20210930_Supplier_Management_MB7

September 30, 2021

[4]:

```
# Visualization of Data

# shift + enter
```

[2]:

```
# Install and import certain Packages
```

[3]:

```
!pip install seaborn
```

```
Requirement already satisfied: seaborn in
/Users/h4/opt/anaconda3/lib/python3.8/site-packages (0.11.1)
Requirement already satisfied: scipy>=1.0 in
/Users/h4/opt/anaconda3/lib/python3.8/site-packages (from seaborn) (1.6.2)
Requirement already satisfied: matplotlib>=2.2 in
/Users/h4/opt/anaconda3/lib/python3.8/site-packages (from seaborn) (3.3.4)
Requirement already satisfied: pandas>=0.23 in
/Users/h4/opt/anaconda3/lib/python3.8/site-packages (from seaborn) (1.2.4)
Requirement already satisfied: numpy>=1.15 in
/Users/h4/opt/anaconda3/lib/python3.8/site-packages (from seaborn) (1.19.5)
Requirement already satisfied: kiwisolver>=1.0.1 in
/Users/h4/opt/anaconda3/lib/python3.8/site-packages (from
matplotlib>=2.2->seaborn) (1.3.1)
Requirement already satisfied: pyparsing!=2.0.4,!=2.1.2,!=2.1.6,>=2.0.3 in
/Users/h4/opt/anaconda3/lib/python3.8/site-packages (from
matplotlib>=2.2->seaborn) (2.4.7)
Requirement already satisfied: pillow>=6.2.0 in
/Users/h4/opt/anaconda3/lib/python3.8/site-packages (from
matplotlib>=2.2->seaborn) (8.3.2)
Requirement already satisfied: python-dateutil>=2.1 in
/Users/h4/opt/anaconda3/lib/python3.8/site-packages (from
matplotlib>=2.2->seaborn) (2.8.1)
Requirement already satisfied: cycler>=0.10 in
/Users/h4/opt/anaconda3/lib/python3.8/site-packages (from
matplotlib>=2.2->seaborn) (0.10.0)
Requirement already satisfied: six in
/Users/h4/opt/anaconda3/lib/python3.8/site-packages (from
cycler>=0.10->matplotlib>=2.2->seaborn) (1.15.0)
Requirement already satisfied: pytz>=2017.3 in
/Users/h4/opt/anaconda3/lib/python3.8/site-packages (from pandas>=0.23->seaborn)
```

```
(2021.1)
```

[5]:

```python
import seaborn as sns # by calling seaborn as "sns" I am creating a variable "sns",
                        # everytime I write "sns" Python understands "seaborn"
```

[6]:

```python
# load a dataset "anscombe"

anscombe = sns.load_dataset('anscombe') # shift + enter
```

[7]:

```python
print(anscombe)
```

```
   dataset     x      y
0        I  10.0   8.04
1        I   8.0   6.95
2        I  13.0   7.58
3        I   9.0   8.81
4        I  11.0   8.33
5        I  14.0   9.96
6        I   6.0   7.24
7        I   4.0   4.26
8        I  12.0  10.84
9        I   7.0   4.82
10       I   5.0   5.68
11      II  10.0   9.14
12      II   8.0   8.14
13      II  13.0   8.74
14      II   9.0   8.77
15      II  11.0   9.26
16      II  14.0   8.10
17      II   6.0   6.13
18      II   4.0   3.10
19      II  12.0   9.13
20      II   7.0   7.26
21      II   5.0   4.74
22     III  10.0   7.46
23     III   8.0   6.77
24     III  13.0  12.74
25     III   9.0   7.11
26     III  11.0   7.81
27     III  14.0   8.84
28     III   6.0   6.08
29     III   4.0   5.39
30     III  12.0   8.15
31     III   7.0   6.42
32     III   5.0   5.73
33      IV   8.0   6.58
34      IV   8.0   5.76
```

```
35      IV   8.0   7.71
36      IV   8.0   8.84
37      IV   8.0   8.47
38      IV   8.0   7.04
39      IV   8.0   5.25
40      IV  19.0  12.50
41      IV   8.0   5.56
42      IV   8.0   7.91
43      IV   8.0   6.89
```

[8]:

```
# create a subset of the dataset
# we choose the data only from class I

dataset_1 = anscombe[anscombe['dataset'] == 'I']
```

[9]:

```
print(dataset_1)
```

```
   dataset     x      y
0        I  10.0   8.04
1        I   8.0   6.95
2        I  13.0   7.58
3        I   9.0   8.81
4        I  11.0   8.33
5        I  14.0   9.96
6        I   6.0   7.24
7        I   4.0   4.26
8        I  12.0  10.84
9        I   7.0   4.82
10       I   5.0   5.68
```

[10]:

```
# for visualization we use also "matplotlib"

!pip install matplotlib

import matplotlib.pyplot as plt
```

