Prédiction des valeurs manquantes dans un entrepôt de données par combinaison de la programmation par contraintes et des KPPV

Fatiha Amanzougarene*, Karine Zeitouni*
Mohamed Chachoua**

*PRISM, 45, Avenue des États Unis, 78000 Versailles
(fatiha.amanzougarene, karine.zeitouni)@prism.uvsq.fr
**EIVP, 80, rue Rébeval, 75019 ParisAutre adresse
chachoua@eivp-paris.fr

Résumé. La présence de données manquantes dans les grandes bases de données scientifiques et statistiques est très courante. Dans cet article, nous proposons un modèle permettant la reconstruction des données manquantes dans le contexte des entrepôts de données. Notre approche de reconstruction de données manquantes consiste à combiner la programmation par contraintes et une technique d'apprentissage automatique, à savoir l'algorithme des k-plus proches voisins. La programmation par contraintes permet d'inférer les données manquantes à partir des contraintes sommaires définies sur les données de base. La méthode des k-plus proches voisins permet d'augmenter la précision de cette inférence. Outre l'application aux entrepôts de données classiques, nous avons étendu notre approche à des entrepôts de données dits qualitatifs correspondant à notre cas d'étude, à savoir l'évaluation des gênes des chantiers urbains.

1 Introduction et motivation

Les entrepôts de données sont particulièrement exposés à des données manquantes ou des valeurs aberrantes en raison de la qualité des sources ou de diverses incohérences lors de l'intégration des données (Ribeiro et al., 2011). Si les valeurs manquantes ne sont pas traitées, des faits importants peuvent ne pas être pris en considération dans le processus d'analyse.

Pour illustrer la problématique et la solution apportée, nous considérons tout au long de cet article un exemple lié à une application sur les gênes [1] des chantiers urbains [2]. Dans cette application, le sujet d'analyse correspond à la gêne. Ce sujet est étudié selon les dimensions suivantes : le temps, l'espace, les nuisances [3] et les catégories de population (par exemple, adulte cadre en bonne santé, mère au foyer, enfant avec problème de respiration). Les mesures associées au fait gêne sont l'effectif pour chaque catégorie de populations, la durée d'exposition ainsi que le degré de gêne. Les deux premières mesures sont numériques tandis que la

1. On définit la gêne comme étant une relation subjective entre un individu et un phénomène nuisible
2. Ce travail rentre dans le cadre du projet ANR, dénommé Furtivité Urbaine Réseaux et Travaux (FURET)
3. On désigne par nuisance une perturbation du milieu ayant un effet négatif comme le bruit, les odeurs, etc.

dernière est qualitative (ex. négligeable, faible ou considérable). Dans (Amanzougarene et al., 2012), nous avons proposé une extension du modèle multidimenstionnel classique permettant la représentation et la manipulation des données qualitatives ordinales de type degré linguistique.

Pour peupler l'entrepôt *Gêne*, deux sources de données sont utilisées. La première source contient une instanciation des faits par des degrés de gêne. Cette approche a été suivie justement à cause du manque de moyens automatique de mesure de la gêne. Ceci implique des faits sans valeurs. La deuxième source contient des règles correspondant à des seuils de gênes à un niveau agrégé à respecter. Ces seuils sont fixés par le gestionnaire de l'espace publique lors de la programmation des demandes de travaux. Parmi les règles définies par le gestionnaire concernant ses préférences, on peut citer par exemple :

– Niveau de gêne < "Important" sur tout le territoire et à tout moment
– Niveau de gêne < "Considérable" devant une maison de retraite et à tout moment

Pendant l'intégration des données sources dans le schéma multidimensionnel de l'entrepôt *Gêne*, les données manquantes sont remplacées par des valeurs non renseignées. Les seuils de la gêne sont stockés dans la partie métadonnées de l'entrepôt sous forme de règles expertes. Dans le reste de cet article nous allons considérer l'exemple suivant. Soit la table de faits *Gêne* présentée par la figure 1. Certains faits ne sont pas connus et ils sont représentés par des points d'interrogation. On dit qu'un fait est inconnu si la valeur de la mesure correspondant à ce fait est non renseignée. L'échelle linguistique, utilisée pour l'instanciation des degrés de gêne dans cet exemple, est représsentée par l'ensemble ordonné L_5 = {*négligeable, faible, considérable, importante, extrême*}. En outre, supposons que : *la gêne moyenne par zone, par moment et par nuisance doit être inférieure ou égale à "importante"*. Nous adoptons la notation suivante :

$(Zone, Moment, Nuisance, Moyenne(DegréDeGêne)) \leq "importante"$

Table de faits Gêne

Faits	Id_Temps	Id_Espace	Id_Nuisance	Id_Catégorie	Degré de gêne
1	27	102	23	2	Considérable
2	27	102	23	5	Négligeable
3	27	102	23	7	Importante
4	27	102	25	47	Négligeable
5	27	102	25	48	?
6	27	102	25	78	Extrême
7	30	104	102	14	?
8	30	104	102	15	Négligeable
9	30	104	102	3	Considérable

FIG. 1 – *Extrait de la table de faits Gêne.*

Les règles sur le niveau de gêne à ne pas dépasser peuvent être considérées comme des contraintes sur les données de la table de faits *Gêne*. Dans l'exemple précédent la moyenne peut être remplacée par une des fonctions d'agrégation qualitatives que nous avons détaillées dans (Amanzougarene et al., 2012).

