Formulaire

NICOLAS LE HIR

TABLE DES MATIÈRES

PRESENTATION

This short document if an informal memo for Epitech students.

1 QUANTIFIERS

In order to write some mathematical expressions, **quantifiers** are often used :

— https://en.wikipedia.org/wiki/Quantifier_(logic)

— https://fr.wikipedia.org/wiki/Quantification_(logique)

For instance :

$$\exists x \in \mathbb{R}, x^2 = 4 \tag{1}$$

means : "There exists a real number x, such that $x^2 = 4$".

$$\forall x \in \mathbb{R}, -x^2 \leqslant 0 \tag{2}$$

means : "For all real number x, $-x^2 \leqslant 0$"

2 ALGORITHMIC COMPLEXITY

Let n be the size of the problem (number of samples in the dataset, number of dimensions, number of integers to sort, ...), and $\mathcal{A}$ an algorithm that processes the problem.

Definition 1. Polynomial time complexity
We say that the algorithm $\mathcal{A}$ has **polynomial** time-complexity if the number of elementary operations $N(n)$ (sums, products, accessing an element in an array, etc.) required for $\mathcal{A}$ to terminate is smaller than a polynomial function of n. Formally, there exsits a fixed integer or float k, and a real number A, such that :

$$\forall n \in \mathbb{N}, N(n) \leqslant A \times n^k \tag{3}$$

The **Landau notation** is often used : $N(n) = \mathcal{O}(n^k)$.
Example : sorting a list of size n.

Definition 2. Exponential complexity
We say that $\mathcal{A}$ has an **exponential** complexity if there exists $k > 1$, and $B \in \mathbb{R}$, such that

$$\forall n \in \mathbb{N}, N(n) \leqslant B \times k^n \tag{4}$$

Similarly, we would write $N(n) = \mathcal{O}(k^n)$.
Example : enumerating the subsets of a set of size n.

3 DISTANCES

Here are some common distances in $\mathbb{R}^2$ and $\mathbb{R}^3$.

3.1 Distances in two dimensions

We consider two points a_1 and a_2 in the 2D space $\mathbb{R}^2$ with coordinates (x_1, y_1), and (x_2, y_2), respectively. Some common distances between a_1 and a_2 are :

L2 distance :

$$d(a_1, a_2) = ||a_1 - a_2||_2 = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \tag{5}$$

L1 distance :

$$d(a_1, a_2) = ||a_1 - a_2||_1 = |x_1 - x_2| + |y_1 - y_2| \tag{6}$$

L∞ distance :

$$d(a_1, a_2) = ||a_1 - a_2||_\infty = \max(|x_1 - x_2|, |y_1 - y_2|) \tag{7}$$

weighted L1 distance : :
let α_1 and α_2 be real, strictly non-negative numbers ($\in \mathbb{R}_+^*$).

$$d(a_1, a_2) = \alpha_1|x_1 - x_2| + \alpha_2|y_1 - y_2| \tag{8}$$

3.2 Distances in three dimensions

We consider two points a_1 and a_2 in the 3D space $\mathbb{R}^3$ with coordinates (x_1, y_1, z_1), and (x_2, y_2, z_2), respectively. Some common distances between a_1 and a_2 are :

L2 distance :

$$d(a_1, a_2) = \|a_1 - a_2\|_2 = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2} \tag{9}$$

L1 distance :

$$d(a_1, a_2) = \|a_1 - a_2\|_1 = |x_1 - x_2| + |y_1 - y_2| + |z_1 - z_2| \tag{10}$$

L∞ distance :

$$d(a_1, a_2) = \|a_1 - a_2\|_\infty = \max(|x_1 - x_2|, |y_1 - y_2|, |z_1 - z_2|) \tag{11}$$

weighted L1 :
let α_1, α_2 and α_3 be real, strictly non-negative numbers ($\in \mathbb{R}_+^*$).

$$d(a_1, a_2) = \alpha_1|x_1 - x_2| + \alpha_2|y_1 - y_2| + \alpha_3|z_1 - z_2| \tag{12}$$

3.3 Distances in d dimensions

$a_1 = (x_1, ..., x_p)$ and $a_2 = (y_1, ..., y_p)$ are now p-dimensional **vectors**.
L2 distance :

$$d(a_1, a_2) = \|a_1 - a_2\|_2 = \sqrt{\sum_{k=1}^{p} (x_k - y_k)^2} \tag{13}$$

L1 distance :

$$d(a_1, a_2) = \|a_1 - a_2\|_1 = \sum_{k=1}^{p} |x_k - y_k| \tag{14}$$

L∞ distance :

$$d(a_1, a_2) = \|a_1 - a_2\|_\infty = \max(x_1, \ldots, x_n) \tag{15}$$

weighted L1 distance : :

$$\sum_{k=1}^{p} \alpha_k |x_k - y_k| \tag{16}$$

with the $\alpha_k, k \in [1, p]$ being p real, strictly non-negative numbers.

