KEY_Lesson20_BarCharts_Histograms

May 26, 2020

1 Bar Charts and Histograms

1.1 Bar Charts

Bar charts are used to display how a *categorical* variable relates to a *continuous* variable. In bar charts the *categorical* varibale is displayed on the x-axis and the *continuous* variable is displayed on the y-axis.

- Categorical variables are variables with different categories or groups.
  - Examples: gender, city
- Continuous variables are numeric variables.
  - Examples: time, height, length

[1]:

```
# import seaborn, matplotlib
import seaborn as sns
import matplotlib.pyplot as plt
# set up inline figures
%matplotlib inline
```

We will be using the titanic dataset in this example. Let's load and preview it.

[2]:

```
# read in titanic data
titanic = sns.load_dataset("titanic")
# preview data
titanic.head()
```

[2]:

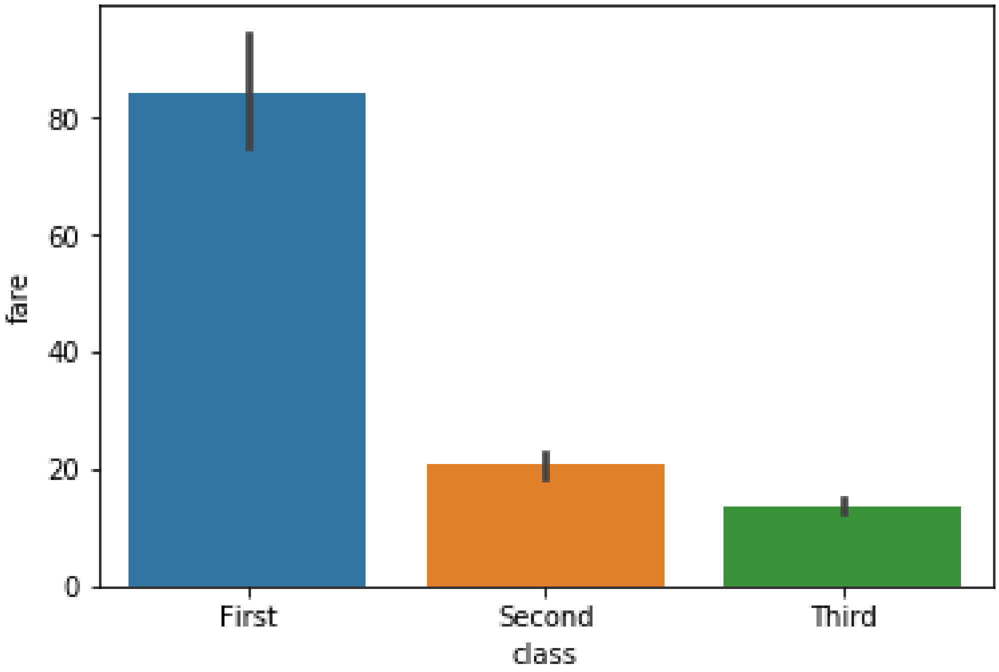
```
   survived  pclass     sex   age  sibsp  parch     fare embarked  class  \
0         0       3    male  22.0      1      0   7.2500        S  Third
1         1       1  female  38.0      1      0  71.2833        C  First
2         1       3  female  26.0      0      0   7.9250        S  Third
3         1       1  female  35.0      1      0  53.1000        S  First
4         0       3    male  35.0      0      0   8.0500        S  Third

     who  adult_male deck  embark_town alive  alone
0    man        True  NaN  Southampton    no  False
1  woman       False    C    Cherbourg   yes  False
2  woman       False  NaN  Southampton   yes   True
3  woman       False    C  Southampton   yes  False
```

```
4    man        True  NaN  Southampton    no   True
```

Let's say we want to compare the mean fare price across the three classes of tickets for all passengers.

```
[3]: # barplot of class vs fare
     sns.barplot(x="class", y = 'fare', data=titanic)
```

```
[3]: <matplotlib.axes._subplots.AxesSubplot at 0x115cb1780>
```

