Data Visualization with Seaborn, Bokeh and Spark

Mahidhar Tatineni
User Services, SDSC

Costa Rica Big Data School
December 8, 2017

Mt. Whitney (14,454 ft)

Data Visualization Tools

- **Several open source tools available to aid visualization with Spark: *Seaborn, Bokeh, and Zeppelin*.**
- **Seaborn**
  - Statistical data visualization package interfaced with matplotlib
- **Bokeh**
  - Interactive viz library targeting web browsers for output, interactivity with large datasets, data applications. Compatible with Jupyter notebooks.
- **Zeppelin**
  - Web based notebook for data driven interactive analytics, with Spark integration.
- **Examples of Seaborn, Bokeh usage on CRBDS resources.**
- **End to end example of data analytics workflow using real world example and Seaborn.**

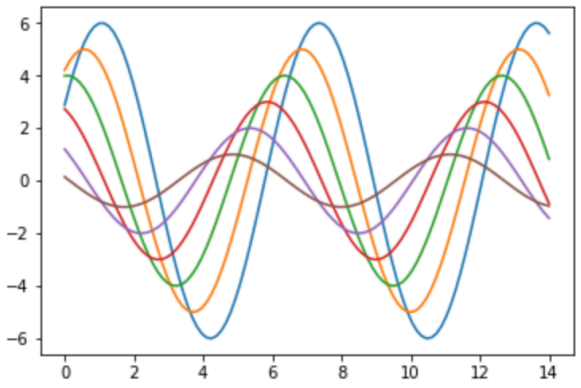
Seaborn Features

(https://seaborn.pydata.org/index.html)

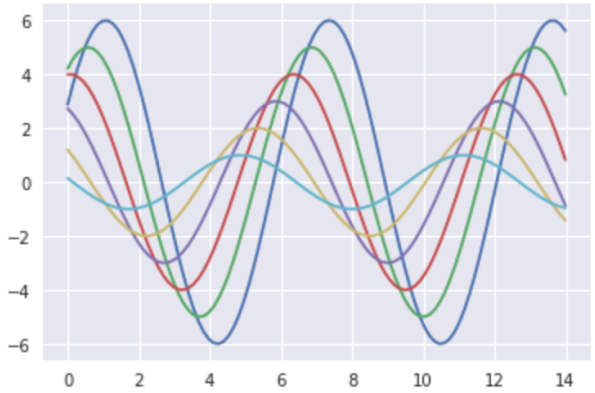
- Integrated with PyData.
- Handles numpy, pandas data structures
- Statistical routines from scipy, statsmodels.
- Built-in themes to help styling of matplotlib graphics.
- Tools to enable custom color palettes.
- Visualization functions for univariate, bivariate distributions. Comparisons between subsets of data.
- Fitting and visualizing linear regression models
- Matrix data visualizations
- Statistical timeseries data
- Grids of plots - combining several plots for complex visualization.

Seaborn: Aesthetic Enhancments of Figures

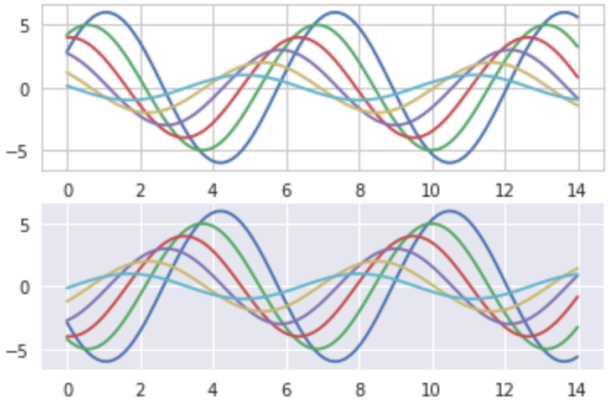
- Seaborn themes - darkgrid, whitegrid, dark, white, ticks.
- Scaling plot elements - paper, notebook, talk, and poster.
- Can choose color palettes
- Example : Lets plot offset sine waves:



- Default matplotlib



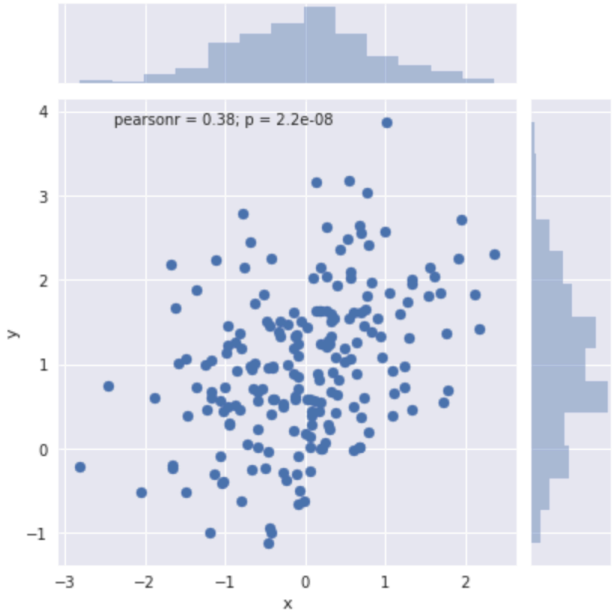
- Default Seaborn



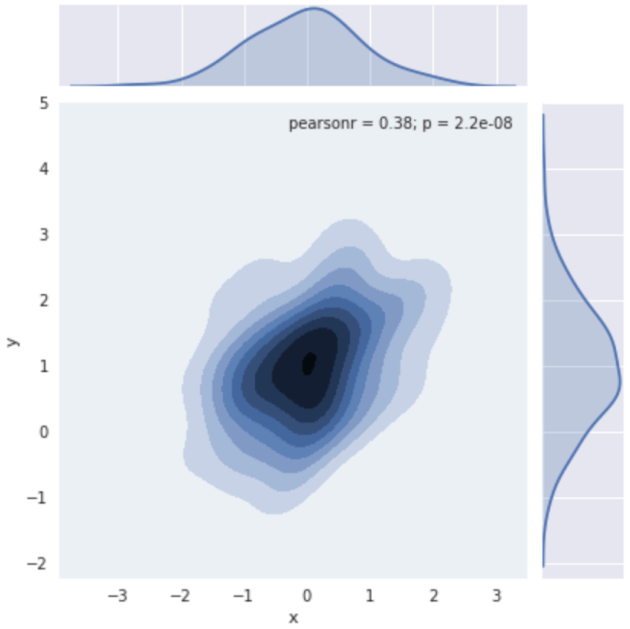
- Whitegrid vs Darkgrid
- Set using axes_style()

Seaborn: Plotting Functions

- Visualizing univariate, bivariate distributions
- Plotting categorical data
- Visualizing linear relationships

