Introduction to Matplotlib and Seaborn

ironhack.school/courses/course-

Lesson Goals

In this lesson we will learn about the basics of Matplotlib and Seaborn

Matplotlib

Introduction

Matplotlib is a data visualization package built on numpy arrays. The package is fairly old, with version 0.1 released in 2003. The main advantage of this package is that it allows for flexible visualization in Python across versions and operating systems.

Getting Started

Since we are using Jupyter notebooks, we start our notebook with this line:

```
%matplotlib inline
```

This ensures that our visualizations will render in the notebook itself.

We now load the matplotlib library:

```
import matplotlib
import matplotlib.pyplot as plt
```

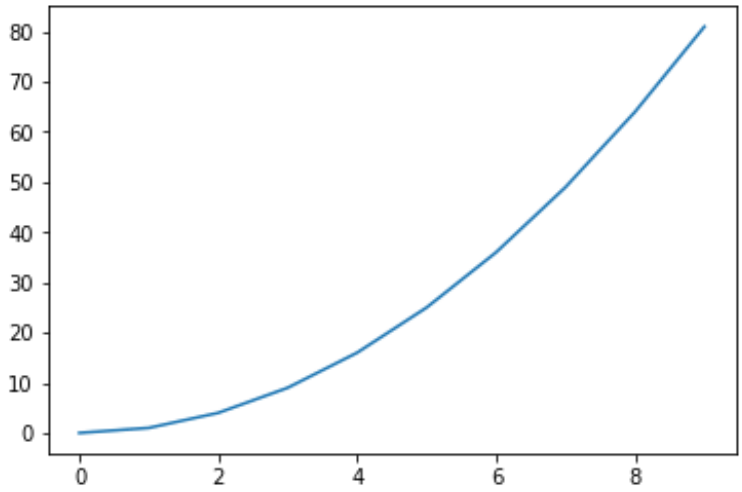
Basic Plots

Line Plot

We can generate two numpy arrays and then plot them as a line plot.

```
import numpy as np


var1 = np.arange(0,10,1)
var2 = var1 ** 2
plt.plot(var1, var2)
```

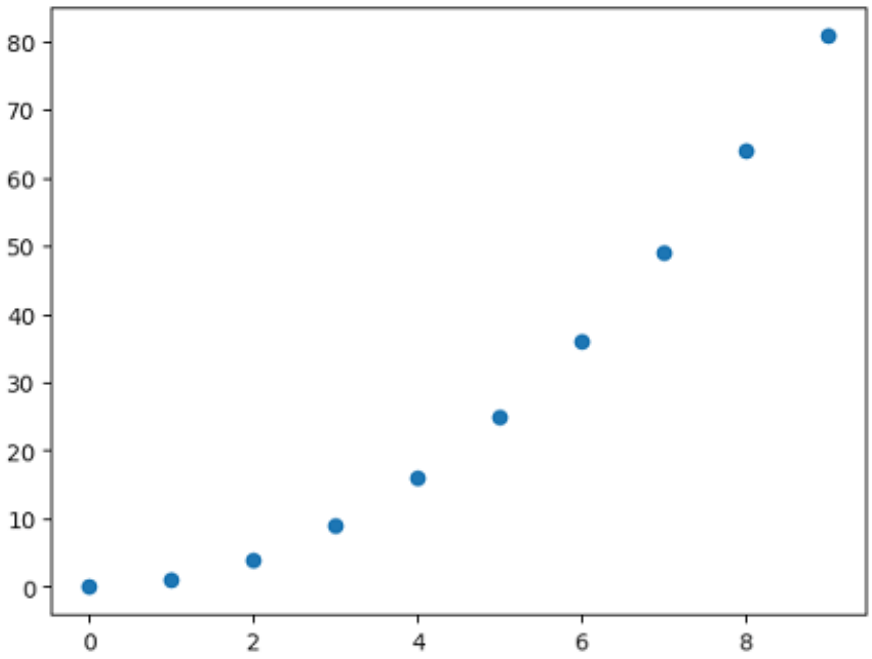


Here we used the numpy function `arange` to generate all integers from 0 to 9. We then plotted those numbers vs. their squares.

Scatter Plot

Here we use the `scatter` function to plot the same two arrays

```
plt.scatter(var1, var2)
```



Seaborn

Introduction

Seaborn is a library that was created to complement matplotlib. It is meant to streamline the amount of code needed to write to create matplotlib visualizations. It also enables us to work with pandas dataframes. This is the main disadvantage of matplotlib that

seaborn resolves.

