

Dimensionality Reduction

Principal Component Analysis

Feature Extraction

- Aims to reduce the number of features in a dataset by creating new features from the existing ones (discarding the original ones)
- **Feature Extraction Techniques:**
 - **Principal Component Analysis:** Linear transformation techniques by finding orthogonal axes that capture the most variance

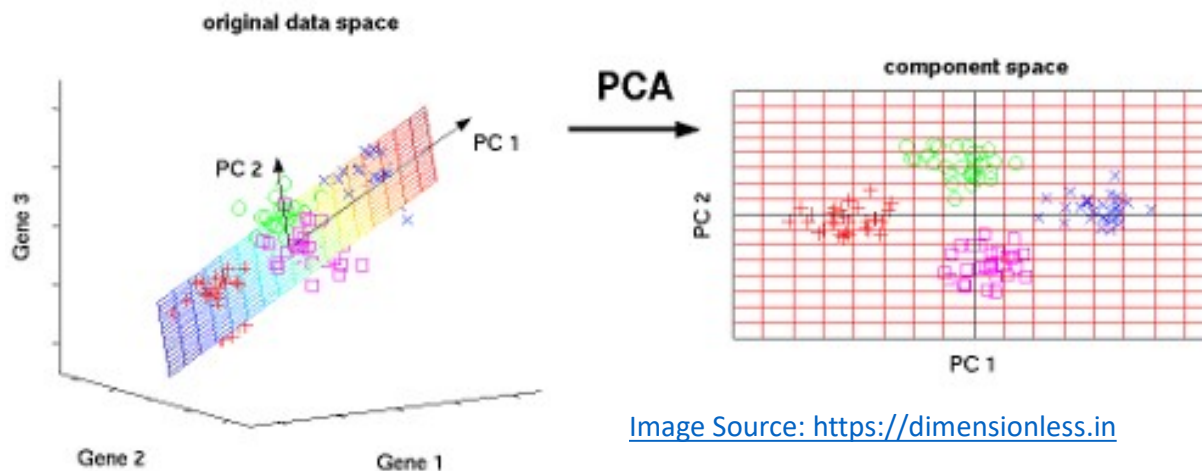
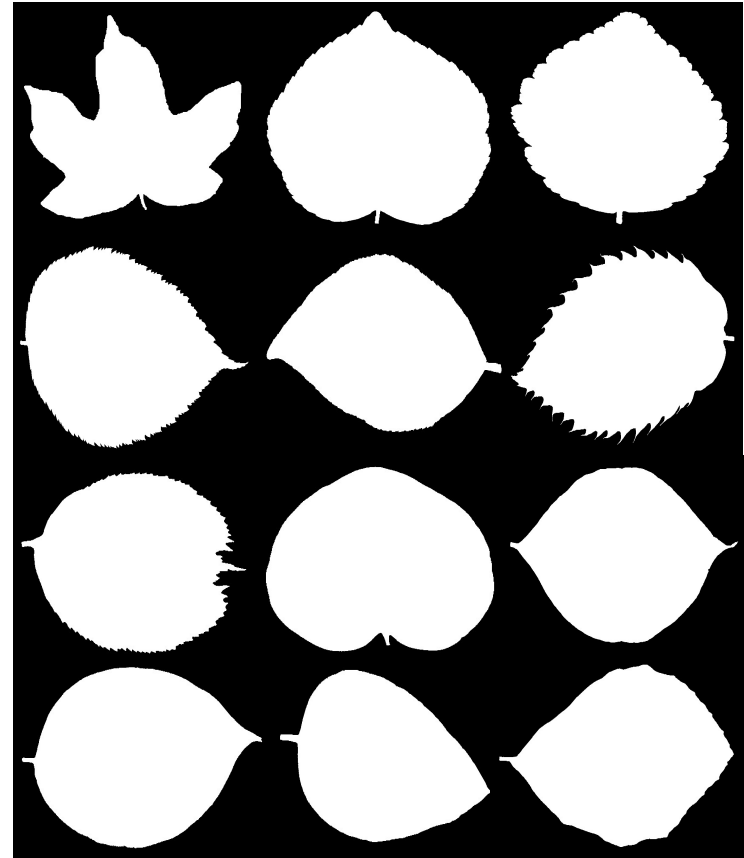


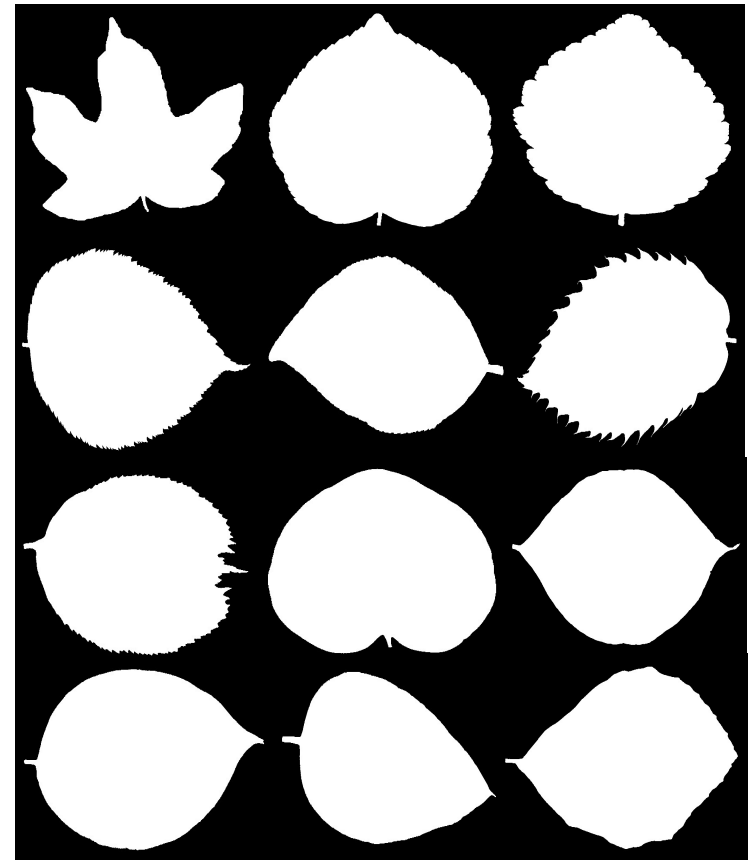
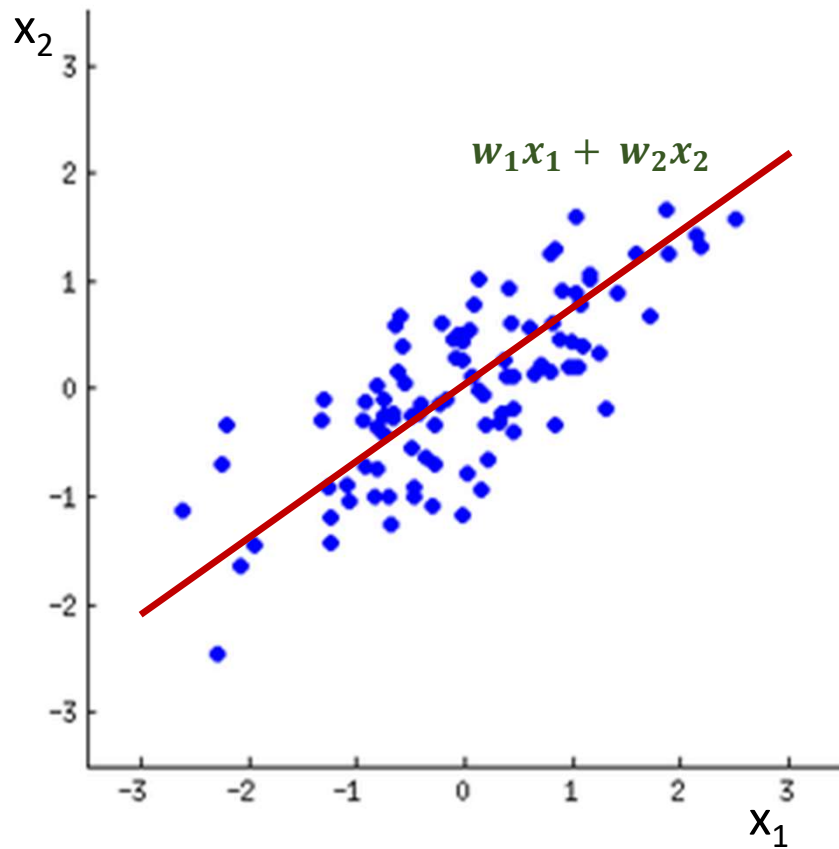
Image Source: <https://dimensionless.in>

