

Data Visualization with Matplotlib and Seaborn



Hello!

I am Eslam Ahmed

I am a software engineer.

You can find me at jeksogsa@gmail.com



Hello!

I am Eman Ehab

I am a ML research engineer.

You can find me at
emanehab.ieee@gmail.com



Agenda

- ⬡ What is Data Visualization
- ⬡ Matplotlib and Seaborn
- ⬡ Distribution Plots
- ⬡ Categorical Plots
- ⬡ Matrix Plots
- ⬡ Regression Plots
- ⬡ Color palettes
- ⬡ Change Plot Size



Agenda

What is Data Visualization

 Matplotlib and Seaborn

 Distribution Plots

 Categorical Plots

 Matrix Plots

 Regression Plots

 Color palettes

 Change Plot Size



What is Data Visualization

Data visualization is the graphical representation of information and data. By using visual elements like charts, graphs, and maps, data visualization tools provide an accessible way to see and understand trends, outliers, and patterns in data.

Our eyes are drawn to colors and patterns. We can quickly identify red from blue, square from circle. Our culture is visual, including everything from art and advertisements to TV and movies.

Data visualization is another form of visual art that grabs our interest and keeps our eyes on the message.



Agenda

- ⬡ What is Data Visualization
- ⬡ Matplotlib and Seaborn
- ⬡ Distribution Plots
- ⬡ Categorical Plots
- ⬡ Matrix Plots
- ⬡ Regression Plots
- ⬡ Color palettes
- ⬡ Change Plot Size



Matplotlib and Seaborn

Seaborn and Matplotlib are two of Python's most powerful visualization libraries. Seaborn is built on top of matplotlib so it's more like a wrapper, Seaborn uses fewer syntax and has stunning default themes and Matplotlib is more easily customizable through accessing the classes.

We will focus on Seaborn as we know it's so much easier and give you a lot of out-of-the-box plots.



<https://matplotlib.org/gallery/index.html>

<https://seaborn.pydata.org/index.html>

Matplotlib and Seaborn

Install



```
1 >_ conda install matplotlib  
2 >_ conda install seaborn
```

Use



```
1 import matplotlib.pyplot as plt  
2 import seaborn as sns
```



Agenda

- ⬡ What is Data Visualization
- ⬡ Matplotlib and Seaborn
- ⬡ **Distribution Plots**
- ⬡ Categorical Plots
- ⬡ Matrix Plots
- ⬡ Regression Plots
- ⬡ Color palettes
- ⬡ Change Plot Size



