# 01-Customizing-Plots

October 19, 2022

---



For more information, visit us at www.pieriandata.com

## 1 Customizing Pandas Plots

In this section we'll show how to control the position and appearance of axis labels and legends. For more info on the following topics visit https://pandas.pydata.org/pandas-docs/stable/generated/pandas.DataFrame.plot.html

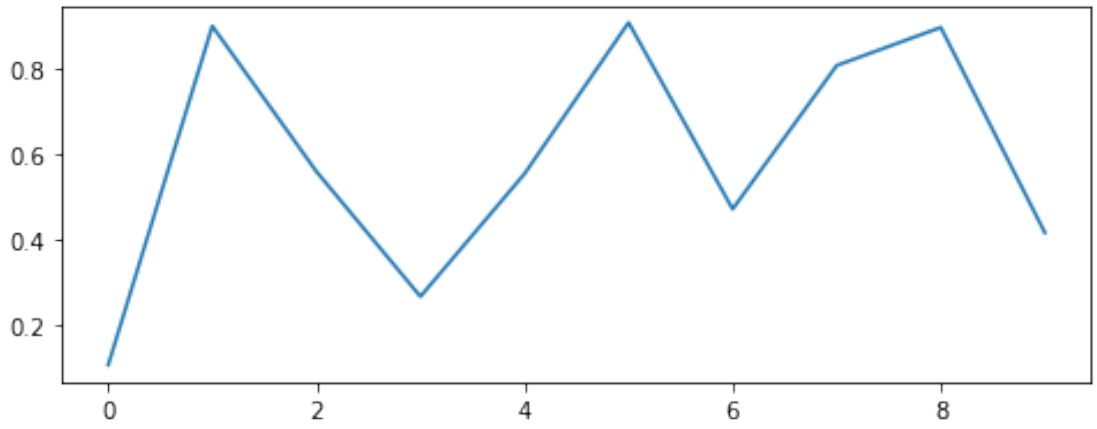
[1]:
```
import pandas as pd
%matplotlib inline

df2 = pd.read_csv('df2.csv')
```

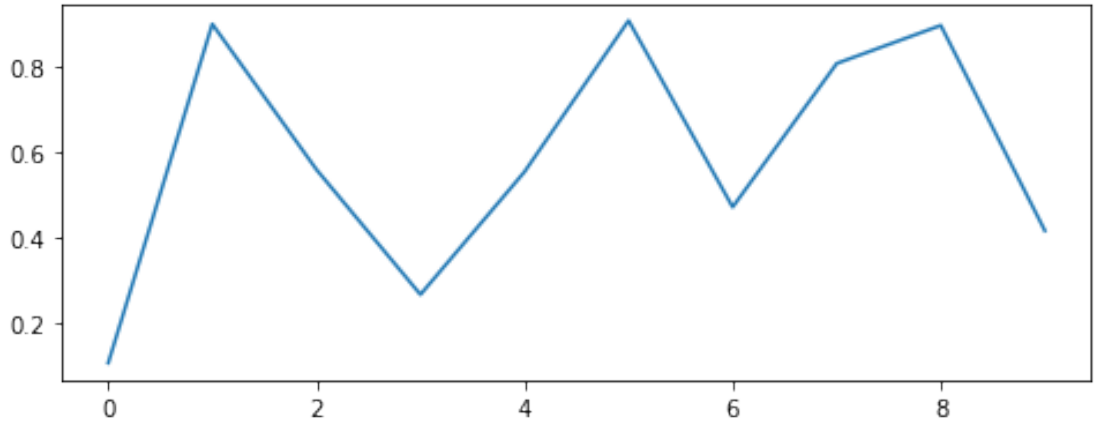
### 1.1 Colors, Widths and Linestyles

The pandas .plot() method takes optional arguments that allow you to control linestyles, colors, widths and more.

