BGSM/CRM
AL&DNN

Informal outline
and some background topics

S. Xambó

UPC & IMTech

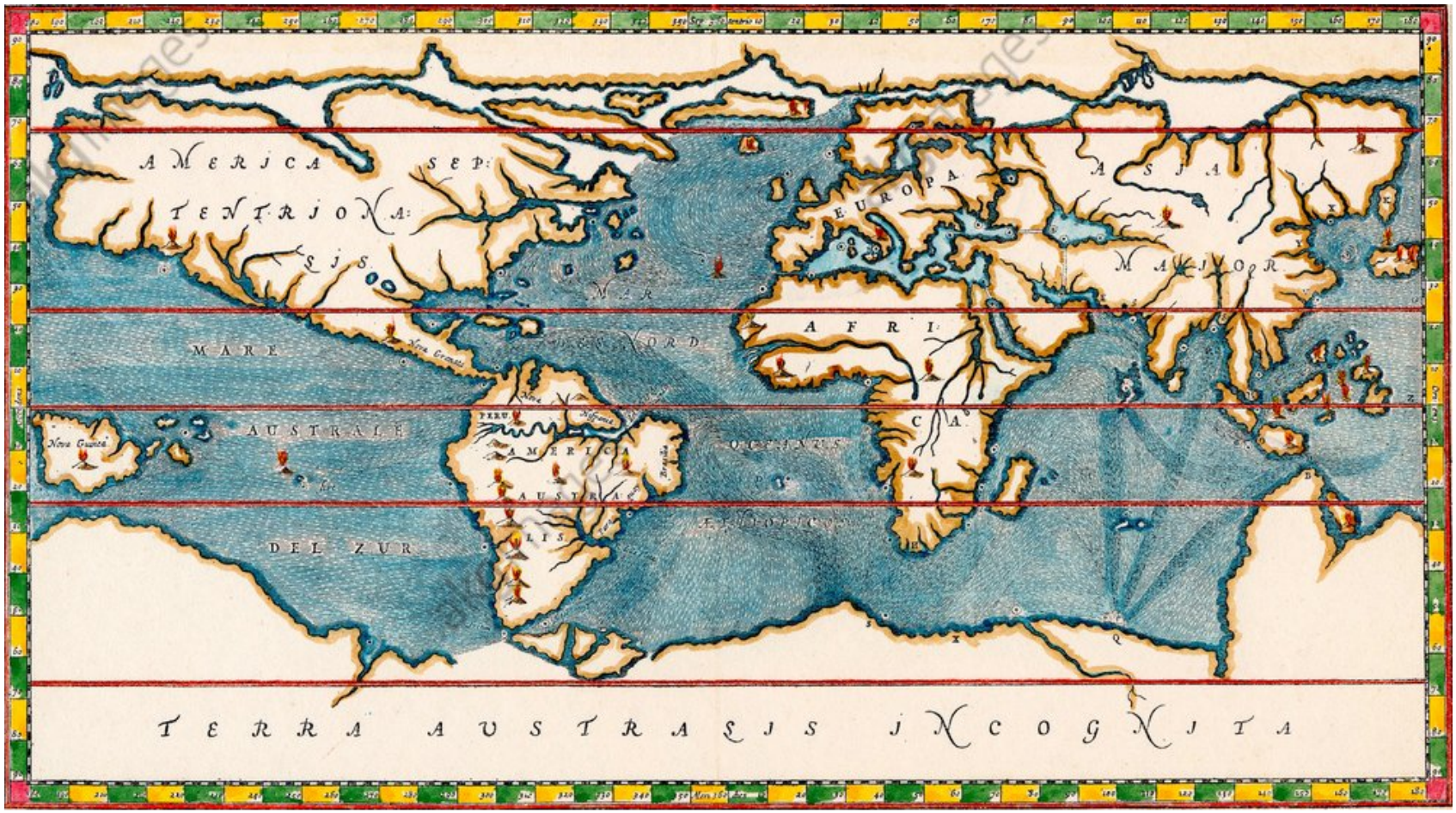
5/10/2021

Session	**Topics**	
1: Tue 10-05	General introduction	SX
2: Thu 10-07*	The curse of dimensionality. NNs and approximation properties	JB
3: Tue 10-13	Reproducing kernel Hilbert spaces	SX
4: Thu 10-14	Gradient descent and stochastic approximation	SX
5: Tue 10-19	Training dynamics: lazy regine and Neural Tangent Kernel	SX
6: Thu 10-21*	Training dynamics: active regime and mean-field description	JB
7: Tue 11-02	Group theory and differential geometry basics. Noether's theorem.	SX
8: Thu 11-04*	Beyond Barron spaces: geometric stability	JB
9: Tue 11-09	Harmonic Analysis: Fourier, Wavelets, Graph spectral transforms.	SX
10: Thu 11-11*	The Scattering Transform	JB
11: Tue 11-16*	Beyond Euclidean Domains: the 5G	JB
12: Thu 11-18*	Open Problems and closing remarks	JB

Informal outline

The most promising words ever written on the maps of human knowledge are **terra incognita**

Daniel J. Boorstin, *The discoverers*



Athanasius Kircher, *Mundus Subterraneus*, Amsterdam 1668

John **McCarthy** first coined the term artificial intelligence in **1956** when he invited a group of researchers from a variety of disciplines to discuss what would ultimately become the field of **AI**. At that time, the researchers came together to clarify and develop the concepts around "thinking machines" which up to this point had been quite divergent.

The **proposal** for the conference said:

> *The study is to proceed on the basis of the* **conjecture** *that every aspect of* **learning** *or any other feature of intelligence* **can in principle be so precisely described that a machine can be made to simulate it**.

Reading: Jeff Hawkins, *A thousand brains: a new theory of intelligence*. Basic books, 2021. See [1].

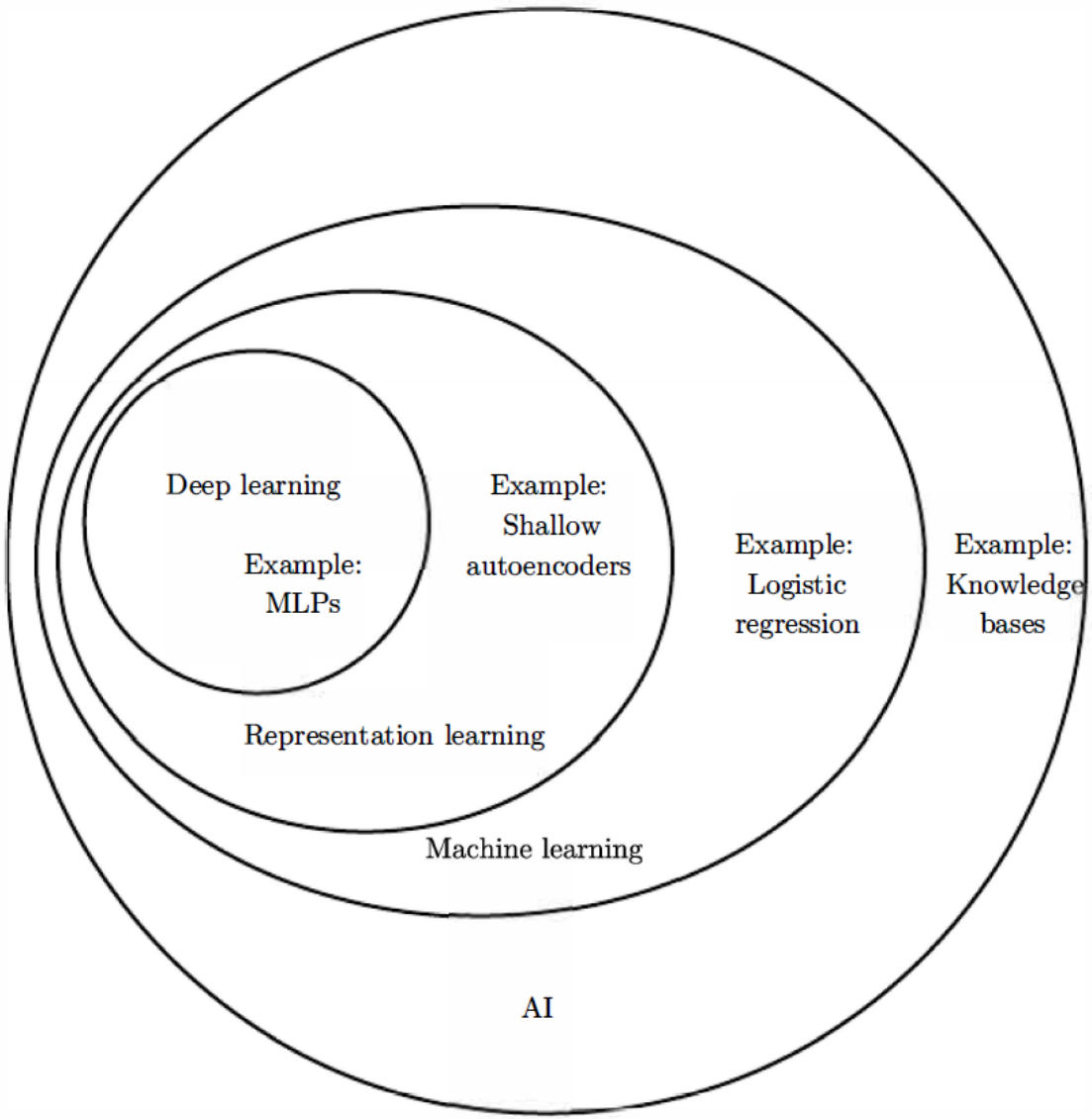


Figure 2.1: Inclusion relations: DL $\subset$ RL $\subset$ ML (AL) $\subset$ AI (cf. [2]).

	AI taxonomy	
	AI domain	**AI subdomain**
Core	**Reasoning**	Knowledge representation
		Automated reasoning
		Common sense reasoning
	Planning	Planning and Scheduling
		Searching
		Optimisation
	Learning	Machine learning
	Communication	Natural language processing
	Perception	Computer vision
		Audio processing
Transversal	**Integration and Interaction**	Multi-agent systems
		Robotics and Automation
		Connected and Automated vehicles
	Services	AI Services
	Ethics and Philosophy	AI Ethics
		Philosophy of AI

Figure 2.2: Extracted from [3, page 11]

Hypothesis space[3]

$$\mathcal{H} = \{h\}$$

Data, expert tables[2]

$$\mathcal{D}$$
$$\{(x^1, y^1), \ldots, (x^m, y^m)\}$$
$$x^j \sim P, \quad y^j = f^*(y^j)$$

World – Observer[1]

Learner/LA[5]

Error measure[4]

$\Longrightarrow$

Prediction rule[6]

$$y = h(x)$$

[1] *Urbi et Orbi* engraving (Flammarion). **Tycho Brahe** (observer model): He experienced the solar eclipse of 21 August 1560 [he was 15], and was greatly impressed by the fact that it *had been predicted, although the prediction based on current observational data was a day off.* He realized that *more accurate observations* would be **the key to making more exact predictions**. [2] **Ephemeris**: Tables of planet and comet positions over time.
[3] *Inductive bias.* Greeks: circles around Earth. [4] *Loss*, *risk*, *regret.* How close are predictions to observations?
[5] Learner model (Kepler): Ellipses with a focus at the Sun. Today: Learning algorithm. [6] Hopefully, $h \approx f^*$.

[4, Abstract] (but see also [5]):

"Neural networks have a reputation for being better at solving statistical or approximate problems than at performing calculations or working with symbolic data.

In this paper, we show that they can be surprisingly good at more elaborated tasks in mathematics, such as *symbolic integration* and solving differential equations.

We propose a syntax for representing mathematical problems, and methods for generating large datasets that can be used to train sequence-to-sequence models.

