

---

# Raisonnement analogique pour la recommandation : premières expérimentations

---

Nicolas Hug      Henri Prade      Gilles Richard

Institut de Recherche en Informatique de Toulouse, 31400 Toulouse  
{nicolas.hug, henri.prade, gilles.richard}@irit.fr

## Résumé

Les systèmes de recommandation ont pour vocation de fournir des suggestions intéressantes à leurs utilisateurs. On distingue deux principaux types d'approche pour construire un système de recommandation : les méthodes dites "*content-based*" et les méthodes dites *collaboratives*. Encouragés par les bons résultats obtenus par les techniques de classification basées sur les proportions analogiques, nous nous intéressons ici à la conception de techniques de recommandation elles aussi basées sur les proportions analogiques. La qualité d'un système de recommandation peut être évaluée selon différents aspects, parmi lesquels la *précision*, qui mesure la capacité d'un système à évaluer avec justesse l'intérêt d'un utilisateur pour un objet donné. D'autres dimensions sont aussi à prendre en compte, notamment la *couverture*, ou la capacité du système à faire découvrir des articles à la fois intéressants et qui n'auraient pas pu être découverts autrement (*surprise*). Dans ce papier, nous comparons deux approches basées sur l'analogie avec d'autres approches classiques du problème de la recommandation.

## Abstract

Recommender systems aim at providing suggestions of interest for end-users. Two basic types of approach underlie existing recommender systems, namely content-based methods and collaborative filtering. In this paper, encouraged by the good results obtained with analogical proportion-based classification techniques, we investigate the possibility of using analogy as the main underlying principle for implementing prediction algorithms. We study two variants of this idea. The quality of a recommender system can be estimated along diverse dimensions, among which the accuracy to predict user's rating for unseen items is clearly important. Still other dimensions like *coverage* and *surprise* are also of great interest. In this paper, we compare the proposed approach with well-known recommender systems.

## 1 Introduction

Dans un monde où le volume de données disponibles ne cesse de croître, les outils de filtrage automatique sont essentiels pour extraire efficacement de l'information et des connaissances. Dans le domaine du commerce électronique, les systèmes de recommandation (SR) jouent le rôle de moteur de recherche : ils filtrent les articles disponibles pour fournir aux clients des recommandations intéressantes. Avec l'essor du commerce en ligne, les systèmes de recommandation deviennent de plus en plus populaires [24, 1]. Ils aident à répondre à des questions aussi diverses que "quel film louer ?", "quel article acheter ?", ou encore "dans quel restaurant dîner ?". Pour ce faire, les SR fournissent aux utilisateurs une liste de recommandations. Evidemment, le système est d'autant meilleur que les recommandations sont personnalisées : un SR ne recommandant que les articles les plus populaires est vraisemblablement inutile, puisque les utilisateurs sont susceptibles de déjà connaître ces articles.

Un moyen très classique de fournir des recommandations pour un utilisateur cible est d'estimer son intérêt (ou son goût) pour les articles dont le système dispose. Le goût d'un utilisateur  $u$  pour un article  $i$  ( $i$  pour l'anglais *item*) est communément représenté par une note (ou score) que  $u$  donnerait à  $i$ . L'échelle des notes peut varier, mais l'intervalle entier  $[1, 5]$  et l'échelle binaire *aime/n'aime pas* sont très communs dans les systèmes réels.

Une fois que ces estimations sont calculées, une option simple est de recommander à l'utilisateur les articles pour lesquels l'estimation de sa note est la plus haute. On utilise ici l'hypothèse implicite que l'utilisateur sera intéressé par les articles avec un score élevé.

Avoir une idée précise de la qualité globale d'un SR n'est pas une chose triviale, et plusieurs points de vue

doivent être pris en compte (voir [24] pour une étude exhaustive).

Le raisonnement par analogie est largement reconnu comme un élément clef de l'intelligence humaine [8, 11, 9, 16]. C'est un moyen puissant pour établir des parallèles entre des objets a priori non apparentés, et ainsi inférer des relations ou des propriétés sur la base des similarités/dissimilarités observées. Aussi, il est envisageable de concevoir un système de recommandation utilisant le raisonnement par analogie comme moyen d'inférence dans le but de proposer des recommandations pertinentes, et si possible surprenantes. Nous avons choisi d'explorer cette piste dans ce papier.

Les "proportions analogiques" permettent d'exprimer une relation analogique en faisant intervenir quatre éléments  $a, b, c, d$ , où l'on pose que " $a$  est à  $b$  ce que  $c$  est à  $d$ ". Une telle proportion exprime le fait que les éléments  $a$  et  $b$  diffèrent de la même manière que les éléments  $c$  et  $d$  : cela permet une approche plus riche que de simplement considérer les plus proches voisins d'un élément, car on va chercher de l'information au-delà de leur voisinage. L'utilisation des proportions analogiques permet de concevoir des classificateurs très performants [2, 23, 5], mais l'analogie s'est aussi montrée efficace sur d'autres tâches d'intelligence artificielle [6, 21].

Les tâches de prédiction et de classification peuvent être considérées comme similaires, puisque les  $n$  notes possibles peuvent être vues comme autant de classes (à ceci près que dans le cas de la prédiction, ces "classes" sont ordonnées). En s'inspirant des travaux effectués en classification, mais aussi en essayant de s'adapter à ce qui semble être des spécificités propres aux SR, deux algorithmes basés sur l'analogie sont proposés dans ce papier, chacun reposant sur des définitions légèrement différentes du principe d'inférence analogique.

Notre papier est organisé comme suit. Dans la prochaine section, nous établissons une brève revue des techniques et outils pour la recommandation, ainsi que pour l'évaluation des SR. Dans la section 3, nous fournissons les prérequis nécessaires sur les proportions analogiques, et expliquons comment elles peuvent servir de principe d'inférence. Dans la section 4, nous nous intéressons à l'utilisation des proportions analogiques pour la conception d'un système de recommandation, en proposant deux implémentations. Dans la section 5, nous reportons les résultats obtenus sur une base de test et nous les comparons avec ceux de techniques existantes bien connues. Enfin, nous concluons et fournissons des pistes de recherche dans la section 6.

