In [1]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
sns.set_style('darkgrid')
```

In [16]:
```
iowa = pd.read_csv("Iowa_Liquor_Sales.csv")
```

```
C:\Users\prane\anaconda3\lib\site-packages\IPython\core\interactiveshell.py:3165: DtypeWarning: Columns (6) have mixed
types.Specify dtype option on import or set low_memory=False.
  has_raised = await self.run_ast_nodes(code_ast.body, cell_name,
```

In [17]:
```
iowa
```

Out[17]:

	Invoice/Item Number	Date	Store Number	Store Name	Address	City	Zip Code	Store Location	County Number	County	...	Item Number
0	S29198800001	11/20/2015	2191	Keokuk Spirits	1013 MAIN	KEOKUK	52632	1013 MAIN\nKEOKUK 52632\n(40.39978, -91.387531)	56.0	Lee	...	297
1	S29195400002	11/21/2015	2205	Ding's Honk And Holler	900 E WASHINGTON	CLARINDA	51632	900 E WASHINGTON\nCLARINDA 51632\n(40.739238, ...	73.0	Page	...	297
2	S29050300001	11/16/2015	3549	Quicker Liquor Store	1414 48TH ST	FORT MADISON	52627	1414 48TH ST\nFORT MADISON 52627\n(40.624226, ...	56.0	Lee	...	249
3	S28867700001	11/04/2015	2513	Hy-Vee Food Store #2 / Iowa City	812 S 1ST AVE	IOWA CITY	52240	812 S 1ST AVE\nIOWA CITY 52240\n	52.0	Johnson	...	237
4	S29050800001	11/17/2015	3942	Twin Town Liquor	104 HIGHWAY 30 WEST	TOLEDO	52342	104 HIGHWAY 30 WEST\nTOLEDO 52342\n(41.985887,...	86.0	Tama	...	249
...	...	...	...	...	...	...	...	...	...	...	...	...
12591072	INV-08368000074	10/31/2017	5423	Stammer Liquor Corp	615 2nd Ave	Sheldon	51201	615 2nd Ave\nSheldon 51201\n(43.184614, -95.85...	71.0	OBRIEN	...	73802
12591073	INV-08368000075	10/31/2017	5423	Stammer Liquor Corp	615 2nd Ave	Sheldon	51201	615 2nd Ave\nSheldon 51201\n(43.184614, -95.85...	71.0	OBRIEN	...	20375
12591074	INV-08368000076	10/31/2017	5423	Stammer Liquor Corp	615 2nd Ave	Sheldon	51201	615 2nd Ave\nSheldon 51201\n(43.184614, -95.85...	71.0	OBRIEN	...	20372
12591075	INV-08368000077	10/31/2017	5423	Stammer Liquor Corp	615 2nd Ave	Sheldon	51201	615 2nd Ave\nSheldon 51201\n(43.184614, -95.85...	71.0	OBRIEN	...	20369
12591076	INV-08368000078	10/31/2017	5423	Stammer Liquor Corp	615 2nd Ave	Sheldon	51201	615 2nd Ave\nSheldon 51201\n(43.184614, -95.85...	71.0	OBRIEN	...	77309

12591077 rows × 24 columns

In [18]:
```
iowa.shape
```

Out[18]:
```
(12591077, 24)
```

In [19]:
```
iowa.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 12591077 entries, 0 to 12591076
Data columns (total 24 columns):
 #   Column                 Dtype
---  ------                 -----
 0   Invoice/Item Number    object
 1   Date                   object
 2   Store Number           int64
 3   Store Name             object
 4   Address                object
 5   City                   object
 6   Zip Code               object
 7   Store Location         object
 8   County Number          float64
 9   County                 object
 10  Category               float64
 11  Category Name          object
 12  Vendor Number          float64
 13  Vendor Name            object
 14  Item Number            int64
 15  Item Description       object
 16  Pack                   int64
 17  Bottle Volume (ml)     int64
 18  State Bottle Cost      object
 19  State Bottle Retail    object
 20  Bottles Sold           int64
 21  Sale (Dollars)         object
 22  Volume Sold (Liters)   float64
 23  Volume Sold (Gallons)  float64
dtypes: float64(5), int64(5), object(14)
memory usage: 2.3+ GB
```

In [20]:
```
iowa.columns
```

Out[20]:
```
Index(['Invoice/Item Number', 'Date', 'Store Number', 'Store Name', 'Address',
       'City', 'Zip Code', 'Store Location', 'County Number', 'County',
       'Category', 'Category Name', 'Vendor Number', 'Vendor Name',
       'Item Number', 'Item Description', 'Pack', 'Bottle Volume (ml)',
       'State Bottle Cost', 'State Bottle Retail', 'Bottles Sold',
       'Sale (Dollars)', 'Volume Sold (Liters)', 'Volume Sold (Gallons)'],
      dtype='object')
```

In [28]:
```
iowa.duplicated().sum()
```

Out[28]:
```
0
```

In [29]:
```
iowa[iowa.duplicated()]
```

Out[29]:

Invoice/Item Number	Date	Store Number	Store Name	Address	City	Zip Code	Store Location	County Number	County	...	Item Number	Item Description	Pack	Bottle Volume (ml)	State Bottle Cost ($)	State Bottle Retail ($)	Bottl Sc

0 rows × 24 columns

In [30]:
```
iowa.describe()
```

Out[30]:

	Store Number	County Number	Category	Vendor Number	Item Number	Pack	Bottle Volume (ml)	Bottles Sold	Volume Sold (Liters)	Volume So (Gallon
count	1.259108e+07	1.251190e+07	1.258306e+07	1.259107e+07	1.259108e+07	1.259108e+07	1.259108e+07	1.259108e+07	1.259108e+07	1.259108e+(
mean	3.565216e+03	5.724050e+01	1.044710e+06	2.574911e+02	4.603682e+04	1.222533e+01	9.289402e+02	8.140392e+00	7.489119e+00	1.977358e+(
std	9.312721e+02	2.726983e+01	5.435094e+04	1.416175e+02	5.301684e+04	7.458673e+00	7.340448e+02	2.217891e+01	2.679149e+01	7.077822e+(
min	2.106000e+03	1.000000e+00	1.012200e+05	1.000000e+01	1.010000e+02	1.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+(
25%	2.604000e+03	3.100000e+01	1.012210e+06	1.150000e+02	2.705600e+04	6.000000e+00	7.500000e+02	2.000000e+00	1.500000e+00	4.000000e-(
50%	3.704000e+03	6.200000e+01	1.031200e+06	2.600000e+02	3.817700e+04	1.200000e+01	7.500000e+02	4.000000e+00	3.000000e+00	7.900000e-(
75%	4.304000e+03	7.700000e+01	1.062310e+06	3.800000e+02	6.375500e+04	1.200000e+01	1.000000e+03	1.200000e+01	9.000000e+00	2.380000e+(
max	9.932000e+03	9.900000e+01	1.901200e+06	9.870000e+02	9.992750e+05	6.000000e+02	3.780000e+05	1.500000e+04	1.500000e+04	3.962580e+(

In [31]:
```
iowa.nunique()
```

Out[31]:
```
Invoice/Item Number        12591077
Date                           1379
Store Number                   1884
Store Name                     1952
Address                        3154
City                            793
Zip Code                        871
Store Location                 4477
County Number                    99
County                          200
Category                        107
Category Name                   130
Vendor Number                   271
Vendor Name                     393
Item Number                    7395
Item Description               5865
Pack                             30
Bottle Volume (ml)               49
State Bottle Cost ($)          2768
State Bottle Retail ($)        3224
Bottles Sold                    450
Sale (Dollars)                23154
Volume Sold (Liters)           1180
Volume Sold (Gallons)          1184
dtype: int64
```

In [32]:
```
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
%config InlineBackend.figure_format = 'retina' # HD!
```

In [33]:
```
iowa.rename(columns={'State Bottle Retail': 'State Bottle Retail ($)', 'State Bottle Cost': 'State Bottle Cost ($)'}, ir
```

In [34]:
```
iowa.head()
```

Out[34]:

	Invoice/Item Number	Date	Store Number	Store Name	Address	City	Zip Code	Store Location	County Number	County	...	Item Number	It Descript
0	S29198800001	11/20/2015	2191	Keokuk Spirits	1013 MAIN	KEOKUK	52632	1013 MAIN\nKEOKUK 52632\n(40.39978, -91.387531)	56.0	Lee	...	297	Temple Rye w/Fl
1	S29195400002	11/21/2015	2205	Ding's Honk And Holler	900 E WASHINGTON	CLARINDA	51632	900 E WASHINGTON\nCLARINDA 51632\n(40.739238, ...	73.0	Page	...	297	Temple Rye w/Fl
2	S29050300001	11/16/2015	3549	Quicker Liquor Store	1414 48TH ST	FORT MADISON	52627	1414 48TH ST\nFORT MADISON 52627\n(40.624226, ...	56.0	Lee	...	249	Disaron Amare Cav Mignor 50ml P
3	S28867700001	11/04/2015	2513	Hy-Vee Food Store #2 / Iowa City	812 S 1ST AVE	IOWA CITY	52240	812 S 1ST AVE\nIOWA CITY 52240\n	52.0	Johnson	...	237	Knob Cr w/ Cry Decar
4	S29050800001	11/17/2015	3942	Twin Town Liquor	104 HIGHWAY 30 WEST	TOLEDO	52342	104 HIGHWAY 30 WEST\nTOLEDO 52342\n(41.985887,...	86.0	Tama	...	249	Disaron Amare Cav Mignor 50ml P

5 rows × 24 columns

In [35]:
```
# Getting rid of $ in each value of column so I can use values as floats
iowa['State Bottle Retail ($)'] = iowa['State Bottle Retail ($)'].str.replace('$', '')
iowa['State Bottle Cost ($)'] = iowa['State Bottle Cost ($)'].str.replace('$', '')
iowa['Sale (Dollars)'] = iowa['Sale (Dollars)'].str.replace('$', '')

# changing data type to floats
iowa['Sale (Dollars)'] = iowa['Sale (Dollars)'].astype('float')
iowa['State Bottle Cost ($)'] = iowa['State Bottle Cost ($)'].astype('float')
iowa['State Bottle Retail ($)'] = iowa['State Bottle Retail ($)'].astype('float')
```

```
<ipython-input-35-18ebd6678347>:2: FutureWarning: The default value of regex will change from True to False in a futur
e version. In addition, single character regular expressions will*not* be treated as literal strings when regex=True.
