Plotting with glyphs

INTERACTIVE DATA VISUALIZATION WITH BOKEH

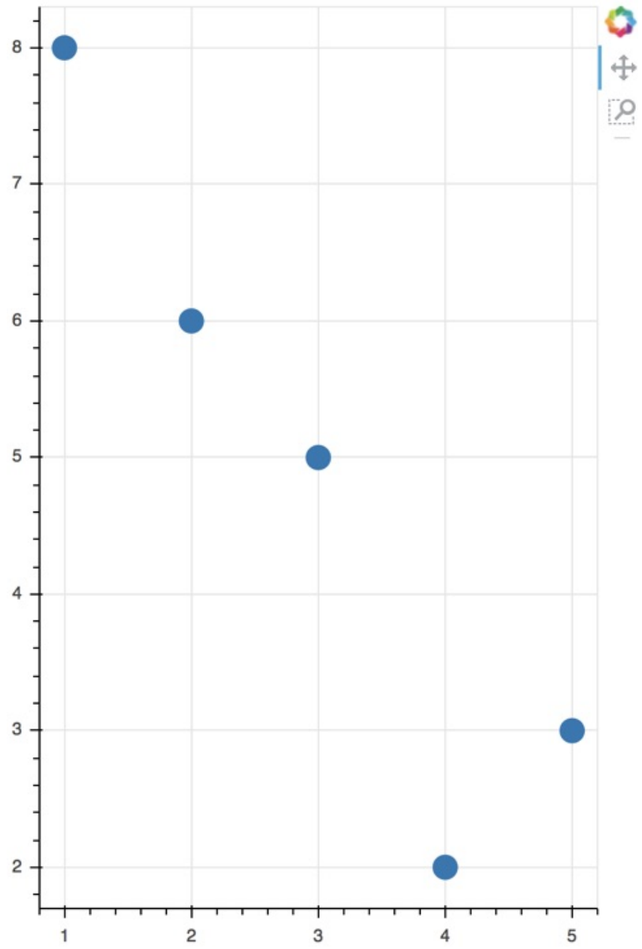
Bryan Van de Ven
Core Developer of Bokeh

What are Glyphs

- **Visual shapes**
  - circles, squares, triangles
  - rectangles, lines, wedges
- **With properties attached to data**
  - coordinates (x, y)
  - size, color, transparency

Typical usage

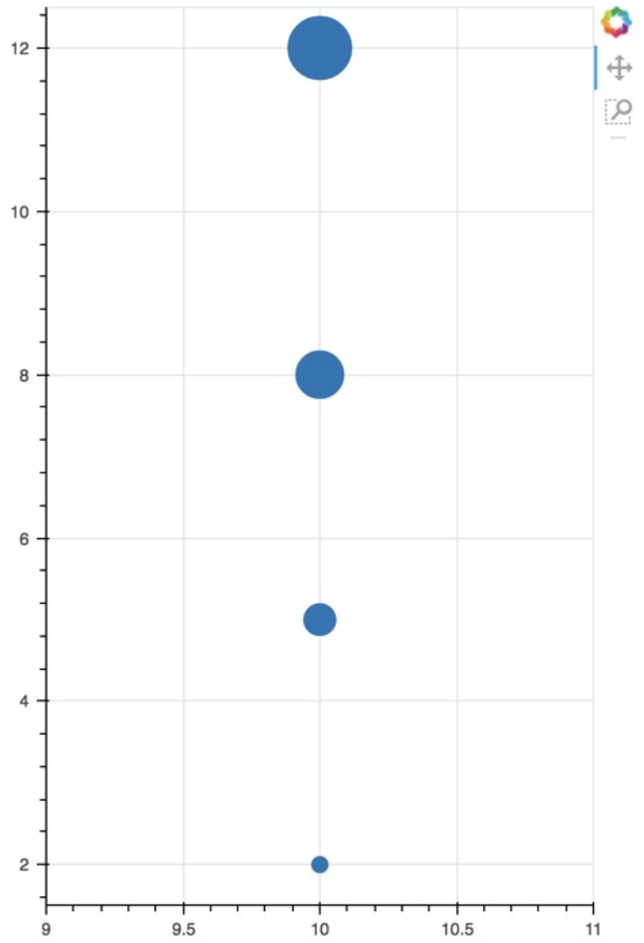
```
from bokeh.io import output_file, show
from bokeh.plotting import figure

plot = figure(plot_width=400, tools='pan,box_zoo

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

output_file('circle.html')
show(plot)
```



Glyph properties

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

```
plot = figure()
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_triangle()`
- `square()`
- `square_cross()`
- `square_x()`
- `triangle()`
- `x()`

Let's practice!

INTERACTIVE DATA VISUALIZATION WITH BOKEH

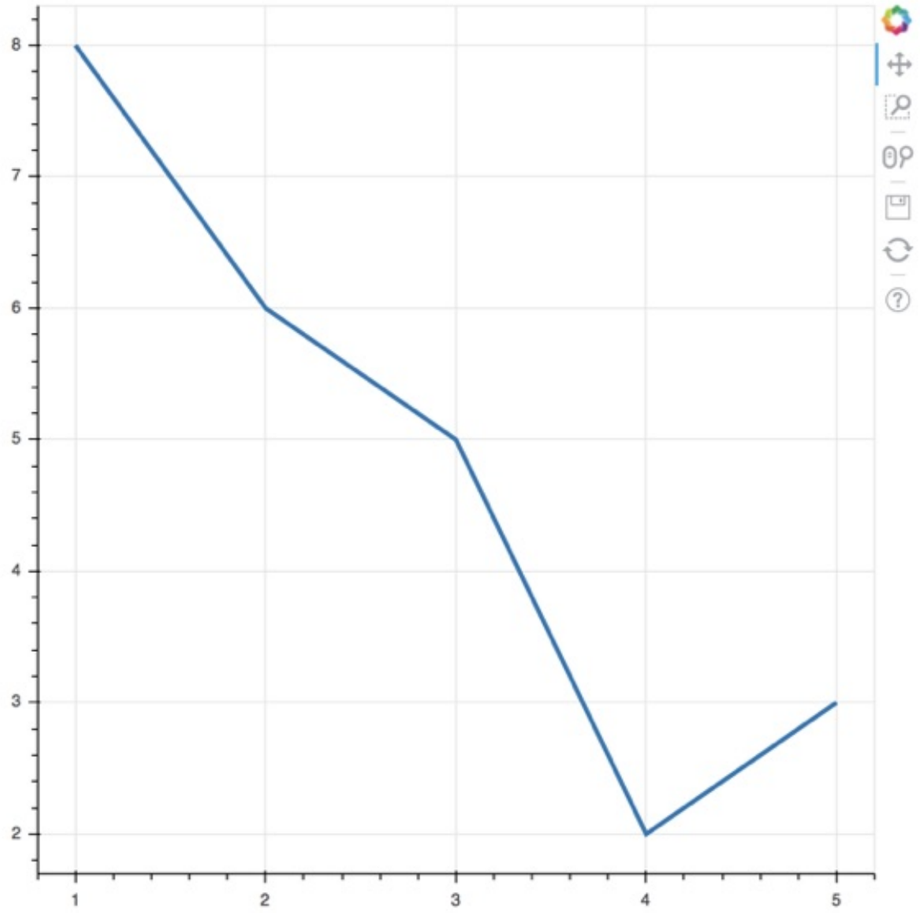
Additional glyphs

INTERACTIVE DATA VISUALIZATION WITH BOKEH

Bryan Van de Ven
Core Developer of Bokeh

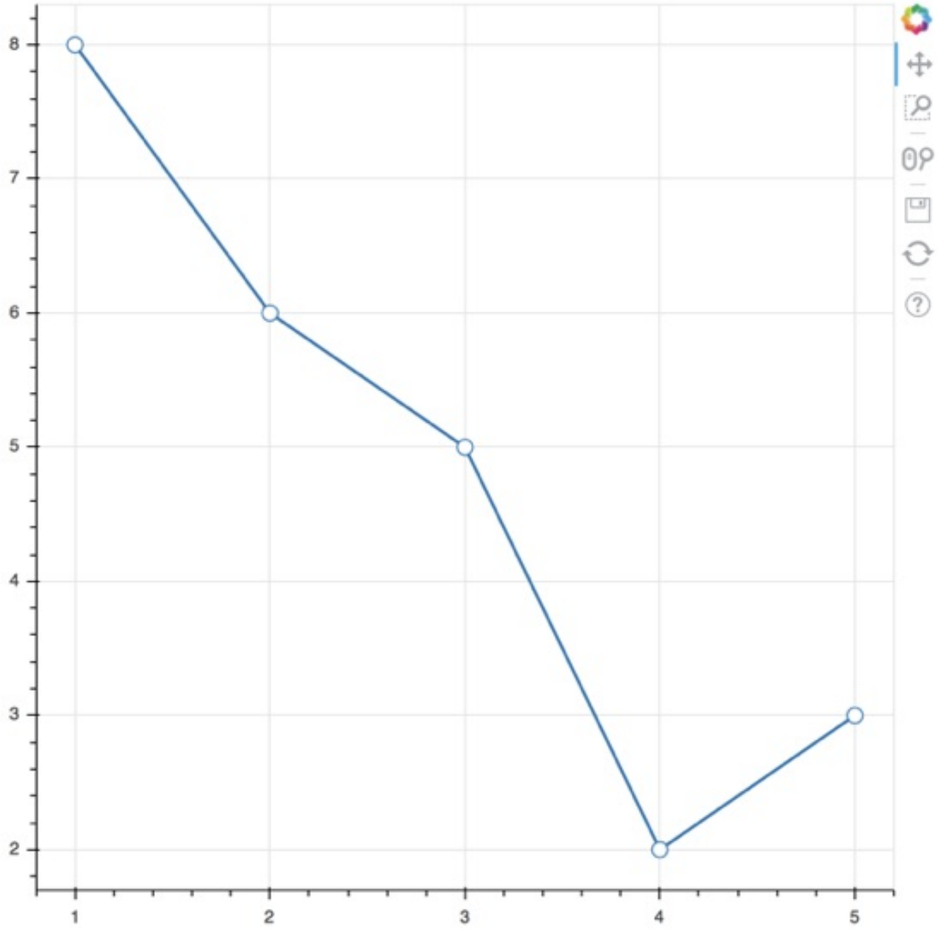
Lines

```
from bokeh.io import output_file, show
from bokeh.plotting import figure

x = [1,2,3,4,5]
y = [8,6,4,2,3]

plot = figure()

plot.line(x, y, line_width=3)

output_file('line.html')
show(plot)
```



Lines and Markers together

```
from bokeh.io import output_file, show
from bokeh.plotting import figure

x = [1,2,3,4,5]
y = [8,6,4,2,3]

plot = figure()

plot.line(x, y, line_width=2)
plot.circle(x, y, fill_color='white', size=10)

output_file('line.html')
show(plot)
```

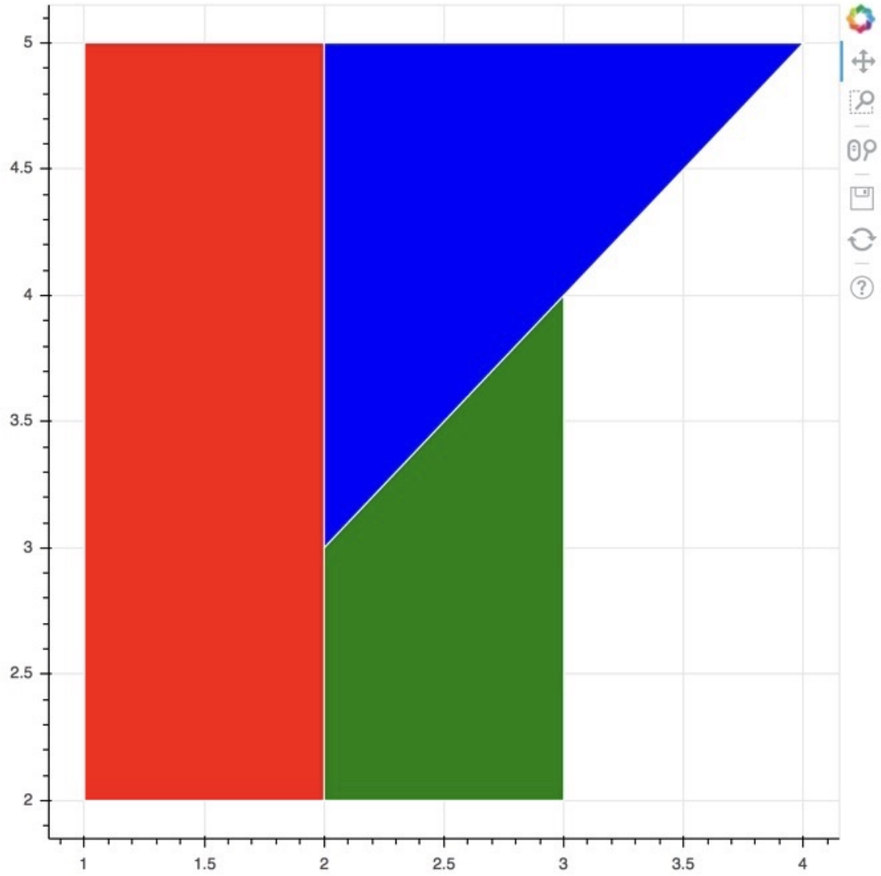


Patches

- Useful for showing geographic regions
- Data given as "list of lists"

Patches

```
from bokeh.io import output_file, show
from bokeh.plotting import figure

xs = [[1,1,2,2], [2,2,4], [2,2,3,3]]
ys = [[2,5,5,2], [3,5,5], [2,3,4,2]]

plot = figure()

plot.patches(xs, ys,
    fill_color=['red', 'blue', 'green'],
    line_color='white')

output_file('patches.html')
show(plot)
```



- `annulus()`
- `annular_wedge()`
- `wedge()`
- `rect()`
- `quad()`
- `vbar()`
- `hbar()`
- `image()`
- `image_rgba()`
- `image_url()`
- `patch()`
- `patches()`
- `line()`
- `multi_line()`
- `circle()`
- `oval()`
- `ellipse()`
- `arc()`
- `quadratic()`
- `bezier()`

Let's practice!

INTERACTIVE DATA VISUALIZATION WITH BOKEH

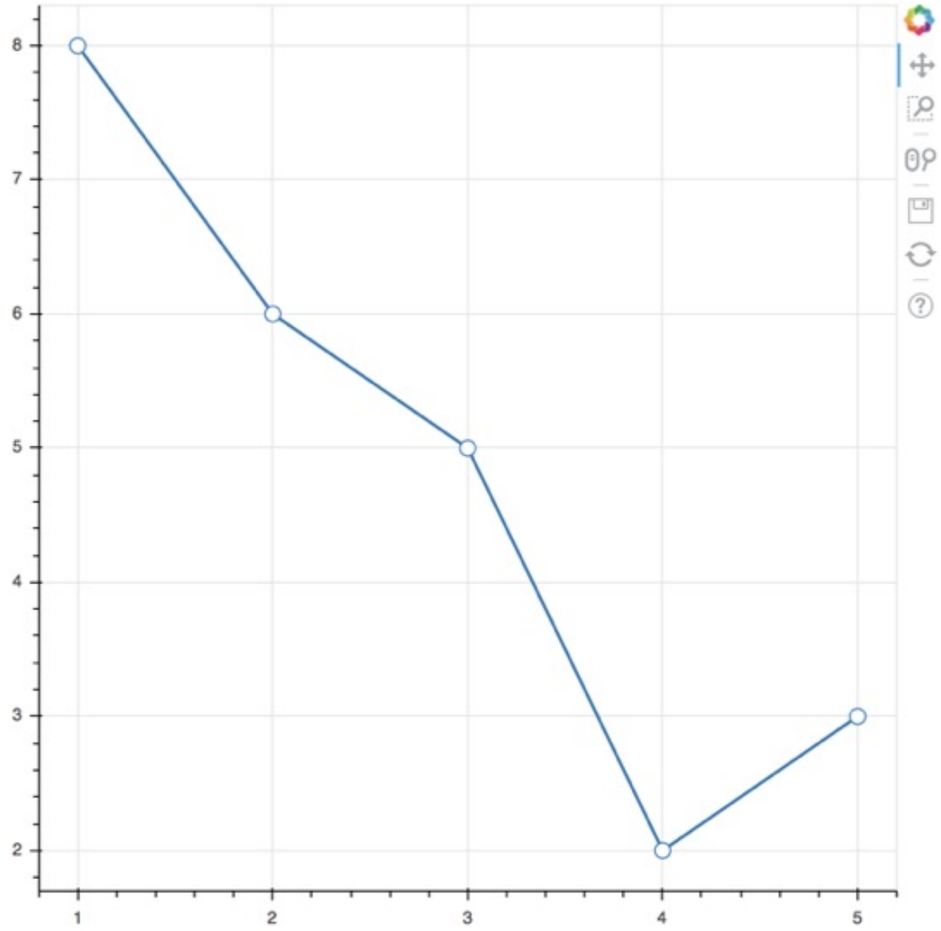
Data formats

INTERACTIVE DATA VISUALIZATION WITH BOKEH

Bryan Van de Ven
Core Developer of Bokeh

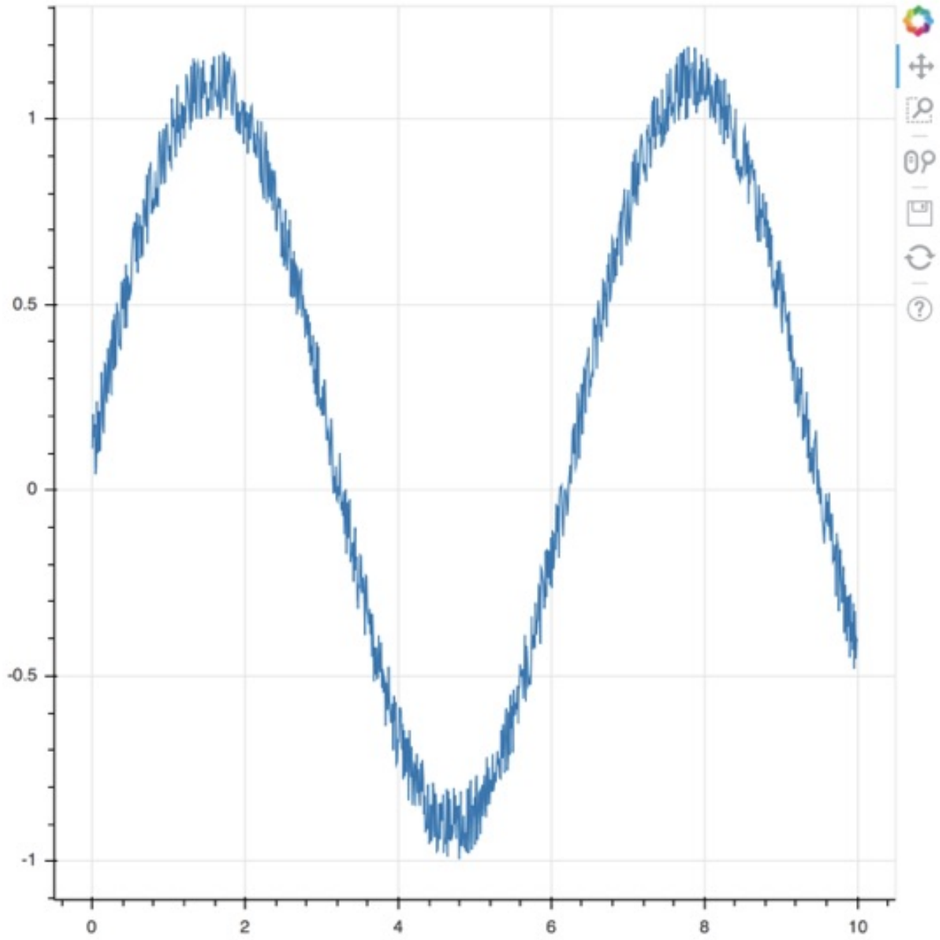
Python Basic Types

```
from bokeh.io import output_file, show
from bokeh.plotting import figure

x = [1,2,3,4,5]
y = [8,6,5,2,3]

plot = figure()

plot.line(x, y, line_width=3)
plot.circle(x, y, fill_color='white', size=10)

output_file('basic.html')
show(plot)
```



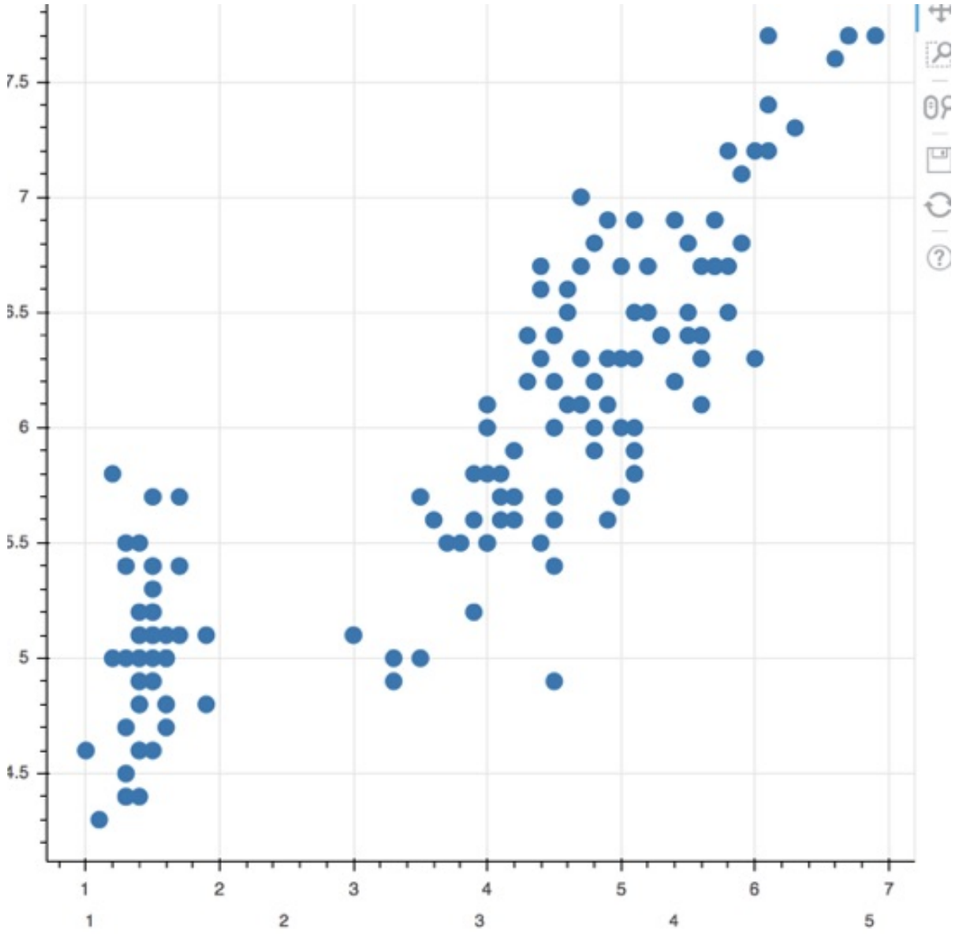
NumPy Arrays

```
from bokeh.io import output_file, show
from bokeh.plotting import figure
import numpy as np

x = np.linspace(0, 10, 1000)
y = np.sin(x) + np.random.random(1000) * 0.2

plot = figure()

plot.line(x, y)

output_file('numpy.html')
show(plot)
```



Pandas

```
from bokeh.io import output_file, show
from bokeh.plotting import figure

# Flowers is a Pandas DataFrame
from bokeh.sampledata.iris import flowers

plot = figure()

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

output_file('pandas.html')
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 glpyhs to link selections
- Extra columns can be used with hover tooltips

Column Data Source

```
from bokeh.models import ColumnDataSource


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

source.data
```

```
{'x': [1, 2, 3, 4, 5], 'y': [8, 6, 5, 2, 3]}
```

Column Data Source

```
from bokeh.models import ColumnDataSource
from bokeh.sampledata.iris import flowers as df

df.head()
```

```
    sepal_length    sepal_width    petal_length    petal_width    species
0        5.1             3.5            1.4             0.2        setosa
1        4.9             3.0            1.4             0.2        setosa
2        4.7             3.2            1.3             0.2        setosa
3        4.6             3.1            1.5             0.2        setosa
4        5.0             3.6            1.4             0.2        setosa
```

```
source = ColumnDataSource(df)
```

Let's practice!

INTERACTIVE DATA VISUALIZATION WITH BOKEH

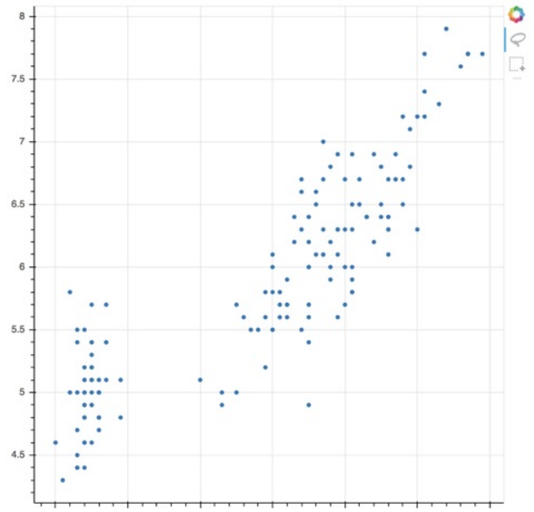
Customizing glyphs

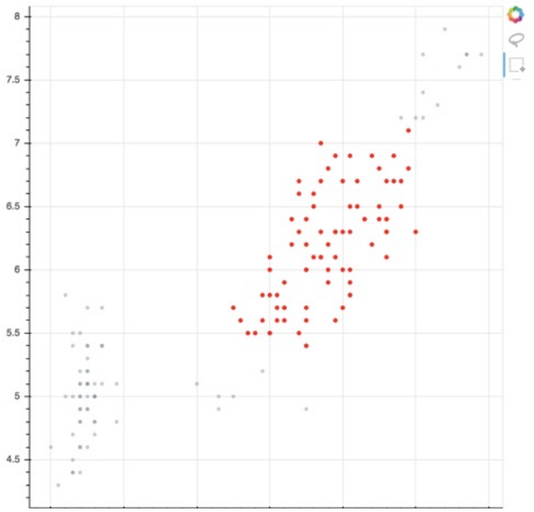
INTERACTIVE DATA VISUALIZATION WITH BOKEH

Bryan Van de Ven
Core Developer of Bokeh

Selection appearance

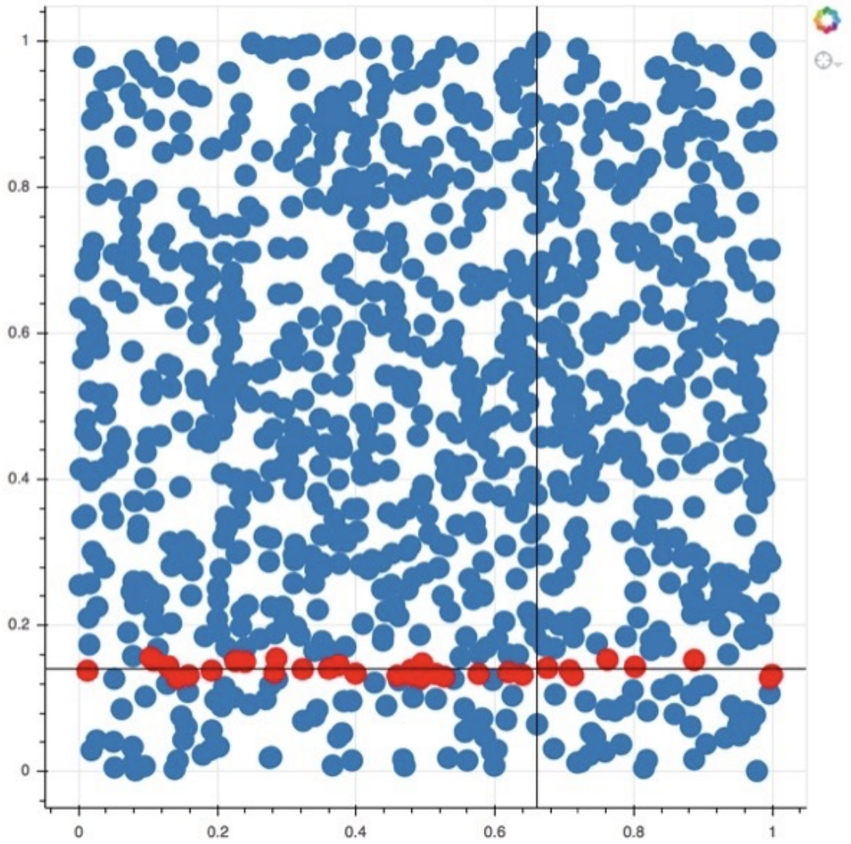
```
plot = figure(tools='box_select, lasso_select')
plot.circle(petal_length, sepal_length,
        selection_color='red',
        nonselection_fill_alpha=0.2,
        nonselection_fill_color='grey')
```