## 2 Prérequis sur les systèmes de recommandation

L'objectif d'un système de recommandation est de fournir à un utilisateur une liste personnalisée d'articles pertinents. Nous formalisons ici le problème.

### 2.1 Formalisation du problème de la recommandation

Soit  $U$  un ensemble d'utilisateurs et  $I$  un ensemble d'articles. Pour certaines paires  $(u, i) \in U \times I$ , on connaît la note  $r_{ui}$ , qui exprime l'intérêt que l'utilisateur  $u$  porte à l'article  $i$ .

Il est assez commun d'avoir  $r_{ui} \in [1, 5]$ , où 5 signifie un intérêt prononcé pour l'article  $i$ , 1 signifie un rejet et 3 une certaine indifférence, ou une note moyenne. Nous dénotons l'ensemble des notes connues du système par  $R$ . Bien sûr, dans les systèmes réels, le cardinal de  $R$  est très petit en comparaison du nombre total de notes possibles qui vaut  $|U| \times |I|$ , puisque de nombreuses notes sont inconnues. Nous notons  $U_i$  l'ensemble des utilisateurs qui ont noté l'article  $i$ , et symétriquement  $I_u$  représente l'ensemble des articles que  $u$  a notés.

Pour recommander des articles aux utilisateurs, un système de recommandation procède ainsi :

1. Avec un algorithme  $A$ , estimer les notes  $r_{ui}$  inconnues (c.à.d.  $r_{ui} \notin R$ ). Cette estimation  $A(u, i)$  est communément notée  $\hat{r}_{ui}$ .
2. Avec une stratégie de recommandation  $S$  et à la lumière des notes précédemment estimées, recommander des articles aux utilisateurs. Par exemple, une stratégie très basique mais assez courante est de suggérer à  $u$  les articles  $i$  pour lesquels  $\hat{r}_{ui}$  est le plus grand.

On distingue trois principales familles de systèmes de recommandation, chacune se distinguant par les techniques employées par l'algorithme de prédiction  $A$  : les systèmes basés sur le contenu (*content-based*), le filtrage collaboratif et les systèmes hybrides. Les méthodes hybrides sont en fait une combinaison des méthodes content-based et collaboratives. Nous nous intéresserons ici à ces deux dernières, sans trop nous attarder sur les méthodes content-based car elles se sont montrées moins performantes que les méthodes collaboratives.

### 2.2 Techniques content-based

Les algorithmes basés sur le contenu utilisent les métadonnées des utilisateurs et des articles pour estimer une note. Les métadonnées sont des informations externes qui peuvent être collectées, typiquement :

- pour les utilisateurs : le genre, l'âge, la profession, l'adresse, etc ;
- pour les articles : cela dépend bien sûr de la nature des articles, mais dans le cas de films on peut penser à leur genre, les acteurs principaux, le réalisateur, etc.

A partir de ces métadonnées, un système basé sur le contenu va essayer de trouver des articles similaires à ceux pour lequel l'utilisateur a déjà exprimé son intérêt (en donnant un score élevé par exemple). Cela suppose l'utilisation d'une mesure de similarité entre les articles. De nombreuses options sont envisageables pour une telle mesure. Elles ne seront pas discutées ici, car notre méthode est de nature collaborative.

Un inconvénient bien connu des techniques content-based est leur tendance à seulement recommander des articles que les utilisateurs connaissent déjà, et les recommandations sont ainsi peu surprenantes et manquent de diversité.

### 2.3 Le filtrage collaboratif

Les algorithmes de filtrage collaboratif reposent sur l'ensemble des notes connues  $R$  pour faire une prédiction : pour une prédiction  $r_{ui}$  ( $r_{ui} \notin R$ ),  $A$  produira  $\hat{r}_{ui}$  en se basant sur un sous-ensemble de  $R$  soigneusement sélectionné. La différence principale entre les méthodes collaboratives et les méthodes content-based est que dans les premières les métadonnées ne sont pas utilisées, et dans les secondes les seules notes qui sont prises en compte sont celles de l'utilisateur cible. Deux approches collaboratives populaires sont l'approche basée sur le voisinage et l'approche par factorisation de matrice, que nous décrirons ici dans leur forme la plus simple.

L'approche par voisinage est une technique extrêmement basique, mais qui se montre néanmoins très efficace. Pour estimer la note d'un utilisateur  $u$  pour un article  $i$ , on sélectionne  $N_i^k(u)$ , l'ensemble des  $k$  utilisateurs les plus proches de  $u$  et qui ont noté  $i$ . Là aussi, on a besoin d'une mesure de similarité entre les utilisateurs. L'estimation de  $r_{ui}$  est calculée comme suit :

$$\hat{r}_{ui} = \operatorname{aggr}_{v \in N_i^k(u)} r_{vi},$$

où l'agrégat est habituellement une moyenne pondérée par la similarité entre  $u$  et  $v$ . Une prédiction un peu plus sophistiquée, popularisée par [3] est calculée ainsi :

$$\hat{r}_{ui} = b_{ui} + \operatorname{aggr}_{v \in N_i^k(u)} (r_{vi} - b_{vi}),$$

où  $b_{ui}$  est un biais lié à l'utilisateur  $u$  et à l'article  $i$ . Le biais caractérise la tendance de  $u$  à donner des notes au dessus ou en dessous de la moyenne générale  $\mu$  et

la tendance des utilisateurs à noter  $i$  au dessus ou en dessous de  $\mu$ . Comme elle utilise le voisinage des individus pour produire une prédiction, cette technique tend à modéliser les relations locales sous-jacentes aux données.

Notons qu'il est tout à fait possible d'adopter un point de vue centré sur les articles plutôt que sur les utilisateurs. En effet, plutôt que de chercher des utilisateurs similaires à  $u$ , on pourrait plutôt chercher des articles similaires à  $i$ , ce qui conduira à des formules duales de celles ci-dessus.

