The three types of machine learning

- Supervised learning

The algorithm is trained on a labeled dataset, where the input data is paired with the correct output.
The goal is to learn the mapping between the input and output, so it can make predictions on new, unseen data.

- Labeled data
- Direct feedback
- Predict outcome / future

Examples:
- Regression (predicting numerical values)
- Classification (predicting categorical ~~tablest~~ labels)

- Unsupervised learning

The algorithm is trained on an unlabeled dataset, meaning there are no predefined correct outputs.
The goal is to discover hidden patterns and structures within the data.

- No labels
- No feedback
- Find hidden structures

Examples:
- Clustering (grouping similar data points together)
- Dimensionality reduction (reducing the number of features in the data)

- Reinforcement Learning

The algorithm learns by interacting with an environment. It takes actions, receives rewards or penalties based on the outcome, and adjusts its behavior to maximize rewards over time.

~~Examp~~

- Decision process
- Reward system
- Learn series of action

Examples:

- Game playing
- Robotics
- Autonomous systems

How does supervised learning (classification) work :

- Given : for the random pair (X, Y) in $\mathbb{R}^d \times \{0, 1\}$ consisting of a random observation X and its random binary level Y (class
  a sample of n i.i.d. : $(x_1, y_1), \ldots, (x_n, y_n)$
- Goal : predict the label of the new (unseen before) observation x
- Method : construct a classification rule

$$g : \mathbb{R}^d \longrightarrow \{0, 1\}, \; x \longmapsto g(x)$$

  so $g(x)$ is the prediction of the label for observation x
- Criterion : of the performance of g, it is the error probability

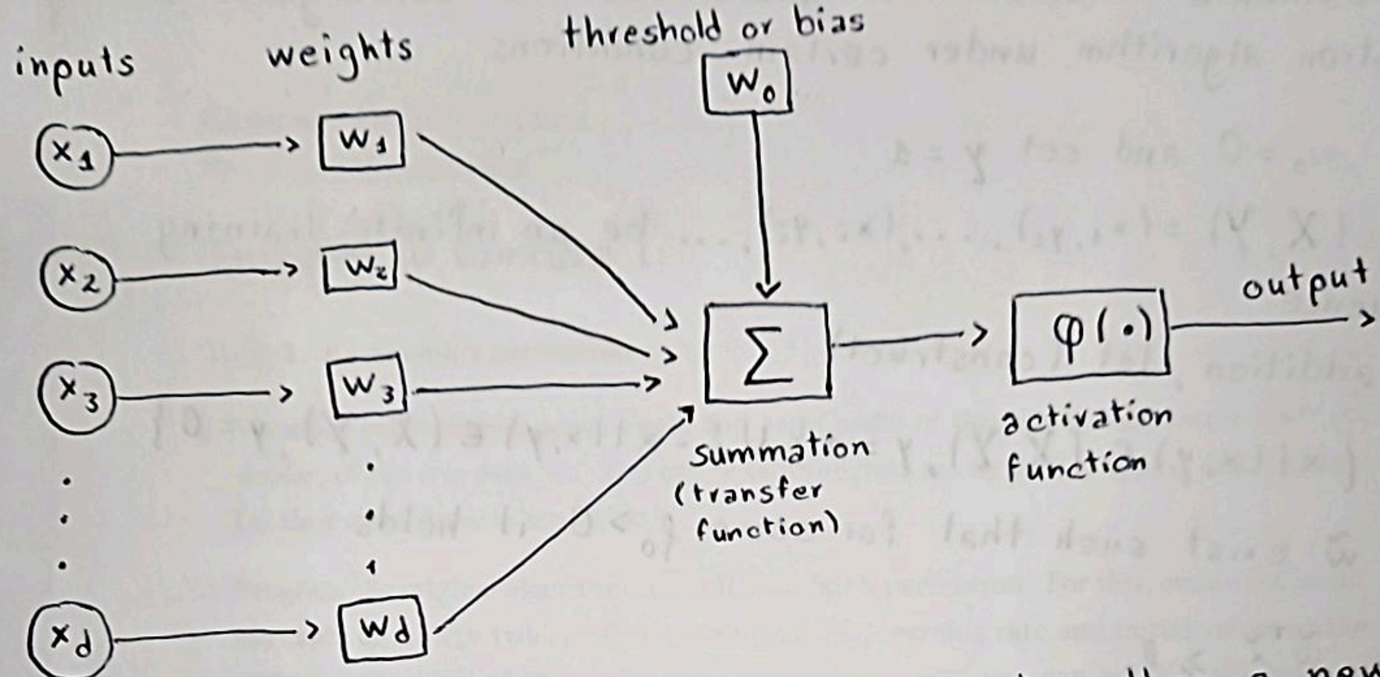
$$\mathbb{P}(g(X) \neq Y) = \mathbb{E}[\mathbb{1}(g(X) \neq Y)]$$

