

Plotly is a library that allows you to create interactive plots that you can use in dashboards or websites (you can save them as html files or static images).

Installation

In order for this all to work, you'll need to install plotly and cufflinks to call plots directly off of a pandas dataframe. These libraries are not currently available through **conda** but are available through **pip**. Install the libraries at your command line/terminal using:

```
!pip install plotly
!pip install cufflinks
```

**** NOTE: Make sure you only have one installation of Python on your computer when you do this, otherwise the installation may not work. ****

Imports and Set-up

Entrée [2]:

```
!pip install plotly
!pip install cufflinks
```

...

Entrée [4]:

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
```

Entrée [5]:

```
from plotly.offline import iplot
import plotly as py
import plotly.tools as tls
```

Entrée [6]:

```
import cufflinks as cf
```

Using Cufflinks and iplot()

- line
- scatter
- bar
- box
- spread
- ratio
- heatmap
- surface
- histogram

- bubble

Entrée [7]:

```
print(py.__version__)
```

3.10.0

Entrée [12]:

```
df = pd.DataFrame()  
help(df.iplot)
```

Help on method `_iplot` in module `cufflinks.plotlytools`:

```
_iplot(kind='scatter', data=None, layout=None, filename='', sharing=None, title='', xTitle='', yTitle='', zTitle='', theme=None, colors=None, colorscale=None, fill=False, width=None, dash='solid', mode='', interpolation='linear', symbol='circle', size=12, barmode='', sortbars=False, bargap=None, bargroupgap=None, bins=None, histnorm='', histfunc='count', orientation='v', boxpoints=False, annotations=None, keys=False, bestfit=False, bestfit_colors=None, mean=False, mean_colors=None, categories='', x='', y='', z='', text='', gridcolor=None, zerolinecolor=None, margin=None, labels=None, values=None, secondary_y='', secondary_y_title='', subplots=False, shape=None, error_x=None, error_y=None, error_type='data', locations=None, lon=None, lat=None, asFrame=False, asDates=False, asFigure=False, asImage=False, dimensions=None, asPlot=False, asUrl=False, online=None, **kwargs) method of pandas.core.frame.DataFrame instance
```

Returns a plotly chart either as inline chart, image of Figure object

Parameters:

Entrée [13]:

```
tls.embed('https://plot.ly/~cufflinks/8')
```

C:\ProgramData\Anaconda3\lib\site-packages\IPython\core\display.py:689: User
Warning:

Consider using IPython.display.IFrame instead

Out[13]:

Entrée [14]:

```
py.offline.init_notebook_mode(connected=True)
```

Entrée [15]:

```
cf.go_offline()
```

Entrée [67]:

```
df = pd.DataFrame(np.random.randn(100,3), columns = ['A', 'B', 'C'])
df.head()
df['A'] = df['A'].cumsum() + 20
df['B'] = df['B'].cumsum() + 20
df['C'] = df['C'].cumsum() + 20
```

Entrée [68]:

```
df.shape
```

Out[68]:

```
(100, 3)
```

Entrée [69]:

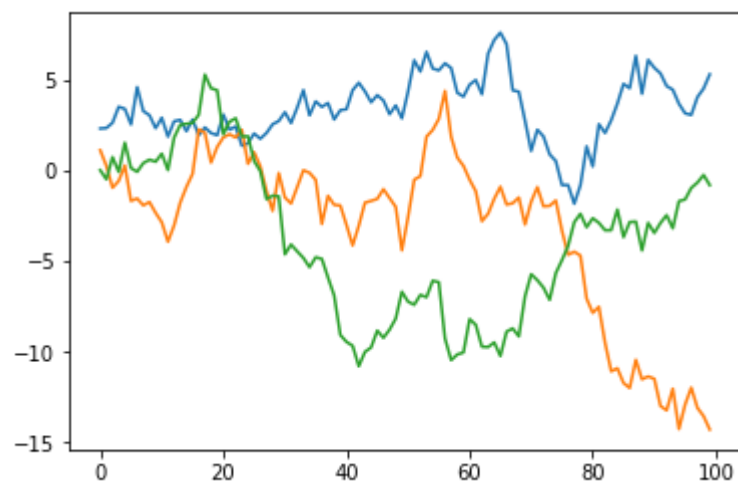
```
df.iplot()
```

Entrée [23]:

```
plt.plot(df)
```

Out[23]:

```
[<matplotlib.lines.Line2D at 0x1f46a1f14a8>,  
<matplotlib.lines.Line2D at 0x1f46a1f15f8>,  
<matplotlib.lines.Line2D at 0x1f46a1f1748>]
```

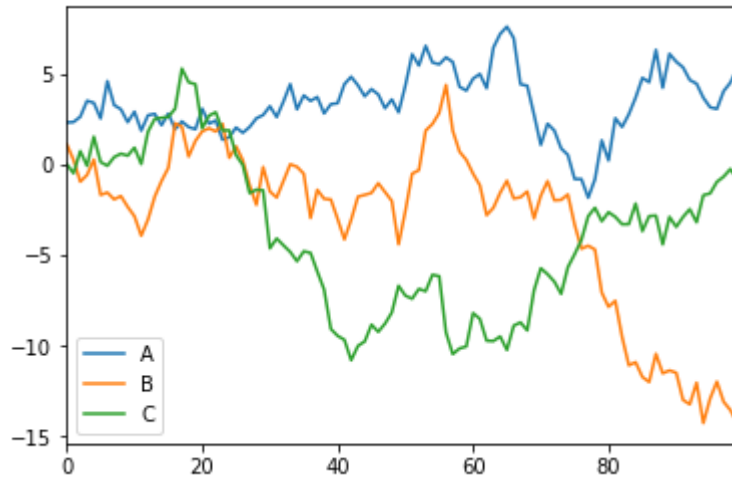


Entrée [24]:

```
df.plot()
```

Out[24]:

<matplotlib.axes._subplots.AxesSubplot at 0x1f46a25acf8>



Entrée [28]:

```
df.iplot(x = 'A', y = 'B', mode = 'markers', size = 25)
```

Entrée [29]:

```
titanic = sns.load_dataset('titanic')  
titanic.head()
```

Out[29]:

	survived	pclass	sex	age	sibsp	parch	fare	embarked	class	who	adult_male
0	0	3	male	22.0	1	0	7.2500	S	Third	man	True
1	1	1	female	38.0	1	0	71.2833	C	First	woman	False
2	1	3	female	26.0	0	0	7.9250	S	Third	woman	False
3	1	1	female	35.0	1	0	53.1000	S	First	woman	False
4	0	3	male	35.0	0	0	8.0500	S	Third	man	True

Entrée [44]:

```
titanic.iplot(kind = 'bar', x = 'sex', y = 'survived', title = 'Survived', xTitle='Sex', yT
```

Entrée [41]:

```
titanic['sex'].value_counts()
```

Out[41]:

```
male      577
female    314
Name: sex, dtype: int64
```

Entrée [54]:

```
cf.getThemes()
```

Out[54]:

```
['ggplot', 'pearl', 'solar', 'space', 'white', 'polar', 'henanigans']
```

Entrée [62]:

```
cf.set_config_file(theme='white')
df.iplot(kind = 'bar', barmode='stack', bargap=0.5)
```


Entrée [50]:

```
df.iplot(kind = 'bar', barmode='stack', bargap=0.5)
```

Entrée [51]:

```
df.iplot(kind = 'barh', barmode='stack', bargap=0.5)
```

Entrée []:

Entrée [63]:

```
df.iplot(kind = 'box')
```

Entrée [70]:

```
df.iplot()
```

Entrée [76]:

```
df.iplot(kind = 'area', fill = True)
```

Entrée [81]:

```
df3 = pd.DataFrame({'X': [10,20,30,20,10], 'Y': [10, 20, 30, 20, 10], 'Z': [10, 20, 30, 20, 10]},  
df3.head())
```

Out[81]:

	X	Y	Z
0	10	10	10
1	20	20	20
2	30	30	30
3	20	20	20
4	10	10	10

Entrée [83]:

```
df3.iplot(kind='surface', colorscale='rdylbu')
```

Entrée [84]:

```
help(cf.datagen)
```

Help on module cufflinks.datagen in cufflinks:

NAME

cufflinks.datagen

FUNCTIONS

bars(n=3, n_categories=3, prefix='category', columns=None, mode='abc')
Returns a DataFrame with the required format for
a bar plot

Parameters:

n : int
 Number of points for each trace
n_categories : int
 Number of categories for each point
prefix : string
 Name for each category
columns : [str]
 ...

Entrée [85]:

```
cf.datagen.sinwave(10, 0.25).iplot(kind = 'surface')
```

C:\ProgramData\Anaconda3\lib\site-packages\cufflinks\datagen.py:380: Runtime
Warning:

invalid value encountered in true_divide

Entrée []:

Entrée [87]:

```
cf.datagen.scatter3d(2, 150, mode = 'stocks').iplot(kind = 'scatter3d', x = 'x', y = 'y', z
```

Entrée []:

Entrée [91]:

```
df[['A', 'B']].iplot(kind = 'spread')
```

Entrée []:

Entrée [98]:

```
df.iplot(kind='hist', bins = 25, barmode = 'overlay', bargap=0.5)
```

Entrée []:

Entrée [104]:

```
cf.datagen.bubble3d(5,4,mode='stocks').iplot(kind='bubble3d',x='x',y='y',z='z', size='size')
```

Entrée []:

Entrée [107]:

```
cf.datagen.heatmap(20,20).iplot(kind = 'heatmap', colorscale='spectral', title='Cufflinks -
```

Entrée []:

Entrée []:

Entrée []:

Entrée []:

Entrée []:

Entrée []:

Entrée []:

Entrée []:

Entrée []:

Entrée []:

Entrée []:

Entrée []:

Entrée []:

Entrée []:

Entrée []:

Entrée []:

Entrée []:

Entrée []:

Entrée []:

Entrée []:

Entrée []:

Entrée []:

Entrée []:

Entrée []:

Entrée []:

Entrée []:

Entrée []:

Entrée []:

Entrée []:

Entrée []:

Entrée []:

Entrée []:

Entrée []:

Entrée []:

Entrée []:

Entrée []:

Entrée []:

Entrée []:

Entrée []:

Entrée []:

Entrée []:

Entrée []:

Entrée []:

Entrée []:

Entrée []:

Entrée []:

Entrée []:

Entrée []:

Entrée []:

Entrée []:

Entrée []:

Entrée []: