THE SINGULAR VALUE DECOMPOSITION AND ITS APPLICATIONS

SUMMER@ICERM 2020

1. Project Overview

The Singular Value Decomposition (SVD) is a popular matrix factorization with fascinating applications. Any matrix $A \in \mathbb{R}^{m\times n}$ can be factored into a SVD,

$$A = USV^T,$$

where $U \in \mathbb{R}^{m\times m}$ and $V \in \mathbb{R}^{n\times n}$ are orthogonal matrices, and $S \in \mathbb{R}^{m\times n}$ is a diagonal matrix. The diagonal entries $S(i,i)$, denoted by σ_i, satisfy $\sigma_1 \geq \sigma_2 \geq \cdot\cdot \sigma_p \geq 0$ where $p = \min(m,n)$. The σ_i's are called the singular values of A, and the number of positive singular values corresponds to the rank of A. The columns of U and V are called the left and right singular vectors respectively.

There is an interesting geometric interpretation of the SVD. Using u_i and v_j to denote the columns of U and V respectively, the SVD of a 2×2 matrix A can be viewed as in Figure 1.

Another way to write the SVD is as a sum of rank one matrices, i.e.,

$$A = \sum_{i=1}^{r} \sigma_i u_i v_i^T, \tag{1.1}$$

where r is the rank of A. (1.1) suggest a natural way to get a low rank approximation of A through the SVD. In other words, since $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r > 0$, the initial terms of the summation (1.1) are more "important" when approximating A. This simple idea explains why the approximation

$$A \approx \sum_{i=1}^{l} \sigma_i u_i v_i^T, \tag{1.2}$$

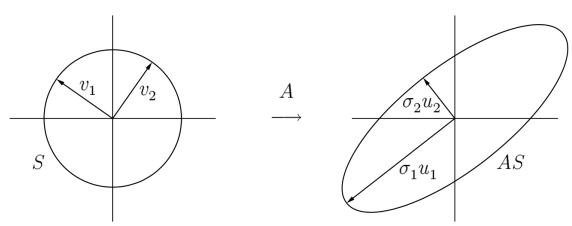

Figure 1. Geometrical Interpretation of the SVD

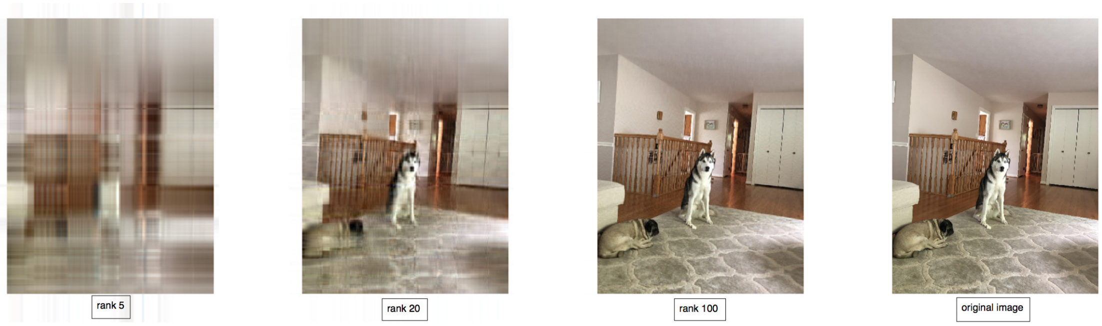

FIGURE 2. Low Rank Approximation of an Image

FIGURE 3. Comparison of the deterministic SVD and the randomized SVD

is often a very good approximation of A even when $l << r$. That is, A, can be well-approximated by a lower rank matrix by using the SVD, and this is in the heart of many applications of the SVD. One fun application of the SVD is in image and video compression. In figure 2, an image of 1510×2232 pixels is reconstructed by using (1.2). We stack the three color channels together to get a 4530×2232 matrix and get an approximation of the original image by using the first $l = 5, 20$, and 100 singular values and vectors. It is hard to detect any visual differences between the original image and the low rank approximation of the image for $l = 100$. This is a massive compression of data. When $l = 100$, we are only using approximately 10% of the memory compared to how the original image was saved, and this approximation faithfully reconstructs the original image.

The SVD indeed has numerous applications and beautiful mathematical properties. It is, however, computationally expensive to exactly compute the SVD of large matrices. Randomized linear algebra plays an essential role to effectively compute the SVD. In figure 3, a comparison between the (deterministic) SVD and the randomized SVD is given. Both figures are rank 200 approximations, and while the quality of the approximation is visually the same, the randomized SVD is much faster.

2. Project Outlook

In this project, we will explore various interesting applications of the SVD. Students will create their own applications where SVD may be useful. We will also investigate ways to improve existing background subtraction algorithms that are used to detect moving objects in a video as well. Additionally, we will consider ways to make our work accessible to the general public, which will give students a chance to practice presenting their work in a fun way that is easy to understand.

Matlab will be the main programming language that will be used in this project.

Suggested readings:

https://sites.math.washington.edu/~morrow/464_16/svd.pdf
https://pdfs.semanticscholar.org/d964/eea67155c2bc26bfc8c03dfc842181af9697.pdf
https://people.maths.ox.ac.uk/trefethen/lec4.ps