AGROCAMPUS OUEST

☐ **CFR Angers**

☑ **CFR Rennes**

Année universitaire : 2015 - 2016

Spécialité : Agroalimentaire

Spécialisation : Statistiques Appliquées

Mémoire de Fin d'Études

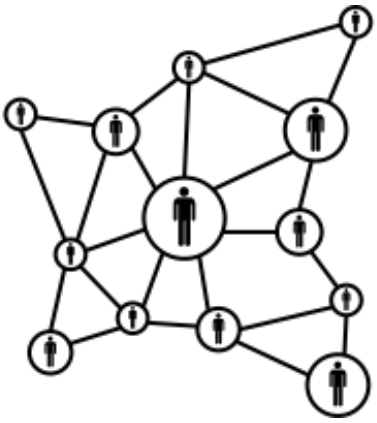
☑ d'Ingénieur de l'Institut Supérieur des Sciences agronomiques, agroalimentaires, horticoles et du paysage

☐ de Master de l'Institut Supérieur des Sciences agronomiques, agroalimentaires, horticoles et du paysage

☐ d'un autre établissement (étudiant arrivé en M2)

Approche comparative de l'imputation des données manquantes en recommandation de produits

Par : Anas EL KHALOUI



Soutenu à Rennes le 08/09/2015

Devant le jury composé de :

Président : Julie Josse

Maître de stage : Valérie Kaufmann

Enseignant référent : François Husson

SOMMAIRE

REMERCIEMENTS **2**

I. INTRODUCTION **3**

II. CONTEXTE DE L'ETUDE **4**

1. ENTREPRISE & MISSIONS **4**

2. LE CRM OU LE MARKETING ANALYTIQUE **4**

3. LA RECOMMANDATION DE PRODUITS, UN ENJEU DECISIF POUR L'ENTREPRISE COMMERCIALE **5**

III. LA RECOMMANDATION DE PRODUITS, UN PROBLEME DE DONNEES MANQUANTES **8**

1. RECOMMANDATION DE PRODUITS ET COMPLETION DE MATRICES **8**

2. CHOIX DES DONNEES & ETUDE PRELIMINAIRE **9**

3. OBJECTIFS OPERATIONNELS & SCIENTIFIQUES **9**

IV. UNE APPROCHE COMPARATIVE DE LA COMPLETION DE MATRICES **11**

1. TYPES ET STRUCTURATION DES DONNEES MANQUANTES **11**

A) TYPES DE DONNEES MANQUANTES 11
B) REPARTITION DES DONNEES MANQUANTES 11

2. CHOIX DES METHODES **12**

A) METHODE DES K-PLUS PROCHES VOISINS (K-NEAREST NEIGHBORS OU KNN) 13
B) FORETS ALEATOIRES (FORETS D'ARBRES DECISIONNELS) 13
C) ACP ITERATIVES REGULARISEES 14

3. METHODE DE COMPARAISON **15**

A) PRINCIPES 15
B) DONNEES 15
C) ALGORITHME 16

V. RESULTATS, DISCUSSION ET LIMITATIONS **16**

1. RESULTATS **16**

A) EFFET DU POURCENTAGE DE DONNEES MANQUANTES 16
B) EFFET DES LIAISONS LINEAIRES ENTRE VARIABLES 18
C) EFFET DU TYPE DE LIAISONS ENTRE VARIABLES : LIAISONS NON LINEAIRES 22
D) EFFET DU NOMBRE D'INDIVIDUS 26
E) EFFET DU NOMBRE DE VARIABLES 27

2. DISCUSSION ET LIMITATIONS **27**

A) DONNEES 27
B) TEMPS DE CALCUL 28
C) PERSPECTIVES 28

VI. CONCLUSION **31**

TABLE DES FIGURES **32**

BIBLIOGRAPHIE **33**

GLOSSAIRE **34**

Remerciements

Je remercie d'abord mon maître de stage Valérie Kaufmann, directrice des Études et du CRM du groupe BUT pour sa disponibilité sans faille, son attention permanente et le temps qu'elle a passé à me former. Je la remercie ensuite vivement pour la confiance qu'elle m'a accordée durant son absence. J'ai grâce à elle pu m'investir dans la découverte et la connaissance d'un secteur absolument passionnant.

Mes remerciements les plus vifs s'adressent ensuite aux membres de l'équipe CRM qui m'ont intégré et impliqué dans leurs projets dès les premiers jours de ma présence au sein de la société. Audrey Raimond et Tamara Szarmach, travailler avec vous est un véritable plaisir.

Un grand merci également à Juliette Laporte pour m'avoir impliqué dans des projets et opérations à l'échelle du groupe. Travailler avec toi est un défi permanent extrêmement enrichissant.

Ma reconnaissance s'adresse ensuite à toutes les personnes, salariés ou prestataires de BUT avec qui j'ai collaboré et qui ont partagé leur expertise avec moi : Sovansokha Suorm (Direction Exploitation BUT), Nicolas Imbert et Fabien Cagnet (SOGEC Datamark Services), Thierry Vallaud (BVA Data Science), etc.

I.Introduction

La prise de conscience de l'importance de la collecte et de l'exploitation de données commerciales a mené à la généralisation récente de bases de données orientées vers le marketing et centrées sur le consommateur. Ces bases ont la particularité de permettre l'identification et le suivi de manière unique de chaque client ou prospect quel que soit le canal (Magasin, web, mobile, etc.) qu'il utilise pour interagir avec l'entreprise. Cela a ouvert la porte à un nouveau marketing, basé sur la connaissance des particularités de chacun et dont le rôle est non plus simplement de relayer massivement les offres et les promotions, mais d'engager une véritable relation entre la marque et sa clientèle, durable et personnalisée dans laquelle le consommateur se sent écouté et estimé.

Ceci est particulièrement le cas dans le secteur de la distribution, où le contexte ultra concurrentiel et l'offre pléthorique imposent de penser à l'avance le parcours du client et de mettre en place des processus permettant le captage et la rétention de ce dernier tout en lui procurant satisfaction.

La Statistique intervient précisément à ce niveau-là, traditionnellement de manière descriptive dans le cadre de segmentations comme la RFM (Récence, fréquence, montant) ou la PMG (Petits, moyens, grands clients), permettant une bonne vision de sa base et un premier niveau de ciblage. Les percées récentes en matière de stockage, de traitement et de mise à disposition des données ainsi qu'une grande effervescence dans le secteur des « Business Analytics » ont cependant marqué le début d'une nouvelle ère qui est celle du Marketing prédictif. Temps passé sur la page d'un produit éventuellement convoité, historique d'achat, activité sur les réseaux sociaux, réaction aux emails, toutes les données d'un utilisateur présentent de précieux indices permettant aux entreprises de réagir à ses besoins et à ses envies en temps réel.

Mon arrivée au sein de l'entreprise BUT International a coïncidé avec une période de très fort développement sur les domaines relatifs aux données et à leur traitement. J'ai pu me familiariser avec différentes sources de données clients et utiliser différents outils pour les requêter, en extraire de l'information et fournir des réponses aux questions que se posent le management et les différents autres services. Cependant, j'ai décidé dans ce document de me limiter à une problématique particulière rencontrée par toute entreprise commercialisant ses produits sur internet, et de l'utiliser pour explorer un des champs les plus actifs de la Statistique, qui est celui de la donnée manquante.

II.Contexte de l'étude

1. Entreprise & missions

Dans ce qui suit, nous nous plaçons dans le cadre d'une entreprise de distribution d'équipement de la maison, qui opère sur un réseau de 300 magasins en France ainsi que sur internet via un site e-commerce. Les travaux présentés dans ce document s'articulent autour de la problématique de la recommandation personnalisée de produits aux clients, dans une optique d'amélioration de la qualité de l'offre et des ventes. En effet, le secteur de l'équipement de la maison se caractérise par un problème persistant de fidélisation des clients se traduisant par de très faibles fréquences d'achat et beaucoup d'inactivité. À titre d'exemple un client BUT fait en moyenne 1,3 achats seulement sur toute sa durée de vie et près de 14% des clients de la base sont inactifs depuis plus de 17 mois et donc considérés comme perdus. Un enjeu important est donc celui de trouver des manières efficaces d'animer sa base clients et de stimuler l'activité commerciale tout au long de l'année plutôt que suivant le calendrier classique des ventes (Soldes d'hiver et d'été, rentrée scolaire, etc.). Cette palette de tâches est attribuée au service CRM & Cross-Canal, qui doit s'adapter et évoluer pour retenir le client, devenu nomade et volatile.

2. Le CRM ou le marketing analytique

De plus en plus répandu dans les entreprises depuis le début des années 2000, le CRM (Consumer Relationship Management) développe une nouvelle manière de considérer la relation client, basée sur la collecte, l'historisation et la consolidation dans de grandes bases de données d'informations provenant de canaux différents : Magasin, applications mobile, sites e-commerce, navigation web, réseaux sociaux, centres d'appel, échanges d'e-mails, contacts avec le service après-vente, etc. Ces données sont ensuite exploitées pour mettre en place une gestion personnalisée, interactive, et hautement analytique du parcours du client. Tout cela suppose bien sûr la mise en place d'une infrastructure matérielle et logicielle bien pensée, alimentée et actualisée en permanence par des flux de données de sources diverses. À titre d'exemple, la base CRM BUT reçoit des données d'une quinzaine de sources différentes traitées puis stockées dans une base relationnelle dont la clef d'entrée est le client.

Le CRM a pour objectif principal d'optimiser et rallonger le cycle de vie du client en lui offrant le service le plus adapté à ses besoins via une connaissance approfondie, évolutive et aussi exhaustive que possible de celui-ci. Comme pour toute démarche de marketing, il s'agit avec le CRM d'augmenter les ventes et de générer de la marge, ce qui implique d'assortir les échanges avec le client d'offres pertinentes, l'idéal étant de lui proposer aussi précisément que possible ce dont il a besoin à l'instant où on le contacte, afin de susciter l'intérêt ou mieux encore, déclencher l'achat. Pour résumer, le CRM est pour une entreprise l'art d'optimiser ses interactions avec ses clients et ses prospects. Sachant qu'une base peut contenir plusieurs millions de clients (dix millions pour BUT), l'automatisation des processus est cruciale.

C'est là qu'intervient une famille diversifiée d'outils, en plein essor depuis quelques années et qui mobilise beaucoup de R&D en informatique et mathématiques appliquées : Les systèmes de recommandation.

3. La recommandation de produits, un enjeu décisif pour l'entreprise commerciale

Un système de recommandation est un outil logiciel qui a pour fonction de choisir et présenter des éléments d'information susceptibles d'intéresser l'utilisateur. Autrement dit, c'est un algorithme construit de manière à prédire la préférence d'un usager pour une entité quelconque qu'il n'a pas encore considéré. Cette entité peut être une personne ou une communauté comme c'est le cas sur les réseaux sociaux, ou alors un livre, un morceau de musique, un film ou encore un produit. Un exemple parlant est celui du site de vente en ligne Amazon qui suggère à ses clients des produits susceptibles de les intéresser selon leur historique d'achat, de navigation et ce qu'ils ont mis dans leur panier,. Le service Google quant à lui affiche des publicités ciblées selon l'historique de recherche et de navigation de l'internaute.

Les systèmes de recommandation sont bâtis sur une hypothèse importante selon laquelle les consommateurs sont similaires et qu'il existe des profils de clients. L'idée derrière ceci est de dire que si une personne *A* a la même opinion (ou les mêmes goûts) qu'une personne *B* à propos d'objet x, alors la personne *A* a plus de chance d'avoir la même opinion que *B* sur un autre objet y, plutôt que d'avoir la même opinion que quelqu'un choisi au hasard pour l'objet y. L'idée fondamentale est donc de supposer que si des utilisateurs ont partagés des intérêts identiques dans le passé, il y a de fortes chances pour qu'ils partagent aussi les mêmes goûts dans le futur. Bien sûr, plus les personnes A et B ont d'opinions communes, et plus le rapprochement se justifie.

Dans un marché très diversifié et en constante évolution, les solutions disponibles aujourd'hui pour recommander des produits sont de trois types :

- **Les solutions Open Source**, gratuites et adaptables au besoin, mais requérant un niveau certain de connaissances en programmation et bases de données. Elles s'appellent PredictionIO, Apache Mahout ou encore MyMediaLite et Python Crab. Elles ont besoin d'être paramétrées et fonctionnent grâce à d'autres modules Open Source comme Apache Spark (Plateforme de calcul distribué), MapReduce, etc. Certaines de ces solutions ont besoin d'une ou plusieurs machines pour fonctionner efficacement. Le configurateur a le choix des algorithmes à mettre en œuvre pour générer les recommandations, selon les données dont il dispose.