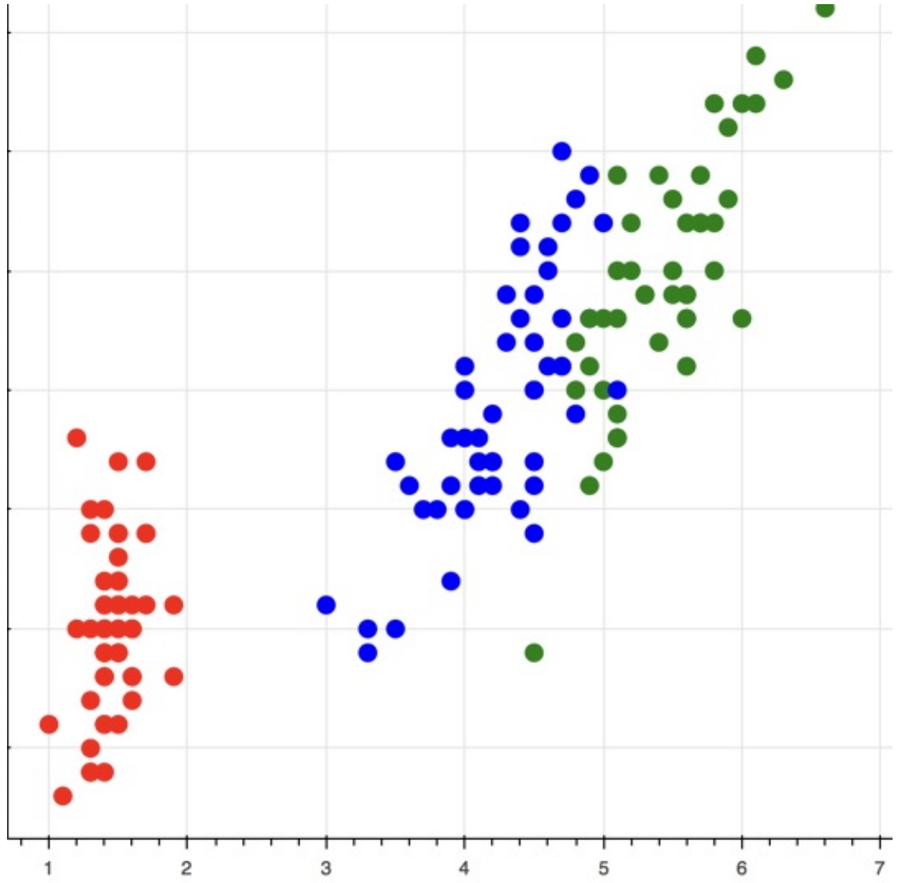
Hover appearance

```
from bokeh.models import HoverTool

hover = HoverTool(tooltips=None, mode='hline')

plot = figure(tools=[hover, 'crosshair'])

# x and y are lists of random points
plot.circle(x, y, size=15, hover_color='red')
```



Color mapping

```
from bokeh.models import CategoricalColorMapper

mapper = CategoricalColorMapper(
        factors=['setosa', 'virginica', 'versicolor'],
        palette=['red', 'green', 'blue'])

plot = figure(x_axis_label='petal_length'
        y_axis_label='sepal_length')

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



Let's practice!

INTERACTIVE DATA VISUALIZATION WITH BOKEH