Martin Mundt, Dr. Iuliia Pliushch, Prof. Dr. Visvanathan Ramesh

Pattern Analysis & Machine Intelligence Praktikum: MLPR-SS21

Week 7: K-Means, PCA and ICA

Unsupervised learning: K-Means, PCA and ICA

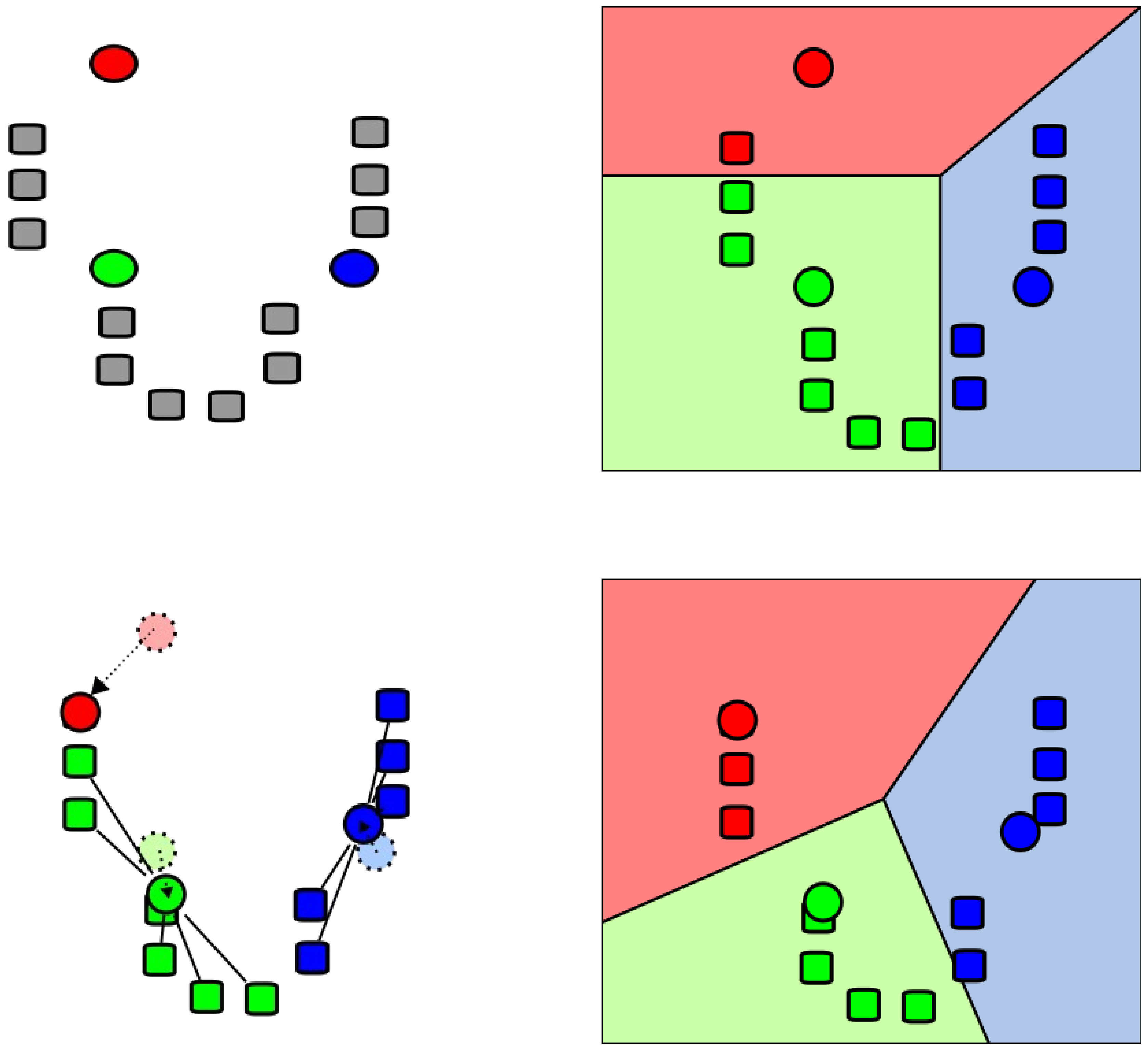
- **K-means**: clustering algorithm which operates on the distances between points and the supposed cluster centers

- **PCA**: helps in analyzing the data variance and can be used for **data compression**

- **ICA:** separating a multivariate signal into non-Gaussian, **independent** subcomponents

K-Means clustering (Lloyd algorithm)

- Input: d-dimensional data points
- Randomly initialize k cluster means
- Assign points to its closest cluster mean
- Update the cluster means and repeat the two previous steps until the means converge



https://de.coursera.org/lecture/genomic-data/the-lloyd-algorithm-for-k-means-clustering-3O9eh