- **Les moteurs développés en interne**. Souvent les plus avancés, ils sont l'apanage des grands acteurs technologiques comme Netflix (Fig. 1), Google ou Facebook. Ils tournent sur des réseaux (clusters) de machines interconnectées appelées nœuds et sont capables de générer des millions de recommandations personnalisées en quasi temps-réel. Ayant été développés spécifiquement pour un type de recommandations et une structure de base de données, ces moteurs peuvent utiliser plusieurs approches et sont constamment améliorés. Ils sont bien sûr propriétaires et croisent plusieurs types de données : Notations, données sociodémographiques, historique d'achat, historique de navigation, panier abandonné, etc.

Figure 1 Exemples de recommandations affichées sur le site Netflix.com (Source : Netlix.com)

- **Les solutions commerciales « clés en main »** destinées à s'intégrer à un site web e-commerce comme Target2Sell, Pigdata, Nuukik, Ezako. Elles sont destinées à être mises en service de manière rapide et sans grands travaux et ont un niveau de complexité faible. Elles proposent en effet au client des produits identiques à ceux qu'il a déjà consultés ou achetés, des produits achetés par des clients ayant renseigné les mêmes attributs que lui, les produits les mieux vendus du moment, ou alors des produits choisis dans une liste selon des règles définies ; par exemple des produits à pourcentage de marge élevé ou prix bas, des produits en fin de stock ou alors des produits complémentaires à ceux consultés (Proposer la carte mémoire avec l'appareil photo numérique). La simplicité des calculs sur lesquels se basent ces moteurs fait qu'ils ne requièrent pas d'infrastructures logicielles ou

matérielles particulières et fonctionnent en temps réel. De plus, ils sont tous en mode « cloud », c'est-à-dire que les calculs sont faits sur le serveur de l'entreprise fournisseur qui envoie alors les recommandations pour affichage sur le site de son client. Ce sont les moteurs les plus utilisés et les plus fréquents sur les sites e-commerce petits et moyens. Le prix de ces services est de l'ordre de quelques centaines d'euros et dépend du nombre de visites sur le site. Le plus connu, Target2Sell est facturé 349€ pour un million de visites annuelles. L'installation et la mise en service sont facturées. Ce dernier type comporte beaucoup de solution « boîtes noires ». C'est ce type de solutions qui est actuellement mis en œuvre sur le site BUT.fr (Fig. 2). Sous R, il est aisé de générer de pareilles recommandations basées sur la proximité entre produits « item-based », entre clients « user-based » ou encore basées sur la popularité du produit grâce au package **recommenderlab.**

Figure 2 Recommandations affichées par le site BUT.fr suite à la consultation de la page d'un réfrigérateur (Source : But.fr)

Du point de vue de la qualité des recommandations, il est clair que ces systèmes sont sous-optimaux selon le type de données qu'ils utilisent. En effet, pour ce qui est des systèmes basés sur de la collecte explicite de données (L'utilisateur indique de manière explicite au système ses préférences via des notes, des « likes », une liste de favoris ou d'envies, etc.), il existe un biais de déclaration dû au fait que l'utilisateur s'exprime différemment selon ce sur quoi il procure des informations, ce qui fausse la récolte de données. Ce type de biais est fréquemment rencontré dans les données de questionnaire autour des sujets de société tabous. Lorsque les données utilisées sont issues d'une récolte implicite de données, c'est-à-dire de l'observation automatique des comportements de l'utilisateur en « arrière-plan » et sans qu'on le lui demande, un autre type de biais entre en jeu, dit biais d'attribution. On ne peut en effet pas, comme dans le cas de la récolte explicite, attribuer les données à un utilisateur en particulier. Par exemple, plusieurs personnes peuvent se servir du même ordinateur, et l'on peut très bien ne pas aimer un ouvrage que l'on a acheté ou acheter un produit pour quelqu'un d'autre.

Malgré tout cela, les algorithmes de recommandation de produits ont très largement prouvé leur efficacité commerciale et il est aujourd'hui impensable de faire fonctionner un site

web e-commerce sans leur appui : 35% des sites e-commerce ne sont pas rentables (55% des petits et moyens), 98% des visiteurs de sites de vente en lignes n'achètent rien, et les paniers contiennent un seul produit dans 90% des cas. En plus d'augmenter le taux de transformation (Nombre d'acheteurs / Nombre de visiteurs) et la rentabilité, les moteurs de recommandation ont une importance dans l'expérience des clients qui passent beaucoup de temps à chercher avant de trouver ce qui leur correspond. L'époque étant à l'instantanéité, il faut faciliter les achats des internautes en les immergeant dans un univers produits qui leur correspond. Le moteur de recommandation de produits n'est donc plus un gadget technologique superflu mais une brique permettant d'optimiser directement les fondamentaux du site à savoir son taux de transformation, son panier moyen qui est le montant moyen acheté, et sa capacité à fidéliser.

III.La recommandation de produits, un problème de données manquantes

1. Recommandation de produits et complétion de matrices

Mettre en place un moteur de recommandation de produits à des consommateurs peut être formalisé comme la prédiction de valeurs manquantes correspondant à l'appréciation des produits n'ayant pas été notés par les individus de l'échantillon dont on dispose. En effet, les données manquantes (DM) sont omniprésentes en statistiques appliquées et leur gestion a été décrite comme l'un des plus grands problèmes dans la recherche lorsqu'il s'agit de concevoir les études et d'analyser les résultats (William Shadish, cité dans Azar, 2002). Notre objectif ici est d'explorer ce champ de la recherche dans le cadre opérationnel de la recommandation de produits à des clients à partir de notes qu'ils ont laissé en ligne.

Le problème se ramène donc à une complétion de matrice par imputation de données, assez commune en statistiques appliqués au domaine de la génétique. L'application la plus fréquente consiste à utiliser un panel référence d'haplotypes[1] afin de prédire les données manquantes dans le génotype d'un échantillon d'individus. Ce procédé sert par exemple à trouver les gènes responsables de maladies génétiques. Cette approche a également gagné en célébrité suite au "Neflix Challenge" qui a eu lieu entre 2006 et 2009 et a créé une véritable effervescence dans le milieu des mathématiques appliquées au marketing. L'objectif de ce challenge était de créer le meilleur algorithme permettant de prédire le vote des utilisateurs dans leurs choix de films. L'algorithme en question a été amélioré depuis avec l'apport d'autres techniques et est aujourd'hui déployé et en fonctionnement sur la plateforme de contenus à la demande Netflix. Le challenge avait vu la mobilisation de 48 000 équipes provenant de 182 pays, dont le but était d'arriver à un algorithme qui améliorait de 10% en termes d'erreur type d'imputation la précision des prédictions générées par le système existant.

[1] Un haplotype est un groupe de gènes hérités d'un parent (Source : nature.com)

2. Choix des données & étude préliminaire

Nous extrayons nos données de départ depuis la base de données du site de vente en ligne But.fr. Elles consistent en la notation sur une échelle de 1 à 5 par les utilisateurs identifiés des produits qu'ils ont achetés.

Comme pour tout site e-commerce, chaque utilisateur ne note qu'un très petit sous-ensemble de produits, et il y a plus de produits notés que d'utilisateurs identifiés. Nous faisons notre extraction de données selon les critères suivants : Nous prenons uniquement les produits notés par deux individus au moins, et les individus ayant noté deux produits au moins ; ce qui a pour effet de réduire de plus de 80% la quantité de données extraites. Nous avons alors un jeu de données avec les 3 colonnes **identifiant client, produit et note** que nous transformons en ce qui sera notre matrice de travail au format utilisateurs x produits et qui contient les notations. Elle contient 807 lignes (utilisateurs) et 982 colonnes (produits) et est incomplète à hauteur de 99.8%.

Notre matrice de travail étant extrêmement incomplète, il faut bien avoir conscience du fait que nous entreprenons de **déduire 99.8% de données (soit 790757 valeurs) à partir des 0.2% (1717 valeurs) dont on dispose.** Nous garderons cette donnée à l'esprit tout au long de l'étude.

En outre, il existe une seconde problématique inhérente à la gestion des DM : Afin que notre démarche soit valide et sensée, il est essentiel de se poser des questions sur la structure des données avant d'entreprendre toute démarche d'imputation. Nous n'avons aucune indication sur cette structure a priori (Liens entre variables), et elle est d'autant plus difficile à appréhender du fait de l'impossibilité du calcul des indicateurs sur les lignes et les colonnes due au caractère hypercreux de la matrice de travail. Elle contient en effet des colonnes et des lignes avec deux valeurs seulement, toutes les autres étant manquantes (NA).

Le jeu de données est donc exceptionnellement incomplet, malgré les mesures prises pour éliminer les produits rarement notés et les utilisateurs qui notent peu. Cela est dû au fait qu'il y ait 982 produits pour 807 clients noteurs seulement, ce qui connote une communauté assez peu active et des clients web encore peu engagés. Chaque produit noté par un utilisateur donne lieu à une variable, et les utilisateurs notent généralement peu de produits. On calcule en effet que :

- Chaque produit est noté en moyenne 1.7 fois par les 807 clients noteurs.
- Chaque utilisateur a noté en moyenne 2.13 produits parmi les 982.

3. Objectifs opérationnels & scientifiques

Du point de vue opérationnel, notre objectif est d'utiliser ce jeu de données, avec toutes ses particularités, pour proposer des solutions permettant de générer des recommandations personnalisées pour nos clients. Ces recommandations sont destinées à être injectées dans la base CRM BUT qui alimente les instruments de marketing direct que sont :

- **La plateforme CRM IBM Unica Campaign,** permettant l'envoi de SMS & d'e-mails personnalisés, et de recueillir les réactions des ciblés : Ouvertures des mails, achats,

souscriptions à un programme de fidélité, etc. Particulièrement dans le cas de l'emailing, il est possible d'automatiser la recommandation grâce à ce que l'on appelle des "Triggers" qui sont des e-mails automatiques qui se déclenchent suite à un comportement du client (Une visite d'une page du site web par exemple, ou un passage en magasin), ou alors de manière régulière pour maintenir une relation durable s'appuyant sur une offre diversifiée et adaptative. En ce moment, un exemple de trigger en fonctionnement et montrant de très bons résultats est le trigger « panier abandonné » qui rappelle au client qu'il a des produits non réglés dans son panier et qui l'invite à finaliser son achat.

- **Le moteur d'affichage du site Web.** Le site But.fr grâce à un fichier texte qu'il diffuse appelé cookie http[2] reconnaît chaque internaute de façon unique et permet donc de le cibler spécifiquement et en temps réel. Ce type de recommandation est extrêmement efficace lorsqu'il est appliqué en fin du tunnel de commande, c'est à dire au moment de la validation de l'achat avant le paiement. On parle alors de cross-selling ou de vente additionnelle. 30 % des ventes d'Amazon seraient générées par les recommandations aux clients d'après l'organisme Forrester Research.

Sachant que le volume des données dont on dispose augmente avec l'activité e-commerce de BUT[3], les recommandations sont destinées à être mises à jour de manière régulière pour en affiner la pertinence et d'accompagner l'évolution des besoins des clients et de leurs projets. L'environnement R dont nous nous servirons n'étant pas utilisé par la direction des systèmes informatiques BUT, il ne pourra donc pas servir au déploiement final de la solution. Un portage dans un autre langage de script (Python) sera donc à prévoir. Dans cette optique, nous apporterons le plus grand soin à l'implémentation en commentant soigneusement le code pour qu'il soit clair et intelligible et en évitant les opérations obscures. Il s'agit de produire une solution claire et opérationnelle, mais surtout adaptée à un besoin.

Cependant, dans le cadre de ce document, l'objectif opérationnel, suffisamment approfondi en entreprise, servira principalement de prétexte pour s'attarder sur le côté scientifique via une approche comparative de la complétion de matrice en relation avec les avancées dans le domaine de la gestion des données manquantes. Après avoir exposé les points à considérer dans le cadre des données manquantes, nous choisirons trois approches distinctes ayant fonctionné sur notre matrice de travail issue du site BUT.fr puis nous en comparerons l'efficacité sous différentes conditions grâce à un algorithme basé sur des simulations que nous développerons à l'aide de fonctions de différents packages de l'environnement R. Il est en effet impératif de chiffrer la qualité de ces imputations avant de proposer une quelconque solution à l'industrialisation.

[2] Un cookie est un fichier stocké par un site web sur le terminal de l'internaute. Son rôle est de permettre au développeur web de conserver des données de l'utilisateur afin de faciliter sa navigation et permettre certaines fonctionnalités de personnalisation.

[3] 150 000 commandes entre le 1er juillet 2013 et le 30 juin 2014, 170 000 sur la même période en 2014-2015 soit une hausse de 13.3% et un chiffre d'affaire internet en augmentation de 12.2%.

