

Parametros más usados en Seaborn

```
In [ ]: import seaborn as sns
import matplotlib.pyplot as plt
import pandas as pd
```

```
In [ ]: #Importando un data set manualmente
tips= pd.read_csv('dataset/tips.csv')
tips
```

Out[]:

	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 × 7 columns

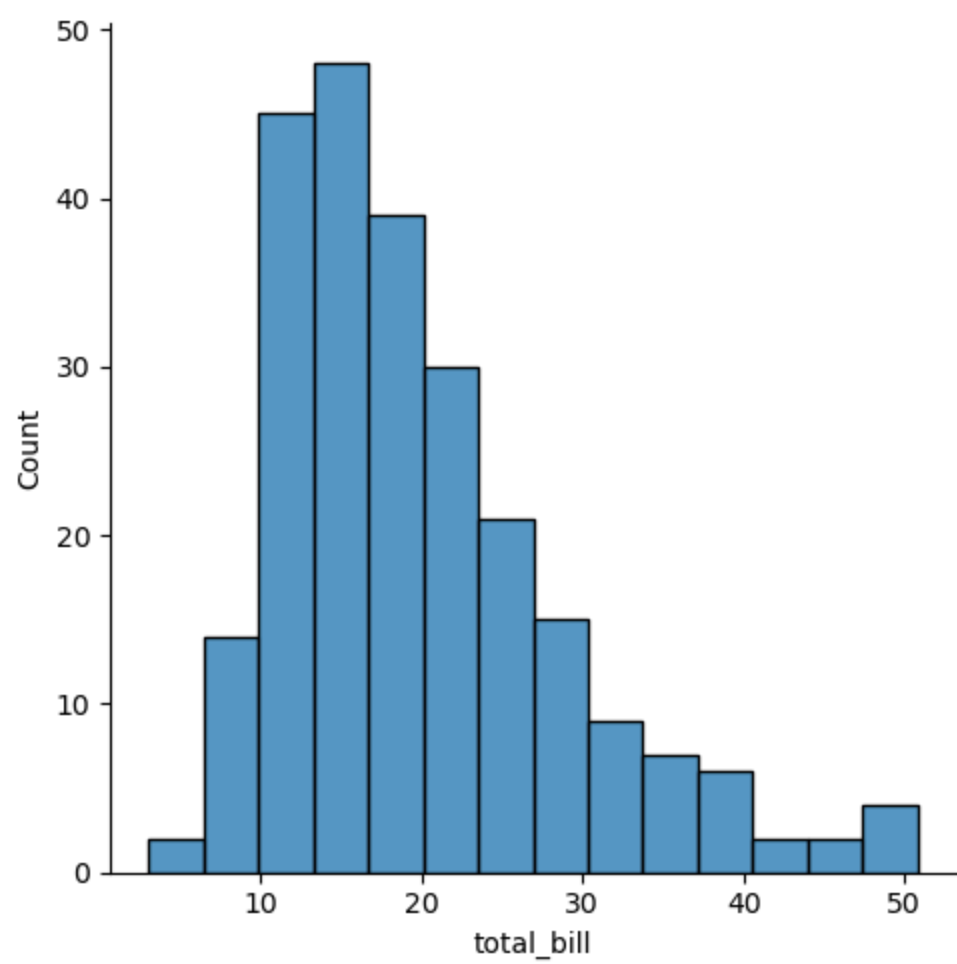
Ver datasets disponibles

```
In [ ]: sns.get_dataset_names()
```

```
['anagrams',
'anscombe',
'attention',
'brain_networks',
'car_crashes',
'diamonds',
'dots',
'dowjones',
'exercise',
'flights',
'fmri',
'geyser',
'glue',
'healthexp',
'iris',
'mpg',
'penguins',
'planets',
'seaice',
'taxis',
'tips',
'titanic']
```

