

Filtrage collaboratif

Projet de fin de module “Développer un algorithme sur une infrastructure”

Certificat sciences des données et big data, 26 novembre 2020

Guiding lines

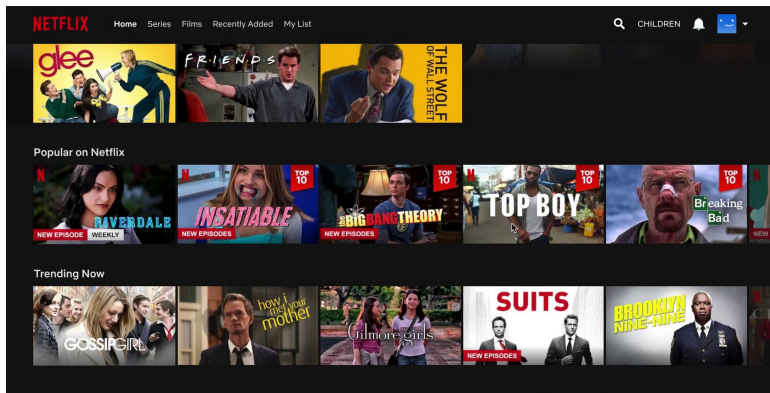
1 Présentation du problème

2 Résolution du problème

Netflix

Example

Le système de recommandation de Netflix permet de prédire l'évaluation d'un film par un utilisateur en se basant sur les évaluations de l'ensemble des utilisateurs.



Le filtrage collaboratif

Definition

- Le **filtrage collaboratif** construit des *systèmes de recommandation* de films en utilisant les *évaluations* d'un groupe pour aider un utilisateur.
- Un **système de recommandation** compare le *profil* d'un utilisateur à certaines *caractéristiques* d'un film pour prédire son *évaluation*.

Le filtrage collaboratif

Definition

- Le **filtrage collaboratif** construit des *systèmes de recommandation* de films en utilisant les *évaluations* d'un groupe pour aider un utilisateur.
- Un **système de recommandation** compare le *profil* d'un utilisateur à certaines *caractéristiques* d'un film pour prédire son *évaluation*.

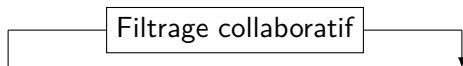
Hypothèse de proximité

- Des utilisateurs qui ont des profils proches auront des évaluations proches sur un même film.
- Des films qui ont des caractéristiques proches auront des évaluations proches pour un même utilisateur.

Hypothèse de proximité

- Des utilisateurs qui ont des profils proches auront des évaluations proches sur un même film.
- Des films qui ont des caractéristiques proches auront des évaluations proches pour un même utilisateur.

FC: Données -> Modèle

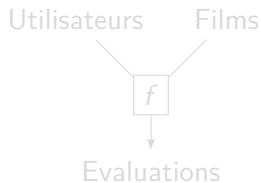


Données
(évaluations du groupe)

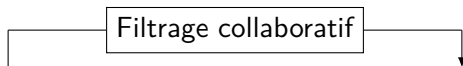
```

1::12::4
1::56::3
.
.
Util_Id :: Film_Id :: Eval
.
.
45257::117::1
  
```

Modèle
(système de recommandation)



FC: Données -> Modèle

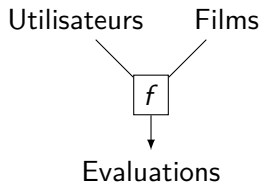


Données
(évaluations du groupe)

```

1::12::4
1::56::3
.
.
Util_Id :: Film_Id :: Eval
.
.
45257::117::1
  
```

Modèle
(système de recommandation)



Fonction de coût

Il faut ensuite sélectionner un *bon* modèle.

- La fonction de coût mesure la proximité du modèle aux données

$$J(f) = \sum_{(i,j,r) \in \text{Données}} (r - f(i,j))^2,$$

avec i l'identifiant de l'utilisateur, j l'identifiant du film (movie), r l'évaluation (rate)

Trop de solutions à ce problème,

il faut ajouter plus d'hypothèses sur f .

Fonction de coût

Il faut ensuite sélectionner un *bon* modèle.

- La fonction de coût mesure la proximité du modèle aux données

$$J(f) = \sum_{(i,j,r) \in \text{Données}} (r - f(i,j))^2,$$

avec i l'identifiant de l'utilisateur, j l'identifiant du film (movie), r l'évaluation (rate)

Trop de solutions à ce problème,

il faut ajouter plus d'hypothèses sur f .

Fonction de coût

Il faut ensuite sélectionner un *bon* modèle.

- La fonction de coût mesure la proximité du modèle aux données

$$J(f) = \sum_{(i,j,r) \in \text{Données}} (r - f(i,j))^2,$$

avec i l'identifiant de l'utilisateur, j l'identifiant du film (movie), r l'évaluation (rate)

Trop de solutions à ce problème,

il faut ajouter plus d'hypothèses sur f .

Profil utilisateur et caractéristiques de films

On peut supposer que,

- Un utilisateur i est associé à un **profil**

$$P_i \in \mathbb{R}^k,$$

contenant k coordonnées.