IV. Une approche comparative de la complétion de matrices

1. Types et structuration des données manquantes

Nous présenterons ci-dessous quelques notions fondamentales développées par Rubin et/ou Little. Les données manquantes se rencontrent très souvent dans des données d'enquête, et peuvent être la réflexion de mécanismes sous-jacents qu'on ne peut ignorer.

a) Types de données manquantes

Les données manquantes dans les variables d'un jeu de données n'étant pas toujours le fruit du hasard, il faut en distinguer les causes afin d'aborder correctement leur imputation. On en distingue trois types :

a. **Missing completely at random (MCAR) :** Une donnée est dite MCAR si sa probabilité d'absence est la même que celle de toutes les autres observations. Elle est manquante complètement aléatoirement, avec une probabilité ne dépendant pas des mesures observées ou non observées. Avec r représentant la présence de la donnée, on écrit : $P(r|x_{obs}, x_{miss}) = P(r)$ et l'on dit que la non-présence est uniformément répartie.
b. **Missing at random (MAR) :** Une donnée MAR ne manque pas de manière totalement aléatoire. Sa probabilité d'absence est liée à une ou plusieurs variables observées. : $P(r|x_{obs}, x_{miss}) = P(r|x_{obs})$
c. **Missing not at random (MNAR) :** La donnée est manquante avec une probabilité dépendant des valeurs que prend la variable qui la contient. L'exemple le plus commun est celui où les personnes ayant un revenu très élevé refusent de répondre à la question du revenu dans un sondage.

On ne sait a priori pas dans quel cas se situe notre matrice de travail. On peut cependant supposer qu'elle suit le cas MCAR. Chaque produit (i.e variable) est noté par les clients qui l'ont acheté parmi le catalogue de plusieurs dizaines de milliers de produits et qui ont bien voulu aller sur le site pour le noter. Les données manquantes correspondent à des notes non laissées à des produits non achetés par le client chez BUT.

b) Répartition des données manquantes

Les DM se répartissent de trois manières différentes dans une matrice de données. Elles sont illustrées dans la figure 3 :

a. **Valeurs manquantes univariées :** Pour une variable j seule, si une valeur x_{ij} est manquante, alors il n'y a plus d'observations de cette variable.
b. **Valeurs manquantes monotones :** Si la valeur x_{ij} de la variable j pour l'individu i est manquante, alors les valeurs de toutes les variables suivantes sont manquantes.
c. **Valeurs manquantes non monotones :** Leur répartition ne suit pas de schéma ou de loi, elle est arbitraire dans la matrice de données.

C'est dans ce troisième cas que se situe notre matrice de travail.

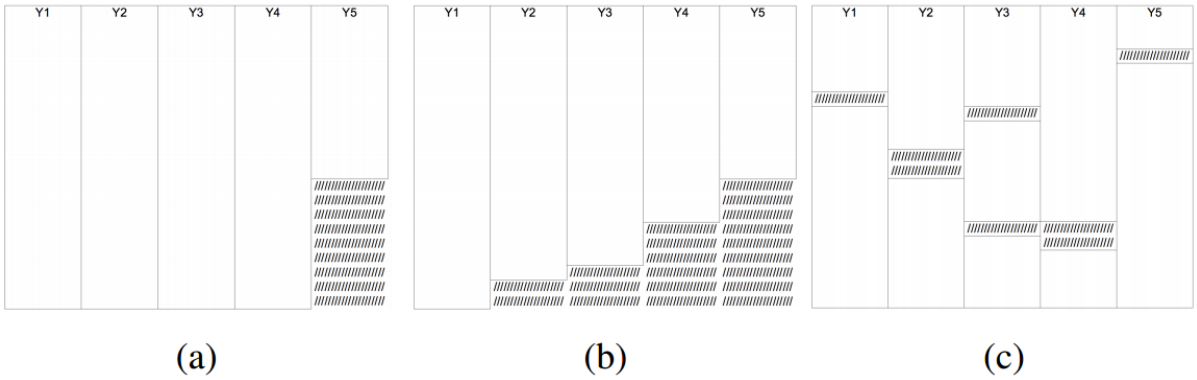


Figure 3 Les trois types de répartition des données manquantes. (Source : Page du Pr. Philippe Besse - goo.gl/uP1rEK)

2. Choix des méthodes

Il existe différentes manière d'imputer des données manquantes, plus ou moins poussées. Les méthodes les plus basiques consistent à remplacer la valeur manquante par la moyenne ou la médiane de la variable. L'imputation est également possible par l'ajustement de modèles de régression linéaire locale (Avec la possibilité d'ajouter du bruit et donc de la variabilité), mais cela implique de ne pas prendre en compte les relations entre variables.

Des méthodes plus évoluées basées sur différentes approches existent. Nous concentrerons notre attention dans ce document sur trois méthodes statistiques dont les propriétés permettent l'imputation de données manquantes : les forêts aléatoires, les k-plus proches voisins et l'ACP. Ce choix se justifie par le fait qu'elles représentent une large palette de méthodes statistiques d'une part, ce qui rend la comparaison d'autant plus intéressante, et par le fait qu'elles soient connues comme performantes d'autre part.

a) Méthode des k-plus proches voisins (K-Nearest Neighbors ou kNN).

Issue du domaine de la classification, la méthode des k-plus proches voisins est également utile en régression. Se basant sur des calculs de distances puis de moyennes pour établir la similarité, elle est computationnellement très simple, en plus d'avoir prouvé son efficacité sur une large variété d'applications. Elle ne requiert par ailleurs aucune hypothèse sur les données d'où un champ d'applications illimité et une mise en œuvre facilitée. La figure 4 ci-dessous en illustre le fonctionnement en classification dans un espace de dimension 2.

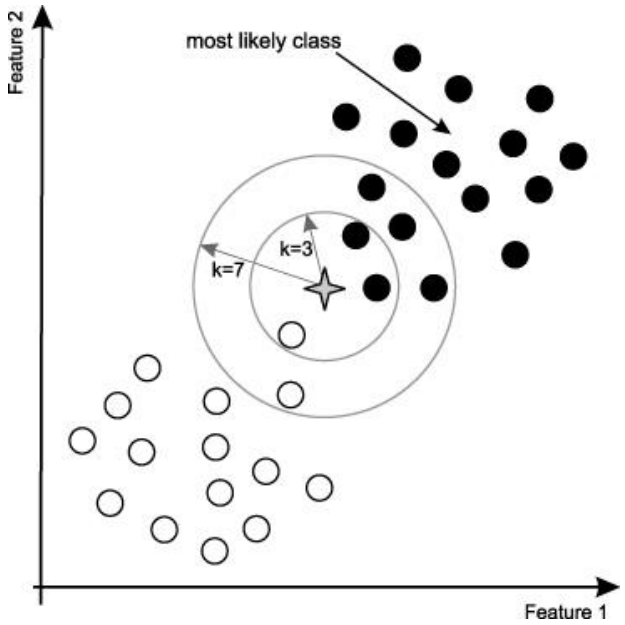


Figure 4 Fonctionnement de la méthode des k-plus proches voisins dans le cadre de la classification pour k = 3 et k = 7 (Source : goo.gl/gbCWwB)

Pour chaque ligne i (client) avec des données manquantes, cette méthode trouve ses k plus proches voisines en termes de distance euclidienne. On affecte ensuite aux valeurs manquantes la moyenne des valeurs des k voisins. On parle d'approche « user-based » car on se base sur les similitudes entre clients noteurs pour prédire les données. Une autre possibilité est de transposer la matrice (intervertir lignes et colonnes) pour adopter une approche dite « item-based » où l'on se sert de la similitude entre produits pour imputer. Cette méthode d'imputation est implémentée par la fonction **impute.knn** du package **imputeR**. Nous prenons k = 10 qui est la valeur par défaut de la fonction.

b) Forêts aléatoires (Forêts d'arbres décisionnels)

La méthode des Forêts Aléatoires (Random Forest ou RF) est une méthode récente basée sur des arbres de décision comme celui de la figure 5 et qui rencontre un franc succès depuis sa formalisation en 2001 par Adele Cutler et Leo Breiman. Particulièrement polyvalente, elle ne requiert pas d'hypothèses particulières sur la structure ou la nature des données qu'on lui soumet. Elle remplit par ailleurs deux fonctions : La classification et la régression et peut donc servir de méthode d'imputation. La méthode consiste à faire coopérer des milliers d'arbres de décision générés automatiquement sur des sous-ensembles tirés avec remise dans le jeu de données pour trouver la valeur à imputer. Cette manière de procéder, qui fait participer une multitude de classifieurs faibles dont la connaissance additionnée permet une prise de décision

robuste s'appelle Bagging (Bootstrap Aggregating) et a été pensée par Breiman en 1996. Elle présente l'avantage de corriger la tendance au surajustement qu'ont les arbres de décision, c'est-à-dire le fait d'être difficilement utilisable sur des données n'ayant pas servi à les générer, ce qui est tout l'intérêt des modèles de prédiction. En apprentissage automatique, ce type de méthodes est de plus en plus utilisé du fait du développement d'algorithmes pouvant tourner sur plusieurs processeurs ou plusieurs machines et de l'augmentation de la puissance des ordinateurs.

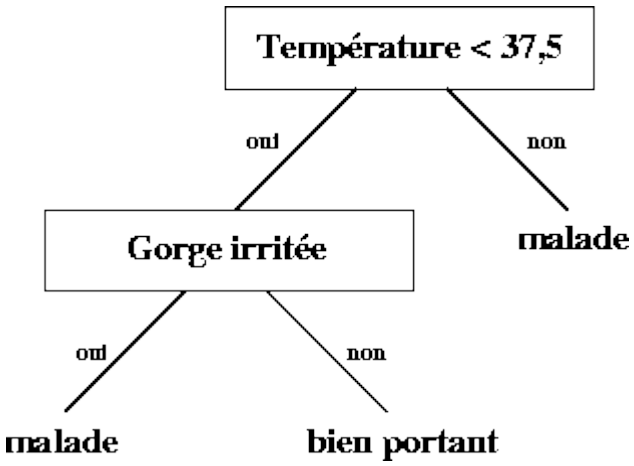


Figure 5 Un exemple simple d'arbre de décision permettant de déterminer si un patient est malade à partir de sa température corporelle (variable quantitative ayant une importance primaire dans l'arbre) et de l'état de sa gorge (variable qualitative binaire). (Source : Site de l'université de Lille 3 - goo.gl/h2wyop)

Nous utilisons les propriétés de cette méthode non paramétrique implémentées dans le package R « **MissForest** » (Stekhoven, 2013) pour prédire la valeur des DM. Cette méthode est itérative et gourmande en ressources, mais l'algorithme peut être configuré pour fonctionner de manière distribuée sur plusieurs processeurs. MissForest commence par ajuster une forêt aléatoire sur les données disponibles dans le jeu de données, puis prédit les manquantes. Ces deux étapes sont répétées jusqu'à la convergence. L'algorithme, malgré sa complexité n'a pas besoin de paramétrage spécifique ce qui le rend simple à utiliser.

c) ACP itératives régularisées

L'ACP est une technique d'analyse des données très populaire qui consiste à projeter un nuage de points existant dans un espace de grande dimension dans un espace de dimension moindre de manière à en permettre la compréhension. Cela se fait selon des axes de projection appelés facteurs et qui ont la particularité de conserver un maximum d'informations des données de départ. C'est de ce point de vue une méthode de réduction de la dimensionnalité qui semble très adaptée à des jeux de données contenant beaucoup de colonnes (produits).

Il existe en plus de cela un package R contenant une fonction spécialisée dans l'imputation de données manquantes se basant sur l'utilisation de l'ACP de manière itérative. La méthode en question, en plus d'être précise, présente des propriétés très intéressantes en termes de conservation des caractéristiques de la matrice à laquelle elle s'applique. Cette méthode tire parti d'un algorithme itératif d'analyse en composantes principales (EM-PCA ou Expectation-Maximization PCA) régularisé afin d'éviter le surajustement (Josse et al., 2012). L'initialisation consiste à imputer les valeurs manquantes par la moyenne. On réalise ensuite

l'ACP de la matrice obtenue ce qui revient à la projeter dans un espace de dimension réduite qui représente au mieux sa variabilité et sert à « reconstruire » les données. On déduit ainsi une première série de valeurs plausibles (par minimisation de la distance entre l'individu et sa projection) que l'on attribue aux DM. On répète ensuite l'opération jusqu'à convergence. Cette approche est utilisée notamment pour reconstruire des images. Nous utilisons ici la fonction **imputePCA** issue du package **MissMDA** (Husson et Josse, 2010). Nous effectuons les imputations pour un nombre de facteurs de l'ACP égal à deux.

3. Méthode de comparaison

a) Principes

Nous mettons au point une méthode de comparaison basée sur l'introduction de manière aléatoire d'un pourcentage contrôlé de DM dans un jeu de données complet. Une fois celui-ci imputé, nous procédons à un calcul permettant d'apprécier l'exactitude des valeurs introduites, en comparaison avec celles du jeu de données de départ complet. Nous faisons donc des séries de simulations, dont nous multiplions le nombre dans un souci de précision et de stabilité des résultats de l'expérimentation.

