Intelligence Artificielle pour l'Analyse de Données

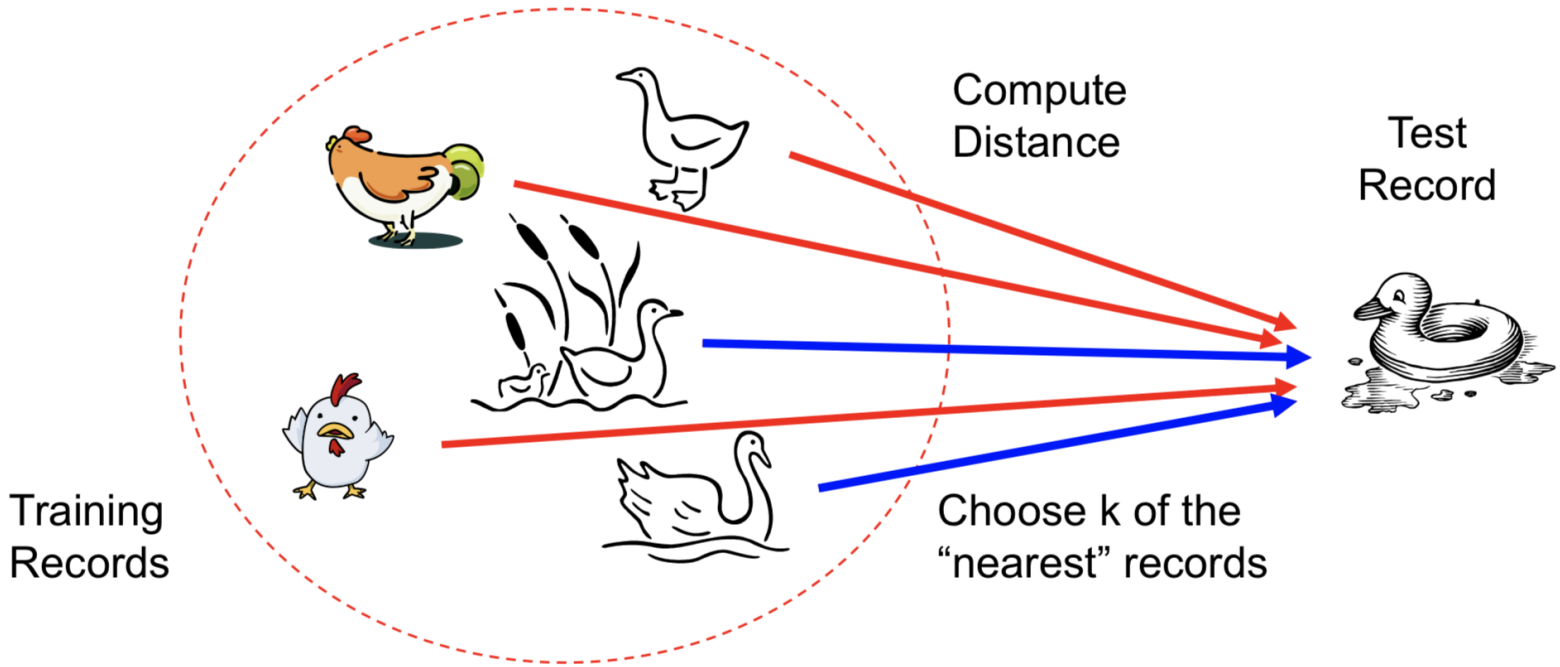
k-NN ou Nearest-Neighbor Classifier

Akka Zemmari
zemmari@u-bordeaux.fr

LaBRI, Université de Bordeaux - CNRS

2019-2020

Idées de base



Si ça marche comme un canard, crie comme un canard, c'est que c'est probablement un canard ... [1]

[1] Schéma de B. S. Panda (IIT Delhi)

