Notes sur la classification par les k plus proches voisins

Maximilien

June 2020

Notions MIASHS Espace vectoriels, norme, distance, espace normé, espace métrique, algorithme de recherche.

1 Introduction

Intuitivement, on cherche une fonction $h : \mathcal{X} \mapsto \mathcal{Y}$ où $\mathcal{X} \subset \mathbb{R}^n$ représente par exemple l'ensemble des photos de chiens et de chats et $\mathcal{Y} = \{0, 1\}$ avec $0 =$ chien et $1 =$ chat. La stratégie la plus naïve permettant d'atteindre cet objectif est de partir d'un jeu de données $\mathcal{D} = \{(\boldsymbol{x_i}, y_i)\}_{i \leq N}$ sur $\mathcal{X} \times \mathcal{Y}$ et de se dire que deux objets appartenant à la même classe se ressemblent probablement. Il s'agit d'une méthode non-paramétrique de classification où la classe d'un nouvel échantillon $\boldsymbol{x}$ dépendra de celle de ses k voisins. On peut voir ici l'aspect non-paramétrique dans le sens où le modèle ne possède pas de paramètre mais se contente de conserver toutes les données. Le choix de ce nombre $k > 0$ permettra de jouer sur la régularité de la frontière de décision comme cela est présenté par la figure 1.

Pour cela, nous allons tout d'abord munir $\mathbb{R}^n$ d'une distance :

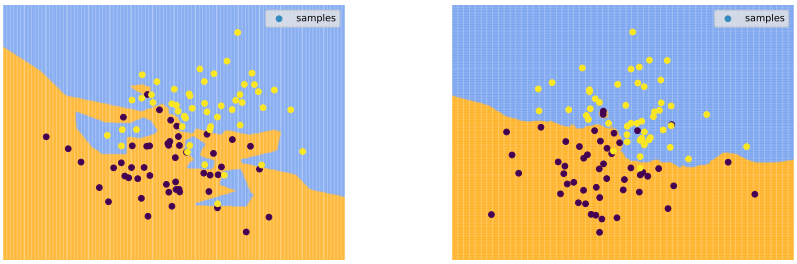
$$d : \mathbb{R}^n \times \mathbb{R}^n \mapsto \mathbb{R}^+.$$

Une distance fréquemment et traditionnellement utilisée est celle issue de la norme dite Euclidienne ℓ_2. Cette dernière se définit comme suit :

$$\|\cdot\|_2 : \boldsymbol{x} \mapsto \sqrt{\sum_{i=1}^{n} e_i^\star(\boldsymbol{x})^2} = \sqrt{\langle \boldsymbol{x}, \boldsymbol{x} \rangle},$$

où $\{e_i^\star\}_{i \leq n}$ est la base duale de $\mathbb{R}^n$. La distance issue de cette dernière est ainsi :

$$\begin{aligned} d : \mathbb{R}^n \times \mathbb{R}^n &\mapsto \mathbb{R}^+ \\ \boldsymbol{x}, \boldsymbol{y} &\to \|\boldsymbol{x} - \boldsymbol{y}\|_2 \end{aligned}$$



(a) (b)

FIGURE 1: Une classification par les k voisins avec $k = 1$ (a) et $k = 10$ (b).

Exercice ? On vérifie facilement que si $\|\cdot\|_2$ est bien une norme, alors d est une distance.

Dernière étape, l'agrégation des prédictions. Étant donné un nouvel échantillon $\boldsymbol{x}$ et ses k voisins $\text{nei}(\boldsymbol{x})$, il convient d'agréger leur label afin de réaliser une prédiction sur $\boldsymbol{x}$. Lorsque $k = 1$, la réponse est triviale. De plus, on peut remarquer que l'algorithme réalise un pavage de Voronoï (ou partition de Voronoï) tel qu'illustré par la figure 2 où chaque région de Voronoï est associée au label de l'échantillon correspondant. À l'inverse, lorsque $k > 1$, d'autres stratégies doivent être mises en place. Le vote à la majorité est l'une d'entre-elles. La classe prédite est celle qui est majoritaire parmi les voisins.

2 Construction de l'algorithme

Nous allons ici proposer des algorithmes afin de prédire le label associé à un nouvel échantillon $\boldsymbol{x}$. Les algorithmes construits ici vont réaliser une recherche exhaustive des voisins. En pratique des structures de données, appelées index voire des stratégies de hashing comme LSH [1] sont utilisées afin de calculer un voisinage efficacement. À noter que les approches de type LSH sont des heuristiques.