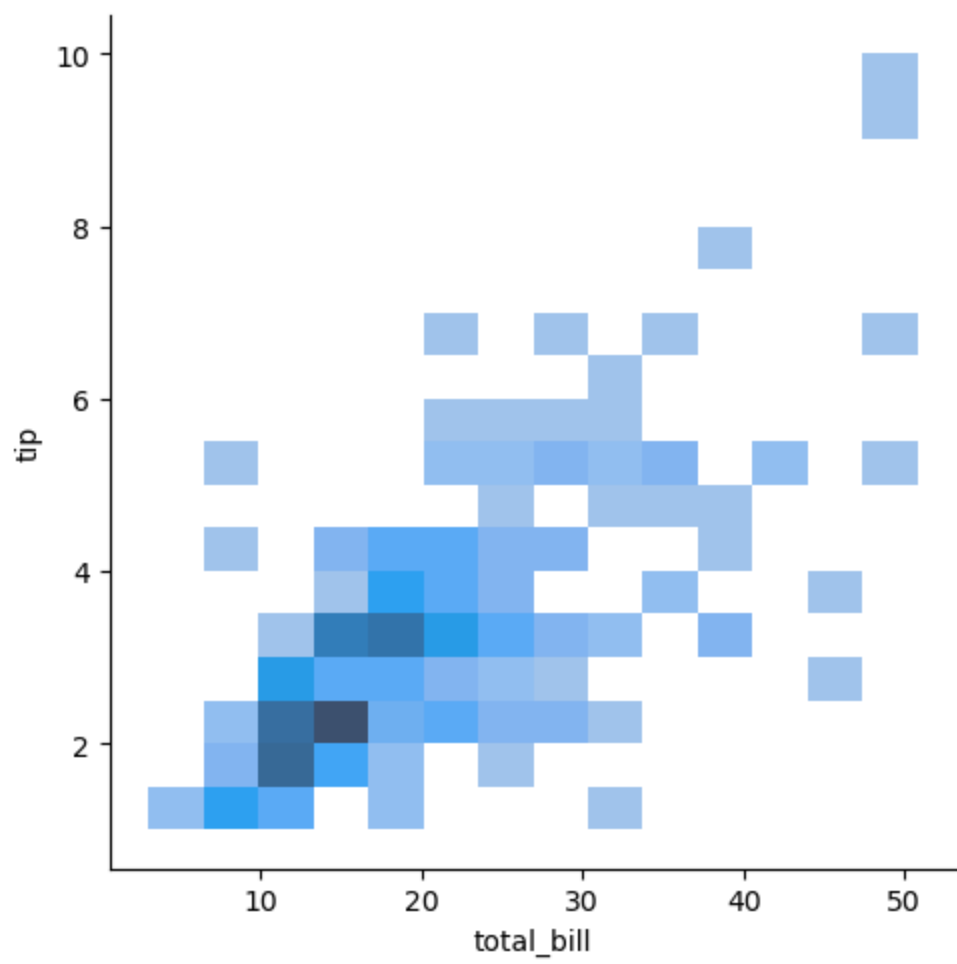
```
In [ ]: #Importando mediante Seaborn
tips = sns.load_dataset('tips')
```

```
In [ ]: #Seaborn está optimizado para Pandas
sns.displot(data=tips, x='total_bill')
plt.show()
```

