

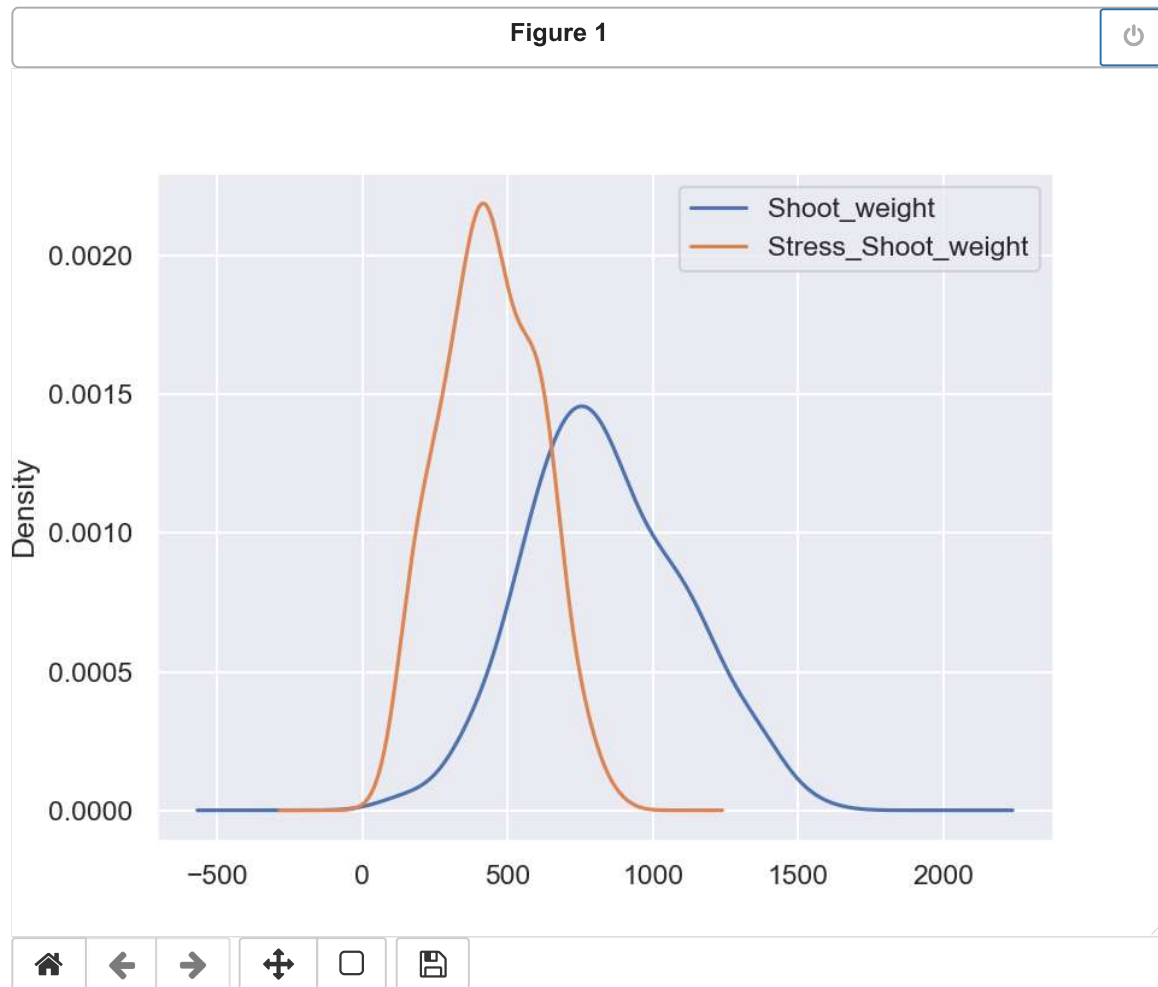
```
In [1]: import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import matplotlib as mpl
import pandas as pd
import seaborn as sns
sns.set()
%matplotlib notebook
```

```
In [2]: pd.read_excel('S3 Table Observed mean phenotype values.xlsx', header = 1).columns
```

```
Out[2]: Index(['accession_name', 'Root_weight', 'Shoot_weight', 'Root_length',
'Shoot_length', 'Root_Na+', 'Root_K+', 'Shoot_Na+', 'Shoot_K+',
'Chlorophyll_A', 'Chlorophyll_B', 'Stress_Root_weight',