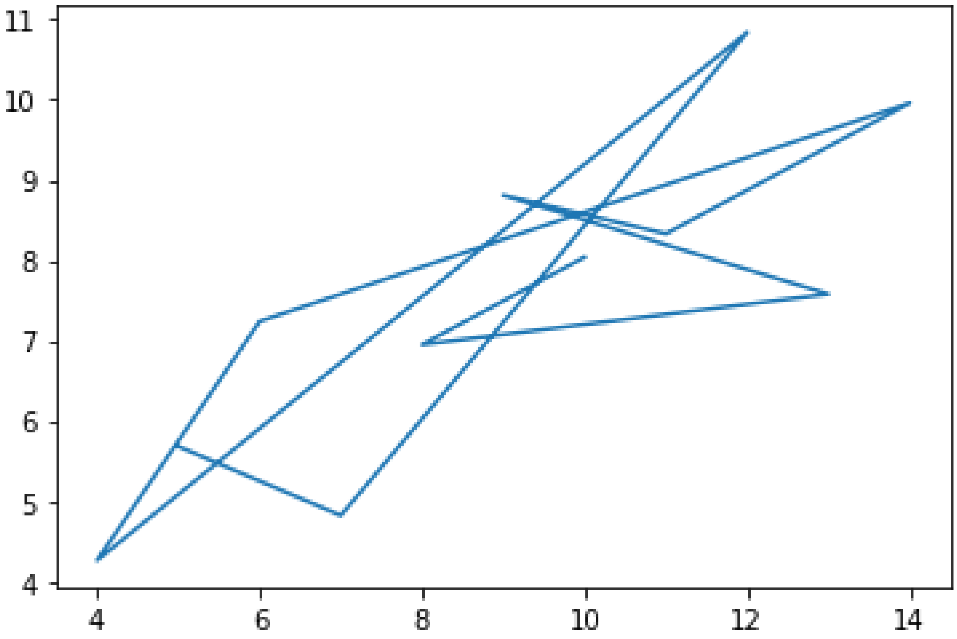
```
Requirement already satisfied: matplotlib in
/Users/h4/opt/anaconda3/lib/python3.8/site-packages (3.3.4)
Requirement already satisfied: cycler>=0.10 in
/Users/h4/opt/anaconda3/lib/python3.8/site-packages (from matplotlib) (0.10.0)
Requirement already satisfied: python-dateutil>=2.1 in
/Users/h4/opt/anaconda3/lib/python3.8/site-packages (from matplotlib) (2.8.1)
Requirement already satisfied: numpy>=1.15 in
/Users/h4/opt/anaconda3/lib/python3.8/site-packages (from matplotlib) (1.19.5)
Requirement already satisfied: pyparsing!=2.0.4,!=2.1.2,!=2.1.6,>=2.0.3 in
/Users/h4/opt/anaconda3/lib/python3.8/site-packages (from matplotlib) (2.4.7)
Requirement already satisfied: pillow>=6.2.0 in
/Users/h4/opt/anaconda3/lib/python3.8/site-packages (from matplotlib) (8.3.2)
```

```
Requirement already satisfied: kiwisolver>=1.0.1 in
/Users/h4/opt/anaconda3/lib/python3.8/site-packages (from matplotlib) (1.3.1)
Requirement already satisfied: six in
/Users/h4/opt/anaconda3/lib/python3.8/site-packages (from
cycler>=0.10->matplotlib) (1.15.0)
```

[12]:
```
# by default plt plots a 2D line

plt.plot(dataset_1['x'], dataset_1['y'])
```

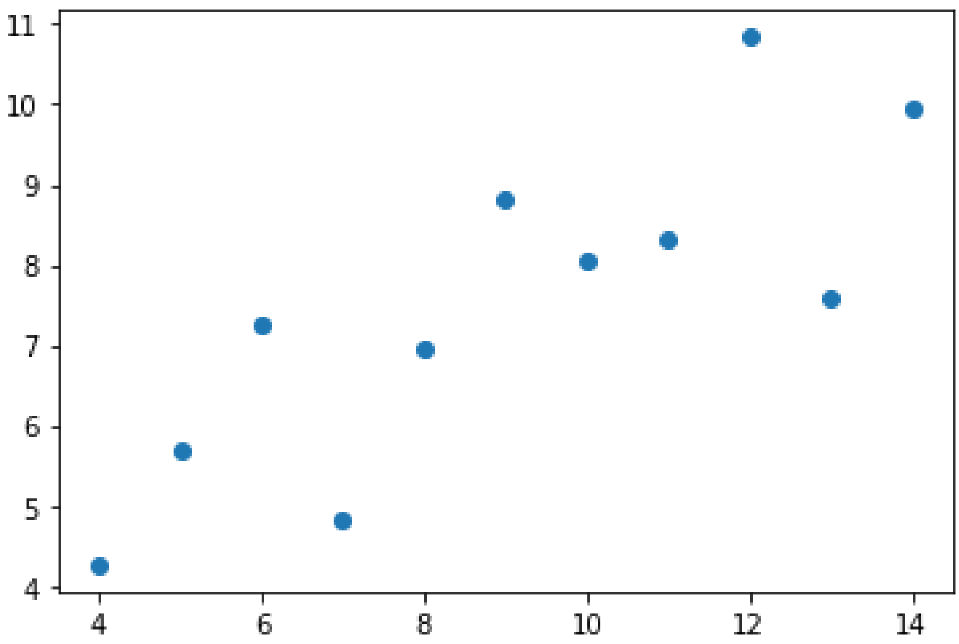
[12]:
```
[<matplotlib.lines.Line2D at 0x7fcdcfd4a280>]
```



[15]:
```
# change the plot to a scatterplot (points)

plt.plot(dataset_1['x'], dataset_1['y'],'o')
```

[15]:
```
[<matplotlib.lines.Line2D at 0x7fcdd00a5c70>]
```



[16]:
```
# Excercise. Please represent following dataset graphically:

dataset_2 = anscombe[anscombe['dataset'] == 'II']
dataset_3 = anscombe[anscombe['dataset'] == 'III']
dataset_4 = anscombe[anscombe['dataset'] == 'IV']
```

[17]:
```
# representation of statistical datsets
```

[18]:
```
tips = sns.load_dataset('tips')
```

[19]:
```
print(tips.head()) # if I call "head()" then I get only the first 5 positions
```

```
   total_bill   tip     sex smoker  day    time  size
0       16.99  1.01  Female     No  Sun  Dinner     2
1       10.34  1.66    Male     No  Sun  Dinner     3
2       21.01  3.50    Male     No  Sun  Dinner     3
3       23.68  3.31    Male     No  Sun  Dinner     2
4       24.59  3.61  Female     No  Sun  Dinner     4
```

