

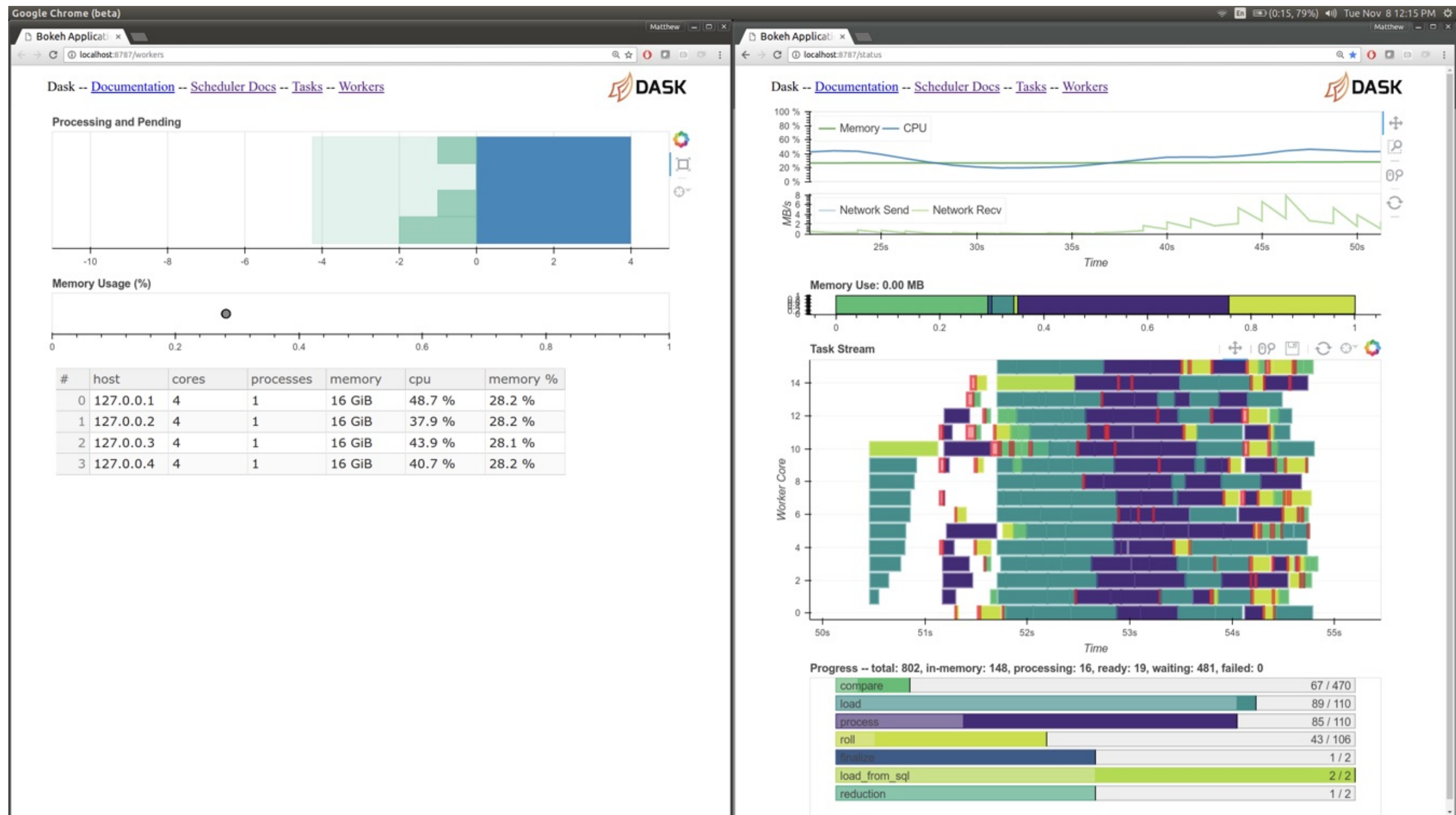
Introducing the Bokeh Server

INTERACTIVE DATA VISUALIZATION WITH BOKEH



Bryan Van de Ven
Core Developer of Bokeh





Basic app outline

outline.py

```
from bokeh.io import curdoc

# Create plots and widgets

# Add callbacks

# Arrange plots and widgets in layouts

curdoc().add_root(layout)
```

Running Bokeh applications

Running single module apps at the shell or Windows command prompt:

```
bokeh serve --show myapp.py
```

"Directory" style apps run similarly:

```
bokeh serve --show myappdir/
```

Let's practice!

INTERACTIVE DATA VISUALIZATION WITH BOKEH

Connecting sliders to plots

INTERACTIVE DATA VISUALIZATION WITH BOKEH



Bryan Van de Ven
Core Developer of Bokeh

A slider example

slider.py

```
from bokeh.io import curdoc
from bokeh.layouts import column
from bokeh.models import ColumnDataSource, Slider
from bokeh.plotting import figure
from numpy.random import random

N = 300
source = ColumnDataSource(data={'x': random(N), 'y': random(N)})

# Create plots and widgets
plot = figure()
plot.circle(x='x', y='y', source=source)

slider = Slider(start=100, end=1000, value=N,
                step=10, title='Number of points')
```


A slider example

slider.py

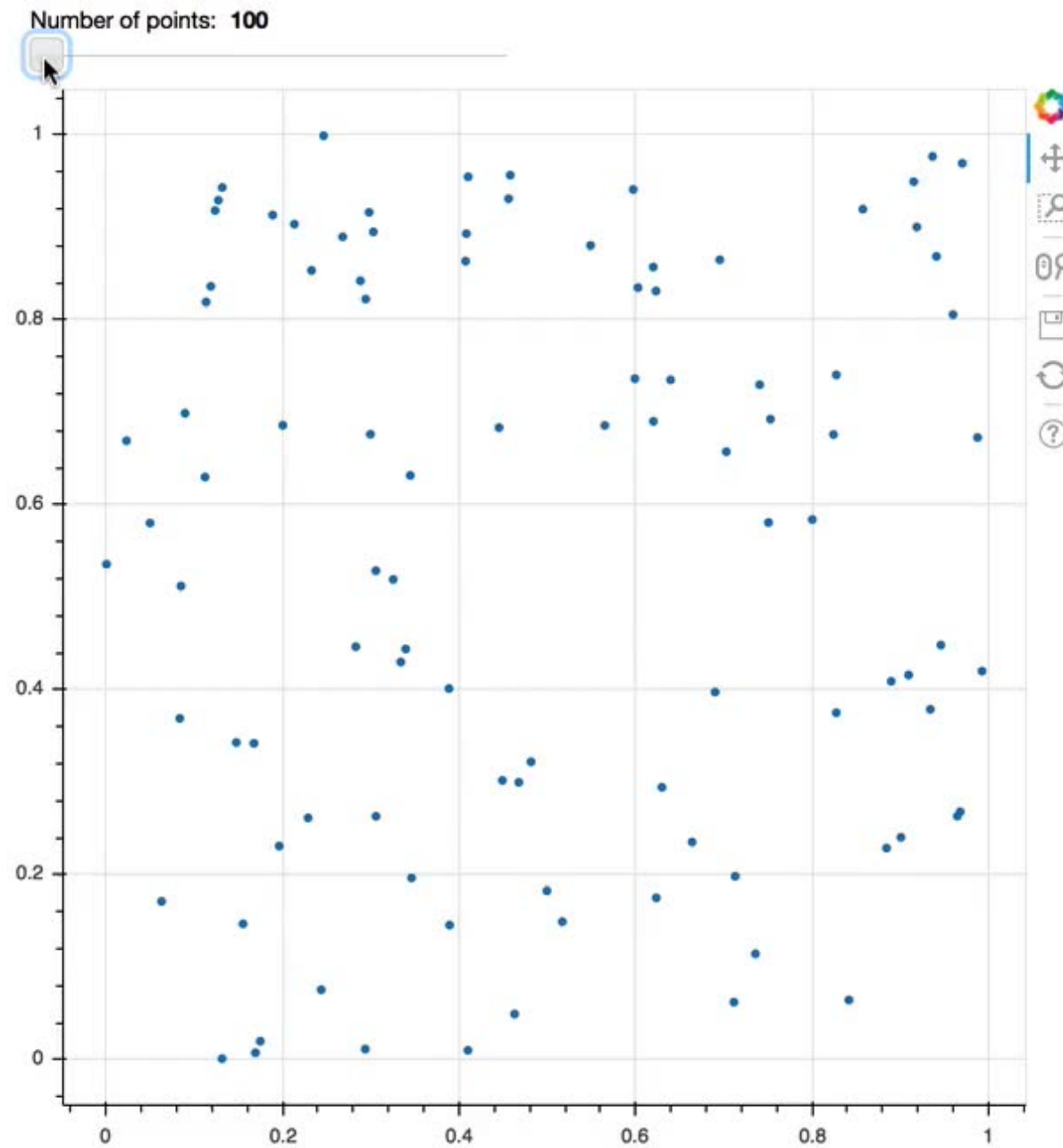
```
# (continued)

# Add callback to widgets
def callback(attr, old, new):
    N = slider.value
    source.data={'x': random(N), 'y': random(N)}

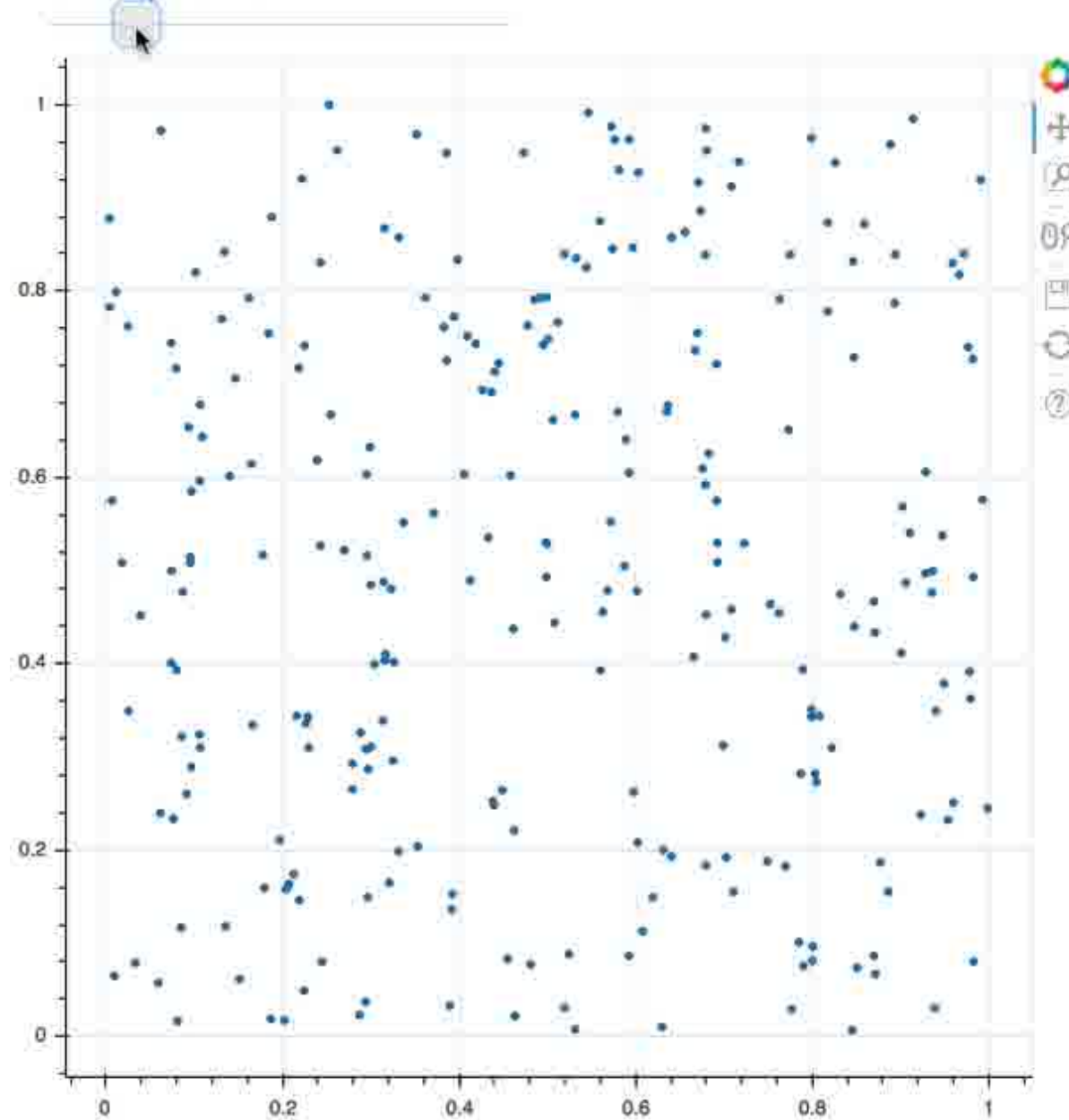
slider.on_change('value', callback)

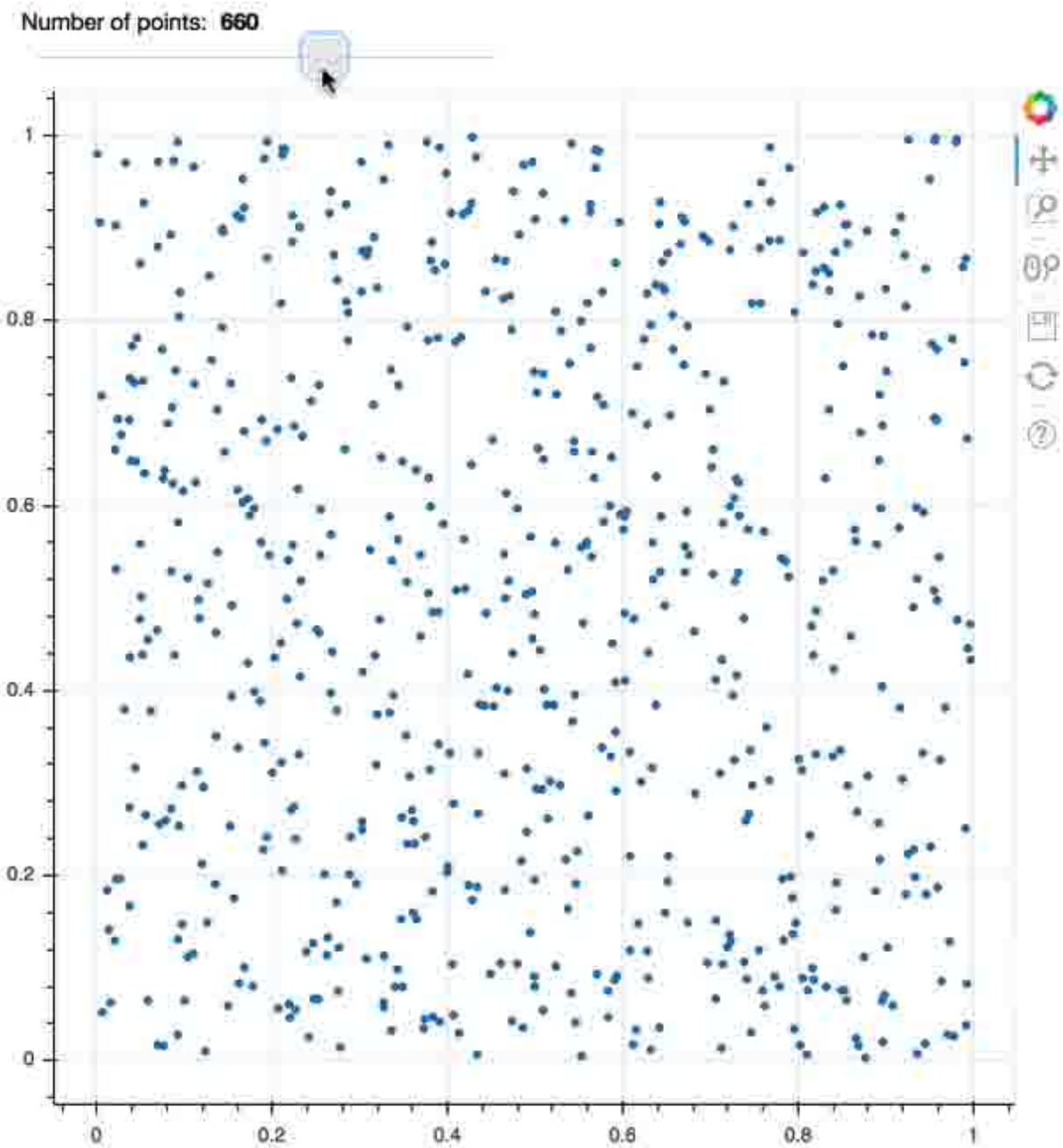
# Arrange plots and widgets in layouts
layout = column(slider, plot)

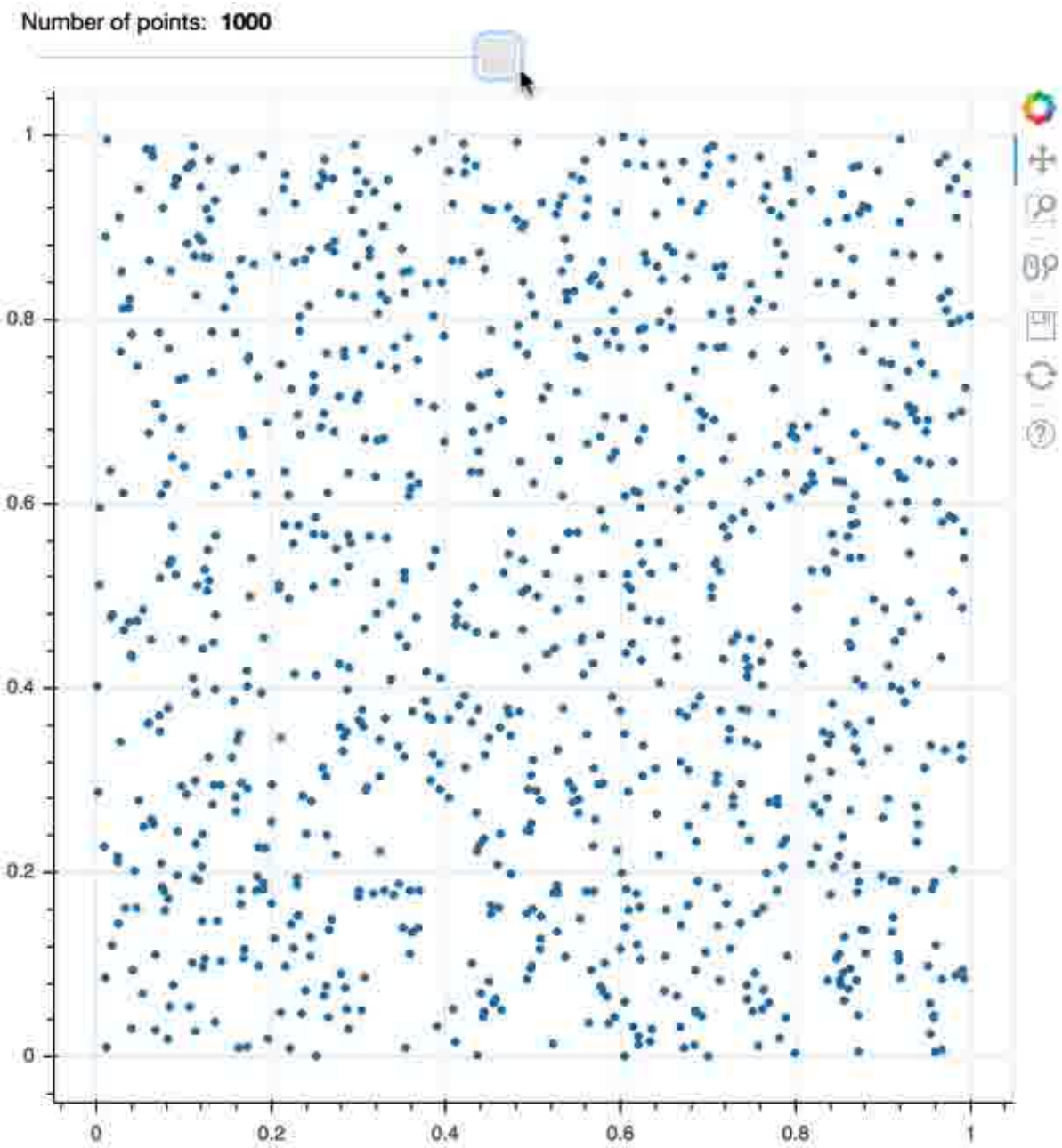
curdoc().add_root(layout)
```

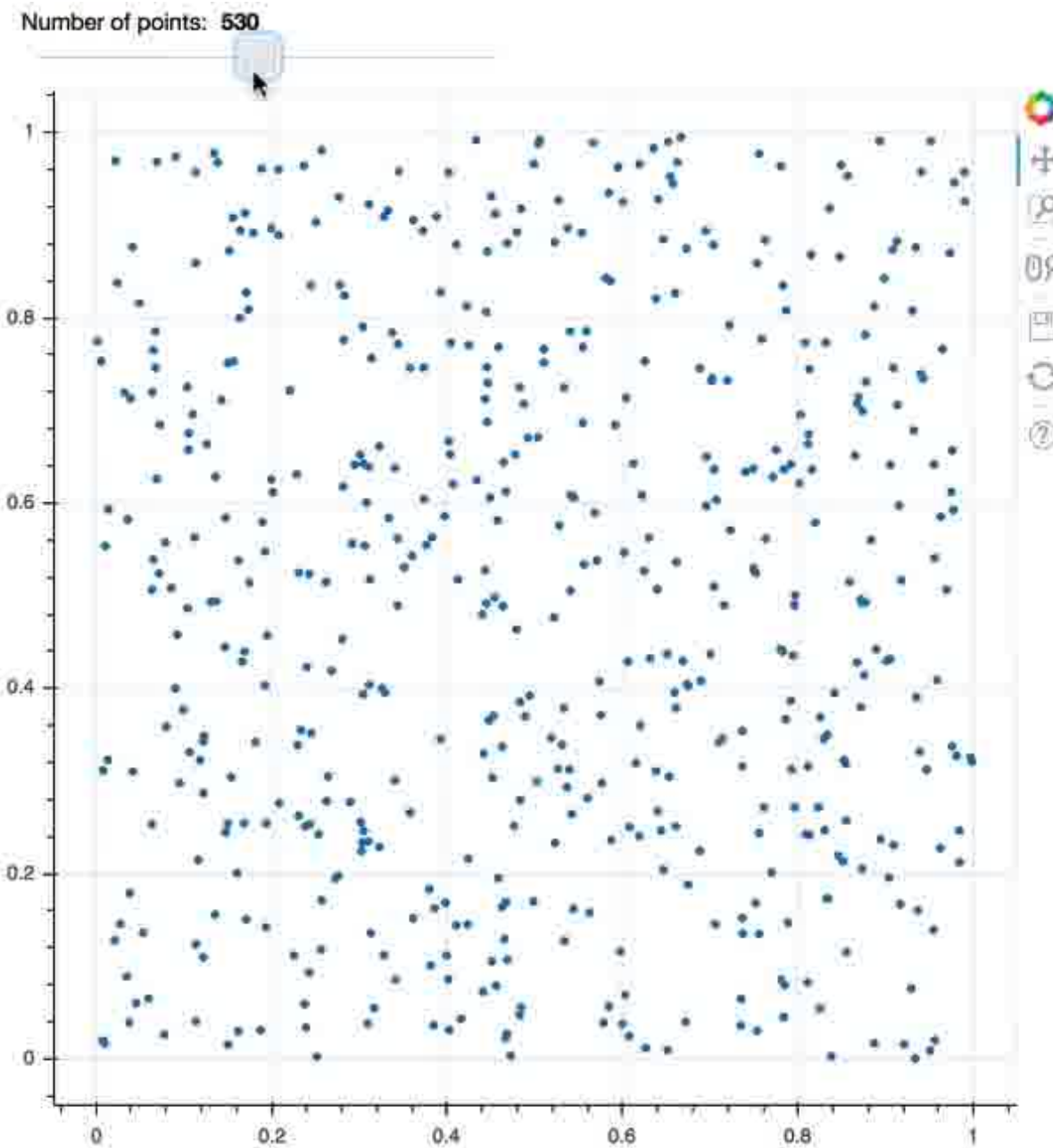


Number of points: 270









Let's practice!

INTERACTIVE DATA VISUALIZATION WITH BOKEH

Updating plots from dropdowns

INTERACTIVE DATA VISUALIZATION WITH BOKEH



Bryan Van de Ven
Core Developer of Bokeh

A Select example

select.py

```
from bokeh.io import curdoc
from bokeh.layouts import import column
from bokeh.models import ColumnDataSource, Select
from bokeh.plotting import figure
from numpy.random import random, normal, lognormal

N = 1000
source = ColumnDataSource(data={'x': random(N), 'y': random(N)})

# Create plots and widgets
plot = figure()
plot.circle(x='x', y='y', source=source)

menu = Select(options=['uniform', 'normal', 'lognormal'],
              value='uniform', title='Distribution')
```

A Select example

select.py

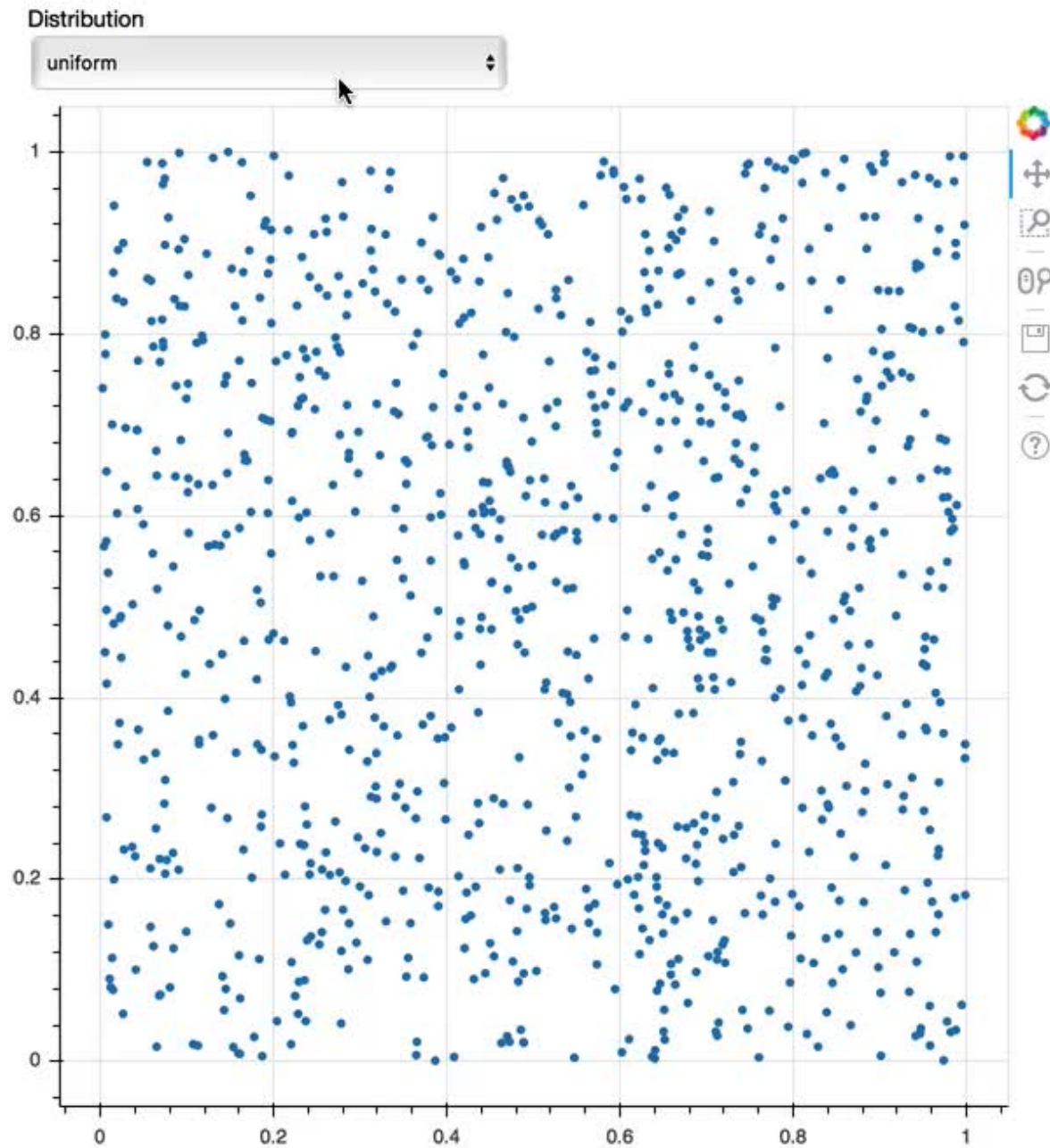
```
# (continued)

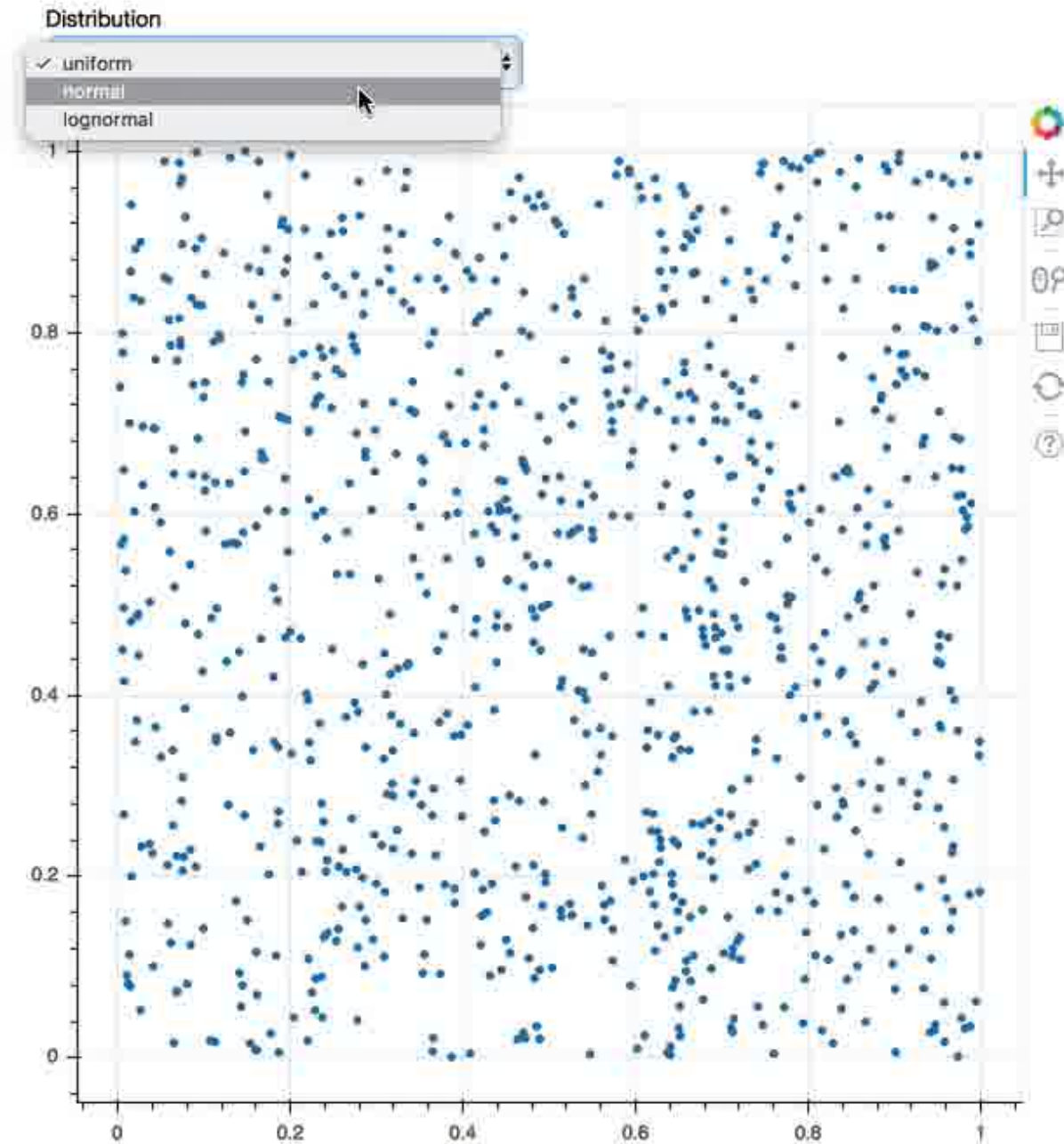
# Add callback to widgets
def callback(attr, old, new):
    if menu.value == 'uniform': f = random
    elif menu.value == 'normal': f = normal
    else: f = lognormal
    source.data={'x': f(size=N), 'y': f(size=N)}

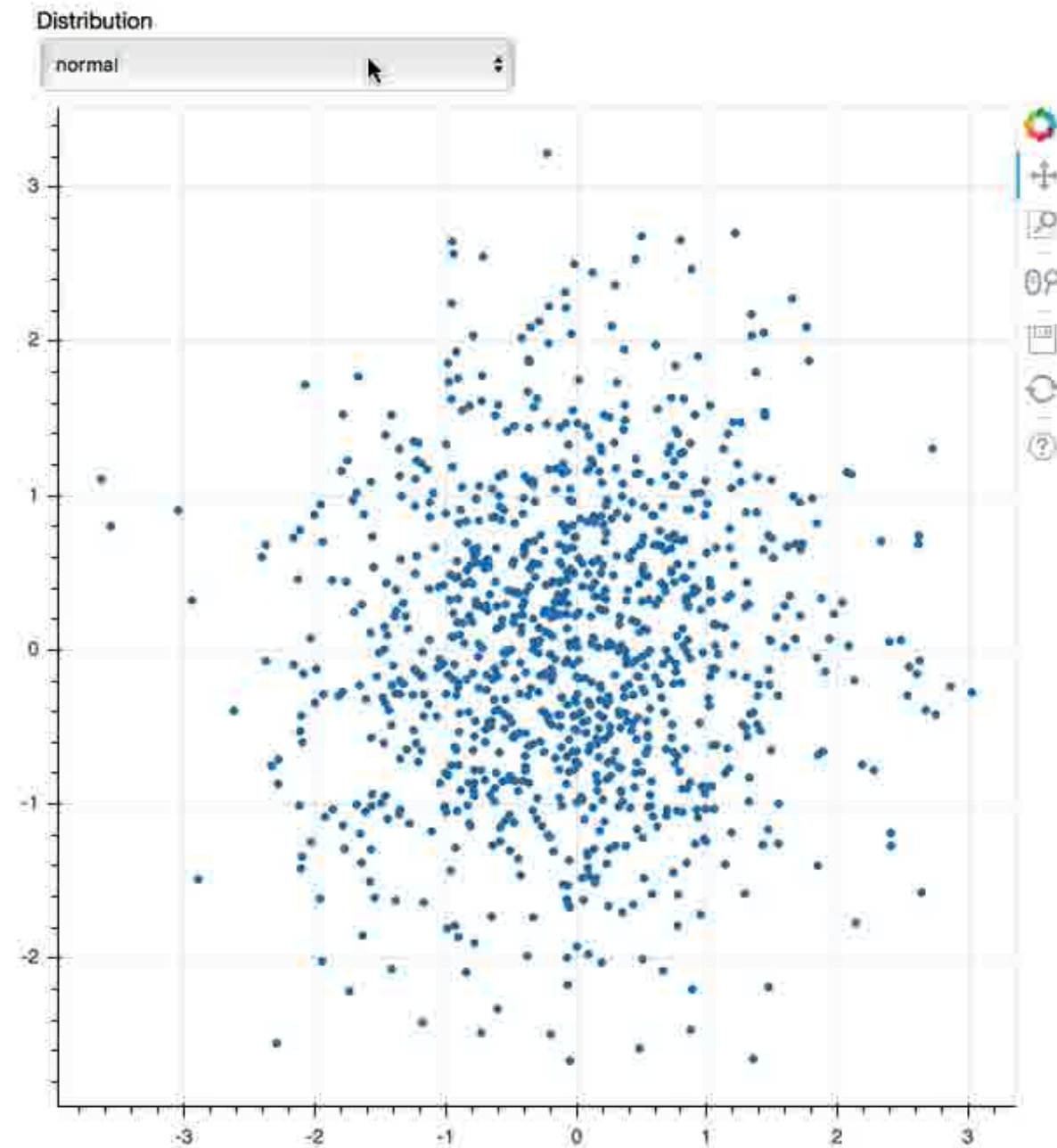
menu.on_change('value', callback)

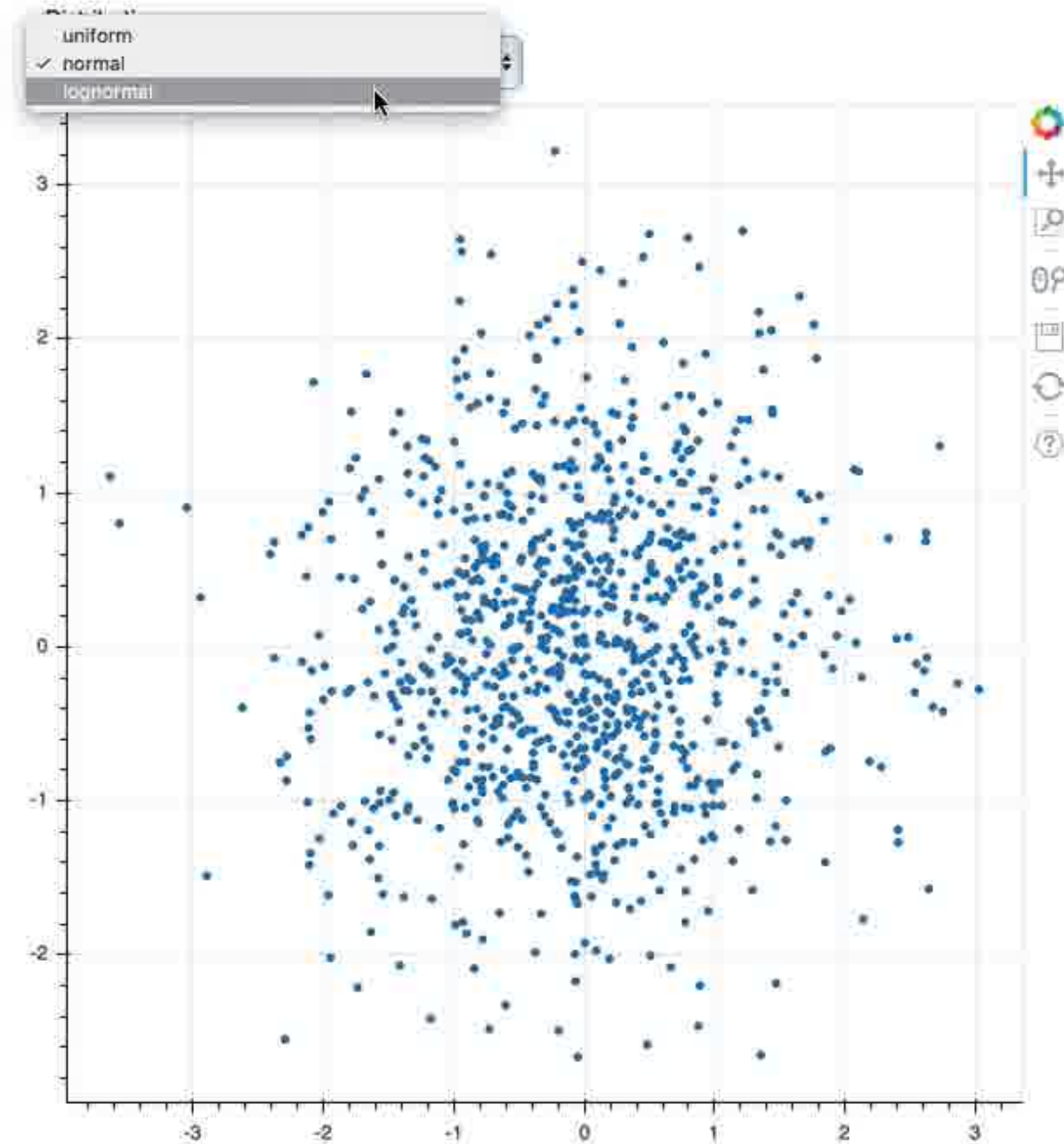
# Arrange plots and widgets in layouts
layout = column(menu, plot)

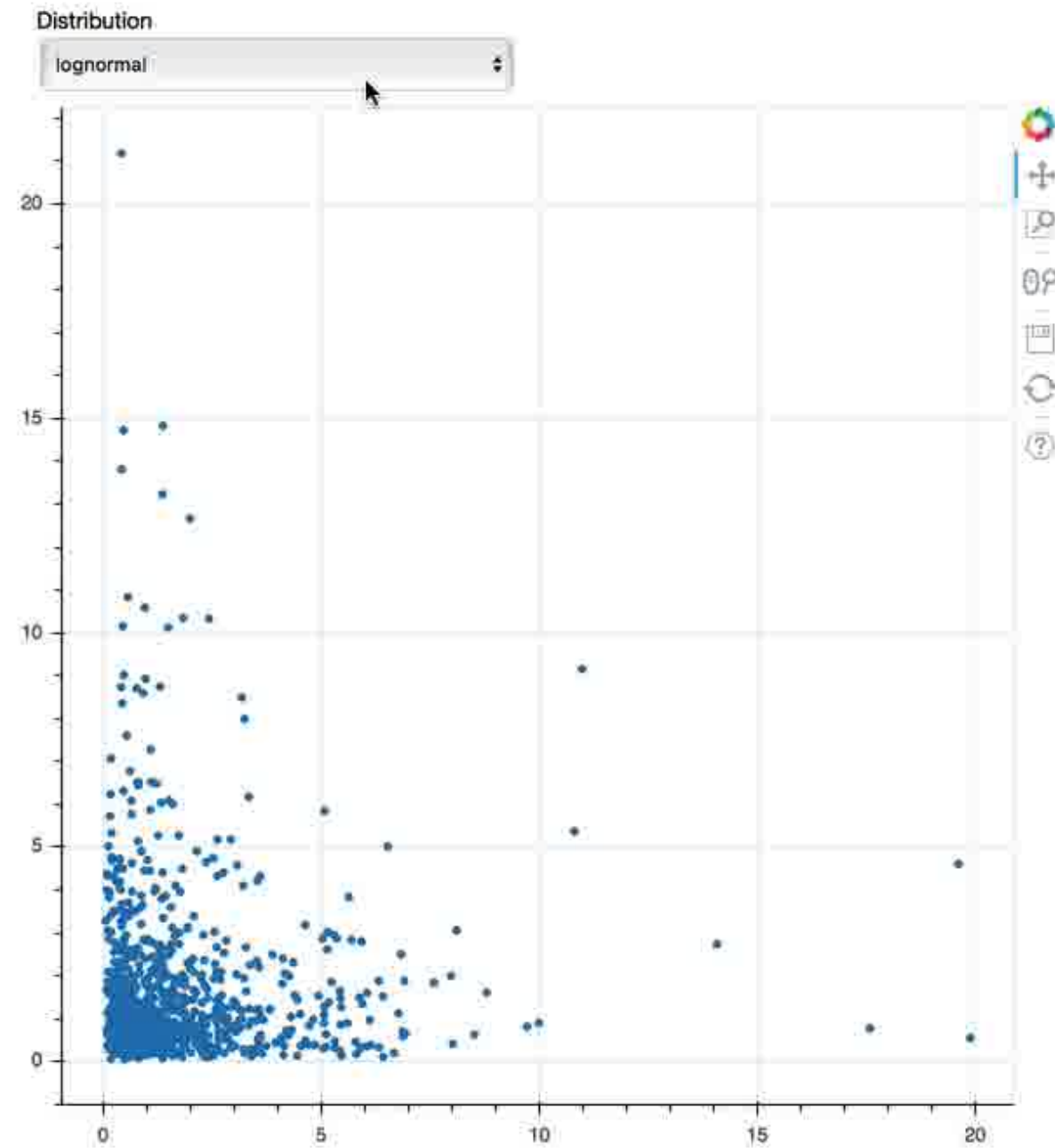
curdoc().add_root(layout)
```











Let's practice!

INTERACTIVE DATA VISUALIZATION WITH BOKEH

Buttons

INTERACTIVE DATA VISUALIZATION WITH BOKEH



Bryan Van de Ven
Core Developer of Bokeh

Button callbacks

select.py

```
from bokeh.models import Button

button = Button(label='press me')

def update():
    # Do something interesting

button.on_click(update)
```

Button types

```
from bokeh.models import CheckboxGroup, RadioGroup, Toggle

toggle = Toggle(label='Some on/off', button_type='success')

checkbox = CheckboxGroup(labels=['foo', 'bar', 'baz'])

radio = RadioGroup(labels=['2000', '2010', '2020'])

def callback(active):
    # Active tells which button is active
```

Button types

Plain button

press me

Toggle

Some on/off

Radio Group

☐ foo

☐ bar

☐ baz

Checkbox Group

☐ 2000

☐ 2010

☐ 2020

Let's practice!

INTERACTIVE DATA VISUALIZATION WITH BOKEH

Hosting applications for wider audiences

INTERACTIVE DATA VISUALIZATION WITH BOKEH



Bryan Van de Ven
Core Developer of Bokeh

Bokeh Application Hosting



<https://anaconda.org>

Let's practice!

INTERACTIVE DATA VISUALIZATION WITH BOKEH