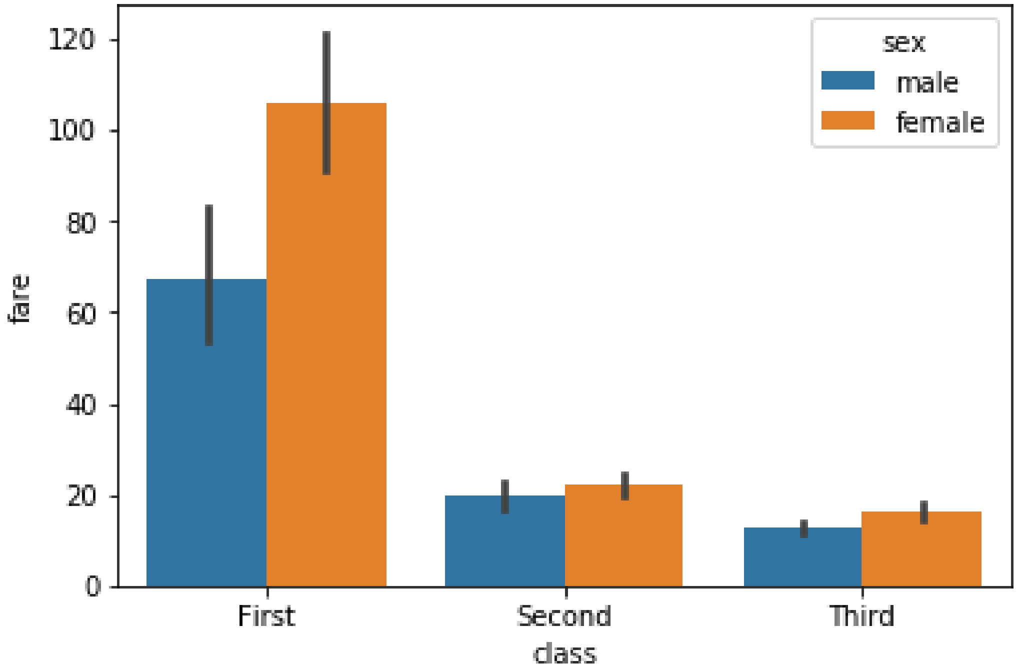


Notice how `seaborn` magically computes the mean fares and generates the plot exactly as we want without us even specifying!

What if we wanted to look at the data more granularly and further *stratify* each `class` bar by the `sex` variable? Based on what you know about `seaborn` so far, how do you think we can do that?

```
[4]: # barplot of class vs fare stratified by sex
     sns.barplot(x="class", y = 'fare', hue = "sex", data=titanic)
```

```
[4]: <matplotlib.axes._subplots.AxesSubplot at 0x1192bbba8>
```



