Introduction to Interactive Visualization in Python

Junying (Alice) Fang

Download Workshop Codes

https://github.com/fjying/DashPlotlyWorkshop

1. Open the Repo Link
2. Click the Green Code Button on the Top Right of the Page
3. Click Download Zip
4. Zip the Download Folder
5. Rename the Folder as DashPlotlyWorkshop
6. Put the Folder on the appropriate place

Set Up Environment to Run Codes

If Anaconda Python is Installed:

1. Open the terminal

2. cd to the repo folder path

cd <.../DashPlotlyWorkshop>

3. Create New Conda Environment

conda env create -f environment.yml

conda activate dashplotlyworkshop

4. Install Package named dashplotlyworkshop

pip install -e .

If Anaconda is not Installed:

1. Install All Necessary Packages

pip install plotly dash==2.6.2 dash-bootstrap-components==1.2.1

2. Install Package named dashplotlyworkshop

pip install -e .

Generate the First Dashboard

Run on Terminal:

1. Run Workshop Codes to Generate the First Dashboard

cd dashplotlyworkshop/onepagedashboard

python app_run.py

2. Look at dashboard:

Enter http://127.0.0.1:8050/ on the browser

3. Stop the running of dashboard

Control C

Run on IDE such as PyCharm and VSCode:

1. Open the Repo Folder named DashPlotlyWorkshop on IDE

2. Run Python File directly on IDE

dashplotlyworkshop/onepagedashboard/app_run.py

3. Look at dashboard:

Enter http://127.0.0.1:8050/ on the browser

4. Stop the running of dashboard

Control C

Outline

1. Plotly Introduction
2. Dash Introduction
3. Time Series Plot using Plotly
4. Single Page Dashboard using Plotly and Dash
5. Multiple Pages Dashboard
6. More Dashboards for Research Analysis: Table, Scatterplot, Heatmap

Plotly

Alternative to Matplotlib:

(1) Interactive Plot

(2) Direct integration with Dash to build the dashboard

https://plotly.com/python/

Dash

Dash allows Python users to create the dashboard (graphical web applications generally speaking) directly without learning other programming languages such as JavaScript, HTML, or CSS.

https://dash.plotly.com/

Full-Stack Development in Python: Traditional versus Dash

	Traditional	Dash
Frontend	HTML, CSS for Visualization Flask in Python to handle dynamic requests from users	Dash allows us to handle visualization and dynamic requests by writing all codes in Python
Backend	Python	Python
Data Layer	Database	Database

Dash is a GREAT tool for researchers as it enables easier implementation and faster learning by merely using Python.

It has all necessary features to create common dashboards for data visualization for research analysis.

Plot time series of risk allocation index by asset type using Plotly

Need to first export package environment into Jupyter:

conda install -c anaconda ipykernel

python -m ipykernel install --user --name=dashplotlyworkshop

Run DashPlotlyWorkshop/dashplotlyworkshop/notebook/plotly.ipynb

How to Generate the Previous Plot of Particular Day based on User Input?

One-Page Dashboard

cd <.../DashPlotlyWorkshop/dashplotlyworkshop/onepagedashboard>

python app_run.py

app.layout

dbc.Row, dbc.Col

html.Label, html.H3, html.Wbr

dcc.Graph, dcc.Dropdown

Callback function to plot the graph based on Dropdown input

Multipages Dashboard

cd <.../DashPlotlyWorkshop/dashplotlyworkshop/multiplepagesdashboard>

python app_layout_run.py

sidebar: dbc.Nav, dbc.NavLink

url: dcc.Location()

Callback function to render each page content based on the related input ip address

Scatterplot

cd <.../DashPlotlyWorkshop/dashplotlyworkshop/moreexamples>

python scatterplots.py

Relationships between GDP per capita, Population, Country Continent and Life Expectancy across Time

Visualize Panel Data

cd <.../DashPlotlyWorkshop/dashplotlyworkshop/moreexamples>

python table.py

Heatmap

cd <.../DashPlotlyWorkshop/dashplotlyworkshop/moreexamples>

python heatmap.py

Could ZOOM IN to look temperature variation within particular region and time range in detail