

# Machine Learning

## Statistical Learning Framework

---

Souhaib Ben Taieb

March 5, 2020

University of Mons

# Table of contents

Components of supervised learning

Target function (optimal prediction)

# Learning from data

“**Machine learning** is a **scientific discipline** that explores the **construction and study of algorithms** that can **learn from data**.”

- The essence of machine learning
  - A pattern exists
  - We cannot pin it down mathematically
  - We have data on it
- Learning examples
  - Spam Detection
  - Product Recommendation
  - Credit Card Fraud Detection
  - Medical Diagnosis
- The goal of **supervised learning** is to find an underlying relationship between one or several **input** variables and one (or several) **output** variable(s) given **data**.

# Table of contents

Components of supervised learning

Target function (optimal prediction)

# Components of supervised learning

- The **input** variables are typically denoted using the symbol  $X$ . It is also called *predictors*, *independent variables*, *features*, *variables* or just *inputs*. If we observe  $p$  different variables, we write  $X = (X_1, X_2, \dots, X_p)$ . The **inputs** belong to an input space  $\mathcal{X}$ . We typically assume that  $\mathcal{X} \subseteq \mathbb{R}^p$ .
- The **output** variable is typically denoted using the symbol  $Y$ . It is often called the *response* or *dependent variable*. The **output** belongs to an output space  $\mathcal{Y}$ .
  - Regression:  $\mathcal{Y} \subseteq \mathbb{R}$
  - Binary classification:  $\mathcal{Y} \in \{-1, 1\}$  or  $\mathcal{Y} \in \{0, 1\}$
  - Multi-class classification (with  $K$  categories):  
 $\mathcal{Y} \in \{1, 2, \dots, K\}$

# Components of supervised learning

- The **data**, also called *training set*, is a set of  $n$  input-output pairs

$$\mathcal{D} = \{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\} = \{(x_i, y_i)\}_{i=1}^n,$$

where  $x_i = (x_{i1}, \dots, x_{ip})$ . Each pair is also called an *example* or *sample* or *data point*. The space  $\mathcal{X} \times \mathcal{Y}$  is called the *data space*.

- In order to learn (or estimate) the input-output relation, we have to postulate the existence of a model for the data.

# Probabilistic data model

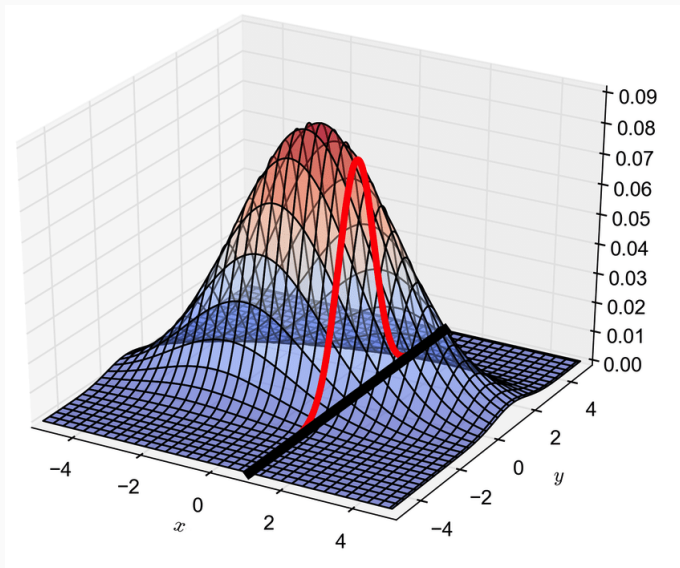
- We assume there exists a fixed unknown data distribution  $p_{X,Y}(x,y)$  according to which the data are identically and independently distributed (i.i.d.).
- The probability distribution  $p_{X,Y}(x,y)$  models different *sources of uncertainty*.
- We assume that it factorizes as

$$p_{X,Y}(x,y) = p_X(x)p_{Y|X}(y|x)$$

where

- the marginal distribution  $p_X(x)$  models uncertainty in the sampling of the input points.
- the conditional distribution  $p_{Y|X}(y|x)$  describes a stochastic (non-deterministic) relation between inputs and output.