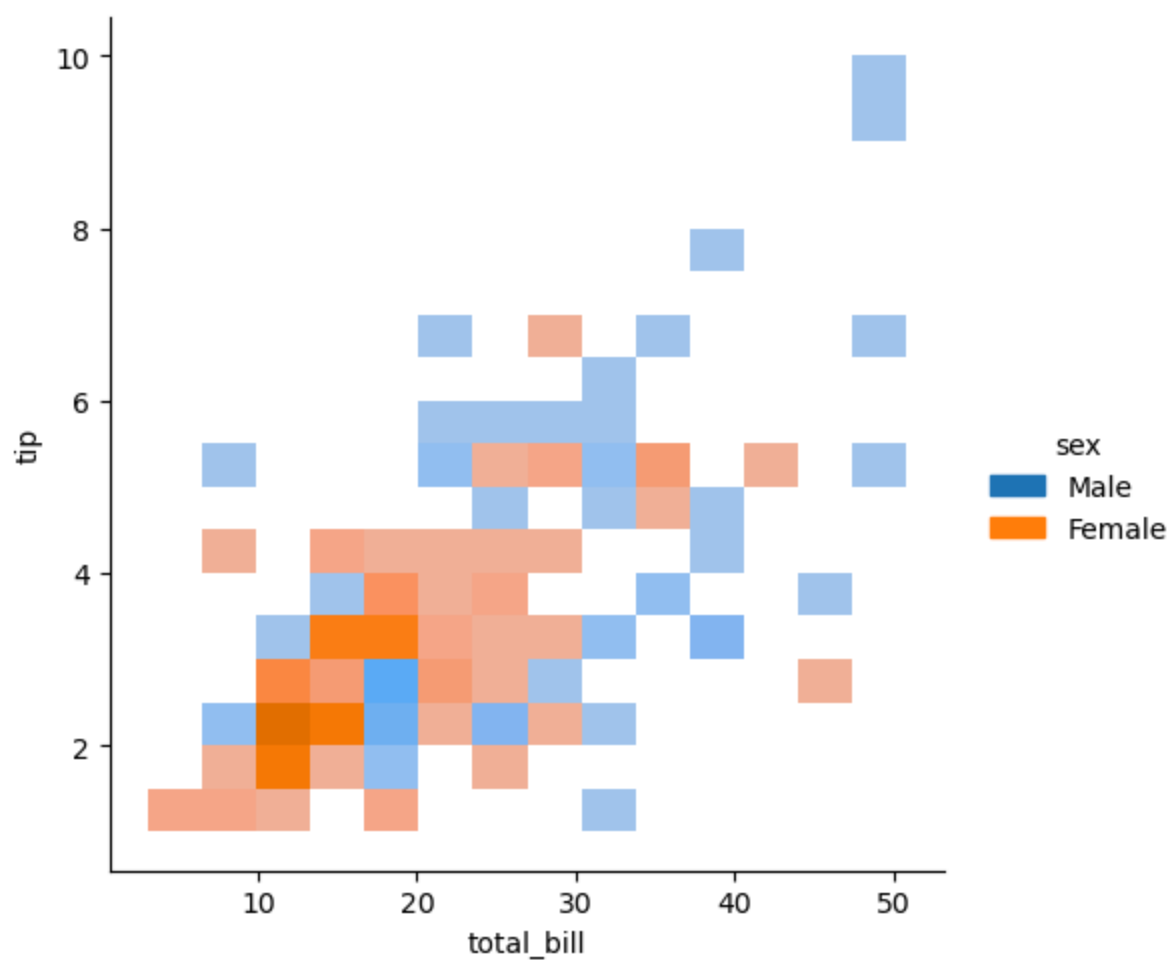


Este es un gráfico de frecuencias, por ello el eje y hace referencia a `Count`

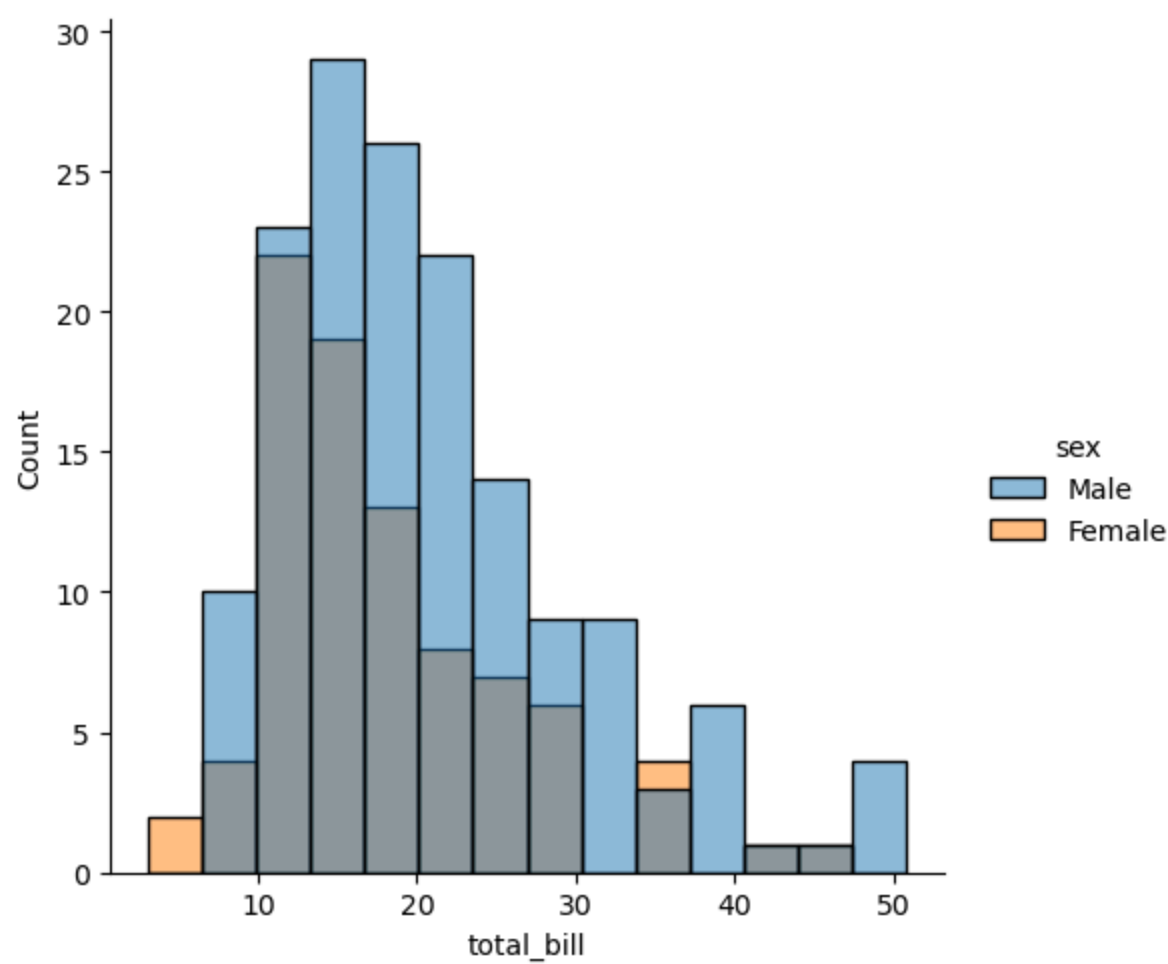
```
In [ ]: #Relacionando 2 variables
sns.displot(data=tips,x='total_bill',y='tip')
plt.show()
```



```
In [ ]: #Relacionando 2 variables
sns.displot(data=tips,x='total_bill',y='tip',hue='sex')
#Haciendo distinción con el parametro hue
plt.show()
```



```
In [ ]: #Funciona mejor cuando tengo un histograma
#Relacionando 2 variables
sns.displot(data=tips,x='total_bill',hue='sex')
#Haciendo distinción con el parametro hue
plt.show()
```



