Main matrix factorizations

\[ A = P L U \]

\( P \) permutation matrix, \( L \) lower triangular, \( U \) upper triangular

Key use: Solve square linear system \( A x = b \).

\[ A = Q R \]

\( Q \) unitary, \( R \) upper triangular

Key use: Solve square or overdetermined linear systems \( A x = b \).

\[ A = U \Sigma V^* \]

\( U \), \( V \) unitary, \( \Sigma \) diagonal with non-increasing, non-negative elements

Key uses:
- Overdetermined linear systems
- Understand effect of matrix-vector product \( A x \).
- Extract and compress information contained in \( A \).
\[ A = PLU \]

Solve \[ A x = b \implies PLU x = b \implies LU x = P^T b \]

Splits original system into two successive triangular ones

\[
\begin{bmatrix}
\ddots & & & & \\
& \ddots & & & \\
& & L & & \\
& & & & \ddots \\
& & & & & \ddots \\
\end{bmatrix}
\begin{bmatrix}
\vdots \\
\vdots \\
y \\
\vdots \\
\vdots \\
\vdots \\
\end{bmatrix}
= \begin{bmatrix}
\vdots \\
\vdots \\
P^T b \\
\vdots \\
\vdots \\
\vdots \\
\end{bmatrix},
\begin{bmatrix}
\ddots & & & & \\
& \ddots & & & \\
& & U & & \\
& & & & \ddots \\
& & & & & \ddots \\
\end{bmatrix}
\begin{bmatrix}
\vdots \\
\vdots \\
x \\
\vdots \\
\vdots \\
\vdots \\
\end{bmatrix}
= \begin{bmatrix}
\vdots \\
\vdots \\
y \\
\vdots \\
\vdots \\
\vdots \\
\end{bmatrix}
\]

Stable, Factorization cost \( O(N^3) \) operations

Can be obtained directly (Crout or Doolittle), or as a byproduct of Gaussian elimination

Cost for each RHS: \( O(N^2) \) operations

One back substitution
\( A = Q R \)

A square or rectangular:

\[
\begin{bmatrix}
  \cdots & A & \cdots \\
  \cdots & \cdots & \cdots \\
  \cdots & \cdots & \cdots \\
  \cdots & \cdots & \cdots \\
  \cdots & \cdots & \cdots \\
\end{bmatrix} =
\begin{bmatrix}
  \cdots & Q & \cdots \\
  \cdots & \cdots & \cdots \\
  \cdots & \cdots & \cdots \\
  \cdots & \cdots & \cdots \\
  \cdots & \cdots & \cdots \\
\end{bmatrix}
\begin{bmatrix}
  \cdots & R & \cdots \\
  \cdots & \cdots & \cdots \\
  \cdots & \cdots & \cdots \\
  \cdots & \cdots & \cdots \\
  \cdots & \cdots & \cdots \\
\end{bmatrix}
\begin{bmatrix}
  \cdots \\
  \cdots \\
  \cdots \\
  \cdots \\
  \cdots \\
\end{bmatrix}
\]

\[
\begin{bmatrix}
  \cdots & A & \cdots \\
  \cdots & \cdots & \cdots \\
  \cdots & \cdots & \cdots \\
  \cdots & \cdots & \cdots \\
  \cdots & \cdots & \cdots \\
\end{bmatrix} =
\begin{bmatrix}
  \cdots & Q & \cdots \\
  \cdots & \cdots & \cdots \\
  \cdots & \cdots & \cdots \\
  \cdots & \cdots & \cdots \\
  \cdots & \cdots & \cdots \\
\end{bmatrix}
\begin{bmatrix}
  \cdots \\
  \cdots \\
  \cdots \\
  \cdots \\
  \cdots \\
\end{bmatrix}
\begin{bmatrix}
  \cdots & R & \cdots \\
  \cdots & \cdots & \cdots \\
  \cdots & \cdots & \cdots \\
  \cdots & \cdots & \cdots \\
  \cdots & \cdots & \cdots \\
\end{bmatrix}
\begin{bmatrix}
  \cdots \\
  \cdots \\
  \cdots \\
  \cdots \\
  \cdots \\
\end{bmatrix}
\]

Significance:

(1) Solve \( A x = b \) \( \Rightarrow \) \( Q R x = b \) \( \Rightarrow \) \( R x = Q^T b \);

Slightly higher cost than \( A = P L U \); however no pivoting needed.