We achieve results that outperform commercial Computer Algebra Systems such as Matlab or Mathematica."

One dataset for learning symbolic integration is formed by a large number of pairs (f', f), where f is an expression and f' its derivative.

Aimed at finding *hidden structure in data*.

***k*-Means.** This algorithm groups unlabeled data $\mathcal{D}$ in k classes:

(1) Select k vectors $z^1, \cdots, z^k \in \mathcal{D}$ at random.

(2) Assign each $x^j \in \mathcal{D}$ to the first z^i nearest to x^j (initial groups).

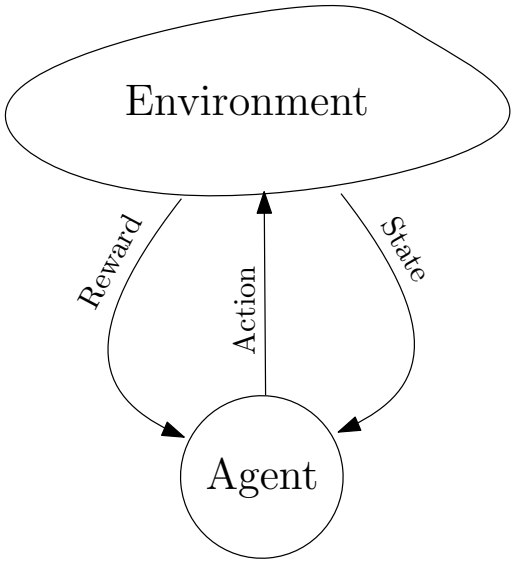
(3) Update each z^i to the centroid (or mean) of the z^i group.

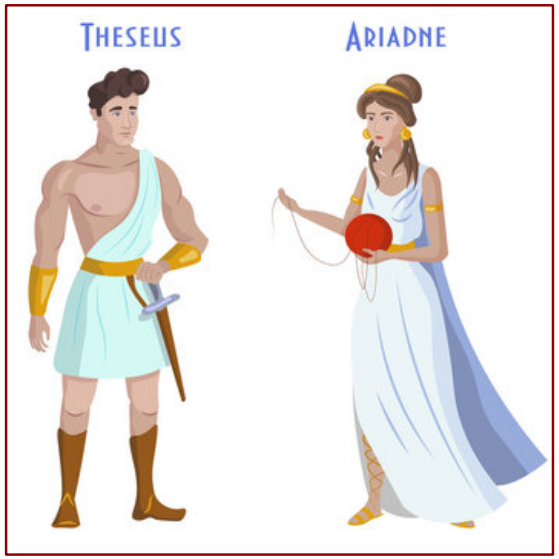
(4) Iterate (2) and (3) until the z^i are stable (up to a tolerance).

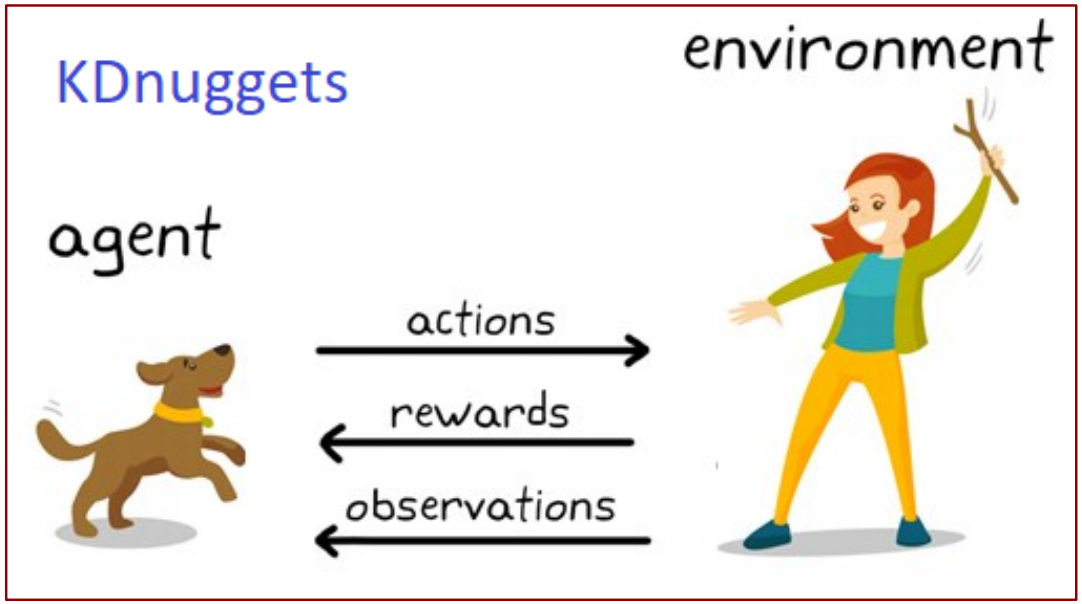
The associated *cluster predictor* assigns x to the first nearest z^i.

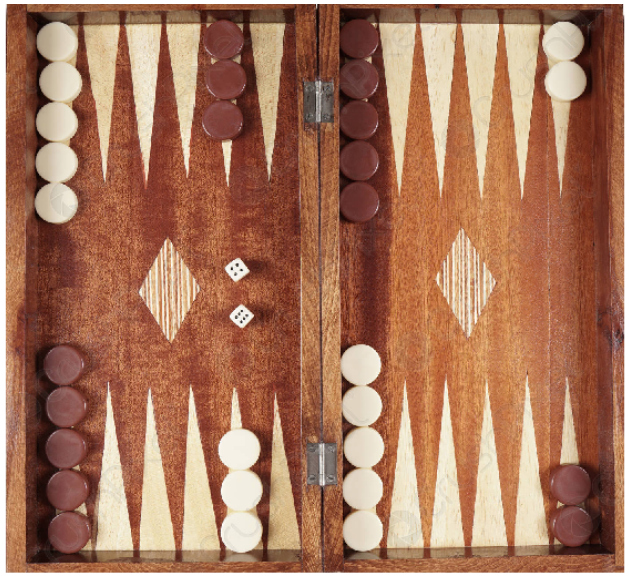
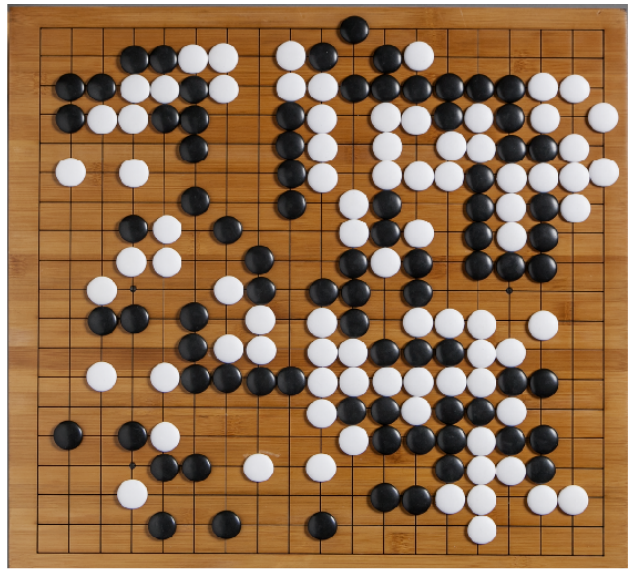
***k*-NN** (nearest neighbors). Let $\mathcal{D} = \{(x^1, y^1), \cdots, (x^m, y^m)\}$ be a labeled set and k a positive integer. The label predictor of the k-NN algorithm assigns a vector x to the mode of $y^{j_1}, \cdots, y^{j_k}$, where $x^{j_1}, \cdots, x^{j_k}$ are the nearest neighbors of x from among $x^1, \cdots, x^m$.

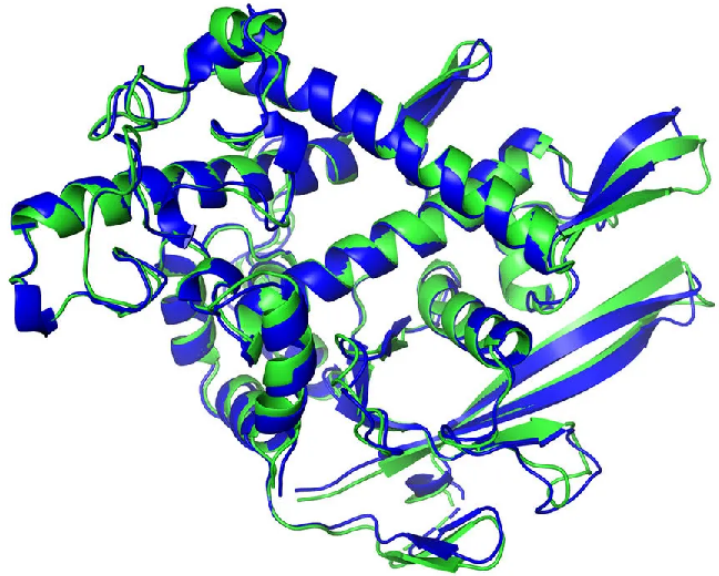
Algorithm learns to *react* to an *environment*









Chess, Backgammon, Go, Console games, Protein folding, ...

"We demonstrate how by using a *reinforcement learning algorithm*, the deep cross-entropy method, *one can find explicit constructions and counterexamples to several open conjectures in extremal combinatorics and graph theory*.

Amongst the conjectures we refute are a question of Brualdi and Cao about maximizing permanents of pattern avoiding matrices, and several problems related to the adjacency and distance eigenvalues of graphs." [6, Abstract].

An interesting feature is that in some cases the learning algorithm does not produce directly a counterexample but graphs which are close to refuting the conjecture; these graphs have a special structure and give a very clear indication about where to search for counterexamples. See IMTech NL01, p. 20 (`https://imtech.upc.edu/en/communication/nesletter/nl01_web.pdf`).

Introductory problems

$\mathcal{X} \subseteq \mathbf{R}^n$, $\mathcal{Y} = \{-1, 1\}$.

$\mathcal{D} = \{(x^1, y^1), \ldots, (x^m, y^m)\}$ (dataset, $x^j \in \mathcal{X}$, $y^j \in \mathcal{Y}$).

$\mathcal{D}$ is *linearly separable* if there is a hyperplane $h(x) = w \cdot x + b$ ($w \in \mathbf{R}^n$, $b \in \mathbb{R}$)) such that $y^j h(x^j) > 0$ ($j \in [m]$).

In general, there are (if any) infinitely many separating hyperplanes.

Problem. Find the separating hyperplane such that the points on either side are as far as possible from the hyperplane.

This idea leads to the notion of *margin* and the appearance of *support vectors*.

