1 Computing some basic statistics

Basic concepts are pretty straightforward. When computing a statistic from data, the term is “sample mean”, “sample variance”, etc. I have left off the “sample” for the same of simplicity.

- **Mean:** sum of the data values divided by the number of data values
  \[ \bar{x} = \frac{1}{N} \sum_{i=0}^{N-1} x_i \] (1)

- **Median:** data value with equal numbers of values above and below it
  \[ \text{median} = \text{sort}(\{x_0, \ldots, x_{N-1}\})[N/2] \] (2)

- **Mode:** data value with the highest frequency (usually applied to categorical or binned data)
  \[ \text{mode} = \max(\text{histogram}(\{x_0, \ldots, x_{N-1}\})) \] (3)

- **Standard Deviation:** a measure of the dispersion of the data as a distance from the mean
  \[ s_N = \sqrt{\frac{1}{N} \sum_{i=0}^{N-1} (x_i - \bar{x})^2} \] (4)

- **Variance:** the square of the standard deviation
  \[ v_N = \frac{1}{N} \sum_{i=0}^{N-1} (x_i - \bar{x})^2 \] (5)

When they fail to be meaningful is a different issue.

- Bi-modal or multi-modal distributions cause problems
- Missing data values can cause problems
2 Distributions inform visualization (Or, range selection and normalization)

How do we automatically select ranges for display? We have to pick ranges for many aspects of a visualization, including spatial extent, colors, sizes, or other methods of representing data values. Part of the purpose of interactive visualization is to avoid having to automatically select ranges, but even if the user has control over the spatial view, we may still have to select ranges for the color and size of the plotted points.

Visualization has several competing goals. A single visualization or series of visualizations has to find an appropriate balance.

- Correct characterization of the data
- Discriminability of relevant characteristics
- Stability and comparability across data sets

Below is a list of potential strategies for range selection. For each strategy, we also describe how one would normalize the data using that strategy.
1. Pick a range based on the min and max of the data.
   
   If we choose a simple property of the data such as max and min to set the range, then plots of similar kinds of data can look extremely different.
   
   • Outliers in the data control the range of colors, locations, sizes, etc.
   • Data with similar values in different data sets may end up with different visualization properties
   • Discriminability in the center of the data range may be impossible

   We would like to use invariant properties of the data space to select the range for visualization. Or even if they can’t be invariant, properties that are less susceptible to the above problems.

   **Normalization.** To normalize the data, we linearly transform it so that the min becomes 0 and the max becomes 1.

   **Example.** We have weights and heights for a population of males. If there are outliers (e.g. infants or giants), then we will not be able to see the trends for the average person as well (see Figure ??).

![Figure 1: The presence of outliers makes it harder to focus on the data we are interested in. The data set of the right includes measurements for an infant and a giant. The data set on the right includes a giant only. At first glance, it looks like the population has gotten smaller from one data set to the next. But really, we had a change in outliers. Also, it is a little hard to see the trends for the more average population.](image-url)
2. Pick a range based on the mean and standard deviation of the data.

\[ \sigma = \sqrt{\frac{1}{N} \sum_{i=0}^{N-1} (x_i - \mu)^2} \]

Data with a Gaussian distribution will be distributed so that most of the data (about 95%) is within 2 standard deviations of the mean. For such data sets, a range such as ±2 standard deviations will generally produce a consistent, useful visualization with reasonable discriminability.

Using standard deviation to set the range does not work as well on data that does not follow a Gaussian distribution or has a significant number of outliers. Outliers cause the standard deviation to expand, compressing the middle of the plot and reducing discriminability.

**Normalization.** To normalize the data, we linearly transform. We may also need to make it possible to omit data points outside the 0 .. 1 range form the display. Doing the linear transformation is a two-step process:

First, compute the z-statistic according to

\[ z_i = \frac{x_i - \mu}{\sigma} \]  

The transformation in (6) creates what is called a z-value in statistics. It represents how far away from the mean a particular value lies in units of the standard deviation of the distribution. The z-value, therefore, is invariant to the particular mean and standard deviation of the variable. It represents a statistically meaningful differential and indicates where in the distribution the particular value lies.

