

Seaborn

January 21, 2021

1 Module 0 - Seaborn

A PDF version of this notebook is available at [Module 0 - Seaborn](#)

Seaborn is a Python data visualization library based on matplotlib. It provides a high-level interface for drawing attractive and informative statistical graphics. The usual alias for seaborn is `sns`.

```
[1]: import seaborn as sns
```

```
[4]: ## Let's import some other modules
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

```
[6]: # let's load a sample dataset from the Seaborn module
tips = sns.load_dataset('tips')
tips
```

```
[6]:
```

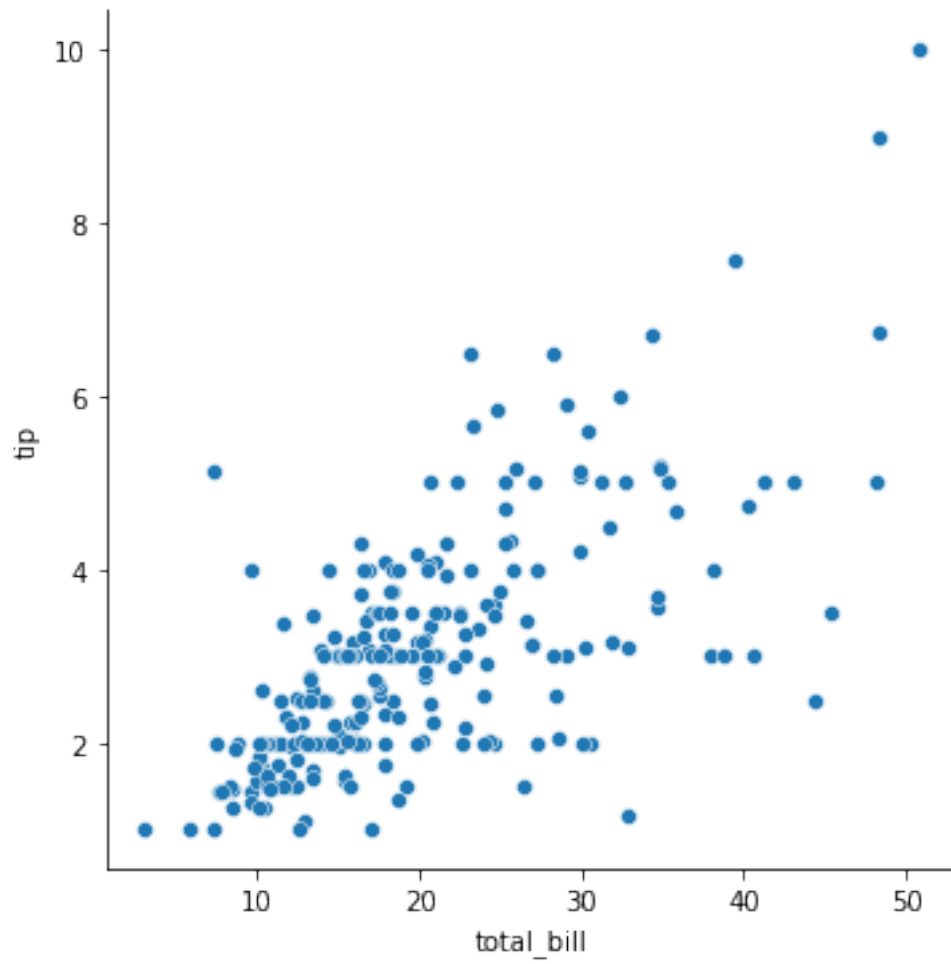
	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2
1	10.34	1.66	Male	No	Sun	Dinner	3
2	21.01	3.50	Male	No	Sun	Dinner	3
3	23.68	3.31	Male	No	Sun	Dinner	2
4	24.59	3.61	Female	No	Sun	Dinner	4
..
239	29.03	5.92	Male	No	Sat	Dinner	3
240	27.18	2.00	Female	Yes	Sat	Dinner	2
241	22.67	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]

1.0.1 Scatter Plot

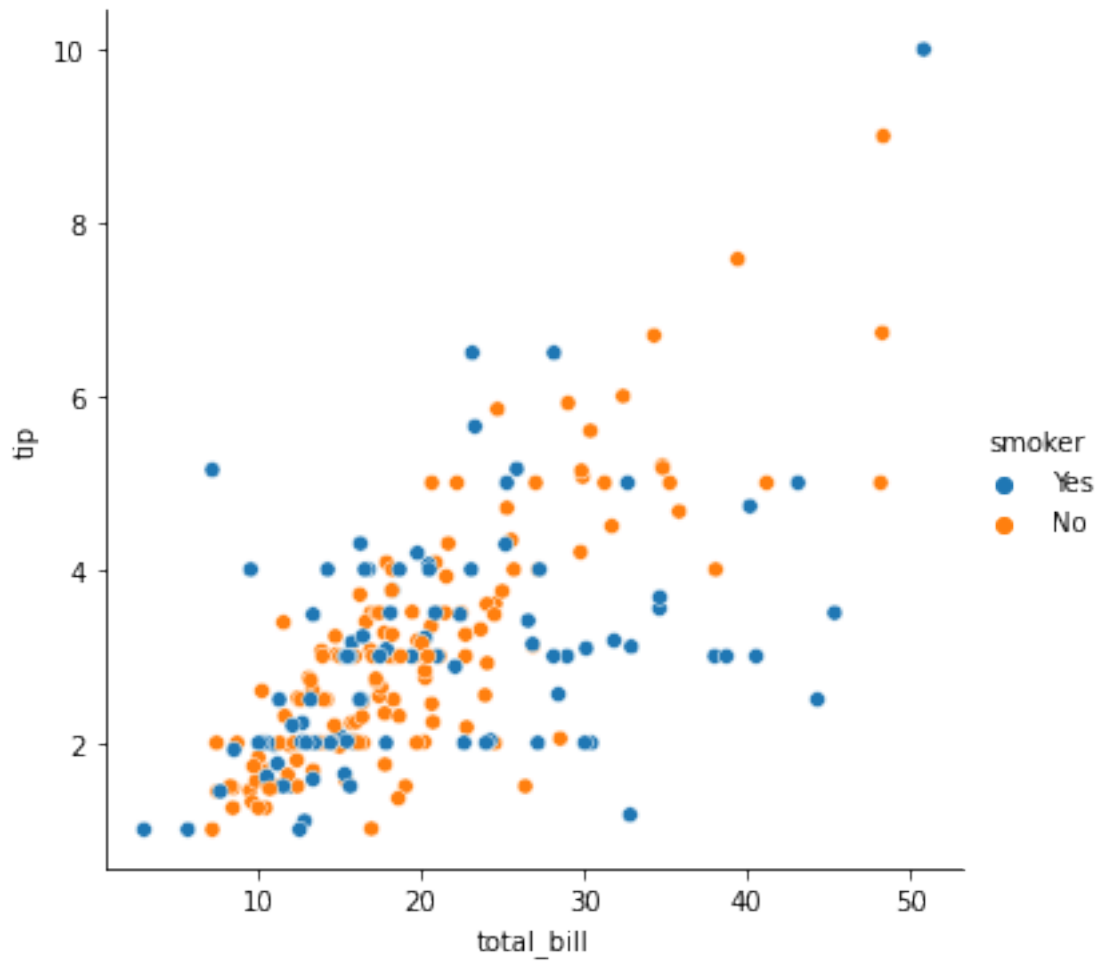
```
[7]: sns.relplot(x = 'total_bill', y = 'tip', data = tips)
```

