Visualizing Data

As a general-purpose programming language, Python is incredibly useful for analyzing data and visualizing results. This activity is a first look at `matplotlib`, one of the most widely used 2D plotting libraries.

Manager:		Recorder:	
Presenter:		Reflector:	

Content Learning Objectives

After completing this activity, students should be able to:

- Explain the basic structure of code for plotting a mathematical function.
- Analyze visually the behavior of the Python random number generator.
- Read data from a CSV file and generate histograms of various columns.

Process Skill Goals

During the activity, students should make progress toward:

- Navigating the documentation for a third-party library. (Information Processing)

Facilitation Notes

Give students a copy of the following files: *simpleplot.py*, *histogram.py*, *scorecard.csv*.

For each of the examples in this activity, it helps to run the code on the projector and discuss the results visually when reporting out. On Model 1, be sure to explain the `arange` function and how it works differently from `range`. For example, `range` requires an integer for the third argument, but `arange` allows floats. If time permits, explain briefly what an `array` is in `numpy`.

In addition to plotting histograms, the point of Model 2 is to give students further insight to how random numbers work. Have teams come up with an explanation of why some random numbers appear to occur more often than others (when the sample size is low). Demonstrate the results of #12 on the board to facilitate discussion.

On Model 3, students may need help counting rows and columns in the *data.csv* file shown. Check their answers for the first three questions before they proceed to the provided Scorecard Data. Questions #18 and #19 can take a lot of time if they are not savvy with Excel. You might want to answer these questions for the entire class on the projector (e.g., demonstrate how to find the range of values by sorting the data).

The NULL values in the Scorecard Data will likely confuse students. Monitor team discussions, and as needed, point out that "NULL" is a literal string in the file contents. It's not interpreted as `None` in Python.

Model 1 Simple Plot

When analyzing data, it's helpful to create charts, plots, and other visualizations. Doing so allows you to see important numerical relationships. Enter the following code into a Python Editor, and run the program.

```
1  import matplotlib.pyplot as plt
2  import numpy as np
3
4  def model_one():
5      x = np.arange(0.0, 2.0, .01)
6      y = np.sin(2 * np.pi * x)
7      plt.plot(x, y)
8      plt.xlabel('time (s)')
9      plt.ylabel('volts (mV)')
10     plt.show()
11
12 model_one()
```

Questions (15 min) **Start time:**

1. Identify in the source code which line numbers:

a) generated the data? 5–6

b) set the axes properties? 8–9

c) displayed the window? 10

d) plotted the actual data? 7

2. Describe in your own words what is being plotted.

It plots a sine wave with an amplitude of 2, a period of 1, and a vertical shift of 1. The x and y values range from 0 to 2.

3. Modify the code to plot only one cycle of the sine wave (instead of two). Write the edited line of code below.

`x = np.arange(0.0, 1.0, .01)` OR `y = np.sin(np.pi * x)`

4. Change the third argument of `np.arange` from 0.01 to 0.15. What is the result?

Less curvature; it almost looks like a triangle.

5. Add `"o"` as a third argument to the `plot` function. What is the result?

Plots points instead of a line.

6. How does the third parameter of `np.arange` affect how the plot looks?

It determines how many points to generate, which makes the plot look smoother, but it also takes longer to draw.

7. How would you modify the code to plot the function $y = x^2 - 1$ instead? Show the results from -2 to +2.

Replace Line 5 with `x = np.arange(-2.0, 2.0, 0.01)` and Line 6 with `y = x ** 2 - 1`.

8. Which two Python libraries are used in Model 1? Quickly search the Internet and find their websites. Write a one-sentence description about each library.

- `matplotlib`: 2D plotting library which produces publication quality figures. It includes `pyplot`, which provides a MATLAB-like plotting framework.
- `numpy`: Fundamental package for scientific computing (part of SciPy).

Model 2 Histograms

Recall that you can generate a sequence of numbers using the `random` module. Merge the code below into your program from Model 1. Run the program, and view the output.

```
1  import matplotlib.pyplot as plt
2  import random
3
4  def model_two(npts):
5      numbers = []
6      for _ in range(npts):
7          numbers.append(random.random())
8      plt.hist(numbers)
9      plt.show()
10
11 model_two(100)
```

Questions (10 min) Start time:

9. Based on the Python code:

a) What is the range of values generated by the `random` function? [0, 1)

b) How many random values are generated? 100

10. Based on the figure plotted:

a) How many bars are displayed? 10

b) What is the width of each bar? 0.1

c) What is the sum of the heights of the bars? 100

11. Based on your answers above, what are appropriate labels for the x and y axes?

The x-axis could be "random number value", and the y-axis could be "frequency" (number of times generated).

12. Increase the argument of `model_two` to 1000, 10000, and 100000. Describe how the output plot changes when you run the program.

The bars get skinnier and more uniform. The y-axis also increases in range. And it takes a lot longer to plot.

13. Add the number 50 as second argument to the `hist` function. What is the meaning of the result?

It plots 50 bars instead of 10, and it looks look less uniform. There are more ranges in which the numbers may fall.

14. In general, describe what the `hist` function does with the list of random numbers to create this type of plot.

It groups them into bins (based on the second argument of `hist`), and draws a bar for the number of values in each bin.

Model 3 CSV Data

"Comma Separated Values" is a common file format when exporting data from spreadsheets and databases. Each line of the file is a row, and each column is separated by a comma. Cells that contain commas are wrapped in quote marks.

`data.csv` file contents:

```
Name,Location,URL,Students
Westminster College,"Salt Lake City, UT",westminstercollege.edu,2135
Muhlenberg College,"Allentown, PA",muhlenberg.edu,2330
University of Maine,"Orono, ME",umaine.edu,8677
James Madison University,"Harrisonburg, VA",jmu.edu,19019
Michigan State University,"East Lansing, MI",msu.edu,38853
```

Python includes a `csv` module (https://docs.python.org/3/library/csv.html) that makes it easy to read and write CSV files.

```
import csv

with open("data.csv") as file:
    csv_data = list(csv.reader(file))
names = csv_data[0]
print("Column names:", names)
for row in csv_data[1:]:
    print(row[1]) # 2nd column
```

Program output:

```
Salt Lake City, UT
Allentown, PA
Orono, ME
Harrisonburg, VA
East Lansing, MI
```

Questions (20 min) Start time:

15. In the example `data.csv` file above:

a) In what way is the first line different? It has the column names (not data)

b) How many rows of data are there? 5 How many columns? 4

16. Compare `data.csv` with the program output:

a) Are quote marks included in data.csv? Yes In the program output? No

b) What is the purpose of the quote marks? To specify a value that contains commas

17. In the Python code above:

a) Which line of code reads the first line of the file? `names = csv_data[0]`

b) What type of data does the variable `row` contain? A list of strings

In 2013, the U.S. Department of Education released the "College Scorecard" website to help students and families compare institutions of higher education. The Scorecard data includes information like average cost of attendance, graduation and retention rates, student body demographics, etc.

18. Go to https://collegescorecard.ed.gov/data/ and download the "Most Recent Institution-Level Data" . Open the CSV file in Excel or a similar program, and skim its contents.

a) How many rows does it have? 6484 (not counting the header)

b) How many columns does it have? 3305 (columns A to DWC)

19. Column KE is named UGDS, which means "Enrollment of undergraduate certificate / degree-seeking students".

a) What is the range of values in this column? 0 to 138,138

b) Which school has the most students enrolled? Southern New Hampshire University

c) Do all rows have an integer value for UGDS? No, many are NULL

20. Based on the code in Model 2 and Model 3, write a program that plots a histogram of the UGDS column. Complete the following steps to consider each part of the program.

a) What two import statements will you need at the top?

```
import csv
import matplotlib.pyplot as plt
```

b) What three statements prepare the csv file for reading?

```
with open("data.csv") as file:
    csv_data = list(csv.reader(file))
names = csv_data[0]
```

c) What code is necessary to read the entire column into a list? (Note: Use `names.index('UGDS')` to find the column index; store it in a variable called `col_index`)

```
col_idx = names.index('UGDS')
ugds = []
for row in csv_data[1:]:
    ugds.append(row[col_idx])
```

d) By default, data from text/csv files are read as strings. Write the code to convert the `row[col_index]` values to integers. Be sure not include the `"NULL"` values in the final list.

```
datum = row[col_idx]
if datum != "NULL":
    ugds.append(int(datum))
```

e) Write the last two lines that plot and show the histogram.

```
plt.hist(ugds)
plt.show()
```

21. Run the program, and compare your results with another team's. What does the histogram tell you about undergraduate enrollments in the United States?

There are thousands of schools with less than 1,000 students. Very few schools have 15,000 students or more.

22. What other questions could you ask about this data? How would you answer them using histograms, line charts, and scatter plots?

Answers will vary. For example, it would be nice to see the number of schools per state.