Machine Learning II

Introduction

Souhaib Ben Taieb

February 8, 2022

University of Mons

Table of contents

S-INFO-075: Machine Learning II

- Everything in English (lectures, labs, communications, etc)
- **Instructor**
    - Prof. Souhaib BEN TAIEB
    - De Vinci Building, second floor, room 2.15
    - Email: souhaib.bentaieb@umons.ac.be
- **Teaching assistant**
    - Sukanya Patra (PhD student)
    - De Vinci Building, ground floor
    - Email: sukanya.patra@umons.ac.be
- **Course Webpage**
    - `https://github.com/bsouhaib/ML2-2022`
    - Lecture notes, project details, etc.
- **Moodle**
    - `https://moodle.umons.ac.be/course/view.php?id=2786`
    - Forum for asking questions, etc.
    - **No email please — use the Moodle forum**

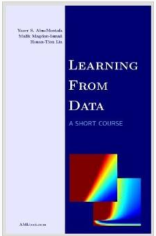
Prerequisites

- Machine learning I (S-INFO-256)
- Probability and Statistics
- Multivariate calculus
- Linear algebra
- Optimization (linear and non-linear)

Key reference

Yaser S. Abu-Mostafa, Malik Magdon-Ismail, Hsuan-Tien Lin (2012) Learning from Data. AMLBook.

https://work.caltech.edu/telecourse.html



Assessment

- Exam (E) (*open book*): **60%**
- Project (P): **20%**
- Four assignments (A): **20%** (5% each)
- Final mark:
  - If $E \geq 45\%$ and $P \geq 45\%$ and $A \geq 45\%$:
    - Final mark $= E \times 0.6 + P \times 0.2 + A \times 0.2$
  - Otherwise:
    - Final mark $= \min(E, P, A)$

Task	**Due Date**	**Value**
Final exam	Official exam period	60%
Project	TBA	20%
Assignments 1–4	TBA	20%

Main topics

- Theory of learning
- Linear models (classification and regression)
- (Stochastic) Gradient descent
- (Deep) Neural networks and backpropagation
- (?) Support Vector Machines
- (?) Recommender systems, text mining, ...

Table of contents

Supervised learning

Components of learning

Metaphor: Credit approval

Applicant information:

age	23 years
gender	male
annual salary	\$30,000
years in residence	1 year
years in job	1 year
current debt	\$15,000
...	...

Approve credit?

Components of learning

Components of learning

Formalization:

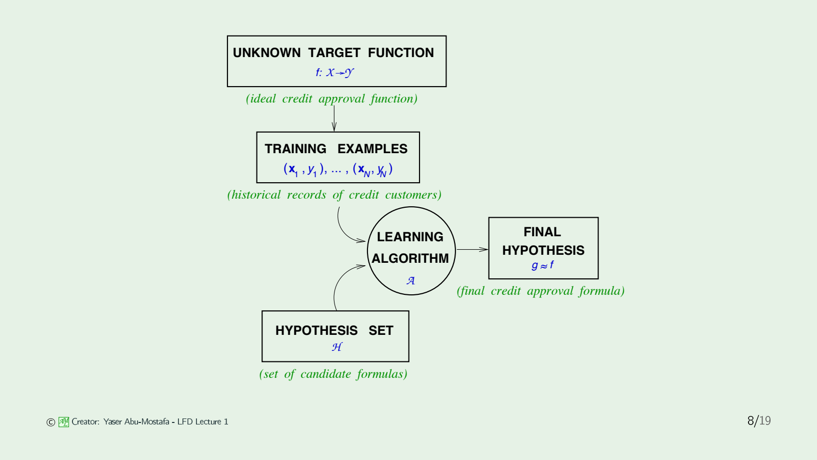
- Input: $\mathbf{x}$ *(customer application)*
- Output: y *(good/bad customer?)*
- Target function: $f : \mathcal{X} \to \mathcal{Y}$ *(ideal credit approval formula)*
- Data: $(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \cdots, (\mathbf{x}_N, y_N)$ *(historical records)*

$\downarrow \quad \downarrow \quad \downarrow$

- Hypothesis: $g : \mathcal{X} \to \mathcal{Y}$ *(formula to be used)*

Note: $\boldsymbol{x} \in \mathcal{X}$ and $y \in \mathcal{Y}$.

The learning process



The learning algorithm $\mathcal{A}$ picks $g \approx f$ from a hypothesis set $\mathcal{H}$ using the training examples (data).

The learning model

Solution components

The 2 solution components of the learning problem:

- The Hypothesis Set

  $\mathcal{H} = \{h\} \qquad g \in \mathcal{H}$

- The Learning Algorithm

Together, they are referred to as the *learning model*.