Le cas de figure où un unique voisin est utilisé afin de réaliser la prédiction est introduit par l'algorithme 2.

1. `https://en.wikipedia.org/wiki/Locality-sensitive_hashing`.

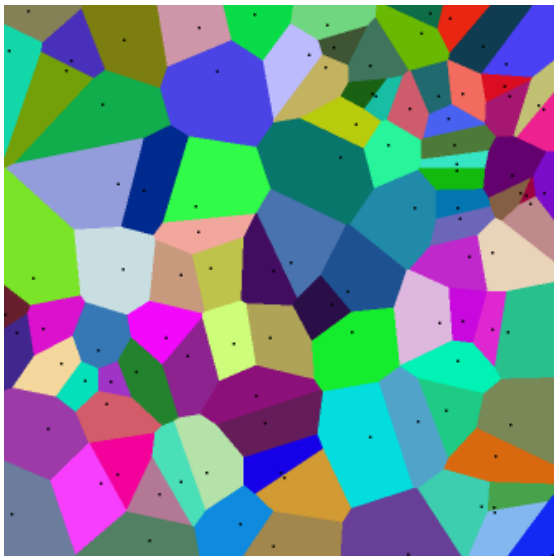


FIGURE 2: Un diagramme de Voronoï

Algorithm 1: La classification avec le plus proche voisin.

input : Un nouvel échantillon $\boldsymbol{x}$.
output: La classe y prédite pour $\boldsymbol{x}$.
distance_min = ∞;
closest = -1;
for $i \leftarrow 1$ **to** N **do**
  if $d(\boldsymbol{x_i}, \boldsymbol{x}) <$*distance_ min* **then**
    distance_min= $d(\boldsymbol{x_i}, \boldsymbol{x})$;
    closest=i;
  end
end
output=y_{closest};

Le cas où $k > 1$ est plus compliqué et oblige de maintenir une liste ordonnée. Cette liste contiendra à tout moments les au plus k points les plus proches de notre nouveau point $\boldsymbol{x}$.

Algorithm 2: La classification avec les k plus proches voisins.

```
input  : Un nouvel échantillon x.
output: La classe y prédite pour x.
resultats=ordered_list();
for i ← 1 to N do
    if length(resultats)< k or d(x_i, x) <last(resultats).distance then
        insert(resultats, (distance=d(x_i, x), pred=y_i));
        if length(resultats)> k then
            remove_last(resultats);
        end
    end
end
output=vote(resultats);
```

3 Les limites en grande dimension

L'algorithme de classification par les k plus proches voisins s'appuie sur la notion de distance entre nos échantillons. Cette dernière souffre malheureusement de ce qu'on appelle la "malédiction de la dimension". Cela se traduit intuitivement par l'idée qu'en "grande dimension", tous les points sont distants les uns des autres. La proximité d'un point avec l'autre est plus liée aux fluctuations aléatoires affectant les vecteurs qu'à l'information déterministe qu'ils contiennent. Une photo de chien jouant dans l'herbe a en réalité toutes les chances de ressembler à une photo de chat jouant dans l'herbe plutôt qu'à celle d'un autre chien. Cette idée est poussée à l'extrême en grande dimension.

Nous allons illustrer cette idée au travers d'un exemple simple sur lequel nous pourrons jouer avec la taille de notre jeu de données $|\mathcal{D}|$ ainsi que sur le nombre de dimensions qu'on considère.

Un exemple paradigmatique. Soit $\mathcal{B}_f(\mathbf{0}, r) \subset \mathbb{R}^d$ la boule fermée de centre $\mathbf{0}$ et de rayon r. Sans aucune perte de généralité, supposons $r = 1$. Sans rentrer dans des considérations liées à la mesure de Lebesgue, notons

$$\mathrm{Vol}(\mathcal{B}_f(\mathbf{0}, r)) \propto r^d$$

le volume de la boule. Enfin, supposons que nous disposions d'un jeu de données $\mathcal{D}$ distribué uniformément (selon la mesure de Lebesgue qu'on normalisera au besoin) dans $\mathcal{B}_1 = \mathcal{B}_f(\mathbf{0}, 1)$.