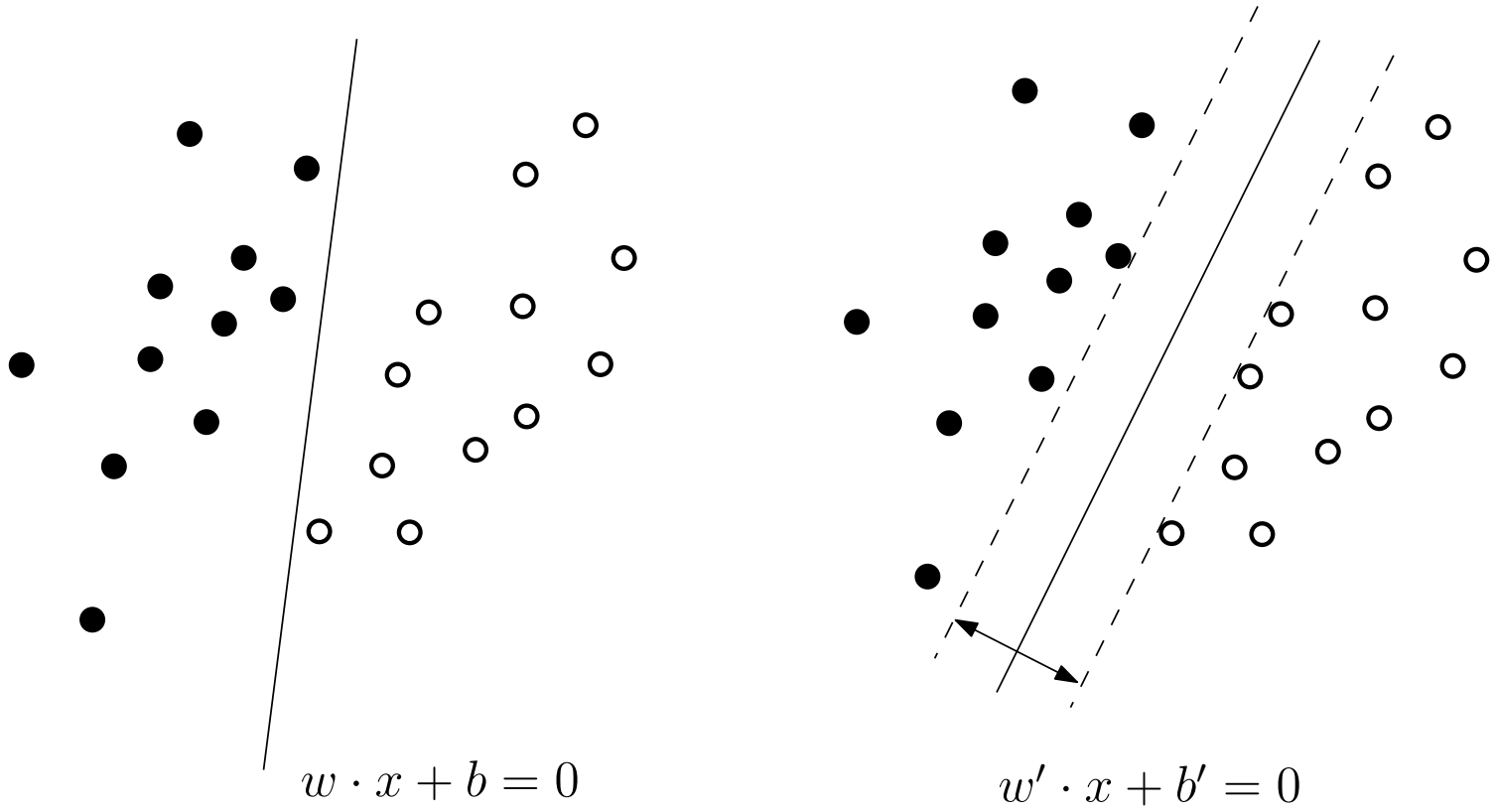
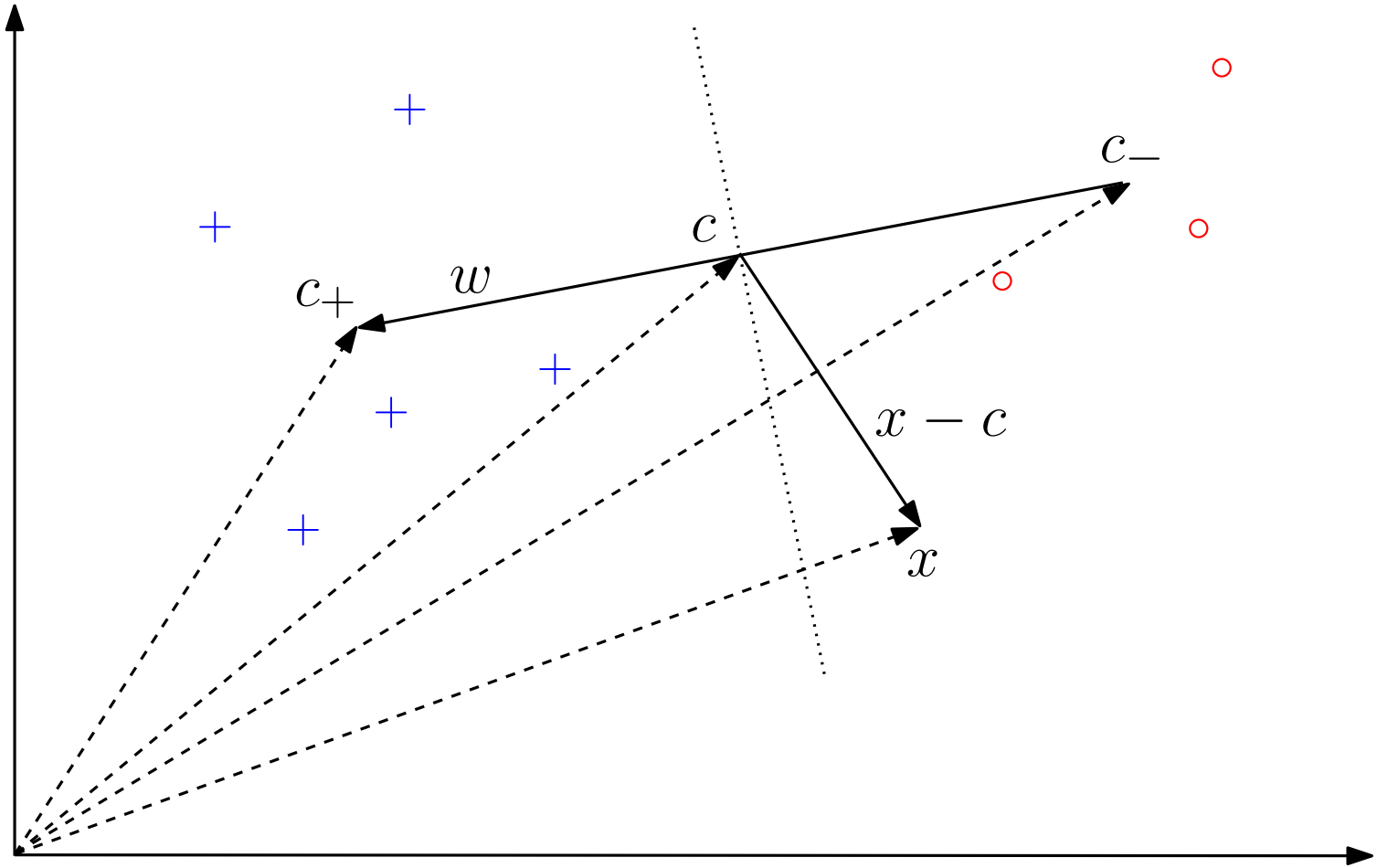


Figure 4.1: Separation by hyperplanes and *support vectors*. On the left, the white points ($+1$) and the black points (-1) are linearly separable. On the right we see the same set of points and the greatest margin separator, which is computed by the **SVM** algorithm described later.



A simple geometric classification algorithm: given two classes of points (depicted by 'o' and '+'), compute their means c_+, c_- and assign a test pattern x to the one whose mean is closer. This can be done by looking at the dot product between $x - c$ (where $c = (c_+ + c_-)/2$) and $w = c_+ - c_-$, which changes sign as the enclosed angle passes through $\pi/2$. Note that the corresponding decision boundary is a hyperplane (the dotted line) orthogonal to w [the perpendicular bisector of c_+c_-].

Adapted from [7, Fig. 1.1].

$\mathcal{X} \subseteq \mathbf{R}^n$ $(n \geqslant 1)$, $\mathcal{Y} = \mathbf{R}$.

$\mathcal{D} = \{(x^1, y^1), \ldots, (x^m, y^m)\}$ (dataset, $x^j \in \mathcal{X}$, $y^j \in \mathcal{Y}$).

Problem. Find an affine linear map $h : \mathbf{R}^n \to \mathbf{R}$,

$$h(x) = w \cdot x + w_0, \quad w \in \mathbf{R}^n, \ w_0 \in \mathbf{R},$$

such that $\hat{y}^j = h(x^j)$ are as close as possible to the y^j (*regression hyperplane*).

For $n = 1$, *regression line*; *regression plane* for $n = 2$.

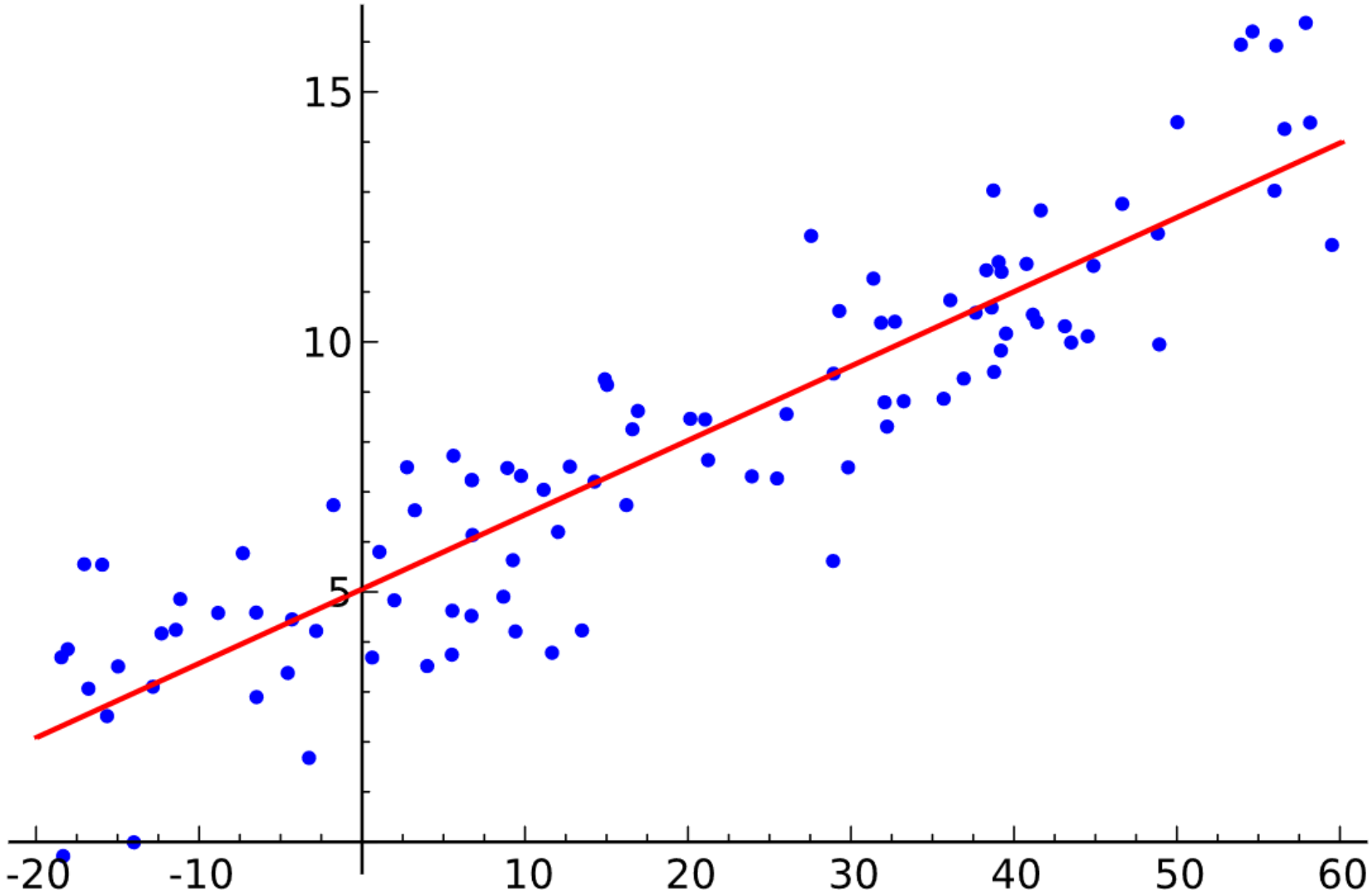


Figure 4.2: Regression line for a dataset in $\mathbf{R} \times \mathbf{R}$. Image from `https://en.wikipedia.org/wiki/Linear_regression`.