Idées de base

Données d'entrainement

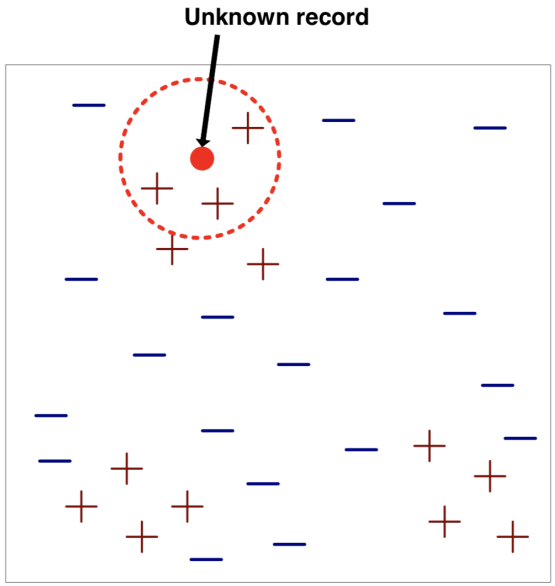
A_1	A_2	$\cdots$	A_n	Classe
				A
				B
				B
				C
				A
				C
				B

- Stocker les données d'entraînement.

- Utiliser ces données pour prédire la classe de l'ensemble de test.

A_1	A_2	$\cdots$	A_n

Nearest-Neighbor Classifiers



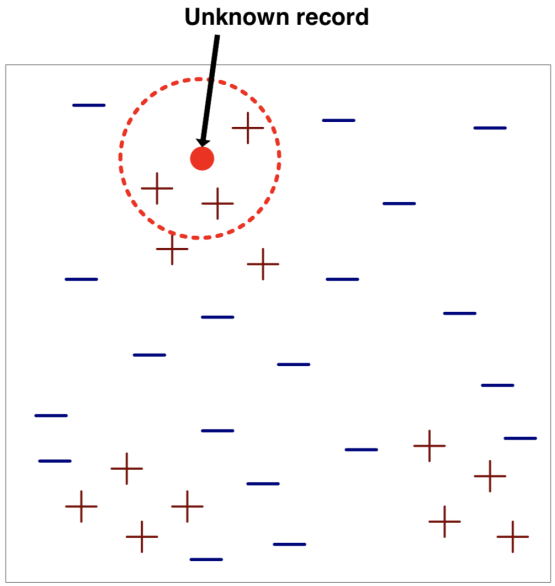
On a besoin de trois choses :

- Un ensemble d'entrainement.
- Une mesure de distance.
- La valeur de k, le nombre de voisins à interroger.

Pour classifier un nouvel enregistrement :

- Calculer la distance vers les autres enregistrements (de l'ensemble d'entraînement).
- Identifier k plus proches voisins.
- Utiliser la classe des k voisins les plus proches pour déterminer la classe du nouvel enregistrement (par un vote majoritaire par exemple).

Nearest-Neighbor Classifiers



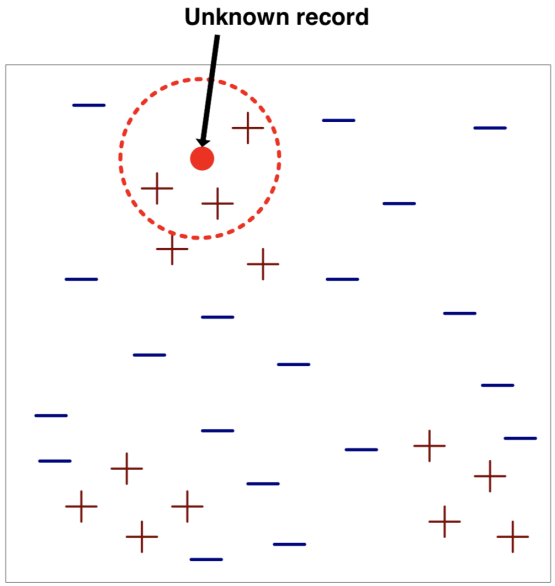
On a besoin de trois choses :

- Un ensemble d'entrainement.
- Une mesure de distance.
- La valeur de k, le nombre de voisins à interroger.

Pour classifier un nouvel enregistrement :

- Calculer la distance vers les autres enregistrements (de l'ensemble d'entraînement).
- Identifier k plus proches voisins.
- Utiliser la classe des k voisins les plus proches pour déterminer la classe du nouvel enregistrement (par un vote majoritaire par exemple).