```
[7]: <seaborn.axisgrid.FacetGrid at 0x7f682273cfd0>
```



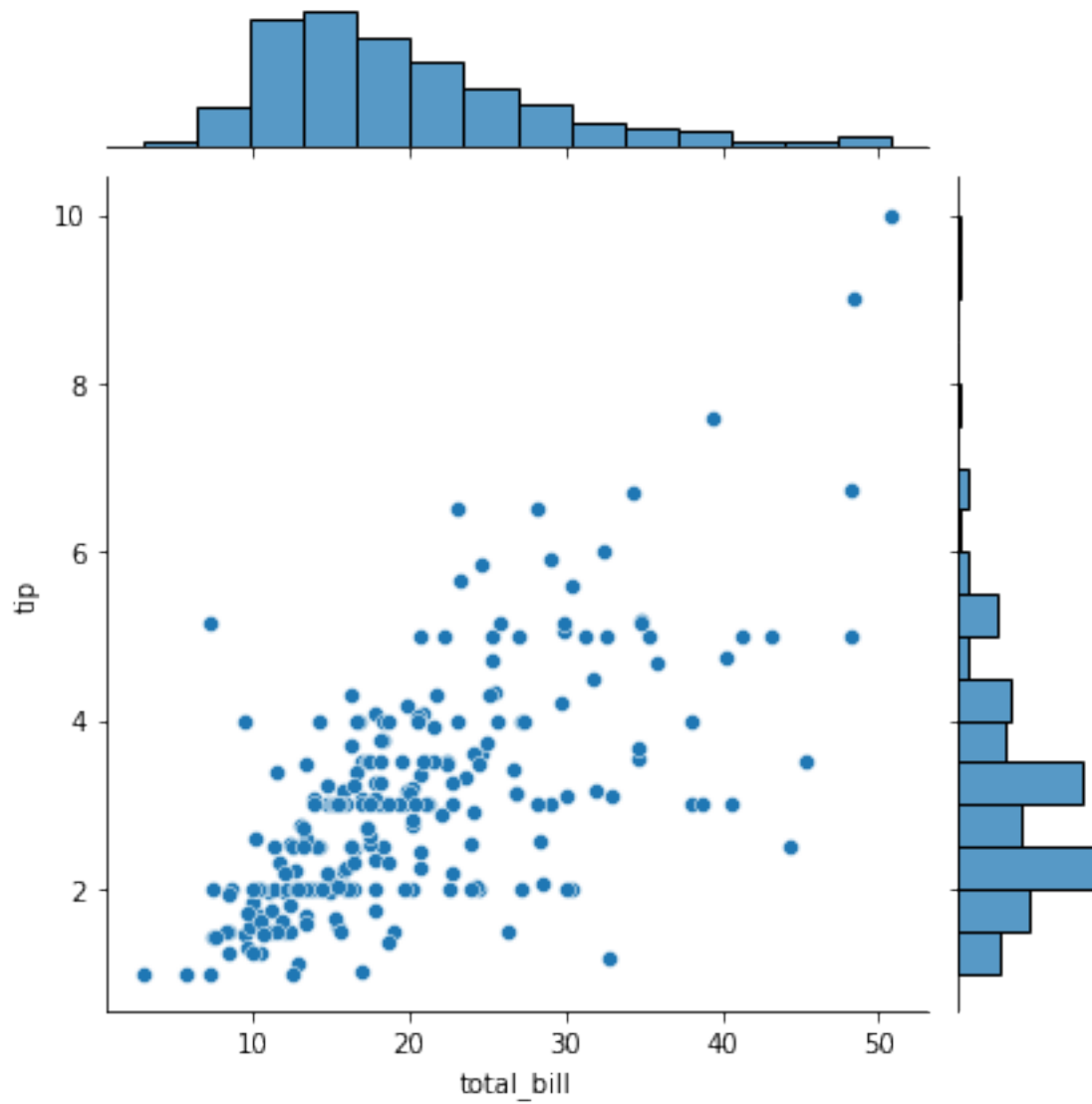
```
[8]: ## color by smoker  
sns.relplot(x = 'total_bill', y = 'tip', data = tips, hue = 'smoker')
```

