0. matplotlib_tutorial

April 3, 2023

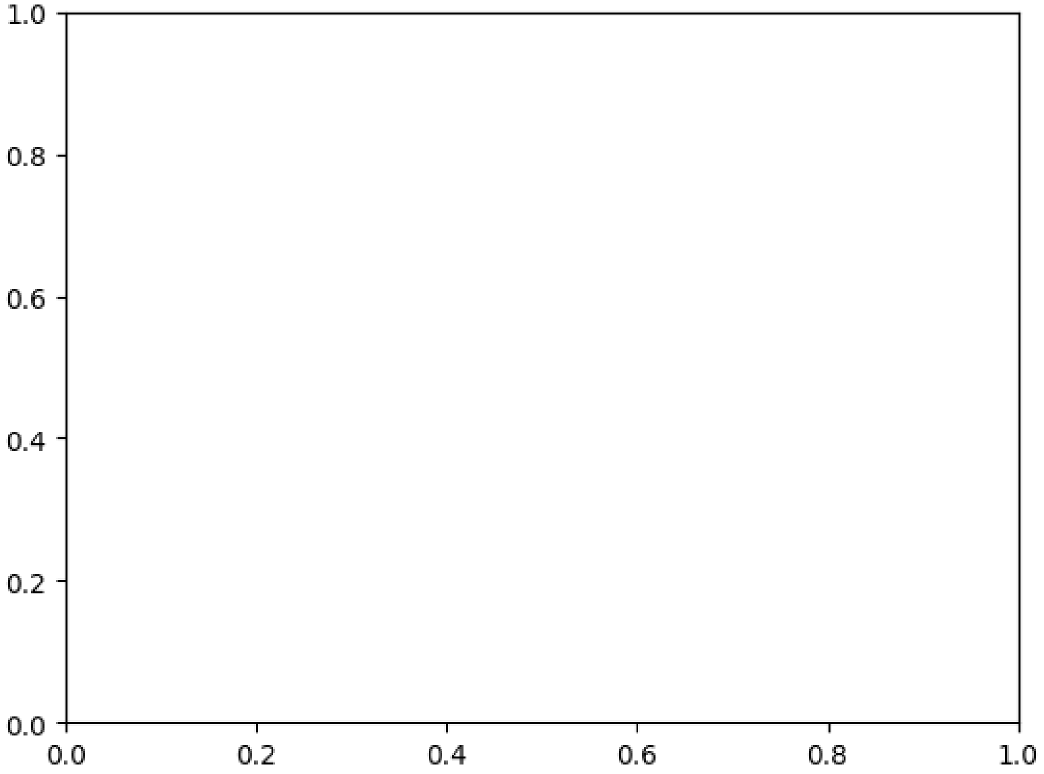
[ ]:

```python
# Create an empty figure with no axis
# To see the figure, save the code as a single file say `figure.py` and run the␣
 ↪file
import matplotlib.pyplot as plt
figure = plt.figure()
plt.show()
```

```
<Figure size 640x480 with 0 Axes>
```

[ ]:

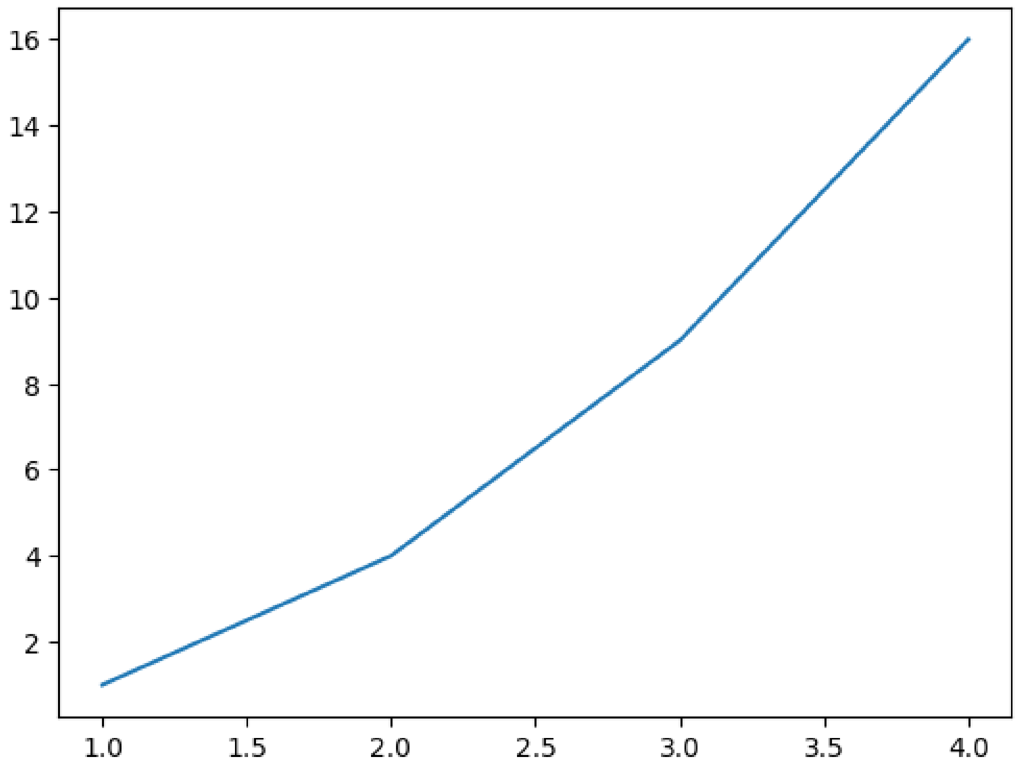
```python
# Create a figure with a single axis
# The figure method call `add_subplot(1, 1, 1)` means
# the figure is grided into `1` row, `1` column, and the axis is placed at␣
 ↪position `1`
import matplotlib.pyplot as plt
figure = plt.figure()
axis = figure.add_subplot(1, 1, 1)
plt.show()
```



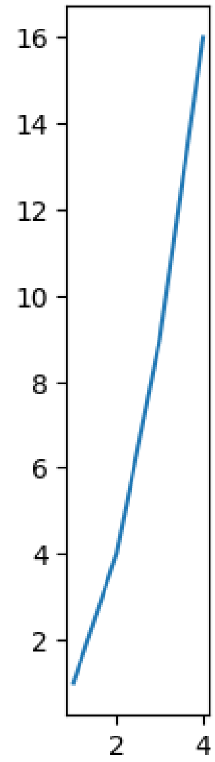
[ ]:

```
# Add a plot of `y=x^2` to the axis
import matplotlib.pyplot as plt
figure = plt.figure()
axis = figure.add_subplot(1, 1, 1)
x = [1, 2, 3, 4]
y = [xi**2 for xi in x]
print(x)
print(y)
axis.plot(x, y)
plt.show()
```

