

Module 02

Categorical Plots

Data Science Developer

Categorical Data Plots

Now let's discuss using seaborn to plot categorical data! There are a few main plot types for this:

- factorplot
- boxplot
- violinplot
- stripplot
- swarmplot
- barplot
- countplot

Let's go through examples of each!

Imports

```
import seaborn as sns  
%matplotlib inline
```

Data

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

```
tips.head()
```

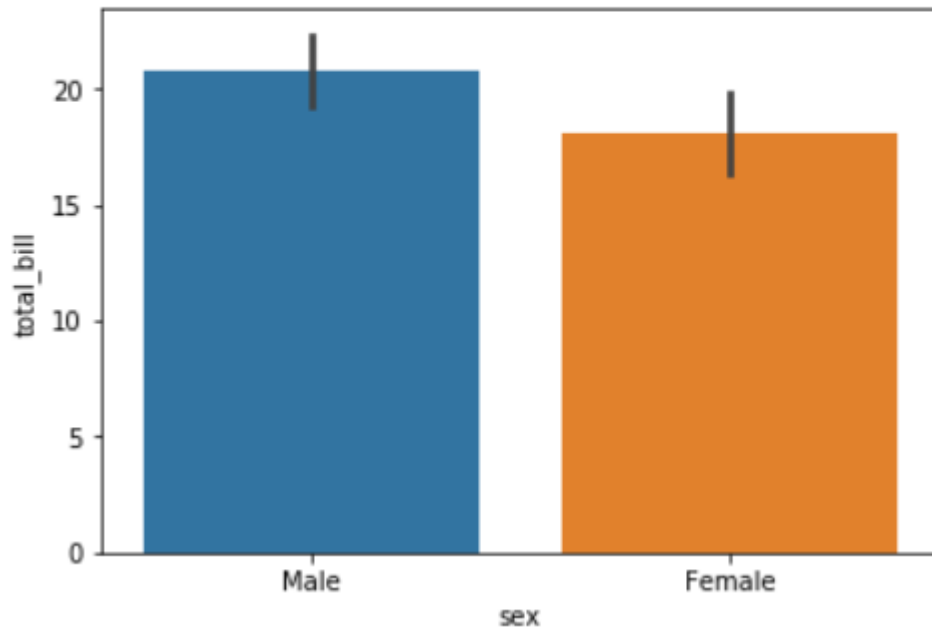
	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2
1	10.34	1.66	Male	No	Sun	Dinner	3
2	21.01	3.50	Male	No	Sun	Dinner	3
3	23.68	3.31	Male	No	Sun	Dinner	2
4	24.59	3.61	Female	No	Sun	Dinner	4

barplot

barplot is a general plot that allows you to aggregate the categorical data based off some function, by default the mean:

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

```
<matplotlib.axes._subplots.AxesSubplot at 0x1cceed8bd68>
```

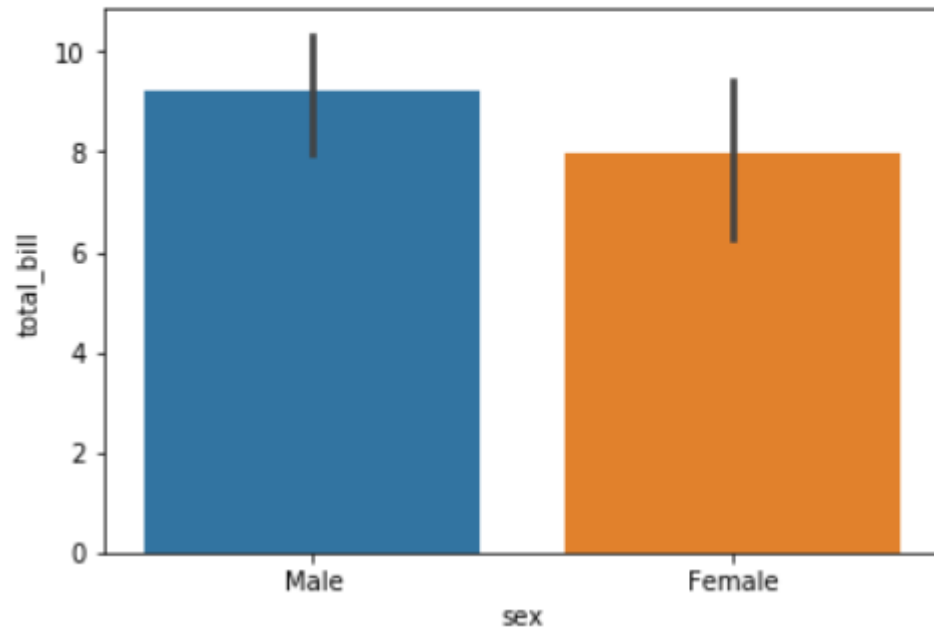


You can change the estimator object to your own function, that converts a vector to a scalar:

```
import numpy as np
```

