Matplotlib Basics

28.09.2019

Python Notes

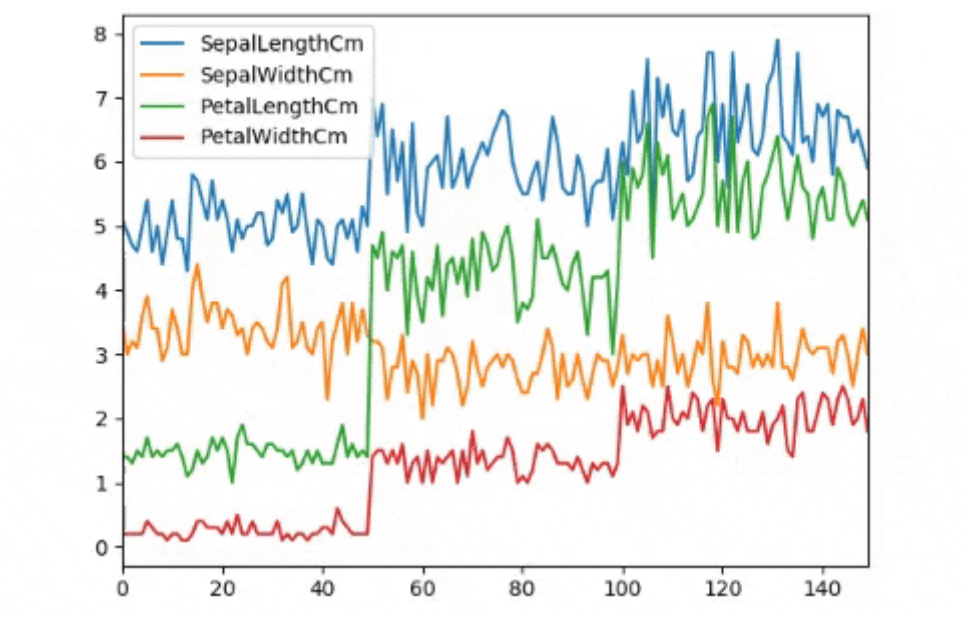
Pandas Review

```
import pandas as pd

# get iris.csv file as a dataFrame
df = pd.read_csv("iris.csv")

#see features
df.columns

#see whole Species column as a series
df.Species

#see how many unique elements inside Species column
df.Species.unique() #returns as an string array

#get a general info
df.info()

#let's create a new dF by collecting Iris-setosa species only from the original dF. Filtering:
setosadf = df[df.Species == "Iris-setosa"]
```

```
#let's create a new dF by collecting Iris-versicolor species only from the original dF
versicolordf = df[df.Species == "Iris-versicolor"]

#Compare numeric features of both dataFrames:
print(setosadf.describe())
print(versicolordf.describe())
```

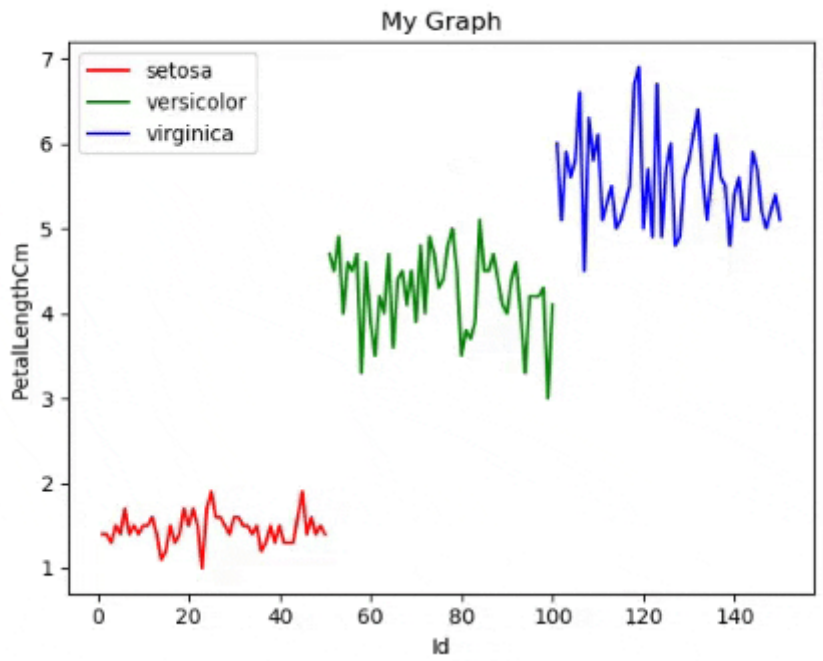
Line Plot

```
import matplotlib.pyplot as plt

#The Id column is not very useful for plotting. Let's drop it.
df1 = df.drop(["Id"],axis=1)

# Plot a line graph for each feature. Indexes vs Values.
df1.plot()
plt.show()
```




```
# Obtain 3 different dFs each has different Species feature
setosadf = df[df.Species == "Iris-setosa"]
versicolordf = df[df.Species == "Iris-versicolor"]
virginicadf = df[df.Species == "Iris-virginica"]

# plot Id series vs PetalLength series
plt.figure()
plt.plot(setosadf.Id, setosadf.PetalLengthCm, color="red", label="setosa")
plt.plot(versicolordf.Id, versicolordf.PetalLengthCm, color="green", label="versicolor")
plt.plot(virginicadf.Id, virginicadf.PetalLengthCm, color="blue", label="virginica")
```

