Chapter 1

SVD, PCA & Pre-processing

Part 1: Linear algebra and SVD

max planck institut
informatik

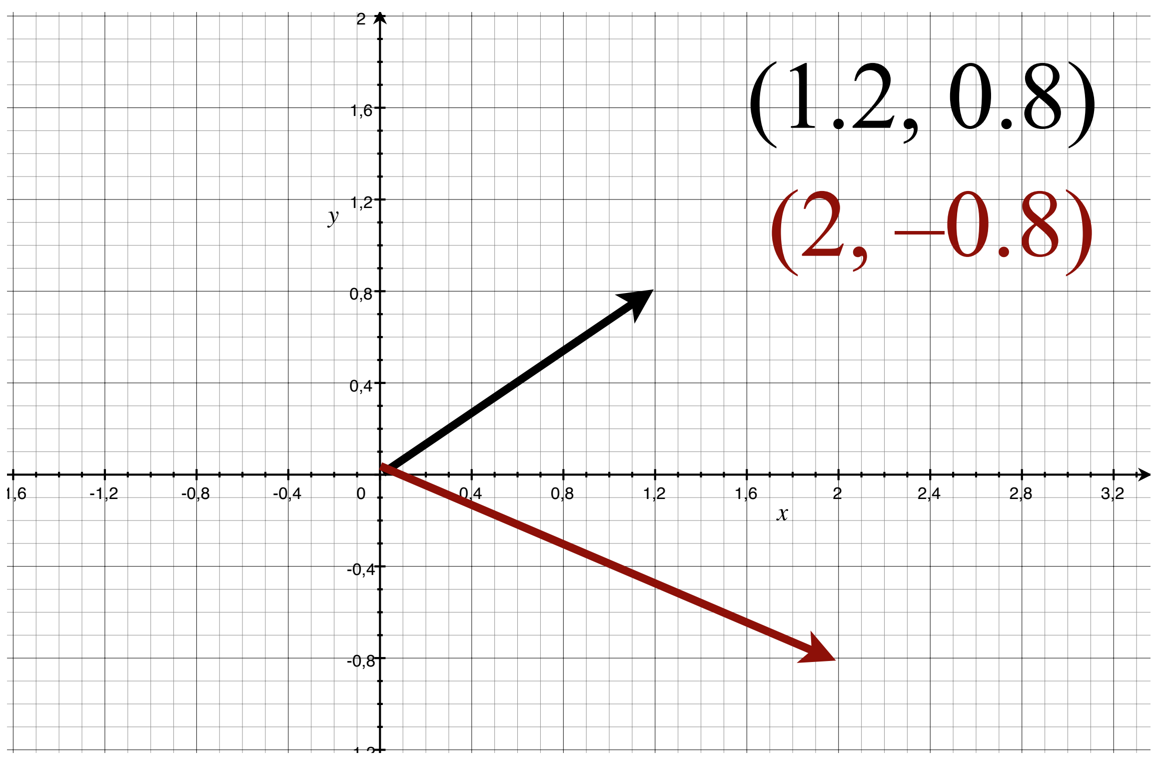
Contents

- Linear algebra crash course
- The singular value decomposition
- Applications of SVD
- Normalization & selecting the rank
- Computing the SVD

Linear Algebra Crash Course

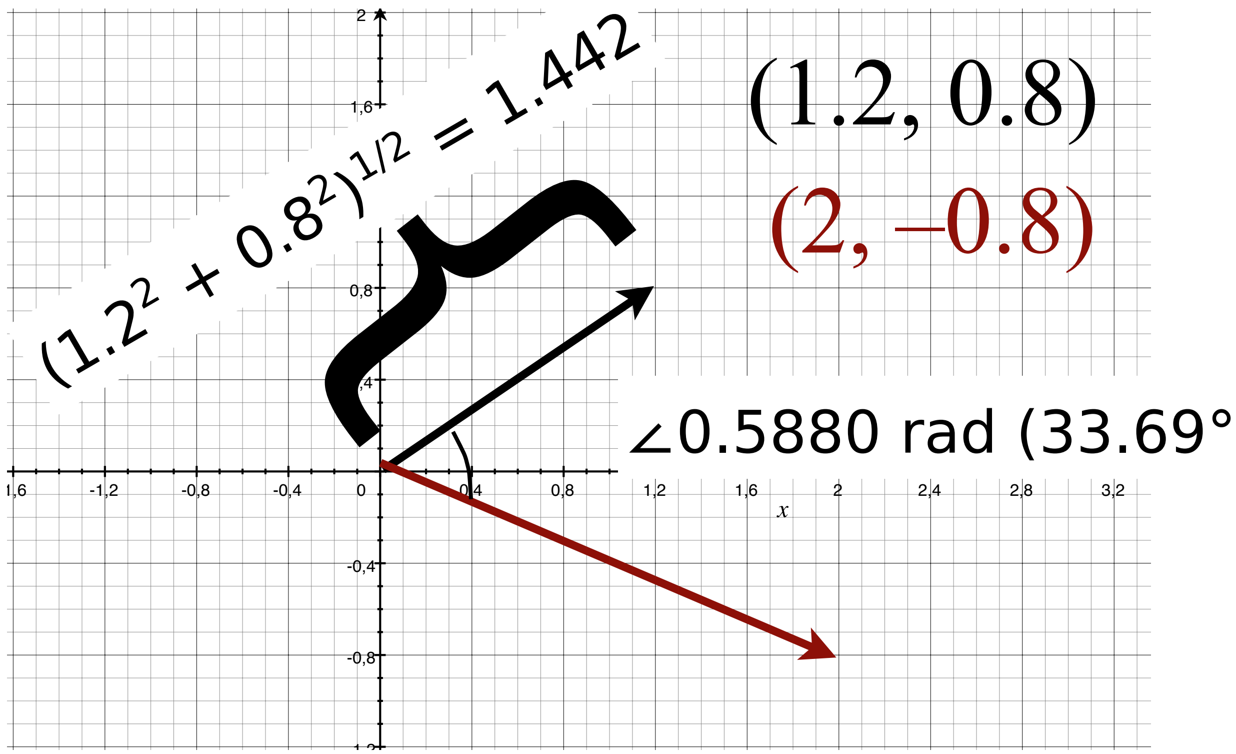
Matrices and vectors

- A **vector** is
  - a 1D array of numbers
  - a geometric entity with magnitude and direction
  - a matrix with exactly one row or column



Norms and angles

- The magnitude is measure by a (vector) **norm**
  - The **Euclidean** norm
    $$\|\boldsymbol{x}\| = \|\boldsymbol{x}\|_2 = \left(\sum_{i=1}^{n} x^2\right)^{1/2}$$
  - General L_p norm $(1 \leq p \leq \infty)$
    $$\|\boldsymbol{x}\|_p = \left(\sum_{i=1}^{n} |x|^p\right)^{1/p}$$
- The direction is measured by the **angle**



Basic vector operations

- The **transpose** of $\boldsymbol{x}$, $\boldsymbol{x}^T$, transposes a row vector into a column vector and vice versa
- A **dot product** of two vectors of the same dimension is $\boldsymbol{x} \cdot \boldsymbol{y} = \sum_{i=1}^{n} x_i y_i$
  - A.k.a. **scalar product** or **inner product**
  - Same as $\langle \boldsymbol{x}, \boldsymbol{y} \rangle$, $\boldsymbol{a}^T\boldsymbol{b}$ (for column vectors), or $\boldsymbol{a}\boldsymbol{b}^T$ (for row vectors)

Orthogonality

- **Orthogonality** is a generalization of perpendicularity
  - $\boldsymbol{x}$ and $\boldsymbol{y}$ are orthogonal if $\boldsymbol{x} \cdot \boldsymbol{y} = 0$
    - HW: this generalizes standard definition

Matrix algebra

- Matrices in $\mathbb{R}^{n \times n}$ form a ring
  - Addition, subtraction, and multiplication
  - But usually no division
  - Multiplication is not commutative
    - $\boldsymbol{AB} \neq \boldsymbol{BA}$ in general

