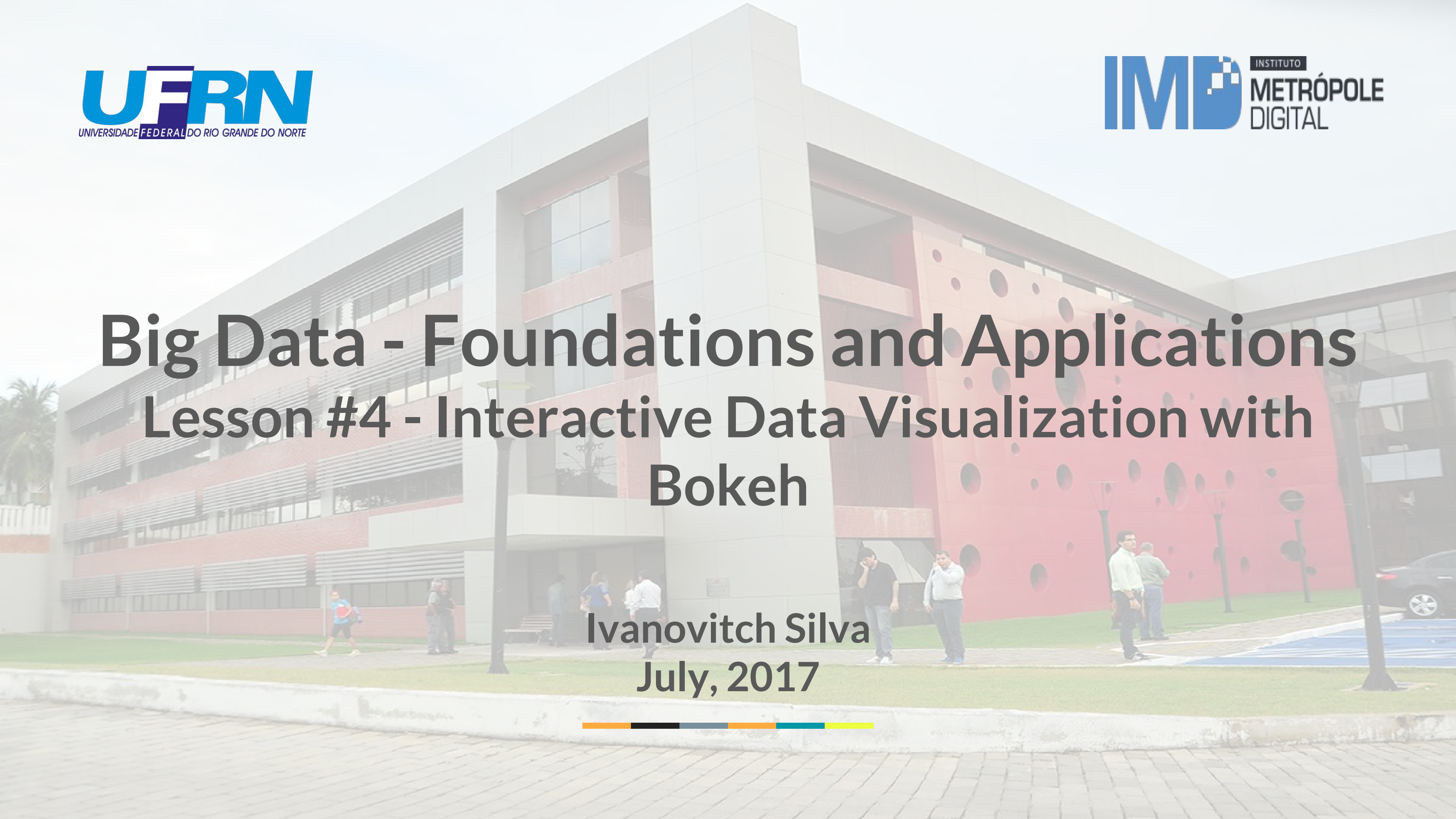


UFRN UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE

IMD INSTITUTO METRÓPOLE DIGITAL

Big Data - Foundations and Applications

Lesson #4 - Interactive Data Visualization with Bokeh

Ivanovitch Silva
July, 2017

Agenda

Part I

- Introduction about Bokeh
- Glyphs
- Lines
- Plot (numpy, pandas)
- Column data source

Part II

- Layout
- Rows of plot
- Columns of plot
- Nested plots
- Grid plots
- Tabbed plots
- Links selections
- Legends

What is Bokeh?

http://bokeh.pydata.org/en/latest/

- Interactive visualization, controls, and tools
- Versatile and high-level graphics
- High-level statistical charts
- Streaming, dynamic, large data
- For the browser, with or without a server
- No JavaScript

#	Manufacture	Model	Displacemer	Year	Cylinders	Transmissio	Drive	Class	City MPG	Highway MP
32	**Chevrolet**	Malibu	2.4	1999	4	auto(l4)	front	midsize	19	27
33	**Chevrolet**	Malibu	2.4	2008	4	auto(l4)	front	midsize	22	30
34	**Chevrolet**	Malibu	3.1	1999	6	auto(l4)	front	midsize	18	26
35	**Chevrolet**	Malibu	3.5	2008	6	auto(l4)	front	midsize	18	29
36	**Chevrolet**	Malibu	3.6	2008	6	auto(s6)	front	midsize	17	26
37	**Dodge**	Caravan ...	2.4	1999	4	auto(l3)	front	minivan	18	24
38	**Dodge**	Caravan ...	3.0	1999	6	auto(l4)	front	minivan	17	24
39	**Dodge**	Caravan ...	3.3	1999	6	auto(l4)	front	minivan	16	22
40	**Dodge**	Caravan ...	3.3	1999	6	auto(l4)	front	minivan	16	22
41	**Dodge**	Caravan ...	3.3	2008	6	auto(l4)	front	minivan	17	24
42	**Dodge**	Caravan ...	3.3	2008	6	auto(l4)	front	minivan	17	24
43	**Dodge**	Caravan ...	3.3	2008	6	auto(l4)	front	minivan	11	17
44	**Dodge**	Caravan ...	3.8	1999	6	auto(l4)	front	minivan	15	22
45	**Dodge**	Caravan ...	3.8	1999	6	auto(l4)	front	minivan	15	21
46	**Dodge**	Caravan ...	3.8	2008	6	auto(l6)	front	minivan	16	23

