**Generating visualizations with pyplot**

|  |
| --- |
| **import** matplotlib**.**pyplot **as** plt  plt**.**plot**([**1**,** 2**,** 3**,** 4**])**  plt**.**ylabel**(**'some numbers'**)**  plt**.**show**()** |
|  |

**Formatting the style of your plot**

|  |
| --- |
| **import** matplotlib**.**pyplot **as** plt  plt**.**plot**([**1**,** 2**,** 3**,** 4**],** **[**1**,** 4**,** 9**,** 16**],** 'ro'**)**  plt**.**axis**([**0**,** 6**,** 0**,** 20**])**  plt**.**show**()** |
|  |

## **Plotting with keyword strings**

|  |
| --- |
| **import** numpy **as** np  **import** matplotlib**.**pyplot **as** plt  data **=** **{**'a'**:** np**.**arange**(**50**),**  'c'**:** np**.**random**.**randint**(**0**,** 50**,** 50**),**  'd'**:** np**.**random**.**randn**(**50**)}**  data**[**'b'**]** **=** data**[**'a'**]** **+** 10 **\*** np**.**random**.**randn**(**50**)**  data**[**'d'**]** **=** np**.abs(**data**[**'d'**])** **\*** 100  plt**.**scatter**(**'a'**,** 'b'**,** c**=**'c'**,** s**=**'d'**,** data**=**data**)**  plt**.**xlabel**(**'entry a'**)**  plt**.**ylabel**(**'entry b'**)**  plt**.**show**()** |
|  |

## **Plotting with categorical variables**

|  |
| --- |
| **import** matplotlib**.**pyplot **as** plt  names **=** **[**'group\_a'**,** 'group\_b'**,** 'group\_c'**]**  values **=** **[**1**,** 10**,** 100**]**  plt**.**figure**(**figsize**=(**9**,** 3**))**  plt**.**subplot**(**131**)**  plt**.**bar**(**names**,** values**)**  plt**.**subplot**(**132**)**  plt**.**scatter**(**names**,** values**)**  plt**.**subplot**(**133**)**  plt**.**plot**(**names**,** values**)**  plt**.**suptitle**(**'Categorical Plotting'**)**  plt**.**show**()** |
|  |

## **Working with multiple figures and axes**

|  |
| --- |
| **import** numpy **as** np  **import** matplotlib**.**pyplot **as** plt  **def** f**(**t**):**  **return** np**.**exp**(-**t**)** **\*** np**.**cos**(**2**\***np**.**pi**\***t**)**  t1 **=** np**.**arange**(**0.0**,** 5.0**,** 0.1**)**  t2 **=** np**.**arange**(**0.0**,** 5.0**,** 0.02**)**  plt**.**figure**()**  plt**.**subplot**(**211**)**  plt**.**plot**(**t1**,** f**(**t1**),** 'bo'**,** t2**,** f**(**t2**),** 'k'**)**  plt**.**subplot**(**212**)**  plt**.**plot**(**t2**,** np**.**cos**(**2**\***np**.**pi**\***t2**),** 'r--'**)**  plt**.**show**()** |
|  |

## **Working with text**

|  |
| --- |
| **import** numpy **as** np  **import** matplotlib**.**pyplot **as** plt  mu**,** sigma **=** 100**,** 15  x **=** mu **+** sigma **\*** np**.**random**.**randn**(**10000**)**  # the histogram of the data  n**,** bins**,** patches **=** plt**.**hist**(**x**,** 50**,** density**=**1**,** facecolor**=**'g'**,** alpha**=**0.75**)**  plt**.**xlabel**(**'Smarts'**)**  plt**.**ylabel**(**'Probability'**)**  plt**.**title**(**'Histogram of IQ'**)**  plt**.**text**(**60**,** .025**,** r'$\mu=100,\ \sigma=15$'**)**  plt**.**axis**([**40**,** 160**,** 0**,** 0.03**])**  plt**.**grid**(True)**  plt**.**show**()** |
|  |

### Using mathematical expressions in text

### Annotating text

|  |
| --- |
| **import** numpy **as** np  **import** matplotlib**.**pyplot **as** plt  ax **=** plt**.**subplot**()**  t **=** np**.**arange**(**0.0**,** 5.0**,** 0.01**)**  s **=** np**.**cos**(**2**\***np**.**pi**\***t**)**  line**,** **=** plt**.**plot**(**t**,** s**,** lw**=**2**)**  plt**.**annotate**(**'local max'**,** xy**=(**2**,** 1**),** xytext**=(**3**,** 1.5**),**  arrowprops**=dict(**facecolor**=**'black'**,** shrink**=**0.05**),**  **)**  plt**.**ylim**(-**2**,** 2**)**  plt**.**show**()** |
|  |

## **Logarithmic and other nonlinear axes**

|  |
| --- |
| **import** numpy **as** np  **import** matplotlib**.**pyplot **as** plt  # Fixing random state for reproducibility  np**.**random**.**seed**(**19680801**)**  # make up some data in the open interval (0, 1)  y **=** np**.**random**.**normal**(**loc**=**0.5**,** scale**=**0.4**,** size**=**1000**)**  y **=** y**[(**y **>** 0**)** **&** **(**y **<** 1**)]**  y**.**sort**()**  x **=** np**.**arange**(len(**y**))**  # plot with various axes scales  plt**.**figure**()**  # linear  plt**.**subplot**(**221**)**  plt**.**plot**(**x**,** y**)**  plt**.**yscale**(**'linear'**)**  plt**.**title**(**'linear'**)**  plt**.**grid**(True)**  # log  plt**.**subplot**(**222**)**  plt**.**plot**(**x**,** y**)**  plt**.**yscale**(**'log'**)**  plt**.**title**(**'log'**)**  plt**.**grid**(True)**  # symmetric log  plt**.**subplot**(**223**)**  plt**.**plot**(**x**,** y **-** y**.**mean**())**  plt**.**yscale**(**'symlog'**,** linthresh**=**0.01**)**  plt**.**title**(**'symlog'**)**  plt**.**grid**(True)**  # logit  plt**.**subplot**(**224**)**  plt**.**plot**(**x**,** y**)**  plt**.**yscale**(**'logit'**)**  plt**.**title**(**'logit'**)**  plt**.**grid**(True)**  # Adjust the subplot layout, because the logit one may take more space  # than usual, due to y-tick labels like "1 - 10^{-3}"  plt**.**subplots\_adjust**(**top**=**0.92**,** bottom**=**0.08**,** left**=**0.10**,** right**=**0.95**,** hspace**=**0.25**,**  wspace**=**0.35**)**  plt**.**show**()** |
|  |