Chaque méthode d'imputation est testée sur l'intervalle de pourcentages de DM allant de 5 jusqu'à 95% afin d'approcher le cas des données de recommandation.

Nous effectuerons l'étude sous l'hypothèse MCAR, habituellement admise dans les études comparant les méthodes de complétion de matrices et comprenant l'introduction artificielle de NA.

Nous limiterons par ailleurs notre étude aux données quantitatives continues comme base unique de test.

Nous évaluerons la qualité de la complétion d'une matrice par la valeur du critère NRMSE (Normalized Root Mean Square Error - Erreur moyenne normalisée) défini comme suit :

$$NRMSE = \sqrt{\frac{moyenne\left[\left(y_{imp} - y_{réel}\right)^2\right]}{variance(y_{réel})}}$$

Où y_{imp} est le vecteur contenant la totalité des valeurs imputées (sa dimension étant égale au nombre de valeurs manquantes) et $y_{réel}$ le vecteur de même dimension contenant l'ensemble des valeurs réelles qui ont été supprimées de la matrice d'origine. Un NRMSE faible est associé à une bonne qualité de prédiction des valeurs imputées. Chaque valeur de NRMSE que nous utiliserons sera accompagnée de son écart-type afin d'apprécier la stabilité de la prédiction associée. Pour ce faire, nous ferons plusieurs simulations pour chaque cas testé.

Sachant que nous n'avons pas d'informations a priori sur les données de notation de produits, nous ferons l'étude dans différentes conditions.

b) Données

Selon nos besoins, nous générerons nos données sous contraintes. Pour que nos résultats soient généralisables, nous ne travaillerons que sur des lots de matrices de données simulées, c'est-à-dire contenant des nombres générés aléatoirement sous une certaine loi.

Les variables sont générées sous loi normale car nous supposons que l'ensemble des notes attribuées à un produit se répartissent selon un écart type σ autour d'une valeur moyenne μ.

c) Algorithme

La comparaison de l'efficacité des méthodes de complétion sur une matrice M donnée se fait comme suit :

- On commence par générer une matrice M qui servira de base de test. On sélectionne ensuite les bornes de l'intervalle P des pourcentages de DM à introduire et le pas qui permet de le subdiviser. Ici on prend l'intervalle allant de 5 à 95% avec un pas de 5% soit 19 valeurs de %NA. Au-delà de 95%, il y a un risque d'introduire 100% de DM dans une colonne ce qui revient à la supprimer.
- On choisit ensuite le nombre de simulations d'introduction de DM à faire pour chaque %NA, nous choisissons d'aller jusqu'à 50 simulations. On génère ainsi les jeux de données incomplets en introduisant les NA à des positions aléatoirement réparties. On a alors 50 x 19 = 950 matrices incomplètes.
- On impute les 950 matrices avec chacune des trois méthodes choisies en plus de deux autres méthodes servant de référence (imputation par la moyenne et par la médiane), on a alors 950 x 5 = 4750 matrices complétées.
- On calcule pour chaque matrice complétée l'erreur type normalisée (NRMSE) par rapport à la matrice de base M.
- On agrège les données pour obtenir pour chaque couple {méthode d'imputation ; %NA} une moyenne de NRMSE et l'écart-type associé. Peu importe la méthode d'imputation utilisée sur une matrice, la qualité de sa complétion est jugée par le même critère ce qui rend possible la comparaison.

V.Résultats, discussion et limitations

1. Résultats

Nous illustrerons l'étude par des graphiques représentant l'erreur type normalisée moyenne en fonction du pourcentage de valeurs manquantes dans la matrice imputée, et ce pour chacune des méthodes étudiées. L'erreur type (NRMSE) doit être interprétée comme suit :

- Si le NRMSE = 0 alors l'imputation est parfaite.
- Si le NRMSE = 1 alors l'imputation n'est pas meilleur que dans le cas où l'on remplace toute valeur manquante par la moyenne de sa série.

Chaque valeur de NRMSE moyen est accompagnée d'une barre d'erreur donnant une idée de la stabilité de l'imputation. Cette barre contient 95% des valeurs de NRMSE calculées sur le lot de 50 matrices imputées pour le %NA et la méthode considérée.

a) Effet du pourcentage de données manquantes

Les données manquantes sont un phénomène anecdotique en analyse des données, du le plus souvent à des évènements anormaux : Enquêté qui refuse de répondre à une question,

erreur d'instrument de mesure. Il en résulte habituellement des pourcentages de DM faibles, dépassant rarement les 30%. Lorsque l'on se place du point de vue de la recommandation de produits, le problème prend une dimension systématique et bien plus étendue. Si les grands acteurs commerciaux disposent de bases de données riches contenant des dizaines de millions de notations, le pourcentage de valeurs manquantes reste la plupart du temps très élevé du fait du nombre de produits à considérer.

Nous étudions ici l'effet du pourcentage de données manquantes sur la précision de nos algorithmes de complétion. Nous utilisons pour ce faire un jeu de données généré sous une loi normale multivariée grâce au package MASS, avec produits dont les notes sont corrélées à r = 0.6. Les résultats sont présentés dans la figure 6 ci-dessous.

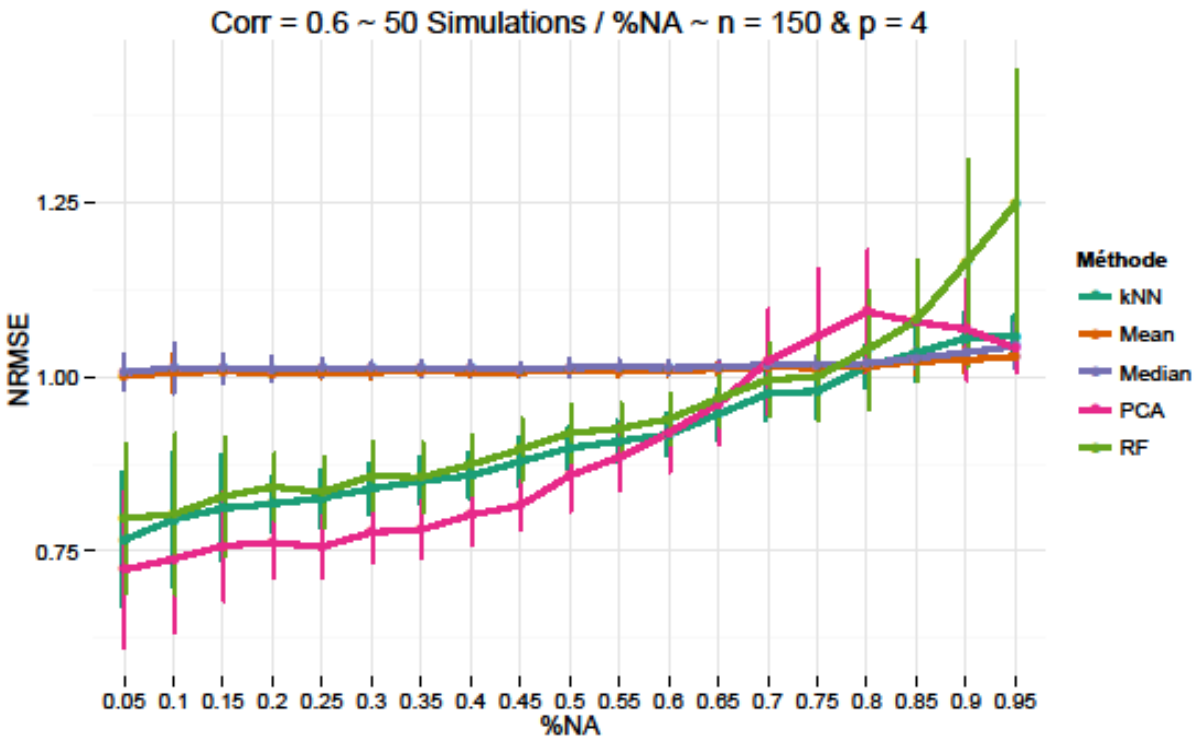


Figure 6 Effet du pourcentage de DM sur l'efficacité de l'imputation par forêts aléatoires (RF), ACP itératives (PCA) et k-plus proches voisins (kNN) par rapport à l'imputation par la moyenne (mean) et la médiane (median) le pourcentage de DM est en abscisse alors que la précision de la prédiction est en ordonnée. .

La méthode par ACP itératives est ici la plus efficace pour %NA < 60% mais devient inefficace (Par rapport à une complétion par la moyenne) et instable à partir de %NA = 70% avant de revenir à un NRMSE = 1 pour %NA = 95%.

Ce n'est pas le cas de la méthode Miss Forest qui présente des NRMSE de plus en plus dispersés et atteignant 1.25 pour des valeurs extrêmes de %NA. Il semblerait donc que la méthode basée sur les forêts aléatoires soit particulièrement inadaptée pour des pourcentages de DM très élevés.

La méthode des k-plus proches voisins par contre est particulièrement stable sur tout le spectre de %NA mais devient équivalente à une complétion par la moyenne à partir de 80% de DM. Elle est en outre aussi précise que la méthode basée sur les arbres de décision, tout en

étant bien plus parcimonieuse, beaucoup moins gourmande en ressources de calcul et plus rapide.

La précision de l'imputation est donc clairement fonction du pourcentage de données disponibles. Une entreprise a tout intérêt à collecter des données pour enrichir ses bases et ainsi améliorer ses recommandations. Dans cette optique-là, un achat quelconque chez BUT déclenche l'envoi automatique vingt et un jours plus tard d'un e-mail demandant au client de noter le ou les produits concernés par l'achat (Figure 7 ci-dessous), puis éventuellement d'un e-mail dit de « relance ». Plusieurs sites par ailleurs proposent de participer à un tirage au sort ou un jeu en notant un produit.

Figure 7 Visuel d'un email de "demande d'avis" envoyé suite à l'achat d'une bibliothèque.

b) Effet des liaisons linéaires entre variables

Tout système de recommandation de produits se base sur l'hypothèse selon laquelle les données laissées par les utilisateurs présentent des structures et des liaisons, à savoir que les produits identiques sont notés de manière identique et donc que les variables qui les représentent sont corrélées. De même, deux utilisateurs ayant les mêmes goûts noteront un produit de manière similaire. Cette hypothèse est essentielle car imputer des DM au sein d'un jeu de données sans structure (Aucune corrélation entre variables) conduira à des valeurs prédites de manière aléatoire.

Nous comparons ici l'efficacité des méthodes de complétion sur trois matrices de données, de plus en plus structurées en termes de coefficient de corrélation de Pearson. Nous faisons l'étude pour r = {0.3, 0.6, 0.9}. Les figure 8 et 9 ci-dessous permettent de s'apercevoir de la différence de structure entre le cas où r = 0.3 et celui où r = 0.9.

