St. Francis Xavier University
Department of Computer Science

CSCI 554: Matrix Computation
Lecture 7: Singular Value Decomposition
Winter 2022

1 Composing Another Decomposition

In our previous lectures, we became familiar with the QR decomposition of a matrix and with the Gram–Schmidt process, and we found that these techniques were, in fact, equivalent. Both of these techniques can be used to solve the least squares problem, but we relied on the crucial assumption that our input matrix A had full rank. If we're given a matrix that is rank-deficient, or if we have a matrix whose rank we haven't determined, we can't use these methods to solve the least squares problem.

For this reason, we will now introduce and study a more powerful tool that allows us to solve the least squares problem for matrices without full rank, among other things. This tool is called the *singular value decomposition*, and it is one of the most fundamental matrix decompositions we will see in this course.

Now, even though we will introduce the singular value decomposition in this lecture, we won't be able to prove very many things about it at the moment, nor will we be able to develop an algorithm for computing the singular value decomposition of a matrix just yet. This is because the singular value decomposition relies on knowledge of eigenvalues and eigenvectors, which we will focus on in the coming lecture.

What we can do at this stage is define the singular value decomposition and study some applications of this decomposition to other aspects of linear algebra and computer science. For example, we will see how to connect the singular value decomposition to our previous notions of the norm and condition number of a matrix. We will also study how to employ the singular value decomposition as yet another method to solve the least squares problem. However, when it comes to all of the results *about* the singular value decomposition—as opposed to the results *using* the singular value decomposition—we will have to wait a little longer.

1.1 The SVD Theorem

Suppose $A \in \mathbb{R}^{n\times m}$ is a real-valued matrix, where $n, m \geq 1$. (We can disregard whichever of n and m is larger, as it makes no difference here.) Recall that the *rank* of A is the maximum number of linearly independent columns.

Traditionally, we define a diagonal matrix as a square matrix $A \in \mathbb{R}^{n\times n}$ whose only nonzero entries are located along the main diagonal; that is, at indices (i, i) for $1 \leq i \leq n$. We may generalize this notion to non-square matrices by defining a *rectangular diagonal matrix* to be a matrix $B \in \mathbb{R}^{n\times m}$ whose only nonzero entries are located at indices (i, i) for $1 \leq i \leq \min\{n, m\}$. For example, the following matrices are non-square but diagonal:

$$\begin{bmatrix} 2 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 8 \\ 0 & 0 & 0 \end{bmatrix} \quad \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 5 & 0 & 0 & 0 \end{bmatrix}$$

With these notions, we can define the singular value decomposition by way of the following theorem.

Theorem 1 (SVD theorem)**.** *Let $A \in \mathbb{R}^{n\times m}$ be a nonzero real-valued matrix where* $\operatorname{rank}(A) = r$*. Then A can be expressed as a matrix product $A = U\Sigma V^\intercal$, where $U \in \mathbb{R}^{n\times n}$ and $V \in \mathbb{R}^{m\times m}$ are orthogonal matrices,* *and* $\Sigma \in \mathbb{R}^{n \times m}$ *is a rectangular diagonal matrix of the form*

$$\Sigma = \begin{bmatrix} \sigma_1 & & & & & \\ & \sigma_2 & & & & \\ & & \ddots & & & \\ & & & \sigma_r & & \\ & & & & 0 & \\ & & & & & \ddots \end{bmatrix},$$

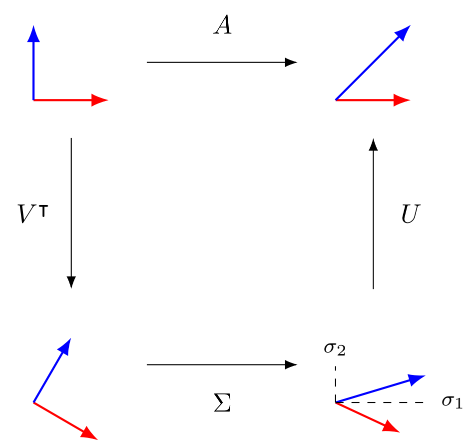
where $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r > 0$.

