

Systèmes des recommandation : K-plus proche voisin

Fouille de données avancées (2016-2017)


UFR MIME

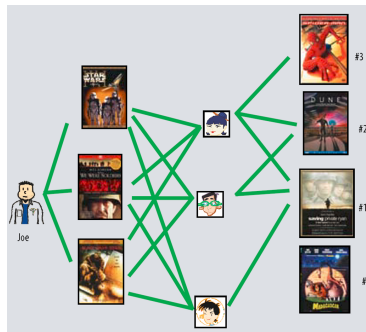
Université Lille 3

11 Janvier 2017

- 1 Reccomendation basé sur utilisateurs
- 2 K-Plus Proche Voisins
- 3 Reccomendation avec K-NN

Donnees Utilisateurs-Filmes :Rappel

				
	1	2	3	2
	2	3	1	?
	5	3	?	5
	2	1	4	1

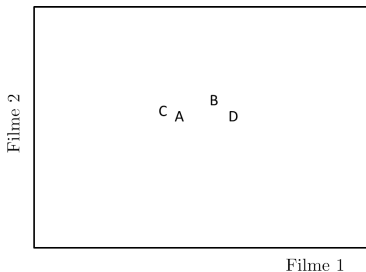


Etant donne un utilisateur i et une filme f , retrouver le valeur de note

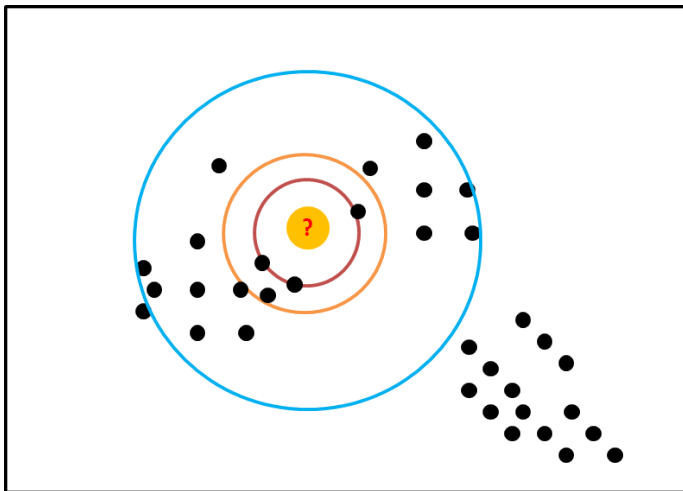
- User-based : Retrouver les autres **utilisateurs** qui sont **similaire** et voir leurs notes pour predire la note pour utilisateur i
- Item-Based : Retrouver les autres **filmes** qui sont **similaire** et voir leurs notes pour predire la note pour filme f

K-Plus proche voisins

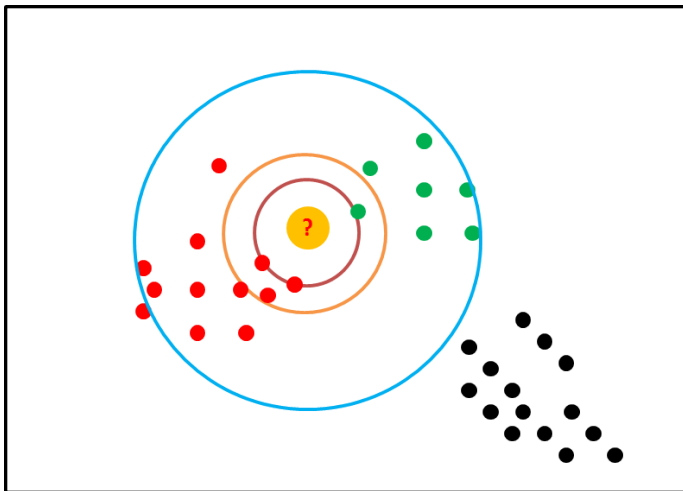
	Filme 1	Filme 2	Filme 3	Filme 4	Filme 5	Filme 6
User A	1	1	1	4	5	5
User B	3	1	5	2	2	5
User C	2	3	1	5	5	?
User D	5	2	4	1	1	3