Matrix multiplication

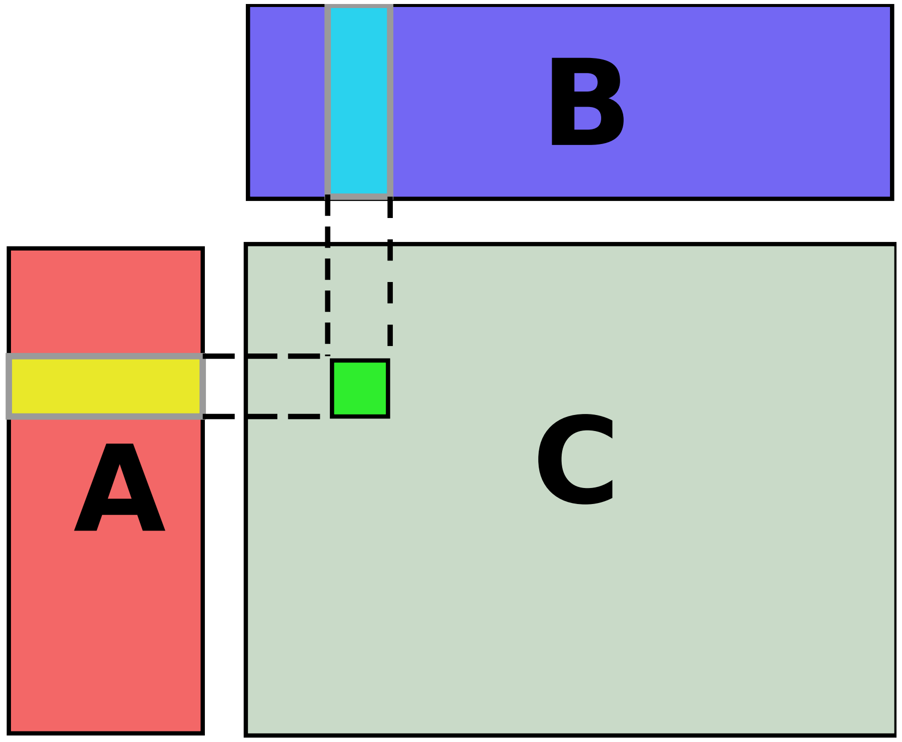
- The product of two matrices, $\boldsymbol{A}$ and $\boldsymbol{B}$, is defined element-wise as

$$(\boldsymbol{AB})_{ij} = \sum_{\ell=1}^{k} a_{i\ell} b_{\ell j}$$

  - The number of columns in $\boldsymbol{A}$ and number of rows in $\boldsymbol{B}$ must agree
    - inner dimension

Intuition for Matrix Multiplication

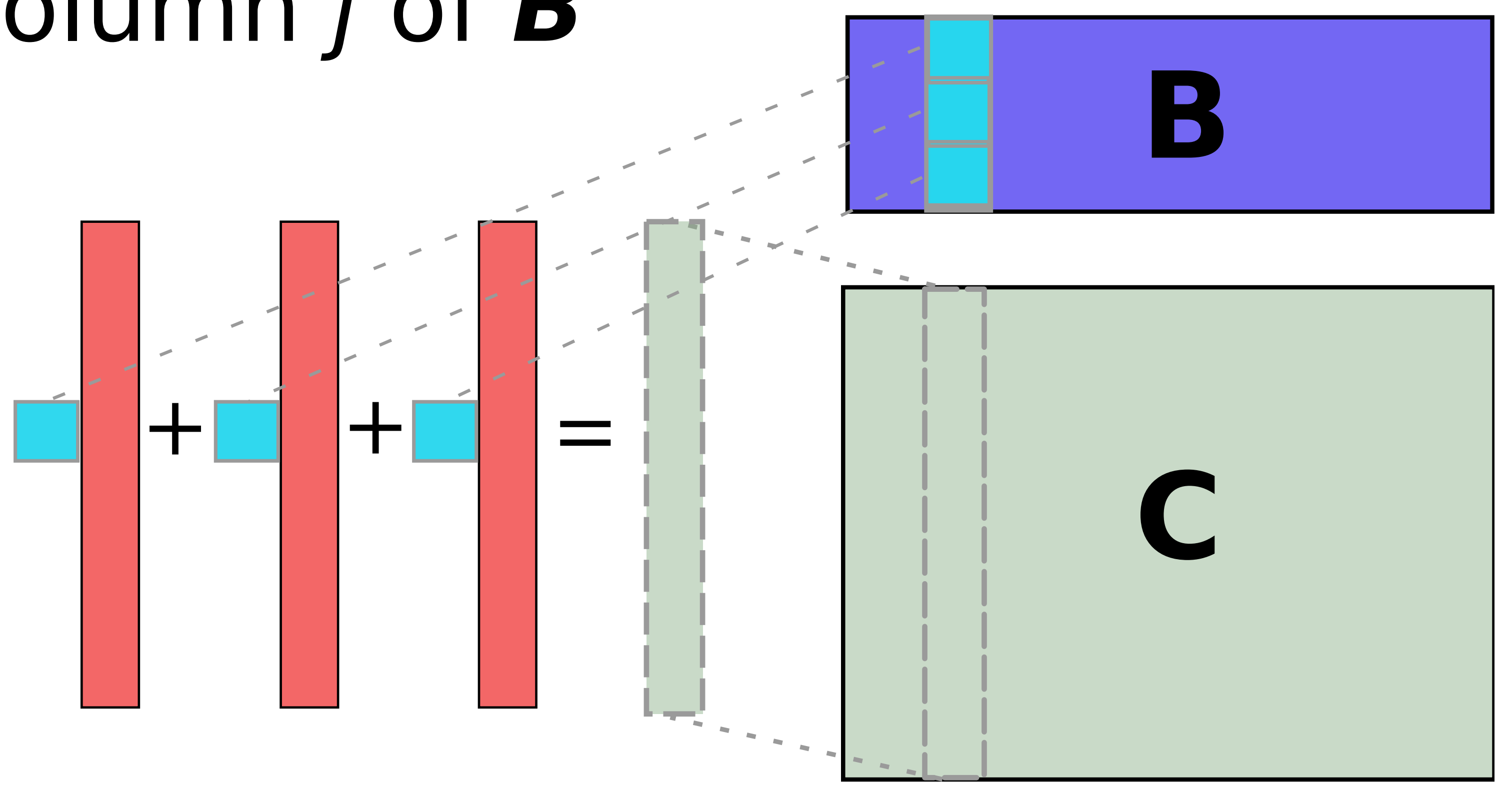
- Element $(\boldsymbol{AB})_{ij}$ is the inner product of row i of $\boldsymbol{A}$ and column j of $\boldsymbol{B}$



$$\boldsymbol{C}_{ij} = \sum_{\ell=1}^{k} a_{i\ell} b_{\ell j}$$

Intuition for Matrix Multiplication

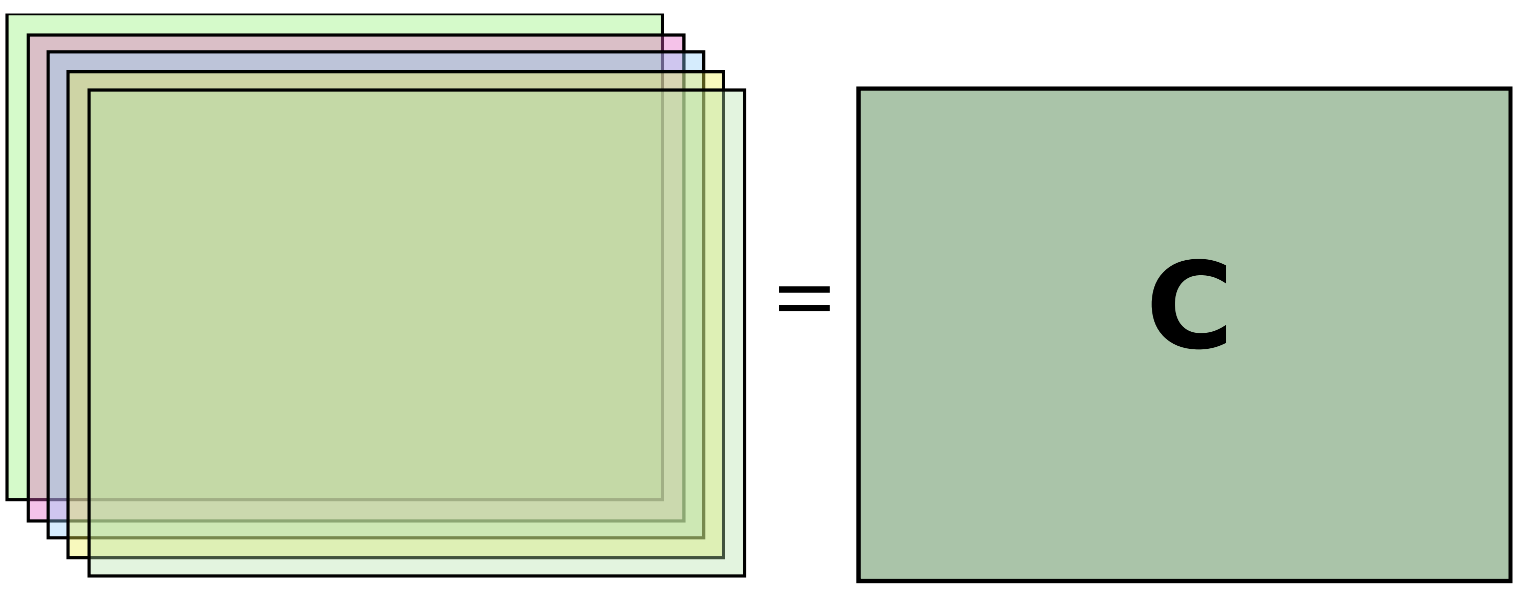
- Column j of $\boldsymbol{AB}$ is the linear combination of columns of $\boldsymbol{A}$ with the coefficients coming from column j of $\boldsymbol{B}$



