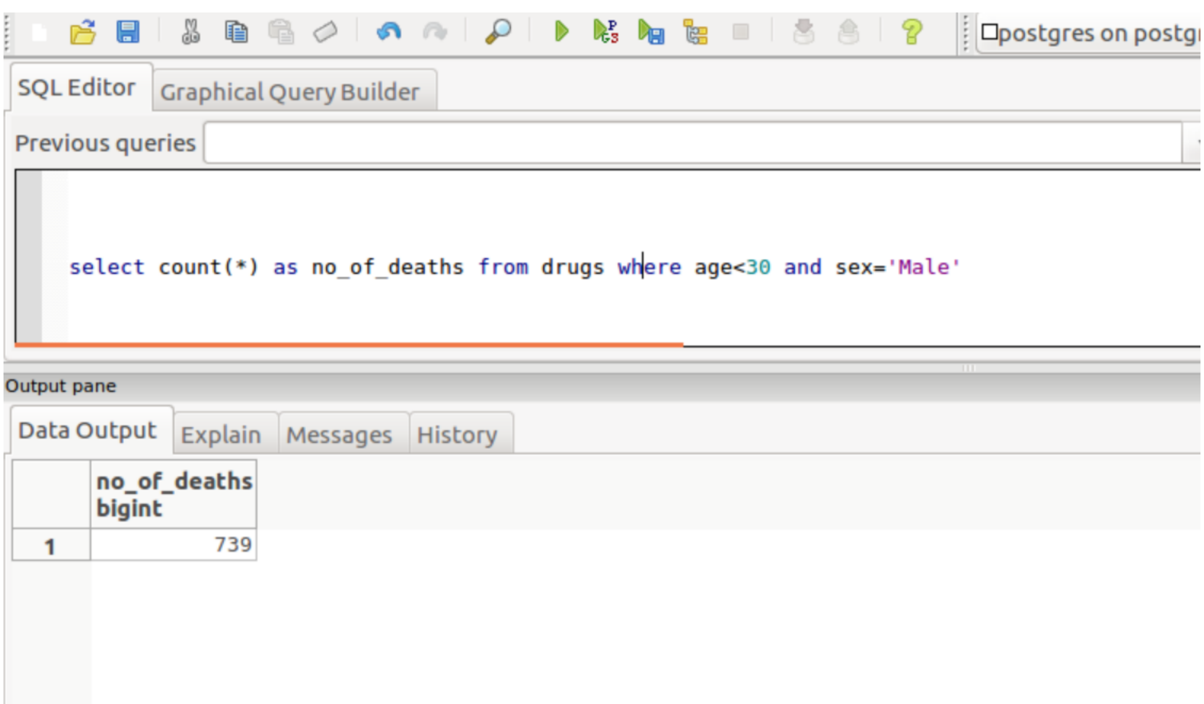
4. Query for finding avg age of male and female in year 2012



