02-01
Recherche des plus proches voisins

Sommaire

Sommaire

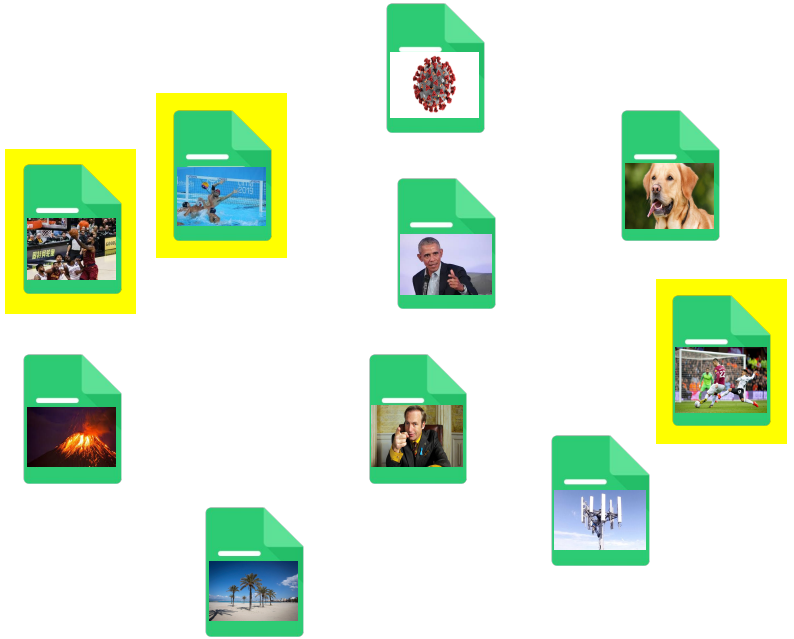
Recherche de documents

- Commençons par un exemple: vous êtes présentement en train de lire un article sur le soccer ...

Vous souhaitez trouver des articles **similaires**

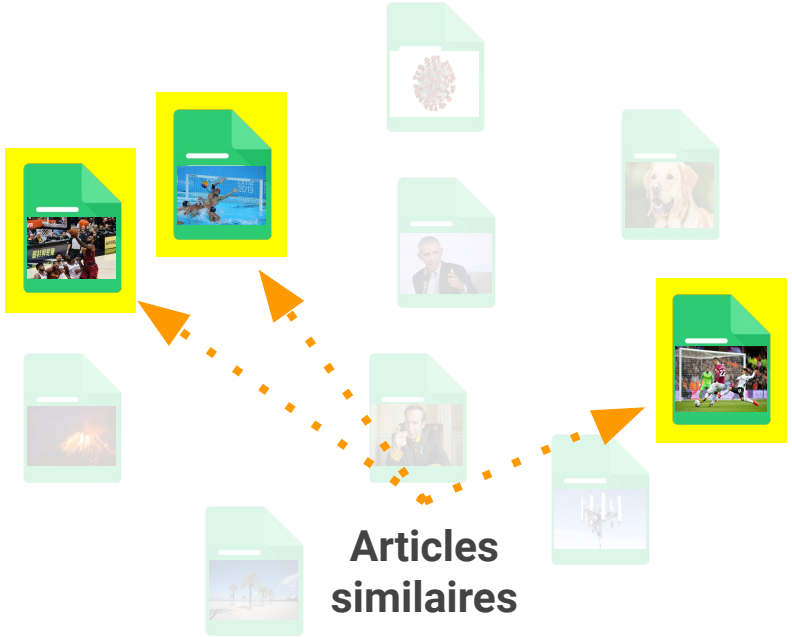
Recherche de documents

- Commençons par un exemple: vous êtes présentement en train de lire un article sur le soccer ...



Recherche de documents

- Commençons par un exemple: vous êtes présentement en train de lire un article sur le soccer ...



Recherche de documents

- Commençons par un exemple: vous êtes présentement en train de lire un article sur le soccer ...