4 LIKELIHOOD / VRAISEMBLANCE

We define the **likelihood** of a **parametric model.**

- Observations : $(x_1, \ldots, x_n)$
- Model : p (for instance a normal law)
- Parameters : θ (for instance (μ, σ), the mean and the standard deviation of the normal law).

The likelihood writes :

$$L(\theta) = p(x_1, \ldots, x_n | \theta) \tag{17}$$

5 DERIVATIVE / DÉRIVÉE

Let $f : \mathbb{R} \to \mathbb{R}$ be a real.
If it exists, the **derivative** of f in x is defined by :

$$f'(x) = \lim_{h \to 0} \frac{f(x+h) - f(x)}{h} \tag{18}$$

Examples :

If $g : x \mapsto 3x$, then the derivative exists and $\forall x \in \mathbb{R}, x, g'(x) = 3$

If $h : x \mapsto x^2$, then the derivative exists and $\forall x \in \mathbb{R}, h'(x) = 2x$.

If $h : x \mapsto |x|$, then the derivative exists only if $x \neq 0$.

6 EXPECTED VALUE / ÉSPÉRANCE

Let Z be a real random variable (https://en.wikipedia.org/wiki/Random_variable).
If it is correctly defined, the **expected value** of Z is defined in the following way.

— If Z is a discrete random variable, that takes the values $\{z_i, i \in \mathbb{N}\}$ with probability $P(Z = z_i)$.

$$E[Z] = \sum_{i=1}^{+\infty} z_i P(Z = z_i) \tag{19}$$

— If Z is a continuous random variable with density p :

$$E[Z] = \int_{-\infty}^{+\infty} z p(z) dz \tag{20}$$

Note that not all random variables do have an expected value.

6.1 Examples

6.1.1 Constant random variable

If Z is a constant random variable : $Z = \alpha \in \mathbb{R}$ with probability 1, $E[Z] = \alpha$.

6.1.2 Dice game

Z represents the outcome of a dice throw :
If the dice is unbiased :

$$E[Z] = \frac{1}{6}[1+2+3+4+5+6] = 3.5 \tag{21}$$

If the dice is cheated, for instance :

$$E[Z] = \frac{1}{100}(1+2+3+4) + \frac{48}{100}(5+6) = 5.38 \tag{22}$$

7 ENTROPY

Definition 3. Shannon entropy
The **Shannon entropy** of a discrete random variable X that takes the values x_i with probability p_i is given by :

$$H(X) = -\sum_{i=1}^{n} p_i \log(p_i) \tag{23}$$

Examples :

— Entropy of certain distribution $H = 0$.

— Entropy of uniform distribution with n values :

$$\begin{aligned} H &= -\sum_{i=1}^{n} \frac{1}{n} \log \frac{1}{n} \\ &= -n \times \frac{1}{n} \times \log \frac{1}{n} \\ &= \log n \end{aligned} \tag{24}$$

8 BINARY DECOMPOSITION ALGORITHM

Result: Integer n in binary form
$L \leftarrow$ liste vide [];
$r \leftarrow 0$;
while $n > 0$ **do**
  $r \leftarrow n\%2$;
  $L \leftarrow L + [r]$;
  $n \leftarrow (n - r)/2$;
end
$L \leftarrow reversed(L)$;
return L

Algorithm 1: Binary decomposition of integer n

9 DESIGN MATRIX FOR MACHINE LEARNING

In machine learning, we often work with a dataset D_n of n samples, each sample being a vector with d **features**, hence in $\mathbb{R}^d$. One sample x might be written as a line vector, like so :

$$x = \begin{pmatrix} x_1 \\ \dots \\ x_i \\ \dots \\ x_d \end{pmatrix} \tag{25}$$

This vector can also be transposed, and written as a line vector :

$$x^T = (x_1, \dots, x_d) \tag{26}$$

The full dataset, with n samples $x_i, i \in [1, n]$ is often stored in the **design matrix** $X \in \mathbb{R}^{n \times d}$.

$$X = \begin{pmatrix} x_1^T \\ \dots \\ x_i^T \\ \dots \\ x_n^T \end{pmatrix} = \begin{pmatrix} x_{11}, \dots, x_{1j}, \dots x_{1d} \\ \dots \\ x_{i1}, \dots, x_{ij}, \dots x_{id} \\ \dots \\ x_{n1}, \dots, x_{nj}, \dots x_{nd} \end{pmatrix} \tag{27}$$