

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

## Distribution plot

we use distribution plot to visualize the distribution of quantatitive data types are:-

- disjoint /histplot
- joint plot
- pair plot
- rug plot

```
tips=sns.load_dataset('tips')
tips
```

	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
...	...	...	...	...	...	...	...
239	29.03	5.92	Male	No	Sat	Dinner	3
240	27.18	2.00	Female	Yes	Sat	Dinner	2
241	22.67	2.00	Male	Yes	Sat	Dinner	2
242	17.82	1.75	Male	No	Sat	Dinner	2
243	18.78	3.00	Female	No	Thur	Dinner	2

```
[244 rows x 7 columns]
```

```
tips['size'].unique()
```

```
array([2, 3, 4, 1, 6, 5])
```

it is consider as category

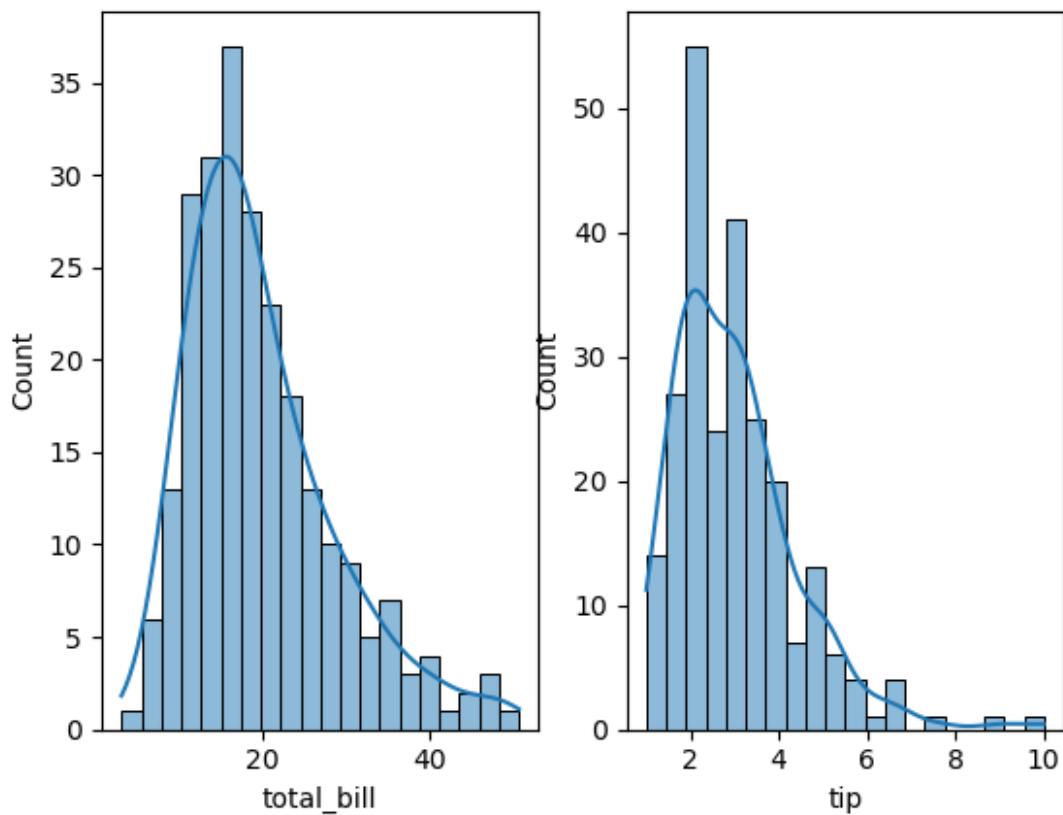
we loaded date set from sns directly,otherwise you can also load yours using pandas.

```
dam=pd.read_csv('dam.csv')
dam
```

	x	y	z
0	1	2	3
1	4	5	6
2	7	8	9

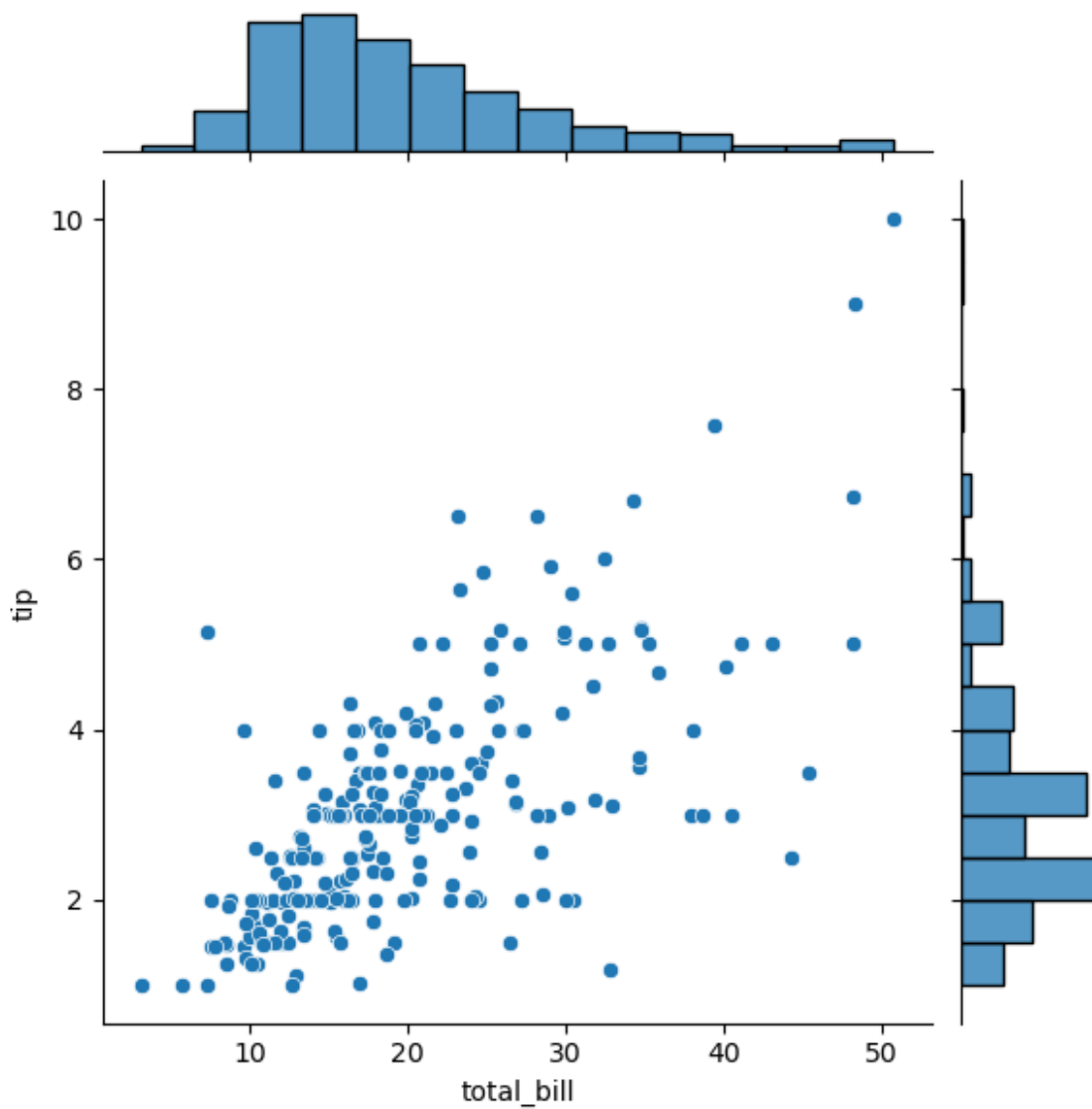
## distplot

```
plt.subplot(1,2,1)
sns.histplot(tips['total_bill'],bins=20,kde=True)
plt.subplot(1,2,2)
sns.histplot(tips['tip'],bins=20,kde=True)
<Axes: xlabel='tip', ylabel='Count'>
```