'Stress_Shoot_weight', 'Stress_Root_length', 'Stress_Shoot_length',
'Stress_Root_Na+', 'Stress_Root_K+', 'Stress_Shoot_Na+',
'Stress_Shoot_K+', 'SES', 'Stress_Chlorophyll_A',
'Stress_Chlorophyll_B', 'Root_thickness', 'Shoot_thickness',
'Stress_Root_thickness', 'Stress_Shoot_thickness', 'Lost_Root_weight',
'Lost_Shoot_weight', 'Lost_Root_length', 'Lost_Shoot_length',
'Lost_Root_thickness', 'Lost_Shoot_thickness', 'Lost_Chlorophyll_A',
'Lost_Chlorophyll_B', 'PCA1'],
dtype='object')
```

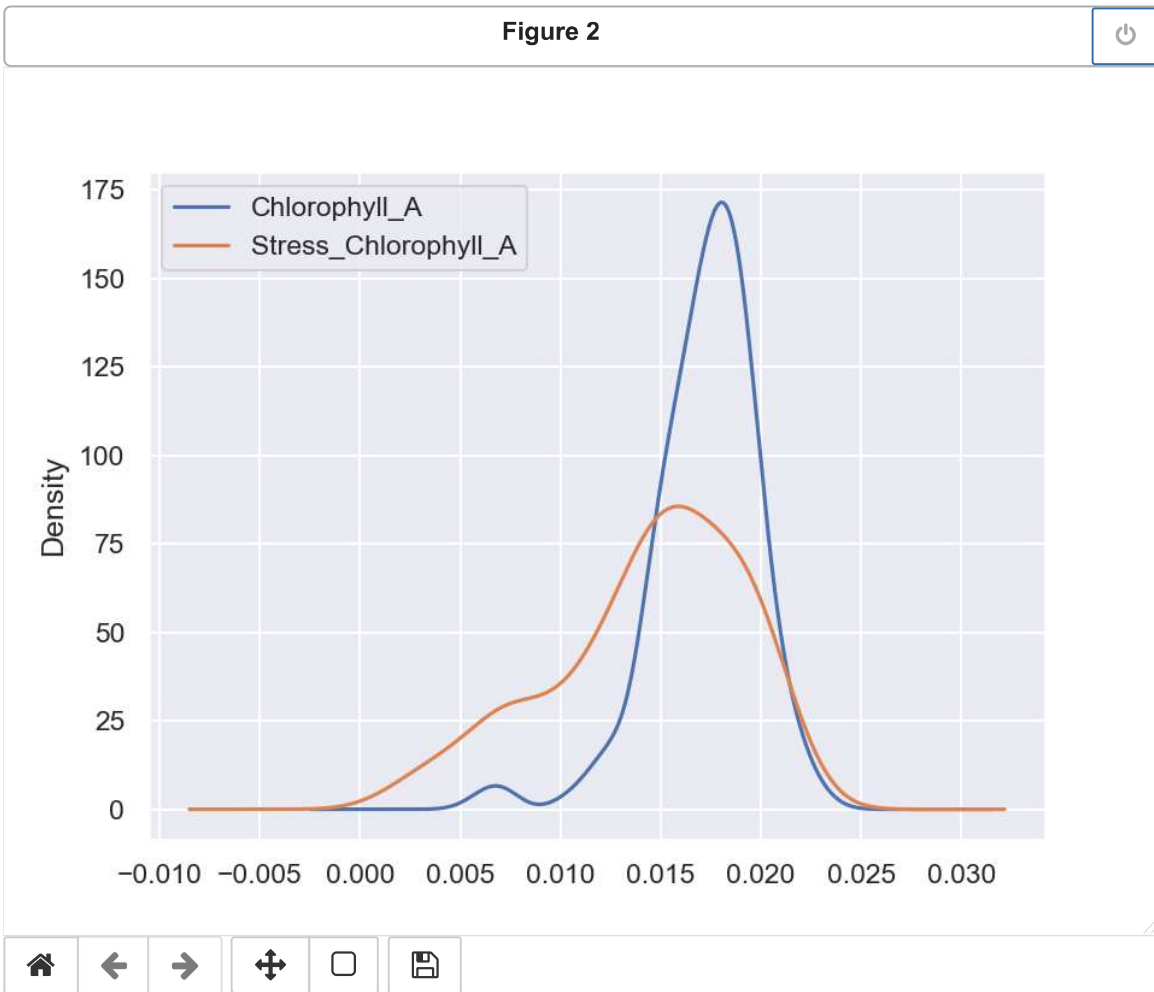