  iowa['State Bottle Retail ($)'] = iowa['State Bottle Retail ($)'].str.replace('$', '')
<ipython-input-35-18ebd6678347>:3: FutureWarning: The default value of regex will change from True to False in a futur
e version. In addition, single character regular expressions will*not* be treated as literal strings when regex=True.
  iowa['State Bottle Cost ($)'] = iowa['State Bottle Cost ($)'].str.replace('$', '')
<ipython-input-35-18ebd6678347>:4: FutureWarning: The default value of regex will change from True to False in a futur
e version. In addition, single character regular expressions will*not* be treated as literal strings when regex=True.
  iowa['Sale (Dollars)'] = iowa['Sale (Dollars)'].str.replace('$', '')
```

In [36]:
```
iowa.dtypes
```

Out[36]:
```
Invoice/Item Number         object
Date                        object
Store Number                 int64
Store Name                  object
Address                     object
City                        object
Zip Code                    object
Store Location              object
County Number              float64
County                      object
Category                   float64
Category Name               object
Vendor Number              float64
Vendor Name                 object
Item Number                  int64
Item Description            object
Pack                         int64
Bottle Volume (ml)           int64
State Bottle Cost ($)      float64
State Bottle Retail ($)    float64
Bottles Sold                 int64
Sale (Dollars)             float64
Volume Sold (Liters)       float64
Volume Sold (Gallons)      float64
dtype: object
```

In [37]:
```
iowa[['State Bottle Cost ($)', 'State Bottle Retail ($)']].describe()
```

Out[37]:

	State Bottle Cost ($)	State Bottle Retail ($)
count	1.259107e+07	1.259107e+07
mean	9.659111e+00	1.451134e+01
std	1.174101e+01	1.761109e+01
min	0.000000e+00	0.000000e+00
25%	5.500000e+00	8.250000e+00
50%	7.960000e+00	1.199000e+01
75%	1.175000e+01	1.763000e+01
max	7.680000e+03	1.152000e+04

In [38]:
```
# Creating new column of net profit per bottle for each bottle
iowa['State Profit Per Bottle ($)'] = iowa['State Bottle Retail ($)'] - iowa['State Bottle Cost ($)']
```

In [41]:
```
# Grouping by vendor
iowa.groupby(['Vendor Number'])[['State Bottle Cost ($)', 'State Bottle Retail ($)']].describe()
```

Out[41]:

	State Bottle Cost ($)								State Bottle Retail ($)							
	count	mean	std	min	25%	50%	75%	max	count	mean	std	min	25%	50%	75%	max