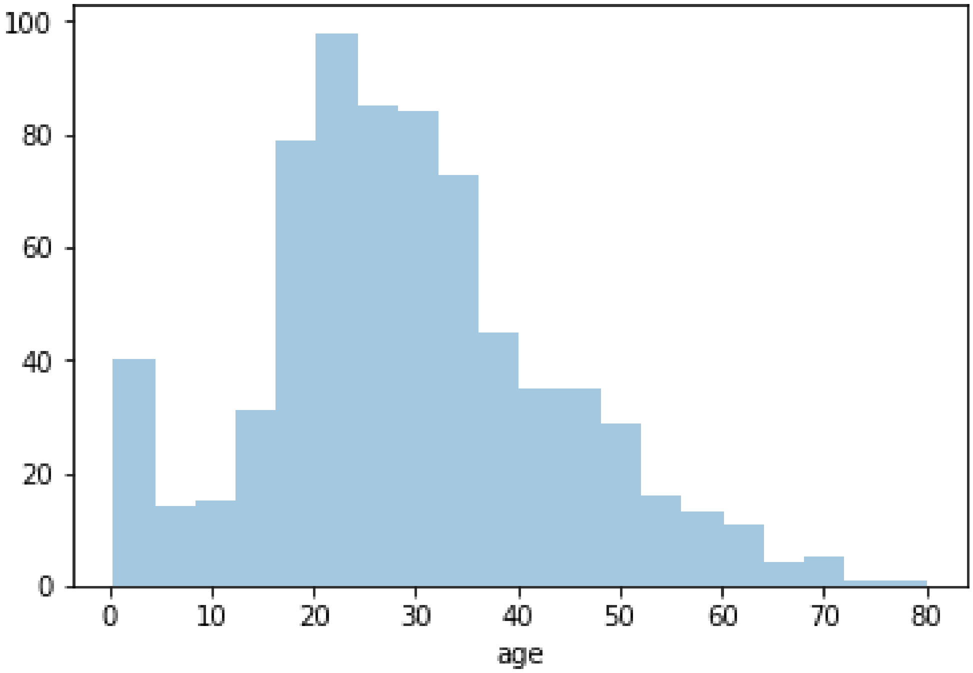
1.2 Histograms

Histograms are used to visualize the *distribution* of a *continuous* variable.

Let's say we wanted to see how the `age` was distributed across all passengers in our dataset. We can use the `distplot` function to generate our histogram.

[5]:
```
# histogram of age
sns.distplot(titanic['age'].dropna(), kde=False)
```

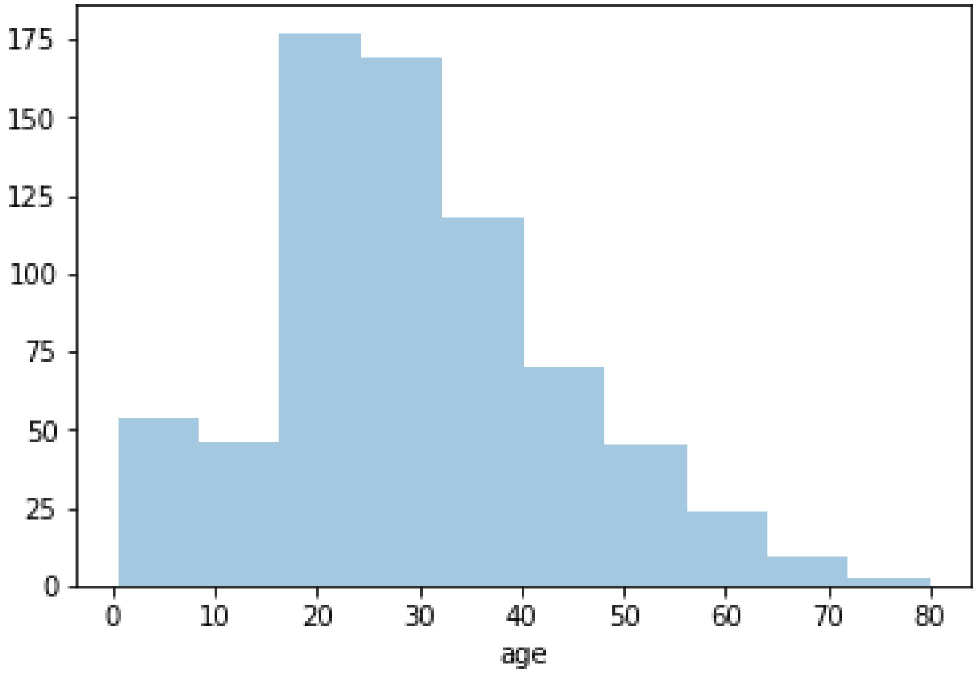
[5]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x119367198>
```



We can change the number of bins used to plot our histogram to change the *granularity* of our distribution plot.

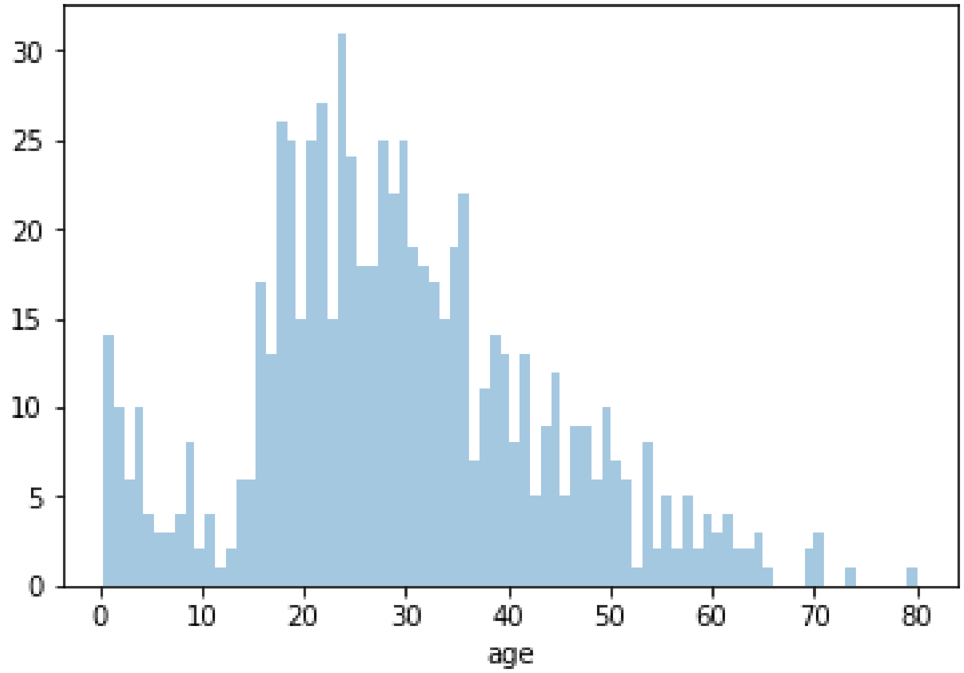
```
[6]: # histogram of age
     sns.distplot(titanic['age'].dropna(), kde=False, bins=10)
```

