Artificial Neural Networks

Lecture 1: Introduction

Dr Juan Neirotti

j.p.neirotti@aston.ac.uk

2024

What is this Module About?

- In the most general terms: Modeling stationary processes, i.e. probabilistic processes where the density distribution does not changes with time.
- In the not-that-much general terms: Pattern recognition (speech, face, hand-written characters).
- All these tasks need a statistics approach to help extract the relevant features (patterns) out of many instances (big data).

Definition

1. Artificial Intelligence (AI) is the area of knowledge that endeavors towards constructing systems (hardware or software) that behave *intelligently*. (Observe that I have not defined what I mean by intelligent just yet).
2. Machine Learning (ML) is a sub-area of AI that tackles problems by extracting the patterns that link questions with correct answers (provided to the AI system during the *training phase*). The ML paradigm differs from the traditional manner to solve problems on the fact that, in the traditional way the rules (patterns) that produce an answer given a question are supposed to be known, whereas in the ML paradigm such rules are unknown and need to be discovered.

Definition

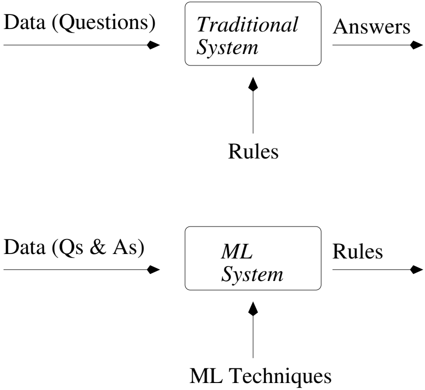


Figure: Different paradigms for solving problems

Definition

1. Statistical ML: The adjective *statistical* acknowledges the nature of the elements in the Data Set to be used $\mathcal{D}_u = \{\boldsymbol{x}_n\}_{n=1}^{M}$, which can be described through a probability distribution $\mathcal{P}(\boldsymbol{x})$, or $\mathcal{D}_s = \{(\boldsymbol{x}_n, t_n)\}_{n=1}^{M}$, which can be described through a probability distribution $\mathcal{P}(\boldsymbol{x}, t)$,
2. Most of the work in SML involves making models of the $\mathcal{P}(\boldsymbol{x})$ or $\mathcal{P}(\boldsymbol{x}, t)$ and produce results based on inferences using such a model (the ML techniques in the figure above).

Definition

1. If the type of data set we use to *train* (more on this later) our intelligent system is $\mathcal{D}_u = \{\boldsymbol{x}_n\}_{n=1}^{M}$ we say that the training process involves a *unsupervised learning*.
2. If the type of data set we use to *train* (more on this later) our intelligent system is $\mathcal{D}_s = \{(\boldsymbol{x}_n, t_n)\}_{n=1}^{M}$, we say that the training process involves a *supervised learning*.

Problems to be Tackled

1. Classification: Given a particular input $\boldsymbol{x} \in \mathscr{X}$ the intelligent system must produce a classification $t \in \mathscr{T} = \{-1, +1\}$. The map $t = t(\boldsymbol{x})$ is discrete.
2. Regression: Estimate the functional form of the map $t : \mathscr{X} \to \mathscr{T}$,i.e. $t = t(\boldsymbol{x})$, which is, in general, a continuous map.

Why Statistics

1. Both characters are drawn in a 256 x 256 pixels figure.
2. The total number of possible figures (in black and white) is $2^{256\times256} \sim 10^{20000}$. The size of the set with all the possible figures is huge.
3. No all the possible figures are meaningful. There are many (many indeed) figures that wouldn't carry any meaning at all.

Figure: Hand-written characters a and b.

Pattern extraction

1. We can define x_1 and x_2 as the horizontal and vertical dimensions of the characters.
2. a's and b's are of similar width, $x_1(a) \simeq x_2(b)$.
3. a's are expected to be shorter than b's, therefore $x_2(a) < x_2(b)$.
4. Both attributes are distributed variables $x_1(a) \sim \mathcal{P}_{1,a}$, $x_2(a) \sim \mathcal{P}_{2,a}$, $x_1(b) \sim \mathcal{P}_{1,b}$, $x_2(b) \sim \mathcal{P}_{2,b}$.