## Conditional distribution (continuous case)





## Probabilistic data model - Regression

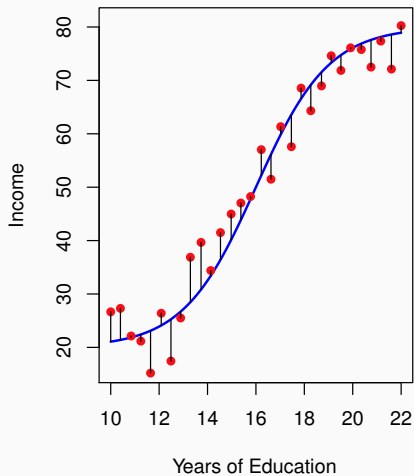
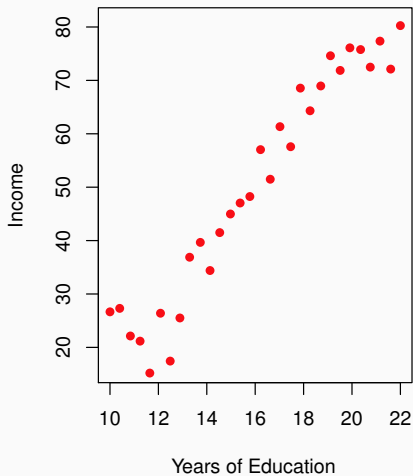
- In regression, the following (additive error) model is often considered:

$$y = f(x) + \varepsilon,$$

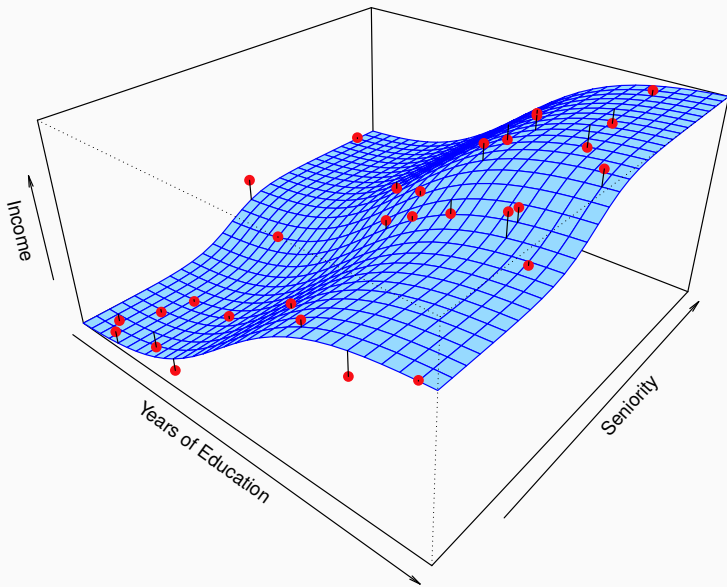
where

- $f$  is a fixed unknown function
- $\varepsilon$  is random noise (independent of  $x$ ) with  $\mathbb{E}[\varepsilon] = 0$  and  $\text{Var}(\varepsilon) = \sigma^2$ , with  $\sigma \in [0, \infty)$ .
- The model implies that the conditional distribution depends on  $x$  only through the conditional mean.
  - $\mathbb{E}[y|x] = f(x)$  and  $\text{Var}[y|x] = \sigma^2$
- For example,  $f$  can be a linear function  $f(x) = \beta_0 + \sum_{j=1}^p \beta_j x_j$  ( $\beta_j \in \mathbb{R}$ ,  $j = 0, 1, \dots, p$ ) and  $\varepsilon \sim \mathcal{N}(0, \sigma^2)$ .
  - $y|x \sim \mathcal{N}(f(x), \sigma^2)$

# Probabilistic data model - Regression



# Probabilistic data model - Regression



# Why estimate $f$ ?

- **Prediction:**

- We want to predict the output for a new input  $x^*$ , i.e.

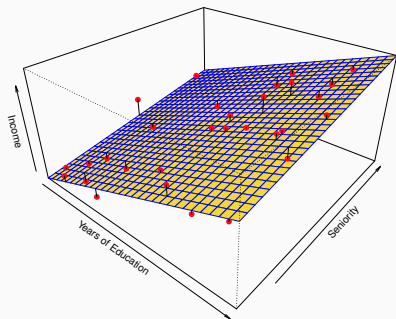
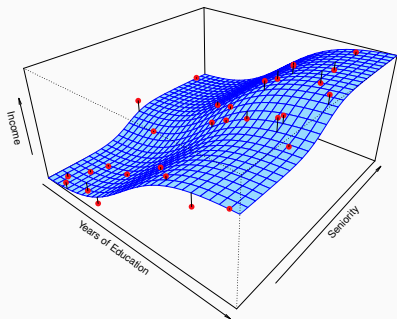
$$\hat{y}^* = \hat{f}(x^*)$$

where  $\hat{f}$  represents our estimate for  $f$ , and  $\hat{y}^*$  represents the resulting prediction for  $y^*$ .

- **Inference (or explanation):**

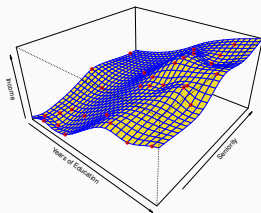
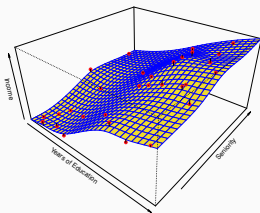
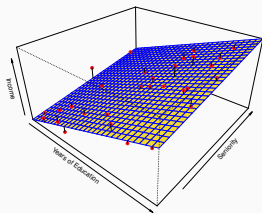
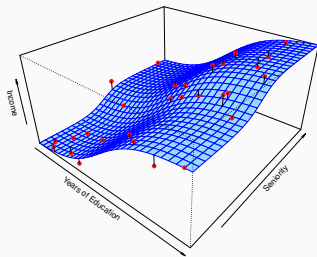
- Which predictors are associated with the response?
- What is the relationship between the response and each predictor?

# How to estimate $f$ ?



$$\hat{f}(\text{education}, \text{seniority}) = \hat{\beta}_0 + \hat{\beta}_1 \times \text{education} + \hat{\beta}_2 \times \text{seniority}$$

# How to estimate $f$ ?



# Two classes of estimation methods

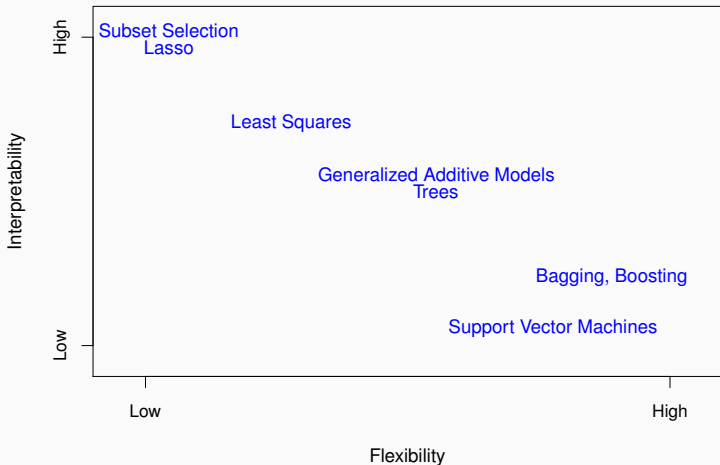
- **Parametric methods**

- Assumption about the form of  $f$ , e.g. linear
- The problem of estimating  $f$  reduces to estimating a set of parameters
- Usually a good starting point for many learning problems
- Poor performance if linearity assumption is wrong
- Example:  $\hat{f}(x) = \hat{\beta}_0 + \hat{\beta}_1 x_1 + \cdots + \hat{\beta}_p x_p$

- **Non-parametric methods**

- No *explicit* assumptions about the form of  $f$
- High flexibility: it can potentially fit a wider range of shapes for  $f$
- A large number of observations is required to estimate  $f$  with good accuracy
- Example:  $\hat{f}(x) = \frac{1}{k} \sum_{x_i \in N_k(x)} y_i$  where  $N_k(x)$  is the set with the  $k$  nearest neighbors of  $x$ .

# Model Interpretability vs flexibility



See “Explainable AI” for a modern view of this tradeoff.