Vendor Number																
10.0	868.0	11.636982	3.246844	10.66	10.66	11.00	11.00	66.66	868.0	17.455495	4.870361e+00	15.99	15.99	16.500	16.50	99.99
14.0	1.0	5.990000	NaN	5.99	5.99	5.99	5.99	5.99	1.0	8.990000	NaN	8.99	8.99	8.990	8.99	8.99
27.0	3.0	4.400000	0.000000	4.40	4.40	4.40	4.40	4.40	3.0	6.600000	1.087792e-15	6.60	6.60	6.600	6.60	6.60
33.0	12.0	43.599167	18.198587	26.00	26.00	43.42	60.17	65.00	12.0	65.401667	2.729910e+01	39.00	39.00	65.135	90.26	97.50
35.0	601459.0	11.627173	6.148947	0.43	7.53	9.50	15.00	88.84	601459.0	17.499314	9.235583e+00	0.65	11.31	14.250	22.50	133.26
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
969.0	1565.0	22.555086	8.474932	4.86	13.00	24.83	29.34	55.92	1565.0	33.831508	1.271478e+01	7.29	19.50	37.250	44.01	83.88
971.0	18654.0	16.826948	6.873388	4.30	13.10	13.10	20.00	131.22	18654.0	25.395054	1.010301e+01	6.45	19.65	19.650	30.00	93.38
977.0	598.0	10.093328	0.705079	9.50	9.59	10.09	10.09	12.08	598.0	15.152525	1.057127e+00	14.38	14.39	15.140	15.14	18.13
978.0	5086.0	13.164847	2.025961	7.46	13.33	14.10	14.10	28.34	5086.0	19.769969	3.038904e+00	11.19	19.99	21.150	21.15	42.51
987.0	31.0	8.938387	0.688230	6.87	9.16	9.16	9.16	9.16	31.0	13.408065	1.030842e+00	10.31	13.74	13.740	13.74	13.74

271 rows × 16 columns

In [42]:
```
iowa['Category'].nunique()
```

Out[42]: 107

In [43]:
```
iowa['Vendor Number'].nunique()
```

Out[43]: 271

In [44]:
```
iowa.groupby('Category Name')[['State Bottle Cost ($)', 'State Bottle Retail ($)', 'State Profit Per Bottle ($)']].std(
```

Out[44]:

Category Name	State Bottle Cost ($)	State Bottle Retail ($)	State Profit Per Bottle ($)
100 PROOF VODKA	2.659414	3.989338	1.330062
100% Agave Tequila	11.867804	17.791870	5.924422
AMARETTO - IMPORTED	0.000000	0.000000	0.000000
AMERICAN ALCOHOL	0.852801	1.278575	0.425779
AMERICAN AMARETTO	1.497517	2.245276	0.747766
...	...	...	...
WHISKEY LIQUEUR	7.360404	11.037647	3.677346
WHITE CREME DE CACAO	0.578364	0.867803	0.289464
WHITE CREME DE MENTHE	0.099751	0.151195	0.051564
Whiskey Liqueur	8.613618	12.919482	4.305864
White Rum	3.352310	5.028339	1.676029

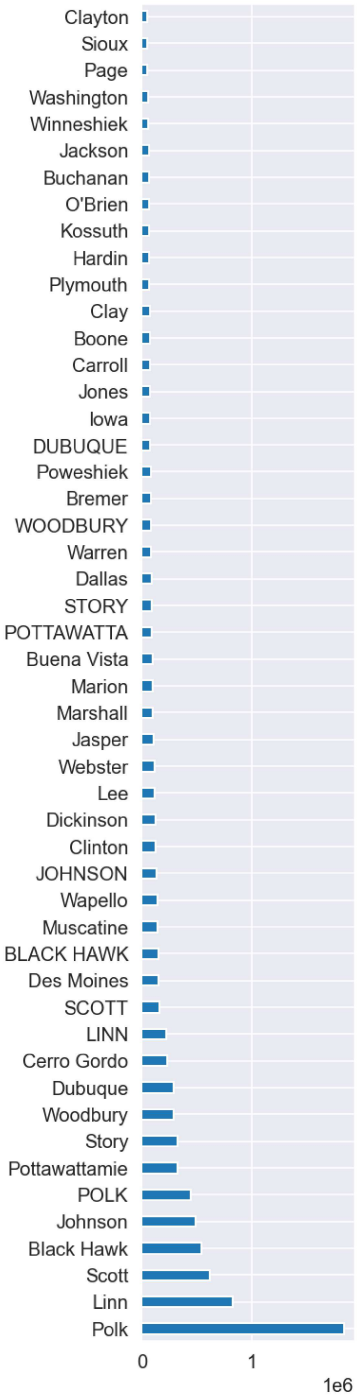
130 rows × 3 columns

In [45]:
```
# dataframe of the most prominent counties in Iowa to show up in the data
pop_counties = iowa['County'].value_counts()
pop_counties
```

Out[45]:
```
Polk          1833520
Linn           827574
Scott          621384
Black Hawk     535908
Johnson        490358
               ...
RINGGOLD         2004
WAYNE            1563
Fremont           779
FREMONT           464
Pottawatta        407
Name: County, Length: 200, dtype: int64
```

In [46]:
```
# bar graph representing the most prominent counties in dataset
# dropoff after Cerro Gordo
pop_counties.head(50).plot(kind='barh', width=0.4, figsize=(2,13))
```

Out[46]:
```
<AxesSubplot:>
```


Clayton
Sioux
Page
Washington
Winneshiek
Jackson
Buchanan
O'Brien
Kossuth
Hardin
Plymouth
Clay
Boone
Carroll
Jones
Iowa
DUBUQUE
Poweshiek
Bremer
WOODBURY
Warren
Dallas
STORY
POTTAWATTA
Buena Vista
Marion
Marshall
Jasper
Webster
Lee
Dickinson
Clinton
JOHNSON
Wapello
Muscatine
BLACK HAWK
Des Moines
SCOTT
LINN
Cerro Gordo
Dubuque
Woodbury
Story
Pottawattamie
POLK
Johnson
Black Hawk
Scott
Linn
Polk
0
1
1e6

In [47]:
```
pop_counties.head(10)
```

Out[47]:
```
Polk             1833520
Linn              827574
Scott             621384
Black Hawk        535908
Johnson           490358
POLK              446035
Pottawattamie     325374
Story             320334
Woodbury          288241
Dubuque           286739
Name: County, dtype: int64
```

In [48]:
```
# most popular categories featured
pop_categories = iowa['Category Name'].value_counts()
pop_categories.head(15).plot(kind='barh', width=0.4, figsize=(2,15))
```

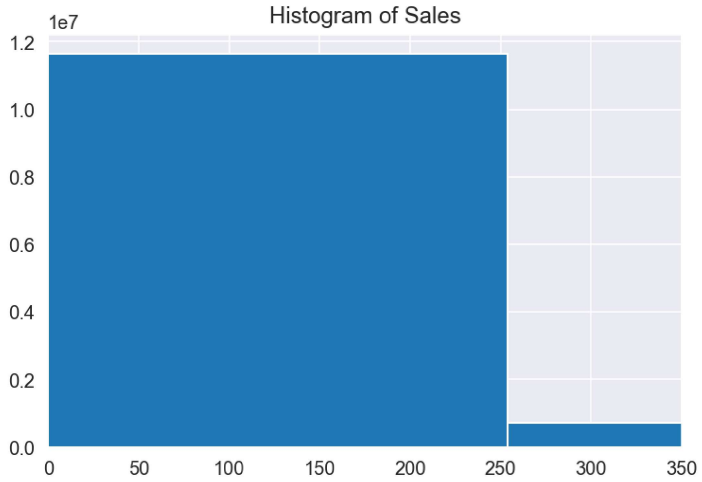
Out[48]:
```
<AxesSubplot:>
```


IMPORTED VODKA - MISC
AMERICAN DRY GINS
Canadian Whiskies
FLAVORED RUM
AMERICAN COCKTAILS
WHISKEY LIQUEUR
PUERTO RICO & VIRGIN ISLANDS RUM
IMPORTED VODKA
TEQUILA
BLENDED WHISKIES
VODKA FLAVORED
SPICED RUM
STRAIGHT BOURBON WHISKIES
CANADIAN WHISKIES

VODKA 80 PROOF

In [56]:
```
# histogram sales data
# only going to 350 bc thats where most of data is clustered, showing more of the graph will not showcase data clearly
plt.hist(iowa['Sale (Dollars)'], bins=1000)
plt.xlabel=('Sale ($)')
plt.ylabel=('Frequency')
plt.title('Histogram of Sales')
plt.xlim([0, 350])
```

Out[56]: (0.0, 350.0)



In [57]:
```
iowa['Store Number'].nunique()
```

Out[57]: 1884

In [58]:
```
# breaking down large table, forming new table of below columns based on city
city_table = iowa[['Store Number', 'City', 'Sale (Dollars)', 'State Bottle Retail ($)',
                           'State Profit Per Bottle ($)', 'Bottles Sold', 'Volume Sold (Liters)',
                          ]]
```

In [89]:
```
city_table.head()
```

Out[89]:

	Store Number	City	Sale (Dollars)	State Bottle Retail ($)	State Profit Per Bottle ($)	Bottles Sold	Volume Sold (Liters)
0	2191	KEOKUK	162.84	27.14	9.05	6	4.50
1	2205	CLARINDA	325.68	27.14	9.05	12	9.00
2	3549	FORT MADISON	19.20	9.60	3.20	2	0.30
3	2513	IOWA CITY	160.02	53.34	17.79	3	5.25
4	3942	TOLEDO	19.20	9.60	3.20	2	0.30

In [60]:
```
# Making similar dataframe but for counties instead
county_table = iowa[['Store Number', 'County', 'Sale (Dollars)', 'State Bottle Retail ($)',
                     'State Profit Per Bottle ($)', 'Bottles Sold', 'Volume Sold (Liters)',
                    ]]
```

In [61]:
```
county_table.head()
```

Out[61]:

	Store Number	County	Sale (Dollars)	State Bottle Retail ($)	State Profit Per Bottle ($)	Bottles Sold	Volume Sold (Liters)
0	2191	Lee	162.84	27.14	9.05	6	4.50
1	2205	Page	325.68	27.14	9.05	12	9.00