```
[8]: <seaborn.axisgrid.FacetGrid at 0x7f6821874400>
```



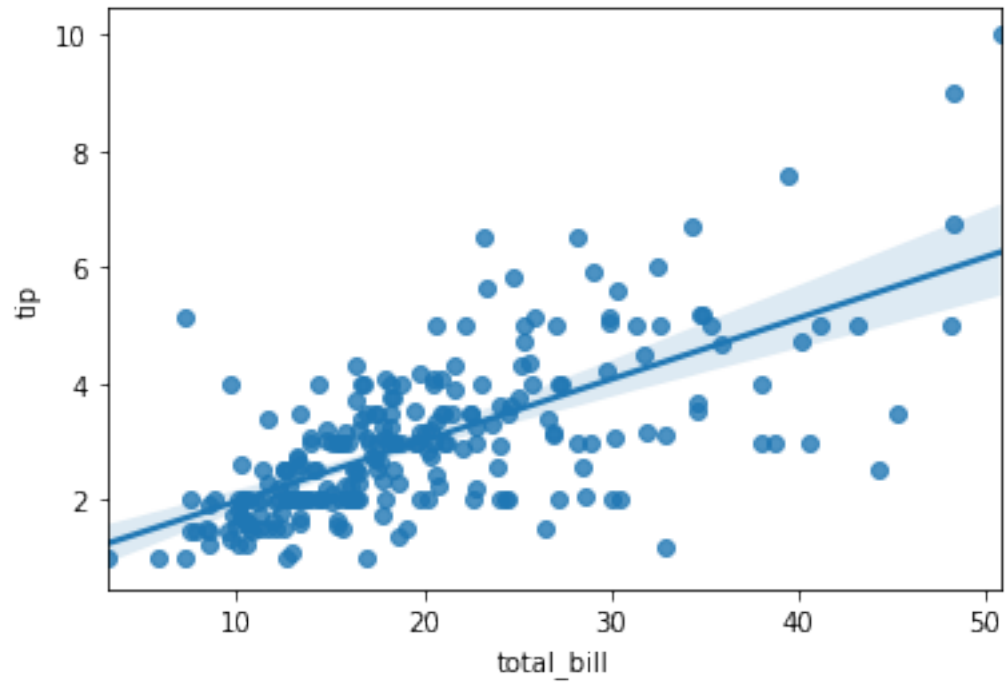
```
[23]: ## The jointplot includes histograms on the sides  
sns.jointplot(x = tips['total_bill'], y = tips['tip'])
```