5. Query for getting number of deaths whose age is less than 30.

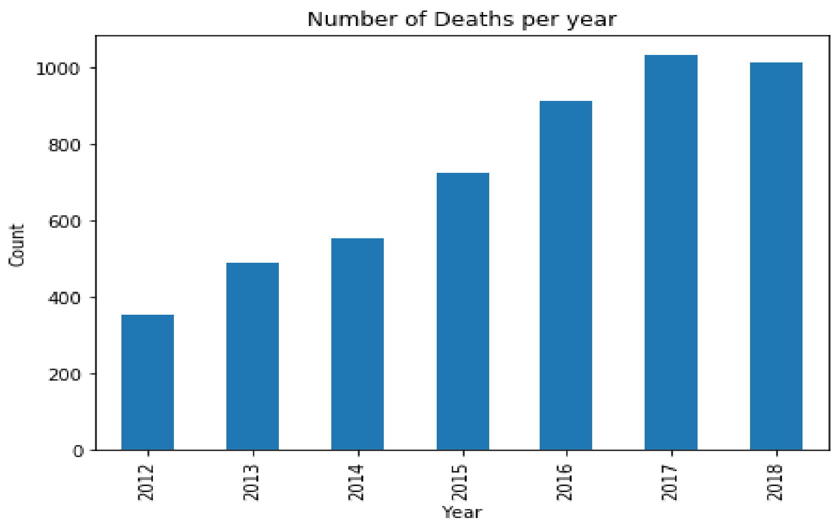


6. Query for finding number of deaths whose age is less than 30 and gender is Male

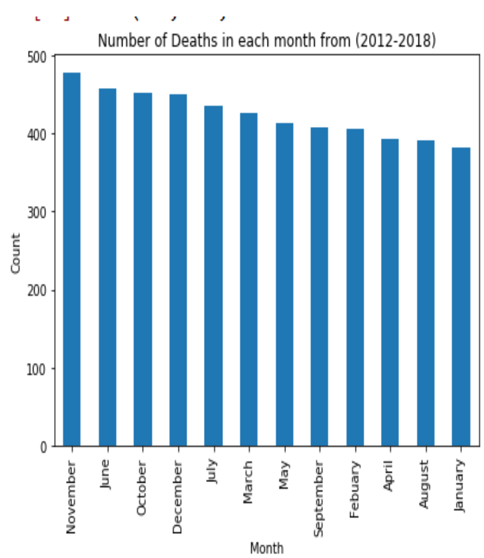


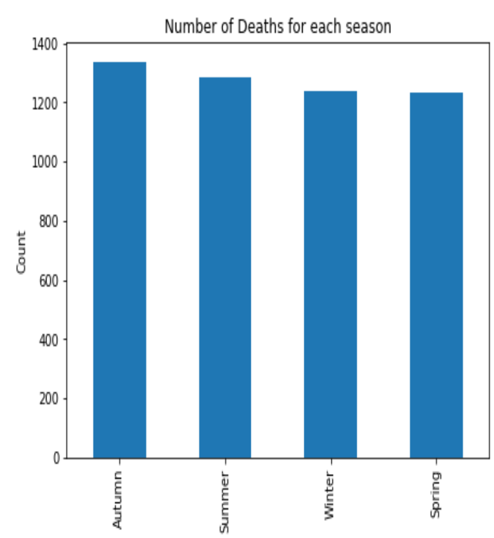
Results:

1. **See if there is an increase in drug related deaths over year**
   From the below graph we can see that number of drug overdose deaths has increased. In 2018 there are around 1000 deaths recorded which is twice of number of deaths in 2012. The highest number of deaths occurred in 2017.



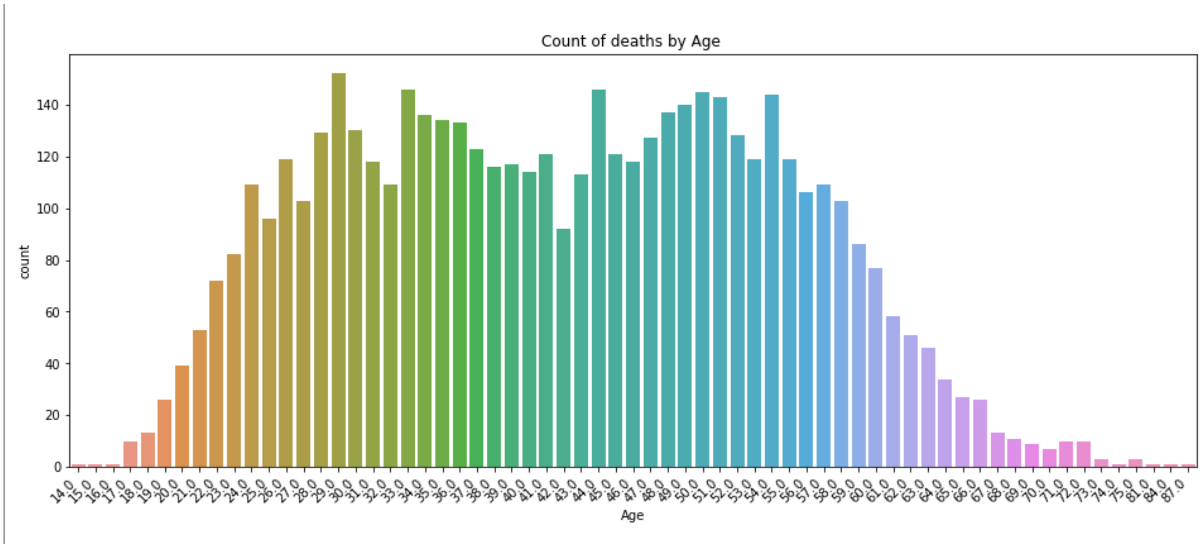
2. **Seasonal impact on drug related deaths**
   We can see that drug overdose deaths doesn't have any seasonal impact. There are at least 400 deaths recorded for each month since 2012.

