

File Edit View Run Kernel Tabs Settings Help

Filter files by name

/ resources /

Name	Last Modified
assets	4 days ago
assignments	a day ago
bonus	4 days ago
3d_plotting...	seconds ago
annotating...	an hour ago
box_and_vi...	6 days ago
Candlestick...	6 days ago
candlestick...	6 days ago
case_study...	3 days ago
distribution...	6 days ago
heat_maps...	6 days ago
histograms...	2 hours ago
mapping.ip...	6 days ago
matplotlib...	6 days ago
probability...	6 days ago
quartiles_a...	6 days ago
scaling_and...	6 days ago
scatterplot...	6 days ago
subplots.ip...	6 days ago
time_series...	6 days ago
tree_maps.i...	6 days ago
two_dimen...	6 days ago
univariate...	2 days ago
widget_de...	6 days ago

```
[1]: # Import the pandas, numpy, scipy and matplotlib Library
import pandas as pd
import numpy as np
from scipy import stats
import matplotlib as mpl
mpl.get_backend()

import matplotlib.pyplot as plt
```

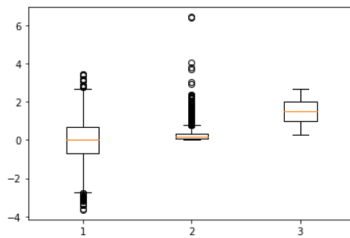
```
[2]: Dist1 = np.random.normal(size = 5000)
Dist2 = np.random.pareto( a = 5, size = 5000)

comp1 = np.random.normal(1, .2, size=5000)
comp2 = np.random.normal(2, .2, size=5000)

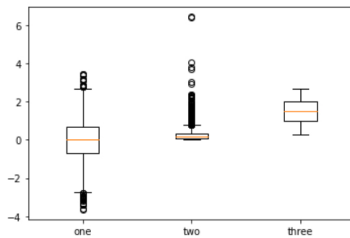
Dist3 = pd.Series(np.concatenate([comp1, comp2]))

Distributions = (Dist1, Dist2, Dist3)
```

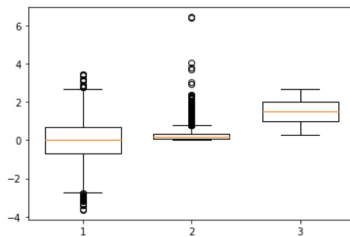
```
[3]: # basic plot
plt.boxplot(Distributions);
```



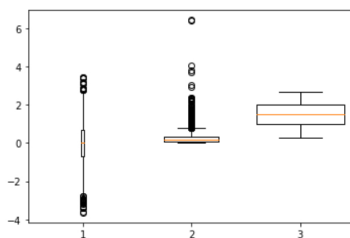
```
[4]: # plot with labels
plt.boxplot(Distributions, labels = ("one", "two", "three"));
```



```
[5]: # width size
plt.boxplot(Distributions, widths = .7);
```

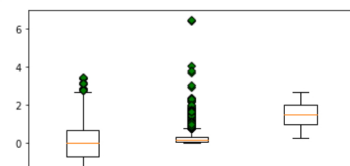


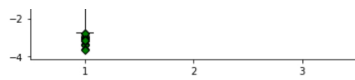
```
[6]: # individual box width size
plt.boxplot(Distributions, widths = (.03, .5, .8));
```



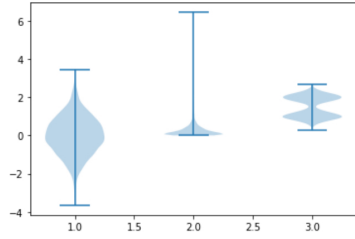
```
[7]: # customize the appearance of your outliers
green_diamond = dict(markerfacecolor='g', marker='D')

plt.boxplot(Distributions, flierprops=green_diamond);
```

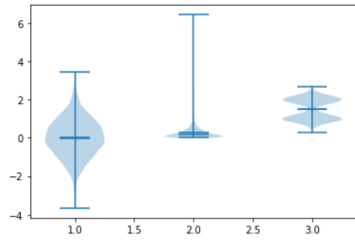




```
[8]: #basic violin plot
plt.violinplot(Distributions);
```



```
[9]: #show means and medians
plt.violinplot(Distributions, showmeans=True, showmedians=True );
```



```
[10]: #show Components
violinplt = plt.violinplot(Distributions, showmeans=True, showmedians=True );
#print (violinplt)
print (violinplt.keys())
violinplt['cmeans'].set_color('y')
violinplt['bodies'][0].set_color('r')
violinplt['bodies'][1].set_color('g')

# note violin plot requires slightly different syntax for each of the components
violinplt['cmedians'].set_color(('k','r', 'g'))

dict_keys(['bodies', 'cmeans', 'cmaxes', 'cmins', 'cbars', 'cmedians'])
```

