**Plotting commands summary**

matplotlib.pyplot.**plotting**()

| **Function** | **Description** |
| --- | --- |
| [**acorr**](http://matplotlib.org/api/pyplot_api.html#matplotlib.pyplot.acorr) | Plot the autocorrelation of **x**. |
| **[angle\_spectrum](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.angle_spectrum" \o "matplotlib.pyplot.angle_spectrum)** | Plot the angle spectrum. |
| [**annotate**](http://matplotlib.org/api/pyplot_api.html#matplotlib.pyplot.annotate) | Create an annotation: a piece of text referring to a data point. |
| **[arrow](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.arrow" \o "matplotlib.pyplot.arrow)** | Add an arrow to the axes. |
| **[autoscale](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.autoscale" \o "matplotlib.pyplot.autoscale)** | Autoscale the axis view to the data (toggle). |
| **[axes](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.axes" \o "matplotlib.pyplot.axes)** | Add an axes to the figure. |
| **[axhline](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.axhline" \o "matplotlib.pyplot.axhline)** | Add a horizontal line across the axis. |
| **[axhspan](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.axhspan" \o "matplotlib.pyplot.axhspan)** | Add a horizontal span (rectangle) across the axis. |
| **[axis](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.axis" \o "matplotlib.pyplot.axis)** | Convenience method to get or set axis properties. |
| **[axvline](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.axvline" \o "matplotlib.pyplot.axvline)** | Add a vertical line across the axes. |
| **[axvspan](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.axvspan" \o "matplotlib.pyplot.axvspan)** | Add a vertical span (rectangle) across the axes. |
| **[bar](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.bar" \o "matplotlib.pyplot.bar)** | Make a bar plot. |
| [**barbs**](http://matplotlib.org/api/pyplot_api.html#matplotlib.pyplot.barbs) | Plot a 2-D field of barbs. |
| **[barh](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.barh" \o "matplotlib.pyplot.barh)** | Make a horizontal bar plot. |
| **[box](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.box" \o "matplotlib.pyplot.box)** | Turn the axes box on or off. |
| **[boxplot](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.boxplot" \o "matplotlib.pyplot.boxplot)** | Make a box and whisker plot. |
| **[broken\_barh](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.broken_barh" \o "matplotlib.pyplot.broken_barh)** | Plot horizontal bars. |
| [**cla**](http://matplotlib.org/api/pyplot_api.html#matplotlib.pyplot.cla) | Clear the current axes. |
| [**clabel**](http://matplotlib.org/api/pyplot_api.html#matplotlib.pyplot.clabel) | Label a contour plot. |
| [**clf**](http://matplotlib.org/api/pyplot_api.html#matplotlib.pyplot.clf) | Clear the current figure. |
| [**clim**](http://matplotlib.org/api/pyplot_api.html#matplotlib.pyplot.clim) | Set the color limits of the current image. |
| **[close](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.close" \o "matplotlib.pyplot.close)** | Close a figure window. |
| [**cohere**](http://matplotlib.org/api/pyplot_api.html#matplotlib.pyplot.cohere) | Plot the coherence between *x* and *y*. |
| **[colorbar](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.colorbar" \o "matplotlib.pyplot.colorbar)** | Add a colorbar to a plot. |
| **[contour](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.contour" \o "matplotlib.pyplot.contour)** | Plot contours. |
| [**contourf**](http://matplotlib.org/api/pyplot_api.html#matplotlib.pyplot.contourf) | Plot contours. |
| [**csd**](http://matplotlib.org/api/pyplot_api.html#matplotlib.pyplot.csd) | Plot the cross-spectral density. |
| **[delaxes](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.delaxes" \o "matplotlib.pyplot.delaxes)** | Remove an axes from the current figure. |
| **[draw](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.draw" \o "matplotlib.pyplot.draw)** | Redraw the current figure. |
| [**errorbar**](http://matplotlib.org/api/pyplot_api.html#matplotlib.pyplot.errorbar) | Plot an errorbar graph. |
| [**eventplot**](http://matplotlib.org/api/pyplot_api.html#matplotlib.pyplot.eventplot) | Plot identical parallel lines at specific positions. |
| **[figimage](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.figimage" \o "matplotlib.pyplot.figimage)** | Adds a non-resampled image to the figure. |
| **[figlegend](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.figlegend" \o "matplotlib.pyplot.figlegend)** | Place a legend in the figure. |
| **[figtext](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.figtext" \o "matplotlib.pyplot.figtext)** | Add text to figure. |