$\mathcal{D} = \{(x^1, y^1), \ldots, (x^m, y^m)\}$ $(x^j, y^j \in \mathbf{R})$.

Problem. Find a polynomial map $p : \mathbf{R} \to \mathbf{R}$ of degree r,

$$p(x) = w_0 + w_1 x + \cdots w_r \cdot x^r, \quad w_0, w_1, \cdots, w_r \in \mathbf{R},$$

such that $\hat{y}^j = p(x^j)$ are as close as possible to the y^j (*polynomial approximation of degree r*).

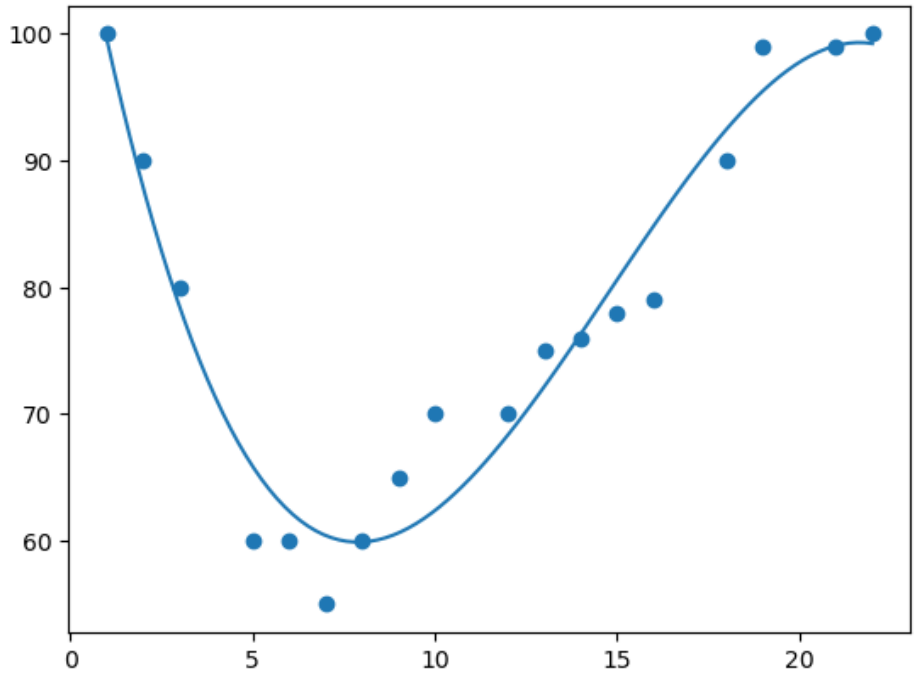


Figure 4.3: Cubic approximation of a dataset in $\mathbf{R} \times \mathbf{R}$.

The *logistic regression* is linear regression of $\log p/(1-p)$.

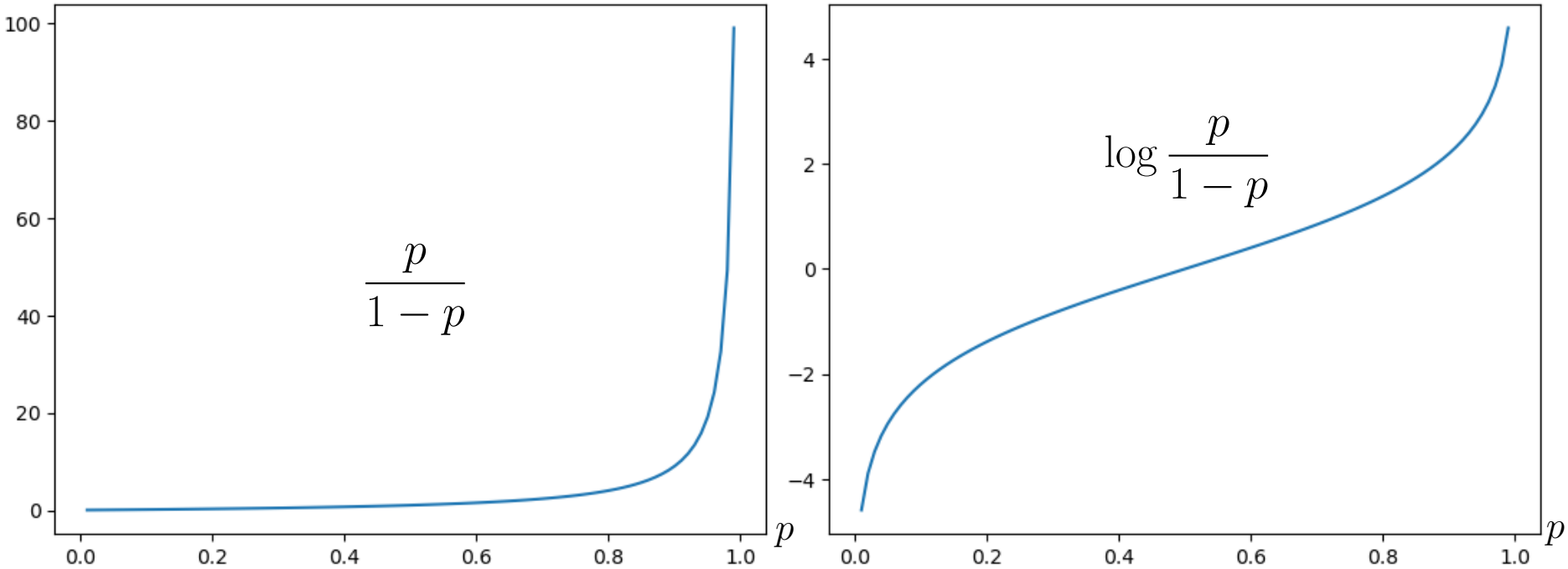


Figure 4.4: For probability values $p \in [0,1]$ it makes no sense to apply linear regression procedures. Left: graph of the *odds* function, $p/(1-p)$, for $p \in [0,1]$. Right: graph of $\log(p/1-p)$, with symmetry about the point $(1/2, 0)$, so linear regression of its values is in principle possible.

If $\log(p/1-p) = w \cdot x$ $(x, w \in \mathbf{R}^n)$, then $p = p(x) = 1/(1+e^{-w \cdot x})$ estimates the probability of the observations x.

The function $1/(1+e^{-t})$ is the *logistic function*. Its range is $(0,1)$. A variation is the function $(1-e^{-t})/(1+e^{-t})$, with range $(-1,1)$.

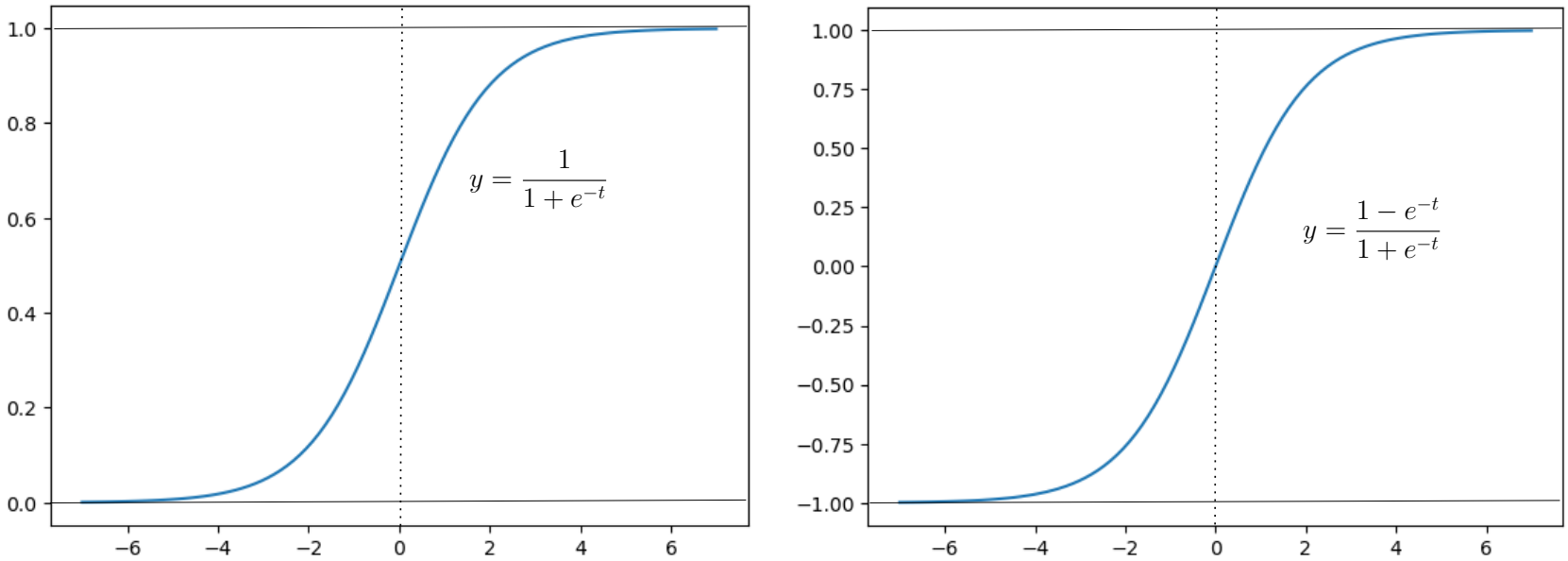


Figure 4.5: Logistic (or *sigmoid*) functions.

General references

General: [8], [9]*, [2]*, [10], [11], [12], [13], [14].

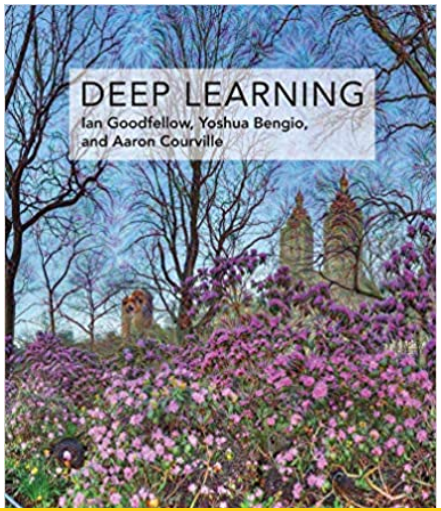
Bayesian approaches: [15], [16], [17].

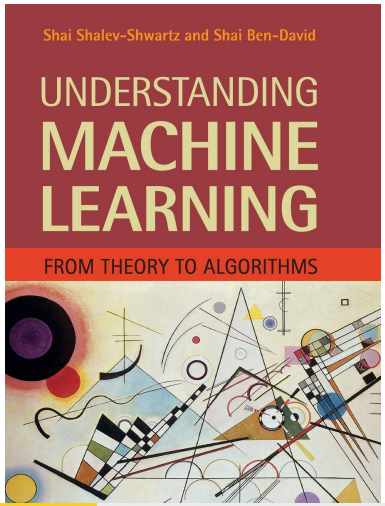
Applications: [18], [19], [20], [21], [22].