Pattern extraction

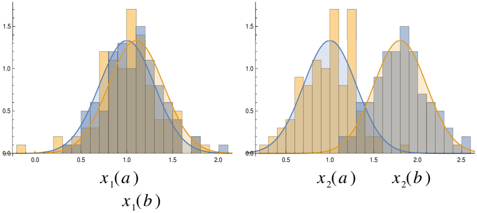
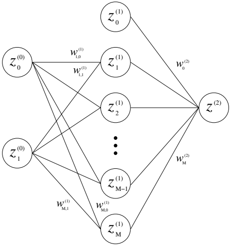


Figure: Histogram and correspondent distribution for x_1 (left) and x_2 (right). Observe that, even for the clear separation between x_2 distributions, there steel exists an *overlapping* range of values where no decision can be taken.

Maps

1. $t : \mathscr{X} \to \mathscr{T}$, where usually $\mathscr{X} \subset \mathbb{R}^d$
2. d is expected to be large.
3. $\mathscr{T}$ could be a discrete set (classification problem) or a continuous set (regression)
4. The *estimate* $\hat{t} = \hat{t}(\boldsymbol{x}, \boldsymbol{w})$, $\boldsymbol{x} \in \mathscr{X}$ is the input (or independent variable) and $\boldsymbol{w} \in \Omega$ are the parameters of the model.
5. Parameters ($\boldsymbol{w}$) and the functional form of the map ($\hat{t}(\boldsymbol{x}, \boldsymbol{w})$) are part of the model.
6. Neural networks are implementations of maps where the functional form t is fixed by the problem and the parameters $\boldsymbol{w}$ are *learn* from the available data.

Maps

1. 

$$z_k^{(\ell+1)} = \sigma\left(\left\langle z^{(\ell)} \,\middle|\, \boldsymbol{w}_k^{(\ell+1)} \right\rangle + w_{k,0}^{(\ell+1)}\right)$$

2. σ is the activation function that can be: sigmoidal, tanh, sgn, Heaviside, identity, ReLU, etc.
3. The activation variable is always linear:
$\left\langle z^{(\ell)} \,\middle|\, \boldsymbol{w}_k^{(\ell+1)} \right\rangle + w_{k,0}^{(\ell+1)} = \sum_{j=1}^{\dim(z^{(\ell)})} z_j^{(\ell)} w_{k,j}^{(\ell+1)} + w_{k,0}^{(\ell+1)}$
4. The number of layers ($0 \leq \ell \leq L$) and the type of units used determine the architecture of the network.

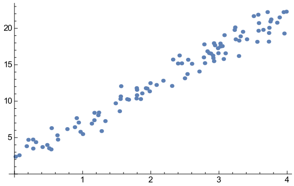
Learning the weights

1. Unsupervised Learning $\mathscr{D} = \{\boldsymbol{x}_\ell\}_{\ell=1}^N$ the objective is to estimate $\mathcal{P}_X(\boldsymbol{x}|\boldsymbol{w})$.
2. Supervised Learning $\mathscr{D} = \{(\boldsymbol{x}_\ell, t_\ell)\}_{\ell=1}^N$ the objective is to estimate the function $h(\boldsymbol{x}) = t$ through the map $\hat{t}(\boldsymbol{x}; \boldsymbol{w})$ (regression).

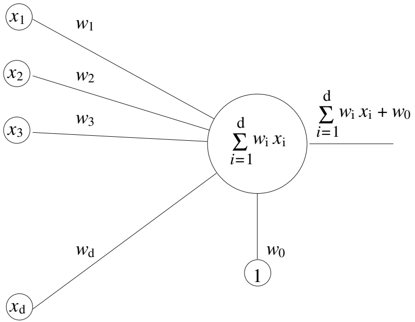
An Example: Regression with a linear function

- Let us suppose we have a set of data presented in the following form:

$$\mathcal{D} = \{(\boldsymbol{x}_\ell, t_\ell) : t_\ell \in \mathbb{R}, \boldsymbol{x}_\ell \in \mathbb{R}^d, \ell = 1, \ldots, L\}$$



- The architecture used for this problem is the following:



- Because the output is the identity function $\sigma(y) = y$ this is known as the linear perceptron.

