Analyzing Chemical Spills in NY State (1985-2021)

Eddieb Sadat

EM-224 Informatics and Software Development
Professor Sang Won (Grace) Bae
May 15, 2021

I pledge my honor that I have abided by the Stevens Honor System.
E.S.

Project Goals

With New York's massive population and economy, it is inevitable that different industries and businesses will require handling chemicals. However, many of these chemicals can be very toxic to humans and the environments, which is why it is crucial to keep track of and to mitigate chemical spills. Thus, the goal of this project is to provide a foundation to better understand where these chemical spills are occurring, how often they are occurring, and what is causing them to occur. This foundation will allow us to better understand where to allocate time and resources in order to ensure the safety of residents and the environment.

To achieve this overall goal, it is necessary to know in which county the most number of spills occurs, what the cause of the spilled chemicals are, how often spills occur, the source of the spills, and the amount of chemicals spilled into the environment. Using this filtered data, it will be possible to create organized visuals that will easily show patterns and important information. By determining these patterns, it will then be possible to conduct field research investigations to attempt to find an underlying issue with chemical handling and regulations.

I hypothesize that initially there have been more chemical spills at the start of 1980, and then a gradual increase of spills around 2005, and then a gradual decrease in spills from 2017-2021. In the 1980's New York was not as populated or congested in the same way that it is today, and thus, I predict lower spill levels. Furthermore, at the turn of the 21st century, there was a big technological boom that soared New York's economy. With increased activity, it seems to be a fair assumption that the number of chemical spills also increased. However, around 2015 there was a push for a more environmentally-aware economy, and New York being a more progressive state, attempted to comply with new regulations to decrease pollution, which is why I predict that the number of spills from about 2017-2021 will decrease.

Overall, this dataset provides very straight-foward information for each spill incident reported. I expect that after filtering and better organizing the data, it will produce very easy-to-understand visuals and will provide very clear patterns to digest that will help solve a critical environmental and economical problem.

Business Understanding

Many of the chemicals that are used in the modern industry are classified as toxic to living organisms. However, more than environmental and health impas, there are also several underlying economic impacts that chemical spills can have.

Because chemical spills are often considered to be hazardous, there is a list of procedures and specialized equipment that must be used in order to effectively and safely contain and clean up a chemical spill. Furthermore, there must be workers who are trained and qualified to safely clean, handle, store, and to make inert the chemicals. All of this comes at a great cost, and because the New York Department of Environmental Conservation is the organization that leads the cleanups, that means chemical spill cleanups come at a cost to taxpayers. By mitigating chemical spills, it could lessen the amount that taxpayers must pay for environmental maintenance.

Another economical issue is potential real-estate value crashes. Depending on the toxicity and the amount of chemical spilled in an area, it could contaminate groundwater, cause radiation or illness concerns, and could potentially take years to cleanup. This will likely deter people from wanting to live near this area and will deplete all real-estate demand. With no demand, it will be very likely that these lands will end up drastically decreasing in price and cause current residents to lose property value.

Lastly, as people and the environment are affected by these chemical spills, it is inevitable that there will be lawsuits filed against the government or the responsible party by citizens, labor unions, and insurance companies. Not only will this lead to more taxpayer money to be spent, but it could also have dire consequences for the responsible company, who may go bankrupt from the insurance premiums or lawsuit costs.