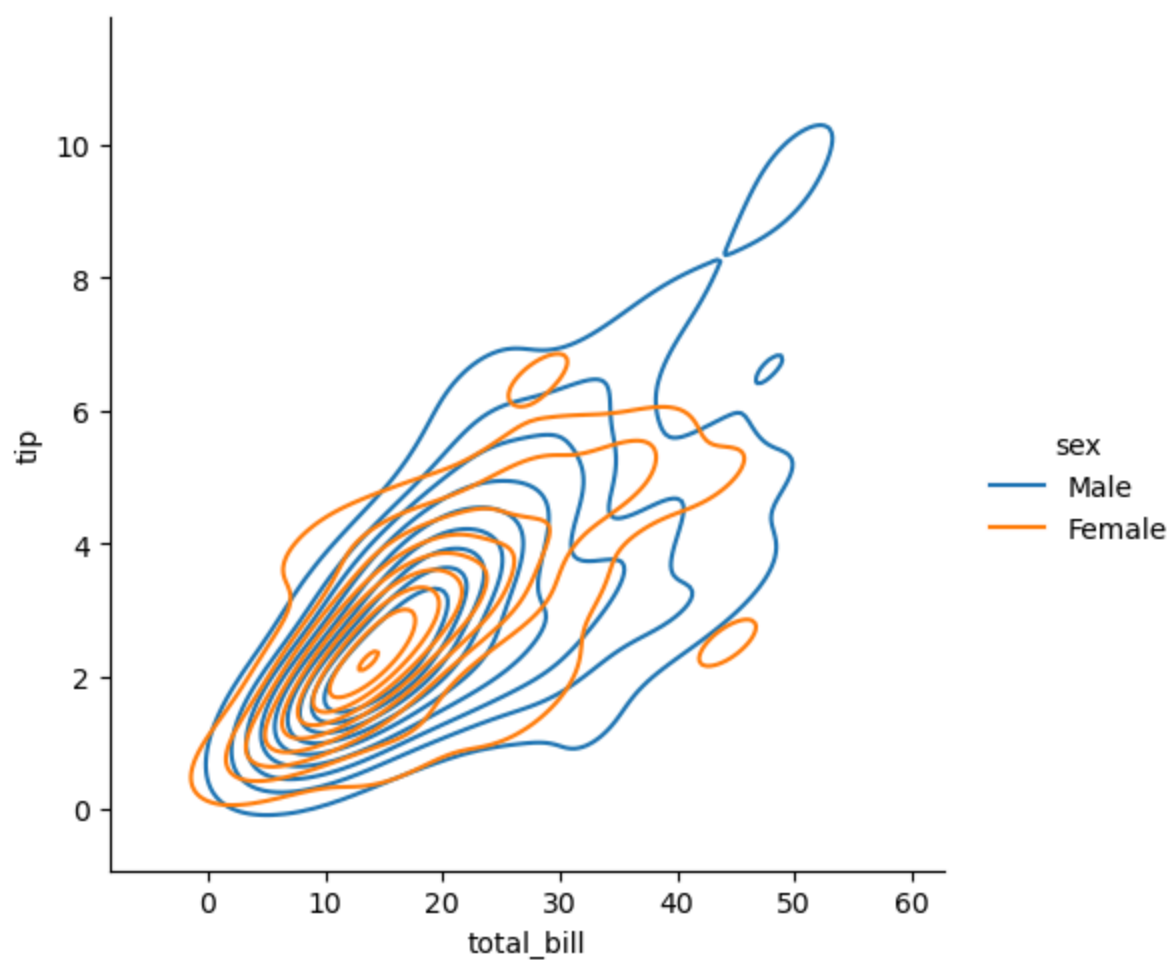
Esto me dice:

Tenemos la data y con `hue` me ayuda a hacer una segmentación o como lo voy a segmentar.

