CPSC 462: Homework #2 Due Tuesday, September 23rd

The goal of this assignment is to experiment with basic data visualization techniques using the `matplotlib` Python module. For this week there is no reading assignment.

Instructions: Perform the following steps and hand in your source code and the visualizations you generate for each step. Keep a log of any assumptions and/or issues you had in generating each chart. Include a cover sheet with your assignment and submit your program to the course submission page. All code used to generate the charts should be placed in a single Python program named `hw2.py` such that running the program should generate the various charts. Use the "preprocessed" `auto-data.txt` dataset you created from HW 1. Pick one method to deal with missing values for this assignment (e.g., eliminate rows with missing values, use means or medians, etc.). Each chart that you generate **must** include a figure title and labels on the x and y axes where appropriate (see the examples in Figure 1). Also, your final program should save each chart as a PDF file using the `savefig('filename.pdf')` function. Saved charts should start with the step name, e.g., `step-1-cylinders.pdf`.

Step 1 (*Categorical Data A*): Create a frequency diagram (sometimes informally referred to as a "histogram") for each of the categorical attributes of the `auto-data.txt` dataset (i.e., cylinders, model year, and origin). Each diagram should show the frequency (i.e., total number) of cars per value of the given attribute. Use a basic bar chart to draw your frequency diagrams. See Figure 1 for an example for the cylinders attribute.

Step 2 (*Categorical Data B*): Create a pie chart showing the frequency of cars for each of the categorical attributes of the `auto-data.txt` dataset. Your pie chart should include the percentages for each attribute value (using `autopct='%1.1f%%'`). See Figure 1 for an example for the cylinders attribute.

Step 3 (*Continuous Data A*): Create a dot (aka strip) chart showing the values for each of the continuous attributes (i.e., mpg, displacement, horsepower, weight, acceleration, and msrp). See Figure 1 for an example for mpg. As shown, darker circles indicate more data instances with that value. Some hints for creating a similar looking dot chart: set the y-axis values for each x value to 1; hide the y-axis using `pyplot.gca().get_yaxis().set_visible(False)`; use the `'.'` marker and set `markersize` to a larger default value and set `alpha=0.2` to make dots transparent.

Step 4 (*Continuous Data B*): There is often a need to transform a continuous attribute into a categorical attribute. Use the following two approaches to convert mpg into a categorical attribute and for each approach create a corresponding frequency diagram.

> *Approach 1.* The US Department of Energy assigns gasoline vehicles a fuel economy rating from 1 (worst) to 10 (best). The ratings are defined in terms of mpg as follows:
>
Rating	*MPG*
10	≥ 45
9	37–44
8	31–36
7	27–30
6	24–26

Rating	*MPG*
5	20–23
4	17–19
3	15–16
2	14
1	≤ 13

> Use these ranges to define category values (denoting rating 1 to 10) for the mpg attribute.
>
> *Approach 2.* Create 5 "equal-width" bins to generate categories. Each bin should divide up the range of mpg values into equal subranges, where value 1 denotes the smallest subrange of values and 5 the largest subrange of values (see Figure 1).

Each frequency diagram should label bins according to their corresponding ranges (e.g., `"27--30"`). See Figure 1 for an example.

Step 5 (*Continuous Data C*): Create a histogram using the `pyplot.hist` for each of the continuous attributes. Use the default of 10 bins (see Figure 1).

Step 6 (*Relationships A*): Create scatter plots that compare displacement, horsepower, weight, acceleration, and msrp to mpg (i.e., where mpg is the y-axis in each scatter plot). Be sure to appropriately label the x and y axes. Figure 1 gives an example for displacement.

Step 7 (*Relationships B*): Write a function to calculate (least-squares) linear regressions and create scatter plots with the corresponding linear regression lines for comparing displacement, horsepower, weight, and msrp to mpg. Create one additional scatter plot with a linear regression line comparing displacement to weight. Label each plot with the correlation coefficient and covariance. Figure 1 gives an example for displacement compared to mpg.

Step 8 (*Relationships C*): Create two charts to compare categorical/continuous and categorical/categorical attributes, respectively. For the first, create a box plot describing MPG (continuous) by year (categorical). For the second, create a frequency diagram of the number of cars from each country of origin (categorical) separated out by model year (categorical). Examples of both charts are shown in Figure 1.