```
[23]: <seaborn.axisgrid.JointGrid at 0x7f680f51e748>
```



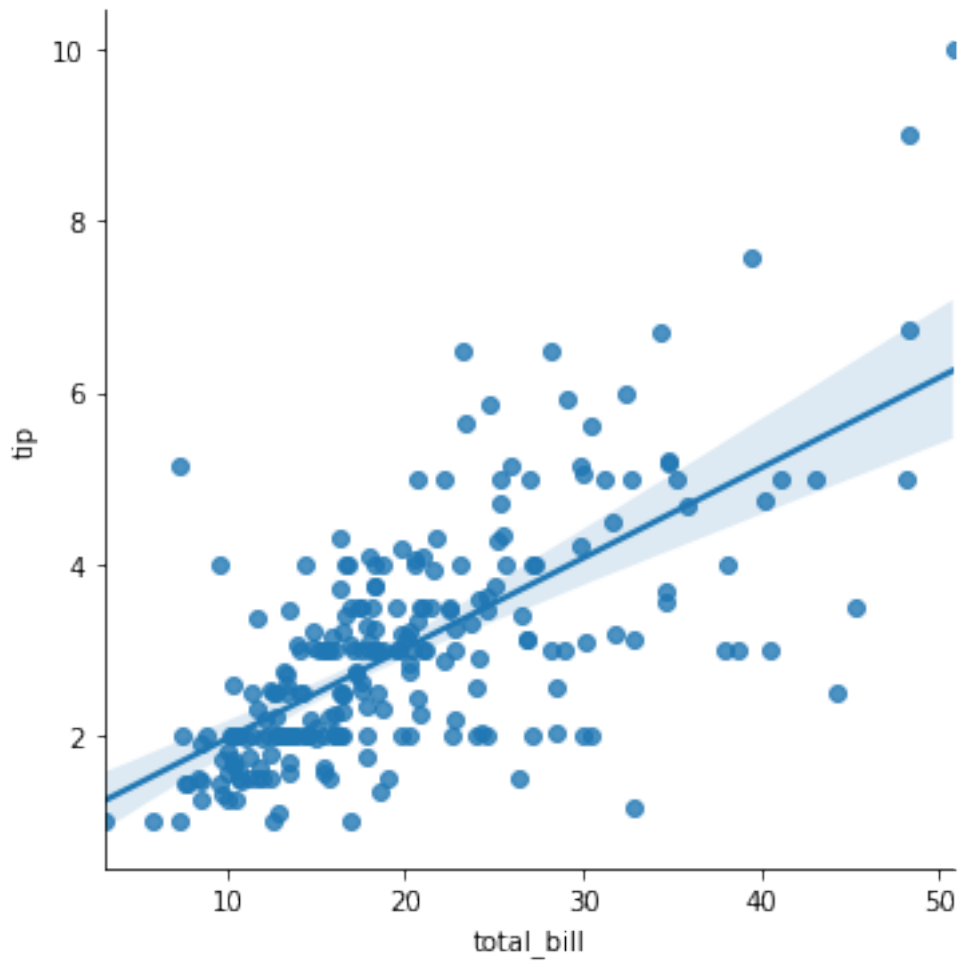
```
[27]: ## the regplot creates a regression line with the scatterplot  
sns.regplot(x = 'total_bill', y = 'tip', data = tips)
```

```
[27]: <matplotlib.axes._subplots.AxesSubplot at 0x7f680ec4b208>
```



```
[28]: ## the lmplot also does the same  
sns.lmplot(x = 'total_bill', y = 'tip', data = tips)
```

