

Statistical Learning

Jordi Villà i Freixa

Universitat de Vic - Universitat Central de Catalunya
Study Abroad

jordi.villa@uvic.cat

course 2023-2024

- 1 Introduction and scope
- 2 Introduction
- 3 Statistical modelling
- 4 Modelling data
- 5 Detailed notation
- 6 Bibliography

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](<https://ctan.math.utah.edu/ctan/tex-archive/macros/latex/contrib/mlmath/mlmath.pdf>) 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 uncertainty 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 $\mathbf{x} = [x_1, \dots, x_p]^\top$, a very general model is to assume that \mathbf{x} is the outcome of a random vector $\mathbf{X} = [X_1, \dots, X_p]^\top$ with some unknown pdf f . The model can be refined by assuming a specific form of f .

Example of modelling data II

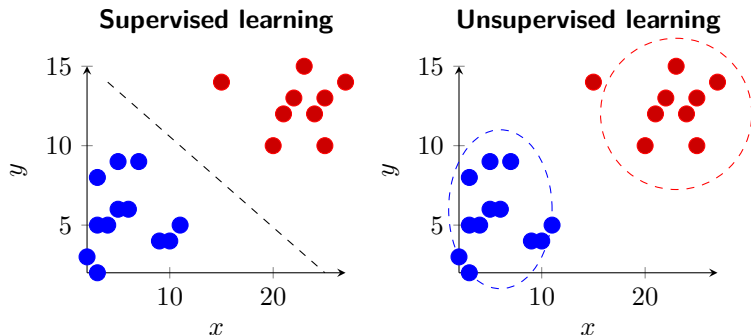


Figure 1: Supervised vs unsupervised ML

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 uncertainty inherent in making predictions based on observed data.

iid data

If we are given a sequence of data vectors $\mathbf{x}_1, \dots, \mathbf{x}_n$ one of the simplest possible models is to assume that the corresponding random vectors $\mathbf{X}_1, \dots, \mathbf{X}_n$ are independent and identically distributed (iid). We express this as:

$$\mathbf{X}_1, \dots, \mathbf{X}_n \stackrel{iid}{\sim} f$$

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 $\mathbf{X}_1, \dots, \mathbf{X}_n$ is the *product* of the marginal ones:

$$f_{\mathbf{X}_1, \dots, \mathbf{X}_n}(\mathbf{x}_1, \dots, \mathbf{x}_n) = f(\mathbf{x}_1) \cdots f(\mathbf{x}_n)$$

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

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

The parameters are typically obtained from the data.

Modeling dilemma

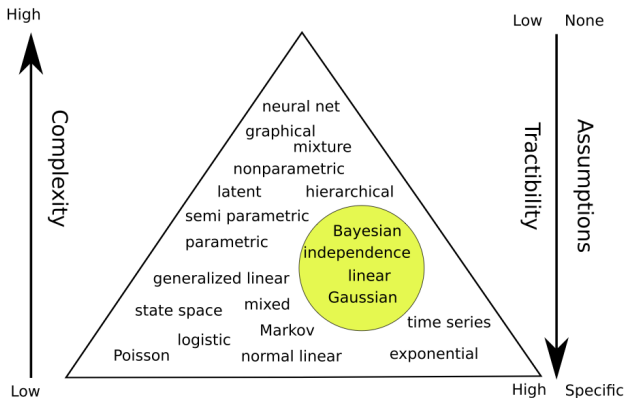


Figure 2: Complex models (very few of them) generally applicable but difficult to analyze. Simple models (a lot of options) very tractable but they do not describe well the data[1].

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* $\tau = \{\mathbf{x}_1, \dots, \mathbf{x}_n\}$ is viewed as the outcome of n iid random variables $\mathbf{X}_1, \dots, \mathbf{X}_n$ for some unknown pdf.

Goal: to learn or estimate f from the finite training set.

Tradeoff vs risk

Imagine the unsupervised learning framework 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 pdf f , and we will call it $g(\mathbf{x})$.
- Such best approximation will minimize some calculated risk.

Some basic notation I

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

$$g: \mathcal{X} \rightarrow \mathcal{Y}$$

$$\mathbf{x} \mapsto \hat{\mathbf{y}} = g(\mathbf{x})$$

Definition

Dataset $S = \{\mathbf{z}_i\}_{i=1}^n = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^n$ 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, \mathbf{y} is a vector of real values.
- In *classification* problems, \mathbf{y} values lie within a finite set of c categories: $y \in \{0, 1, \dots, c - 1\}$.

Definition

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

θ denotes the set of parameters of f_{θ} .

If there exists a target function, it is denoted by f^* or $f : \mathcal{X} \rightarrow \mathcal{Y}$ satisfying $\mathbf{y}_i = f^*(\mathbf{x}_i)$ for $i = 1, \dots, n$.

Some basic notation III

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

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

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

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

for convenience.

(In the case of a classification, $\ell(f_\theta, \mathbf{y}) = \mathbb{1}\{y \neq \hat{\mathbf{y}}\}$)

We will see other useful loss functions (cross entropy or *hinge* loss functions) later in this course.

It is unlikely that a mathematical function $g \equiv f_\theta : \mathcal{X} \rightarrow \mathcal{Y}$ would be able to make accurate predictions of all possible pairs $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$.

Some basic notation IV

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

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

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

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

where $\mathbf{z} = (\mathbf{x}, \mathbf{y})$ follows the distribution \mathcal{D} .

(In the case of a classification, we denote $L_{\mathcal{D}}(g) \equiv L_{\mathcal{D}}(\boldsymbol{\theta}) = \mathbb{P}_{\mathcal{D}}[f_{\boldsymbol{\theta}}(\mathbf{x}) \neq \mathbf{y}]$ 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 $g^* := \operatorname{argmin}_g \mathbb{E}_{\mathcal{D}} \ell(f_{\boldsymbol{\theta}}(\mathbf{x}), \mathbf{y})$

Some basic notation V

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

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 $\mathbf{X} = \mathbf{x}$.*

which leads to write the random response Y as:

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

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



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.