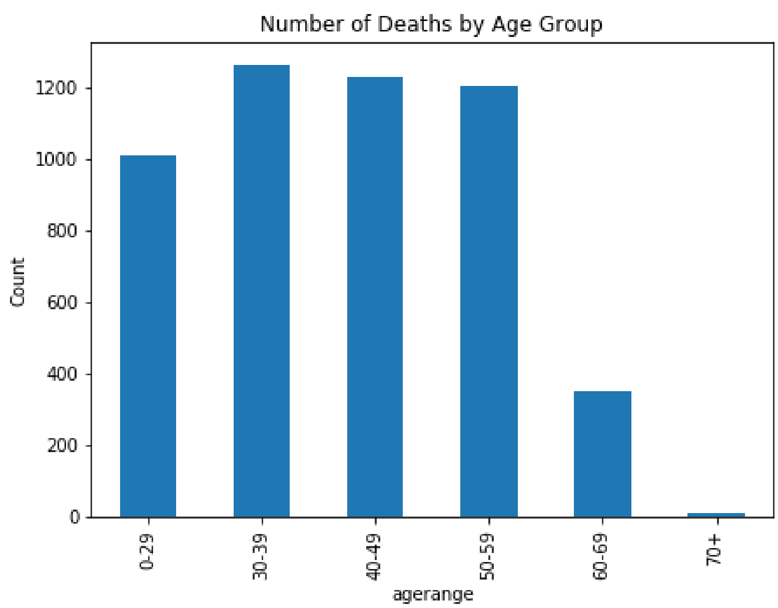




3. Which age group, sex, race has more accidental drug deaths?

There are few deaths for underage `people(<18) . Age group of 30-39 has highest number of deaths. The number of deaths for age groups 40-49, 50-59 are slightly less than 30-39's.

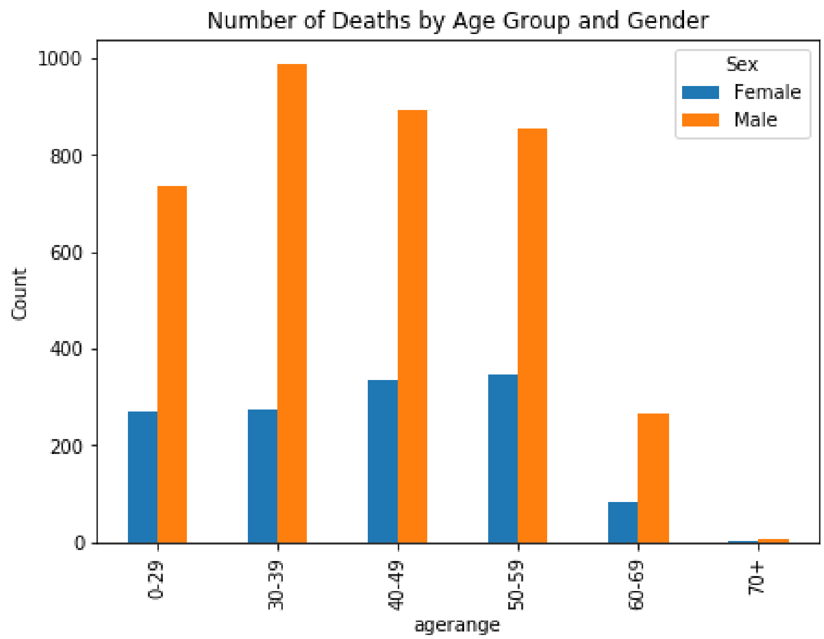




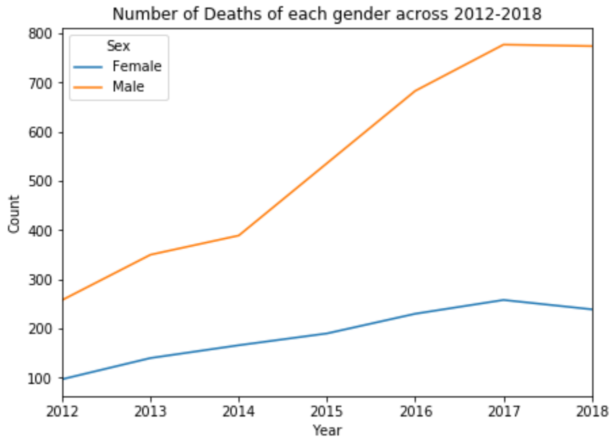
```
In [77]: mydata.groupby('agerange')['agerange'].count()
Out[77]:
agerange
0-29     1006
30-39    1262
40-49    1229
50-59    1202
60-69     352
70+         7
Name: agerange, dtype: int64
```