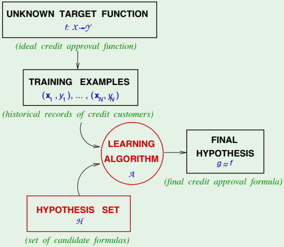


Table of contents

The perceptron learning model

Let $\mathcal{X} = \mathbb{R}^d$ be the input space, and let $\mathcal{Y} = \{+1, -1\}$ be the output space, denoting a binary (yes/no) decision.

To specify the perceptron **learning model**, we need to define:

1. The perceptron **hypothesis set**
2. The perceptron **learning algorithm**

For $s \in \mathbb{R}$, we define the sign function as

$$\text{sign}(s) = \begin{cases} -1 & \text{if } s < 0, \\ 1 & \text{if } s > 0. \end{cases}$$

Note: for the moment, sign(0) is ignored (technicality).

The perceptron hypothesis set

A simple hypothesis set - the 'perceptron'

For input $\mathbf{x} = (x_1, \cdots, x_d)$ 'attributes of a customer'

$$\text{Approve credit if} \quad \sum_{i=1}^{d} w_i x_i > \text{threshold},$$
$$\text{Deny credit if} \quad \sum_{i=1}^{d} w_i x_i < \text{threshold}.$$

This linear formula $h \in \mathcal{H}$ can be written as

$$h(\mathbf{x}) = \text{sign}\left(\left(\sum_{i=1}^{d} w_i x_i\right) - \text{threshold}\right)$$

The perceptron hypothesis set

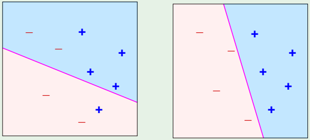
$$h(\mathbf{x}) = \text{sign}\left(\left(\sum_{i=1}^{d} w_i\, x_i\right) + \quad w_0 \quad \right)$$

Introduce an artificial coordinate $x_0 = 1$:

$$h(\mathbf{x}) = \text{sign}\left(\sum_{i=0}^{d} w_i\, x_i\right)$$

In vector form, the perceptron implements

$$h(\mathbf{x}) = \text{sign}(\mathbf{w}^\intercal \mathbf{x})$$



'linearly separable' data

Note

The L_2 norm of $\boldsymbol{z} = (z_0, z_1, \ldots, z_d)^T$ is given by

$$\|\boldsymbol{z}\|_2 = \sqrt{\sum_{i=0}^{d} z_i^2}.$$

The dot product of two vectors $\boldsymbol{w}$ and $\boldsymbol{x}$ is defined by

$$\boldsymbol{w} \cdot \boldsymbol{x} = \boldsymbol{w}^T \boldsymbol{x} = \|\boldsymbol{w}\|_2 \, \|\boldsymbol{x}\|_2 \cos(\theta),$$

where $\|\cdot\|_2$ is the L_2 norm, θ is the angle between $\boldsymbol{w}$ and $\boldsymbol{x}$.

- If the angle between $\boldsymbol{w}$ and $\boldsymbol{x}$ is <u>less than 90 degrees</u>, the dot product will be positive, as $\cos(\theta)$ will be positive.
- If the angle between $\boldsymbol{w}$ and $\boldsymbol{x}$ is <u>greater than 90 degrees</u>, the dot product will be negative, as $\cos(\theta)$ will be negative.
- If $\boldsymbol{w}$ and $\boldsymbol{x}$ are <u>perpendicular (at 90 degrees to each other)</u>, the result of the dot product will be zero, because $\cos(\theta)$ will be zero.

The perceptron learning algorithm

A simple learning algorithm - PLA

The perceptron implements

$$h(\mathbf{x}) = \text{sign}(\mathbf{w}^\intercal \mathbf{x})$$

Given the training set:

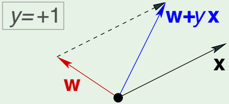
$$(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \cdots, (\mathbf{x}_N, y_N)$$

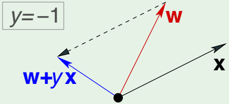
pick a misclassified point:

$$\text{sign}(\mathbf{w}^\intercal \mathbf{x}_n) \neq y_n$$

and update the weight vector:

$$\boxed{\mathbf{w} \leftarrow \mathbf{w} + y_n \mathbf{x}_n}$$





The perceptron learning algorithm

Data: $\{(\boldsymbol{x}_n, y_n)\}_{n=1}^{N}$
initialise weights at $t = 0$ to $\boldsymbol{w}(0)$;
for $t = 0, 1, 2, \ldots$ **do**
    select a misclassified point $(\boldsymbol{x}_n, y_n)$;
    update the weights: $\boldsymbol{w}(t+1) = \boldsymbol{w}(t) + y_n \boldsymbol{x}_n$;
    iterate to the next step until all points are well classified;
end
Return the final weights $\boldsymbol{w}(t+1)$.

