1 Visualization with Bokeh

- Bokeh offers both powerful and flexible features which imparts simplicity and highly advanced customization.
- It provides multiple visualization interfaces to the user as shown below:`Bokeh_Interface`

Charts: a high-level interface that is used to build complex statistical plots as quickly and in a simplistic manner.

Plotting: an intermediate-level interface that is centered around composing visual glyphs.

Models: a low-level interface that provides the maximum flexibility to application developers.

In this article, we will look at first two interfaces charts and plotting only.

We will discuss models and other advance feature of this library in next post.

2 Charts

As mentioned above, it is a high level interface used to present information in standard visualization form. These forms include box plot, bar chart, area plot, heat map, donut chart and many others. You can generate these plots just by passing data frames, numpy arrays and dictionaries.

Lets look at the common methodology to create a chart:

- Import the library and functions/ methods
- Prepare the data
- Set the output mode (Notebook, Web Browser or Server)
- Create chart with styling option (if required)
- Visualize the chart

To understand these steps better, let me demonstrate these steps using example below:

Charts Example-1:

Create a bar chart and visualize it on web browser using Bokeh

We will follow above listed steps to create a chart:

```
#Import library

from bokeh.charts import Bar, output_file, show
#use output_notebook to visualize it in notebook

# prepare data (dummy data)
data = {"y": [1, 2, 3, 4, 5]}

# Output to Line.HTML
output_file("lines.html", title="line plot example") #put output_n

# create a new line chat with a title and axis labels
p = Bar(data, title="Line Chart Example", xlabel=’x’, ylabel=’valu

# show the results
show(p)
% % % --- Bar_Chart
```

- **Important:** In the chart above, you can see the tools at the top (zoom, resize, reset, wheel zoom) and these tools allows you to interact with chart.
- You can also look at the multiple chart options (legend, xlabel, ylabel, xgrid, width, height and many other) and various example of charts here.

Chart Example-2: Compare the distribution of sepal length and petal length of IRIS data set using Box plot on notebook

To create this visualization, firstly, Ill import the iris data set using ***sklearn library***. Then, follow the steps as discussed above to visualize chart in ipython notebook.

```
#IRIS Data Set

from sklearn.datasets import load_iris
import pandas as pd
iris = load_iris()
df=pd.DataFrame(iris.data)
df.columns=[’petal_width’,’petal_length’,’sepal_width’,’sepal_leng

#Import library
from bokeh.charts import BoxPlot, output_notebook, show
data=df[[’petal_length’,’sepal_length’]]

# Output to Notebook
output_notebook()
# create a new line chat with a title and axis labels
p = BoxPlot(data, width=400, height=400)

# show the results
show(p)
Bokeh_Box_Plot
```

Chart Example-3: Create a line plot to bokeh server

Prior to plotting visualization to Bokeh server, you need to run it.

If you are using a conda package, you can use run command bokeh-server from any directory using command. Else, `python ./bokeh-server` command should work in general. For more detail on this please refer this link Deploying Bokeh Server.

There are multiple benefits of Plotting visualization on Bokeh server:

- Plots can be published to larger audience
- Visualize large data set interactively
- Streaming data to automatically updating plots
- Building dashboards and apps

To start plotting on Bokeh server, I have executed the command bokeh-server to initialize it followed by the commands used for visualization.

```
### Bokeh_Server

from bokeh.plotting import figure, output_server, show
output_server("line")
p = figure(plot_width=400, plot_height=400)
# add a line renderer
p.line([5, 2, 3, 4, 5], [5, 7, 2, 4, 5], line_width=2)
show(p)
% % % %---- Bokeh_Server_Visualization
```