Plotly and Cufflinks Crash Course

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

In [1]:

```
1 !pip install plotly
2 !pip install cufflinks
3
```

```
2.0->ipywidgets>=7.0.0->cufflinks) (3.2.0)
Requirement already satisfied: jupyter-core in c:\users\jaypr\appdata\loca
l\programs\python\python38\lib\site-packages (from nbformat>=4.2.0->ipywid
gets>=7.0.0->cufflinks) (4.6.3)
Requirement already satisfied: jupyter-client in c:\users\jaypr\appdata\lo
cal\programs\python\python38\lib\site-packages (from ipykernel>=4.5.1->ipy
widgets>=7.0.0->cufflinks) (6.1.0)
Requirement already satisfied: tornado>=4.2 in c:\users\jaypr\appdata\loca
l\programs\python\python38\lib\site-packages (from ipykernel>=4.5.1->ipywi
dgets>=7.0.0->cufflinks) (6.0.4)
Requirement already satisfied: notebook>=4.4.1 in c:\users\jaypr\appdata\l
ocal\programs\python\python38\lib\site-packages (from widgetsnbextension~=
3.5.0->ipywidgets>=7.0.0->cufflinks) (6.0.3)
Requirement already satisfied: pyrsistent>=0.14.0 in c:\users\jaypr\appdat
a\local\programs\python\python38\lib\site-packages (from jsonschema!=2.5.
0,>=2.4->nbformat>=4.2.0->ipywidgets>=7.0.0->cufflinks) (0.15.7)
Requirement already satisfied: attrs>=17.4.0 in c:\users\jaypr\appdata\loc
al\programs\python\python38\lib\site-packages (from jsonschema!=2.5.0,>=2.
4->nbformat>=4.2.0->ipywidgets>=7.0.0->cufflinks) (19.3.0)
Requirement already satisfied: pywin32>=1.0; sys_platform == "win32" in
```

In [2]:

```
!pip install mpl_toolkits
```

```
ERROR: Could not find a version that satisfies the requirement mpl_toolkits
(from versions: none)
ERROR: No matching distribution found for mpl_toolkits
WARNING: You are using pip version 20.2.3; however, version 21.0.1 is availa
ble.
You should consider upgrading via the 'c:\users\jaypr\appdata\local\programs
\python\python38\python.exe -m pip install --upgrade pip' command.
```

In [3]:

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

In [4]:

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

In [5]:

```
import cufflinks as cf
```

Using Cufflinks and iplot()

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

In [6]:

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

```
4.11.0
```

In [7]:

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

In [8]:

```
cf.go_offline()
```

In [9]:

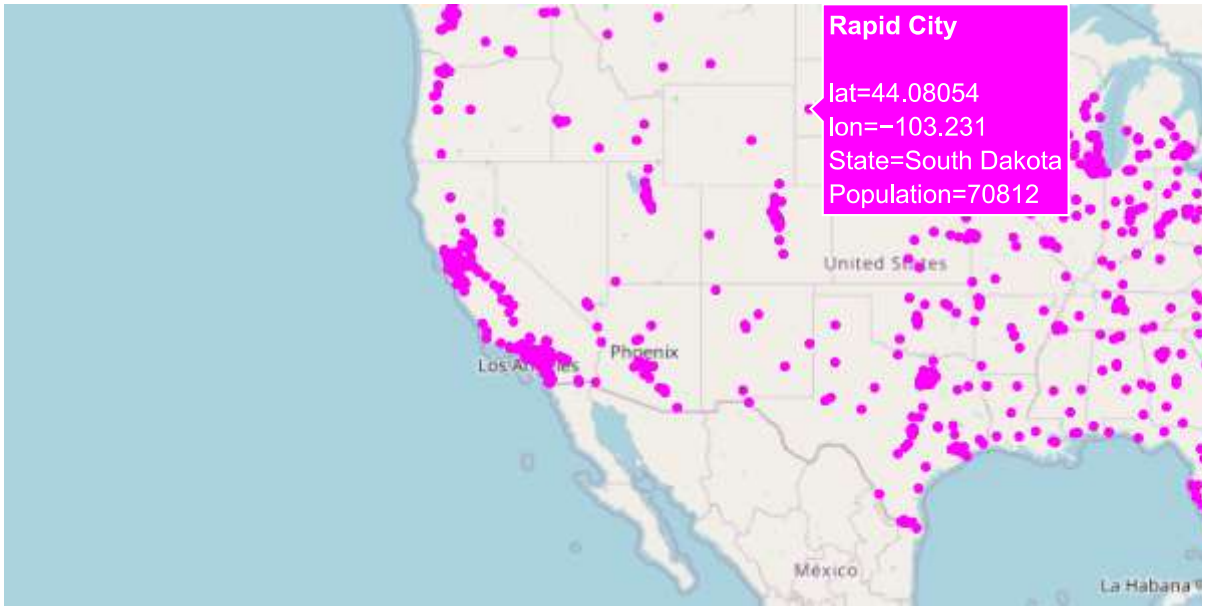
```
df = pd.DataFrame(np.random.randn(100,3), columns = ['A', 'B', 'C'])
df.head()
df['A'] = df['A'].cumsum() + 20
df['B'] = df['B'].cumsum() + 20
df['C'] = df['C'].cumsum() + 20
```

In [10]:

```
import pandas as pd
us_cities = pd.read_csv("https://raw.githubusercontent.com/plotly/datasets/master/us-ci

import plotly.express as px

fig = px.scatter_mapbox(us_cities, lat="lat", lon="lon", hover_name="City", hover_data=
                        color_discrete_sequence=["fuchsia"], zoom=3, height=300)
fig.update_layout(mapbox_style="open-street-map")
fig.update_layout(margin={"r":0,"t":0,"l":0,"b":0})
fig.show()
```



In [11]:

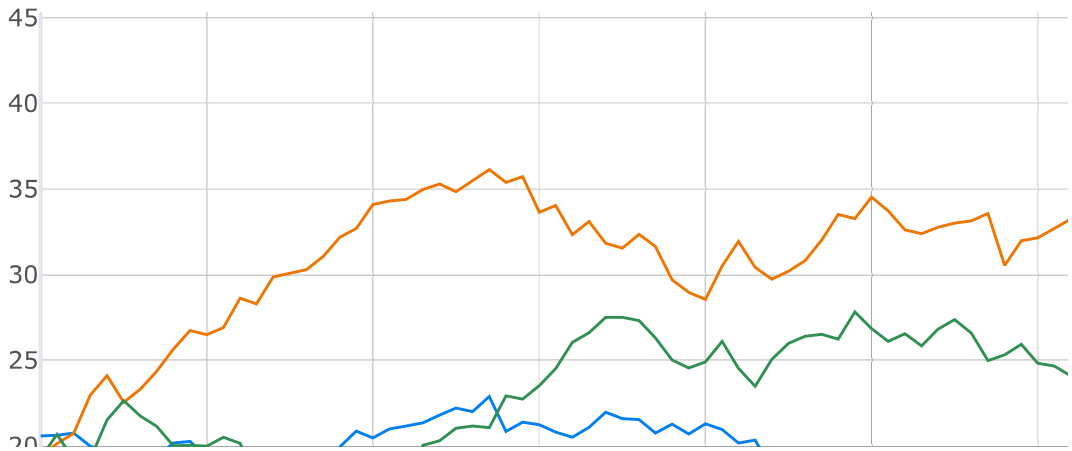
```
1 df.head()
```

Out[11]:

	A	B	C
0	20.582106	19.400658	19.287206
1	20.617086	20.141296	20.676650
2	20.750607	20.739580	19.199534
3	19.976993	22.980277	19.255901
4	19.669725	24.101417	21.509767

In [12]:

```
1 df.iplot()
```

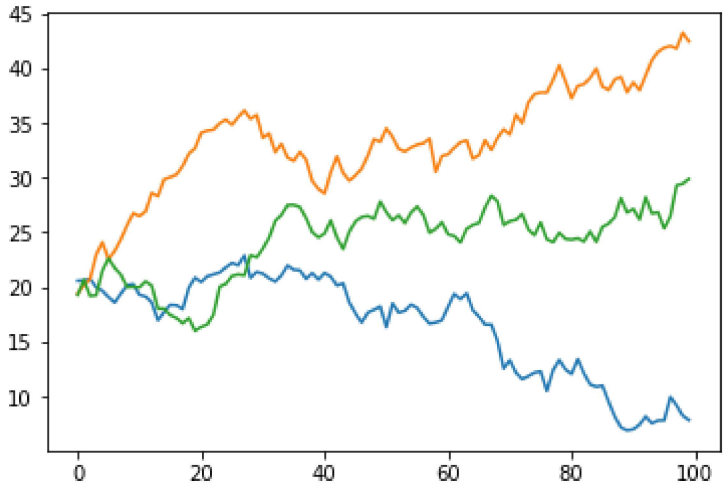


In [13]:

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

Out[13]:

```
[<matplotlib.lines.Line2D at 0x1c6b17cc3d0>,
 <matplotlib.lines.Line2D at 0x1c6b17cc4c0>,
 <matplotlib.lines.Line2D at 0x1c6b17cc580>]
```

