LECTURER: TAI LE QUY

MACHINE LEARNING

UNSUPERVISED LEARNING AND FEATURE ENGINEERING

MACHINE LEARNING—UNSUPERVISED LEARNING AND FEATURE ENGINEERING
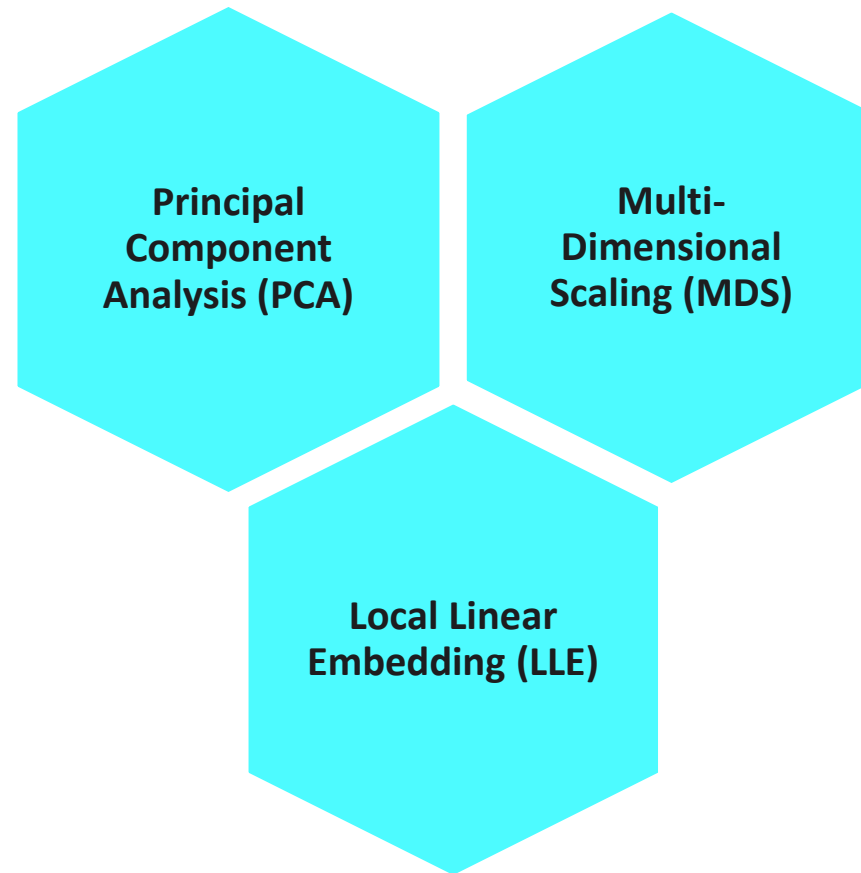**TOPIC OUTLINE**

DIMENSIONALITY REDUCTION

— Explain **how dimensionality reduction** approaches **work**.

— **Apply** dimensionality reduction approaches.

— **Choose an appropriate** dimensionality reduction **approach** with respect to the challenges and constraints of the dataset.
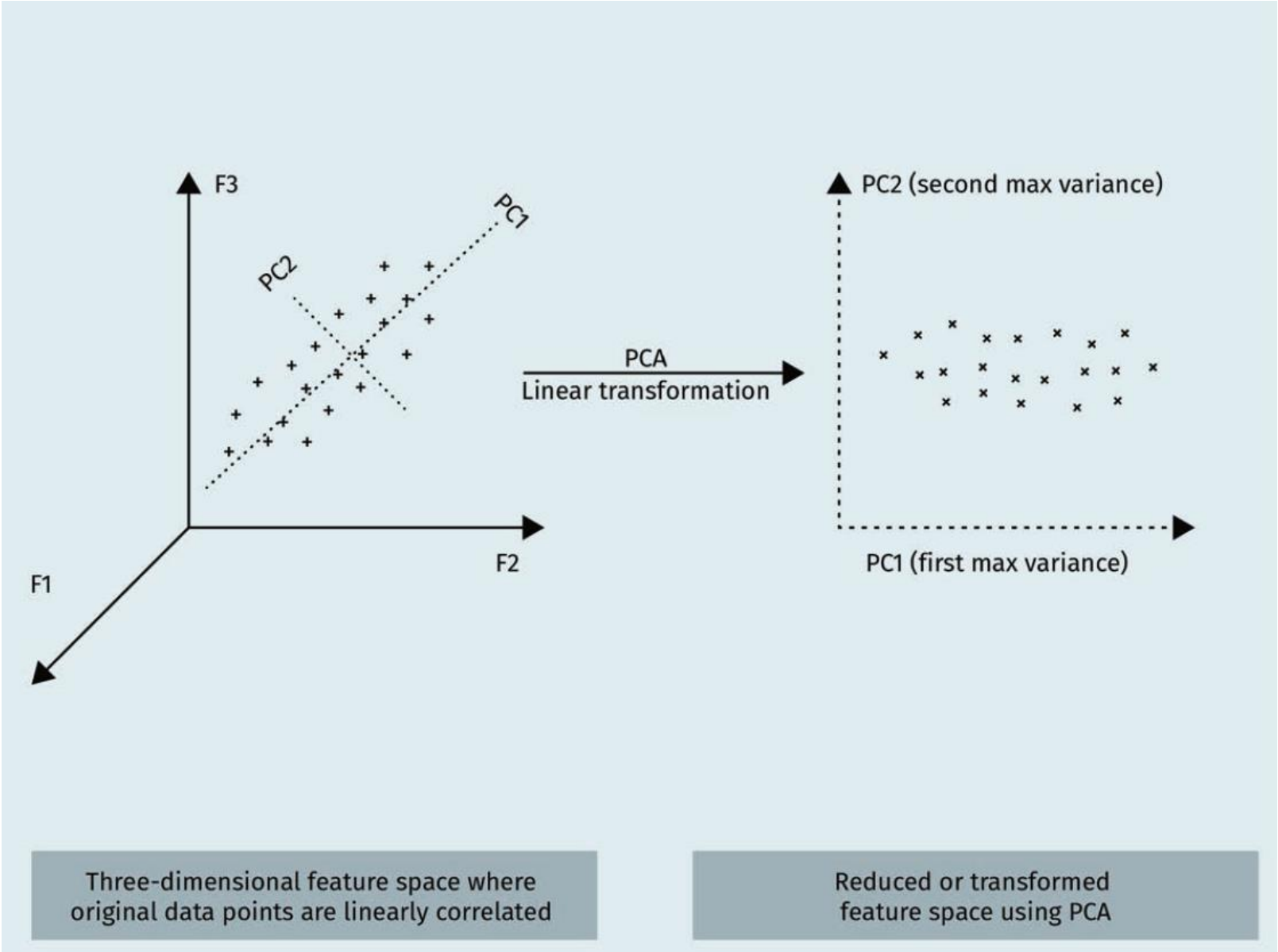
1. Name one **reason** why **dimensionality reduction** is conducted **before clustering** algorithms are applied.