https://de.wikipedia.org/wiki/K-Means-Algorithmus

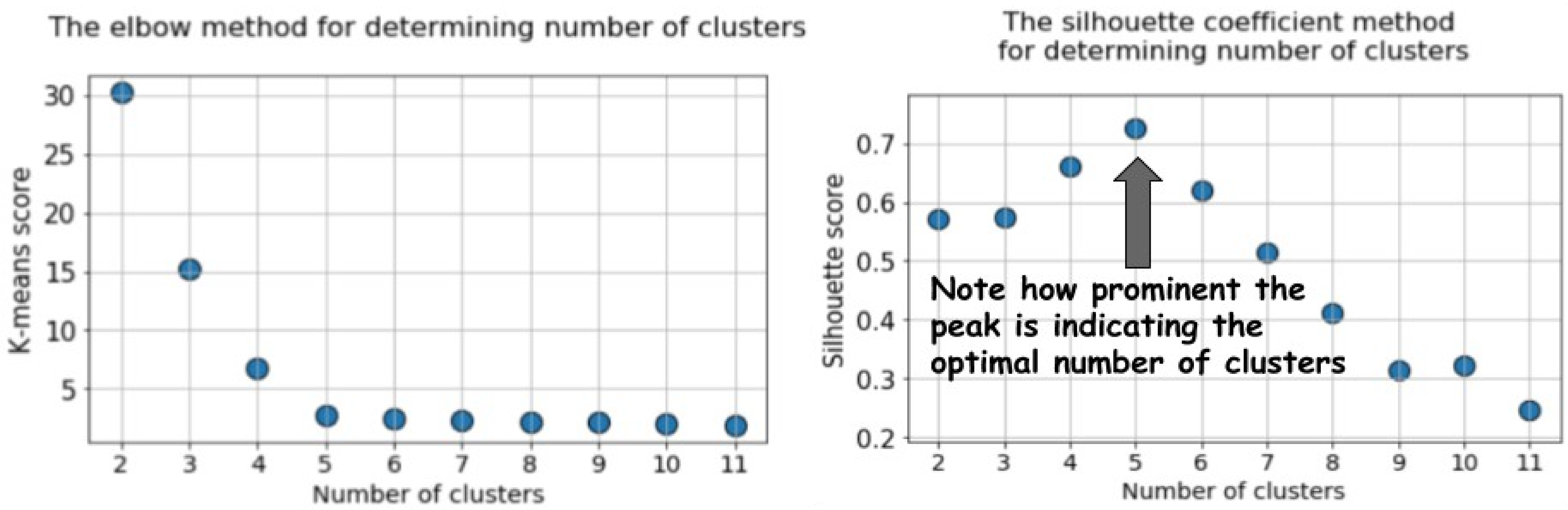
K-Means: Difficulties

- How do we determine **k – ** the number of clusters to split the data into?

Elbow	Silhouette
- run k-means for different values of k - calculate **WCSS**: within cluster sum of squares - plot WCSS for growing k - take the k where ‘the elbow bends’	- run k-means for different values of k - calculate the **average silhouette** - plot the measure for growing k - take the k at the peak

K-Means: Difficulties (1)

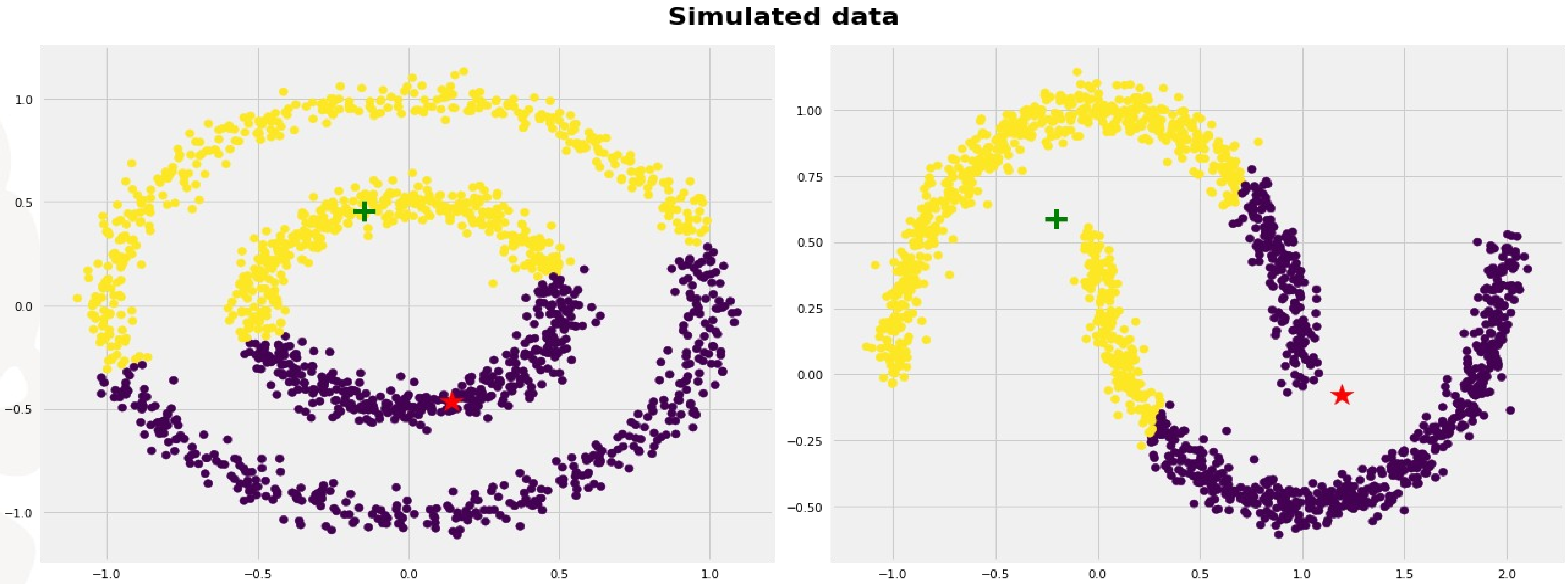
- How do we determine **k –** the number of clusters to split the data into?



https://towardsdatascience.com/clustering-metrics-better-than-the-elbow-method-6926e1f723a6