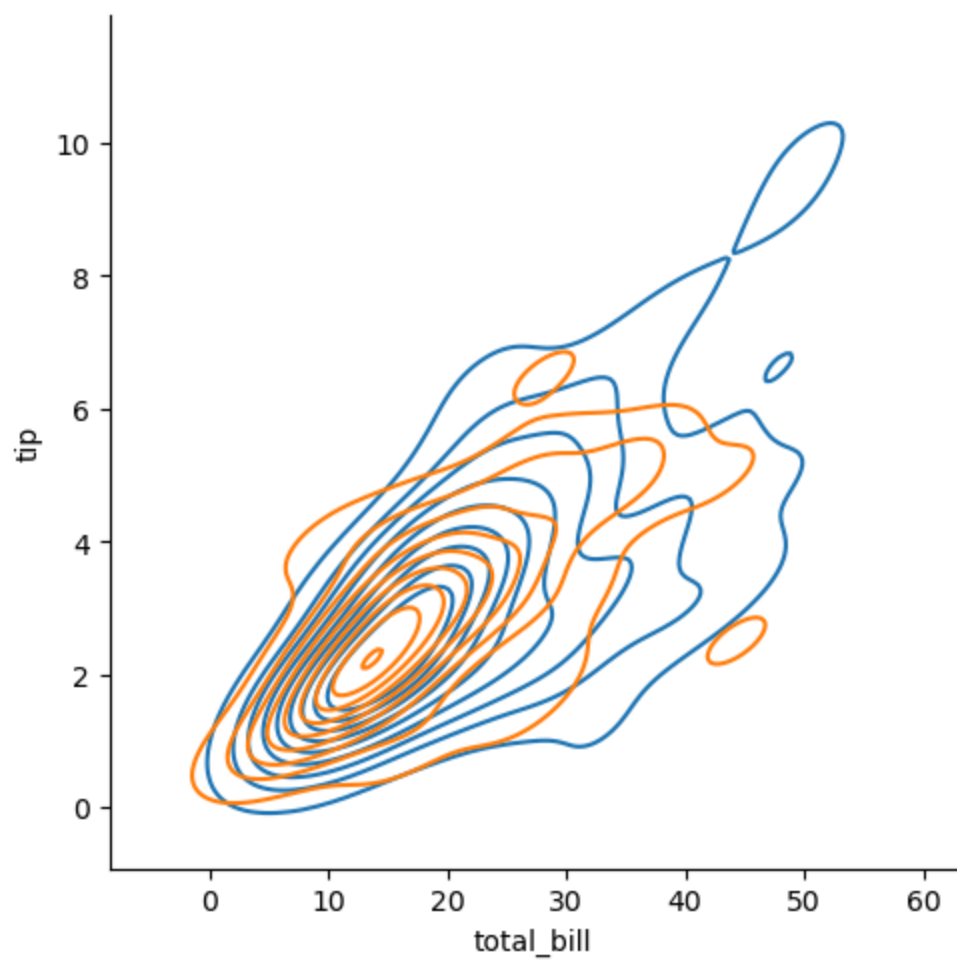
Kind

Me va a decir con que tipo de gráfico quiero trabajar

```
In [ ]: #Relacionando 2 variables
sns.displot(data=tips,x='total_bill',y='tip',hue='sex',kind='kde')
#Haciendo distinción con el parametro hue
plt.show()
```



```
In [ ]: #Relacionando 2 variables
sns.displot(data=tips,x='total_bill',y='tip',hue='sex',kind='kde',legend=False)
#Haciendo distinción con el parametro hue
plt.show()
```