[2]:
```
# START WITH A SIMPLE LINE PLOT
df2['c'].plot(figsize=(8,3));
```

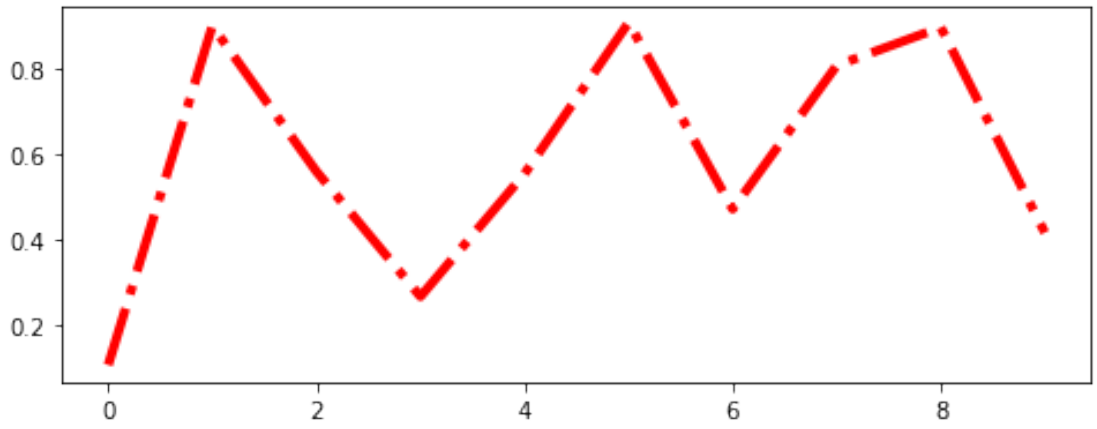



[3]:
```
df2['c'].plot.line(figsize=(8,3));
```




```
<tr><th>PROPERTY</th><th>CODE</th><th>VALUE</th><th>EFFECT</th></tr>
<tr><td>linestyle</td><td><tt>ls</tt></td><td><tt>'-'</tt></td><td>solid line (default)</td></t
<tr><td>linestyle</td><td><tt>ls</tt></td><td><tt>'--'</tt></td><td>dashed line</td></tr>
<tr><td>linestyle</td><td><tt>ls</tt></td><td><tt>'-.'</tt></td><td>dashed/dotted line</td></tr
<tr><td>linestyle</td><td><tt>ls</tt></td><td><tt>':'</tt></td><td>dotted line</td></tr>
<tr><td>color</td><td><tt>c</tt></td><td>string</td><td></td></tr>
<tr><td>linewidth</td><td><tt>lw</tt></td><td>float</td><td></td></tr>
```

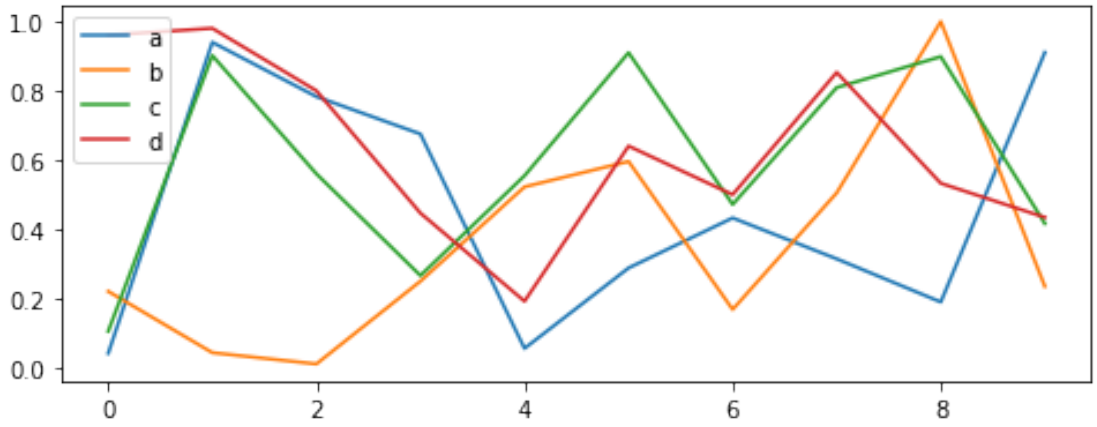
[4]:
```
df2['c'].plot.line(ls='-.', c='r', lw='4', figsize=(8,3));
```




For more on linestyles, click here.

## 1.2 Adding Titles and Axis Labels

[5]:
```
# START WITH A SIMPLE MULTILINE PLOT
df2.plot(figsize=(8,3));
```




Before we tackle the issue of legend placement, let's add a title and axis labels.

In the previous section we learned how to pass a title into the pandas .plot() function; as it turns out, we can't pass axis labels this way.

### 1.2.1 Object-oriented plotting

When we call df.plot(), pandas returns a matplotlib.axes.AxesSubplot object. We can set labels on that object so long as we do it in the same jupyter cell. Setting an autoscale is done the same way.

