

Introduction to Seaborn

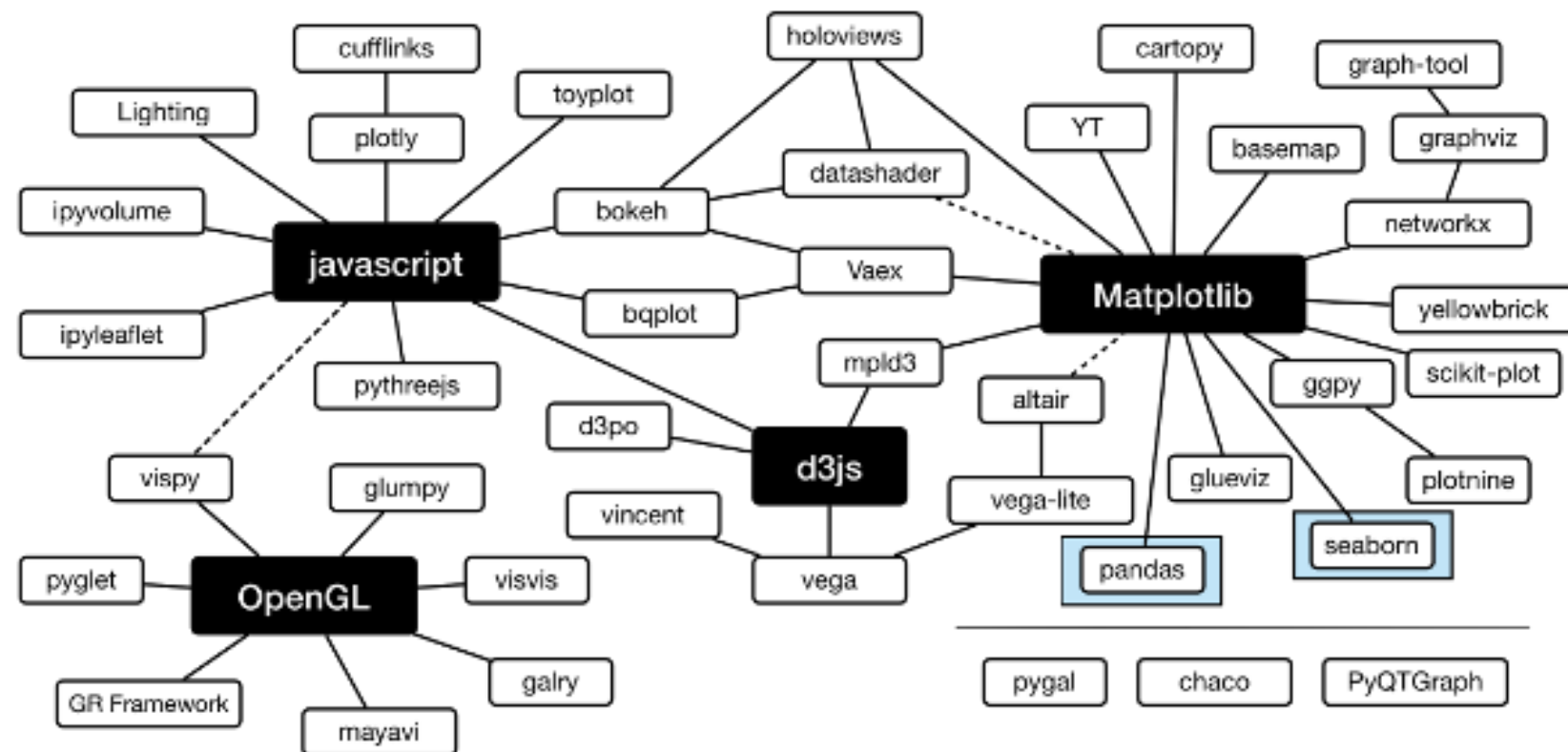
INTERMEDIATE DATA VISUALIZATION WITH SEABORN