2. Explain what the **curse of dimensionality** is.

3. Explain how dimensionality reduction can be **used during** the **reporting** phase of a project.

UNIT CONTENT


Image 1: Unit content - Dimensionality reduction

Principal Component Analysis (PCA)

Multi-Dimensional Scaling (MDS)

Local Linear Embedding (LLE)

PRINCIPAL COMPONENT ANALYSIS (PCA)

Three-dimensional feature space where original data points are linearly correlated

Reduced or transformed feature space using PCA

- The data has some amount of variance/information. We would like to choose a direction u so that if we were to approximate the data as lying in the direction/subspace corresponding to u, as much as possible of this variance is still retained.

Initial data   Direction 1   Direction 2
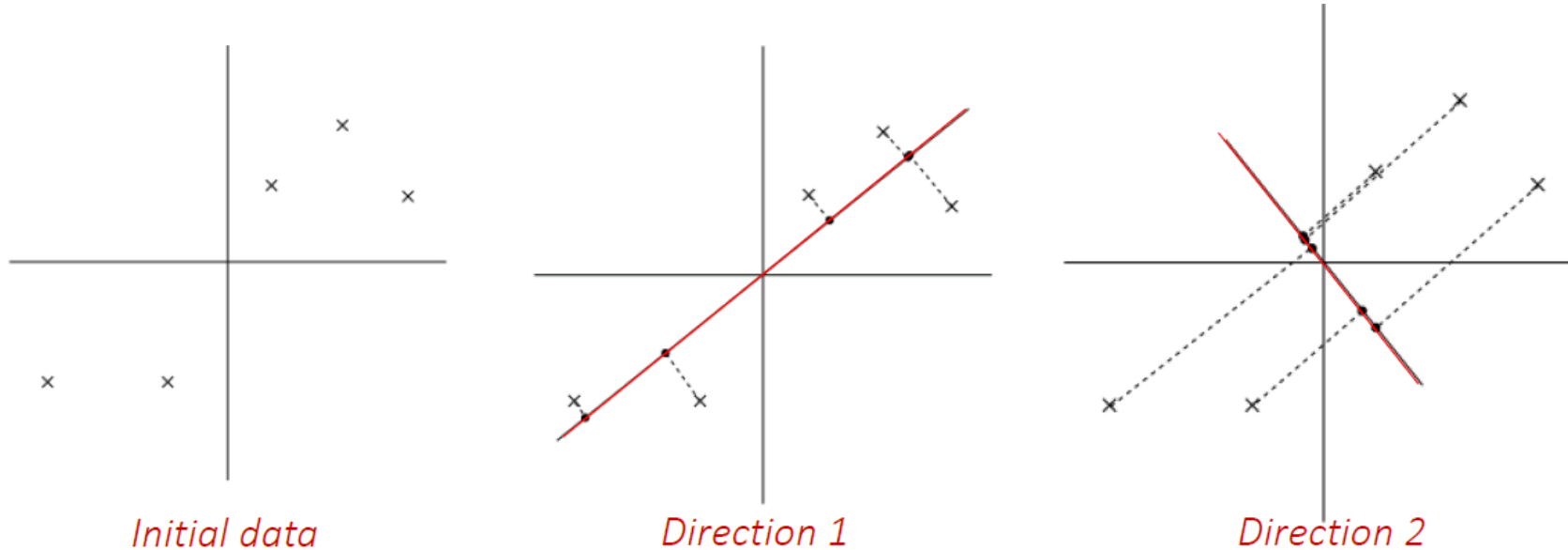
Idea: Choose the direction that maximizes the variance of the projected data

- General form: DV = D'
  - **D_{nxd} is the original dataset**
  - **V_{dxk} is a linear transformation**
  - **D'_{nxk} is the re-representation of the dataset**
  - **d: original dimensionality**
  - **k: reduced dimensionality**

- Geometrically, *V* is a *rotation* and a *stretch* which transforms *D* into *D'*
  - The eigenvectors are the rotations to the new axes
  - The eigenvalues are the amount of stretching that needs to be done
  - The eigenvectors (principal components) determine the directions of the new feature space
  - The eigenvalues explain the variance of the data along the new feature axes.