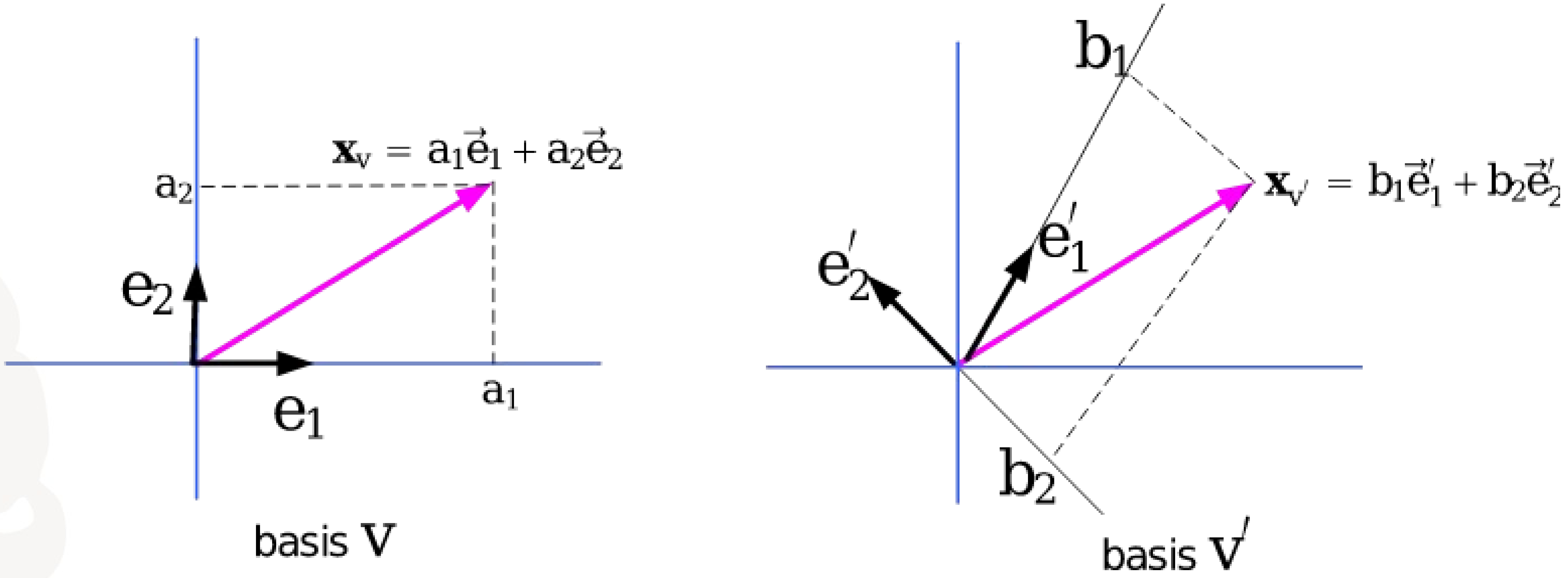
K-Means: Difficulties (2)

- What if the cluster are not of a spherical shape?



https://towardsdatascience.com/k-means-clustering-algorithm-applications-evaluation-methods-and-drawbacks-aa03e644b48a