Nearest-Neighbor Classifiers



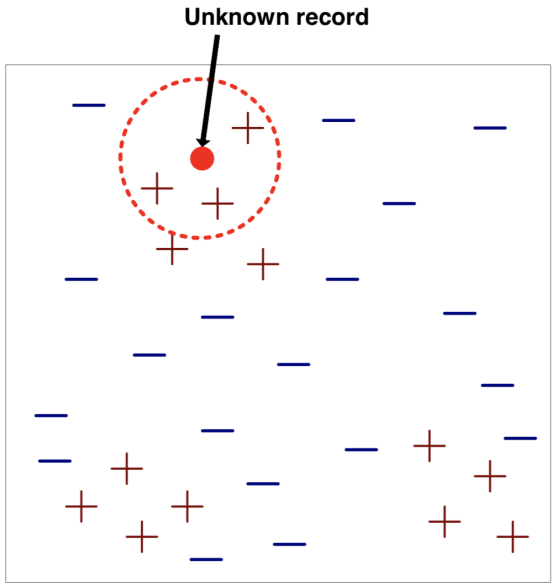
On a besoin de trois choses :

- Un ensemble d'entrainement.
- Une mesure de distance.
- La valeur de k, le nombre de voisins à interroger.

Pour classifier un nouvel enregistrement :

- Calculer la distance vers les autres enregistrements (de l'ensemble d'entraînement).
- Identifier k plus proches voisins.
- Utiliser la classe des k voisins les plus proches pour déterminer la classe du nouvel enregistrement (par un vote majoritaire par exemple).

Nearest-Neighbor Classifiers



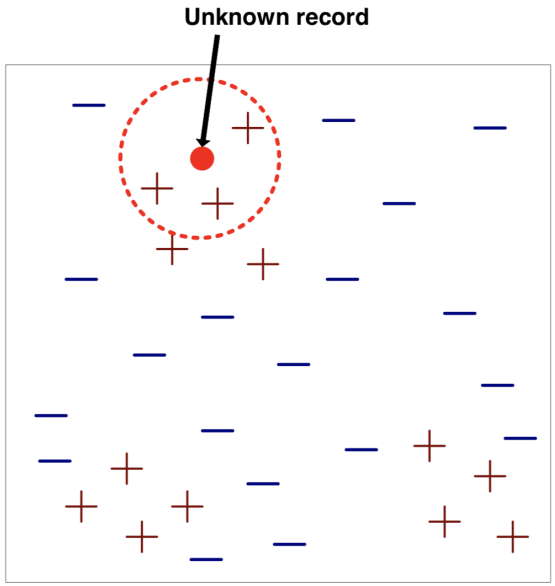
On a besoin de trois choses :

- Un ensemble d'entrainement.
- Une mesure de distance.
- La valeur de k, le nombre de voisins à interroger.

Pour classifier un nouvel enregistrement :

- Calculer la distance vers les autres enregistrements (de l'ensemble d'entraînement).
- Identifier k plus proches voisins.
- Utiliser la classe des k voisins les plus proches pour déterminer la classe du nouvel enregistrement (par un vote majoritaire par exemple).

Nearest-Neighbor Classifiers



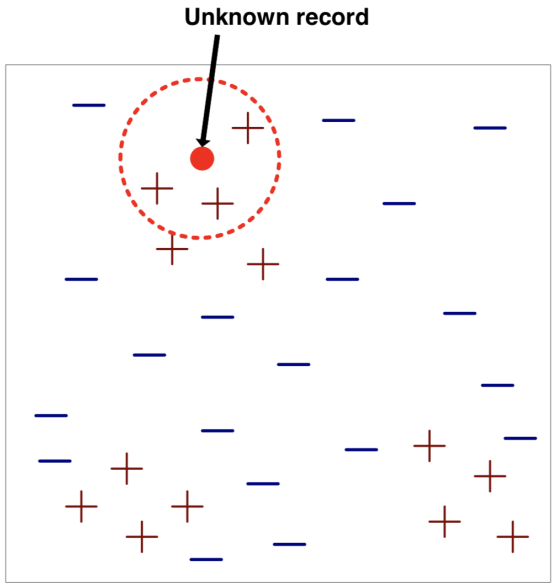
On a besoin de trois choses :

- Un ensemble d'entrainement.
- Une mesure de distance.
- La valeur de k, le nombre de voisins à interroger.

Pour classifier un nouvel enregistrement :

- Calculer la distance vers les autres enregistrements (de l'ensemble d'entraînement).
- Identifier k plus proches voisins.
- Utiliser la classe des k voisins les plus proches pour déterminer la classe du nouvel enregistrement (par un vote majoritaire par exemple).