- PCA computes the **eigenvalues** and **eigenvectors** of covariance matrix

Source of the image: Müller-Kett, 2021.

— Describes the **variance** of all features (in the diagonal) and feature pairwise correlations/**covariances**

$$D = \begin{pmatrix} v_1 \\ \vdots \\ v_n \end{pmatrix} = \begin{pmatrix} v_{1,1} & \cdots & v_{1,d} \\ \vdots & \ddots & \vdots \\ v_{n,1} & \cdots & v_{n,d} \end{pmatrix} \qquad \Sigma_D = \begin{pmatrix} VAR(X_1) & \cdots & COV(X_1,X_d) \\ \vdots & \ddots & \vdots \\ COV(X_d,X_1) & \cdots & VAR(X_d) \end{pmatrix}$$

- For *d*-dimensional data, *dxd* covariance matrix
- Symmetric matrix as *COV(X,Y)=COV(Y,X)*

— Variance is a measure of the spread of the data along a dimension X
It measures how far the values of X are spread out from the average X value (μ)

$$VAR(X) = \frac{1}{n} \sum_{i=1}^{n} (x_i - \mu)^2$$

ID	Height
1	100
2	100
3	100
4	100
5	100

Variance =0

ID	Height
1	100
2	100
3	105
4	100
5	100

Small variance (4)

ID	Height
1	100
2	100
3	100
4	200
5	100

Large variance (1600)

– Covariance provides a measure of the strength of the correlation between two variables X,Y

$$COV(X,Y) = \frac{1}{n}\sum_{i=1}^{n}(x_i - \mu_x)(y_i - \mu_y)$$

COVARIANCE

Large Negative Covariance   Near Zero Covariance   Large Positive Covariance
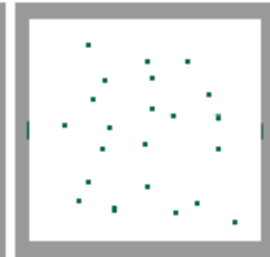
– μ_x, μ_y: the means of X, Y

– What the covariance values mean

- – Zero value: the variables are uncorrelated.
- – Positive values: both dimensions move together (increase or decrease)
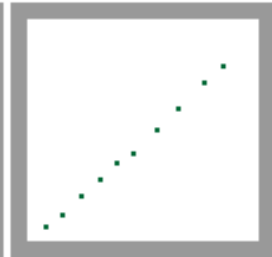- – Negative values: while one dimension increases the other decreases

– Let Σ be square dxd matrix

– A non-zero vector v is called an *eigenvector* of Σ if and only if there exists a scalar (i.e., a single number) λ such that:

$$\Sigma v = \lambda v$$

– If such a number λ exists it is called and eigenvalue of Σ

– The vector v is call the eigenvector associated with λ

– How to find the eigenvalues/eigenvectors of Σ ?

  – By solving $\Sigma v = \lambda v$, which can be rewritten as $\Sigma - \lambda I)v = 0$ (I is the identity matrix, 0 is a vector of all zero.

  – From linear algebra, in order for $(\Sigma - \lambda I)v = 0$ to hold, the determinant should be zero, i.e., $\det(\Sigma - \lambda I) = 0$

  – Solving $\det(\Sigma - \lambda I) = 0$ we get the eigenvalues λ

  – For each eigenvalue λ we can find the eigenvector by solving: $\Sigma v = \lambda v$

$$I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

$$|A| = \begin{vmatrix} a & b \\ c & d \end{vmatrix} = ad - bc.$$

PRINCIPAL COMPONENT ANALYSIS (PCA)

1. **Standardize** the data.

2. Compute the **variance-covariance matrix.**

3. Compute **eigenvalues** and **eigenvectors** (matrix eigen decomposition).

4. **Select** the top k **Principle Components.**

5. Construct the **projection matrix/loading scores.**

**6.** **Project** the data and **visualize** it in the PC feature space.

Image 2: PCA

PRINCIPAL COMPONENT ANALYSIS (PCA)