Principal Component Analysis (PCA)

- We can compose a whole list of different characteristics of different plant-species leaves in the list
- Some of the characteristics will be redundant, while others will strongly differ among species
- PCA is about **constructing some new characteristics** (linear combinations of some old characteristics) that turn out to **summarize our different plant species well**

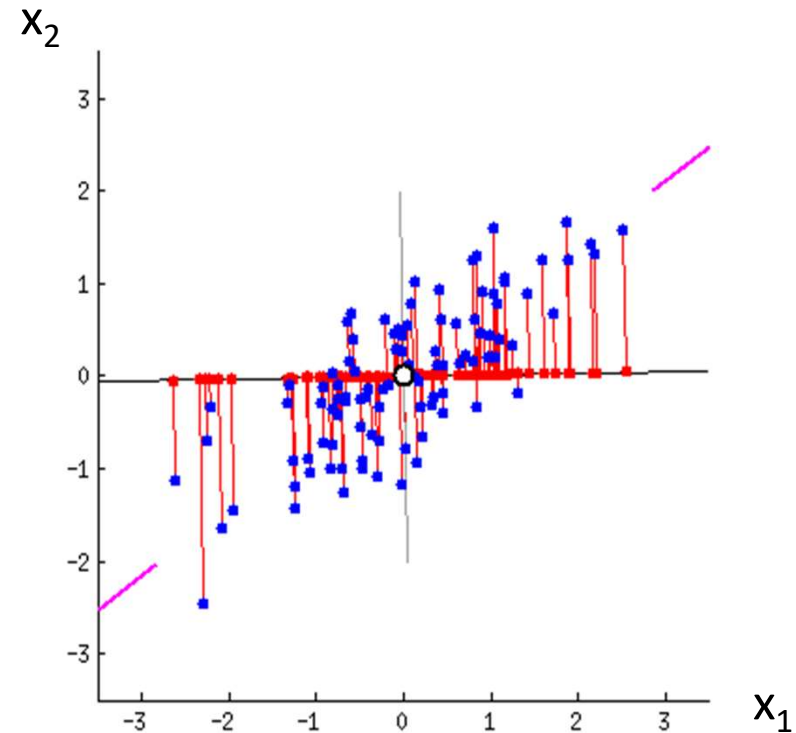
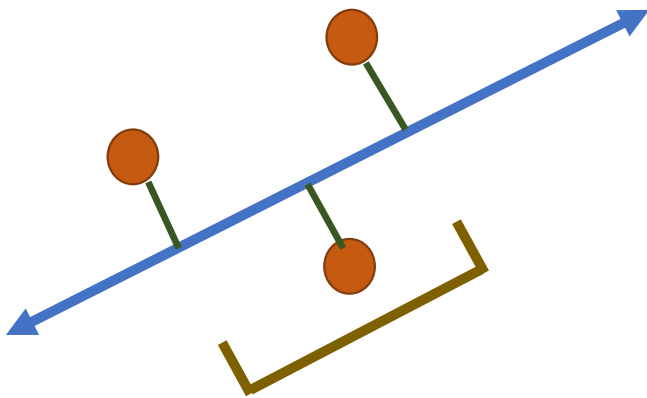


Principal Component Analysis (PCA)



Principal Component Analysis (PCA)

- PCA will find the best line depending on:
 - ✓ The variation of values along the line should be maximal (largest variance)
 - ✓ The error should be minimal (smallest projection error)



Source: <https://stats.stackexchange.com/>

Summary

- Given a set of points, how do we know that they can be expressed like the example before?
 - We need to look at the correlation between points
- How do we find the lines to keep / discard?
 - The tool we use is PCA
 - The directions are obtained by Eigen Analysis of the Covariance Matrix of the data
 - Another approach is to do Singular Vector Decomposition of data matrix
 - Either approach will give us the “best” directions to project the data to.
 - In general, the projection is to a lower dimensional sub-space

Variance

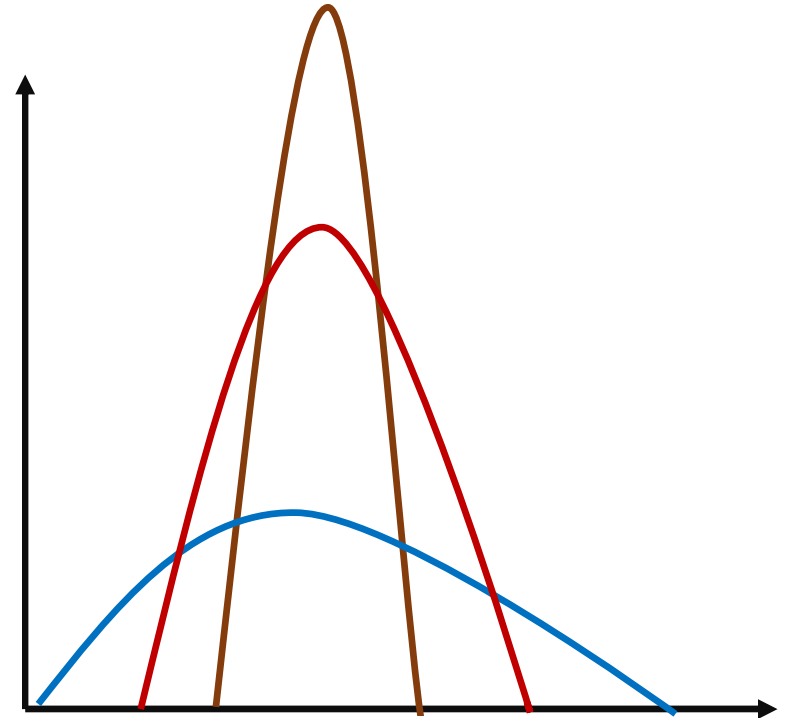
- The variance of a variable describes how much the values are spread out
- It quantifies the average squared difference between each data point and the mean of the dataset

$$s^2 = \frac{\sum (X - \bar{X})^2}{N}$$

Population Variance

$$s^2 = \frac{\sum (x - \mu)^2}{n - 1}$$

Sample Variance



Covariance

- Covariance tells us about the amount of dependency between two variables
- The covariance matrix summarizes the variances and covariances of a set of vectors

$$\begin{array}{ccc}
 M = \begin{bmatrix} 5 & 3 & 1 \\ 1 & 4 & 5 \\ 6 & 8 & 3 \end{bmatrix} & \xrightarrow{\text{Variance}} & \begin{bmatrix} 4.67 & \dots & \dots \\ \dots & 4.67 & \dots \\ \dots & \dots & 2.67 \end{bmatrix} \\
 \underbrace{\hspace{1.5cm}}_{\text{Covariance}} & & \begin{bmatrix} 4.67 & V(1,2) & \dots \\ V(2,1) & 4.67 & \dots \\ \dots & \dots & 2.67 \end{bmatrix}
 \end{array}$$

$$\begin{aligned}
 \mu &= \frac{3 + 4 + 8}{3} = 5 \\
 V &= \frac{(3 - 5)^2 + (4 - 5)^2 + (8 - 5)^2}{3} \\
 &= 4.67
 \end{aligned}$$