Nearest-Neighbor Classifiers



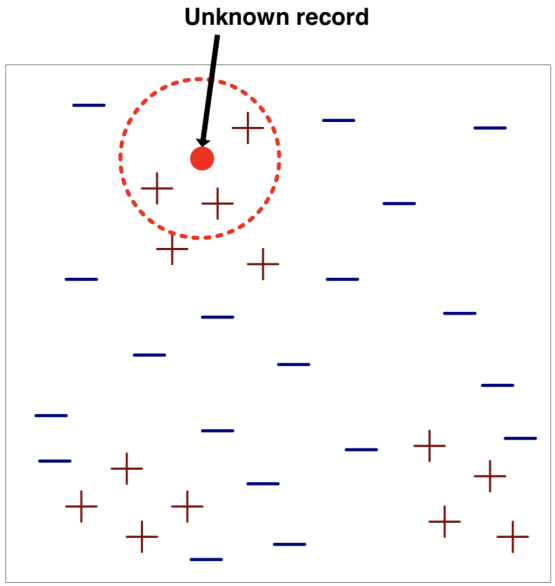
On a besoin de trois choses :

- Un ensemble d'entrainement.
- Une mesure de distance.
- La valeur de k, le nombre de voisins à interroger.

Pour classifier un nouvel enregistrement :

- Calculer la distance vers les autres enregistrements (de l'ensemble d'entraînement).
- Identifier k plus proches voisins.
- Utiliser la classe des k voisins les plus proches pour déterminer la classe du nouvel enregistrement (par un vote majoritaire par exemple).

Nearest-Neighbor Classifiers



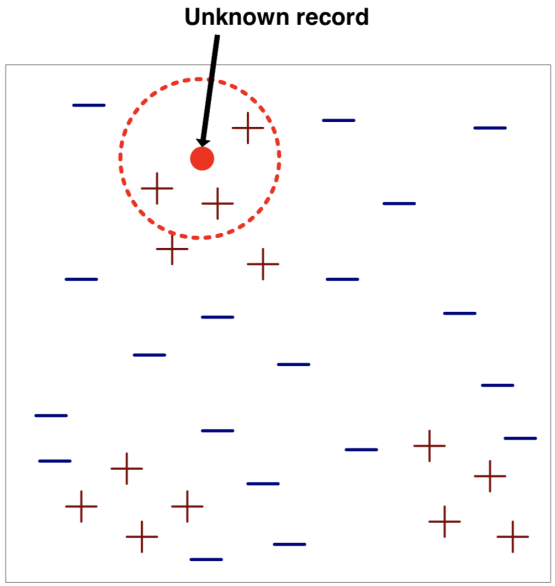
On a besoin de trois choses :

- Un ensemble d'entrainement.
- Une mesure de distance.
- La valeur de k, le nombre de voisins à interroger.

Pour classifier un nouvel enregistrement :

- Calculer la distance vers les autres enregistrements (de l'ensemble d'entraînement).
- Identifier k plus proches voisins.
- Utiliser la classe des k voisins les plus proches pour déterminer la classe du nouvel enregistrement (par un vote majoritaire par exemple).

Nearest-Neighbor Classifiers



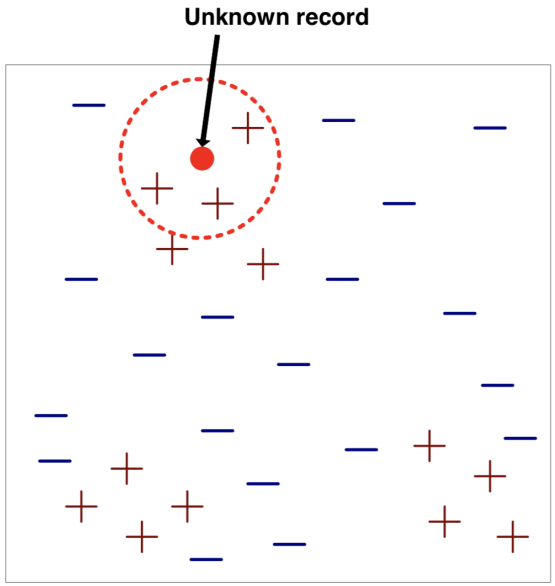
On a besoin de trois choses :

- Un ensemble d'entrainement.
- Une mesure de distance.
- La valeur de k, le nombre de voisins à interroger.

Pour classifier un nouvel enregistrement :

- Calculer la distance vers les autres enregistrements (de l'ensemble d'entraînement).
- Identifier k plus proches voisins.
- Utiliser la classe des k voisins les plus proches pour déterminer la classe du nouvel enregistrement (par un vote majoritaire par exemple).