Implementing PCA

Dataset X → Standardize X → Xstd → Compute variance-covariance matrix → Σ → Compute eigenvalues and eigenvectors → λ, V → Select top k principal component → PCs → Construct the projection matrix → PR → Project and visualize in PCs feature space → $Y = Xstd \cdot PR$
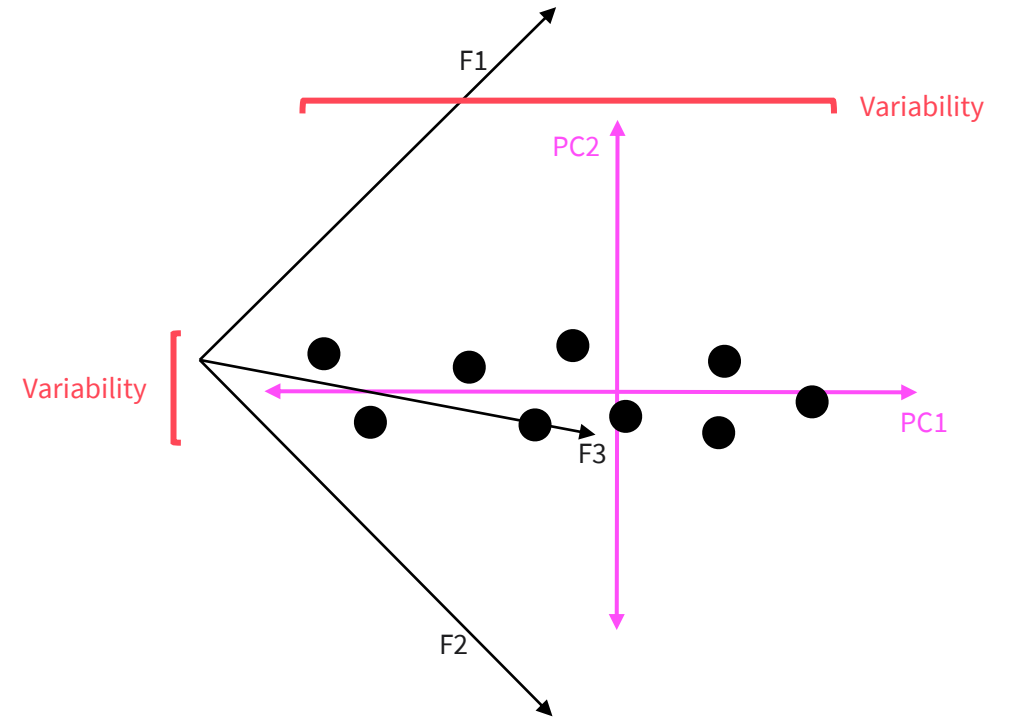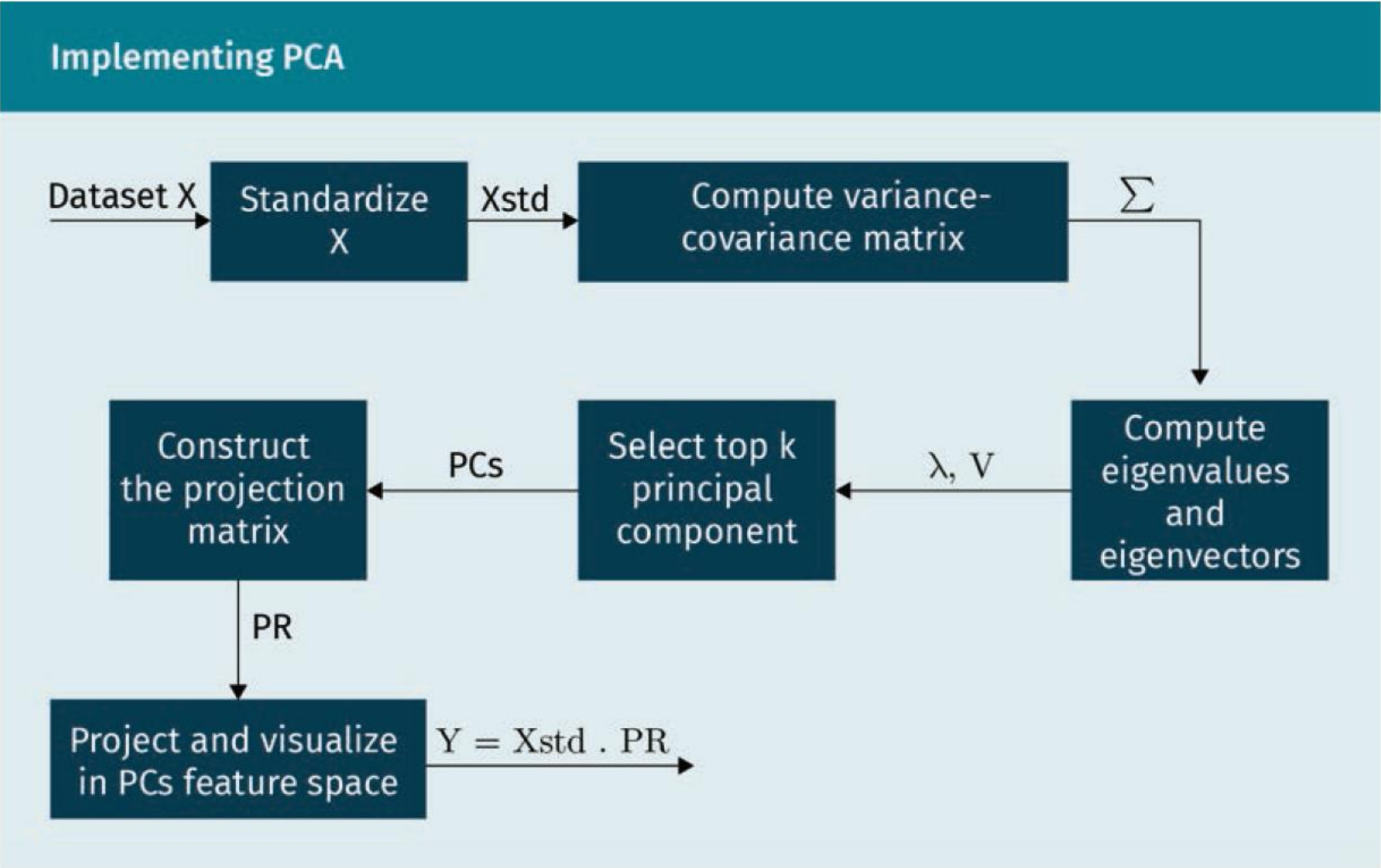
Source of the image: Müller-Kett, 2021.