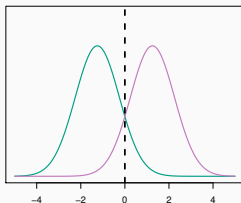


# Probabilistic data model - Classification

- In classification,  $y$  is a discrete random variable ( $p(y|x)$  is a conditional pmf). We cannot use the previous additive error model. The notion of “noise” is different.
- Using Bayes' rule, we can write

$$p(y|x) = \frac{p(x|y)p(y)}{p(x)} \propto p(x|y)p(y) \stackrel{y \text{ uniform}}{\propto} p(x|y)$$

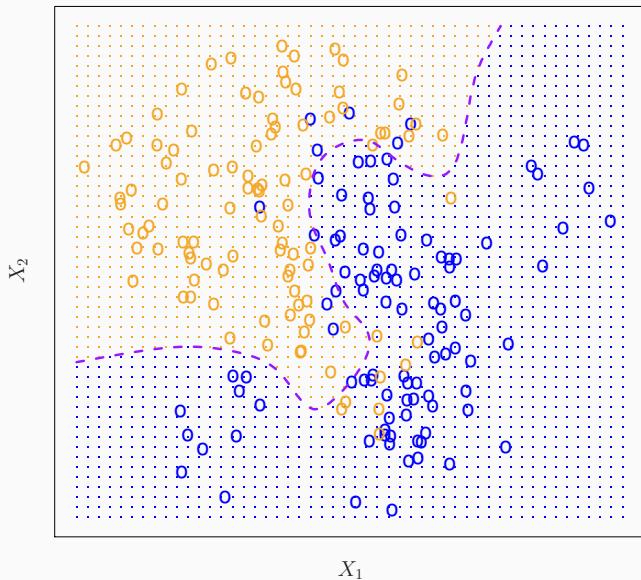
- Let us consider  $K = 2$  and  $p = 1$ .



$p(x|y = -1)$

$p(x|y = +1)$

# Probabilistic data model - Classification



## Components of supervised learning (continued)

- The goal of learning is to estimate the “best” input-output relation, rather than the whole distribution  $p(x, y)$ . In other words, we want the “best” prediction (value) for a given input  $x$ .
- Given the conditional distribution  $p(y|x)$ , which **value** should we use as output prediction for an input  $x$ ?
- To do that, we need to define “best” by specifying a **loss function**

$$L : \mathcal{Y} \times \mathcal{Y} \rightarrow [0, \infty),$$

which is a (pointwise) measure of the error  $L(y, f(x))$  we incur in when predicting  $f(x)$  in place of  $y$ .

- $L(y, f(x)) = (y - f(x))^2$  (square error loss)
- $L(y, f(x)) = |y - f(x)|$  (absolute error loss)
- $L(y, f(x)) = \mathbb{1}\{y \neq f(x)\}$  (zero-one loss)
- ...

## Components of supervised learning (continued)

- Given a loss function, the “best” input-output relation is the **target function**  $f^* : \mathcal{X} \rightarrow \mathcal{Y}$  which minimizes the **expected loss** (or **expected risk**)

$$\mathbb{E}[L(y, f(x))] = \int L(y, f(x)) p(x, y) dx dy,$$

also called the **out-of-sample error** or **generalization error**.

# Table of contents

Components of supervised learning

Target function (optimal prediction)

## Target function (optimal prediction) in regression

In regression (with squared error loss), the target function  $f^*$  minimizes

$$\mathbb{E}_{x,y}[(y - f(x))^2] = \mathbb{E}_x[\mathbb{E}_{y|x}[(y - f(x))^2|x]].$$

It suffices to minimize the error pointwise, i.e.

$$f(x) = \operatorname{argmin}_{c \in \mathbb{R}} \mathbb{E}_{y|x}[(y - c)^2|x = x].$$

The solution is

$$f(x) = \mathbb{E}_{y|x}[y|x],$$

the conditional expectation, also known as the regression function.

Thus the best prediction of  $y$  at any point  $x$  is the conditional mean, when *best is measured by average squared error*.

The target (regression) function has the lowest mean squared errors (MSE) given by  $\sigma^2$ , the variance of  $y|x$ .