- We represent the vectors $|\boldsymbol{x}\rangle, |\boldsymbol{z}\rangle \in \mathscr{X} \subseteq \mathbb{R}^d$ and their transposes $\langle\boldsymbol{x}|, \langle\boldsymbol{z}|$ using the bra-ket notation.
- We suppose $\mathscr{X}$ is a vector space over $\mathbb{R}$ or $\mathbb{C}$ with an inner product $\langle \boldsymbol{z} | \boldsymbol{x} \rangle$.
- The quadratic error (or loss) function, given the data set $\mathcal{D} = \{(|\boldsymbol{y}_m\rangle, t_m)\}_{m=1}^{L}$ (where $|\boldsymbol{y}\rangle = |1, \boldsymbol{x}\rangle$) is defined a

$$
\begin{aligned}
\mathcal{E}(|\boldsymbol{w}\rangle | \mathcal{D}) &= \frac{1}{L} \sum_{\ell=1}^{L} \mathcal{E}_\ell(|\boldsymbol{w}\rangle) \\
&= \frac{1}{2L} \sum_{\ell=1}^{L} (t_\ell - \langle \boldsymbol{y}_\ell | \boldsymbol{w} \rangle)^2 \\
&= \frac{1}{2L} \sum_{\ell=1}^{L} \left(t_\ell^2 - 2t_\ell \langle \boldsymbol{y}_\ell | \boldsymbol{w} \rangle + \langle \boldsymbol{w} | \boldsymbol{y}_\ell \rangle \langle \boldsymbol{y}_\ell | \boldsymbol{w} \rangle\right) \\
&= \frac{1}{2L} \sum_{\ell=1}^{L} t_\ell^2 - \left(\frac{1}{L} \sum_{\ell=1}^{L} t_\ell \langle \boldsymbol{y}_\ell|\right) |\boldsymbol{w}\rangle + \frac{1}{2} \left\langle \boldsymbol{w} \left| \frac{1}{L} \sum_{\ell=1}^{L} |\boldsymbol{y}_\ell\rangle \langle \boldsymbol{y}_\ell| \right| \boldsymbol{w} \right\rangle .
\end{aligned}
$$

- We can identify three terms in the last expression, each term has a component that only depends on the elements of the data set:

$$\mathbb{R}^+ \ni C = \frac{1}{2L}\sum_{\ell=1}^{L} t_\ell^2$$

$$\{1\} \times \mathscr{X} \ni \langle \boldsymbol{b}| = \frac{1}{L}\sum_{\ell=1}^{L} t_\ell \langle \boldsymbol{y}_\ell|$$

$$\mathscr{X}^{(d+1)\times(d+1)} \ni \boldsymbol{G} = \frac{1}{L}\sum_{\ell=1}^{L} |\boldsymbol{y}_\ell\rangle \langle \boldsymbol{y}_\ell| \,.$$

- Therefore

$$\mathcal{E}(|\boldsymbol{w}\rangle\, |\mathcal{D}) = C - \langle \boldsymbol{b}\,|\boldsymbol{w}\,\rangle + \frac{1}{2}\langle \boldsymbol{w}\,|\boldsymbol{G}|\,\boldsymbol{w}\rangle \,.$$

- Observe that $\boldsymbol{G}$ is a symmetric matrix and its eigenvectors and eigenvalues satisfy the following properties:

$$\begin{aligned}
\boldsymbol{G}\,|\boldsymbol{\mu}\rangle &= \mu\,|\boldsymbol{\mu}\rangle \\
\langle\boldsymbol{\mu}|\,\boldsymbol{G} &= \mu\,\langle\boldsymbol{\mu}| \\
\left\langle\boldsymbol{\mu}'\right|\boldsymbol{G}\,|\boldsymbol{\mu}\rangle &= \mu\left\langle\boldsymbol{\mu}'\,|\boldsymbol{\mu}\right\rangle = \mu'\left\langle\boldsymbol{\mu}'\,|\boldsymbol{\mu}\right\rangle \\
(\mu-\mu')\left\langle\boldsymbol{\mu}'\,|\boldsymbol{\mu}\right\rangle &= 0
\end{aligned}$$