L'approche par factorisation de matrice est fortement inspirée par la décomposition en valeurs singulières : on suppose l'existence de  $f$  facteurs (ou critères) dont la nature n'est pas nécessairement connue et qui déterminent la valeur de chaque note  $r_{ui}$ . Un utilisateur  $u$  est modélisé par un vecteur  $p_u \in \mathbb{R}^f$ , où chaque composante de  $p_u$  caractérise l'importance du facteur correspondant pour  $u$ . D'une façon similaire, un article  $i$  est modélisé par un vecteur  $q_i \in \mathbb{R}^f$ , où chaque composante de  $q_i$  caractérise à quel point  $i$  satisfait le critère en question. A partir de là, une prédiction  $\hat{r}_{ui}$  est calculée comme le produit scalaire des deux vecteurs  $p_u$  et  $q_i$  :

$$\hat{r}_{ui} = p_u^t \cdot q_i.$$

Une fois le nombre de facteurs  $f$  fixé, le problème est ici d'estimer les vecteurs  $p_u$  et  $q_i$  pour tous les utilisateurs et tous les articles. Deux méthodes sont fréquemment utilisées pour cela : la régression des moindres carrés alternée et la descente de gradient stochastique, popularisée par [7].

Contrairement à la méthode de voisinage, la factorisation de matrice a tendance à modéliser et capturer les aspects globaux et généraux des données. Les deux méthodes ont été combinées avec succès dans les travaux de [13].

### 2.4 L'évaluation des systèmes de recommandation

Avoir une idée précise de la qualité globale d'un SR n'est pas une chose évidente, et plusieurs points de vue doivent être pris en compte (voir [24] pour une étude exhaustive). La précision des prédictions de l'algorithme est capitale, mais d'autres propriétés sont requises pour déployer un système efficace. Par exemple la *couverture*, qui mesure l'étendue des articles du système qui peuvent faire l'objet d'une recommandation, ou encore la *surprise* qui renseigne sur le caractère inattendu d'une recommandation.

#### Exactitude

La performance de l'algorithme de prédiction  $A$  est généralement évaluée du point de vue de l'exactitude

(*accuracy* en anglais), qui mesure à quel point les estimations de notes  $\hat{r}_{ui}$  sont proches de leurs vraies valeurs  $r_{ui}$ . Pour évaluer l'exactitude d'un algorithme de prédiction, on utilise généralement la technique de validation croisée bien connue en apprentissage artificiel : l'ensemble  $R$  des notes est divisé aléatoirement en deux ensembles disjoints  $R_{appr}$  et  $R_{test}$ , et l'algorithme  $A$  prédit les notes de  $R_{test}$  en se basant uniquement sur les notes de l'ensemble d'apprentissage  $R_{train}$ .

La racine de la moyenne du carré des erreurs (RMSE pour *Root Mean Squared Error*), est un indicateur très commun de l'exactitude d'un algorithme. On la calcule comme suit :

$$\text{RMSE}(A) = \sqrt{\frac{1}{|R_{test}|} \sum_{r_{ui} \in R_{test}} (\hat{r}_{ui} - r_{ui})^2}.$$

Un autre indicateur de l'exactitude est la moyenne des erreurs absolues (MAE pour *Mean Absolute Error*), où les grandes erreurs ne sont pas plus pénalisantes que les petites :

$$\text{MAE}(A) = \frac{1}{|R_{test}|} \sum_{r_{ui} \in R_{test}} |\hat{r}_{ui} - r_{ui}|.$$

### Précision et rappel

La précision et le rappel permettent de mesurer la capacité d'un système à fournir des recommandations pertinentes. Dans ce qui suit, on notera  $I_S$  l'ensemble des articles que la stratégie  $S$  suggèrera à n'importe quel utilisateur, à partir des prédictions venant de l'algorithme  $A$ . Pour des notes appartenant à l'intervalle  $[1, 5]$ , une stratégie simple serait par exemple de recommander l'article  $i$  à l'utilisateur  $u$  si l'estimation  $\hat{r}_{ui}$  est supérieure à 4.

$$I_S = \{i \in I \mid \exists u \in U, \hat{r}_{ui} \geq 4\}.$$

Soit  $I_{pert}$  l'ensemble des articles qui sont *effective-ment* pertinents pour les utilisateurs (c.à.d. l'ensemble des articles qui auraient été recommandés si toutes les prédictions de  $A$  étaient exactes) :

$$I_{pert} = \{i \in I \mid \exists u \in U, r_{ui} \geq 4\}.$$

La précision du système est définie comme la fraction d'articles recommandés qui sont pertinents pour les utilisateurs :

$$\text{Précision} = \frac{|I_S \cap I_{pert}|}{|I_S|},$$

et le rappel est défini comme la fraction d'articles recommandés qui sont pertinents, parmi tous les articles pertinents possibles :

$$\text{Rappel} = \frac{|I_S \cap I_{relev}|}{|I_{relev}|}.$$

Si des prédictions précises sont cruciales, il est largement admis que cela reste insuffisant pour déployer un système de recommandation efficace. D'autres mesures sont nécessaires [15, 10, 12].

Par exemple, on peut naturellement attendre d'un système de recommandation qu'il soit non seulement précis, mais aussi surprenant, et être capable de recommander le plus d'articles possible.

### Couverture

La couverture, dans sa forme la plus simple, est utilisée pour mesurer la capacité d'un système à recommander une grande quantité d'articles. Il est assez commun en effet de tomber dans l'écueil classique qui consiste à ne recommander que des articles très populaires. Un tel système n'apporterait aucune information intéressante à ses utilisateurs.