Data Understanding

In order to prove the validity of this data set, it is important to understand how and by whom it was collected. This chemical spill data was collected by the New York Department of Environmental Conservation; Companies are required to notify authorities for any spill that occurs and the NYDEC is responsible for collecting data on each incident. Every few months, the data entries will be updated to data.gov, which is where I downloaded the 2021 updated version. As of the newest update, the CSV file contains over 550,000 data entries.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
1	Spill Num	Program F	Street 1	Street 2	Locality	County	ZIP Code	SWIS Code	DEC Regio	Spill Date	Received D	Contributi	Waterbod	Source	Close Date	Material N	Material F	Quantity	Units	Recovered
2	107132	MH 864	RT 119/MI	LLWOOD R	ELMSFORD	Westchester		6000	3	########	########	Unknown		Unknown	########	unknown (	Other	10	Gallons	0
3	405586	BOWRY BA	WATER POLL	CONTRC	QUEENS	Queens		4101	2	########	########	Other	EAST RIVEI	Unknown	########	raw sewag	Other	0	Pounds	0
4	405586	BOWRY BA	WATER POLL	CONTRC	QUEENS	Queens		4101	2	########	########	Other	EAST RIVEI	Unknown	########	raw sewag	Other	0		0
5	204667	POLE 160S	GRACE AVE/	BURKE AV	BRONX	Bronx		301	2	8/2/2002	8/2/2002	Equipment	Failure	Commerci	########	transform	Petroleum	1	Gallons	0
6	210559	POLE ON	FERDALE	LOMIS RD /	LIBERTY	Sullivan		5336	3	########	########	Traffic	Accident	Commerci	########	transform	Petroleum	6	Gallons	6
7	311484	PRIVATE R	6568 GLEN	HAVEN RD	SCOTT	Cortland		1238	7	########	########	Equipment	Failure	Private Dv	########	#2 fuel oil	Petroleum	75	Gallons	0
8	104307	149TH RD	183RD ST,	149TH AVE	QUEENS	Queens		4101	2	########	########	Abandoned	Drums	Unknown	8/1/2001	unknown (	Other	0	Gallons	0
9	160046	ABANDON	BAKER	SCHOOL HOUS	SOLON	Cortland		1200	7	########	########	Abandoned	Drums	Unknown	########	unknown (	Other	0	Gallons	0
10	9606869	AMSTERDA	WEST 79 TH	STREET	NYC	New York		3101	2	########	########	Traffic	Accident	Commerci	########	antifreeze	Other	2	Gallons	2
11	312198	APARTMEN	4 SOUTH	SIDE TERRAC	NEW PALT	Ulster		5638	3	2/2/2004	2/2/2004	Tank	Failure	Commerci	4/5/2004	#2 fuel oil	Petroleum	0	Pounds	0
12	109039	APT BUILD	94-06	34TH RD	QUEENS	Queens		4101	2	########	########	Equipment	Failure	Tank Truc	########	#2 fuel oil	Petroleum	5	Gallons	5
13	312848	BROOKLYN	29 FORT	GREEN PLAC	BROOKLYN	Kings		2401	2	########	########	Human	Error	Institutior	########	#6 fuel oil	Petroleum	10	Gallons	0
14	260048	GRAINS MI	GRAINS	MILL ROAD	TRUXTON	Cortland		1244	7	########	########	Deliberate		Unknown	########	unknown (	Other	0	Gallons	0

This data set includes incident reports dating from 1900 to 2021, and provides information on the chemical type, location of incident, amount spilled, amount recovered, and more information (refer to figure below). Once the CSV file is opened into a PANDAS dataframe, it will be very easy to visualize all the data entries.

	Spill Number	Program Facility Name	Street 1	Street 2	Locality	County	ZIP Code	SWIS Code	DEC Region	Spill Date	Received Date	Contributing Factor	Waterbody	Source	Close Date	Material Name	Material Family	Quantity	Units	Recovered
0	107132	MH 864	RT 119/MILLWOOD RD	NaN	ELMSFORD	Westchester	NaN	6000	3	10/10/2001	10/10/2001	Unknown	NaN	Unknown	10/15/2001	unknown material	Other	10.0	Gallons	0.0
1	405586	BOWRY BAY	WATER POLL CONTROL	NaN	QUEENS	Queens	NaN	4101	2	08/21/2004	08/21/2004	Other	EAST RIVER	Unknown	09/17/2004	raw sewage	Other	0.0	Pounds	0.0
2	405586	BOWRY BAY	WATER POLL CONTROL	NaN	QUEENS	Queens	NaN	4101	2	08/21/2004	08/21/2004	Other	EAST RIVER	Unknown	09/17/2004	raw sewage	Other	0.0	NaN	0.0
3	204667	POLE 16091	GRACE AVE/BURKE AVE	NaN	BRONX	Bronx	NaN	301	2	08/02/2002	08/02/2002	Equipment Failure	NaN	Commercial/Industrial	10/28/2002	transformer oil	Petroleum	1.0	Gallons	0.0
4	210559	POLE ON	FERDALE LOMIS RD / RT 52	NaN	LIBERTY	Sullivan	NaN	5336	3	01/20/2003	01/20/2003	Traffic Accident	NaN	Commercial/Industrial	01/22/2003	transformer oil	Petroleum	6.0	Gallons	6.0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...

However, as can be seen, it is quite difficult to read since there are many columns that are unneeded. Additionally, many data entries from 1900-1985 are empty or contain bad information (most likely due to not having computers to organize data), which could create outliers when organizing the data. Furthermore, many of the data entries between 1985 and 2021 are ‘NaN’, which means that PANDAS was unable to process the information due to errors that exist within the CSV data inputs. Thus, the next logical step is to use the PANDAS dataframe to filter out all of the bad data.