Distribution Plots

Univariate Plots

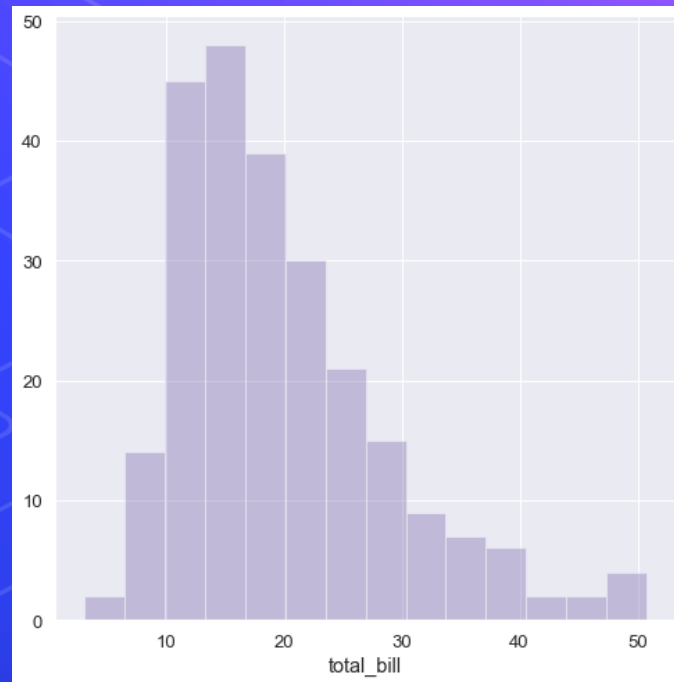
distplot

kdeplot

Bivariate Plots

jointplot

pairplot



```
1 sns.distplot(tips['total_bill'])
```

Distribution Plots

Univariate Plots

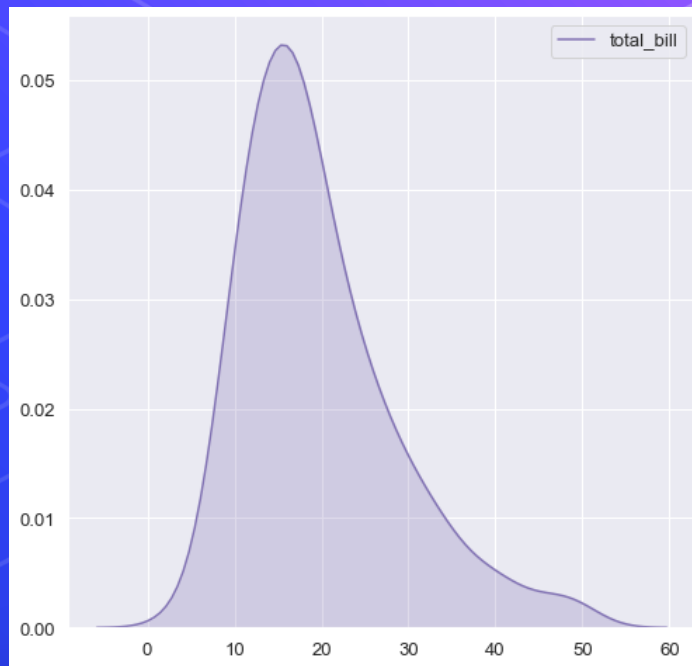
`distplot`

`kdeplot`

Bivariate Plots

`jointplot`

`pairplot`



```
1 sns.kdeplot(tips['total_bill'], shade=True)
```

Distribution Plots

Univariate Plots

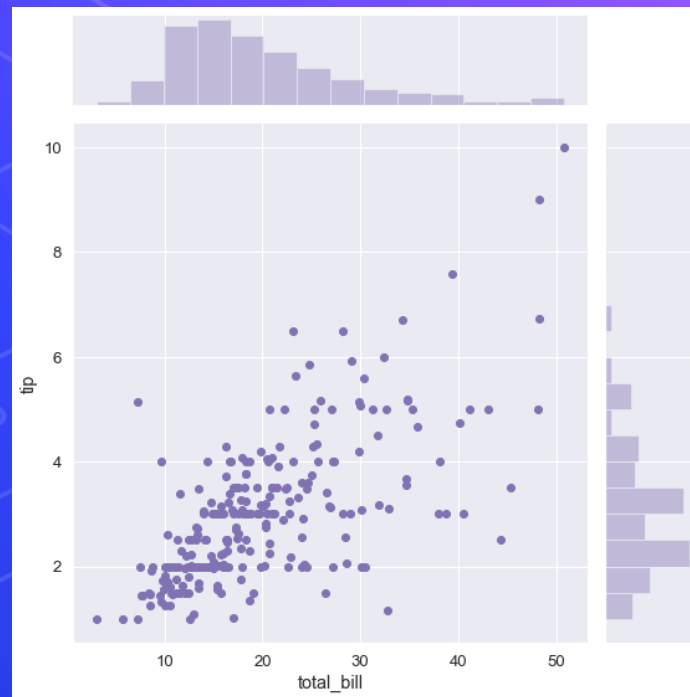
`distplot`

`kdeplot`

Bivariate Plots

`jointplot`

`pairplot`



```
1 sns.jointplot(x='total_bill', y='tip', data=tips, kind='scatter')
```

Distribution Plots



Univariate Plots

`distplot`

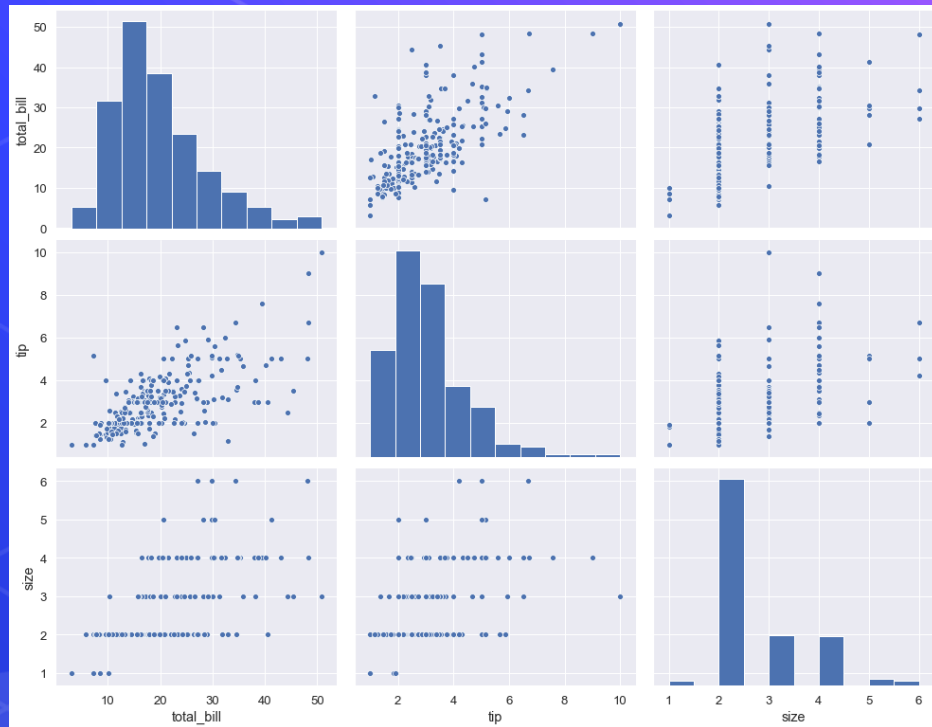
`kdeplot`



Bivariate Plots

`jointplot`

`pairplot`



```
1 sns.pairplot(tips)
```

Agenda

- ⬡ What is Data Visualization
- ⬡ Matplotlib and Seaborn
- ⬡ Distribution Plots
- ⬡ **Categorical Plots**
- ⬡ Matrix Plots
- ⬡ Regression Plots
- ⬡ Color palettes
- ⬡ Change Plot Size



Categorical Plots

⬡ Categorical Distribution Plots

boxplot

violinplot

⬡ Categorical Scatter Plots

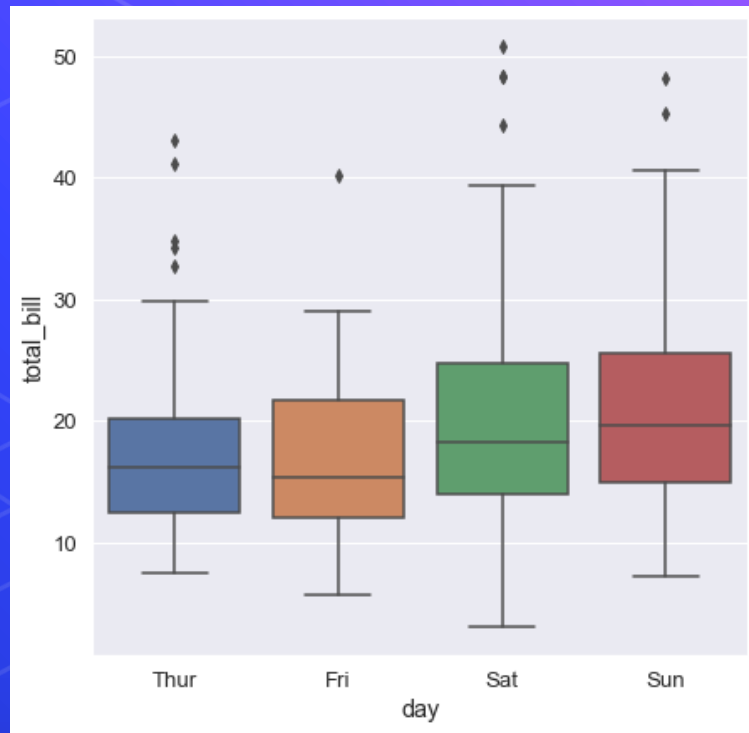
stripplot

swarmplot

⬡ Categorical Estimate Plots

barplot

countplot



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

Categorical Plots

Category Distribution Plots

boxplot

violinplot

Category Scatter Plots

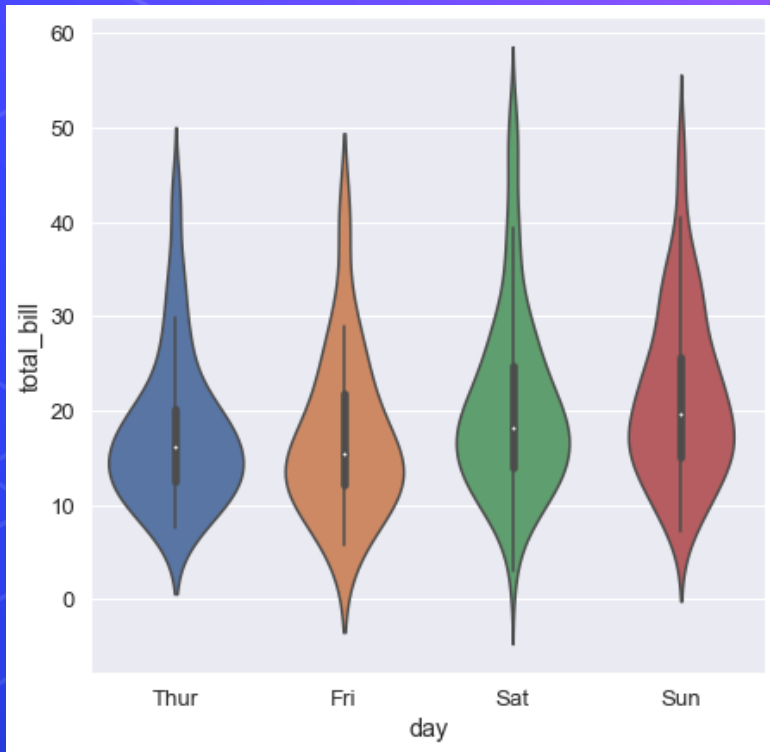
stripplot

swarmplot

Category Estimate Plots

barplot

countplot



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

Categorical Plots

⬡ Categorical Distribution Plots

boxplot

violinplot

⬡ Categorical Scatter Plots