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

```
<matplotlib.axes._subplots.AxesSubplot at 0x1cceb007860>
```

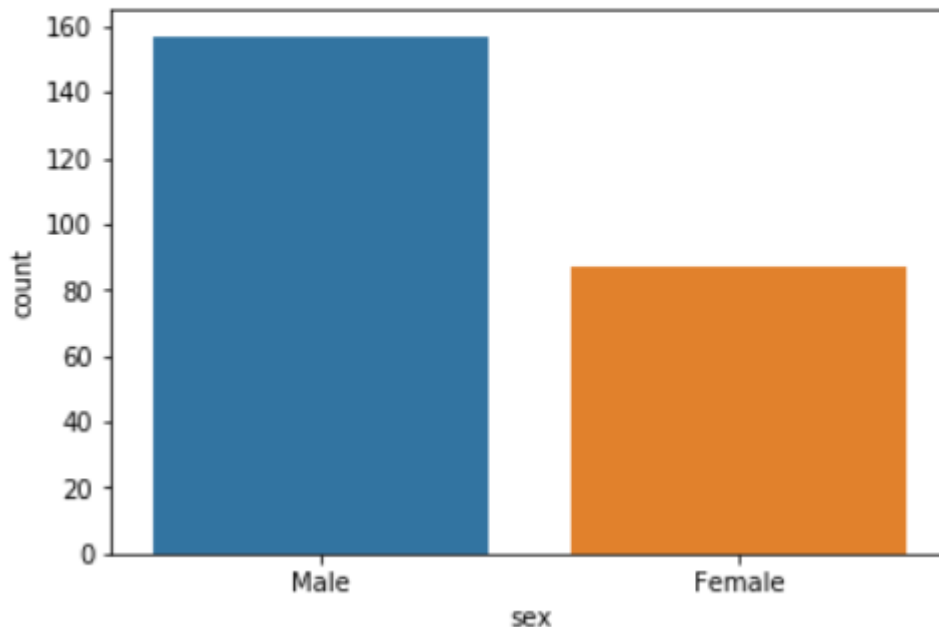


countplot

This is essentially the same as barplot except the estimator is explicitly counting the number of occurrences. Which is why we only pass the x value:

```
sns.countplot(x='sex',data=tips)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x1cceb068dd8>
```

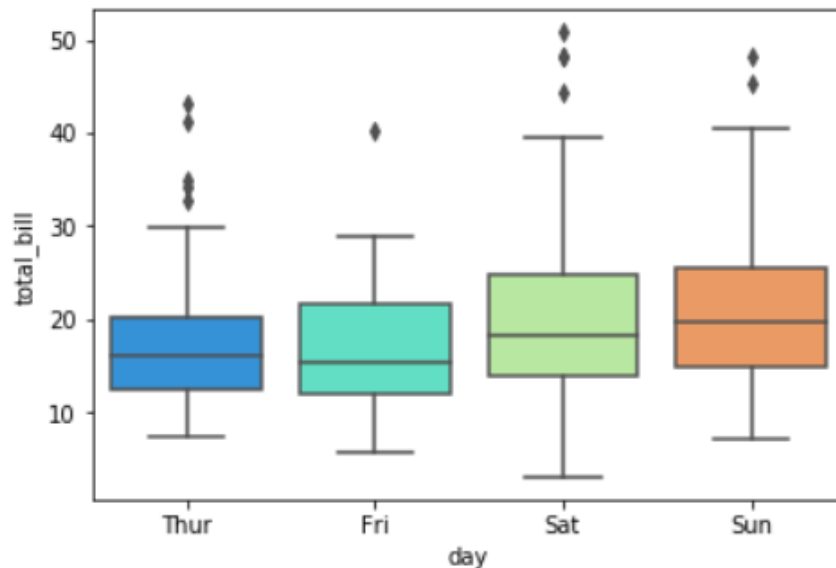


boxplot

A box plot (or box-and-whisker plot) shows the distribution of quantitative data in a way that facilitates comparisons between variables or across levels of a categorical variable. The box shows the quartiles of the dataset while the whiskers extend to show the rest of the distribution, except for points that are determined to be “outliers” using a method that is a function of the inter-quartile range.

