Singular Value Decomposition is a method of generating a series of rank one matrices (matrices with only one linearly independent column) that sum to approximate a matrix. It will work with a matrix of any shape.

The first step is finding three specific matrices, U, Σ, and V that decompose the given matrix A such that

A = UΣVT

This is very similar to the A = PDP-1 decomposition, where P is the horizontal concatenation of the eigenvectors of A, and D is a diagonal matrix of the associated eigenvalues.

Since we cannot find the eigenvectors of a nonsquare matrix, we instead find the eigenvectors of two matrices that are derived from A and are square and symmetrical no matter A’s shape: AAT and ATA. If A is an MxN matrix, the sizes of these matrices will be NxN and MxM, respectively.

The nonzero eigenvalues of AAT and ATA are the same. Their associated eigenvectors ui and vi are related by ui = Avi. The columns of U, an NxN matrix, are the eigenvectors of AAT in order of decreasing associated eigenvalues, and the columns of V, an MxM matrix, are the associated eigenvectors of ATA. Rather than the eigenvalues, Σ is a diagonal matrix of the singular values of A, ‘filled in’ with zeros so it has size MxN. The singular value .

The best rank 1 approximation of A can then be calculated as u1v1T. The best rank 2 approximation is u1v1T +u2v2T , and so on.

HOW WE USED IT

We used a Python script to generate an adjacency matrix (such as the one below for English) showing how frequently one letter followed another in the words of a large sample text. This matrix is conveniently square.

[insert colorized adjacency matrix here]

Maybe explain plotting and leave this stuff for the bottom right corner?

We then found U, Σ, and V for the adjacency matrix. In a first-rank approximation, all the rows are scalar multiples of v1T and all the columns are scalar multiples of u1. Therefore, the ith element of u1 increases with the frequency that the ith letter in the alphabet precedes other letters, and the ith element of v1 increases with the frequency that the ith letter in the alphabet follows other letters. Plotting these two values on the Cartesian plane as (u1, v1) shows both the relative frequency of the letters, with more common letters landing further from the origin, and the frequency of those letters beginning or ending a word, with letters occurring mostly at the beginning of words landing further from the x-axis and letters occurring mostly at the end of words landing further from the y-axis.

Second-rank: ???