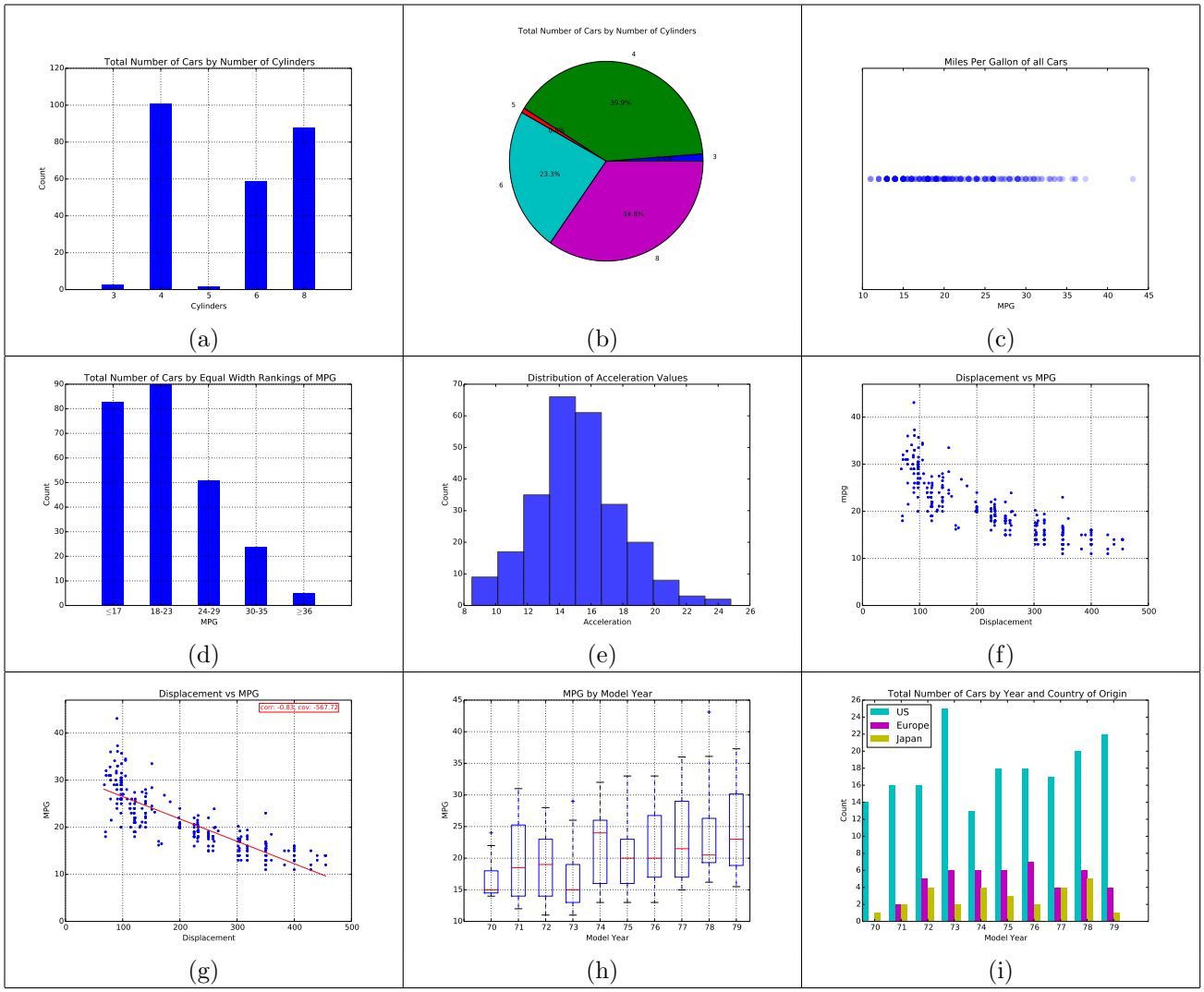

Figure 1: Example visualizations for each step: (a) frequency diagram; (b) pie chart; (c) dot chart; (d) frequency diagram of equal width binning; (e) histogram of acceleration values generated from `pyplot` with 10 bins; (f) scatterplot comparing displacement to mpg; (g) similar plot as in (f) but with linear regression line; (h) box and whisker plot; and (i) multiple frequency diagram.