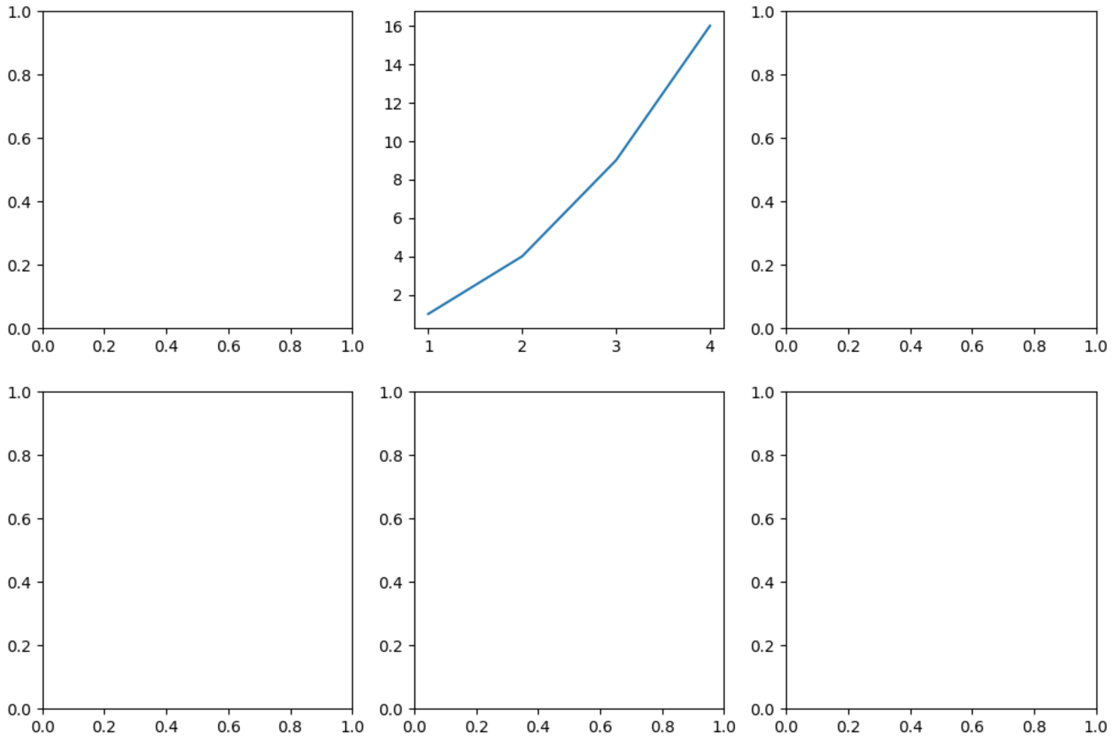
```
[1, 2, 3, 4]
[1, 4, 9, 16]
```



[ ]:
```
# Change aspect ratio to `1`, or equivalent to "equal"
import matplotlib.pyplot as plt
figure = plt.figure()
axis = figure.add_subplot(1, 1, 1)
axis.set_aspect(1)
# axis.set_aspect("equal")
# axis.set_aspect("auto")
# axis.set_aspect(1/2)
x = [1, 2, 3, 4]
y = [xi**2 for xi in x]
axis.plot(x, y)
plt.show()
```



```
[ ]: # Add plot to a specific axis
     import matplotlib.pyplot as plt
     figure, axes = plt.subplots(2, 3)
     figure.set(figheight=8, figwidth=12)
     x = [1, 2, 3, 4]
     y = [xi**2 for xi in x]
     axes[0, 1].plot(x, y)
     plt.show()
```