2	3549	Lee	19.20	9.60	3.20	2	0.30
3	2513	Johnson	160.02	53.34	17.79	3	5.25
4	3942	Tama	19.20	9.60	3.20	2	0.30

In [62]:
```
# Final dataframe by zip codes
zcode_table = iowa[['Store Number', 'Zip Code', 'Sale (Dollars)', 'State Bottle Retail ($)',
                     'State Profit Per Bottle ($)', 'Bottles Sold', 'Volume Sold (Liters)',
                    ]]
```

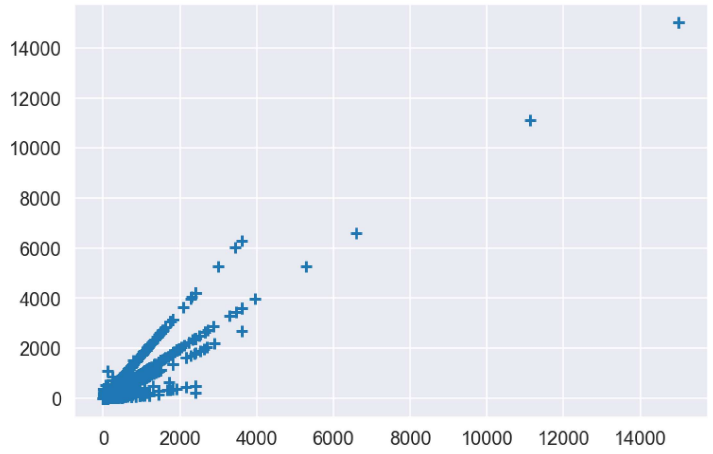
In [63]:
```
zcode_table.head()
```

Out[63]:

	Store Number	Zip Code	Sale (Dollars)	State Bottle Retail ($)	State Profit Per Bottle ($)	Bottles Sold	Volume Sold (Liters)
0	2191	52632	162.84	27.14	9.05	6	4.50
1	2205	51632	325.68	27.14	9.05	12	9.00
2	3549	52627	19.20	9.60	3.20	2	0.30
3	2513	52240	160.02	53.34	17.79	3	5.25
4	3942	52342	19.20	9.60	3.20	2	0.30

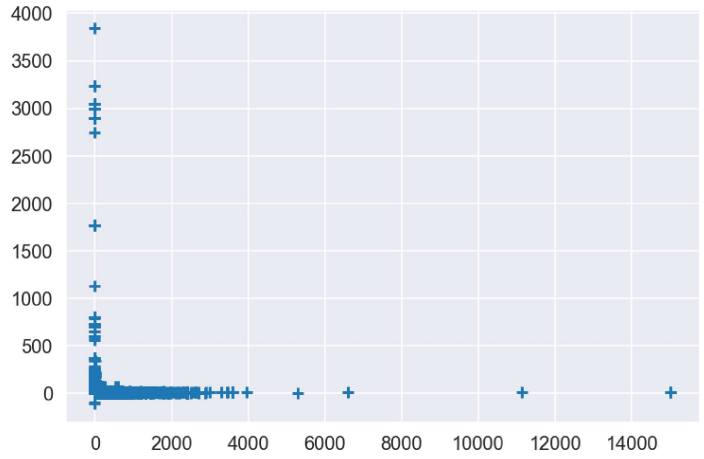
In [64]:
```
# plotting relationship to see if any outliers exist
plt.scatter(x=zcode_table['Bottles Sold'], y=zcode_table['Volume Sold (Liters)'], marker='+')
plt.xlabel('Bottles Sold')
plt.ylabel('Volume Sold (Liters)')
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
<ipython-input-64-5e94e86b5756> in <module>
      1 # plotting relationship to see if any outliers exist
      2 plt.scatter(x=zcode_table['Bottles Sold'], y=zcode_table['Volume Sold (Liters)'], marker='+')
----> 3 plt.xlabel('Bottles Sold')
      4 plt.ylabel('Volume Sold (Liters)')

TypeError: 'str' object is not callable
```



In [65]:
```
# plotting relationship to see if any outliers exist
plt.scatter(x=zcode_table['Bottles Sold'], y=zcode_table['State Profit Per Bottle ($)'], marker='+')
plt.xlabel('Bottles Sold')
plt.ylabel('State Profit Per Bottle')
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
<ipython-input-65-0f9c84a37088> in <module>
      1 # plotting relationship to see if any outliers exist
      2 plt.scatter(x=zcode_table['Bottles Sold'], y=zcode_table['State Profit Per Bottle ($)'], marker='+')
----> 3 plt.xlabel('Bottles Sold')
      4 plt.ylabel('State Profit Per Bottle')

TypeError: 'str' object is not callable
```



In [66]:
```
sales_by_zip = zcode_table.groupby('Zip Code')['Sale (Dollars)'].sum().to_frame().sort_values('Sale (Dollars)', ascendi

# resetting the index
sales_by_zip['Zip Code'] = sales_by_zip.index
```

In [67]:
```
# creating an index of range 0 - length of the
sales_by_zip.index = range(0, len(sales_by_zip))
```

In [72]:
```
print (sales_by_zip.shape)
sales_by_zip.head()
```

```
(871, 2)
```

Out[72]:

	Sale (Dollars)	Zip Code
0	4.578575e+07	50320
1	4.561466e+07	50314
2	4.128033e+07	52402
3	3.928887e+07	52240
4	3.225044e+07	50010

In [70]:
```
volume_by_zip = zcode_table.groupby('Zip Code')['Volume Sold (Liters)'].sum().to_frame()
volume_by_zip['Zip Code'] = volume_by_zip.index
volume_by_zip.index = range(0, len(volume_by_zip))
volume_by_zip.head()
```

Out[70]:

	Volume Sold (Liters)	Zip Code
0	992.12	50002
1	10843.85	50003
2	368.00	50005
3	2811.49	50006
4	57972.79	50009

In [73]:
```
# dataframe of bottle profit per zip code
bottleprofit_per_zip = zcode_table.groupby('Zip Code')['State Profit Per Bottle ($)'].sum().to_frame()
bottleprofit_per_zip['Zip Code'] = bottleprofit_per_zip.index
bottleprofit_per_zip.index = range(0, len(bottleprofit_per_zip))
bottleprofit_per_zip.head()
