0.0.1 Data Visualization using Seaborn (a Python library)

Tutorial by: Navie Narula, Digital Centers Teaching Intern

Created for: Research Data Services at Columbia University Libraries

Resources used to create tutorial: DataCamp’s Introductory Tutorial Pandey’s Visualization Examples Seaborn PyData Swarm Plots Seaborn PyData Heat Maps List of Colors in Python

In [3]: # import libraries
    import pandas as pd
    import numpy as np
    import matplotlib.pyplot as plt
    import seaborn

    %matplotlib inline

The seaborn library has many in-house datasets. You may find them here. We’ll be starting off with the tips dataset.

In [4]: # load in data and save to a variable
df = seaborn.load_dataset("tips")

In [5]: # first five rows of dataset
df.head()

Out[5]:
<table>
<thead>
<tr>
<th>total_bill</th>
<th>tip</th>
<th>sex</th>
<th>smoker</th>
<th>day</th>
<th>time</th>
<th>size</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>16.99</td>
<td>Female</td>
<td>No</td>
<td>Sun</td>
<td>Dinner</td>
<td>2</td>
</tr>
<tr>
<td>1</td>
<td>10.34</td>
<td>Male</td>
<td>No</td>
<td>Sun</td>
<td>Dinner</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>21.01</td>
<td>Male</td>
<td>No</td>
<td>Sun</td>
<td>Dinner</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>23.68</td>
<td>Male</td>
<td>No</td>
<td>Sun</td>
<td>Dinner</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>24.59</td>
<td>Female</td>
<td>No</td>
<td>Sun</td>
<td>Dinner</td>
<td>4</td>
</tr>
</tbody>
</table>

In [6]: # last five rows of dataset
df.tail()

Out[6]:
<table>
<thead>
<tr>
<th>total_bill</th>
<th>tip</th>
<th>sex</th>
<th>smoker</th>
<th>day</th>
<th>time</th>
<th>size</th>
</tr>
</thead>
<tbody>
<tr>
<td>239</td>
<td>29.03</td>
<td>Male</td>
<td>Yes</td>
<td>Sat</td>
<td>Dinner</td>
<td>3</td>
</tr>
<tr>
<td>240</td>
<td>27.18</td>
<td>Female</td>
<td>Yes</td>
<td>Sat</td>
<td>Dinner</td>
<td>2</td>
</tr>
<tr>
<td>241</td>
<td>22.67</td>
<td>Male</td>
<td>Yes</td>
<td>Sat</td>
<td>Dinner</td>
<td>2</td>
</tr>
<tr>
<td>242</td>
<td>17.82</td>
<td>Male</td>
<td>No</td>
<td>Sat</td>
<td>Dinner</td>
<td>2</td>
</tr>
<tr>
<td>243</td>
<td>18.78</td>
<td>Female</td>
<td>No</td>
<td>Thur</td>
<td>Dinner</td>
<td>2</td>
</tr>
</tbody>
</table>
Swarm Plots

In [12]: # use swarmplot to visualize tip observations and amounts
   # by day of the week
   seaborn.swarmplot(x="day", y="tip", data=df)
   seaborn.set_style("whitegrid")
   plt.show()

In [16]: # visualize tip observations
   seaborn.swarmplot(x=df["tip"])
   seaborn.set_style("darkgrid")
   plt.show()
In [28]: # color points by category
    # create customized palette
gender_palette = ['#A833FF', '#FFAF33']
seaborn.swarmplot(x='day', y='tip', hue='sex', palette=gender_palette, data=df)
plt.show()
In [41]: # control plot order on x-axis
seaborn.swarmplot(x="smoker", y="total_bill", data=df, palette="husl", order=["Yes", "No"],
plt.show()

Violin Plots

In [50]: # plot tips
seaborn.violinplot(x = df["tip"], color="gold")
plt.show()
In [54]: # draw plot based on variable
    seaborn.violinplot(x="sex", y="total_bill", data=df)
    plt.show()
In [57]: # Split drawings to compare with hue/legend variables
seaborn.violinplot(x = "time", y = "tip", data=df, hue = "sex", palette = "dark", split = True)
plt.legend()
plt.show()

Facet Grids

In [72]: # draw facet grid based on tip variable
fg = seaborn.FacetGrid(df, col = "time", row = "sex")
fg = fg.map(plt.hist, "tip", color = "tomato")
In [73]: # we can also change the type of plot
    # ...and the colors around the points
    fg = seaborn.FacetGrid(df, col="time", row="sex")
    fg = fg.map(plt.scatter, "total_bill", "tip", color="floralwhite", edgecolor="hotpink")
In [84]: # plot by category
    
    x = seaborn.FacetGrid(df, col="time", hue="sex")
    
    x = x.map(plt.scatter,"total_bill","tip")
    
    x=x.add_legend()
# create random data
uniform_data = np.random.rand(5, 3) # five rows, 3 columns
print(uniform_data)
seaborn.heatmap(uniform_data)
plt.show()

[[0.39376482 0.61566449 0.94105178]
[0.57360765 0.66858876 0.03326495]
[0.35962929 0.46553437 0.28784689]
[0.32919801 0.02822342 0.38018925]
[0.69303348 0.559752 0.61115946]]
In [147]: # load in flights dataset
flights = seaborn.load_dataset("flights")

In [148]: # print first five rows
flights.head()

Out[148]:
<table>
<thead>
<tr>
<th>year</th>
<th>month</th>
<th>passengers</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>January</td>
<td>112</td>
</tr>
<tr>
<td>1</td>
<td>February</td>
<td>118</td>
</tr>
<tr>
<td>2</td>
<td>March</td>
<td>132</td>
</tr>
<tr>
<td>3</td>
<td>April</td>
<td>129</td>
</tr>
<tr>
<td>4</td>
<td>May</td>
<td>121</td>
</tr>
</tbody>
</table>

In [149]: # print last five rows
flights.tail()

Out[149]:
<table>
<thead>
<tr>
<th>year</th>
<th>month</th>
<th>passengers</th>
</tr>
</thead>
<tbody>
<tr>
<td>139</td>
<td>1960</td>
<td>August</td>
</tr>
<tr>
<td>140</td>
<td>1960</td>
<td>September</td>
</tr>
<tr>
<td>141</td>
<td>1960</td>
<td>October</td>
</tr>
<tr>
<td>142</td>
<td>1960</td>
<td>November</td>
</tr>
<tr>
<td>143</td>
<td>1960</td>
<td>December</td>
</tr>
</tbody>
</table>

In [150]: flights = flights.pivot("month", "year", "passengers")
# draw border
x = seaborn.heatmap(flights, linewidths=0.3)
In [152]: # change color and add value
   x = seaborn.heatmap(flights, annot=True, fmt="d", cmap="YlGnBu")
Now, it’s time for you to start working with your own data of choice and produce the visualizations you like! You can use one of seaborn’s in-house datasets or load in your own. If you’d like to use in your own .csv file, you can load that into a dataframe by doing something like this:

```python
import pandas as pd
df = pd.read_csv("<filename>", sep=",")
```