```
[28]: <seaborn.axisgrid.FacetGrid at 0x7f680ec33d30>
```



1.0.2 Line Plot

```
[12]: ## let's create a dataset. Make sure you understand the code.
df = pd.DataFrame(dict(time = np.arange(500), value = np.random.randn(500).
    ↪ cumsum()))
```

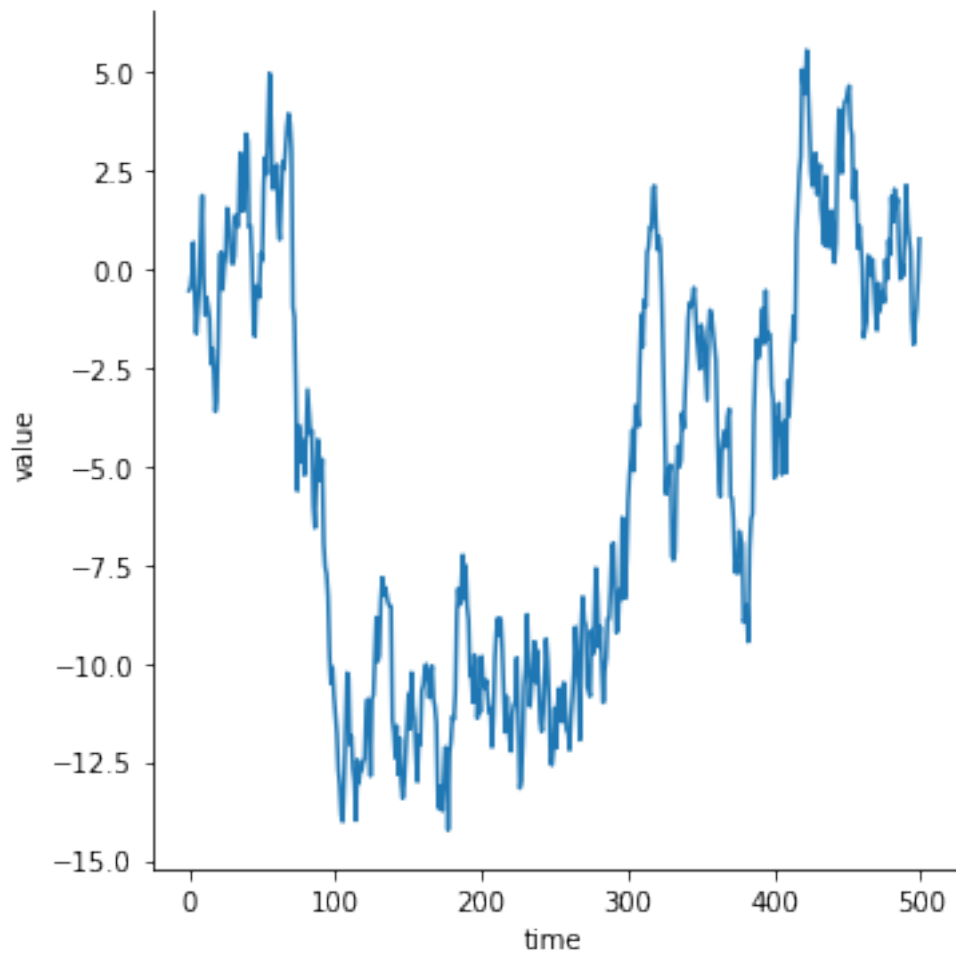
```
[13]: df.head(10)
```

```
[13]:   time    value
0     0 -0.534917
1     1 -0.461070
2     2 -0.115084
3     3  0.687616
4     4 -0.800890
5     5 -1.628512
6     6 -0.986690
7     7 -0.373504
```

