KEY_Practice20_BarCharts_Histograms

May 25, 2020

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
# set up for inline plotting
%matplotlib inline
```

[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	

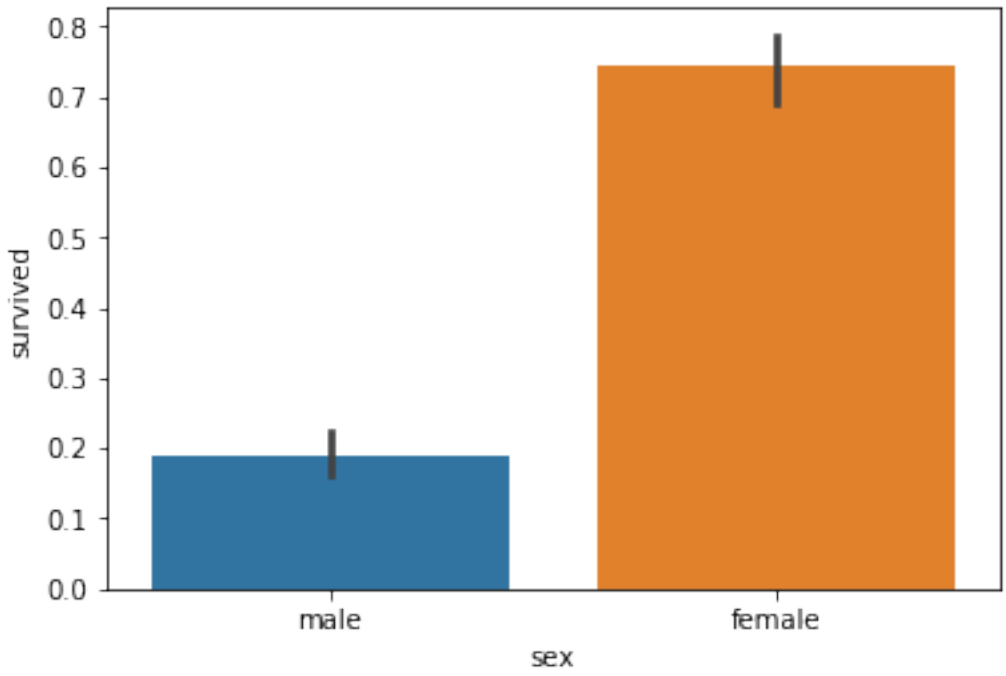
	who	adult_male	deck	embark_town	alive	alone
0	man	True	NaN	Southampton	no	False
1	woman	False	C	Cherbourg	yes	False
