# Statistical Learning

Jordi Villà i Freixa

Universitat de Vic - Universitat Central de Catalunya
Study Abroad

*jordi.villa@uvic.cat*

course 2023-2024

# Índex

# Preliminary note

The material in these slides is strongly based on [1]. When other materials are used, they are cited accordingly.
Mathematical notation follows as good as it can a good practices proposal from the Beijing Academy of Artificial Intelligence.

# What to expect?

In this session we will discuss:

- Modelling data
- Models with independent and identically distributed (iid) data
- The modelling dilemma
- Linear models
- Multivariate normal models
-

# How is data analyzed and used?

Statistical learning interpret the model and quantify the uncertainity of the data.

Machine learning (or *data mining* making predictions using large scale data.

The goals of modelling data are:

- to predict data, based on existing one;
- to discover unusual or interesting patterns in data.

# Example of modelling data I

Imagine an unsupervised learning problem, with data represented by a vector $\boldsymbol{x} = [x_1, \ldots, x_p]^\intercal$, a very general model is to assume that $\boldsymbol{x}$ is the outcome of a random vector $\boldsymbol{X} = [X_1, \ldots, X_p]^\intercal$ with some unknown pdf $f$. The model can be refined by assuming a specific form of $f$.

# Example of modelling data II

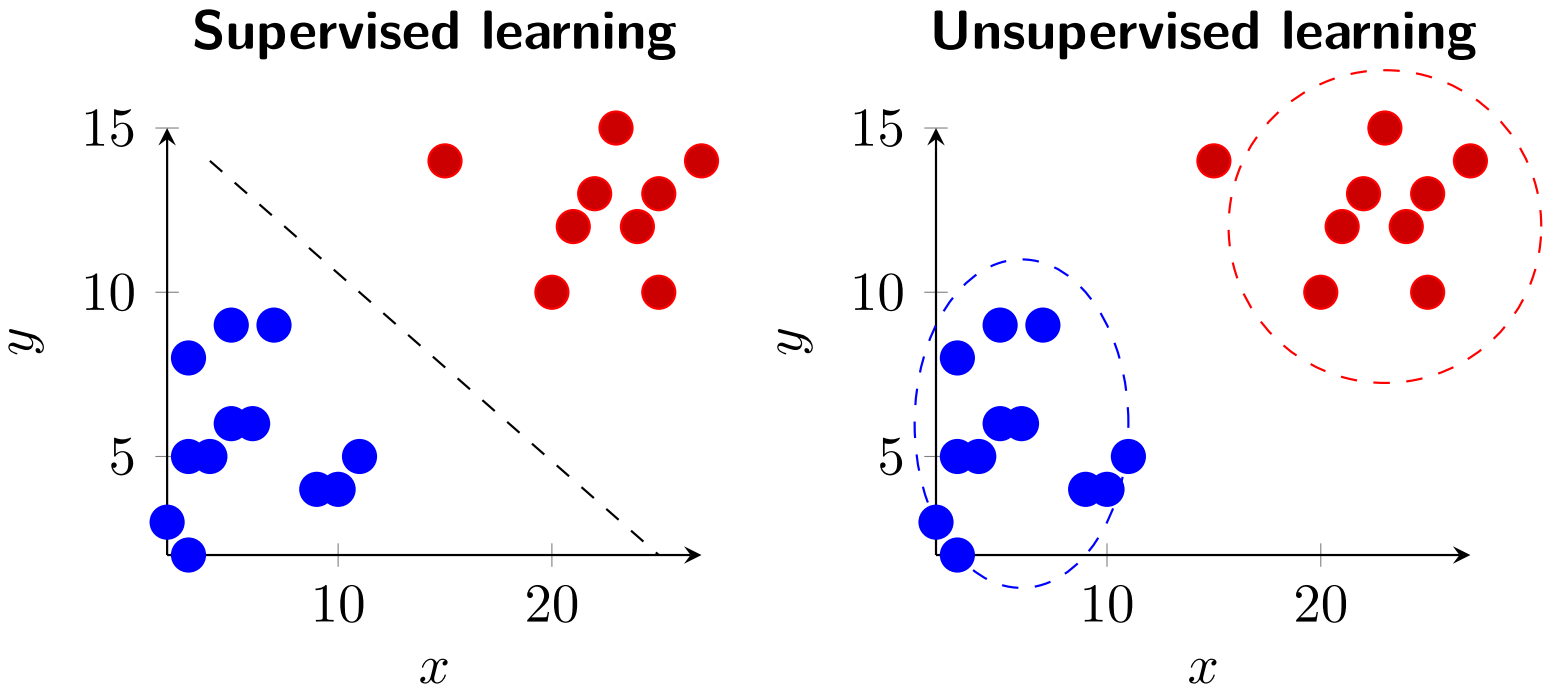


Figure 1: Supervised vs unsupervised ML

# Tools to model data

**Function approximation** Model data with approximate and simple functions or maps.

**Optimization** Given a set of feasible mathematical models to the data, we may need to find the optimal one by fitting or callibrating a function to observed data.