Proof. Not yet. □

We omit the proof of the theorem here because, again, it relies on a knowledge of eigenvalues and eigenvectors that we will gain in a later lecture. Instead, let's focus on the statement of the theorem itself.

The matrix product $A = U\Sigma V^\intercal$ is known as the singular value decomposition, hence the name of the theorem. In the matrix Σ, the nonzero entries σ_1 through σ_r are called the *singular values* of A. Moreover, the columns of U are the *left-singular vectors* of A, and the columns of V are the *right-singular vectors* of A. Both the columns of U and the columns of V form orthonormal bases.

In essence, the singular value decomposition breaks down some linear transformation specified by A into three components that, taken together, perform the same linear transformation. These three components are a rotation, a scaling, and another rotation. The singular value decomposition allows us to separate and disregard the rotation components, leaving us to focus only on the scaling component given by the matrix Σ, which operates with respect to the orthonormal bases given by $\{v_1, \ldots, v_m\}$ and $\{u_1, \ldots, u_n\}$. In the illustration below, Σ performs a horizontal scaling by a factor of σ_1 and a vertical scaling by a factor of σ_2.



We say that two matrices $A \in \mathbb{R}^{n \times m}$ and $B \in \mathbb{R}^{n \times m}$ are *orthogonally equivalent* if there exist invertible matrices $X \in \mathbb{R}^{n \times n}$ and $Y \in \mathbb{R}^{m \times m}$ such that $A = XBY$. From this perspective, we can see that the SVD theorem tells us that A is orthogonally equivalent to the rectangular diagonal "scaling" matrix Σ.

We can alternatively frame the SVD theorem from this geometric perspective, using what we know about the decomposition $A = U\Sigma V^\intercal$.

Theorem 2 (Geometric SVD theorem)**.** *Let* $A \in \mathbb{R}^{n \times m}$ *be a nonzero real-valued matrix where* $\operatorname{rank}(A) = r$. *Then* $\mathbb{R}^m$ *has an orthonormal basis* $\{v_1, \ldots, v_m\}$, $\mathbb{R}^n$ *has an orthonormal basis* $\{u_1, \ldots, u_n\}$, *and there exist values* $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r > 0$ *such that*

$$Av_i = \begin{cases} \sigma_i u_i & \textit{for } 1 \leq i \leq r; \\ 0 & \textit{for } (r+1) \leq i \leq m, \end{cases} \quad \textit{and } A^\intercal u_i = \begin{cases} \sigma_i v_i & \textit{for } 1 \leq i \leq r; \\ 0 & \textit{for } (r+1) \leq i \leq n. \end{cases}$$

2 Applications of the Singular Value Decomposition

As we mentioned in the introduction to this lecture, we will be focusing less on the singular value decomposition itself for the moment, instead directing our focus on applications of this decomposition. In this section, we review a few applications as we prepare (once again) to discuss the least squares problem in the following section.

2.1 Computing Norms and Condition Numbers

Recall our definition of the 2-norm (or Euclidean norm) of a matrix A:

$$\|A\|_2 = \max_{x \neq 0} \frac{\|Ax\|_2}{\|x\|_2}.$$

As we observed when we first studied norms, an induced matrix norm like the 2-norm corresponds to the maximum magnification that takes place during the mapping from vectors $x \in \mathbb{R}^n$ to vectors $Ax \in \mathbb{R}^n$.

Although we initially defined matrix norms to apply only to square matrices, there's no real reason why we need to abide by such a restriction. We can just as easily use norms to measure the maximum magnification induced by a non-square matrix.

From what we know so far about the singular value decomposition, the entries in the matrix Σ each perform a scaling transformation in some dimension. Since we took the entries σ_i of Σ to be sorted from greatest to least, we can easily obtain the greatest scaling factor by simply looking at σ_1, and this greatest scaling factor corresponds to the maximum magnification value of A.

Theorem 3. *Let $A \in \mathbb{R}^{n \times m}$ be a matrix with singular values $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r > 0$. Then $\|A\|_2 = \sigma_1$.*

Proof. Recall from the statement of the geometric SVD theorem that $Av_1 = \sigma_1 u_1$. Then, taking $x = v_1$, we have that

$$\frac{\|Av_1\|_2}{\|v_1\|_2} = \sigma_1 \frac{\|u_1\|_2}{\|v_1\|_2} = \sigma_1,$$

and this tells us that $\|A\|_2 = \max_{x \neq 0} \|Ax\|_2 / \|x\|_2 \geq \sigma_1$.

Now, we must establish equality by showing that no other vector x is magnified by a factor greater than σ_1. Let $x \in \mathbb{R}^m$. We can express x as a linear combination of the right-singular vectors of A:

$$x = c_1 v_1 + c_2 v_2 + \cdots + c_m v_m.$$

Since we know that each vector v_i is orthonormal, we have that $\|x\|_2^2 = |c_1|^2 + \cdots + |c_m|^2$.

Now, multiply the previous linear combination by A to get

$$\begin{aligned} Ax &= c_1 Av_1 + c_2 Av_2 + \cdots + c_m Av_m \\ &= c_1 \sigma_1 u_1 + c_2 \sigma_2 u_2 + \cdots + c_r \sigma_r u_r + 0 + \cdots + 0, \end{aligned}$$

where r is the rank of A. Similarly, since we know that each vector u_i is orthonormal, we have that $\|Ax\|_2^2 = |c_1 \sigma_1|^2 + \cdots + |c_r \sigma_r|^2$.

We can obtain the following upper bound on this expression by taking $\sigma_i = \sigma_1$ for all i:

$$\begin{aligned} \|Ax\|_2^2 &\leq \sigma_1^2 \left(|c_1|^2 + \cdots + |c_r|^2 \right) \\ &\leq \sigma_1^2 \|x\|_2^2, \end{aligned}$$

which implies that $\|Ax\|_2 / \|x\|_2 \leq \sigma_1$, establishing equality as desired. $\square$

Moving on to condition numbers, recall our definition of the condition number of an invertible square matrix $A \in \mathbb{R}^{n \times n}$ under the 2-norm:

$$\kappa_2(A) = \|A\|_2 \, \|A^{-1}\|_2.$$

As we had with the 2-norm itself, it would be nice to have a characterization of the condition number in terms of the singular value decomposition.

Since A is of dimension $n \times n$ and invertible, we know that $\text{rank}(A) = n$. Therefore, A has n singular values σ_i that are each positive, and each of these values σ_i acts as a mapping from v_i to u_i. This implies that the inverse values σ_i^{-1} each act as a mapping from u_i back to v_i. In matrix form, this means that both $A = U\Sigma V^\intercal$ and $A^{-1} = V^{-\intercal}\Sigma^{-1}U^{-1} = V\Sigma^{-1}U^\intercal$.

Just as we know that $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n > 0$, the previous observations tell us that $\sigma_n^{-1} \geq \sigma_{n-1}^{-1} \geq \cdots \geq \sigma_1^{-1} > 0$. Then, by Theorem 3, we know that $\|A^{-1}\|_2 = \sigma_n^{-1}$, and this leads to the following result.

Theorem 4. *Let $A \in \mathbb{R}^{n \times n}$ be an invertible matrix with singular values $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n > 0$. Then*

$$\kappa_2(A) = \frac{\sigma_1}{\sigma_n}.$$

Proof. Follows from an application of Theorem 3 to both A and A^{-1}. □

2.2 Computing the Pseudoinverse

Recall that, given a square matrix $A \in \mathbb{R}^{n \times n}$, A has an inverse if and only if there exists a matrix A^{-1} such that $AA^{-1} = A^{-1}A = I_n$, where I_n is the $n \times n$ identity matrix.

Inverse matrices are only well-defined for invertible square matrices A. It is possible, however, to generalize the notion of an inverse to work for non-square matrices. This generalization, known as the *pseudoinverse* or the *Moore–Penrose inverse*, works for all matrices $A \in \mathbb{R}^{n \times m}$.

Definition 5 (Pseudoinverse)**.** Given a matrix $A \in \mathbb{R}^{n \times m}$, the pseudoinverse of A, denoted A^+, is the unique matrix of dimension $m \times n$ that satisfies each of the following four conditions:

1. $AA^+A = A$;
2. $A^+AA^+ = A^+$;
3. $(AA^+)^\intercal = AA^+$; and
4. $(A^+A)^\intercal = A^+A$.

The four conditions given in the definition are sometimes referred to as the Moore–Penrose conditions, and they collectively ensure that a pseudoinverse "behaves like" any other inverse matrix.

Just as we're able to express the solution to a system of linear equations $Ax = b$ involving a square coefficient matrix A by taking $x = A^{-1}b$, we can express the solution to a general system of linear equations $Ax = b$ with $A \in \mathbb{R}^{n \times m}$ using the pseudoinverse. In this case, we would have $x = A^+b$. Thus, the pseudoinverse finds an application in solving the least squares problem.

How do we compute the pseudoinverse, though? Well, to maintain the property that the pseudoinverse should behave like any other inverse, we should require that

$$A^+u_i = \begin{cases} \sigma_i^{-1}v_i & \text{for } 1 \leq i \leq r; \text{ and} \\ 0 & \text{for } (r+1) \leq i \leq n. \end{cases}$$

This expression can be written as a matrix product $A^+U = \Sigma^+V$, where

$$\Sigma^+ = \begin{bmatrix} \widehat{\Sigma}^{-1} & 0 \\ 0 & 0 \end{bmatrix} \in \mathbb{R}^{m \times n}$$

and $\widehat{\Sigma}$ is the $r \times r$ matrix with entries σ_1 to σ_r along its main diagonal. Then, rearranging this matrix product, we get that $A^+ = V\Sigma^+U^\intercal$. Therefore, we can compute the pseudoinverse using the singular value decomposition of A!

With the pseudoinverse, we can then draw a connection to the least squares problem.

Proposition 6. *Let $A \in \mathbb{R}^{n\times m}$ be a matrix and let $b \in \mathbb{R}^n$ be a vector. Furthermore, let $x \in \mathbb{R}^m$ be the solution to the instance of the least squares problem*

$$\|b - Ax\|_2 = \min_{w\in\mathbb{R}^m} \|b - Aw\|_2.$$

Then we have that $x = A^+b$.

2.3 Computing the Numerical Rank

As we saw in the previous lecture, the rank of a matrix A is the maximum number of linearly independent columns of A. On paper, it's easy for us to determine the rank of a matrix; for example, consider the matrix

$$A = \begin{bmatrix} 1/3 & 1/3 & 2/3 \\ 1/3 & 2/3 & 3/3 \\ 2/3 & 2/3 & 4/3 \end{bmatrix}.$$

We can clearly see that $\text{rank}(A) = 2$, since the third column of A is a linear combination of the first and second columns. However, if we represent this matrix in a computer's memory, the computer suffers from rounding errors as a result of how it stores these fractional values. As a consequence, if we plug this matrix into computer software and compute the singular value decomposition, the software will tell us that *three* nonzero singular values exist, thus implying that the rank of A is 3. This is a consequence of the computer failing to recognize that $1/3 + 1/3 = 2/3$, for example, because it instead calculates $0.333 + 0.333 \neq 0.667$.

Interestingly, if we use the same computer software to find the rank of A using the software's built-in rank method, it gives us the correct answer of 2. This is because the built-in method uses a different approach to finding the rank of a matrix—it finds something known as the *numerical rank*. How does the numerical rank differ from our attempt to find the rank directly from the singular value decomposition? Let's reveal what the computer software gives us in terms of singular values for A:

$$\sigma_1 = 2.2024,\ \sigma_2 = 0.19539,\ \text{and } \sigma_3 = 5.9693 \times 10^{-17}.$$

One of these values is not like the others! The singular value σ_3 is extremely small compared to the other two singular values, and the fact that σ_3—despite being extremely small—is nonzero is what leads to the software erroneously claiming that the rank of A is 3.

The numerical rank of a matrix A takes into account not just the singular values themselves, but also a specified threshold value ϵ. This threshold value can be defined in any number of ways, but it's usually defined in terms of the machine precision, the roundoff error amount, or some similar computer limitation.

Definition 7 (Numerical rank)**.** Given a matrix A, we say that A has a numerical rank of k if the singular value decomposition of A produces k singular values that are much greater than the threshold value ϵ; that is,

$$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_k \gg \epsilon \geq \sigma_{k+1} \geq \cdots \geq \sigma_r > 0.$$

While small perturbations or errors in the data stored in a matrix can affect the result when computing the rank, the numeric rank is capable of tolerating this error while still producing the correct answer. Thus, the numeric rank gives us a reliable metric for measuring the rank of a matrix, provided we use an appropriate threshold value.

3 Singular Value Decomposition and Least Squares

Let's now wrap up our introduction to the singular value decomposition by tying it to our very familiar least squares problem. Suppose that $A \in \mathbb{R}^{n\times m}$ is a matrix where $\operatorname{rank}(A) = r$, and let $b \in \mathbb{R}^n$. Consider the system of linear equations $Ax = b$. Recall that if $n > m$, we say that this linear system is overdetermined, and in this case we must solve the least squares problem to find a solution vector x that is as close to the exact solution as possible.

Previously, we saw that if $n \geq m$ and $\operatorname{rank}(A) = m$, then there is a unique solution to the least squares problem. However, in the case where $\operatorname{rank}(A) < m$, there may be many solutions x that minimize the residual $\|b - Ax\|_2$. Thus, we will focus our efforts on finding the *smallest* vector $x \in \mathbb{R}^m$ minimizing $\|b - Ax\|_2$. By "smallest", we mean the vector x having the smallest 2-norm. Fortunately for us, we are always able to find a unique solution in this specialized case.

Suppose we have computed the singular value decomposition $A = U\Sigma V^\intercal$, where $U \in \mathbb{R}^{n\times n}$ and $V \in \mathbb{R}^{m\times m}$ are orthogonal matrices and

$$\Sigma = \begin{bmatrix} \widehat{\Sigma} & 0 \\ 0 & 0 \end{bmatrix} \in \mathbb{R}^{n\times m}$$

is such that $\widehat{\Sigma}$ is the $r \times r$ matrix with entries σ_1 to σ_r along its main diagonal, where $\sigma_1 \geq \cdots \geq \sigma_r > 0$.

Since U is orthogonal, we know that

$$\begin{aligned} \|b - Ax\|_2 &= \|U^\intercal(b - Ax)\|_2 \\ &= \|U^\intercal b - \Sigma(V^\intercal x)\|_2. \end{aligned}$$

Taking $c = U^\intercal b$ and $y = V^\intercal x$ then squaring, we obtain the expression

$$\begin{aligned} \|b - Ax\|_2^2 &= \|c - \Sigma y\|_2^2 \\ &= \sum_{i=1}^{r} |c_i - \sigma_i y_i|^2 + \sum_{i=r+1}^{n} |c_i|^2. \end{aligned}$$

We can see that this expression is minimized when $y_i = c_i/\sigma_i$, as the first component of the term gets zeroed out. We also need only to focus on values y_1 through y_r when $r < m$, since in this case the values y_{r+1} through y_m have no effect. Thus, we can take these values to be arbitrary, and indeed, $\|y\|_2$ is minimized when y_{r+1} through y_m are all zero. Then, since $x = Vy$ and V is an orthogonal matrix, we know that $\|x\|_2 = \|y\|_2$, and $\|x\|_2$ is likewise minimized—exactly what we're trying to solve!

The procedure we just described gives us a method of solving the least squares problem using the singular value decomposition of A:

Algorithm 1: Least squares—SVD

$c \leftarrow \begin{bmatrix} \widehat{c} \\ d \end{bmatrix}$ ▷ $c = U^\intercal b$

$\widehat{y} \leftarrow \widehat{\Sigma}^{-1}\widehat{c}$ ▷ y_i is minimized when $y_i = c_i/\Sigma_i$

if $r < m$ **then**

  choose $z \in \mathbb{R}^{m-r}$ arbitrarily

$y \leftarrow \begin{bmatrix} \widehat{y} \\ z \end{bmatrix}$

$x \leftarrow Vy$

Naturally, we're not able to implement this algorithm just yet, since we don't yet know how to compute the singular value decomposition of A. However, if we're given the singular value decomposition in advance, then we can use U, Σ, and V to solve the least squares problem in just a few steps.

One additional benefit with this approach comes in handling the information necessary to solve this problem. Since we need to determine the deficient rank of A, we can additionally use the singular value decomposition to compute the numerical rank using the approach we studied in the previous section.