| [**figure**](http://matplotlib.org/api/pyplot_api.html#matplotlib.pyplot.figure) | Creates a new figure. |
| [**fill**](http://matplotlib.org/api/pyplot_api.html#matplotlib.pyplot.fill) | Plot filled polygons. |
| [**fill\_between**](http://matplotlib.org/api/pyplot_api.html#matplotlib.pyplot.fill_between) | Make filled polygons between two curves. |
| **[fill\_betweenx](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.fill_betweenx" \o "matplotlib.pyplot.fill_betweenx)** | Make filled polygons between two horizontal curves. |
| **[findobj](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.findobj" \o "matplotlib.pyplot.findobj)** | Find artist objects. |
| [**gca**](http://matplotlib.org/api/pyplot_api.html#matplotlib.pyplot.gca) | Get the current **[Axes](http://matplotlib.org/api/axes_api.html" \l "matplotlib.axes.Axes" \o "matplotlib.axes.Axes)** instance on the current figure matching the given keyword args, or create one. |
| **[gcf](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.gcf" \o "matplotlib.pyplot.gcf)** | Get a reference to the current figure. |
| **[gci](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.gci" \o "matplotlib.pyplot.gci)** | Get the current colorable artist. |
| **[get\_figlabels](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.get_figlabels" \o "matplotlib.pyplot.get_figlabels)** | Return a list of existing figure labels. |
| **[get\_fignums](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.get_fignums" \o "matplotlib.pyplot.get_fignums)** | Return a list of existing figure numbers. |
| **[grid](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.grid" \o "matplotlib.pyplot.grid)** | Turn the axes grids on or off. |
| **[hexbin](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.hexbin" \o "matplotlib.pyplot.hexbin)** | Make a hexagonal binning plot. |
| **[hist](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.hist" \o "matplotlib.pyplot.hist)** | Plot a histogram. |
| [**hist2d**](http://matplotlib.org/api/pyplot_api.html#matplotlib.pyplot.hist2d) | Make a 2D histogram plot. |
| **[hlines](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.hlines" \o "matplotlib.pyplot.hlines)** | Plot horizontal lines at each **y** from **xmin** to**xmax**. |
| **[hold](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.hold" \o "matplotlib.pyplot.hold)** | Set the hold state. |
| [**imread**](http://matplotlib.org/api/pyplot_api.html#matplotlib.pyplot.imread) | Read an image from a file into an array. |
| **[imsave](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.imsave" \o "matplotlib.pyplot.imsave)** | Save an array as in image file. |
| **[imshow](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.imshow" \o "matplotlib.pyplot.imshow)** | Display an image on the axes. |
| **[ioff](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.ioff" \o "matplotlib.pyplot.ioff)** | Turn interactive mode off. |
| [**ion**](http://matplotlib.org/api/pyplot_api.html#matplotlib.pyplot.ion) | Turn interactive mode on. |
| [**ishold**](http://matplotlib.org/api/pyplot_api.html#matplotlib.pyplot.ishold) | Return the hold status of the current axes. |
| **[isinteractive](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.isinteractive" \o "matplotlib.pyplot.isinteractive)** | Return status of interactive mode. |
| **[legend](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.legend" \o "matplotlib.pyplot.legend)** | Places a legend on the axes. |
| **[locator\_params](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.locator_params" \o "matplotlib.pyplot.locator_params)** | Control behavior of tick locators. |
| **[loglog](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.loglog" \o "matplotlib.pyplot.loglog)** | Make a plot with log scaling on both the *x* and*y* axis. |
| **[magnitude\_spectrum](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.magnitude_spectrum" \o "matplotlib.pyplot.magnitude_spectrum)** | Plot the magnitude spectrum. |
| [**margins**](http://matplotlib.org/api/pyplot_api.html#matplotlib.pyplot.margins) | Set or retrieve autoscaling margins. |
| **[matshow](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.matshow" \o "matplotlib.pyplot.matshow)** | Display an array as a matrix in a new figure window. |
| **[minorticks\_off](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.minorticks_off" \o "matplotlib.pyplot.minorticks_off)** | Remove minor ticks from the current plot. |
| **[minorticks\_on](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.minorticks_on" \o "matplotlib.pyplot.minorticks_on)** | Display minor ticks on the current plot. |
| **[over](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.over" \o "matplotlib.pyplot.over)** | Call a function with hold(True). |