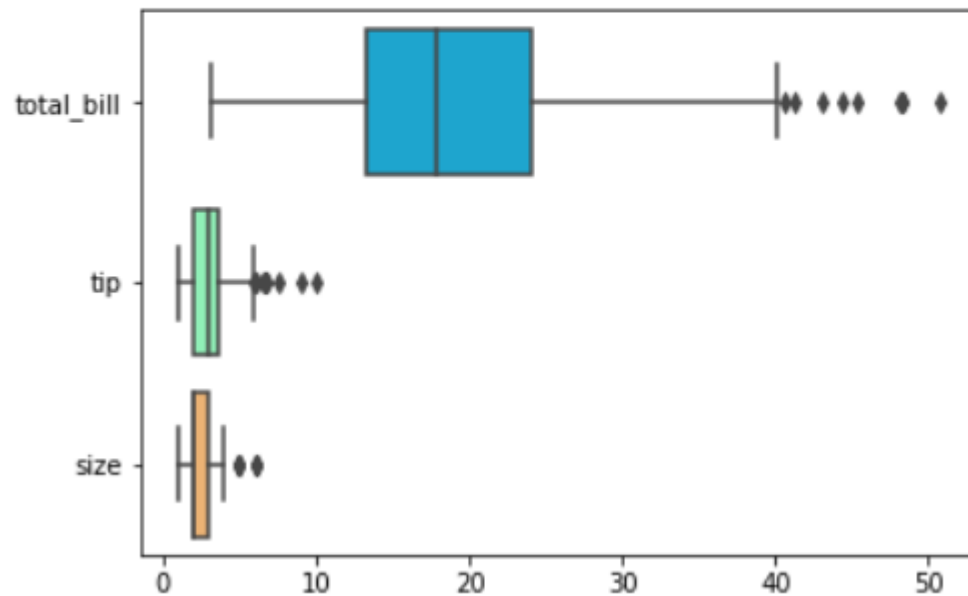
```
sns.boxplot(x="day", y="total_bill", data=tips,palette='rainbow')
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x1ccec0886d8>
```



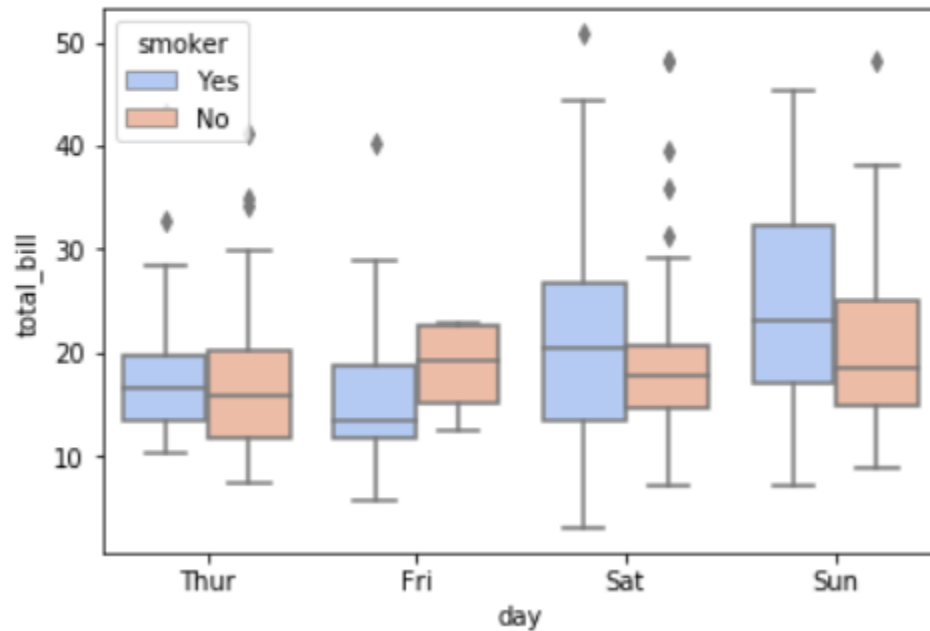

```
# Can do entire dataframe with orient='h'  
sns.boxplot(data=tips,palette='rainbow',orient='h')
```

