Scatter, Line, and Bar Charts using Matplotlib

Matplotlib is a popular Python module that is used to create charts. In this tutorial, I’ll use simple examples to show you how to create Scatter, Line and Bar charts using matplotlib.

But before we begin, here is the general structure that you may use to create your charts using matplotlib:

### ****Scatter plot****

import matplotlib.pyplot as plt

plt.scatter(xAxis,yAxis)

plt.title('title name')

plt.xlabel('xAxis name')

plt.ylabel('yAxis name')

plt.show()

### ****Line chart****

import matplotlib.pyplot as plt

plt.plot(xAxis,yAxis)

plt.title('title name')

plt.xlabel('xAxis name')

plt.ylabel('yAxis name')

plt.show()

### ****Bar chart****

import matplotlib.pyplot as plt

xAxis = [i + 0.5 for i, \_ in enumerate(xAxis)]

plt.bar(xAxis,yAxis)

plt.title('title name')

plt.xlabel('xAxis name')

plt.ylabel('yAxis name')

plt.xticks([i + 0.5 for i, \_ in enumerate(xAxis)], xAxis)

plt.show()

## How to Create Scatter Plots using Matplotlib

Scatter plots are used to depict a relationship between two variables.

For example, let’s say that you want to depict the relationship between:

* Unemployment Rate; and
* Stock Index Price

Before you plot that data, you’ll need to capture it in Python. I’ll use 2 different approaches to capture the data in Python via:

* List
* Pandas DataFrame

### Create Scatter Plot using a List

You can create a simple list, which will contain the values for the Unemployment Rate and Stock Index Price, as follows:

Unemployment\_Rate = [6.1,5.8,5.7,5.7,5.8,5.6,5.5,5.3,5.2,5.2]

Stock\_Index\_Price = [1500,1520,1525,1523,1515,1540,1545,1560,1555,1565]

To create the scatter plot based on that data, you can apply the generic structure that we saw at the beginning of the tutorial. Your full Python code should look like this:

import matplotlib.pyplot as plt

Unemployment\_Rate = [6.1,5.8,5.7,5.7,5.8,5.6,5.5,5.3,5.2,5.2]

Stock\_Index\_Price = [1500,1520,1525,1523,1515,1540,1545,1560,1555,1565]

plt.scatter(Unemployment\_Rate, Stock\_Index\_Price, color='green')

plt.title('Unemployment Rate Vs Stock Index Price', fontsize=14)

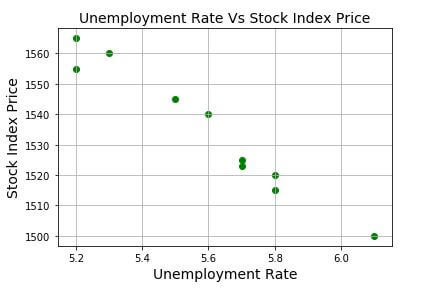
plt.xlabel('Unemployment Rate', fontsize=14)

plt.ylabel('Stock Index Price', fontsize=14)

plt.grid(True)

plt.show()

Once you run the Python code, you’ll get the following Scatter plot:



As indicated earlier, this plot depicts the relationship between the Unemployment Rate and the Stock Index Price.

You may notice that a *negative* relationship exists between those two variables, meaning that when the Unemployment Rate increases, the Stock Index Price falls.

### Create Scatter Plot using Pandas DataFrame

Another way in which you can capture the data in Python is by using pandas DataFrame.

from pandas import DataFrame

Data = {'Unemployment\_Rate': [6.1,5.8,5.7,5.7,5.8,5.6,5.5,5.3,5.2,5.2],

'Stock\_Index\_Price': [1500,1520,1525,1523,1515,1540,1545,1560,1555,1565]

}

df = DataFrame(Data,columns=['Unemployment\_Rate','Stock\_Index\_Price'])

And here is the full Python code to display the Scatter plot using the DataFrame:

from pandas import DataFrame

import matplotlib.pyplot as plt

Data = {'Unemployment\_Rate': [6.1,5.8,5.7,5.7,5.8,5.6,5.5,5.3,5.2,5.2],

'Stock\_Index\_Price': [1500,1520,1525,1523,1515,1540,1545,1560,1555,1565]

}

df = DataFrame(Data,columns=['Unemployment\_Rate','Stock\_Index\_Price'])

plt.scatter(df['Unemployment\_Rate'], df['Stock\_Index\_Price'], color='green')

plt.title('Unemployment Rate Vs Stock Index Price', fontsize=14)

plt.xlabel('Unemployment Rate', fontsize=14)

plt.ylabel('Stock Index Price', fontsize=14)

plt.grid(True)

plt.show()

## How to Create Line Charts using Matplotlib

Line charts are often used to display trends overtime.