$$\boldsymbol{C} = \left[\left[\sum_{\ell=1}^{k} b_{\ell 1} \boldsymbol{a}_\ell \right] \quad \left[\sum_{\ell=1}^{k} b_{\ell 2} \boldsymbol{a}_\ell \right] \cdots \left[\sum_{\ell=1}^{k} b_{\ell m} \boldsymbol{a}_\ell \right] \right]$$

Intuition for Matrix Multiplication

- Matrix $\boldsymbol{AB}$ is a sum of k matrices $\boldsymbol{a}_l\boldsymbol{b}_l^T$ obtained by multiplying the l-th column of $\boldsymbol{A}$ with the l-th row of $\boldsymbol{B}$



$$\boldsymbol{C} = \sum_{\ell=1}^{k} \boldsymbol{a}_\ell \boldsymbol{b}_\ell^T$$

Matrix decompositions

- A **decomposition** of matrix $\boldsymbol{A}$ expresses it as a product of two (or more) **factor matrices**
  - $\boldsymbol{A} = \boldsymbol{BC}$
- Every matrix has decomposition $\boldsymbol{A} = \boldsymbol{AI}$ (or $\boldsymbol{A} = \boldsymbol{IA}$ if $n < m$)
- The size of the decomposition is the inner dimension of the product

Matrices as linear maps

- Matrix $\boldsymbol{A} \in \mathbb{R}^{n \times m}$ is a **linear mapping** from $\mathbb{R}^m$ to $\mathbb{R}^n$
  - $\boldsymbol{A}(\boldsymbol{x}) = \boldsymbol{A}\boldsymbol{x}$
- If $\boldsymbol{A} \in \mathbb{R}^{n \times k}$ and $\boldsymbol{B} \in \mathbb{R}^{k \times m}$, then $\boldsymbol{AB}$ is a mapping from $\mathbb{R}^m$ to $\mathbb{R}^n$
- The transpose $\boldsymbol{A}^T$ is a mapping from $\mathbb{R}^n$ to $\mathbb{R}^m$
  - $(\boldsymbol{A}^T)_{ij} = \boldsymbol{A}_{ji}$
  - $(\boldsymbol{AB})^T = \boldsymbol{B}^T\boldsymbol{A}^T$

Matrix inverse

- Square matrix $\boldsymbol{A}$ is **invertible** if there is a matrix $\boldsymbol{B}$ s.t. $\boldsymbol{AB} = \boldsymbol{BA} = \boldsymbol{I}$
  - $\boldsymbol{B}$ is the inverse of $\boldsymbol{A}$, denoted $\boldsymbol{A}^{-1}$
  - Usually the transpose is **not** the inverse
- Non-square matrices don't have general inverses
  - Can have left or right inverse:
    $\boldsymbol{AR} = \boldsymbol{I}$ or $\boldsymbol{LA} = \boldsymbol{I}$

Linear independency

- Vector $\boldsymbol{u}$ is **linearly dependent** on a set of vectors $\boldsymbol{V} = \{\boldsymbol{v}_i\}$ if $\boldsymbol{u}$ is a linear combination of $\boldsymbol{v}_i$
  - $\boldsymbol{u} = \sum_i a_i \boldsymbol{v}_i$ for some a_i
  - If $\boldsymbol{u}$ is not linearly dependent, it is **linearly independent**
- Set V of vectors is **linearly independent** if all $\boldsymbol{v}_i$ are linearly independent of $V \setminus \{\boldsymbol{v}_i\}$

Matrix ranks

- The **column rank** of a matrix $\boldsymbol{A}$ is the number of linearly independent columns of $\boldsymbol{A}$
- The **row rank** of $\boldsymbol{A}$ is the number of linearly independent rows of $\boldsymbol{A}$
- The **Schein rank** of $\boldsymbol{A}$ is the least integer k such that $\boldsymbol{A}$ can be expressed as a sum of k rank-1 matrices
  - Rank-1 matrix is an outer product of two vectors

Orthogonal matrices

- Set of vectors $\{\boldsymbol{v}_i\}$ is **orthogonal** if all $\boldsymbol{v}_i$ are mutually orthogonal, i.e. $\langle \boldsymbol{v}_i, \boldsymbol{v}_j \rangle = 0$ for all $i \neq j$
  - If $\|\boldsymbol{v}_i\|_2 = 1$ for all $\boldsymbol{v}_i$, the set is **orthonormal**
- Square matrix $\boldsymbol{A}$ is orthogonal if its columns form a set of orthonormal vectors
  - Non-square matrices can be row- or column-orthogonal
- If $\boldsymbol{A}$ is orthogonal, then $\boldsymbol{A}^{-1} = \boldsymbol{A}^T$

Properties of orthogonal matrices

- The inverse of orthogonal matrices is easy to compute
- Orthogonal matrices perform a rotation
  - Only the angle of the vector is changed, the length stays the same

Matrix norms

- **Matrix norms** measure the magnitude of the matrix
  - the magnitude of the values or the image
- **Operator norms**:

  $||\boldsymbol{A}||_p = \max\{||\boldsymbol{Mx}||_p : ||\boldsymbol{x}||_p = 1\}$ for $p \geq 1$
- **Frobenius norm**:

  $\|\boldsymbol{A}\|_F = \left(\sum_{i=1}^{n} \sum_{j=1}^{m} a_{ij}^2\right)^{1/2}$

Singular Value Decomposition

Skillicorn Chapter 3; Golub & Van Loan Chapters 2.4–2.6, Leskovec et al. Chapter 11.3

“The SVD is the Swiss Army knife of matrix decompositions”