There are a greater number of male deaths in every age group. Accidental number of deaths of males is almost 3 times of Females. In the 2nd graph below, we can that number of deaths of males has increased largely after 2014 while female deaths trend appears to flatten after 2017.

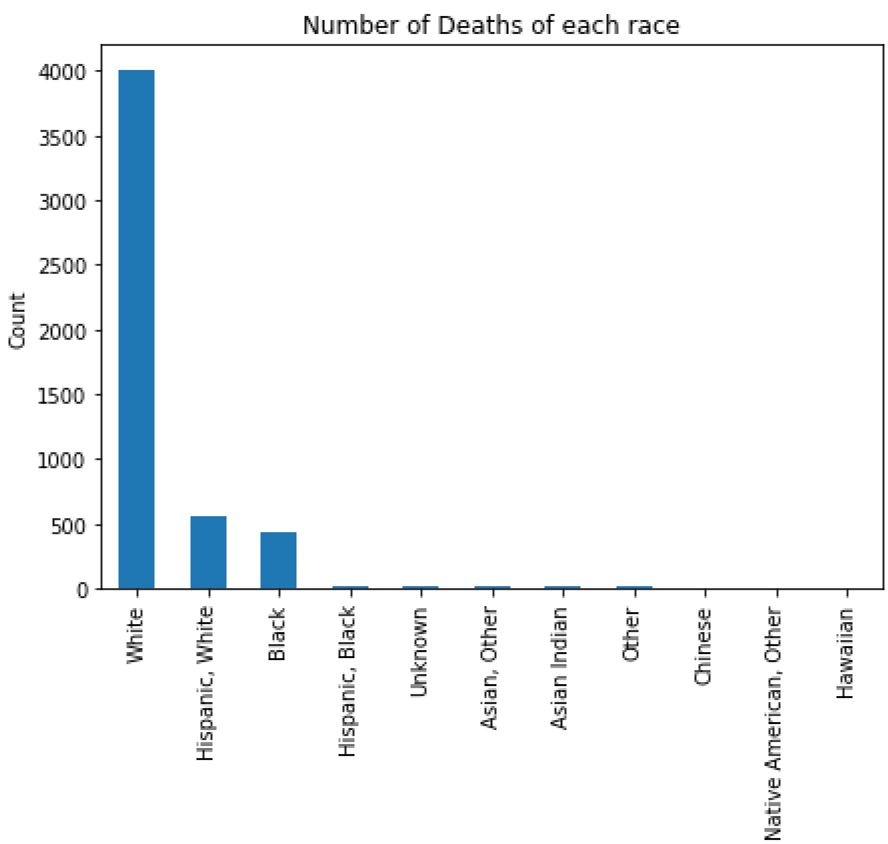
```
In [80]: mydata[mydata['Sex']!='Unknown'].groupby(['agerange','Sex'])
['Sex'].count().unstack().plot(kind='bar',figsize=(7,5))
    ...: plt.ylabel('Count')
    ...: plt.title('Number of Deaths by Age Group and Gender')
Out[80]: Text(0.5, 1.0, 'Number of Deaths by Age Group and Gender')
```



```
In [82]: mydata[mydata['Sex']!='Unknown'].groupby(['Year','Sex'])
['Sex'].count().unstack().plot(kind='line',figsize=(7,5))
    ...: plt.ylabel('Count')
    ...: plt.title('Number of Deaths of each gender across 2012-2018')
Out[82]: Text(0.5, 1.0, 'Number of Deaths of each gender across 2012-2018')
```



