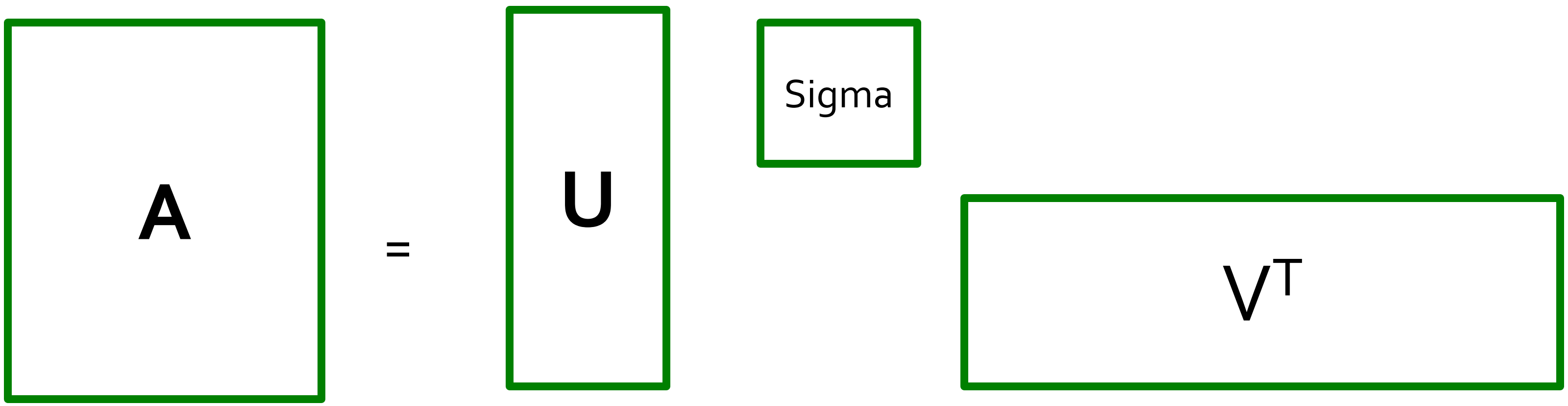
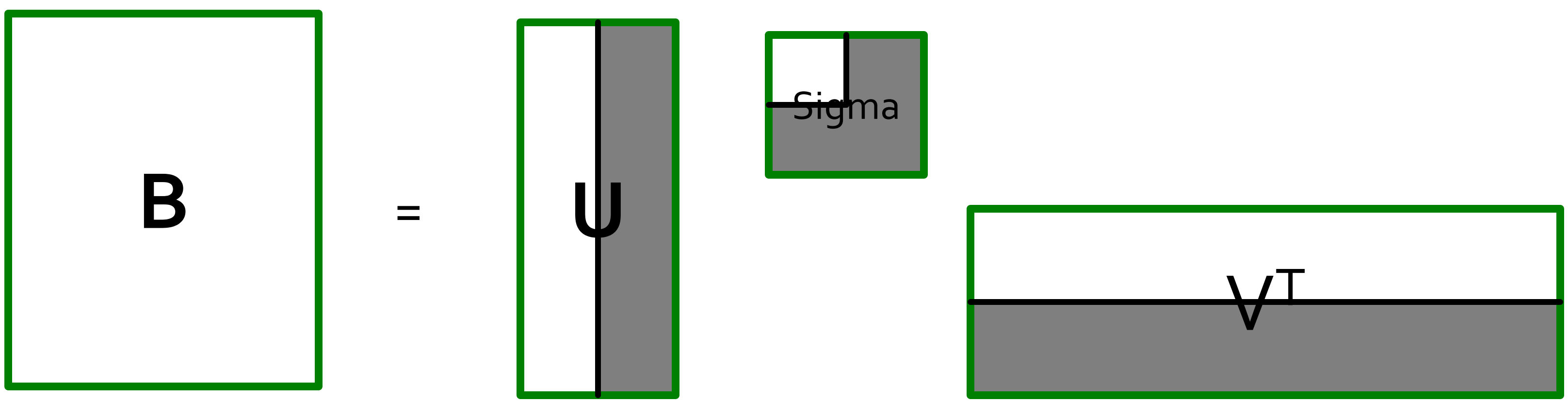
SVD – Best Low Rank Approx.