PCA: Basis transformation



The same vector having different representation depending on basis used

https://www.12000.org/my_notes/similarity_transformation_and_SVD/index.htm

PCA: Eigenvalues & eigenvectors

eigenvalue

eigenvector

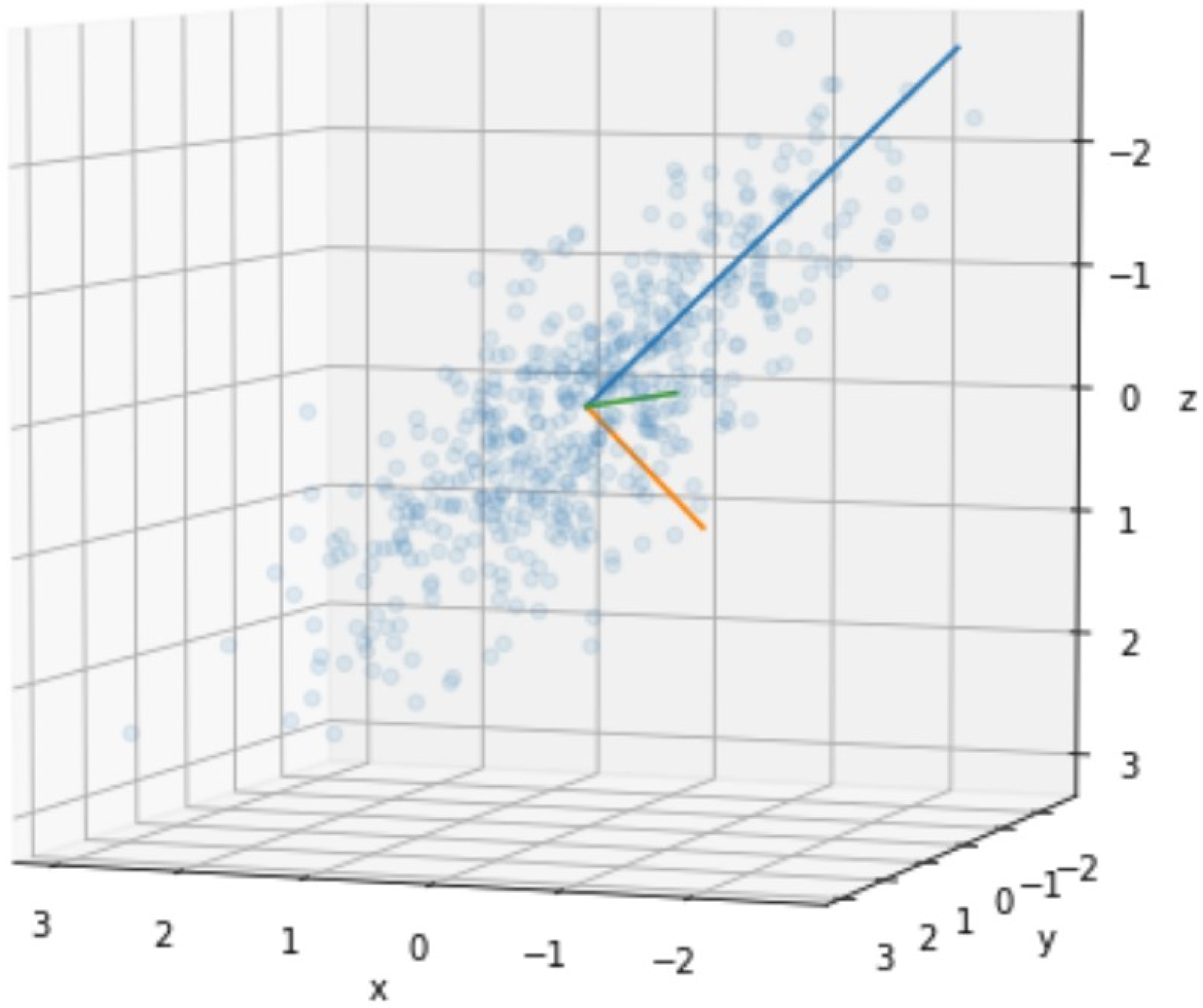
$$Av = \lambda v$$

Interpretation: the eigenvector v does not change (its direction) when multiplied by A, it is only scaled.

PCA – Principal Component Analysis

- Input: d-dimensional data

- Subtract the mean from your data
- Compute the covariance matrix for your zero-mean data
- Compute the eigenvalues and eigenvectors of the **covariance matrix**
- Sort the **eigenvectors** (=principal components) in descending order according to the eigenvalues
- Pick a subset of them and transform your data



http://www.iro.umontreal.ca/~pift6080/H09/documents/papers/pca_tutorial.pdf

Covariance matrix

- **Variance:**

$$var(X) = \frac{\sum_{i \in N} (x_i - \mu_x)^2}{N-1} = \frac{\sum_{i \in N} (x_i - \mu_x) * (x_i - \mu_x)}{N-1}$$