(2) Solve (least squares) overdetermined linear systems

Normal equations \( A^* A x = A^* b \) often ill conditioned. If starting with \( A \) in the form \( A = Q R \):

\[
\begin{bmatrix}
  \cdots & \cdots & R^* \\
  \cdots & \cdots & \cdots \\
\end{bmatrix}
\begin{bmatrix}
  \cdots & Q^* & \cdots \\
  \cdots & \cdots & \cdots \\
\end{bmatrix}
\begin{bmatrix}
  \cdots & Q & \cdots \\
  \cdots & \cdots & \cdots \\
\end{bmatrix}
\begin{bmatrix}
  \cdots & R & \cdots \\
  \cdots & \cdots & \cdots \\
\end{bmatrix}
\begin{bmatrix}
  \cdots \\
  \cdots \\
  \cdots \\
  \cdots \\
\end{bmatrix}
\begin{bmatrix}
  \cdots & R^* & b \\
  \cdots & \cdots & \cdots \\
\end{bmatrix}
\]

Identity matrix; vanishes

\[
\begin{bmatrix}
  \cdots & \cdots & R^* \\
  \cdots & \cdots & \cdots \\
\end{bmatrix}
\begin{bmatrix}
  \cdots & R & \cdots \\
  \cdots & \cdots & \cdots \\
\end{bmatrix}
\begin{bmatrix}
  \cdots \\
  \cdots \\
  \cdots \\
\end{bmatrix}
\begin{bmatrix}
  \cdots & R^* & \hat{b} \\
  \cdots & \cdots & \cdots \\
\end{bmatrix}
\]

Now \( R^* \) vanishes \textit{analytically}, then well conditioned.
\[ A = U \Sigma V^* \]  
\textbf{SVD}

Applies to \( A_{m \times n} \) of any shape:

\( \text{If } m < n : \)

\[
\begin{bmatrix}
\vdots & \vdots & \vdots \\
\cdot & \cdot & A \\
\vdots & \vdots & \vdots
\end{bmatrix}
= 
\begin{bmatrix}
\cdot \\
\cdot \\
U & \cdot
\end{bmatrix}
\begin{bmatrix}
\Sigma & \cdot \\
\cdot & \cdot
\end{bmatrix}
\begin{bmatrix}
\cdot & V^* \\
\cdot & \cdot
\end{bmatrix}
\]

Reduced form:

\[
\begin{bmatrix}
\vdots & \vdots & \vdots \\
\cdot & \cdot & A \\
\vdots & \vdots & \vdots
\end{bmatrix}
= 
\begin{bmatrix}
\cdot \\
\cdot \\
U & \cdot
\end{bmatrix}
\begin{bmatrix}
\Sigma & \cdot \\
\cdot & \cdot
\end{bmatrix}
\begin{bmatrix}
\cdot & \cdot \\
\cdot & \cdot
\end{bmatrix}
\]

\( \text{If } m > n : \)

\[
\begin{bmatrix}
\vdots & \vdots & \vdots \\
\cdot & \cdot & \cdot \\
A & \cdot & \cdot
\end{bmatrix}
= 
\begin{bmatrix}
\cdot \\
\cdot \\
U & \cdot
\end{bmatrix}
\begin{bmatrix}
\Sigma & \cdot \\
\cdot & \cdot
\end{bmatrix}
\begin{bmatrix}
\cdot & V^* \\
\cdot & \cdot
\end{bmatrix}
\]

