

# Algorithme des k-plus-proches-voisins

## 1 Préliminaires

Avant de s'intéresser à la programmation de l'algorithme des k-plus-proches-voisins, ou k-ppv il faudra avoir fini les étapes suivantes :

- tâches de base concernant les dessins (rectangle, manipulation de pixels) ;
- transformation en HSV ;
- enregistrer une image/partie d'image dans un fichier ;
- découpage en modules!!!! Et écriture d'un Makefile.

## 2 Algorithme des k-ppv

L'algorithme des k-plus-proches-voisins est l'un des algorithmes les plus simple d'apprentissage automatique supervisé. En supposant qu'une base d'apprentissage correctement étiquetée soit à disposition, cette méthode permet d'obtenir de très bon résultats de classification.

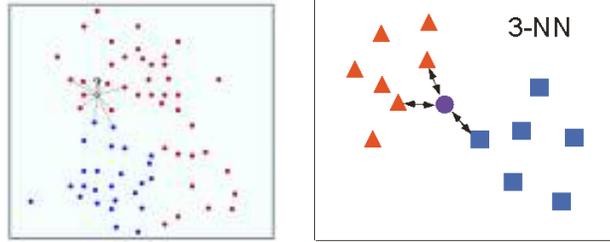
### 2.1 Contexte : classification supervisée

Par classification supervisée, on entend le problème suivant : à partir d'un échantillon *fini* d'objets étiquetés/classés, on désire construire une fonction capable d'étiqueter/classer au mieux de nouveaux objets (ne faisant pas partie de l'échantillon initial). Cela correspond parfaitement à la situation d'un élève de CP qui apprend à lire : sa maîtresse lui dessine au tableau des symboles représentant des lettres (les classes) et l'idée d'un apprentissage réussi est que l'élève arrive à lire n'importe quels caractères, quels que soient les scripteurs. .le concept-clé de l'apprentissage est donc la *généralisation*. C'est également le cas en apprentissage automatique, où de multiples algorithmes ont été proposés pour la tâche de classification. Les meilleurs algorithmes d'apprentissage reposent sur des résultats de statistique théorique extrêmement forts.

Pour la détection de visage, les objets que nous allons considérer sont simplement des images, ou des portions d'image, et les classes utilisées sont simplement `visage/non visage`.

### 2.2 K-ppv : classifieur « paresseux »

L'algorithme k-ppv est un algorithme de la famille des algorithmes dits « paresseux » : à l'inverse de beaucoup d'autres méthodes d'apprentissage automatique (tels que la régression logistique, les réseaux de neurones artificiels, les méthodes à noyaux, etc.) aucun apprentissage ne prend réellement place, c'est-à-dire qu'il n'y a pas de phase de détermination de paramètres d'une fonction par le biais d'une optimisation mathématique (cf. les prochains TD sur la régression logistique). Le principe est le suivant : étant donnée une base d'apprentissage d'images étiquetées correctement et un entier  $k$ , le classifieur k-ppv détermine la classe d'un nouvel objet en lui attribuant la classe majoritaire des  $k$  objets lui ressemblant le plus dans la base d'apprentissage. Les figures suivantes illustrent le principe de fonctionnement de ce classifieur (sources : [http://www.vias.org/tmdatanaleng/cc\\_classif\\_knn.html](http://www.vias.org/tmdatanaleng/cc_classif_knn.html) et <http://www.nysaes.cornell.edu/fst/faculty/siebert/FS608/syllabus.html>).



On note deux aspects importants de l'algorithme k-ppv : d'une part, à chaque nouvelle classification il est nécessaire de parcourir l'ensemble de la base d'apprentissage, ce qui en fait un algorithme qui n'est pas nécessairement très efficaces (surtout que, habituellement, on cherche à avoir la base d'apprentissage la plus grande possible afin d'avoir un meilleur classifieur), et d'autre part, un point crucial de cet algorithme est la fonction de distance utilisée pour mesurer la proximité des objets. Il n'existe pas de distance/similarité universellement optimale et une bonne connaissance du problème traité guide généralement le choix de cette distance/similarité.

### 3 Travail à faire

Le travail à faire concernant l'algorithme k-ppv s'étalera sur deux séances car il comporte plusieurs phases plus ou moins conséquentes :

1. récupération d'une base d'images représentant des visages et d'autres des non visages : ces bases constitueront l'ensemble d'apprentissage ;
2. calcul des histogrammes d'une image en représentation RGB et HSV ;
3. programmation de l'algorithme k-ppv ;
4. utilisation de la distance euclidienne et du cosinus dans le k-ppv pour mesurer l'efficacité de la classification effectuée par l'algorithme ;
5. connexion du module des k-ppv à l'interface graphique et encadrement des régions des images représentant des visages ;
6. (partie difficile) ré-étiquetage à la souris des régions d'image mal classées pour l'enrichissement de la base d'apprentissage.