Data Preparation

As previously stated, the first and most important step is to get rid of all inaccurate and unwanted data from the dataframe. To do this, I first started by deleting all the columns I would not be using by using the .drop() function. Next, I sorted all the rows in the descending order of the Spill Date column, which would organize all the rows in order from 2021 to 1985. This will make reading the dataframe a lot easier, as well as allowing me to create the ‘spills per year’

graph significantly more easily. After sorting the rows, I also removed all rows that contained 'NaN' from the dataframe. This is a crucial step because Seaborn would be unable to graph the column if there are 'NaN' values. Next, I had to remove all dates prior to 1985. Because all the rows were sorted by date, it was very easy to do this, and all it required was for me to manually go into the dataframe and index all the rows from 1985 to 2021. Finally, I reset the index of the dataframe so that it would start from 0, 1, 2... . This step is another crucial step for the 'spills per year' graph since I needed to iterate through the rows using the ordered index.

This finalized dataframe (df2) now only contains rows with valid information, contains

```
import pandas as pd
import seaborn as sb
import matplotlib.pyplot as plt


df = pd.read_csv('Spill_Incidents.csv')

#filtering dataframe by removing all unneeded columns
df1 = df.drop(['Program Facility Name','Street 1', 'Street 2','Locality',
               'ZIP Code', 'SWIS Code', 'DEC Region', 'Waterbody', 'Spill Number',
               'Received Date', 'Close Date', 'Units'], axis = 1)

#removing all rows with 'NaN'
df2 = df1.sort_values('Spill Date', ascending = False).dropna()

#removing all rows prior to 1985
df2 = df2[:-58]
df2 = df2.reset_index()
df2
```

	index	County	Spill Date	Contributing Factor	Source	Material Name	Material Family	Quantity	Recovered
0	21635	Onondaga	12/31/2020	Equipment Failure	Commercial/Industrial	gasoline	Petroleum	0.0	0.0
1	327901	Westchester	12/31/2020	Equipment Failure	Private Dwelling	#2 fuel oil	Petroleum	25.0	0.0
2	162073	Dutchess	12/31/2020	Equipment Failure	Commercial Vehicle	hydraulic oil	Petroleum	7.0	0.0
3	137134	Nassau	12/31/2020	Equipment Failure	Commercial/Industrial	#2 fuel oil	Petroleum	2.0	0.0
4	368207	Nassau	12/31/2020	Equipment Failure	Private Dwelling	#2 fuel oil	Petroleum	0.0	0.0
...	...	...	...	...	...	...	...	...	...
514107	431871	Richmond	01/01/1985	Other	Unknown	unknown material	Other	0.0	0.0
514108	467330	Kings	01/01/1985	Unknown	Unknown	unknown material	Other	0.0	0.0
514109	211206	Genesee	01/01/1985	Tank Failure	Commercial/Industrial	gasoline	Petroleum	0.0	0.0
514110	471686	Nassau	01/01/1985	Unknown	Unknown	unknown petroleum	Petroleum	0.0	0.0
514111	481711	Nassau	01/01/1985	Housekeeping	Commercial/Industrial	waste oil/used oil	Petroleum	0.0	0.0

514112 rows × 9 columns

less potential outliers that could dramatically alter the graphs, and is ready to have data isolated and plugged into Seaborn to produce graphic visuals.