```
[6]: <matplotlib.axes._subplots.AxesSubplot at 0x119428080>
```



```
[7]: # histogram of age
     sns.distplot(titanic['age'].dropna(), kde=False, bins=80)
```

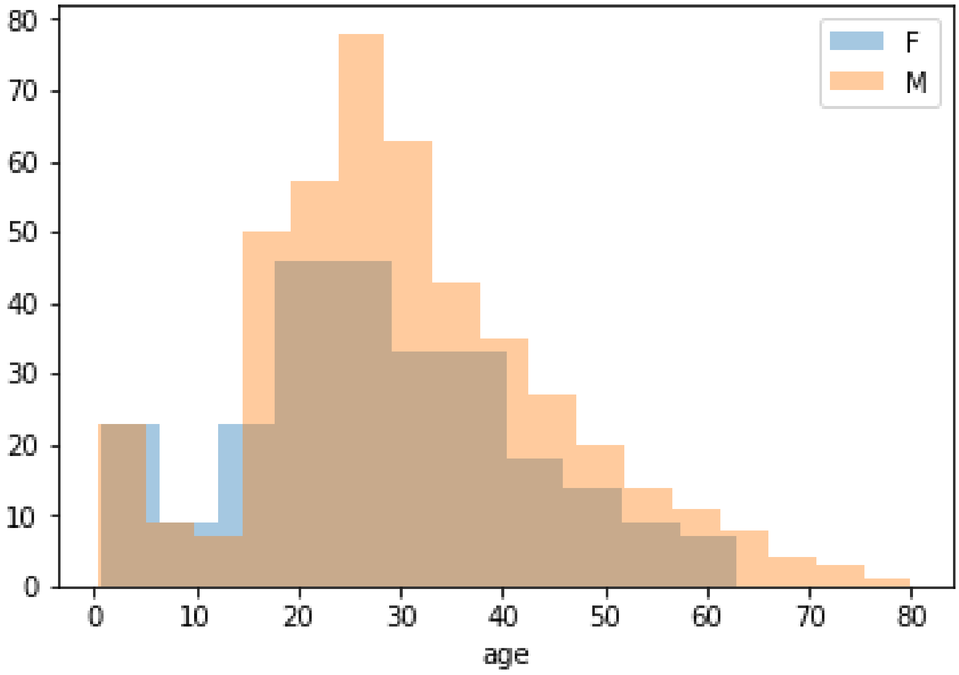
```
[7]: <matplotlib.axes._subplots.AxesSubplot at 0x1194b5518>
```



Unfortunately we can't color our histograms by another variable, but we can compare the distributions of certain variables between *subsets* of our DataFrame by *layering* them.

[8]:
```
# histogram of age for females
sns.distplot(titanic.query('sex == "female"')['age'].dropna(), kde=False,␣