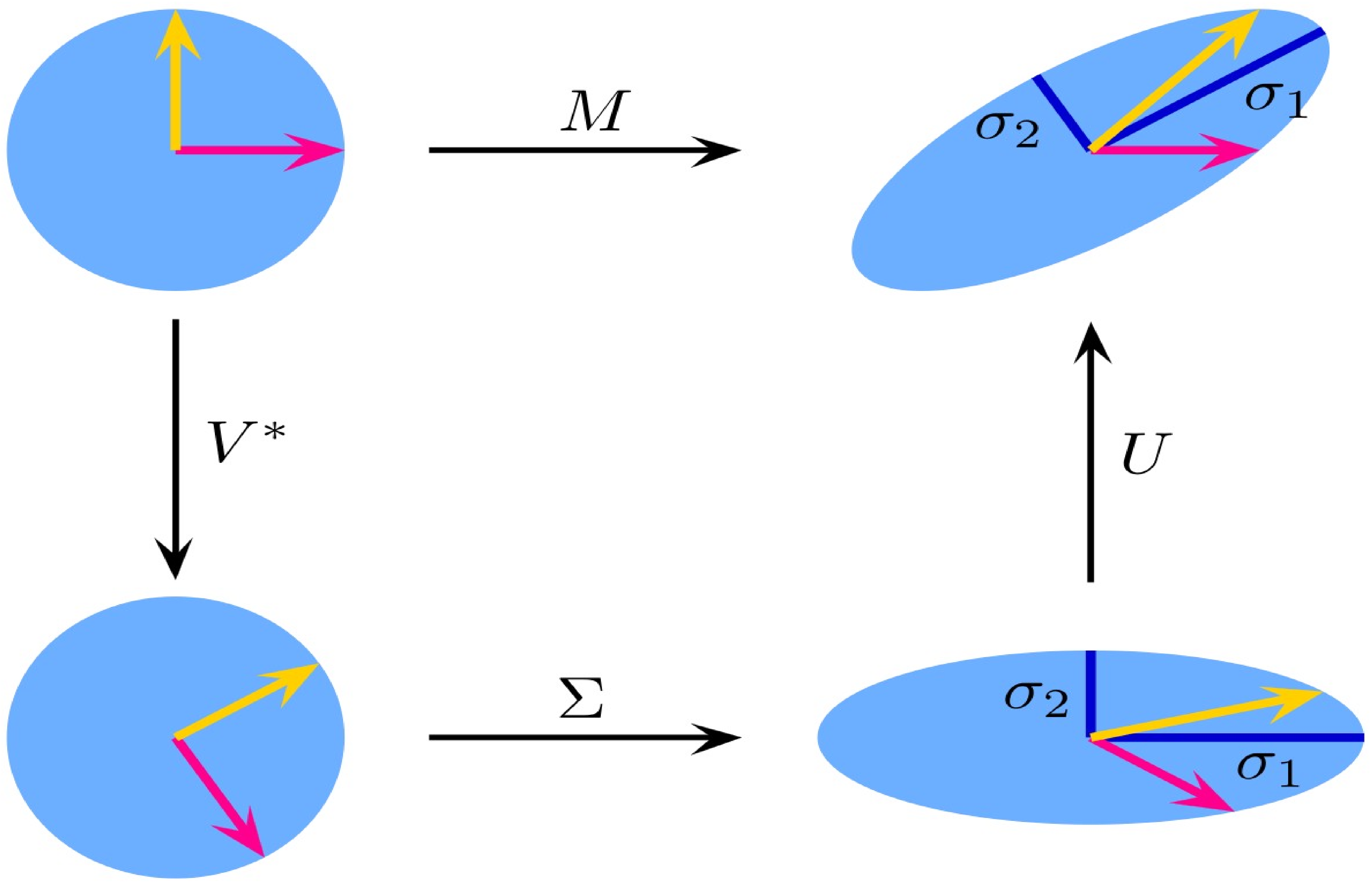
- **Covariance:**

$$covar(X, Y) = \frac{\sum_{i \in N} (x_i - \mu_x) * (y_i - \mu_y)}{N-1}$$

http://www.iro.umontreal.ca/~pift6080/H09/documents/papers/pca_tutorial.pdf

PCA: Compute the eigenvalues, -vectors

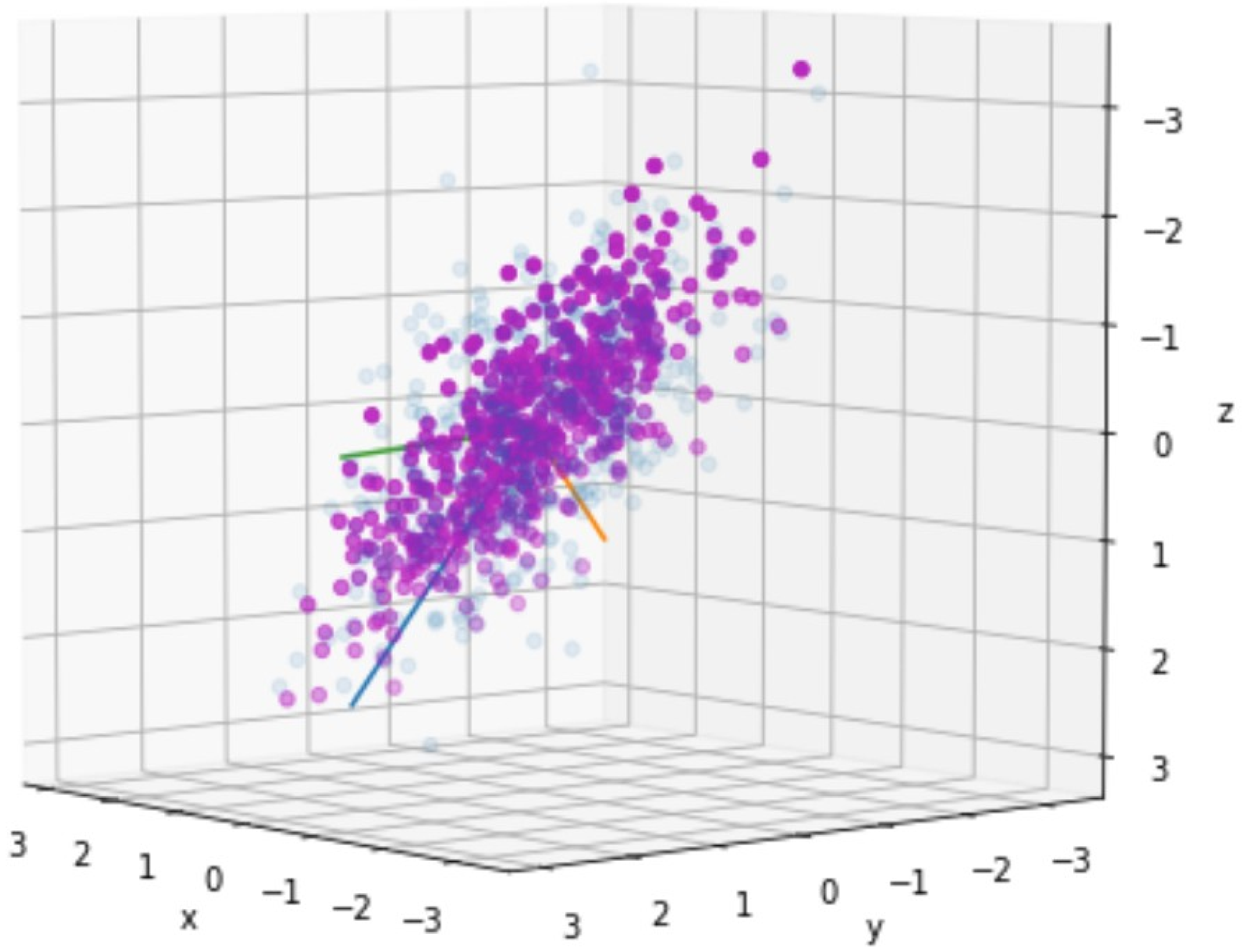
SVD (singular value decomposition) as a generalization of **eigendecomposition** (which is only applicable to square matrices)