## Proof

We want to minimize

$$r(c; x) = \mathbb{E}_{y|x}[(y - c)^2 | x = x].$$

The necessary condition for optimality is given by

$$\begin{aligned} \frac{\partial r(c; x)}{\partial c} = 0 &\iff \frac{\partial \mathbb{E}_{y|x}[(y - c)^2 | x = x]}{\partial c} = 0 \\ &\iff \mathbb{E}_{y|x} \left[ \frac{\partial (y - c)^2}{\partial c} | x = x \right] = 0 \\ &\iff \mathbb{E}_{y|x} [-2(y - c) | x = x] = 0 \\ &\iff c = \mathbb{E}_{y|x} [y | x = x] \end{aligned}$$

The sufficient condition for optimality (a minimum) is given by

$$\frac{\partial^2 r(c; x)}{\partial c^2} > 0 \iff 2 > 0.$$

## Target function (optimal prediction) in classification

Let us consider multi-class classification with  $K$  categories where  $y \in \mathcal{C} = \{C_1, \dots, C_K\}$ .

With the zero-one loss, the target function  $f^*$  minimizes

$$\mathbb{E}_{x,y}[\mathbb{1}\{y \neq f(x)\}] = p(y \neq f(x)) = \mathbb{E}_x[\mathbb{E}_{y|x}[\mathbb{1}\{y \neq f(x)\}|x]].$$

It suffices to minimize the error pointwise, i.e.

$$f(x) = \operatorname{argmin}_{c \in \mathcal{C}} \mathbb{E}_{y|x}[\mathbb{1}\{y \neq c\}|x = x].$$



We have that

$$\begin{aligned} & \mathbb{E}_{y|x}[\mathbb{1}\{y \neq c\}|x = x] \\ &= \sum_{k=1}^K \mathbb{1}\{C_k \neq c\} p(y = C_k|x = x) \\ &= \sum_{k:C_k \neq c} 1 \times p(y = C_k|x = x) + 0 \times p(y = c|x = x) \\ &= \sum_{k:C_k \neq c} p(y = C_k|x = x) \\ &= \sum_{k:C_k \neq c} p(y = C_k|x = x) + p(y = c|x = x) - p(y = c|x = x) \\ &= \sum_{k=1}^K p(y = C_k|x = x) - p(y = c|x = x) \\ &= 1 - p(y = c|x = x). \end{aligned}$$

## Target function (optimal prediction) in classification

This implies that

$$\begin{aligned} f(x) &= \operatorname{argmin}_{c \in \mathcal{C}} \mathbb{E}_{y|x}[\mathbb{1}\{y \neq f(x)\} | x = x] \\ &= \operatorname{argmin}_{c \in \mathcal{C}} 1 - p(y = c | x = x) \\ &= \operatorname{argmax}_{c \in \mathcal{C}} p(y = c | x = x). \end{aligned}$$

Equivalently, we have that

$$f(x) = C_k \text{ if } p(y = C_k | x = x) = \max_{c \in \mathcal{C}} p(y = c | x = x).$$

This target function (classifier) is called the Bayes classifier, which has the following error rate at  $x$ :

$$1 - \max_{k=1, \dots, K} p(y = C_k | x = x),$$

also called the Bayes error rate, which gives the lowest possible error rate that could be achieved if we knew the true probability distribution of the data.

## Supervised learning in practice

- In practice, the target function cannot be computed exactly since we do not know  $p(x, y)$ .
- Instead, we observe a dataset  $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^n$  where  $(x_i, y_i) \stackrel{\text{i.i.d.}}{\sim} p(x, y)$ , i.e.  $x_i \stackrel{\text{i.i.d.}}{\sim} p(x)$  and  $y_i \stackrel{\text{i.i.d.}}{\sim} p(y|x = x_i)$ .
- Using the **dataset**  $\mathcal{D}$ , the goal of the **learning algorithm** is to pick the “best” hypothesis or function, denoted  $\hat{f}_{\mathcal{D}}$  or  $\hat{f}$ , from a **hypothesis set**  $\mathcal{H}$ , where “best” is defined by the **loss function**  $L$ .
- The **hypothesis set**  $\mathcal{H}$  contains all the hypotheses (functions) we consider. It is often implicitly defined.
- In other words, the learning algorithm will **estimate** the target function.