- The best solution : it is to Know the distribution of (X, Y)

$$g(x) = \mathbb{1}(\mathbb{E}[Y | X = x] > 0{,}5)$$

such $g(x)$ is Known as the Bayes classifier

Rosenblatt's perceptron algorithm



Let $w = (w_1, w_2, \ldots, w_d)^T$ be the weight vector, then a new observation $x = (x_1, x_2, \ldots, x_d)^T$ is classified as

$$g(x) = \begin{cases} 1 & \text{if } \phi\left(\sum_{k=1}^{d} w_k x_k + w_0\right) > 0 \\ 0 & \text{otherwise} \end{cases}$$

Inizialize w_0 and w randomly or set $w_0 = 0$ and $w = 0$

Choose constant $\gamma \in (0, 1]$ controlling the learning speed

Feed training pairs (x, y) and for each of them update current threshold and weights $w_0^{(i)}$ and $w^{(i)}$ to $w_0^{(i+1)}$ and $w^{(i+1)}$ as follows:

1) Classify current observation x:

$$o^{(i)} = \begin{cases} 1 & \text{if } \sum_{k=1}^{d} w_k x_k + w_0 > 0 \\ 0 & \text{otherwise} \end{cases}$$

2) Calculate correction:

$$\delta^{(i)} = \begin{cases} 0 & \text{if } o^{(i)} = y \\ 1 & \text{if } o^{(i)} = 0 \text{ but } y = 1 \\ -1 & \text{if } o^{(i)} = 0 \text{ but } y = 0 \end{cases}$$

3) Update threshold and weights:

$$w^{(i+1)} = w^{(i)} + \gamma \delta^{(i)} x$$

$$w_0^{(i+1)} = w_0^{(i)} + \gamma \delta^{(i)}$$

Novikoff's convergence theorem

A fundamental theorem that guarantees the convergence of the perceptron algorithm under certain conditions

- Let $w_0 = 0$ and set $\gamma = 1$
- Let $(\mathcal{X}, \mathcal{Y}) = (x_1, y_1), \dots, (x_i, y_i), \dots$ be an infinite training sequence
- In addition, let (construct)
  $\tilde{\mathcal{X}} = \{x \mid (x, y) \in (\mathcal{X}, \mathcal{Y}), y = 1\} \cup \{-x \mid (x, y) \in (\mathcal{X}, \mathcal{Y}), y = 0\}$
- Let $\tilde{w}$ exist such that for some $\rho_0 > 0$ it holds

$$\min_{\tilde{x} \in \tilde{\mathcal{X}}} \frac{\tilde{w}^T \tilde{x}}{\|\tilde{w}\|} \geq \rho_0$$

  i.e. the classes are linearly separable via the origin with margin ρ_0
- Let $0 < D < \infty$ exist such that it holds

$$\max_{x \in \mathcal{X}} \|x\| < D$$

Theorem:
The perceptron constructs a hyperplane that correctly separates all pairs $(x, y) \in (\mathcal{X}, \mathcal{Y})$ with the number of corrections at most

$$\left\lceil \frac{D^2}{\rho_0^2} \right\rceil$$