Chapter 1

SVD, PCA & Pre-processing

Part 2: Pre-processing and selecting the rank

max planck institut
informatik

Pre-processing

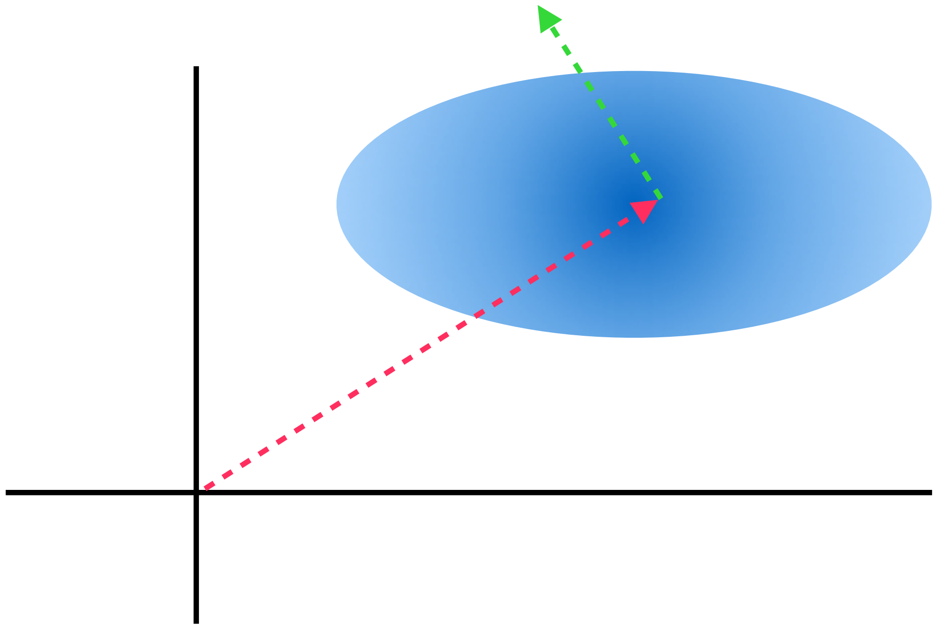
Skillicorn chapter 3.1

Why pre-process?

- Consider matrix of weather data
  - Monthly temperatures in degrees Celsius
    - Typical range [–20, +25]
  - Monthly precipitation in millimeters
    - Typical range [0, 100]
- Precipitation seems much more important
  - But what if the temperatures where in degrees Kelvin?
    - The range is now [250, 300]

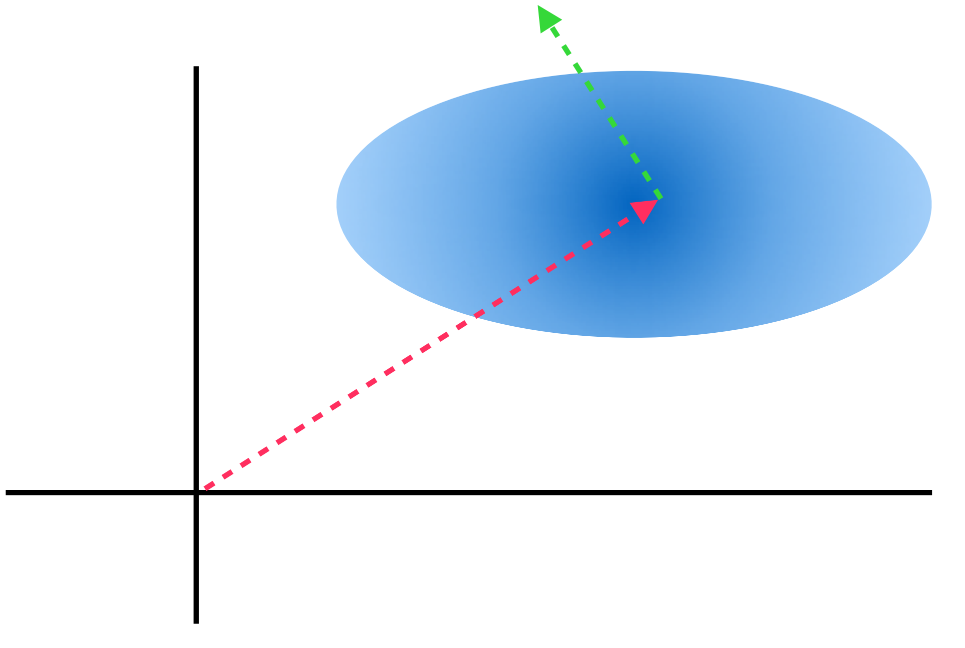
Why pre-process

- If $\boldsymbol{A}$ is nonnegative, the first singular vector just shows where the average of $\boldsymbol{A}$ is
  - The remaining vectors still have to be orthogonal to the first