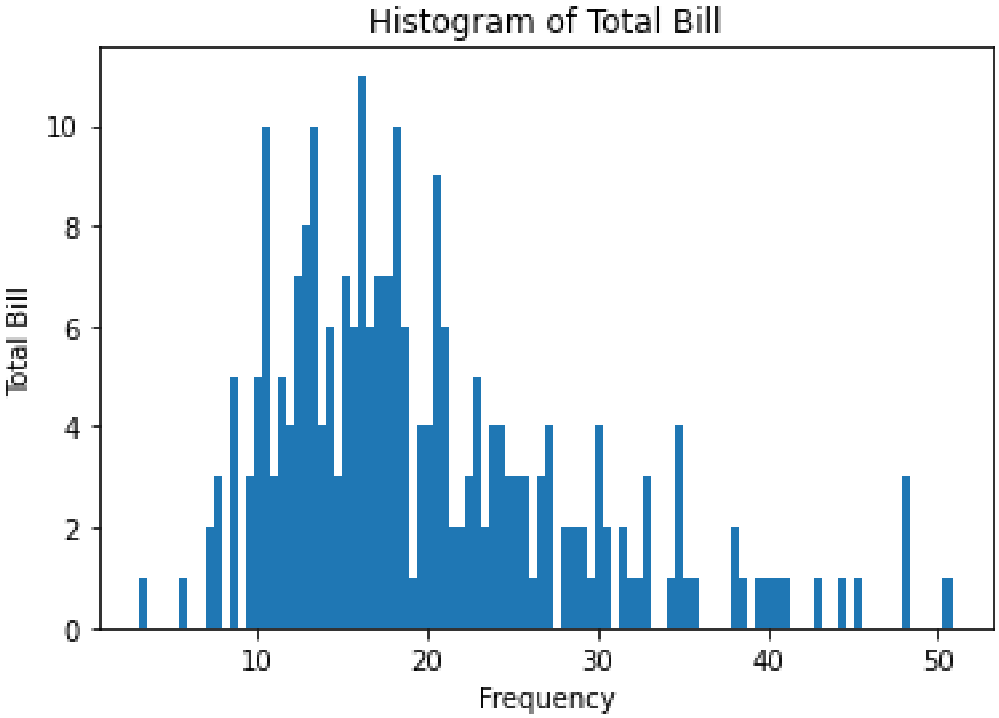
[20]:
```
# Histogram (1 Variable representation)
```

[24]:
```
# Example: plot the frequency of Total_Bills

fig = plt.figure()  # reserve some memory for the plot by creating a variable
axes1 = fig.add_subplot(1,1,1)  # created a mosaic of 1 picture
```

```
axes1.hist(tips['total_bill'], bins=100) # we tell Python to plot a Histogram␣
↪of the colum
                                            # "total_bill" with 100 groups
axes1.set_title('Histogram of Total Bill') # Title of the histogram
axes1.set_xlabel('Frequency')    # label for x-Axis
axes1.set_ylabel('Total Bill')   # label for y-Axis
fig.show()                       # show the Figure # shift+Enter
```

```
<ipython-input-24-d7d273859594>:10: UserWarning: Matplotlib is currently using
module://ipykernel.pylab.backend_inline, which is a non-GUI backend, so cannot
show the figure.
  fig.show()                       # show the Figure # shift+Enter
```



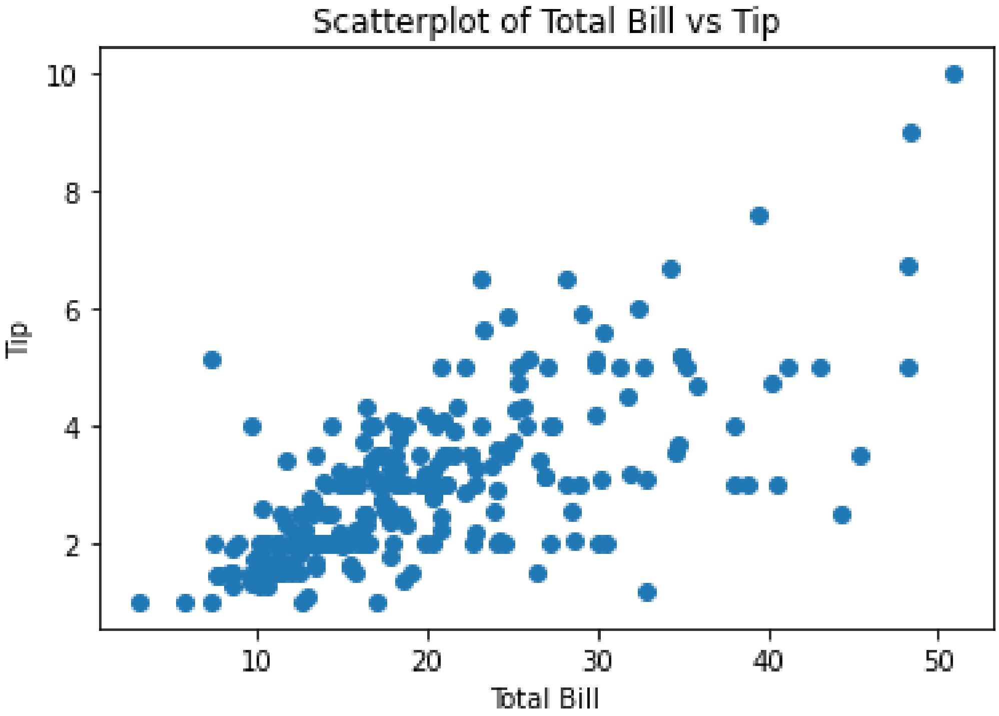
[25]:
```
# Scatterplot (2 Variable representation)
```

[26]:
```
# Example: Total Bill (x-axis) and Tip (y-axis)

scatter_plot = plt.figure()
axes1 = scatter_plot.add_subplot(1,1,1)
axes1.scatter(tips['total_bill'], tips['tip'])
axes1.set_title('Scatterplot of Total Bill vs Tip')
axes1.set_xlabel('Total Bill')
axes1.set_ylabel('Tip')
```

```
scatter_plot.show()
```

```
<ipython-input-26-5a3fa8c26df1>:9: UserWarning: Matplotlib is currently using
module://ipykernel.pylab.backend_inline, which is a non-GUI backend, so cannot
show the figure.
  scatter_plot.show()
```