| **[pause](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.pause" \o "matplotlib.pyplot.pause)** | Pause for *interval* seconds. |
| [**pcolor**](http://matplotlib.org/api/pyplot_api.html#matplotlib.pyplot.pcolor) | Create a pseudocolor plot of a 2-D array. |
| **[pcolormesh](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.pcolormesh" \o "matplotlib.pyplot.pcolormesh)** | Plot a quadrilateral mesh. |
| [**phase\_spectrum**](http://matplotlib.org/api/pyplot_api.html#matplotlib.pyplot.phase_spectrum) | Plot the phase spectrum. |
| [**pie**](http://matplotlib.org/api/pyplot_api.html#matplotlib.pyplot.pie) | Plot a pie chart. |
| [**plot**](http://matplotlib.org/api/pyplot_api.html#matplotlib.pyplot.plot) | Plot lines and/or markers to the **[Axes](http://matplotlib.org/api/axes_api.html" \l "matplotlib.axes.Axes" \o "matplotlib.axes.Axes)**. |
| **[plot\_date](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.plot_date" \o "matplotlib.pyplot.plot_date)** | Plot with data with dates. |
| **[plotfile](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.plotfile" \o "matplotlib.pyplot.plotfile)** | Plot the data in in a file. |
| **[polar](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.polar" \o "matplotlib.pyplot.polar)** | Make a polar plot. |
| [**psd**](http://matplotlib.org/api/pyplot_api.html#matplotlib.pyplot.psd) | Plot the power spectral density. |
| **[quiver](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.quiver" \o "matplotlib.pyplot.quiver)** | Plot a 2-D field of arrows. |
| **[quiverkey](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.quiverkey" \o "matplotlib.pyplot.quiverkey)** | Add a key to a quiver plot. |
| **[rc](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.rc" \o "matplotlib.pyplot.rc)** | Set the current rc params. |
| **[rc\_context](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.rc_context" \o "matplotlib.pyplot.rc_context)** | Return a context manager for managing rc settings. |
| **[rcdefaults](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.rcdefaults" \o "matplotlib.pyplot.rcdefaults)** | Restore the default rc params. |
| **[rgrids](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.rgrids" \o "matplotlib.pyplot.rgrids)** | Get or set the radial gridlines on a polar plot. |
| **[savefig](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.savefig" \o "matplotlib.pyplot.savefig)** | Save the current figure. |
| [**sca**](http://matplotlib.org/api/pyplot_api.html#matplotlib.pyplot.sca) | Set the current Axes instance to *ax*. |
| **[scatter](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.scatter" \o "matplotlib.pyplot.scatter)** | Make a scatter plot of x vs y, where x and y are sequence like objects of the same lengths. |
| **[sci](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.sci" \o "matplotlib.pyplot.sci)** | Set the current image. |
| [**semilogx**](http://matplotlib.org/api/pyplot_api.html#matplotlib.pyplot.semilogx) | Make a plot with log scaling on the *x* axis. |
| **[semilogy](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.semilogy" \o "matplotlib.pyplot.semilogy)** | Make a plot with log scaling on the *y* axis. |
| **[set\_cmap](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.set_cmap" \o "matplotlib.pyplot.set_cmap)** | Set the default colormap. |
| [**setp**](http://matplotlib.org/api/pyplot_api.html#matplotlib.pyplot.setp) | Set a property on an artist object. |
| **[show](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.show" \o "matplotlib.pyplot.show)** | Display a figure. |
| [**specgram**](http://matplotlib.org/api/pyplot_api.html#matplotlib.pyplot.specgram) | Plot a spectrogram. |
| [**spy**](http://matplotlib.org/api/pyplot_api.html#matplotlib.pyplot.spy) | Plot the sparsity pattern on a 2-D array. |
| **[stackplot](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.stackplot" \o "matplotlib.pyplot.stackplot)** | Draws a stacked area plot. |
| **[stem](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.stem" \o "matplotlib.pyplot.stem)** | Create a stem plot. |
| [**step**](http://matplotlib.org/api/pyplot_api.html#matplotlib.pyplot.step) | Make a step plot. |
| [**streamplot**](http://matplotlib.org/api/pyplot_api.html#matplotlib.pyplot.streamplot) | Draws streamlines of a vector flow. |
| **[subplot](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.subplot" \o "matplotlib.pyplot.subplot)** | Return a subplot axes positioned by the given grid definition. |
| **[subplot2grid](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.subplot2grid" \o "matplotlib.pyplot.subplot2grid)** | Create a subplot in a grid. |
| **[subplot\_tool](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.subplot_tool" \o "matplotlib.pyplot.subplot_tool)** | Launch a subplot tool window for a figure. |