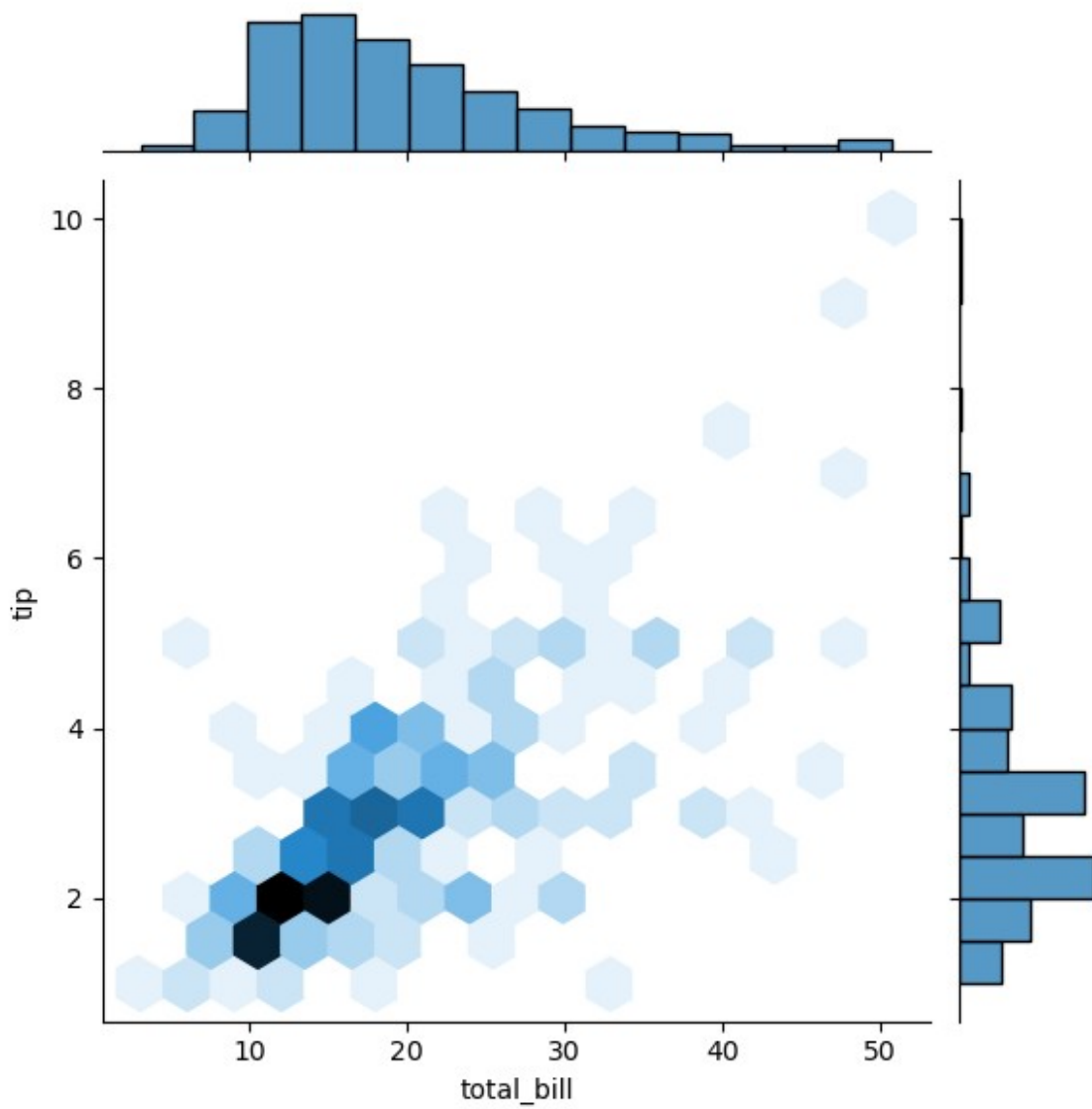


## Joint Plot

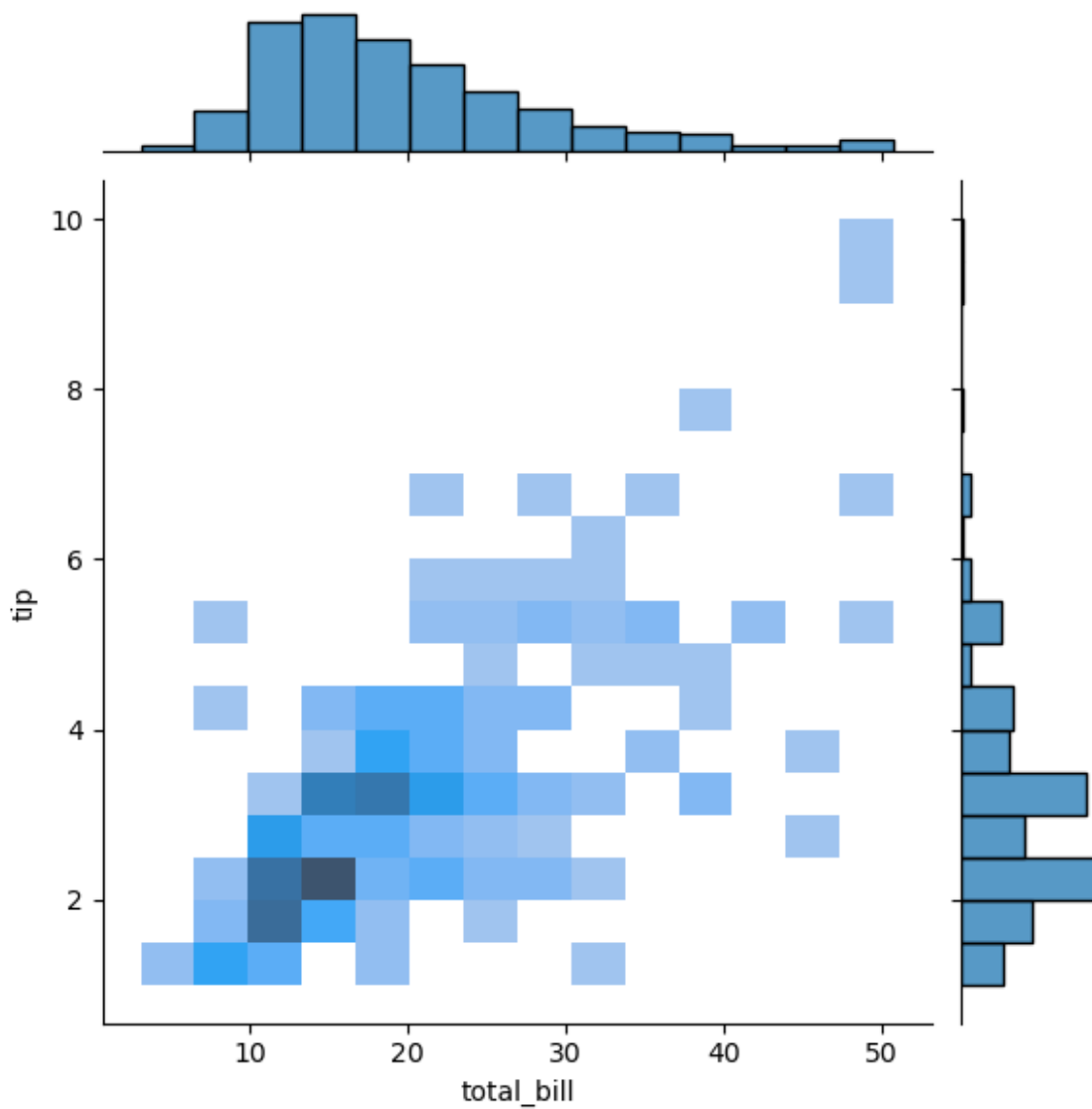
```
sns.jointplot(x='total_bill',y='tip',data=tips,kind='scatter')
<seaborn.axisgrid.JointGrid at 0x1335a574590>
```



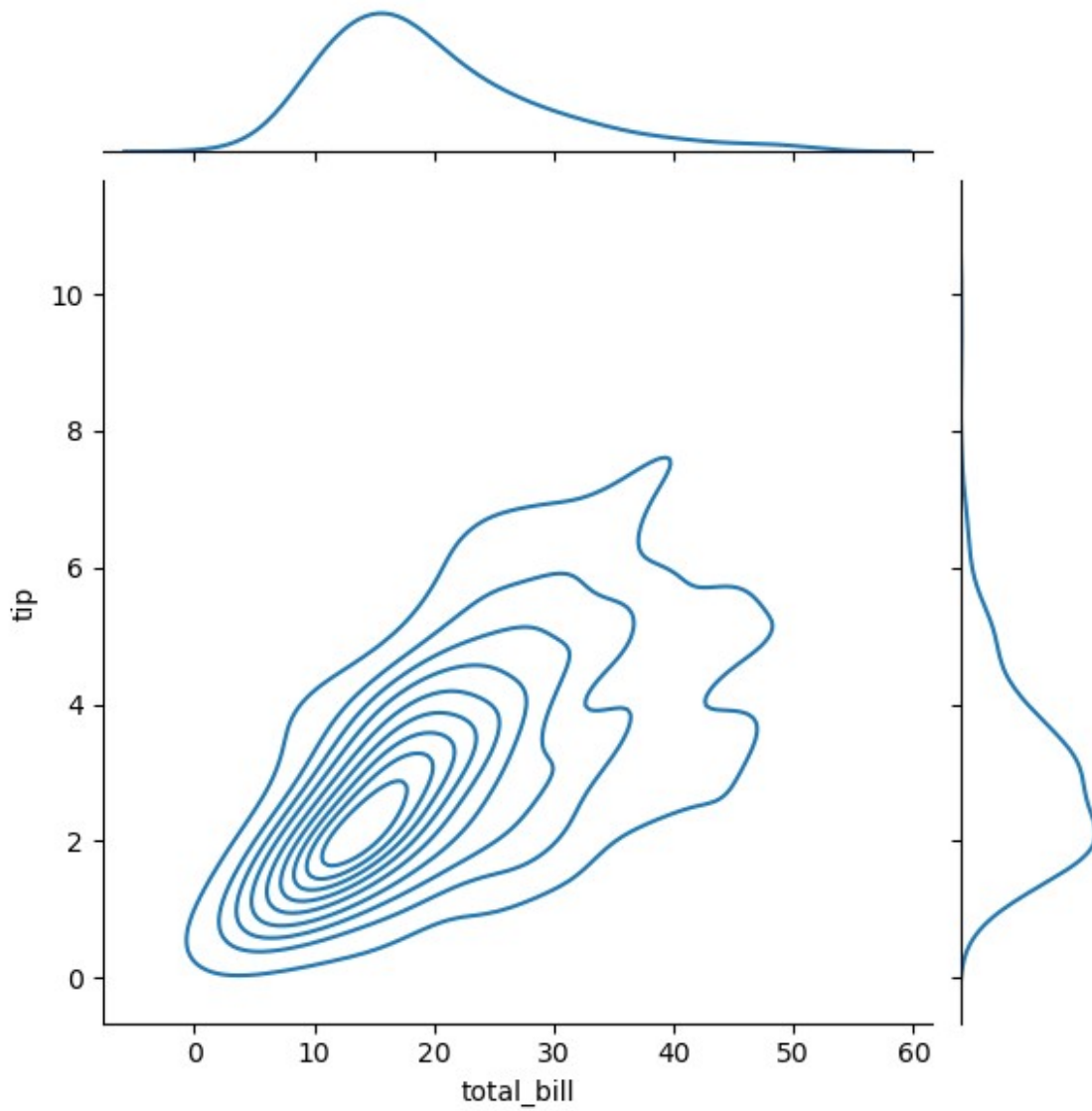
```
sns.jointplot(x='total_bill',y='tip',data=tips,kind='hex')  
<seaborn.axisgrid.JointGrid at 0x1335cf607d0>
```



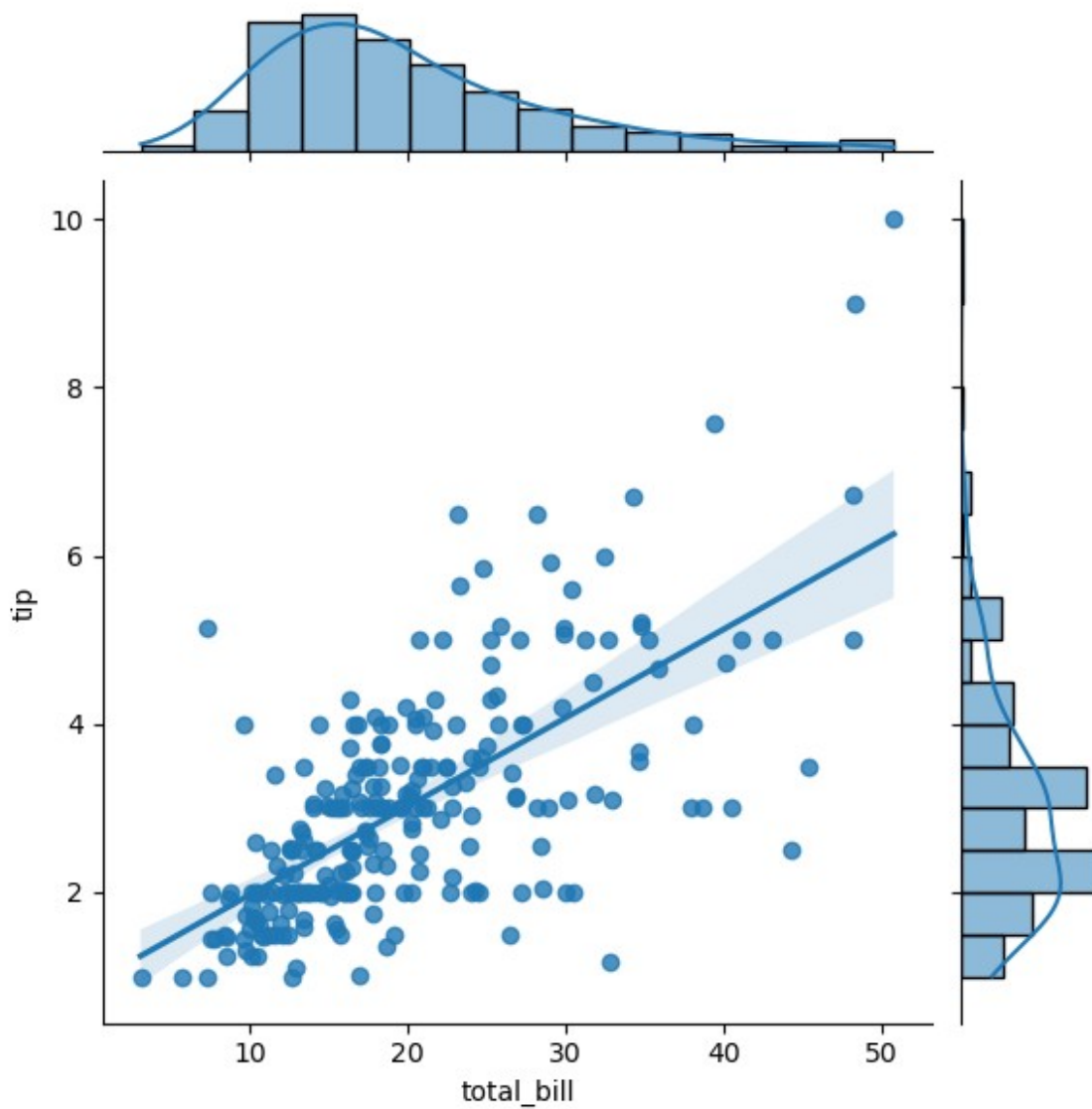
```
sns.jointplot(x='total_bill',y='tip',data=tips,kind='hist')  
<seaborn.axisgrid.JointGrid at 0x1335d49a350>
```