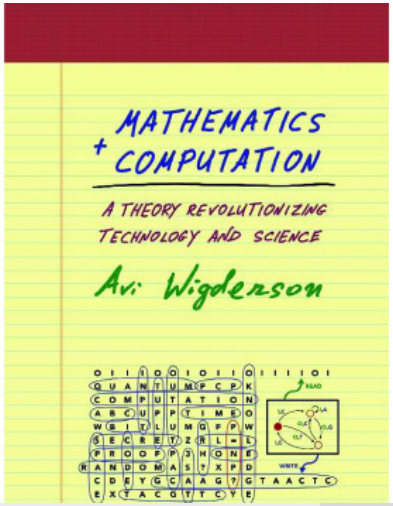
The many facets of the symbiosis Mathematics & Computation are appraised in [23]* (in particular, Chapter 17 is devoted to computational learning theory).

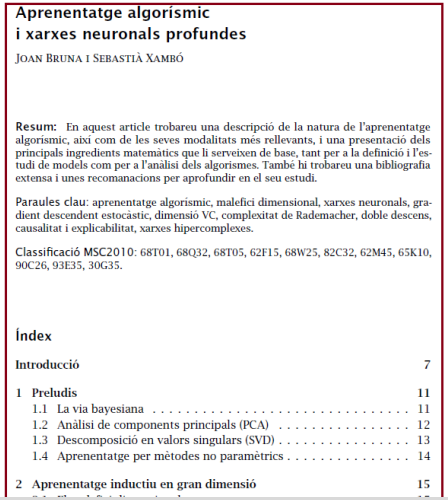
See also the extensive survey [24] and Marr's blog [25].

Expository (in Catalan): [26]*.








Aprenentatge algorísmic
i xarxes neuronals profundes

Joan Bruna i Sebastià Xambó

Resum: En aquest article trobareu una descripció de la natura de l'aprenentatge algorísmic, així com de les seves modalitats més rellevants, i una presentació dels principals ingredients matemàtics que li serveixen de base, tant per a la definició i l'estudi de models com per a l'anàlisi dels algorismes. També hi trobareu una bibliografia extensa i unes recomanacions per aprofundir en el seu estudi.

Paraules clau: aprenentatge algorísmic, malefici dimensional, xarxes neuronals, gradient descendent estocàstic, dimensió VC, complexitat de Rademacher, doble descens, causalitat i explicabilitat, xarxes hipercomplexes.

Classificació MSC2010: 68T01, 68Q32, 68T05, 62F15, 68W25, 82C32, 62M45, 65K10, 90C26, 93E35, 30G35.

Índex

A model for supervised learning

Data sources and experts/supervisors
Inductive hypotheses and their complexity
Loss functions
Empirical risk and the task of a supervised learner
Training and validation
Error decomposition and regret bound

$\mathcal{X}$, the space (or set) from which data is extracted.

Information theory allows to assume, in important cases, that this data is represented by *vectors* of some $\mathbf{R}^n$.

- For N pixel monochrome images, $\mathcal{X}$ is (a region of) $\mathbf{R}^N$.
- For RGB images, a region of $\mathbf{R}^{3N}$.

The dimension of these spaces for images of interest (N or $3N$) is very large.

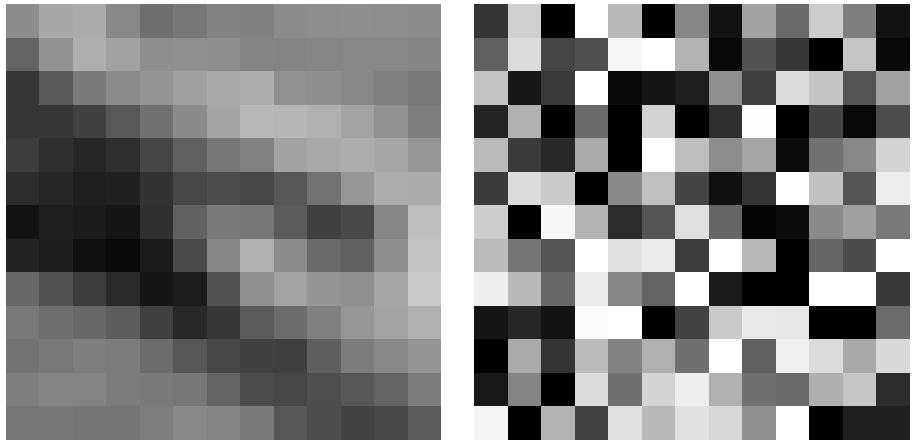
In general, then, we must be prepared to have to deal with input spaces $\mathcal{X}$ of very large dimension, a scenario in which the mathematical methods that work for low dimensions are no longer valid.

It is governed by the random selection of elements x of $\mathcal{X}$ according to a probability distribution P over $\mathcal{X}$, $x \sim P$ in symbols.

Generally, P is very far from the uniform distribution.

For example, the images we usually run into have regularities that markedly distinguish them from those formed by randomly and independently selected pixels according to the uniform distribution.

And a similar observation holds true for other forms of data, such as voice or music signals.



A function $f^* : \mathcal{X} \to \mathcal{Y}$.

For each $x \in \mathcal{X}$, $x \sim P$, the expert produces an *example*:

(x, y) with $y = f^*(x)$.

To deal with 'uncertainties' and 'noise', we may allow $f^*(x)$ to be a probability distribution P_x on $\mathcal{Y}$, which amounts to a probability distribution, still denoted by P, on $\mathcal{X} \times \mathcal{Y}$:

$P(x, y) = P(x)P_x(y)$.

It is a space $\mathcal{H}$ of functions $h : \mathcal{X} \to \mathcal{Y}$.

The selection of such a space is known as *inductive bias*, as it usually transcribes *a priori* heuristics about the expected form of the solution.

The space $\mathcal{H}$ is often specified as a parameterized set of functions:

$$\mathcal{H} = \{h_w(x) = H(w, x)\}_{w \in W},$$

where W is a set and $H : W \times \mathcal{X} \to \mathcal{Y}$.

- In the case of linear regression, $\mathcal{H}$ is the space of linear affine maps, or multivariate polynomials of degree 1. The same space is involved in the linearly separable binary classification.

- As we will see, a neural network can be seen as a parameterized family of nonlinear functions, a feature on which rests their resilience in algorithmic learning. In this context, the parameters are usually called *weights* of the network.

- In case the expert belongs to $\mathcal{H}$, we say that it is *realizable*.

The measure of the *complexity* of the hypotheses is a function $\gamma : \mathcal{H} \to \mathbf{R}_+$.

- *Example*: $\gamma(h) = \|h\|$ (the *norm* of h) when $\mathcal{H}$ has this resource.

For all $\delta \in \mathbf{R}_+$, we set $\mathcal{H}_\delta = \{h \in \mathcal{H} : \gamma(h) \leqslant \delta\}$.

- In the case of the norm, it is the (closed) ball of $\mathcal{H}$ of radius δ, which has the advantage of being a convex set.

The sets $\mathcal{H}_\delta$ provide a means for grading the search effort within $\mathcal{H}$ according to increasing complexity.

The *loss* of $h \in \mathcal{H}$ is an indicator, denoted by $L(h)$, of the separation between h i f^*.

- *Example.* The expectation $L(h) = \mathbb{E}_P |h(x) - f^*(x)|^2$.
- More generally, we can choose $L(h) = \mathbb{E}_P \, \ell(h(x), f^*(x))$, where $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbf{R}$ is a non-negative function with $\ell(y, y') = 0$ if and only if $y = y'$ (we say ℓ is a *point-wise loss* function).

The minimum loss achievable by functions $h \in \mathcal{H}$, $\min_{h \in \mathcal{H}} L(h)$, will be denoted by $L_{\mathcal{H}}$.

If $f^*(x)$ is a probability distribution P_x on $\mathcal{Y}$, $\ell(h(x), f^*(x))$ has to be replaced by $\mathbb{E}_{P_x} \ell(h(x), y)$.

In some application domains, the loss is called *risk*, and still in others, *error*.

A *training data set* is a set of examples

$$\mathcal{D} = \{(x^i, y^i = f^*(x^i)) : x^i \in \mathcal{X})\}_{i \in [m]}$$

produced by the expert, where $x^i \sim P$ independently.

In general terms, the goal of the learner is to approximate the expert f^*.

In terms of the ingredients available to the learner ($\mathcal{D}$, $\mathcal{H}$, L), this goal can be specified as

> *the production of an estimator* $\hat{f} \in \mathcal{H}$, *using only* $\mathcal{D}$, *such that* $L(\hat{f}) \approx L_{\mathcal{H}} = \min_{h \in \mathcal{H}} L(h)$.

The estimator $\hat{f}$ is constructed by minimizing the *empirical risk* $\hat{L}_{\mathcal{D}}(h)$ (for $h \in \mathcal{H}$), which is defined by the formula

$$\hat{L}_{\mathcal{D}}(h) = \frac{1}{m} \sum_{i=1}^{m} |h(x^i) - y^i|^2.$$

This minimization problem is called *empirical risk minimization* (ERM).

- The expression $|h(x^i) - y^i|^2$ has to be replaced by $\ell(h(x^i), y^i)$ if the loss function L is defined in terms of the point-wise loss ℓ.

As we will see later, the empirical risk minimization is carried out by methods that are generically known as *gradient descend algorithms*.

Since there is a (random) discrepancy between the functional we would like to minimize ($L(h)$, which is unknown to the learner) and the functional available to the learner (the empirical risk $\hat{L}_{\mathcal{D}}(h)$), it is necessary to introduce some kind of *restriction* or *regularization* in order to be able to handle these fluctuations.

For example, we can consider the δ-restricted empirical risk,

$$\hat{L}_{\mathcal{D},\delta} = \min_{h \in \mathcal{H}_\delta} \hat{L}_{\mathcal{D}}(h) \ , \tag{1}$$

or the λ-*regularized*, or λ-*penalized*, empirical risk,

$$\min_{h \in \mathcal{H}} \left(\hat{L}_{\mathcal{D}}(h) + \lambda \gamma(h) \right), \tag{2}$$

where $\gamma(h)$ is the complexity of h (introduced on page 31) and λ is a fixed positive constant.

- The role of the term $\lambda\gamma(h)$ is to penalize high complexity hypothesis.

