Singular value decomposition (SVD)

A general $m \times n$ matrix $\boldsymbol{A}$ (m rows, n columns) can always be represented as

$$\boldsymbol{A} = \boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^T$$

$\boldsymbol{U}$ is $m \times m$ and orthogonal, $\boldsymbol{V}$ is $n \times n$ and orthogonal, $\boldsymbol{D}$ is $m \times n$ and diagonal:

$$\boldsymbol{D} = \begin{pmatrix} \sigma_1 & 0 & 0 & \dots & 0 \\ 0 & \sigma_2 & 0 & \dots & 0 \\ 0 & 0 & \sigma_3 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & \sigma_n \\ 0 & 0 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & 0 \end{pmatrix} \text{ or } \boldsymbol{D} = \begin{pmatrix} \sigma_1 & 0 & 0 & \dots & 0 & 0 & \dots & 0 \\ 0 & \sigma_2 & 0 & \dots & 0 & 0 & \dots & 0 \\ 0 & 0 & \sigma_3 & \dots & 0 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & \sigma_m & 0 & \dots & 0 \end{pmatrix}$$

Note this differs from the definition in the last lecture, where we had $m > n$, $\boldsymbol{U}$ was $m \times n$ and $\boldsymbol{D}$ was $n \times m$. But the previous definition (called "thin" SVD) is equivalent in the sense that when the last m-n rows of $\boldsymbol{D}$ are 0's, it does not matter what the last m-n columns of $\boldsymbol{U}$ contain.

σ_i are called the singular values, columns of U are left-singular vectors and those of V are right-singular vectors.

$$A = UDV^T$$

What is the meaning of SVD? AA^T is a symmetric square $(m \times m)$ matrix.

$$AA^T = (UDV^T)(UDV^T)^T = UDV^T VD^T U^T = UDD^T U^T$$

$$U^T AA^T U = DD^T$$

$m>n$

$$DD^T = \begin{pmatrix} \sigma_1 & 0 & 0 & \dots & 0 \\ 0 & \sigma_2 & 0 & \dots & 0 \\ 0 & 0 & \sigma_3 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & \sigma_n \\ 0 & 0 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & 0 \end{pmatrix} \begin{pmatrix} \sigma_1 & 0 & 0 & \dots & 0 & 0 & \dots & 0 \\ 0 & \sigma_2 & 0 & \dots & 0 & 0 & \dots & 0 \\ 0 & 0 & \sigma_3 & \dots & 0 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & \sigma_n & 0 & \dots & 0 \end{pmatrix} = \begin{pmatrix} \sigma_1^2 & 0 & 0 & \dots & 0 & 0 & \dots & 0 \\ 0 & \sigma_2^2 & 0 & \dots & 0 & 0 & \dots & 0 \\ 0 & 0 & \sigma_3^2 & \dots & 0 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & \sigma_n^2 & 0 & \dots & 0 \\ 0 & 0 & 0 & \dots & 0 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & 0 & 0 & \dots & 0 \end{pmatrix}$$

$m<n$

$$DD^T = \begin{pmatrix} \sigma_1 & 0 & 0 & \dots & 0 & 0 & \dots & 0 \\ 0 & \sigma_2 & 0 & \dots & 0 & 0 & \dots & 0 \\ 0 & 0 & \sigma_3 & \dots & 0 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & \sigma_m & 0 & \dots & 0 \end{pmatrix} \begin{pmatrix} \sigma_1 & 0 & 0 & \dots & 0 \\ 0 & \sigma_2 & 0 & \dots & 0 \\ 0 & 0 & \sigma_3 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & \sigma_m \\ 0 & 0 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & 0 \end{pmatrix} = \begin{pmatrix} \sigma_1^2 & 0 & 0 & \dots & 0 \\ 0 & \sigma_2^2 & 0 & \dots & 0 \\ 0 & 0 & \sigma_3^2 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & \sigma_m^2 \end{pmatrix}$$

$$A = UDV^T$$
$$U^T AA^T U = DD^T$$

So ***U*** is the matrix of the eigenvectors of $\boldsymbol{AA^T}$, and ***D*** contains the square roots of the corresponding eigenvalues of $\boldsymbol{AA^T}$.