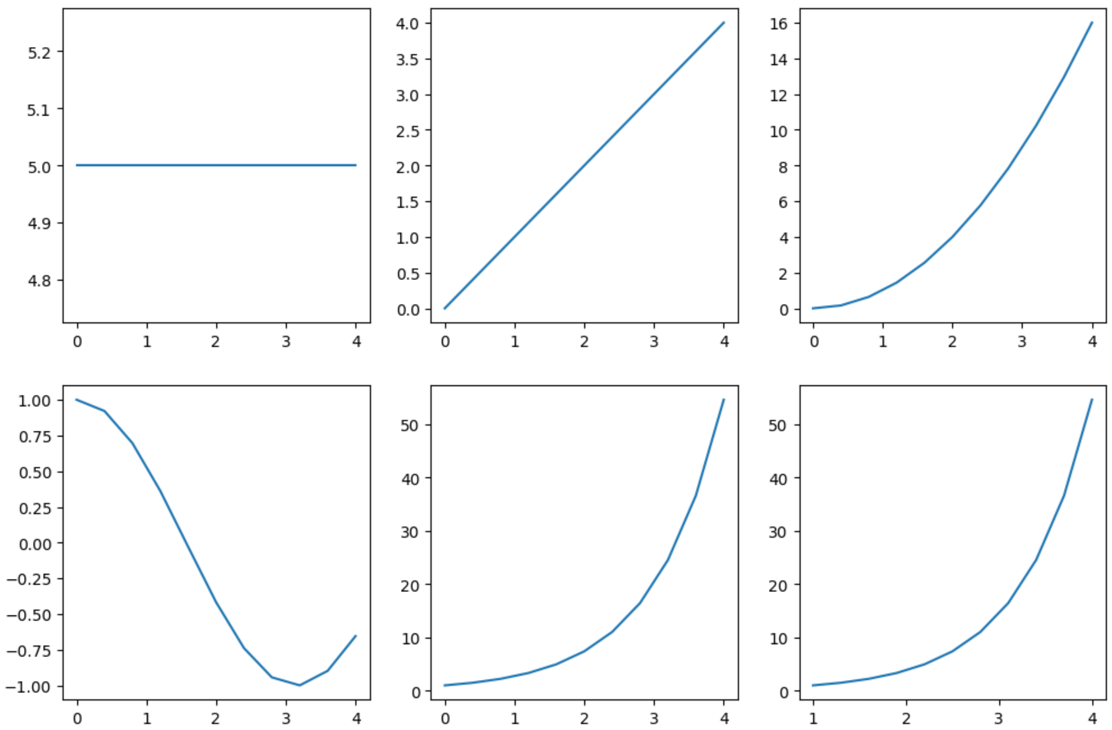


[ ]:

```
# Add a plot to each axis
import numpy as np
import matplotlib.pyplot as plt
figure, axes = plt.subplots(2, 3)
figure.set(figheight=8, figwidth=12)
x1 = np.linspace(0, 4, num=11)
y1 = np.repeat(a=5, repeats=11)
print(f"x1 = {x1}")
print(f"y1 = {y1}")
axes[0, 0].plot(x1, y1)
x2 = np.linspace(0, 4, num=11)
y2 = x2
print(f"x2 = {x2}")
print(f"y2 = {y2}")
axes[0, 1].plot(x2, y2)
x3 = np.linspace(0, 4, num=11)
y3 = x3**2
print(f"x3 = {x3}")
print(f"y3 = {y3}")
axes[0, 2].plot(x3, y3)
x4 = np.linspace(0, 4, num=11)
y4 = np.cos(x4)
print(f"x4 = {x4}")
```