Why pre-process

- If ***A*** is centered to the origin, the singular vectors show the directions of variance in ***A***
  - This is the basis of KLT/PCA…



The z-scores

- The z-**scores** are attributes whose values are transformed by
  - centering them to 0 by removing the (column) mean from each value
  - normalizing the magnitudes by dividing every value with the (column) standard deviation

$$X' = \frac{X - \mu}{\sigma}$$

When z-scores?

- Attribute values are approximately normally distributed, c.f. $X' = \frac{X-\mu}{\sigma}$
- All attributes are equally important
- Data does not have any important structure that is destroyed
  - Non-negativity, sparsity, integer values, …

Other normalizations

- Large values can be reduced in importance by
  - taking logarithms (from positive values)
  - taking cubic roots
- Sparsity can be preserved by only considering non-zero values
- **The effects of normalization must always be considered**

Selecting the rank

Skillicorn chapter 3.3

How many factors?

- Assume we want to compute rank-k truncated SVD to analyze some data
- But how to select the k?
  - Too big, and we have to handle unimportant factors
  - Too small, and we loose important structure
- So we need a way to select a good k

Guttman–Kaiser criterion and captured energy

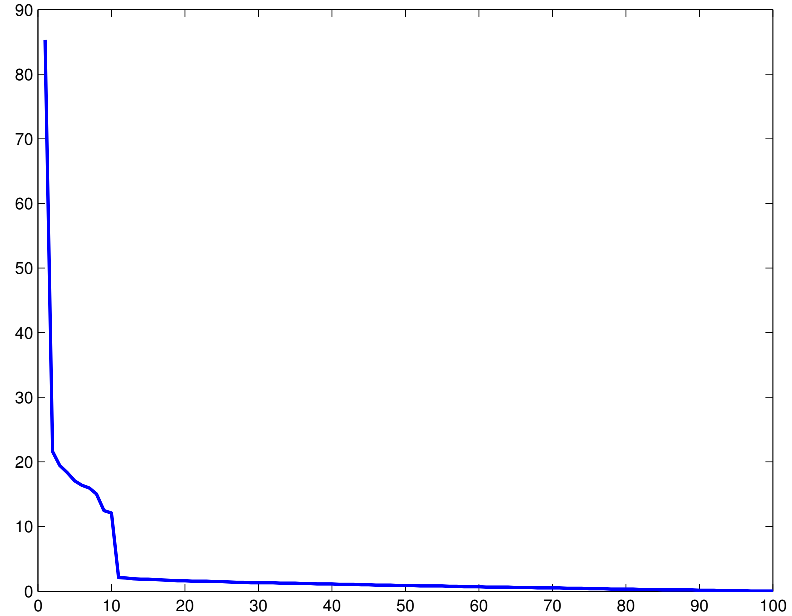
- **Method 1**: select k s.t. for all $i > k$, $\sigma_i < 1$
  - Motivation: components with singular values < 1 are uninteresting
- **Method 2**: select smallest k s.t.