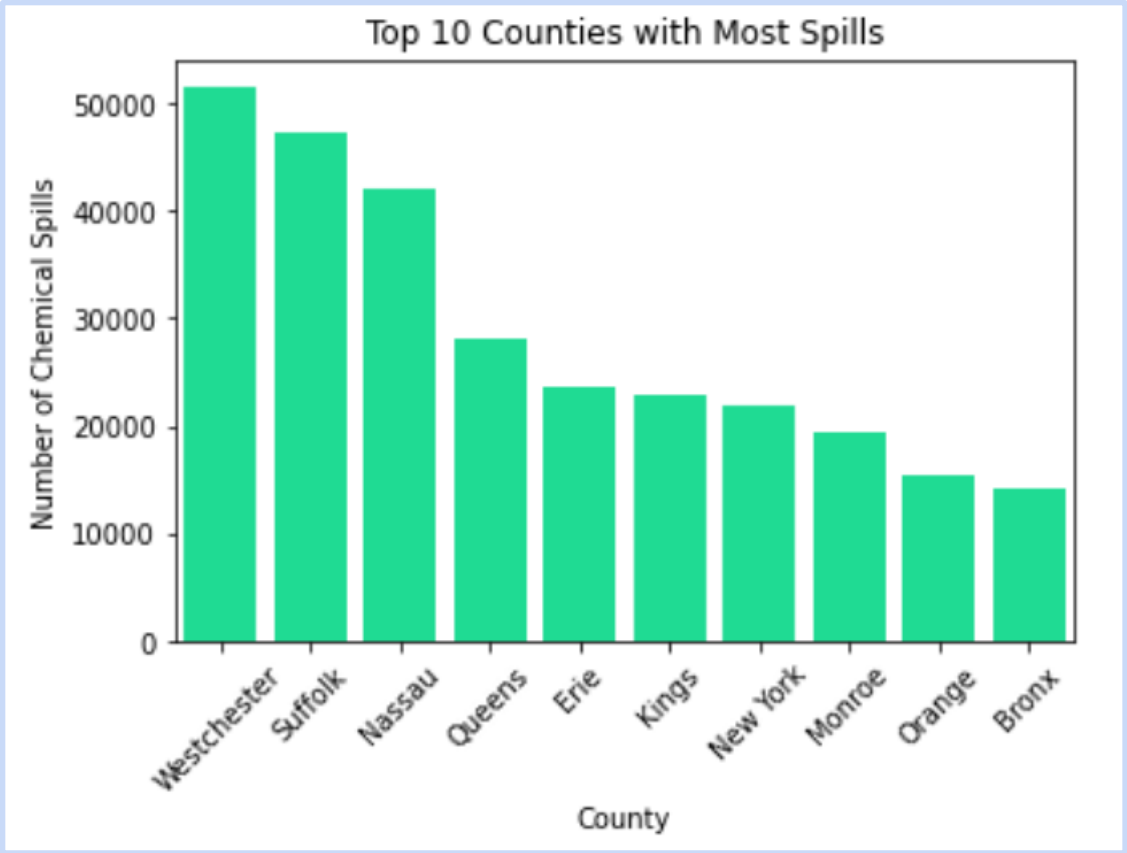
Data Representation

A very important step in creating graphic visuals is to understand which datasets to use and which graphic plot is best to visualize the desired pattern. For example, one of my key questions was to be able to answer which New York county has experienced the most number of chemical spill incidents. In order to find this, I needed to use the 'County' column, and a function called .value_counts() to take the summation of all unique occurrences in the column - essentially acting as a counter for all unique items. The way .value_counts() stores this information by creating a new dataframe using the County names as the index and the number of occurrences as the first column.

```
Westchester                                51535
Suffolk                                    47189
Nassau                                     41954
Queens                                     28083
Erie                                       23714
                                           ...
Oil Springs Indian Reservation                 1
Canada - Region 5                              1
Connecticut - Region 4                         1
St. Regis Indian Reservation - Region 5        1
Shinnecock Indian Reservation                  1
Name: County, Length: 86, dtype: int64
```

The next step is to understand how to use this output in order to create a Seaborn plot. Since I am trying to compare the number of spill incidents per county, I need to use the index (county name) as my x-axis data, and the first column (number of spill incidents) as my y-axis. Since I am trying to compare the top counties to each other, I need a data-representation that will isolate each county name and compare the incident values to each other. Thus, the plot type I thought would be best to use is the barplot, since it compares all individual counties by the total number of incidents.

