# Autoencoders research

Lorenzo Bozzoni

Politecnico di Milano

February 2024

# Table of content

# General framework

The general framework of autoencoders is:

$$\mathcal{X} \ni x_t = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \xrightarrow{B} \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_p \end{bmatrix} \xrightarrow{A} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = y_t \in \mathcal{Y}$$

Where $B \in \mathcal{B}$ which is a set of functions from $\mathbb{F}^n$ to $\mathbb{G}^p$ while $A \in \mathcal{A}$ which is a set of functions from $\mathbb{G}^p$ to $\mathbb{F}^n$.

# General framework

The goal is to find a pair of functions $A, B$ such that the generic dissimilarity function $\Delta$ is minimized:

$$\min E(A,B) = \min_{A,B} \sum_{1}^{m} E(x_t, y_t) = \min_{A,B} \sum_{1}^{m} \Delta(A \circ B(x_t), y_t)$$

In the auto-associative case the right side of the autoencoder is again $x_t$. **The focus is not on the reconstruction of the input but rather on how well we can compress the input data in the hidden layer without losing information.**

# Linear autoencoders

In the case of **linear autoencoders** we have:

- $\mathbb{F}, \mathbb{G}$ are fields
- $\mathcal{A}, \mathcal{B}$ are the classes of linear transformations: $A, B$ are respectively matrices of shape $p \times n$ and $n \times p$
- $\Delta$ is the squared Euclidean distance ($L_2^2$ norm)

# Linear autoencoders

In general the problem of finding the matrices $A, B$ that minimize the error function $E$ is a non-convex optimization problem.

However, fixing one of the two matrices, the problem becomes convex so **we can find the optimal value by alternating the optimization of the two matrices.** Fixing $A$ the optimal $B$ is:

$$B = \hat{B}(A) = (A^\intercal A)^{-1} A^\intercal$$

While fixing $B$ the optimal $A$ is:

$$A = \hat{A}(B) = \Sigma_{XX} B^\intercal (B \Sigma_{XX} B^\intercal)^{-1}$$

Where $\Sigma_{XX}$ is the covariance matrix of the input data.

# Linear autoencoders

- Since we are applying only linear transformations, the best compression we can achieve is the one that projects the input data on the subspace spanned by the eigenvectors of the covariance matrix of the input data.
- This corresponds to the **Principal Component Analysis (PCA)** when the input is normalized as follows:

$$\hat{x}_{i,j} = \frac{1}{\sqrt{m}} \left( x_{ij} - \frac{1}{m} \sum_{k=1}^{m} x_{kj} \right)$$

# Boolean autoencoders

In the case of **Boolean autoencoders** we have:

- $\mathbb{F}, \mathbb{G}$ are the Boolean fields, i.e $\{0, 1\}$, the Galois field $\mathbb{F}_2$
- $\mathcal{A}, \mathcal{B}$ are the classes of Boolean transformations: $A, B$ are respectively matrices of shape $p \times n$ and $n \times p$ with entries in $\{0, 1\}$
- $\Delta$ is the Hamming distance