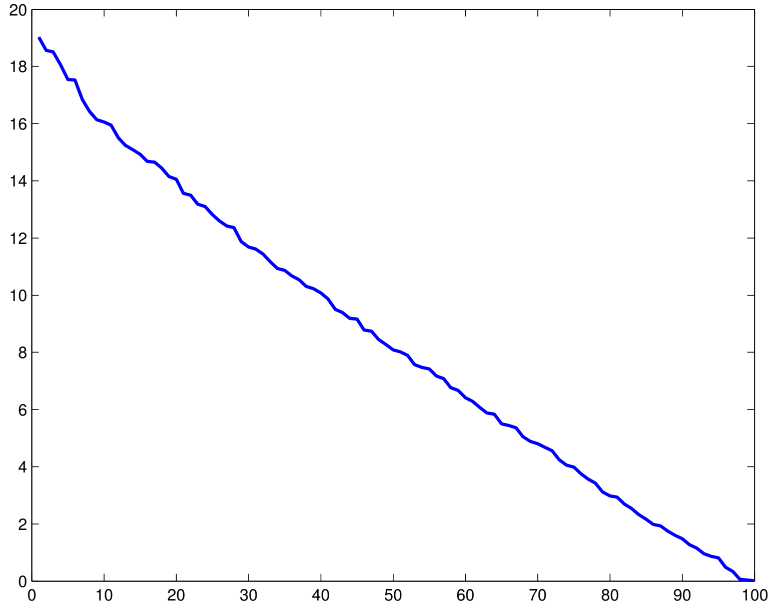
$$\sum_{i=1}^{k} \sigma_i^2 \geq 0.9 \sum_{i=1}^{\min\{n,m\}} \sigma_i^2$$

  - Motivation: this explains 90% of the Frobenius norm (a.k.a. energy)
- Both methods are based on arbitrary thresholds

Cattell's Scree test

- The **scree plot** has the singular values plotted in decreasing order
- In scree test, the rank is selected s.t. in the plot
  - there is a clear drop in the magnitudes; or
  - the singular values start to even out





Entropy-based method

- The **relative contribution** of σ_k is $r_k = \sigma_k^2 / \sum_i \sigma_i^2$
- The **entropy** E of singular values is

$$E = -\frac{1}{\log(\min\{n,m\})} \sum_{i=1}^{\min\{n,m\}} r_i \log r_i$$

$0 \cdot \infty = 0$

- Set the rank to the smallest k s.t. $\sum_{i=1}^{k} r_i \geq E$
- Intuition: low entropy = the mass of the singular values is packed to the begin

Random flip of signs

- Consider a random matrix ***A***′ created by multiplying every element of ***A*** by 1 or –1 u.a.r.
  - The Frobenius norm doesn't change, but the spectral norm does change
  - How much the spectral norm changes depends on the amount of "structure" in ***A***
- Idea: use this to select *k* that isolates the structure from the noise

Using random flips

- The **residual matrix** $\mathbf{A}_{-k}$ is

$$\boldsymbol{A}_{-k} = \boldsymbol{A} - \boldsymbol{A}_k = \boldsymbol{U}_{-k}\boldsymbol{\Sigma}_{-k}\boldsymbol{V}_{-k}^T$$

  - $\boldsymbol{U}_{-k}$ ($\boldsymbol{V}_{-k}$) contains the last $n - k$ ($m - k$) left (right) singular vectors
- Let $\boldsymbol{A}_{-k}$ be the residual of $\boldsymbol{A}$ and $\boldsymbol{A}'_{-k}$ that of $\boldsymbol{A}'$
- Select k s.t. $|\ ||\boldsymbol{A}_{-k}||_2 - ||\boldsymbol{A}'_{-k}||_2\ |\ /\ ||\boldsymbol{A}_{-k}||_F$ is small
  - On average, over multiple random matrices

Issues with the methods

- Require computing the full SVD first or otherwise computationally heavy

Guttman–Kaiser scree entropy-based random flips

- Require subjective evaluation

scree random flips

- Based on arbitrary thresholds

Guttman–Kaiser entropy-based

Summary

- Pre-processing can make all the difference
  - Often overlooked
- Selecting the rank is non-trivial
  - Guttman–Kaiser and scree test are often used in other fields

Chapter 1

SVD, PCA & Pre-processing

Part 3: Computing the SVD

max planck institut
informatik

Computing the SVD

Golub & Van Loan chapters 5.1, 5.4.8, and 8.6

Very general idea

- SVD is unique
  - If $\boldsymbol{U}$ and $\boldsymbol{V}$ are orthogonal s.t. $\boldsymbol{U}^T\boldsymbol{A}\boldsymbol{V} = \boldsymbol{\Sigma}$, then $\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T$ is the SVD of $\boldsymbol{A}$
- Idea: find orthogonal $\boldsymbol{U}$ and $\boldsymbol{V}$ s.t. $\boldsymbol{U}^T\boldsymbol{A}\boldsymbol{V}$ is as desired
  - Iterative process: find orthogonal $\boldsymbol{U}_1$, $\boldsymbol{U}_2$, … and set $\boldsymbol{U} = \boldsymbol{U}_1\boldsymbol{U}_2\boldsymbol{U}_3\cdots$
    - Still orthogonal

