# Introduction to Unsupervised Learning

## 1 Unsupervised Learning Overview

Unsupervised learning is a class of machine learning techniques aimed at finding patterns or structure in data without relying on labelled examples. The primary tasks in unsupervised learning include:

- Clustering: Grouping similar data points together.
- Dimensionality Reduction: Reducing the number of features while preserving as much variance as possible.

In this document, we introduce clustering methods with a focus on the k-means algorithm, and discuss Principal Component Analysis (PCA) using Singular Value Decomposition (SVD) for dimensionality reduction.

## 2 Clustering

Clustering is the task of dividing a dataset into groups, or clusters, such that data points within a cluster are more similar to each other than to those in other clusters. It aims to partition $N$ data points in $D$ dimensional space into $k$ clusters by minimizing the within-cluster sum of squared distances. The input is an unlabelled data set $\langle \mathbf{x}_i \rangle_{i=1}^{N}$, the goal is to output a partition $C_1, \dots C_k$ of the data set $\{\mathbf{x}_1, \dots \mathbf{x}_N\}$, where $C_i$ are non-empty, pairwise disjoint (share no common elements) and their union covers the entire data set.

### 2.1 The k-means Algorithm

The k-means algorithm is one of the most widely used and simplest clustering techniques. First we assign k-means $\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \dots, \boldsymbol{\mu}_k$ which are simply points in the euclidean space $\mathbb{R}^D$, each $\boldsymbol{\mu}_i$ representing the centre of one cluster. Each data point is then assigned to a cluster based on which mean $\boldsymbol{\mu}_i$ is is closest to in the euclidean space. The cluster means $\boldsymbol{\mu}_i$ are then update based on the new partition $C_1, \dots C_k$.

The steps of the algorithm are as follows:

1. **Initialization:** Randomly initialize $k$ cluster centroids, $\boldsymbol{\mu}_1, \boldsymbol{\mu}_2, \dots, \boldsymbol{\mu}_k$.
2. **Assignment Step:** Assign each data point $\mathbf{x}_i$ to the nearest cluster centroid:

$$C_j = \{\mathbf{x}_i : \|\mathbf{x}_i - \boldsymbol{\mu}_j\|^2 \leq \|\mathbf{x}_i - \boldsymbol{\mu}_l\|^2, \forall l = 1, \dots, k\}.$$

3. **Update Step:** Update the centroids by computing the mean of the points assigned to each cluster:
$$\mu_j = \frac{1}{|C_j|} \sum_{\mathbf{x}_i \in C_j} \mathbf{x}_i.$$

4. Repeat the assignment and update steps until convergence (e.g., when cluster assignments no longer change).

The k-means algorithm assumes that clusters are spherical and separable, making it well-suited for simple, low-dimensional data.

### 2.2 Advanced Clustering Methods

For more complex data, advanced clustering methods may be more appropriate. These include:

- **Hierarchical Clustering:** Builds a hierarchy of clusters and is suitable for data with a nested structure.
- **DBSCAN (Density-Based Spatial Clustering of Applications with Noise):** Identifies clusters of arbitrary shape and is robust to noise in spatial data.
- **Gaussian Mixture Models (GMM):** Assumes data is generated from a mixture of Gaussian distributions, making it suitable for probabilistic modelling of data.
- **Spectral Clustering:** Uses the eigenvalues of a similarity matrix to perform clustering, ideal for data with non-convex structures.

## 3 Principal Component Analysis (PCA) Using SVD

Principal Component Analysis (PCA) is a technique for dimensionality reduction that identifies the directions (principal components) in which the data varies the most. PCA transforms data into a lower-dimensional space while preserving as much variance as possible. Here is a great video explaining this concept `https://www.youtube.com/watch?v=FD4DeN81ODY`

### 3.1 Problem Setting

Given a dataset $\mathbf{X} \in \mathbb{R}^{N \times D}$ (where $N$ is the number of data points and $D$ is the number of features), PCA seeks a set of orthonormal vectors $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_k$ ($k \leq D$) such that the projections of $\mathbf{X}$ onto these vectors maximize the variance. Mathematically, this involves solving:
$$\max_{\mathbf{v} \in \mathbb{R}^D, \|\mathbf{v}\|=1} \text{Var}(\mathbf{X}\mathbf{v}),$$
where $\text{Var}(\mathbf{X}\mathbf{v})$ represents the variance of the data along direction $\mathbf{v}$.

## 3.2 Singular Value Decomposition (SVD)

The Singular Value Decomposition (SVD) provides a computationally efficient way to solve PCA. For a centred dataset $\mathbf{X}$ (i.e., mean subtracted), SVD decomposes $\mathbf{X}$ as:

$$\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T,$$

where:

- $\mathbf{U} \in \mathbb{R}^{N \times N}$ is an orthogonal matrix whose columns are the left singular vectors.
- $\mathbf{\Sigma} \in \mathbb{R}^{N \times D}$ is a diagonal matrix containing the singular values $\sigma_1, \sigma_2, \ldots, \sigma_r$ ($r = \min(N, D)$) in descending order.
- $\mathbf{V} \in \mathbb{R}^{D \times D}$ is an orthogonal matrix whose columns are the right singular vectors.

The principal components are the first $k$ columns of $\mathbf{V}$, and the corresponding explained variance is proportional to the square of the singular values ($\sigma_i^2$).

## 3.3 Steps in PCA using SVD

1. Center the data matrix $\mathbf{X}$ by subtracting the mean of each feature.
2. Compute the SVD of $\mathbf{X}$: $\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$.
3. Select the first $k$ columns of $\mathbf{V}$ (right singular vectors) as the principal components.
4. Project $\mathbf{X}$ onto the selected principal components to obtain the reduced dataset.

The mathematical details for why this works, and what a good choice of $k$ might be are quite technical. The key takeaway is to understand that SVD is the most common way of performing PCA, which is a linear dimensionality reduction technique.