```
In [3]: t = pd.read_excel('S3 Table Observed mean phenotype values.xlsx', header = 1)[['accession_name', 'Shoo
'Stress_Shoot_weight', 'Stre
```

```
In [4]: t.iloc[:,[1,3]].plot(kind = 'kde')
```



```
Out[4]: <AxesSubplot:ylabel='Density'>
```

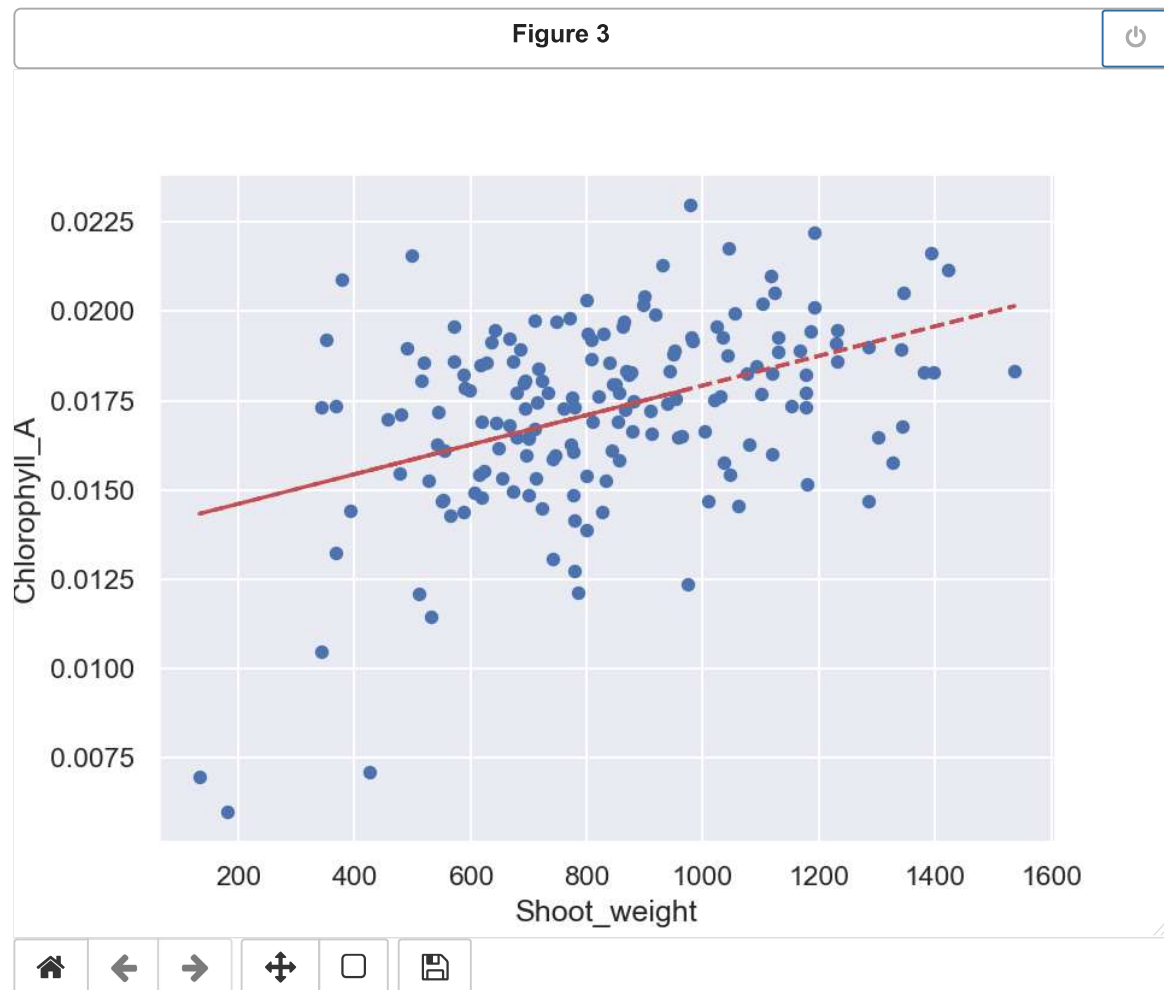
```
In [5]: t.iloc[:,[2,4]].plot(kind = 'kde')
```



```
Out[5]: <AxesSubplot:ylabel='Density'>
```

```
In [6]: t.iloc[:,[1,2]].plot(kind = 'scatter', x = 'Shoot_weight', y = 'Chlorophyll_A')