PRINCIPAL COMPONENT ANALYSIS (PCA): EXAMPLE OF TRANSFORMATION
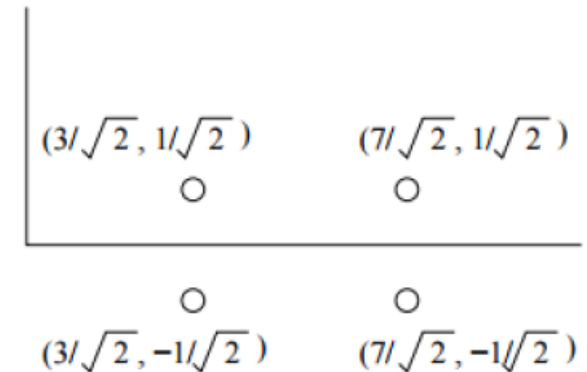
■ Original

(3,4)

(4,3)

(1,2)

(2,1)
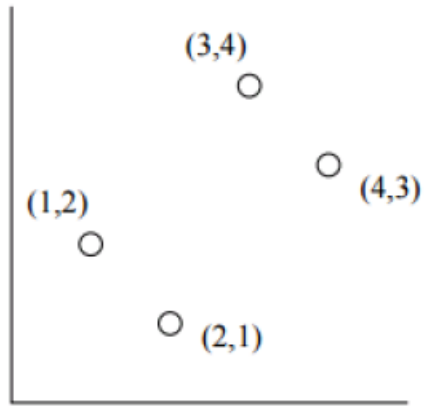
Eigenvectors

$$\begin{bmatrix} 1/\sqrt{2} \\ 1/\sqrt{2} \end{bmatrix} \qquad \begin{bmatrix} -1/\sqrt{2} \\ 1/\sqrt{2} \end{bmatrix}$$

In the rotated coordinate system

$(3/\sqrt{2}, 1/\sqrt{2})$ $(7/\sqrt{2}, 1/\sqrt{2})$

$(3/\sqrt{2}, -1/\sqrt{2})$ $(7/\sqrt{2}, -1/\sqrt{2})$

■ Transformed data

$$\begin{bmatrix} 1 & 2 \\ 2 & 1 \\ 3 & 4 \\ 4 & 3 \end{bmatrix} \begin{bmatrix} 1/\sqrt{2} & -1/\sqrt{2} \\ 1/\sqrt{2} & 1/\sqrt{2} \end{bmatrix} = \begin{bmatrix} 3/\sqrt{2} & 1/\sqrt{2} \\ 3/\sqrt{2} & -1/\sqrt{2} \\ 7/\sqrt{2} & 1/\sqrt{2} \\ 7/\sqrt{2} & -1/\sqrt{2} \end{bmatrix}$$

PERCENTAGE OF VARIANCE EXPLAINED BY PCA

– Let k be the number of top eigenvalues out of d (d is the number of dimensions)

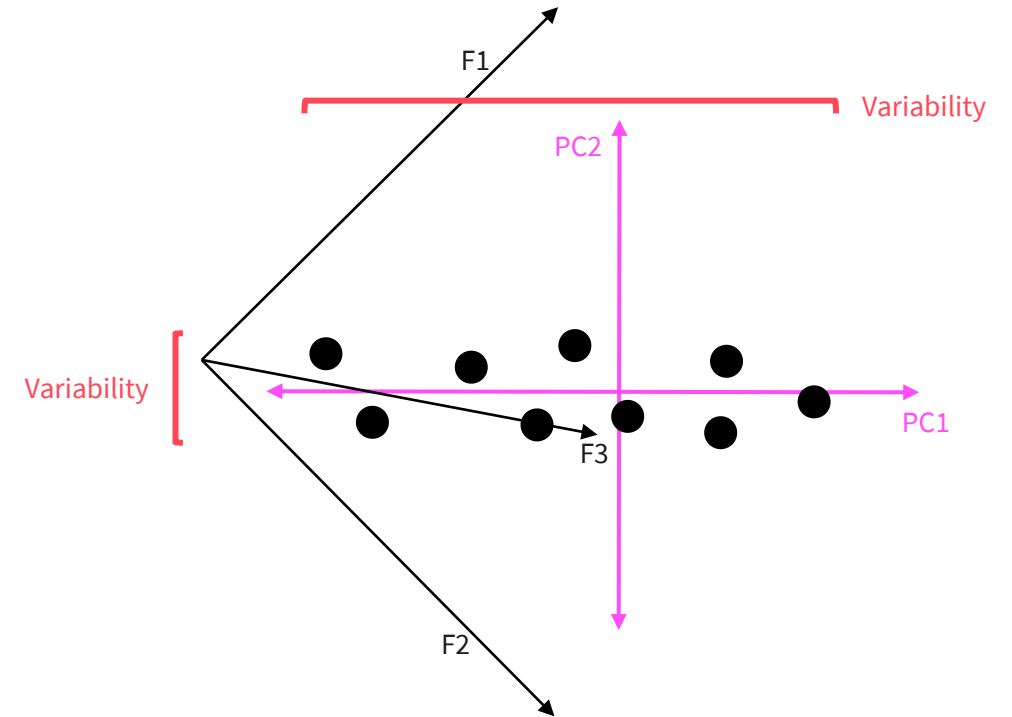– The percentage of variance in the dataset explained by the k selected eigenvalues $\lambda_1, \ldots, \lambda_k$ is:

$$\frac{\sum_{i=1}^{k} \lambda_i}{\sum_{i=1}^{d} \lambda_i}$$

– Similarly, you can find the variance explained by each principal component

– Rule of thumb: keep enough to explain **85%** of the variance

Drawbacks

Image 3: PCA

— PCA performs **poorly on non-linearly correlated** data

— **Thresholds** and **PC selection** is highly **use-case specific** and **subjective**

— **Only** for **numerical** features

— **Bad interpretability**

PRINCIPAL COMPONENT ANALYSIS (PCA)