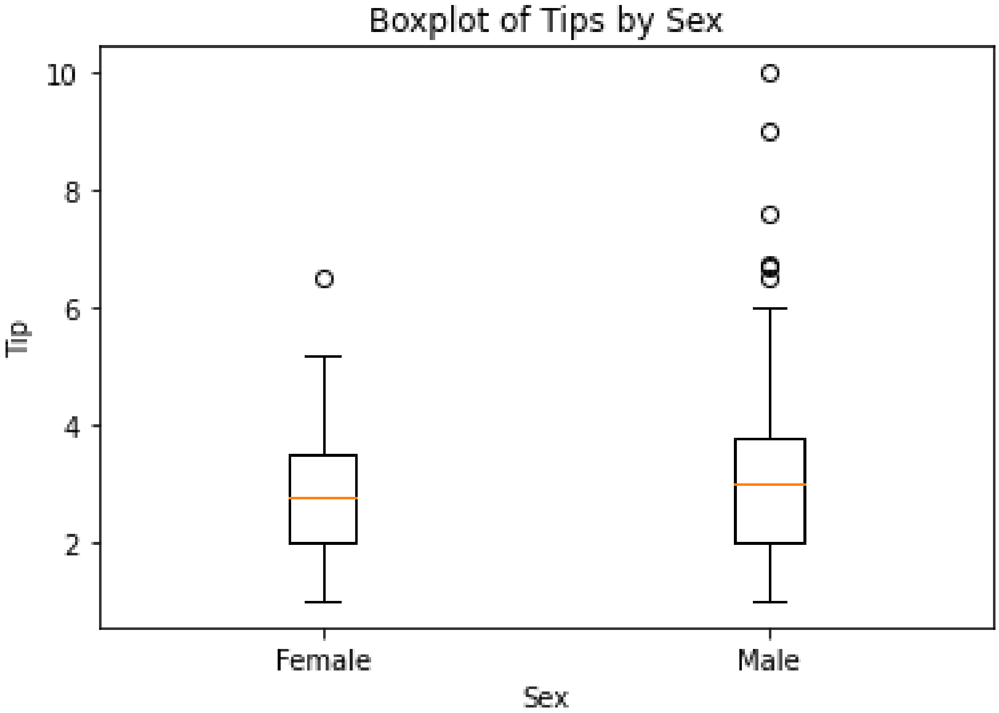
[27]:
```
# Boxplots (Plot one categorical variable and a numerical variable)
```

[30]:
```
# Example: plot boxplot of "sex" vs. "tip"

boxplot = plt.figure()
axes1 = boxplot.add_subplot(1,1,1)
axes1.boxplot([tips[tips['sex']=='Female']['tip'],
              tips[tips['sex']=='Male']['tip']],
             labels=['Female', 'Male'])
axes1.set_xlabel('Sex')
axes1.set_ylabel('Tip')
axes1.set_title('Boxplot of Tips by Sex')
boxplot.show()
```

```
<ipython-input-30-2f1e27df3786>:11: UserWarning: Matplotlib is currently using
module://ipykernel.pylab.backend_inline, which is a non-GUI backend, so cannot
show the figure.
```

```
boxplot.show()
```



[31]:
```
# Excercise. Represent several Boxplots Tip vs. "smoker", "time", "day"
```

[32]:
```
# Multivariate Datasets (many variables in one plot)
```

[33]:
```
# Example: Scatterplot the total bill vs. tip, colored by Sex and Sized by␣
 ↪Amount
```

[34]:
```
# First we need to re-code "sex" to a numerical variable (!)

def recode_sex(sex):
    if sex=='Female':
        return 0
    else:
        return 1

# if the variable "sex" is Female, then transform it to a "0", otherwise␣
 ↪transform it to "1"
```

[35]:
```
# we add a column to our dataset with the sex re-coded

tips['sex_color'] = tips['sex'].apply(recode_sex) # here we apply the function␣
 ↪"recode_sex"
```

```
                                                    # to the column "sex" of␣
↪tips
```

[36]:

```
print(tips.head())
```

```
   total_bill   tip     sex smoker  day    time  size  sex_color
0       16.99  1.01  Female     No  Sun  Dinner     2          0
1       10.34  1.66    Male     No  Sun  Dinner     3          1
2       21.01  3.50    Male     No  Sun  Dinner     3          1
3       23.68  3.31    Male     No  Sun  Dinner     2          1
4       24.59  3.61  Female     No  Sun  Dinner     4          0
```

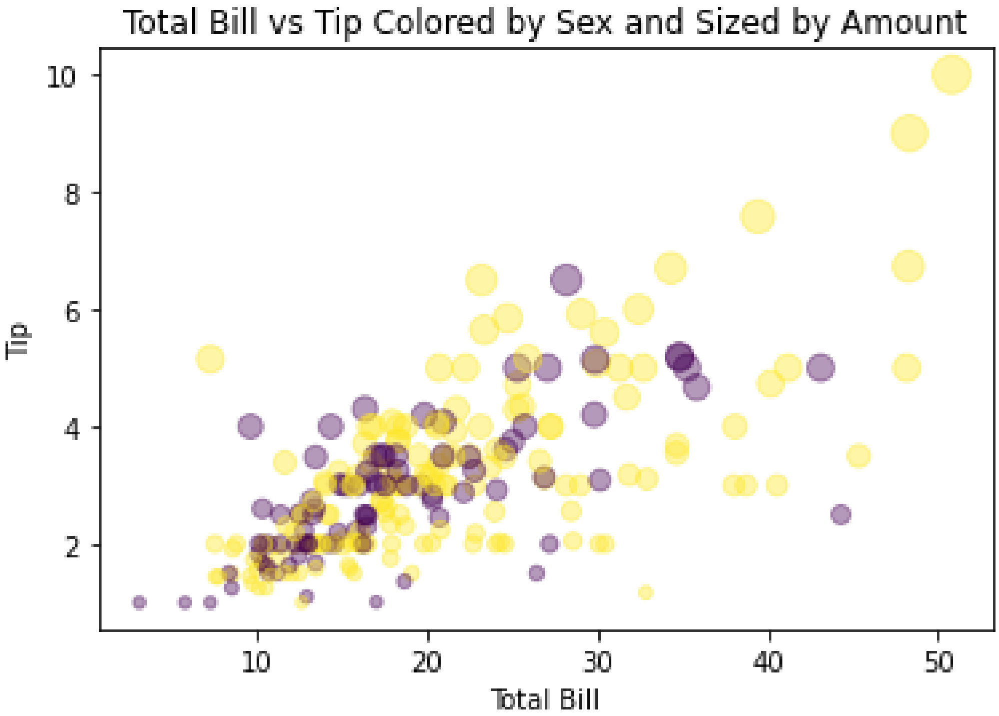
[37]:

```
# Excercise: recode the variables "smoker", "day", "time" so that you get␣
↪numerical variables
```