| **[subplots](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.subplots" \o "matplotlib.pyplot.subplots)** | Create a figure with a set of subplots already made. |
| **[subplots\_adjust](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.subplots_adjust" \o "matplotlib.pyplot.subplots_adjust)** | Tune the subplot layout. |
| [**suptitle**](http://matplotlib.org/api/pyplot_api.html#matplotlib.pyplot.suptitle) | Add a centered title to the figure. |
| **[switch\_backend](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.switch_backend" \o "matplotlib.pyplot.switch_backend)** | Switch the default backend. |
| [**table**](http://matplotlib.org/api/pyplot_api.html#matplotlib.pyplot.table) | Add a table to the current axes. |
| **[text](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.text" \o "matplotlib.pyplot.text)** | Add text to the axes. |
| **[thetagrids](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.thetagrids" \o "matplotlib.pyplot.thetagrids)** | Get or set the theta locations of the gridlines in a polar plot. |
| **[tick\_params](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.tick_params" \o "matplotlib.pyplot.tick_params)** | Change the appearance of ticks and tick labels. |
| **[ticklabel\_format](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.ticklabel_format" \o "matplotlib.pyplot.ticklabel_format)** | Change the **[ScalarFormatter](http://matplotlib.org/api/ticker_api.html" \l "matplotlib.ticker.ScalarFormatter" \o "matplotlib.ticker.ScalarFormatter)** used by default for linear axes. |
| **[tight\_layout](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.tight_layout" \o "matplotlib.pyplot.tight_layout)** | Automatically adjust subplot parameters to give specified padding. |
| **[title](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.title" \o "matplotlib.pyplot.title)** | Set a title of the current axes. |
| **[tricontour](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.tricontour" \o "matplotlib.pyplot.tricontour)** | Draw contours on an unstructured triangular grid. |
| **[tricontourf](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.tricontourf" \o "matplotlib.pyplot.tricontourf)** | Draw contours on an unstructured triangular grid. |
| **[tripcolor](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.tripcolor" \o "matplotlib.pyplot.tripcolor)** | Create a pseudocolor plot of an unstructured triangular grid. |
| **[triplot](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.triplot" \o "matplotlib.pyplot.triplot)** | Draw a unstructured triangular grid as lines and/or markers. |
| **[twinx](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.twinx" \o "matplotlib.pyplot.twinx)** | Make a second axes that shares the *x*-axis. |
| **[twiny](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.twiny" \o "matplotlib.pyplot.twiny)** | Make a second axes that shares the *y*-axis. |
| **[violinplot](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.violinplot" \o "matplotlib.pyplot.violinplot)** | Make a violin plot. |
| [**vlines**](http://matplotlib.org/api/pyplot_api.html#matplotlib.pyplot.vlines) | Plot vertical lines. |
| [**xcorr**](http://matplotlib.org/api/pyplot_api.html#matplotlib.pyplot.xcorr) | Plot the cross correlation between *x* and *y*. |
| **[xkcd](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.xkcd" \o "matplotlib.pyplot.xkcd)** | Turns on [xkcd](http://xkcd.com/) sketch-style drawing mode. |
| **[xlabel](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.xlabel" \o "matplotlib.pyplot.xlabel)** | Set the *x* axis label of the current axis. |
| **[xlim](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.xlim" \o "matplotlib.pyplot.xlim)** | Get or set the *x* limits of the current axes. |
| **[xscale](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.xscale" \o "matplotlib.pyplot.xscale)** | Set the scaling of the *x*-axis. |
| **[xticks](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.xticks" \o "matplotlib.pyplot.xticks)** | Get or set the *x*-limits of the current tick locations and labels. |
| **[ylabel](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.ylabel" \o "matplotlib.pyplot.ylabel)** | Set the *y* axis label of the current axis. |
| **[ylim](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.ylim" \o "matplotlib.pyplot.ylim)** | Get or set the *y*-limits of the current axes. |
| **[yscale](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.yscale" \o "matplotlib.pyplot.yscale)** | Set the scaling of the *y*-axis. |
| **[yticks](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.yticks" \o "matplotlib.pyplot.yticks)** | Get or set the *y*-limits of the current tick locations and labels. |