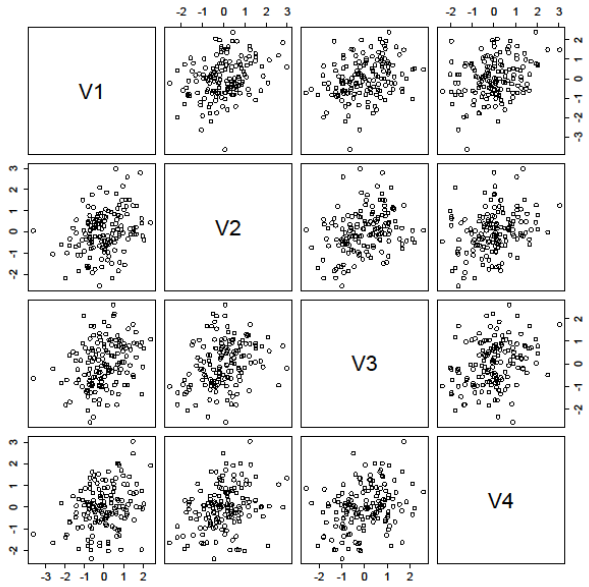


Figure 8 Représentation graphique des liens entre variables lorsqu'elles sont reliées pars une corrélation linéaire de valeur r = 0.3 (Coefficient de Pearson)

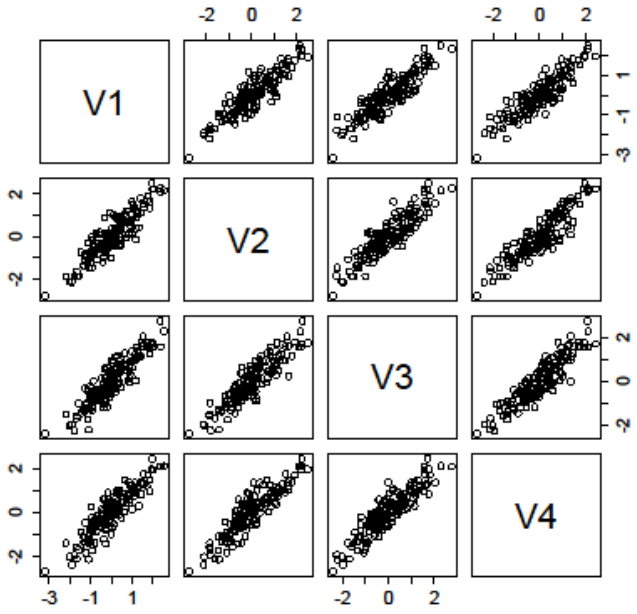


Figure 9 Représentation graphique des liens entre variables lorsqu'elles sont reliées pars une corrélation linéaire de valeur r = 0.9 (Coefficient de Pearson)

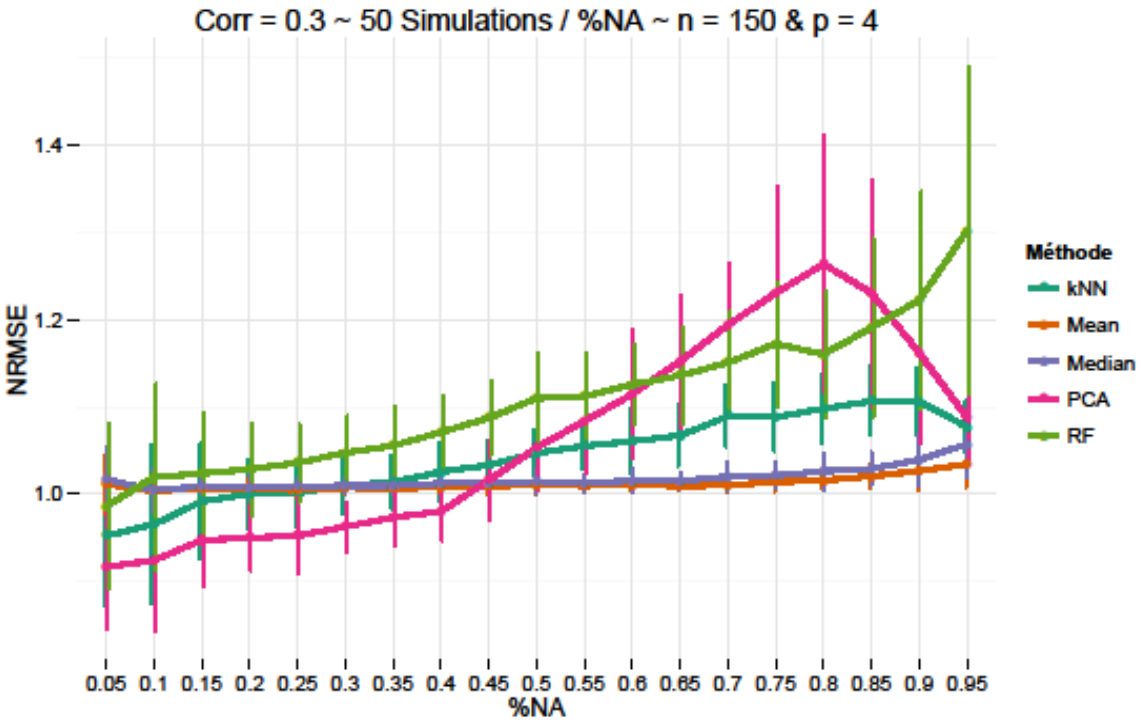


Figure 10 Effet de la liaison linéaire entre variables, cas faible r = 0.3

À la vue du graphique de la figure 10, il semblerait que la structuration des données soit un facteur très impactant en imputation de matrices. Une faible structure différencie bien les trois méthodes. La méthode Miss Forest par exemple est totalement inefficace quelle que soit la proportion de DM et très peu stable pour des %NA élevés alors que celle des k-plus proches voisins le devient à partir de 20%.

Seules les ACP itératives performent mieux qu'une imputation par la moyenne jusqu'à 45% de DM dans la matrice. On remarque cependant avec cette méthode un maximum d'imprécision pour 80% de DM après une forte croissance du NRMSE qui commence à 40% et que l'on peut expliquer par ce qui suit : Pour 80% de DM, il y a beaucoup de composantes principales possibles, dont une majorité d'erronées. L'algorithme tourne et force l'imputation sur la base du rapprochement avec les axes trouvés lors de la première itération, puis va jusqu'au bout. Ceci explique également la taille des barres d'erreur : Les composantes choisies par l'algorithme diffèrent d'une imputation à une autre, selon le placement des NA. Ce pic d'imprécision est suivi par une phase de décroissance du NRMSE qui atteint un minimum local pour %NA = 95%. Pour un tel pourcentage et avec un jeu de données aussi petit, il reste tellement peu de données que la première imputation par la moyenne ne bouge pas au fil des itérations, ce qui explique le NRMSE = 1, comme pour l'imputation par la moyenne.

Globalement, pour un jeu de données peu structuré, la prédiction est difficile pour les trois méthodes et est rendue impossible par une hausse forte du %NA. Un moteur de recommandation efficace se doit donc d'être basé sur des données collectées « proprement », c'est-à-dire sans introduction de biais aucun. En outre, plus on dispose de données, et mieux on peut générer des recommandations. Une marque doit donc savoir établir un échange avec ses clients, et recueillir autant de retours de leur part que possible : avis, notes, etc.

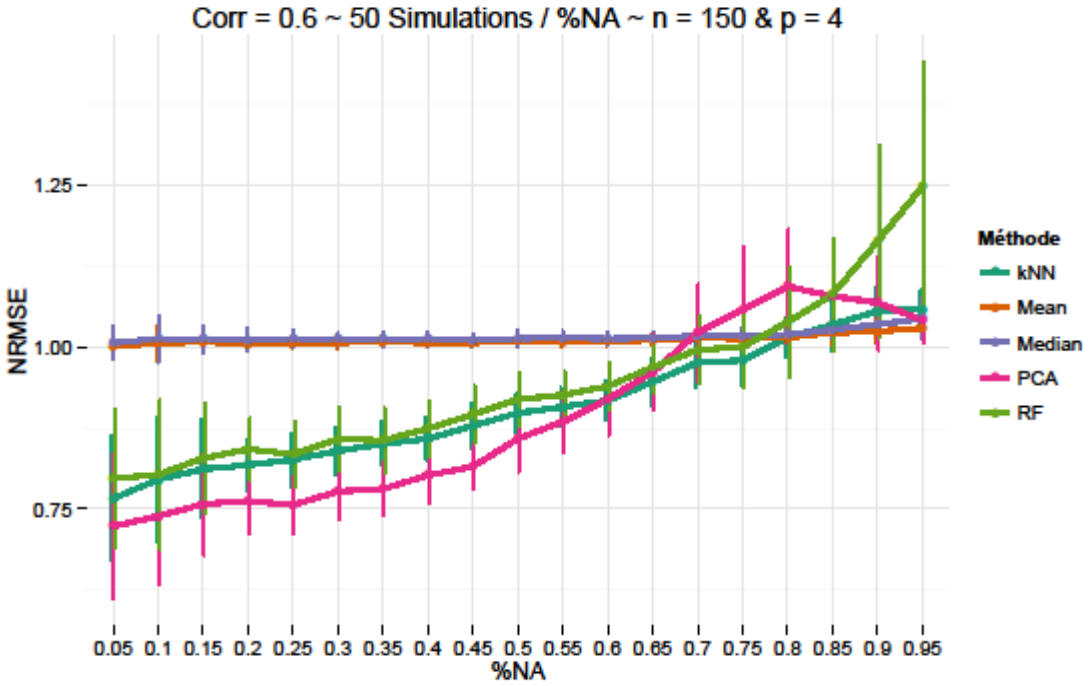


Figure 11 Effet de la liaison linéaire entre variables, cas moyen-fort r = 0.6

La précision de la prédiction est largement améliorée pour un jeu de données plus structuré. Pour une valeur de corrélation entre les variables égale à 0.6 (Fig. 11) les trois méthodes fonctionnent jusqu'à %NA = 70, toujours avec un avantage pour la méthode par ACP jusqu'à 60%. Cette dernière présente toujours un pic d'imprécision pour 80% de DM, mais moins marqué que dans le cas précédent. La méthode Random Forest ne montre à ce stade toujours pas son intérêt par rapport à une simple méthode des N-plus proches voisins.

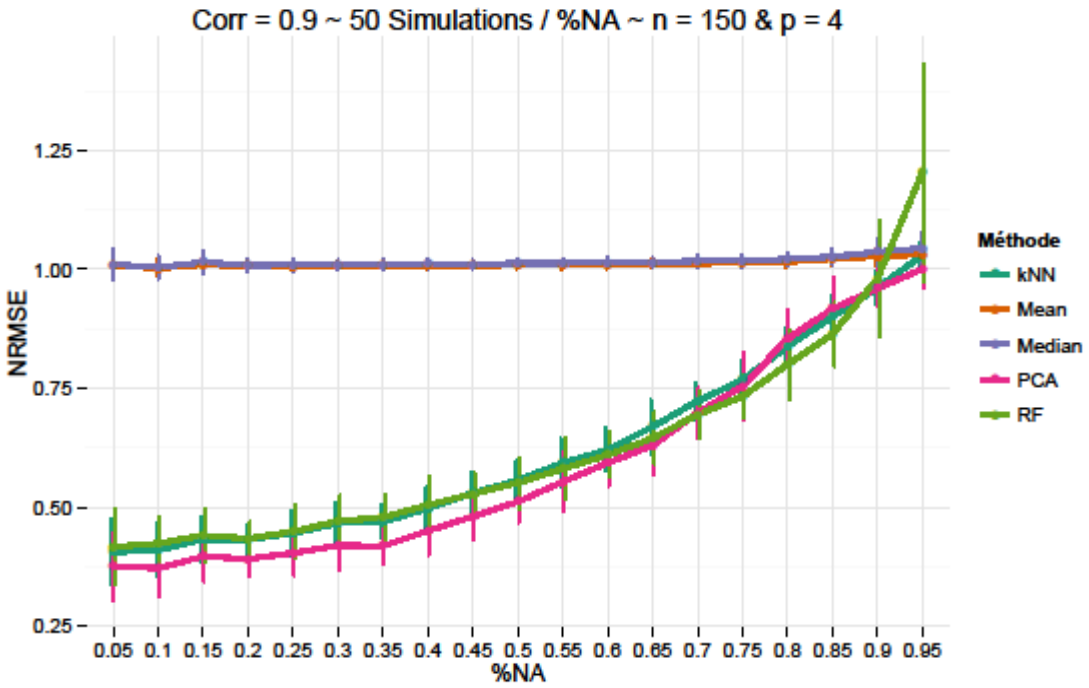


Figure 12 Effet de la liaison linéaire entre variables, cas fort r = 0.9

Dans le cas d'un jeu de données très structuré (Fig. 12), la qualité de la prédiction est améliorée pour les trois méthodes qui montrent des performances très similaires mis à part Random Forest qui supporte toujours aussi mal les %NA extrêmes. La méthode par ACP ne montre pas de pic d'imprécision pour 80% de DM. Travailler sur un jeu de données très structuré a donc pour effet de faciliter la prédiction et d'en stabiliser la qualité. Un algorithme d'imputation « saisit » mieux la structure de la matrice sur laquelle il travaille lorsque celle-ci est plus évidente. Il est donc ensuite plus simple de « deviner » les valeurs absentes à partir de celles fournies en entrée. De ce point de vue, les algorithmes dont nous nous servons sont des méthodes d'apprentissage artificiel : Ils sondent les données fournies en entrée à la recherche de motifs ou patterns qui servent ensuite comme base pour prédire des valeurs absentes.

c) Effet du type de liaisons entre variables : Liaisons non linéaires

Non détectée par le coefficient de corrélation de Pearson, la corrélation non linéaire apparait quand deux variables sont liées par une fonction puissance, exponentielle, logarithme ou toute autre liaison non affine (forme $x_2 = ax_1 + b$). N'ayant pas d'a priori sur les structures de données possibles dans une matrice de notes, nous testons nos méthodes dans le cas d'un jeu de données contenant des variables qui sont liées par des fonctions puissance les unes aux autres. Ainsi, nous avons : $Var3 = Var1^2$ et $Var4 = Var2^3$. La figure 13 ci-dessous permet de se rendre compte de la structure entre les quatre variables.