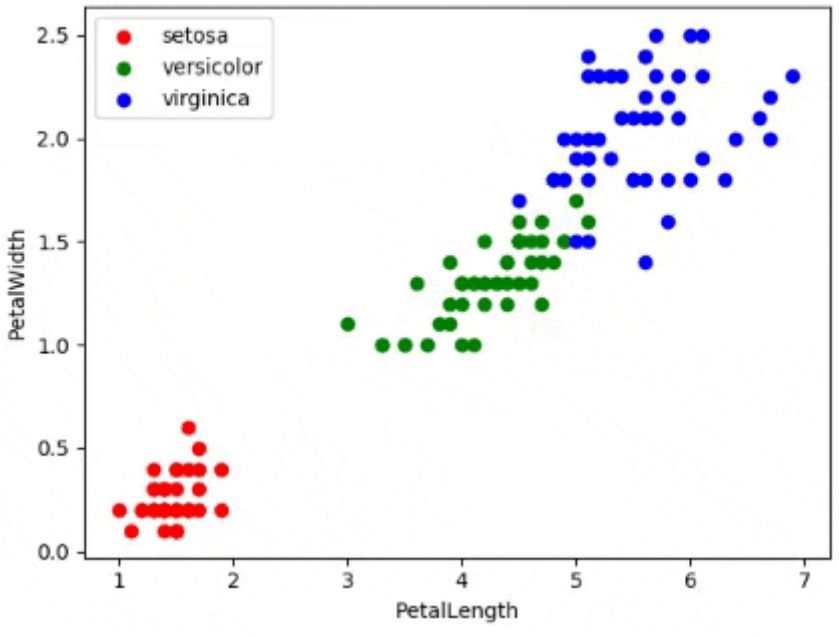
```
# Labeling
plt.title("My Graph")
plt.xlabel("Id")
plt.ylabel("PetalLengthCm")
plt.legend()
plt.show()
```



Scatter Plot

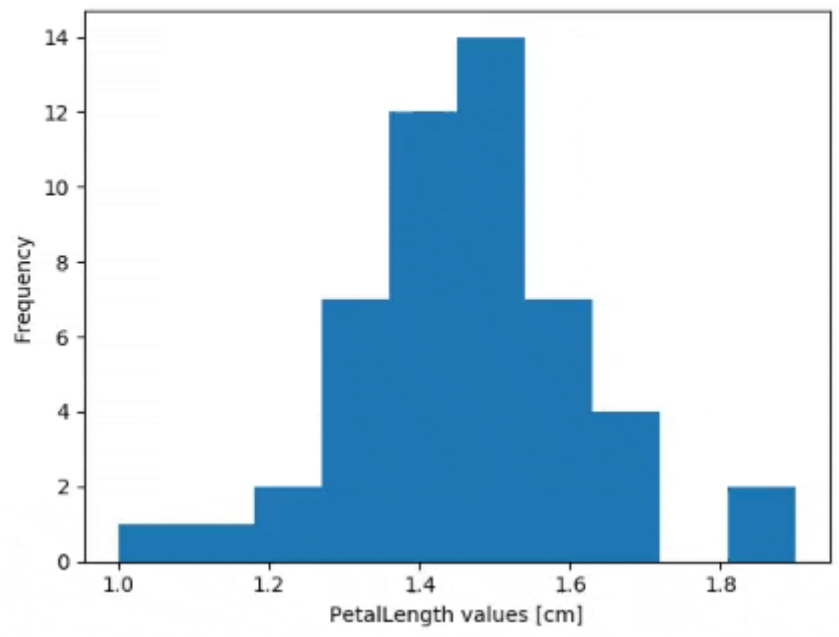
```
# Obtain 3 different dFs each has different Species feature
setosadf = df[df.Species == "Iris-setosa"]
versicolordf = df[df.Species == "Iris-versicolor"]
virginicadf = df[df.Species == "Iris-virginica"]

# Scater plot
plt.scatter(setosadf.PetalLengthCm, setosadf.PetalWidthCm, color="red", label="setosa")
plt.scatter(versicolordf.PetalLengthCm,      versicolordf.PetalWidthCm,      color="green",
label="versicolor")
plt.scatter(virginicadf.PetalLengthCm,      virginicadf.PetalWidthCm,      color="blue",
label="virginica")

# Labeling
plt.xlabel("PetalLength")
plt.ylabel("PetalWidth")
plt.legend()
```

```
plt.show()
```