For example, imagine that you want to present the Unemployment Rate across time using the below data-set:

### Create Line Chart using a List

You may store the Years and the associated Unemployment Rates as a list:

Year = [1920,1930,1940,1950,1960,1970,1980,1990,2000,2010]

Unemployment\_Rate = [9.8,12,8,7.2,6.9,7,6.5,6.2,5.5,6.3]

Using the Line chart structure that we saw at the beginning of this tutorial, your full Python code should be:

import matplotlib.pyplot as plt

Year = [1920,1930,1940,1950,1960,1970,1980,1990,2000,2010]

Unemployment\_Rate = [9.8,12,8,7.2,6.9,7,6.5,6.2,5.5,6.3]

plt.plot(Year, Unemployment\_Rate, color='red', marker='o')

plt.title('Unemployment Rate Vs Year', fontsize=14)

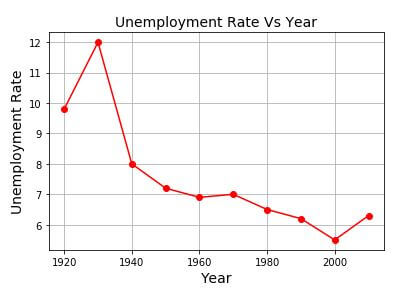
plt.xlabel('Year', fontsize=14)

plt.ylabel('Unemployment Rate', fontsize=14)

plt.grid(True)

plt.show()

And once you run the Python code, you’ll see the trend of the Unemployment across the years:



You’ll notice that based on the data captured, the unemployment rate generally falls over time.

### Create Line Chart using pandas DataFrame

The DataFrame, for our example, should look as follows:

from pandas import DataFrame

Data = {'Year': [1920,1930,1940,1950,1960,1970,1980,1990,2000,2010],

'Unemployment\_Rate': [9.8,12,8,7.2,6.9,7,6.5,6.2,5.5,6.3]

}

df = DataFrame(Data,columns=['Year','Unemployment\_Rate'])

Putting everything together:

from pandas import DataFrame

import matplotlib.pyplot as plt

Data = {'Year': [1920,1930,1940,1950,1960,1970,1980,1990,2000,2010],

'Unemployment\_Rate': [9.8,12,8,7.2,6.9,7,6.5,6.2,5.5,6.3]

}

df = DataFrame(Data,columns=['Year','Unemployment\_Rate'])

plt.plot(df['Year'], df['Unemployment\_Rate'], color='red', marker='o')

plt.title('Unemployment Rate Vs Year', fontsize=14)

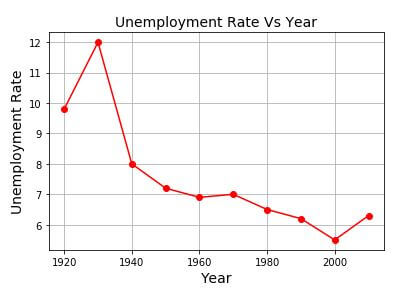
plt.xlabel('Year', fontsize=14)

plt.ylabel('Unemployment Rate', fontsize=14)

plt.grid(True)

plt.show()

You should get the same Line chart when running the code in Python:



## How to Create Bar Charts using Matplotlib

Bar charts are used to display categorical data.

Let’s say that you want to use a Bar chart to display the GDP Per Capita for a sample of 5 countries:

### Create a Bar chart using a List

First, create the list as follows:

Country = ['USA','Canada','Germany','UK','France']

GDP\_Per\_Capita = [45000,42000,52000,49000,47000]

Notice that the Country column contains text/strings (wrapped around quotations for each value), while the GDP\_Per\_Capita column contains numerical values without the quotations.

Since our data-set contains both text and numerical values, you’ll need to add the following syntax:

xAxis = [i + 0.5 for i, \_ in enumerate(Country)]

Without that portion, you’ll face the following error in Python:

**unsupported operand type(s) for -: ‘str’ and ‘float’**

You’ll also need to incorporate the following section when depicting the bar chart:

plt.xticks([i + 0.5 for i, \_ in enumerate(Country)], Country)

When you put all the components together, your full code to create a Bar chart should look like this:

import matplotlib.pyplot as plt

Country = ['USA','Canada','Germany','UK','France']

GDP\_Per\_Capita = [45000,42000,52000,49000,47000]

xAxis = [i + 0.5 for i, \_ in enumerate(Country)]

plt.bar(xAxis, GDP\_Per\_Capita, color='teal')

plt.title('Country Vs GDP Per Capita', fontsize=14)

plt.xlabel('Country', fontsize=14)

plt.ylabel('GDP Per Capita', fontsize=14)

plt.xticks([i + 0.5 for i, \_ in enumerate(Country)], Country)

plt.show()