Nearest-Neighbor Classifiers



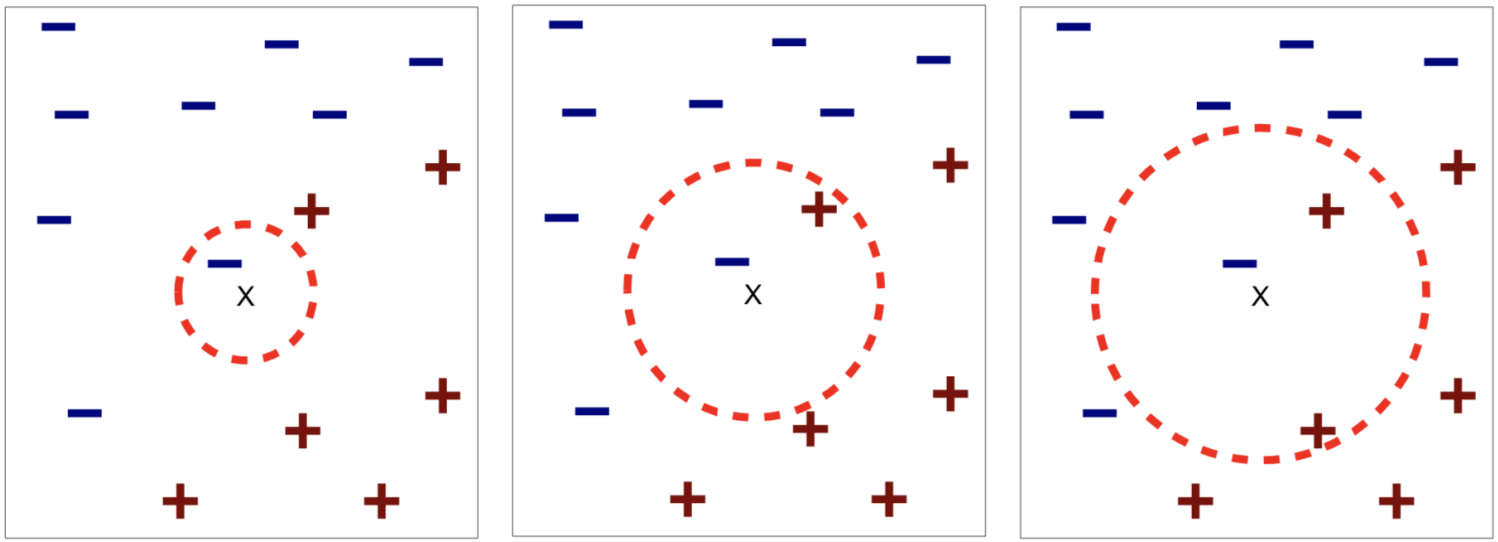
On a besoin de trois choses :

- Un ensemble d'entrainement.
- Une mesure de distance.
- La valeur de k, le nombre de voisins à interroger.

Pour classifier un nouvel enregistrement :

- Calculer la distance vers les autres enregistrements (de l'ensemble d'entraînement).
- Identifier k plus proches voisins.
- Utiliser la classe des k voisins les plus proches pour déterminer la classe du nouvel enregistrement (par un vote majoritaire par exemple).

Nearest-Neighbor ?



(a) 1-nearest neighbor (b) 2-nearest neighbor (c) 3-nearest neighbor

NN Classification

Calculer la distance entre deux points :

- Distance euclidienne :

$$d(p,q) = \sqrt{\sum_i (p_i - q_i)^2}. \tag{1}$$

- Distance de Manhatten :

$$d(p,q) = \sum_i \mid p_i - q_i \mid . \tag{2}$$

- Norme r :

$$d(p,q) = \left(\sum_i \mid p_i - q_i \mid^r \right)^{1/r} . \tag{3}$$

NN Classification

Calculer la distance entre deux points :

- Distance euclidienne :

$$d(p,q) = \sqrt{\sum_i (p_i - q_i)^2}. \tag{1}$$

- Distance de Manhatten :

$$d(p,q) = \sum_i \mid p_i - q_i \mid . \tag{2}$$

- Norme r :

$$d(p,q) = \left(\sum_i \mid p_i - q_i \mid^r \right)^{1/r} . \tag{3}$$

NN Classification

Calculer la distance entre deux points :