  which implies that $\langle\boldsymbol{\mu}'\,|\boldsymbol{\mu}\rangle = 0$ for all $\mu \neq \mu'$.
- Also:

$$\begin{aligned}
\mu\,\langle\boldsymbol{\mu}\,|\boldsymbol{\mu}\rangle &= \langle\boldsymbol{\mu}|\,\boldsymbol{G}\,|\boldsymbol{\mu}\rangle \\
&= \frac{1}{L}\sum_{\ell=1}^{L}\langle\boldsymbol{\mu}\,|\boldsymbol{y}_\ell\rangle\,\langle\boldsymbol{y}_\ell\,|\boldsymbol{\mu}\rangle \\
&= \frac{1}{L}\sum_{\ell=1}^{L}\langle\boldsymbol{\mu}\,|\boldsymbol{y}_\ell\rangle^2 \geq 0
\end{aligned}$$

  thus $\mu \geq 0$.

Optimization

- Let us compute the gradient of the loss function:

$$\nabla_{\boldsymbol{w}} \mathcal{E}(|\boldsymbol{w}\rangle \,|\mathcal{D}) = -\,|\boldsymbol{b}\rangle + \boldsymbol{G}\,|\boldsymbol{w}\rangle \,.$$

- The analytical solution of the problem $\nabla_{\boldsymbol{w}} \mathcal{E}(|\boldsymbol{w}\rangle \,|\mathcal{D}) = \mathbf{0}_{d+1}$ is found if $\boldsymbol{G}$ admits an inverse (which is true if all its eigenvalues are positive):

$$|\boldsymbol{w}^{\star}\rangle = \boldsymbol{G}^{-1}\,|\boldsymbol{b}\rangle \,.$$

- This solution is not achievable if the condition number is too big $\frac{\mu_{\mathsf{max}}}{\mu_{\mathsf{min}}} \gg 1$, where $\mu_{\mathsf{max(min)}}$ is the largest (smallest) eigenvalue of $\boldsymbol{G}$.

Optimization

- Observation. Assume that the formal expression $|\boldsymbol{w}^{\star}\rangle = \boldsymbol{G}^{-1}|\boldsymbol{b}\rangle$ is given (not computed). Then:

$$
\begin{aligned}
\mathcal{E}(|\boldsymbol{w}\rangle|\mathcal{D}) &= C - \langle \boldsymbol{b}|\boldsymbol{w}-\boldsymbol{w}^{\star}+\boldsymbol{w}^{\star}\rangle + \frac{1}{2}\langle \boldsymbol{w}-\boldsymbol{w}^{\star}+\boldsymbol{w}^{\star}|\boldsymbol{G}|\boldsymbol{w}-\boldsymbol{w}^{\star}+\boldsymbol{w}^{\star}\rangle \\
&= C - \langle \boldsymbol{b}|\boldsymbol{w}-\boldsymbol{w}^{\star}\rangle - \langle \boldsymbol{b}|\boldsymbol{w}^{\star}\rangle + \langle \boldsymbol{w}-\boldsymbol{w}^{\star}|\boldsymbol{G}|\boldsymbol{w}^{\star}\rangle + \\
&+ \frac{1}{2}\langle \boldsymbol{w}-\boldsymbol{w}^{\star}|\boldsymbol{G}|\boldsymbol{w}-\boldsymbol{w}^{\star}\rangle \\
&= C - \langle \boldsymbol{b}|\boldsymbol{w}^{\star}\rangle + \frac{1}{2}\langle \boldsymbol{w}-\boldsymbol{w}^{\star}|\boldsymbol{G}|\boldsymbol{w}-\boldsymbol{w}^{\star}\rangle \\
&= \mathcal{E}(|\boldsymbol{w}^{\star}\rangle|\mathcal{D}) + \frac{1}{2}\langle \boldsymbol{w}-\boldsymbol{w}^{\star}|\boldsymbol{G}|\boldsymbol{w}-\boldsymbol{w}^{\star}\rangle.
\end{aligned}
$$

where $\mathcal{E}(|\boldsymbol{w}^{\star}\rangle|\mathcal{D})$ is the residual error of the optimal model.