Rotations and reflections

2D **rotation**

$$\begin{pmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{pmatrix}$$

Rotates counterclockwise through an angle θ

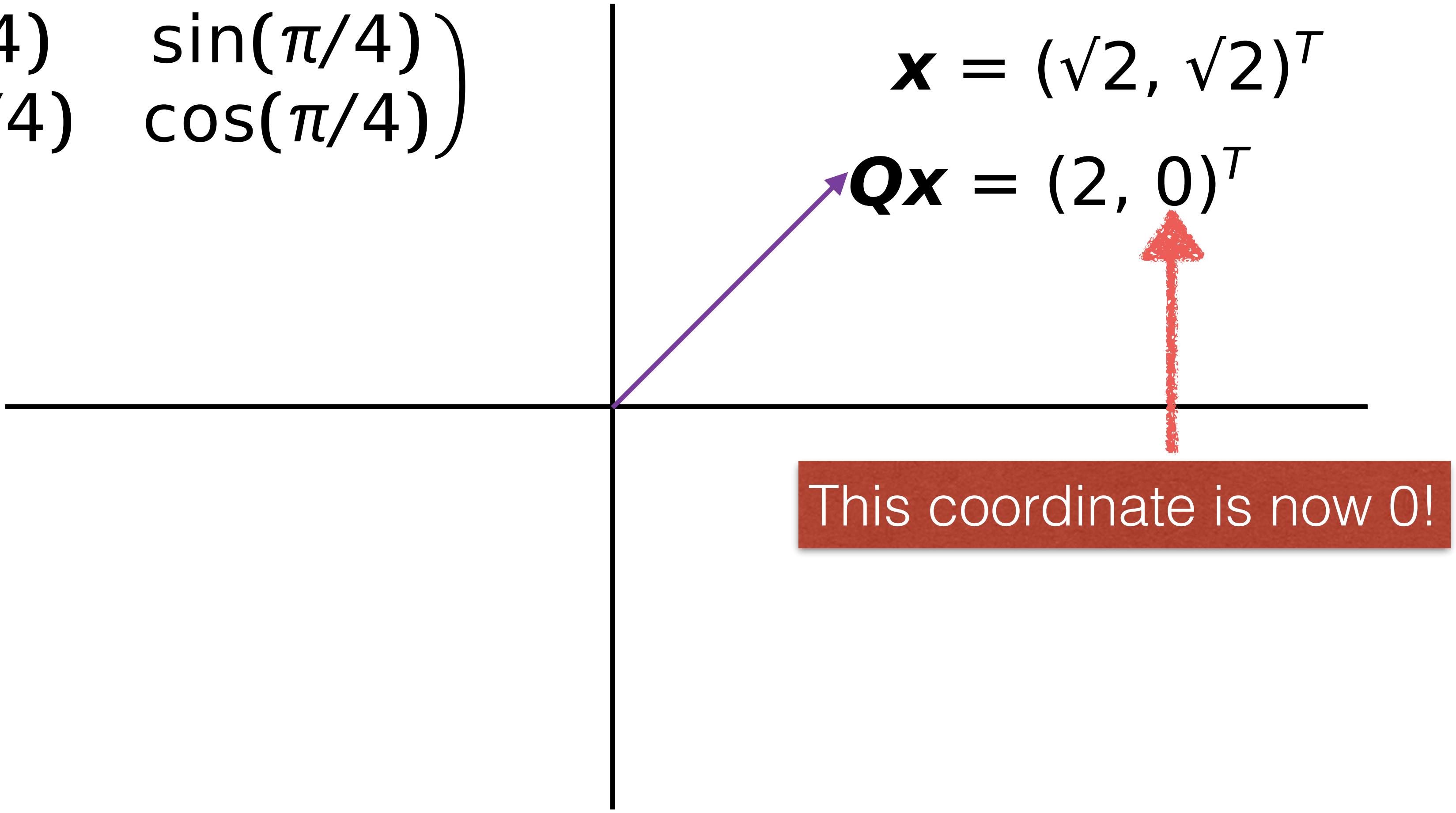
2D **reflection**

$$\begin{pmatrix} \cos(\theta) & \sin(\theta) \\ \sin(\theta) & -\cos(\theta) \end{pmatrix}$$