La couverture peut être définie comme la proportion d'articles recommandés par rapport à tous les articles du système :

$$\text{Coverage} = \frac{|I_S|}{|I|}.$$

### Surprise

Les utilisateurs attendent d'un système de recommandation qu'il soit surprenant, c'est à dire qu'il suggère des articles qu'ils n'auraient pas pu découvrir sans son aide. Selon les travaux de [12], la surprise d'un système de recommandation peut être évaluée avec l'aide de l'information mutuelle ponctuelle (PMI pour *Point-wise Mutual Information*). La PMI entre deux articles  $i$  et  $j$  est définie ainsi :

$$\text{PMI}(i, j) = -\log_2 \frac{p(i, j)}{p(i)p(j)} / \log_2 p(i, j),$$

où  $p(i)$  et  $p(j)$  représentent les probabilités pour les articles  $i$  et  $j$  d'être notés par n'importe quel utilisateur, et  $p(i, j)$  est la probabilité que  $i$  et  $j$  soient notés ensemble :  $p(i) = \frac{|U_i|}{|U|}$  et  $p(i, j) = \frac{|U_i \cap U_j|}{|U|}$  (on rappelle que  $U_i$  est l'ensemble des utilisateurs qui ont noté l'article  $i$ ). Pour estimer la surprise apportée par la recommandation d'un article  $i$  à un utilisateur  $u$ , nous avons deux choix :

- soit prendre le maximum des valeurs de PMI pour  $i$  et tous les autres articles notés par  $u$  :

$$\text{Surp}^{max}(u, i) = \max_{j \in I_u} \text{PMI}(i, j)$$

- soit prendre la moyenne de ces valeurs de PMI :

$$\text{Surp}^{avg}(u, i) = \frac{\sum_{j \in I_u} \text{PMI}(i, j)}{|I_u|}$$

Alors, la capacité globale du système à surprendre ses utilisateurs est la moyenne de toutes les valeurs de surprise sur l'ensemble des prédictions.

### 3 Prérequis sur les proportions analogiques

Une proportion analogique “ $a$  est à  $b$  ce que  $c$  est à  $d$ ” exprime une relation analogique entre les paires  $(a, b)$  et  $(c, d)$ , ainsi qu’entre les paires  $(a, c)$  et  $(b, d)$ . On peut trouver de nombreux exemples naturels de telles proportions analogiques, comme par exemple “le veau est à la vache ce que le poulain est à la jument”, ou encore “le pinceau est au peintre ce que la craie est au professeur”. Cependant, ce n’est que récemment que des définitions formelles ont été proposées pour les proportions analogiques, dans des cadres différents [26, 14, 18]. Dans cette section, nous donnons un bref résumé de la formalisation des proportions analogiques qui nous aideront à décrire nos algorithmes par la suite. Pour une introduction générale, nous invitons le lecteur à se référer à [20, 19, 22].

#### 3.1 Cadre formel

Il est entendu depuis l’époque d’Aristote qu’une proportion analogique  $T$  est une relation quaternaire qui satisfait les trois axiomes suivants :

1.  $T(a, b, a, b)$  (reflexivité)
2.  $T(a, b, c, d) \implies T(c, d, a, b)$  (symétrie)
3.  $T(a, b, c, d) \implies T(a, c, b, d)$  (permutation centrale)

Il existe plusieurs modèles de proportions analogiques, dépendant du domaine de définition. Lorsque le domaine est défini,  $T(a, b, c, d)$  s’écrit plus simplement  $a : b :: c : d$ . Dans ce papier, on s’intéresse à évaluer des proportions analogiques sur des notes, qui peuvent être soit booléennes (*aime/n’aime pas*), soit dans notre cas d’étude des valeurs entières (dans l’intervalle  $[1, 5]$ ). L’idée de proportion analogique est similaire à celle de proportion numérique pour les nombres :

- Domaine  $\mathbb{R}$  :  
 $a : b :: c : d \iff \frac{a}{b} = \frac{c}{d} \iff ad = bc$   
(proportion géométrique).
- Domaine  $\mathbb{R}$  :  
 $a : b :: c : d \iff a - b = c - d \iff a + d = b + c$   
(proportion arithmétique).
- Domaine  $\mathbb{R}^n$  :  
 $\vec{a} : \vec{b} :: \vec{c} : \vec{d} \iff \vec{a} - \vec{b} = \vec{c} - \vec{d}$ . C’est simplement une extension de la proportion arithmétique aux vecteurs réels. Dans ce cas, les quatre vecteurs  $\vec{a}, \vec{b}, \vec{c}, \vec{d}$  forment un parallélogramme.
- Domaine  $\mathbb{B} = \{0, 1\}$  :  
 $a : b :: c : d \iff (a \wedge d \equiv b \wedge c) \wedge (a \vee d \equiv b \vee c)$ .
- Domaine  $\mathbb{B}^n$  :  
 $\vec{a} : \vec{b} :: \vec{c} : \vec{d} \iff \forall i \in [1, n], a_i : b_i :: c_i : d_i$ .  
C’est une extension directe du cas booléen.

D’autres définitions existent pour les matrices, les concepts formels, les treillis, etc. (voir [26, 14, 17]).

#### Résolution d’équation

A partir de trois éléments  $a, b, c$ , le problème de la résolution d’équation consiste à trouver un quatrième élément  $x$  de telle sorte que la proportion  $a : b :: c : x$  soit valide.

Dans le cas des proportions booléennes, le problème n’est pas toujours soluble : bien qu’il soit facile de compléter  $1 : 0 :: 1 : x$  pour construire une proportion valide, il n’existe pas de solution à l’équation  $1 : 0 :: 0 : x$ <sup>1</sup>.

#### Inférence analogique

Le raisonnement analogique peut être vu comme un moyen d’inférer de nouvelles connaissances à partir de proportions analogiques observées. Nous formalisons ici ce raisonnement dans le cas où les éléments  $\vec{a}, \vec{b}, \vec{c}, \vec{d}$  sont des vecteurs de  $\mathbb{R}^n$  ou de  $\mathbb{B}^n$ . Le *saut analogique* est un principe d’inférence non sûr qui postule que pour quatre vecteurs  $\vec{a}, \vec{b}, \vec{c}, \vec{d}$  tels que la proportion est vraie sur certaines composantes, alors elle reste vraie aussi sur les composantes restantes. On peut formuler cela de la manière suivante (où  $\vec{a} = (a_1, a_2, \dots, a_n)$  et  $J \subset [1, n]$ ) :

$$\frac{\forall j \in J, a_j : b_j :: c_j : d_j}{\forall i \in [1, n] \setminus J, a_i : b_i :: c_i : d_i} \quad (\text{inférence analogique})$$

Ce principe amène à une règle de prédiction dans le contexte suivant :