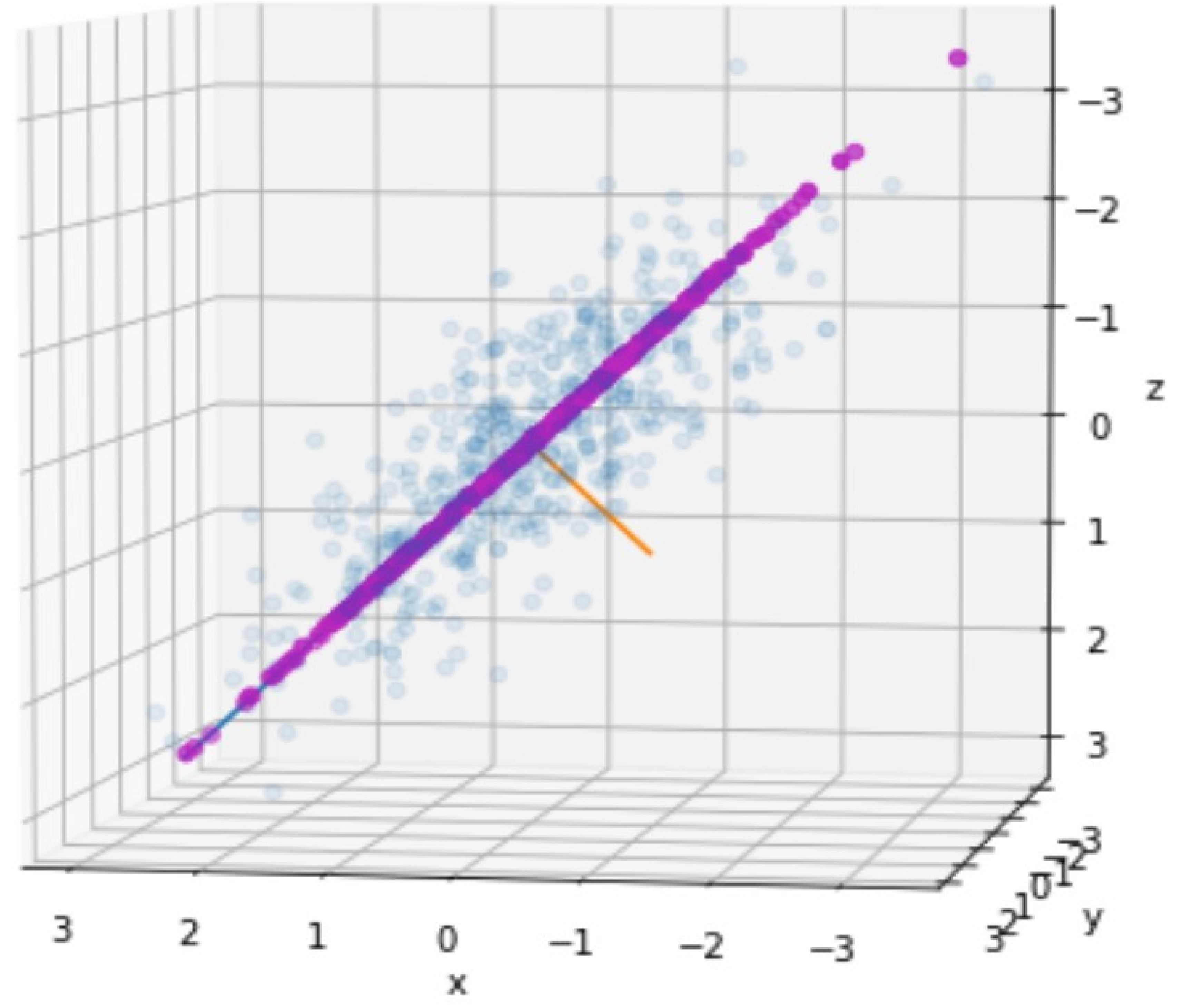


$$M = U \cdot \Sigma \cdot V^*$$

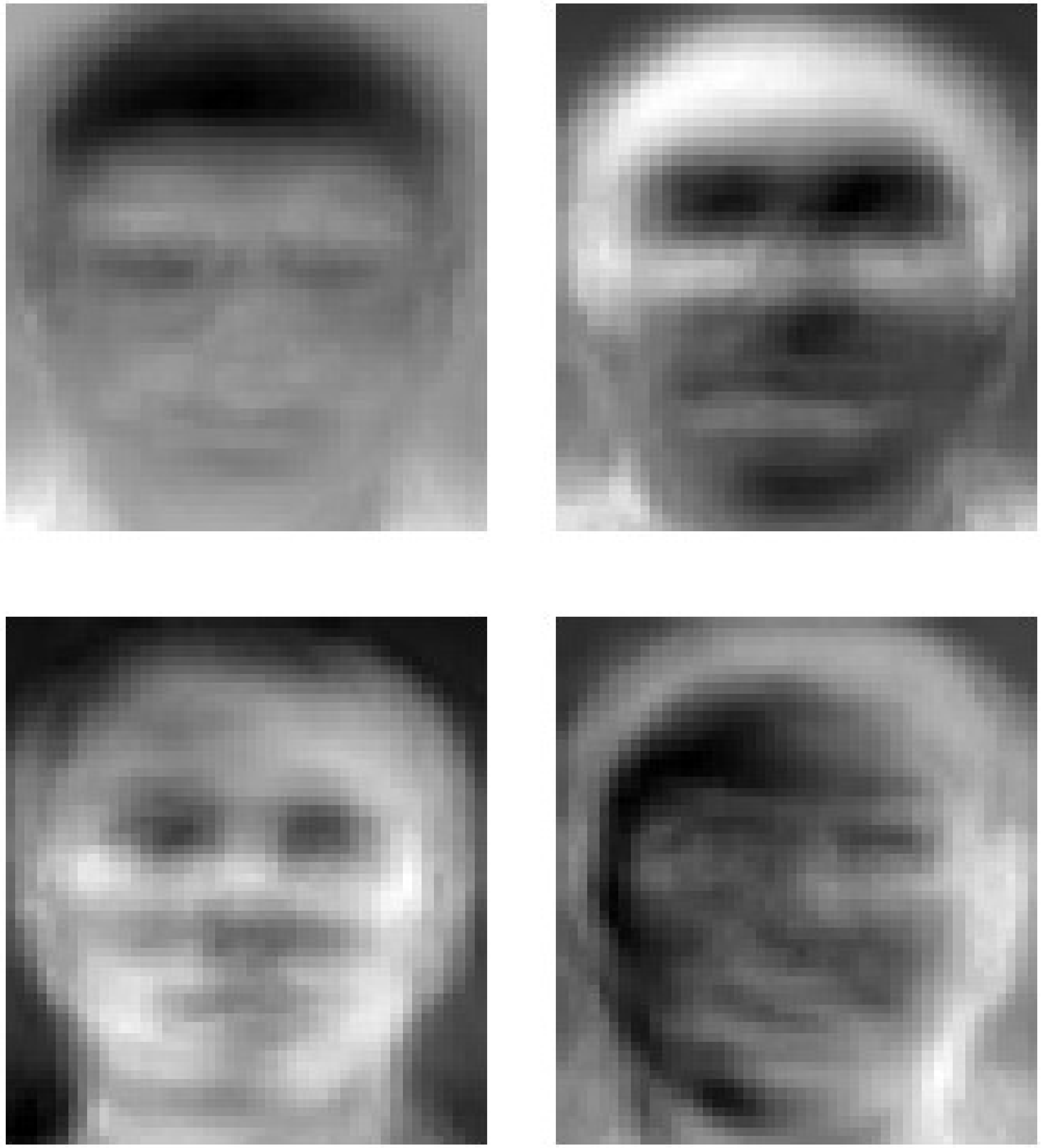
https://en.wikipedia.org/wiki/Singular_value_decomposition