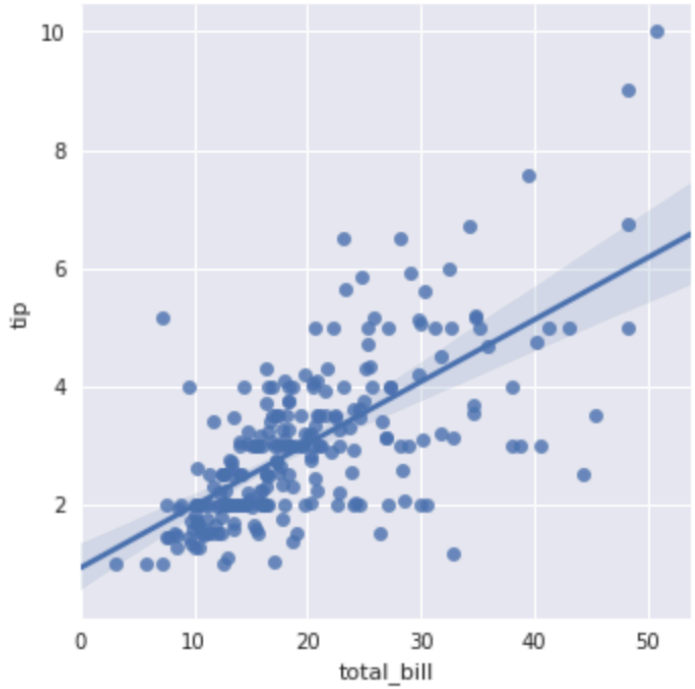


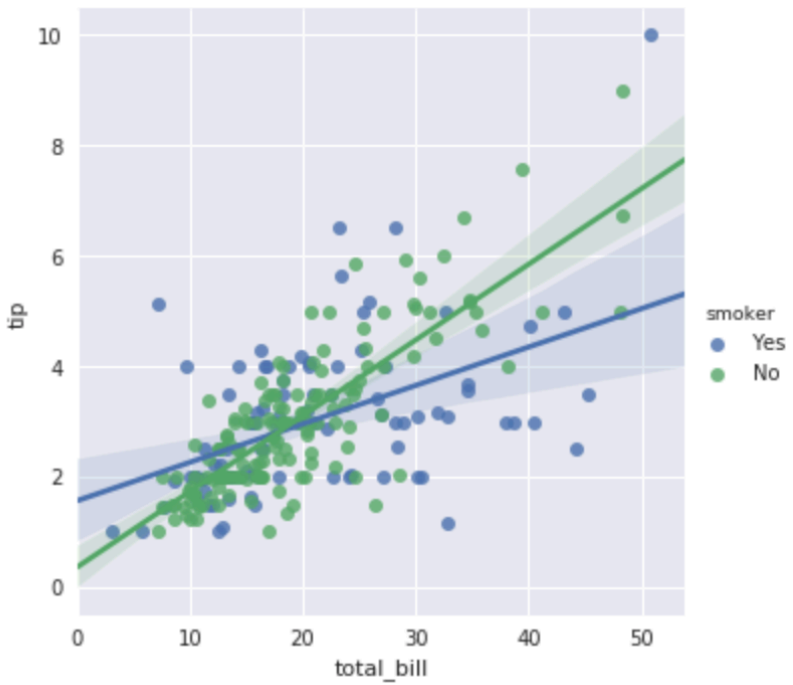
Bivariate distribution scatterplot



Bivariate distribution kernel density estimation

Seaborn: Plotting Functions



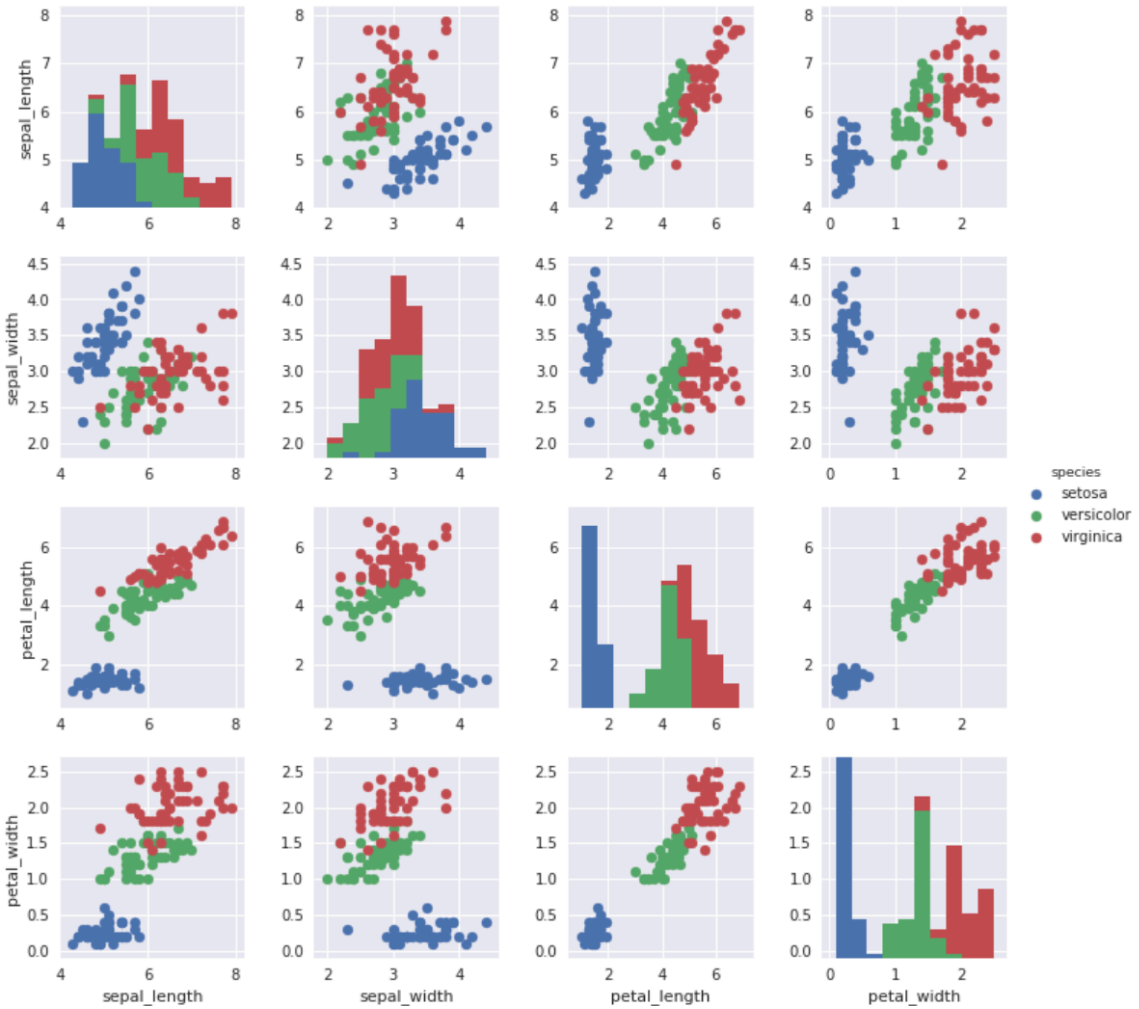


Two variable scatterplot, linear regression + 95% confidence interval
sns.lmplot(x="total_bill", y="tip", data=tips);

Two variable scatterplot, linear regression + 95% confidence interval + conditional
sns.lmplot(x="total_bill", y="tip", hue="smoker", data=tips);

Seaborn: Grid of Plots

```
In [34]: g = sns.PairGrid(iris, hue="species")
         g.map_diag(plt.hist)
         g.map_offdiag(plt.scatter)
         g.add_legend();
```