Gradient Descent Method

- Let us propose the following iterative method to estimate the solution ($|\boldsymbol{w}^{\star}\rangle$) of our problem:

$$|\boldsymbol{w}_{k+1}\rangle = |\boldsymbol{w}_k\rangle - \alpha_k |\boldsymbol{g}_k\rangle$$

where

$$\begin{aligned} |\boldsymbol{g}_k\rangle &= \nabla_{\boldsymbol{w}}\mathcal{E}\left(|\boldsymbol{w}_k\rangle \, |\mathcal{D}\right) = -|\boldsymbol{b}\rangle + \boldsymbol{G}\,|\boldsymbol{w}_k\rangle \\ \alpha_k &\geq 0 \end{aligned}$$

are the gradient and the learning rate (or step-size) respectively.

- We want to find $\alpha \geq 0$ that improves the estimates of the solution, i.e. $\mathcal{E}\left(|\boldsymbol{w}_{k+1}\rangle \,|\mathcal{D}\right) \leq \mathcal{E}\left(|\boldsymbol{w}_k\rangle \,|\mathcal{D}\right)$.
- Let us define $\phi(\alpha) = \mathcal{E}\left(|\boldsymbol{w}_k\rangle - \alpha\,|\boldsymbol{g}_k\rangle \,|\mathcal{D}\right)$, thus

$$
\begin{aligned}
\phi(\alpha) = C - \langle \boldsymbol{b}| \left(|\boldsymbol{w}_k\rangle - \alpha\,|\boldsymbol{g}_k\rangle\right) + \\
+ \frac{1}{2}\left(\langle \boldsymbol{w}_k| - \alpha\,\langle \boldsymbol{g}_k|\right) \boldsymbol{G} \left(|\boldsymbol{w}_k\rangle - \alpha\,|\boldsymbol{g}_k\rangle\right) \\
= \mathcal{E}\left(|\boldsymbol{w}_k\rangle \,|\mathcal{D}\right) - \alpha\,\langle \boldsymbol{g}_k| \left(|\boldsymbol{b}\rangle - \boldsymbol{G}\,|\boldsymbol{w}_k\rangle\right) + \frac{\alpha^2}{2}\,\langle \boldsymbol{g}_k|\,\boldsymbol{G}\,|\boldsymbol{g}_k\rangle \\
= \phi(0) - \alpha\,\langle \boldsymbol{g}_k|\boldsymbol{g}_k\rangle + \frac{\alpha^2}{2}\,\langle \boldsymbol{g}_k|\,\boldsymbol{G}\,|\boldsymbol{g}_k\rangle
\end{aligned}
$$

.

- If
$$\frac{2\left\langle \boldsymbol{g}_k \middle| \boldsymbol{g}_k \right\rangle}{\left\langle \boldsymbol{g}_k \right| \boldsymbol{G} \left| \boldsymbol{g}_k \right\rangle} > \alpha > 0$$
then the error function decreases.
- Rayleigh's inequality:
$$\mu_{\text{min}} \left\langle \boldsymbol{\mu}_{\text{min}} \middle| \boldsymbol{\mu}_{\text{min}} \right\rangle \le \left\langle \boldsymbol{g}_k \right| \boldsymbol{G} \left| \boldsymbol{g}_k \right\rangle \le \mu_{\text{max}} \left\langle \boldsymbol{\mu}_{\text{max}} \middle| \boldsymbol{\mu}_{\text{max}} \right\rangle$$
thus
$$\frac{2}{\mu_{\text{max}}} > \alpha > 0$$
is the condition to ensure convergence.

.

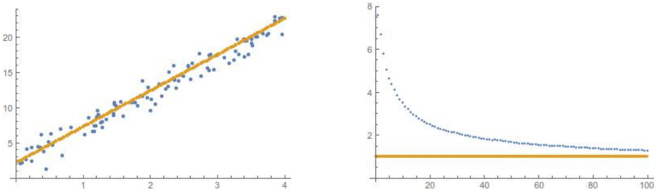
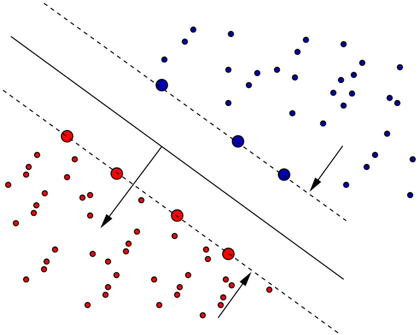