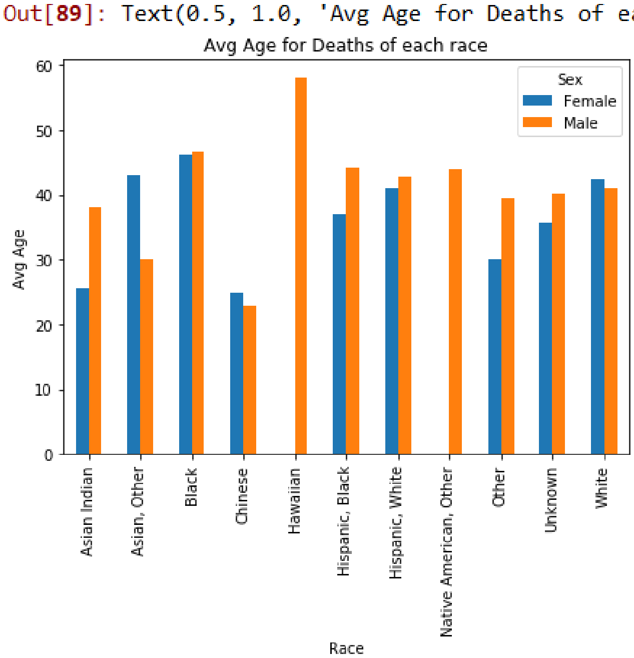
White people have recorded nearly 4000 deaths. There are less than 500 deaths recorded for Hispanic and Black race.



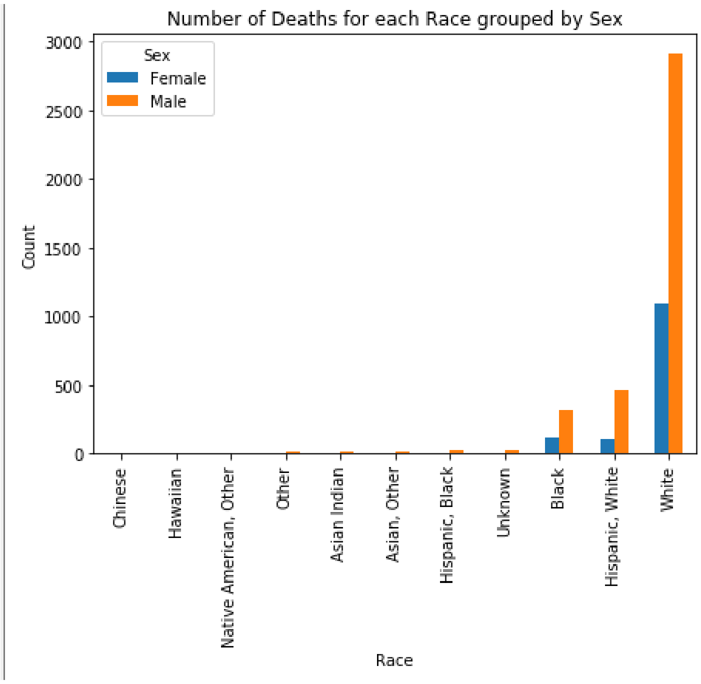
In the below graph we can see the average age of White, Black and Hispanic people is between 40-49 for both male and female.

```
In [91]: mydata[mydata['Sex']!='Unknown'].groupby(['Race','Sex'])['Age'].mean().unstack()
Out[91]:
Sex                         Female       Male
Race
Asian Indian             25.666667  38.181818
Asian, Other             43.142857  30.090909
Black                    46.142857  46.700935
Chinese                  25.000000  23.000000
Hawaiian                       NaN  58.000000
Hispanic, Black          37.000000  44.263158
Hispanic, White          41.010204  42.870130
Native American, Other         NaN  44.000000
Other                    30.000000  39.400000
Unknown                  35.750000  40.052632
White                    42.399449  41.090003
```

```
In [89]: mydata[mydata['Sex']!='Unknown'].groupby(['Race','Sex'])
['Age'].mean().unstack().plot(kind='bar',figsize=(7,5))
    ...: plt.ylabel('Avg Age')
    ...: plt.title('Avg Age for Deaths of each race')
Out[89]: Text(0.5, 1.0, 'Avg Age for Deaths of each race')
```