 ↪label="F")
sns.distplot(titanic.query('sex == "male"')['age'].dropna(), kde=False,␣
 ↪label="M")
plt.legend()
```

[8]:
```
<matplotlib.legend.Legend at 0x119436978>
```



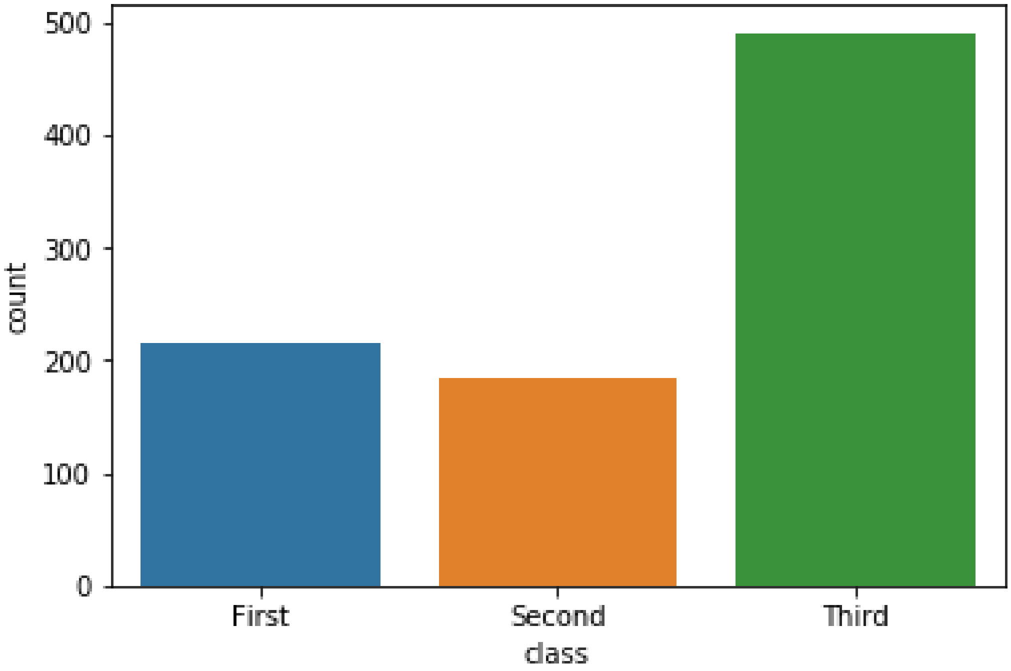
1.3 Count Plots

Count plots can be thought of as histograms for categorical variables.

Let's say we wanted to visualize how many passengers there were in each `class`.

[9]:
```
# count plot of class
sns.countplot(x="class", data=titanic)
```

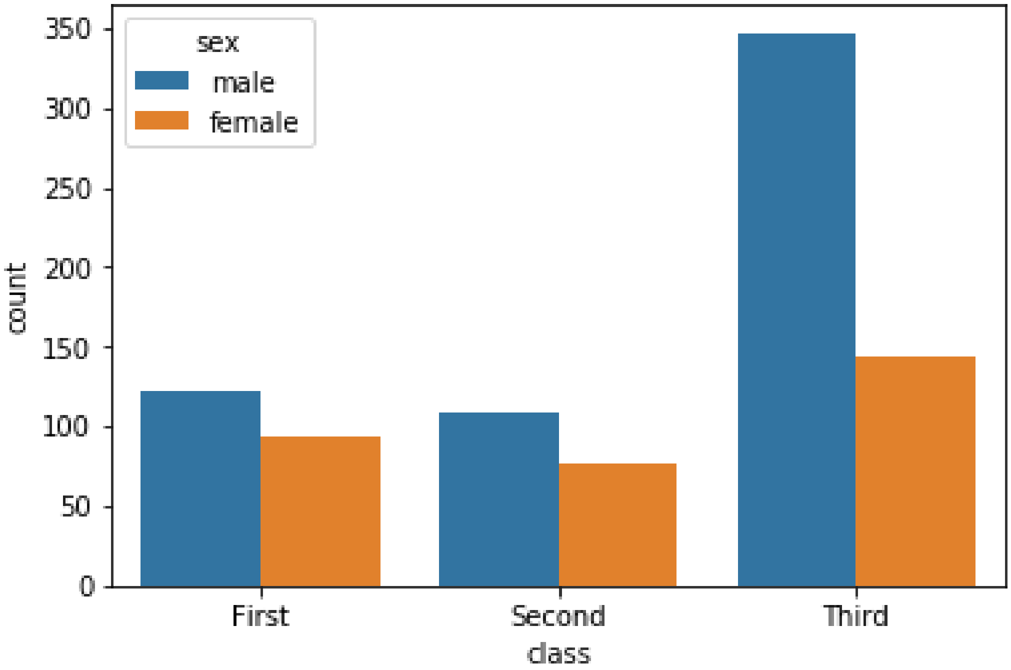
[9]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x119585828>
```



