

<http://devfest.gdgnantes.com>

DevFest
Nantes

2014



7 novembre
8h - 18h30

Cité des
Congrès

XSQLI

{EPITECH}
L'ÉCOLE DE L'EXPERTISE INFORMATIQUE

MyScript.
The Power of Handwriting.

lesjeudis.com
LE JEU DES GÉNÉRALISTES INFORMATIQUES

ASI
Le sens technologique

mozilla
FOUNDATION

SNCF

zenika
ARCHITECTURE INFORMATIQUE



NET/PSYS
ingénierie informatique

SOFTEAM Cadextan

SERLI

IPPON
Digital, technology, Hosting



Restlet

Nantes Métropole
COMMUNAUTÉ URBAINE

Google

intel

Braintree_developers

ici : moteurs de recommandation

Cette présentation pourrait vous intéresser. . .

Une introduction aux moteurs de recommandation

Jeff Abrahamson

Jellybooks

Étant donné information sur l'utilisateur, son environnement, et des objets d'intérêt, déterminer les objets recommandables les plus pertinents.

Étant donné information sur l'utilisateur, son environnement, et des objets d'intérêt, déterminer les objets recommandables les plus pertinents.

On n'est pas contraint à trouver les max k .

Il suffit de trouver k parmi un max n .

Exemples

- Amazon
- Google News (ou Le Monde)
- Facebook
- Diagnostique médicale
- App Store / Google Play
- Youtube
- Publicités
- Netflix, last.fm, Spotify, Pandora, . . .
- Browser (propositions de URL)
- Recherche

Valeur pour le client

- Trouver — intéressant
- Réduire choix
- Explorer options
- Découverte (“long tail”)
- Loisirs

Valeur pour le fournisseur

- Prestation unique / supplémentaire
- Confiance, fidélité
- Augmenter ventes, CTR, conversions
- Mieux comprendre le client

Attacher vos ceintures. . .

- On va beaucoup sauter
- Il restera des trous
- Trop de matériel, rien n'est vrai tout le temp

Attacher vos ceintures. . .

- On va beaucoup sauter
- Il restera des trous
- Trop de matériel, rien n'est vrai tout le temp
- Tout projet est un projet de recherche

Attacher vos ceintures. . .

- On va beaucoup sauter
 - Il restera des trous
 - Trop de matériel, rien n'est vrai tout le temp
-
- Tout projet est un projet de recherche
 - Objectif : plus de perspectif

La recommandation

Content-based filtering <i>(filtrage basée sur le continu)</i>	Plus de ce qui ressemble à ce que j'aime.
Collaborative filtering <i>(filtrage collaboratif)</i>	Plus de ce que d'autres qui aiment ce que j'aime aiment.
Knowledge-based filtering <i>(filtrage basée sur connaissance)</i>	Plus de ce qu'il faut.

Content-based filtering

Pour

- Pas besoin de communauté
- Comparaison entre objets possible.

Contre :

- Comprendre contenu
- Cold start (nouveaux utilisateurs)
- Pas de sérendipité

Collaborative filtering

Pour

- Sans études du contenu
- Sérendipité
- Apprentissage du marché

Contre :

- Retours des utilisateurs
- Cold start (nouveaux utilisateurs)
- Cold start (nouveaux objets)

Knowledge-based filtering

Pour

- Déterministe
- Sûr
- Pas de cold start
- Expertise commerciale

Contre :

- Étude pour bootstrap
- Statique, ne répond pas aux tendances

Knowledge-based filtering

Pour

- Déterministe
- Sûr
- Pas de cold start
- Expertise commerciale

Contre :

- Étude pour bootstrap
- Statique, ne répond pas aux tendances

On n'en parle plus.

Utility Matrix

- Users (utilisateurs)
- Items (objets)

Utility Matrix

- Users (utilisateurs)
- Items (objets)

Le but est de trouver les blancs.

	I_1	I_2	I_3	I_4	I_5
U_1	1				
U_2			1	1	1
U_3		1		1	1

Par exemple, vente de livres chez Amazon.

Mais des milliers ou des millions de colonnes et de lignes.

Utility Matrix

- Users (utilisateurs)
- Items (objets)

Le but est de trouver les blancs.