Figure: Linear regression of data points generated by the function $y(x) = 5x + 2 + \varepsilon$, where ε is a random Gaussian noise with zero mean and unit variance (left) and error $\sqrt{\frac{1}{L}\sum_{\ell=1}^{L}(y_\ell - w_0 - w_1 x_\ell)^2}$ vs iterations (right). Observe that the error converges to the noise's standard deviation. The eigenvalues of $\frac{1}{L}\sum_{\ell=1}^{L} \mathbf{Y}_\ell \mathbf{Y}_\ell^{\mathrm{T}}$ are $\mu_{\max} = 6.603 \pm 0.001$ and $\mu_{\min} = 0.196 \pm 0.001$. Thus $\eta \approx 0.03 \pm 0.01$.

Second Example: Classification with a binary perceptron

- Let us suppose we have a set of data presented in the following form:

$$\mathcal{D} = \{(t_\ell; \mathbf{x}_\ell) : t_\ell \in \{-1, +1\}, \mathbf{x}_\ell \in \mathbb{R}^d, \ell = 1, \dots, L\}.$$



Linear separability

Perceptron Learning

- The choice of the function $\Theta(x)$, where $\Theta(x > 0) = 1$ and 0 otherwise, is well suited for the classification problem.
- Suppose we have a set $\mathcal{D} = \{(t_\ell; \boldsymbol{x}_\ell) : t_\ell \in \{-1, +1\}, \boldsymbol{x}_\ell \in \mathbb{R}^d, \ell = 1, \dots, L\}$, thus there are two classes of vectors $\boldsymbol{x}$, those which carry the label $t = +1$ and those with $t = -1$.
- The error is defined as the total number of elements in $\mathcal{D}$ that have been wrongly classified by $\boldsymbol{w}$:

$$E(\boldsymbol{w}) = \sum_{\ell=1}^{L} \Theta\left(-t_\ell \left\langle \boldsymbol{x}_\ell \middle| \boldsymbol{w} \right\rangle\right)$$

- Suppose that there exists $\boldsymbol{w}^\star \in \mathbb{R}^d$ such that $\operatorname{sgn}\left(t_\ell \left\langle \boldsymbol{x}_\ell \middle| \boldsymbol{w} \right\rangle\right) = 1$ for all $1 \le \ell \le L$.

Update:

$$\left|\boldsymbol{w}_{\ell+1}\right\rangle = \left|\boldsymbol{w}_\ell\right\rangle + \eta\, \Theta\left(-t_\ell \left\langle \boldsymbol{x}_\ell \middle| \boldsymbol{w}_\ell \right\rangle\right) t_\ell \left|\boldsymbol{x}_\ell\right\rangle . \tag{1}$$

Perceptron Convergence Theorem

- Theorem: For any linearly separable set $\mathcal{D} = \{(t_\ell; \boldsymbol{x}_\ell) : t_\ell \in \{-1, +1\}, \boldsymbol{x}_\ell \in \mathbb{R}^d, \ell = 1, \ldots, L\}$, the learning rule (1) is guaranteed to find a solution in a finite number of steps.
- Proof: Suppose τ is the total number of real updates

$$
\begin{aligned}
|\boldsymbol{w}_{\ell+1}\rangle &= |\boldsymbol{w}_\ell\rangle + \eta\, \Theta\left(-t_\ell \langle \boldsymbol{x}_\ell | \boldsymbol{w}_\ell \rangle\right) t_\ell |\boldsymbol{x}_\ell\rangle \\
&\vdots \\
&= |\boldsymbol{w}_0\rangle + \eta \sum_{j=1}^{\ell} \Theta\left(-t_j \langle \boldsymbol{x}_j | \boldsymbol{w}_j \rangle\right) t_j |\boldsymbol{x}_j\rangle \\
&= \eta \sum_{j=1}^{L} \tau_j t_j |\boldsymbol{x}_j\rangle
\end{aligned}
$$

where $0 \leq \tau_j$ is the number of times the vector $\boldsymbol{x}_j$ has been misclassified and $|\boldsymbol{w}_0\rangle = |\boldsymbol{0}\rangle$.