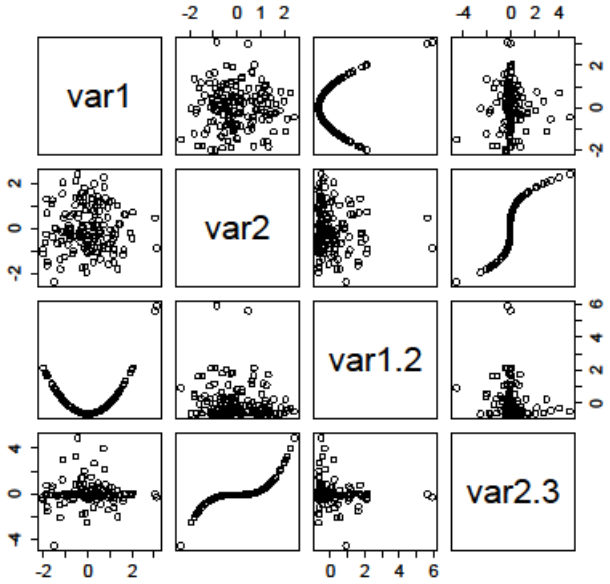


Figure 13 Représentation graphique des liens entre variables liées non linéairement.

Le résultat des simulations est le suivant (Figure 14).

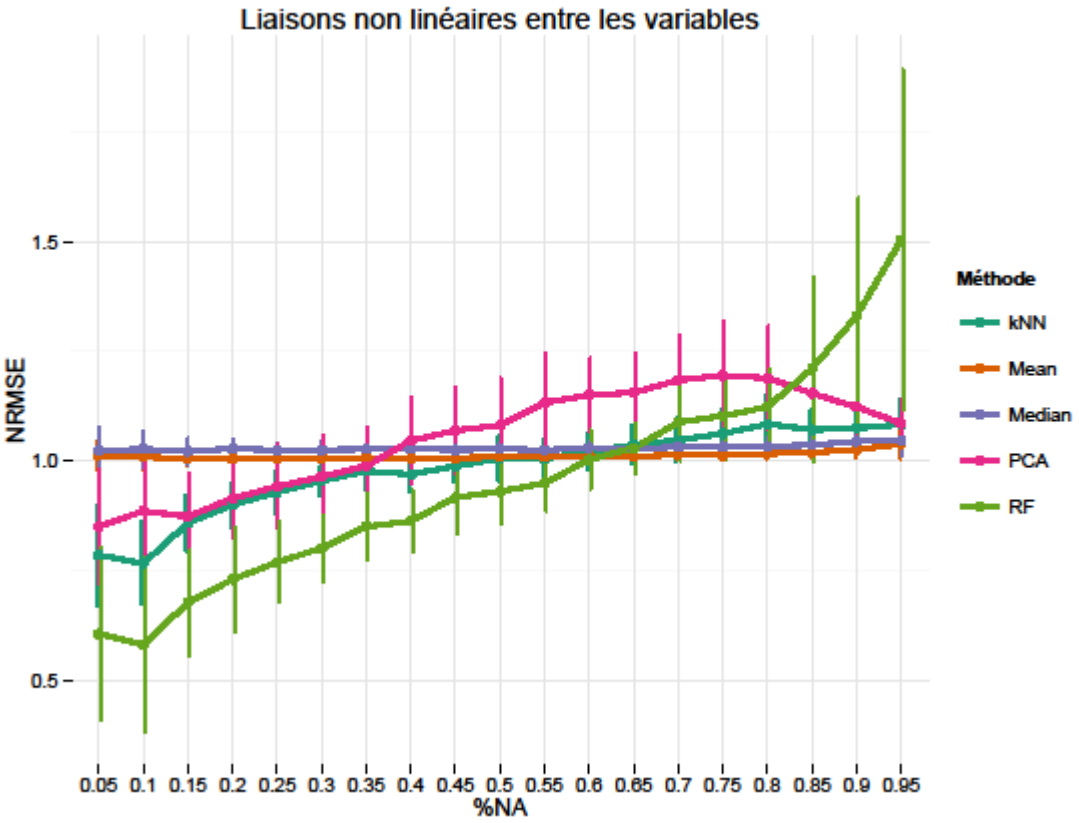


Figure 14 Qualité de la prédiction en conditions non-linéaires

La méthode basée sur l'ACP est la moins efficace dans ce cas. L'ACP en effet est une méthode linéaire et ne tient pas compte d'éventuelles liaisons de nature non linéaire entre variables. Elle est de ce point de vue inadaptée au traitement de données structurées non

linéairement. Cela se voit dans le fait qu'elle présente un NRMSE moyen de 0.7 pour 5% de DM, valeur qui dépasse 1 pour des %NA > 40%.

La méthode des k-plus proches voisins se comporte de manière similaire sur les %NA faibles (<0.4), puis imite le comportement de l'imputation par la moyenne pour des proportions de DM plus élevées.

La méthode Random Forest, pour des proportions de DM inférieures à 65% est bien plus précise que les autres et montre une véritable capacité à imputer des matrices structurées non linéairement. En effet, la force de cette méthode réside dans sa grande polyvalence. En plus d'être efficace sur des données mixtes (qualitatives et quantitatives d'où plusieurs applications dans l'analyse de données clients), le fait qu'elle soit basée sur des arbres de décision rend l'hypothèse de linéarité non nécessaire. De ce fait, Random Forest prend en compte les dépendances même complexes comprenant par exemple des combinaisons de fonctions trigonométriques et puissance.

Les figure 15 et 16 ci-dessous montre les résultats de la même expérimentation avec des liaisons non linéaires plus complexes faisant intervenir les fonctions sinus, cosinus et exponentielle et permet de valider ce qui a été dit précédemment.

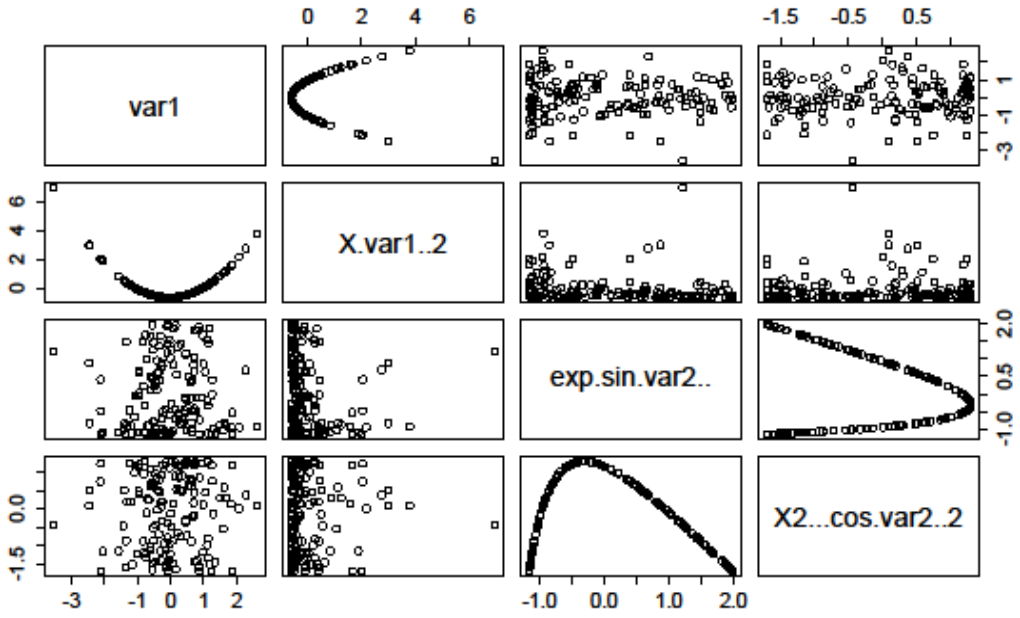


Figure 15 Représentation graphique des liens entre variables liées non linéairement de manière complexe.

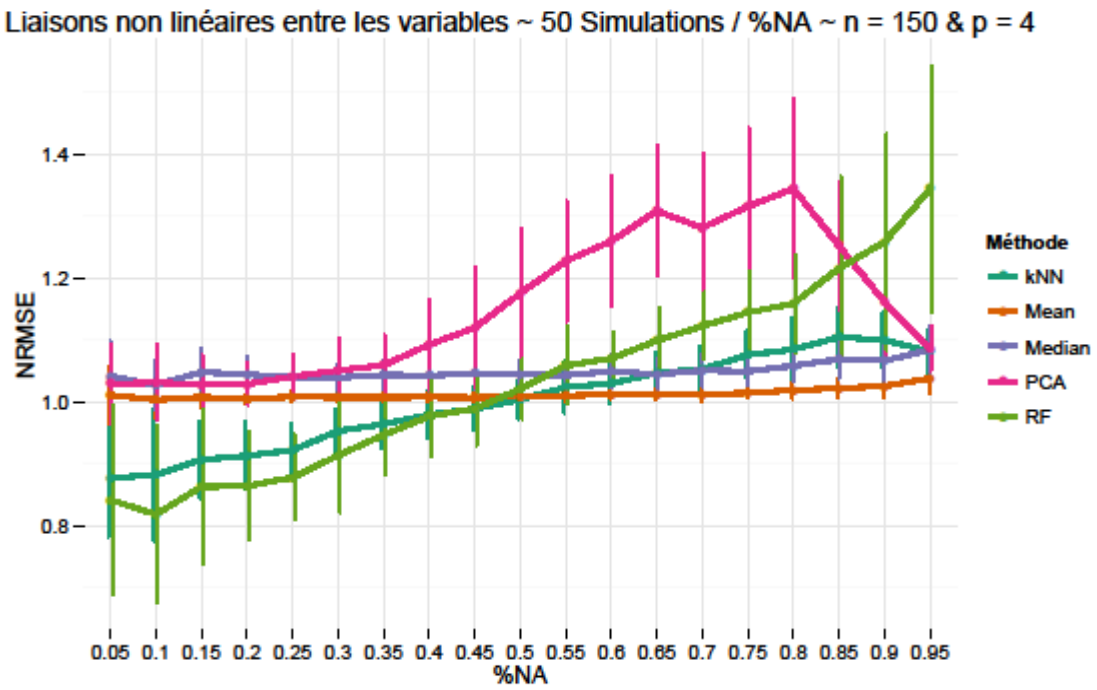


Figure 16 Impact de liaisons non-linéaires complexes entre les variables sur la qualité de la prédiction.

Random Forest est encore une fois la seule méthode à prendre en compte la non-linéarité tandis que la méthode par ACP est encore plus inadaptée. Celle des k-plus proches voisins présente des résultats satisfaisants également du fait de son fonctionnement « naïf » basé sur la similarité, et ce même pour des %NA supérieurs à 50%. Tout comme la méthode RF, elle ne requiert pas d'hypothèses sur la linéarité.

d) Effet du nombre d'individus

Nous simulons pour cette série de tests les jeux de données standard, mais avec dix fois plus de lignes (Figure 17).

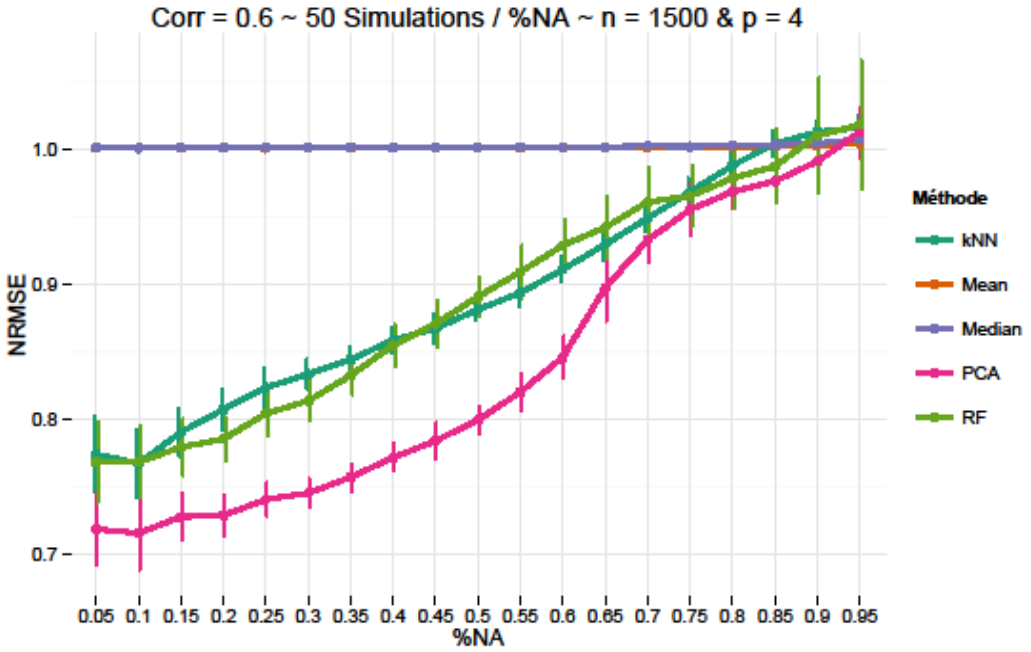


Figure 17 Impact de l'augmentation du nombre de lignes sur la qualité de prédiction

La qualité des prédictions s'en voit comme attendu améliorée en comparaison avec la figure 11. En effet, les méthodes ont dix fois plus de données d'apprentissage lorsque la taille de l'échantillon est multipliée par dix, et ce quel que soit la proportion de NA. Pour %NA = 95% par exemple, on a 7.5 données par variable en moyenne quand la matrice compte 150 lignes alors que l'on monte à 75 données quand elle en compte 1500. L'apprentissage se fait donc sur des données plus représentatives de celles de la matrice complète et il en résulte une meilleure précision de prédiction. De plus on ne retrouve plus de comportements instables pour RF lorsque %NA prend des valeurs extrêmes, ni pour la méthode par ACP pour des valeurs de %NA autour de 80%. L'algorithme d'ACP est le plus efficace ici sur l'ensemble du spectre, simplement parce qu'il est plus aisé de trouver une composante principale lorsque l'on dispose plus de données.

e) Effet du nombre de variables

Nous faisons cette fois-ci l'expérience avec dix fois plus de variables que précédemment et présentons les résultats en figure 18.