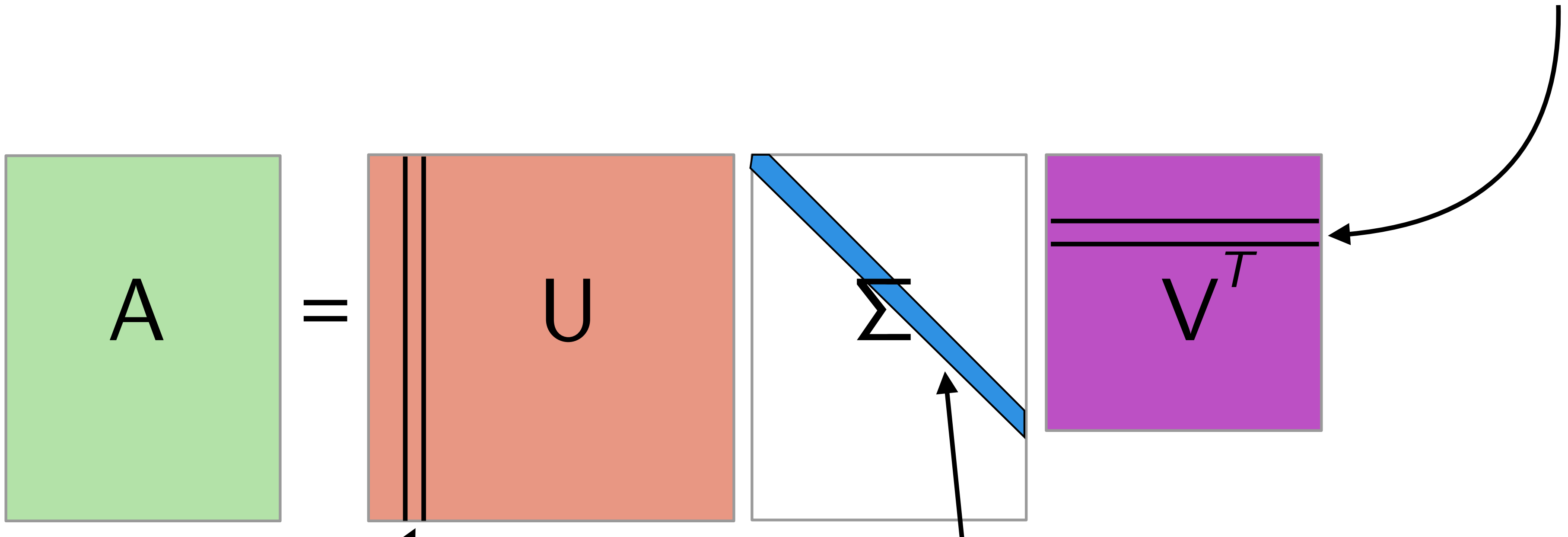
– Diane O’Leary, 2006

The definition

- **Theorem**. For every $\boldsymbol{A} \in \mathbb{R}^{n \times m}$ there exists an n-by-n *orthogonal* matrix $\boldsymbol{U}$ and an m-by-m *orthogonal* matrix $\boldsymbol{V}$ such that $\boldsymbol{U}^T\boldsymbol{A}\boldsymbol{V}$ is an n-by-m *diagonal* matrix $\boldsymbol{\Sigma}$ that has values $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_{\min\{n,m\}} \geq 0$ in its diagonal
  - I.e. every $\boldsymbol{A}$ has decomposition $\boldsymbol{A} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T$
  - The **singular value decomposition** of $\boldsymbol{A}$

In picture

$\boldsymbol{v}_i$ are the **right singular vectors**



σ_i are the **singular values**

$\boldsymbol{u}_i$ are the **left singular vectors**

Some useful equations

- $\boldsymbol{A} = \boldsymbol{U\Sigma V}^T = \sum_i \sigma_i \boldsymbol{u}_i \boldsymbol{v}_i^T$
  - Expresses $\boldsymbol{A}$ as a sum of rank-1 matrices
- $\boldsymbol{A}^{-1} = (\boldsymbol{U\Sigma V}^T)^{-1} = \mathbf{V}\boldsymbol{\Sigma}^{-1}\boldsymbol{U}^T$ (if $\boldsymbol{A}$ is invertible)
- $\boldsymbol{A}^T\boldsymbol{A}\boldsymbol{v}_i = \sigma_i^2\boldsymbol{v}_i$ (for any $\boldsymbol{A}$)
- $\boldsymbol{A}\boldsymbol{A}^T\boldsymbol{u}_i = \sigma_i^2\boldsymbol{u}_i$ (for any $\boldsymbol{A}$)

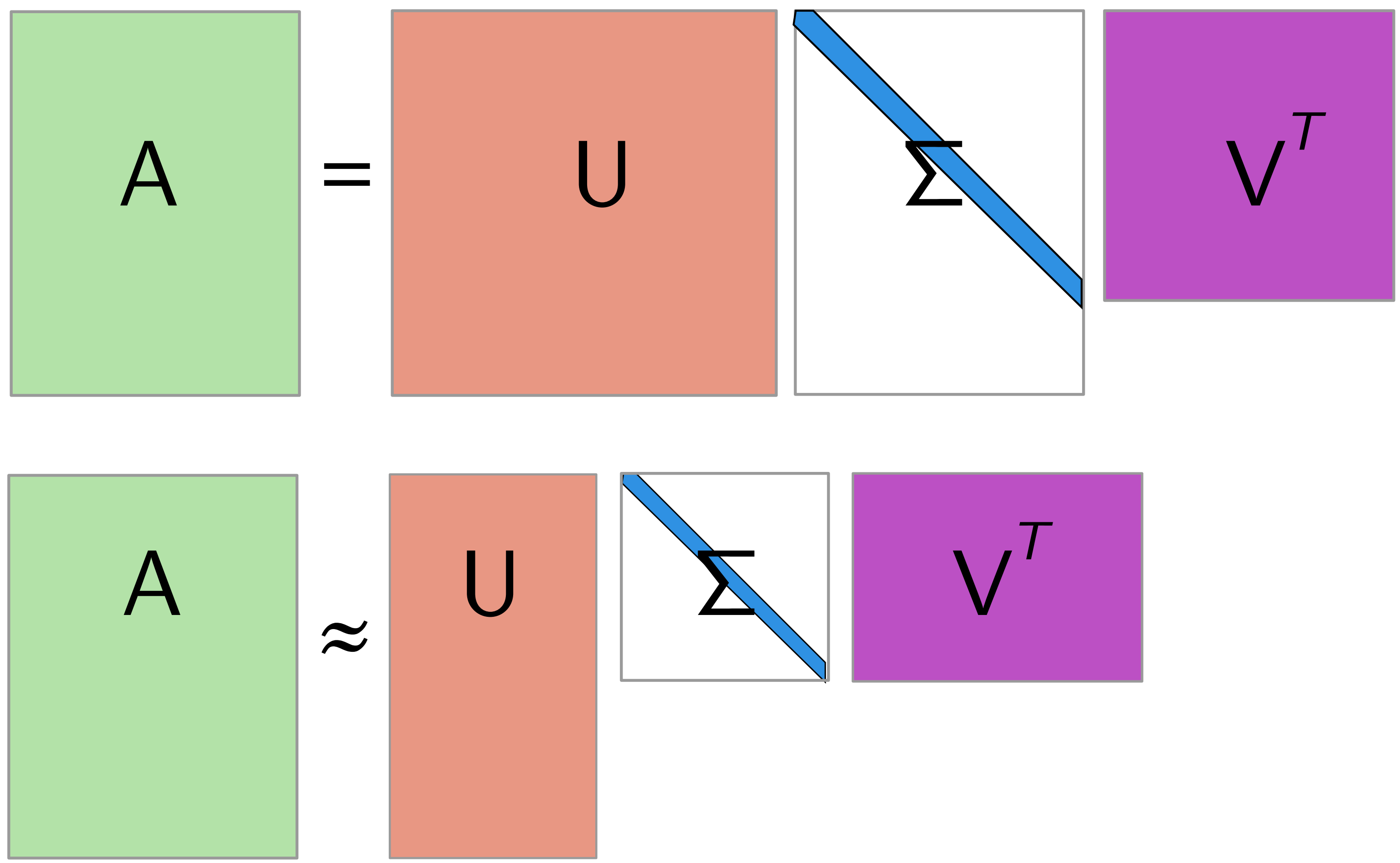
Truncated SVD

- The rank of the matrix is the number of its non-zero singular values (write $\boldsymbol{A} = \sum_i \sigma_i \boldsymbol{u}_i \boldsymbol{v}_i^T$)
- The **truncated SVD** takes the first k columns of $\boldsymbol{U}$ and $\boldsymbol{V}$ and the main k-by-k submatrix of $\boldsymbol{\Sigma}$
  - $\boldsymbol{A}_k = \boldsymbol{U}_k \boldsymbol{\Sigma}_k \boldsymbol{V}_k^T$
  - $\boldsymbol{U}_k$ and $\boldsymbol{V}_k$ are column-orthogonal

Truncated SVD

Full

Truncated



Why is SVD important?