Getting Started

After setting `%matplotlib inline` and importing matplotlib, we proceed to import seaborn as sns. We can also set the default style to the seaborn style.

```
import seaborn as sns
sns.set()
```

Basic Plots

Line Plots

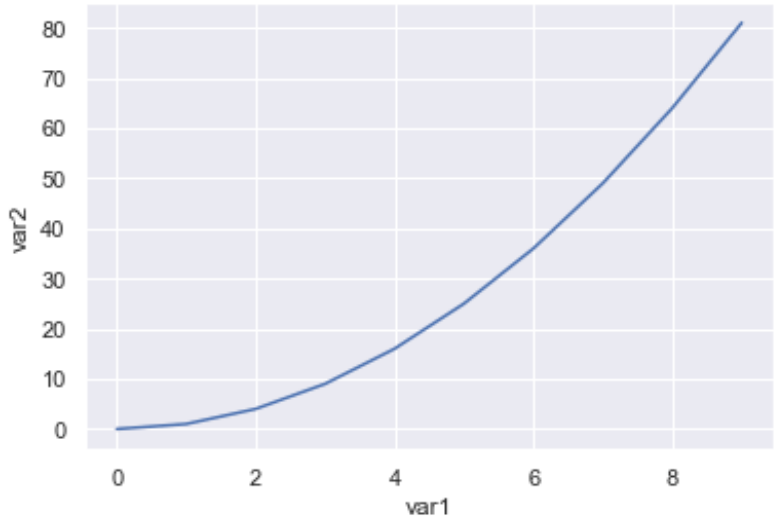
We can create a Pandas DataFrame using the two variables we created earlier.

```
import pandas as pd
vars = pd.DataFrame()
vars['var1'] = var1
vars['var2'] = var2
vars.head()
```

	var1	var2
0	0	0
1	1	1
2	2	4
3	3	9
4	4	16

Now that we have our DataFrame, we can plot the data using seaborn.

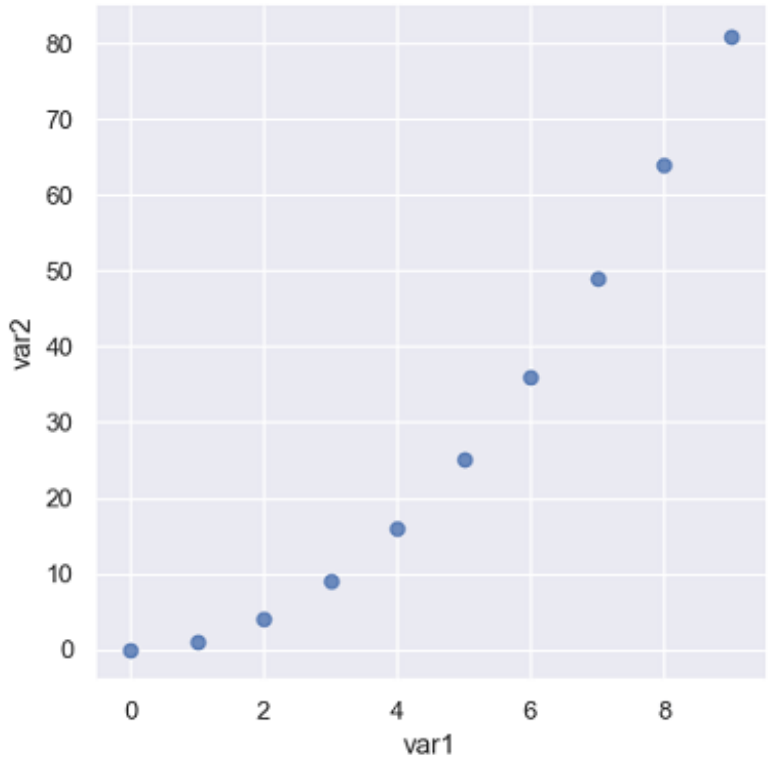
```
sns.lineplot(vars.var1, vars.var2)
```



Scatter Plots

Here we use the seaborn `lmplot` function and set the option to fit the regression line to false.

```
sns.lmplot("var1", "var2", data=vars, fit_reg=False)
```



Using Pandas Directly

Matplotlib has been integrated into Pandas and now we are able to plot matplotlib visualizations directly from Pandas without having to use the matplotlib or seaborn syntax. However, we still need to import matplotlib and set `%matplotlib inline`