$\boldsymbol{A^TA}$ is a symmetric square ($n\times n$) matrix.

$$A^T A = (UDV^T)^T (UDV^T) = VD^T U^T UDV^T = VD^T DV^T$$
$$V^T A^T AV = D^T D$$

So ***V*** is the matrix of the eigenvectors of $\boldsymbol{A^TA}$, and ***D*** contains the square roots of the corresponding eigenvalues of $\boldsymbol{A^TA}$. (same as those of $\boldsymbol{AA^T}$, except for $n>m$ extra eigenvalues of $\boldsymbol{A^TA}$ are 0, and for $m>n$ extra eigenvalues of $\boldsymbol{AA^T}$ are 0).

If A is square and symmetric, then $\boldsymbol{AA^T}=\boldsymbol{A^TA} = \boldsymbol{A}^2$, the eigenvectors are the same as those of ***A*** and the eigenvalues are squares of those of ***A***. So up to a sign the singular values are the eigenvalues of ***A*** and the columns of ***U*** and ***V*** are the eigenvectors. ***U*** and ***V*** are identical or can be chosen identical.

$$A = UDV^T$$

For a general matrix ***A***, $\quad AV = UDVV^T = UD.$

$$U = (\vec{u}_{1,} \vec{u}_{2,} .., \vec{u}_m) \qquad D = \begin{pmatrix} \sigma_1 & 0 & 0 & ... & 0 \\ 0 & \sigma_2 & 0 & ... & 0 \\ 0 & 0 & \sigma_3 & ... & 0 \\ ... & ... & ... & ... & ... \\ 0 & 0 & 0 & ... & \sigma_n \\ 0 & 0 & 0 & ... & 0 \\ ... & ... & ... & ... & ... \\ 0 & 0 & 0 & ... & 0 \end{pmatrix} \text{ or } D = \begin{pmatrix} \sigma_1 & 0 & 0 & ... & 0 & 0 & ... & 0 \\ 0 & \sigma_2 & 0 & ... & 0 & 0 & ... & 0 \\ 0 & 0 & \sigma_3 & ... & 0 & 0 & ... & 0 \\ ... & ... & ... & ... & ... & ... & ... & ... \\ 0 & 0 & 0 & ... & \sigma_m & 0 & ... & 0 \end{pmatrix}$$

$$A \cdot (\vec{v}_{1,} \vec{v}_{2,} .., \vec{v}_n) = \begin{cases} (\sigma_1 \vec{u}_{1,} \sigma_2 \vec{u}_{2,} .., \sigma_n \vec{u}_n), & m \geq n, \\ (\sigma_1 \vec{u}_{1,} \sigma_2 \vec{u}_{2,} .., \sigma_m \vec{u}_m, 0, ..., 0), & m < n. \end{cases}$$

$$A\vec{v}_i = \sigma_i \vec{u}_i$$

$$U^T A = U^T UDV^T = DV^T$$

$$A^T U = V D^T$$

$$A^T \cdot (\vec{u}_{1,} \vec{u}_{2,} .., \vec{u}_m) = \begin{cases} (\sigma_1 \vec{u}_{1,} \sigma_2 \vec{u}_{2,} .., \sigma_m \vec{u}_n, 0, ..., 0), & m \geq n, \\ (\sigma_1 \vec{u}_{1,} \sigma_2 \vec{u}_{2,} .., \sigma_n \vec{u}_m), & m < n. \end{cases}$$

$$A^T \vec{u}_i = \sigma_i \vec{v}_i$$

$$A = UDV^T \qquad\qquad A\vec{v}_i = \sigma_i \vec{u}_i$$

For those i for which σ_i=0, $A\vec{v}_i = \vec{0}$. So such $\vec{v}_i$ span the nullspace of ***A***.

Nullspace (or kernel) is spanned by the columns of ***V*** with corresponding σ=0.