Covariance

$$M = \begin{bmatrix} 5 & 3 & 1 \\ 1 & 4 & 5 \\ 6 & 8 & 3 \end{bmatrix} \xrightarrow{\text{Covariance}} \begin{bmatrix} 4.67 & \text{cov}(1,2) & \text{cov}(1,3) \\ \text{cov}(2,1) & 4.67 & \text{cov}(2,3) \\ \text{cov}(3,1) & \text{cov}(3,2) & 2.67 \end{bmatrix}$$

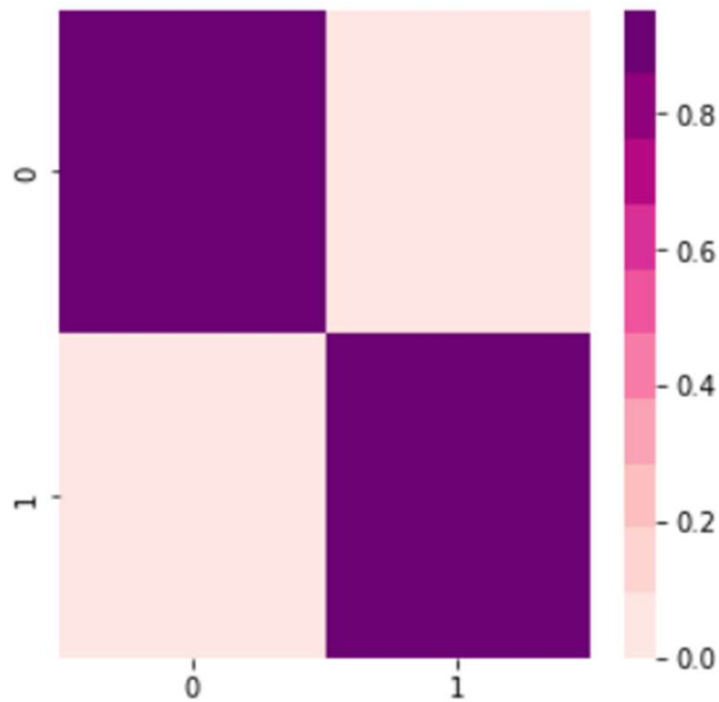
$$\text{cov}(X, Y) = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{n}$$

$$\begin{bmatrix} 4.67 & 2.33 & -2.67 \\ 2.33 & 4.67 & 0.67 \\ -2.67 & 0.67 & 2.67 \end{bmatrix}$$

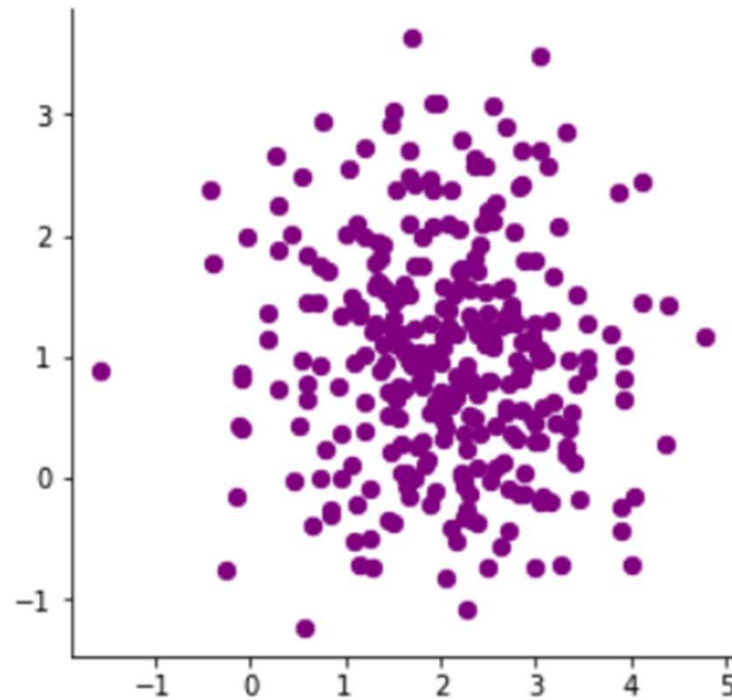
$$X = \begin{bmatrix} 5 \\ 1 \\ 6 \end{bmatrix}, \quad Y = \begin{bmatrix} 3 \\ 4 \\ 8 \end{bmatrix} \quad \bar{x} = 4, \bar{y} = 5$$

$$\begin{aligned} \text{cov}(X, Y) &= \frac{(5-4)(3-5) + (1-4)(4-5) + (6-4)(8-5)}{3} \\ &= 2.33 \end{aligned}$$