[6]:
```
title='Custom Pandas Plot'
ylabel='Y-axis data'
xlabel='X-axis data'

ax = df2.plot(figsize=(8,3),title=title)
ax.set(xlabel=xlabel, ylabel=ylabel)
ax.autoscale(axis='x',tight=True);
```


NOTE: The plot shrinks a bit so that the figure size can accommodate the new axis labels.

## 1.3 Plot Legend Placement

Pandas tries to optimize placement of the legend to reduce overlap on the plot itself. However, we can control the placement ourselves, and even place the legend outside of the plot. We do this through the .legend() method. [reference]

For starters we can pass a location code. .legend(loc=1) places the legend in the upper-right corner of the plot.Alternatively we can pass a location string: .legend(loc='upper right') does the same thing.

LOCATION CODE

LOCATION STRING

0

'best'

1

'upper right'

2

'upper left'

3

'lower left'

4

'lower right'

5

'right'

6

'center left'

7

'center right'

8

'lower center'

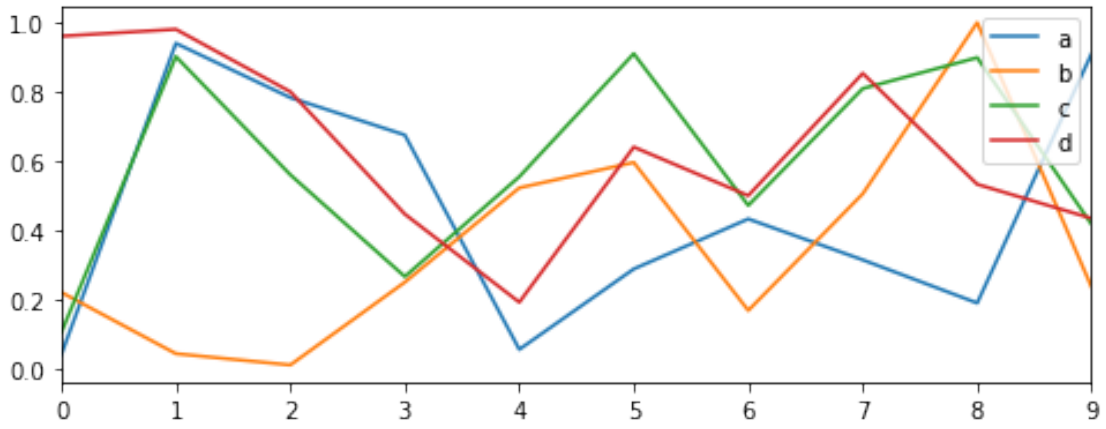
9

'upper center'

10

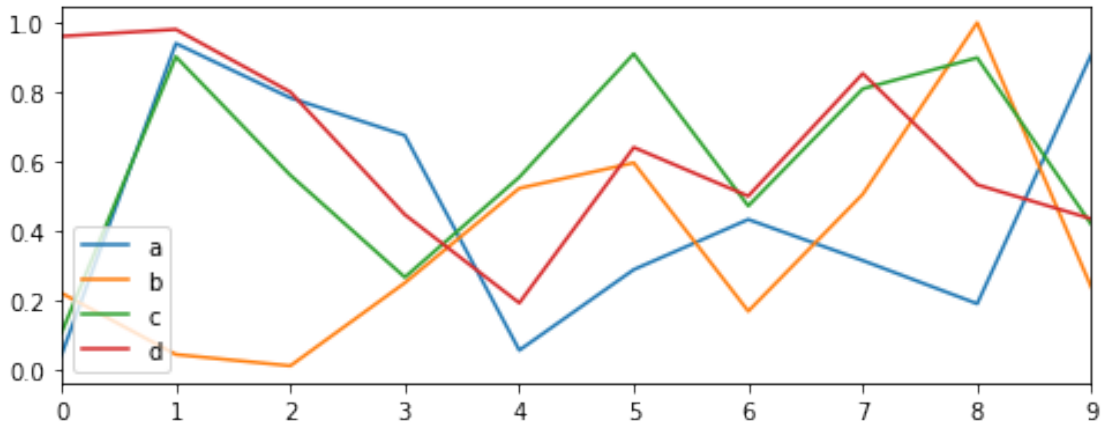
'center'

[7]:
```
ax = df2.plot(figsize=(8,3))
ax.autoscale(axis='x',tight=True)
ax.legend(loc=1);
```