If we take all $\vec{v}_i,\ i=1,\ldots,n,$ they form a full basis, so an arbitrary vector $\vec{x}$ can be represented as their linear combination: $\vec{x} = \sum_{i=1}^{n} c_i \vec{v}_i$.

$$A\vec{x} = A\left(\sum_{i=1}^{n} c_i \vec{v}_i\right) = \sum_{i=1}^{n} c_i \sigma_i \vec{u}_i.$$

Depending on c_i, any linear combination of $\vec{u}_i$ with $\sigma_i \neq 0$ can be obtained, and no vectors that cannot be represented as such a combination are obtainable. Thus, the range of ***A*** is spanned by the columns of ***U*** with corresponding $\sigma \neq 0$.

For n×n square matrices, the sum of the dimension of the nullspace (nullity) and the dimension of the range (rank) is n.

$$\boldsymbol{A} = \boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^T$$

If ***A*** is square and invertible, then

$$\boldsymbol{A}^{-1} = (\boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^T)^{-1} = \boldsymbol{V}\boldsymbol{D}^{-1}\boldsymbol{U}^T = \boldsymbol{V}\cdot\mathrm{diag}(\sigma_1^{-1}, \sigma_2^{-1}, \ldots, \sigma_n^{-1})\cdot\boldsymbol{U}^T$$

Invertible if and only if all $\sigma_i \neq 0$. Generalization: pseudoinverse

$$\boldsymbol{A}^+ = \boldsymbol{V}\boldsymbol{D}^+\boldsymbol{U}^T,$$

where $\boldsymbol{D}^+$ is an $n \times m$ diagonal matrix that has σ_i^{-1} on the diagonal, except infinities are replaced by zeros ($\infty \to 0$!).

$$\boldsymbol{A}\vec{x} = \vec{b}$$

If $\vec{b}$ is in the range of ***A***, then $\vec{x} = \vec{x}_0 = \boldsymbol{A}^+\vec{b}$ is a solution.

$\boldsymbol{A}\boldsymbol{A}^+\vec{b} = \boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^T\boldsymbol{V}\boldsymbol{D}^+\boldsymbol{U}^T\vec{b} = \boldsymbol{U}\boldsymbol{D}\boldsymbol{D}^+\boldsymbol{U}^T\vec{b}$. $\boldsymbol{U}^T\vec{b}$ is a vector with components $(\vec{u}_i\cdot\vec{b})$. $\boldsymbol{D}\boldsymbol{D}^+$ is a diagonal matrix $m \times m$ with diagonal elements s_i = 1 or 0, with 0 only where σ_i = 0. $\boldsymbol{U}\boldsymbol{D}\boldsymbol{D}^+\boldsymbol{U}^T\vec{b} = \sum_{i=1}^m \vec{u}_i s_i(\vec{u}_i\cdot\vec{b})$. Since $\vec{b}$ is in the range of ***A***, it has a nonzero $(\vec{u}_i\cdot\vec{b})$ only for those i for which s_i = 1, not 0.

$$\sum_{i=1}^m \vec{u}_i s_i(\vec{u}_i\cdot\vec{b}) = \sum_{i=1}^m \vec{u}_i(\vec{u}_i\cdot\vec{b}) = \vec{b}.$$

$$\boldsymbol{A} = \boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^T$$

$$\boldsymbol{A}\vec{x} = \vec{b} \qquad \vec{x}_0 = \boldsymbol{A}^+ \vec{b}$$

To this, any vector from the nullspace can be added, so the general solution is

$$\vec{x} = \boldsymbol{A}^+ \vec{b} + \sum_{i:\sigma_i=0} c_i \vec{v}_i$$

Solution $\vec{x} = \vec{x}_0 = \boldsymbol{A}^+ \vec{b}$ is special, because it has the smallest norm of all.

$\boldsymbol{A}^+ \vec{b} = \boldsymbol{V}\boldsymbol{D}^+ \boldsymbol{U}^T \vec{b}$ $\quad \boldsymbol{V} = (\vec{v}_{1,} \vec{v}_{2,} .., \vec{v}_n)$ $\quad$ When multiplied by $\boldsymbol{D}^+$,

only those components *i* remain nonzero for which $\sigma_i \neq 0$. Conversely, the

second term (the sum) only contains the components for which $\sigma_i = 0$.

