Data Visualization with Python

IBM Course Eight Week One

Objectives

- Make visualizations effective, attractive, and impactive
- Explain Darkhorse Analytics and apply the approach to your own visualizations
- Use Matplotlibs to create plots
- Identify the three layers of Matplotlibs, and understand the functions of each layer

Intro to Data Visualization

- A picture is worth 1000 words!
- Creating and transforming visuals that are effective, attractive, and impactful
- Graphical form □ easy to understand
- Good for when you need to get acquainted with the data
- Don’t distract from the data – find a balance

Why Build Visuals?

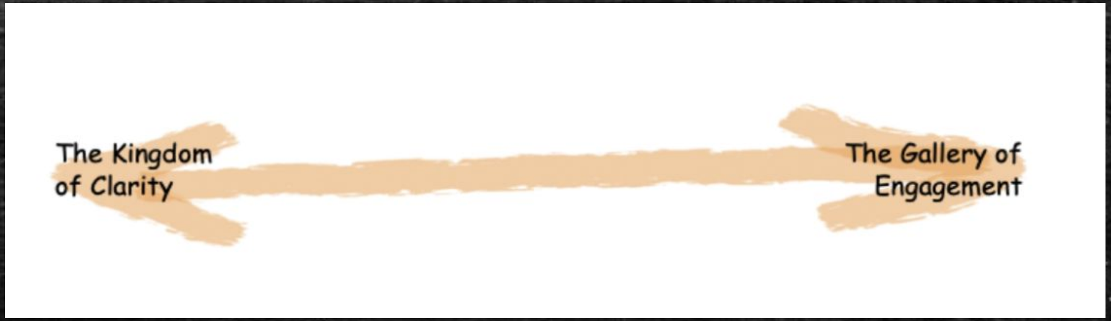
Convey a clear description of the data to disclose findings to audience

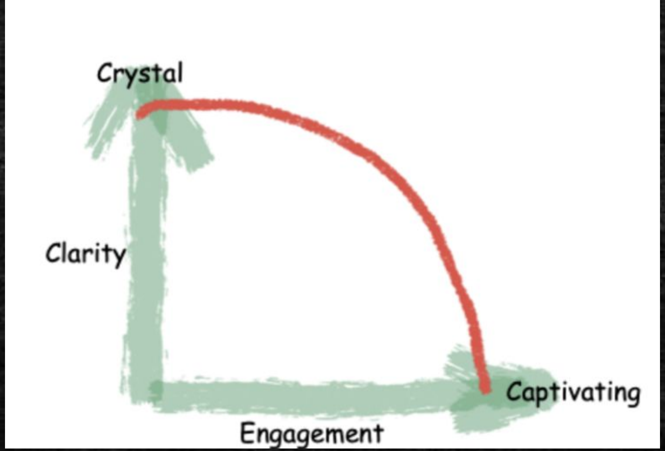
Darkhorse Analytics

“Less is more…”