[41]:

```
scatter_plot = plt.figure()
axes1 = scatter_plot.add_subplot(1,1,1)
axes1.scatter(x = tips['total_bill'],
             y = tips['tip'],
             s = tips['tip']*20, # size
             c = tips['sex_color'], # color
             alpha = 0.4) # transparency
axes1.set_title('Total Bill vs Tip Colored by Sex and Sized by Amount')
axes1.set_xlabel('Total Bill')
axes1.set_ylabel('Tip')
scatter_plot.show()
```

```
<ipython-input-41-b567227f7fea>:11: UserWarning: Matplotlib is currently using
module://ipykernel.pylab.backend_inline, which is a non-GUI backend, so cannot
show the figure.
  scatter_plot.show()
```



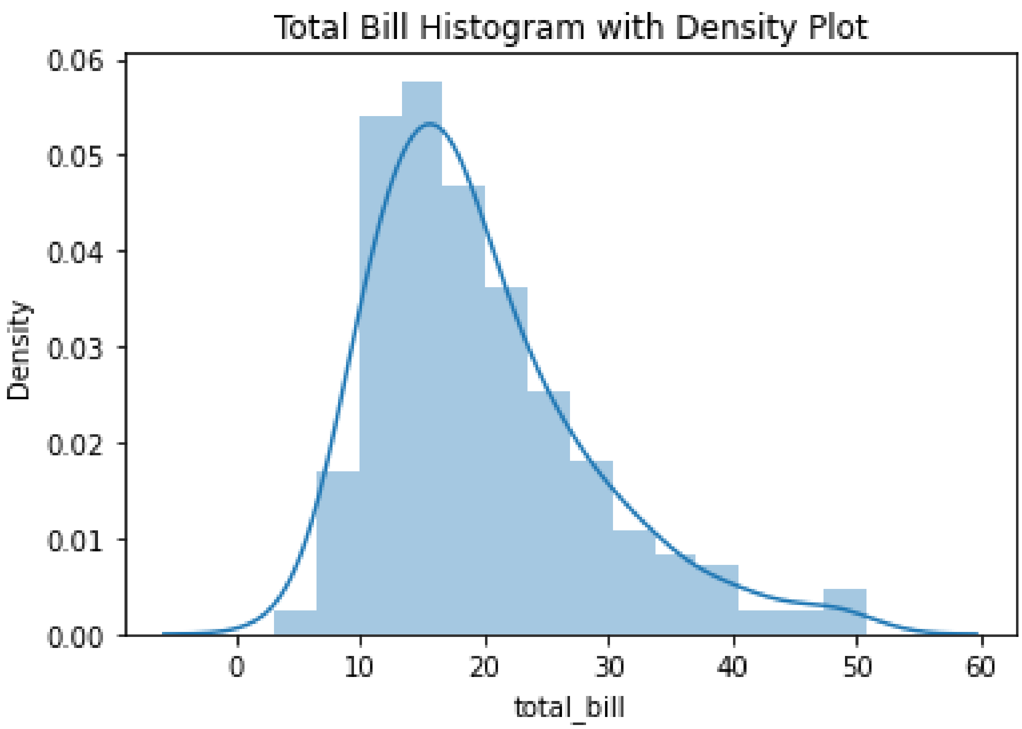
[42]:
```
# Excercise: represent Total Bill vs. Tip by daytime
# Excercise: represent Total Bill vs. Tip by smoker
```

[44]:
```
# Represent two plots in one figure
```

[45]:
```
# Example: epresentation of Histogram with a Density Plot (Density function)
```

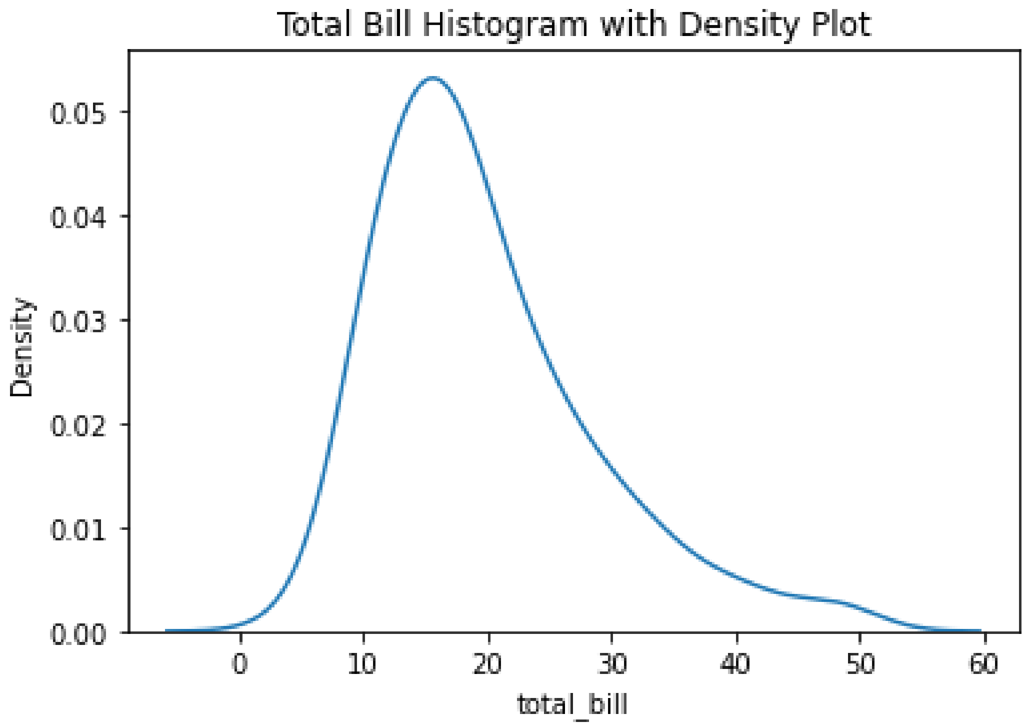
[47]:
```
hist, ax = plt.subplots()
ax = sns.distplot(tips['total_bill']) # density plot + histogram
ax.set_title('Total Bill Histogram with Density Plot')
plt.show()
```

```
/Users/h4/opt/anaconda3/lib/python3.8/site-
packages/seaborn/distributions.py:2557: FutureWarning: `distplot` is a
deprecated function and will be removed in a future version. Please adapt your
code to use either `displot` (a figure-level function with similar flexibility)
or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```



[48]:
```
hist, ax = plt.subplots()
ax = sns.distplot(tips['total_bill'], hist=False) # density plot without␣
 ↪histogram
ax.set_title('Total Bill Histogram with Density Plot')
plt.show()
```