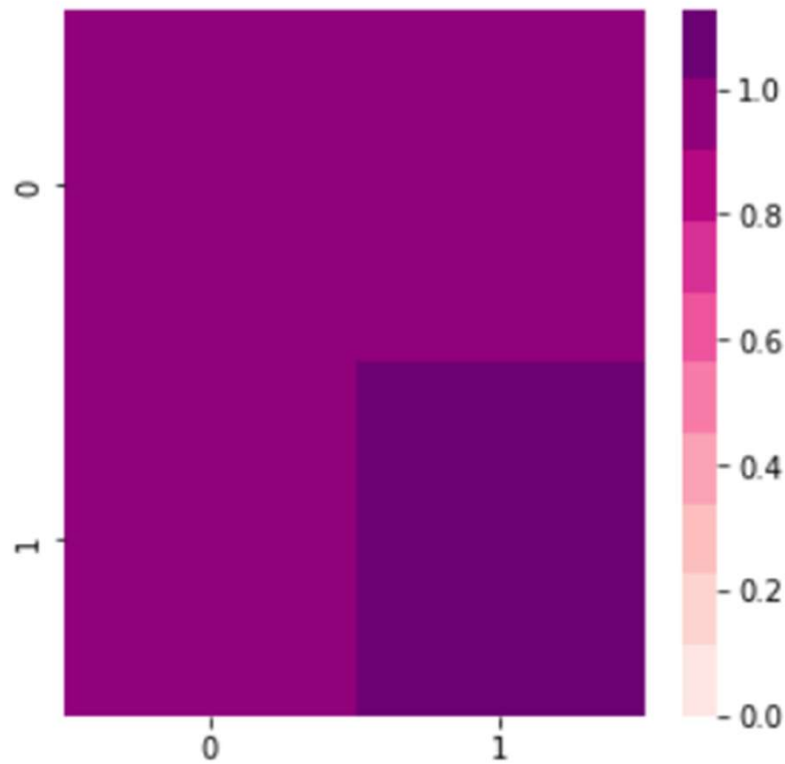
Covariance Matrix - Uncorrelated



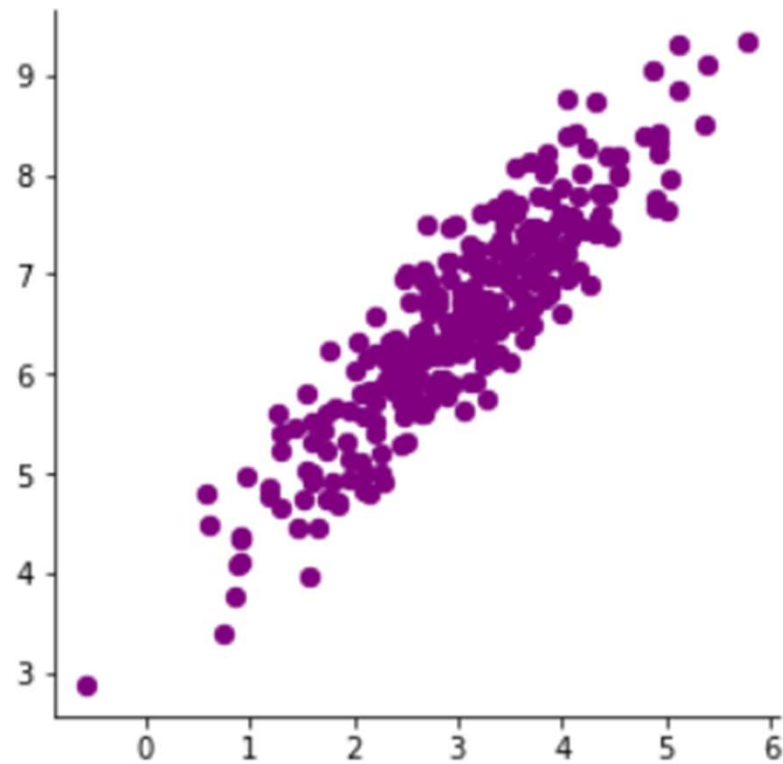
$$\text{cov}(X, Y) = \begin{bmatrix} 0.95 & -0.05 \\ -0.05 & 0.88 \end{bmatrix}$$



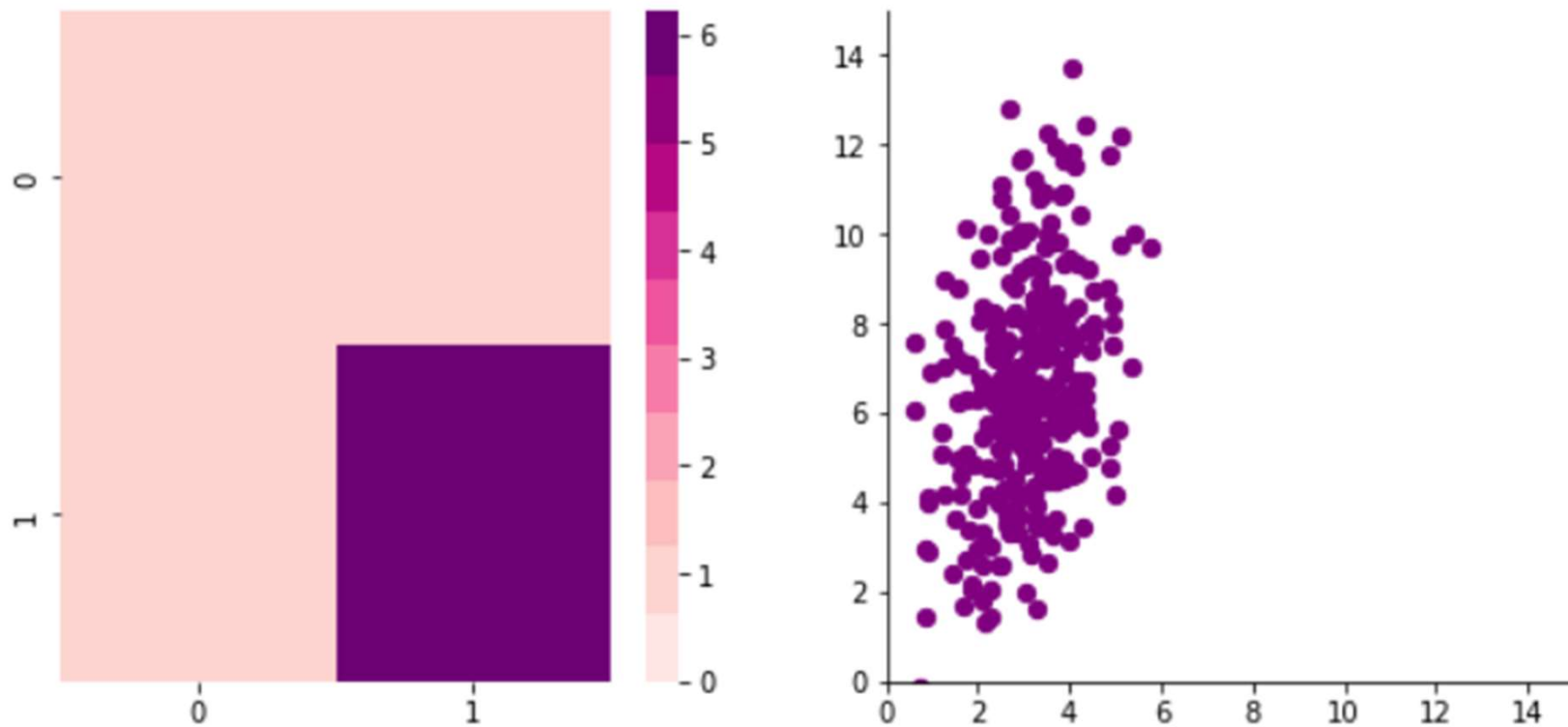
Covariance Matrix - Correlation



$$\text{cov}(X, Y) = \begin{bmatrix} 0.95 & 0.94 \\ 0.92 & 1.12 \end{bmatrix}$$

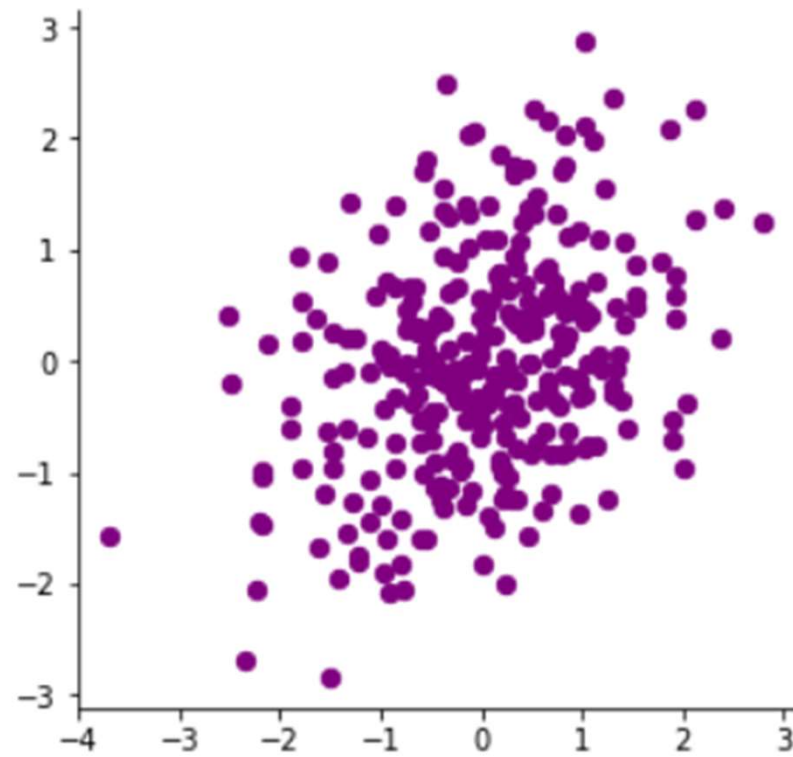
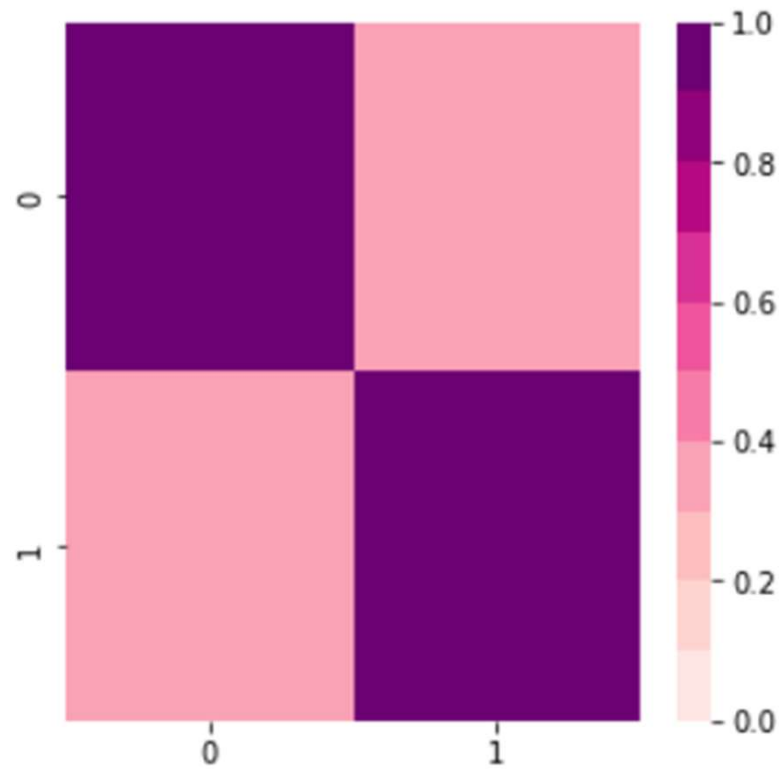


Covariance Matrix – Non-Standardized Data




$$\text{cov}(X, Y) = \begin{bmatrix} 0.95 & 0.84 \\ 0.84 & 6.24 \end{bmatrix}$$

Covariance/Correlation Matrix



$$\text{cov}(X, Y) = \begin{bmatrix} 1 & 0.35 \\ 0.35 & 1 \end{bmatrix}$$

Covariance: Dataset with m samples, n features

$$\begin{bmatrix} & X_1 & X_2 & X_3 & \dots & X_n \\ S_1 & q_{1,1} & q_{1,2} & q_{1,3} & \dots & q_{1,n} \\ S_2 & q_{2,1} & q_{2,2} & q_{2,3} & \dots & q_{2,n} \\ S_3 & q_{3,1} & q_{3,2} & q_{3,3} & \dots & q_{3,n} \\ & \dots & \dots & \dots & \dots & \dots \\ S_m & q_{m,1} & q_{m,2} & q_{m,3} & \dots & q_{m,n} \end{bmatrix}$$
$$cov(X_a, X_b) = \frac{1}{m} \sum_{i=1}^m (q_{i,a} - \overline{q_a})(q_{i,b} - \overline{q_b})$$

$$C = \begin{bmatrix} cov(X_1, X_1) & cov(X_1, X_2) & \dots & cov(X_1, X_n) \\ cov(X_2, X_1) & cov(X_2, X_2) & \dots & cov(X_2, X_n) \\ \dots & \dots & \dots & \dots \\ cov(X_n, X_1) & cov(X_n, X_2) & \dots & cov(X_n, X_n) \end{bmatrix}$$

n-dimensional Covariance Matrix

Summary

- Given a set of points, how do we know that they can be expressed like the example before?
 - We need to look at the correlation between points
- How do we find the lines to keep / discard?
 - The tool we use is PCA
 - The directions are obtained by Eigen Analysis of the Covariance Matrix of the data
 - Another approach is to do Singular Vector Decomposition of data matrix
 - Either approach will give us the “best” directions to project the data to.
 - In general, the projection is to a lower dimensional sub-space

Eigen-story

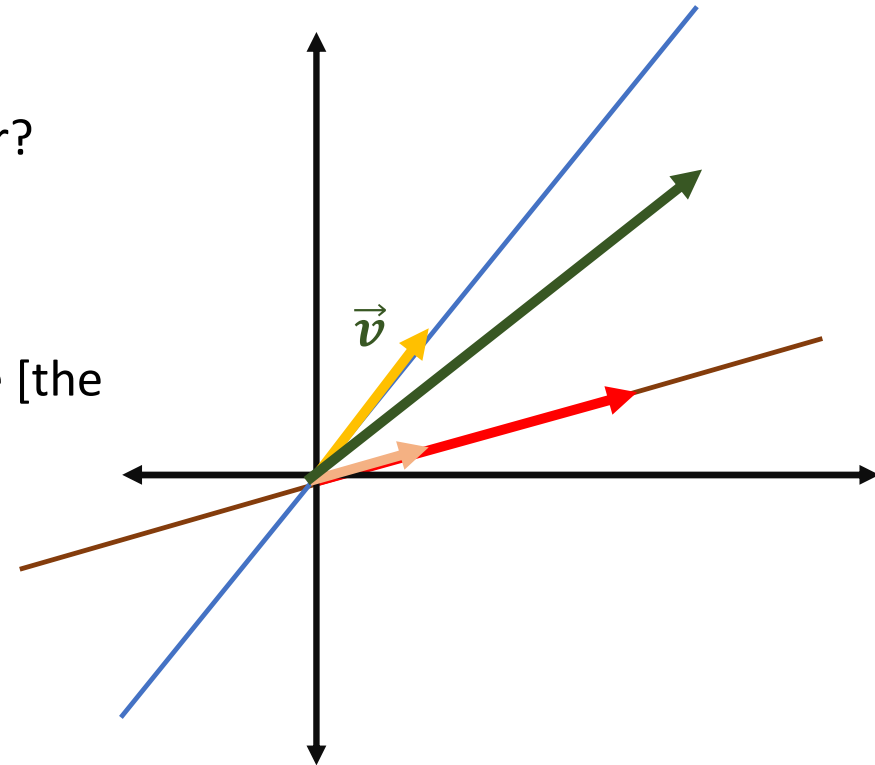
- What happens with a matrix operation on the vector?

$$A \vec{v} = \begin{bmatrix} 3 & 2 \\ 1 & 3 \end{bmatrix} \begin{bmatrix} 2 \\ 1 \end{bmatrix} = \begin{bmatrix} 8 \\ 5 \end{bmatrix}$$

- The vectors change their directions most of the time [the span changes]
- We have special cases where the vectors stay on the span [only getting scaled]

$$A \vec{v} = \begin{bmatrix} 1 & 3 \\ 3 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 4 \\ 4 \end{bmatrix} = 4 \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

- The vector here is the eigenvector and the scaling 4 is the eigenvalue



The span of a vector is a line $c \vec{v} \forall c \in \mathbb{R}$

$$A \vec{v} = \lambda \vec{v}$$

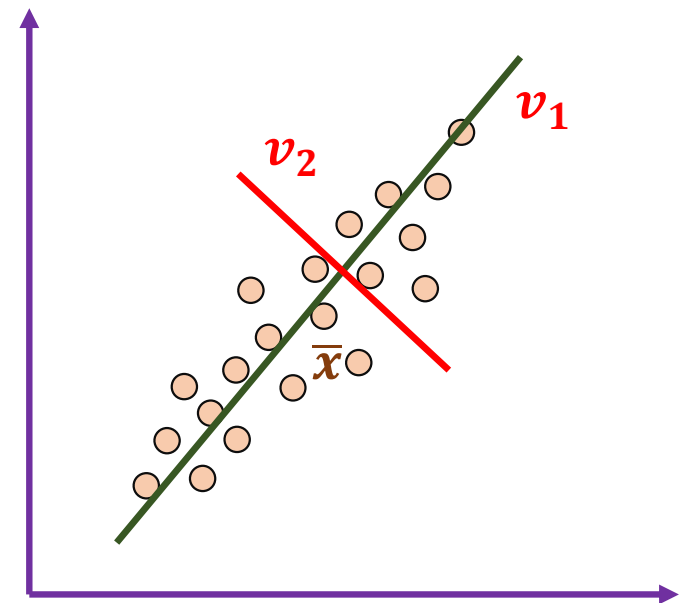
Demo - PCA

Let $x_1, \dots, x_n \in \mathbb{R}^d$ be the set of sample points you are dealing with. Projecting onto a unit vector $v \in \mathbb{R}^d$ you get the samples $z_i := x_i^T v \in \mathbb{R}$ for $i = 1, \dots, n$. The sample mean of this list of numbers is

$$\bar{z} = \frac{1}{n} \sum_i z_i = \frac{1}{n} \sum_i x_i^T v = \left(\frac{1}{n} \sum_i x_i^T \right) v = \bar{x}^T v,$$

where $\bar{x} = \frac{1}{n} \sum_i x_i$ is the sample mean of the unprojected data. Now the sample variance of the projected data is, by definition

$$\begin{aligned} \sigma_z^2 &= \frac{1}{n} \sum_i (z_i - \bar{z})^2 \\ &= \frac{1}{n} \sum_i \left(x_i^T v - \bar{x}^T v \right)^2 \\ &= \frac{1}{n} \sum_i \left((x_i - \bar{x})^T v \right)^2. \end{aligned}$$



\bar{x} is the mean of the points

Demo - PCA

- Consider the variation along a direction \mathbf{v} among all the points

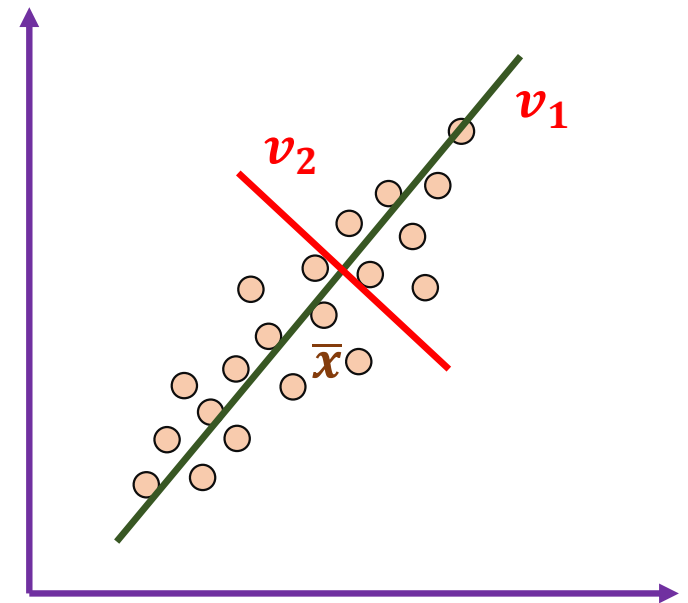
$$\mathit{var}(\mathbf{v}) = \frac{1}{n} \sum_{x \text{ points}} |(\mathbf{x} - \bar{\mathbf{x}})^T \cdot \mathbf{v}|^2$$

- Which unit vector \mathbf{v} maximizes var ?

$$\mathbf{v}_1 = \max_{\mathbf{v}} \{\mathit{var}(\mathbf{v})\}$$

- Which unit vector \mathbf{v} minimizes var ?

$$\mathbf{v}_2 = \min_{\mathbf{v}} \{\mathit{var}(\mathbf{v})\}$$



$\bar{\mathbf{x}}$ is the mean of the points

Demo - PCA

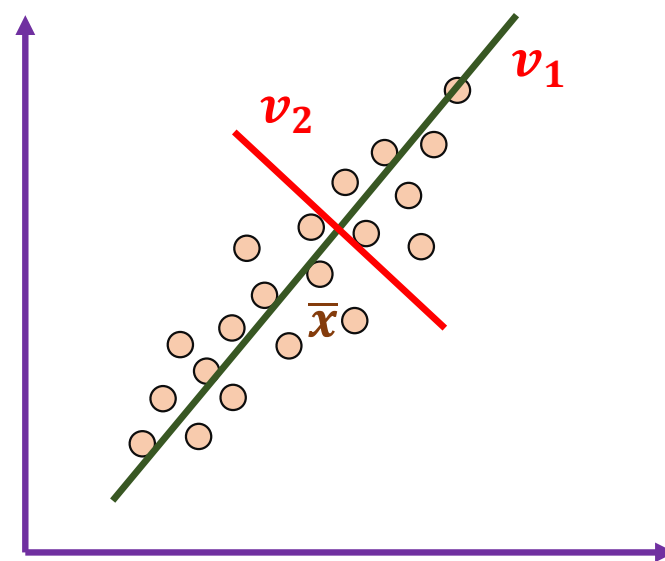
- Consider the variation along a direction \mathbf{v} among all the points

$$\begin{aligned}\text{var}(\mathbf{v}) &= \frac{1}{n} \sum_{x \text{ points}} |(\mathbf{x} - \bar{\mathbf{x}})^T \cdot \mathbf{v}|^2 \\ &= \frac{1}{n} \sum_{x \text{ points}} \mathbf{v}^T (\mathbf{x} - \bar{\mathbf{x}}) (\mathbf{x} - \bar{\mathbf{x}})^T \mathbf{v} \\ &= \frac{1}{n} \mathbf{v}^T \left[\sum_x (\mathbf{x} - \bar{\mathbf{x}}) (\mathbf{x} - \bar{\mathbf{x}})^T \right] \mathbf{v} = \mathbf{v}^T \mathbf{A} \mathbf{v}\end{aligned}$$

- Where \mathbf{A} is the covariance matrix $(\frac{1}{n} \sum_x (\mathbf{x} - \bar{\mathbf{x}}) (\mathbf{x} - \bar{\mathbf{x}})^T)$ of the datapoints

- ✓ \mathbf{v}_1 is the eigenvector of \mathbf{A} with largest eigenvalue
- ✓ \mathbf{v}_2 is the eigenvector of \mathbf{A} with smallest eigenvalue

iHub-Data-FMML 2023



The eigenvectors of the covariance matrix give you the direction that maximizes the variance. The direction of the green line is where the variance is maximum. Compare that with the projection on the red line: the spread is very small.

PCA

- $(XX^T)v = \lambda v$, with v as the eigenvector of the standardized covariance matrix XX^T
- The projection of the variance, $v^T XX^T v = \lambda v^T v = \lambda$
- The eigenvalue, λ denotes the amount of variability captured along that dimension
- Eigenvalues $\lambda_1 \geq \lambda_2 \geq \lambda_3 \geq \dots$
 - The **1st PC** v_1 is the eigenvector of the sample covariance matrix $X^T X$ associated with the largest eigenvalue
 - The **2nd PC** is the eigenvector of the sample covariance matrix $X^T X$ associated with the second largest eigenvalue
 - And so on ...