```

Out[73]:

	State Profit Per Bottle ($)	Zip Code
0	1679.09	50002
1	13958.18	50003
2	709.19	50005
3	5496.36	50006
4	96626.62	50009

In [78]:
```
# renaming column to appropriate description
stores_by_zip.rename(columns={'Store Number' : 'Number of Stores Per Zip'}, inplace=True)
stores_by_zip.head()
```

Out[78]:

	Number of Stores Per Zip	Zip Code
0	39	52402
1	33	51501
2	30	52404
3	27	52402
4	26	51501

In [79]:
```
bottles_by_zip = iowa.groupby('Zip Code')['Bottles Sold'].sum().to_frame()
bottles_by_zip['Zip Code'] = bottles_by_zip.index
bottles_by_zip.index = range(0, len(stores_by_zip))
bottles_by_zip.head()
```

Out[79]:

	Bottles Sold	Zip Code
0	1380	50002
1	11264	50003
2	495	50005
3	2711	50006
4	63721	50009

In [81]:
```
# merging stores and sales dataframe to matching zip codes
zip_frame = pd.merge(stores_by_zip, sales_by_zip, how='inner', on='Zip Code')
print (zip_frame.shape)
zip_frame.head()
```

```
(871, 3)
```

Out[81]:

	Number of Stores Per Zip	Zip Code	Sale (Dollars)
0	39	52402	4.128033e+07
1	33	51501	2.502018e+07
2	30	52404	1.707932e+07
3	27	52402	4.811544e+06
4	26	51501	3.545061e+06

In [82]:
```
# merging volume frame
zip_frame = pd.merge(zip_frame, volume_by_zip, how='inner', on='Zip Code')
print (zip_frame.shape)
zip_frame.head()
```

```
(871, 4)
```

Out[82]:

	Number of Stores Per Zip	Zip Code	Sale (Dollars)	Volume Sold (Liters)
0	39	52402	4.128033e+07	2878962.74
1	33	51501	2.502018e+07	1629347.08
2	30	52404	1.707932e+07	1246460.86
3	27	52402	4.811544e+06	172066.71
4	26	51501	3.545061e+06	125226.49

In [83]:
```
# mergine profit per bottle frame
zip_frame = pd.merge(zip_frame, bottleprofit_per_zip, how='inner', on='Zip Code')
print (zip_frame.shape)
zip_frame.head()
```

```
(871, 5)
```

Out[83]:

	Number of Stores Per Zip	Zip Code	Sale (Dollars)	Volume Sold (Liters)	State Profit Per Bottle ($)
0	39	52402	4.128033e+07	2878962.74	1.175481e+06
1	33	51501	2.502018e+07	1629347.08	8.052069e+05
2	30	52404	1.707932e+07	1246460.86	6.945509e+05
3	27	52402	4.811544e+06	172066.71	2.735682e+05
4	26	51501	3.545061e+06	125226.49	1.988386e+05

In [84]:
```
# merging bottles per zip as well
zip_frame = pd.merge(zip_frame, bottles_by_zip, how='inner', on='Zip Code')
print (zip_frame.shape)
zip_frame.head()
```

```
(871, 6)
```

Out[84]:

	Number of Stores Per Zip	Zip Code	Sale (Dollars)	Volume Sold (Liters)	State Profit Per Bottle ($)	Bottles Sold
0	39	52402	4.128033e+07	2878962.74	1.175481e+06	3123488
1	33	51501	2.502018e+07	1629347.08	8.052069e+05	1886250
2	30	52404	1.707932e+07	1246460.86	6.945509e+05	1456608
3	27	52402	4.811544e+06	172066.71	2.735682e+05	202685
4	26	51501	3.545061e+06	125226.49	1.988386e+05	165868

In [85]:
```
# Feature Engineering
zip_frame['Sales Per Store'] = zip_frame['Sale (Dollars)'] / zip_frame['Number of Stores Per Zip']
zip_frame.head()
```

Out[85]:

	Number of Stores Per Zip	Zip Code	Sale (Dollars)	Volume Sold (Liters)	State Profit Per Bottle ($)	Bottles Sold	Sales Per Store
0	39	52402	4.128033e+07	2878962.74	1.175481e+06	3123488	1.058470e+06
1	33	51501	2.502018e+07	1629347.08	8.052069e+05	1886250	7.581872e+05
2	30	52404	1.707932e+07	1246460.86	6.945509e+05	1456608	5.693106e+05
3	27	52402	4.811544e+06	172066.71	2.735682e+05	202685	1.782053e+05