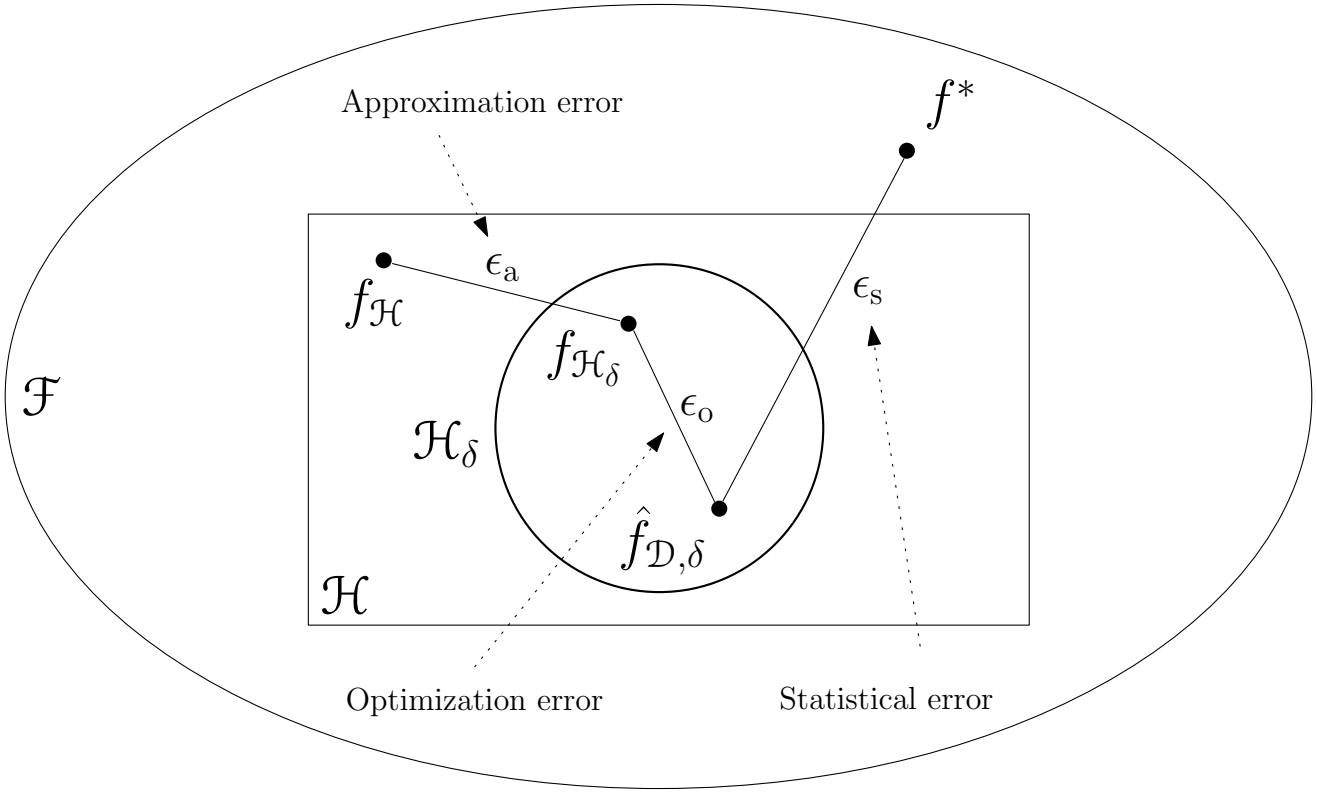


Figure 8.1: $\mathcal{F}$ denotes the (large) unknown universe to which the supervisor f^* belongs (thus $\mathcal{F}$ is a subset of the set of all maps $\mathcal{X} \to \mathcal{Y}$). We also depict the hypotheses space $\mathcal{H}$ as a subset of $\mathcal{F}$ (in general $f^* \notin \mathcal{H}$) and $\mathcal{H}_\delta$ (a ball of radius δ if the complexity is a norm). $f_{\mathcal{H}} \in \mathcal{H}$ and $f_{\mathcal{H}_\delta} \in \mathcal{H}_\delta$ denote functions achieving the least cost $L_{\mathcal{H}}$ and $L_{\mathcal{H}_\delta}$ for functions in $\mathcal{H}$ and in $\mathcal{H}_\delta$, respectively. And $\hat{f} = \hat{f}_{\mathcal{D},\delta} \in \mathcal{H}_\delta$ denotes the function returned by an ERM algorithm selected so that $\hat{L}_{\mathcal{D}}(\hat{f}) - \hat{L}_{\mathcal{D},\delta} \leqslant \epsilon$. It depends on $\mathcal{D}$, δ and ϵ and its computation cost increases when δ increases or ϵ decreases.

Given an estimator $\hat{f} \in \mathcal{H}$ supplied by ERM, now we seek to bound the difference $L(\hat{f}) - L_{\mathcal{H}}$, an amount that some authors call *regret* (of having chosen $\hat{f}$), as it expresses the discrepancy that would be reported by an orable that knew $L(\hat{f})$ and $L_{\mathcal{H}}$.

This regret can be decomposed as follows ([27], [28], [13]):

$$L(\hat{f}) - L_{\mathcal{H}} = L(\hat{f}) - L_{\mathcal{H}_\delta} + L_{\mathcal{H}_\delta} - L_{\mathcal{H}}.$$

The significance of $L_{\mathcal{H}_\delta} - L_{\mathcal{H}}$ is described below under the name of *approximation error* and is denoted ϵ_{a} (cf. Fig. 8.1).

On the other hand we can write

$$L(\hat{f}) - L_{\mathcal{H}_\delta} = L(\hat{f}) - \hat{L}_{\mathcal{D}}(\hat{f}) + \hat{L}_{\mathcal{D}}(\hat{f}) - \hat{L}_{\mathcal{D},\delta} + \hat{L}_{\mathcal{D},\delta} - L_{\mathcal{H}_\delta}.$$

The difference $\epsilon_{\mathrm{o}} = \hat{L}_{\mathcal{D}}(\hat{f}) - \hat{L}_{\mathcal{D},\delta}$ is analyzed below under the label of *optimization error* (cf. Fig. 8.1).

It remains to consider the sum $L(\hat{f}) - \hat{L}_{\mathcal{D}}(\hat{f}) + \hat{L}_{\mathcal{D},\delta} - L_{\mathcal{H}_\delta}$, which we proceed to bound above.

On one hand, it is clear that $L(\hat{f}) - \hat{L}_{\mathcal{D}}(\hat{f}) \leqslant \epsilon_{\mathrm{s}}$, where $\epsilon_{\mathrm{s}} = \sup_{h \in \mathcal{H}_\delta} |L(h) - \hat{L}_{\mathcal{D}}(h)|$ and studied below under the name *statistical error*, or also *fluctuation error*.

And on the other hand we also have $\hat{L}_{\mathcal{D},\delta} - L_{\mathcal{H}_\delta} \leqslant \epsilon_{\mathrm{s}}$, because if $h \in \mathcal{H}_\delta$ satisfies $L_{\mathcal{H}_\delta} = L(h)$, then

$$\hat{L}_{\mathcal{D},\delta} - L_{\mathcal{H}_\delta} \leqslant \hat{L}_{\mathcal{D}}(h) - L(h) \leqslant |L(h) - \hat{L}_{\mathcal{D}}(h)| \leqslant \epsilon_{\mathrm{s}}.$$

These considerations can be summarized as follows (cf. Fig. 8.1):

Theorem (Regret bound)

$$L(\hat{f}) - L_{\mathcal{H}} \leqslant \epsilon_{\mathrm{a}} + \epsilon_{\mathrm{o}} + 2\epsilon_{\mathrm{s}}.$$

It has been defined as the difference $\epsilon_{\mathrm{a}} = L_{\mathcal{H}_\delta} - L_{\mathcal{H}}$.

- It is non-negative and decreases when δ increases;
- It does not depend on the data $\mathcal{D}$;
- It measures the approximation of the minimum loss $L_{\mathcal{H}}$ of $\mathcal{H}$ that can be achieved with functions from $\mathcal{H}_\delta$.

For $h \in \mathcal{H}_\delta$, $\epsilon_{\mathrm{o}} = \hat{L}_{\mathcal{D}}(h) - \hat{L}_{\mathcal{D},\delta}$, where $\hat{L}_{\mathcal{D},\delta}$ is the least empirical loss of functions from $\mathcal{H}_\delta$.

In practice, a tolerance $\epsilon > 0$ is fixed and an ERM algorithm is run to produce $\hat{f} \in \mathcal{H}_\delta$ such the the optimization error is $\leqslant \epsilon$, that is,

$$\hat{L}_{\mathcal{D}}(\hat{f}) - \hat{L}_{\mathcal{D},\delta} \leqslant \epsilon. \tag{3}$$

The main effort to achieve (3) is computational, and it increases when ϵ decreases.

For $h \in \mathcal{H}_\delta$, $|L(h) - \hat{L}_\mathcal{D}(h)|$ is the error produced on substituting the loss $L(h)$ by the empirical loss $\hat{L}_\mathcal{D}(h)$.

In the worst case, this error is $\epsilon_{\rm s} = \sup_{h \in \mathcal{H}_\delta} |L(h) - \hat{L}_\mathcal{D}(h)|$, a quantity introduced before as *statistical error*, or *fluctuation error*. It is also *generalization error*.

Appendices

The Vapnik-Chervonenkis capacity
The Pollard dimension
The Rademacher complexity

Assume that $\mathcal{H}$ is a space binary hipothesis $h : \mathcal{X} \to \{0, 1\}$, which we can identify with subsets of $\mathcal{X}$: $h \leftrightarrow h^{-1}(1) = 1_{h(x)=1}$.

We say that $\mathcal{H}$ *shatters* a finite subset $Z \subset \mathcal{X}$ when any binary function of Z is the restriction to Z of some $h \in \mathcal{H}$.

The Vapnik-Chervonenkis *dimension* (or *capacity*) of $\mathcal{H}$, denoted by $\mathrm{VC}(\mathcal{H})$, is *the maximum cardinal of a finite subset $Z \subset \mathcal{X}$ shattered by $\mathcal{H}$*, if this maximum exists, and ∞ otherwise.

To determine that $\mathrm{VC}(\mathcal{H}) = k < \infty$, if suffices to exhibit a subset Z of cardinal k shattered by $\mathcal{H}$ and to show that *no* subset of cardinal $k + 1$ can be shattered by $\mathcal{H}$.

In the case of infinite capacity, it is required to establish that for any $k \in \mathbf{N}$ there is a subset of cardinal k that is shattered by $\mathcal{H}$.

Example (1). Let $\mathcal{X} = \mathbf{R}$ and let $\mathcal{H}$ be the set of *positive semilines* $[a, \infty)$, $a \in \mathbf{R}$. Then $\mathrm{VC}(\mathcal{H}) = 1$. Indeed, given $Z = \{z\}$, $z \in \mathcal{X}$, there are semilines that contain z and others that do not, which says that Z is shattered by $\mathcal{F}$. On the other hand, if $Z = \{z, z'\} \subset \mathbf{R}$, $z < z'$, any semiline containing z also contains z', which shows that Z is not shattered by $\mathcal{F}$.

Example (2). Still in $\mathbf{R}$, consider the set $\mathcal{H}$ of *closed intervals*. Then $\mathrm{VC}(\mathcal{H}) = 2$. Indeed, if $Z = \{z, z'\}$ as in the previous example, there are closed intervals that are disjoint from Z, others that contain Z, and others that contain z and not z' or that contain z' and not z. But if $Z = \{z, z', z''\}$, $z < z' < z''$, there is no closed interval containing z and z'' that does not contain z', which shows that no Z of cardinal 3 can be shattered by $\mathcal{H}$. Notice that in the example (1) we would also have capacity 2 if we had considered positive and negative semilines.

Example (3). Let $\mathcal{H}$ be the space of *half-planes* in the plane $\mathcal{X} = \mathbf{R}^2$. A set Z of three non-colinear points of $\mathcal{X}$ is shattered by $\mathcal{F}$, as for any subset Z' of Z there are half-planes h such that $h \cap Z = Z'$. On the other hand, no subset of $\mathcal{X}$ of cardinal 4 can be shattered by $\mathcal{H}$. To see this, first note that we may assume that Z does not contain three colinear points, for a half-plane that contains the endpoints of a segment in fact contains the whole segment. We can also assume that none of the four points lies in the interior of the triangle formed by the other three, for this point would automatically belong to any half-plane containing the triangle. So we may assume that the four points form a convex quadrilateral. But in this case no half-plane that contains the endpoints of one diagonal can exclude the endpoints of the other diagonal. In conclusion, $\mathrm{VC}(\mathcal{H}) = 3$.