Reduced form:

\[
\begin{bmatrix}
\vdots & \vdots & \vdots \\
\cdot & \cdot & \cdot \\
A & \cdot & \cdot
\end{bmatrix}
= 
\begin{bmatrix}
\cdot \\
\cdot \\
U & \cdot
\end{bmatrix}
\begin{bmatrix}
\Sigma & \cdot \\
\cdot & \cdot
\end{bmatrix}
\begin{bmatrix}
\cdot & \cdot \\
\cdot & \cdot
\end{bmatrix}
\]

Alternate illustration of reduced form:
Existence and uniqueness of SVD factorization

Theorem: Every $A_{m,n}$ matrix has an SVD factorization

$$A = U \Sigma V^*$$

The singular values $\sigma_j$ are uniquely determined.

For a square: If the $\sigma_j$ are distinct, then the singular vectors are uniquely determined apart from complex factors of magnitude one.
SVD - The effect of performing $A \times x$

2 x 2 illustration

$$
\begin{bmatrix}
\vdots \\
A \\
\vdots \\
\end{bmatrix} = \begin{bmatrix}
\begin{pmatrix}u_1 \\ u_2\end{pmatrix} \\
U \\
\begin{pmatrix}v_1 \\ v_2\end{pmatrix} \\
V^* \\
\end{bmatrix} \begin{bmatrix}
\sigma_1 \\
\sigma_2 \\
\Sigma \\
\end{bmatrix} \begin{pmatrix}v_1 \\ v_2\end{pmatrix} \\

The range of $A \times x$, when $A$ is a 2 x 2 matrix and $x$ is a unit length vector, is an ellipse with semiaxes $\sigma_1 u_1$ and $\sigma_2 u_2$; the pre-images are $v_1$ and $v_2$.

The concept generalizes to all sizes of $A$. 
The four fundamental subspaces of a matrix

Every matrix \( \begin{bmatrix} \cdot & \cdot & \cdot & A & \cdot & \cdot & \cdot \end{bmatrix} \) has four fundamental subspaces:

- **Row space:** Space spanned by all the rows of \( A \)
- **Null space:** Space of all column vectors \( \nu \) such that \( A \nu = 0 \)
- **Column space:** Space spanned by all the columns of \( A \)
- **Left null space:** Space spanned by all row vectors \( \mu \) such that \( \mu A = 0 \)

\[
\begin{bmatrix}
A
\end{bmatrix}
=\begin{bmatrix}
\begin{array}{c}
U
\end{array}
\end{bmatrix}
\begin{bmatrix}
\begin{array}{c}
\Sigma
\end{array}
\end{bmatrix}
\begin{bmatrix}
\begin{array}{c}
V^*
\end{array}
\end{bmatrix}
\]

- \( U \) comprises the first \( r \) basis for column space of \( A \)
- \( V^* \) comprises the first \( r \) basis for row space of \( A \)
- \( \Sigma \) comprises \( r \) singular values of \( A \) and \( n \) all other entries zero
The orthogonality of the four fundamental subspaces.
An immediate consequence of the SVD decomposition of any matrix $A$. 
A can be written as a sum of \( r \) rank one matrices. 

Recall

\[
\begin{bmatrix}
\vdots & \vdots & A & \vdots & \vdots & \vdots \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
\end{bmatrix} = \begin{bmatrix}
\vdots & U & \vdots \\
\vdots & \vdots & \vdots \\
\end{bmatrix} \begin{bmatrix}
\Sigma \\
\vdots & \vdots & \vdots \\
\end{bmatrix} \begin{bmatrix}
\vdots & V^* & \vdots \\
\vdots & \vdots & \vdots \\
\end{bmatrix}
\]

\[
\Rightarrow A = \begin{bmatrix}
\begin{array}{c}
u_1 \\
u_2 \\
\end{array}
\end{bmatrix} \sigma_1 \begin{bmatrix}
\begin{array}{c}

V_1^* \\
\end{array}
\end{bmatrix} + \begin{bmatrix}
\begin{array}{c}
u_2 \\
u_3 \\
\end{array}
\end{bmatrix} \sigma_2 \begin{bmatrix}
\begin{array}{c}