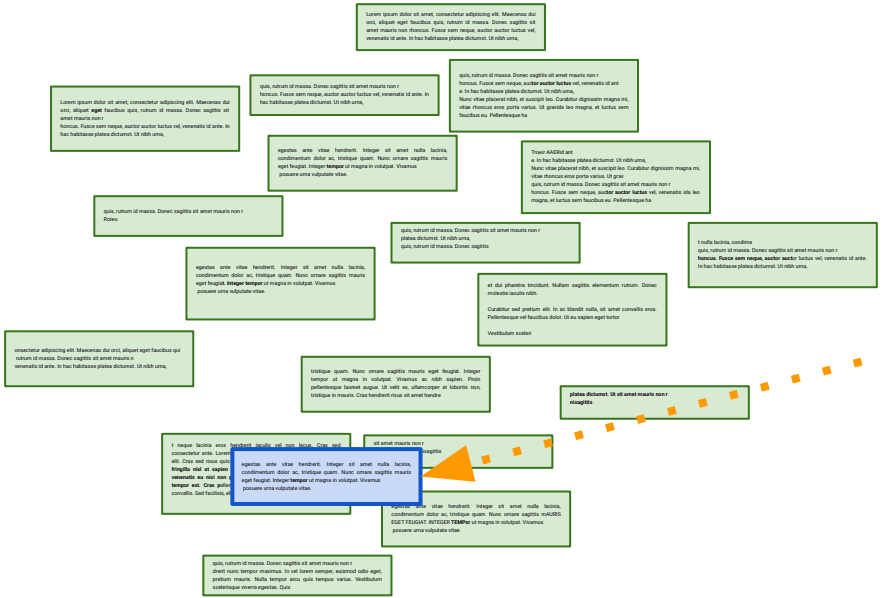
Comment **chercher** parmi tous les **documents** disponibles ?

Comment **mesurer la similarité** ?

Sommaire

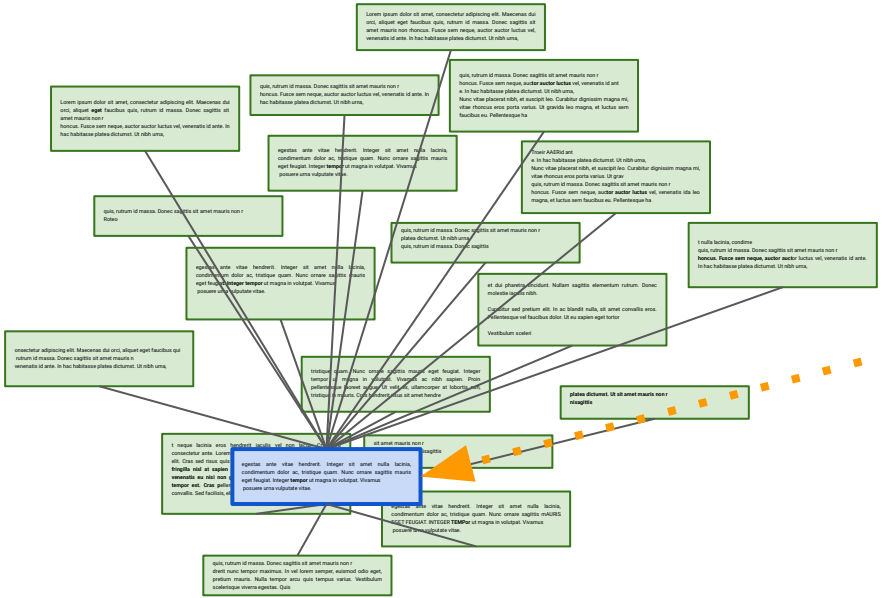
Recherche 1-NN

- Ci-dessous, est représenté l'espace de tous les documents organisés par **similarité du texte**



Recherche 1-NN

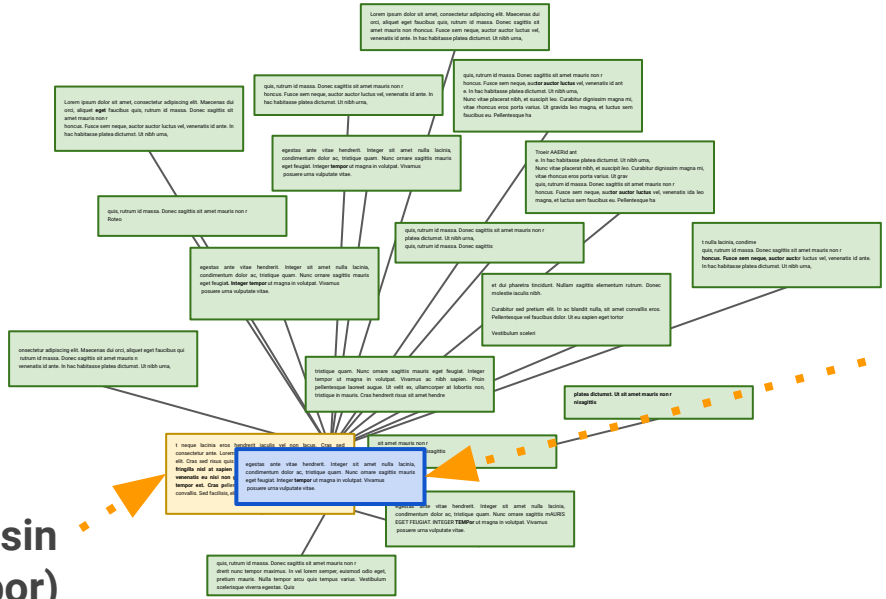
- On calcul de la **distance** vers tous les documents