Reflects across the line spanned by $(\cos(\theta/2), \sin(\theta/2))^T$

Example

$$\boldsymbol{Q} = \begin{pmatrix} \cos(\pi/4) & \sin(\pi/4) \\ -\sin(\pi/4) & \cos(\pi/4) \end{pmatrix}$$



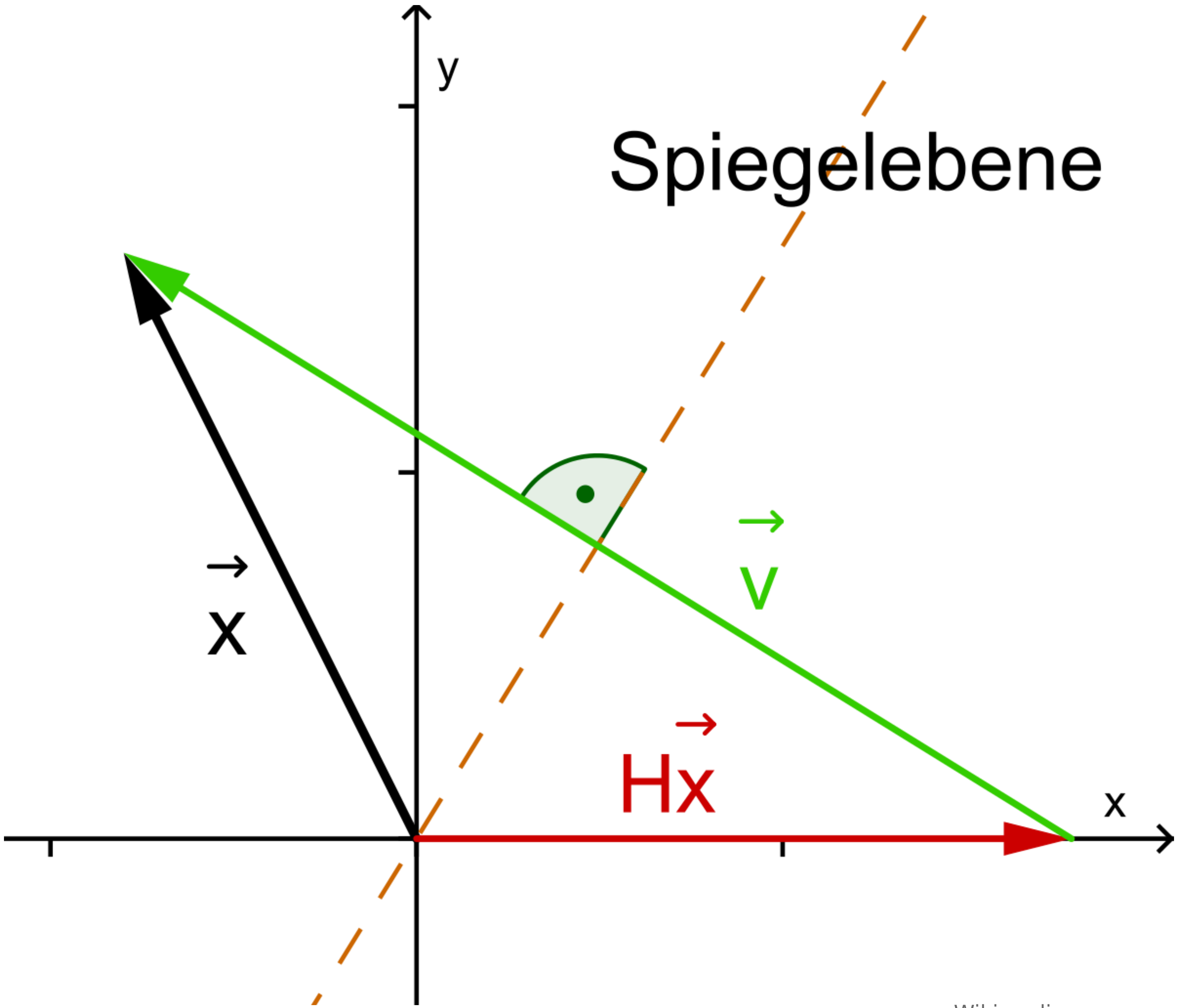
Householder reflections

- A **Householder reflection** is n-by-n matrix

$$\boldsymbol{P} = \boldsymbol{I} - \beta \boldsymbol{v}\boldsymbol{v}^T \quad \text{where} \quad \beta = \frac{2}{\boldsymbol{v}^T\boldsymbol{v}}$$