<matplotlib.axes._subplots.AxesSubplot at 0x1ccec106128>



```
sns.boxplot(x="day", y="total_bill", hue="smoker", data=tips, palette="coolwarm")
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x1ccec1a2828>
```

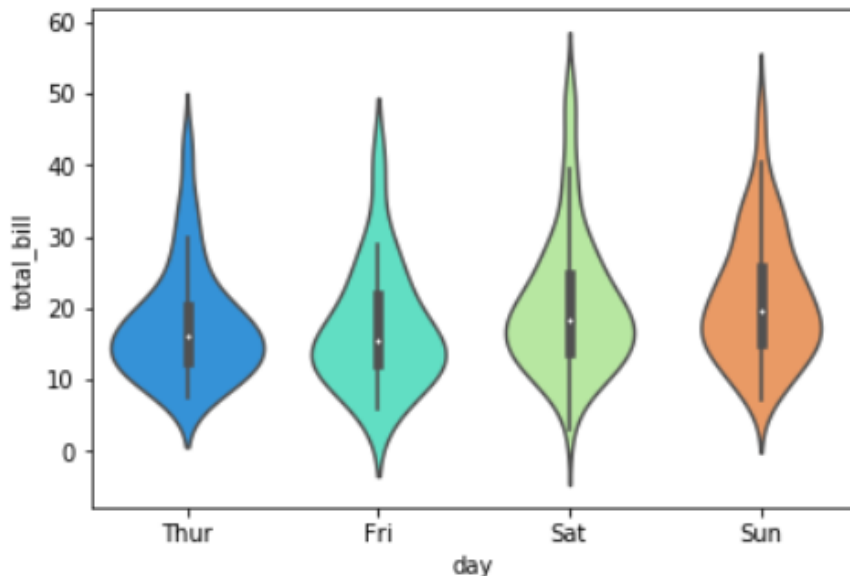


violinplot

A violin plot plays a similar role as a box and whisker plot. It shows the distribution of quantitative data across several levels of one (or more) categorical variables such that those distributions can be compared. Unlike a box plot, in which all of the plot components correspond to actual datapoints, the violin plot features a kernel density estimation of the underlying distribution.

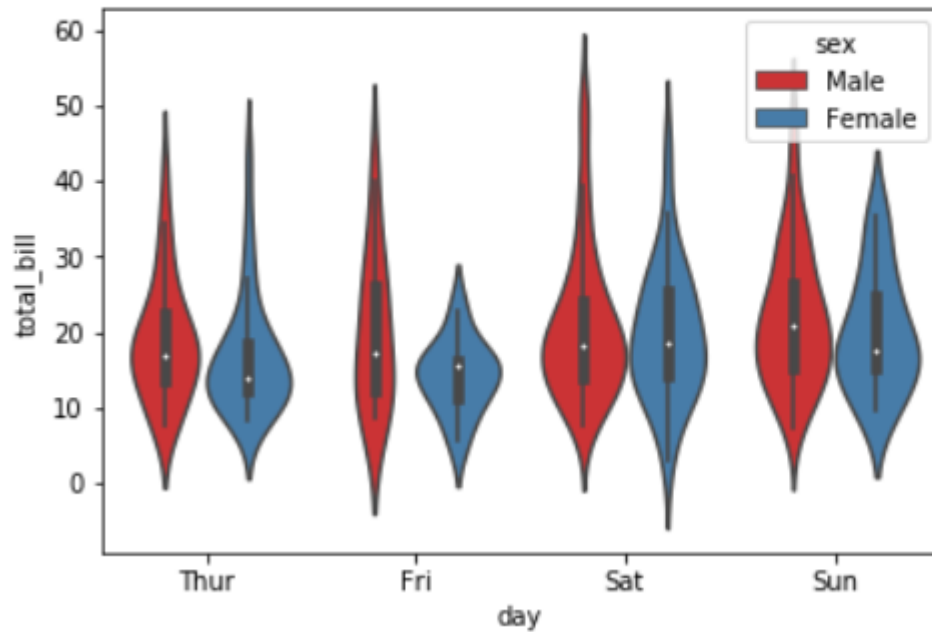
```
sns.violinplot(x="day", y="total_bill", data=tips,palette='rainbow')
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x1ccec267d68>
```