Article requête (query article)

Recherche 1-NN

- Identification du **plus proche voisin**

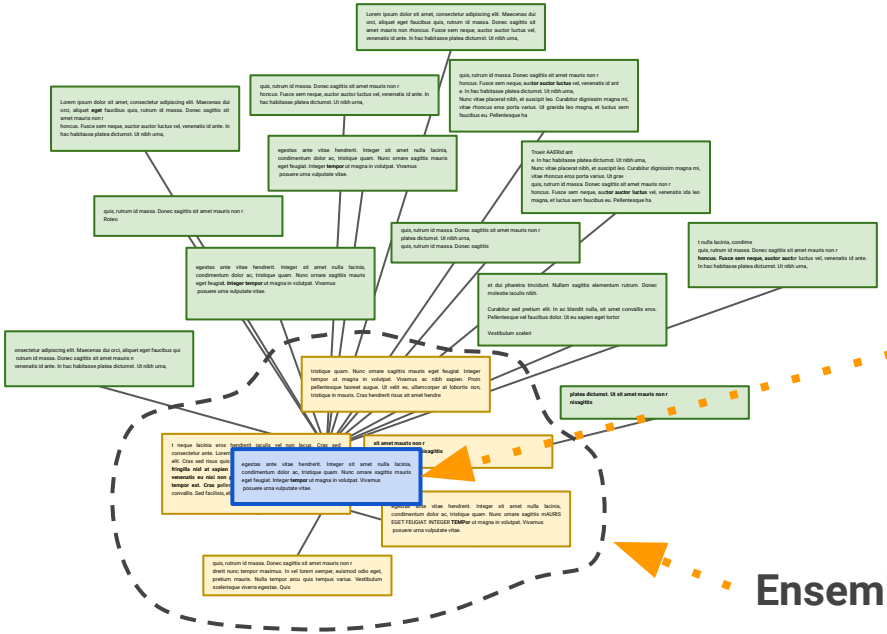


Article requête (query article)

Plus proche voisin (nearest neighbor)

Recherche k-NN

- Identification d'un ensemble de **plus proches voisins**

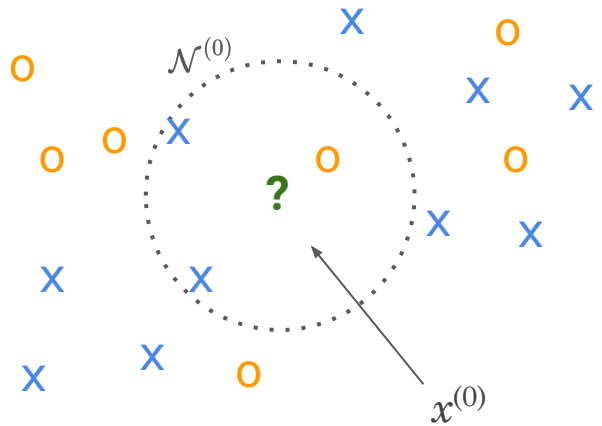


Article requête (query article)

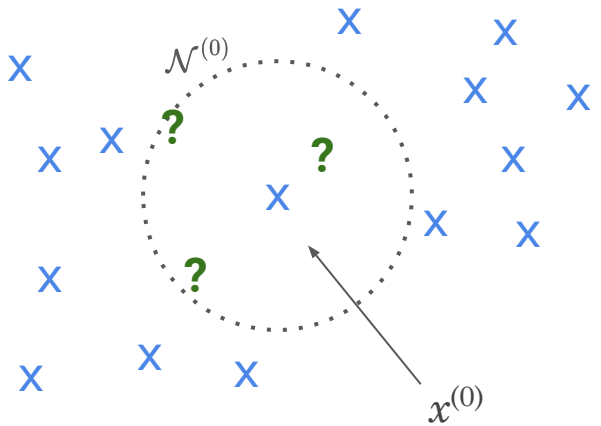
Ensemble de plus proches voisins

k-NN: comparaison supervisé / non supervisé

Apprentissage supervisé



Apprentissage non supervisé



Sommaire

Algorithme 1-NN: notations

- **Entrée:**
  - Document “query”: x_q 📙
  - Corpus de documents: x_1, x_2, …, x_M 
- **Sortie:**
  - Document le plus similaire: x^{NN} 📕
- Formellement, nous cherchons le document x_i ayant la distance minimale avec x_q

$$x^{NN} = \min_{x_i} \text{distance}\left(x_q, x_i\right)$$

Algorithme 1-NN: pseudo-code

Initialiser Dist2NN = ∞, x^{NN} = ∅

Itérer sur tous les documents x_1, x_2, …, x_M:

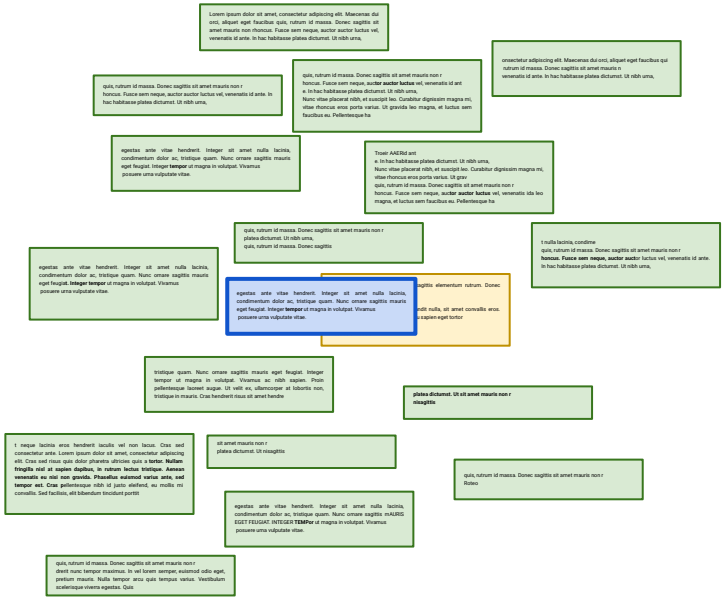
    Calculer la distance δ entre x_q et x_i

    Si δ < Dist2NN

        $x^{NN} = x_i$

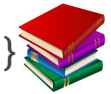
        Dist2NN = δ

Retourner le document le plus similaire x^{NN}
(document du corpus le plus proche de x_q)



Sommaire

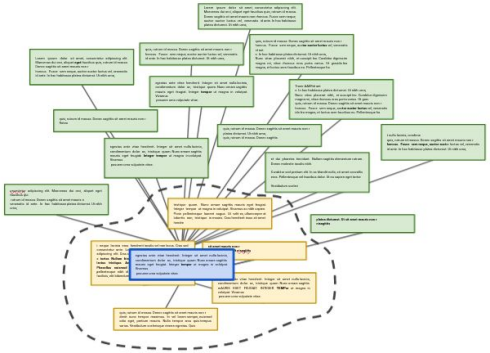
Algorithme k-NN

- **Entrée:**
  - Document “query”: x_q 
  - Corpus de documents: x_1, x_2, …, x_M 
- **Sortie:**
  - Liste des k documents les plus similaires: $X^{NN} = \{ x^{NN1}, x^{NN2}, \ldots, x^{NNk} \}$ 
- Formellement, nous cherchons tous les documents x_i tels que

$$\forall x_i \notin X^{NN}, \text{distance}(x_q, x_i) \geq \max_{x_i^{NNj}, j=1\ldots,k} \text{distance}(x_q, x^{NNj})$$

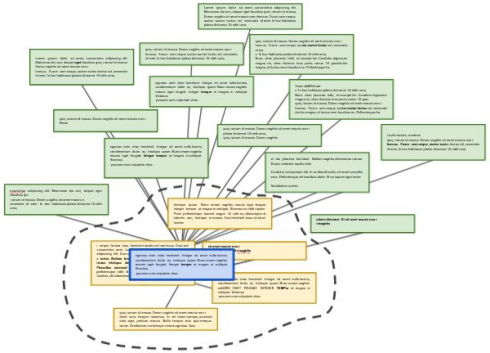
Algorithme k-NN

```
Initialiser Dist2kNN = sort(δ1, δ2, …, δk)
Itérer sur tous les documents de k+1 à M: xk+1, …, xM:
    Calculer la distance δ entre xq et xi
    Si  δ < Dist2kNN[k]
        Trouver j tel que δ > Dist2kNN[j-1] et δ < Dist2kNN[j]
        Retirer le document le plus loin et déplacer les autres
        Dist2kNN[j+1:k] = Dist2kNN[j:k-1]
        Dist2kNN[j] = δ
Retourne les k documents les plus similaires
```



Trier les k premiers documents selon leur distance au document “query”