Nonlinearly Correlated Data Points and Their Projection into the Transformed Feature Space Using PCA
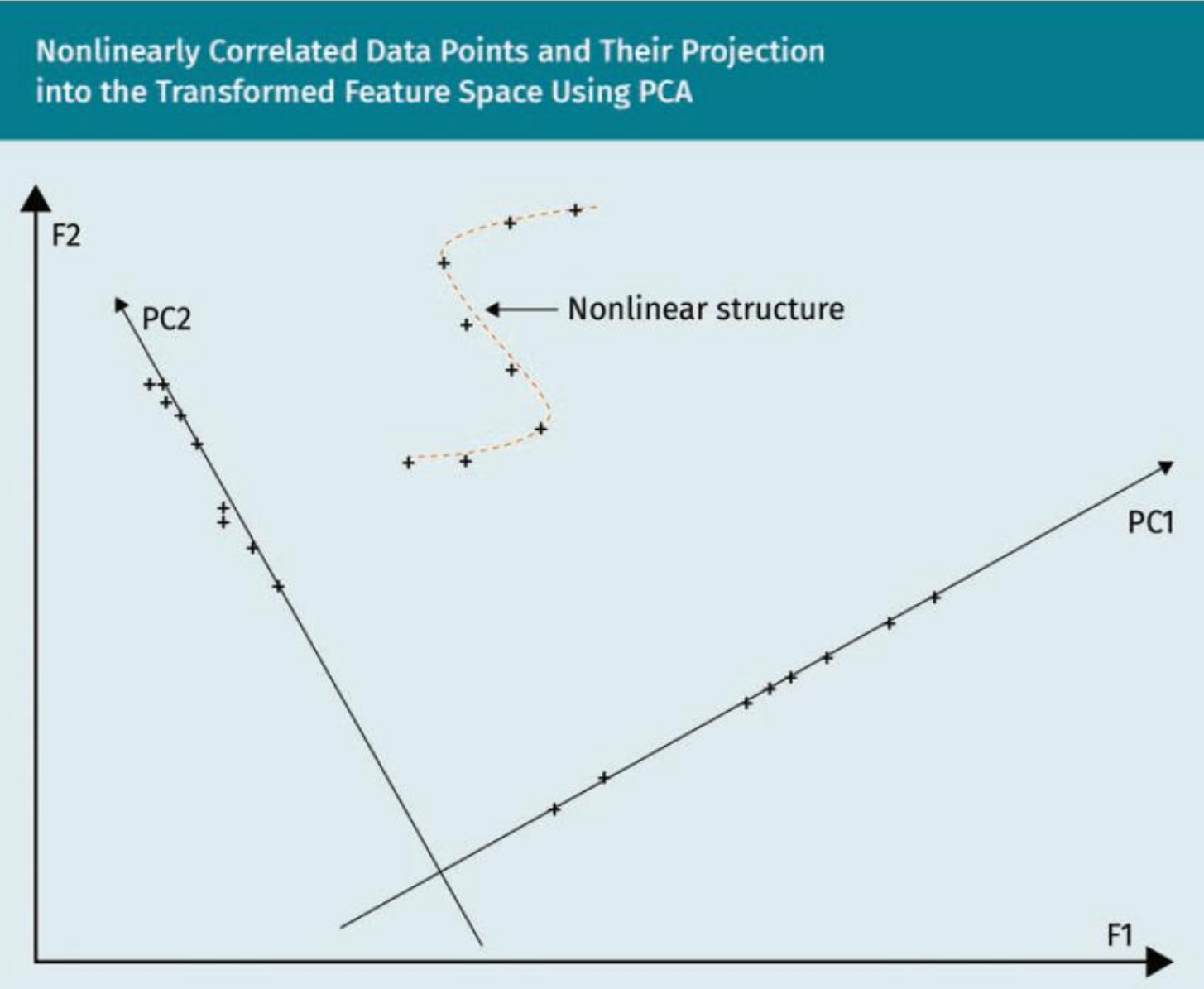
MULTI-DIMENSIONAL SCALING (MDS)

— Similar to PCA

— Instead of maximizing the variance, MDS **preserves pairwise similarities**…

— …by **minimizing the difference** between **distances in the original and reduced feature spaces.**

— Based on a **dissimilarity matrix**

  — **Metric MDS** for numeric data

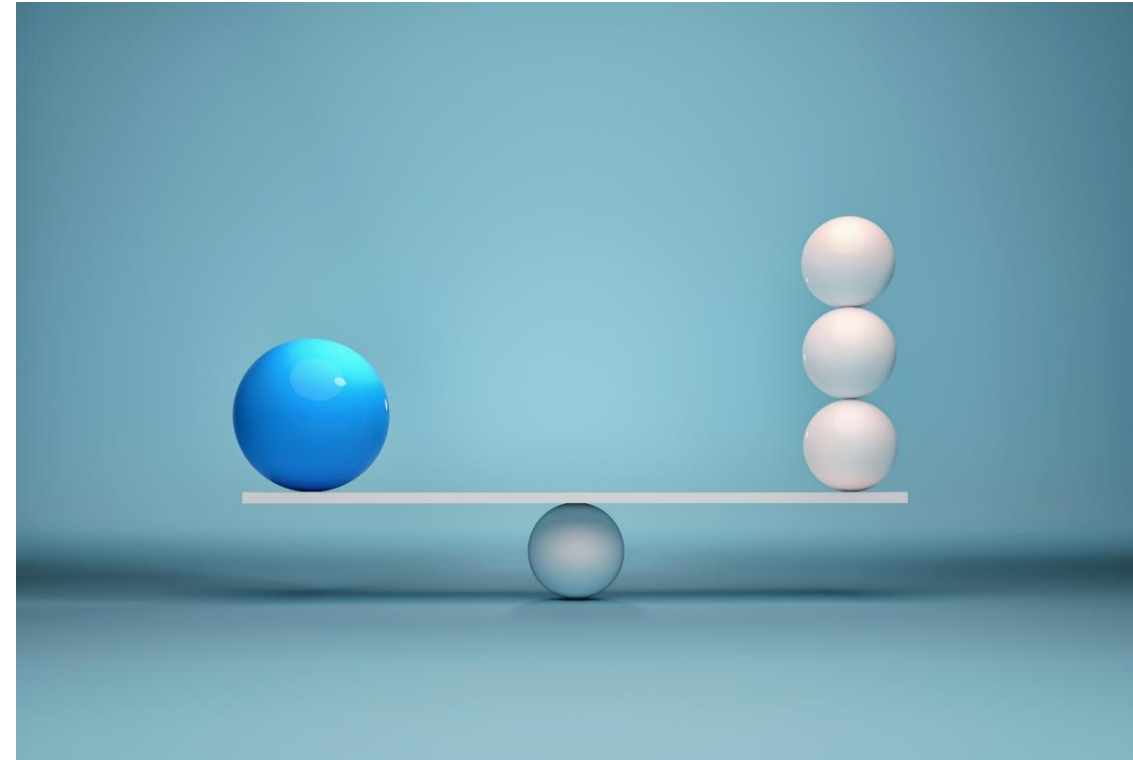 — **Non-Metric MDS** for categorical data