Second, pick a cutoff, such as ±2σ as the extent of the view volume. Note that this means that not all of the data is initially visible. Since the “units” are in σ, the additional transformation is straight-forward. Here it is for the range [−σ, σ]

\[ \hat{x}_i = \frac{z_i + 1}{2} \]  

**Example.** We continue looking at the heights/weights data. Looking within 2 standard deviations of data in both the x and y directions allows us to focus on the part of the data we are interested in. (see Figure ??).
Figure 2: Two standard deviations lets us look at most of our data. On the left is the original plot, with the area covered by 2 standard deviations outlined in magenta. On the right is the view restricted to that box.
3. Pick a range based on the mean and mean absolute deviation (or average absolute deviation).

\[ \text{MAD} = \frac{1}{N} \sum_{i=0}^{N-1} |x_i - \mu| \]

The mean absolute deviation is similar to the standard deviation, but it is less sensitive to outliers. For a Gaussian distribution, the MAD will be approximately 0.8 times the standard deviation.

**Normalization.** To normalize the data, we linearly transform. We may also need to make it possible to omit data points outside the 0 .. 1 range form the display.

**Example 2.** We continue looking at the heights/weights data. Looking within 2 MADs of data in both the x and y directions allows us to focus on the part of the data we are interested in. (see Figure ??). Note that although these data are normally distributed, they need not be.

![Figure 3: Two standard deviations lets us look at most of our data. On the left is the original plot, with the area covered by 2 standard deviations outlined in magenta and the area covered by two MADs outlined in red. On the right is the view restricted to the red box.](image)

4. Pick the range based on the potential data values.

For some data sets, the range of possible values is known. Selecting the range base on the set of possible values then enables consistent visualizations across different data sets with the same variables. If the actual data values cluster in one section of the range, this method of range selection will compress them, reducing discriminability of individual data points. However, if the intent is to compare the behavior of data sets and not individual points, this method enables consistent, comparable visualizations.

**Normalization.** To normalize the data, we linearly transform. We may also need to make it possible to omit data points outside the 0 .. 1 range form the display.

6
5. Pick the range based on potential values, but use bookend categories that capture outliers.

For example, pick a color range such as magenta to green. Let pure magenta represent any data value larger than an upper bound, and pure green any data value smaller than a lower bound. Pick the range between the upper and lower bound so that it covers most of the data.

We can achieve the same effect by using a sigmoid function to map data values to visualization values, such as colors. A sigmoid is a squashing function that takes the number line and squashes it to the range [0, 1], maintaining the monotonicity of the original number line. There are a number of functions used as sigmoids, but the most common is the logistic function. All sigmoid functions are s-shaped, mapping large negative numbers to values close to zero and large positive numbers to values close to one.

\[
S(x) = \frac{1}{1 + e^{-B(x-x_0)}}
\]  

(8)

The parameter \(x_0\) corresponds to the central point of the sigmoid curve. The parameter \(B\) controls the slope of the central part of the curve. Large \(B\) values produce steeper slopes, forcing the curve to be closer to a step function. Small values of \(B\) flatten the central slope, slowing the transition of the range from values close to 0 to values close to 1. The sigmoid is a generally useful function in data analysis, and we’ll explore it more fully later in the course.

Normalization. To normalize the data, we use whichever function we want to use. For the example, this means we use the sigmoid function to accomplish the normalization for us.