2	woman	False	NaN	Southampton	yes	True
3	woman	False	C	Southampton	yes	False
4	man	True	NaN	Southampton	no	True

1.1 Bar Charts

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

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

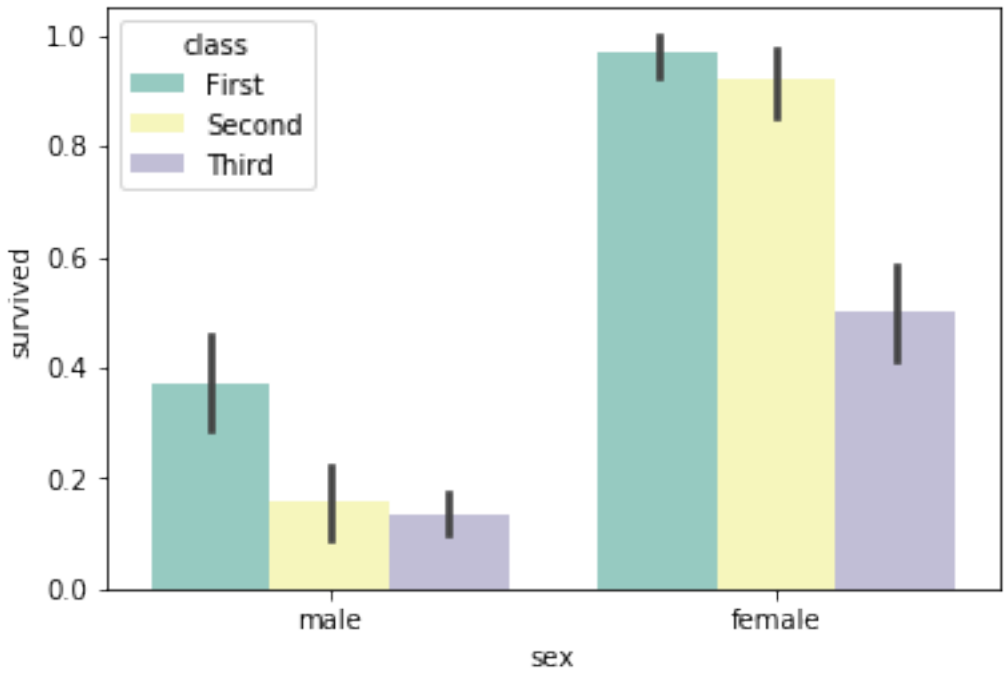
[3]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x118fea588>
```