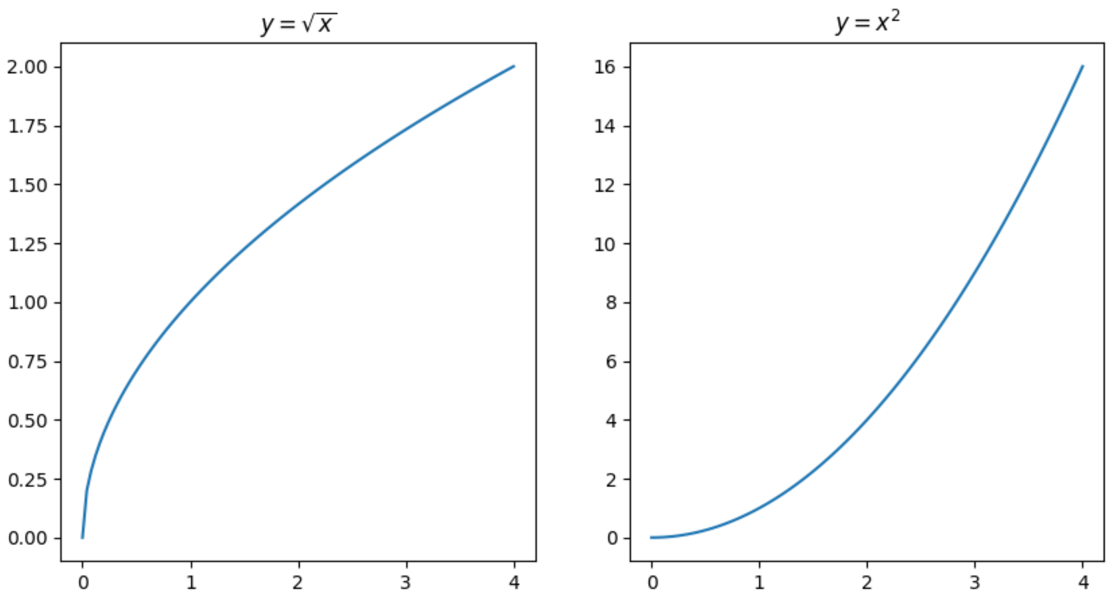
```python
print(f"y4 = {y4}")
axes[1, 0].plot(x4, y4)
x5 = np.linspace(0, 4, num=11)
y5 = np.exp(x5)
print(f"x5 = {x5}")
print(f"y5 = {y5}")
axes[1, 1].plot(x5, y5)
x6 = np.linspace(1, 4, num=11)
y6 = np.log(x6)
print(f"x6 = {x6}")
print(f"y6 = {y6}")
axes[1, 2].plot(x6, y5)
plt.show()
```

```
x1 = [0.  0.4 0.8 1.2 1.6 2.  2.4 2.8 3.2 3.6 4. ]
y1 = [5 5 5 5 5 5 5 5 5 5 5]
x2 = [0.  0.4 0.8 1.2 1.6 2.  2.4 2.8 3.2 3.6 4. ]
y2 = [0.  0.4 0.8 1.2 1.6 2.  2.4 2.8 3.2 3.6 4. ]
x3 = [0.  0.4 0.8 1.2 1.6 2.  2.4 2.8 3.2 3.6 4. ]
y3 = [ 0.    0.16  0.64  1.44  2.56  4.    5.76  7.84 10.24 12.96 16.  ]
x4 = [0.  0.4 0.8 1.2 1.6 2.  2.4 2.8 3.2 3.6 4. ]
y4 = [ 1.          0.92106099  0.69670671  0.36235775 -0.02919952 -0.41614684
 -0.73739372 -0.94222234 -0.99829478 -0.89675842 -0.65364362]
x5 = [0.  0.4 0.8 1.2 1.6 2.  2.4 2.8 3.2 3.6 4. ]
y5 = [ 1.          1.4918247   2.22554093  3.32011692  4.95303242  7.3890561
 11.02317638 16.44464677 24.5325302  36.59823444 54.59815003]
x6 = [1.  1.3 1.6 1.9 2.2 2.5 2.8 3.1 3.4 3.7 4. ]
y6 = [0.         0.26236426 0.47000363 0.64185389 0.78845736 0.91629073
 1.02961942 1.13140211 1.22377543 1.30833282 1.38629436]
```