Example. We continue looking at the heights/weights data. Now we want to include the amount each person exercises using color. In Figure ?? we show the height/weight data (no outliers) and a histogram of the exercise data (totally fabricated and in arbitrary units). There are outliers. So when we map the minimum exercise value to yellow and the maximum to blue, we find most of the data is a hideous middling color and we don’t have much ability to make sense out of it (see Figure ??). If we simply bookend it between 8 and 32, we get much more discrimination (see Figure ??) and can conclude that the people who exercise more also weigh more for their height (this must be people working out in a weight-lifting gym!). We could run it through the logistic function, but it only makes sense if the curve isn’t too steep. When the curve is steep, more data points end up pure yellow or blue and we essentially split the population into two.
Figure 4: The data and how what we will use to color it. On the left are the heights and weights. We will color with exercise levels. On the right is a histogram of the exercise levels. There are outliers.

Figure 5: Color-coding linearly doesn’t make sense. The outliers are yellow and blue. Everyone else is icky and too similarly colored.

Figure 6: Bookending the data helps a lot. By making every value at or below 8 yellow and every value at or above 32 blue, we can now discriminate between the other data points. The bluer points are clearly lying “above” the yellow points.
Figure 7: If we use a sigmoid, we don’t want it to be that steep. On the left we use a sigmoid with $B=0.2$ and on the right we use a sigmoid with $B=1$. When $B=1$, the data become split into groups and there aren’t many in the middle.
6. Pick the range based on desired visualization outcomes.

For example, if we’re plotting temperature for a weather report, any temperature above 95F should appear as hot. Likewise, any temperature below 0F should appear as cold. These are subjective impressions, and in many situations it is not necessary to discriminate further. **Normalization.** To normalize the data, we linearly transform. We may also need to make it possible to omit data points outside the 0 .. 1 range form the display.

7. Logarithmic scales to capture measurements with high dynamic range.

Data with high dynamic range, such as sound amplitudes, challenge linear visualizations. As the amplitude of a lawnmower may be several orders of magnitude higher than the sound of a person talking, trying to plot both on the same linear chart compresses most typical sounds into a very small range. If we were to add the amplitude of a jet engine, the lawnmower would also end up compressed.

A logarithmic scale enables us to discriminate between different quiet sounds and different loud sounds, but it makes distances less meaningful. Equal distance in log space means equal ratio transitions. Therefore, the change from 1 to 2 is spatially the same as the change from 10 to 20 or 1000 to 2000.

**Normalization.** To normalize the data, we transform by putting the data into log scale.

The above concerns apply equally well to color ranges or spatial ranges in visualization.

In good plotting programs, the user can control all aspects of the visualization. There is no universal method of picking a range that will work for all visualizations. There are methods that work reasonably well over a wide range of situations. If you know your data set, you can probably develop an algorithm that will work effectively for the user’s purposes. Nevertheless, it is important to build flexibility into a visualization system and enable the user to tweak or adjust most range selection parameters.

### 2.0.1 Color

RGB: additive color scheme, used internally by computers to represent colors. While, as a programmer, we have to ultimately write pixel colors using RGB, the space is not always appropriate for color selection. In particular, distances in RGB space do not correspond to perceptual distances, as Figure ?? demonstrates.

There are two useful magnitude axes in RGB space. People with normal color vision perceive red and green as opposite colors and yellow and blue as opposite colors. We can use this to display magnitudes on a red-green color axis or yellow-blue color axis.

\[ C_{RG} = (\alpha, 1 - \alpha, 0.0) \]
Figure 8: Color distances. The color in the first column is equally different in RGB space from the colors in the second and third column

\[ C_{BY} = (1 - \alpha, 1 - \alpha, \alpha) \]

Interesting note: A common color spectrum for scientific data (e.g. the heat plots used to display microarray data) has been from red to black to green. But this is problematic for people with red-green color blindness (nearly 10% of the male population!). The field has been moving away from this, and in 2007, Nature Structure & Molecular Biology revised their style guide to ask that contrasts of red and green be avoided in graphs, models and schematics.

- Intensity is usually the average value of the color channels \( I = (R + G + B)/3 \).
- Intensity may also be a weighted average of the color channels \( Y = 0.299R + 0.587G + 0.114B \)

Acknowledgement: Most of the material in this lecture is taken from Bruce Maxwell’s spring 2012 CS251 notes.