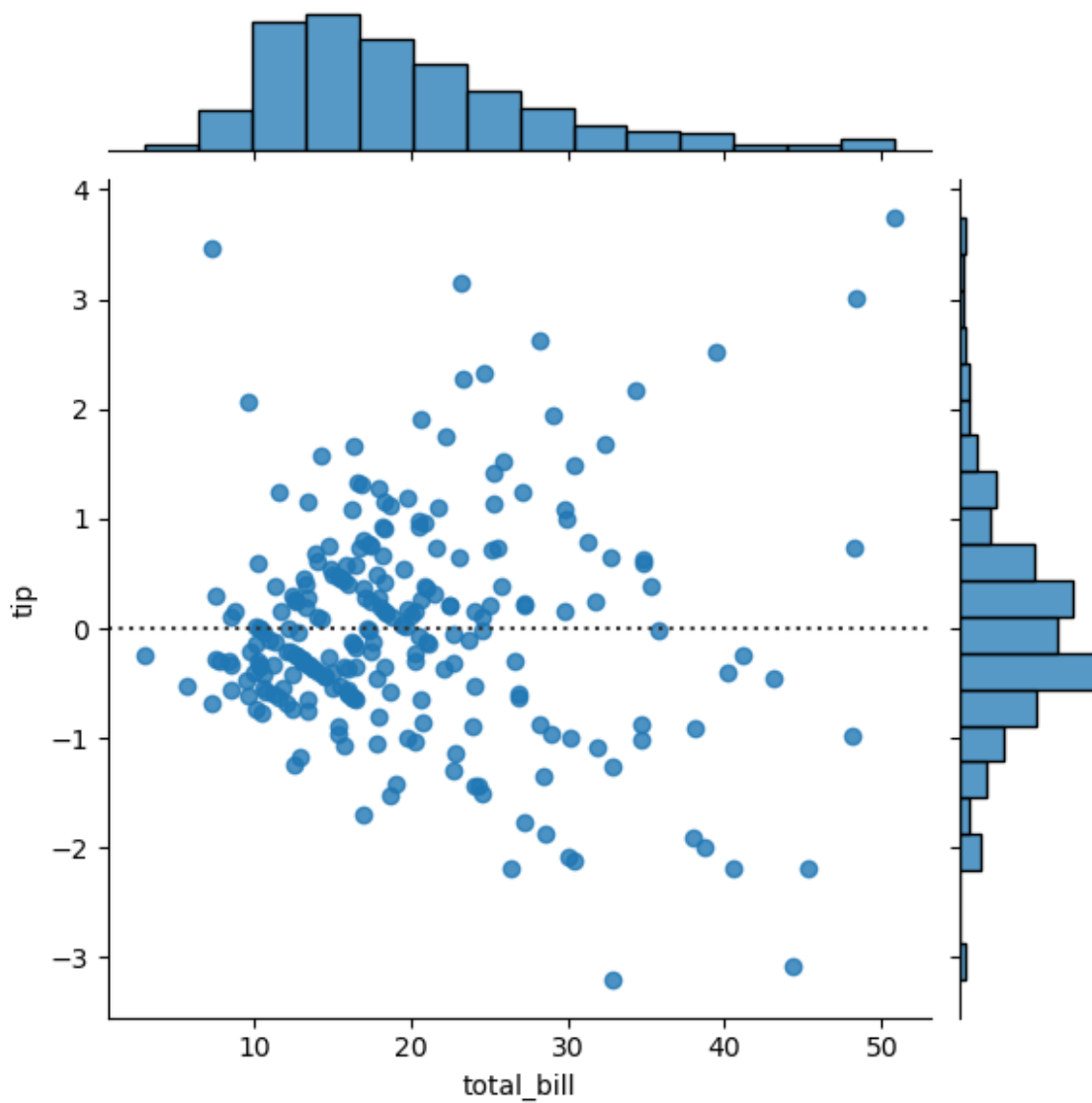
```
sns.jointplot(x='total_bill',y='tip',data=tips,kind='kde')  
<seaborn.axisgrid.JointGrid at 0x1335d041940>
```



```
sns.jointplot(x='total_bill',y='tip',data=tips,kind='reg')  
<seaborn.axisgrid.JointGrid at 0x1335d042c40>
```



```
sns.jointplot(x='total_bill',y='tip',data=tips,kind='resid')  
<seaborn.axisgrid.JointGrid at 0x1335d0e8b90>
```

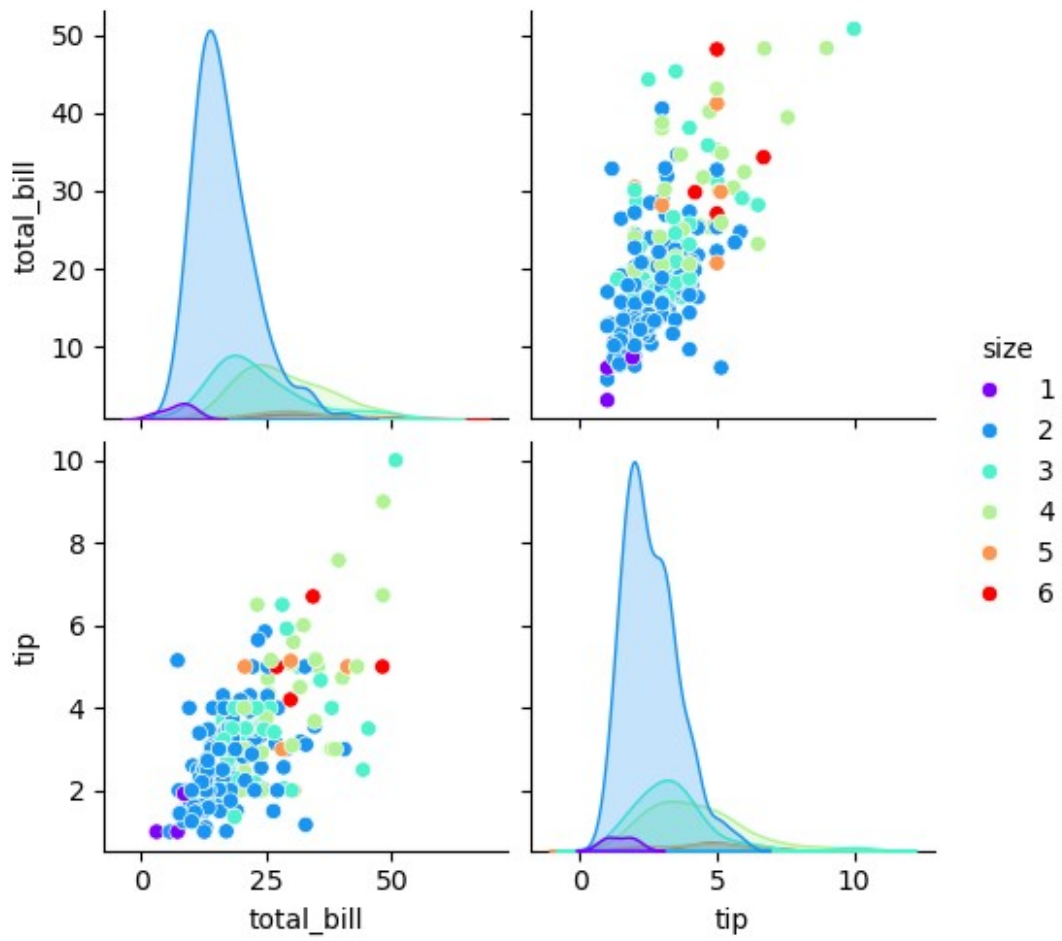


Pair PLOT(imp)

```
sns.pairplot(tips,hue='size',palette='rainbow')
```

```
<seaborn.axisgrid.PairGrid at 0x1335a5763c0>
```

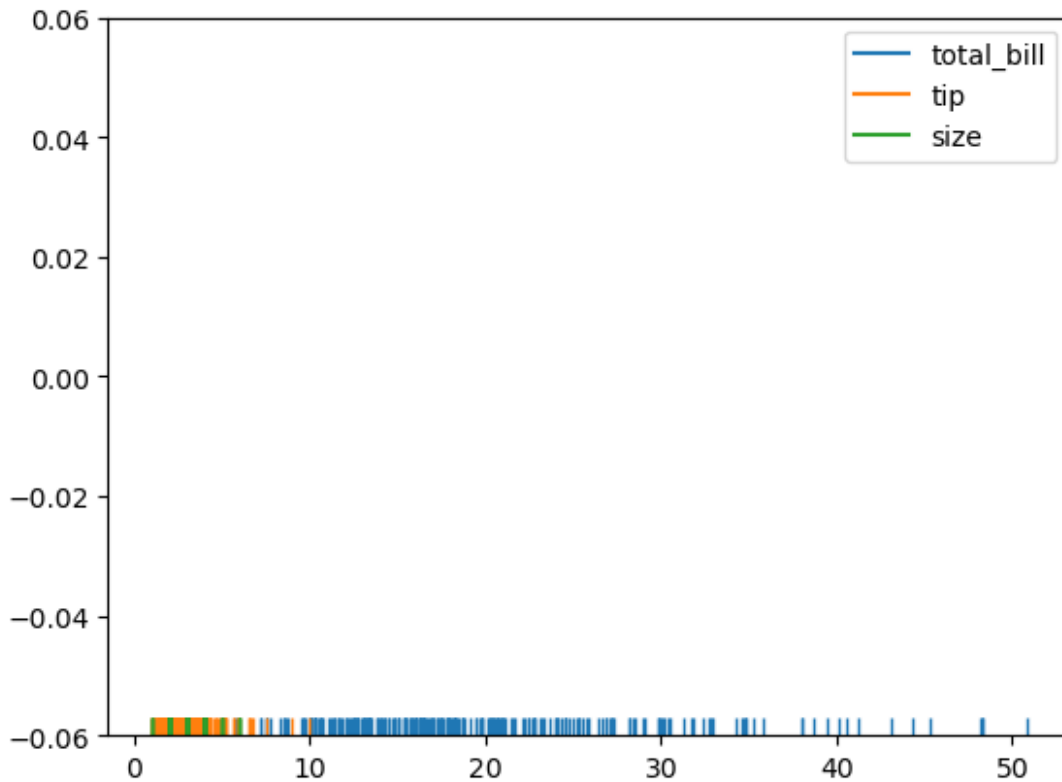




rugplot

```
sns.rugplot(tips)
```

<Axes: >



## Categorical Plots

types:-

- box plot
- violin plot
- strripp plot
- swarm plot
- barplot
- countplot

tips

	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
...	...	...	...	...	...	...	...
239	29.03	5.92	Male	No	Sat	Dinner	3
240	27.18	2.00	Female	Yes	Sat	Dinner	2
241	22.67	2.00	Male	Yes	Sat	Dinner	2

