

KEY_Practice19B_BarCharts_Histograms

July 19, 2019

1 Bar Charts and Histograms

As always, let's begin by importing our necessary packages and reading in/previewing our data. In this practice we will continue to explore the titanic dataset.

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

```
[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
```

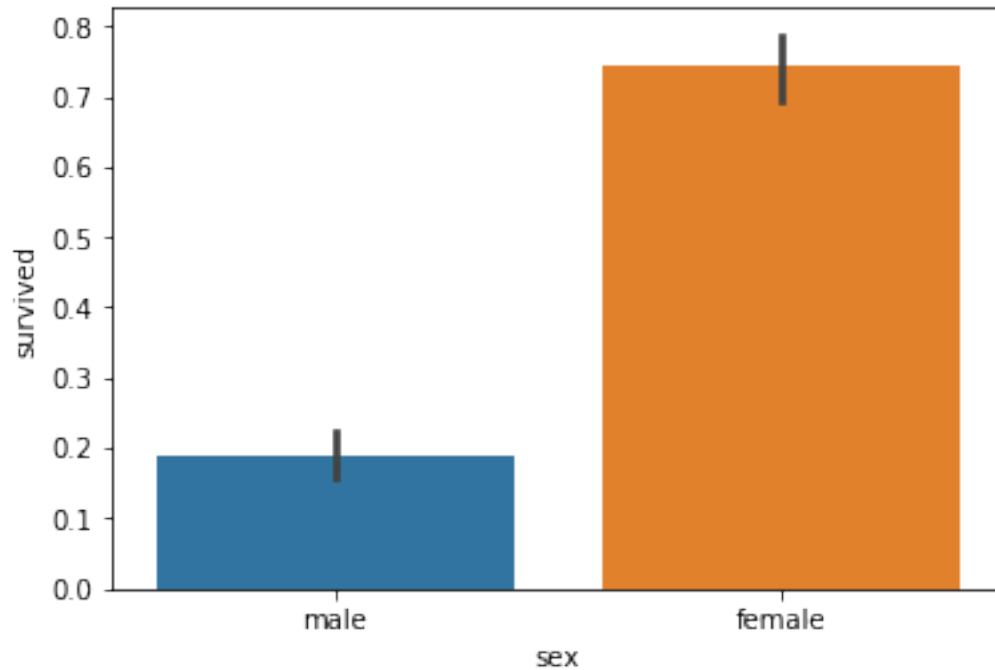
```
   who  adult_male  deck  embark_town  alive  alone
0  man          True  NaN  Southampton    no  False
1 woman         False   C   Cherbourg   yes  False
2 woman         False  NaN  Southampton   yes   True
3 woman         False   C   Southampton   yes  False
4  man          True  NaN  Southampton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

1.1 Bar Charts

Generate a barplot of survived (y-axis) across sex (x-axis).

```
[5]: # barplot of sex vs survived
sns.barplot(x="sex", y = 'survived', data=titanic)
```

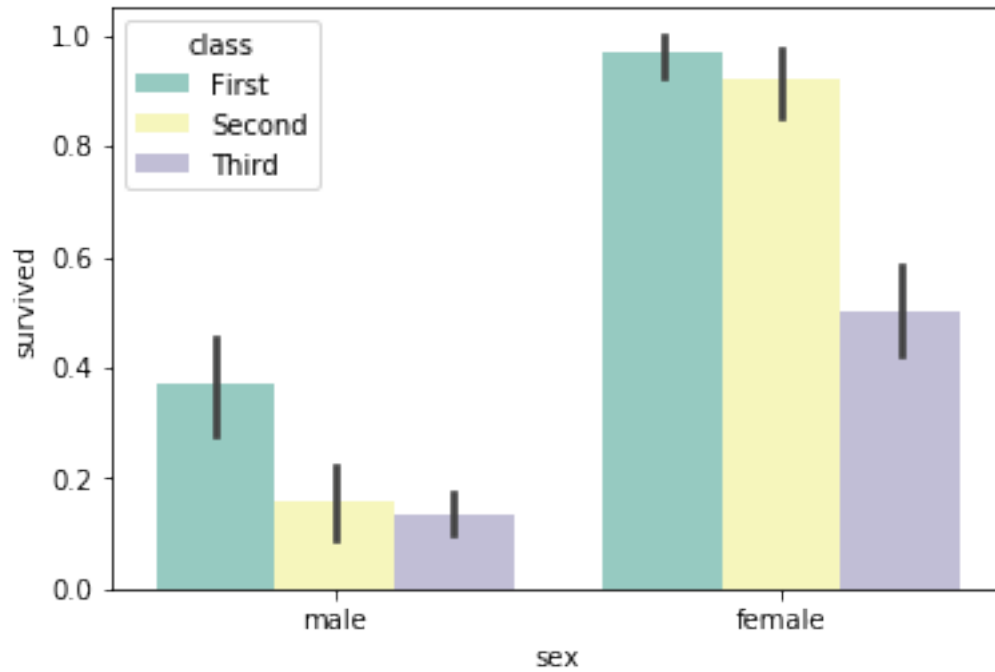
```
[5]: <matplotlib.axes._subplots.AxesSubplot at 0x1a162ba438>
```



Now, generate a barplot of survived (y-axis) across sex (x-axis), stratified by class. Choose your favorite color palette.