Example (4). Example (3) is valid in any dimension $d \geqslant 2$, in the sense that the capacity of the set $\mathcal{H}$ of *half-spaces* of $\mathcal{X} = \mathbf{R}^d$ is $d+1$. To see this, the easiest part is to find a set Z of cardinal $d+1$ that is shattered by $\mathcal{H}$. Let $Z = \{z_0, z_1, \ldots, z_d\}$, where z_0 is the origin and z_j, $j = 1, \cdots, d$, the unit point on j-th coordinate axis. Let $\{0, 1, \ldots, d\} = A \sqcup B$ be an arbitrary partition of $\{0, 1, \ldots, d\}$ and set $w_j = 1$ if $j \in A$ and $w_j = -1$ if $j \in B$. Then the half-space defined by the hyperplane $h(x) = w_0/2 + w_1 x_1 + \cdots + w_d x_d = 0$ contains (excludes) the points z_j for $j \in A$ ($j \in B$). Indeed, we have $h(z_0) = w_0/2$ and $h(z_j) = w_0/2 + w_j$ for $j > 0$, which imply that $h(z_j)$ has the same sign as w_j for any j. Thus Z is shattered by $\mathcal{H}$. To conclude, we need to show that no subset Z of cardinal $d+2$ can be shattered by $\mathcal{H}$. This is a consequence of the fact that there exists (Radon lemma) a partition $Z = Z' \sqcup Z''$ such that $[Z'] \cap [Z''] \neq \emptyset$, where $[Z']$ and $[Z'']$ denote the convex hulls of Z' and Z''. Indeed, if there were a half-space separating Z' and Z'', it would also separate $[Z']$ and $[Z'']$, which is impossible.

Example (5). Consider the family $\mathcal{H}$ of *convex polygons* in $\mathbf{R}^2$ with r sides at most. We are going to see that the VC capacity of this family is $2r+1$ (for the discussions that follow, see the illustrations in Fig. 10.1). Indeed, let Z be a set of $2r+1$ points on a circumference C and form an arbitrary partition $Z = Z' \sqcup Z''$. Since either $|Z'| \leqslant r$ or $|Z''| \leqslant r$, we will deal with each case separately. If $|Z'| \leqslant r$, the convex polygon P' whose vertices are Z' satisfies $P' \cap C = Z'$ and hence $P' \cap Z = Z'$. If instead we have $|Z''| \leqslant r$, the polygon P'' whose sides are the tangents to C at the points of Z'', displaced (the tangents) infinitesimally toward the center of C, satisfies $P'' \cap Z = Z'$.

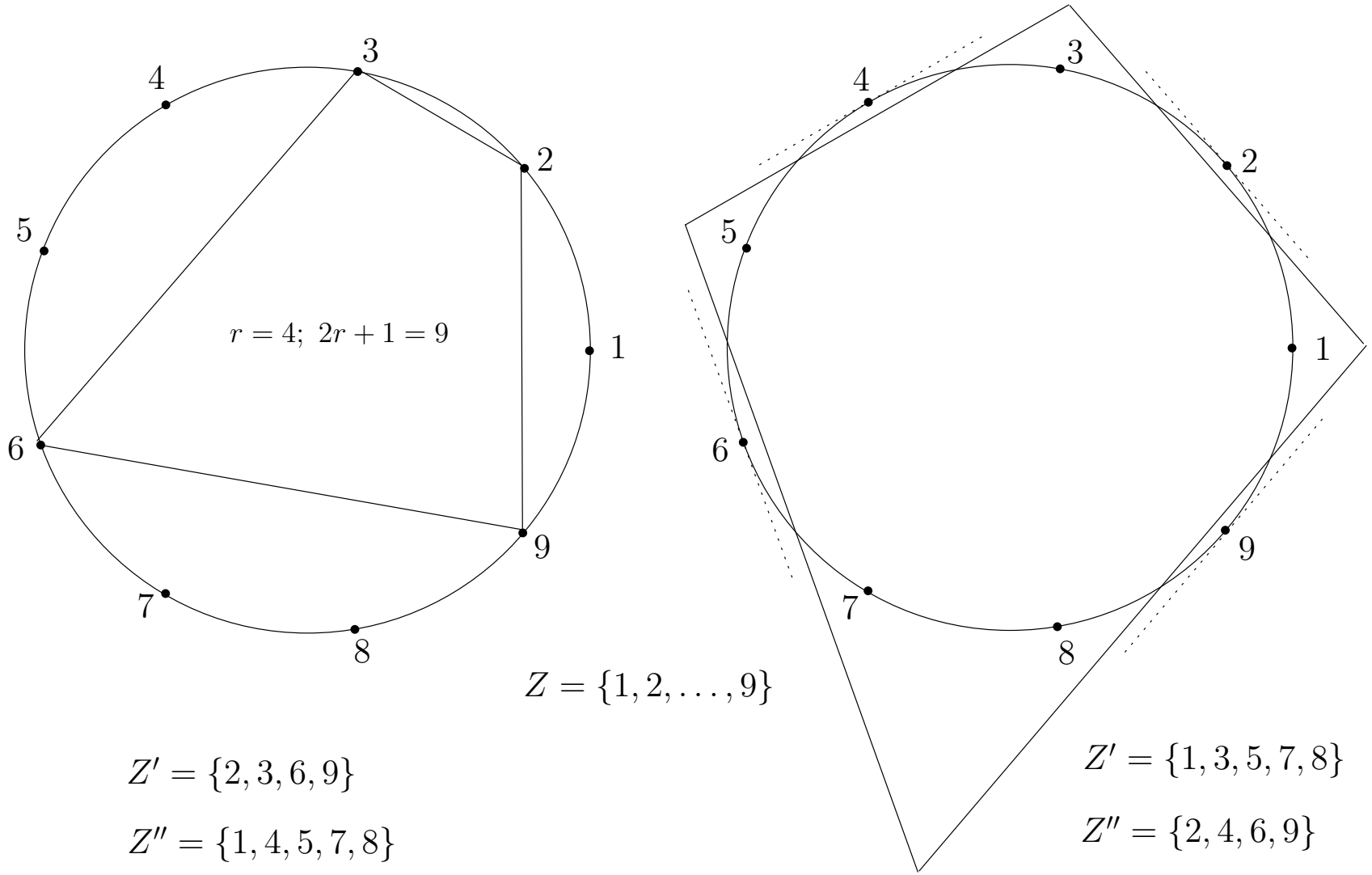


Figure 10.1: The VC capacity of convex 4-gons is 9. Left: the convex hull $[\{2, 3, 6, 9\}]$ excludes $\{1, 4, 5, 7, 8\}$. Right: construction of a convex 4-gon including $\{1, 3, 5, 7, 8\}$ and excluding $\{2, 4, 6, 9\}$.

Now we have to show that no set Z of $2r+2$ points can be shattered by $\mathcal{H}$. Assume first that the points are the vertices of a convex $2r+2$-gon, and number them consecutively around with $j = 1, \cdots, 2r+2$. Let Z' (Z'') be the subset of Z whose index is odd (even). Then there is no convex r-gon P such that $P \cap Z = Z'$ because the r sides of P ought to separate the $r+1$ pairs of vertices with indices $2j-1, 2j$, $j = 1, \cdots, r+1$ (see Fig. 10.2). To end, note that if the points Z are not the vertices of a convex $2r+2$-gon, then one of the points, say z, lies in the convex hull of the remaining points, which implies that there is no convex r-gon P such that $P \cap Z = Z - \{z\}$.

The reasoning so far can be easily modified in order to show that the family of convex polygons with exactly r sides also has capacity $2r+1$. Finally, note that the family of all convex polygons, with any number of sides, has capacity ∞.

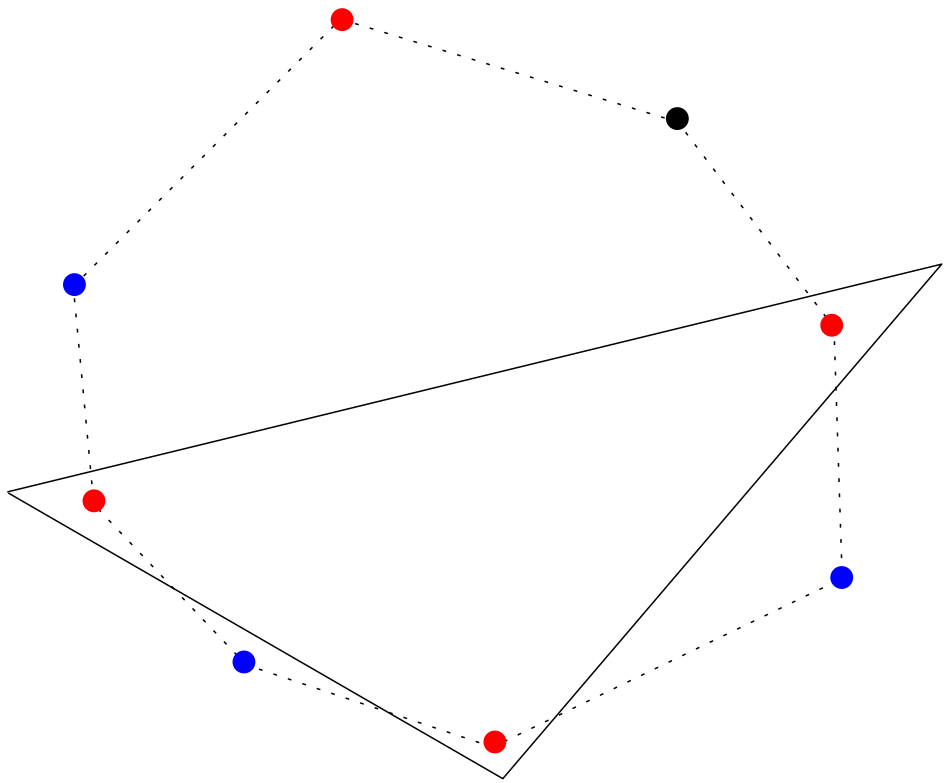


Figure 10.2: The VC capacity of triangles is 7. The image illustrates that no triangle can contain the four red dots and exclude the four blue dots.

The role of VC capacity with respect to the *generalization problem* is encapsulated in the following statement.

Theorem. Let $k = \mathrm{VC}(\mathcal{H})$ and assume that $k < \infty$. Then we have:

(1) $L(h) \leqslant_\delta \hat{L}_{\mathcal{D}}(h) + O\left(\sqrt{\frac{k+\ln(1/\delta)}{n}}\right)$.

Alternatively, $n \geqslant O\left(\frac{1}{\epsilon^2}(k + \ln(1/\delta))\right)$ guarantees that $|L(h) - \hat{L}_{\mathcal{D}}(h)| \leqslant_\delta \epsilon$.

(2) If $h \in \mathcal{H}$ satisfies $L_{\mathcal{D}}(h) = 0$, then $n \geqslant \frac{8}{\epsilon}(k \ln(16/\epsilon) + \ln(2/\delta))$ guarantees that $L(h) \leqslant_\delta \epsilon$. *Remark: the expression of the lower bound on n is $O\left(\frac{k}{\epsilon}\ln(1/\epsilon) + \frac{1}{\epsilon}\ln(1/\delta)\right)$.*

In terms of loss, $L(h) \leqslant_\delta O\left(\frac{1}{n}(k \ln(n/k) + \ln(1/\delta))\right)$ if $L_{\mathcal{D}}(h) = 0$.

To end, let us also take notice of what happens when $\mathrm{VC} = \infty$ (cf. [9, Th 6.6].

Theorem For a class of binary hypothesis $\mathcal{H}$ such that $\mathrm{VC}(\mathcal{H}) = \infty$, *there exist oracles that no algorithm can learn*.

This notion amounts to an extension of the VC capacity to the case of real-valued functions.

Assume that $\mathcal{Y} = [0, K] \subset \mathbf{R}$, so that $\mathcal{H}$ is a set of functions $h : \mathcal{X} \to [0, K]$. Let $X = \{x^1, \ldots, x^m\}$ be a subset of $\mathcal{X}$. We say that $\mathcal{H}$ *shatters* X with *witnesses* $t_1, \ldots, t_m \in \mathbf{R}$ if for any subset X' of X there is a function $h \in \mathcal{H}$ such that $h(x^j) \leqslant t_j$ or $h(x_j) > t_j$ according to whether $x_j \in X'$ or $x_j \notin X'$.