Plotting with Glyphs

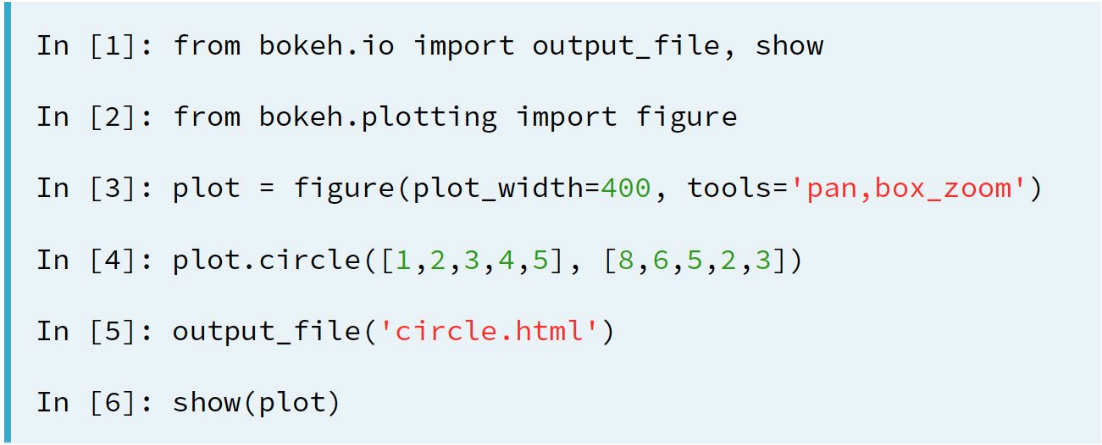
What are Glyphs

- visual shapes
- circles, squares, triangles
- rectangles, lines, wedges

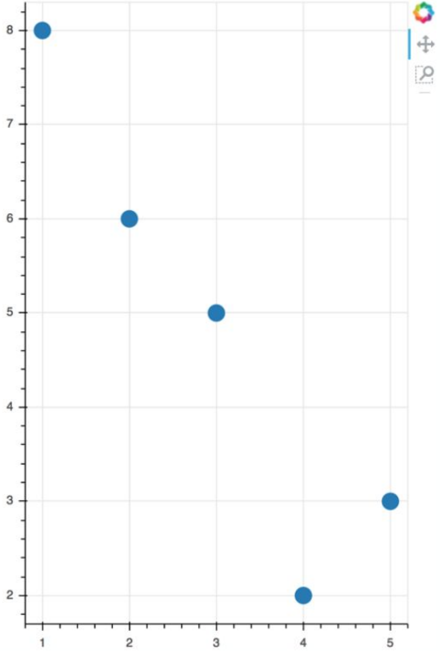
With properties attached to data

- coordinates (x,y)
- size, color, transparency

Typical usage



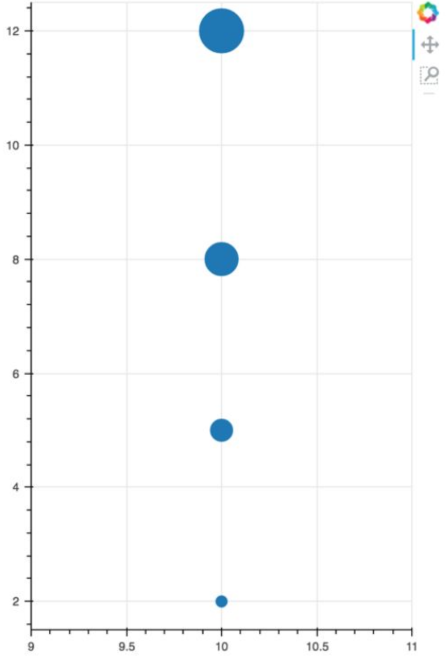
```
In [1]: from bokeh.io import output_file, show

In [2]: from bokeh.plotting import figure

In [3]: plot = figure(plot_width=400, tools='pan,box_zoom')

In [4]: plot.circle([1,2,3,4,5], [8,6,5,2,3])

In [5]: output_file('circle.html')

In [6]: show(plot)
```



Glyph properties

- Lists, arrays, sequences of values
- Single fixed values

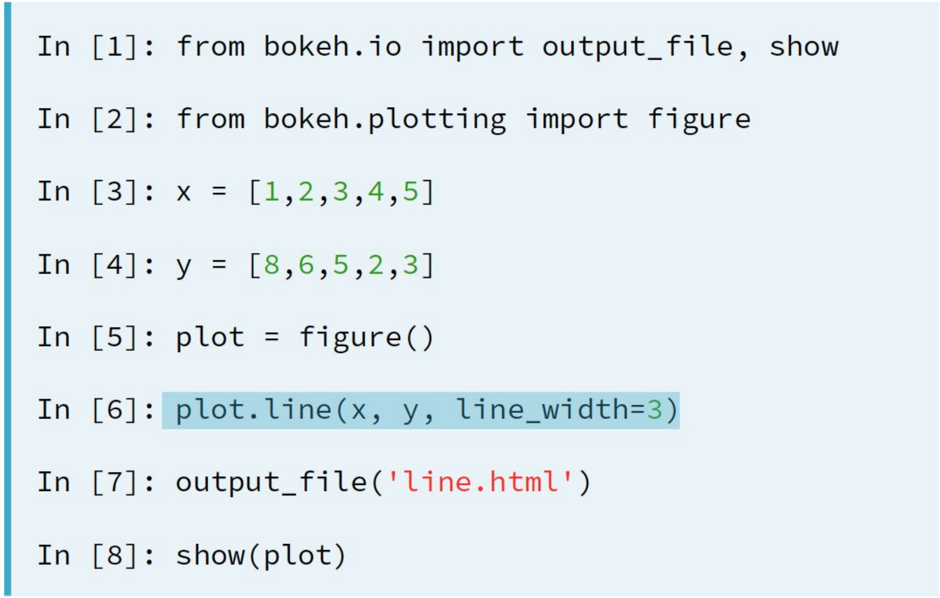
```
In [1]: plot = figure()

In [2]: plot.circle(x=10, y=[2,5,8,12], size=[10,20,30,40])
```



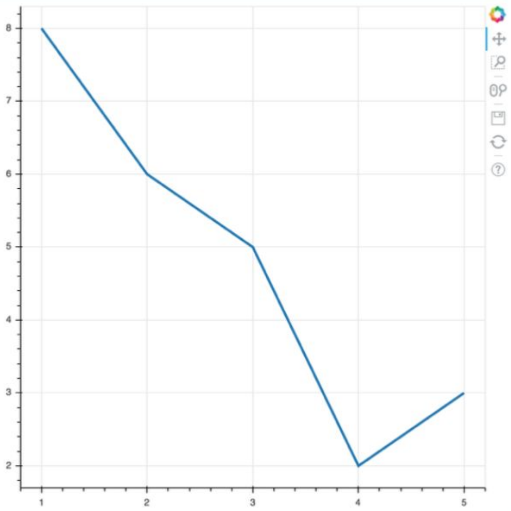
Markers

- asterisk()
- circle()
- circle_cross()
- circle_x()
- cross()
- diamond()
- diamond_cross()
- inverted_triangl()
- square()
- square_cross()
- square_x()
- triangle()
- x()

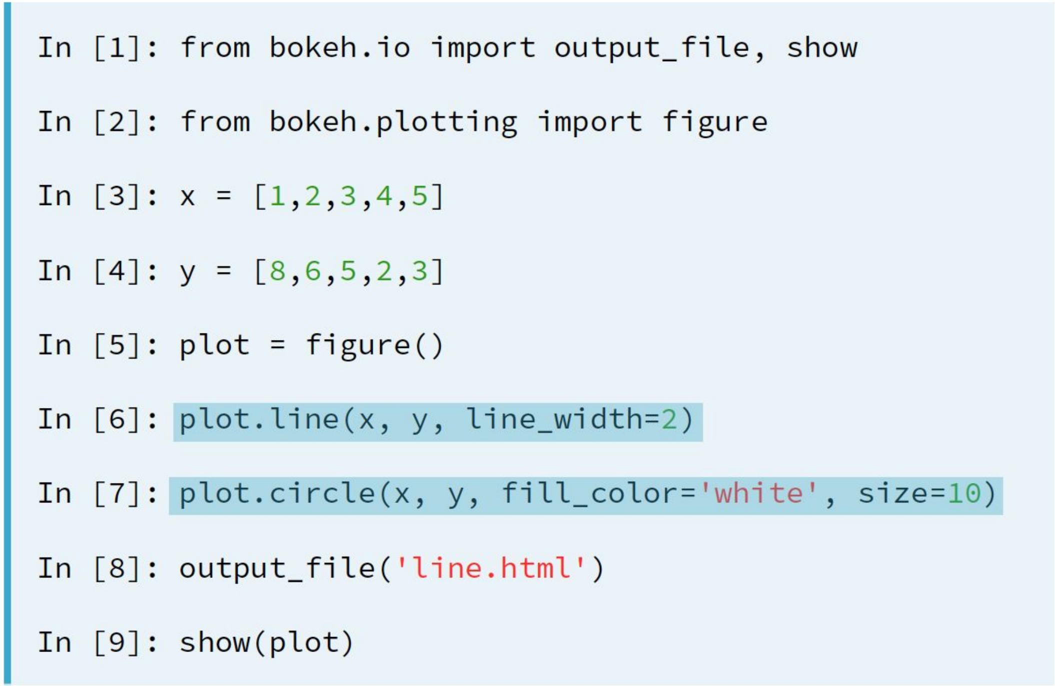
Lines



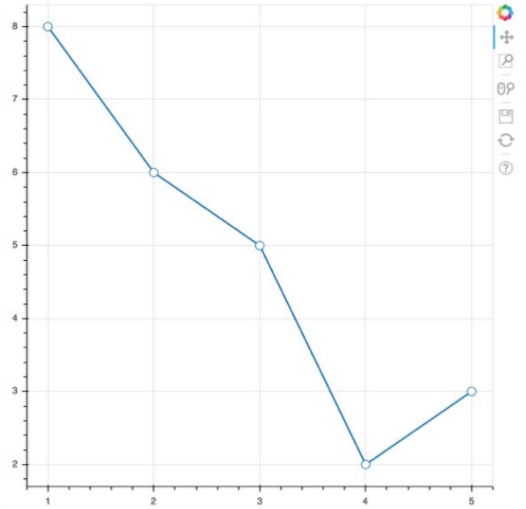
```
In [1]: from bokeh.io import output_file, show

In [2]: from bokeh.plotting import figure

In [3]: x = [1,2,3,4,5]

In [4]: y = [8,6,5,2,3]

In [5]: plot = figure()

In [6]: plot.line(x, y, line_width=3)

In [7]: output_file('line.html')

In [8]: show(plot)
```