```
[4]: # barplot of sex vs survived stratified by class
sns.barplot(x="sex", y = 'survived', hue = "class", palette = "Set3", data=titanic)
```

```
[4]: <matplotlib.axes._subplots.AxesSubplot at 0x1a161fac18>
```

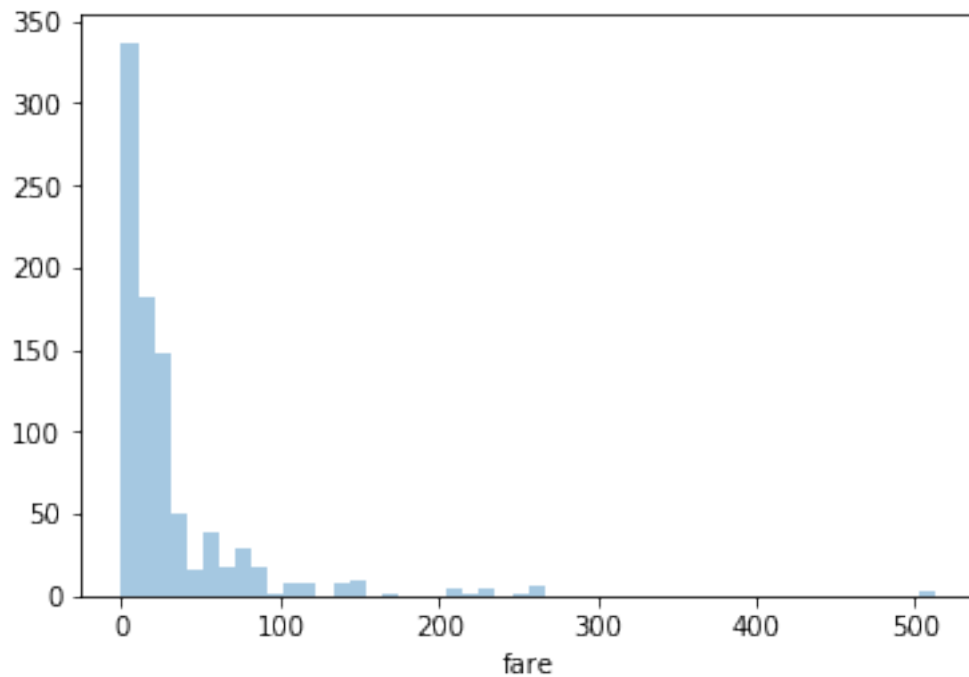


1.2 Histograms

Generate a histogram of fare