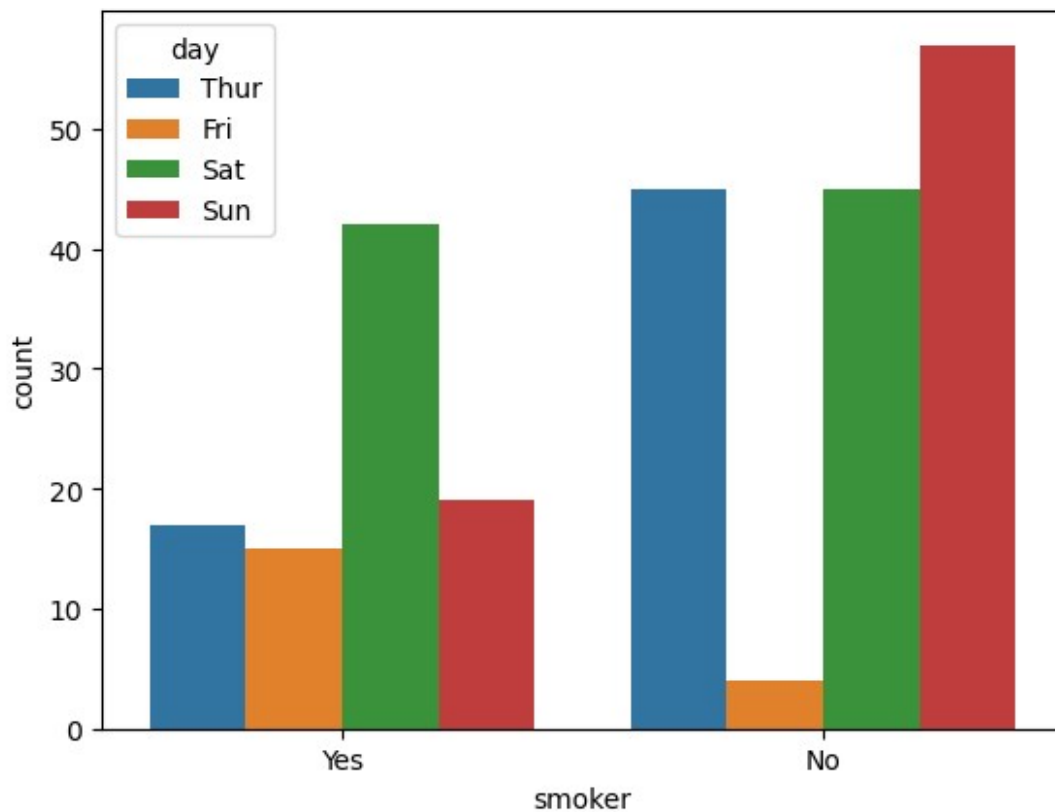
242	17.82	1.75	Male	No	Sat	Dinner	2
243	18.78	3.00	Female	No	Thur	Dinner	2

[244 rows x 7 columns]

## count plot

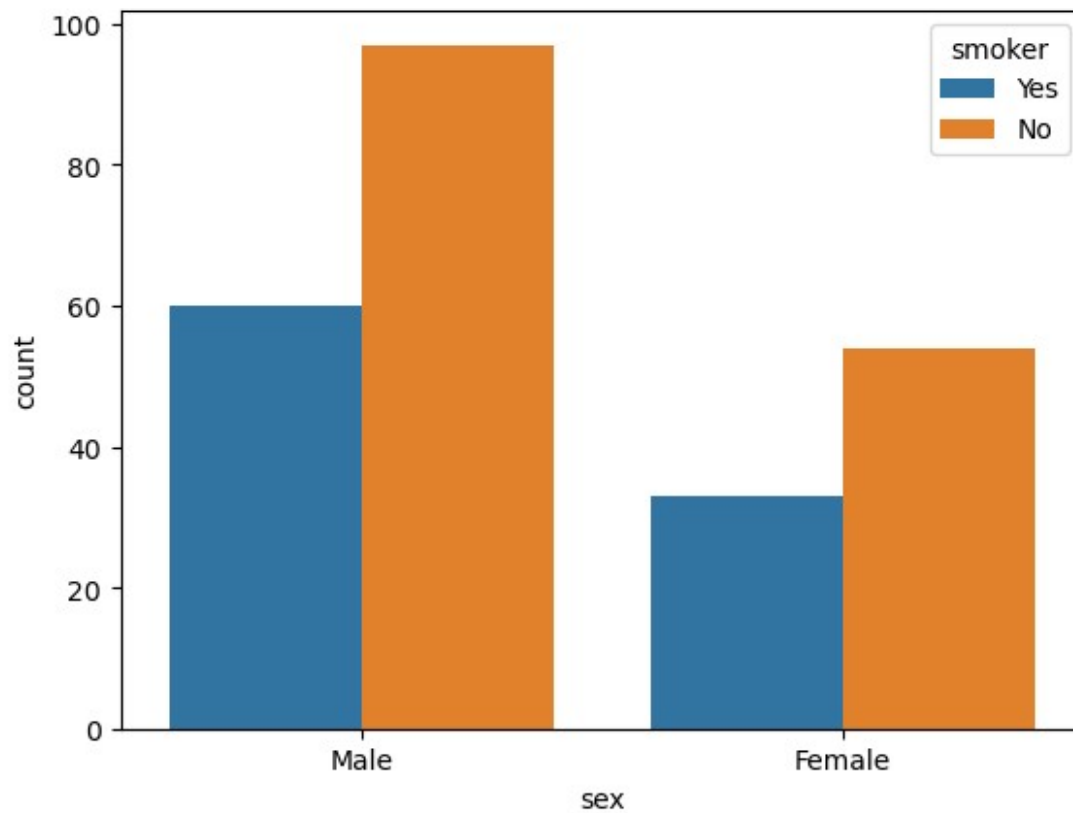
```
sns.countplot(x='smoker',hue='day',data=tips)
```

<Axes: xlabel='smoker', ylabel='count'>

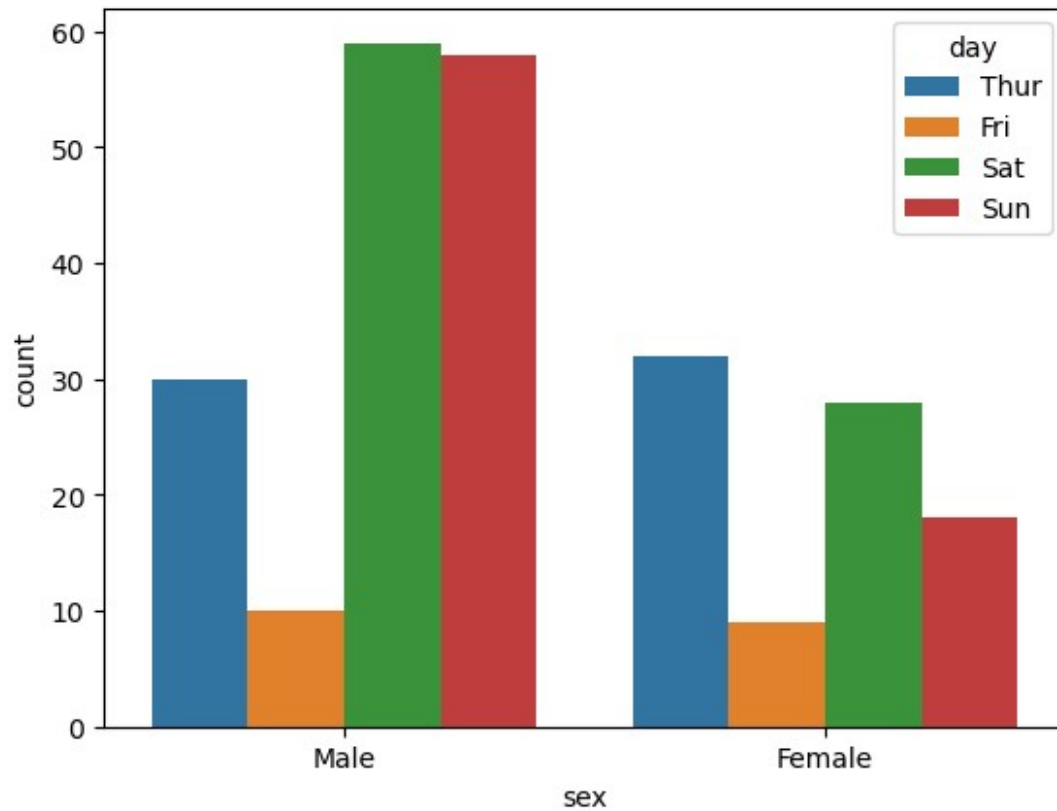


```
sns.countplot(x=tips['sex'],hue=tips['smoker'])
```

<Axes: xlabel='sex', ylabel='count'>



```
sns.countplot(x='sex', hue='smoker', data=tips)  
<Axes: xlabel='sex', ylabel='count'>
```

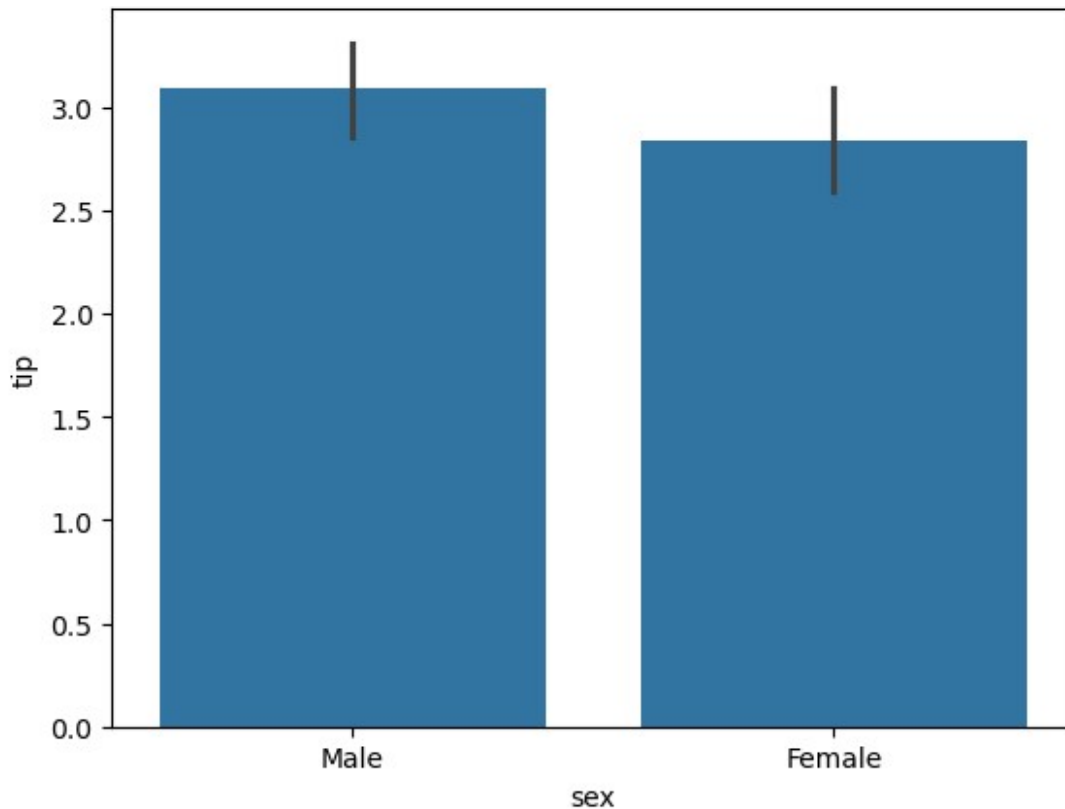


## barplot

y=quantity x=qualitative

```
sns.barplot(x='sex', y='tip', data=tips)
```

```
<Axes: xlabel='sex', ylabel='tip'>
```