	I_1	I_2	I_3	I_4	I_5
U_1	3				
U_2			5	1	4
U_3		2		5	1

Par exemple, avis de films à Netflix.

Mais des milliers ou des millions de colonnes et de lignes.

Utility Matrix

D'où viennent la matrice ?

- Demande aux utilisateurs
- Observer nos utilisateurs

Ça peut coûter cher...

Item Profiles

Exemples :

- Filmes $\Rightarrow ?$
- Livres $\Rightarrow ?$
- Actualités $\Rightarrow ?$
- Images $\Rightarrow ?$

Item Profiles

Exemples :

- Filmes $\Rightarrow ?$
- Livres $\Rightarrow ?$
- Actualités $\Rightarrow ?$
- Images $\Rightarrow ?$

Filmes :

Contenu : acteurs, cinéaste, année (décennie, etc.), durée

Collaboratif : vu, avis (1–5), ou vu, quand vu relatif à sa sortie

Item Profiles

Exemples :

- Filmes $\Rightarrow ?$
- Livres $\Rightarrow ?$
- Actualités $\Rightarrow ?$
- Images $\Rightarrow ?$

Livres :

Contenu : auteur, genre, année (décennie, etc.), nombre de pages, contenu (très difficile)

Collaboratif : lu, avis (1–5), comment lu

Item Profiles

Exemples :

- Filmes $\Rightarrow ?$
- Livres $\Rightarrow ?$
- Actualités $\Rightarrow ?$
- Images $\Rightarrow ?$

Actualités :

Contenu : source, section, vecteurs de mots avec TF-IDF important

Collaboratif :

Item Profiles

Exemples :

- Filmes $\Rightarrow ?$
- Livres $\Rightarrow ?$
- Actualités $\Rightarrow ?$
- Images $\Rightarrow ?$

Images :

Contenu :

Collaboratif :

Item Profiles

Exemples :

- Filmes $\Rightarrow ?$
- Livres $\Rightarrow ?$
- Actualités $\Rightarrow ?$
- Images $\Rightarrow ?$

Et puis profils d'utilisateur.

Le comportement de l'utilisateur lui attribue des composants des vecteurs.

Vecteurs

Similarité

Similarité : Indice de Jaccard

ou : *Jaccard index*, *Jaccard similarity coefficient*

Similarité :

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|}$$

Similarité : Indice de Jaccard

ou : *Jaccard index*, *Jaccard similarity coefficient*

Similarité :

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|}$$

Distance :

$$J_{\delta}(A, B) = 1 - J(A, B)$$

Similarité cosinus

ou : mesure cosinus, *cosine similarity*

Similarité :

$$\cos \theta = \frac{A \cdot B}{\|A\| \|B\|}$$

Similarité cosinus

ou : mesure cosinus, *cosine similarity*

Similarité :

$$S_C(A, B) = \frac{A \cdot B}{\|A\| \|B\|}$$

Similarité cosinus

ou : mesure cosinus, *cosine similarity*

Similarité :

$$S_C(A, B) = \frac{A \cdot B}{\|A\| \|B\|}$$

Distance :

$$D_C(A, B) = 1 - S_C(A, B)$$

Similarité cosinus

ou : mesure cosinus, *cosine similarity*

Similarité :

$$S_C(A, B) = \frac{A \cdot B}{\|A\| \|B\|}$$

Distance :

$$D_C(A, B) = 1 - S_C(A, B)$$

On ne prend que les composants non-vides.

Textes : TF-IDF

- Vecteurs de fréquences de mots
- Fréquence \nRightarrow signification

Textes : TF-IDF

- Vecteurs de fréquences de mots
- Fréquence $\not\Rightarrow$ signification
- Term Frequency - Inverse Document Frequency

Textes : TF-IDF

$$TF_{ij} = \frac{f_{ij}}{\max_k f_{kj}} \qquad IDF_i = \log_2 \left(\frac{N}{n_i} \right)$$

$$TF-IDF_{ij} = TF_{ij} \cdot IDF_i$$

avec :

f_{ij} = fréquence de mot i dans document j

N = nombre de document

n_i = nombre de document dans lesquels on trouve terme i

Textes : TF-IDF

$$TF_{ij} = \frac{f_{ij}}{\max_k f_{kj}} \quad IDF_i = \log_2 \left(\frac{N}{n_i} \right)$$

$$TF\text{-}IDF_{ij} = TF_{ij} \cdot IDF_i$$

avec :

f_{ij} = fréquence de mot i dans document j

N = nombre de document

n_i = nombre de document dans lesquels on trouve terme i

IDF est une mesure de l'information que porte un mot.

