0123_holoviews_basic_example

July 31, 2019

1 Holoviews basic example

This is a very simple of example of producing an interactive visualisation using Holoviews (which calls on Bokeh). These visualisations can be viewed in Jupyter notebooks, or may be saved as a single html page which needs only a web browser to see. Here we show room temperature and humidity, with the plots allowing the choice of which room to show.

To create these interactive plots you will need to install pyviz, holoviews and bokeh as described below.

1.1 Install libraries if needed

From terminal run:

```
conda install -c pyviz holoviews bokeh
holoviews --install-examples
```

1.2 Import libraries

```python
[1]: import numpy as np
     import pandas as pd
     import holoviews as hv
     import panel as pn
     from holoviews import opts
     hv.extension('bokeh')
```

1.3 Create some dummy data

```python
[2]: # Set length of data set to create
     length = 25

     # Build strings for location data
     location1 = ['Window'] * length
     location2 = ['Porch'] * length
     location3 = ['Fridge'] * length

     # Set temperature to normal distribution (mu, sigma, length)
     temperature1 = np.random.normal(25 ,5, length)
```

```python
temperature2 = np.random.normal(15 ,3, length)
temperature3 = np.random.normal(4, 0.5,length)

# Set temperature to uniform distribution (min, max, length)
humidity1 = np.random.uniform(30, 60, length)
humidity2 = np.random.uniform(60, 80, length)
humidity3 = np.random.uniform(80, 99, length)

# Record mean temperature/humidity (use np.repeat to repeata single value)
mean_temp1 = np.repeat(np.mean(temperature1), length)
mean_temp2 = np.repeat(np.mean(temperature2), length)
mean_temp3 = np.repeat(np.mean(temperature3), length)

mean_humidity1 = np.repeat(np.mean(humidity1), length)
mean_humidity2 = np.repeat(np.mean(humidity2), length)
mean_humidity3 = np.repeat(np.mean(humidity3), length)

# Concatenate three sets of data into single list/arrays
location = location1 + location2 + location3
temperature = np.concatenate((temperature1, temperature2, temperature3))
mean_temperature = np.concatenate((mean_temp1, mean_temp2, mean_temp3))
humidity = np.concatenate((humidity1, humidity2, humidity3))
mean_humidity = np.concatenate((mean_humidity1, mean_humidity2, mean_humidity3))

# Create list of days
days = list(range(1,length + 1))
day = days * 3 # times 3 as there are three locations

# Transfer data to pandas DataFrame
data = pd.DataFrame()
data['day'] = day
data['location'] = location
data['temperature'] = temperature
data['humidity'] = humidity
data['mean_temperature'] = mean_temperature
data['mean_humidity'] = mean_humidity
```

```python
[3]: data.head()
```

```
[3]:    day location  temperature   humidity  mean_temperature  mean_humidity
     0    1   Window    26.081745  49.611333         25.222169       45.43133
     1    2   Window    31.452276  39.027559         25.222169       45.43133
     2    3   Window    19.031828  58.825912         25.222169       45.43133
     3    4   Window    21.309825  52.741160         25.222169       45.43133
     4    5   Window    13.529042  39.977335         25.222169       45.43133
```

1.4 Build bar chart

```
[4]: # Make holoviews data table
     key_dimensions   = ['location']
     value_dimensions = ['day', 'temperature', 'humidity', 'mean_temperature',␣
       ↪'mean_humidity']
     hv_data = hv.Table(data, key_dimensions, value_dimensions)

     # Build bar charts
     bars1 = hv_data.to.bars(['day'], ['temperature'])
     bars2 = hv_data.to.bars(['day'], ['humidity']).opts(color='Red')

     # Compose plot
     bar_plot = bars1 + bars2

     # Show plot (only work in Jupyter notebook)
     bar_plot
```

```
[4]: :Layout
        .HoloMap.I  :HoloMap   [location]
           :Bars   [day]   (temperature)
        .HoloMap.II :HoloMap   [location]
           :Bars   [day]   (humidity)
```

1.5 Build scatter chart

```
[5]: # Build scatter charts
     scatter1 = hv_data.to.scatter(['day'], ['temperature'])
     scatter2 = hv_data.to.scatter(['day'], ['humidity']).opts(color='Red')

     # Compose plot
     scatter_plot = scatter1 + scatter2

     # Show plot
     scatter_plot
```

```
[5]: :Layout
        .HoloMap.I  :HoloMap   [location]
           :Scatter   [day]   (temperature)
        .HoloMap.II :HoloMap   [location]
           :Scatter   [day]   (humidity)
```

1.6 Build line chart for mean termperature and humidity

```
[6]: # Build line charts
     line1 = hv_data.to.curve(['day'], ['mean_temperature'])
     line2 = hv_data.to.curve(['day'], ['mean_humidity']).opts(color='r')
```

```
# Compose plot
line_chart = line1 + line2

# Show poy
line_chart
```

[6]: :Layout
    .HoloMap.I :HoloMap   [location]
       :Curve   [day] (mean_temperature)
    .HoloMap.II :HoloMap   [location]
       :Curve   [day] (mean_humidity)

1.7 Combine line and scatter charts

```
[7]: # Compose plot (* creates overlays of two or more plots)
combined_plot = line1 * scatter1 + line2 * scatter2

# Show plot
combined_plot
```

[7]: :Layout
    .HoloMap.I :HoloMap   [location]
       :Overlay
          .Curve.I   :Curve   [day] (mean_temperature)
          .Scatter.I :Scatter [day]   (temperature)
    .HoloMap.II :HoloMap   [location]
       :Overlay
          .Curve.I   :Curve   [day] (mean_humidity)
          .Scatter.I :Scatter [day]   (humidity)

1.8 Save to html

```
[8]: hv.save(combined_plot, 'holoviews_example.html')
```