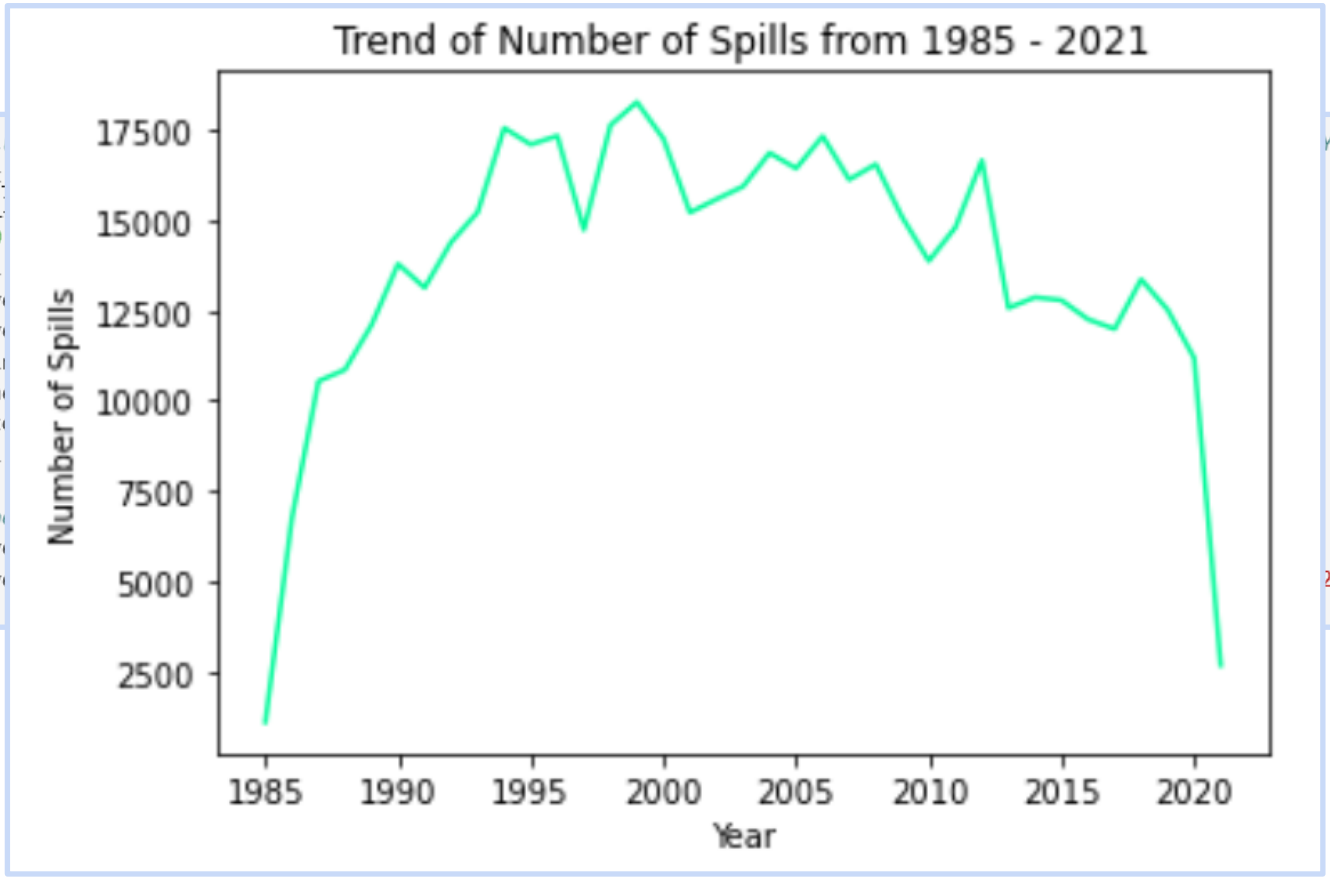
```
#counting number of reoccurances per unique item (county)
county = df2['County'].value_counts()

#bar graph using top 10 counties
barcounty = sb.barplot(x = county.index[:10], y = county.head(10), color = 'mediumspringgreen')
barcounty.set_xticklabels(barcounty.get_xticklabels(), rotation=45) #Rotate xlabel for legibilaty
barcounty.set(xlabel = 'County', ylabel = 'Number of Chemical Spills', title = 'Top 10 Counties with Most Spills')
```



For all key questions like “Top 10 Counties with Most Spills'” where it compared the number of spill incidents to a column, I decided that the best plot type would be barplot. I could have used other visuals, such as a pieplot, but I wanted to keep all my outputs consistent, and thus stuck with the same plot-type for all of them.

Only one dataset was not a barplot, and it was the “Trend of Number of Spills from 1985 - 2021” plot. For this one, I had to manually compile the total occurrences per year, and because I was trying to show the trend of the number of spills per year, I decided that a lineplot would best plot for this. For my x-axis I used the year, and for my y-axis I used the number of occurrences. When graphed, this lineplot will plot and connect each coordinate, which will clearly show the slope trends of incidents per year; A negative slope indicates that the number of

incidents has decreased, and a positive slope means that the number of incidents has increased.



Now that I had a visual-type for each key question, I compiled all the data and represented them in their individual Seaborn plots. All code and visual plots can be found in the Attachments section of this report.

Practical Results - Conclusion

To start, I had to find the most updated CSV file of the New York State Chemical Spills incident report. By using the PANDAS library and reading the CSV into a dataframe, I was able to manually go through each column and row to determine which were unnecessary to reach my goals. Furthermore, by quickly scanning the rows, I determined that any date prior to 1985 could potentially be inaccurate, as well as identifying the need to remove any rows with 'NaN' values.

Once the dataframe was filtered, I was finally able to start answering my key questions by using tools such as .value_counts() to find the total number of occurrences of unique items, and str.contains() to find all rows specific to a single year. Next, using this refined data, I was able to create Seaborn barplots and a lineplot to accuracelty represent a visual that displays patterns that help answer my key questions.

Analysis

In the end, I was able to answer all my key questions and identify patterns that would have otherwise been near-impossible by reading the CSV. I found that of all counties in New York, the most likely county to experience a chemical spill are Weschetser, Suffolk, and Nassau.