V_2^* \\
\end{array}
\end{bmatrix} + \cdots + \begin{bmatrix}
\begin{array}{c}
u_r \\
u_r+1 \\
\end{array}
\end{bmatrix} \sigma_r \begin{bmatrix}
\begin{array}{c}

V_r^* \\
\end{array}
\end{bmatrix}
\]

Let

\[
\begin{bmatrix}
U_i \\
\end{bmatrix} = \begin{bmatrix}
\begin{array}{c} u_i \\
\end{array}
\end{bmatrix} \begin{bmatrix}
\begin{array}{c} v_i^* \\
\end{array}
\end{bmatrix} , \quad \begin{bmatrix}
U_j \\
\end{bmatrix} = \begin{bmatrix}
\begin{array}{c} u_j \\
\end{array}
\end{bmatrix} \begin{bmatrix}
\begin{array}{c} v_j^* \\
\end{array}
\end{bmatrix} , \quad i \neq j ,
\]

- Then every column (row) of \( U_i \) is orthogonal to every column (row) of \( U_j \).
- If we re-order \( U_i, U_j \) into single columns, they become orthogonal.

Key use of SVD:

If the \( \sigma_k \) decay fast, expansion can be truncated early \( \Rightarrow \) \{ Data compression, Feature extraction \}
First example of data compression

Top left panel: A 256x256 image (matrix)

Below: Its singular values:

The singular values.

Remaining panels: The image (matrix) reconstruction when keeping 20, 50, and 75 out of the 256 singular values.

However: .jpg etc. are more effective for image compression - utilize more application specific features.
Extraction of translation and rotation information

Left panel: A signature, digitized as a 2 x 262 matrix of x,y - coordinates at 262 sample points.

Computation: Form the SVD decomposition of A: \( A = U \Sigma V^* \):

\[
U = \begin{bmatrix}
-0.9851 & 0.1718 \\
-0.1718 & -0.9851
\end{bmatrix}, \quad \Sigma = \begin{bmatrix}
73.79 & \\
& 19.52
\end{bmatrix}
\]

Right panel: Principle axis and (hyper) ellipse associated with U and \( \Sigma \).
A comparison of some different biometrics:

<table>
<thead>
<tr>
<th>Biometric</th>
<th>Reliability</th>
<th>Acceptance</th>
<th>Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fingerprint</td>
<td>4</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Hand geometry</td>
<td>3</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>Eye retina</td>
<td>5</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>Iris</td>
<td>5</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>Face</td>
<td>3</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>Voice</td>
<td>3</td>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td>Handwriting</td>
<td>2</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>Keystroke</td>
<td>1</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>Signature</td>
<td>3</td>
<td>5</td>
<td>1</td>
</tr>
</tbody>
</table>

Dependent on the application, facial image recognition may be a reasonable compromise between reliability, acceptance, and cost.
Data set for facial image recognition

Two images in 'Training set' of around 600 images.

Each image 100x100 pixels rearranged into a 10,000 x 1 vector.

All the data together forms a 10000 x 600 matrix: $A$
Average face

Previous two faces

Caricatures - faces with average subtracted

⇒

⇑

⇓

⇓
SVD of matrix $A$ based on caricatures

$$A = U \Sigma V^* ; \text{ Plot of } \lambda = \sigma^2 \Rightarrow$$

To each singular value corresponds an 'eigenface':

Orthogonal eigenfaces form a very efficient base for expressing real faces. Vast data reduction from length 10,000 vectors to less than about 100 coefficients.
Representation of pictures not seen before

Find coefficients such that
\[ X \approx \lambda_1 u_1 + \ldots + \lambda_r u_r \]

Projection of \( X \) on space of eigenfaces

\[ \Rightarrow \]

New data vector \( X \)

Reconstructions with 10 and 40 eigenfaces, resp.

\[ \Rightarrow \]

Best fit to the rose within the range of the training set
Recognizing faces

Facial image not in training set

Its coefficients, and those that match the best from the training set

Identified image in the training set

Clearly the same person

Many practical issues not discussed here, incl. corrections for perspective, lightning, changes in hair style, eye glasses, etc.
Numerical computation of the SVD factorization