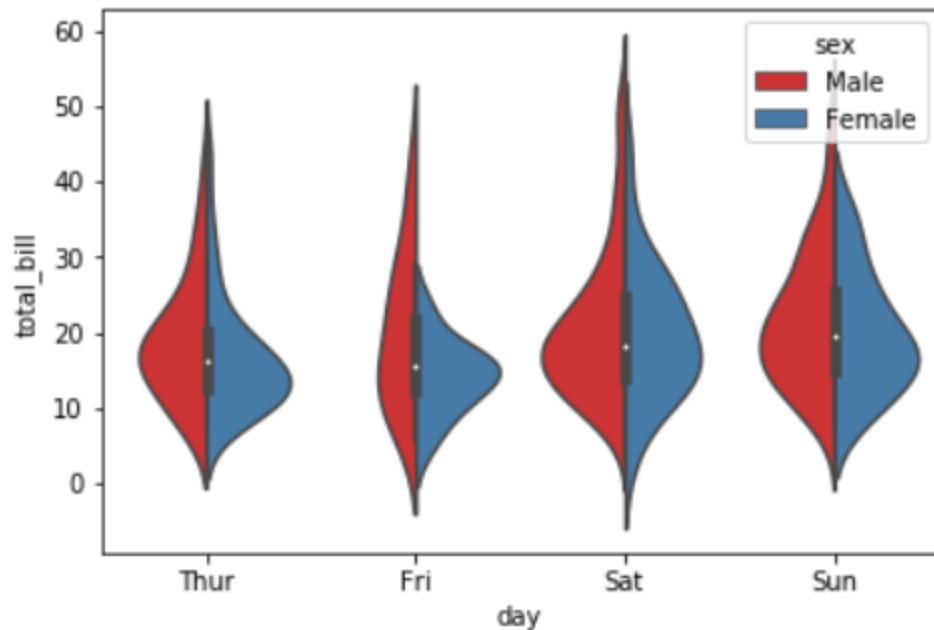
```
sns.violinplot(x="day", y="total_bill", data=tips, hue='sex', palette='Set1')
```

<matplotlib.axes._subplots.AxesSubplot at 0x1ccec2ff2b0>



```
sns.violinplot(x="day", y="total_bill", data=tips, hue='sex', split=True, palette='Set1')
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x1ccec3a4f60>
```

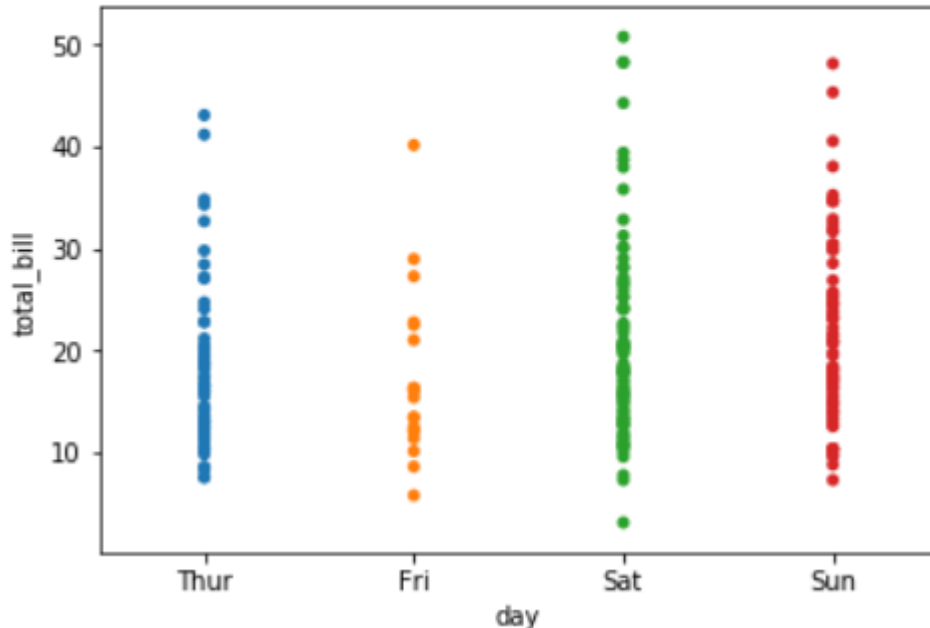


stripplot

The stripplot will draw a scatterplot where one variable is categorical. A strip plot can be drawn on its own, but it is also a good complement to a box or violin plot in cases where you want to show all observations along with some representation of the underlying distribution.