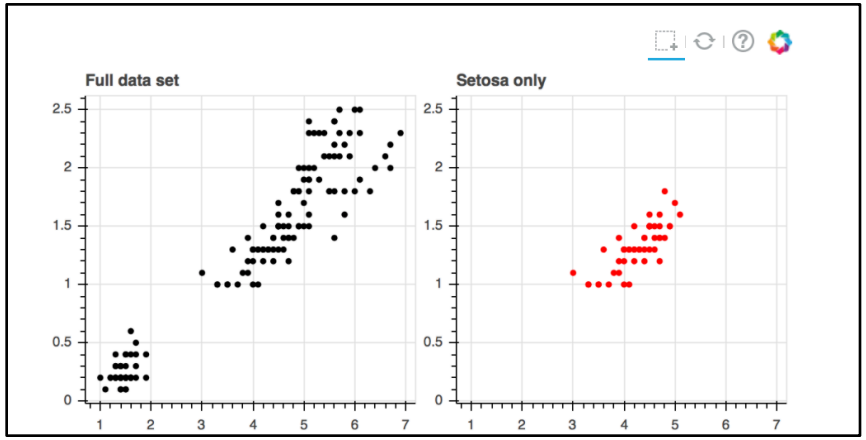


Bokeh

- Python interactive visualization library
- Visualization via web browsers
- Interactive visualization over very large or streaming datasets
- Available via Jupyter notebooks
- Can be connected to Spark
- Tools available for handling:
  - Categorical data
  - Network graphs with configurable node and edge interactions
  - Mapping Geo Data
  - Making interactive tools such as pan, zoom, select etc.
  - Styling visual attributes
- Server option to build and publish applications
- Plots/Apps can be embedded into HTML documents
- Can leverage other libraries such as Datashader, HoloViews.

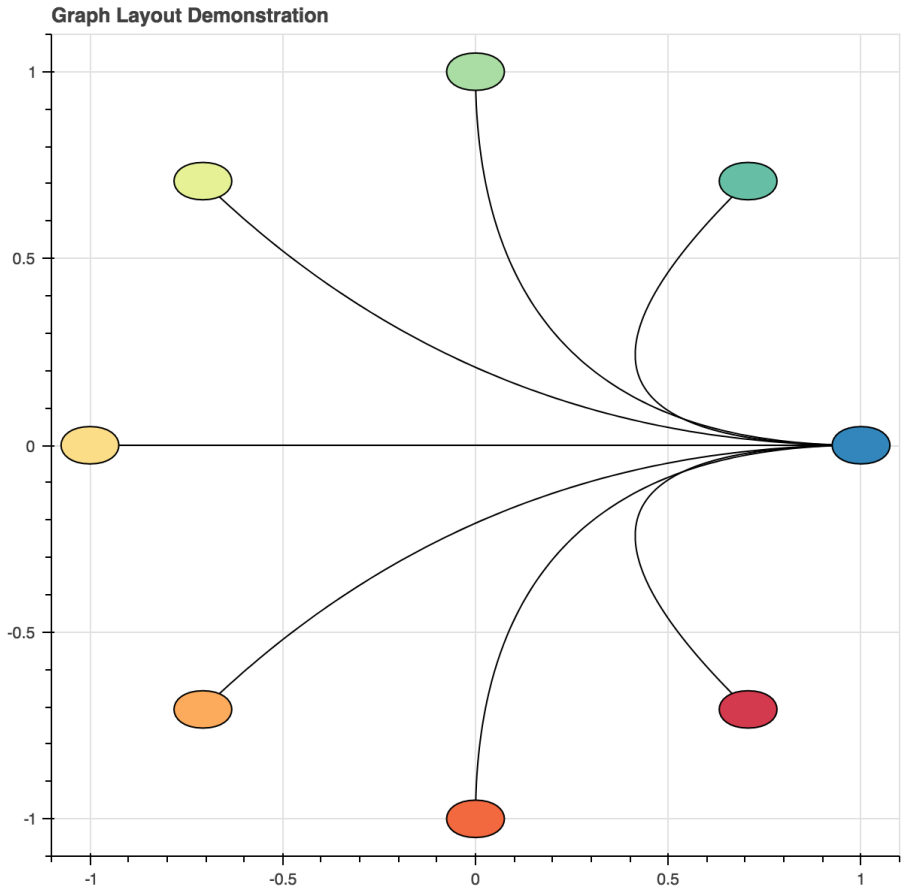
Bokeh: Providing Data for Plots

- Directly pass list of values
- ColumnDataSource, DataFrames
- Filtered data with CDSView - IndexFilter, Boolean Filter, GroupFilter, CustomJSFilter
- Code snippet from notebook below:



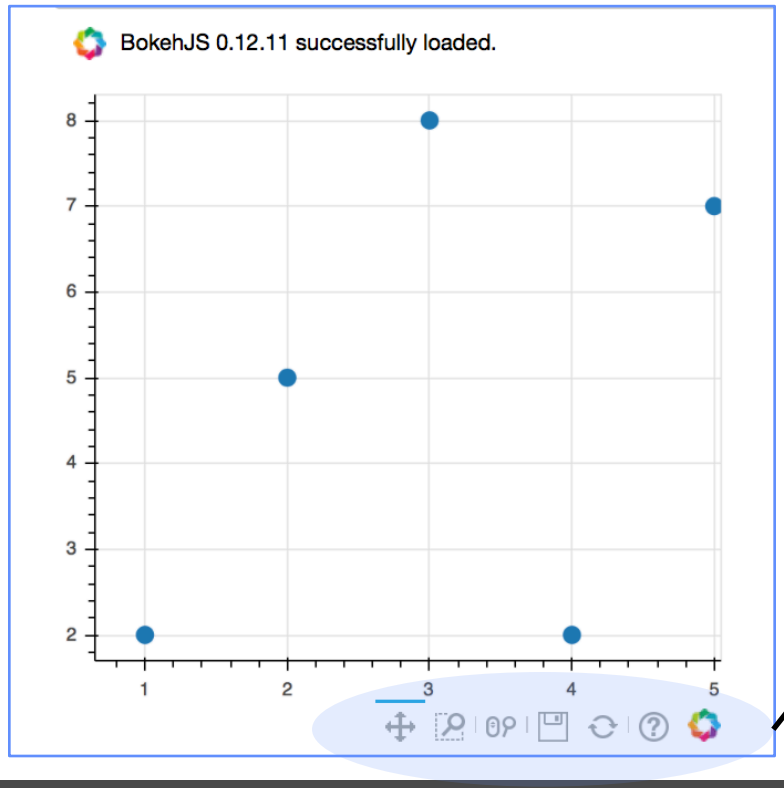

```
from bokeh.plotting import figure, output_file, show
from bokeh.layouts import gridplot
from bokeh.models import ColumnDataSource, CDSView, GroupFilter

from bokeh.sampledata.iris import flowers
from bokeh.io import output_notebook

output_notebook()

source = ColumnDataSource(flowers)
view1 = CDSView(source=source, filters=[GroupFilter(column_name='species', group='versicolor')])

plot_size_and_tools = {'plot_height': 300, 'plot_width': 300,
                       'tools':['box_select', 'reset', 'help']}

p1 = figure(title="Full data set", **plot_size_and_tools)
p1.circle(x='petal_length', y='petal_width', source=source, color='black')

p2 = figure(title="Setosa only", x_range=p1.x_range, y_range=p1.y_range, **plot_size_and_tools)
p2.circle(x='petal_length', y='petal_width', source=source, view=view1, color='red')

show(gridplot([[p1, p2]]))
```

Bokeh: Network Graphs example



Bokeh: Interactive Tools

- Gestures - *Pan/Drag, Click/Tap, Scroll/Pinch*
- Actions - *Undo, Redo, Reset, Save, Zoom In, Zoom Out.*
- Inspectors - *Crosshair, Hover*




```
p = figure(plot_width=400, plot_height=400,
title=None, toolbar_location="below",
toolbar_sticky=False)
```

Bokeh in Notebooks

- Easy integration with Jupyter, Zeppelin Notebooks.
- Import output_notebook() from bokeh.io
- Keep everything else the same and the next show() will display the content in the notebook.
- All the snapshots in this presentation are from documentation examples run in a notebook on CRBDS resources.

Hands On / Demo Session

- Seaborn.ipynb - notebook with Seaborn examples (from documentation)
- Bokeh.ipynb - notebook with Bokeh examples (from documentation)
- spark_nba.ipynb - notebook with an sports analytics example that leverages ipython, Spark, and Seaborn. Example developed by Chris Rawles. Reference: https://content.pivotal.io/blog/how-data-science-assists-sports

References

- **Seaborn**

https://seaborn.pydata.org/index.html

- **Bokeh:**

https://bokeh.pydata.org/en/0.12.11/docs/user_guide.html#

- **Sports Data Science Example:**

https://content.pivotal.io/blog/how-data-science-assists-sports

https://github.com/crawles/spark-nba-analytics