z = np.polyfit(t['Shoot_weight'].to_list(), t['Chlorophyll_A'].to_list(), 1)
p = np.poly1d(z)
plt.plot(t['Shoot_weight'].to_list(),p(t['Shoot_weight'].to_list()),"r--")

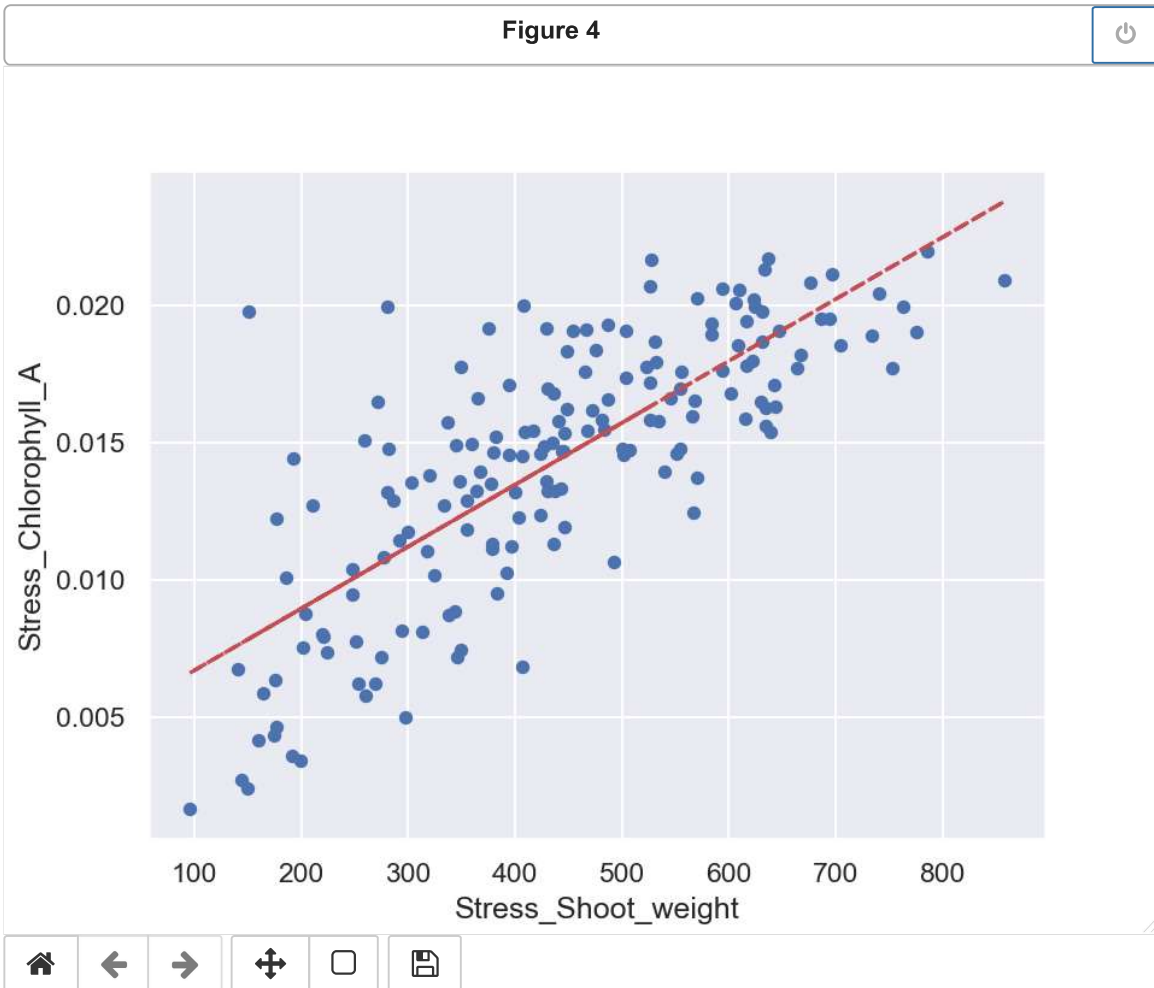
plt.show()
```



c argument looks like a single numeric RGB or RGBA sequence, which should be avoided as value-mapping will have precedence in case its length matches with *x* & *y*. Please use the *color* keyword-argument or provide a 2-D array with a single row if you intend to specify the same RGB or RGBA value for all points.