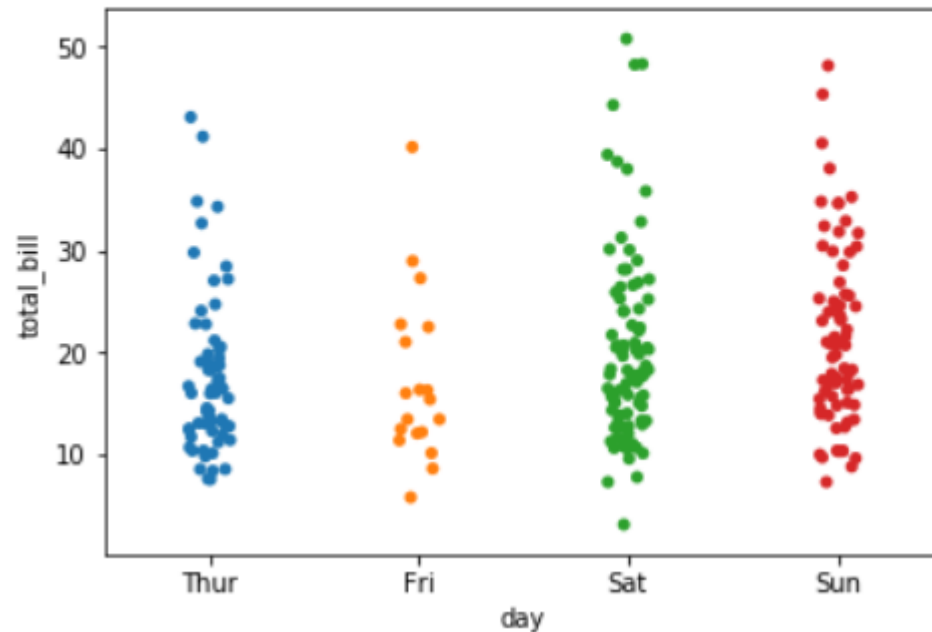
```
sns.stripplot(x="day", y="total_bill", data=tips)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x1ccec4139e8>
```

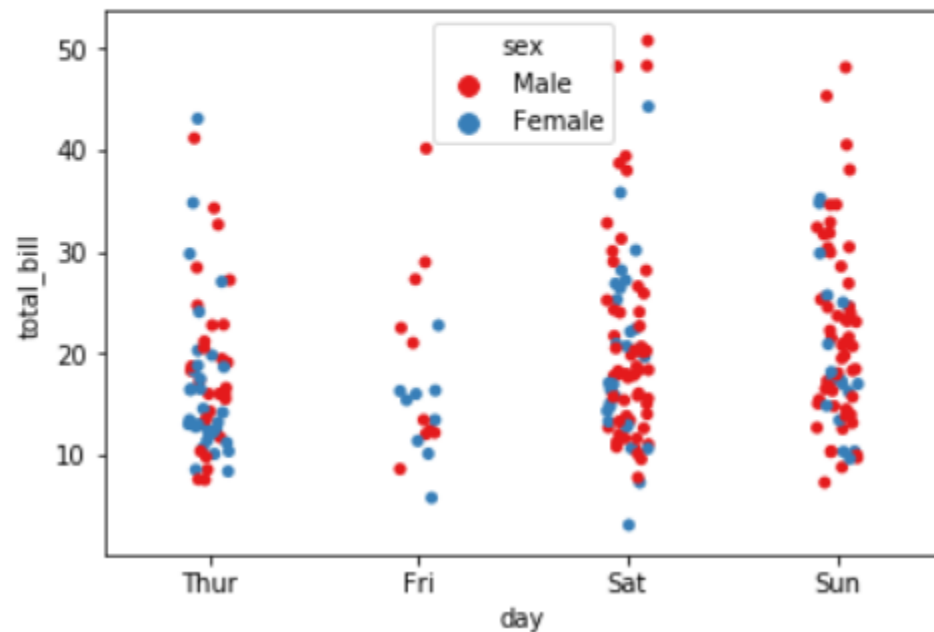


```
sns.stripplot(x="day", y="total_bill", data=tips, jitter=True)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x1ccec485390>
```