TF-IDF nous dit quels mots caractérisent le mieux le document.

Textes : TF-IDF

$$TF_{ij} = \frac{f_{ij}}{\max_k f_{kj}} \qquad IDF_i = \log_2 \left(\frac{N}{n_i} \right)$$

$$TF-IDF_{ij} = TF_{ij} \cdot IDF_i$$

avec :

f_{ij} = fréquence de mot i dans document j

N = nombre de document

n_i = nombre de document dans lesquels on trouve terme i

IDF est une mesure de l'information que porte un mot.

TF-IDF nous dit quels mots caractérisent le mieux le document.

Variation : boolean, log, filtrage de mots vides

Content-Based Filtering

	I_1	I_2	I_3	I_4	I_5
U_1	3				
U_2			5	1	4
U_3		2		5	1

Plus de ce qui ressemble à ce que j'aime

Content-Based Filtering

	I_1	I_2	I_3	I_4	I_5
U_1	3				
U_2			5	1	4
U_3		2		5	1

Plus de ce qui ressemble à ce que j'aime

Et puis, partitionnement de données (regroupement, *clustering*), etc.

Content-Based Filtering

Basé sur les profils d'objets (*item profiles*)

- Plus stable (en principe)
- $O(n^2)$ (mais souvent moins, objets sans co-classement)
- Peut réduire le seuil
- Pré-calcul possible, requête (plus) rapide

Collaborative Filtering

	I_1	I_2	I_3	I_4	I_5
U_1	3		4	2	
U_2			5	1	4
U_3		2		5	1

Plus de ce que d'autres qui aiment ce que j'aime aiment.

Collaborative Filtering

	I_1	I_2	I_3	I_4	I_5
U_1	3				
U_2			5	1	4
U_3		2		5	1

Le Profil d'utilisateur.

Collaborative Filtering

	I_1	I_2	I_3	I_4	I_5
U_1	3		4	2	
U_2			5	1	4
U_3		2		5	1

Le Profil d'objet.

Utility Matrix Symmetry

- Proposer d'objets basés sur des utilisateurs
- Proposer d'utilisateurs basés sur des objets

Utility Matrix Symmetry

- Proposer d'objets basés sur des utilisateurs
- Proposer d'utilisateurs basés sur des objets

Mais, par exemple : 2 objets similaire $\not\equiv$ 2 utilisateurs similaires.

Utility Matrix Symmetry

- Proposer d'objets basés sur des utilisateurs
- Proposer d'utilisateurs basés sur des objets

Pour estimer $m_{u,i}$,

- Trouver k utilisateur comme U_u
- Trouver k objets comme I_i

Utility Matrix : Estimer $m_{u,i}$

	l_1	l_2	l_3	l_4	l_5
U_1	3		4	2	<input type="text"/>
U_2			5	1	4
U_3		2		2	3

- Trouver k utilisateur comme U_u , prendre $\frac{1}{k} \sum_{j=1}^k m_{u_j,i}$
- Trouver k objets comme l_i , prendre $\frac{1}{k} \sum_{j=1}^k m_{u,i_j}$

Utility Matrix : Estimer $m_{u,i}$

	l_1	l_2	l_3	l_4	l_5
U_1	3		4	2	<input type="text"/>
U_2			5	1	4
U_3		2		2	3

- Trouver k utilisateur comme U_u , prendre $\frac{1}{k} \sum_{j=1}^k m_{u_j,i}$
- Trouver k objets comme l_i , prendre $\frac{1}{k} \sum_{j=1}^k m_{u,i_j}$

Il faut calculer la ligne complète (ou la partie qui est susceptible d'être important)

Utility Matrix : Estimer $m_{u,i}$

	I_1	I_2	I_3	I_4	I_5
U_1	3		4	2	<div></div>
U_2			5	1	4
U_3		2		2	3

- Trouver k utilisateur comme U_u , prendre $\frac{1}{k} \sum_{j=1}^k m_{u_j,i}$
- Trouver k objets comme I_i , prendre $\frac{1}{k} \sum_{j=1}^k m_{u,i_j}$

Une fois pour U_u , les k autres nous donne un raccourcis.