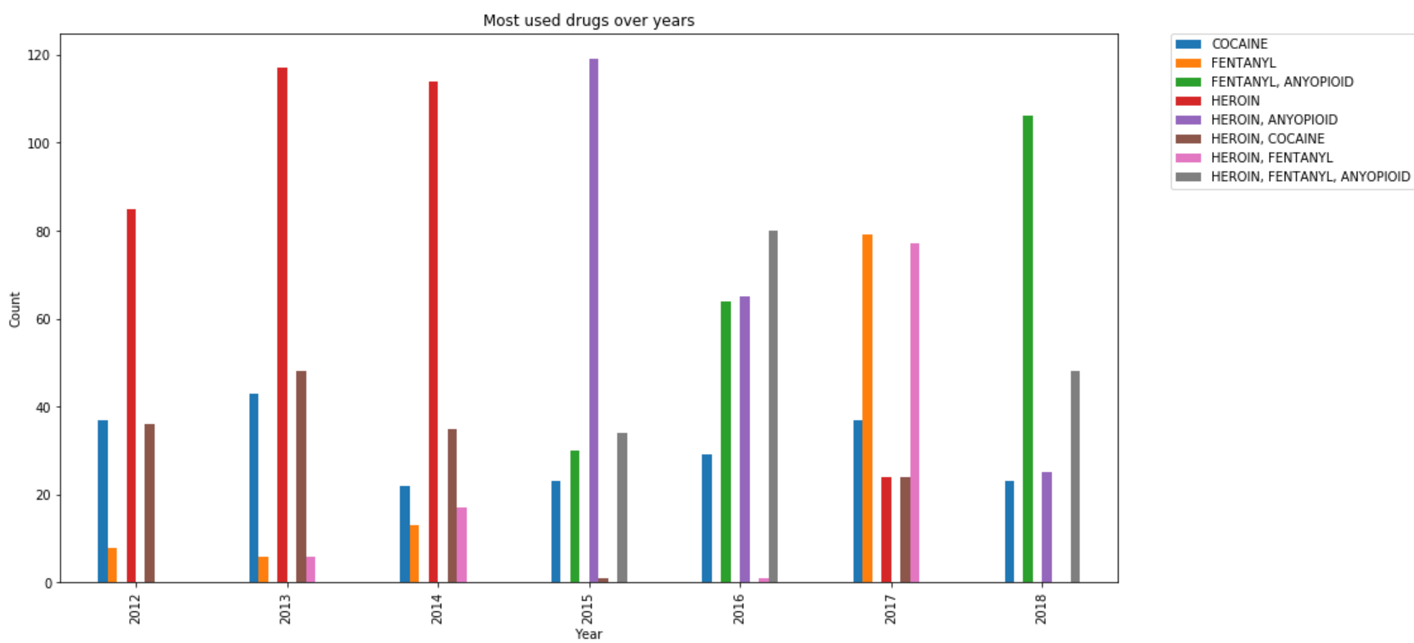


For every type of race the number of deaths of male is greater than that of Females