```
/Users/h4/opt/anaconda3/lib/python3.8/site-
packages/seaborn/distributions.py:2557: FutureWarning: `distplot` is a
deprecated function and will be removed in a future version. Please adapt your
code to use either `displot` (a figure-level function with similar flexibility)
or `kdeplot` (an axes-level function for kernel density plots).
  warnings.warn(msg, FutureWarning)
```



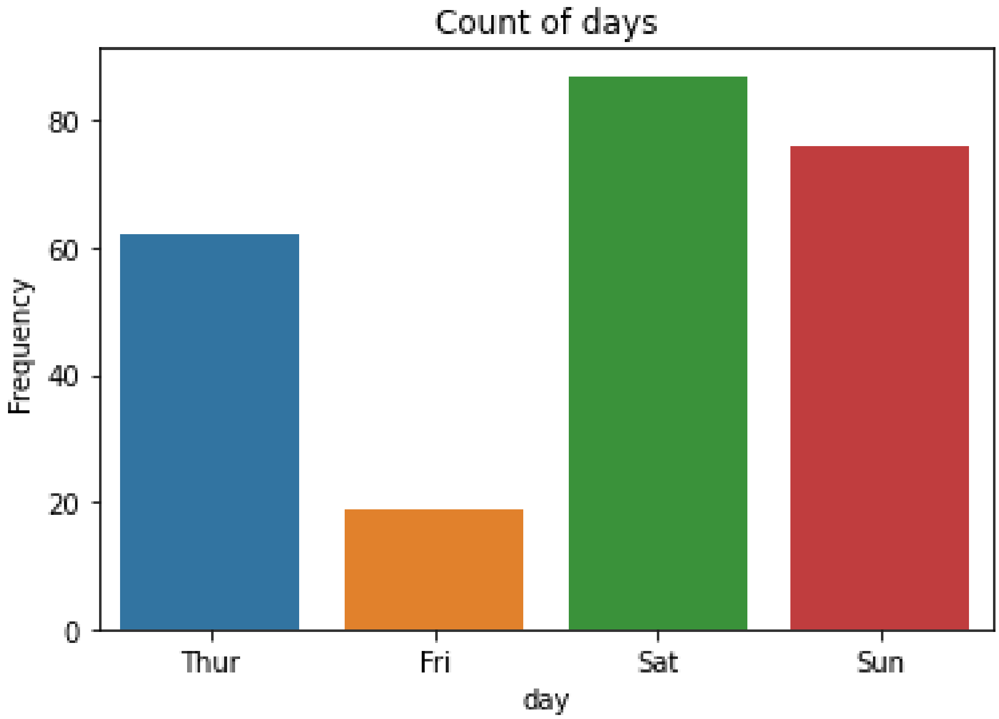
[49]:

```
# Bar Plot
```

[50]:

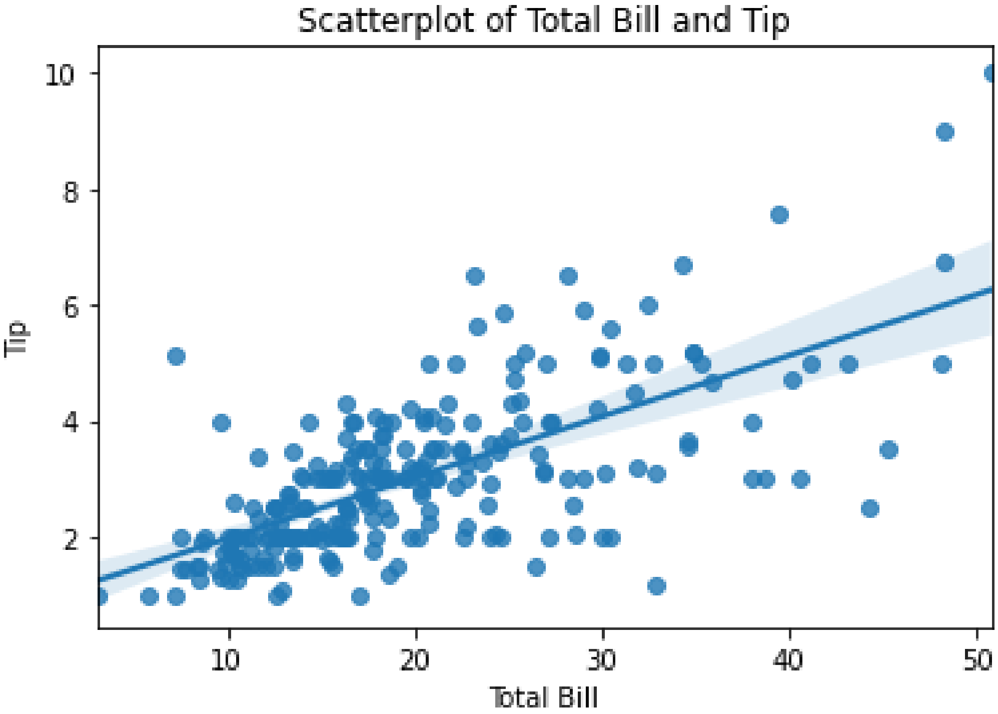
```
coun, ax = plt.subplots()
ax = sns.countplot('day', data=tips)
ax.set_title('Count of days')
ax.set_ylabel('Days of the week')
ax.set_ylabel('Frequency')
plt.show()
```