- On a 4 vecteurs  $\vec{a}, \vec{b}, \vec{c}, \vec{d}$  où  $\vec{d}$  n’est que partiellement connu : seules les composantes de  $\vec{d}$  dont les indices sont dans  $J$  sont connues.
- En utilisant l’inférence analogique, on peut prédire les composantes manquantes de  $\vec{d}$  en résolvant l’ensemble des équations (dans le cas où elles sont solubles)

$$\forall i \in [1, n] \setminus J, a_i : b_i :: c_i : d_i = 1.$$

Dans le cas où les éléments sont tels que leur dernière composante est une classe (ou label), appliquer ce raisonnement sur  $\vec{d}$  revient à prédire une classe candidate pour  $\vec{d}$ . Lorsque l’on peut trouver plusieurs triplets  $\vec{a}, \vec{b}, \vec{c}$  qui satisfont la proportion analogique sur les indices dans  $J$ , il est possible d’obtenir plusieurs classes candidates et il est alors nécessaire de faire un choix (par une procédure de vote ou n’importe quelle technique d’agrégation), afin d’obtenir une seule et unique classe. Comme évoqué dans l’introduction, ce principe d’inférence a été appliqué avec succès à des problèmes de classification.

1. Dans le cadre booléen, la proportion  $a : b :: c : d$  est vraie pour seulement six quadruplets (contre  $2^4$  possibles).

## 4 Recommandation analogique

Nous avons développé deux algorithmes pour la recommandation, chacun étant inspiré par deux vues différentes du principe d'inférence analogique. Dans les deux cas, l'idée principale est que si une proportion analogique est satisfaite entre trois utilisateurs  $a, b, c, d$  (c.à.d. que pour chaque article  $j$  qu'ils ont noté en commun la proportion analogique  $r_{aj} : r_{bj} :: r_{cj} : r_{dj}$  est vraie), alors la proportion doit aussi être vraie pour un film  $i$  que  $a, b, c$  ont noté mais que  $d$  n'a pas noté : la valeur  $r_{di}$  est la composante manquante. Cela nous conduit à estimer  $r_{di}$  comme la solution  $\hat{r}_{di}$  de l'équation analogique suivante :

$$r_{ai} : r_{bi} :: r_{ci} : x$$

Etant donnée une paire  $(u, i)$ , telle que  $r_{ui} \notin R$  (on ne connaît pas la note de  $u$  pour  $i$ ) la procédure principale est la suivante :

1. Trouver l'ensemble des triplets d'utilisateurs  $a, b, c$  tels qu'une proportion analogique est vraie entre  $a, b, c$ , et  $u$  et tels que l'équation

$$r_{ai} : r_{bi} :: r_{ci} : x$$

est soluble.

2. Résoudre l'équation  $r_{ai} : r_{bi} :: r_{ci} : x$ , et considérer que la solution  $r$  est une note candidate pour  $r_{ui}$ .
3. Définir  $\hat{r}_{ui}$  comme un agrégat de toutes les notes candidates.

La première étape stipule simplement que dans l'espace vectoriel de dimension  $|I_{abcu}|$ , les utilisateurs  $a, b, c$  et  $u$ , considérés comme des vecteurs de notes sur  $I_{abcu}$ , forment un parallélogramme. Evidemment, cette condition est relativement stricte et il peut être nécessaire de la relaxer, en permettant au parallélogramme de se déformer. On peut donc remplacer la condition 3 par  $\|(a-b) - (c-d)\| \leq \lambda$ , où  $\lambda$  est un seuil adéquat et où  $\|\cdot\|$  correspond à la norme euclidienne.

Une autre modification de l'algorithme serait de chercher les utilisateurs  $a, b$  et  $c$  uniquement dans un sous-ensemble de  $U$ . On peut penser à l'ensemble des  $k$  plus proches voisins de  $d$ , en utilisant l'hypothèse que ce sont les voisins de  $d$  qui sont les plus à même d'estimer correctement sa note pour  $i$ . Cela permet aussi de réduire la taille de l'espace de recherche des triplets et ainsi le temps d'exécution.

Evidemment, l'analogie peut être appliquée avec une approche basée sur les articles plutôt que sur les utilisateurs, comme c'est le cas pour les techniques classiques. Les deux approches seront étudiées dans l'expérimentation.

### 4.1 Algorithme utilisant les proportions analogiques

Ici, on utilise la définition de la proportion analogique basée sur la proportion arithmétique :

$$a : b :: c : d \iff a - b = c - d.$$

Etendue aux vecteurs, cette définition indique que  $(\vec{a}, \vec{b}, \vec{c}, \vec{d})$  sont les sommets d'un parallélogramme. Une implémentation directe de la procédure ci-dessus mène à l'algorithme 1, où  $I_{abcu}$  désigne l'ensemble des articles  $j$  que les utilisateurs  $a, b, c$  et  $u$  ont conjointement notés ( $r_{aj} \in R, r_{bj} \in R, r_{cj} \in R$  et  $r_{uj} \in R$ ).

---

#### Algorithm 1

---

**Entrée :** Un ensemble de notes connues  $R$ , un utilisateur  $u$  et un article  $i$  tels que  $r_{ui} \notin R$

**Sortie :**  $\hat{r}_{ui}$  : une estimation de  $r_{ui}$

**Initialisation :**

$C = \emptyset$  // liste de solutions candidates

**for all** utilisateurs  $\vec{a}, \vec{b}, \vec{c}$  voisins de  $u$  tels que

1.  $r_{ai} \in R, r_{bi} \in R, r_{ci} \in R$

2.  $r_{ai} - r_{bi} = r_{ci} - x$  est soluble

3.  $\|(\vec{a}-\vec{b}) - (\vec{c}-\vec{d})\| \leq \lambda$  // l'analogie est presque vérifiée entre  $a, b, c$  et  $u$

**do**

$x \leftarrow r_{ci} - r_{ai} + r_{bi}$

$C \leftarrow C \cup \{x\}$  // ajouter  $x$  comme une solution candidate

**end for**

$\hat{r}_{ui} = \text{aggr } x$   
 $x \in C$

---

Nous avons considéré une condition stricte de résolubilité pour l'équation  $r_{ai} - r_{bi} = r_{ci} - x$  : comme le résultat exact  $x = r_{ci} + r_{bi} - r_{ai}$  n'appartient pas nécessairement à l'intervalle  $[1, 5]$ , nous considérons l'équation soluble uniquement lorsque  $r_{ai} = r_{bi}$  ou  $r_{ai} = r_{ci}$ . Cela assure que la solution sera bien dans  $[1, 5]$ .