Chris Moffitt
Instructor

Python Visualization Landscape

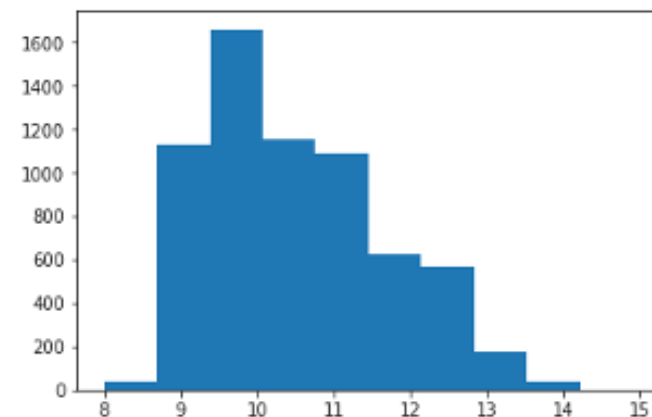
- The python visualization landscape is complex and can be overwhelming



Matplotlib

- `matplotlib` provides the raw building blocks for Seaborn's visualizations
- It can also be used on its own to plot data

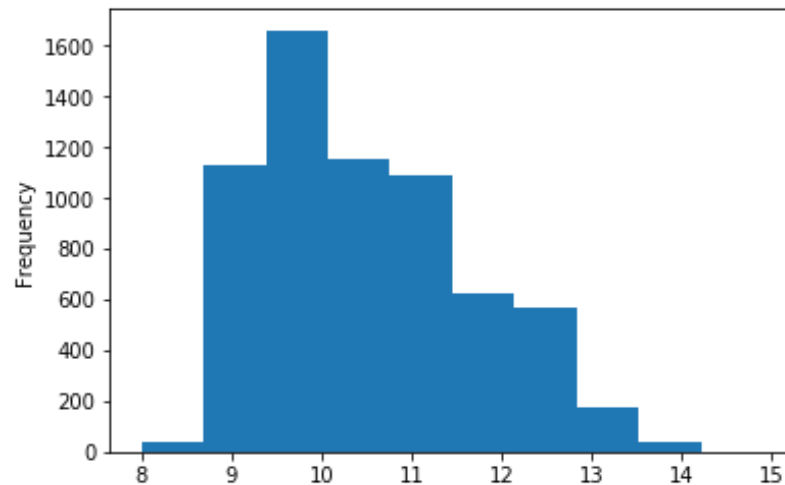
```
import matplotlib.pyplot as plt
import pandas as pd
df = pd.read_csv("wines.csv")
fig, ax = plt.subplots()
ax.hist(df['alcohol'])
```



Pandas

- `pandas` is a foundational library for analyzing data
- It also supports basic plotting capability

```
import pandas as pd
df = pd.read_csv("wines.csv")
df['alcohol'].plot.hist()
```



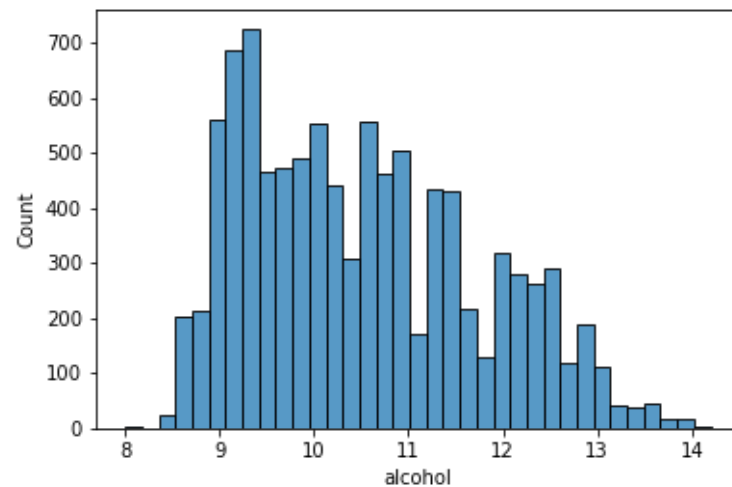
Seaborn

- Seaborn supports complex visualizations of data
- It is built on matplotlib and works best with pandas' dataframes