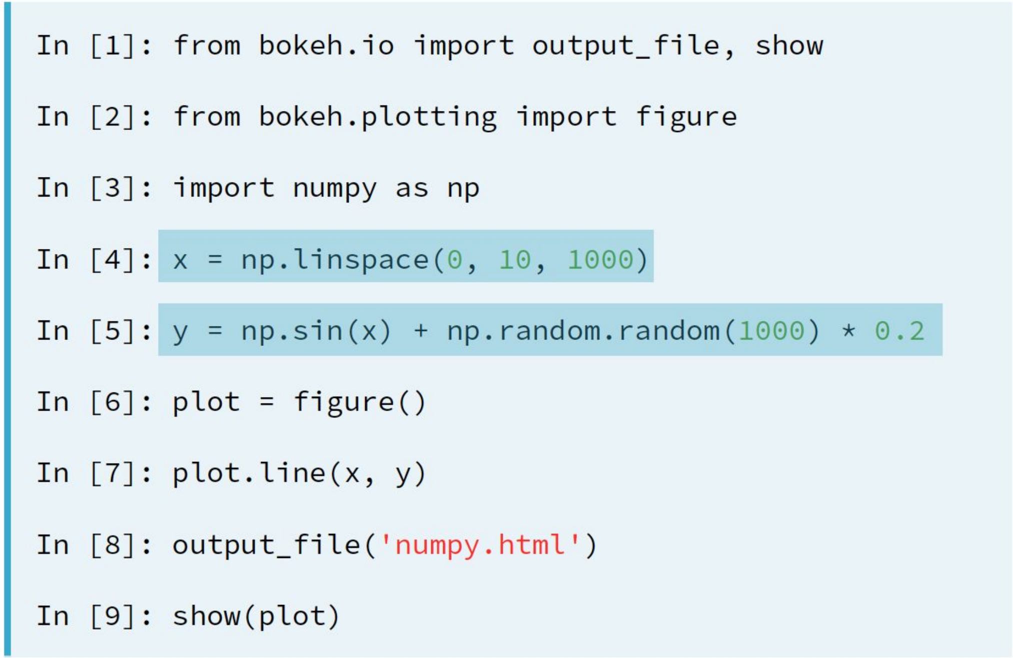
Lines and markers together



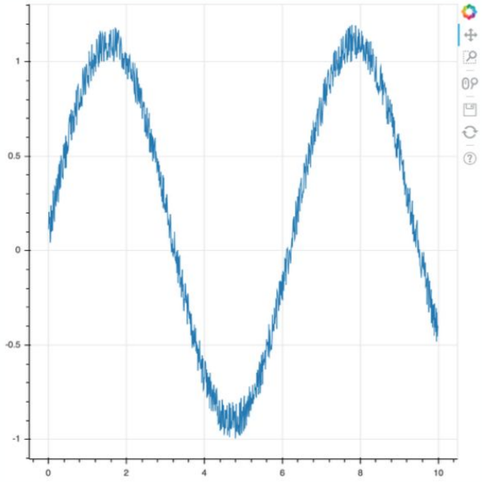
```
In [1]: from bokeh.io import output_file, show

In [2]: from bokeh.plotting import figure

In [3]: x = [1,2,3,4,5]

In [4]: y = [8,6,5,2,3]

In [5]: plot = figure()

In [6]: plot.line(x, y, line_width=2)

In [7]: plot.circle(x, y, fill_color='white', size=10)

In [8]: output_file('line.html')

In [9]: show(plot)
```



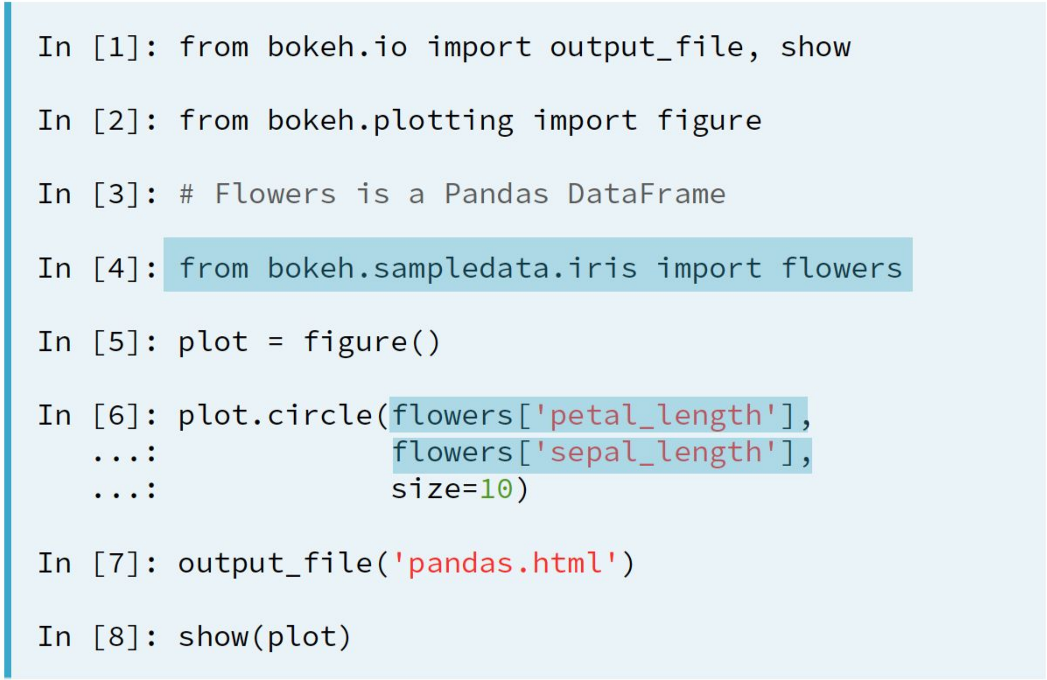
Numpy arrays



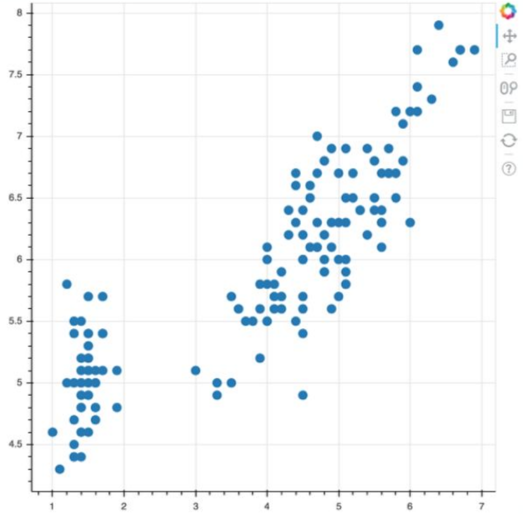
```
In [1]: from bokeh.io import output_file, show

In [2]: from bokeh.plotting import figure

In [3]: import numpy as np

In [4]: x = np.linspace(0, 10, 1000)

In [5]: y = np.sin(x) + np.random.random(1000) * 0.2

In [6]: plot = figure()

In [7]: plot.line(x, y)

In [8]: output_file('numpy.html')

In [9]: show(plot)
```