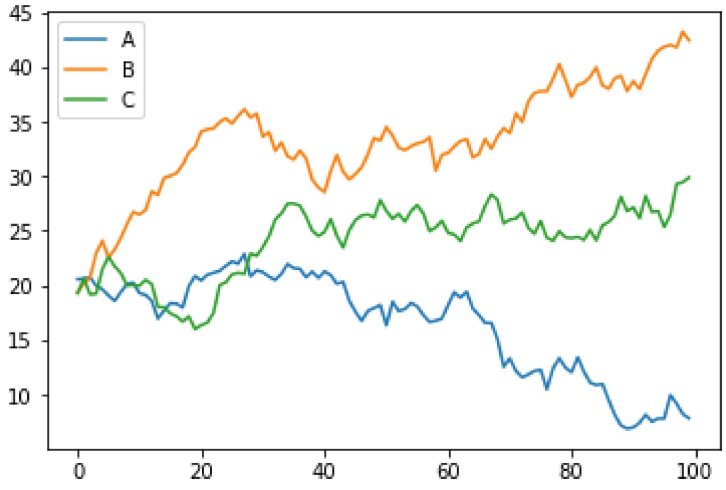


In [14]:

```
1 df.plot()
```

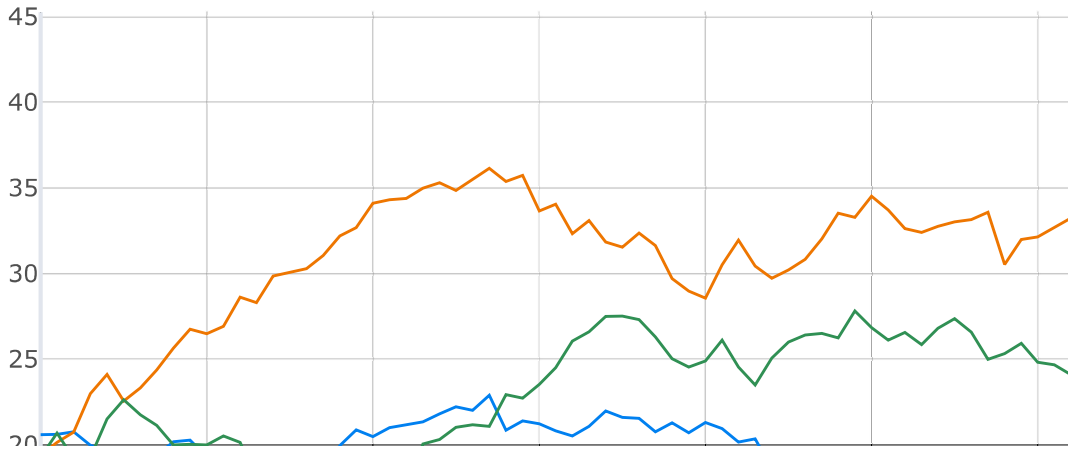
Out[14]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x1c6b282b280>
```



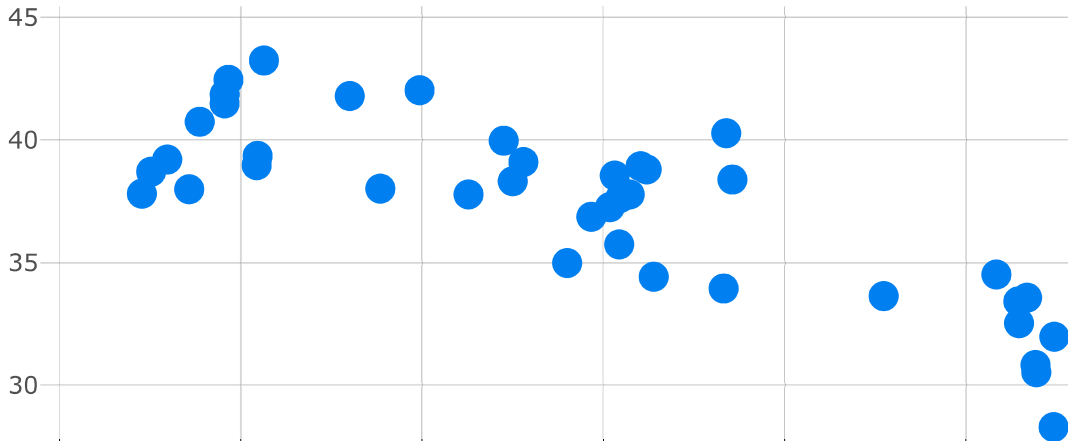
In [15]:

```
1 df.iplot()
```



In [16]:

```
df.iplot(x = 'A', y = 'B', mode = 'markers', size = 15)
```



In [17]:

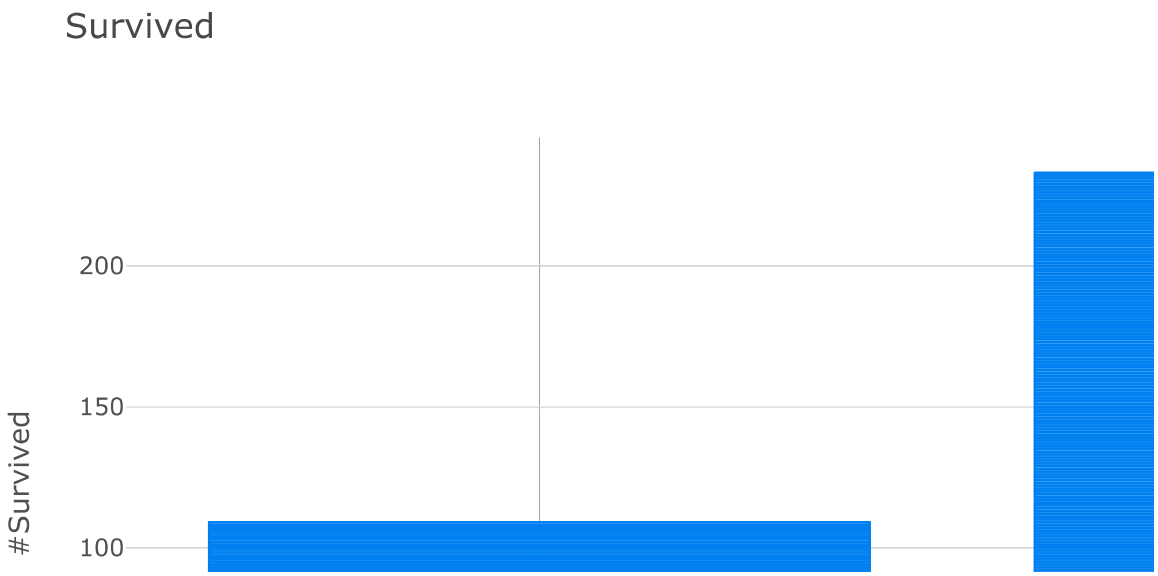
```
titanic = sns.load_dataset('titanic')
titanic.head()
```

Out[17]:

	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

In [18]:

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



In [19]:

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

Out[19]:

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

In [20]:

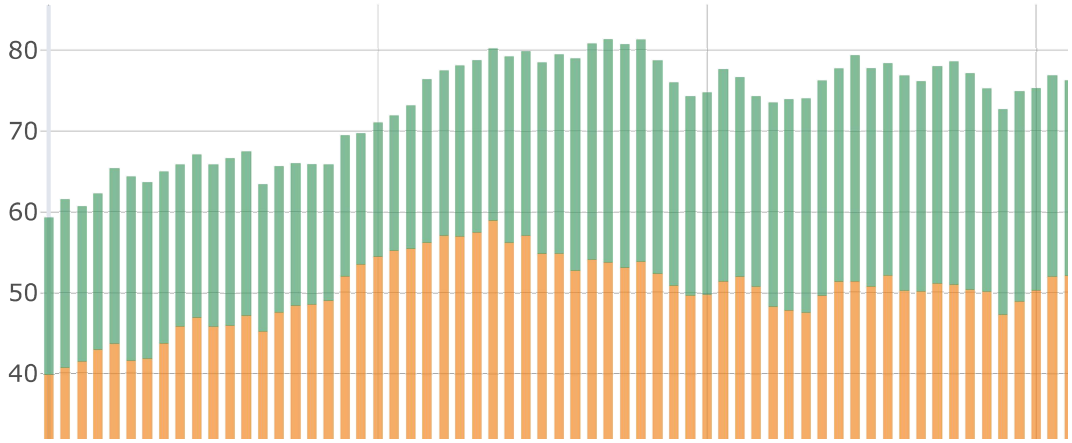
```
1 cf.getThemes()
2
```

Out[20]:

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

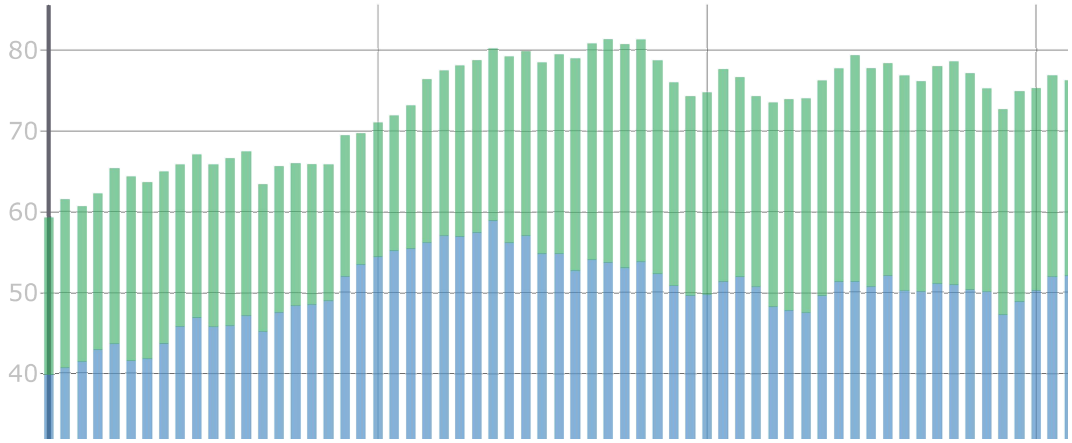
In [21]:

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



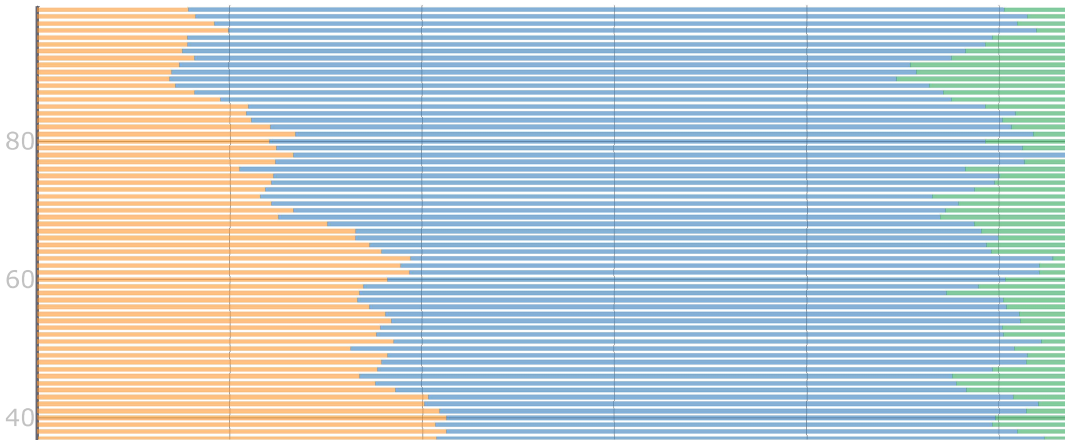
In [22]:

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



In [23]:

```
1
2 df.iplot(kind = 'barh', barmode='stack', bargap=0.5)
3
```



In [24]:

```
1 1,2,3,4,5,6,7
```

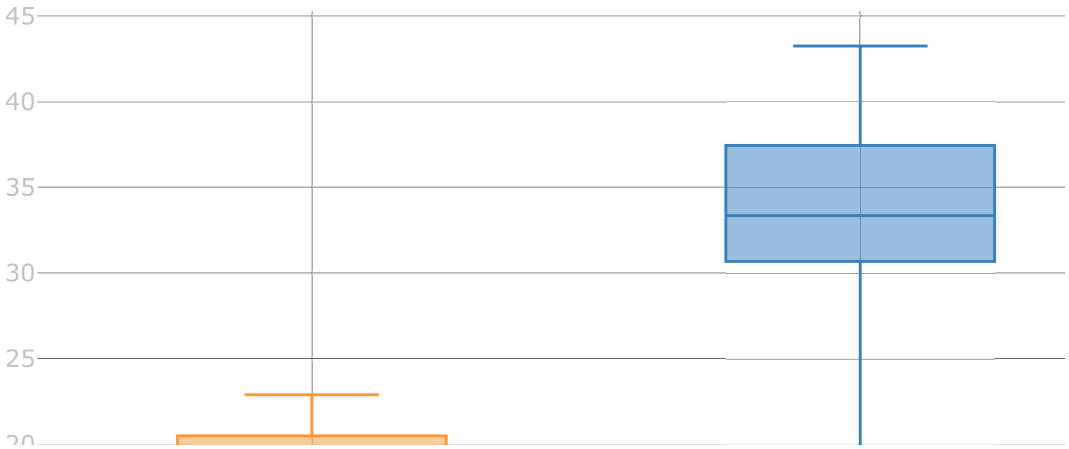
Out[24]:

```
(1, 2, 3, 4, 5, 6, 7)
```

In [25]:

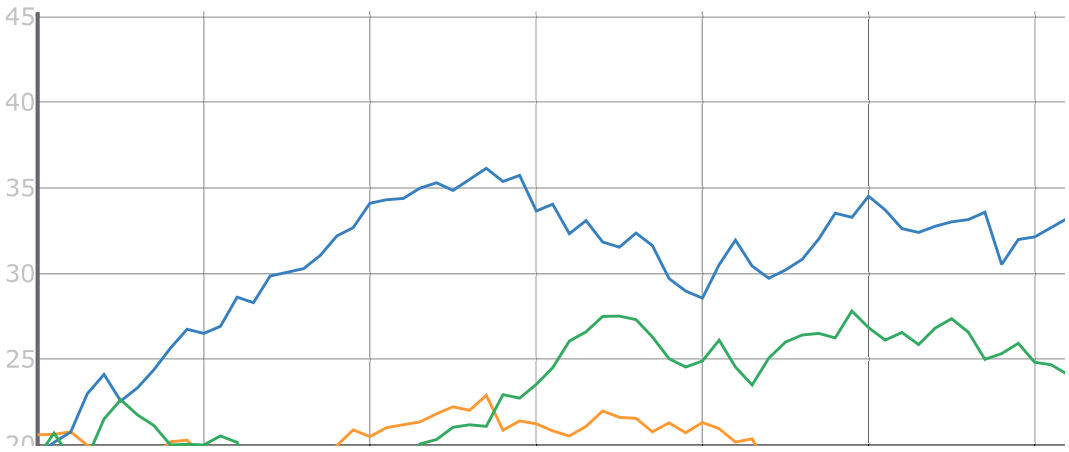
```
df.iplot(kind = 'box')
# max is Q3+1.5(IQR-Interquartile range)
# box is interquartile range
# top of box  is 75 percentile so say Q3
# middile line in box is median
# lower one of box is 25 percentile so we say Q1

```



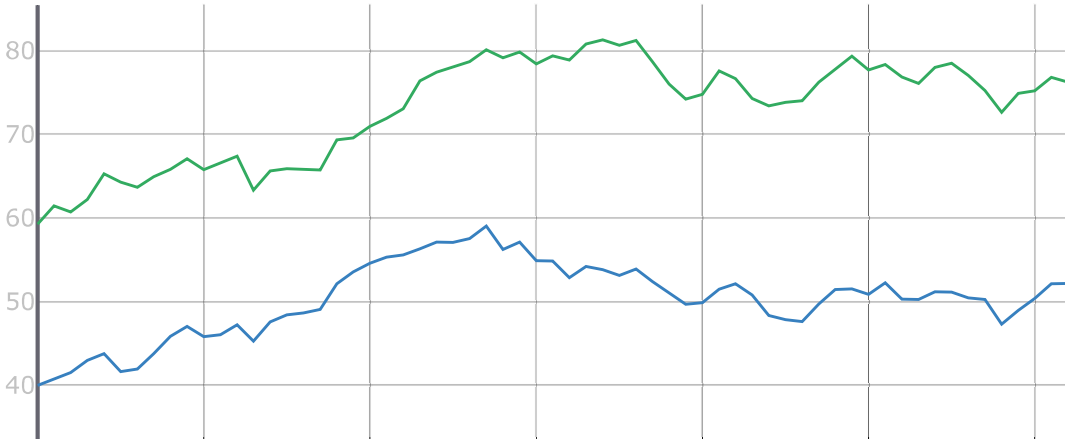
In [26]:

```
1 df.iplot()
```



In [27]:

```
1 df.iplot(kind = 'area')
```



In [28]:

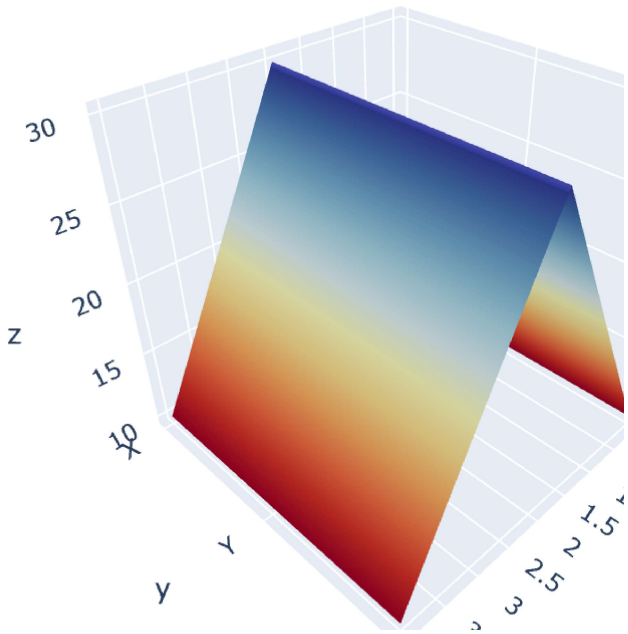
```
1 df3 = pd.DataFrame({'X': [10,20,30,20,10], 'Y': [10, 20, 30, 20, 10], 'Z': [10, 20, 30,
2 df3.head()
```

Out[28]:

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

In [29]:

```
df3.iplot(kind='surface', colorscale='rdylbu')
#  3-D graph given by kind id surface
```



In [30]:

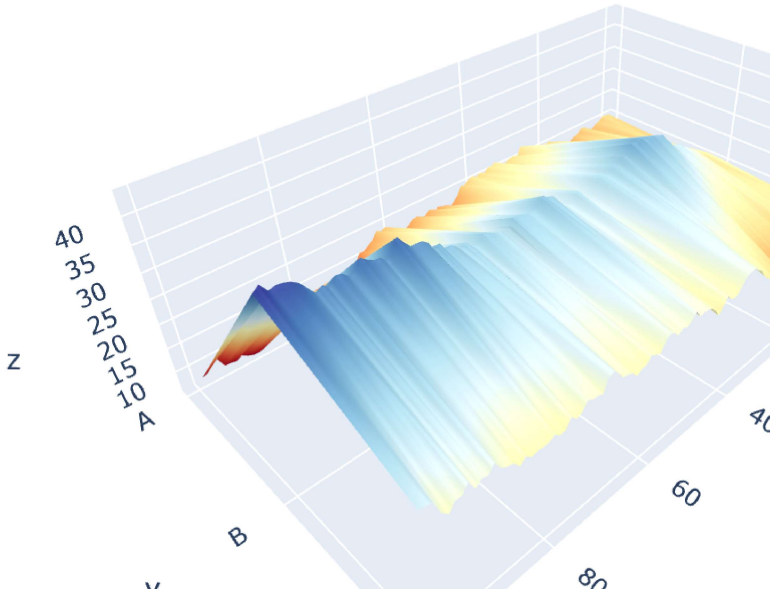
```
df.head()
```

Out[30]:

	A	B	C
0	20.582106	19.400658	19.287206
1	20.617086	20.141296	20.676650
2	20.750607	20.739580	19.199534
3	19.976993	22.980277	19.255901
4	19.669725	24.101417	21.509767

In [31]:

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



In [32]:

```
1 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]
```

In [33]:

```
1 cf.datagen.sinwave(10, 0.25)
```

```
c:\users\jaypr\appdata\local\programs\python\python38\lib\site-packages\cuff
links\datagen.py:380: RuntimeWarning:

invalid value encountered in true_divide
```

Out[33]:

	-10.00	-9.75	-9.50	-9.25	-9.00	-8.75	-8.50	-8.25	
-10.00	0.141420	0.141119	0.136502	0.127774	0.115261	0.099396	0.080692	0.059725	0.03
-9.75	0.141119	0.136324	0.127190	0.114059	0.097388	0.077730	0.055706	0.031980	0.00
-9.50	0.136502	0.127190	0.113653	0.096368	0.075924	0.052985	0.028270	0.002521	-0.02
-9.25	0.127774	0.114059	0.096368	0.075317	0.051612	0.026026	-0.000641	-0.027588	-0.05
-9.00	0.115261	0.097388	0.075924	0.051612	0.025275	-0.002227	-0.030028	-0.057286	-0.08
...	...	...	...	...	...	...	...	...	
8.75	0.099396	0.077730	0.052985	0.026026	-0.002227	-0.030842	-0.058904	-0.085555	-0.11
9.00	0.115261	0.097388	0.075924	0.051612	0.025275	-0.002227	-0.030028	-0.057286	-0.08
9.25	0.127774	0.114059	0.096368	0.075317	0.051612	0.026026	-0.000641	-0.027588	-0.05
9.50	0.136502	0.127190	0.113653	0.096368	0.075924	0.052985	0.028270	0.002521	-0.02
9.75	0.141119	0.136324	0.127190	0.114059	0.097388	0.077730	0.055706	0.031980	0.00

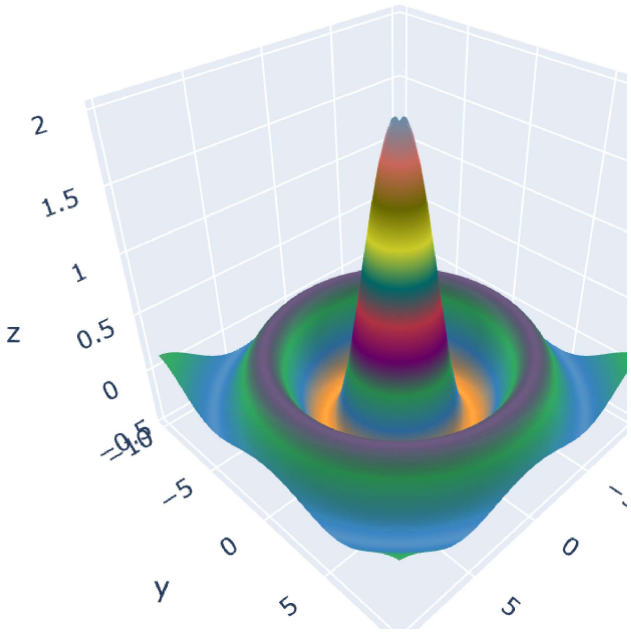
80 rows × 80 columns

In [34]:

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

```
c:\users\jaypr\appdata\local\programs\python\python38\lib\site-packages\cuff
links\datagen.py:380: RuntimeWarning:

invalid value encountered in true_divide
```



In [ ]:

```
1
```

In [35]:

```
1 cf.datagen.scatter3d(2, 150, mode = 'stocks')
```

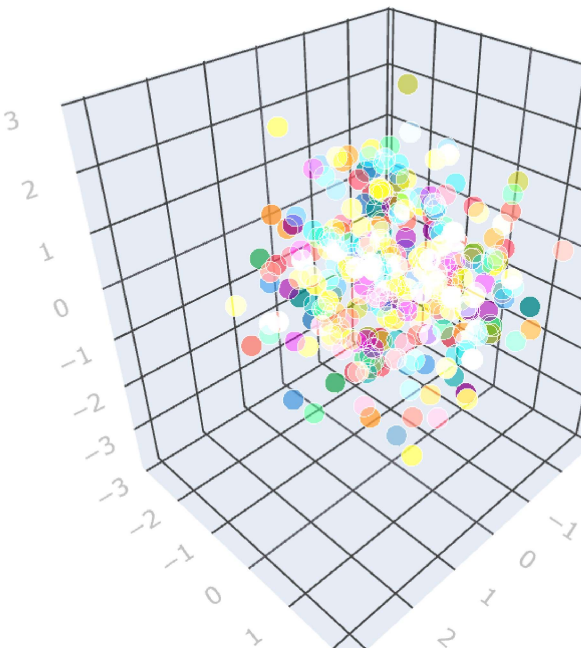
Out[35]:

	x	y	z	text	categories
0	-0.787235	-0.605613	-0.726719	HBC.HE	category1
1	-0.051558	-0.587636	-0.124863	XRY.TK	category1
2	0.382181	0.902776	0.096859	GYY.KL	category1
3	-0.503643	-0.424391	-0.897089	FUY.NB	category1
4	1.580290	0.315600	-0.381231	QGL.DF	category1
...	...	...	...	...	...
295	-0.174544	-0.796032	-1.150135	RDU.FI	category2
296	1.984021	-0.704577	-0.433376	WVR.ZE	category2
297	0.082780	-0.191479	0.488212	GTX.PO	category2
298	-1.572571	-0.334735	1.269286	SVU.DP	category2
299	-1.084338	-0.346620	-1.534560	RLO.SS	category2

300 rows × 5 columns

In [36]:

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



In [ ]:

```
1
```

In [37]:

```
1 df
```

Out[37]:

	A	B	C
0	20.582106	19.400658	19.287206
1	20.617086	20.141296	20.676650
2	20.750607	20.739580	19.199534
3	19.976993	22.980277	19.255901
4	19.669725	24.101417	21.509767
...	...	...	...
95	7.823389	41.864354	25.371801
96	9.973049	42.031431	26.457230
97	9.202944	41.795870	29.301233
98	8.256570	43.242544	29.448288
99	7.864555	42.454844	29.887756

100 rows × 3 columns

In [38]:

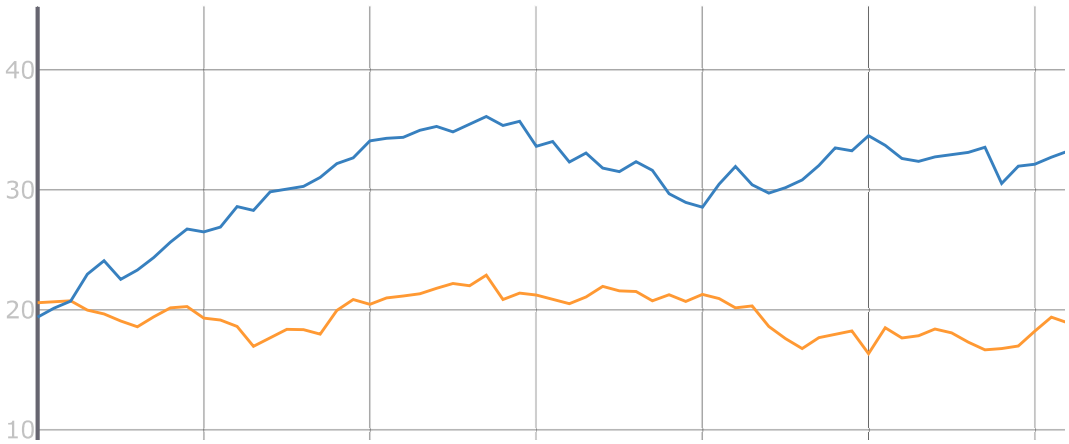
```
1 df[['A', 'B']].iplot(kind = 'spread')
```

```
c:\users\jaypr\appdata\local\programs\python\python38\lib\site-packages\cuff
links\plotlytools.py:849: FutureWarning:

The pandas.np module is deprecated and will be removed from pandas in a futu
re version. Import numpy directly instead

c:\users\jaypr\appdata\local\programs\python\python38\lib\site-packages\cuff
links\plotlytools.py:850: FutureWarning:

The pandas.np module is deprecated and will be removed from pandas in a futu
re version. Import numpy directly instead
```

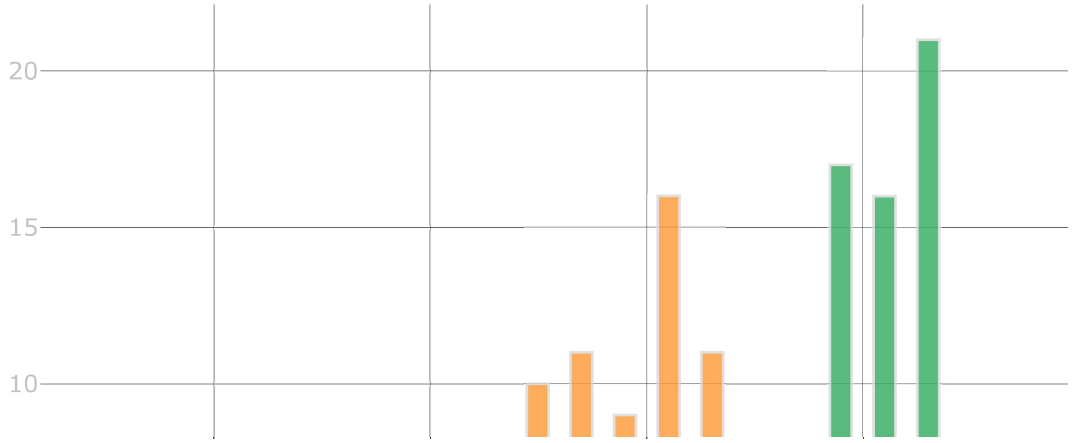


In [ ]:

```
1
```

In [39]:

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



In [ ]:

```
1
```

In [40]:

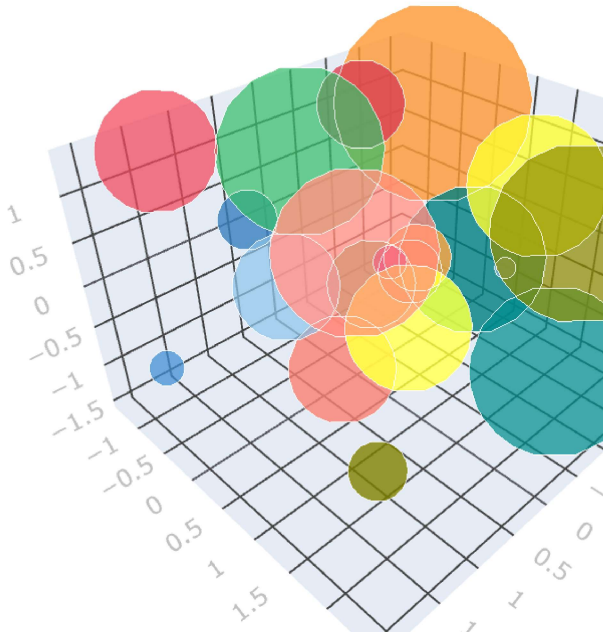
```
1 cf.datagen.bubble3d(5,4,mode='stocks')
```

Out[40]:

	x	y	z	size	text	categories
0	-0.221965	1.333085	1.200487	78	PFP.ZM	category1
1	0.442760	3.365535	-0.014687	97	KSE.CS	category1
2	0.522115	-0.971033	0.008742	76	NRJ.VX	category1
3	-0.236167	-0.366586	0.352369	73	HCY.HT	category1
4	0.339240	-0.441110	-1.082037	70	DET.YE	category2
5	-1.023237	-0.583674	0.507725	98	ORQ.XM	category2
6	1.891476	0.364198	1.286128	76	ULZ.TS	category2
7	-1.227526	0.823692	-1.930924	68	BVD.LY	category2
8	-0.179229	-0.036810	0.335511	48	OCF.HL	category3
9	0.973710	-0.997492	-0.349691	87	CXM.ZA	category3
10	-1.370509	0.528471	0.589016	2	ZRT.WN	category3
11	1.159664	1.105182	-0.981121	59	FRW.QO	category3
12	0.866367	-0.376844	-1.410100	54	VVJ.CE	category4
13	0.841819	-0.440566	-0.479827	15	CCQ.JH	category4
14	0.966933	0.110908	2.101624	88	DWA.RY	category4
15	-1.636845	-1.153394	-1.879674	63	OLO.KT	category4
16	-0.104114	-0.609884	0.374178	37	BIM.MD	category5
17	0.179745	0.470610	-0.360603	74	HQY.CH	category5
18	-0.609819	1.438276	0.098682	22	QBB.JH	category5
19	-1.235129	-0.041499	-0.459269	15	LPP.YD	category5

In [41]:

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



In [ ]:

```
1
```

In [42]:

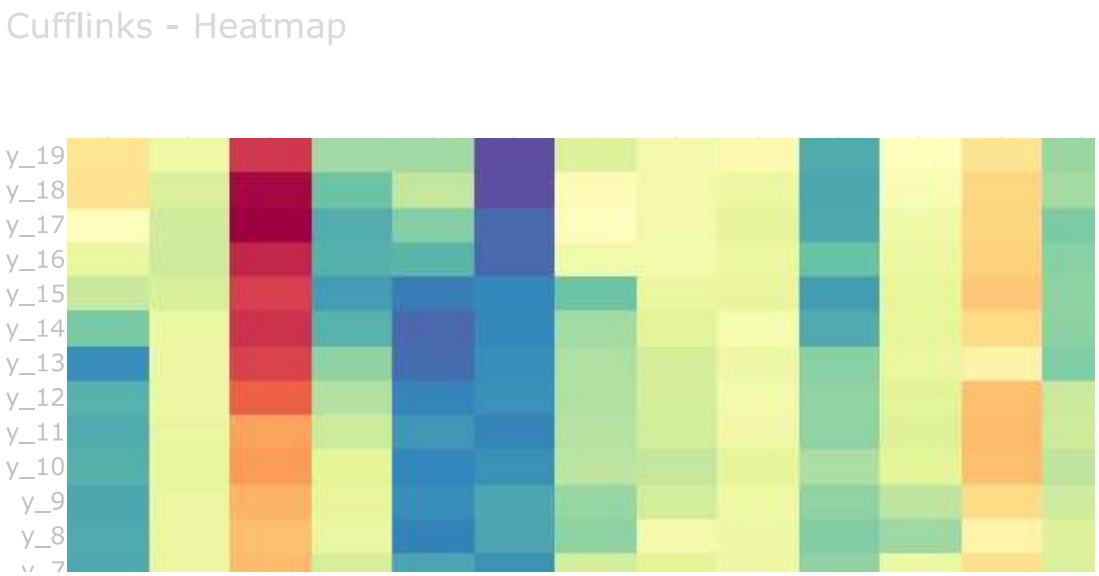
```
1 cf.datagen.heatmap(20,20)
```

Out[42]:

	y_0	y_1	y_2	y_3	y_4	y_5	y_6	y_7
x_0	20.000000	12.815639	23.584434	27.502176	29.211126	29.658153	26.106582	36.866867
x_1	28.881715	26.600531	26.354974	28.636361	29.498894	47.055816	53.812515	43.315063
x_2	10.986595	8.660168	10.489846	13.061253	17.692225	18.859744	12.294009	12.097621
x_3	22.955034	25.639004	47.603354	56.043637	51.527148	57.376643	54.013891	54.033378
x_4	29.645976	20.730300	12.829655	13.356252	12.811447	13.402311	-3.626040	0.307931
x_5	18.074550	13.564282	11.151259	12.733715	11.751998	6.288627	4.188528	5.738439
x_6	17.815414	34.534622	34.371605	36.134899	34.127301	48.998260	51.106415	56.301358
x_7	27.782605	21.026356	20.547670	21.469379	20.629419	12.673273	18.152383	19.402623
x_8	20.225015	17.240489	14.579349	22.747390	22.502548	20.007169	21.760372	23.238962
x_9	8.186232	11.935129	14.429447	7.652064	9.601932	8.901834	21.669899	14.474571
x_10	14.570039	16.761173	10.235238	10.102743	0.706147	-3.540093	1.903163	6.439741
x_11	8.419724	25.563518	21.436474	23.541222	20.257744	18.952720	23.922634	17.322056
x_12	14.844220	10.478002	8.483961	12.472113	16.015786	4.870890	11.025941	23.351586
x_13	12.424139	12.767488	11.854315	14.982388	15.028074	9.657451	6.665247	8.122691
x_14	21.414308	26.660124	25.200476	24.235043	23.744712	22.992339	25.368800	25.051051
x_15	19.632183	17.359251	15.016375	16.721032	22.814731	21.657124	24.601431	25.516511
x_16	17.148918	15.317571	13.876974	6.511071	6.510806	19.577176	17.288995	21.961542
x_17	21.643937	17.262621	16.551132	8.280606	6.098290	6.172476	4.968679	6.480927
x_18	23.446610	31.850077	31.242579	28.102081	18.162703	13.791778	16.429204	6.457712
x_19	21.190701	17.147439	15.746571	13.052404	-6.388501	-3.037469	4.970217	4.803506

In [43]:

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



In [ ]:

```
1
```

In [ ]:

```
1
```