Assignment 3: Data Visualization in Python

(60 Points Total)

Data available under Resources>Assignment Data.

This exercise focuses on the use of **Matplotlib** and **Seaborn** to graph and visualize data. You will need to import the NumPy, Pandas, Matplotlib, and Seaborn libraries. Produce code to generate the following graphs. Deliver your results in a Jupyter Notebook.

The **high_plains_data.csv** file contains information for all counties in the high plains region of the United States. The elevation, temperature, and precipitation data were extracted from raster grids provided by the PRISM Climate Group at Oregon State University (https://prism.oregonstate.edu/). I also summarized percent forest cover from the 2011 National Land Cover Database (NLCD). NLCD data can be obtained from the Multi-Resolution Land Characteristics Consortium (MRLC) (https://www.mrlc.gov/).

Variable	Explanation
NAME_1	County name
STATE_NAME	State in which county occurs
TOTPOP10	Total county population in 2010
per_for	Percent forest cover
elev	Elevation in meters
temp	30-year annual normal mean temperature in Celsius
precip	30-year annual normal precipitation in millimeters

The **portland_park_trees.csv** file contains information about individual trees in city parks in Portland, Oregon. These data were obtained from the City of Portland Office of Parks & Recreation (https://www.portlandoregon.gov/parks/article/433143).

Variable	Explanation
DBH	Diameter at breast height in inches
Condition	Condition of tree
TreeHeight	Height of tree in feet
CrownWidth	Crown width in feet
Family	Tree family
Genus	Tree genus
Common	Tree common name
Genus_spec	Tree genus and species
Native	Whether or not tree is a native species

Graphs Relating to Portland Tree Data

Graph 1: Create a **scatterplot** for just trees in the *Ulmus* genus with DBH mapped to the x-axis and tree height mapped to the y-axis (Hint: You will need to use the "Genus", "DBH", and "TreeHeight" attributes.). **(3 Points)**

Graph 2: Create a **scatterplot** for just trees in the *Ulmus* genus with DBH mapped to the x-axis, tree height mapped to the y-axis, and tree species mapped to hue (Hint: You will need to use the "Genus", "Genus_spec", "DBH", and "TreeHeight" attributes.). **(6 Points)**

Graph 3: Create a **boxplot** of DBH for just the *Ulmus* genus. **(3 Points)**

Graph 4: Create a **boxplot** of DBH for just the *Ulmus* genus differentiated by species (or, each species should have its own boxplot). **(6 Points)**

Graph 5: Combine Graphs 1 through 4 into a single figure with 2 rows and two columns. Do not plot a legend for any of the graphs. **(6 Points)**

Graph 6: Refine Graph 4 as follows: **(9 Points)**

- Provide a plot title and increase the font size of the title.
- Define colors to use to fill the boxplots as opposed to using the default colors.
- Remove the legend.
- Change the x-axis title to "Species" and the y-axis title to "DBH". Set a font size for both.
- Set limits, tick intervals, tick interval labels, and a font size for the y-axis.
- Edit the species names on the x-axis. You will need to shorten some of them. Set a font size and italicize the scientific names. Apply a rotation to the labels.
- Add y-axis grid lines.

Graph 7: Create a figure containing Graph 4 and an additional graph of DBH vs. tree height for the *Ulmus* trees. For the new subplot, refine as follows: **(9 Points)**

- Provide a plot title and increase the font size of the title.
- Define colors to use to fill the points. These should match the colors used for the same species in the boxplot.
- Remove the legend.
- Change the x-axis title to "DBH" and the y-axis title to "Tree Height". Set a font size for both.
- Set limits, tick intervals, tick interval labels, and a font size for the x-axis.
- Set limits, tick intervals, tick interval labels, and a font size for the y-axis.
- Add y-axis grid lines.

Graphs Relating to High Plains Data

Graph 8: Create a **boxplot** of temperature with separate plots for each state (Hint: Use the "temp" and "STATE_NAME" fields.). **(3 Points)**

Graph 9: Create a **scatterplot** with elevation mapped to the x-axis and temperature mapped to the y-axis. Use hue to differentiate the states (Hint: Use the "elev", "temp", and "STATE_NAME" fields.). **(6 Points)**

Graph 10: Combine Graphs 8 and 9 into a single figure with two subplots. Refine the graphs as follows: **(9 Points)**

- Provide a plot title for each subplot and increase the font size of the titles.
- Define colors to use to fill the points and boxplots. Use the same color for each state in both plots.
- Remove legends.
- Change the y-axis title to "Temperature (Celsius)" for both graphs.
- In the boxplot, change the x-axis title to "States".
- For the scatterplot, change the x-axis title to "Elevation (meters)".
- Set limits, tick intervals, tick interval labels, and a font size for the x-axis of both subplots.
- Set limits, tick intervals, tick interval labels, and a font size for the y-axis of both subplots.
- Make sure the y-axis for each plot has the same range, tick intervals, and tick interval labels so that they are directly comparable.
- Add y-axis grid lines to both subplots.
- Rotate the boxplot x-axis labels (i.e., state names) to minimize overlap.

Deliverables

- Jupyter Notebook