B is best approximation of A: $\|A-B\|_F = \sqrt{\Sigma_{ij} (A_{ij}-B_{ij})^2}$

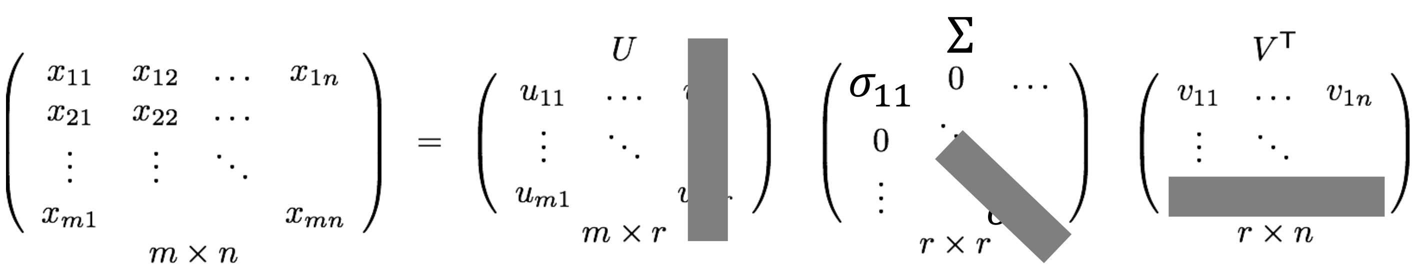


SVD – Best Low Rank Approx.

- **Theorem:**
  Let A = U Σ V^T where Σ: $\sigma_1 \geq \sigma_2 \geq \ldots$, and rank(***A***)=***r***
  then B = U S V^T is a **best** rank-***k*** approx. to ***A***
  - **Where:**
    S = diagonal *n*x*n* matrix where ***s$_i$*=*σ$_i$*** (*i=1…k*) else ***s$_i$*=0**

What do we mean by "best":

  - ***B* is a solution to min$_B$ ‖*A*-*B*‖$_F$ where rank(*B*)=*k***



$$\|A-B\|_F = \sqrt{\Sigma_{ij} (A_{ij}-B_{ij})^2}$$

SVD - Interpretation #3

Equivalent:

'spectral decomposition' of the matrix:

$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 3 & 3 & 3 & 0 & 0 \\ 4 & 4 & 4 & 0 & 0 \\ 5 & 5 & 5 & 0 & 0 \\ 0 & 2 & 0 & 4 & 4 \\ 0 & 0 & 0 & 5 & 5 \\ 0 & 1 & 0 & 2 & 2 \end{bmatrix} = \begin{bmatrix} | & | \\ u_1 & u_2 \\ | & | \end{bmatrix} \times \begin{bmatrix} \sigma_1 & \emptyset \\ \emptyset & \sigma_2 \end{bmatrix} \times \begin{bmatrix} \text{---} \; v_1 \; \text{---} \\ \text{---} \; v_2 \; \text{---} \end{bmatrix}$$

SVD - Interpretation #3

Equivalent:

'spectral decomposition' of the matrix

$$
\begin{array}{c}
\xleftarrow{\quad\quad} \text{m} \xrightarrow{\quad\quad} \\
\text{n} \updownarrow
\begin{bmatrix}
1 & 1 & 1 & 0 & 0 \\
3 & 3 & 3 & 0 & 0 \\
4 & 4 & 4 & 0 & 0 \\
5 & 5 & 5 & 0 & 0 \\
0 & 2 & 0 & 4 & 4 \\
0 & 0 & 0 & 5 & 5 \\
0 & 1 & 0 & 2 & 2
\end{bmatrix}
\end{array}
= \overbrace{\sigma_1 \underbrace{u_1}_{n \times 1} \underbrace{v^T_1}_{1 \times m} + \sigma_2 u_2 v^T_2 + \ldots}^{\text{k terms}}
$$

Assume: $\sigma_1 \geq \sigma_2 \geq \sigma_3 \geq \ldots \geq 0$

Why is setting small σ_i to 0 the right thing to do?

Vectors $\mathbf{u}_i$ and $\mathbf{v}_i$ are unit length, so σ_i scales them.

So, zeroing small σ_i introduces less error.

SVD - Interpretation #3

Q: How many σs to keep?

A: Rule-of-a thumb:

keep 80-90% of 'energy' $(=\sum \sigma_i^2)$

$$
\begin{array}{cc}
 & m \\
n & \begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 3 & 3 & 3 & 0 & 0 \\ 4 & 4 & 4 & 0 & 0 \\ 5 & 5 & 5 & 0 & 0 \\ 0 & 2 & 0 & 4 & 4 \\ 0 & 0 & 0 & 5 & 5 \\ 0 & 1 & 0 & 2 & 2 \end{bmatrix}
\end{array}
= \sigma_1 \; u_1 \; v^T_1 + \sigma_2 \; u_2 \; v^T_2 + \ldots
$$

Assume: $\sigma_1 \geq \sigma_2 \geq \sigma_3 \geq \ldots$

SVD - Complexity

- **To compute SVD:**
  - **$O(nm^2)$** or **$O(n^2m)$** (whichever is less)
- **But:**
  - Less work, if we just want singular values
  - or if we want first *k* singular vectors
  - or if the matrix is sparse
- **Implemented in** linear algebra packages like
  - LINPACK, Matlab, SPlus, Mathematica ...

SVD - Conclusions so far

- **SVD:** $A = U \Sigma V^T$: **unique**
  - **U**: user-to-concept similarities
  - **V**: movie-to-concept similarities
  - Σ : strength of each concept
- **Dimensionality reduction:**
  - keep the few largest singular values (80-90% of ‘energy’)
  - SVD: picks up linear correlations