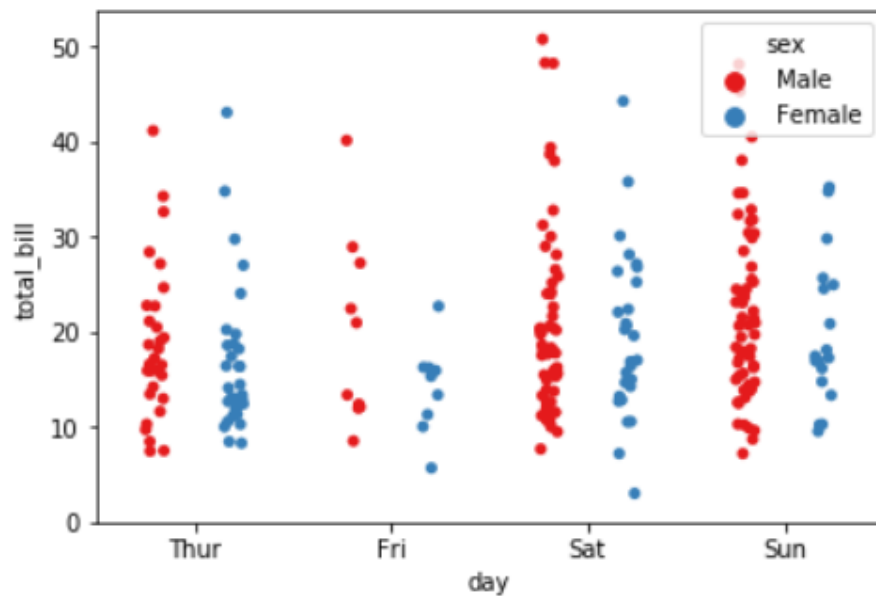


```
sns.stripplot(x="day", y="total_bill", data=tips, jitter=True, hue='sex', palette='Set1')  
<matplotlib.axes._subplots.AxesSubplot at 0x1ccec4d5f98>
```




```
sns.stripplot(x="day", y="total_bill", data=tips, jitter=True, hue='sex', palette='Set1', dodge=True)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x1ccec648a90>
```

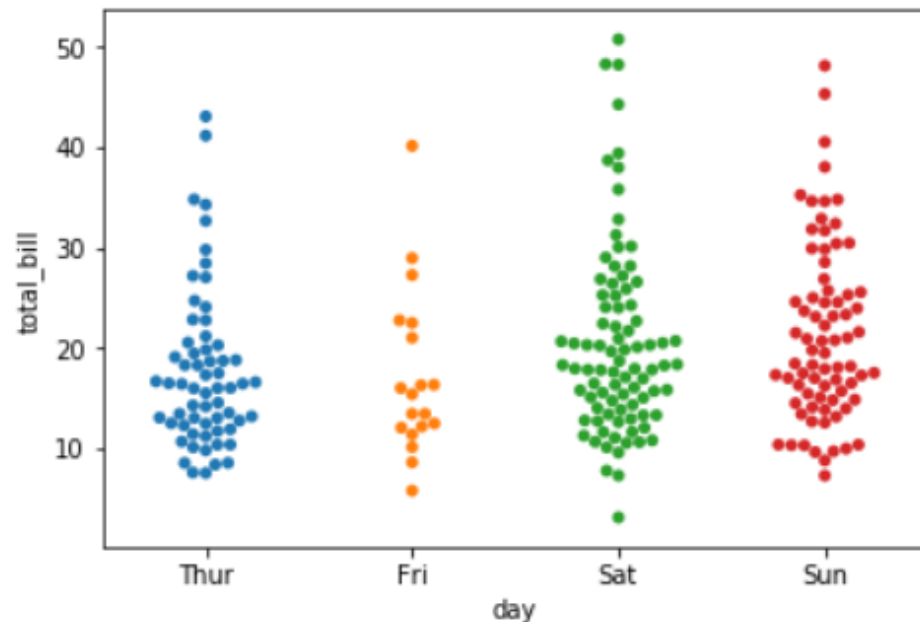


swarmplot

The swarmplot is similar to stripplot(), but the points are adjusted (only along the categorical axis) so that they don't overlap. This gives a better representation of the distribution of values, although it does not scale as well to large numbers of observations (both in terms of the ability to show all the points and in terms of the computation needed to arrange them).

