Matplotlib for Python Developers `pyplot` Summary Sheet

B.W. Keller

Section 1: Diving in to Matplotlib

1.3   Beginning with the most basic plots

Importing Matplotlib

- `import matplotlib.pyplot as plt` will import all of the matplotlib convenience methods

- `%matplotlib inline` will allow the Jupyter notebook to display matplotlib figures.

matplotlib.pyplot methods

Method	Argument	Description
`plot()`		Generate a line or point plot
	[first]	X values, required
	[second]	Y values, required
	[third]	Format string, optional
	label=	String to be used for labelling in the legend
	marker=	Marker character to be used for points
	color=	Color of line and marker
	linestyle=	Style of line joining points
`xlabel()`		Set the x label
	[first]	String to show for the x label
`ylabel()`		Set the y label
	[first]	String to show for the y label
`xlim()`		Set the domain (x max/min)
	[first]	Minimum value for x axis
	[second]	Maximum value for x axis
`legend()`		Display a legend showing the labels for each plot
`hist()`		Use bins to display a Probability Distribution Function of an array of numbers
`imshow()`		Display a NxM array, coloring each point using the value in the array
	cmap=	select the color map that translates values to colors
`colorbar()`		Display a colorbar showing the numerical values of colors

Section 2: Basic Plotting Functions

2.4   Differentiating Line and Scatter Plots

matplotlib.pyplot methods

Method	Argument	Description
`plot()`		Generate a line or point plot
	[first]	X values, required
	[second]	Y values, required
	[third]	Format string, optional
	label=	String to be used for labelling in the legend
	marker=	Marker character to be used for points (can be LaTeXstring)
	color=	Color of line and marker
	linestyle=	Style of line joining points
	markerevery=	How frequently to put a marker down
	ms=	Size of the marker
	linewidth=	Width of the line joining points
	zorder=	"height" of line and marker, relative to others
	alpha=	Level of transparency
`scatter()`		Generate a scatter plot, with unique markers/colors
	[first]	X values, required
	[second]	Y values, required
	c=	single color or array of colors for markers
	cmap=	select the color map that translates values to colors
	s=	single size or array of sizes for markers
	edgecolor=	color for marker edges

2.5   Bar Plots and Histograms

matplotlib.pyplot methods

Method	Argument	Description
`bar()`		Generate a vertical bar plot
	[first]	left edges of the bars
	height=	array with height of the bars
	color=	fill color of the bars
	edgecolor=	edge color of the bars
	align=	Where to place the bars relative to the first argument
	hatch=	What pattern to fill the bars with
	width=	Width of the bars
	bottom=	array of bottoms for the bars
`barh()`		Generate a horizontal bar plot
	[first]	left edges of the bars
	width=	array with width of the bars
`hist()`		Use bins to display a Probability Distribution Function of an array of numbers
	bins=	Number of bins to use
	histtype=	Style of histogram to show
	normed=	Normalize the histogram
	cumulative=	Make a cumulative distribution function
	stacked=	Stack multiple histograms on top of each other

2.6 Images and Contours

matplotlib.pyplot methods

Method	Argument	Description
`imshow()`		Generate an image plot
	[first]	NxM array to generate image with
	cmap=	select the color map that translates values to colors
	extent=	4 element tuple for the corners: (xmin, xmax, ymin, ymax)
	vmin=	Minimum value for colormap
	vmax=	Maximum value for colormap
	aspect=	Aspect ration (height/width) of image
	interpolation=	Method to interpolate between pixels
`matshow()`		Generate an image plot without interpolation and alternative label positions
	[first]	NxM array to generate image with
`contour()`		Generate contours of isovalues
	[first]	NxM array to generate image with
	[second]	Number of contours to apply
	levels=	What values to apply contours to
	colors=	List of colors to use for contours
	linestyles=	List of line styles to use for contours
	cmap=	select the color map that translates values to colors
`contourf()`		Generate filled contours of isovalues
	[first]	NxM array to generate image with
	[second]	Number of contours to apply
	hatch=	What pattern to fill the contours with
`clabel()`		Apply a text label to contours

2.7 Plots with Uncertainties

matplotlib.pyplot methods

Method	Argument	Description
`errorbar()`		Make a line or scatter plot with uncertainties
	[first]	X values, required
	[second]	Y values, required
	xerr=	Lengths of x error bars
	yerr=	Lengths of y error bars
	ecolor=	Color of error bars
	elinewidth=	Line width of error bars
	capsize=	Size of error bar caps
`bar()`		Generate a vertical bar plot
	[first]	left edges of the bars
	height=	array with height of the bars
	xerr=	Lengths of x error bars
	yerr=	Lengths of y error bars

2.8   Other Useful Plot Types

matplotlib.pyplot methods

Method	Argument	Description
`fill_between()`		Generate a filled area between two curves
	[first]	x positions
	[second]	y positions of top curve
	[third]	y positions of bottom curve (optional)
	hatch=	What pattern to fill the area with
`hexbin()`		Generate a 2D histogram of points using hexagonal bins
	[first]	x positions of points
	[second]	y positions of points
	bins=	Number of color bins to use
	mincnt=	Minimum number of points to display a bin
`hist2d()`		Generate a 2D histogram of points using rectangular bins

2.9   Multiple Panel Plots

matplotlib.pyplot methods

Method	Argument	Description
`subplot()`		Generate a grid of subplots, and activate the current subplot
	[first]	Three digits showing number of rows, columns, and subplot position
`subplots()`		Return a figure object and a list of subplot axes
	[first]	Number of rows
	[second]	Number of columns

2.10   Legends and Colorbars

matplotlib.pyplot methods

Method	Argument	Description
`legend()`		Make a legend showing labels of plots
	loc=	Where to place the legend
	ncol=	Number of columns in the legend
	title=	Title for the legend
	fontsize=	Size of the legend font
`colorbar()`		Make a colorbar
	orientation=	Horizontal or vertical colorbar