Darkhorse Analytics Approach

- Crystal
- Clarity
- Engagement
- Captivating



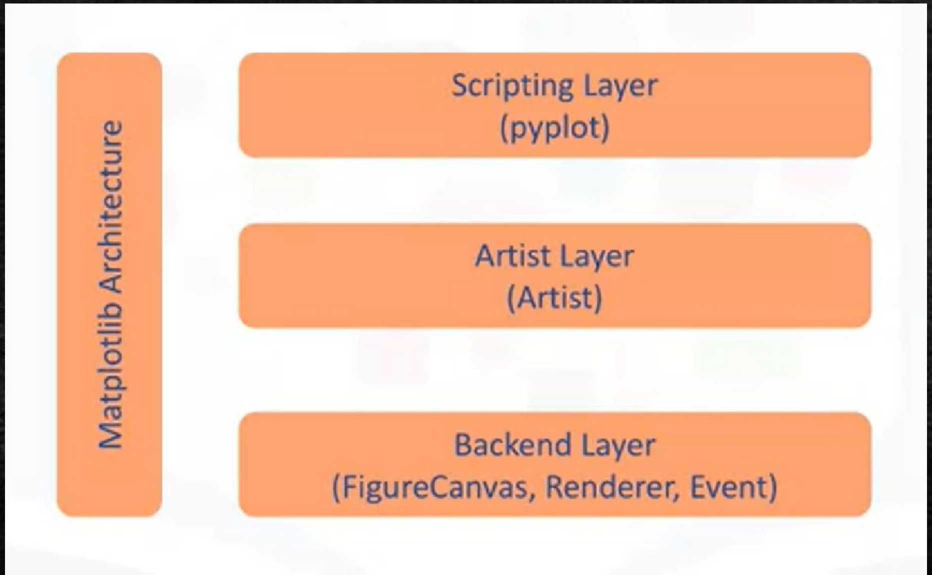


Ask the Right Questions

- Who is your target audience?
- What is the main message you're trying to convey?
- What do you want your audience to do about it?
- *Be as specific as possible when exploring these questions.*

Matplotlib

- Created by John Hunter , neurobiologist
- Most popular data visualization library in Python
- Three main layers: backend, artist, scripting layer



Matplotlib Architecture

Scripting Layer
(pyplot)

Artist Layer
(Artist)

Backend Layer
(FigureCanvas, Renderer, Event)

Matplotlib’s Layers

Backend

- 3 built-in abstract interface classes: FigureCanvas, Renderer, Event

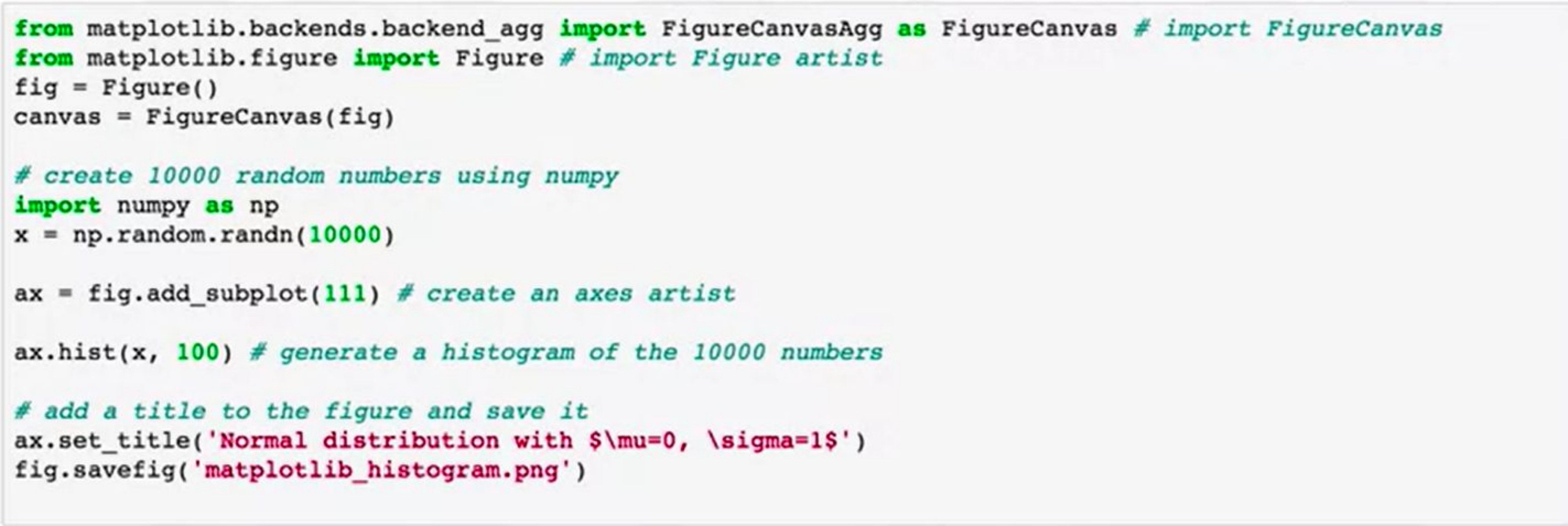
Artist

- One main object: the artist
- Two types of artist objects: primitive and composite

Scripting

- Meant for individuals who are not professional programmers
- Automates processes