```
[6]: #histogram of fare
sns.distplot(titanic['fare'], kde=False)
```

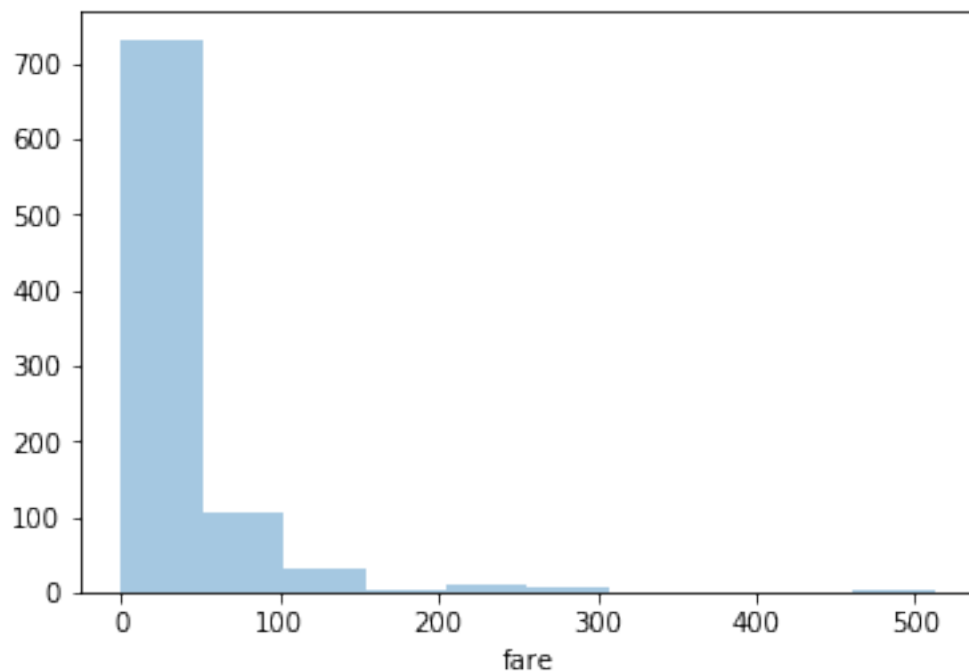
[6]: <matplotlib.axes._subplots.AxesSubplot at 0x1a16399048>



Now let's play with the bins parameter. First let's try a small number of bins, like 10:

```
[12]: #histogram of fare with 10 bins  
sns.distplot(titanic['fare'], kde=False, bins=10)
```

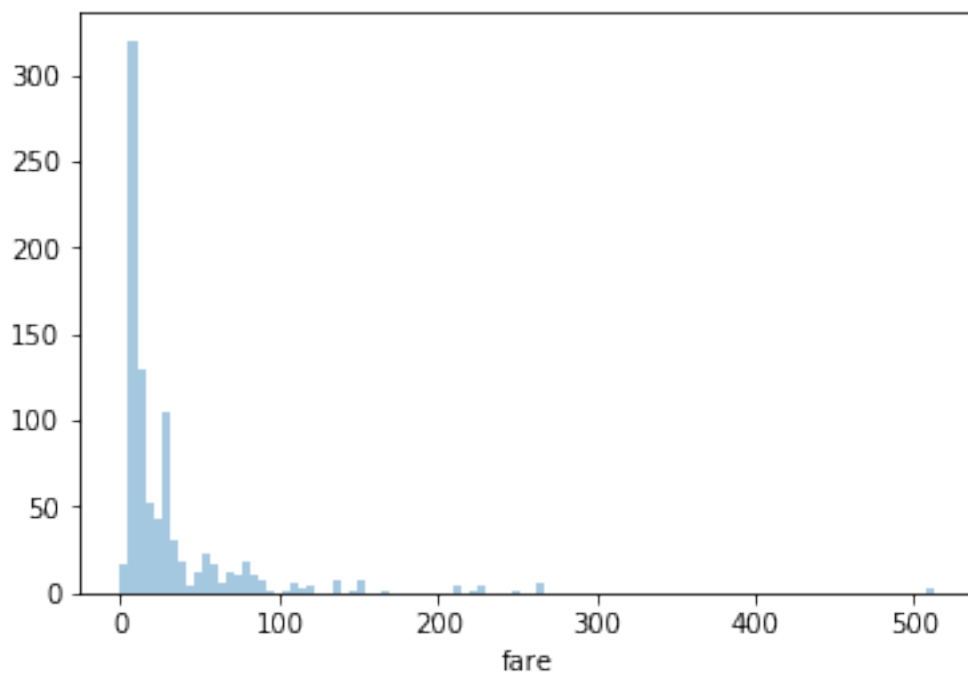
```
[12]: <matplotlib.axes._subplots.AxesSubplot at 0x1a16be4a20>
```



Now, let's try a larger number of bins, say 100.