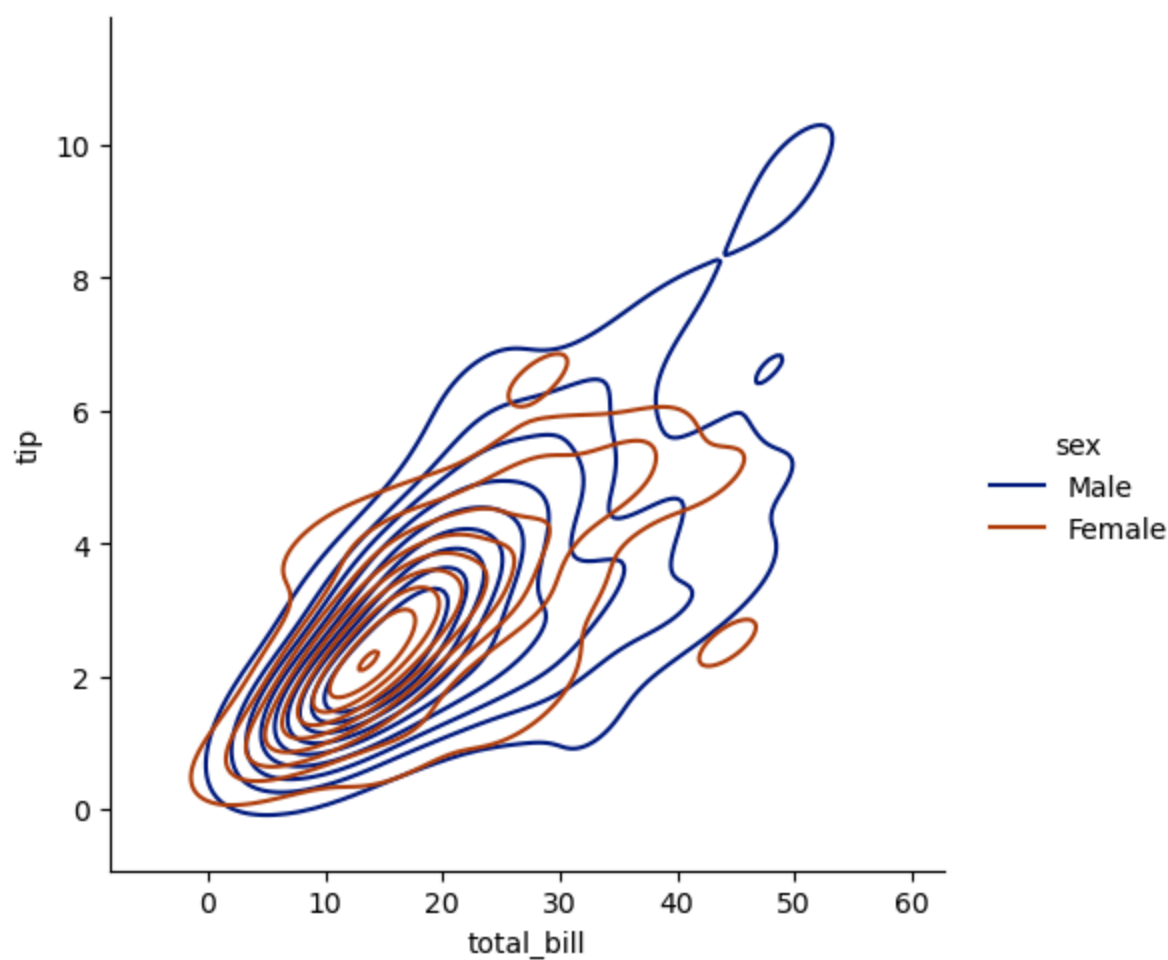


Sin leyendas

Funciona en todos los tipos de gráficas

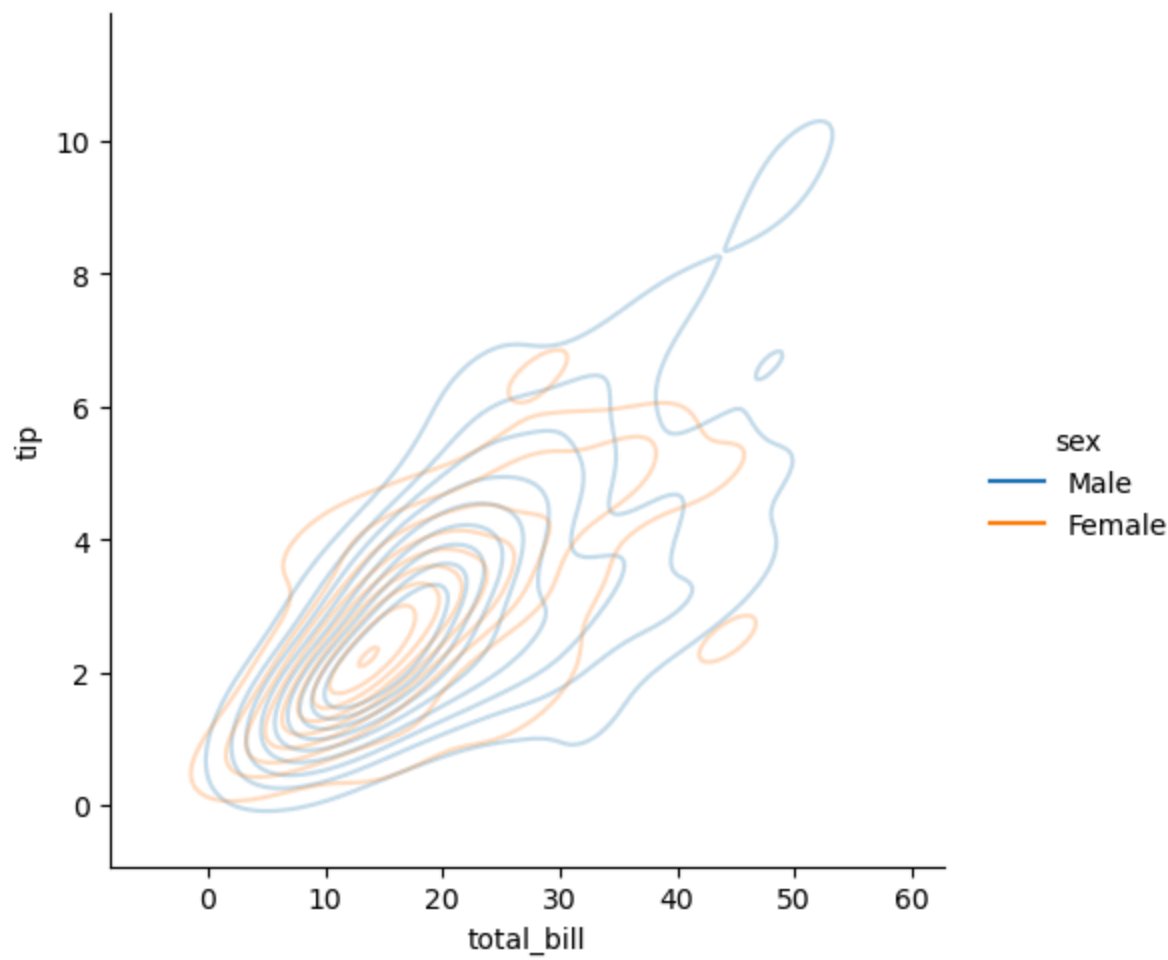
Paleta de colores

```
In [ ]: #Relacionando 2 variables
sns.displot(data=tips,x='total_bill',y='tip',hue='sex',kind='kde',palette='dark')
#Haciendo distinción con el parametro hue
plt.show()
```



Transparencia con alpha

```
In [ ]: #Relacionando 2 variables
sns.displot(data=tips,x='total_bill',y='tip',hue='sex',kind='kde',alpha=0.25)
#Haciendo distinción con el parametro hue
plt.show()
```



Dato extra:

Si desean que las imágenes salgan más nítidas, pueden aumentar el dpi (densidad de píxeles) usando la siguiente instrucción:

```
sns.set(rc={"figure.dpi":300})
```

 De preferencia la podrían colocar después de importar la librería

```
import seaborn as sns
import matplotlib.pyplot as plt
sns.set(rc={"figure.dpi":300})
```

Referencias:

- [Seaborn Load dataset](#)
- [Git Hub Seaborn data](#)
- [Examples Seaborn](#)