# § Methods of Unsupervised Learning

## Introduction

Most of the available work in machine learning is done based upon supervised learning. On the other hand, most of the data available in the world is unlabeled and with little or no hope to become labelled. Thus, most of the data available in the world is not supervised learning material.

Supervised learning makes predictions using models trained on ordered pairs $\mathcal{D} = \{(\vec{x}_i, y_i)\}_{i=1}^{N}$ that results as the cartesian product of the vectors in the feature vector space $X = \{\vec{x}_i\}_{i=1}^{N}$ and the target vector space $Y = \{y\}_{i=1}^{N}$. Both both feature and target are random variables $(X, Y) \in \mathbb{R}^{N \times D} \otimes \mathbb{R}^{N}$, where $D$ is the number of features and $N$ the number of observations, described by a joint probability density $P(X, Y)$ and the goal is to learn a model that can predict the value or infer the form of $P(Y|X)$, using a variational approach over a loss function $L$:

$$(1) \quad p(\alpha) = \underset{\vartheta}{\operatorname{argmin}} \, E_{Y|X}\left(L(Y, \hat{Y})\right)$$

In unsupervised learning, we have only the feature vector space $X = \{\vec{x}_i\}_{i=1}^{N}$, distributed along a joint probability distribution $P(\vec{x})$. In general, dim $(X)$ is much higher than in supervised learn, both in number of observations and number of variables. and one is interested in much more complex properties than in a prediction. In unsupervised methods, we are interested on the bare properties of $X$, much beyond the methods of descriptive statistics.

The most known supervised methods:

i) Cluster Analysis

ii) Principal component Analysis

iii) Density Estimation

Cluster analysis attempts to find multiple convex regions in $d$-space, containing representative modes of $P(X)$. Principal component analysis and its associates, such as self-organizing maps, multidimensional scaling and principal surfaces looks for

low-dimensional manifolds in X that are represented by high density data. This provides information about associations between data and wether or not they can be described as a combination of small set of latent variables. Mixture Models attempts to find the modes of a probability density as a combination of trial known distributions.

## § Clustering Methods

Clustering is a collection of methods used to split a data sample into groups, so that the elements of each group are similar between them and different from the members of other groups.

The most used clustering methods are based on the so-called dissimilarity matrix, a $N \times N$ pairwise matrix $D_{ij} \geq 0$ where the entries are point distances of each datapoint, one-by-one.

Formally, we want to partition the space into $K$ subsets, containing groups of data points, so that:

$$(2) \quad D = \bigcup_{k=1}^{K} C_k$$

$$(3) \quad C_i \cap C_j = \phi \quad \forall \; i \neq j$$

where $C_k$ is the $k$-th partition of the feature space.

The dissimilarity matrix is defined based on a distance measure of each datapoint:

(4) $$D(x_i, x_j) = \sum_{q=1}^{D} d_q(x_{iq}, x_{jq})$$

where the sum runs over all features of each datapoint. For continuous qualitative variables, the most common choice is the Euclidean Distance:

(5) $$d^2(x_i, x_j) = \sum_{q=1}^{D} (x_{iq} - x_{jq})^2$$

In situations where similarities are more interesting than dissimilarities, Pearson correlation can be a good choice:

(6) $$\rho(x_i, x_j) = \frac{\sum_q (x_{iq} - \bar{x}_i)(x_{jq} - \bar{x}_j)}{\sqrt{\sum_q (x_{iq} - \bar{x}_i)^2 \sum_q (x_{jq} - \bar{x}_j)^2}}$$

Also called similarity based clustering. The so-called Mahalanobis measure is a good call in these correlated scenario.

## § Object Dissimilarity

There are many possibilities in measuring dissimilarities, although Euclidean distance still is the most known, it is only suited to quantitative variables. Most of the works upon clustering use four metrics:

- Euclidean Distance: $d_q^2(x_i, x_j) = (x_{iq} - x_{jq})^2$
- Mahalanobis Distance: $d_q(x_i, x_j) = |x_{iq} - x_{jq}|$
- Cosine Distance: $d(\vec{x_i}, \vec{x_j}) = 1 - \dfrac{\vec{x_i} \cdot \vec{x_j}}{\|\vec{x_i}\| \, \|\vec{x_j}\|}$
- Jaccard Distance: Let $x_i, x_j \in [0,1]$, we have:

$$d_q(x_1 x_2) = 1 - \frac{x_{iq} x_{jq}}{\sum_q x_{iq} + \sum_q x_{jq} - \sum_q x_{iq} x_{jq}}$$

The total dissimilarity of a pair $(x_i, x_j)$ is:

$$(7) \qquad D(x_i x_j) = \sum_{q=1}^{D} w_q \, d_q(x_i x_j)$$

where $w_q$ is the weight of the $q$-th feature.