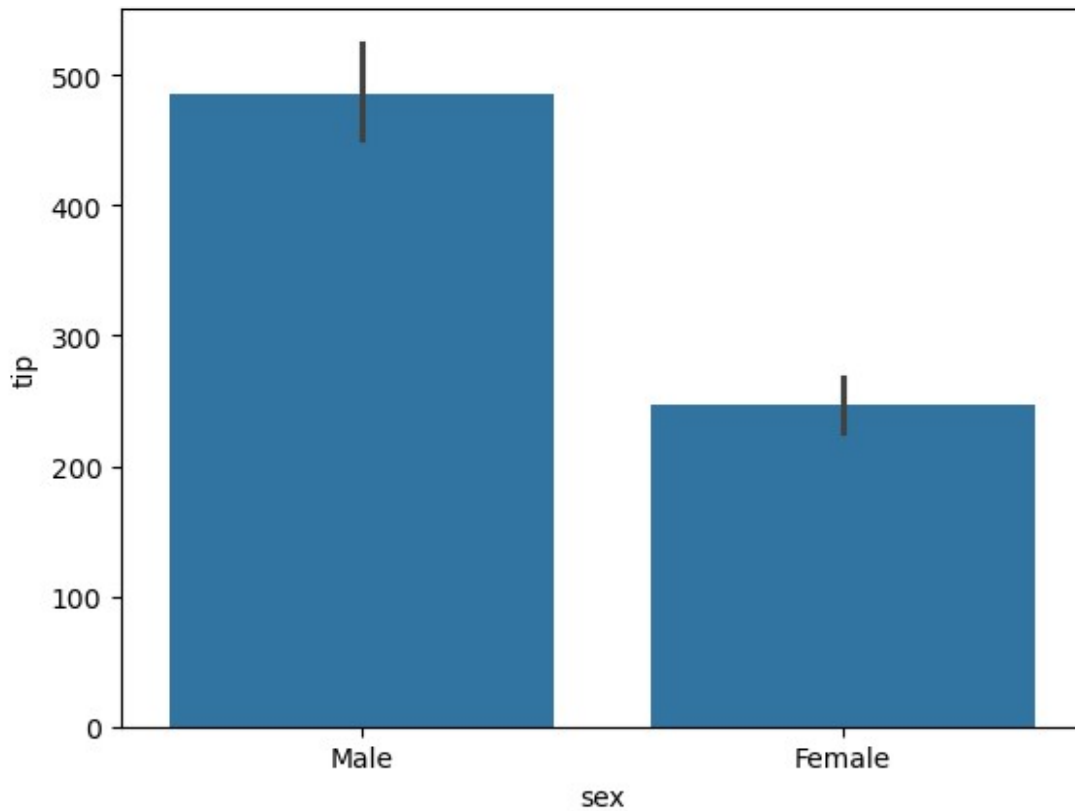


you can change the estimator object to your own function ,that converts a vector to scalar

- Mean(default):np.mean
- Median :np.median
- sum: np.sum
- count:len (to count the number of observation)
- standard deviation : np.std
- minimum:np.min
- Maximum:np.max

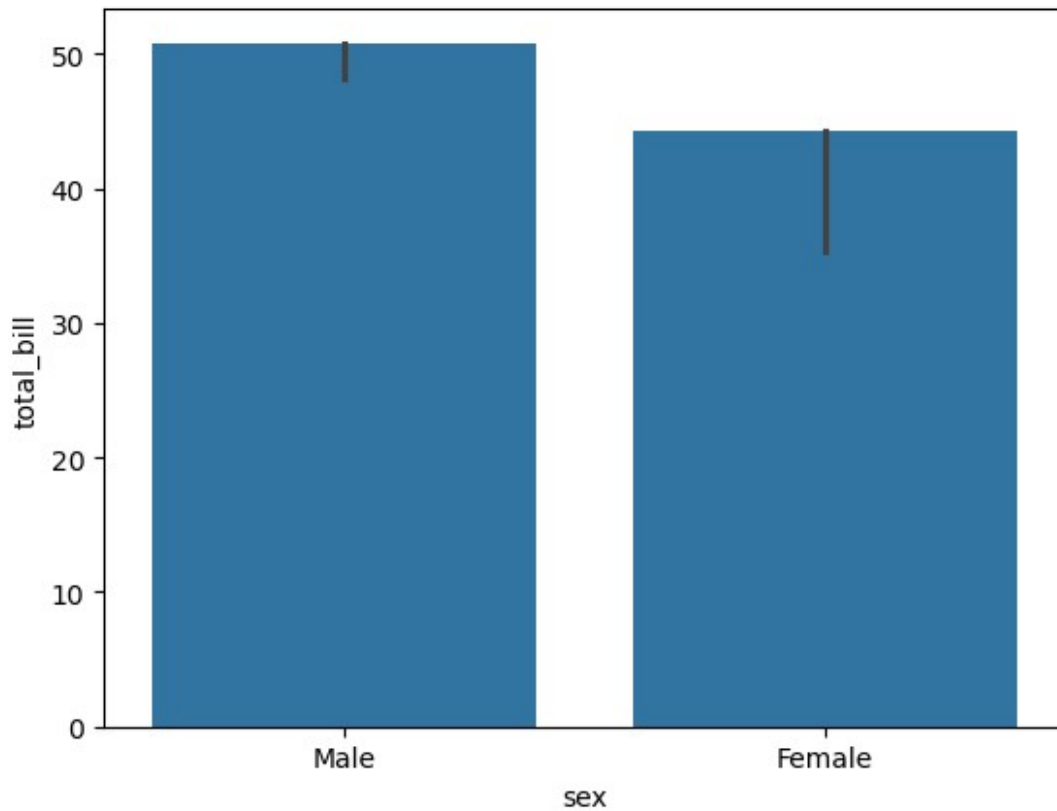
```
sns.barplot(x='sex' ,y='tip',data=tips,estimator=np.sum)
```

```
<Axes: xlabel='sex', ylabel='tip'>
```



means male gives total tip 400 while female gives total tip 200

```
sns.barplot(x='sex' ,y='total_bill',data=tips,estimator=np.max)  
<Axes: xlabel='sex', ylabel='total_bill'>
```

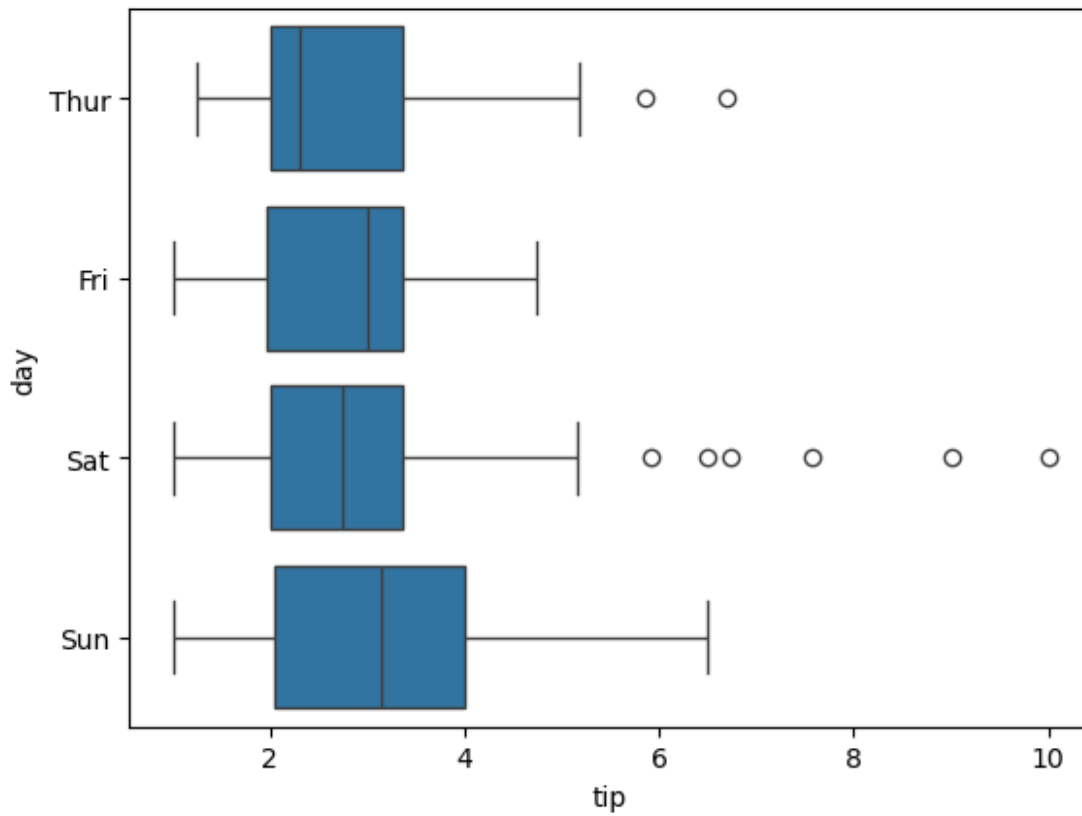


```
tips[tips['sex'] == 'Female']['total_bill'].max() #female max
44.3
tips[tips['sex'] == 'Male']['total_bill'].max() #male max
50.81
```

boxplot(IMP)

```
sns.boxplot(x='tip' , y='day' ,data=tips)
<Axes: xlabel='tip', ylabel='day'>
```