From this, we can infer that these three counties are not only highly populated, but are also travel hubs for Fuel and Chemical tankers. Next, I determined that the most commonly spilled chemical is Fuel #2, which is the same fuel that heats residential and commercial buildings, as well as powers stovetops. Again, this makes sense because it is the most commonly used fuel, followed by gasoline and diesel - which were also among the top five most spilled chemicals. Lastly, I was able to determine that my hypothesis on the number of spills per year was fairly correct. From 1985-1998, the number of incidents per year steadily rose, but from around 2005-2021, there seems to be a steady decrease in the number of occurrences per year.

In summary, I think that the New York Department of Environmental Conservation can improve their field data collection, because many of the data entries turned out to be 'NaN' , non-existent, or inaccurate. I would recommend giving more training to field workers and looking into chemical containment regulations to possibly mitigate future chemical spills from Industrial companies or residential pipelines.

Overall I thought this project was really fun to do, and provided a more real-life example of the methodologies used for data visualization. I definitely see the value in using PANDAS and Seaborn, because trying to manually read through half-a-million data entries is nearly impossible, and frankly, sounds like a migraine.

Attachments

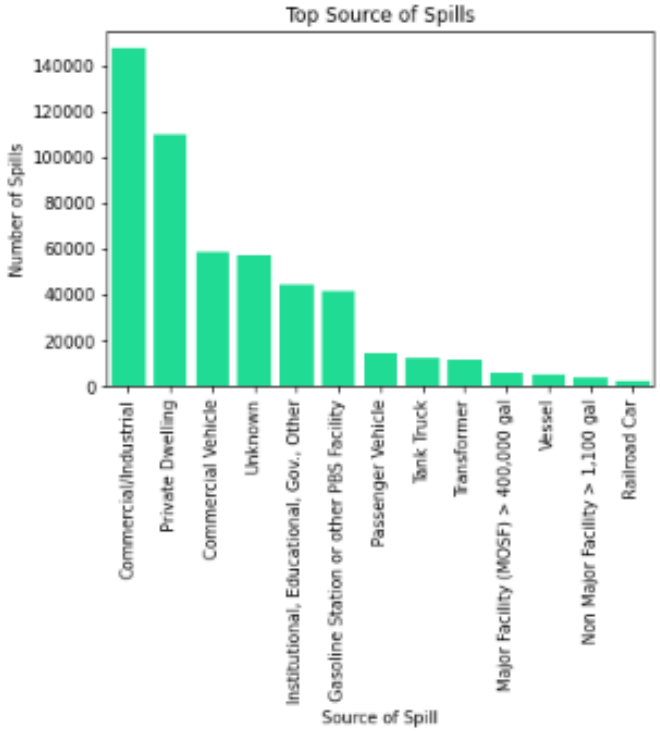
```
import pandas as pd
import seaborn as sb
import matplotlib.pyplot as plt


df = pd.read_csv('Spill_Incidents.csv')

#filtering dataframe by removing all unneeded columns
df1 = df.drop(['Program Facility Name','Street 1', 'Street 2','Locality',
               'ZIP Code', 'SWIS Code', 'DEC Region', 'Waterbody', 'Spill Number',
               'Received Date', 'Close Date', 'Units'], axis = 1)

#removing all rows with 'NaN'
df2 = df1.sort_values('Spill Date', ascending = False).dropna()

#removing all rows prior to 1985
df2 = df2[:-58]
df2 = df2.reset_index()
df2
```

	index	County	Spill Date	Contributing Factor	Source	Material Name	Material Family	Quantity	Recovered
0	21635	Onondaga	12/31/2020	Equipment Failure	Commercial/Industrial	gasoline	Petroleum	0.0	0.0
1	327901	Westchester	12/31/2020	Equipment Failure	Private Dwelling	#2 fuel oil	Petroleum	25.0	0.0
2	162073	Dutchess	12/31/2020	Equipment Failure	Commercial Vehicle	hydraulic oil	Petroleum	7.0	0.0
3	137134	Nassau	12/31/2020	Equipment Failure	Commercial/Industrial	#2 fuel oil	Petroleum	2.0	0.0
4	368207	Nassau	12/31/2020	Equipment Failure	Private Dwelling	#2 fuel oil	Petroleum	0.0	0.0
...	...	...	...	...	...	...	...	...	...
514107	431871	Richmond	01/01/1985	Other	Unknown	unknown material	Other	0.0	0.0
514108	467330	Kings	01/01/1985	Unknown	Unknown	unknown material	Other	0.0	0.0
514109	211206	Genesee	01/01/1985	Tank Failure	Commercial/Industrial	gasoline	Petroleum	0.0	0.0
514110	471686	Nassau	01/01/1985	Unknown	Unknown	unknown petroleum	Petroleum	0.0	0.0
514111	481711	Nassau	01/01/1985	Housekeeping	Commercial/Industrial	waste oil/used oil	Petroleum	0.0	0.0