.

- Then

$$\langle \boldsymbol{w}^\star \, | \boldsymbol{w}_{\ell+1} \rangle = \eta \sum_{j=1}^{L} \tau_j t_j \, \langle \boldsymbol{w}^\star \, | \boldsymbol{x}_j \rangle$$

where $t_j \langle \boldsymbol{w}^\star \, | \boldsymbol{x}_j \rangle > 0$ because $|\boldsymbol{w}^\star\rangle$ is the solution to the problem (and, therefore, classifies correctly all examples from the data set $\mathcal{D}$).

- Then

$$\langle \boldsymbol{w}^\star \, | \boldsymbol{w}_{\ell+1} \rangle \geq \eta\tau \min_{(\boldsymbol{x}_j, t_j) \in \mathcal{D}} \left\{ t_j \, \langle \boldsymbol{w}^\star \, | \boldsymbol{x}_j \rangle \right\},$$

where $\tau = \sum_{j=1}^{L} \tau_j$.

- Thus

$$\| \boldsymbol{w}_{\ell+1} \|_2 \geq \eta\tau \min_{(\boldsymbol{x}_j, t_j) \in \mathcal{D}} \left\{ t_j \, \langle \boldsymbol{w}^\star \, | \boldsymbol{x}_j \rangle \right\}$$

- Also

$$\langle \boldsymbol{w}_{\ell+1} | \boldsymbol{w}_{\ell+1} \rangle = \langle \boldsymbol{w}_{\ell} | \boldsymbol{w}_{\ell} \rangle + \eta^2 \Theta\left(-t_\ell \langle \boldsymbol{x}_\ell | \boldsymbol{w}_\ell \rangle\right) \langle \boldsymbol{x}_\ell | \boldsymbol{x}_\ell \rangle + \\ + 2\eta \Theta\left(-t_\ell \langle \boldsymbol{x}_\ell | \boldsymbol{w}_\ell \rangle\right) t_\ell \langle \boldsymbol{w}_\ell | \boldsymbol{x}_\ell \rangle ,$$

  and observe that the last term is always less than zero.
- Thus

$$\begin{aligned} \langle \boldsymbol{w}_{\ell+1} | \boldsymbol{w}_{\ell+1} \rangle &\leq \langle \boldsymbol{w}_{\ell} | \boldsymbol{w}_{\ell} \rangle + \eta^2 \Theta\left(-t_\ell \langle \boldsymbol{x}_\ell | \boldsymbol{w}_\ell \rangle\right) \langle \boldsymbol{x}_\ell | \boldsymbol{x}_\ell \rangle \\ &\leq \langle \boldsymbol{w}_0 | \boldsymbol{w}_0 \rangle + \eta^2 \sum_{j=1}^{\ell} \Theta\left(-t_j \langle \boldsymbol{x}_j | \boldsymbol{w}_j \rangle\right) \langle \boldsymbol{x}_j | \boldsymbol{x}_j \rangle \\ &\leq \eta^2 \tau \max_{\boldsymbol{x}_j \in \mathcal{D}} \left\{ \langle \boldsymbol{x}_j | \boldsymbol{x}_j \rangle \right\} . \end{aligned}$$

- Thus
$$\|\boldsymbol{w}_{\ell+1}\|_2 \leq \eta \sqrt{\tau \max_{\boldsymbol{x}_j \in \mathcal{D}} \{\langle \boldsymbol{x}_j \,|\boldsymbol{x}_j \rangle\}}.$$
- Both conditions are satisfied if
$$0 \leq \tau \leq \frac{\max_{\boldsymbol{x}_j \in \mathcal{D}} \{\langle \boldsymbol{x}_j \,|\boldsymbol{x}_j \rangle\}}{\left(\min_{(\boldsymbol{x}_j, t_j) \in \mathcal{D}} \{t_j \langle \boldsymbol{w}^\star \,|\boldsymbol{x}_j \rangle\}\right)^2} < \infty.$$
- That indicates that there is a maximum number of mistakes the algorithm could make.