So the two terms are orthogonal, and the minimum of the sum is when the second term vanishes.

If $\vec{b}$ is not in the range of A, then there is no solution. However, $\vec{x}_0 = \boldsymbol{A}^+ \vec{b}$ minimizes the residual $\boldsymbol{A}\vec{x} - \vec{b}$. Consider $\vec{x} = \vec{x}_0 + \vec{x}\,'$.

$$\boldsymbol{A}\vec{x} - \vec{b} = \boldsymbol{A}\boldsymbol{A}^+ \vec{b} - \vec{b} + \boldsymbol{A}\vec{x}\,' = \boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^T \boldsymbol{V}\boldsymbol{D}^+ \boldsymbol{U}^T \vec{b} - \vec{b} + \boldsymbol{A}\vec{x}\,'$$
$$= \boldsymbol{U}\boldsymbol{D}\boldsymbol{D}^+ \boldsymbol{U}^T \vec{b} - \vec{b} + \boldsymbol{A}\vec{x}\,'$$

$$\boldsymbol{U}\boldsymbol{D}\boldsymbol{D}^+ \boldsymbol{U}^T \vec{b} - \vec{b} = \sum_{i=1}^{m} \vec{u}_i s_i (\vec{u}_i \cdot \vec{b}) - \sum_{i=1}^{m} \vec{u}_i (\vec{u}_i \cdot \vec{b}) = \sum_{i=1}^{m} \vec{u}_i (s_i - 1)(\vec{u}_i \cdot \vec{b}).$$

s_i = 1 or 0, with 0 only where σ_i = 0. So this is a linear combination of $\vec{u}_i$ with σ_i = 0. Conversely, $\boldsymbol{A}\vec{x}\,'$ belongs to the range of **A** and thus is a linear combination of $\vec{u}_i$ with $\sigma_i \neq 0$. So $\boldsymbol{A}\vec{x}_0 - \vec{b}$ and $\boldsymbol{A}\vec{x}\,'$ are mutually orthogonal and the minimum is achieved when $\vec{x}\,' = \vec{0}$.

Often, it is found numerically that all singular values are nonzero, but some are very small, such that the ratio between the largest and the smallest by absolute value becomes comparable to machine precision. The corresponding entries in $\boldsymbol{D}^{+}$ will be very large.

In such cases, it may actually be better to introduce a cutoff and treat all singular values below the cutoff as zero.

This changes the rank of the matrix, so it's not very well defined.

This is a situation where using SVD may be preferable to direct methods of solving linear systems.

Other applications:

Linear least square fitting.

Principal component analysis – directions of largest variance of data.

Approximation of a matrix by a lower rank matrix.

$$\boldsymbol{A} = \boldsymbol{U}\boldsymbol{D}\boldsymbol{V}^T$$

$$\boldsymbol{A} = \sum_{k=1}^{\min(m,n)} \sigma_k \vec{u}_k \otimes \vec{v}_k \qquad a_{ij} = \sum_{k=1}^{\min(m,n)} \sigma_k u_{k,i} v_{k,j}$$

Each term is a rank-1 matrix.

Matrix A can be represented as a collection of $\vec{u}_k,\ \vec{v}_k,\ \sigma_k.$

Neglect terms with small σ. Reduces storage requirements. (Lossy) data compression. Suitable for images.

512x512

Fig. 6.5. The original image (*left*) and those obtained using the first 20 (*center*) and 60 (*right*) singular values, respectively

A. Quarteroni, F. Saleri, P. Gervasio, Scientific Computing with MATLAB and Octave, Springer, Berlin, 2010.

Computing SVD (Golub-Kahan algorithm)

1. Bidiagonalization using Householder reflections.

$$U^T A V = \begin{pmatrix} d_1 & f_1 & 0 & 0 & \dots & 0 & 0 \\ 0 & d_2 & f_2 & 0 & \dots & 0 & 0 \\ 0 & 0 & d_3 & f_3 & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & 0 & \dots & d_{n-1} & f_{n-1} \\ 0 & 0 & 0 & 0 & \dots & 0 & d_n \end{pmatrix}$$

2. Diagonalization using Givens rotations.