**Probability and Statistics** Probability theory and statistical inference provides ways to quantify the uncertainity inherent in making predictions based on observed data.

# *iid* data

If we are given a sequence of data vectors $\boldsymbol{x_1}, \ldots, \boldsymbol{x_n}$ one of the simplest possible models is to assume that the corresponding random vectors $\boldsymbol{X_1}, \ldots, \boldsymbol{X_n}$ are independent and identically distributed (iid). We express this as:

$$\boldsymbol{X_1}, \ldots, \boldsymbol{X_n} \overset{iid}{\sim} f$$

meaning that the random vectors form an iid sample from a pdf $f$ or sampling distribution *Dist*.
This is the same as saying that knowing about one variable does not provide information about another variable.

# Independent data models

In independent data models, the joint density of the random vectors $\boldsymbol{X_1}, \ldots, \boldsymbol{X_n}$ is the *product* of the marginal ones:

$$f_{\boldsymbol{X_1},\ldots,\boldsymbol{X_n}}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n) = f(\boldsymbol{x}_1) \cdots f(\boldsymbol{x}_n)$$

The function $g(\boldsymbol{x})$, the "model" for $f(\boldsymbol{x})$ is usually specified up to a small number of parameters, corresponding to :

- $\mathcal{N}(\mu, \sigma^2)$
- $\mathrm{Bin}(n, p)$
- $\mathrm{Exp}(\lambda)$

The parameters are typically obtained from the data.