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 0x11c5e1be0>
```

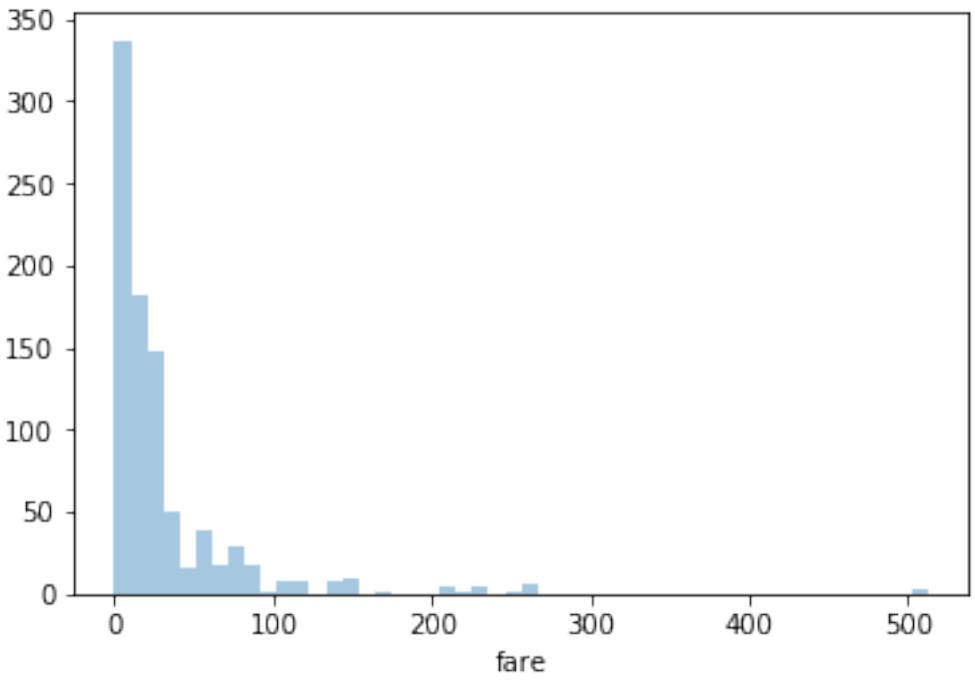


1.2 Histograms

Generate a histogram of `fare`

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

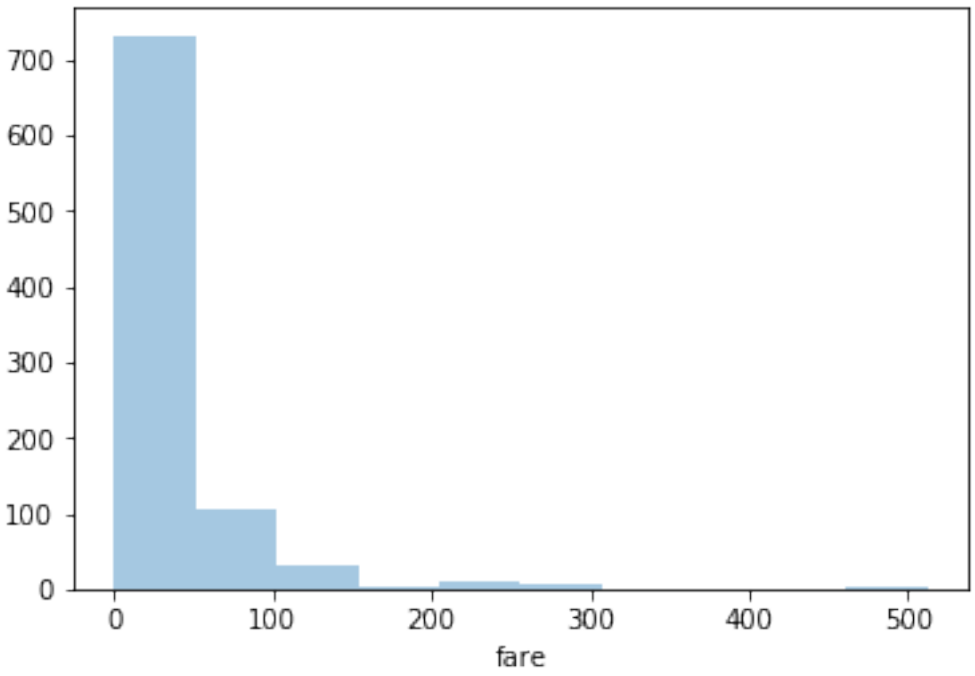
[5]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x11c606fd0>
```