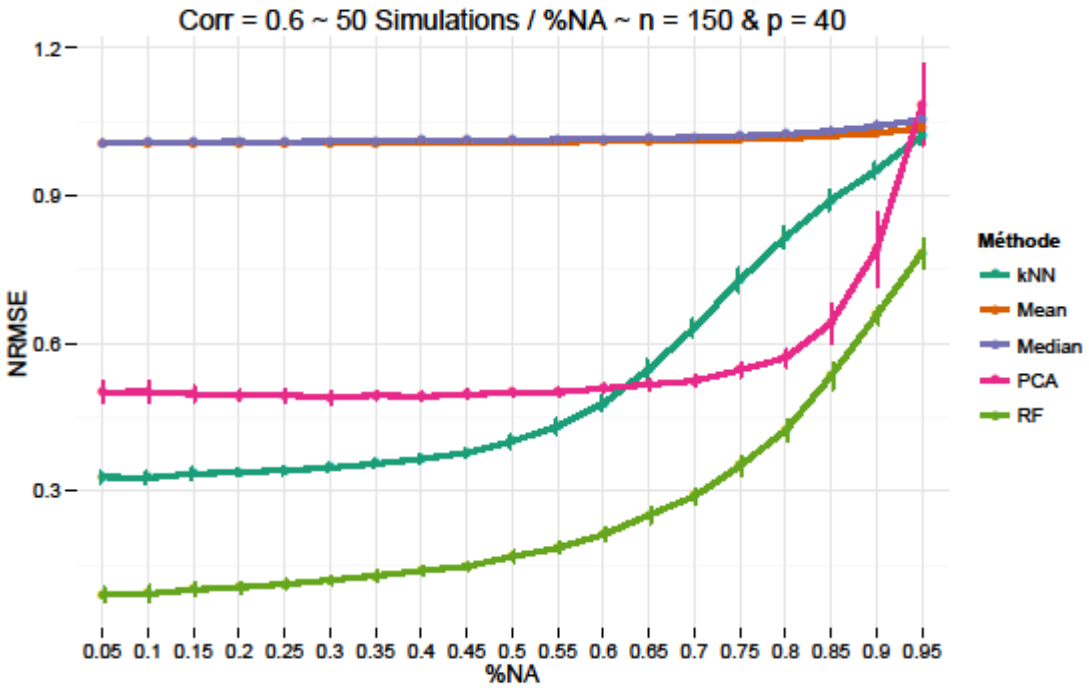


Figure 18 Impact de l'augmentation du nombre de colonnes sur la qualité de prédiction.

De manière identique à précédemment, nous multiplions la quantité de donnés par 10. Le gain de précision est supérieur à celui obtenant en multipliant le nombre de lignes et ce particulièrement pour la méthode RF qui montre une exactitude remarquable, inférieure à 0.3 jusqu'à %NA = 70%. Toutes les méthodes sont par ailleurs bien plus stables que dans les autres cas. Cependant, l'efficacité de la prévision par ACP itératives pourrait être largement améliorée en prenant plus de composantes principales. On se sert ici de 2 composantes seulement sur les 40 possibles.

2. Discussion et limitations

a) Données

Les expériences citées dans ce document utilisent des données manquantes complètement aléatoirement. Les résultats présentés peuvent donc ne pas être généralisables à des cas où les données manquent de manière biaisée et non aléatoire.

Ensuite, nous avons ici fait l'usage de données générées de manière contrôlées. Dans la réalité, on n'a aucun a priori dessus et elles peuvent comporter des problèmes de qualité. Le plus grand soin doit donc être apporté à leur collecte, leur stockage et les opérations de prétraitement qu'elles subissent. Elles servent en effet de matière à un algorithme d'apprentissage, un biais dedans entraînerait forcément des prédictions biaisées.

Dans le cas des données But.fr qui ont un pourcentage de DM à 99.8%, il est difficile de générer des prédictions fiables du fait d'un problème bien connu dans le domaine de la recommandation de produits appelé « cold start », que l'on pourrait traduire par démarrage à froid. Ce problème est la conséquence pure et simple du manque de données ; on a la possibilité de bâtir un système, mais l'apprentissage est tellement maigre que les prédictions sont très approximatives. Ce problème est partiellement corrigé en supplémentant les systèmes avec des données sur l'utilisateur comme son âge, son sexe, son appartenance à tel ou tel segment, etc qui entrent alors aussi en compte dans la suggestion quand la donnée est rare. On parle alors de filtrage collaboratif hybride, utilisé dans tous les grands moteurs (Netflix et autres), qui considèrent que la recommandation doit prendre en compte toutes les données disponibles à propos de l'utilisateur pour coller au mieux à la réalité. Le point de départ est alors non pas une mais deux voir plusieurs matrices de données, selon le nombre de sources à prendre en compte. Typiquement, il y a trois sources en entrée : matrice de notations, informations sur les clients, et informations sur les produits. Ces moteurs sont bien sûr plus compliqués à mettre en place. Ce problème de « cold-start » est très fréquemment rencontré par les entreprises de moyenne et petite taille n'ayant pas la chance d'avoir des millions de données de clients et explique le recours coutumier aux moteurs clé en main simples précités.

b) Temps de calcul

En se plaçant dans le cas d'un jeu de donné de grande taille et bien peuplé, les trois méthodes sont efficaces. Simplement, elles sont très différentes en termes de besoins en calculs. Les méthodes par ACP et par forêts aléatoires sont complexes et fonctionnent de manière itérative, c'est-à-dire en exécutant plusieurs fois la même fonction sur les données et en les mettant à jour jusqu'à l'atteinte d'un objectif. Cela occasionne de longues phases de calcul, incompatibles avec une mise à jour fréquente des données servant à la recommandation. Cependant, l'algorithme **MissForest** dans R est facilement parallélisable, ce qui entraine un gain considérable de vitesse en distribuant les calculs sur plusieurs cœurs ou processeurs.

La méthode des kNN quant à elle a montré une bonne efficacité, quelle que soit la situation et présente un temps de calcul négligeable. Parcimonieuse et facile à programmer, elle est aussi particulièrement stable et peut en conséquent présenter un excellent choix lorsque la réactivité rentre en compte. De plus, la précision de la méthode peut être optimisée en ajustant le nombre k de voisins à choisir avec une démarche de validation croisée, à savoir faire la complétion avec plusieurs valeurs de k et choisir celle qui minimise l'erreur de prédiction.

c) Perspectives

Mettre en place un moteur de recommandation de produits est une tâche complexe qui requiert une longue période d'étude, beaucoup de choix et de développement logiciel. De plus, il faut disposer d'infrastructures lourdes : modules de récolte de données, traitements de déduplication, nettoyage, organisation, serveurs de stockage, etc. ; en pratique, peu d'entreprises disposent de tout cela. En revanche, tout distributeur recueille et stocke les achats faits en magasin lors du passage en caisse, principalement à des fins de reporting, de gestion des stocks et de comptabilité.

Une autre approche consiste donc à vouloir travailler sur ces données-là et à y chercher des « motifs » permettant de déduire des règles. Concrètement, cela consiste à extraire tous les tickets de caisse effectués par ses clients, puis chercher dans ce corpus des associations entre produits, qui sont achetés ensemble : on parle de cooccurrences d'achat. Extraire de telle règles sert dans la grande distribution à concevoir et optimiser le placement des produits dans les rayons, mais on peut aussi voir un moyen de faire des recommandations immédiates à n'importe quel client ayant fait un achat ou visité une ou plusieurs pages du site, ou encore une manière de mettre en place des promotions sur des achats groupés. On s'affranchit alors par la même occasion de la nécessité de reconnaissance unique du client. Il suffit alors qu'un client montre un intérêt vis-à-vis d'un produit faisant l'objet d'une règle pour qu'on puisse lui faire des recommandations.

Pour obtenir ces règles, nous allons utiliser une implémentation en R de l'algorithme Apriori disponible dans le package **arules** ou le logiciel **IBM SPSS Modeler Suite** qui comprend des outils d'analyse du panier (Basket Analysis). Afin d'avoir des règles pertinentes et actuelles, nous extrayons le contenu des transactions de 10 000 clients qui ont montré une activité depuis janvier 2015. On commence par dédoublonner le contenu de chaque transaction (Seule la présence du produit nous intéresse et pas sa quantité) et supprimer les produits achetés une seule fois puis l'on supprime les transactions mono-produits. L'algorithme fouille l'ensemble des transactions à la recherche de produits achetés ensemble de manière fréquente pouvant donner lieu à des règles de la forme : $\{Antécedent\} \rightarrow \{Conséquence\}$ comme figure 19 ci-dessous. L'exemple en accompagnement montre bien que ce type de recommandation comprend la notion de « style ».

Antécédent	Conséquence	% de support	% de confiance
MAYA ARMOIRE 2 P COUL. and MAYA LIT 140x190 CM CHENE	MAYA CHEVET 2 TIROIRS	0,185	100
SHELBY NOIR OPTION BALDAQUIN	SHELBY NOIR LIT 140X190 CM	0,152	100
BIBOX CAISSON BAS 90X70 CB05 and BIBOX CAIS.BAS 40X70 BLC CB01	BIBOX CAIS.BAS 60X70 BLC CB02	0,152	100
RIVA MIROIR BLANC	RIVA BUFFET 2P/3T BLANC/NOIR	0,118	100
RIVA VITRINE 4P BLC/NOIR	RIVA BUFFET 2P/3T BLANC/NOIR	0,118	100
RAMASSE COUVERT L.90 DPA	BIBOX CAIS.BAS 60X70 BLC CB02	0,118	100
HANNA MIROIR L192 and HANNA TABLE BASSE CARREE	HANNA BUFFET 4P	0,118	100
HANNA MIROIR L192 and HANNA TABLE CARREE	HANNA BUFFET 4P	0,118	100

Figure 19 Exemple de règle d'association en images. Sortie SPSS, huit premières règles d'association du tri sur le support.

Chaque règle d'association est accompagnée de deux valeurs donnant une idée de sa robustesse :

- Le pourcentage de support : Pourcentage du total des transactions où apparaissent l'antécédent et la conséquence. C'est un indicateur de « fiabilité » de la règle.
- Le pourcentage de confiance : Fréquence à laquelle la règle est vérifiée. Il dénote la « force » de la règle.

L'algorithme demande à fixer des seuils minimaux pour les deux valeurs ci-dessus afin d'isoler les règles les plus pertinentes et les plus généralisables, car une bonne règle a un support et une confiance élevés.

Nous générons ainsi, en fixant le support minimal à 70% et la confiance minimale à 100% 135 règles claires et directement intégrables dans les systèmes BUT.

VI.Conclusion

Le filtrage collaboratif draine beaucoup d'énergie et de recherche en statistiques appliquées au marketing. Des outils intégrables avec les systèmes d'information existants et de plus en plus complets sont développés par une communauté très active.

Une grande majorité des outils de recommandations sont cantonnés à l'affichage sur des pages web, mais l'évolution de l'industrie de la distribution vers l'omnicanal pousse à penser de nouvelles manières de faire, comprenant l'utilisation combinée de tous les moyens de contact à disposition. Il faut en effet aujourd'hui pouvoir être présent en magasin, sur internet, sur mobile, etc.

Dans une optique d'étude, nous sommes partis d'une matrice contenant les notes laissées par les utilisateurs du site BUT.fr aux produits, chacun en laissant un petit nombre. Nous avons ensuite prédit les notes que donneraient les utilisateurs aux produits qu'ils n'ont pas évalués de trois manières différentes. Afin d'évaluer la qualité de nos prédictions, nous avons monté une étude basées sur des données simulées puis effectué des tests sous plusieurs cas de figures possibles. Le moins que l'on puisse dire est que le jeu de données But.fr dont nous nous sommes servis contenaient trop peu de données pour permettre un apprentissage correct et la génération de recommandations précises avec les méthodes testées.