The *Pollard capacity* (or *dimension*) of $\mathcal{H}$, which we will denote $\mathrm{Pol}(\mathcal{H})$, is the greatest cardinal of a subset X of $\mathcal{X}$ that $\mathcal{H}$ can shatter. In the binary case, $\mathcal{Y} = \{0, 1\}$, we clearly have $\mathrm{Pol}(\mathcal{F}) = \mathrm{VC}(\mathcal{F})$. Another example is the equality $\mathrm{Pol}(\mathcal{H}) = \dim(\mathcal{H})$ when $\mathcal{H}$ is a vector space of real functions of finite dimension.

With the notations introduced above, let $p = \mathrm{Pol}(\mathcal{H})$ and $\mathcal{D} \sim P^m$, that is, a subset of m elements of $\mathcal{X}$ drawn independently according to the probability P. Then we have:

Theorem (Pollard, 1984) For any $h \in \mathcal{H}$,

$$\left|\hat{L}_{\mathcal{D}}(h) - \mathbb{E}_{x\sim P}[h(x)]\right| \leqslant_{\delta} K\sqrt{\frac{2p}{m}\ln\frac{em}{p}} + K\sqrt{\frac{1}{2m}\ln\frac{1}{\delta}}.$$

Alternatively,

$$\left|\hat{L}_{\mathcal{D}}(h) - \mathbb{E}_{x\sim P}[h(x)]\right| \leqslant_{\delta} \epsilon \text{ for } m \geqslant \frac{8K^2}{\epsilon^2}\left(p\ln\frac{8K^2}{\epsilon^2} + \frac{1}{4}\ln\frac{1}{\delta}\right).$$

This notion can be defined for any family $\widetilde{\mathcal{H}}$ of functions $\tilde{h} : \mathcal{Z} \to [a, b] \subset \mathbf{R}$, where $(\mathcal{Z}, P)$ is a probability space, and its purpose is to gauge the capacity of $\widetilde{\mathcal{H}}$ to accommodate a certain kind of binary random noise.

In the application to the basic model, we will have $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$, with $P(z) = P(x, y) = P(y|x)P(x) = P_x(y)P(x)$, and $\widetilde{\mathcal{H}}$ will be the family of functions $\tilde{h} : \mathcal{X} \times \mathcal{Y} \to \mathbf{R}$, $h \in \mathcal{H}$, defined by the relation:

$$\tilde{h}(x, y) = \ell(h(x), y), \tag{4}$$

where ℓ is a point-wise loss function.

In fact, it is convenient to cosider two notions of Rademacher complexity: $\mathrm{RAD}_{\boldsymbol{z}}(\widetilde{\mathcal{H}})$, where $\boldsymbol{z} = \{z_1, \ldots, z_m\}$ is a sample ($z_j \sim P$), and $\mathrm{RAD}_m(\widetilde{\mathcal{F}})$.

The latter is the *Rademacher complexity* (for samples of length m) and the former the *empirical Rademacher complexity* (relative to the sample $\boldsymbol{z}$).

To define them, we need to introduce the *Rademacher variables* $\boldsymbol{\sigma} = \sigma_1, \ldots, \sigma_m$ to represent a binary random noise. These are independent random variables, one for each item in the sample, with equiprobable values $\{-1, 1\}$.

The correlation of a sample of this noise and the values $\tilde{h}(z) = \tilde{h}(z_1), \ldots, \tilde{h}(z_m)$ is expressed by the sample mean of the dot product $\boldsymbol{\sigma} \cdot \tilde{h}(z)$, namely $\boldsymbol{\sigma} \cdot \tilde{h}(z)/m$.

With this notion we are ready to define the two flavors of the Rademacher complexity:

$$\mathrm{RAD}_{\boldsymbol{z}}(\widetilde{\mathcal{H}}) = \mathbb{E}_{\boldsymbol{\sigma}}\left[\sup_{\tilde{h}\in\widetilde{\mathcal{F}}} \frac{\boldsymbol{\sigma}\cdot\tilde{h}(\boldsymbol{z})}{m}\right] \text{ and } \mathrm{RAD}_m(\widetilde{\mathcal{H}}) = \mathbb{E}_{\boldsymbol{z}\sim P^m}[\mathrm{RAD}_{\boldsymbol{z}}(\widetilde{\mathcal{F}})]. \quad (5)$$

Theorem For all $\delta > 0$ and $\tilde{h} \in \widetilde{\mathcal{H}}$, the following inequalities hold:

$$\boldsymbol{E}_{z\sim P}[\tilde{h}(z)] \leqslant_\delta \hat{L}_{\boldsymbol{z}}(\tilde{h}) + 2\mathrm{RAD}_{\boldsymbol{z}}(\widetilde{\mathcal{H}}) + 3\sqrt{\frac{\ln\frac{2}{\delta}}{2n}} \quad (6)$$

$$\boldsymbol{E}_{z\sim P}[\tilde{h}(z)] \leqslant_\delta \hat{L}_{\boldsymbol{z}}(\tilde{h}) + 2\mathrm{RAD}_n(\widetilde{\mathcal{F}}) + \sqrt{\frac{\ln\frac{1}{\delta}}{2n}}. \quad (7)$$

Translating these results to the basic model, via the space $\widetilde{\mathcal{H}}$ associated to $\mathcal{H}$ described at the beginning of this section, we obtain:

Theorem (Rademacher bound) Let $\mathcal{H}$ a hypothesis space and $\mathcal{D}$ a dataset of length m. Then, for all $h \in \mathcal{H}$, the following inequality holds:

$$L(h) \leqslant_\delta \hat{L}_{\mathcal{D}}(h) + 2\mathrm{RAD}_m(\mathcal{H}) + \sqrt{\frac{\ln(1/\delta)}{2m}}. \tag{8}$$

If we regard $\tilde{h}(z) \in \mathbf{R}^m$ as an abstract vector $\boldsymbol{a} \in \mathbf{R}^m$, we can define $\mathrm{RAD}(A) = \boldsymbol{E}_{\boldsymbol{\sigma}}[\sup_{\boldsymbol{a}\in A} \frac{\boldsymbol{\sigma}\cdot\boldsymbol{a}}{m}]$ for any $A \subseteq \mathbf{R}^n$, a notion that facilitates the study of the RAD properties.

Theorem (RAD bound) Let $A \subset \mathbf{R}^m$ be finite, and let $L = \sup_{\boldsymbol{a} \in A} \|\boldsymbol{a} - \bar{\boldsymbol{a}}\|$, where $\bar{\boldsymbol{a}}$ is the centroid of A. Then

$$\mathrm{RAD}(A) \leqslant \frac{L\sqrt{2 \ln |A|}}{m}. \tag{9}$$

If $\mathcal{H}$ is a hypothesis class such that $\mathrm{VC}(\mathcal{H}) = k$, and $\mathcal{D} \sim P^m$, then

$$\mathrm{RAD}_{\mathcal{D}}(\mathcal{H}) \leqslant \sqrt{\frac{2k \ln m}{m}}. \tag{10}$$

References I

[1] J. Hawkins, *A thousand brains: a new theory of intelligence*.

Basic Books New York, 2021.

Foreword by Richard Dawkins.
https://numenta.com/a-thousand-brains-by-jeff-hawkins.

[2] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*.

MIT press, 2016.

http://www.deeplearningbook.org.

References II

[3] S. Samoili, M. López Cobo, E. Gómez, G. De Prato, F. Martínez-Plumed, and B. Delipetrev, "AI Watch. Defining Artificial Intelligence. Towards an operational definition and taxonomy of artificial intelligence," *Publications Office of the European Union*, 2020.

[4] G. Lample and F. Charton, "Deep learning for symbolic mathematics," 2019.

`https://arxiv.org/pdf/1912.01412.pdf`.

[5] E. Davis, "The use of deep learning for symbolic integration: A review of (Lample and Charton, 2019)," 2019.

`https://arxiv.org/pdf/1912.05752.pdf`.

References III

[6] A. Z. Wagner, "Constructions in combinatorics via neural networks," 2021.

`https://arxiv.org/pdf/2104.14516.pdf`.

[7] B. Schölkopf and A. J. Smola, *Learning with kernels: support vector machines, regularization, optimization, and beyond*.

MIT Press, 2002.

xviii + 626 pp.

[8] S. Haykin, "Neural networks and learning machines," 2009.

xxx + 906 pp.

References IV

[9] S. Shalev-Shwartz and S. Ben-David, *Understanding machine learning: From theory to algorithms*.

Cambridge university press, 2014.

440 pp.

[10] M. Mohri, A. Rostamizadeh, and A. Talwalkar, *Foundations of machine learning*.

MIT press, 2018.

xvi + 486 pp.

[11] G. Rebala, A. Ravi, and S. Churiwala, *An Introduction to Machine Learning*.

Springer, 2019.

References V

[12] E. Alpaydin, *Introduction to machine learning (fourth edition)*.

Adaptive computation and machine learning, MIT press, 2020.

xxiv + 682 pp. 1st edition: 2004; 2nd, 2010; 3rd, 2014.

[13] F. Bach, "Learning theory from first principles, draft," 2021.

`https://www.di.ens.fr/~fbach/ltfp_book.pdf`.

[14] D. A. Roberts, S. Yaida, and B. Hanin, *The Principles of Deep Learning Theory: An Effective Theory Approach to Understanding Neural Networks*.

Cambridge University Press, 2021.

Forthcoming. Draft: `https://arxiv.org/pdf/2106.10165.pdf`.

References VI

[15] S. Theodoridis, *Machine learning: a Bayesian and optimization perspective*.

Academic Press, 2015.

[16] A. B. Patel, T. Nguyen, and R. G. Baraniuk, "A probabilistic theory of deep learning," *arXiv preprint arXiv:1504.00641*, 2015.

[17] S. Russell and P. Norvig, *Artificial intelligence: A modern approach (Third edition)*.

Pearson, 2016.

[18] A. Said and V. Torra, *Data Science in Practice*.

Springer, 2019.

References VII

[19] V. E. Balas, S. S. Roy, D. Sharma, and P. Samui, *Handbook of deep learning applications*, vol. 136.

Springer, 2019.

[20] J. Guttag, *Introduction to computation and programming using Python: With application to understanding data*.

MIT Press, 2016.

[21] A. Nandy and M. Biswas, *Reinforcement Learning: With Open AI, TensorFlow and Keras Using Python*.

Apress, 2017.

[22] F. Chollet, *Deep learning with Python*.

Manning, 2018.

References VIII

[23] A. Wigderson, *Mathematics and computation*.

Princeton University Press, 2019.

https://www.math.ias.edu/files/mathandcomp.pdf.

[24] G. Nguyen, S. Dlugolinsky, M. Bobák, V. Tran, Á. L. García, I. Heredia, P. Malík, and L. Hluchỳ, "Machine Learning and Deep Learning frameworks and libraries for large-scale data mining: a survey," *Artificial Intelligence Review*, vol. 52, no. 1, pp. 77–124, 2019.

[25] B. Marr, "The Key Definitions Of Artificial Intelligence (AI) That Explain Its Importance," Consulted 2021.

https://bernardmarr.com/
the-key-definitions-of-artificial-intelligence-ai-that-explain

References IX

[26] J. Bruna and S. Xambó-Descamps, "Aprenentatge algorísmic i xarxes neuronals profundes," Butlletí de la SCM, vol. 36, no. 1, pp. 5–67.

[27] L. Bottou and O. Bousquet, "The tradeoffs of large scale learning," in *Advances in neural information processing systems*, pp. 161–168, 2008.

[28] F. Bach, "Breaking the curse of dimensionality with convex neural networks," *The Journal of Machine Learning Research*, vol. 18, no. 1, pp. 629–681, 2017.