Pandas



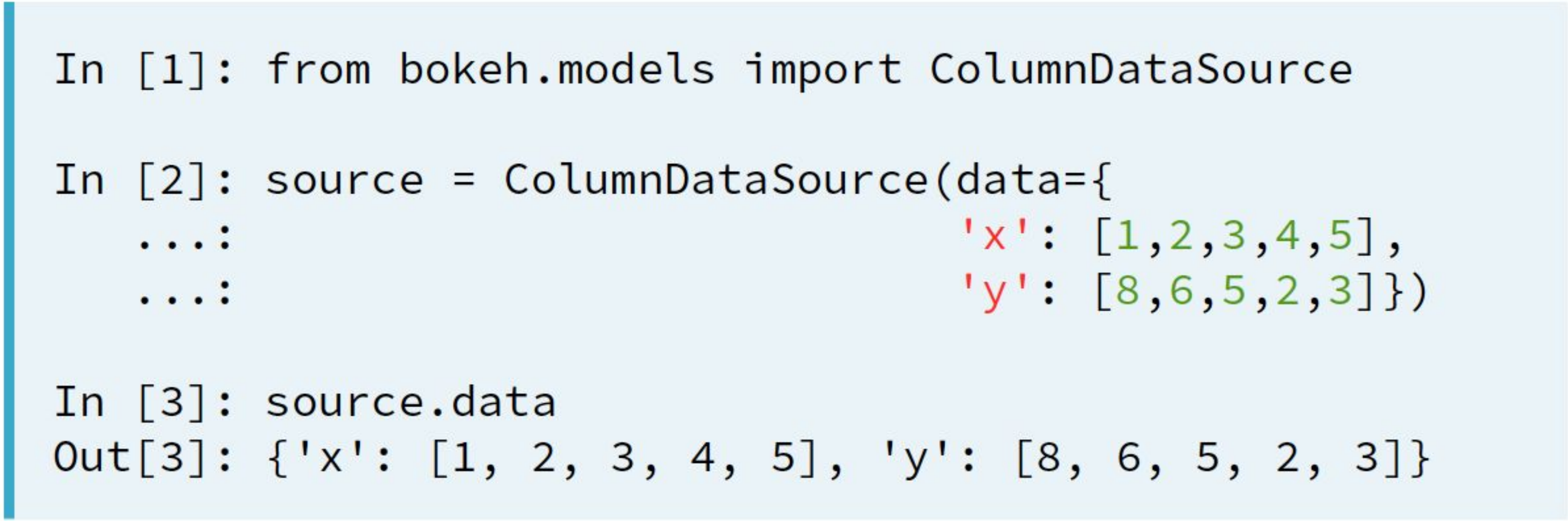
```
In [1]: from bokeh.io import output_file, show

In [2]: from bokeh.plotting import figure

In [3]: # Flowers is a Pandas DataFrame

In [4]: from bokeh.sampledata.iris import flowers

In [5]: plot = figure()

In [6]: plot.circle(flowers['petal_length'],
   ...:             flowers['sepal_length'],
   ...:             size=10)

In [7]: output_file('pandas.html')

In [8]: show(plot)
```



Column data source

- Common fundamental data structure for Bokeh
- Maps string column names to sequences of data
- Often created automatically for you
- Can be shared between glyphs to link selections
- Extra columns can be used with hover tooltips

Column data source



```
In [1]: from bokeh.models import ColumnDataSource

In [2]: source = ColumnDataSource(data={
   ...:                           'x': [1,2,3,4,5],
   ...:                           'y': [8,6,5,2,3]})

In [3]: source.data
Out[3]: {'x': [1, 2, 3, 4, 5], 'y': [8, 6, 5, 2, 3]}
```

References

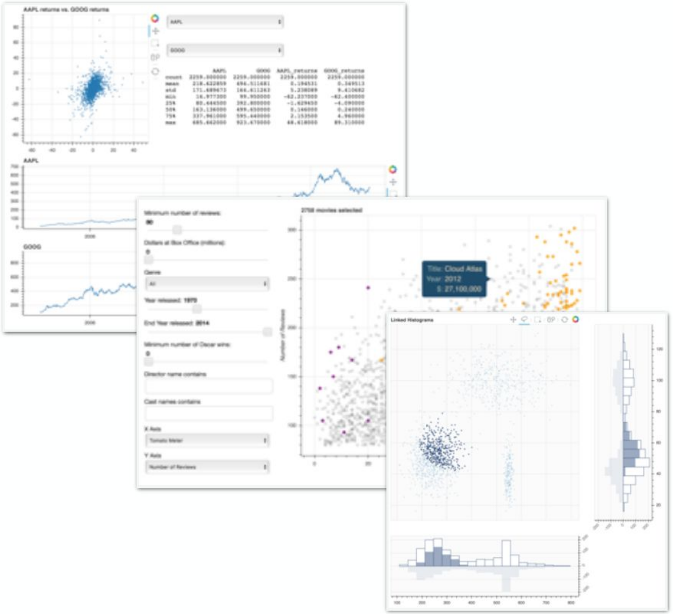
Notebook: "Interactive Data Visualization with Bokeh Part I.ipynb"
Dataset #1: "fertility.csv"
Dataset #2: "olympics.csv"

Introduction
to
Layout

Arranging multiple plots

Arrange plots (and controls) visually on a page:

- rows, columns
- grid arrangements
- tabbed layouts



AAPL

GOOG

```
              AAPL          GOOG  AAPL_returns  GOOG_returns
count  2259.000000   2259.000000   2259.000000   2259.000000
mean    218.622859    496.511681      0.196531      0.369513
std     171.689673    166.611263      5.238089      9.410682
min      16.977300     99.950000    -42.237000    -62.400000
25%      80.444500    392.800000     -1.629450     -4.090000
50%     163.136000    499.650000      0.146000      0.240000
75%     337.961000    595.440000      2.153500      4.960000
max     685.662000    923.470000     48.618000     89.310000
```