Dimensionality reduction: 3D - 2D





Facial recognition: Eigenfaces



https://en.wikipedia.org/wiki/Eigenface

ICA – Independent component analysis

- separate a signal into **non-Gaussian** and **statistically independent** subcomponents

https://edtech.engineering.utoronto.ca/files/item-76-cocktail-party-effectpng

ICA – Independent component analysis

1. Center **x** by subtracting the mean
2. Whiten **x** (PCA or ZCA)

--------**fastICA**--

3. Choose a random initial value for the de-mixing matrix **W**
4. Calculate the **new value** for w
5. **Normalize** w
6. Check whether algorithm has converged and if it hasn’t, return to step 4

----

7. Take the **dot product of W and x** to get the independent source signals

https://towardsdatascience.com/independent-component-analysis-ica-in-python-a0ef0db0955e

ICA: Whitening (2)

- **Reminder**: covariance matrix (**C**) of a symmetric positive semi-definite matrix can be decomposed into a diagonal matrix **D** and a transformation matrix **E**

$$C = E * D * E^T$$

- The transformation matrix for **PCA-whitening** is $W_{PCA} = D^{1/2} * E^T$

- For **ZCA-whitening** is $W_{ZCA} = E * D^{1/2} * E^T$

https://stats.stackexchange.com/questions/117427/what-is-the-difference-between-zca-whitening-and-pca-whitening

ICA: fastICA (3-6)

- we search for such a matrix that the **independent sources** $s = Wx$

- take as **w** a vector that maximizes the non-Gaussianity of $w^T x$
- several measures of non-Gaussianity, for example **negentropy** using higher-order moments
- **decorrelate** outputs $w_1^T x, \ldots, w_n^T x$ so that no 2 vectors converge to the same maxima

- **convergence** when old and new w point into the same direction (their dot-product is 1)

A.Hyvärinen, E. Oja (2000). Independent component analysis: algorithms and applications. Neural Networks 13, 411-430

https://itb.biologie.hu-berlin.de/~kempter/Teaching/2006_SS/hyvarinen_oja_2000.pdf

ICA application: Image superposition

https://upload.wikimedia.org/wikipedia/commons/7/74/Major_Mitchell%27s_Cockatoo_1_-_Mt_Grenfell.jpg

https://upload.wikimedia.org/wikipedia/commons/7/73/Lion_waiting_in_Namibia.jpg

Difference between PCA and ICA

PCA:

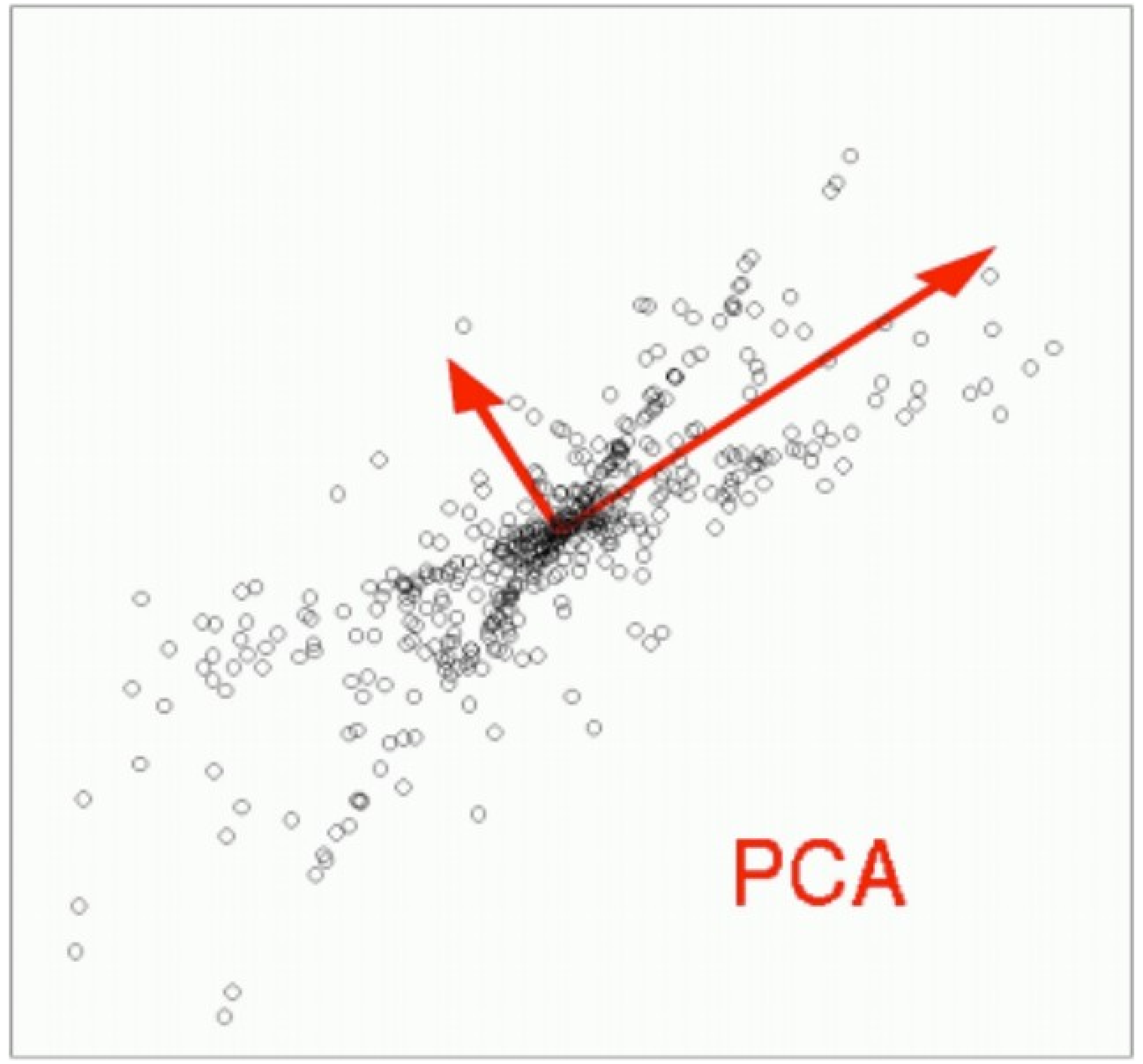
- Removes correlation
- Some components (eigenvalues) are **more important** than others
- (Eigen)vectors are **orthogonal**