matplotlib.pyplot.**colormaps**()

Matplotlib provides a number of colormaps, and others can be added using **[register\_cmap()](http://matplotlib.org/api/cm_api.html" \l "matplotlib.cm.register_cmap" \o "matplotlib.cm.register_cmap)**. This function documents the built-in colormaps, and will also return a list of all registered colormaps if called.

You can set the colormap for an image, pcolor, scatter, etc, using a keyword argument:

imshow(X, cmap=cm.hot)

or using the **[set\_cmap()](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.set_cmap" \o "matplotlib.pyplot.set_cmap)** function:

imshow(X)

pyplot.set\_cmap('hot')

pyplot.set\_cmap('jet')

In interactive mode, **[set\_cmap()](http://matplotlib.org/api/pyplot_api.html" \l "matplotlib.pyplot.set_cmap" \o "matplotlib.pyplot.set_cmap)** will update the colormap post-hoc, allowing you to see which one works best for your data.

All built-in colormaps can be reversed by appending \_r: For instance,gray\_r is the reverse of gray.

There are several common color schemes used in visualization:

Sequential schemes

for unipolar data that progresses from low to high

Diverging schemes

for bipolar data that emphasizes positive or negative deviations from a central value

Cyclic schemes

meant for plotting values that wrap around at the endpoints, such as phase angle, wind direction, or time of day

Qualitative schemes

for nominal data that has no inherent ordering, where color is used only to distinguish categories