Singular Value Decomposition (SVD)

- SVD gives the decomposition for any arbitrary matrix, $\mathbf{M} = \mathbf{U} \mathbf{\Lambda} \mathbf{V}^T$

$$\mathbf{M}_{m \times n} = \mathbf{U}_{m \times r} \mathbf{\Lambda}_{r \times r} \mathbf{V}_{r \times n}^T$$

- ✓ $\mathbf{\Lambda}$ is the diagonal matrix equal to the root of the positive eigenvalues of $\mathbf{M}[\mathbf{X}^T \mathbf{X}$ or $\mathbf{X} \mathbf{X}^T]$
- ✓ \mathbf{U} and \mathbf{V} are the orthogonal matrices, $\mathbf{U}^T \mathbf{U} = \mathbf{1}, \mathbf{V}^T \mathbf{V} = \mathbf{1}$
- ✓ \mathbf{U} consists of orthonormal eigenvectors of $\mathbf{M}[\mathbf{X} \mathbf{X}^T]$
- ✓ \mathbf{V} consists of orthonormal eigenvectors of $\mathbf{M}^T[\mathbf{X}^T \mathbf{X}]$

Singular Value Decomposition (SVD)

- The SVD of the data matrix, $\mathbf{X} = \mathbf{U} \mathbf{\Lambda} \mathbf{V}^T$
- After standardization, the covariance matrix of the data matrix, $\Sigma = \frac{1}{m} \mathbf{X}^T \mathbf{X}$

$$\begin{aligned}\Sigma &= \frac{1}{m} \mathbf{X}^T \mathbf{X} = \frac{1}{m} (\mathbf{U} \mathbf{\Lambda} \mathbf{V}^T)^T (\mathbf{U} \mathbf{\Lambda} \mathbf{V}^T) = \frac{1}{m} (\mathbf{V} \mathbf{\Lambda}^T \mathbf{U}^T) (\mathbf{U} \mathbf{\Lambda} \mathbf{V}^T) \\ &= \frac{1}{m} (\mathbf{V} \mathbf{\Lambda}^T \mathbf{\Lambda} \mathbf{V}^T) = \frac{1}{m} (\mathbf{V} (\mathbf{\Lambda})^2 \mathbf{V}^T)\end{aligned}$$

➤ We can run SVD on \mathbf{X} without ever instantiating the large $\mathbf{X}^T \mathbf{X}$ to obtain the necessary principal components more efficiently

- Both \mathbf{X} and $\mathbf{X}^T \mathbf{X}$ share the same eigenvectors in their SVD

Singular Value Decomposition (SVD)

- The SVD of the data matrix, $\mathbf{X} = \mathbf{U} \mathbf{\Lambda} \mathbf{V}^T$
- After standardization, the covariance matrix of the data matrix, $\Sigma = \frac{1}{m} \mathbf{X}^T \mathbf{X}$

$$\begin{aligned}\Sigma &= \frac{1}{m} \mathbf{X}^T \mathbf{X} = \frac{1}{m} (\mathbf{U} \mathbf{\Lambda} \mathbf{V}^T)^T (\mathbf{U} \mathbf{\Lambda} \mathbf{V}^T) = \frac{1}{m} (\mathbf{V} \mathbf{\Lambda}^T \mathbf{U}^T) (\mathbf{U} \mathbf{\Lambda} \mathbf{V}^T) \\ &= \frac{1}{m} (\mathbf{V} \mathbf{\Lambda}^T \mathbf{\Lambda} \mathbf{V}^T) = \frac{1}{m} (\mathbf{V} (\mathbf{\Lambda})^2 \mathbf{V}^T)\end{aligned}$$

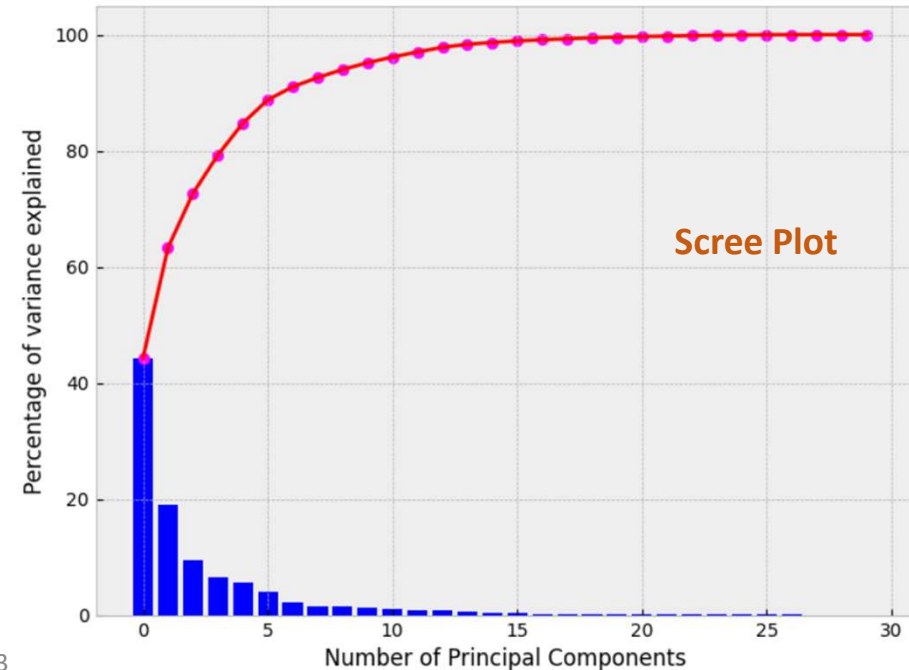
- $(\mathbf{\Lambda})^2$ is a diagonal matrix whose entries are $\Lambda_{ii} = \lambda_i^2$, the squares of the eigenvalues of the SVD of \mathbf{X}
- Both \mathbf{X} and $\mathbf{X}^T \mathbf{X}$ share the same eigenvectors in their SVD

How many PCs

- A dataset with **m samples** and **n features**, will give rise to a **$n \times n$** covariance matrix.
- The **$n \times n$** covariance matrix will have **n** eigenvectors, so **n** PCs.

n features \longrightarrow **n principal components**

- Where does dimensionality reduction come from?
- Can always ignore the components of lesser significance