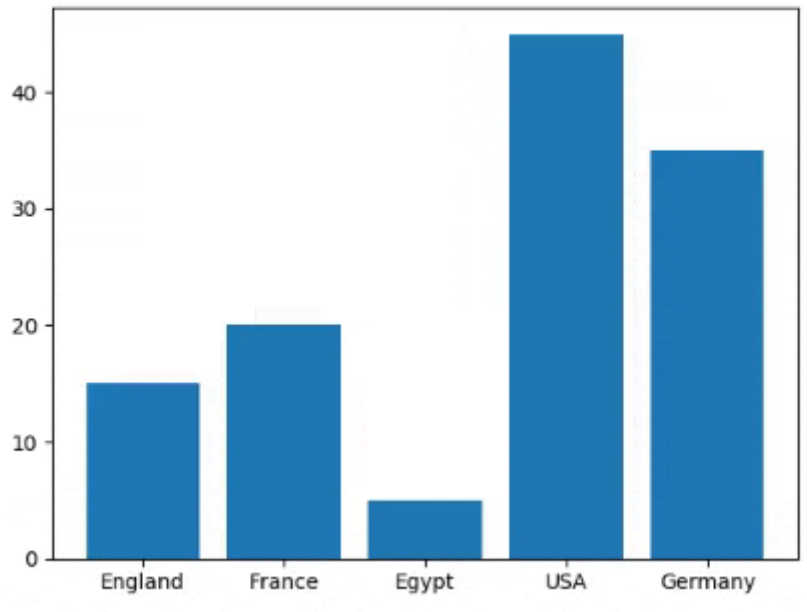
Histogram

```
#Useful to plot how frequently a data occur.

# Hist plot and labels
plt.hist(setosadf.PetalLengthCm)
plt.xlabel("PetalLength values [cm]")
plt.ylabel("Frequency")
```



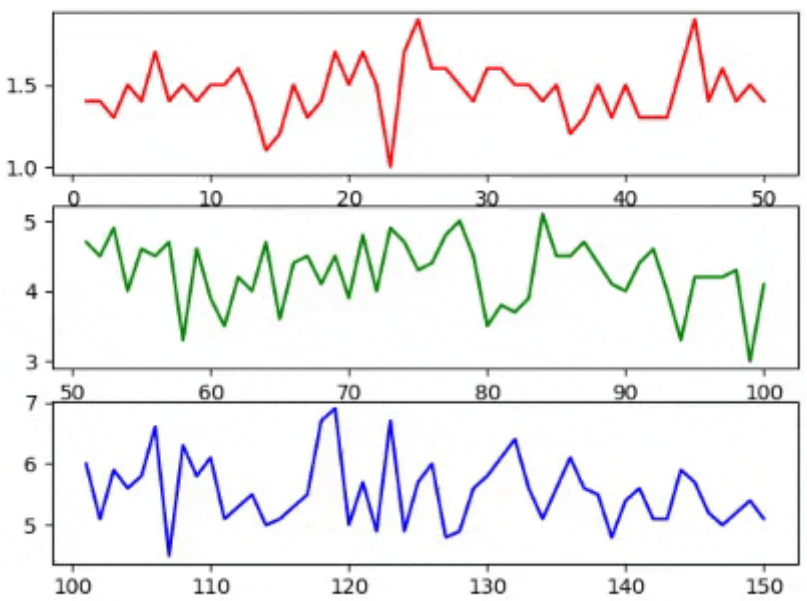
Bar Plot

```
x = ["England","France","Egypt","USA","Germany"]
y = [15,20,5,45,35]
plt.bar(x,y)
```



Subplots

```
# Obtain 3 different dFs each has different Species feature
setosadf = df[df.Species == "Iris-setosa"]
versicolordf = df[df.Species == "Iris-versicolor"]
virginicadf = df[df.Species == "Iris-virginica"]

# Divide a figure into 3 rows 1 column.
plt.subplot(3,1,1)
plt.plot(setosadf.Id, setosadf.PetalLengthCm, color="red", label="setosa")
plt.subplot(3,1,2)
plt.plot(versicolordf.Id, versicolordf.PetalLengthCm, color="green", label="versicolor")
plt.subplot(3,1,3)
plt.plot(virginicadf.Id, virginicadf.PetalLengthCm, color="blue", label="virginica")
```



More blogs