- Distance euclidienne :

$$d(p,q) = \sqrt{\sum_i (p_i - q_i)^2}. \tag{1}$$

- Distance de Manhatten :

$$d(p,q) = \sum_i \mid p_i - q_i \mid . \tag{2}$$

- Norme r :

$$d(p,q) = \left(\sum_i \mid p_i - q_i \mid^r \right)^{1/r} . \tag{3}$$

NN Classification

Calculer la distance entre deux points :

- Distance euclidienne :

$$d(p,q) = \sqrt{\sum_i (p_i - q_i)^2}. \tag{1}$$

- Distance de Manhatten :

$$d(p,q) = \sum_i \mid p_i - q_i \mid . \tag{2}$$

- Norme r :

$$d(p,q) = \left(\sum_i \mid p_i - q_i \mid^r \right)^{1/r} . \tag{3}$$

NN Classification

Calculer la distance entre deux points :

- Distance euclidienne :

$$d(p,q) = \sqrt{\sum_i (p_i - q_i)^2}. \tag{1}$$

- Distance de Manhatten :

$$d(p,q) = \sum_i \mid p_i - q_i \mid . \tag{2}$$

- Norme r :

$$d(p,q) = \left(\sum_i \mid p_i - q_i \mid^r \right)^{1/r} . \tag{3}$$

NN Classification

Déterminer la classe à partir de la liste de voisins (pour classifier un exemple z) :

- choisir la classe majoritaire dans le k-voisinage :

$$y' = \arg\max_{v} \sum_{(x_i, y_i) \in D_z} I\left(v = y_i\right), \tag{4}$$

où D_z est l'ensemble des k exemples les plus proches de z.

NN Classification

Déterminer la classe à partir de la liste de voisins (pour classifier un exemple z) :

- choisir la classe majoritaire dans le k-voisinage :

$$y' = \arg\max_{v} \sum_{(x_i, y_i) \in D_z} I\left(v = y_i\right), \tag{4}$$

où D_z est l'ensemble des k exemples les plus proches de z.

k-NN en une diapo

Soit k le nombre de voisins les plus proches et D l'ensemble d'entraînement.

1. pour chaque exemple $z = (x', ?)$ de l'ensemble de test :
   - 1.1 Calculer $d(x, x')$, la distance de z et chaque exemple (x, y) de D;
   - 1.2 Choisir $D_z \subset D$, l'ensemble des k exemples les plus proches de z;
   - 1.3 $y' = \arg\max_v \sum_{(x_i, y_i) \in D_z} I(v = y_i)$
2. Fin pour

k-NN en une diapo

Soit k le nombre de voisins les plus proches et D l'ensemble d'entraînement.

1. pour chaque exemple $z = (x', ?)$ de l'ensemble de test :
   1.1 Calculer $d(x, x')$, la distance de z et chaque exemple (x, y) de D;
   1.2 Choisir $D_z \subset D$, l'ensemble des k exemples les plus proches de z;
   1.3 $y' = \arg\max_v \sum_{(x_i, y_i) \in D_z} I(v = y_i)$
2. Fin pour

k-NN en une diapo

Soit k le nombre de voisins les plus proches et D l'ensemble d'entraînement.

1. pour chaque exemple $z = (x', ?)$ de l'ensemble de test :
   1.1 Calculer $d(x, x')$, la distance de z et chaque exemple (x, y) de D;
   1.2 Choisir $D_z \subset D$, l'ensemble des k exemples les plus proches de z;
   1.3 $y' = \arg\max_v \sum_{(x_i, y_i) \in D_z} I(v = y_i)$
2. Fin pour

k-NN en une diapo

Soit k le nombre de voisins les plus proches et D l'ensemble d'entraînement.

1. pour chaque exemple $z = (x', ?)$ de l'ensemble de test :
   - 1.1 Calculer $d(x, x')$, la distance de z et chaque exemple (x, y) de D;
   - 1.2 Choisir $D_z \subset D$, l'ensemble des k exemples les plus proches de z;
   - 1.3 $y' = \arg\max_v \sum_{(x_i, y_i) \in D_z} I(v = y_i)$
2. Fin pour

k-NN en une diapo

Soit k le nombre de voisins les plus proches et D l'ensemble d'entraînement.