PLA is guaranteed to converge if data is **linearly separable** (see labs).

Exercise I

Problem 1.2 Consider the perceptron in two dimensions: $h(\mathbf{x}) = \text{sign}(\mathbf{w}^{\mathrm{T}}\mathbf{x})$ where $\mathbf{w} = [w_0, w_1, w_2]^{\mathrm{T}}$ and $\mathbf{x} = [1, x_1, x_2]^{\mathrm{T}}$. Technically, $\mathbf{x}$ has three coordinates, but we call this perceptron two-dimensional because the first coordinate is fixed at 1.

(a) Show that the regions on the plane where $h(\mathbf{x}) = +1$ and $h(\mathbf{x}) = -1$ are separated by a line. If we express this line by the equation $x_2 = ax_1 + b$, what are the slope a and intercept b in terms of w_0, w_1, w_2?

(b) Draw a picture for the cases $\mathbf{w} = [1, 2, 3]^{\mathrm{T}}$ and $\mathbf{w} = -[1, 2, 3]^{\mathrm{T}}$.

In more than two dimensions, the $+1$ and -1 regions are separated by a *hyperplane*, the generalization of a line.

(Source: Abu-Mostafa et al. Learning from data. AMLbook)

Solution to Exercise I

(a) We have $\boldsymbol{w}^T\boldsymbol{x} > 0$ if $h(\boldsymbol{x}) = +1$, and $\boldsymbol{w}^T\boldsymbol{x} < 0$ if $h(\boldsymbol{x}) = -1$. These two regions are separated by the line $\boldsymbol{w}^T\boldsymbol{x} = 0$. This can also be written as $w_0 + w_1x_1 + w_2x_2 = 0$. In other words, if $w_2 \neq 0$, $a = -\frac{w_1}{w_2}$ and $b = -\frac{w_0}{w_2}$.

(b) See board.

Exercise II

Consider the following dataset
$\boldsymbol{x}_1 = (3, 1), \boldsymbol{x}_2 = (1, -3), \boldsymbol{x}_3 = (-1, 3), \boldsymbol{x}_4 = (2.5, -1)$ and
$y_1 = 1, y_2 = -1, y_3 = 1, y_4 = 1$.

1. Plot the data set in $[-1, 3] \times [-3, 3]$
2. Is the data linearly separable?
3. Run the perceptron algorithm with $\boldsymbol{w}(0) = (-3, 1, 1)^T$.

Exercise III

The weight update rule, at time step $t+1$,

$$\boldsymbol{w}(t+1) \leftarrow \boldsymbol{w}(t) + y(t)\boldsymbol{x}(t),$$

has the nice interpretation that it moves in the direction of classifying $\boldsymbol{x}(t)$ correctly.

1. Show that $y(t)\boldsymbol{w}^T(t)\boldsymbol{x}(t) < 0$.
   [Hint: $\boldsymbol{x}(t)$ is misclassified by $\boldsymbol{w}(t)$]
2. Show that $y(t)\boldsymbol{w}^T(t+1)\boldsymbol{x}(t) > y(t)\boldsymbol{w}^T(t)\boldsymbol{x}(t)$.
   [Hint: Use the update rule].

Solution to Exercise III

1. Let $s(t) = \boldsymbol{w}^T(t)\boldsymbol{x}(t)$. If $\boldsymbol{x}(t)$ is misclassified by $\boldsymbol{w}(t)$, then we have $\operatorname{sign}(s(t)) = +1$ and $y(t) = -1$, or $\operatorname{sign}(s(t)) = -1$ and $y(t) = +1$. In other words, we have

$$y(t)s(t) < 0 \equiv y(t)\boldsymbol{w}^T(t)\boldsymbol{x}(t) < 0.$$

2. We have

$$\begin{aligned} y(t)\boldsymbol{w}^T(t+1)\boldsymbol{x}(t) &= y(t)[\boldsymbol{w}^T(t) + y(t)\boldsymbol{x}^T(t)]\boldsymbol{x}(t) && (1)\\ &= y(t)\boldsymbol{w}^T(t)\boldsymbol{x}(t) + [y(t)]^2\boldsymbol{x}^T(t)\boldsymbol{x}(t) && (2)\\ &> y(t)\boldsymbol{w}^T(t)\boldsymbol{x}(t), && (3) \end{aligned}$$

since $[y(t)]^2 > 0$ and $\boldsymbol{x}^T(t)\boldsymbol{x}(t) > 0$ (since $x_0(t) = 1$).