Pour I_i , il faut calculer presque tous les I_j avant de pouvoir remplir une seule ligne. Par contre, objet-objet souvent plus fiable (exemple de genre d'objet / genre de personne).

Utility Matrix : Estimer $m_{u,i}$

	l_1	l_2	l_3	l_4	l_5
U_1	3		4	2	<input type="text"/>
U_2			5	1	4
U_3		2		2	3

- Trouver k utilisateur comme U_u , prendre $\frac{1}{k} \sum_{j=1}^k m_{u_j,i}$
- Trouver k objets comme l_i , prendre $\frac{1}{k} \sum_{j=1}^k m_{u,i_j}$

En tout cas, on fait la plupart en avance.

Utility Matrix

La matrice est creuse.

\Rightarrow clustering \Rightarrow matrice réduites

Utility Matrix

La matrice est creuse.

\implies clustering \implies matrice réduites

Estimation sur la matrice réduite, puis prendre d'objets et d'utilisateurs représentatif dans le *cluster* (paquet).

Amazon : Item-to-Item Collaborative Filtering

Observations :

Clustering coûte cher, diminue qualité

Amazon : Item-to-Item Collaborative Filtering

Observations :

Dimension reduction diminue qualité

Amazon : Item-to-Item Collaborative Filtering

Observations :

Utilisateurs touchent peu d'objets

Amazon : Item-to-Item Collaborative Filtering

Observations :

Réponse rapide désirable

Amazon : Item-to-Item Collaborative Filtering

Scale indépendamment du nombre d'utilisateur et du nombre d'objets

- Online
- Offline

G. Linden, B. Smith, J. York, *Amazon.com Recommendations: Item-to-Item Collaborative Filtering*, Internet Computing (7, 1), 22 Jan 2003.

Amazon : Item-to-Item Collaborative Filtering

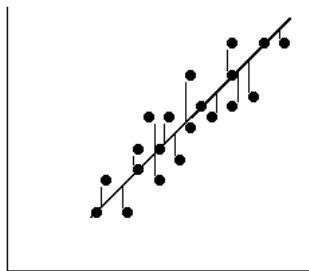
Offline (Precomputation)

```
for chaque objet  $l_1$  à vendre do  
  |  
  for chaque utilisateur  $C$  qui a acheté  $l_1$  do  
    |  
    for chaque objet  $l_2$  acheté par  $C$  do  
      |  
       $(l_1, l_2)++$   
    end  
  end  
  for chaque object  $l_2$  do  
    |  
     $S_{l_1, l_2} \leftarrow S(l_1, l_2)$   
  end  
end
```

Régression linéaire sur les avis des utilisateurs.

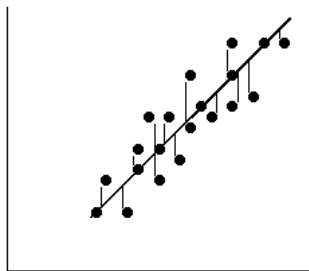
Daniel Lemire and Anna Maclachlan, *Slope One Predictors for Online Rating-Based Collaborative Filtering*, Proceedings of SIAM Data Mining (SDM) 2005.

Slope One : Régression



<http://www.upa.pdx.edu/IOA/newsom/pa551/Image255.gif>

Slope One : Régression



$$\min \sum (y_i - (ax_i + b))^2$$

<http://www.upa.pdx.edu/IOA/newsom/pa551/Image255.gif>

Slope One : algorithme

Offline :

for *chaque* l_i, l_j **do**

$\mathcal{U} \leftarrow \{\text{utilisateurs qui ont exprimé un avis sur } l_i, l_j\}$
 $\text{dev}_{i,j} \leftarrow \frac{1}{\|\mathcal{U}\|} \sum_{u \in \mathcal{U}} (r_u(i) - r_u(j))$