```
8      8  0.927968
9      9  1.876880
```

```
[14]: sns.relplot(x = 'time', y = 'value', kind = 'line', data = df)
```

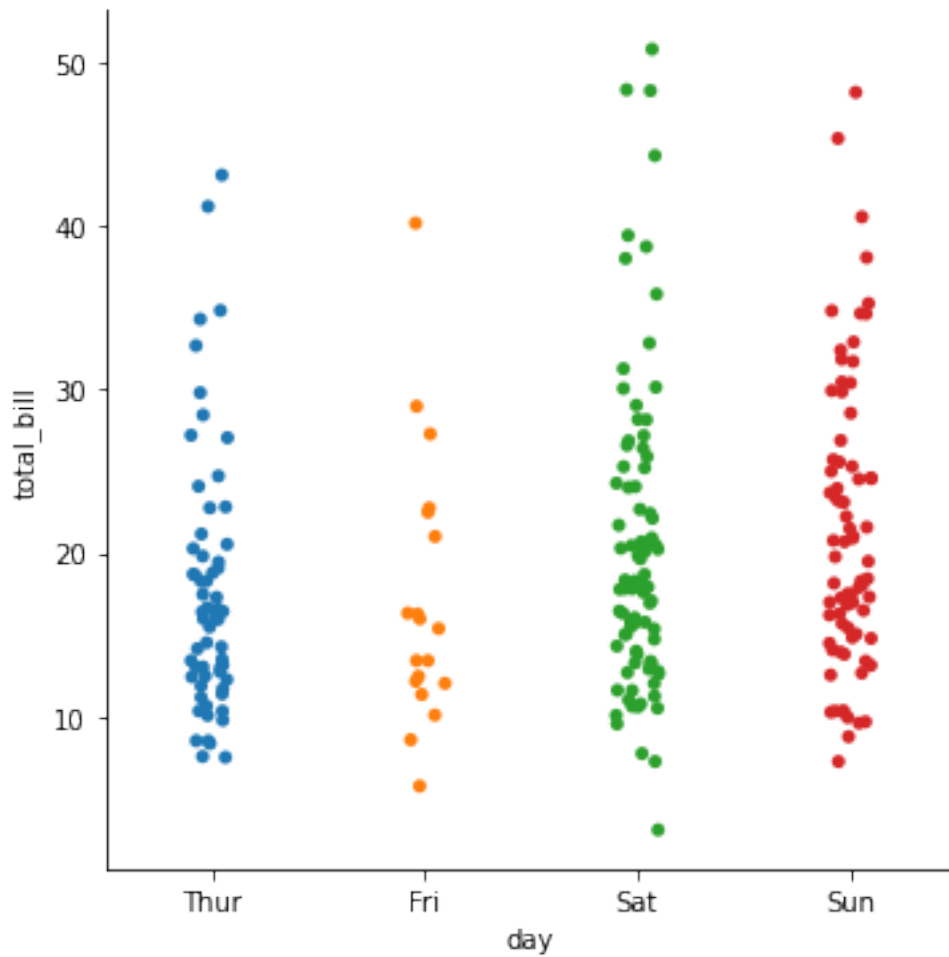
```
[14]: <seaborn.axisgrid.FacetGrid at 0x7f68218e9860>
```



1.0.3 Categorical Plots

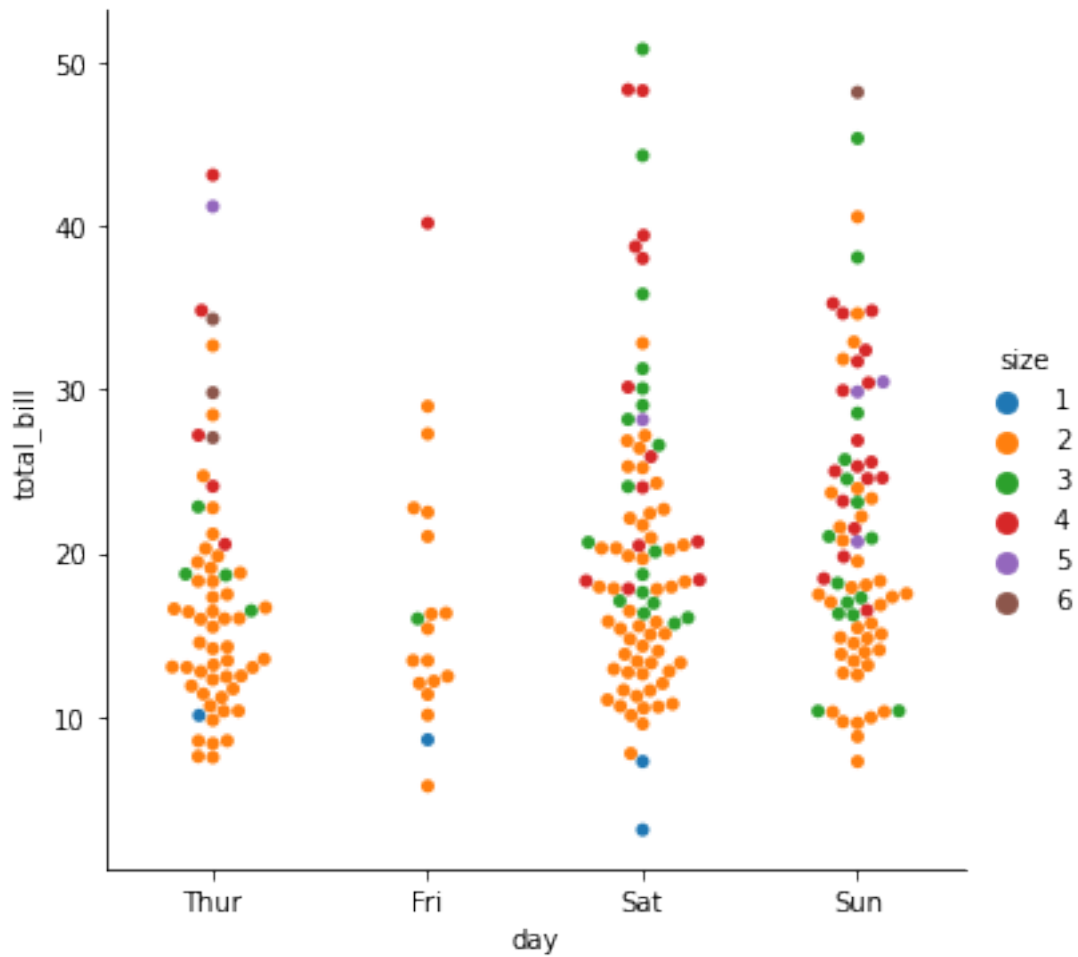
```
[15]: sns.catplot(x = 'day', y = 'total_bill', data = tips)
```