Now, let's stratify each class by the `sex` variable using color. By now you're an expert in this!

[10]:
```
# stratify class by sex variable
sns.countplot(x="class", hue = "sex",  data=titanic)
```

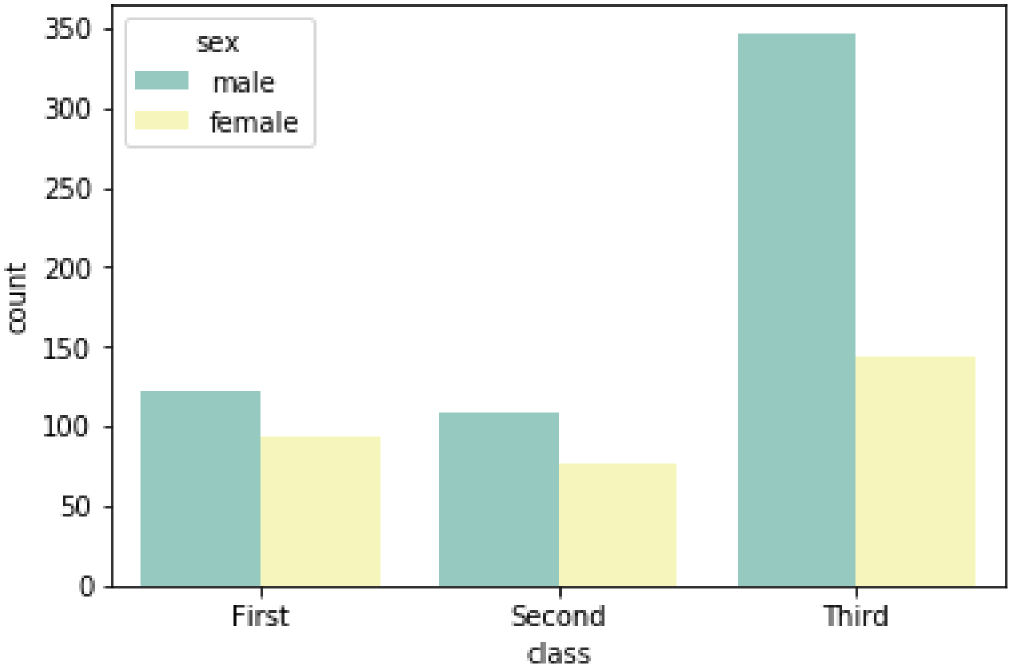
[10]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x1198e82e8>
```



As always we can change the color palette:

[11]:
```
# change color palette
sns.countplot(x="class", hue = "sex", palette = "Set3", data=titanic)
```

[11]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x1199a0e80>
```



In this lesson you learned: * How to create barplots in seaborn * How to stratify barplots by another variable using color (`hue`) * How to create histograms in seaborn * Changing the granularity of the histograms (`bins`) * How to create count plots in seaborn * How to stratify count plots by another variable using color (`hue`)