Nécessairement $\forall \boldsymbol{x} \in \mathcal{D}, \boldsymbol{x} \in \mathcal{B}_1$ et $d(\boldsymbol{x}, \mathbf{0}) \leq 1$. Autrement dit la distance maximale d'un point dans la boule au centre de la boule est 1. La question qui nous vient rapidement est la suivante. Sachant que nous avons $|\mathcal{D}|$ points dans notre jeu de données, à quelle distance du centre se situe le point le plus proche. De manière plus formelle, nous souhaitons calculer la médiane de cette distance. De manière évidente, nous avons $\mathbb{P}(\boldsymbol{x} \in \mathcal{B}_1) = 1$. De plus, soit $\mathcal{B}_r$ une boule centrée en $\mathbf{0}$ de rayon r inférieur à 1. On peut observer que la probabilité

qu'un point tombe dans la "petite" boule est le ratio des volumes des deux boules :

$$\mathbb{P}(\boldsymbol{x} \in \mathcal{B}_r) = r^{-d}.$$

Puisque nous sommes intéressé par la médiane de la distance la plus faible, nous cherchons le rayon de la boule $\mathcal{B}_r$ tel que l'ensemble des points de notre jeu de données ne tombent pas à l'intérieur 50% du temps. Autrement dit, nous cherchons r tel que :

$$\mathbb{P}(\forall \boldsymbol{x},\ \boldsymbol{x} \notin \mathcal{B}_r) = \frac{1}{2}.$$

La probabilité qu'un unique point ne tombe pas dans la boule est donnée par :

$$\mathbb{P}(\boldsymbol{x} \notin \mathcal{B}_r) = 1 - r^{-d}.$$

Et qu'aucun ne tombe dans la boule par :

$$\mathbb{P}(\forall \boldsymbol{x},\ \boldsymbol{x} \notin \mathcal{B}_r) = (1 - r^{-d})^N.$$

Ce qui nous mène à l'équation à une inconnue suivante :

$$(1 - r^{-d})^N = \frac{1}{2} \Leftrightarrow r = \big(1 - \frac{1}{2^{1/N}}\big)^{1/d}.$$

Afin d'illustrer l'implication de ce résultat, supposons que notre jeu de données contiennent des images de taille $16 \times 16 = 256$ et $32 \times 32 = 1024$. Ces images sont ridiculement petites comparées à ce que nous manipulons en vrai. Cependant, dès ces tailles là, la malédiction de la dimension commence à frapper. La figure 3 illustre l'évolution de la distance médiane en fonction du nombre de points dans notre jeu de données ainsi que de la dimension du problème.

Avec une taille de jeu de données acceptable, on se retrouve avec la totalité des points quasi équidistants du centre de la boule.

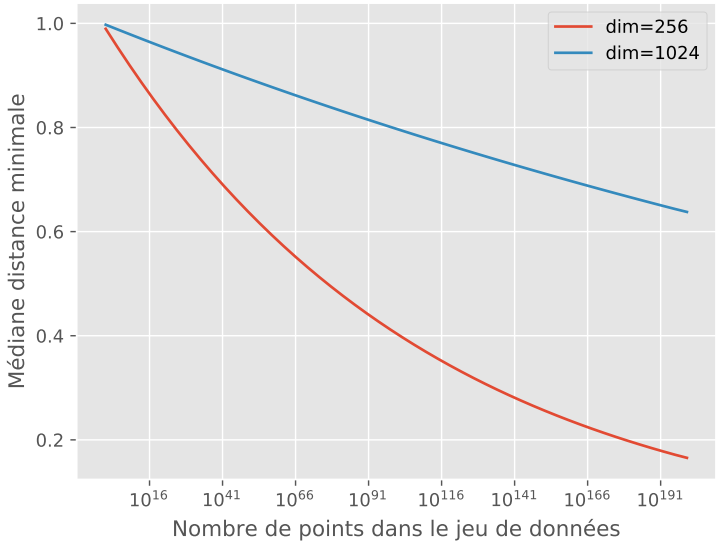


FIGURE 3: Évolution de la médiane de la distance du point le plus proche au centre en fonction de la dimension et du nombre de points dans le jeu de données. Rappelons que 10^{80} est le nombre estimé d'atomes dans l'univers et qu'il représente ici la quantité de photos de chiens et de chats dans notre jeu de données.