Break it Down



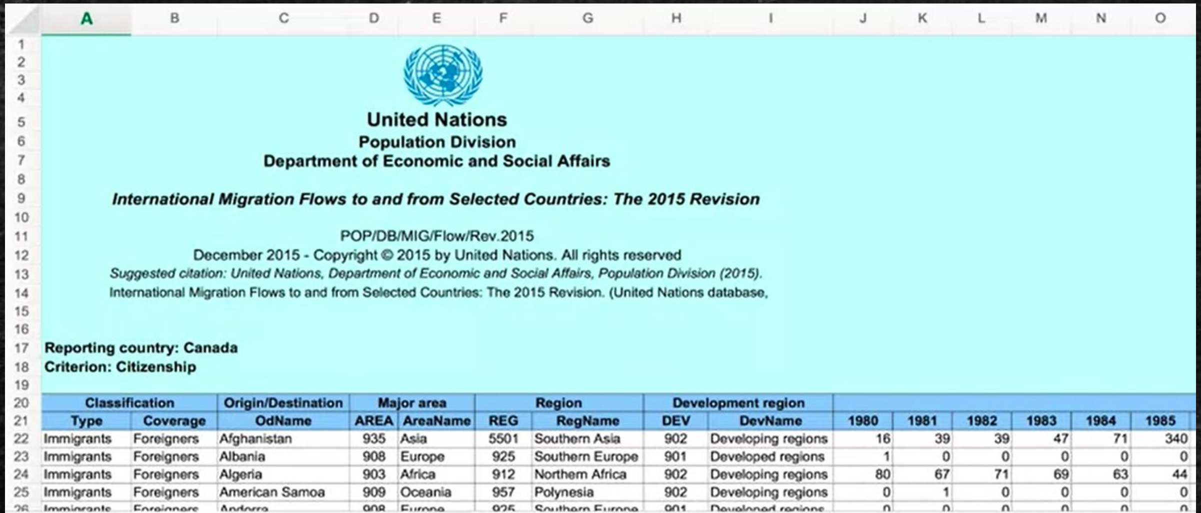
```
from matplotlib.backends.backend_agg import FigureCanvasAgg as FigureCanvas # import FigureCanvas
from matplotlib.figure import Figure # import Figure artist
fig = Figure()
canvas = FigureCanvas(fig)

# create 10000 random numbers using numpy
import numpy as np
x = np.random.randn(10000)

ax = fig.add_subplot(111) # create an axes artist

ax.hist(x, 100) # generate a histogram of the 10000 numbers

# add a title to the figure and save it
ax.set_title('Normal distribution with $\mu=0, \sigma=1$')
fig.savefig('matplotlib_histogram.png')
```

Basic Plotting

- Jupyter Notebook
- Pandas □ built in
- Plot function
- Magic Function
- Notebook Backend

What is a limitation of the Magic function that the Notebook backend can be used to overcome?

Info on the Dataset □ Immigration to Canada



United Nations
Population Division
Department of Economic and Social Affairs

International Migration Flows to and from Selected Countries: The 2015 Revision

POP/DB/MIG/Flow/Rev.2015
December 2015 - Copyright © 2015 by United Nations. All rights reserved
Suggested citation: United Nations, Department of Economic and Social Affairs, Population Division (2015).
International Migration Flows to and from Selected Countries: The 2015 Revision. (United Nations database,

Reporting country: Canada
Criterion: Citizenship

Classification		Origin/Destination	Major area		Region		Development region							
Type	Coverage	OdName	AREA	AreaName	REG	RegName	DEV	DevName	1980	1981	1982	1983	1984	1985
Immigrants	Foreigners	Afghanistan	935	Asia	5501	Southern Asia	902	Developing regions	16	39	39	47	71	340
Immigrants	Foreigners	Albania	908	Europe	925	Southern Europe	901	Developed regions	1	0	0	0	0	0
Immigrants	Foreigners	Algeria	903	Africa	912	Northern Africa	902	Developing regions	80	67	71	69	63	44
Immigrants	Foreigners	American Samoa	909	Oceania	957	Polynesia	902	Developing regions	0	1	0	0	0	0
Immigrants	Foreigners	Andorra	908	Europe	925	Southern Europe	901	Developed regions	0	0	0	0	0	0

- Data pertaining to 45 countries
- Will create plots for the data for exploratory analysis and presentation
- Use Pandas and xlrd □ import libraries
- Import data and check you've imported the right rows/columns

What is a Line Plot?

Take the next 1.5 hours to do the lab, quiz, and final reading.