Algorithme k-NN

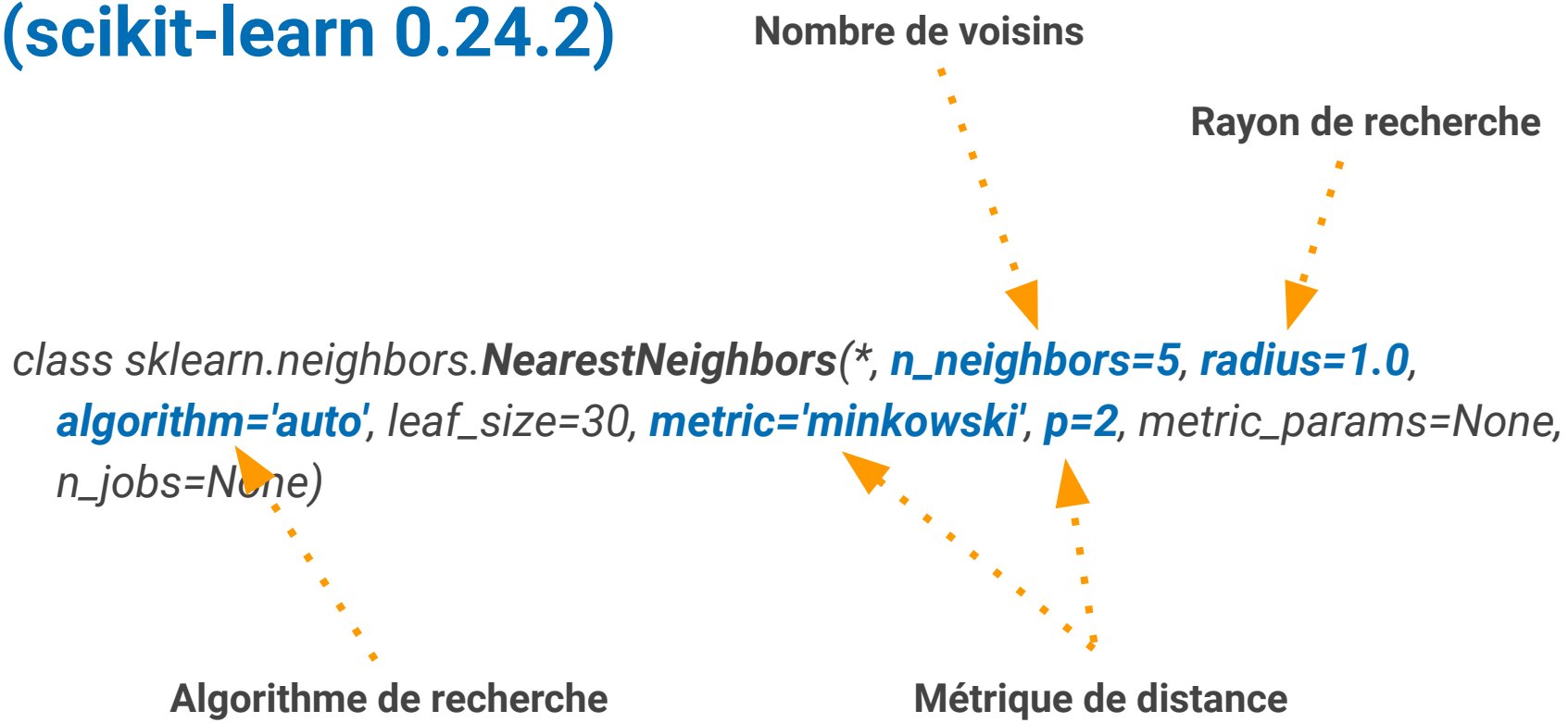


```
Initialiser Dist2kNN = sort(δ1, δ2, …, δk)
Itérer sur tous les documents de k+1 à M: xk+1, …, xM:
    Calculer la distance δ entre xq et xi
    Si  δ < Dist2kNN[k]
        Trouver j tel que δ > Dist2kNN[j-1] et δ < Dist2kNN[j]
        Retirer le document le plus loin et déplacer les autres
        Dist2kNN[j+1:k] = Dist2kNN[j:k-1]
        Dist2kNN[j] = δ
Retourne les k documents les plus similaires
```

Insertion du document

Sommaire

NearestNeighbors (scikit-learn 0.24.2)



Sommaire

Questions ouvertes

- Comment représenter les documents ? 📙 $\rightarrow x_q$
- Quelle mesure utiliser pour calculer la distance entre les documents ? δ = **distance** (x_i, x_q)

Sommaire

Lectures et références

[1] Machine Learning: Clustering and Retrieval - Emily Fox & Carlos Guestrin - University of Washington