- Un film j est associé à ses **caractéristiques**

$$Q_j \in \mathbb{R}^k,$$

contenant k coordonnées.

- L'**évaluation prédite** est le produit scalaire du profil P_i de l'utilisateur i et des caractéristiques Q_j du film j

$$R_{i,j} = \langle P_i, Q_j \rangle = f(i,j),$$

Profil utilisateur et caractéristiques de films

On peut supposer que,

- Un utilisateur i est associé à un **profil**

$$P_i \in \mathbb{R}^k,$$

contenant k coordonnées.

- Un film j est associé à ses **caractéristiques**

$$Q_j \in \mathbb{R}^k,$$

contenant k coordonnées.

- L'**évaluation prédite** est le produit scalaire du profil P_i de l'utilisateur i et des caractéristiques Q_j du film j

$$R_{i,j} = \langle P_i, Q_j \rangle = f(i, j),$$

Profil utilisateur et caractéristiques de films

On peut supposer que,

- Un utilisateur i est associé à un **profil**

$$P_i \in \mathbb{R}^k,$$

contenant k coordonnées.

- Un film j est associé à ses **caractéristiques**

$$Q_j \in \mathbb{R}^k,$$

contenant k coordonnées.

- L'**évaluation prédite** est le produit scalaire du profil P_i de l'utilisateur i et des caractéristiques Q_j du film j

$$R_{i,j} = \langle P_i, Q_j \rangle = f(i,j),$$

Profil utilisateur et caractéristiques de films

On peut supposer que,

- Un utilisateur i est associé à un **profil**

$$P_i \in \mathbb{R}^k,$$

contenant k coordonnées.

- Un film j est associé à ses **caractéristiques**

$$Q_j \in \mathbb{R}^k,$$

contenant k coordonnées.

- L'**évaluation prédite** est le produit scalaire du profil P_i de l'utilisateur i et des caractéristiques Q_j du film j

$$R_{i,j} = \langle P_i, Q_j \rangle = f(i,j),$$

Hypothèse de proximité

- L'inégalité de Cauchy-Schwarz assure ainsi les hypothèses de proximité:

$$\begin{aligned} |R_{i_1,j} - R_{i_2,j}| &\leq \|Q_j\| \times \|P_{i_1} - P_{i_2}\|, \\ |R_{i,j_1} - R_{i,j_2}| &\leq \|P_i\| \times \|Q_{j_1} - Q_{j_2}\|. \end{aligned}$$

- L'ajout d'un terme de régularisation les renforce en favorisant les faibles norme:

$$J(P, Q) = \sum_{(i,j,r) \in \text{Données}} (r - \langle P_i, Q_j \rangle)^2 + \lambda (\|P_i\|^2 + \|Q_j\|^2),$$

$$\text{avec } P = \begin{bmatrix} P_1^\top \\ \vdots \\ P_m^\top \end{bmatrix}, \quad Q = \begin{bmatrix} Q_1^\top \\ \vdots \\ Q_n^\top \end{bmatrix}.$$

Hypothèse de proximité

- L'inégalité de Cauchy-Schwarz assure ainsi les hypothèses de proximité:

$$\begin{aligned} |R_{i_1,j} - R_{i_2,j}| &\leq \|Q_j\| \times \|P_{i_1} - P_{i_2}\|, \\ |R_{i,j_1} - R_{i,j_2}| &\leq \|P_i\| \times \|Q_{j_1} - Q_{j_2}\|. \end{aligned}$$

- L'ajout d'un terme de régularisation les renforce en favorisant les faibles norme:

$$J(P, Q) = \sum_{(i,j,r) \in \text{Données}} (r - \langle P_i, Q_j \rangle)^2 + \lambda (\|P_i\|^2 + \|Q_j\|^2),$$

$$\text{avec } P = \begin{bmatrix} P_1^\top \\ \vdots \\ P_m^\top \end{bmatrix}, \quad Q = \begin{bmatrix} Q_1^\top \\ \vdots \\ Q_n^\top \end{bmatrix}.$$

Guiding lines

1 Présentation du problème

2 Résolution du problème

Alternating Least Squares

$$J(P, Q) = \sum_{(i,j,r) \in \text{Données}} (r - \langle P_i, Q_j \rangle)^2 + \lambda (\|P_i\|^2 + \|Q_j\|^2),$$

- Non quadratique en (P, Q) . Mais si l'on fixe Q , elle est quadratique en P . Son minimiseur est

$$P^* = RQ (\lambda I + Q^T Q)^{-1}.$$

- Puisque J est symétrique, on peut à présent fixer $P = P^*$ et résoudre en Q , etc...

Projet

Vous utiliserez la fonction ALS de la librairie `pyspark.mllib`.