- If we set $\boldsymbol{v} = \boldsymbol{x} - ||\boldsymbol{x}||_2\boldsymbol{e}_1$, then $\boldsymbol{Px} = ||\boldsymbol{x}||_2\boldsymbol{e}_1$
  - $\boldsymbol{e}_1 = (1, 0, 0, \ldots, 0)^T$
- Note: $\boldsymbol{PA} = \boldsymbol{A} - (\beta\boldsymbol{v})(\boldsymbol{v}^T\boldsymbol{A})$ where $\beta = 2/(\boldsymbol{v}^T\boldsymbol{v})$
  - We never have to compute matrix $\boldsymbol{P}$

Example



Wikimedia commons

Almost there: bidiagonalization

- Given n-by-m ($n \geq m$) $\boldsymbol{A}$, we can **bidiagonalize** it with Householder transformations
  - Fix $\boldsymbol{A}[1{:}n,1]$, $\boldsymbol{A}[1,2{:}m]$, $\boldsymbol{A}[2{:}n,2]$, $\boldsymbol{A}[2,3{:}m]$, A[3:n,3], A[3,4:m]…
- The results has non-zeros in main diagonal and the one above it

Example

$$\boldsymbol{U}_4^T \boldsymbol{U}_3^T \boldsymbol{U}_2^T \boldsymbol{U}_1^T \boldsymbol{A} \boldsymbol{V}_1 \boldsymbol{V}_2 = \begin{pmatrix} * & * & 0 & 0 \\ 0 & * & * & 0 \\ 0 & 0 & * & * \\ 0 & 0 & 0 & * \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

Givens rotations

- Householder is too crude to give identity
- **Givens rotations** are rank-2 corrections to the identity of form

$$
\boldsymbol{G}(i,k,\theta) = \begin{pmatrix}
1 & \cdots & 0 & \cdots & 0 & \cdots & 0 \\
\vdots & \ddots & \vdots & & \vdots & & \vdots \\
0 & \cdots & \cos(\theta) & \cdots & \sin(\theta) & \cdots & 0 \\
\vdots & & \vdots & \ddots & \vdots & & \vdots \\
0 & \cdots & -\sin(\theta) & \cdots & \cos(\theta) & \cdots & 0 \\
\vdots & & \vdots & & \vdots & \ddots & \vdots \\
0 & \cdots & 0 & \cdots & 0 & \cdots & 1
\end{pmatrix}
\begin{matrix} \\ \\ i \\ \\ k \\ \\ \\ \end{matrix}
$$

$$\qquad\qquad\qquad\qquad i \qquad\qquad\qquad k$$

Applying Givens

- Set θ s.t.

  $\cos(\theta) = \frac{x_i}{\sqrt{x_i^2 + x_k^2}}$ and $\sin(\theta) = \frac{-x_k}{\sqrt{x_i^2 + x_k^2}}$