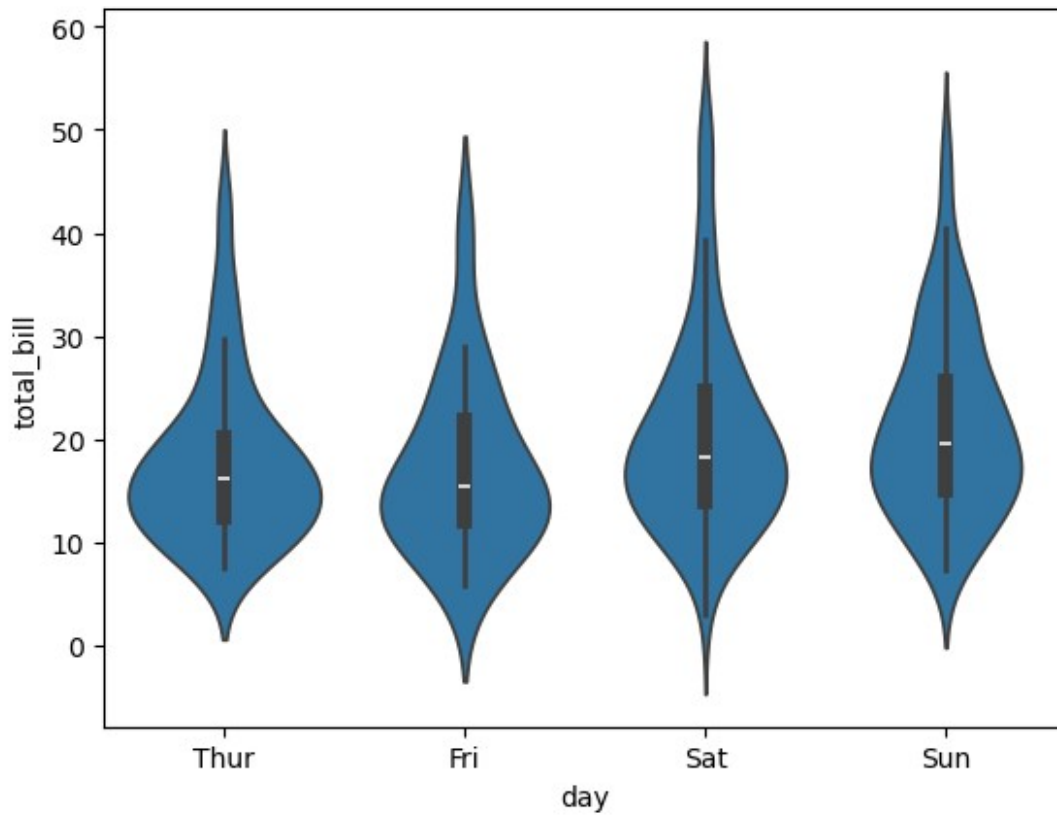


y=qualatative x=quantatitive

## Violen Plot

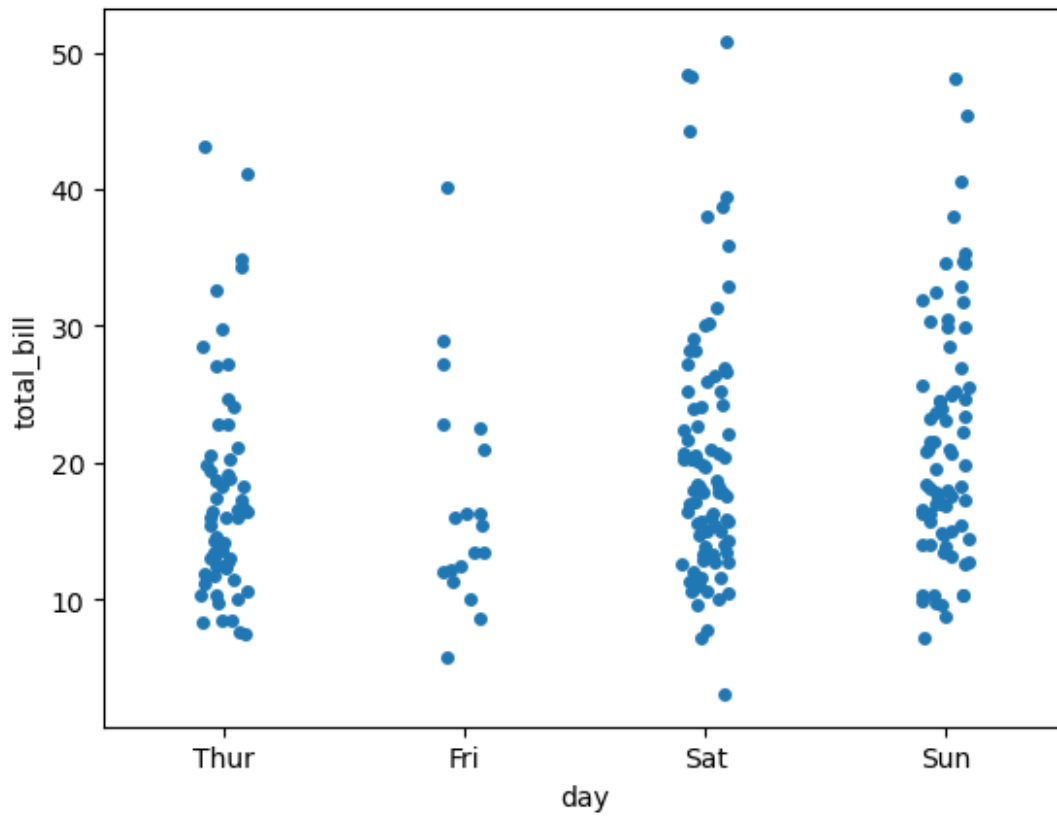
```
sns.violinplot(x='day',y='total_bill',data=tips)
```

```
<Axes: xlabel='day', ylabel='total_bill'>
```



## Stripplot

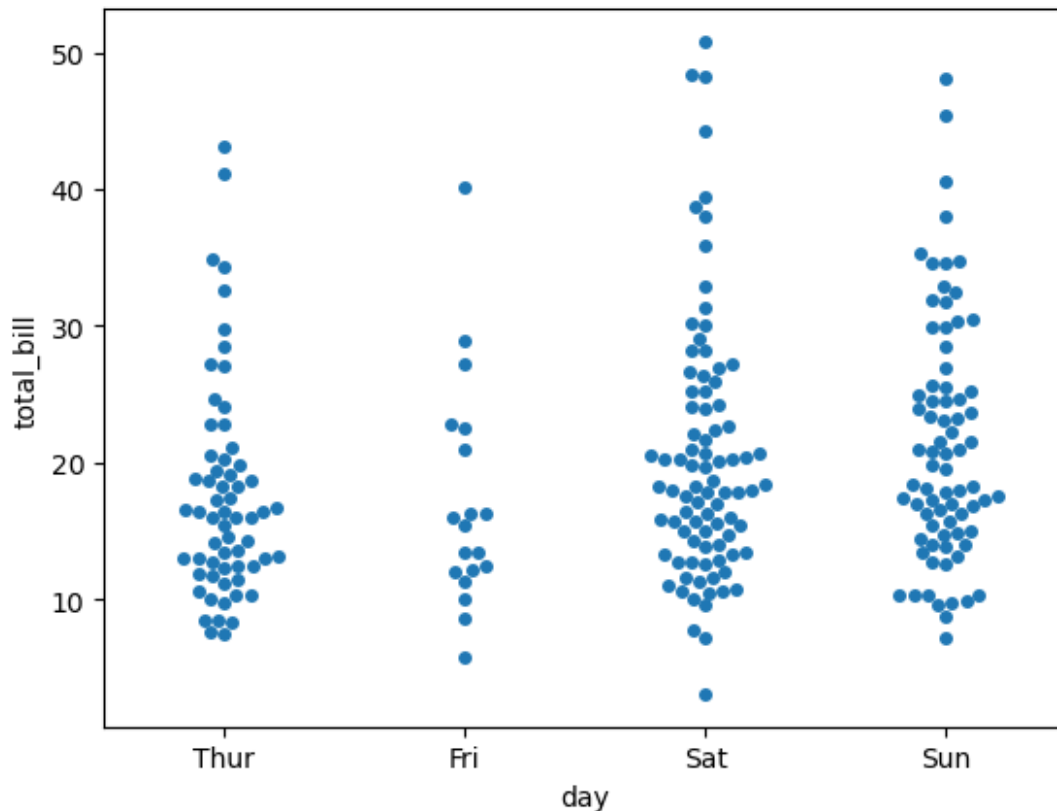
```
sns.stripplot(x='day',y='total_bill',data=tips)  
<Axes: xlabel='day', ylabel='total_bill'>
```



## SwarmPlot

```
sns.swarmplot(x='day',y='total_bill',data=tips)
```

```
<Axes: xlabel='day', ylabel='total_bill'>
```



### violin plus swarm plot

```
sns.violinplot(x='tip',y='day',data=tips)
sns.swarmplot(x='tip',y='day',data=tips)
```

```
C:\Users\ACER\anaconda3\Lib\site-packages\seaborn\categorical.py:3399:
UserWarning: 9.7% of the points cannot be placed; you may want to
decrease the size of the markers or use stripplot.
```

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

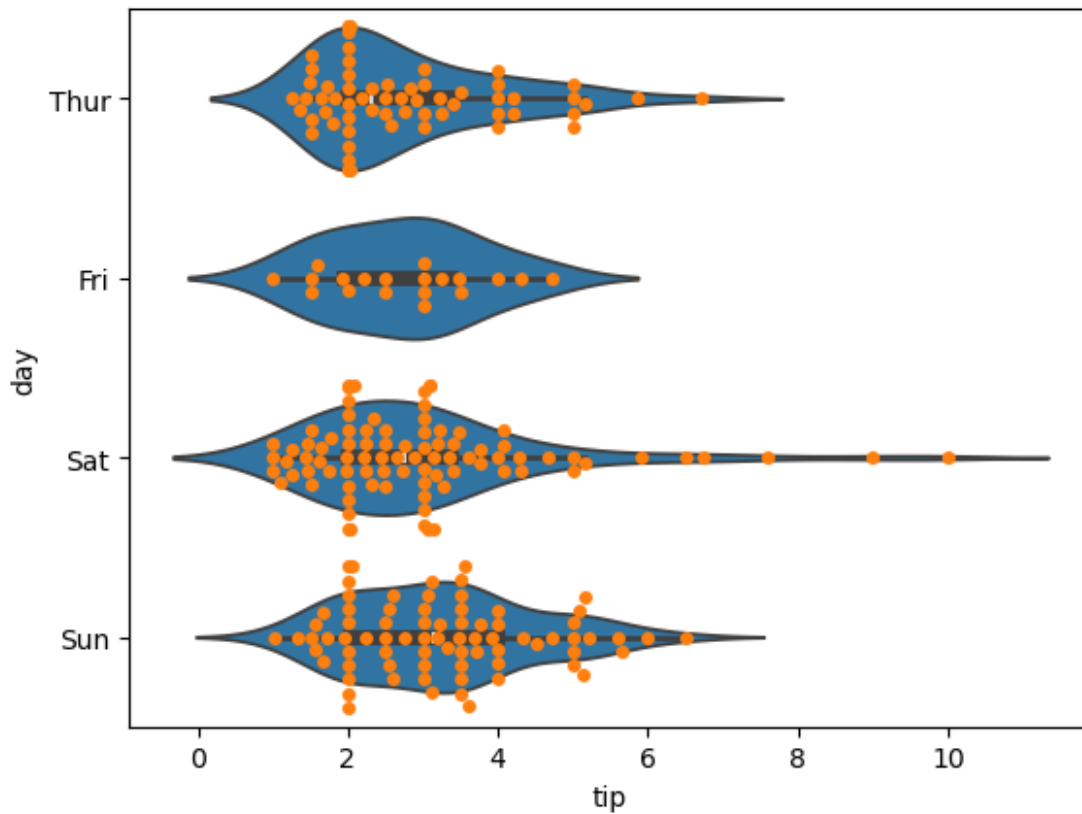
```
C:\Users\ACER\anaconda3\Lib\site-packages\seaborn\categorical.py:3399:
UserWarning: 10.3% of the points cannot be placed; you may want to
decrease the size of the markers or use stripplot.
```

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

```
<Axes: xlabel='tip', ylabel='day'>
```

```
C:\Users\ACER\anaconda3\Lib\site-packages\seaborn\categorical.py:3399:
UserWarning: 8.0% of the points cannot be placed; you may want to
decrease the size of the markers or use stripplot.
```

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



## MATRIX PLOT

- heat MAP
- Cluster map
- pivot table heat map

```
flights=sns.load_dataset('flights')
```

```
flights
```

	year	month	passengers
0	1949	Jan	112
1	1949	Feb	118
2	1949	Mar	132
3	1949	Apr	129
4	1949	May	121
...	...	...	...
139	1960	Aug	606
140	1960	Sep	508
141	1960	Oct	461
142	1960	Nov	390
143	1960	Dec	432

```
[144 rows x 3 columns]
```

```
tips
```

	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
...	...	...	...	...	...	...	...
239	29.03	5.92	Male	No	Sat	Dinner	3
240	27.18	2.00	Female	Yes	Sat	Dinner	2
241	22.67	2.00	Male	Yes	Sat	Dinner	2
242	17.82	1.75	Male	No	Sat	Dinner	2
243	18.78	3.00	Female	No	Thur	Dinner	2

```
[244 rows x 7 columns]
```

## Heat MAP

use correlation which is used to find relation btw two cloumns for prediction

-1<=0<=1 range,0 means not useful

positive corelation means as one increase another term also increase while for negative correlation vice versa

```
tipscorr=tips[['total_bill','tip','size']]
```

```
tipscorr
```

	total_bill	tip	size
0	16.99	1.01	2
1	10.34	1.66	3
2	21.01	3.50	3
3	23.68	3.31	2
4	24.59	3.61	4
...	...	...	...
239	29.03	5.92	3
240	27.18	2.00	2
241	22.67	2.00	2
242	17.82	1.75	2
243	18.78	3.00	2