- It gives us the **dimensions of the fundamental subspaces**
- It lets us **compute various norms**
- It tells about **sensitivity of linear systems**
- It gives us optimal solutions to **least-squares linear systems**
- It gives us the **least-error rank-*k* decomposition**
- **Every matrix has one**

SVD and norms

- Let $\boldsymbol{A} = \boldsymbol{U\Sigma V}^T$ be the SVD of $\boldsymbol{A}$.
  - $\|\boldsymbol{A}\|_F^2 = \sum_{i=1}^{\min\{n,m\}} \sigma_i^2$
  - $\|\boldsymbol{A}\|_2 = \sigma_1$
- Therefore $\|\boldsymbol{A}\|_2 \leq \|\boldsymbol{A}\|_F \leq \sqrt{\min\{n,m\}}\,\|\boldsymbol{A}\|_2$
- For truncated SVD, $\|\boldsymbol{A}_k\|_F^2 = \sum_{i=1}^{k} \sigma_i^2$

Sensitivity of linear systems

- The solution for system $\boldsymbol{Ax} = \boldsymbol{b}$ is $\boldsymbol{x} = \boldsymbol{A}^{-1}\boldsymbol{b}$
    - Requires that $\boldsymbol{A}$ is invertible
- Hence $\boldsymbol{x} = \left(\boldsymbol{U\Sigma V}^T\right)^{-1}\boldsymbol{b} = \sum_{i=1}^{n} \frac{\boldsymbol{u}_i^T\boldsymbol{b}}{\sigma_i}\boldsymbol{v}_i$
    - Small changes in $\boldsymbol{A}$ or $\boldsymbol{b}$ yield large changes in $\boldsymbol{x}$ if σ_n is small
    - Can we characterize this sensitivity?

Condition number

- The **condition number** $\kappa_p(\boldsymbol{A})$ of a square matrix $\boldsymbol{A}$ is $||\boldsymbol{A}||_p \; ||\boldsymbol{A}^{-1}||_p$
  - Particularly $\kappa_2(\boldsymbol{A}) = \sigma_1(\boldsymbol{A})/\sigma_n(\boldsymbol{A})$
    - $\kappa_2(\boldsymbol{A}) = \infty$ for singular $\boldsymbol{A}$
- If κ is large, the matrix is **ill-conditioned**
  - The solution is sensitive for small perturbations

Least-squares linear systems

- **Problem.** Given $\boldsymbol{A} \in \mathbb{R}^{n \times m}$ and $\boldsymbol{b} \in \mathbb{R}^{n}$, find $\boldsymbol{x} \in \mathbb{R}^{m}$ minimizing $||\boldsymbol{Ax} - \boldsymbol{b}||_2$.
- If $\boldsymbol{A}$ is invertible, $\boldsymbol{x} = \boldsymbol{A}^{-1}\boldsymbol{b}$ is an exact solution
- For non-invertible $\boldsymbol{A}$ we have to find other solution

The Moore–Penrose pseudo-inverse

- n-by-m matrix $\boldsymbol{B}$ is the **Moore–Penrose pseudo-inverse** of n-by-m matrix $\boldsymbol{A}$ if
  - $\boldsymbol{ABA} = \boldsymbol{A}$ (but possibly $\boldsymbol{AB} \neq \boldsymbol{I}$)
  - $\boldsymbol{BAB} = \boldsymbol{B}$
  - $(\boldsymbol{AB})^T = \boldsymbol{AB}$ ($\boldsymbol{AB}$ is symmetric)
  - $(\boldsymbol{BA})^T = \boldsymbol{BA}$
- Pseudo-inverse of $\boldsymbol{A}$ is denoted by $\boldsymbol{A}^+$

Pseudo-inverse and SVD

- If $\boldsymbol{A} = \boldsymbol{U\Sigma V}^T$ is the SVD of $\boldsymbol{A}$, then $\boldsymbol{A}^+ = \boldsymbol{V\Sigma}^{-1}\boldsymbol{U}^T$
  - $\boldsymbol{\Sigma}^{-1}$ replaces non-zero σ_i's with $1/\sigma_i$ and transposes the result
    - N.B. not a real inverse
- **Theorem**. Setting $\boldsymbol{x} = \boldsymbol{A}^+\boldsymbol{y}$ gives the optimal solution to $||\boldsymbol{Ax} - \boldsymbol{y}||$

The Eckart–Young theorem

- **Theorem.** Let $\boldsymbol{A}_k = \boldsymbol{U}_k\boldsymbol{\Sigma}_k\boldsymbol{V}_k^{\boldsymbol{T}}$ be the rank-k truncated SVD of $\boldsymbol{A}$. Then $\boldsymbol{A}_k$ is the closest rank-k matrix of $\boldsymbol{A}$ in the Frobenius sense, that is,

  $||\boldsymbol{A} - \boldsymbol{A}_k||_F \leq ||\boldsymbol{A} - \boldsymbol{B}||_F$ for all rank-k matrices $\boldsymbol{B}$

  - Holds for any unitarily invariant norm

Interpreting SVD

Skillicorn chapter 3.2

Factor interpretation

- Let $\boldsymbol{A}$ be objects-by-attributes and $\boldsymbol{U\Sigma V}^T$ its SVD
  - If two columns have similar values in a row of $\boldsymbol{V}^T$, these attributes are similar (have strong correlation)
  - If two rows have similar values in a column of $\boldsymbol{U}$, these objects are similar

Example

- Data: people's ratings on different wines
- Scatterplot of first two LSV
    - SVD doesn't know what the data is
- Conclusion: winelovers like red and white alike, others are more biased