Chaque utilisateur d'un triplet  $(a, b, c)$  est représenté comme un vecteur de notes de dimension  $m$ . Il est clair que cette dimension  $m$  varie en fonction du triplet considéré, puisqu'il est très probable que les articles conjointement notés par les utilisateurs de deux triplets distincts soient eux aussi différents. Aussi, il est nécessaire que le seuil  $\lambda$  soit une fonction de  $m$ , car l'étendue des valeurs que  $\|\cdot\|$  peut prendre dépend de  $m$ . Nous avons observé expérimentalement que  $\lambda = \frac{3}{2} \cdot \sqrt{m}$  permettait d'obtenir des résultats optimaux.

## 4.2 Algorithme avec principe d'inférence reposant sur des motifs

Une idée intuitive sous-jacente à la recommandation par filtrage collaboratif est que des utilisateurs ayant exprimé des goûts similaires sur plusieurs articles devraient, de même, avoir des goûts similaires sur de nouveaux articles. Cependant, on peut observer que dans un groupe d'utilisateurs ayant des goûts similaires, les avis peuvent diverger sur certains articles donnés.

Avec l'algorithme précédent, il peut se produire la situation (non désirée) suivante :

- i quatre utilisateurs  $a, b, c, u$  ont des goûts assez similaires sur tous les articles qu'ils ont conjointement notés (avec peut-être quelques divergences négligeables et permises par la déformation du parallélogramme), alors que
- ii l'équation à résoudre  $r_{ai}, r_{bi}, r_{ci}, x$  où  $i$  est l'article que  $u$  n'a pas noté est telle que  $r_{ai}$  est à peu près égal à  $r_{ci}$  mais diffère fortement de  $r_{bi}$  (par exemple  $1 : 4 :: 1 : x$ , ou encore  $5 : 2 :: 5 : x$ ).

Dans ce cas précis, le motif de l'équation à résoudre ne se retrouve pas dans les analogies construites par les articles conjointement notés, qui sont du type  $1 : 1 :: 1 : 1$ ,  $4 : 4 :: 4 : 5$ , etc.

Or, il semble naturel de souhaiter éviter le genre de situation où le motif de l'équation à résoudre ne se retrouve pas (ou peu) parmi les composantes. En effet, dans ce cas, on peut considérer que l'on ne dispose pas suffisamment d'information pertinente pour appliquer l'inférence analogique puisque l'on n'a pas observé sur les composantes un motif identique à celui de l'équation à résoudre.

Cette observation amène à la conception d'un second algorithme où le principe d'inférence est légèrement modifié, comme expliqué ci-dessous.

Tout comme pour l'algorithme précédent, on cherche des triplets d'utilisateurs  $a, b, c$  tels que l'équation analogique  $r_{ai} : r_{bi} :: r_{ci} : x$  soit soluble.

Pour chaque article  $j \in I_{abcu}$ , on assigne le quadruplet  $(r_{aj}, r_{bj}, r_{cj}, r_{uj})$  à trois catégories :

1. *accord* :  $(r_{aj}, r_{bj}, r_{cj}, r_{uj}) \in \text{Cat 1}$  ssi trois d'entre eux sont égaux et le quatrième diffère avec au plus un écart de 1.  
Ex :  $(4, 4, 4, 4) \in \text{Cat 1}$ ,  $(4, 3, 4, 4) \in \text{Cat 1}$ , mais  $(4, 3, 4, 3) \notin \text{Cat 1}$
2. *Changement dans la même direction* :  
 $(r_{aj}, r_{bj}, r_{cj}, r_{uj}) \in \text{Cat 2}$  ssi
  - (a)  $r_{aj}$  diffère de  $r_{bj}$  avec une différence d'au moins 2, et de même pour  $r_{cj}$  et  $r_{uj}$  ;
  - (b) le changement entre  $r_{aj}$  et  $r_{bj}$  est fait dans la même direction que celui entre  $r_{cj}$  et  $r_{uj}$ .

3. *autres* (Cat 3) : tous les quadruplets qui n'appartiennent pas aux deux premières catégories appartiennent à la troisième.

Il est clair que lorsque  $|\text{Cat 3}|$  est négligeable par rapport à  $|\text{Cat 1}| + |\text{Cat 2}|$ , la proportion analogique est vraie (au moins approximativement) entre les quatre ensembles de notes.

## Résolution d'équation et filtrage des solutions candidates

On observe trois cas différents, selon le motif de l'équation analogique  $r_{ai} : r_{bi} :: r_{ci} : x$  :

1.  $a : a :: a : x \implies x = a$ .  
 $x$  est candidat ssi  $|\text{Cat 1}| \geq |\text{Cat 2}| + |\text{Cat 3}|$ .
2.  $a : b :: a : x, |a - b| \geq 2 \implies x = b$ .  
 $x$  est candidat ssi  $|\text{Cat 2}| > |\text{Cat 3}|$ .
3.  $a : b :: a : x, |a - b| = 1 \implies x = b$ .  
 $x$  est candidat ssi  $|\text{Cat 1}| \geq |\text{Cat 2}| + |\text{Cat 3}|$  ou  $|\text{Cat 2}| > |\text{Cat 3}|$ .

Là aussi, toutes les solutions candidates sont agrégées pour obtenir une seule et unique estimation  $\hat{r}_{ui}$ .

## 5 Expérimentations et résultats

Nous avons implémentés en Python et testés nos deux algorithmes, auxquels nous nous référons par *Paral* et par *Pattern*. Pour ces deux algorithmes, nous cherchons les utilisateurs  $a, b, c$  dans l'ensemble des  $k$  plus proches voisins de  $u$  avec  $k = 20$ . Ils sont comparés avec les algorithmes de voisinage décrits dans la section 2.3, auxquels nous nous référons par *k-NN* pour le modèle de base et *k-NNBiais* pour le modèle sophistiqué utilisant un biais. Pour ces deux algorithmes, nous avons choisi d'utiliser  $k = 40$ .