```
In [7]: t.iloc[:,[3,4]].plot(kind = 'scatter', x = 'Stress_Shoot_weight', y = 'Stress_Chlorophyll_A')
z = np.polyfit(t['Stress_Shoot_weight'].to_list(), t['Stress_Chlorophyll_A'].to_list(), 1)
p = np.poly1d(z)
plt.plot(t['Stress_Shoot_weight'].to_list(),p(t['Stress_Shoot_weight'].to_list()),"r--")

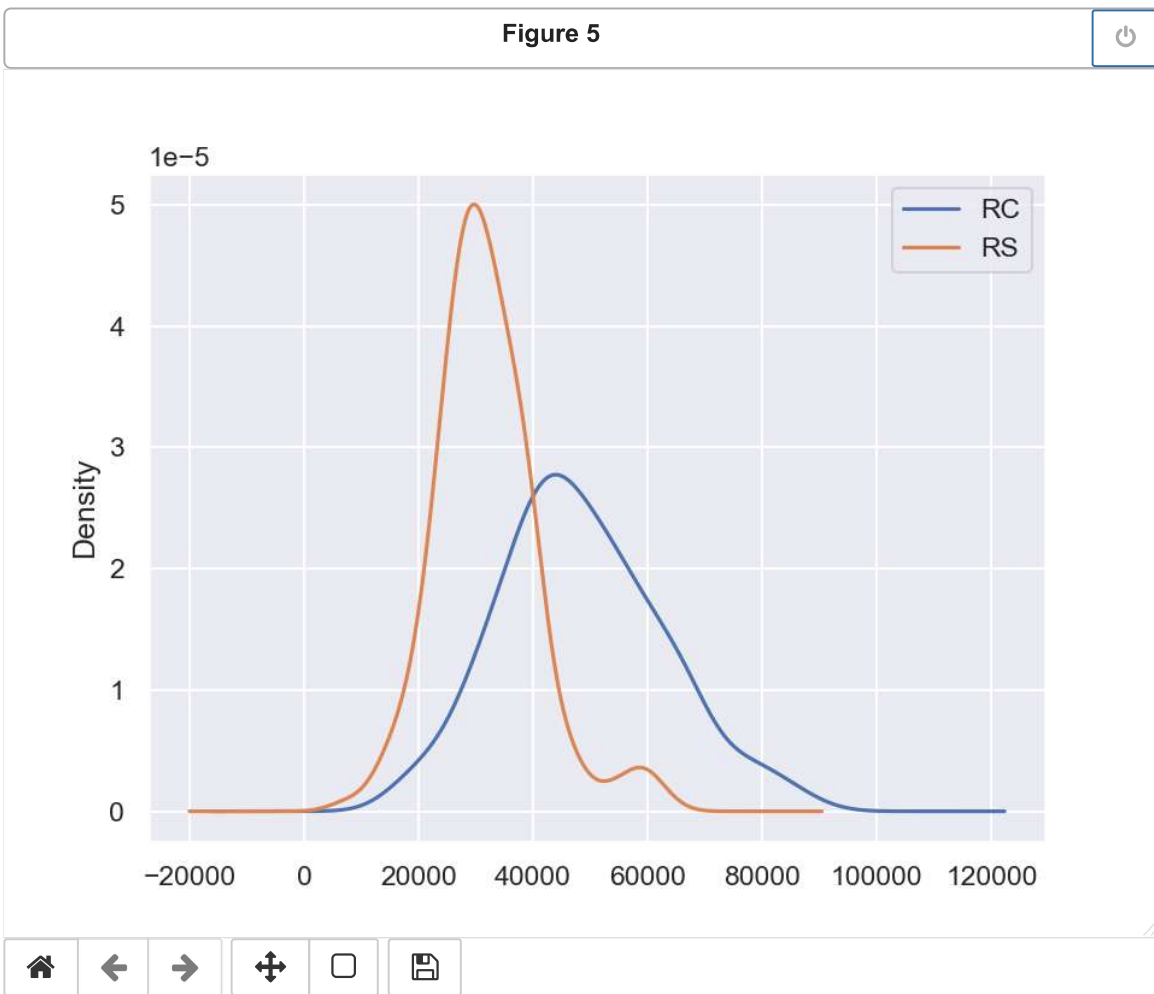
plt.show()
```



c argument looks like a single numeric RGB or RGBA sequence, which should be avoided as value-mapping will have precedence in case its length matches with *x* & *y*. Please use the *color* keyword-argument or provide a 2-D array with a single row if you intend to specify the same RGB or RGBA value for all points.

```
In [8]: t['RC'] = t['Shoot_weight'] / t['Chlorophyll_A']
t['RS'] = t['Stress_Shoot_weight'] / t['Stress_Chlorophyll_A']
```

```
In [9]: t.iloc[:, -2:].plot(kind = 'kde')
```



```
Out[9]: <AxesSubplot:ylabel='Density'>
```

```
In [10]: from IPython.display import Image
Image(filename='Extra for r2.jpeg')
```

Out[10]:

