The Singular Value Decomposition

Jeremy Teitelbaum

The singular value decomposition (SVD)

The SVD is a way to study rectangular matrices using tools that come from our work with symmetric matrices.

It doesn't make direct sense to diagonalize a rectangular matrix, but in some sense the SVD is the closest we can come.

It is a widely used result in applied mathematics.

Singular Values

Let A by an $m \times n$ matrix. The *singular values* σ_i of A are the (positive) square roots of the eigenvalues of the $n \times n$ symmetric matrix $A^T A$

$$\sigma_i = \sqrt{\lambda_i}$$

Remember that, by the spectral theorem, $A^T A$ has real, nonnegative eigenvalues, so these square roots make sense.

We arrange the singular values in decreasing order so that

$$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n \geq 0$$

Singular values

If $v_1, \ldots, v_n$ are the unit eigenvectors of $A^T A$, then

$$\|Av_i\|^2 = (Av_i) \cdot (Av_i) = v_i^T A^T A v_i = \lambda_i \|v_i\|^2$$

so the singular values σ_i measure the amount that A “stretches” v_i.

Nonzero singular values give rank

Some of the singular values σ_i of A and corresponding eigenvalues λ_i of $A^T A$ could be zero.

If λ_k is zero, then

$$Av_k \cdot Av_k = v_k^T A^T A v_k = \lambda_k (v_k \cdot v_k) = 0$$

so $Av_k = 0$.

Suppose that the first r of them are non zero. Then, if v_i are the corresponding eigenvectors of $A^T A$, the vectors

$$Av_1, \dots , Av_r$$

form an orthogonal basis for the column space $\mathrm{Col}(A)$, and A has rank r.

Nonzero singular values give rank (continued)

To see that they are orthogonal, compute

$$Av_i \cdot Av_j = v_i^T A^T A v_j = \lambda_j v_i^T v_j = 0$$

since the v_i are orthogonal. The Av_i also all belong to the column space of A.

Suppose that y is any vector in the column space of A. Then $y = Ax$ for some x, and

$$x = \sum_{i=1}^{n} (x \cdot v_i) v_i.$$

Apply A to this and since $Av_k = 0$ for $k > r$, we see that Ax is in the span of $Av_1, \dots, Av_r$.

So $Av_1, \dots, Av_r$ are orthogonal (hence linearly independent) and span the column space of A.

The SVD

Suppose that A is an $m \times n$ matrix of rank r. Then there exists an $m \times n$ matrix Σ which is "diagonal" in the sense that it looks like this:

$$\begin{array}{ll} \Sigma = \left[\begin{array}{cc} D & 0 \\ 0 & 0 \end{array}\right] & \begin{array}{l} \\ \leftarrow m - r \text{ rows} \end{array} \\ \qquad\qquad\quad \uparrow\!\lfloor\; n - r \text{ columns} & \end{array}$$

Figure 1: "Diagonal" Matrix for SVD

where D is a truly diagonal $r \times r$ matrix whose entries are the nonzero singular values of A (in descending order), and orthogonal matrices U of size $m \times m$ and V of size $n \times n$ such that

$$A = U \Sigma V^T.$$

Constructing the SVD

1. Let $u_i = \frac{Av_i}{\|Av_i\|} = \sigma_i^{-1} A v_i$ for $i = 1, \ldots, r$. This gives an orthonormal family. Extend this to an orthonormal basis $u_1 \ldots, u_m$ of $\mathbf{R}^m$.
2. Let U be the matrix whose columns are the u_i and V be the matrix whose columns are the v_i.
3. Notice that AV has columns $\sigma_i u_i$ for $i = 1, \ldots, r$ and the rest zero. That's what you get if you compute $U\Sigma$.
4. So $AV = U\Sigma$ or $A = U\Sigma V^{-1} = U\Sigma V^T$.

Terminology

Let $A = U\Sigma V^T$ be a singular value decomposition of A.

The *columns* of U are called the *left singular vectors* of A.

The *columns* of V are called the *right singular vectors* of A.

Numerical Example

```python
import numpy as np
from sympy import latex, Matrix
from IPython.display import Latex

A = np.array([[1, 3, 2], [2, 5, 6]])
# note that the routine returns "V"
# but we would call it "V^T"
U, Sigma, V = np.linalg.svd(A, full_matrices=True)
```

Results

Matrix $A = \begin{bmatrix} 1 & 3 & 2 \\ 2 & 5 & 6 \end{bmatrix}$

$$U = \begin{bmatrix} -0.41 & 0.91 \\ -0.91 & -0.41 \end{bmatrix}$$

$$\Sigma = \begin{bmatrix} 8.84 & 0 & 0 \\ 0 & 0.94 & 0 \end{bmatrix}$$

$$V^T = \begin{bmatrix} -0.25 & -0.66 & -0.71 \\ 0.09 & 0.72 & -0.69 \\ -0.96 & 0.24 & 0.12 \end{bmatrix}$$

Four Fundamental Subspaces

Let A be an $m \times n$ matrix with left singular vectors $u_1, \ldots, u_m$, right singular vectors $v_1, \ldots, v_n$, singular values $\sigma_1, \ldots, \sigma_n$, and rank r.

1. $u_1, \ldots, u_r$ form an orthonormal basis for the column space of A. Remember that the u_i are normalized versions of Av_i where v_i are the right singular vectors. The Av_i for $i = 1, \ldots, r$ span the column space of A.
2. $u_{r+1}, \ldots, r_n$ form an orthonormal basis for the null space of A^T (which is the same as the $Col(A)^{\perp}$.)

Four subspaces continued

3. $v_{r+1}, \ldots, v_n$ form an orthonormal basis for the null space of A. This is because $Av_j = 0$ for $j = r+1, \ldots, n$, they are independent (because orthonormal), and the rank of A is r so the dimension of the null space is $n - r$.

4. $v_1, \ldots, v_r$ form an orthonormal basis for the row space of A. This is because they are an orthonormal basis for $Null(A)^{\perp}$ which is the same as $Col(A^T)$ which is $Row(A)$.

The Pseudoinverse

One can use the SVD to solve linear systems and to "approximate" the inverse of matrices that aren't square.

Suppose that A has rank r, so that Σ has r nonzero entries. Let Σ_r be the square $r \times r$ matrix with the nonzero signular values.

Split up U and V so that U_r consists of only the first r columns of U, and V_r consists of only the first r columns of V. Then U is $m \times r$ and V is $n \times r$.

Then $A = U_r \Sigma_r V_r^T$. The "pseudoinverse" of A is

$$V_r \Sigma_r^{-1} U_r^T$$

.

Application of the pseudoinverse

Consider the matrix equation $Ax = b$ (here we don't assume A is square).

If we set $\hat{x} = A^{+}b$ where A^{+} is the pseudoinverse, then $\hat{x}$ is the vector so that $A\hat{x}$ is as close to b as possible.

If A is invertible, or if b is in the column space of A, then $\hat{x}$ is an exact solution.

Example

```python
import numpy as np
from sympy import latex, Matrix
from IPython.display import Latex

A = np.array([[1, 3, 2], [2, 5, 6]])
# note that the routine returns "V"
# but we would call it "V^T"
U, Sigma, V = np.linalg.svd(A, full_matrices=True)
Sigmar=np.diag(Sigma)
Vt=V.transpose()
Ap = Vt[:,:2]@np.linalg.inv(np.diag(Sigma))@U.transpose()
display(Latex("$A^{+}=" + f"{latex(Matrix(np.round(Ap,2)))}
```

$$A^{+} = \begin{bmatrix} 0.1 & -0.01 \\ 0.72 & -0.25 \\ -0.64 & 0.38 \end{bmatrix}$$

Example continued.

Consider the equation

$$Ax = \begin{bmatrix} 6 \\ 13 \end{bmatrix}$$

It has infinitely many solutions. If we use the pseudo inverse, we find one of them:

$$A^{+} \begin{bmatrix} 6 \\ 13 \end{bmatrix} = \hat{x}$$

$$\hat{x} = \begin{bmatrix} 0.42 \\ 1.145 \\ 1.072 \end{bmatrix}$$

This turns out to be the solution with minimal norm (that is, the shortest vector that solves the linear system).