```
/Users/h4/opt/anaconda3/lib/python3.8/site-packages/seaborn/_decorators.py:36:
FutureWarning: Pass the following variable as a keyword arg: x. From version
0.12, the only valid positional argument will be `data`, and passing other
arguments without an explicit keyword will result in an error or
misinterpretation.
  warnings.warn(
```



[51]:
```
# Linear regression with scatterplot
```

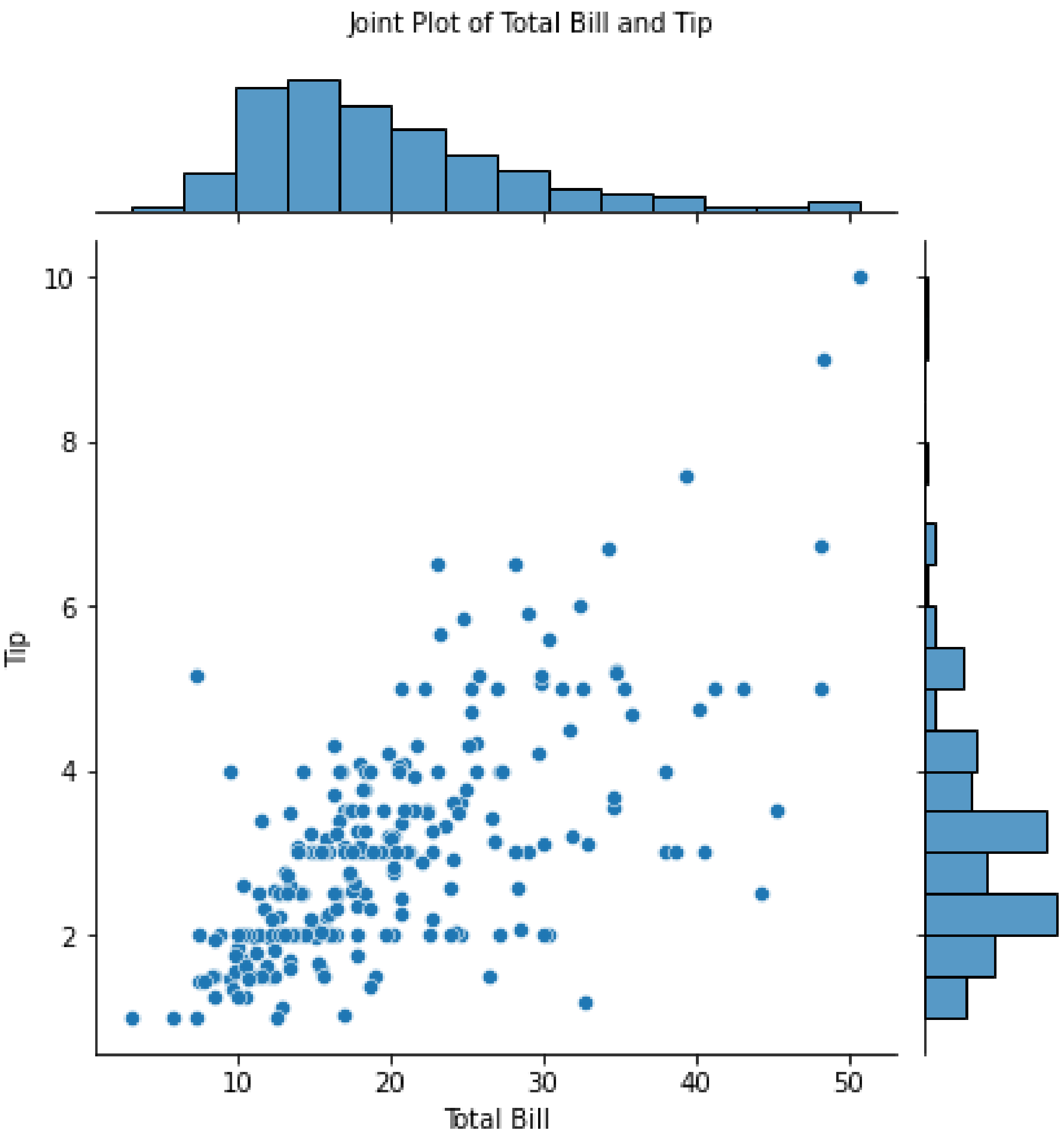
[52]:
```
scatter, ax = plt.subplots()
ax = sns.regplot(x='total_bill', y='tip', data=tips)
ax.set_title('Scatterplot of Total Bill and Tip')
ax.set_xlabel('Total Bill')
ax.set_ylabel('Tip')
plt.show()
```