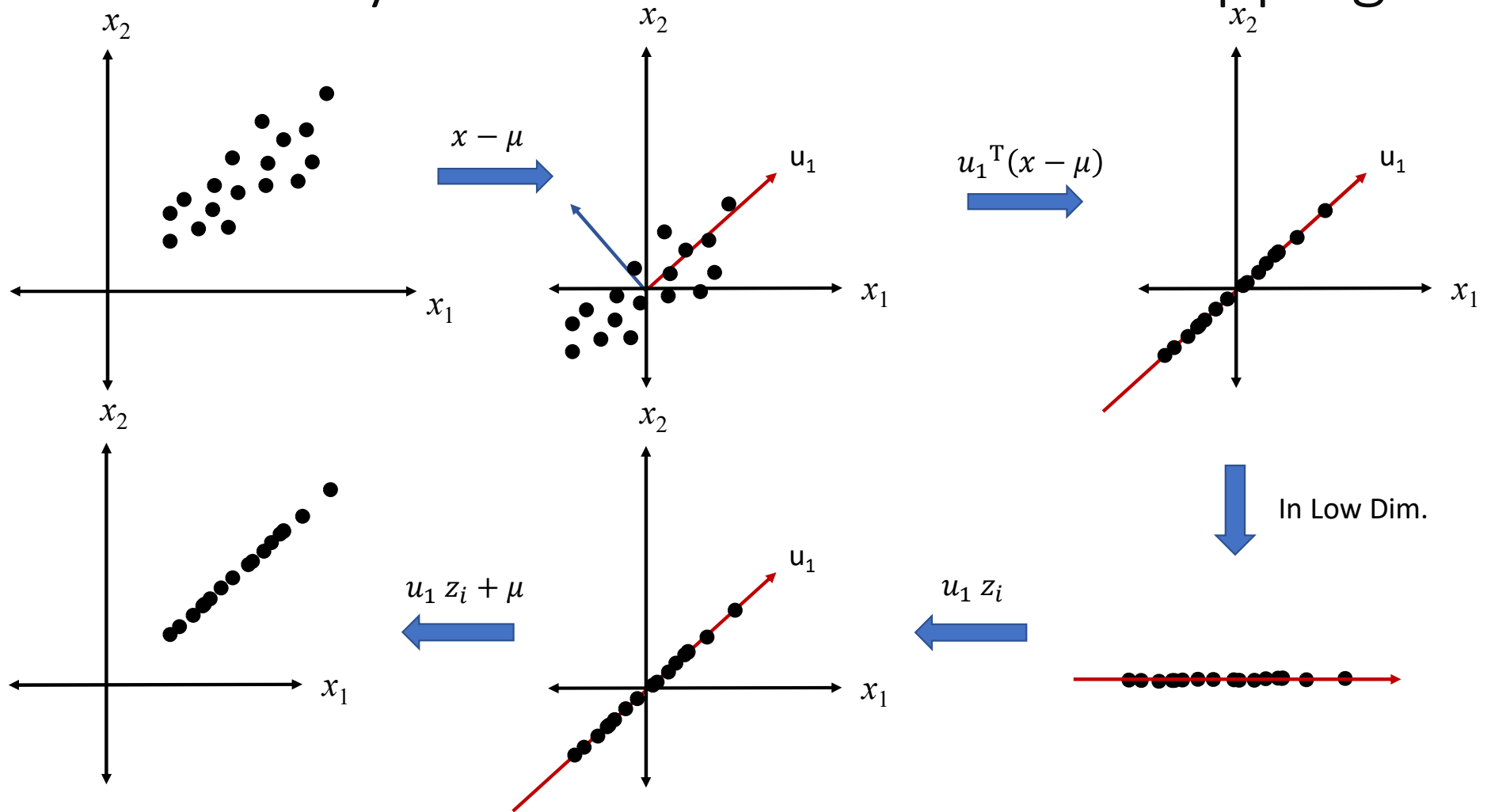
How many PCs

- You do lose some information, but if the eigenvalues are small, you don't lose much
 - n dimensions in original data
 - calculate n eigenvectors and eigenvalues
 - choose only the first D eigenvectors, based on their eigenvalues
 - final data set has only D dimensions

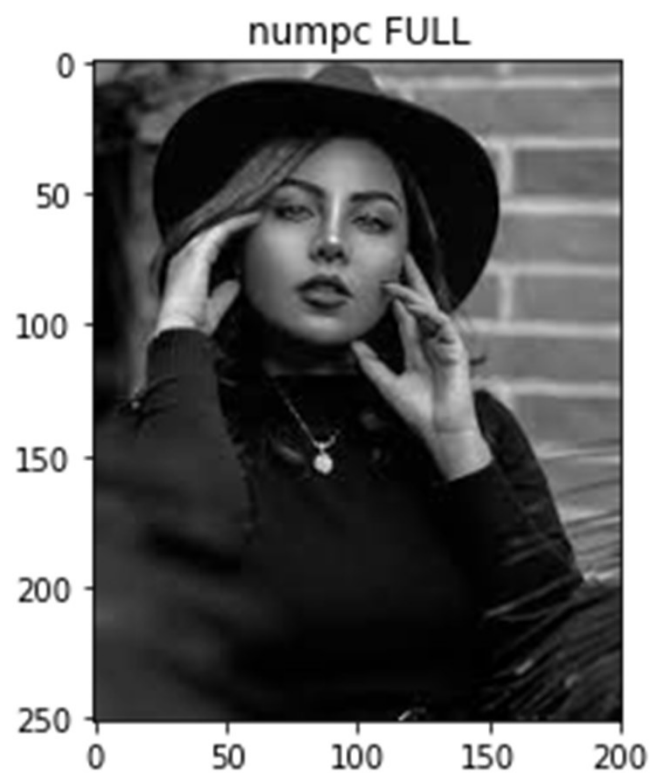
Step by Step Computation

- ❑ Step 1: Standardization of the data
- ❑ Step 2: Compute the covariance matrix
- ❑ Step 3: Calculate the eigenvalues and the eigenvectors of the covariance matrix
- ❑ Step 4: Compute the principal components by selecting the first D eigenvectors
- ❑ Step 5: Reduces the dimensions of the dataset

Dimensionality Reduction and Inverse Mapping



Example



PCs # 0



PCs # 10



PCs # 20



PCs # 30



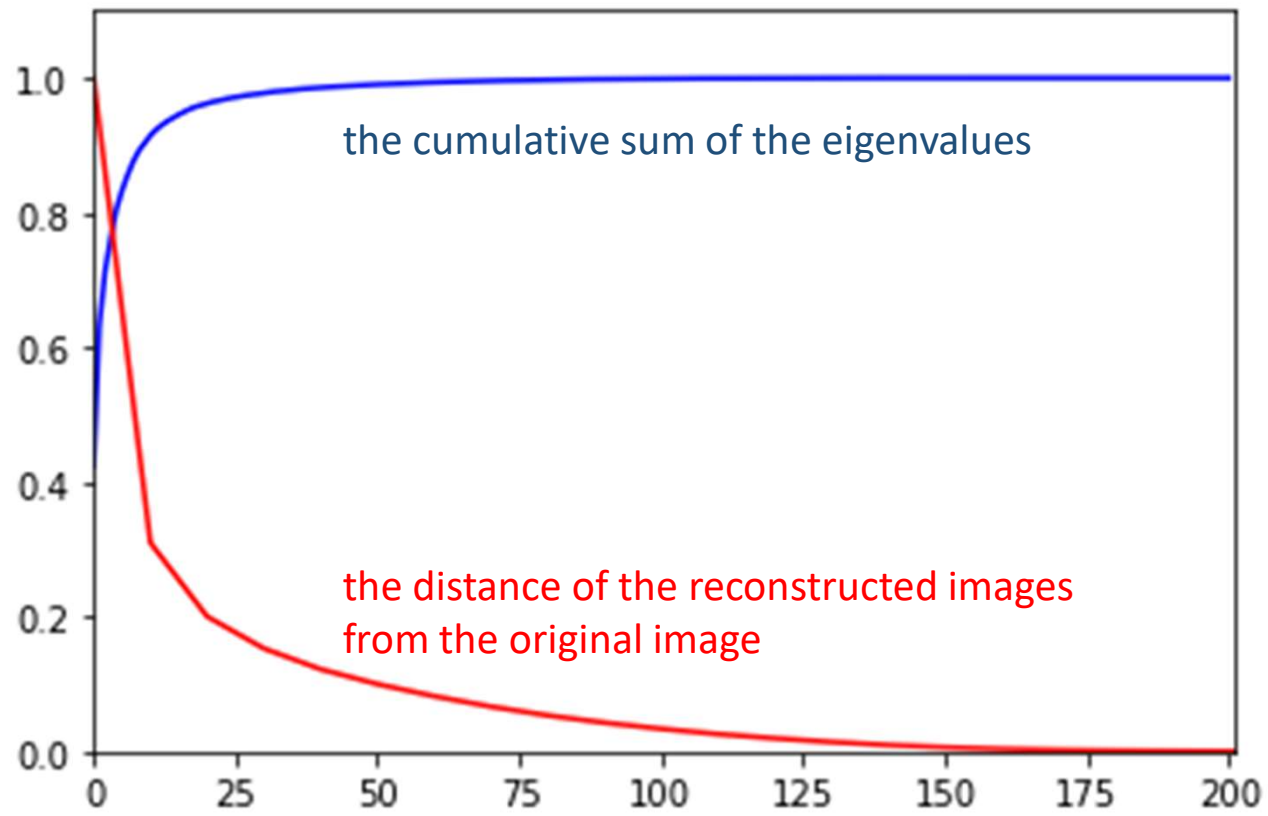
PCs # 40



PCs # 50



Example



Principal Component Analysis: Summary

- PCA allows us to find the highest variance (lowest sq. distance) direction to project to
- Eigen values gives an indication of the number of dimensions to choose.
- Can be computed in multiple ways (SVD is popular)
- Is an unsupervised algorithm
- Ensure pre-processing for effectiveness
- Is used in a variety of applications