Pandas Data Visualization

* Pandas data visualization tools are built off matplotlib.
* Pandas data visualization lets you call plots off a data frame directly.
* For a time series data, you can set the time column as the index by calling index\_col = a where a is the time column title.
* To visualize with pandas, simply call the plot type on the data frame or column for plotting the whole data frame or column alone respectively. E.g., df.hist() to plot a histogram of the data frame or df[‘b’].hist() to plot a histogram of a given column with column head ‘b’ in data frame df.
* You can improve the visual aesthetics of the pandas plots automatically to look like seaborn plots by default by simply importing seaborn in the notebook.
* You can customize the charts with seaborn or matplotlib like arguments like bins, bin size, marker, marker size etc. by passing the argument in the brackets. E.g., df[‘b’].hist(bins = l) where l is an integer.
* The plot types available in pandas are: area plot, density plots, histograms (hist), line plot, scatter plot, bar plot, box plot, hexbin, kde, pie.
* You can plot any of the plot types with df[‘b’].plot(kind = ‘c’) where c is the desired plot type e.g., df[‘b’].plot(kind = ‘scatter’)
* If you use the above method, you can then pass in the typical customization arguments for that plot type from matplotlib or seaborn. E.g., df[‘b’].plot(kind = ‘hist’, bins=l).
* An easier way is to just call the plot type off the ‘.plot’ command e.g., df[‘b’].plot.hist()
* An area plot can also take an alpha argument to make it more transparent.
* You can stack a bar chat by passing in stacked = True.
* You can make the scatter plot from pandas visualization show the points or marker size based off the value of another column that wasn’t plotted e.g. df.plot.scatter(x = ‘A’, y = ‘B’, c = ‘C’) where c represents color and C is the column head or df.plot.scatter(x = ‘A’, y = ‘B’, s = df[‘C’]) where s is size.
* You can use hexbin to plot bivariate data.
* You can increase the size of the hexagons by passing in the gridsize argument, the color with cmap argument etc.

Note: Seaborn and matplotlib are better off for advanced plot types, customization, etc.