A BAD way: If \( A = U \Sigma V^* \), then

\[
A^* A = (U \Sigma V^*)^* (U \Sigma V^*) = V \Sigma^* U^* U \Sigma V^* = V (\Sigma^* \Sigma) V^*
\]

\[
\Rightarrow (A^* A) V = V (\Sigma^* \Sigma)
\]
Hermitean  Diagonal

\[
\Rightarrow \text{The columns of } V \text{ are the eigenvectors of } A^* A
\]

The singular values are \( \sigma_i = \sqrt{\lambda_i} \) where \( \lambda_i \) are the eigenvalues of \( A^* A \).

\( U \) follows directly from \( A = U \Sigma V^* \).
Numerical computation of the SVD factorization

A **GOOD** way (for \( A \) of size up to around 1000 x 1000)

Compare with QR algorithm for eigenvalues:

\[
\begin{bmatrix}
\ddots & \ddots & \ddots & \ddots \\
& A & & \\
& & \ddots & \ddots \\
& & & \ddots \\
\end{bmatrix}
\]

phase 1

\( \rightarrow \rightarrow \rightarrow \) direct,

\( O(n^3) \) op.

\[
\begin{bmatrix}
\ddots & \ddots & \ddots & \ddots \\
& & \ddots & \ddots \\
& & & \ddots \\
& & & & \\
\end{bmatrix}
\]

phase 2

\( \rightarrow \rightarrow \rightarrow \) iterative,

\( O(n) \) iter,

\( O(n^2) \) op / iter.

\[
\begin{bmatrix}
\ddots & \ddots & \ddots & \ddots \\
& & & \\
& & & \\
& & & \\
\end{bmatrix}
\]

Upper triangular

Now, for the SVD:

\[
\begin{bmatrix}
\ddots & \ddots & \ddots & \ddots \\
& A & & \\
& & \ddots & \ddots \\
& & & \ddots \\
\end{bmatrix}
\]

phase 1

\( \rightarrow \rightarrow \rightarrow \) direct,

\( O(n^3) \) op.

\[
\begin{bmatrix}
\ddots & \ddots & \ddots \\
& & \ddots \\
& & & \ddots \\
& & & & \\
\end{bmatrix}
\]

phase 2

\( \rightarrow \rightarrow \rightarrow \) iterative,

\( O(n) \) iter,

\( O(n) \) op / iter.

\[
\begin{bmatrix}
\ddots & \ddots & \ddots \\
& & \ddots \\
& & & \ddots \\
& & & & \\
\end{bmatrix}
\]

Diagonal

Key tool in both phases in both cases: **Householder transformations**.

**For \( A \) sparse and very large:** Effective iterative methods available for leading singular values / vectors (Lanczos and subspace iteration methods).
## Comparison of SVD vs. eigenvalues / eigenvectors

<table>
<thead>
<tr>
<th></th>
<th>SVD</th>
<th>Eigenvalues / eigenvectors</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Formulas:</strong></td>
<td>$A = U \Sigma V^*$</td>
<td>$A = X \Lambda X^{-1}$</td>
</tr>
<tr>
<td><strong>Properties:</strong></td>
<td>$U$, $V$ unitary</td>
<td>$X$ typically not unitary</td>
</tr>
<tr>
<td></td>
<td>$\Sigma$ diagonal, non-negative, ordered</td>
<td>(only the case if $A$ 'normal', i.e. $AA^* = A^*A$)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$\Lambda$ may contain 'Jordan boxes'</td>
</tr>
<tr>
<td><strong>Uses:</strong></td>
<td>Reveals 'Character' of $A$, and of its fundamental subspaces, Gives most general solution to $Ax = b$, whatever shape $A$ is. Feature extraction Data compression</td>
<td>Matrix powers $A^k$, functions of $A$ Linear systems of ODEs Stability analysis</td>
</tr>
</tbody>
</table>
SVD