The base colormaps are derived from those of the same name provided with Matlab:

| **Colormap** | **Description** |
| --- | --- |
| autumn | sequential linearly-increasing shades of red-orange-yellow |
| bone | sequential increasing black-white color map with a tinge of blue, to emulate X-ray film |
| cool | linearly-decreasing shades of cyan-magenta |
| copper | sequential increasing shades of black-copper |
| flag | repetitive red-white-blue-black pattern (not cyclic at endpoints) |
| gray | sequential linearly-increasing black-to-white grayscale |
| hot | sequential black-red-yellow-white, to emulate blackbody radiation from an object at increasing temperatures |
| hsv | cyclic red-yellow-green-cyan-blue-magenta-red, formed by changing the hue component in the HSV color space |
| jet | a spectral map with dark endpoints, blue-cyan-yellow-red; based on a fluid-jet simulation by NCSA [[1]](http://matplotlib.org/api/pyplot_summary.html#id6) |
| pink | sequential increasing pastel black-pink-white, meant for sepia tone colorization of photographs |
| prism | repetitive red-yellow-green-blue-purple-...-green pattern (not cyclic at endpoints) |
| spring | linearly-increasing shades of magenta-yellow |
| summer | sequential linearly-increasing shades of green-yellow |
| winter | linearly-increasing shades of blue-green |

For the above list only, you can also set the colormap using the corresponding pylab shortcut interface function, similar to Matlab:

imshow(X)

hot()

jet()

The next set of palettes are from the [Yorick scientific visualisation package](http://yorick.sourceforge.net/index.php), an evolution of the GIST package, both by David H. Munro:

| **Colormap** | **Description** |
| --- | --- |
| gist\_earth | mapmaker’s colors from dark blue deep ocean to green lowlands to brown highlands to white mountains |
| gist\_heat | sequential increasing black-red-orange-white, to emulate blackbody radiation from an iron bar as it grows hotter |
| gist\_ncar | pseudo-spectral black-blue-green-yellow-red-purple-white colormap from National Center for Atmospheric Research [[2]](http://matplotlib.org/api/pyplot_summary.html#id7) |
| gist\_rainbow | runs through the colors in spectral order from red to violet at full saturation (like *hsv*but not cyclic) |
| gist\_stern | “Stern special” color table from Interactive Data Language software |

The following colormaps are based on the [ColorBrewer](http://colorbrewer.org/) color specifications and designs developed by Cynthia Brewer:

ColorBrewer Diverging (luminance is highest at the midpoint, and decreases towards differently-colored endpoints):

| **Colormap** | **Description** |
| --- | --- |
| BrBG | brown, white, blue-green |
| PiYG | pink, white, yellow-green |
| PRGn | purple, white, green |
| PuOr | orange, white, purple |
| RdBu | red, white, blue |
| RdGy | red, white, gray |
| RdYlBu | red, yellow, blue |
| RdYlGn | red, yellow, green |
| Spectral | red, orange, yellow, green, blue |

ColorBrewer Sequential (luminance decreases monotonically):

| **Colormap** | **Description** |
| --- | --- |
| Blues | white to dark blue |
| BuGn | white, light blue, dark green |
| BuPu | white, light blue, dark purple |
| GnBu | white, light green, dark blue |
| Greens | white to dark green |
| Greys | white to black (not linear) |
| Oranges | white, orange, dark brown |
| OrRd | white, orange, dark red |
| PuBu | white, light purple, dark blue |
| PuBuGn | white, light purple, dark green |
| PuRd | white, light purple, dark red |
| Purples | white to dark purple |
| RdPu | white, pink, dark purple |
| Reds | white to dark red |
| YlGn | light yellow, dark green |
| YlGnBu | light yellow, light green, dark blue |
| YlOrBr | light yellow, orange, dark brown |
| YlOrRd | light yellow, orange, dark red |

ColorBrewer Qualitative:

(For plotting nominal data, **ListedColormap** should be used, not**LinearSegmentedColormap**. Different sets of colors are recommended for different numbers of categories. These continuous versions of the qualitative schemes may be removed or converted in the future.)