Mahalanobis distance has a very interesting property of taking the correlation between the variables into account.

In fact, let $\vec{x}_i = \{x_{iq}\}_{q=1}^{D}$ and $\vec{x}_j = \{x_{jq}\}_{q=1}^{D}$ be two datapoints, the Mahalanobis distance can be written as:

$$(8)\quad d(\vec{x}_i, \vec{x}_j) = \sqrt{(\vec{x}_i - \vec{x}_j)^T \overleftrightarrow{S} (\vec{x}_i - \vec{x}_j)}$$

is the covariance matrix, that is related with Pearson's correlation:

$$(9)\quad \overleftrightarrow{\rho}(\vec{x}_i, \vec{x}_j) = \sigma_{\vec{x}_i} \sigma_{\vec{x}_j} \overleftrightarrow{S}(\vec{x}_i, \vec{x}_j)$$

Now, the weight of a variable has to do with a few things:

a) A choice of which variables are more relevant.

b) A geometrical aspect of the dataset

c) A difference in unit and scales between data.

The first one is not of mathematical reasons, so will not be treated. The last one is a common mistake in many data science projects. Use variables of different scales and units may conduct to mis interpretation of the relevance of the variable in the dissimilarity calculation. This can be easily solved by doing a proper preprocessing.

Of much more interesting is the geometrical view point. In physics, we often encounters such problems that is of analytical solution only in a specific systems of coordinates these cases, are those ones in which the phase-space has an specific symmetry to be exploited in the solution of the problem. In here, care must be taken. The obscurse of Liouville's theorem may turn difficult a canonical transformation, or then the geometrical aspect has to be compensated with the weight of each variable. However, most of the cases, one may consider the latent phase space to be uniform.

The average dissimilarity of the entire dataset is:

$$(10)\quad \bar{D} = \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N} D(x_i x_j)$$

$$= \sum_{q=1}^{p} w_q \bar{d}_q$$

where

$$(11)\quad \bar{d}_q = \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N} d_q(x_i x_j)$$

under the assumption of a uniform phase space:

$$(12)\quad w_q = 1/\bar{d}_q$$

To a quantitative variable, Euclidean distance gives:

$$(13)\quad \bar{d}_q = \frac{1}{N^2}\sum_i\sum_j (x_{iq} - x_{jq})^2 = 2\,Var(\vec{x}_q)$$

## § Clustering Algorithm

There are three main problems to solve:

1- How to define the clusters

2- How to assign points to clusters

3- How to define the number of clusters.

Since we don't have an objective metric to be optimized. We need to relies upon heuristic arguments to both motive and justify our unsupervised learning algorithm.

The methodology includes successive trials of the same algorithm with different numbers of clusters and combinatorial optimization

# Combinatorial Algorithms

A clustering algorithm wants to group together the closer points to a given reference point, so that the elements of the cluster are much alike as possible and the most different to the elements of other clusters as possible.

Each observation is labelled with an univoque index $i \in \mathbb{N}$ and the number of clusters is of ad-hoc nature, $k < N$, and each datapoint is assigned to one cluster and only.

This assignment is performed by a many-to-one map, called <u>encoder</u>, $k = C(i)$, that address the $i$-th observation to the $k$-cluster. A encoder is choosen as the best solution of a loss function and a given dissimilarity matrix, both specified by the user. In this kind of problem, a loss function in order to assure that the cluster requirement are met.

One possible approach is to direct specify a mathematical representation of a loss function and attempt a solution through combinatorial optimization. The goal is to collect the close points into some cluster, so that they are as much alike as possible of each other and less alike as possible from the points of the other clusters. A natural loss function would be:

$$(14)\ W(C) = \frac{1}{2}\sum_{k=1}^{K}\sum_{C(i)=u}\sum_{C(j)=u} d(x_i, x_u)$$

the so-called within clustering scattering rate, and measures the sum of distances of the intracluster points of all clusters. The complete metric is the Total Clustering scattering rate:

$$(15)\ T(C) = \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} d_{ij}$$

$$= \frac{1}{2}\sum_{k=1}^{K}\sum_{C(i)=u}\left(\sum_{C(j)=u} d_{ij} + \sum_{C(j)\neq u} d_{ij}\right)$$

$$= W(C) + B(C)$$

where $B(C)$ is the between or out clustering scattering rate. The optimal cluster, is the one that minimizes $W(C)$ and maximize $B(C)$.