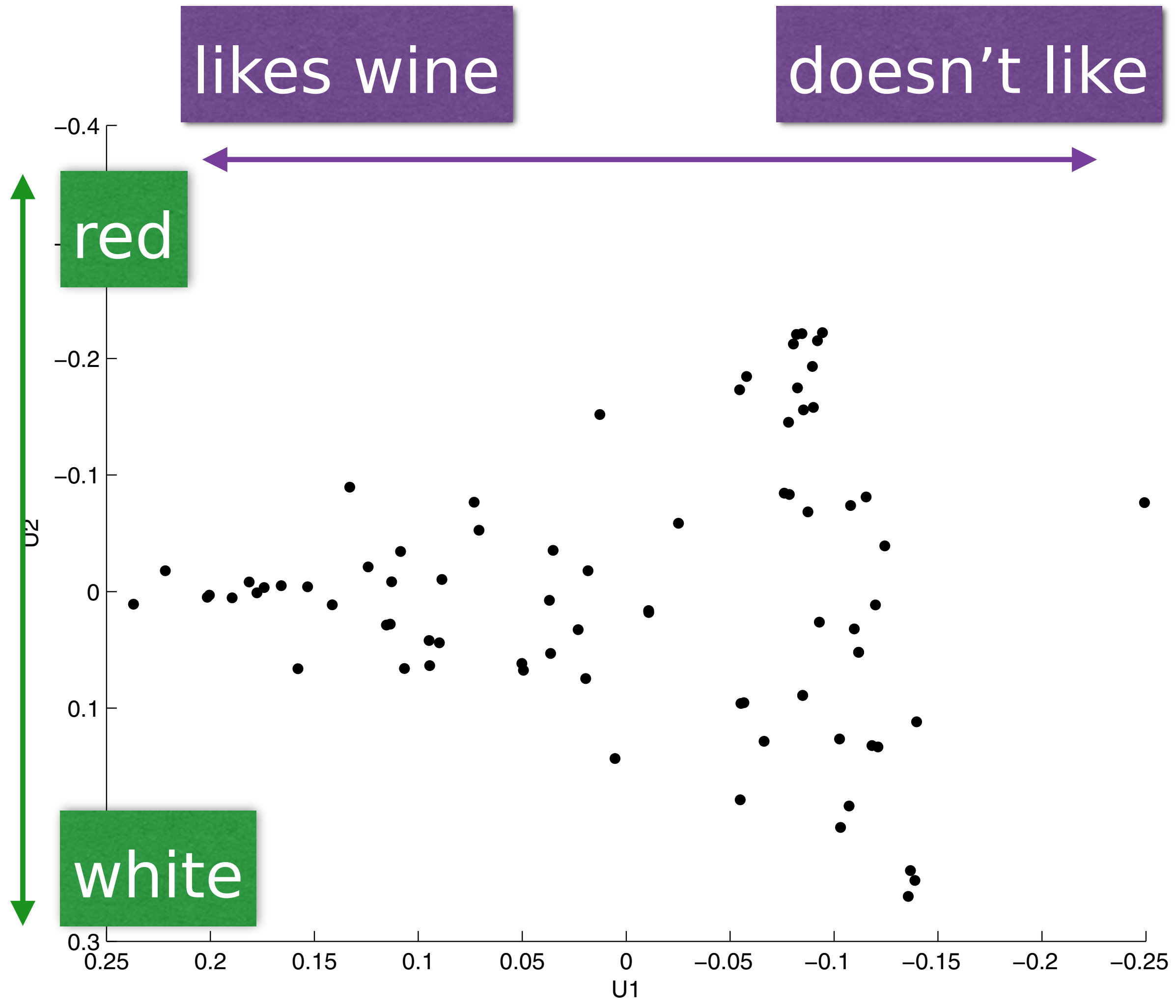
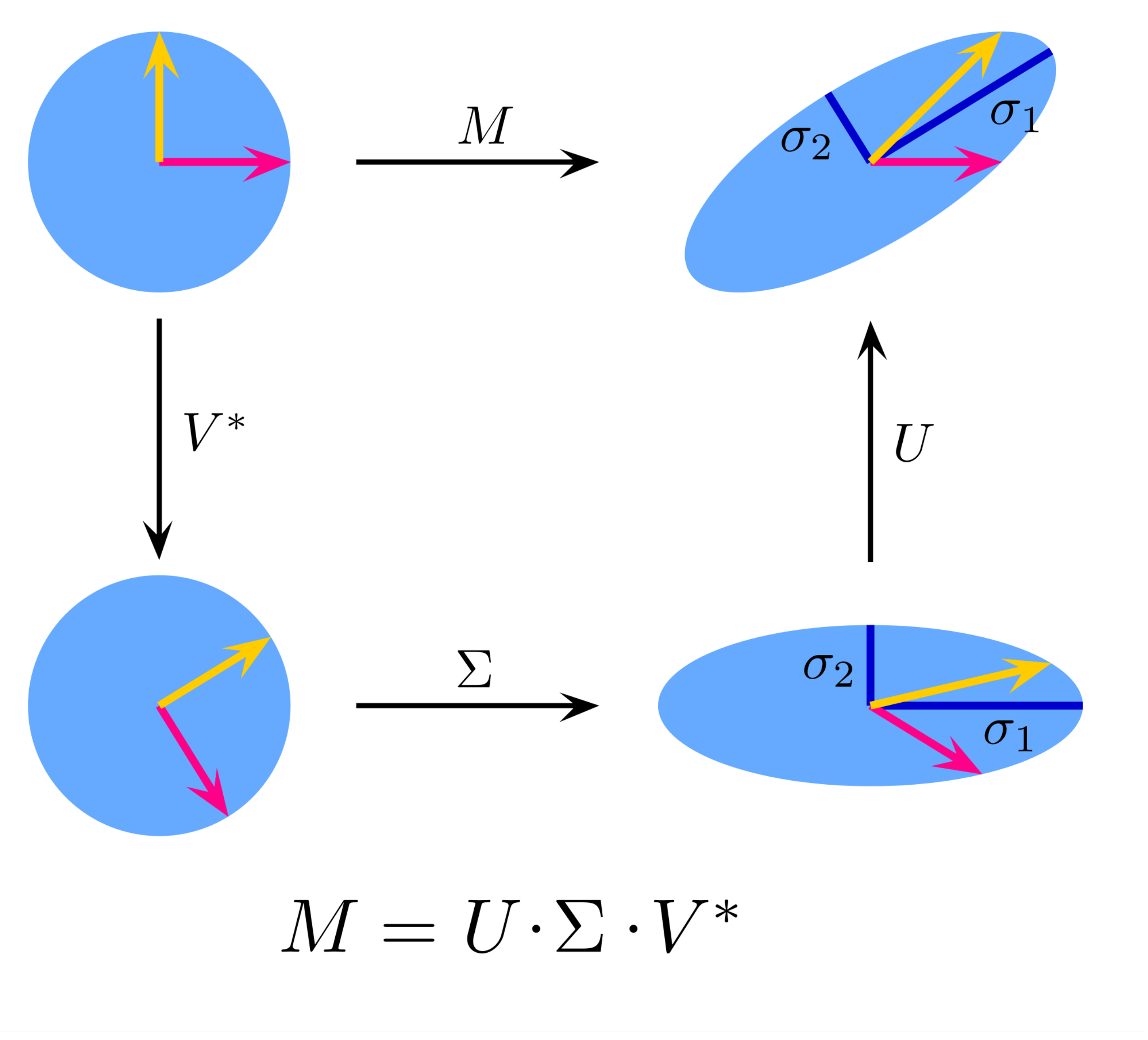


Figure 3.2. *The first two factors for a dataset ranking wines.*

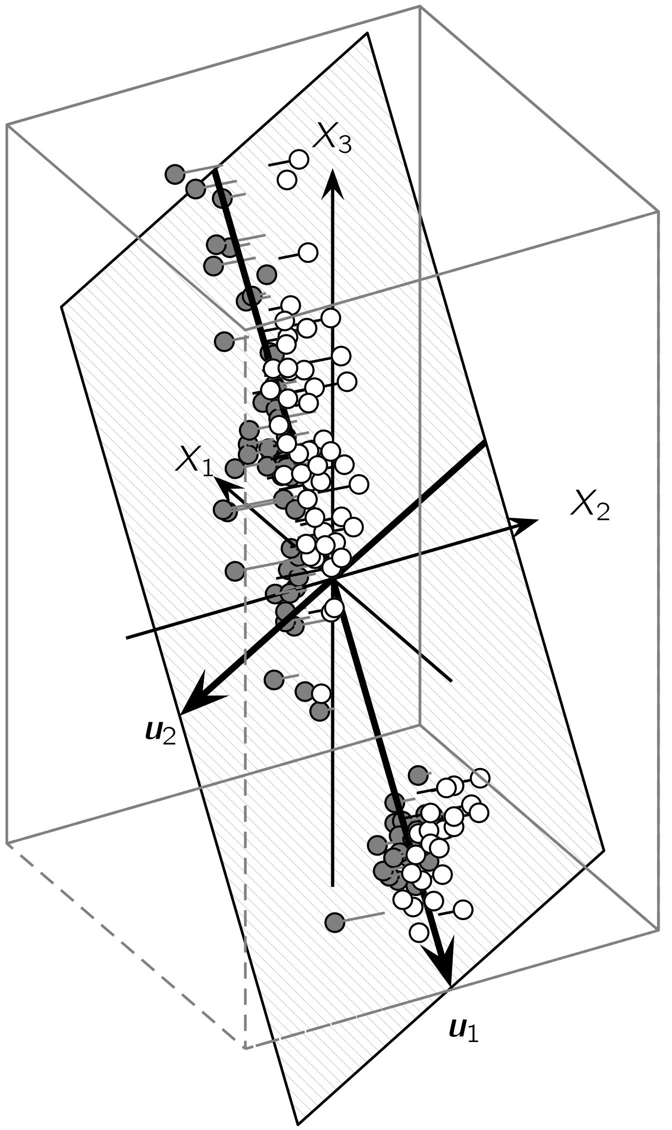
Geometric interpretation

- Let $\boldsymbol{M} = \boldsymbol{U\Sigma V}^T$
- Any linear mapping $\boldsymbol{y}=\boldsymbol{Mx}$ can be expressed as a rotation, stretching, and rotation operation
  - $\boldsymbol{y}_1 = \boldsymbol{V}^T\boldsymbol{x}$ is the first rotation
  - $\boldsymbol{y}_2 = \boldsymbol{\Sigma y}_1$ is the stretching
  - $\boldsymbol{y} = \boldsymbol{U y}_2$ is the final rotation