Advantage over PCA

— Can be used for **non-linear correlations.**

Disadvantage

— **Computationally expensive** compared to PCA.

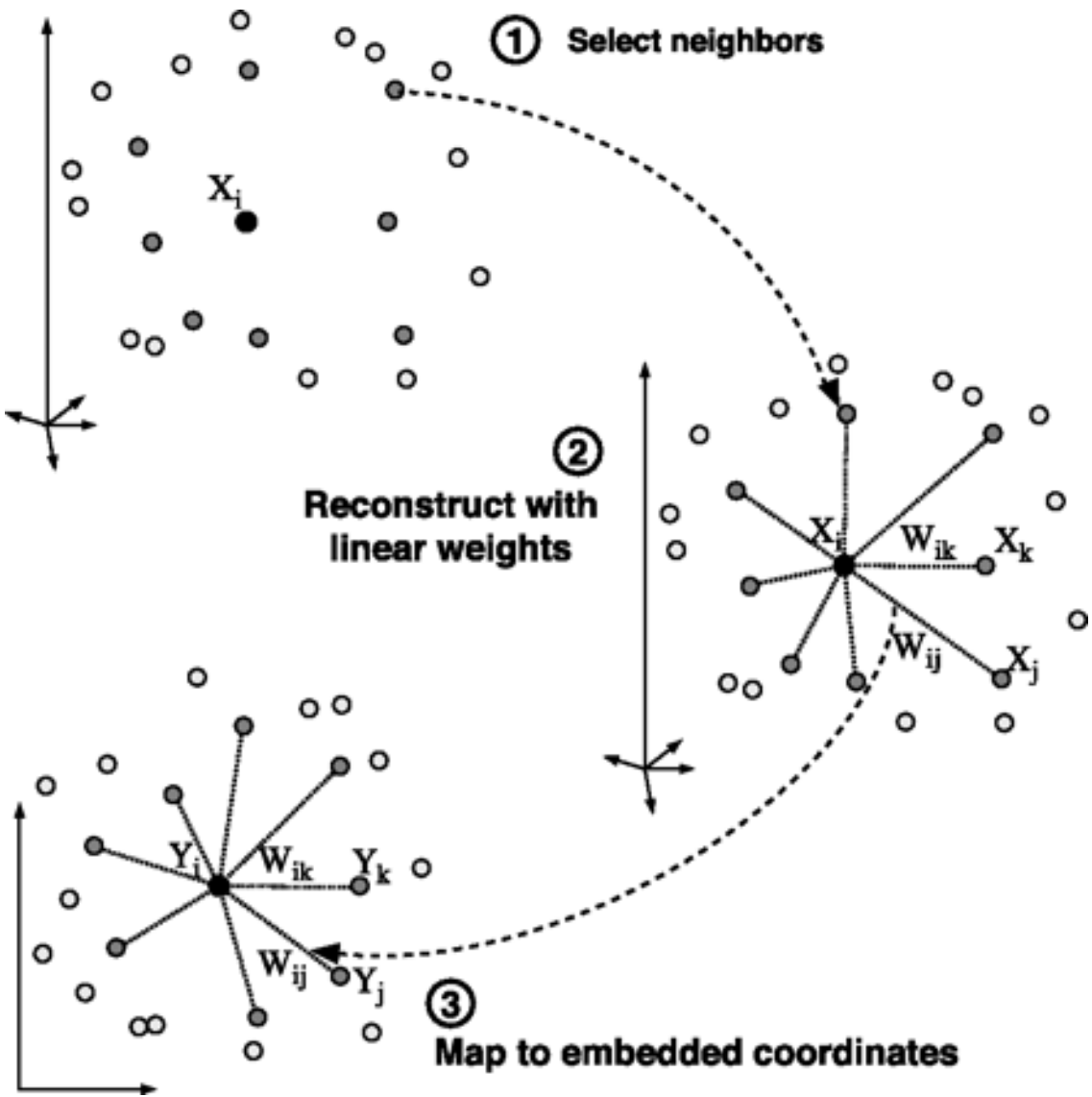Image 4: Pros and cons

— Preserves **local geometries** of the "manifold".