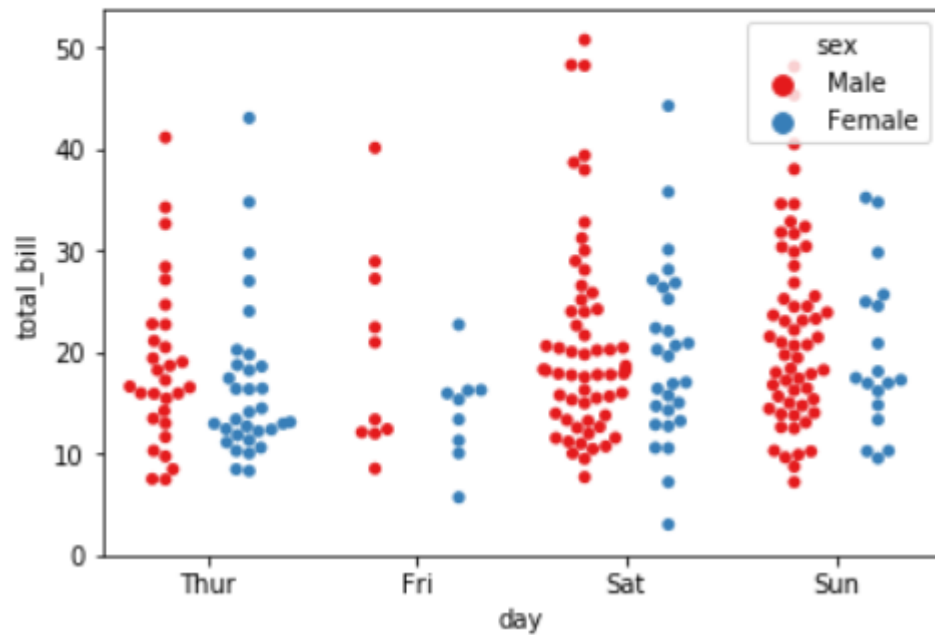
```
sns.swarmplot(x="day", y="total_bill", data=tips)
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x1ccec715cc0>
```



```
sns.swarmplot(x="day", y="total_bill", hue='sex', data=tips, palette="Set1", dodge=True)
```

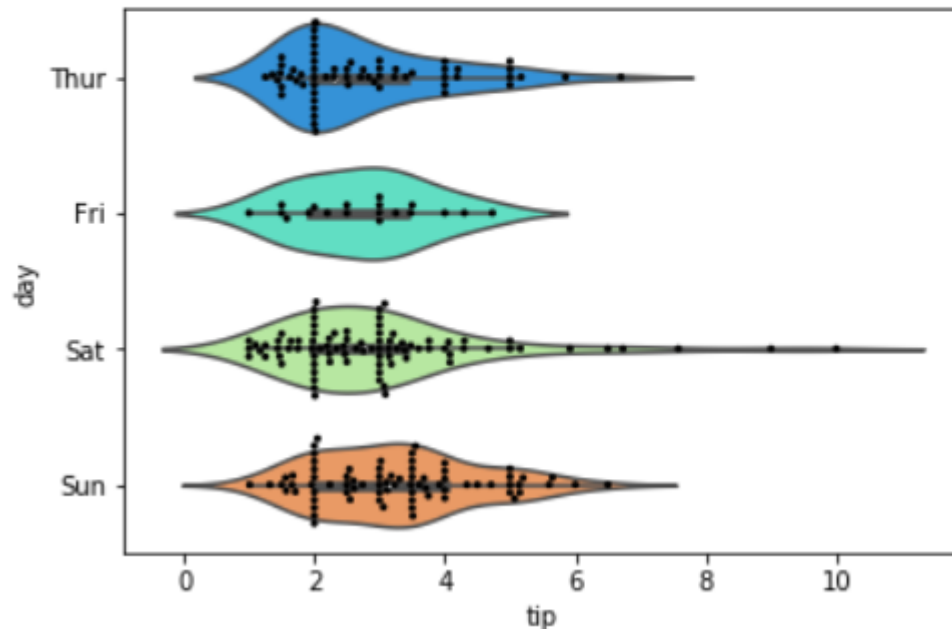
<matplotlib.axes._subplots.AxesSubplot at 0x1ccec7dab38>



Combining Categorical Plot

```
sns.violinplot(x="tip", y="day", data=tips,palette='rainbow')  
sns.swarmplot(x="tip", y="day", data=tips,color='black',size=3)
```

<matplotlib.axes._subplots.AxesSubplot at 0x1ccec83d320>



factorplot

factorplot is the most general form of a categorical plot. It can take in a kind parameter to adjust the plot type:

```
sns.factorplot(x='sex',y='total_bill',data=tips,kind='bar')
```

```
<seaborn.axisgrid.FacetGrid at 0x1ccec765390>
```