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

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

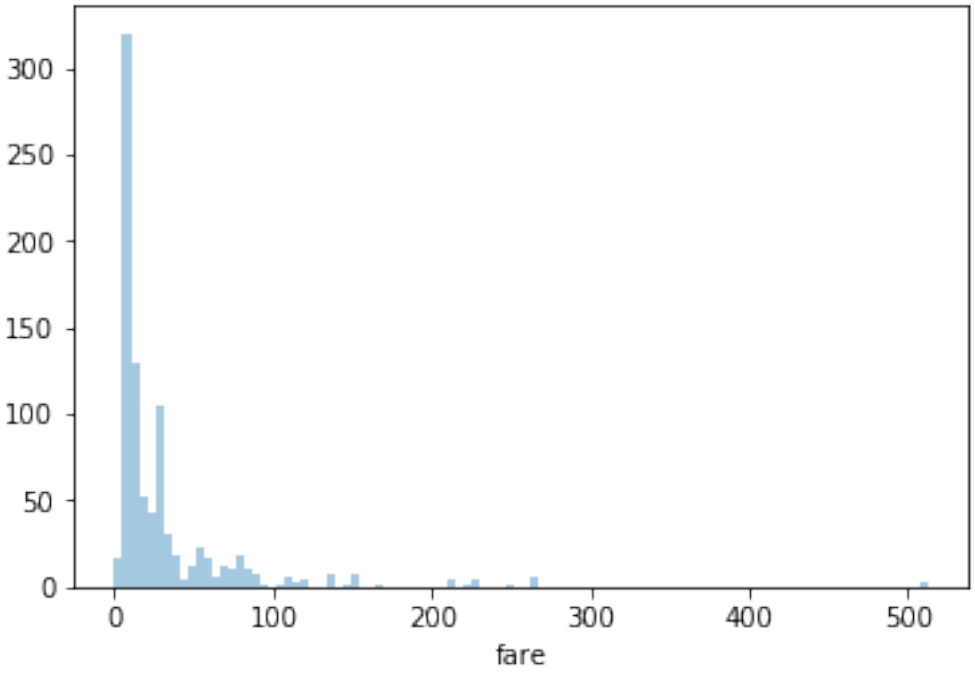
[6]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x11c723d30>
```



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

[7]:
```
#histogram of fare with 100 bins
sns.distplot(titanic['fare'], kde=False, bins=100)
```

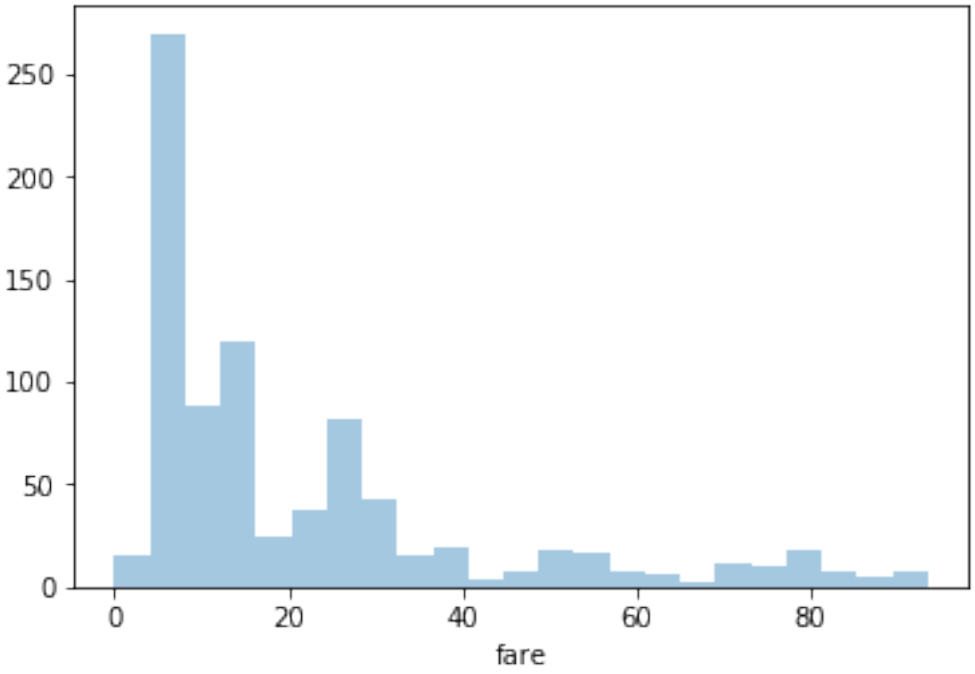
[7]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x11c843ac8>
```



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?

[8]:
```
#histogram of fare
sns.distplot(titanic.query('fare < 100')['fare'], kde=False)
```