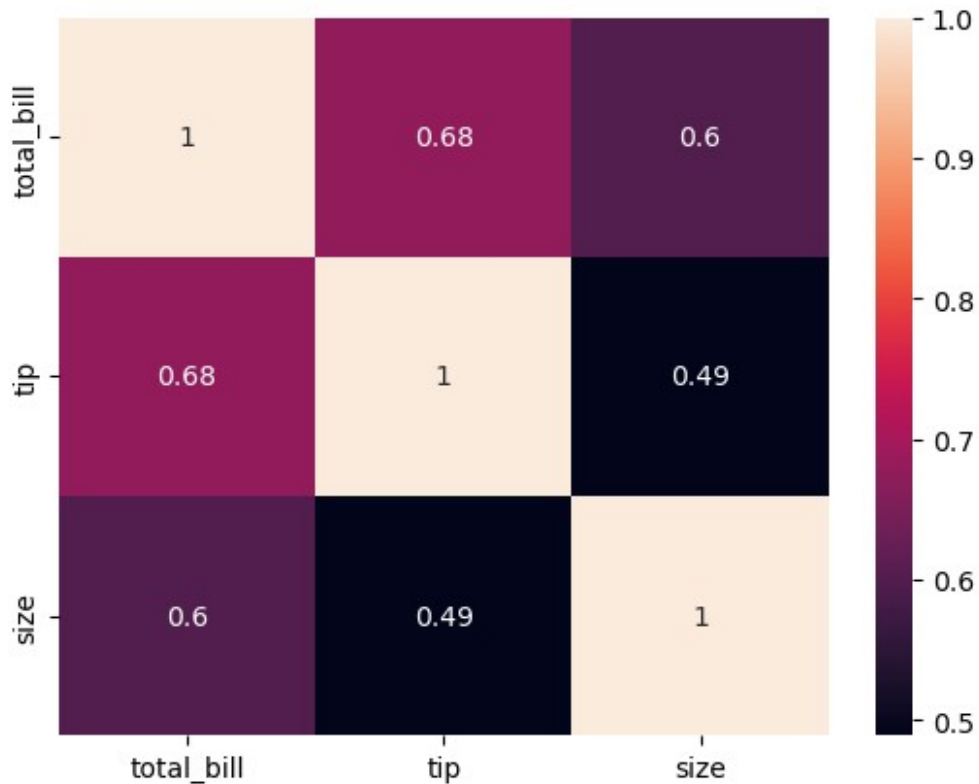
```
[244 rows x 3 columns]
```

```
tipscorr.corr()
```

	total_bill	tip	size
total_bill	1.000000	0.675734	0.598315
tip	0.675734	1.000000	0.489299
size	0.598315	0.489299	1.000000

```
sns.heatmap(tipscorr.corr(),annot=True)
```

```
<Axes: >
```



## Cluster Map

```
sns.clustermap(tipscorr.corr())
```

```
<seaborn.matrix.ClusterGrid at 0x1335a576900>
```

```
Error in callback <function _draw_all_if_interactive at  
0x000001335818F9C0> (for post_execute), with arguments args (),kwargs  
{}
```

```
KeyboardInterrupt
```

```
Error in callback <function flush_figures at 0x000001335A568720> (for  
post_execute), with arguments args (),kwargs {}:
```

```
KeyboardInterrupt
```

## Pivot table Heat map

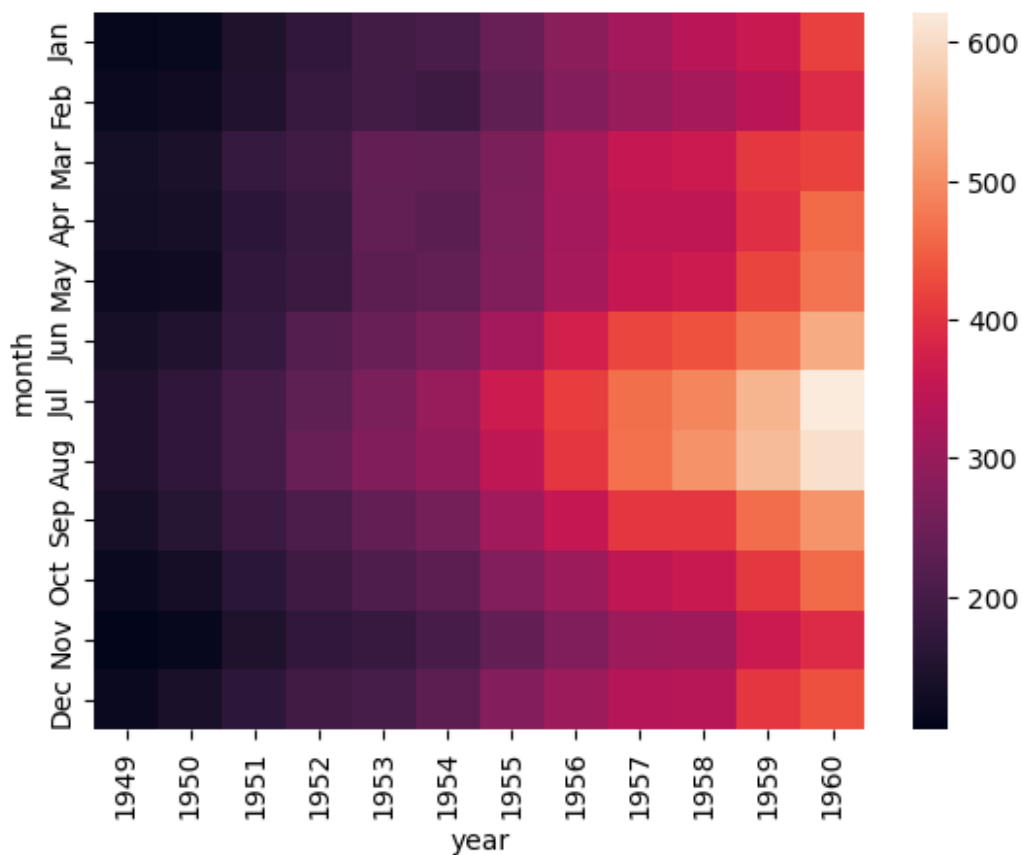
```
pivflight=flights.pivot_table(values='passengers',index='month',columns='year')
```

```
C:\Users\ACER\AppData\Local\Temp\ipykernel_2712\2105619155.py:1:  
FutureWarning: The default value of observed=False is deprecated and  
will change to observed=True in a future version of pandas. Specify  
observed=False to silence this warning and retain the current behavior
```

```
pivflight=flights.pivot_table(values='passengers',index='month',columns='year')
```

```
sns.heatmap(pivflight)
```

```
<Axes: xlabel='year', ylabel='month'>
```



## Regerssion PLOT

it is used in ML ,using lmpot()

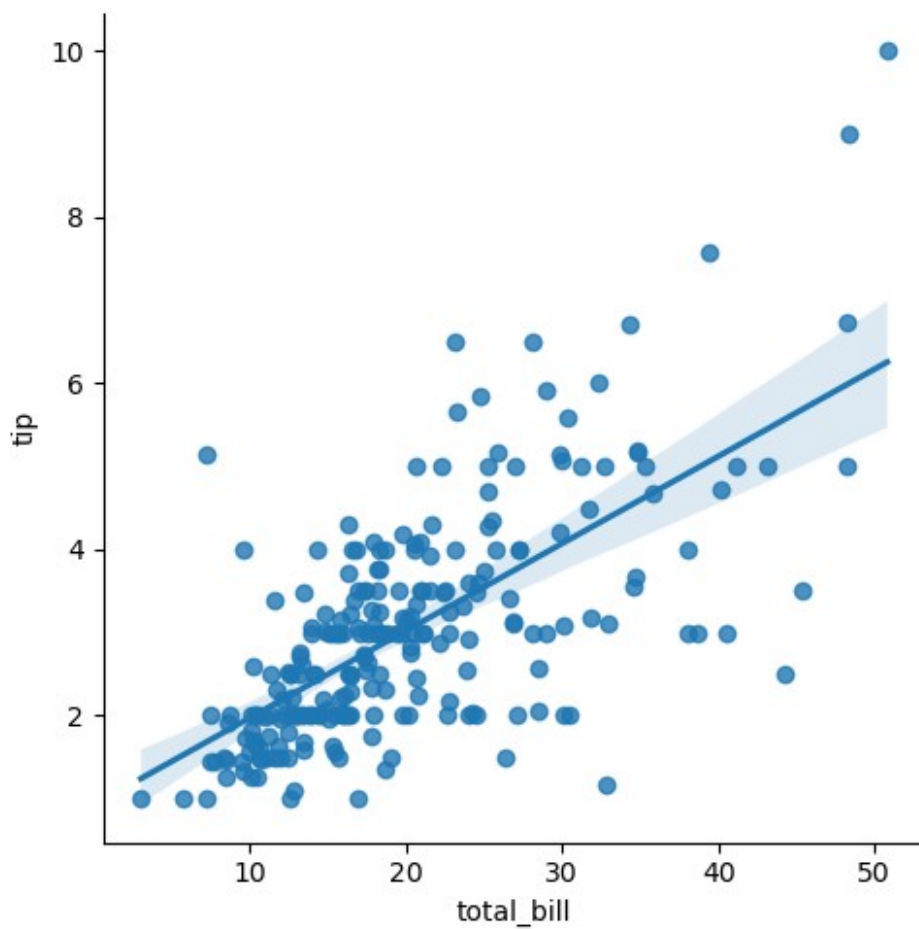
tips



	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
...	...	...	...	...	...	...	...
239	29.03	5.92	Male	No	Sat	Dinner	3
240	27.18	2.00	Female	Yes	Sat	Dinner	2
241	22.67	2.00	Male	Yes	Sat	Dinner	2
242	17.82	1.75	Male	No	Sat	Dinner	2
243	18.78	3.00	Female	No	Thur	Dinner	2

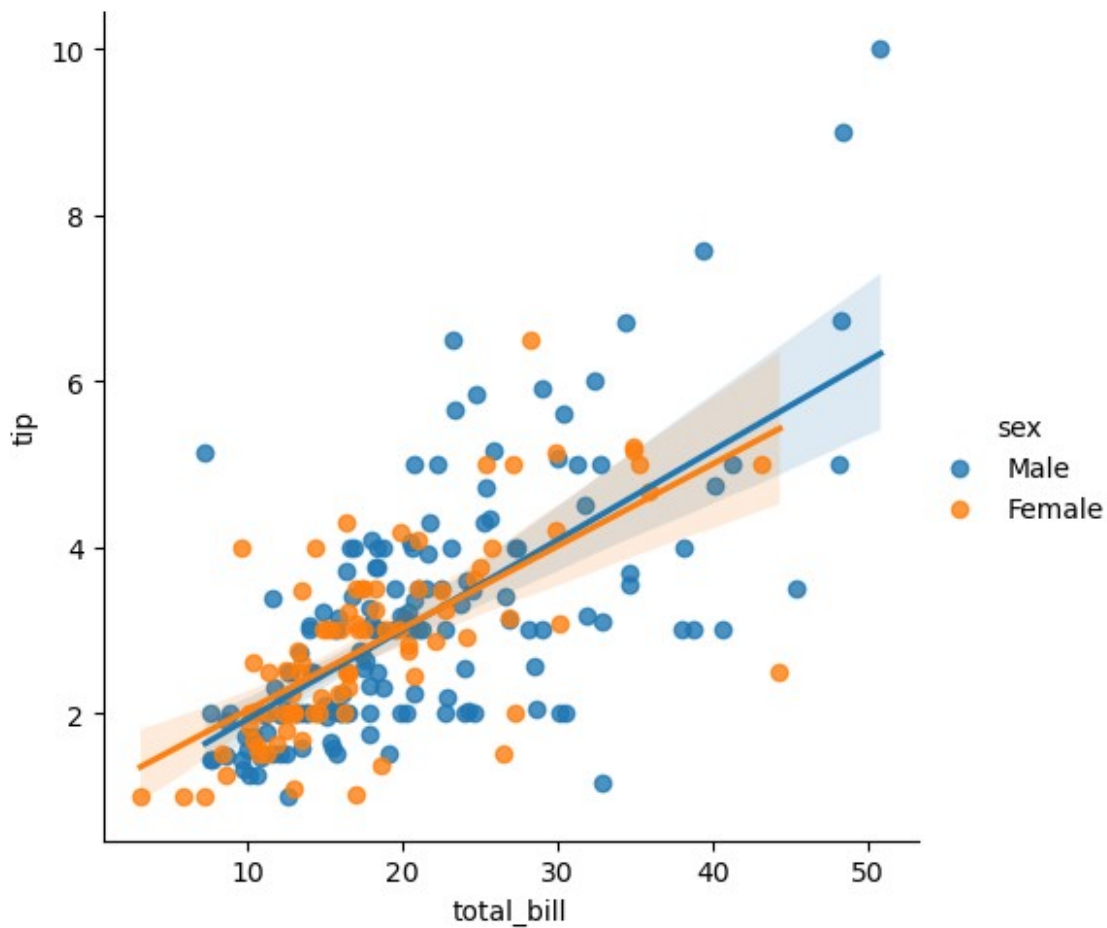
[244 rows x 7 columns]

```
sns.lmplot(x='total_bill',y='tip',data=tips) # normal
<seaborn.axisgrid.FacetGrid at 0x13363f3d7f0>
```