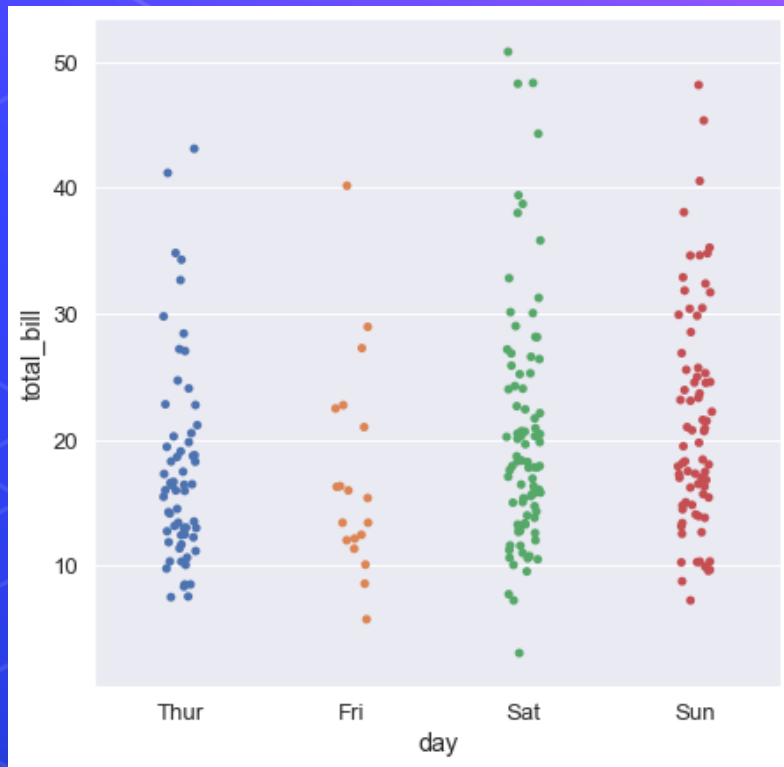
stripplot

swarmplot

⬡ Categorical Estimate Plots

barplot

countplot



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

Categorical Plots

⬡ Categorical Distribution Plots

boxplot

violinplot

⬡ Categorical Scatter Plots

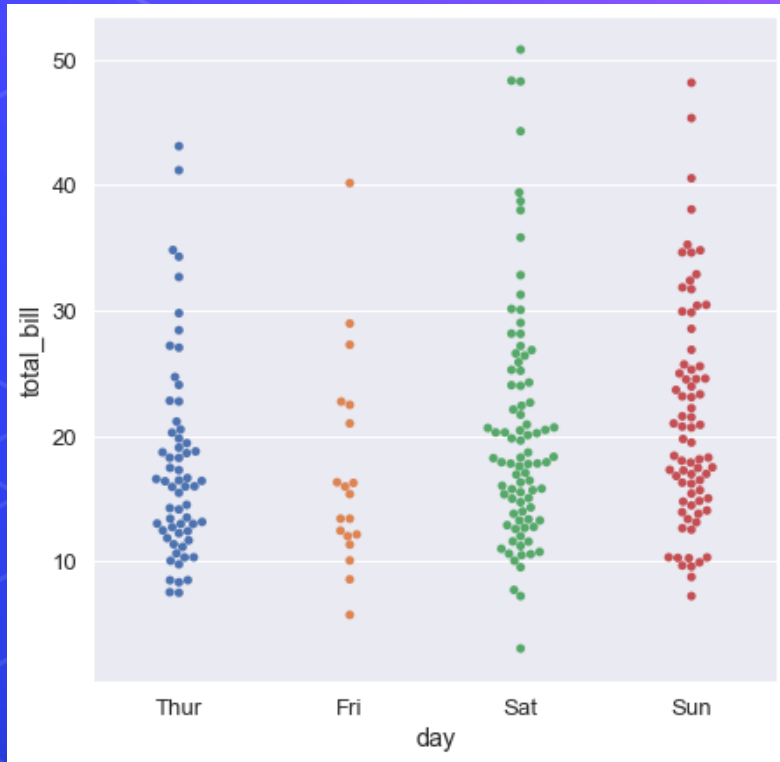
stripplot

swarmplot

⬡ Categorical Estimate Plots

barplot

countplot



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

Categorical Plots

⬡ Categorical Distribution Plots

boxplot

violinplot

⬡ Categorical Scatter Plots

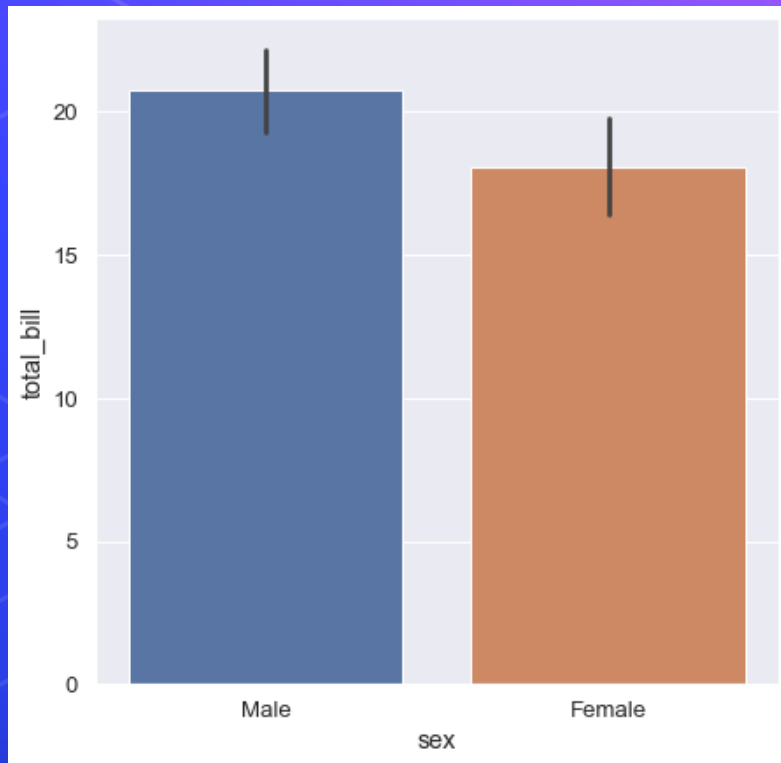
stripplot

swarmplot