# Modeling dilemma

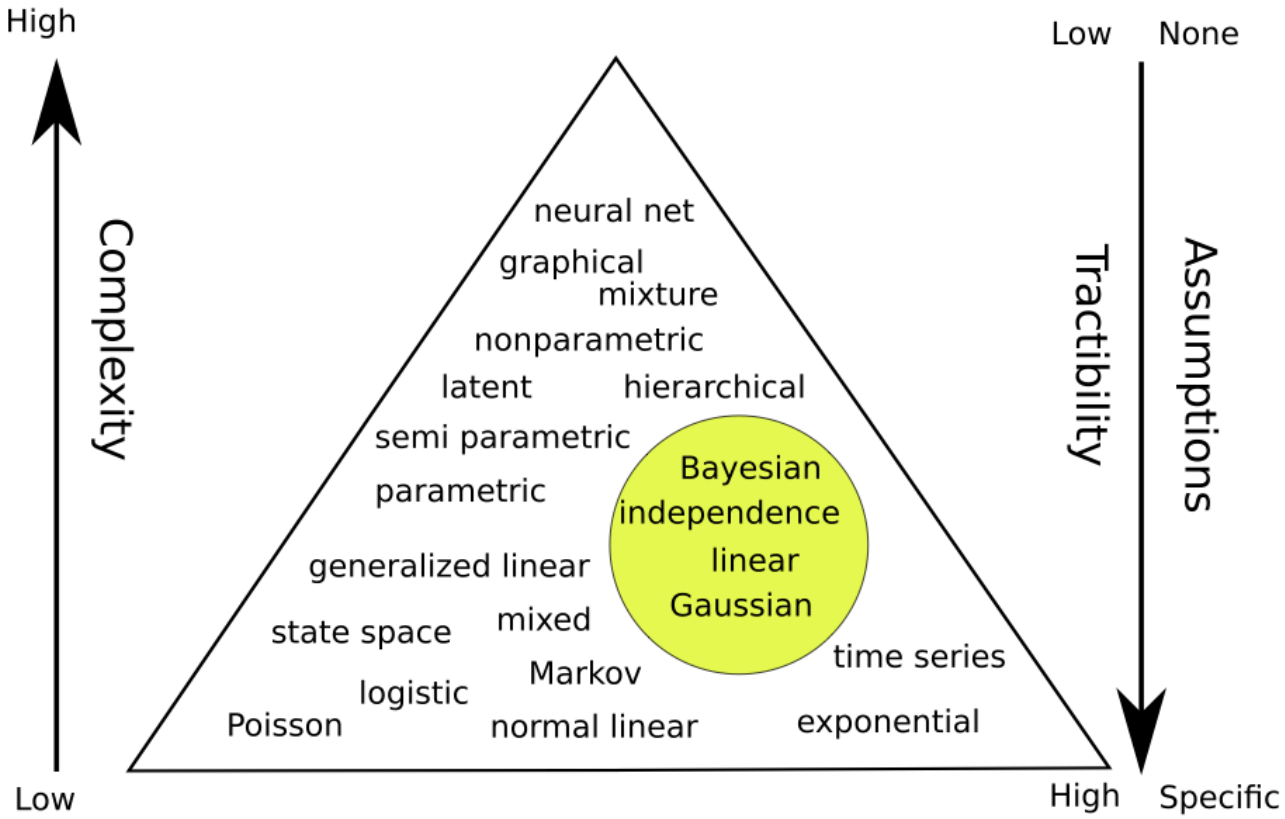


Figure 2: Complex models (very few of them) generally applicable but difficult to analyze. Simple models (a lot of options) very tractable but they do not describe well the data[1].

# Tradeoff

There exists a tradeoff between model tractability and applicability, as seen in Figure 2. Coming back to the example in page 6, the *training set* $\tau = \{\boldsymbol{x_1}, \ldots, \boldsymbol{x_n}\}$ is viewed as the outcome of $n$ iid random variables $\boldsymbol{X_1}, \ldots, \boldsymbol{X_n}$ for some unknown pdf.
**Goal:** to learn or estimate $f$ from the finite training set.

# Tradeoff vs risk

Imagine the **unsupervised learning** framewrok shown before. We can specify a class (a collection) of pdfs that we will call $\mathcal{G}_p$:

- We seek within $\mathcal{G}$ the best approximation to the true model pdf $f(\boldsymbol{x}$, and we will call it $g(\boldsymbol{x}|\boldsymbol{\Theta})$.
- Such best approximation will minimize some calculated risk.

$$\text{Loss}(f(\boldsymbol{x}), g(\boldsymbol{x}|\boldsymbol{\Theta}))) = \ln f(\boldsymbol{x}) - \ln g(\boldsymbol{x}|\boldsymbol{\Theta}))$$

with expected value, this is, the **risk** as

$$\ell(g) = \mathbb{E} \ln \frac{f(\boldsymbol{X})}{g(\boldsymbol{X}|\boldsymbol{\Theta})} = \int f(\boldsymbol{x}) \frac{f(\boldsymbol{x})}{g(\boldsymbol{x}|\boldsymbol{\Theta})} \, \mathrm{d}\boldsymbol{x}$$

# Some basic notation I

Given an input or *feature* vector $\boldsymbol{x}$, ML aims at predicting an ouput or *response* variable vector $\boldsymbol{y}$. In particular, we search for a mathematical *prediction function* $g$ such that we can *guess* an approximation to $\boldsymbol{y}$, $\hat{\boldsymbol{y}}$:

$$g\colon \mathcal{X} \to \mathcal{Y}$$
$$\boldsymbol{x} \mapsto \hat{\boldsymbol{y}} = g(\boldsymbol{x})$$

## Definition

Dataset $S = \{\boldsymbol{z}_i\}_{i=1}^n = \{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^n$ is sampled from a distribution $\mathcal{D}$ over a domain $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$.
$\mathcal{X}$ is the instance domain (a set), $\mathcal{Y}$ is the label domain (a set), and $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$ is the example domain (a set).

# Some basic notation II

Usually, $\mathcal{X}$ is a subset of $\mathbb{R}^d$ and $\mathcal{Y}$ is a subset of $\mathbb{R}^{d_o}$, where $d$ is the input dimension, $d_o$ is the output dimension.
$n = \#S$ is the number of samples. Without specification, $S$ and $n$ are for the training set.

- In *regression* problems, $\boldsymbol{y}$ is a vector of real values.
- In *classification* problems, $\boldsymbol{y}$ values lie within a finite set of $c$ categories: $y \in \{0, 1, \ldots, c-1\}$.

## Definition

A hypothesis space is denoted by $\mathcal{H}$. A hypothesis function is denoted by $f_{\boldsymbol{\theta}}(\boldsymbol{x}) \in \mathcal{H}$ or $f(\boldsymbol{x}; \boldsymbol{\theta}) \in \mathcal{H}$ with $f_{\boldsymbol{\theta}} : \mathcal{X} \to \mathcal{Y}$.

$\boldsymbol{\theta}$ denotes the set of parameters of $f_{\boldsymbol{\theta}}$.
If there exists a target function, it is denoted by $f^*$or $f : \mathcal{X} \to \mathcal{Y}$ satisfying $\boldsymbol{y}_i = f^*(\boldsymbol{x}_i)$ for $i = 1, \ldots, n$.

FACULTAT DE CIÈNCIES, TECNOLOGIA I ENGINYERIES
UVIC | UVIC-UCC

# Some basic notation III

A loss function, denoted by $\ell : \mathcal{H} \times \mathcal{Z} \to \mathbb{R}_+ := [0, +\infty)$, measures the difference (or error) between a predicted label and a true label, e.g., $L^2$ loss:

$$\ell(f_{\boldsymbol{\theta}}, \boldsymbol{z}) = \frac{1}{2}(f_{\boldsymbol{\theta}}(\boldsymbol{x}) - \boldsymbol{y})^2,$$

where $\boldsymbol{z} = (\boldsymbol{x}, \boldsymbol{y})$. $\ell(f_{\boldsymbol{\theta}}, \boldsymbol{z})$ can also be written as

$$\ell(f_{\boldsymbol{\theta}}(\boldsymbol{x}), \boldsymbol{y})$$

for convenience.
(In the case of a classification, $\ell(f_{\boldsymbol{\theta}}, \boldsymbol{y}) = \mathbb{1}\{y \neq \hat{\boldsymbol{y}}\}$)
We will see other useful loss functions ({em cross entropy} or *hinge* loss functions) later in this course.
It is unlikely that a mathematical function $g \equiv f_{\boldsymbol{\theta}} : \mathcal{X} \to \mathcal{Y}$ would be able to make accurate predictions of all possible pairs $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$.

FACULTAT DE CIÈNCIES, TECNOLOGIA I ENGINYERIES
UVIC | UVIC-UCC

# Some basic notation IV

So, we use a probabilistic approach here to mpirical risk or training loss for a set $S = \{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^n$ is denoted by $L_S(\boldsymbol{\theta})$ or $L_n(\boldsymbol{\theta})$ or $R_n(\boldsymbol{\theta})$ or $R_S(\boldsymbol{\theta})$,

$$L_S(\boldsymbol{\theta}) = \frac{1}{n} \sum_{i=1}^{n} \ell(f_{\boldsymbol{\theta}}(\boldsymbol{x}_i), \boldsymbol{y}_i). \tag{1}$$

The population risk or expected loss is denoted by $L_{\mathcal{D}}(\boldsymbol{\theta})$ or $R_{\mathcal{D}}(\boldsymbol{\theta})$

$$L_{\mathcal{D}}(\boldsymbol{\theta}) = \mathbb{E}_{\mathcal{D}} \ell(f_{\boldsymbol{\theta}}(\boldsymbol{x}), \boldsymbol{y}), \tag{2}$$

where $\boldsymbol{z} = (\boldsymbol{x}, \boldsymbol{y})$ follows the distribution $\mathcal{D}$.
(In the case of a classification, we denote $L_{\mathcal{D}}(g) \equiv L_{\mathcal{D}}(\boldsymbol{\theta}) = \mathbb{P}_{\mathcal{D}}[f_{\boldsymbol{\theta}}(\boldsymbol{x}) \neq \boldsymbol{y}]$ and we say that $g$ is a classifier.)
Because we are interested in minimizing the risk in our prediction, we are looking for the best possible $g* := \operatorname{argmin}_g \mathbb{E}_{\mathcal{D}} \ell(f_{\boldsymbol{\theta}}(\boldsymbol{x}), \boldsymbol{y})$

FACULTAT DE CIÈNCIES, TECNOLOGIA I ENGINYERIES
UVIC | UVIC-UCC

# Some basic notation V

(In classification, we look for $g*(\boldsymbol{x}) = \underset{y \in \{0,1,\ldots,c-1\}}{\operatorname{argmax}} \mathbb{P}[Y = y \; X = x]$.)

**Theorem**

*For the squared-error loss $\ell(y, \hat{y}) = (y - \hat{y})^2$, the optimal prediction function $g*$ is equal to the conditional expectation of $Y$ given $\boldsymbol{X} = \boldsymbol{x}$.*

which leads to write the random response $Y$ as:

$$Y = g*(\boldsymbol{x}) + \varepsilon(\boldsymbol{x})$$

Note that such random deviation satisfies $\mathbb{E}\varepsilon(\boldsymbol{x}) = 0$


Dirk P. Kroese, Zdravko Botev, Thomas Taimre, and Radislav: Vaisman.
*Data Science and Machine Learning: Mathematical and Statistical Methods*.
Machine Learning & Pattern Recognition. Chapman & Hall/CRC, 2020.