Minimum number of reviews: 80

Dollars at Box Office (millions): 0

Genre: All

Year released: 1970

End Year released: 2014

Minimum number of Oscar wins: 0

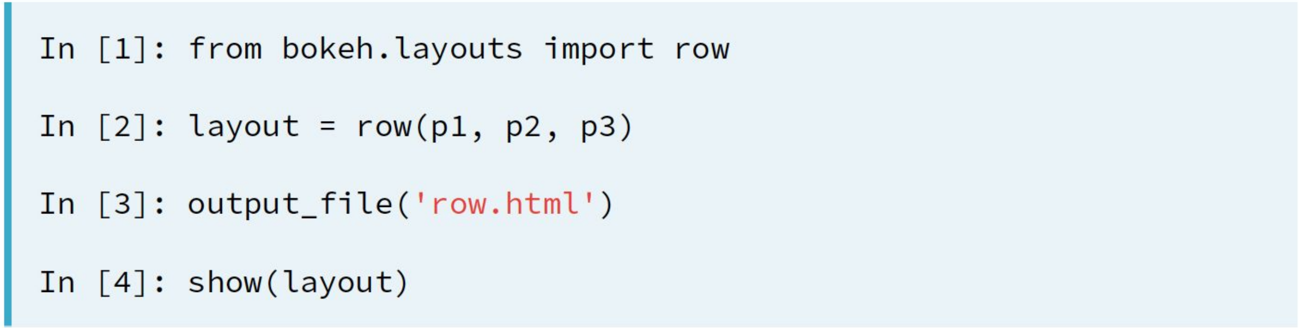
Director name contains

Cast names contains

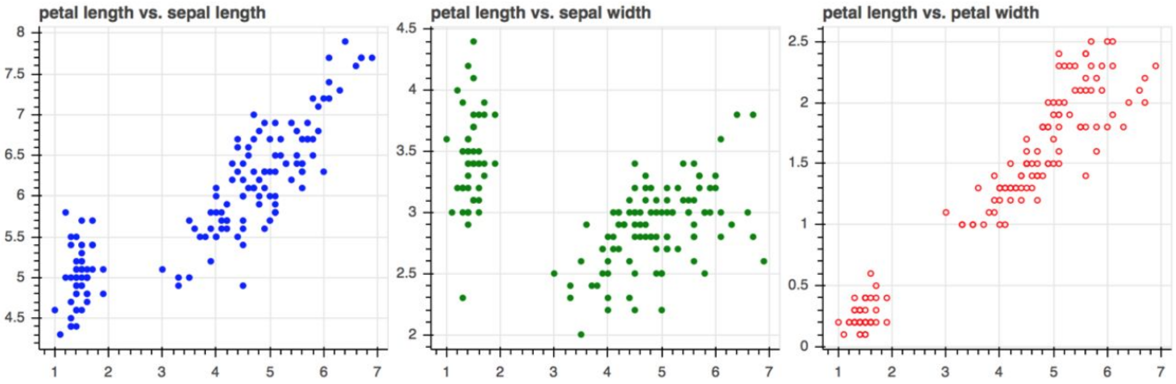
X Axis: Tomato Meter

Y Axis: Number of Reviews

Rows of plot



```
In [1]: from bokeh.layouts import row

In [2]: layout = row(p1, p2, p3)

In [3]: output_file('row.html')

In [4]: show(layout)
```

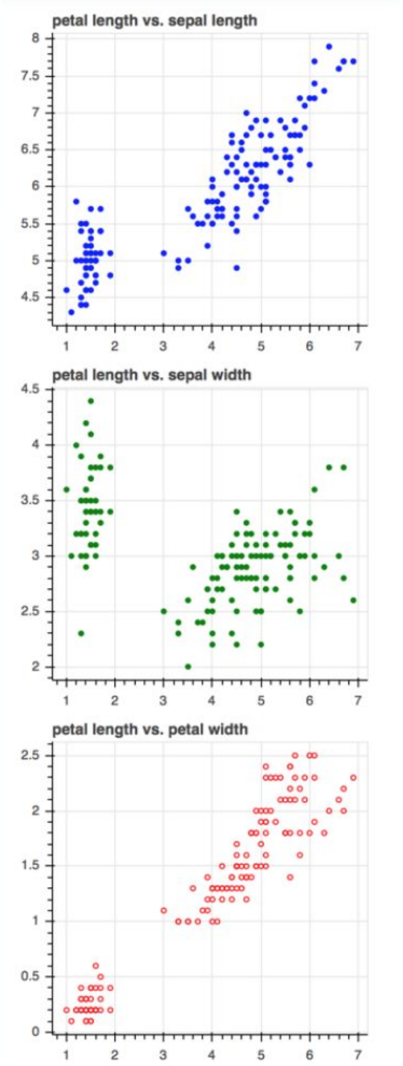


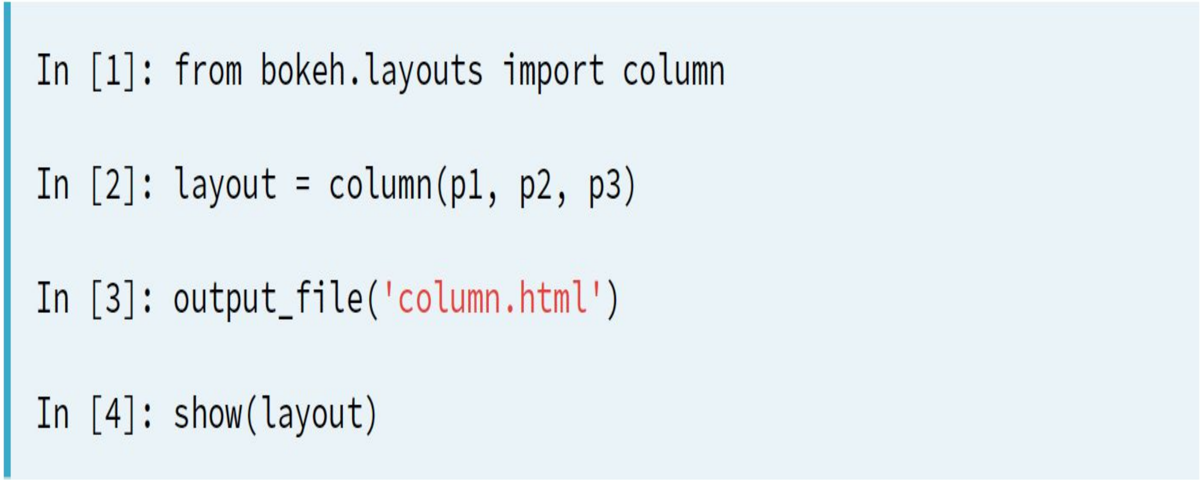
petal length vs. sepal length

petal length vs. sepal width

petal length vs. petal width

Columns of plot

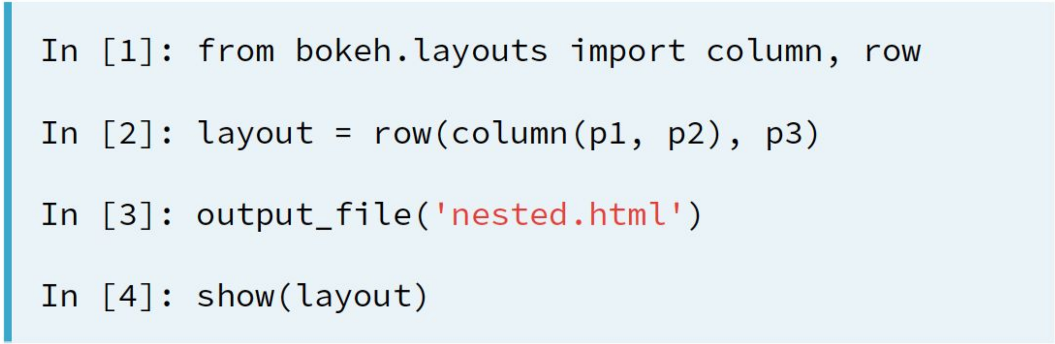




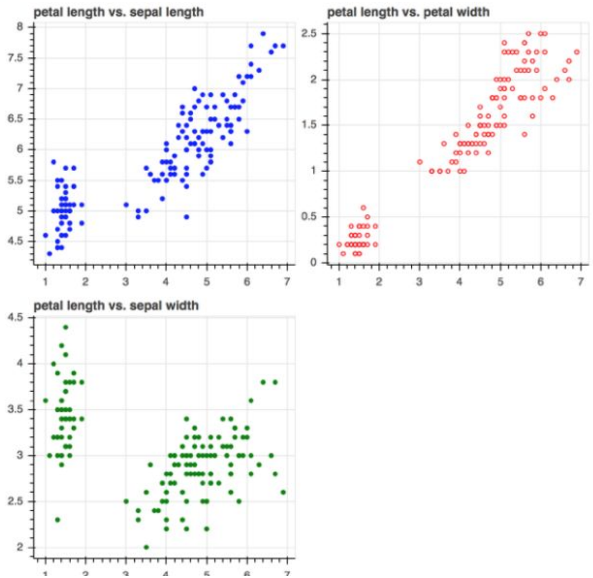
```
In [1]: from bokeh.layouts import column

In [2]: layout = column(p1, p2, p3)

In [3]: output_file('column.html')

In [4]: show(layout)
```

Nested plots

- Rows and column can be nested for more sophisticated layouts



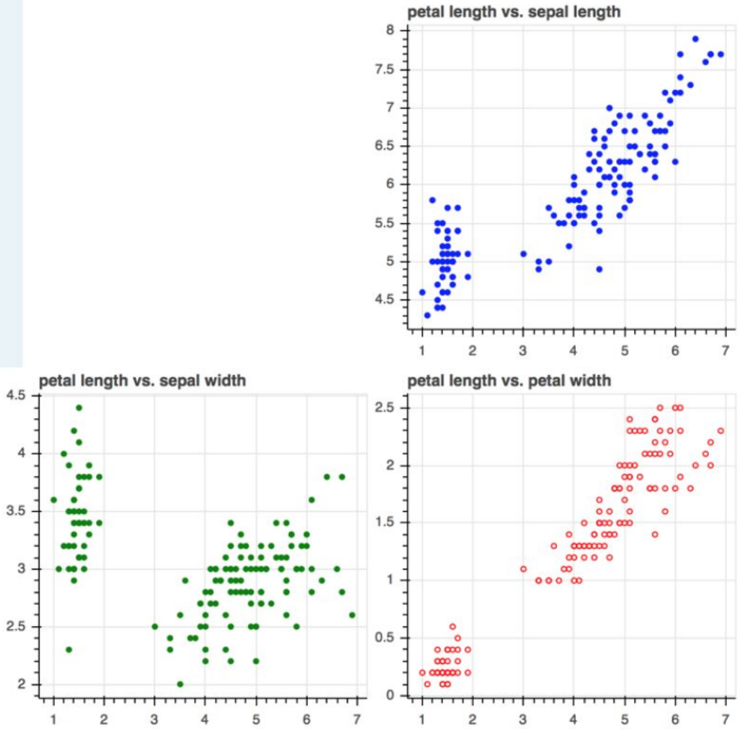
```
In [1]: from bokeh.layouts import column, row

In [2]: layout = row(column(p1, p2), p3)

In [3]: output_file('nested.html')

In [4]: show(layout)
```



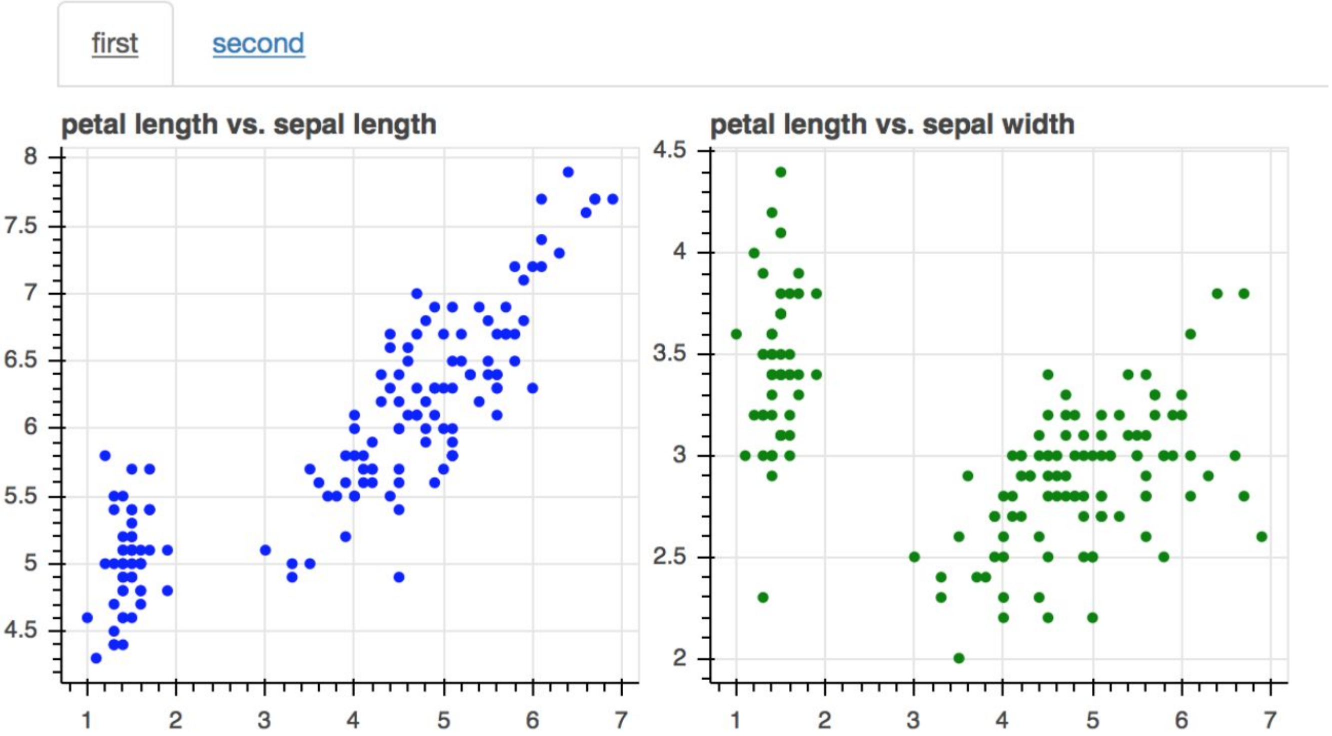
Gridplots

```
In [1]: from bokeh.layouts import gridplot

In [2]: layout = gridplot([[None, p1], [p2, p3]],
   ...:                  toolbar_location=None)

In [3]: output_file('nested.html')

In [4]: show(layout)
```

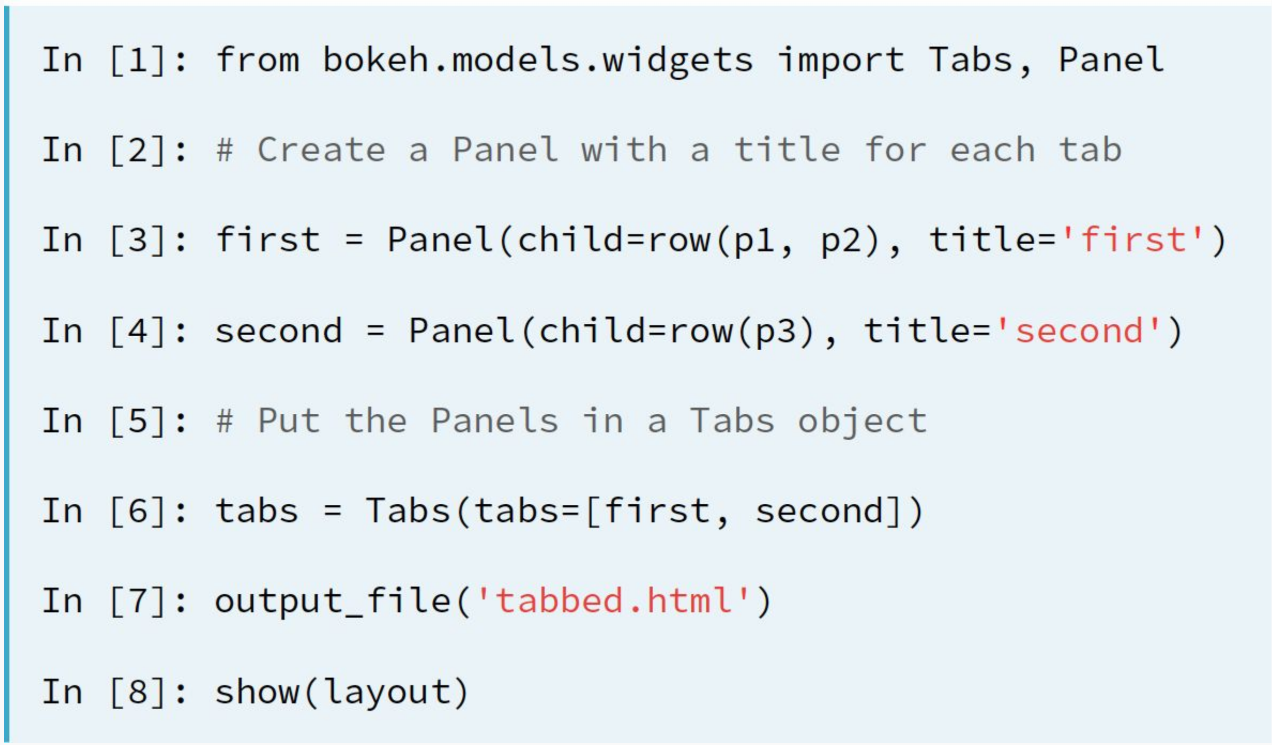
- Give a “list of rows” for layout
- can use None as a placeholder
- Accepts toolbar_location