- Now

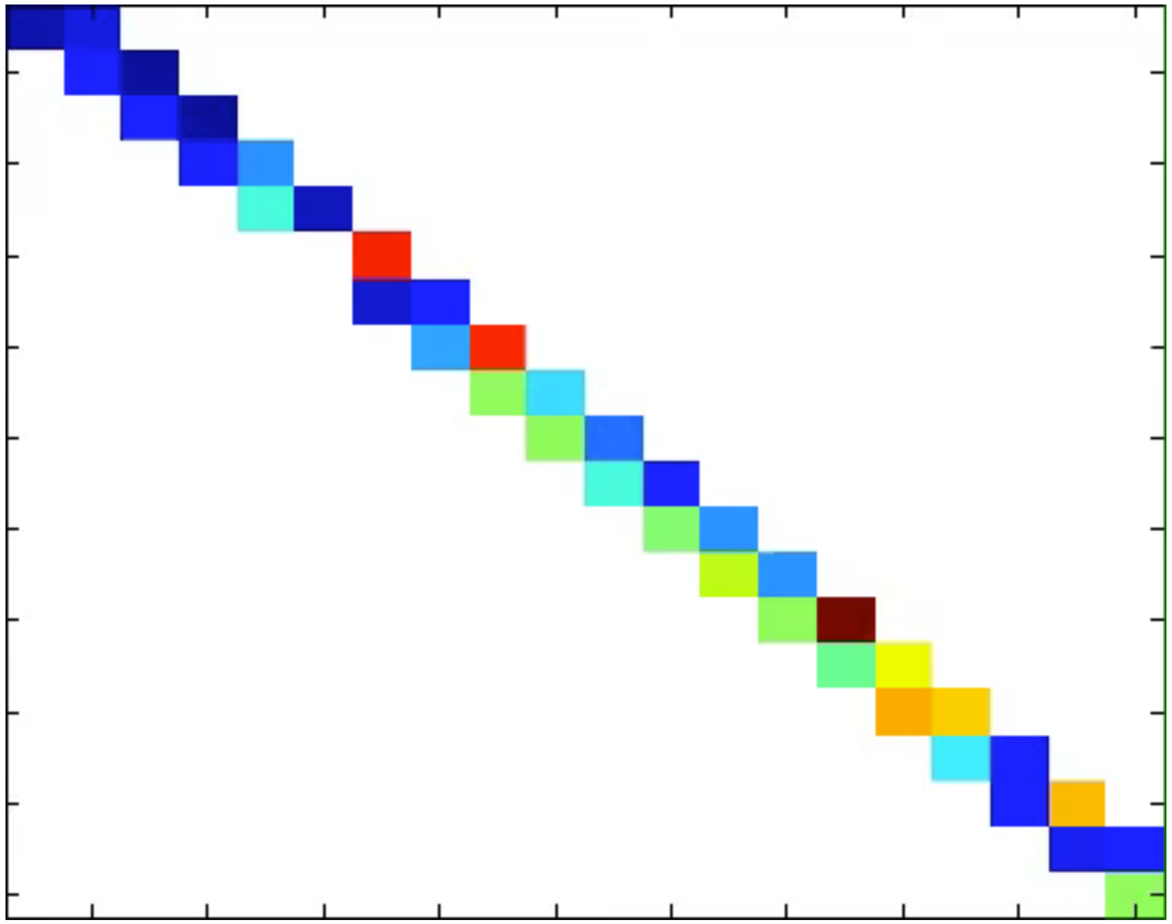
$$\begin{pmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{pmatrix}^T \begin{pmatrix} x_i \\ x_k \end{pmatrix} = \begin{pmatrix} r \\ 0 \end{pmatrix}$$

- N.B. $\boldsymbol{G}(i, k, \theta)^T \boldsymbol{A}$ only affects to the 2 rows $\boldsymbol{A}[\mathrm{c}(i, k),]$
  - Also, no inverse trig. operations are needed

Givens in SVD

- We use Givens transformations to erase the superdiagonal
  - Consider principal 2-by-2 submatrices $\boldsymbol{A}[k{:}k+1, k{:}k+1]$
  - Rotations can introduce unwanted non-zeros to $\boldsymbol{A}[k+2, k]$ (or $\boldsymbol{A}[k, k+2]$)
    - Fix them in the next sub-matrix

Example



Putting it all together

1. Compute the bidiagonal matrix $\boldsymbol{B}$ from $\boldsymbol{A}$ using Householder transformations
2. Apply the Givens rotations to $\boldsymbol{B}$ until it is fully diagonal
3. Collect the required results

Time complexity

Output	Time
Σ	$4nm^2 - 4m^3/3$
Σ, V	$4nm^2 + 8m^3$
Σ, U	$4n^2m - 8nm^2$
Σ, U_1	$14nm^2 - 2m^3$
Σ, U, V	$4n^2m + 8nm^2 + 9m^3$
Σ, U_1, V	$14nm^2 + 8m^3$

Summary of computing SVD

- Rotations and reflections allow us to selectively zero elements of a matrix with orthogonal transformations
  - Used in many, many decompositions
- Fast and accurate results require careful implementations
- Other techniques are faster for truncated SVD in large, sparse matrices

Summary of SVD

- Truly the workhorse of numerical linear algebra
  - Many useful theoretical properties
    - Rank-revealing, pseudo-inverses, scalar norm computation, …
  - Reasonably easy to compute
- But it also has some major shortcomings in data analysis… *stay tuned!*