1. pour chaque exemple $z = (x', ?)$ de l'ensemble de test :
   - 1.1 Calculer $d(x, x')$, la distance de z et chaque exemple (x, y) de D;
   - 1.2 Choisir $D_z \subset D$, l'ensemble des k exemples les plus proches de z;
   - 1.3 $y' = \arg\max_v \sum_{(x_i, y_i) \in D_z} I(v = y_i)$
2. Fin pour

k-NN en une diapo

Soit k le nombre de voisins les plus proches et D l'ensemble d'entraînement.

1. pour chaque exemple $z = (x', ?)$ de l'ensemble de test :
   - 1.1 Calculer $d(x, x')$, la distance de z et chaque exemple (x, y) de D;
   - 1.2 Choisir $D_z \subset D$, l'ensemble des k exemples les plus proches de z;
   - 1.3 $y' = \arg\max_v \sum_{(x_i, y_i) \in D_z} I(v = y_i)$
2. Fin pour

k-NN en une diapo

Soit k le nombre de voisins les plus proches et D l'ensemble d'entraînement.

1. pour chaque exemple $z = (x', ?)$ de l'ensemble de test :
   - 1.1 Calculer $d(x, x')$, la distance de z et chaque exemple (x, y) de D;
   - 1.2 Choisir $D_z \subset D$, l'ensemble des k exemples les plus proches de z;
   - 1.3 $y' = \arg\max_v \sum_{(x_i, y_i) \in D_z} I(v = y_i)$
2. Fin pour

NN Classification

Choix de le valeur de k :

- Si k est trop petit, la classification sera trop sensible au "bruit".
- Si k est trop grand, le voisinage peut contenir des éléments d'autres classes.

NN Classification

Choix de le valeur de k :

- Si k est trop petit, la classification sera trop sensible au "bruit".
- Si k est trop grand, le voisinage peut contenir des éléments d'autres classes.

NN Classification

Choix de le valeur de k :

- Si k est trop petit, la classification sera trop sensible au "bruit".
- Si k est trop grand, le voisinage peut contenir des éléments d'autres classes.

NN Classification

Quelques précautions à prendre :

- Les attributs doivent être normalisés pour éviter que les distances soient faussées par des attributs à grande valeur.
- Exemple : Taille (H), Poids (W) et revenu (I) d'une personne avec :
  - $H \in [1.5m, 1.8m]$
  - $W \in [60kg, 100kg]$
  - $I \in [15ke, 60ke]$.

NN Classification

Quelques précautions à prendre :

- Les attributs doivent être normalisés pour éviter que les distances soient faussées par des attributs à grande valeur.
- Exemple : Taille (H), Poids (W) et revenu (I) d'une personne avec :
  - $H \in [1.5m, 1.8m]$
  - $W \in [60kg, 100kg]$
  - $I \in [15ke, 60ke]$.

NN Classification

Quelques précautions à prendre :

- Les attributs doivent être normalisés pour éviter que les distances soient faussées par des attributs à grande valeur.
- Exemple : Taille (H), Poids (W) et revenu (I) d'une personne avec :
  - $H \in [1.5m, 1.8m]$
  - $W \in [60kg, 100kg]$
  - $I \in [15ke, 60ke]$.

NN Classification

Quelques précautions à prendre :

Attention à la distance euclidienne ...

- Vecteurs de features à grande dimension
  $\rightarrow$ presque tous les vecteurs sont à la même distance de l'exemple qu'on veut classifier.
- Solution : réduire la dimension des vecteurs (ACP par exemple)

NN Classification

Quelques précautions à prendre :
Attention à la distance euclidienne ...

- Vecteurs de features à grande dimension
  $\rightarrow$ presque tous les vecteurs sont à la même distance de l'exemple qu'on veut classifier.
- Solution : réduire la dimension des vecteurs (ACP par exemple)

NN Classification

Quelques précautions à prendre :
Attention à la distance euclidienne ...

- Vecteurs de features à grande dimension
  $\rightarrow$ presque tous les vecteurs sont à la même distance de l'exemple qu'on veut classifier.
- Solution : réduire la dimension des vecteurs (ACP par exemple)

NN Classification

Quelques précautions à prendre :
Attention à la distance euclidienne ...

- Vecteurs de features à grande dimension
  $\rightarrow$ presque tous les vecteurs sont à la même distance de l'exemple qu'on veut classifier.
- Solution : réduire la dimension des vecteurs (ACP par exemple)

Démo

Voyons comment ça marche en pratique