[8]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x11cb77630>
```

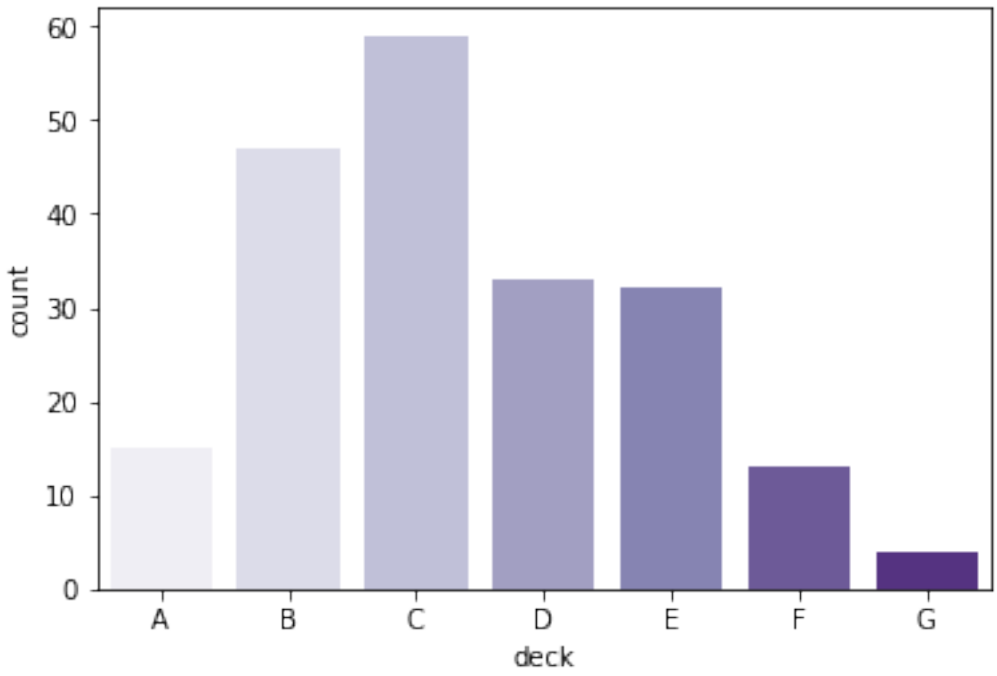


1.3 Count Plots

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

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

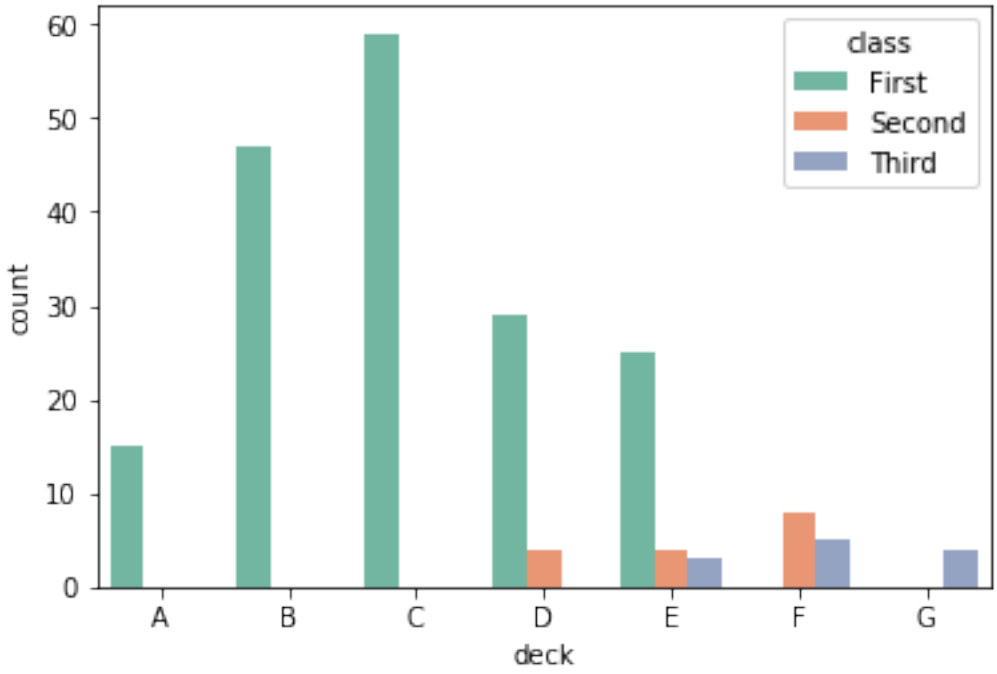
[9]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x11c87b668>
```



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

[10]:
```
# count plot of deck stratified by class
sns.countplot(x="deck", hue = "class", palette = "Set2", data=titanic)
```

[10]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x11ccf8198>
```



1.4 Congratulations!

You just completed all of the core lessons in the GWC Club! Now it's time to keep putting your data science skills to the test in your projects.