```
[15]: <seaborn.axisgrid.FacetGrid at 0x7f68176e9f60>
```



```
[17]: ## There are interesting variations using the kind and hue options
sns.catplot(x = 'day', y = 'total_bill', data = tips, kind = 'swarm', hue = 'day',
            size = 'total_bill')
```

```
[17]: <seaborn.axisgrid.FacetGrid at 0x7f681771dac8>
```

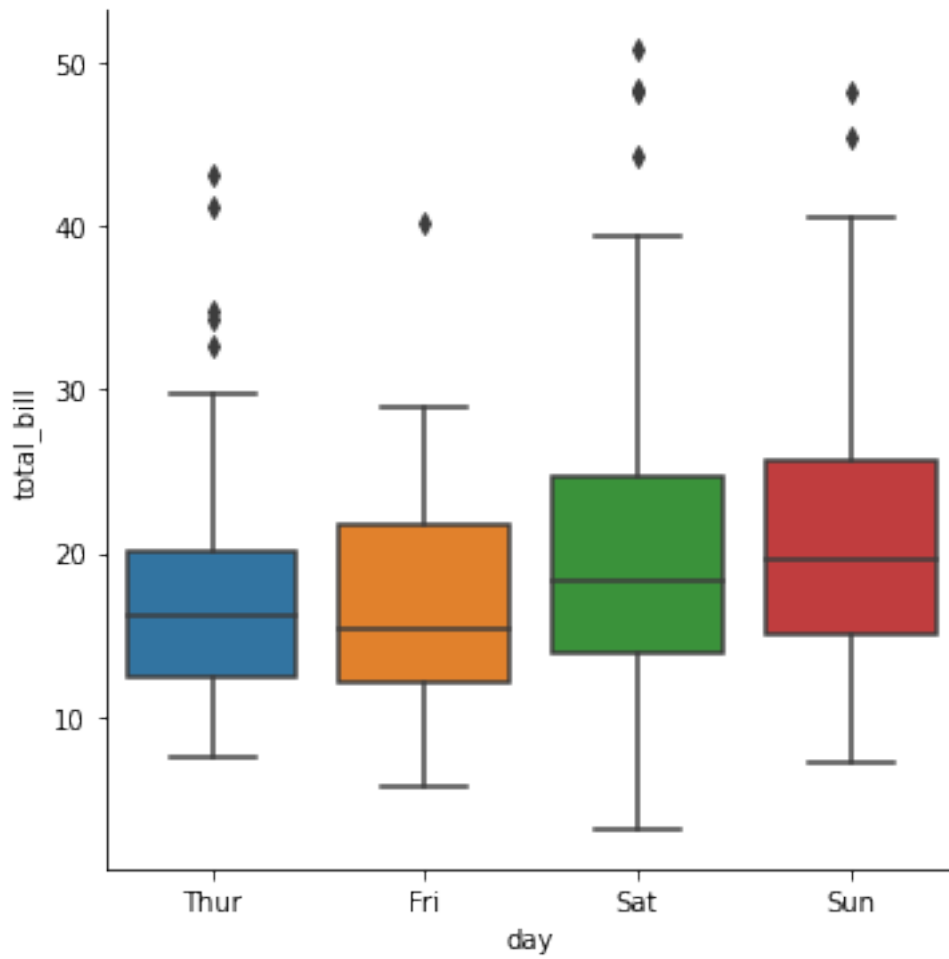



1.0.4 Box Plots

You can use the `kind = "box"` for box plots

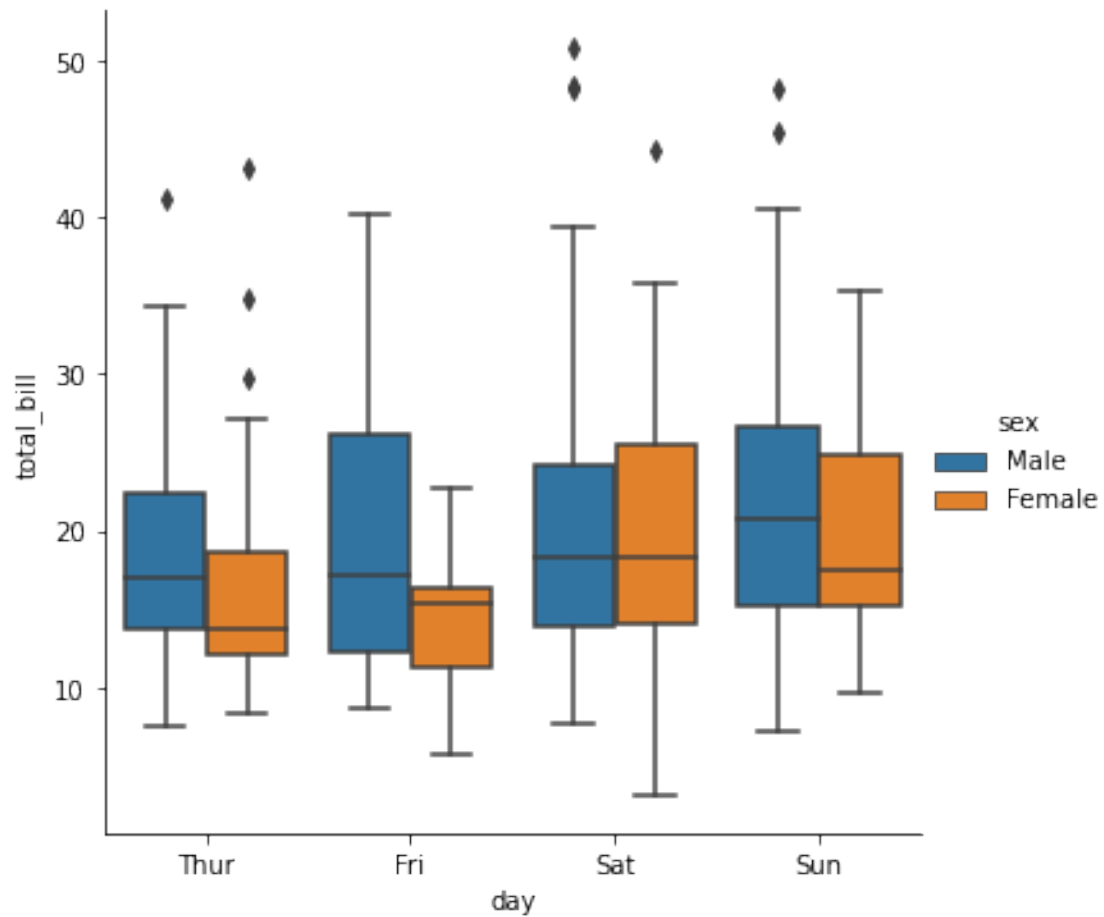
```
[21]: sns.catplot(x = 'day', y = 'total_bill', kind = 'box', data = tips)
```

```
[21]: <seaborn.axisgrid.FacetGrid at 0x7f680f737a90>
```



```
[22]: sns.catplot(x = 'day', y = 'total_bill', kind = 'box', data = tips, hue = 'sex')
```

```
[22]: <seaborn.axisgrid.FacetGrid at 0x7f680f663ba8>
```



1.0.5 Scatter Plot Matrix

```
[25]: ## You can pass the whole dataset to the pairplot function and it will create a
      ↪ scatterplot matrix of the numeric variables
      sns.pairplot(tips)
```

```
[25]: <seaborn.axisgrid.PairGrid at 0x7f680f039a90>
```