```
[14]: #histogram of fare with 100 bins
sns.distplot(titanic['fare'], kde=False, bins=100)
```

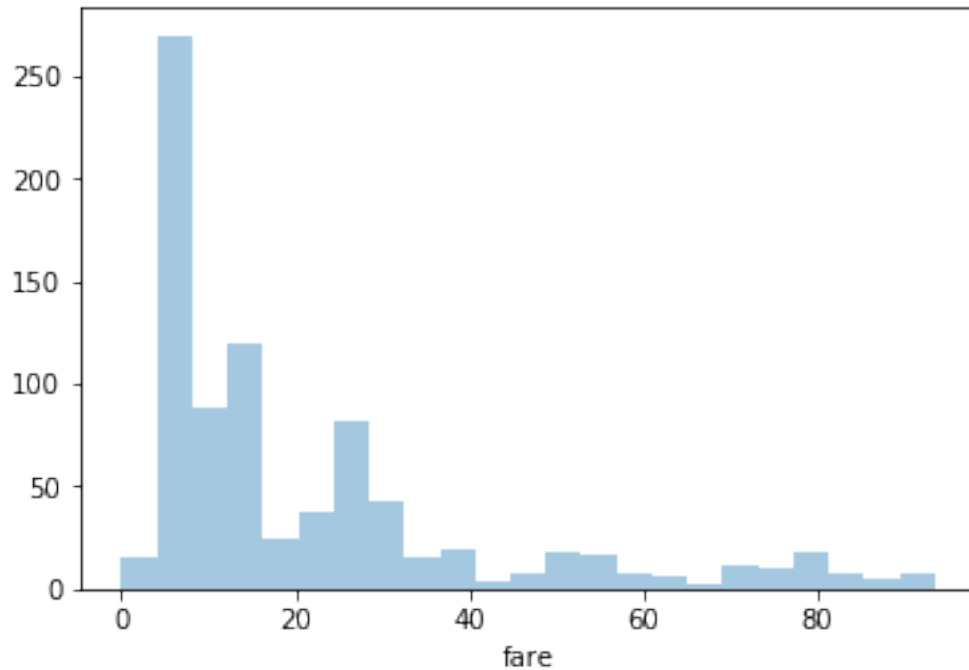
```
[14]: <matplotlib.axes._subplots.AxesSubplot at 0x1a16e180b8>
```



As you can see, most of the values are pretty small (< 100 dollars), and there are only a few really large values. This is called a long-tailed distribution. Can you plot a histogram of just the fares that are less than 100 dollars?

```
[15]: #histogram of fare
sns.distplot(titanic.query('fare < 100')['fare'], kde=False)
```