514112 rows × 9 columns

41]:
```
#number of occurances
source = df2['Source'].value_counts()

#Top 13 spill sources
barsource = sb.barplot(x = source.index[:13], y = source.head(13), color = 'mediumspringgreen')
barsource.set_xticklabels(barsource.get_xticklabels(), rotation=90) #Rotate xlabel for legibilaty
barsource.set(xlabel = 'Source of Spill', ylabel = 'Number of Spills', title = 'Top Source of Spills')
```

41]:
```
[Text(0.5, 0, 'Source of Spill'),
 Text(0, 0.5, 'Number of Spills'),
 Text(0.5, 1.0, 'Top Source of Spills')]
```

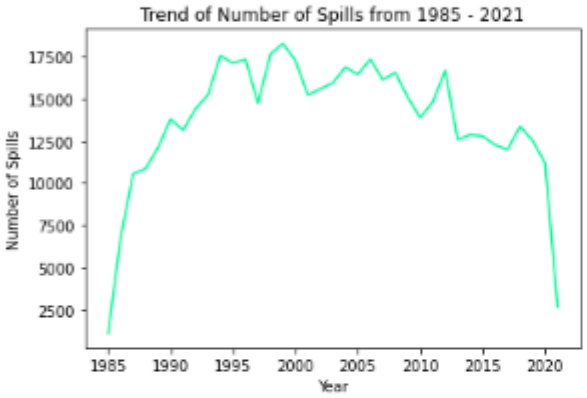


47]:
```
#summation of quantity column
quantity = df2['Quantity'].sum()
print("Total Number of Spilled Chemicals from 1985-2021")
print(quantity, '\n')

#summation of recovered column
recovered = df2['Recovered'].sum()
print("Total Number of Recovered Chemical Spillage from 1985-2021")
print(recovered, '\n')

#difference of quantity and recovered
print("Total Number of Non-Recovered Spillage")
print(quantity - recovered)
```

```
Total Number of Spilled Chemicals from 1985-2021
101139970220.8

Total Number of Recovered Chemical Spillage from 1985-2021
243350045.44999996

Total Number of Non-Recovered Spillage
100896620175.35
```

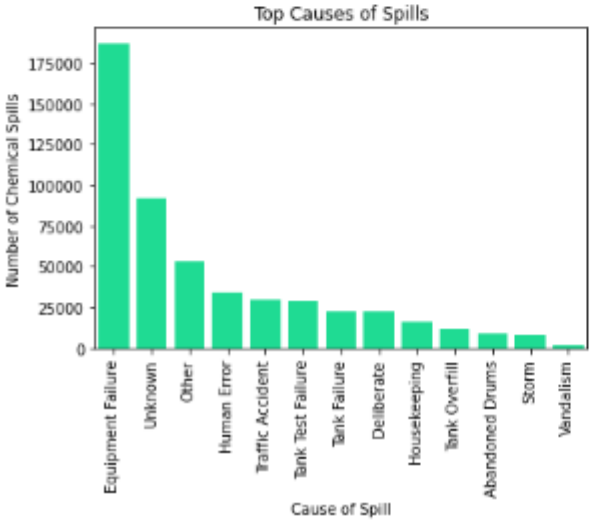
```
#Spill Date is in MM/DD/YYYY format, however this does not organize items well enough - Isolating to only YYYY
count_list = []
year_list = []
i = 0
for i in range(37): #range of 36 years from 1985-2021
    year = 1985 + i
    year_list.append(year)
    index_df = df2['Spill Date'].str.contains(str(year)) #extracing specific years
    newdf = df2.loc[index_df]
    count_list.append(len(newdf)) #appending years with number of spills per year
    i += 1

#number of spills per year from 1985-2021
lineyear = sb.lineplot(x = year_list, y = count_list, color = "mediumspringgreen")
lineyear.set(xlabel = "Year", ylabel = 'Number of Spills', title = 'Trend of Number of Spills from 1985 - 2021')
```

```
[Text(0.5, 0, 'Year'),
 Text(0, 0.5, 'Number of Spills'),
 Text(0.5, 1.0, 'Trend of Number of Spills from 1985 - 2021')]
```