Pour chaque algorithme, nous avons calculé les mesures suivantes (décrites dans la section 2.4) :

- L'exactitude mesurée par la RMSE : une bonne exactitude correspond à une valeur de RMSE proche de 0.
- Précision et rappel : de bonnes valeurs correspondent à un pourcentage élevé.
- Couverture : idem.
- Surprise : ce sont en fait des mesures de non-surprise. Aussi, plus le nombre est faible, plus les recommandations sont surprenantes. Ces mesures sont notées  $S_{max}$  et  $S_{avg}$ .
- Le temps utilisateur nécessaire pour effectuer les recommandations (on ne prend pas en compte le temps d'apprentissage qui est le même pour les différents algorithmes).
- Pourcentage de prédictions impossibles.

Pour chaque algorithme, la stratégie de recommandation  $S$  est de recommander l'article  $i$  à l'utilisateur  $u$  si  $\hat{r}_{ui} \geq 4$ .

## 5.1 Jeu de données et protocole d'évaluation

### Jeu de données

Nous avons comparé les algorithmes sur la base de données Movielens-100K<sup>2</sup>, qui contient 100.000 notes venant de 1000 utilisateurs pour 1700 films. Chaque note appartient à l'intervalle  $[1, 5]$ . La base de données propose des métadonnées sur les films et les utilisateurs, mais nous ne les avons pas utilisées.

### Protocole d'évaluation

Afin d'obtenir des mesures légitimes, nous avons procédé à une campagne de validation croisée à cinq plis : pour chaque pli, l'ensemble des notes  $R$  est scindé aléatoirement en deux sous-ensembles disjoints  $R_{appr}$  et  $R_{test}$ , où  $R_{appr}$  contient cinq fois plus de notes que  $R_{test}$ . Les mesures reportées sont des moyennes sur les cinq plis.

## 5.2 Résultats et analyse

La table 5.2 illustre les performances des algorithmes. Des expériences similaires ont été menées en raisonnant sur des articles plutôt que sur des utilisateurs, montrant des résultats très semblables mais légèrement moins bons en RMSE (environ 5% plus grand).

	RMSE	Prec	Rap	Couv	$S_{max}$	$S_{avg}$	Time	Imp
Paral	.898	89.1	43.3	31.2	0.433	0.199	2h	1.1
Pattern	.927	84.9	50.0	47.7	0.433	0.198	5h	1.2
$k$ -NN	.894	89.1	44.1	27.8	0.432	0.198	10s	0.1
$k$ -NN <i>Biais</i>	.865	88.4	44.0	44.7	0.431	0.199	10s	0.1

TABLE 1 – Performance des algorithmes

Comme attendu, l'algorithme  $k$ -NN *Biais* est plus performant que l'algorithme de filtrage collaboratif basique  $k$ -NN. Cependant, ces deux algorithmes classiques sont supérieurs aux algorithmes analogiques proposés en ce qui concerne la RMSE. Il semble néanmoins qu'une certaine marge de progression soit envisageable pour ces deux nouveaux algorithmes, à la condition d'une analyse minutieuse de leur comportement. Pour les mesures autres que la RMSE, les résultats obtenus sont assez proches pour les quatre algorithmes. En ce qui concerne la surprise, qui est une notion délicate à quantifier, on peut se demander si la mesure utilisée ici est vraiment appropriée.

Des deux algorithmes analogiques, c'est étonnamment le premier qui donne les meilleurs résultats. Ceci

peut être dû au fait que le second est trop permissif dans la manière où les proportions analogiques sont satisfaites. De plus, le nombre de prédictions impossibles est supérieur, puisque les conditions de résolubilité des équations analogiques sont plus strictes. Une option envisageable serait d'hybrider les deux procédures.

Les deux algorithmes analogiques souffrent de leur complexité cubique, caractéristique de ce type d'approche. Dans le cas des systèmes de recommandation où des millions d'utilisateurs et d'articles entrent en jeu, cela devient un inconvénient non négligeable qui devra être contourné d'une manière ou d'une autre.

## 6 Conclusion et futures pistes

Nous en sommes clairement à un stade d'études préliminaires sur l'approche analogique pour la prédiction dans les systèmes de recommandation, qui est un sujet qui n'a jamais été exploré auparavant. Nos premiers résultats ne sont pas meilleurs que ceux obtenus avec les approches standards. Même s'ils ne semblent pas en être trop éloignés, les différences restent suffisamment grandes pour avoir un impact sur l'expérience utilisateur du système. Cependant, il est intéressant d'observer que des approches basées sur des idées assez différentes mènent à des résultats comparables.

De plus, il doit être noté que le problème de la prédiction pour la recommandation, bien que similaire à un problème de classification (pour lequel les classifieurs analogiques sont performants), présente des différences significatives, puisque les notes des articles qui jouent ici le rôle de composantes descriptives peuvent être assez redondantes et en quelque sorte incomplètes pour fournir un profil cohérent d'un utilisateur. Ceci pourrait expliquer le fait que l'application d'une approche analogique est moins performante pour la recommandation que pour la classification.

Des méthodes basées sur la proportion analogique ont aussi été utilisées récemment pour prédire des valeurs booléennes manquantes dans des bases de données [4]. Le problème de la recommandation peut aussi être vu comme un problème de valeurs manquantes, mais ici la proportion de données inconnues est considérable, et les données ne sont pas nécessairement booléennes. Cela aussi suggère que la tâche de recommandation est plus difficile.

Il reste cependant un certain nombre de pistes à explorer, tel que comprendre dans quelles situations une approche analogique serait plus performante que les approches classiques, et dans quelles autres situations une autre approche est préférable.

Enfin, l'exploitation du potentiel créatif de l'analogie pour proposer des articles jamais considérés par un utilisateur mais possédant des caractéristiques com-

2. <http://grouplens.org/datasets/movielens/>



munes avec des articles qu'il apprécie [25] ainsi que l'exploitation de son pouvoir explicatif pour suggérer pourquoi un article peut être intéressant, restent entièrement à explorer.

