3D Data Visualization

Oliver W. Layton

CS251: Data analysis and visualization

Lecture 6, Fall 2018

Monday February 18

Plan

- Review: vectorization
- Loose end: How do we normalize a data vector?
- Visualizing data in 3D
- Orthographic viewing pipeline
- Cross product

Broadcasting with Numpy

`matA` is a 10x5 matrix. `matAB` is a 10x3 matrix.

- Is doing this valid: `matA + 3`?
- Is doing this valid: `matA + matA`?
- Is doing this valid: `matA + matB`?
- Is doing this valid: `matA / matB`?
- (NEW) What operation is this? `matA.T * matB`.
  - What are the final dimensions?
- (NEW) Is doing this valid: `matA - np.mean(matA, axis=1)`?
- (NEW) Is doing this valid: `matA - np.mean(matB, axis=0)`?

How do we normalize a data vector?

Review: 2D data visualization workflow

- What was the process, starting with the data?
- Convert data -> view volume coordinates.
  - T1 matrix to move data to origin (corner of view volume).
- Convert view volume coordinates -> normalized view volume coordinates.
  - S1 matrix by view volume extent.
- Convert normalized view volume coordinates -> screen coordinates.
  - S2 matrix to screen window size. What happens to y?
  - T2 matrix by height to move origin to top-left corner.
- How do we do this in one step?
  - Chain matrix left multiplications to data vector (or matrix). What's the order?
  - $T_2 S_2 S_1 T_1 D$ (Later stage transformation matrices get appended to left hand side).

How do we adapt our 2D data visualization workflow to 3D?

Overall approach is similar, but some main differences are:

1. We need to do the 3D-to-2D projection (remember perspective vs. orthographic projections? Which did we say that we're doing?).

2. This is 3D not 2D. With the added homogeneous coordinate, how many rows and columns will our transformation matrices have?

Orthographic viewing pipeline

Because we chain together a bunch of matrix multiplications to do the viewing, we call the data-to-visualization process **orthographic viewing pipeline**.

- What does 'orthographic' mean again?

Orthographic projection

We do the 3D-to-2D projection when we convert normalized viewing volume coords -> screen coords:

$$\begin{bmatrix} x_{norm} \\ y_{norm} \\ z_{norm} \end{bmatrix} \rightarrow \begin{bmatrix} x_{norm} \\ y_{norm} \end{bmatrix} = \begin{bmatrix} \bar{x} \\ \bar{y} \end{bmatrix}$$

In other words, we simply drop the z coordinate! Easy!

Let's draw the 3D viewing setup

Goal (this week): Display 3D data volume

Goal (next week): Interactive control 3D view