Terms

The terms will be used to refer to:
- df = Pandas DataFrame
- series = Pandas Series
- data = Pandas DataFrame or Series

Plot whith Series and DataFrames

`- series.plot()`	Series
`- datafr ame.pl ot( x=' Non e', y=' None')`	DataFrame
`- data.p lot.<k ind >()`	Another method
`'bar' or 'barh', 'hist', 'box', 'kde' or 'density' , 'hexbin', 'pie' and 'scatter'`	Kinds

Bar Plot

`- data.p lot.bar() /.barh()`	Bar plot Horizontal plot
`- data.p lot.ba r(s tac ked = True)`	Stacked bar plot

Area Plots

`- data.p lot.area()`	Area plot
`- data.p lot.ar ea( sta cke d=F alse)`	Non-St-acked area plot

Pie plot

`- series.pl ot.p ie()`	Pie plot for Series
`- DataFr ame.pl ot.p ie (su bpl ots =True)`	Pie plot for DataFrame
`- series.pl ot.p ie (la bels= ['A',' B', 'C'], colors= ['r',' b', 'g'], autopct= '%.2f')`	Wedge labels

It's valid:
fontsize and figsize

Scatter plot

```
-DataF ram e.p lot.sc att er(x=' ', y=' ')
```

```
-ax= df.plo t.s cat ter (x= 'A' ,y='B' , color= 'None', label= 'Gr oup
```

```
-df.pl ot.s ca tte r(x ='C ',y ='D', color= 'Ot her', label= 'Gr oup
2', ax= ax)
```

Histograms

`- data.p lot.hist()`	Histogram plot
`- data.p lot.hi st( sta cke d=True, bins=10)`	Stacked and bins size
`- data.p lot.hi st( ori ent ati on= 'ho riz ontal', cumula tiv e=True)`	Horizontal and cumulative
`data.d iff ().h is t(c olo r='g', alpha=0.5)`	Subplots histograms

Box Plots

`- data.p lot.box()`	Box p
`- dict={ 'bo xes':' ','whi ske rs':' ', 'media ns':' ', 'caps': ' '}`	Color Boxe
`- data.p lot.bo x(c olo r=dict)`	
`- data.p lot.bo x(v ert =False)`	Horizo box p
`- df.box plo t(b y=' col umn')`	Group
`- df.box plo t(c olu mn=[' ',' '], by=[' ',' '])`	
`- df.gro upb y(' g').bo xplot()`	for ran choic

The "choice random" is:

```
-np.r and om_ cho ice ([' A',
'B' ],s ize =50])
```

Hexagonal bin plot

```
- DataFr ame.pl ot.h ex bin (x= 'N
```

```
- DataFr ame.pl ot.h ex bin (x= 'N
on= np.max)
```

```
- DataFr ame.pl ot.h ex bin (x= 'N
```

Density plot

`- data.p lot.kde()`	Density plot

Plot for data .CSV

```
> data= pd.rea d_c sv( 'Name or di
```

Plotting Tools from Pandas Plotting

```
> import pandas.pl otting as pdt
```

Scatter matrix plot

```
- pdt.sc at t e r_ mat ri x(
e = ( 6, 6)')
```

Lag plot

```
- pdt.la g_p lot (se ries)
```

Autocorrelation plot

```
- pdt.au toc orr ela tio n_p lot (
```

Bootstrap plot

```
- pdt.bo ots tra p_p lot (se ries,
gr een')
```

Andrews curves

```
- pdt.an dre ws_ cur ves (data, 'c
```

Parallel coordinates

```
- pdt.pa ral lel _co ord ina tes (
names')
```

RadViz

```
- pdt.ra dvi z(data, 'column name
```

Plot formatting

Plot style

```
- series.pl ot( sty le= 'k--')
```

Controlling the legend

```
- DataFr ame.pl ot( legend= False)
```

Color map

```
- DataFr ame.pl ot( col ormap=' ')
```

Scales (logarithmic)

```
- data.p lot (logy= True) or logx or
g
```

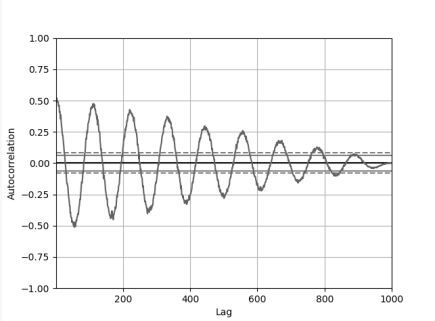
Plotting on a seconday y-axis

```
- DataFr ame.co lum n1.p lot()
- DataFr ame.co lum n2.p lo t(s eco nda ry
_ y=True)
```

Suppressing tick resolution adjustament

```
- data.p lot (x_ compat= True)
```

j



Subplots

`- data.p lot (su bpl ots =True)`	Subplots
`- data.p lot (su bplots= True, layout= (2 ,3)`	Multiple axes

It´s valid:
figsize and sharex

Plotting with errors bars

```
DataFr ame.pl ot.b ar (ye rr= df_err, xerr= df1_err, caps
ize=3)
```

df_err and *df1_err* are DataFrame of the errors of X and Y

Plotting tables

```
- ax.get _xa xis ().s et _vi sib le( False)
- DataFr ame.pl ot( table= True, ax=ax)
```

Adds table to:

```
- fig, ax= plt.su bpl ots (1,1)
- pdt.ta ble(ax, DataFrame, loc='upper right', colWid ths =[0.2, 0.2, 0.
2])
- DataFr ame.pl ot(ax= ax)
```