⬡ Categorical Estimate Plots

barplot

countplot



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

Categorical Plots

⬡ Categorical Distribution Plots

boxplot

violinplot

⬡ Categorical Scatter Plots

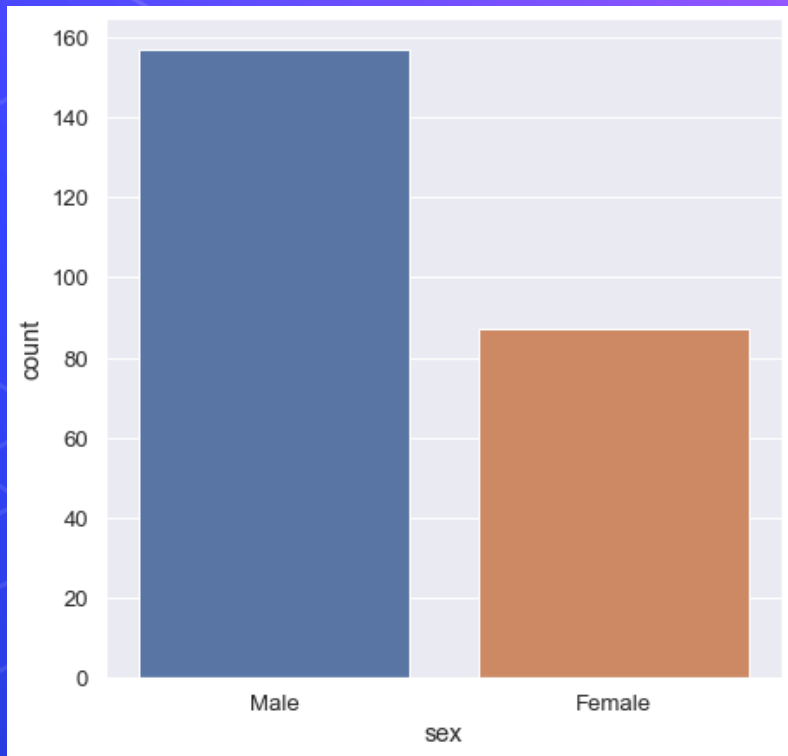
stripplot

swarmplot

⬡ Categorical Estimate Plots

barplot

countplot



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

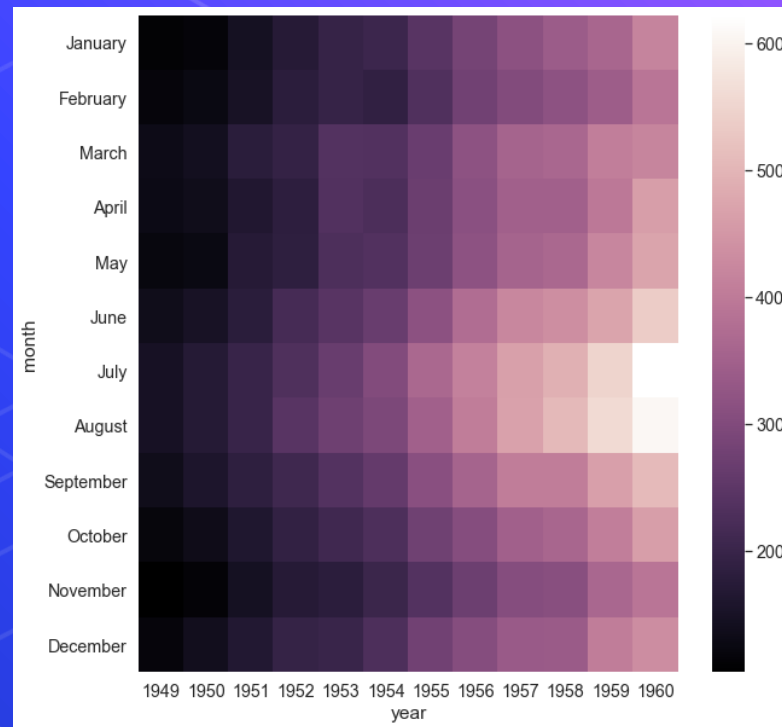
Agenda

- ⬡ What is Data Visualization
- ⬡ Matplotlib and Seaborn
- ⬡ Distribution Plots
- ⬡ Categorical Plots
- ⬡ **Matrix Plots**
- ⬡ Regression Plots
- ⬡ Color palettes
- ⬡ Change Plot Size



Matrix Plots

heatmap



```
1 sns.heatmap(flights)
```

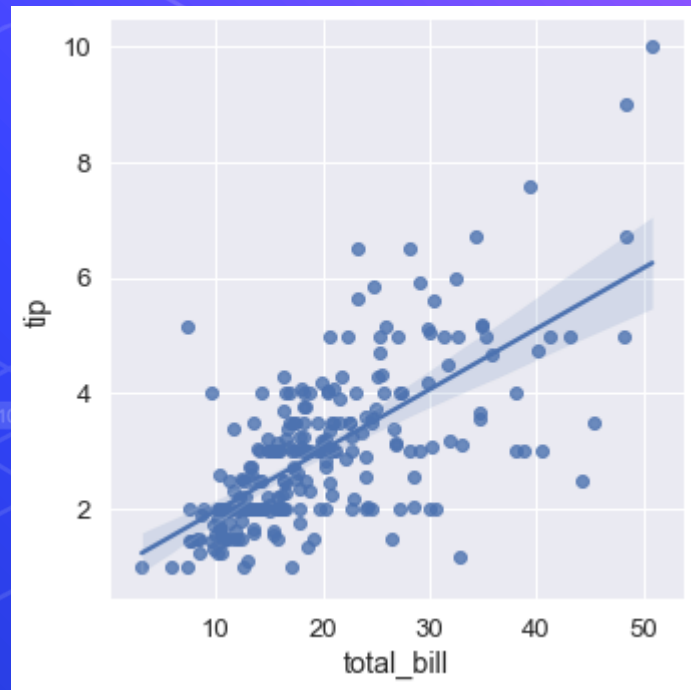
Agenda

- ⬡ What is Data Visualization
- ⬡ Matplotlib and Seaborn
- ⬡ Distribution Plots
- ⬡ Categorical Plots
- ⬡ Matrix Plots
- ⬡ Regression Plots
- ⬡ Color palettes
- ⬡ Change Plot Size



Regression Plots

lmplot



```
1 sns.lmplot(x='total_bill', y='tip', data=tips)
```

Agenda

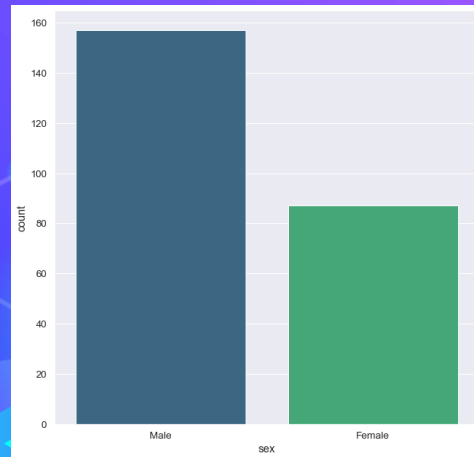
- ⬡ What is Data Visualization
- ⬡ Matplotlib and Seaborn
- ⬡ Distribution Plots
- ⬡ Categorical Plots
- ⬡ Matrix Plots
- ⬡ Regression Plots
- ⬡ Color palettes
- ⬡ Change Plot Size



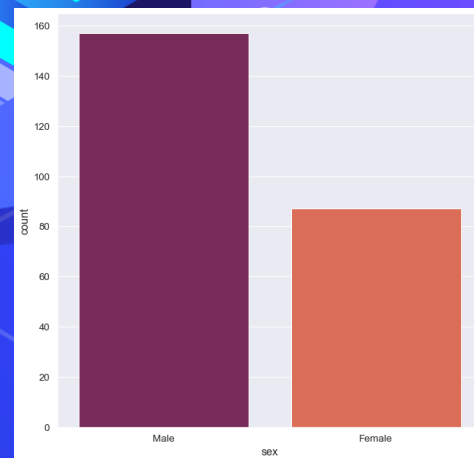
Color palettes

- Set1
- Set2
- Set3
- rocket
- cubehelix
- viridis
- ...

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



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



You can check all possible values for different color palettes from here

https://seaborn.pydata.org/tutorial/color_palettes.html

Agenda

- ⬡ What is Data Visualization
- ⬡ Matplotlib and Seaborn
- ⬡ Distribution Plots
- ⬡ Categorical Plots
- ⬡ Matrix Plots
- ⬡ Regression Plots
- ⬡ Color palettes
- ⬡ Change Plot Size



Change Plot Size

- ⬡ Change size with `sns.set()`
- ⬡ Change size with `aspect` and `height`
- ⬡ Change Font Scale

```
1 sns.set(rc={'figure.figsize': [10, 10]})
```


Change Plot Size


- ⬡ Change size with `sns.set()`
- ⬡ Change size with aspect and height
- ⬡ Change Font Scale



```
1 sns.lmplot('total_bill', 'tip', tips, height=10, aspect=1.2)
```

Change Plot Size

- ⬡ Change size with `sns.set()`
- ⬡ Change size with `aspect` and `height`
- ⬡ Change Font Scale



```
1 sns.set(font_scale=1.2)
```

Questions ?!



Thanks!

>_ Live long and prosper