* Accent
* Dark2
* Paired
* Pastel1
* Pastel2
* Set1
* Set2
* Set3

Other miscellaneous schemes:

| **Colormap** | **Description** |
| --- | --- |
| afmhot | sequential black-orange-yellow-white blackbody spectrum, commonly used in atomic force microscopy |
| brg | blue-red-green |
| bwr | diverging blue-white-red |
| coolwarm | diverging blue-gray-red, meant to avoid issues with 3D shading, color blindness, and ordering of colors [[3]](http://matplotlib.org/api/pyplot_summary.html#id8) |
| CMRmap | “Default colormaps on color images often reproduce to confusing grayscale images. The proposed colormap maintains an aesthetically pleasing color image that automatically reproduces to a monotonic grayscale with discrete, quantifiable saturation levels.” [[4]](http://matplotlib.org/api/pyplot_summary.html#id9) |
| cubehelix | Unlike most other color schemes cubehelix was designed by D.A. Green to be monotonically increasing in terms of perceived brightness. Also, when printed on a black and white postscript printer, the scheme results in a greyscale with monotonically increasing brightness. This color scheme is named cubehelix because the r,g,b values produced can be visualised as a squashed helix around the diagonal in the r,g,b color cube. |
| gnuplot | gnuplot’s traditional pm3d scheme (black-blue-red-yellow) |
| gnuplot2 | sequential color printable as gray (black-blue-violet-yellow-white) |
| ocean | green-blue-white |
| rainbow | spectral purple-blue-green-yellow-orange-red colormap with diverging luminance |
| seismic | diverging blue-white-red |
| nipy\_spectral | black-purple-blue-green-yellow-red-white spectrum, originally from the Neuroimaging in Python project |
| terrain | mapmaker’s colors, blue-green-yellow-brown-white, originally from IGOR Pro |

The following colormaps are redundant and may be removed in future versions. It’s recommended to use the names in the descriptions instead, which produce identical output:

| **Colormap** | **Description** |
| --- | --- |
| gist\_gray | identical to *gray* |
| gist\_yarg | identical to *gray\_r* |
| binary | identical to *gray\_r* |
| spectral | identical to *nipy\_spectral* [[5]](http://matplotlib.org/api/pyplot_summary.html#id10) |

**Footnotes**

|  |  |
| --- | --- |
| [[1]](http://matplotlib.org/api/pyplot_summary.html#id1) | Rainbow colormaps, jet in particular, are considered a poor choice for scientific visualization by many researchers: [Rainbow Color Map (Still) Considered Harmful](http://www.jwave.vt.edu/~rkriz/Projects/create_color_table/color_07.pdf) |

|  |  |
| --- | --- |
| [[2]](http://matplotlib.org/api/pyplot_summary.html#id2) | Resembles “BkBlAqGrYeOrReViWh200” from NCAR Command Language. See [Color Table Gallery](http://www.ncl.ucar.edu/Document/Graphics/color_table_gallery.shtml) |

|  |  |
| --- | --- |
| [[3]](http://matplotlib.org/api/pyplot_summary.html#id3) | See [Diverging Color Maps for Scientific Visualization](http://www.cs.unm.edu/~kmorel/documents/ColorMaps/) by Kenneth Moreland. |

|  |  |
| --- | --- |
| [[4]](http://matplotlib.org/api/pyplot_summary.html#id4) | See [A Color Map for Effective Black-and-White Rendering of Color-Scale Images](http://www.mathworks.com/matlabcentral/fileexchange/2662-cmrmap-m) by Carey Rappaport |

|  |  |
| --- | --- |
| [[5]](http://matplotlib.org/api/pyplot_summary.html#id5) | Changed to distinguish from ColorBrewer’s *Spectral* map.[**spectral()**](http://matplotlib.org/api/pyplot_api.html#matplotlib.pyplot.spectral) still works, but set\_cmap('nipy\_spectral') is recommended for clarity. |