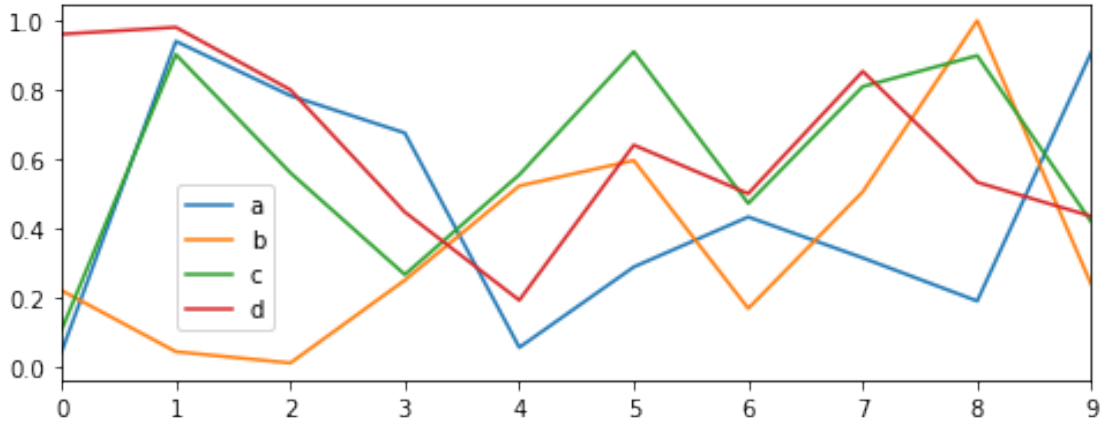


We can pass a second argument, bbox_to_anchor that treats the value passed in through loc as an anchor point, and positions the legend along the x and y axes based on a two-value tuple.

[8]:
```
# FIRST, PLACE THE LEGEND IN THE LOWER-LEFT
ax = df2.plot(figsize=(8,3))
ax.autoscale(axis='x',tight=True)
ax.legend(loc=3);
```



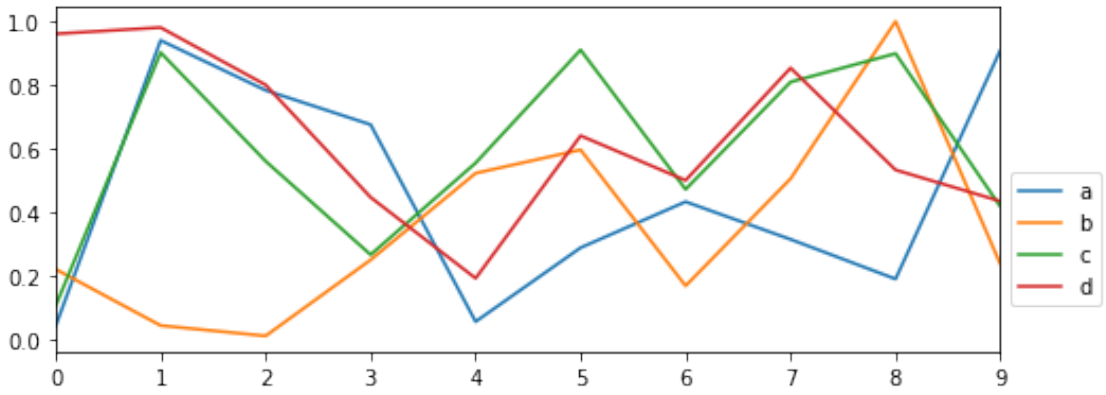

[9]:
```
# NEXT, MOVE THE LEGEND A LITTLE TO THE RIGHT AND UP
ax = df2.plot(figsize=(8,3))
ax.autoscale(axis='x',tight=True)
ax.legend(loc=3, bbox_to_anchor=(0.1,0.1));
```



### 1.3.1 Placing the Legend Outside the Plot

In the above plot we passed (0.1,0.1) as our two-item tuple. This places the legend slightly to the right and slightly upward.To place the legend outside the plot on the right-hand side, pass a value greater than or equal to 1 as the first item in the tuple.

[10]:
```
ax = df2.plot(figsize=(8,3))
ax.autoscale(axis='x',tight=True)
ax.legend(loc=3, bbox_to_anchor=(1.0,0.1));
```




That's it! Feel free to experiment with different settings and plot types.