Les tests ont montré que dans le cas classique où les données sont liées linéairement, la méthode par ACP reste la plus précise. Elle semble l'être d'autant plus quand le volume des données augmente. Elle se montre donc comme étant le meilleur choix quand la précision compte. Elle reste cependant très demandeuse en ressources et donc lente à exécuter. La solution à cela est encore une fois de distribuer les calculs. Cette conclusion prend tout son sens quand on sait que l'équipe ayant gagné le million de dollars du Netflix Challenge s'est servi notamment d'un algorithme se basant tout comme l'ACP sur la recherche de facteurs latents dans la matrice de notations, avec une régularisation.

Enfin, nous avons présenté une manière qui implique une personnalisation moindre des recommandations mais qui néanmoins permet à tout vendeur de générer un ensemble de règles pouvant être très utiles dans l'optimisation de son marketing, tout en lui permettant de mieux comprendre ses ventes : Association de produits, ambiances, etc. Cette approche permet de tirer parti du fait que la majorité des ventes de BUT se font en magasin et que la base centrale BUT contient les transactions faites par plus de 10 millions de clients.

Table des figures

FIGURE 1 EXEMPLES DE RECOMMANDATIONS AFFICHEES SUR LE SITE NETFLIX.COM (SOURCE : NETLIX.COM) 6
FIGURE 2 RECOMMANDATIONS AFFICHEES PAR LE SITE BUT.FR SUITE A LA CONSULTATION DE LA PAGE D'UN REFRIGERATEUR (SOURCE : BUT.FR) 7
FIGURE 3 LES TROIS TYPES DE REPARTITION DES DONNEES MANQUANTES. (SOURCE : PAGE DU PR. PHILIPPE BESSE - GOO.GL/UP1REK) 12
FIGURE 4 FONCTIONNEMENT DE LA METHODE DES K-PLUS PROCHES VOISINS DANS LE CADRE DE LA CLASSIFICATION POUR K = 3 ET K = 7 (SOURCE : GOO.GL/GBCWWB) 13
FIGURE 5 UN EXEMPLE SIMPLE D'ARBRE DE DECISION PERMETTANT DE DETERMINER SI UN PATIENT EST MALADE A PARTIR DE SA TEMPERATURE CORPORELLE (VARIABLE QUANTITATIVE AYANT UNE IMPORTANCE PRIMAIRE DANS L'ARBRE) ET DE L'ETAT DE SA GORGE (VARIABLE QUALITATIVE BINAIRE). (SOURCE : SITE DE L'UNIVERSITE DE LILLE 3 - GOO.GL/H2WYOP) 14
FIGURE 6 EFFET DU POURCENTAGE DE DM SUR L'EFFICACITE DE L'IMPUTATION PAR FORETS ALEATOIRES (RF), ACP ITERATIVES (PCA) ET K-PLUS PROCHES VOISINS (KNN) PAR RAPPORT A L'IMPUTATION PAR LA MOYENNE (MEAN) ET LA MEDIANE (MEDIAN) LE POURCENTAGE DE DM EST EN ABSCISSE ALORS QUE LA PRECISION DE LA PREDICTION EST EN ORDONNEE. . 17
FIGURE 7 VISUEL D'UN EMAIL DE "DEMANDE D'AVIS" ENVOYE SUITE A L'ACHAT D'UNE BIBLIOTHEQUE. 18
FIGURE 8 REPRESENTATION GRAPHIQUE DES LIENS ENTRE VARIABLES LORSQU'ELLES SONT RELIEES PARS UNE CORRELATION LINEAIRE DE VALEUR R = 0.3 (COEFFICIENT DE PEARSON) 19
FIGURE 9 REPRESENTATION GRAPHIQUE DES LIENS ENTRE VARIABLES LORSQU'ELLES SONT RELIEES PARS UNE CORRELATION LINEAIRE DE VALEUR R = 0.9 (COEFFICIENT DE PEARSON) 19
FIGURE 10 EFFET DE LA LIAISON LINEAIRE ENTRE VARIABLES, CAS FAIBLE R = 0.3 20
FIGURE 11 EFFET DE LA LIAISON LINEAIRE ENTRE VARIABLES, CAS MOYEN-FORT R = 0.6 21
FIGURE 12 EFFET DE LA LIAISON LINEAIRE ENTRE VARIABLES, CAS FORT R = 0.9 22
FIGURE 13 REPRESENTATION GRAPHIQUE DES LIENS ENTRE VARIABLES LIEES NON LINEAIREMENT. 23
FIGURE 14 QUALITE DE LA PREDICTION EN CONDITIONS NON-LINEAIRES 23
FIGURE 15 REPRESENTATION GRAPHIQUE DES LIENS ENTRE VARIABLES LIEES NON LINEAIREMENT DE MANIERE COMPLEXE. 24
FIGURE 16 IMPACT DE LIAISONS NON-LINEAIRES COMPLEXES ENTRE LES VARIABLES SUR LA QUALITE DE LA PREDICTION. 25
FIGURE 17 IMPACT DE L'AUGMENTATION DU NOMBRE DE LIGNES SUR LA QUALITE DE PREDICTION 26
FIGURE 18 IMPACT DE L'AUGMENTATION DU NOMBRE DE COLONNES SUR LA QUALITE DE PREDICTION. 27
FIGURE 19 EXEMPLE DE REGLE D'ASSOCIATION EN IMAGES. SORTIE SPSS, HUIT PREMIERES REGLES D'ASSOCIATION DU TRI SUR LE SUPPORT. 30

Bibliographie

Agrawal, Rakesh, Tomasz Imieliński, and Arun Swami. "Mining Association Rules between Sets of Items in Large Databases." ACM SIGMOD Record SIGMOD Rec. 22.2 (1993): 207–216. Web.

Azur, Melissa J. et al. "Multiple Imputation by Chained Equations: What Is It and How Does It Work?" *International Journal of Methods in Psychiatric Research Int. J. Methods Psychiatr. Res.* 20.1 (2011): 40–49. Web.

Baraldi, Amanda N., and Craig K. Enders. "An Introduction to Modern Missing Data Analyses." *Journal of School Psychology* 48.1 (2010): 5–37. Web.

Daniel J. Stekhoven (2013). missForest: Nonparametric Missing Value Imputation using Random Forest. R package version 1.4.

Dray, Stéphane, and Julie Josse. "Principal Component Analysis with Missing Values: a Comparative Survey of Methods." *Plant Ecol Plant Ecology* 216.5 (2014): 657–667. Web.

Francois Husson and Julie Josse (2015). missMDA: Handling Missing Values with Multivariate Data Analysis. R package version 1.8.2.

H. Wickham. ggplot2: elegant graphics for data analysis. Springer New York, 2009.

Ishioka, Tsunenori. "Imputation Of Missing Values for Semi-Supervised Data Using the Proximity in Random Forests." *IJBIDM International Journal of Business Intelligence and Data Mining* 8.2 (2013): 155. Web.

Lingbing Feng, Gen Nowak, Alan. H. Welsh and Terry. J. O'Neill (2014). imputeR: A General Imputation Framework in R. R package version 1.0.0.

Little, Roderick J. A., and Donald B. Rubin. *Statistical Analysis with Missing Data*. New York: Wiley, 1987. Print.

Marlin, Benjamin M., and Richard S. Zemel. "Collaborative Prediction and Ranking with Non-Random Missing Data." *Proceedings of the third ACM conference on Recommender systems - RecSys '09* (2009): n. pag. Web.

Stekhoven D. J., & Buehlmann, P. (2012). MissForest - non-parametric missing value imputation for mixed-type data. Bioinformatics, 28(1), 112-118.

Trevor Hastie, Robert Tibshirani, Balasubramanian Narasimhan and Gilbert Chu (). impute: impute: Imputation for microarray data. R package version 1.40.0.

Venables, W. N. & Ripley, B. D. (2002) Modern Applied Statistics with S. Fourth Edition. Springer, New York. ISBN 0-387-95457-0

Zhu, Xiaoping. "Comparison Of Four Methods for Handing Missing Data in Longitudinal Data Analysis through a Simulation Study." Open Journal of Statistics OJS 04.11 (2014): 933–944. Web.

Glossaire

Calcul parallèle : Le parallélisme en informatique désigne la mise en œuvre d'architecture matérielles et logicielles permettant de traiter des informations de manière simultanée. Le but de ces techniques est de réaliser le plus grand nombre d'opérations en un temps le plus petit possible. Ce domaine a véritablement décollé avec l'avènement au début des années 2000 du processeur multi-cœurs et son intégration dans l'ordinateur de bureau par la suite. Le calcul parallèle est très utilisé en calcul scientifique.

Classification : Ensemble de méthodes permettant d'organiser des objets par similarité.

Cross-selling : Vente croisée, vente additionnelle. Stratégie de vente qui consiste à profiter d'une vente ou de l'intérêt que manifeste un consommateur pour un produit donné pour proposer et vendre d'autres produits traditionnellement complémentaires ou supérieurs. En contexte e-commerce, l'utilisation des statistiques permet de suggérer des produits encore plus pertinents car personnalisés et ayant donc plus de chances d'intéresser le client, sans que cela soit évident.

Régression : Ensemble de méthodes statistiques permettant d'analyser et de comprendre la relation d'une variable par rapport à une ou plusieurs autres.

Trigger : Un trigger est une action marketing déclenchée à la suite d'évènements ou de comportements prédéfinis d'un client ou d'un prospect. En plus de délivrer un message personnalisé et ciblé au moment adéquat, le trigger permet aussi de diminuer la saturation liée à la profusion de la communication de masse (Rarement pertinente) et signe la transition vers un marketing personnalisé qui s'adapte sans être intrusif. Un exemple de Trigger mis en place chez BUT est l'emailing panier abandonné qui relance automatiquement le consommateur ayant ajouté des produits dans son panier BUT.fr, mais qui a quitté le tunnel de commande au moment du paiement.

Tunnel de commande : Dans un contexte e-commerce, le tunnel de commande (ou tunnel d'achat) est la succession d'étapes entre l'arrivée du consommateur sur le site web et la validation finale de son achat. Tout tunnel requiert un travail minutieux d'optimisation et de simplification car l'abondance d'offre et la facilité d'accès que procure l'outil internet au shopper fait qu'il abandonnera facilement le processus d'achat entamé si celui-ci devient fastidieux ou pas assez clair.

Abstract

Product recommendation is one of today's main methods to generate business and sales online. It thus represents a crucial matter for every modern business-to-consumer firm. Also called collaborative filtering, product recommendation is based on different kinds of calculations that result in the suggestion of suitable products to consumers, in other terms of the presentation of products they may like and/or purchase. This process of course must be automated through designing and implementation of algorithms that manipulate big amounts of consumer data to generate creditable recommendations. In this paper, we explore considering the issue from the missing data imputation point of view. For this purpose, we set up a comparative study approach and implement a simulation algorithm. These make it possible for us to benchmark three popular imputation methods following different constraints that we settle for the study. The imputation methods we assess are Missing Forest, Iterative PCAs and k-Nearest Neighbors. We will compare them to basic imputation with mean and median in extreme missing data conditions. After exploring the limitations of these methods, we expose and carry out another way of product recommendation that uses standard transactional data to look for buying patterns. This approach uses association rules mining and gives good and immediately actionable results.

Keywords: *missing data, recommendation, random forest, iterative PCA, k-Nearest Neighbors, simulations, association rules, collaborative filtering*.

Résumé

La recommandation de produit est un enjeu crucial pour toute entreprise commerciale aujourd'hui. Cette discipline, qui consiste à automatiser l'étude du comportement du client pour ensuite lui suggérer des produits qu'il a des chances d'apprécier est née avec la vente en ligne et connait un fort développement. Au sein du département Études et CRM du groupe BUT, et dans un cadre opérationnel, nous mettons en place une approche statistique de la recommandation de produits. Après avoir mis en relation la recommandation de produits avec la gestion des données manquantes, nous développons dans le présent document une étude comparative de trois algorithmes d'imputation (Forêts aléatoires, ACP itératives et méthodes des k-plus proches voisins). Nous implémentons pour ce faire un algorithme de simulations rendant possible la comparaison précise de l'efficacité des différentes méthodes sur des données que nous générons sous contraintes. L'étude est faite pour des pourcentages de valeurs manquantes atteignant les 95% et considère plusieurs structures possibles de données. Nous mettons ensuite les résultats en lien avec le contexte de l'entreprise BUT avant de présenter une seconde méthode de recommander des produits présentant l'avantage de pouvoir être mise en place relativement aisément, utilisant les règles d'association.

Mots-clés : *données manquantes, recommandation, forêts aléatoires, ACP itératives, k-plus proches voisins, simulations, règles d'association, filtrage collaboratif*.