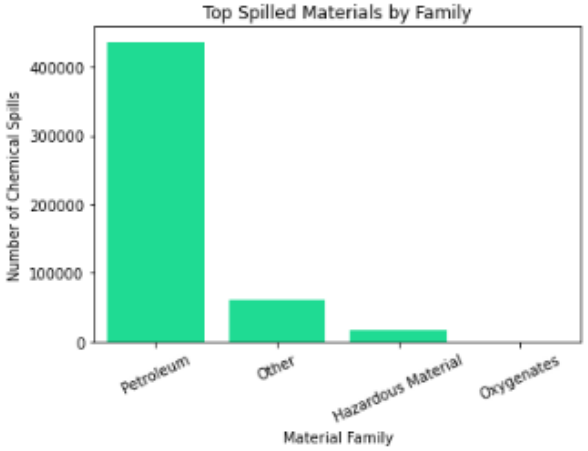

```
#number of reoccurances
factor = df2['Contributing Factor'].value_counts()

#Top 13 spill causes
barfactor = sb.barplot(x = factor.index[:13], y = factor.head(13), color = 'mediumspringgreen')
barfactor.set_xticklabels(barfactor.get_xticklabels(), rotation=90) #Rotate xlabel for legibility
```

```
[Text(0.5, 0, 'Cause of Spill'),
 Text(0, 0.5, 'Number of Chemical Spills'),
 Text(0.5, 1.0, 'Top Causes of Spills')]
```



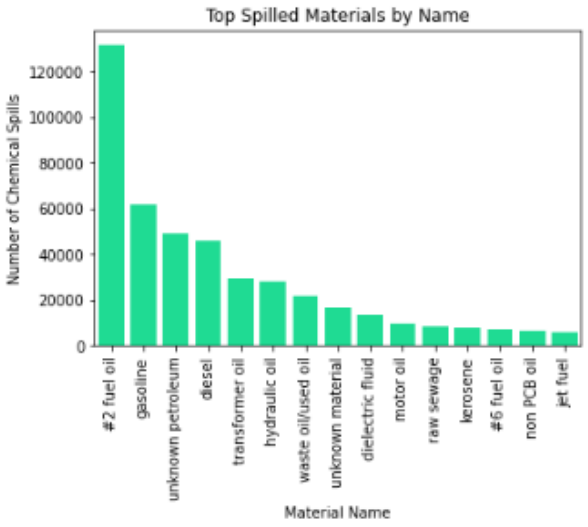
```
[17]: #number of occurances
materialfamily = df2['Material Family'].value_counts()

#Top 5 material families
barmf = sb.barplot(x=materialfamily.index[:5], y = materialfamily.head(), color = 'mediumspringgreen')
barmf.set_xticklabels(barmf.get_xticklabels(), rotation=25) #Rotate xlabel for legibilaty
barmf.set(xlabel = "Material Family", ylabel = "Number of Chemical Spills", title = "Top Spilled Materials by Family")
```

```
[17]: [Text(0.5, 0, 'Material Family'),
 Text(0, 0.5, 'Number of Chemical Spills'),
 Text(0.5, 1.0, 'Top Spilled Materials by Family')]
```




```
[49]: #number of occurances
materialname = df2["Material Name"].value_counts()

#top 15 materials by name
barmn = sb.barplot(x = materialname.index[:15], y = materialname.head(15), color = 'mediumspringgreen')
barmn.set_xticklabels(barmn.get_xticklabels(), rotation=90) #Rotate xlabel for legibilaty
barmn.set(xlabel = 'Material Name', ylabel = 'Number of Chemical Spills', title = 'Top Spilled Materials by Name')
```

```
[49]: [Text(0.5, 0, 'Material Name'),
 Text(0, 0.5, 'Number of Chemical Spills'),
 Text(0.5, 1.0, 'Top Spilled Materials by Name')]
```



```
#summation of quantity column
quantity = df2['Quantity'].sum()
print("Total Number of Spilled Chemicals from 1985-2021")
print(quantity, '\n')

#summation of recovered column
recovered = df2['Recovered'].sum()
print("Total Number of Recovered Chemical Spillage from 1985-2021")
print(recovered, '\n')

#difference of quantity and recovered
print("Total Number of Non-Recovered Spillage")
print(quantity - recovered)
```

```
Total Number of Spilled Chemicals from 1985-2021
101139970220.8

Total Number of Recovered Chemical Spillage from 1985-2021
243350045.44999996

Total Number of Non-Recovered Spillage
100896620175.35
```

Source:

https://catalog.data.gov/dataset/spill-incidents/resource/a8f9d3c8-c3fa-4ca1-a97a-55e55ca6f8c0