Tabbed Layout



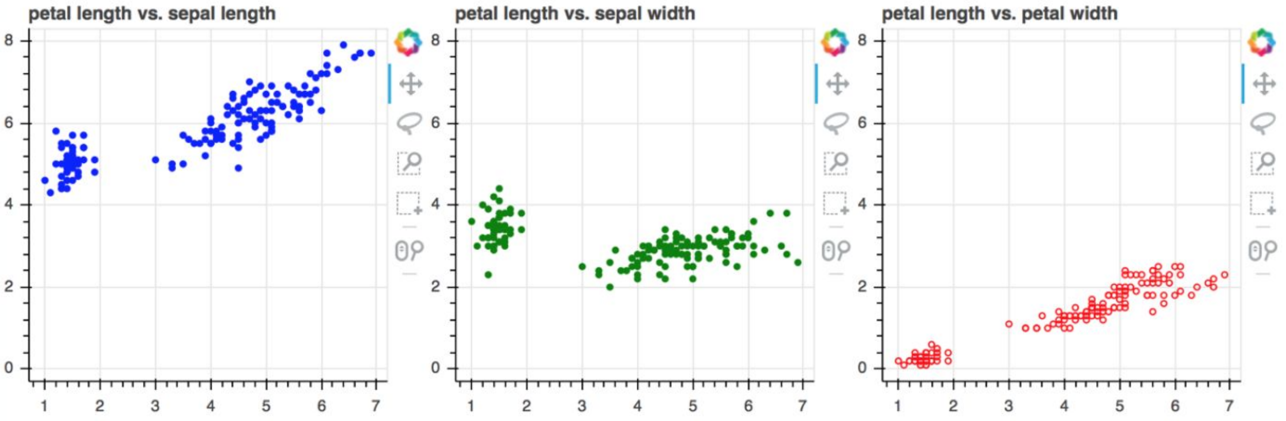
Tabbed Layout



```
In [1]: from bokeh.models.widgets import Tabs, Panel

In [2]: # Create a Panel with a title for each tab

In [3]: first = Panel(child=row(p1, p2), title='first')

In [4]: second = Panel(child=row(p3), title='second')

In [5]: # Put the Panels in a Tabs object

In [6]: tabs = Tabs(tabs=[first, second])

In [7]: output_file('tabbed.html')

In [8]: show(layout)
```

Linking axes

```
In [1]: p3.x_range = p2.x_range = p1.x_range

In [2]: p3.y_range = p2.y_range = p1.y_range
```

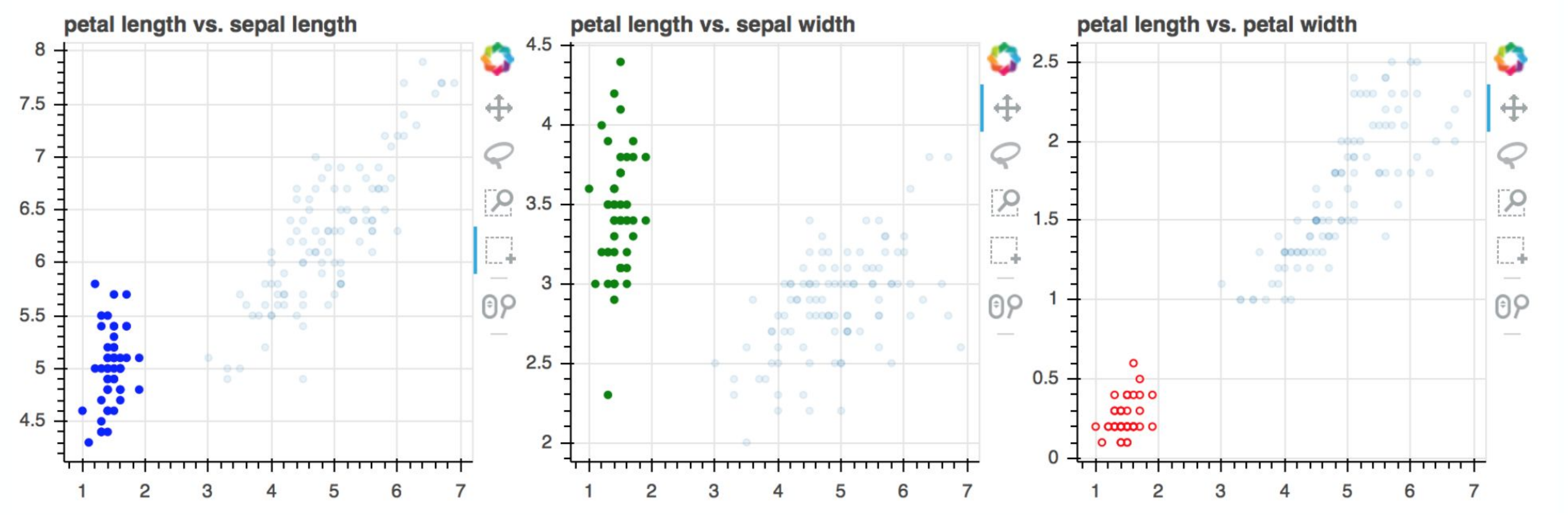


petal length vs. sepal length

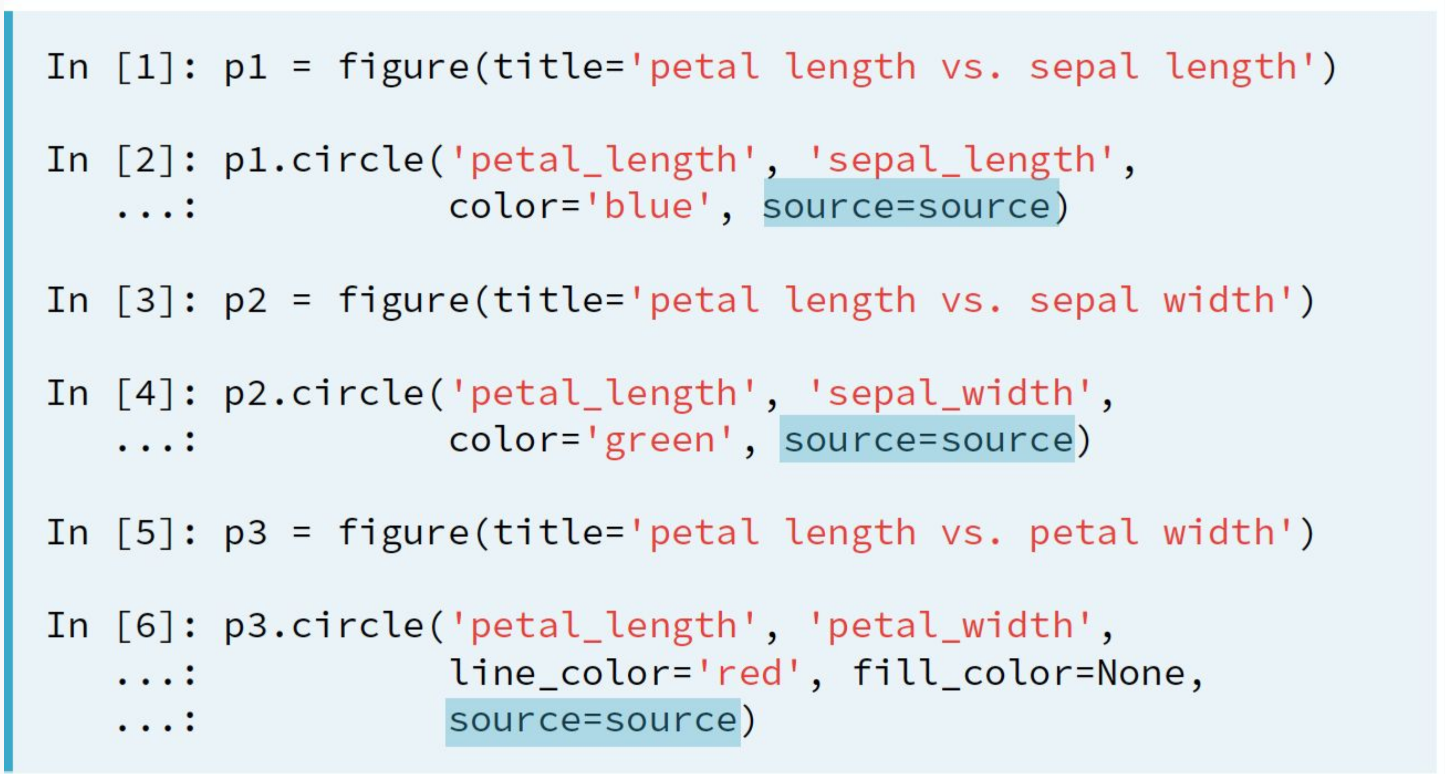
petal length vs. sepal width

petal length vs. petal width

Link selections

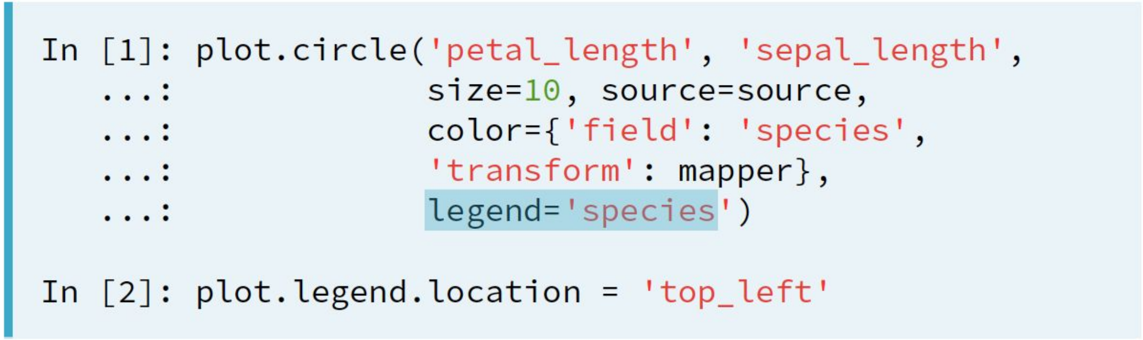


Link selections

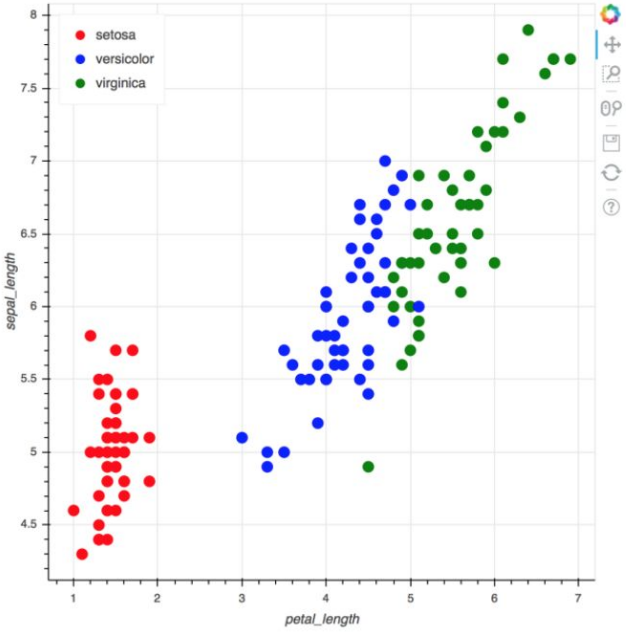


```
In [1]: p1 = figure(title='petal length vs. sepal length')

In [2]: p1.circle('petal_length', 'sepal_length',
   ...:           color='blue', source=source)

In [3]: p2 = figure(title='petal length vs. sepal width')

In [4]: p2.circle('petal_length', 'sepal_width',
   ...:           color='green', source=source)

In [5]: p3 = figure(title='petal length vs. petal width')

In [6]: p3.circle('petal_length', 'petal_width',
   ...:           line_color='red', fill_color=None,
   ...:           source=source)
```

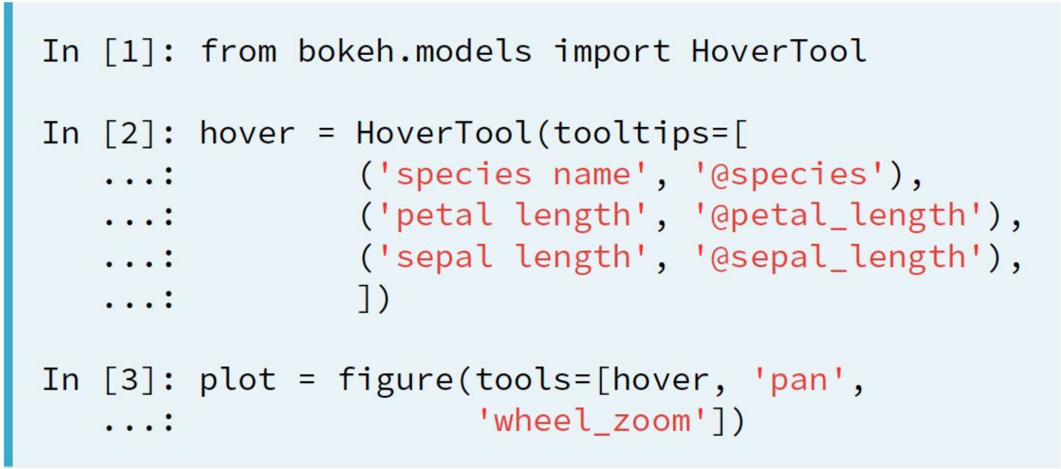
Legends



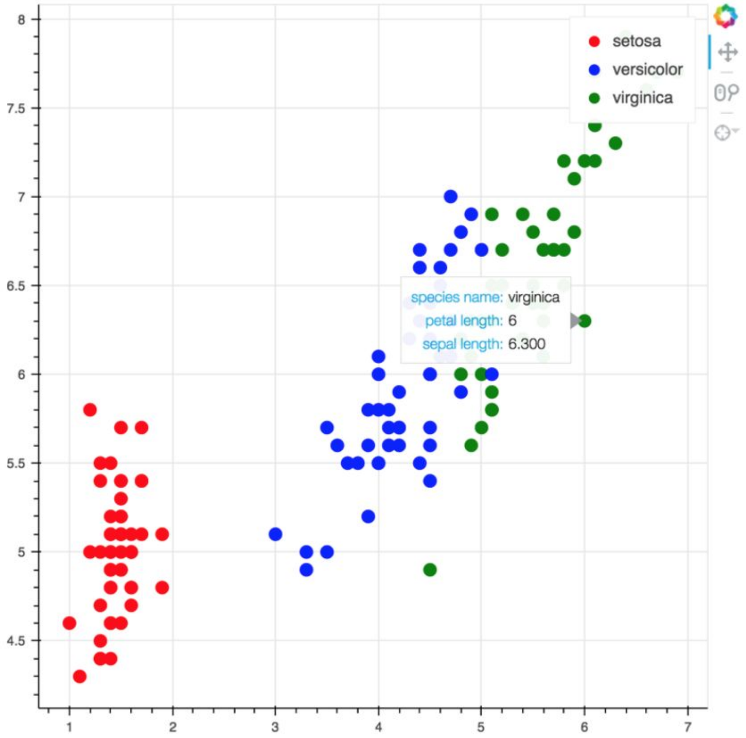
```
In [1]: plot.circle('petal_length', 'sepal_length',
   ...:             size=10, source=source,
   ...:             color={'field': 'species',
   ...:             'transform': mapper},
   ...:             legend='species')

In [2]: plot.legend.location = 'top_left'
```



Hover tips



```
In [1]: from bokeh.models import HoverTool

In [2]: hover = HoverTool(tooltips=[
   ...:             ('species name', '@species'),
   ...:             ('petal length', '@petal_length'),
   ...:             ('sepal length', '@sepal_length'),
   ...:             ])

In [3]: plot = figure(tools=[hover, 'pan',
   ...:                     'wheel_zoom'])
```



References

- Notebooks:
  - "Interactive Data Visualization with Bokeh - Part II.ipynb"
- Datasets:
  - "fertility.csv"
  - "olympics.csv"