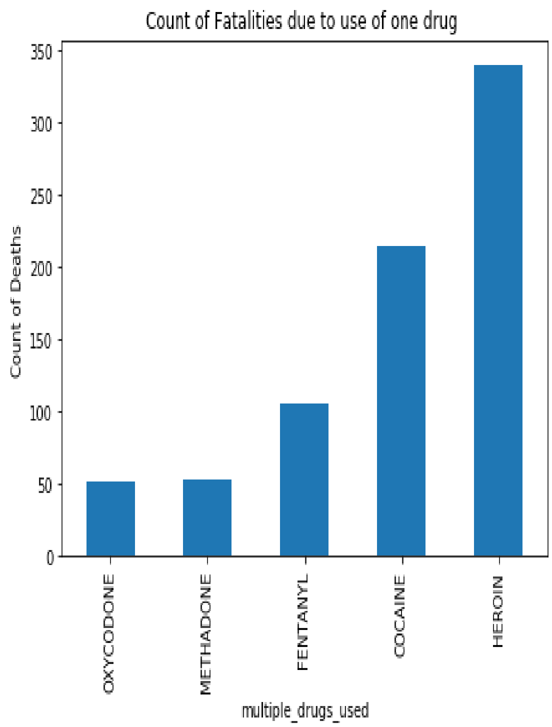


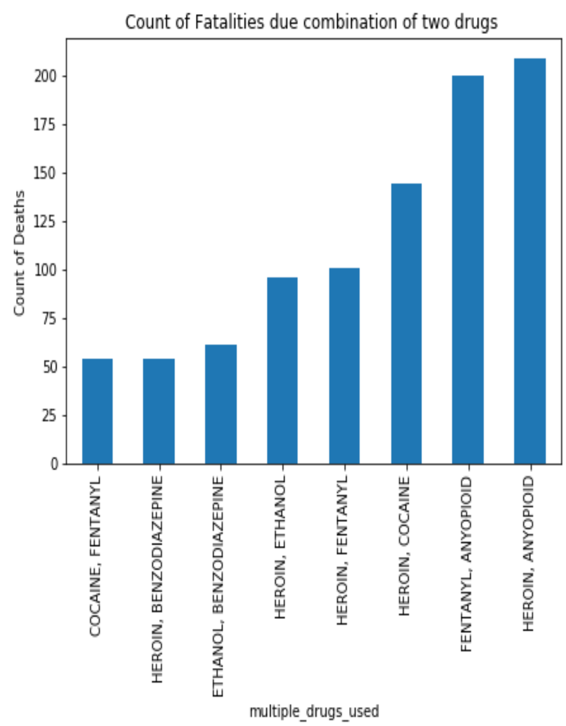
4. **What drugs causes more deaths and which city in Connecticut has more cases.**

Below we can see that from years 2012 to 2015 Heroin and opioid has been used largely. Deaths associated with Cocaine are around 40 in years 2012, 2013, 2016 and 2017. In year 2017 and 2018 there are a greater number of Fentanyl opioid associated deaths.

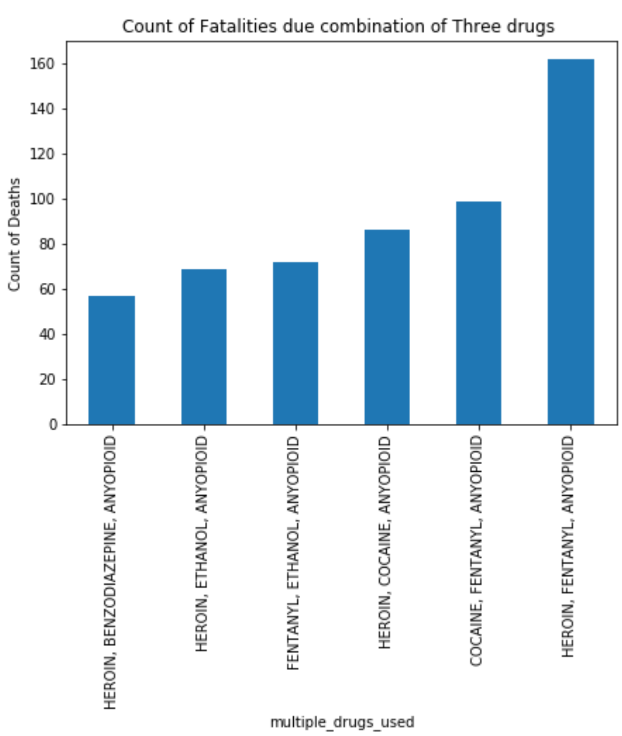


In this below we can the number of deaths caused by using one drug and combination of two drugs