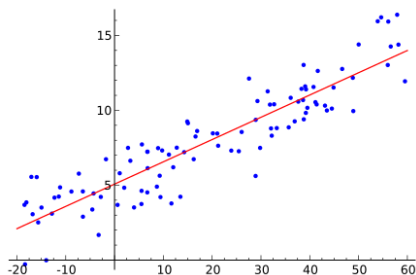
end

Online (pour u) :

$\mathcal{V} \leftarrow \{j \mid u \text{ a exprimé un avis sur } l_j\}$

$r_u(i) \leftarrow \frac{1}{\|\mathcal{V}\|} \sum_{u \in \mathcal{V}} (\text{dev}_{i,j} - r_u(j))$

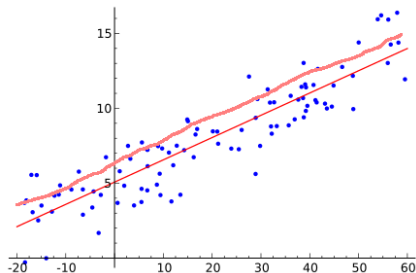
Slope One : Régression



"Linear regression" by Sewaqu - Own work. Licensed under Public domain via Wikimedia Commons -

http://commons.wikimedia.org/wiki/File:Linear_regression.svg#mediaviewer/File:Linear_regression.svg

Slope One : Régression



Dimensionality reduction

SVD, $k = 20 \dots 100$ typiquement

Décomposition en valeurs singulières

$$M = U \Sigma V^*$$

Dimensionality reduction

SVD, $k = 20 \dots 100$ typiquement

Décomposition en valeurs singulières

$$(a_1 \quad \cdots \quad a_m) \begin{pmatrix} b_1 \\ \vdots \\ b_n \end{pmatrix} = \text{scalaire}$$

Dimensionality reduction

SVD, $k = 20 \dots 100$ typiquement

Décomposition en valeurs singulières

$$\begin{pmatrix} a_1 \\ \vdots \\ a_m \end{pmatrix} (b_1 \quad \cdots \quad b_n) = \begin{pmatrix} c_{1,1} & \cdots & c_{1,n} \\ \vdots & & \vdots \\ c_{m,1} & \cdots & c_{m,n} \end{pmatrix}$$

Dimensionality reduction

SVD, $k = 20 \dots 100$ typiquement

Décomposition en valeurs singulières

$$\begin{pmatrix} a_{1,1} & a_{1,2} & a_{1,3} \\ \vdots & \vdots & \vdots \\ a_{m,1} & a_{m,2} & a_{m,3} \end{pmatrix} \begin{pmatrix} b_{1,1} & \cdots & b_{1,n} \\ b_{2,1} & \cdots & b_{2,n} \\ b_{3,1} & \cdots & b_{3,n} \end{pmatrix} = \begin{pmatrix} c_{1,1} & \cdots & c_{1,n} \\ \vdots & & \vdots \\ c_{m,1} & \cdots & c_{m,n} \end{pmatrix}$$

Dimensionality reduction

SVD, $k = 20 \dots 100$ typiquement

Décomposition en valeurs singulières

$$\begin{pmatrix} a_{1,1} & \cdots & a_{1,k} \\ \vdots & & \vdots \\ a_{m,1} & \cdots & a_{m,k} \end{pmatrix} \begin{pmatrix} c_{1,1} & \cdots & c_{1,n} \\ \vdots & & \vdots \\ c_{k,1} & \cdots & c_{k,n} \end{pmatrix} = \begin{pmatrix} c_{1,1} & \cdots & c_{1,n} \\ \vdots & & \vdots \\ c_{m,1} & \cdots & c_{m,n} \end{pmatrix}$$

Problèmes particuliers

- Comment mesurer la réussite ?
- Quelles sont des critères ?

Clustering

- kNN
- Curse of Dimensionality
- Scalabilité

Clustering

- kNN *k*-Nearest Neighbor
- Curse of Dimensionality Fléau (ou : malédiction) de la dimension
- Scalabilité 10^7 clients, 10^6 objets

Questions ?

Sondage : purple.com/devfest

Et si vous êtes dans la région, rdv sur meetup.com :

[Nantes Machine Learning Meetup](#)

github.com/JeffAbrahamson/talks/