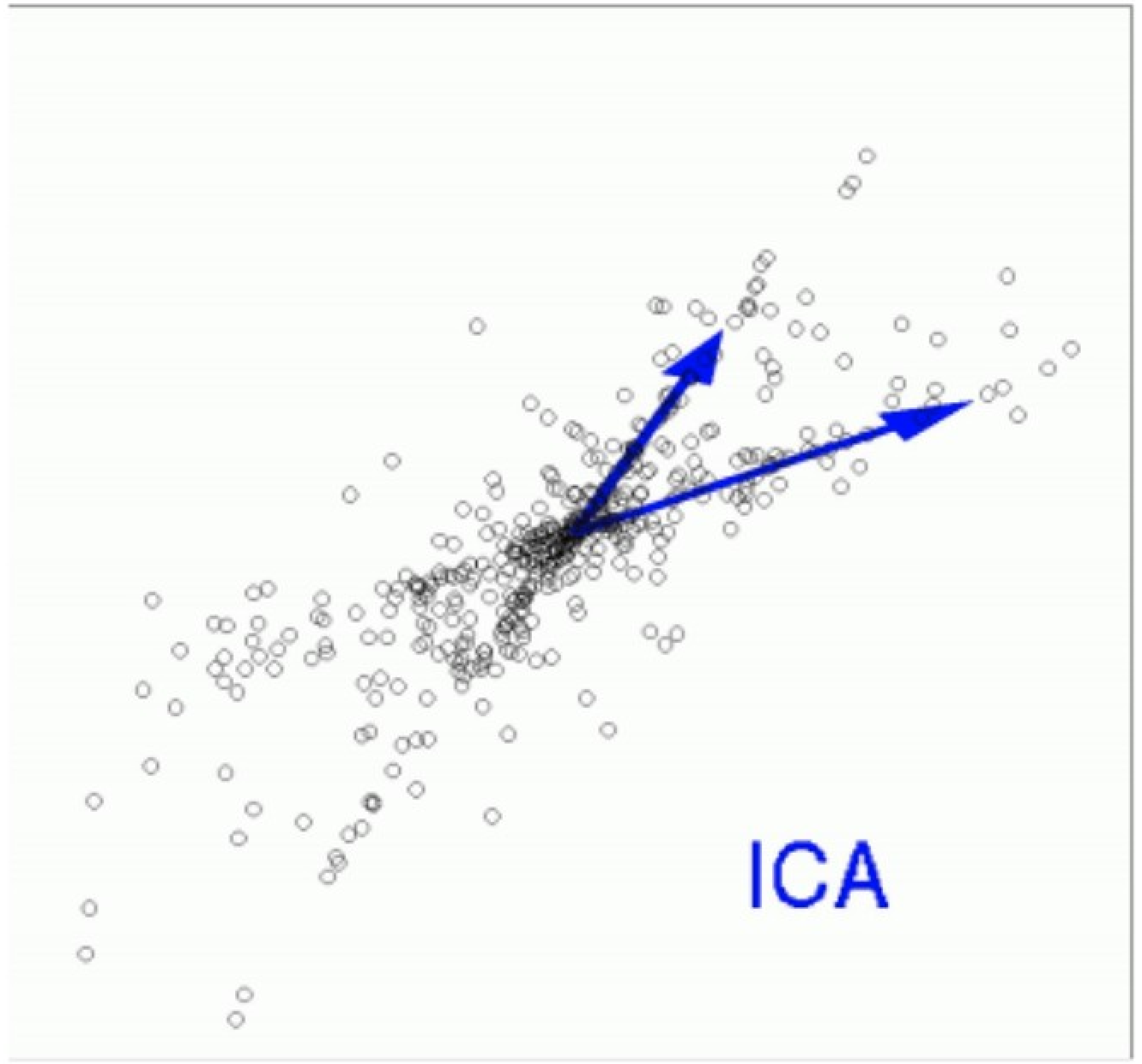
ICA:

- Removes correlation **and** higher-order dependence
- All components are **equally important**
- ICA vectors are **not orthogonal**

http://compneurosci.com/wiki/images/4/42/Intro_to_PCA_and_ICA.pdf

Difference between PCA and ICA





http://compneurosci.com/wiki/images/4/42/Intro_to_PCA_and_ICA.pdf

Thank you for your attention!