Direction of largest variances

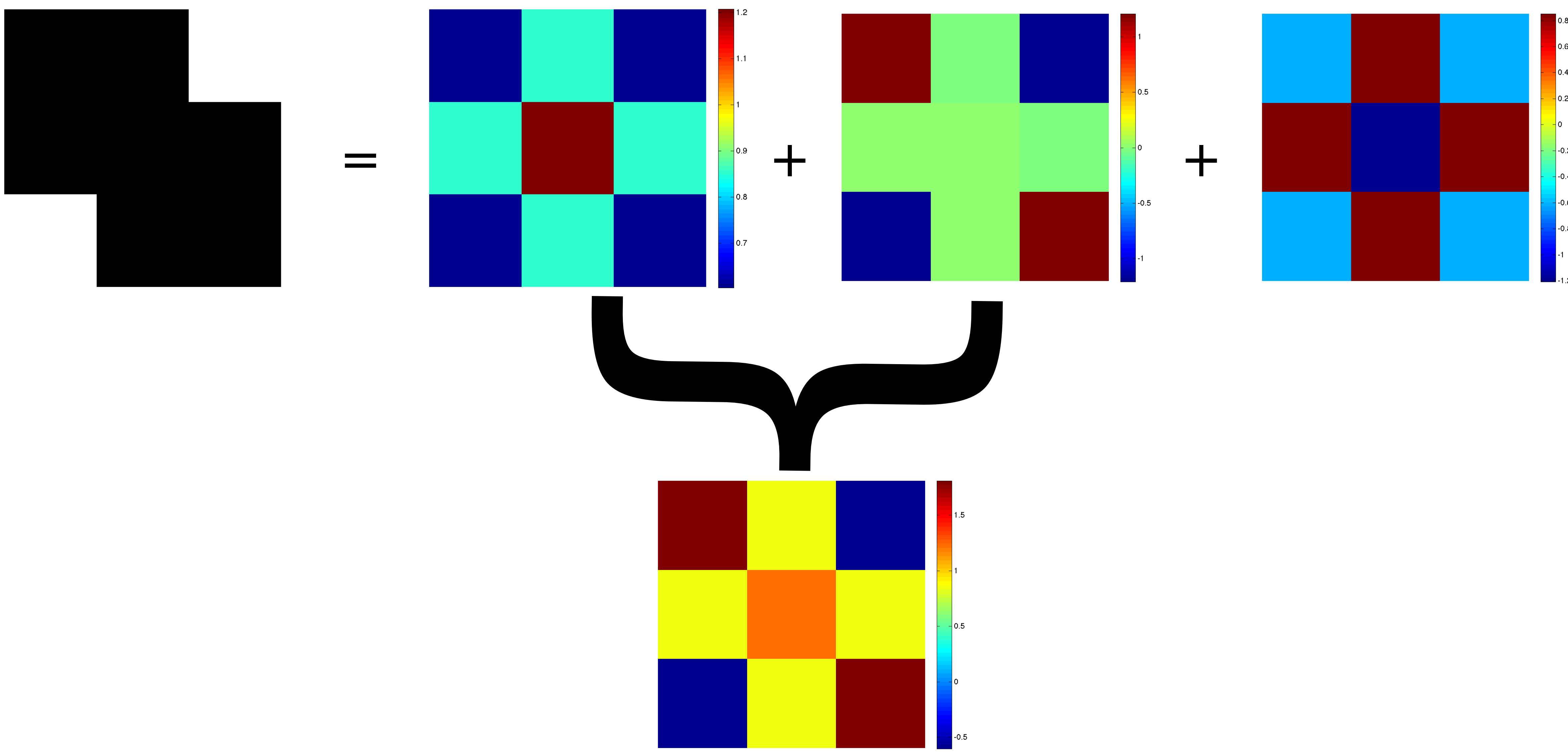
- The singular vectors give the directions of the largest variances
  - First singular vector points to the direction of the largest variance
  - Second to the second-largest
    - Spans a hyperplane with the first
- The projection distance to these hyperplanes is minimal over all hyperplanes (Eckart–Young)



Component interpretation

- We can write $\boldsymbol{A} = \boldsymbol{U\Sigma V}^T = \sum_i \sigma_i \boldsymbol{u}_i \boldsymbol{v}_i^T = \sum_i \boldsymbol{A}_i$
- This explains the data as a sum of rank-1 layers
  - First layer explains the most, the second updates that, the third updates that, …
- Each individual layer don’t have to be very intuitive

Example

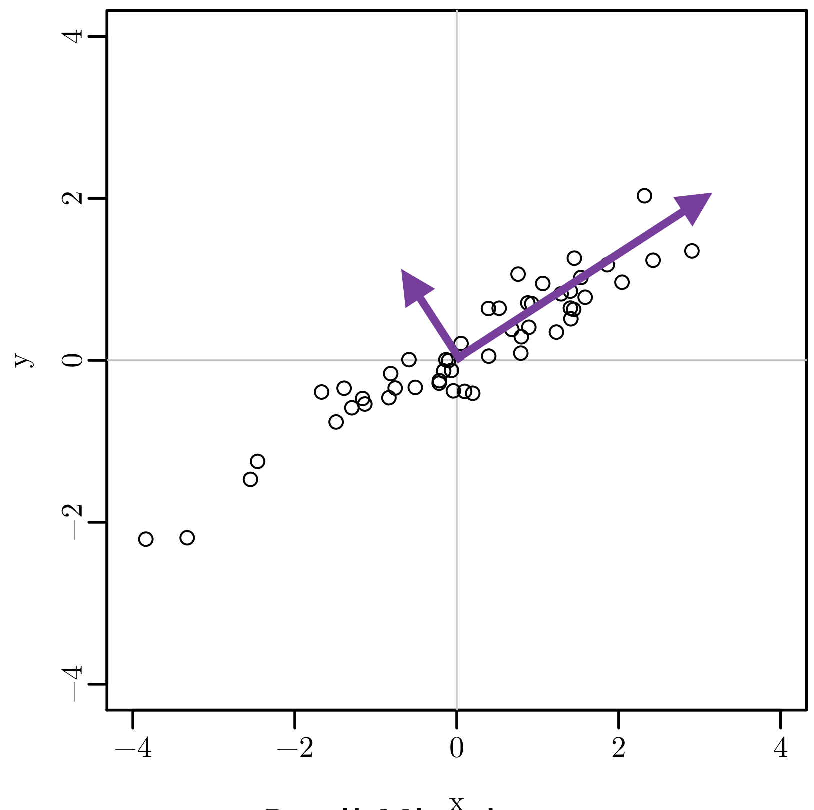


Applications of SVD

Skillicorn chapter 3.5; Leskovec et al. chapter 11.3

Removing noise

- SVD is often used as a pre-processing step to remove noise from the data
  - The rank-*k* truncated SVD with proper *k*



$\sigma_1 = 11.73$

$\sigma_2 = 1.71$

Removing dimensions

- SVD can be used to project the data to smaller-dimensional subspace
  - Original dimensions can have complex correlations
  - Subsequent analysis is faster
  - Points seem close to each other in high-dimensional space

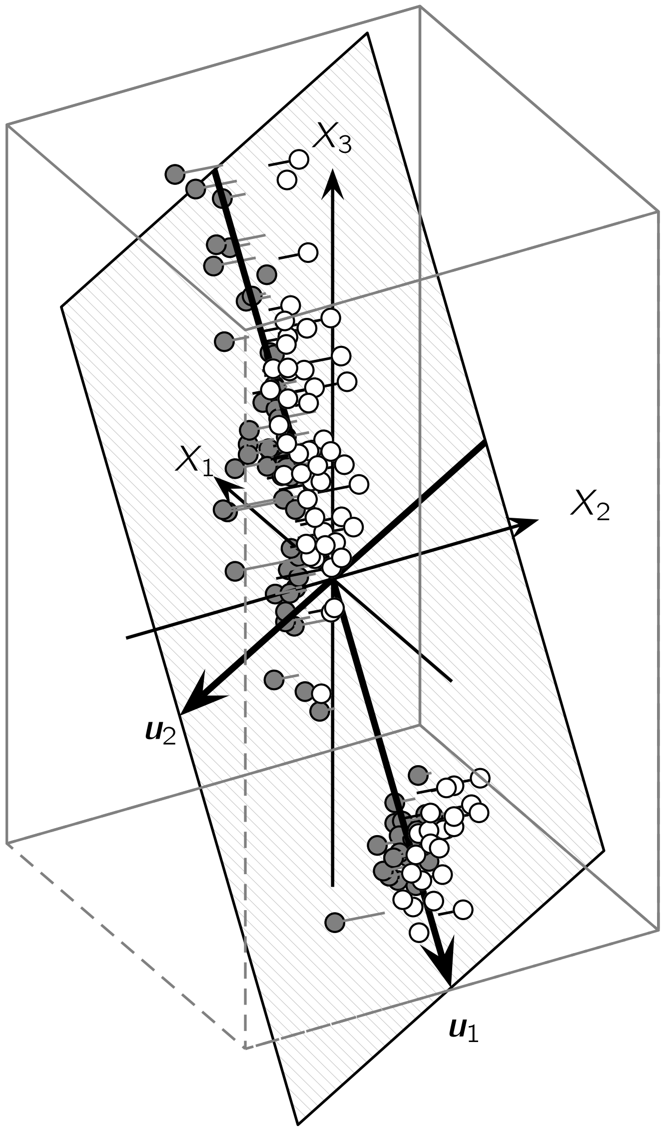
Curse of dimensionality

Karhunen–Loève transform

- The **Karhunen–Loève transform** (KLT) works as follows:
  - Normalize $\boldsymbol{A} \in \mathbb{R}^{n \times m}$ to z-scores
  - Compute the SVD $\boldsymbol{U\Sigma V}^T = \boldsymbol{A}$
  - Project $\boldsymbol{A} \mapsto \boldsymbol{AV}_k \in \mathbb{R}^{n \times k}$
    - $\boldsymbol{V}_k$ = top-k right singular vectors
- A.k.a. the **principal component analysis** (PCA)