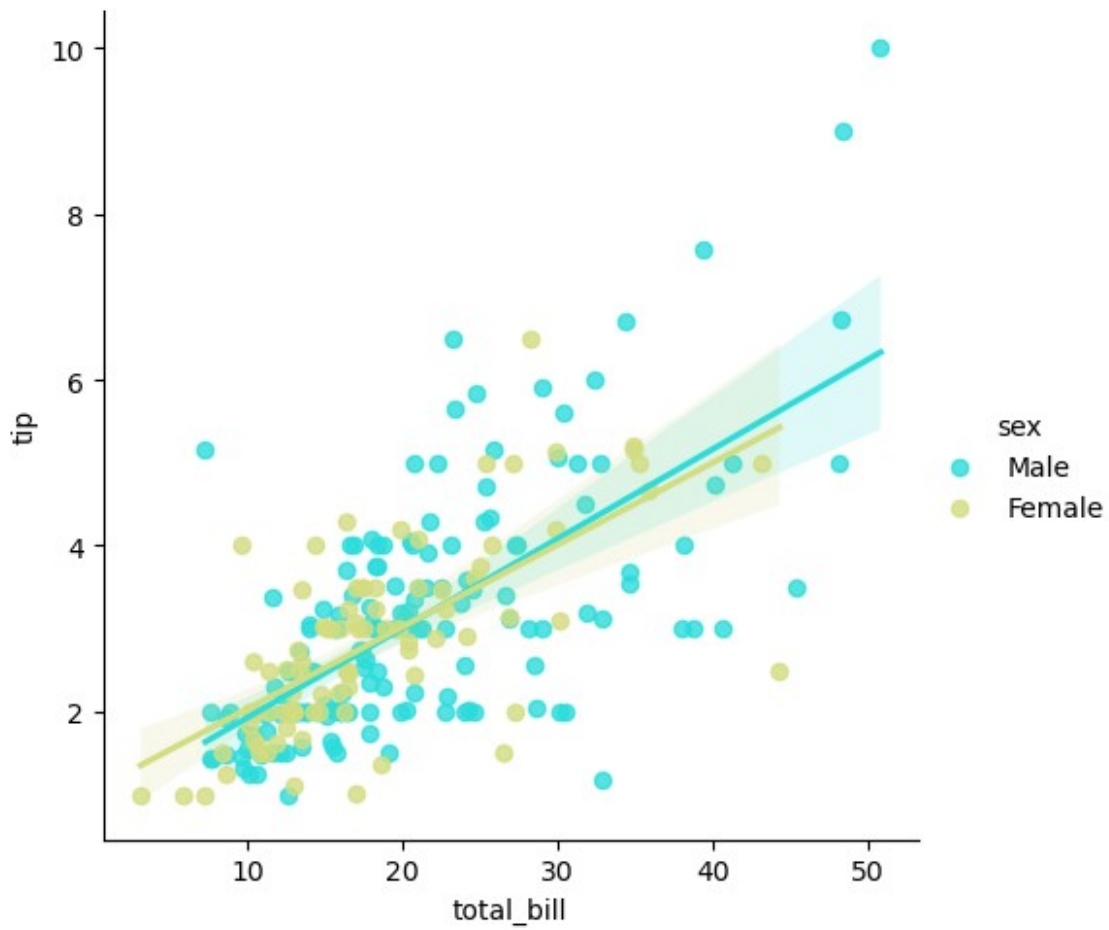
```
sns.lmplot(x='total_bill',y='tip',data=tips,hue='sex') # hue based on
category
```

```
<seaborn.axisgrid.FacetGrid at 0x1336384a990>
```

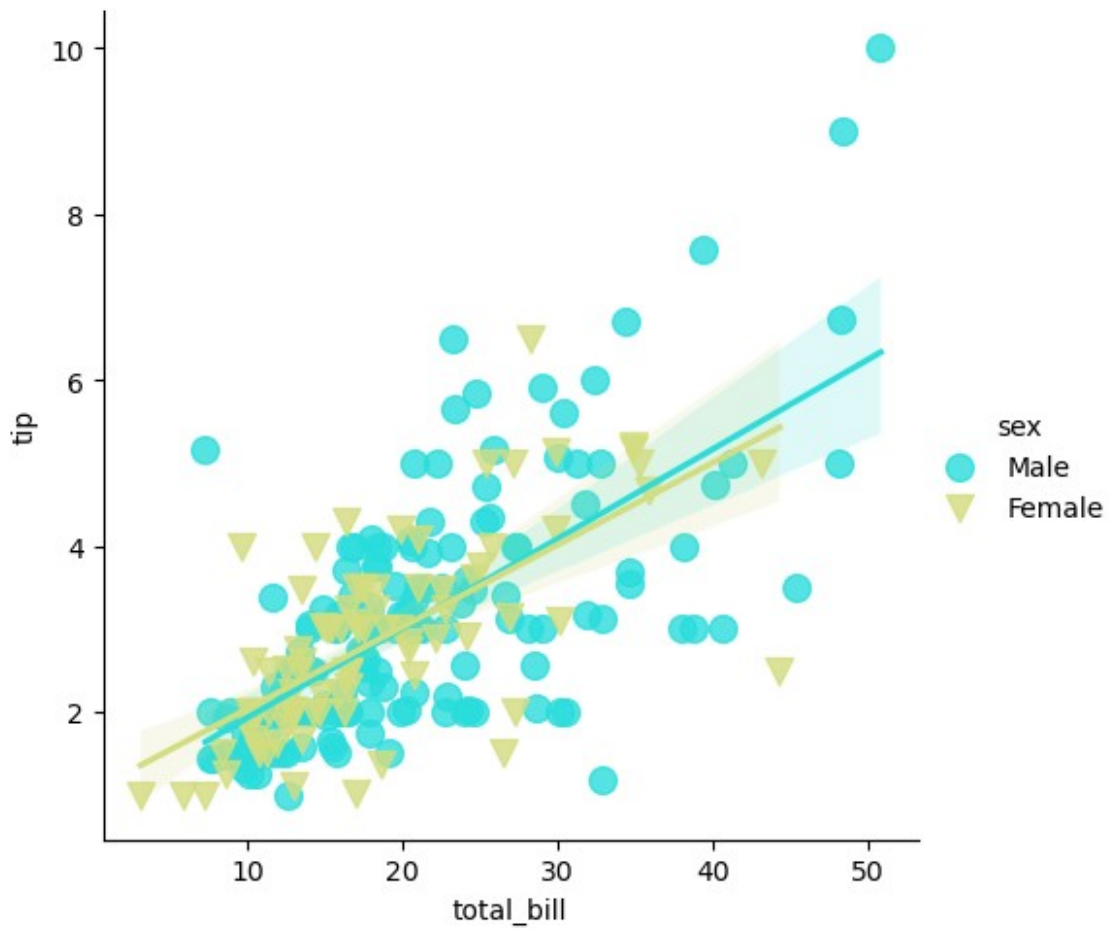


```
sns.lmplot(x='total_bill',y='tip',data=tips,hue='sex',palette='rainbow') #palette design
```

```
<seaborn.axisgrid.FacetGrid at 0x1336384aad0>
```

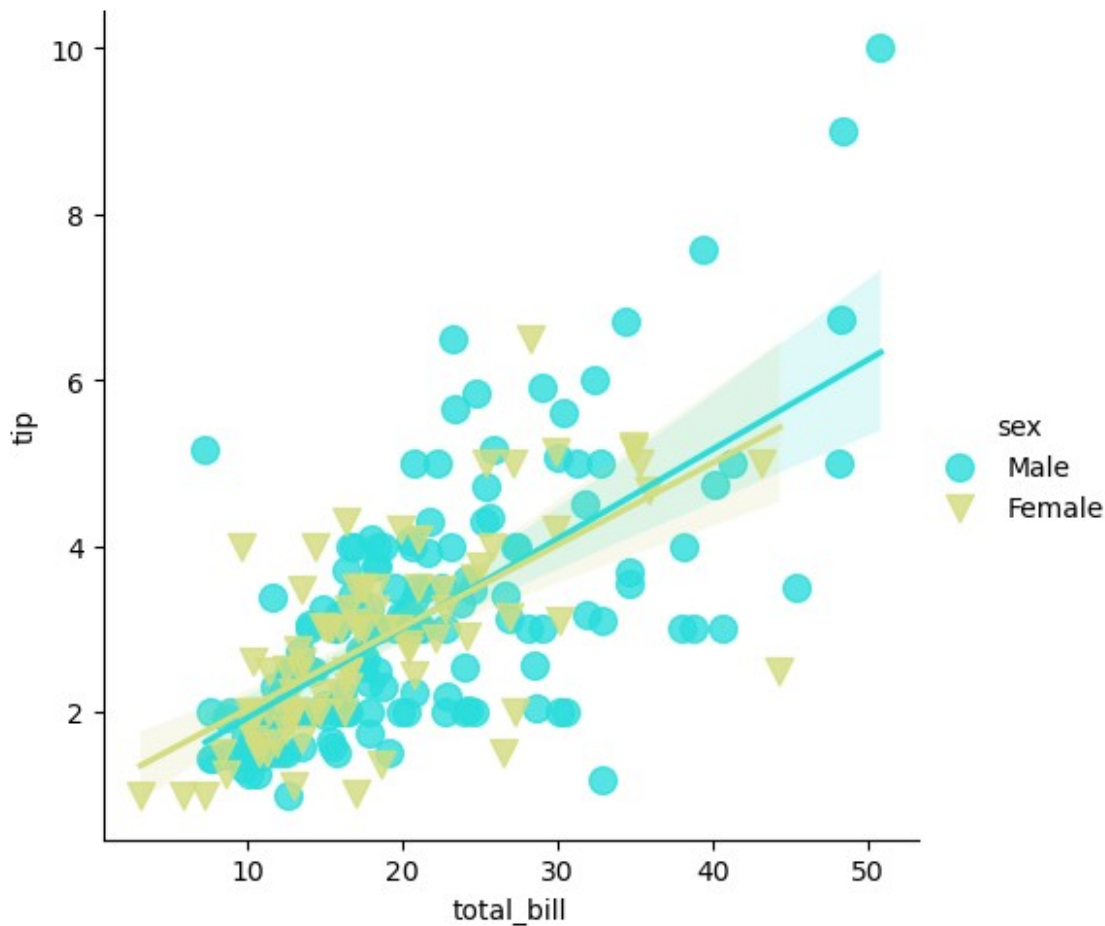


```
sns.lmplot(x='total_bill',y='tip',data=tips,hue='sex',palette='rainbow',markers=['o','v'],scatter_kws={'s':100}) # size and shape of markers<seaborn.axisgrid.FacetGrid at 0x133675a0050>
```



```
sns.lmplot(x='total_bill',y='tip',data=tips,hue='sex',palette='rainbow',markers=['o','v'],scatter_kws={'s':100})
```

```
<seaborn.axisgrid.FacetGrid at 0x133675d2c10>
```



## Plotly and cufflinks

```
# !pip install plotly==4.14.3
```

```
Requirement already satisfied: plotly==4.14.3 in c:\users\acer\anaconda3\lib\site-packages (4.14.3)  
Requirement already satisfied: retrying>=1.3.3 in c:\users\acer\anaconda3\lib\site-packages (from plotly==4.14.3) (1.4.2)  
Requirement already satisfied: six in c:\users\acer\anaconda3\lib\site-packages (from plotly==4.14.3) (1.17.0)
```

```
# !pip install cufflinks
```

not working leave it for now