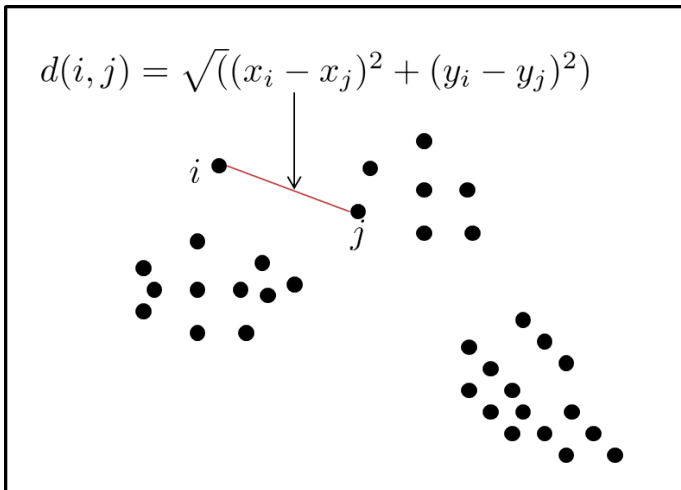
K-Plus proche voisins



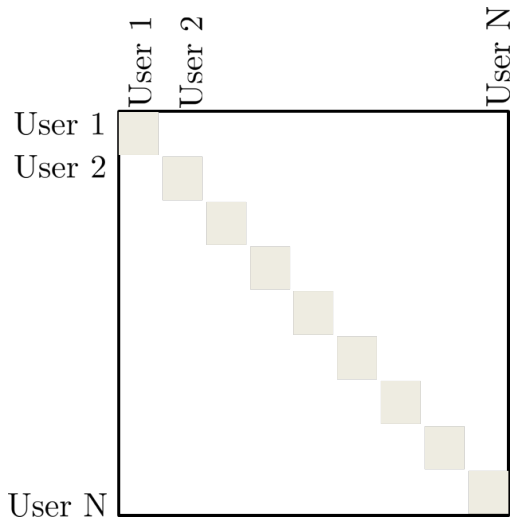
K-Plus proche voisins



Distance a pair ou similarité



Distance a pair ou similarité



- Comment mesurer la similarité ?
- Comment décider la nombre de plus proche voisin k ?
- Comment calculer la valeur/note pour un nouveau (utilisateur, filme) ?
- Quelle est la complexité d'algorithme de k -plus proche voisin ?

- Distance Euclidean entre les vecteurs $\|x_i - x_j\|^2 = \sum_k (x_{ik} - x_{jk})^2$
- Similarité par cosinus entre les vecteurs $d(x_i, x_j) = \frac{x_i \cdot x_j}{\|x_i\| \|x_j\|}$
- Coefficient de corrélation de Pearson

$$\text{sim}(i, j) = \frac{\sum_{p \in P} (n(i, p) - \bar{n}_i) (n(j, p) - \bar{n}_j)}{(\sum_{p \in P} (n(i, p) - \bar{n}_i)^2) \cdot (\sum_{p \in P} (n(j, p) - \bar{n}_j)^2)} \quad (1)$$

Faire une recommandation avec KNN

Etant donnee un user i , et un filme f

- Calculer une distance/similarite entre chaque paires des utilisateurs
- Retrouver les k -plus proche voisin de i parmi tous les users
- Calculer une proportion de confiance pour chaque k -voisin $v \in knn(i)$

$$poid(v) = \frac{d(i, v)}{\sum_{v \in knn(i)} d(i, v)}$$

- Calculer la recommandation/prediction comme :

$$note(i, f) = \sum_{v \in knn(i)} poid(v) * note(v, f) \quad (2)$$

- Tache I : Coder la fonction pour calculer la K-plus proche voisin et donner la fonction distance entre films et la paramètre k
- Tache II : Coder la fonction pour calculer la recommandation pour un nouveau user i et film f étant donné la fonction de distance, k

D'autres questions

- Comment retrouver le meilleur k ?
- Par évaluation d'erreur sur un ensemble masqué T (dernier cours)
- Comment faire la recommandations purment sur les filmes ?
- Quelle est plus rapides : recherche des k -voisin parmi filmes ou users ?