Seaborn histplot

- The `histplot` is similar to the histogram shown in previous examples
- By default, generates a histogram but can also generate other complex plots

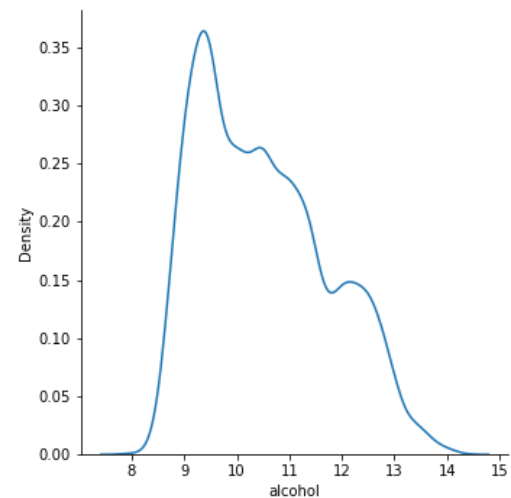
```
import seaborn as sns
sns.histplot(df['alcohol'])
```



Seaborn displot

- The `displot` leverages the `histplot` and other functions for distribution plots
- By default, it generates a histogram but can also generate other plot types

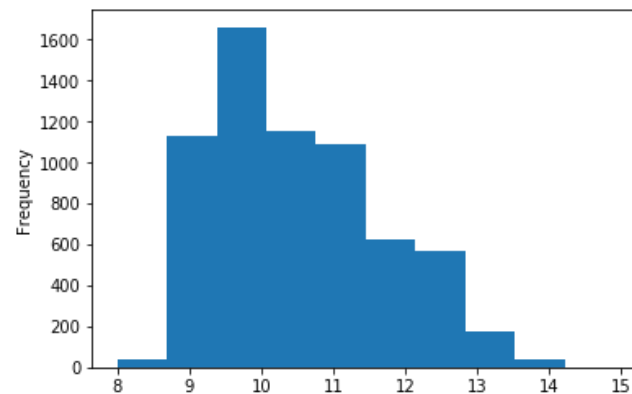
```
import seaborn as sns  
sns.displot(df['alcohol'], kind='kde')
```



pandas Histogram vs. Displot

- Pandas histogram

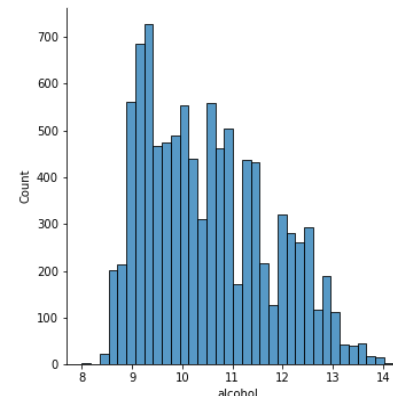
```
df['alcohol'].plot.hist()
```



- Actual frequency of observations
- No outline of bars
- Wide bins
- No x-axis label

- Seaborn displot

```
sns.displot(df['alcohol'])
```



- Automatic label on x-axis
- Muted color palette
- Cleaner plot

Let's practice!

INTERMEDIATE DATA VISUALIZATION WITH SEABORN

Using the distribution plot

INTERMEDIATE DATA VISUALIZATION WITH SEABORN

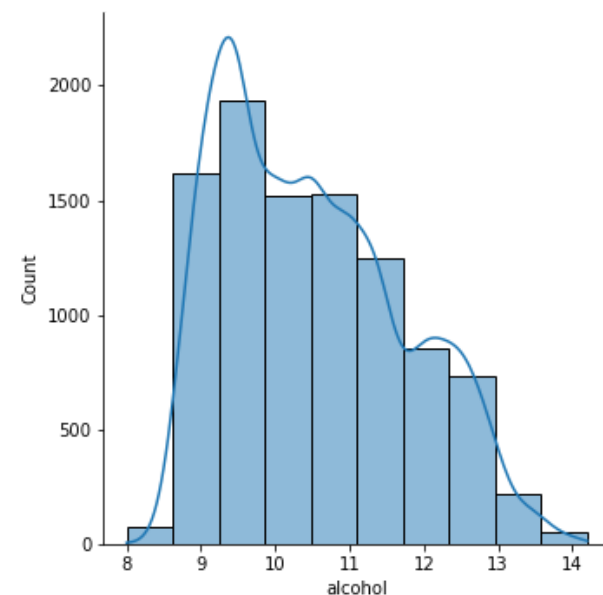


Chris Moffitt
Instructor

Creating a histogram

- The `displot` function has multiple optional arguments
- You can overlay a KDE plot on the histogram and specify the number of bins to use

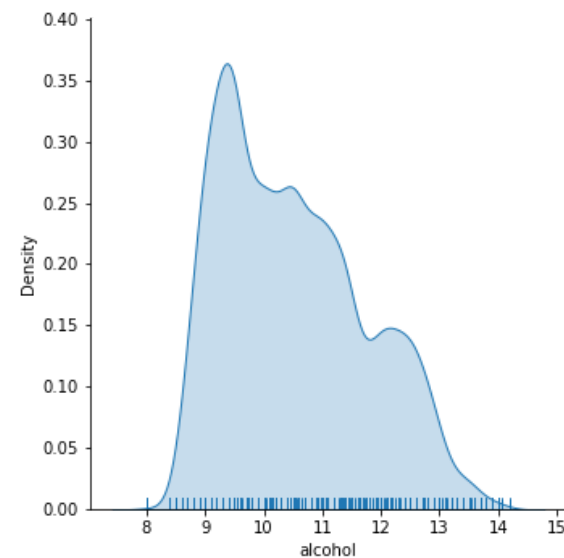
```
sns.displot(df['alcohol'], kde=True, bins=10)
```



Alternative data distributions

- A rug plot is an alternative way to view the distribution of data by including small tickmarks along the x axis
- A kde curve and rug plot can be combined

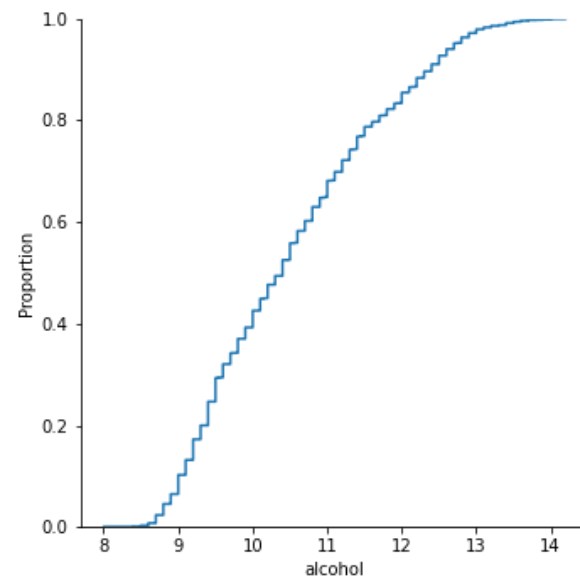
```
sns.displot(df['alcohol'], kind='kde', rug=True, fill=True)
```



Further plot types

- The `displot` function uses several functions including `kdeplot`, `rugplot` and `ecdfplot`
- The `ecdfplot` shows the cumulative distribution of the data

```
sns.displot(df['alcohol'], kind='ecdf')
```



Let's practice!

INTERMEDIATE DATA VISUALIZATION WITH SEABORN

Regression Plots in Seaborn

INTERMEDIATE DATA VISUALIZATION WITH SEABORN

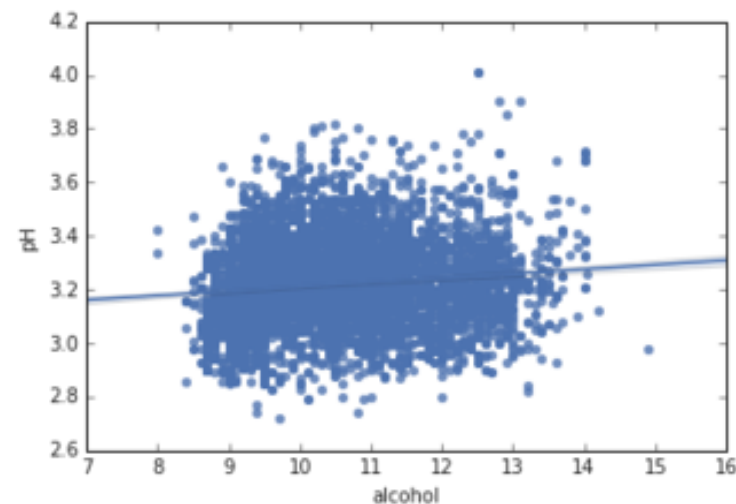


Chris Moffitt
Instructor

Introduction to regplot

- The `regplot` function generates a scatter plot with a regression line
- Usage is similar to the `displot`
- The `data` and `x` and `y` variables must be defined

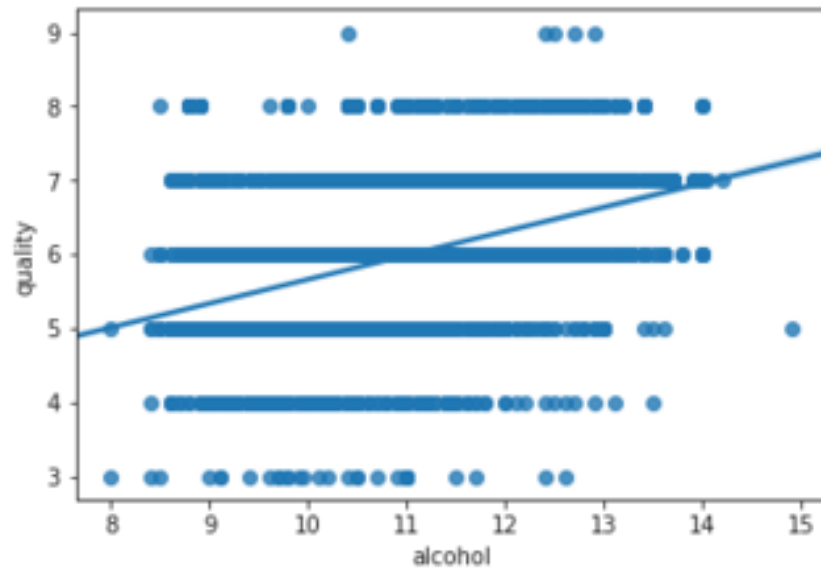
```
sns.regplot(data=df, x="alcohol", y="pH" )
```



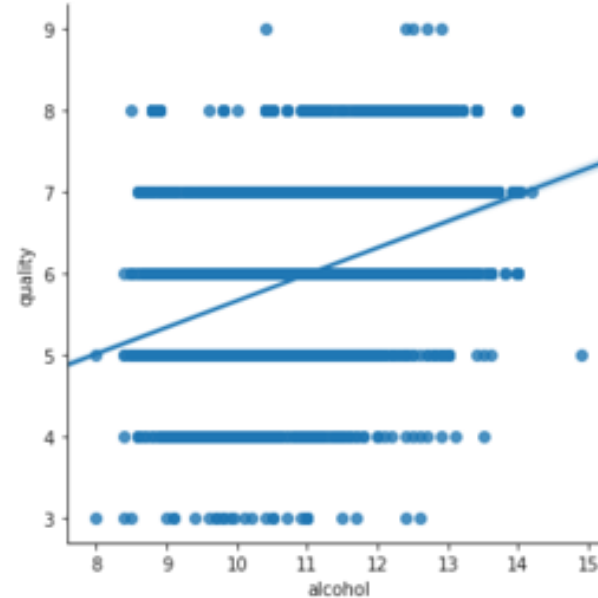
Implot() builds on top of the base regplot()

- `regplot` - low level
- `lmpplot` - high level

```
sns.regplot(data=df,  
            x="alcohol",  
            y="quality")
```



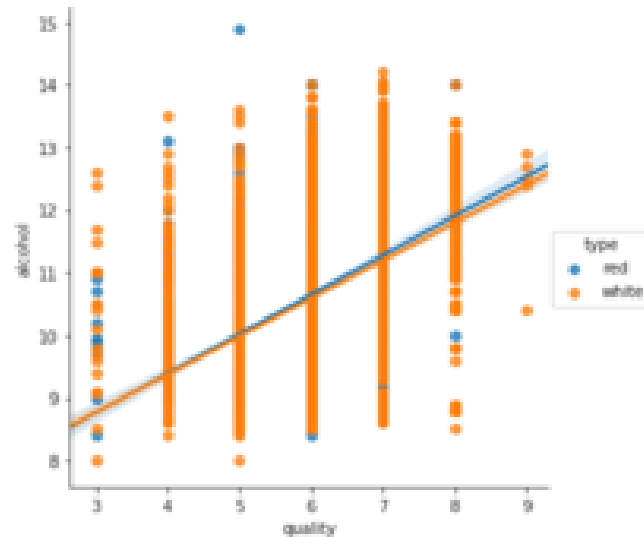
```
sns.lmpplot(data=df,  
            x="alcohol",  
            y="quality")
```



Implot faceting

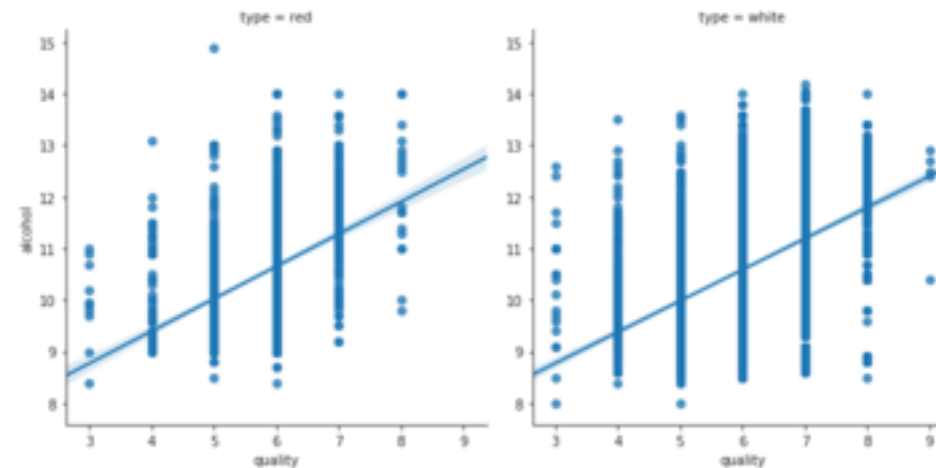
- Organize data by colors (`hue`)

```
sns.lmplot(data=df,  
           x="quality",  
           y="alcohol",  
           hue="type")
```



- Organize data by columns (`col`)

```
sns.lmplot(data=df,  
           x="quality",  
           y="alcohol",  
           col="type")
```



Let's practice!

INTERMEDIATE DATA VISUALIZATION WITH SEABORN