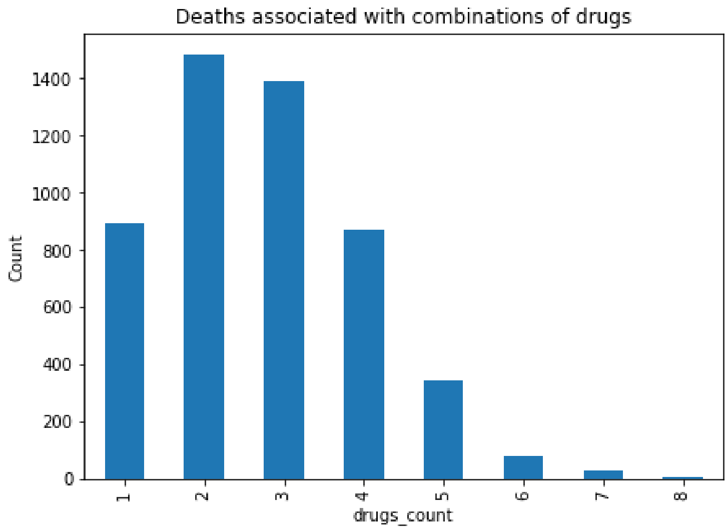




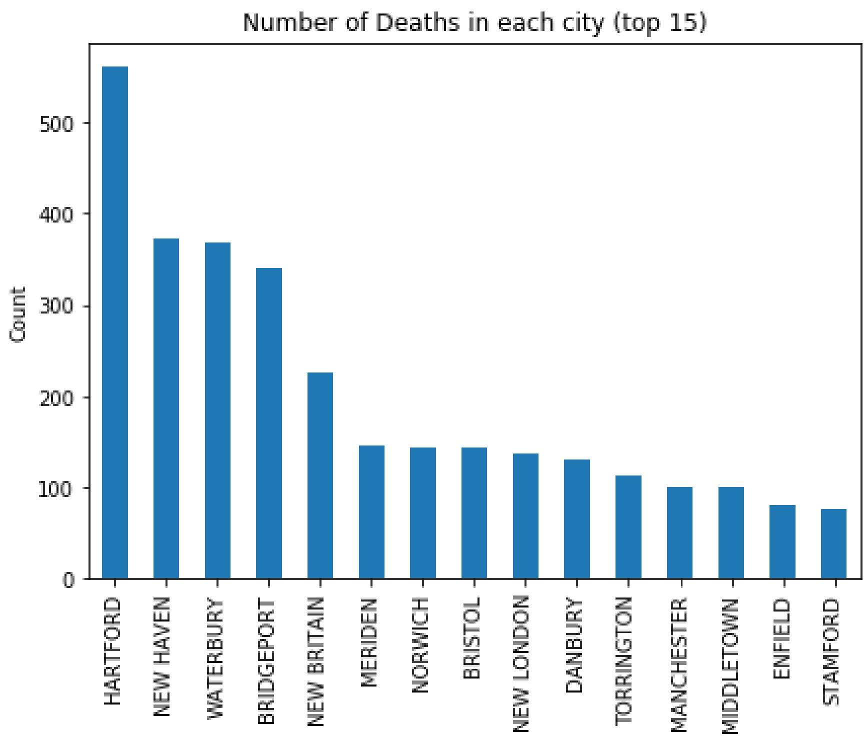
Combination of Heroin, Fentanyl and opioid has lead to more deaths.



Number of deaths caused due to combination of two and three drugs is greater use of one drug.



In the below we can that Hartford city has most of deaths which is greater than 500. New Haven, Waterbury, Bridgeport has around 350 deaths.



5. **Fit a model to predict number of deaths in next 12 months.**

```
> actuals_preds <- data.frame(cbind(actuals=final_df$Frequency, predicteds=predict(lm.fit)))
> correlation_accuracy <- cor(actuals_preds)
> correlation_accuracy
           actuals predicteds
actuals    1.0000000  0.9119513
predicteds 0.9119513  1.0000000
>
```

Better accuracy of correlation means that the actual and expected values have identical directional motion

```
> head(actuals_preds)
  actuals predicteds
1      31   25.36723
2      27   26.21982
3      24   27.07242
4      30   27.92502
5      28   28.77761
6      28   29.63021
```

Below are the prediction values for 12 months in 2019

```
> pred_vals=85:96
> predict(lm.fit, data.frame(Index = pred_vals))
        1         2         3         4         5         6         7         8         9        10        11        12 
 96.98537  97.83797  98.69056  99.54316 100.39576 101.24835 102.10095 102.95355 103.80615 104.65874 105.51134 106.36394
```

Limitations:

In this analysis I wanted to know if there is any seasonal impact on drug related deaths. But it appears that there is no seasonal impact there are almost equal number of deaths in every season. It would be great if there is a user interface for displaying all related graphs. I also wanted to know if we can prediction of city which will record highest number of deaths in a month or week.

Conclusion:

The only drug that has been declining in use over the years is cocaine alone. Heroin-including opioid poisoning fatalities started to rise dramatically, with heroin and fentanyl addiction more than rising from 2012 to 2018 . The number of overdose deaths are significantly large among Male. Hartford city has drastic number of fatalities. The correlation accuracy for linear model is 91.1% which says our model is good enough for the data to be linearly fitted.
This analysis can be used by government officials to target cities where drug related deaths are more.

References:

- [1] Accidental Drug Related Deaths 2012-2018. (2019, May 8). Retrieved March 3, 2020, from https://catalog.data.gov/dataset/accidental-drug-related-deaths-january-2012-sept-2015
- [2] Why Connecticut's drug overdose crisis isn't slowing down. (n.d.). Retrieved May 10, 2020, from https://overdose.trendct.org/story/main