```
[53]: # Jointplot - scatter and histogram in two dimensions
```

```
[54]: joint = sns.jointplot(x='total_bill', y='tip', data=tips)
      joint.set_axis_labels(xlabel='Total Bill', ylabel='Tip')
      joint.fig.suptitle('Joint Plot of Total Bill and Tip', fontsize=10, y=1.03)
```

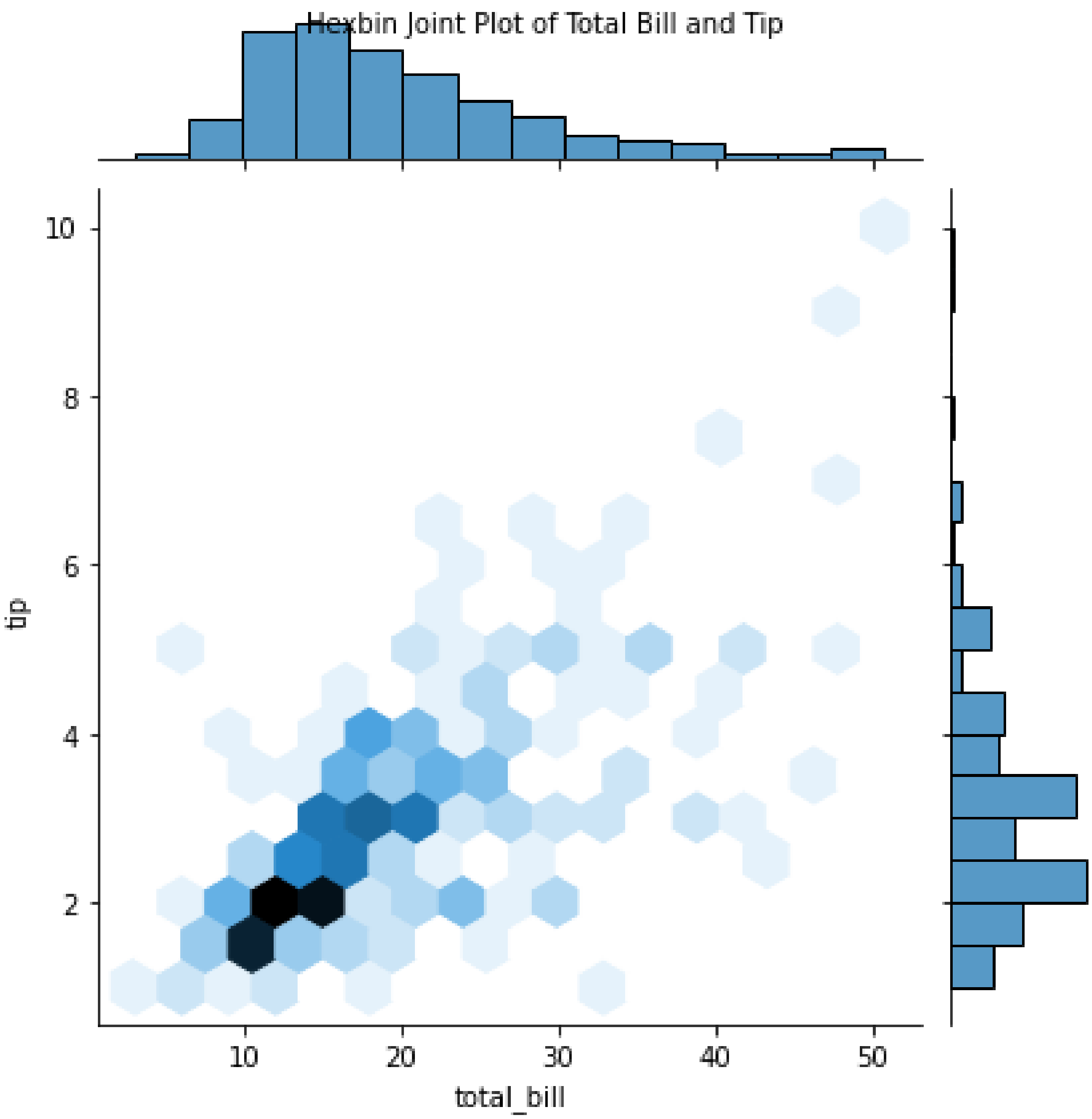
```
[54]: Text(0.5, 1.03, 'Joint Plot of Total Bill and Tip')
```



```
[55]: # Hexbin: density of scatterplots with hexagons

      hexbin = sns.jointplot(x='total_bill', y='tip', data=tips, kind='hex')
      hexbin.fig.suptitle('Hexbin Joint Plot of Total Bill and Tip', fontsize=10)
```

```
[55]: Text(0.5, 0.98, 'Hexbin Joint Plot of Total Bill and Tip')
```



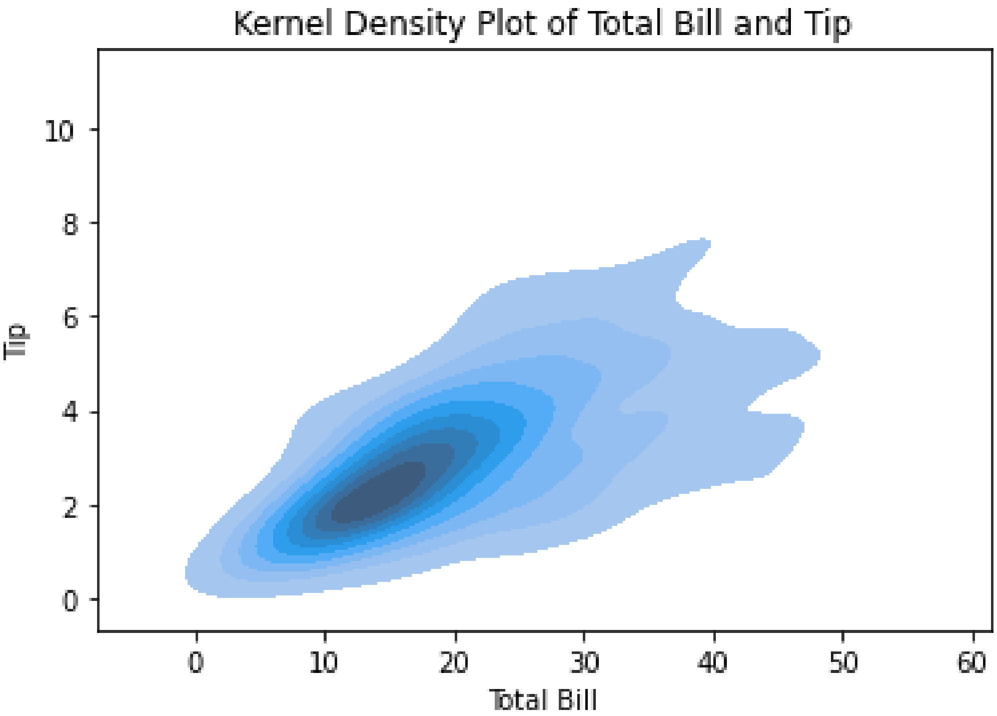
```
[56]: # 2D density plot

      kde, ax = plt.subplots() # KDE = kernel density plot
      ax = sns.kdeplot(data=tips['total_bill'],
                       data2=tips['tip'],
                       shade=True)
      ax.set_title('Kernel Density Plot of Total Bill and Tip')
      ax.set_xlabel('Total Bill')
      ax.set_ylabel('Tip')
```

```
/Users/h4/opt/anaconda3/lib/python3.8/site-
packages/seaborn/distributions.py:1639: FutureWarning: Use `x` and `y` rather
than `data` `and `data2`
```

```
    warnings.warn(msg, FutureWarning)
```

```
[56]: Text(0, 0.5, 'Tip')
```



```
[57]: # www.profh4.com --> Lectures --> Playlist "Supplier Management WS2021 2022
       ↪MBW7"
```

```
[ ]:
```