1. For each sample, choose **k nearest neighbors.**

2. Find **weights** that, when being multiplied with the neighbors, **reconstruct the sample** in question.

3. Use the optimized weights to find an **embedding vector** containing the coordinates in the reduced feature space.

Source of the text: Saul & Roweis, 2000.

LOCAL LINEAR EMBEDDING (LLE)

① Select neighbors

② Reconstruct with linear weights
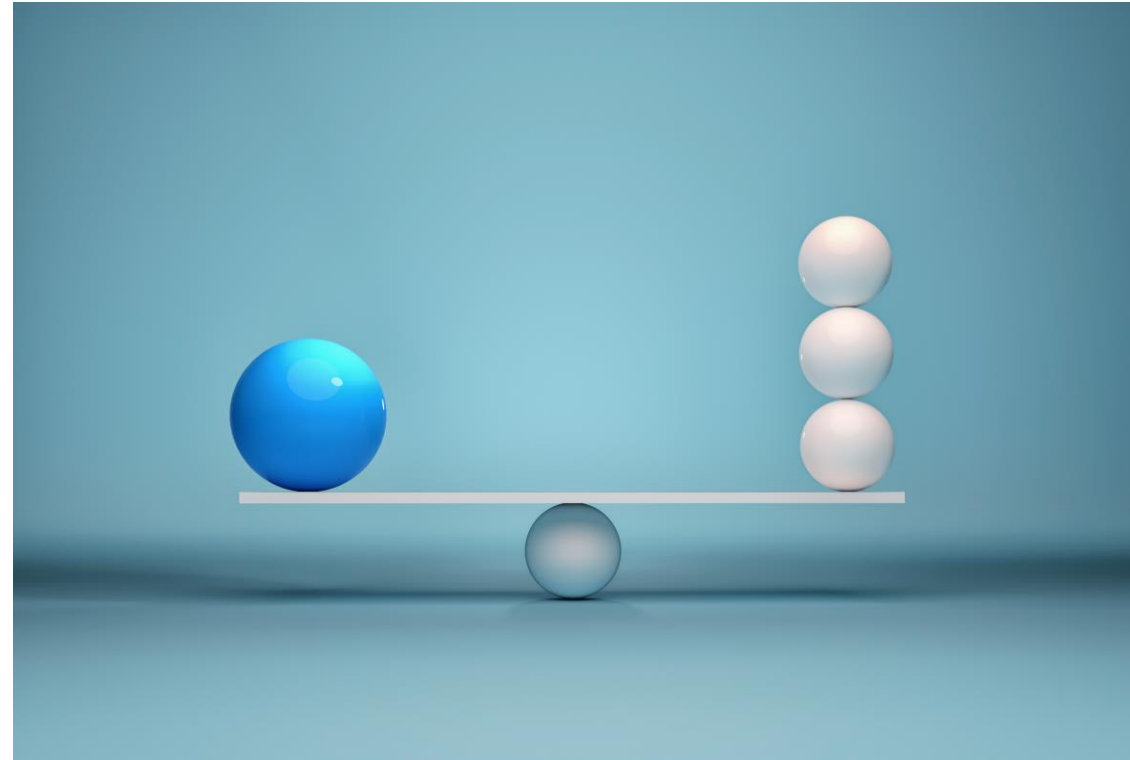
③ Map to embedded coordinates

Advantage

— **Computationally cheaper** than MDS

Disadvantages

— **Susceptible to outliers**

Image 4: Pros and cons

— Explain **how dimensionality reduction** approaches **work**.

— **Apply** dimensionality reduction approaches.

— **Choose an appropriate** dimensionality reduction **approach** with respect to the challenges and constraints of the dataset.

TRANSFER TASK

A start-up that sells **sustainable products in smaller stores** has been very successful in recent years. As a result, more stores are to be opened worldwide.

To keep an **overview of the offered products**, you and your team of Data Scientists are tasked to **define homogeneous groups of products** to facilitate ordering, marketing, and distribution. For this purpose, you create **data visualizations** that show product groups as data points close to each other.

After some exploratory trials, you realize that the products seem to be **grouped in irregular shapes** in the original feature space. Also, your team is relatively new and still waiting for big PCs, so you still **work on your laptop**.

Discuss which **method** is **appropriate** for this use case. In your reasoning, also explain why other methods are not suitable in this context.

Please present your results.

The results will be discussed in plenary.

1. Which of the following propositions is correct regarding principal component analysis (PCA)?

   a) Eigenvectors form the axes in the reduced feature space.

   b) Eigenvalues represent the overall variance in the entire dataset.

   c) Eigenvalues form the axes in the transformed feature space.

   d) Eigenvectors represent the proportion of explained variances according to each Eigenvalue.

2. Which of the following does MDS seek to preserve during dimensionality reduction?
   a) the variance and distance between data samples
   b) the variance between data samples
   c) the variance-covariance between data samples
   d) the distance between data samples

3. In which of the following is it recommended to use LLE?

    a) in datasets with noise and outliers

    b) in small datasets

    c) in large datasets

    d) in data samples with linear correlations

LIST OF SOURCES

Text

Buja, A., Swayne, D. F., Littman, M. L., Dean, N., Hofmann, H., & Chen, L. (2008). Data visualization with multidimensional scaling. Journal of Computational and Graphical Statistics, 17(2), 444—472. https://doi.org/10.1198/106186008X318440

Saul, L. K., & Roweis, S. T. (2000). An introduction to locally linear embedding. [Unpublished article]. https://cs.nyu.edu/~roweis/lle/papers/lleintro.pdf

Images

Müller-Kett, 2021.

Microsoft Archive.