## Références

- [1] G. Adomavicius and A. Tuzhilin. Toward the next generation of recommender systems : a survey of the state-of-the-art and possible extensions. *Knowledge and Data Engineering, IEEE Transactions on*, 17(6) :734–749, June 2005.
- [2] S. Bayouhd, L. Miclet, and A. Delhay. Learning by analogy : A classification rule for binary and nominal data. *Proc. Inter. Joint Conf. on Artificial Intelligence IJCAI07*, pages 678–683, 2007.
- [3] R. M. Bell and Y. Koren. Lessons from the netflix prize challenge. *SIGKDD Explor. Newsl.*, 9(2) :75–79, December 2007.
- [4] W. Correa Beltran, H. Jaudoin, and O. Pivert. Estimating null values in relational databases using analogical proportions. In *Information Processing and Management of Uncertainty in Knowledge-Based Systems - 15th International Conference, IPMU 2014, Montpellier, France, July 15-19, 2014, Proceedings, Part III*, pages 110–119, 2014.
- [5] Myriam Bounhas, Henri Prade, and Gilles Richard. Analogical classification : A new way to deal with examples. In *ECAI 2014 - 21st European Conference on Artificial Intelligence, 18-22 August 2014, Prague, Czech Republic*, volume 263 of *Frontiers in Artificial Intelligence and Applications*, pages 135–140. IOS Press, 2014.
- [6] W. Correa, H. Prade, and G. Richard. When intelligence is just a matter of copying. In *Proc. 20th Eur. Conf. on Artificial Intelligence, Montpellier, Aug. 27-31*, pages 276–281. IOS Press, 2012.
- [7] S. Funk. Netflix update : Try this at home. <http://sifter.org/simon/journal/20061211.html>, 2006.
- [8] D. Gentner. Structure-mapping : A theoretical framework for analogy. *Cognitive Science*, 7(2) :155–170, 1983.
- [9] D. Gentner, K. J. Holyoak, and B. N. Kokinov. *The Analogical Mind : Perspectives from Cognitive Science*. Cognitive Science, and Philosophy. MIT Press, Cambridge, MA, 2001.
- [10] J. L. Herlocker, J. A. Konstan, Loren G. Terveen, and J. T. Riedl. Evaluating collaborative filtering recommender systems. *ACM Trans. Inf. Syst.*, 22(1) :5–53, 2004.
- [11] D. Hofstadter and M. Mitchell. The Copycat project : A model of mental fluidity and analogy-making. In D. Hofstadter and The Fluid Analogies Research Group, editors, *Fluid Concepts and Creative Analogies : Computer Models of the Fundamental Mechanisms of Thought*, pages 205–267, New York, NY, 1995. Basic Books, Inc.
- [12] M. Kaminskas and D. Bridge. Measuring surprise in recommender systems. In P. Adamopoulos et al., editor, *Procs. of the Workshop on Recommender Systems Evaluation : Dimensions and Design (Workshop Programme of the 8th ACM Conference on Recommender Systems)*, 2014.
- [13] Yehuda Koren. Factor in the neighbors : Scalable and accurate collaborative filtering. *ACM Trans. Knowl. Discov. Data*, 4(1) :1 :1–1 :24, January 2010.
- [14] Y. Lepage. De l'analogie rendant compte de la commutation en linguistique. *Habilit. à Diriger des Recher., Univ. J. Fourier, Grenoble*, 2003.
- [15] S. M. McNee, J. Riedl, and J. A. Konstan. Being accurate is not enough : how accuracy metrics have hurt recommender systems. In G. M. Olson and R. Jeffries, editors, *Extended Abstracts Proceedings of the 2006 Conference on Human Factors in Computing Systems, CHI 2006, Montréal, Québec, Canada, April 22-27, 2006*, pages 1097–1101, 2006.
- [16] E. Melis and M. Veloso. Analogy in problem solving. In *Handbook of Practical Reasoning : Computational and Theoretical Aspects*. Oxford Univ. Press, 1998.
- [17] L. Miclet, N. Barbot, and B. Jeudy. Analogical proportions in a lattice of sets of alignments built on the common subwords in a finite language. In H. Prade and G. Richard, editors, *Computational Approaches to Analogical Reasoning - Current Trends*. Springer, 2013.
- [18] L. Miclet and H. Prade. Handling analogical proportions in classical logic and fuzzy logics settings. In *Proc. 10th Eur. Conf. on Symbolic and Quantitative Approaches to Reasoning with Uncertainty (ECSQARU'09), Verona*, pages 638–650. Springer, LNCS 5590, 2009.
- [19] H. Prade and G. Richard. Analogical proportions and multiple-valued logics. In L. C. van der Gaag, editor, *Proc. 12th Eur. Conf. on Symb. and Quantit. Appr. to Reasoning with Uncertainty (ECSQARU'13), Utrecht, Jul. 7-10*, LNAI 7958, pages 497–509. Springer, 2012.

- [20] H. Prade and G. Richard. From analogical proportion to logical proportions. *Logica Universalis*, 7(4) :441–505, 2013.
- [21] H. Prade and G. Richard. Picking the one that does not fit - a matter of logical proportions. In *Proc. 8th Conf. Europ. Soc. for Fuzzy Logic and Technology (EUSFLAT), Milano, Sept. 11-13*, 2013.
- [22] H. Prade and G. Richard. Homogenous and heterogeneous logical proportions. *IfCoLog J. of Logics and their Applications*, 1(1) :1–51, 2014.
- [23] H. Prade, G. Richard, and B. Yao. Classification by means of fuzzy analogy-related proportions. A preliminary report. In *Proc. 2d IEEE Int. Conf. Soft Comp. & Pattern Recog. (SocPar'10), Evry*. 2010, 297-302.
- [24] F. Ricci, L. Rokach, B. Shapira, and P.B. Kantor, editors. *Recommender Systems Handbook*. Springer, 2011.
- [25] T. Sakaguchi, Y. Akaho, K. Okada, T. Date, T. Takagi, N. Kamimaeda, M. Miyahara, and T. Tsunoda. Recommendation system with multi-dimensional and parallel-case four-term analogy. *IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 3137 – 3143, 2011.
- [26] F. Yvon and N. Stroppa. Formal models of analogical proportions. Technical report, Ecole Nationale Supérieure des Télécommunications, no D008, 2006.