More on KLT

- The columns of $\boldsymbol{V}_k$ show the main directions of variance in columns
- The data is expressed in a new coordinate system
- The average projection distance is minimized



Visualization

Scatter plots

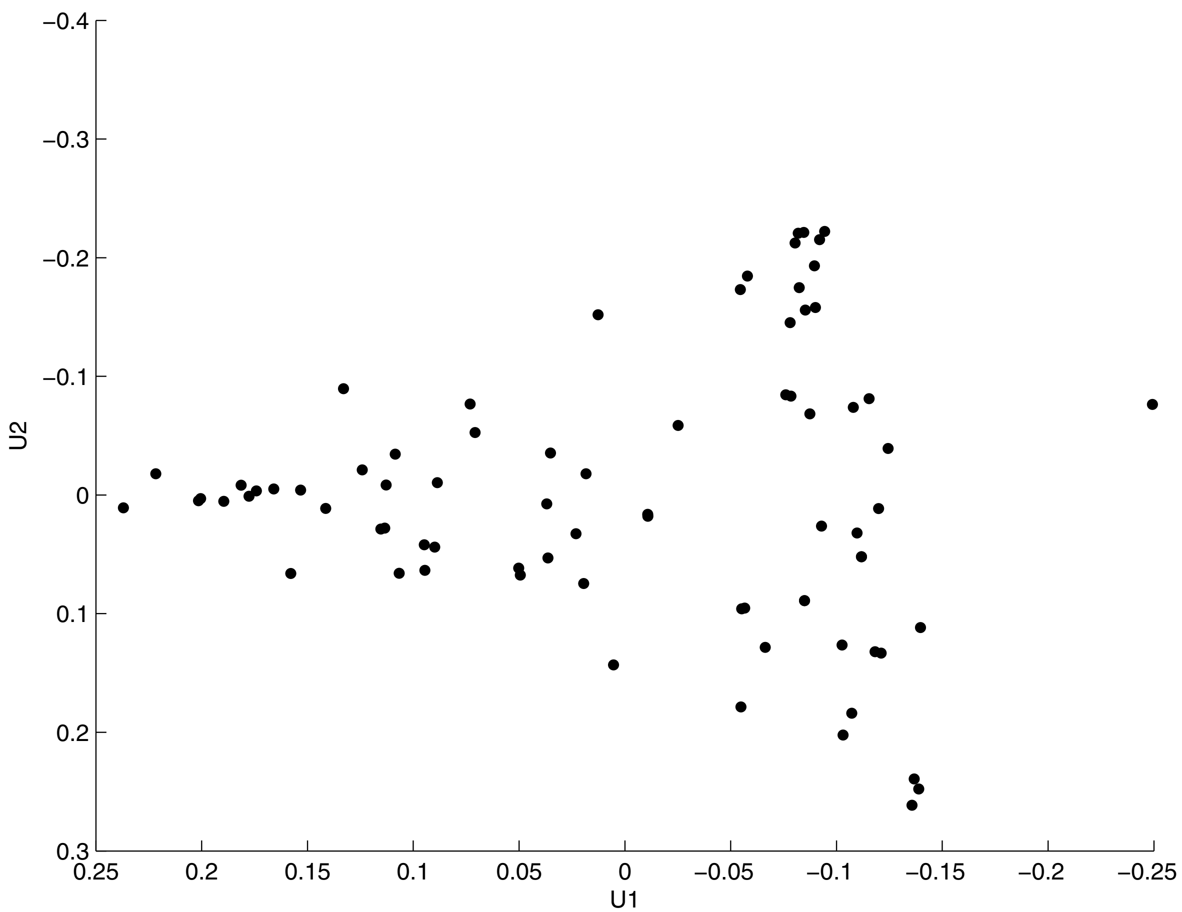
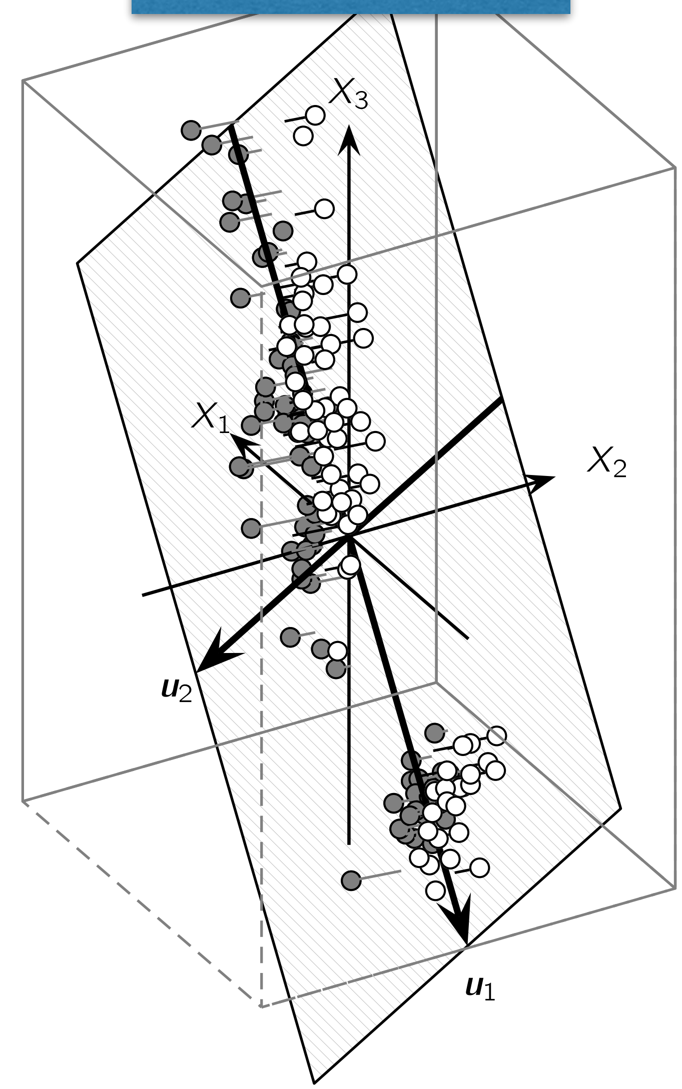


Figure 3.2. *The first two factors for a dataset ranking wines.*

2D or 3D KLT



Latent Semantic Analysis & Indexing

- **Latent semantic analysis** (LSA) is a **latent topic model**
  - Documents-by-terms matrix $\boldsymbol{A}$
    - Typically normalized (e.g. tf/idf)
- Goal is to find the "topics" doing SVD
  - $\boldsymbol{U}$ associates documents to topics
  - $\boldsymbol{V}$ associates topics to terms
- Queries can be answered by projecting the query vector $\boldsymbol{q}$ to $\boldsymbol{q}' = \boldsymbol{q}\boldsymbol{V}\boldsymbol{\Sigma}^{-1}$ and returning rows of $\boldsymbol{U}$ that are similar to $\boldsymbol{q}'$

And many more…

- Determining the rank, finding the least-squares solution, recommending the movies, ordering results of queries, …
- Next week: and how do we compute this SVD, again? *Stay tuned!*