```
[15]: <matplotlib.axes._subplots.AxesSubplot at 0x1a16fc14e0>
```

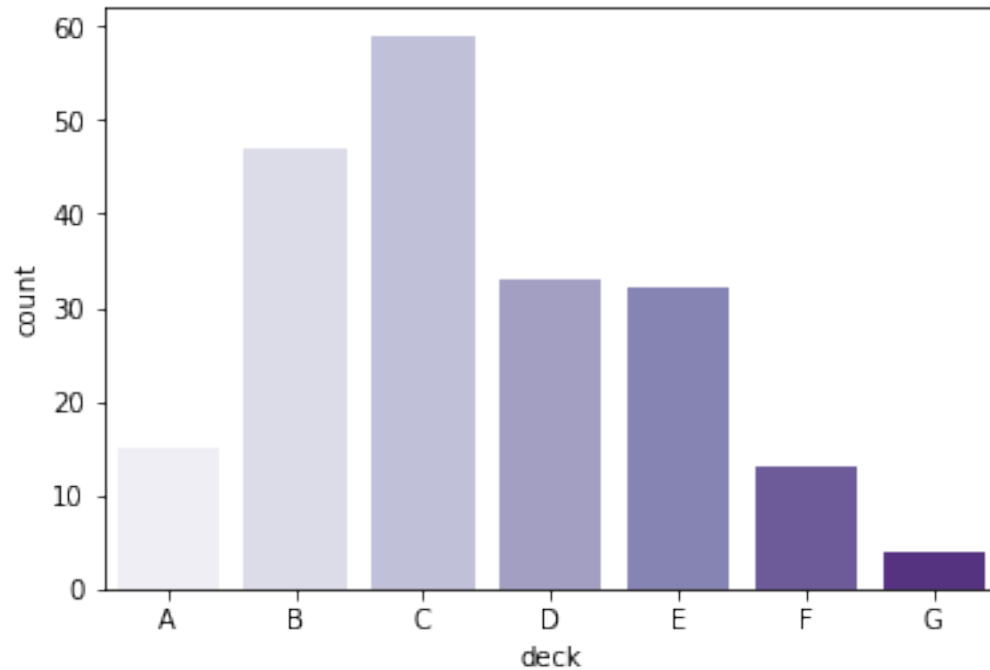


1.3 Count Plots

Generate a count plot to visualize the distribution the deck variable across all passengers in our data set. Choose a sequential color palette.

```
[16]: # count plot of deck  
sns.countplot(x="deck", palette = "Purples", data=titanic)
```

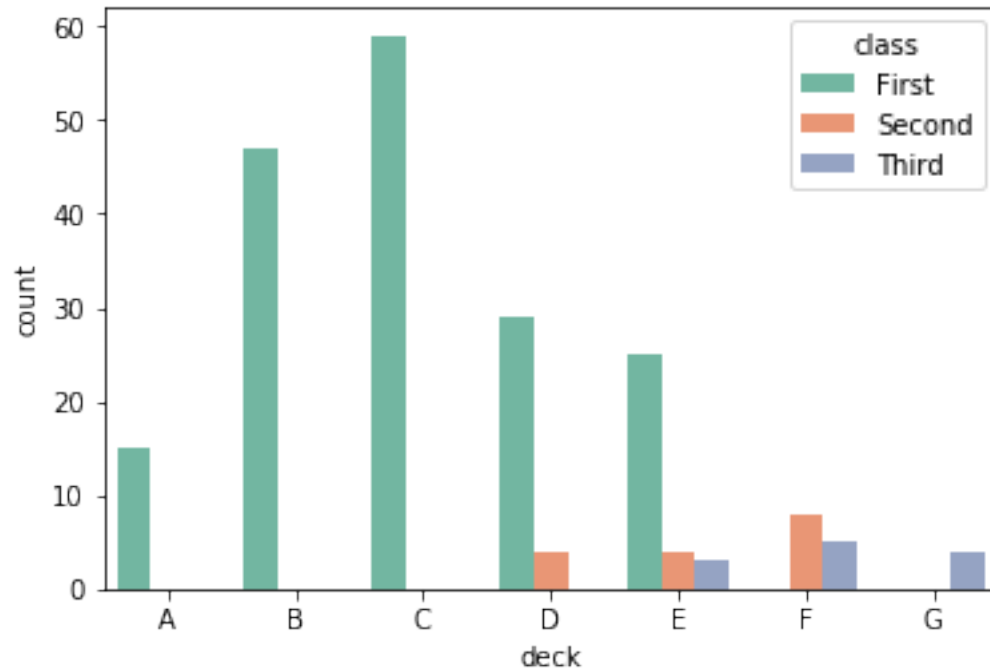
```
[16]: <matplotlib.axes._subplots.AxesSubplot at 0x1a171378d0>
```



Now stratify your plot using the class variable. What kind of color palette is appropriate now?

```
[17]: # count plot of deck stratified by class
sns.countplot(x="deck", hue = "class", palette = "Set2", data=titanic)
```

```
[17]: <matplotlib.axes._subplots.AxesSubplot at 0x1a17218710>
```



1.4 Congratulations!

You just completed all of the lessons in the GWC Data Science Summer Experience! Now it's time to put your data science skills to the test in your projects.