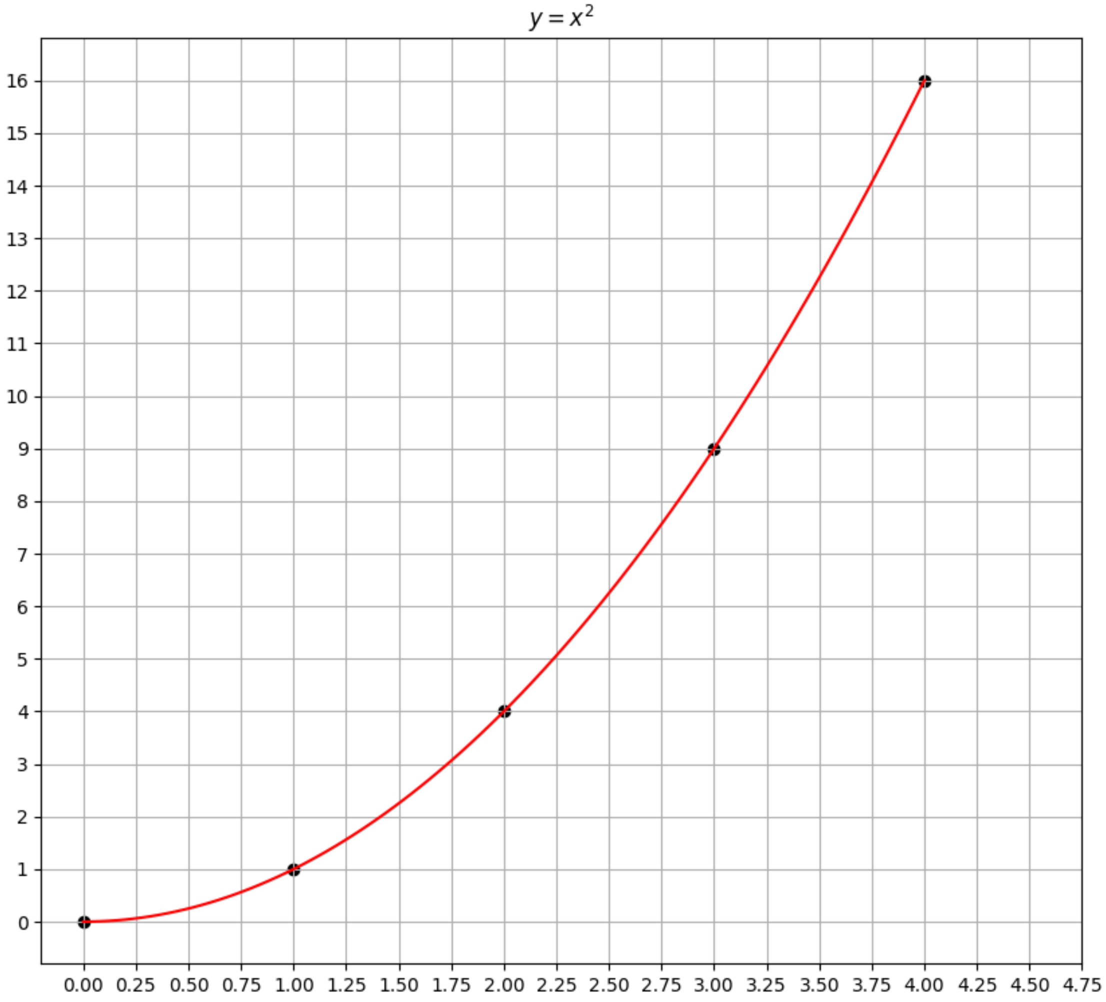
[ ]:

```
# One dimensional grid
import numpy as np
import matplotlib.pyplot as plt
figure, axes = plt.subplots(1, 2)
figure.set(figheight=5, figwidth=10)
x1 = np.linspace(0, 4, num=101)
y1 = np.sqrt(x1)
# print(f"x1 = {x1}")
# print(f"y1 = {y1}")
axes[0].plot(x1, y1)
text0 = axes[0].set_title("$ y=\sqrt{x} $")
x2 = np.linspace(0, 4, num=101)
y2 = np.square(x2)
# print(f"x2 = {x2}")
# print(f"y2 = {y2}")
axes[1].plot(x2, y2)
text1 = axes[1].set_title("$ y=x^2 $")
plt.show()
```



[ ]:

```
# Add two or more plots to an axis
import matplotlib.pyplot as plt
figure, axis = plt.subplots(1, 1)
figure.set(figheight=10, figwidth=10)
x = np.linspace(0, 4, 5)
y = np.square(x)
axis.scatter(x, y, c='black')
x_mesh = np.linspace(0, 4, 101)
y_mesh = np.square(x_mesh)
axis.plot(x_mesh, y_mesh, c='red')
axis.set_title("$ y=x^2 $")
axis.set_aspect(1 / 4)
axis.set_xticks(np.arange(0, 5, 0.25))
axis.set_yticks(np.arange(0, 17, 1))
axis.grid(axis='both', which='both')
plt.show()
```

$y = x^2$
16
15
14
13
12
11
10
9
8
7
6
5
4
3
2
1
0
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00
2.25
2.50
2.75
3.00
3.25
3.50
3.75
4.00
4.25
4.50
4.75