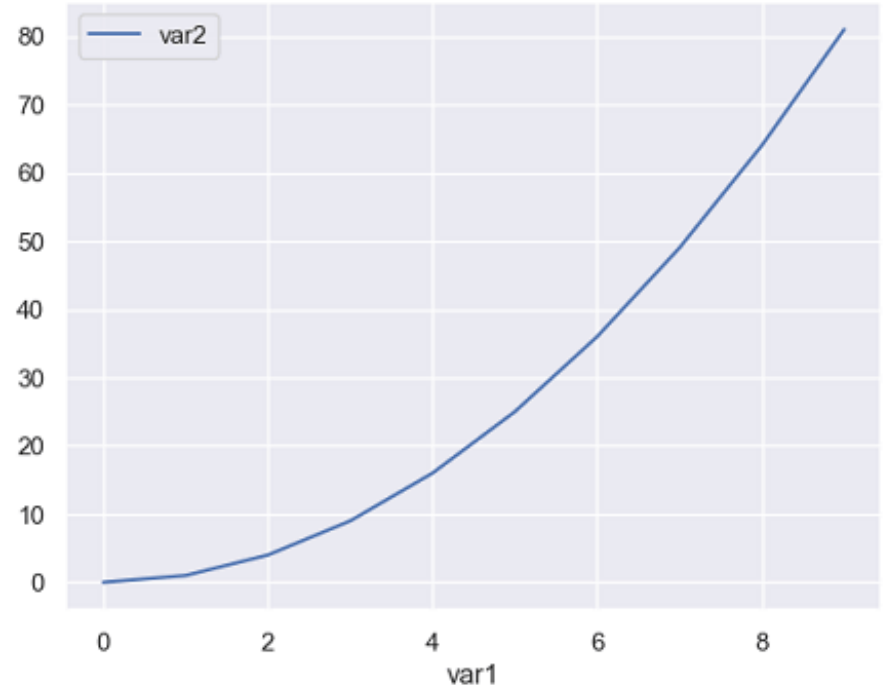
Basic Plots

We plot by accessing `.plot` through the dataframe itself. We then specify what plot we would like and pass the appropriate columns to the function.

Let's use our numpy generated data to plot directly from pandas

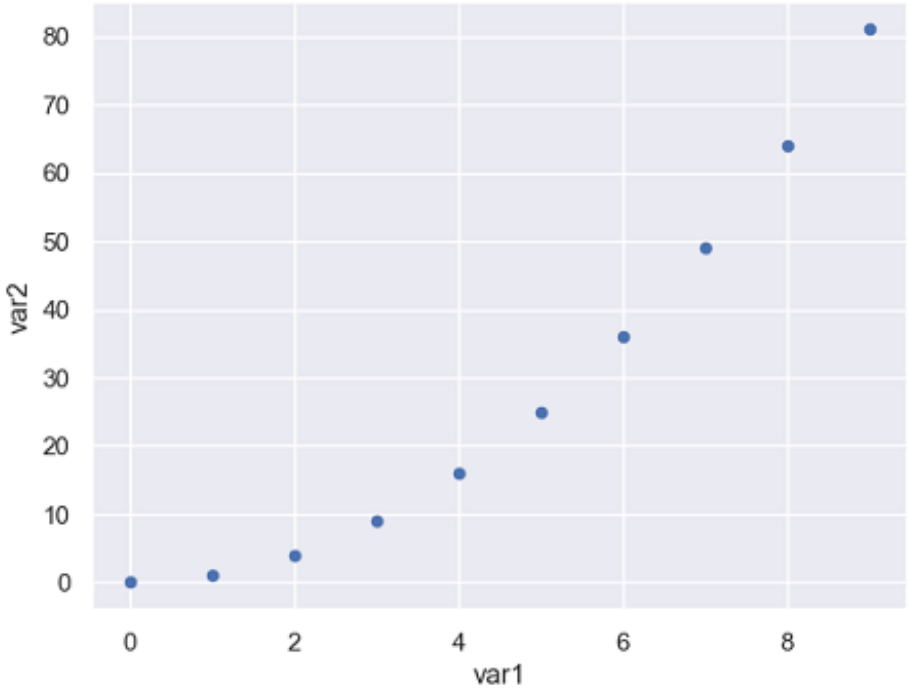
Line Plot

```
vars.plot.line(x="var1", y="var2")
```



Scatter Plot

```
vars.plot.scatter(x="var1", y="var2)
```



Summary

In this lesson we learned how to plot visualizations using three different tools. All three use matplotlib under the hood. However, Seaborn and Pandas have improved on the original Python library and enabled us to visualize DataFrames as well.