4	26	51501	3.545061e+06	125226.49	1.988386e+05	165868	1.363485e+05

In [94]:
```
# define X and y
X = zip_frame[['State Profit Per Bottle ($)', 'Number of Stores Per Zip', 'Volume Sold (Liters)', 'Bottles Sold', 'Sales
y = zip_frame['Sale (Dollarsfrom sklearn.ensemble import RandomForestRegressor
regressor_forest = RandomForestRegressor(n_estimators = 10, random_state = 42)
regressor_forest.fit(X_train, y_train)
regressor_forest.score(X_test,y_test))']
```

In [95]:
```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X,y,random_state = 42, test_size = 0.3)
```

In [96]:
```
from sklearn.linear_model import LinearRegression
regressor = LinearRegression()
regressor.fit(X_train, y_train)
regressor.score(X_test, y_test)
```

Out[96]: 0.9889234427671271

In [97]:
```
from sklearn.ensemble import RandomForestRegressor
regressor_forest = RandomForestRegressor(n_estimators = 10, random_state = 42)
regressor_forest.fit(X_train, y_train)
regressor_forest.score(X_test,y_test)
```

Out[97]: 0.9660903946423354

In [98]:
```
from sklearn.tree import DecisionTreeRegressor

regressor = DecisionTreeRegressor(random_state = 42)
regressor.fit(X_train, y_train)
regressor.score(X_test,y_test)
```

Out[98]: 0.9776151699984184

In [99]:
```
from sklearn.ensemble import GradientBoostingRegressor
boost = GradientBoostingRegressor()
boost.fit(X_train,y_train)
boost.score(X_test,y_test)
```

Out[99]: 0.9833350349173261

In [100]:
```
from sklearn.ensemble import AdaBoostRegressor

ada = AdaBoostRegressor()
ada.fit(X_train,y_train)
ada.score(X_test,y_test)
```

Out[100]: 0.9769751848378644

In [106]:
```
a = iowa[iowa['Zip Code'] == 50314]
```

In [110]:
```
a['Category Name'].value_counts()
```

Out[110]:
```
American Flavored Vodka    2422
American Vodkas            2139
Imported Flavored Vodka    2072
Canadian Whiskies          1886
100% Agave Tequila         1867
                           ...
TROPICAL FRUIT SCHNAPPS       2
CORN WHISKIES                 1
OTHER PROOF VODKA             1
AMARETTO - IMPORTED           1
AMERICAN SLOE GINS            1
Name: Category Name, Length: 120, dtype: int64
```

In [111]:
```
b = iowa[iowa['Zip Code'] == 52807]
```

In [112]:
```
b['Category Name'].value_counts()
```

Out[112]:
```
American Vodkas              2472
Straight Bourbon Whiskies    1712
American Flavored Vodka      1697
Canadian Whiskies            1676
Spiced Rum                   1086
                             ...
GREEN CREME DE MENTHE           2
OTHER PROOF VODKA               2
Corn Whiskies                   2
AMERICAN SLOE GINS              2
CREME DE ALMOND                 1
Name: Category Name, Length: 115, dtype: int64
```

In [113]:
```
c = iowa[iowa['Zip Code'] == 52402]
```

In [114]:
```
c['Category Name'].value_counts()
```

Out[114]:
```
American Vodkas              5923
Canadian Whiskies            3837
American Flavored Vodka      2857
Straight Bourbon Whiskies    2788
Spiced Rum                   2706
                             ...
CORN WHISKIES                   2
BARBADOS RUM                    2
WHITE CREME DE CACAO            1
GREEN CREME DE MENTHE           1
AMERICAN SLOE GINS              1
Name: Category Name, Length: 117, dtype: int64
```

In [ ]:
```

```