Alternating Least Squares

$$J(P, Q) = \sum_{(i,j,r) \in \text{Données}} (r - \langle P_i, Q_j \rangle)^2 + \lambda (\|P_i\|^2 + \|Q_j\|^2),$$

- Non quadratique en (P, Q) . Mais si l'on fixe Q , elle est quadratique en P . Son minimiseur est

$$P^* = RQ (\lambda I + Q^T Q)^{-1}.$$

- Puisque J est symétrique, on peut à présent fixer $P = P^*$ et résoudre en Q , etc...

Projet

Vous utiliserez la fonction ALS de la librairie `pyspark.mllib`.

Alternating Least Squares

$$J(P, Q) = \sum_{(i,j,r) \in \text{Données}} (r - \langle P_i, Q_j \rangle)^2 + \lambda (\|P_i\|^2 + \|Q_j\|^2),$$

- Non quadratique en (P, Q) . Mais si l'on fixe Q , elle est quadratique en P . Son minimiseur est

$$P^* = RQ (\lambda I + Q^T Q)^{-1}.$$

- Puisque J est symétrique, on peut à présent fixer $P = P^*$ et résoudre en Q , etc...

Projet

Vous utiliserez la fonction ALS de la librairie `pyspark.mllib`.

Descente de gradient (stochastique)

$$J(P, Q) = \sum_{(i,j,r) \in \text{Données}} l(r, P_i, Q_j),$$

$$l(r, p, q) = (r - \langle p, q \rangle)^2 + \lambda (\|p\|^2 + \|q\|^2).$$

- Pour chaque ligne (i, j, r) des données, SGD fait un pas dans la direction de plus grande pente de $l(r, P_i, Q_j)$ seulement

$$P_i^{(c+1)} = P_i^{(c)} - \alpha \nabla_p l(r, P_i^{(c)}, Q_j^{(c)}),$$

$$Q_j^{(c+1)} = Q_j^{(c)} - \alpha \nabla_q l(r, P_i^{(c)}, Q_j^{(c)}).$$

- Lorsque toutes les lignes sont traitées, une “époque” est conclue. On répète ensuite plusieurs époques.

Projet

Vous implémenterez cette descente de gradient.

Descente de gradient (stochastique)

$$J(P, Q) = \sum_{(i,j,r) \in \text{Données}} l(r, P_i, Q_j),$$

$$l(r, p, q) = (r - \langle p, q \rangle)^2 + \lambda (\|p\|^2 + \|q\|^2).$$

- Pour chaque ligne (i, j, r) des données, SGD fait un pas dans la direction de plus grande pente de $l(r, P_i, Q_j)$ seulement

$$P_i^{(c+1)} = P_i^{(c)} - \alpha \nabla_p l(r, P_i^{(c)}, Q_j^{(c)}),$$

$$Q_j^{(c+1)} = Q_j^{(c)} - \alpha \nabla_q l(r, P_i^{(c)}, Q_j^{(c)}).$$

- Lorsque toutes les lignes sont traitées, une “époque” est conclue. On répète ensuite plusieurs époques.

Projet

Vous implémenterez cette descente de gradient.

Descente de gradient (stochastique)

$$J(P, Q) = \sum_{(i,j,r) \in \text{Données}} l(r, P_i, Q_j),$$

$$l(r, p, q) = (r - \langle p, q \rangle)^2 + \lambda (\|p\|^2 + \|q\|^2).$$

- Pour chaque ligne (i, j, r) des données, SGD fait un pas dans la direction de plus grande pente de $l(r, P_i, Q_j)$ seulement

$$P_i^{(c+1)} = P_i^{(c)} - \alpha \nabla_p l(r, P_i^{(c)}, Q_j^{(c)}),$$

$$Q_j^{(c+1)} = Q_j^{(c)} - \alpha \nabla_q l(r, P_i^{(c)}, Q_j^{(c)}).$$

- Lorsque toutes les lignes sont traitées, une “époque” est conclue. On répète ensuite plusieurs époques.

Projet

Vous implémenterez cette descente de gradient.

Descente de gradient (stochastique)

$$J(P, Q) = \sum_{(i,j,r) \in \text{Données}} l(r, P_i, Q_j),$$

$$l(r, p, q) = (r - \langle p, q \rangle)^2 + \lambda (\|p\|^2 + \|q\|^2).$$

- Pour chaque ligne (i, j, r) des données, SGD fait un pas dans la direction de plus grande pente de $l(r, P_i, Q_j)$ seulement

$$P_i^{(c+1)} = P_i^{(c)} - \alpha \nabla_p l(r, P_i^{(c)}, Q_j^{(c)}),$$

$$Q_j^{(c+1)} = Q_j^{(c)} - \alpha \nabla_q l(r, P_i^{(c)}, Q_j^{(c)}).$$

- Lorsque toutes les lignes sont traitées, une “époque” est conclue. On répète ensuite plusieurs époques.

Projet

Vous implémenterez cette descente de gradient.

Aide pour le calcul des dérivées

Pour dériver par rapport à des vecteurs, il est parfois plus simple de revenir à la définition de la dérivée

$$l(r, p + \delta p, q) = l(r, p, q) + \left\langle \nabla_p l \left(P_i^{(c)}, Q_j^{(c)} \right), \delta p \right\rangle + o(\|\delta p\|)$$