Notre objectif général est la reconstruction des valeurs manquantes dans les fait d'un entrepôt de données, y compris les entrepôts qualitatifs. Dans l'état de l'art, il existe principalement deux approches de reconstrcution de données manquantes. L'une consiste à induire les données manquantes par similarité en se basant sur l'algorithme des k plus proches voisins. Quant

à la seconde, elle se base sur des contraintes supposées sur les agrégats et utilise la programmation par contraintes. La qualité de la prédiction est néanmoins limitée. Le modèle que nous proposons tire profit à la fois des contraintes sur les agrégats et de la similarité dans le but d'améliorer la précision de cette prédiction.

Nos principales contributions sont : (i) la proposition d'un modèle combinant la programmation par contraintes et la méthode des k-plus proches voisins pour prédire les données manquantes dans les entrepôts de données. A notre connaissance, notre travail est le premier à considérer l'utilisation conjointe de ces deux méthodes pour la prédiction des données manquantes dans le contexte des entrepôts de données ; (ii) la prédiction de données manquantes s'applique aussi bien dans le cas d'un entrepôt de données classique que dans le cas d'un entrepôt qualitatif. A notre connaissance, il n'existe aucune proposition pour des mesures qualitatives ; (iii) l'optimisation du traitement de ces données manquantes par un partitionnement préalable de la base de données.

Le reste de cet article est organisé comme suit. Dans la section 2, nous présentons un aperçu de l'état de l'art. Dans la section 3, nous décrivons notre modèle permettant la reconstruction de données manquantes dans le contexte des entrepôts de données. La section 4 est dédiée à l'étude expérimentale. Enfin, nous résumons nos contributions en conclusion de cet article et traçons quelques perspectives.

2 Travaux connexes

Les travaux concernant le traitement de données manquantes dans le contexte des bases de données multidimensionnelles peuvent être classés en deux catégories. La première catégorie se focalise sur la représentation et la modélisation d'agrégats multidimensionnels en présence de données manquantes au niveau des mesures ou au niveau des dimensions. Cette catégorie peut à son tour être décomposée en deux classes. Dans la première classe, on trouve les travaux de (Dyreson, 1997), (Barbará et Sullivan, 1997) qui s'intéressent au traitement de requêtes multidimensionnelles portant sur des agrégats incomplets sans recours aux données de base. L'objectif principal de ces travaux est d'améliorer le temps d'exécution des requêtes ou de diminuer l'espace de stockage des données. Dans la seconde classe, on trouve les travaux de (Jensen et al., 2004), (Timko et al., 2005) qui proposent des méthodes de représentation et d'agrégation des données permettant la prise en compte des données manquantes au niveau des hiérarchies de dimensions. La deuxième catégorie de travaux s'intéresse à la reconstruction des données de base (Dyreson et al., 2003). Notre travail se situe dans la deuxième catégorie de travaux. Dans cette catégorie on distingue deux sous-catégories de travaux. La première sous-catégorie propose des méthodes de déduction de données manquantes à partir des informations sommaires disponibles dans la base en s'appuyant sur des méthodes statistiques et mathématiques comme la régression linéaire et la programmation par contraintes (Wu et Barbará, 2002). Dans la deuxième sous catégorie (Ribeiro et al., 2011) les auteurs proposent une méthode de reconstruction de données de base en s'appuyant sur la méthode des k-plus proches voisins et la provenance des données. Dans un domaine différent, le Reverse Data Management (*RDM*) (Meliou et al., 2011) permet de générer des données de base à partir de données agrégées. Son but est de modifier les données sources d'une base de données afin d'atteindre un objectif souhaité. Un exemple d'objectifs peut être l'indicateur clef de performance. Parmi les tâches de gestion de données qui révèlent du paradigme *RDM*, on trouve la mise à jour au travers

des vues (Feng et al., 2012), la génération de données synthétiques (Arasu et al., 2011) et les requêtes de type *What-If* et *How-To* (Antova et al., 2007), (Meliou et al., 2011).

3 Modèle proposé

3.1 Définition du problème

Cette section présente la définition du problème de reconstruction des données manquantes dans le contexte d'une base de données multidimensionnelle. Cette reconstruction s'appuie sur la combinaison de la programmtion par contraintes (Wu et Barbará, 2002) et l'algorithme *KPPV* (Hruschka et al., 2004) et suppose deux points importants : (1) la présence de valeurs manquantes uniquement au niveau des données de base ; (2) l'existence des informations sommaires tels que des agrégats définis sur les données de base.

Définition 1 Nous définissons un problème de reconstruction de données manquantes par un quadruplet (X, D, C, D') où :

- $X = \{X_1, X_2, ..., X_n\}$ est l'ensemble des variables correspondant aux valeurs manquantes d'une mesure donnée, soit m ;
- D correspond au domaine de définition de la mesure m, tel que D est un domaine numérique si m est de nature quantitative et D est un ensemble de termes linguistiques si m est de nature qualitative ;
- $C = \{C_1, C_2, ..., C_p\}$ est l'ensemble des contraintes définies sur m tel que :
  $C_i : f_{AG}(..., X_h, ..., X_k, ...) \leq ValeurAgrégat$ où $h, k \in [1, n]$ et f_{AG} est une fonction d'agrégation numérique ou qualitative. Nous appelons C_i une contrainte numérique si elle est définie par une fonction d'agrégation numérique et une contrainte qualitative si elle est définie par une fonction d'agrégation qualitative ;
- D' est une fonction d'élagage qui définit le domaine réduit de chaque variable X_i, notée $(D'(X_i))$. D' résulte de l'étape de raffinement par *KPPV*.

3.2 Méthode et architecture

Notre approche s'inspire du paradigme *RDM*, où *KPPV* est utilisée pour améliorer la qualité de la prédiction. Ainsi, nous proposons un modèle de reconstruction de données manquantes nommé *RDM2* pour «Reconstruction de Données Manquantes par Reverse Data Management». En effet, notre approche consiste à :

1. Calculer les domaines $(D'(X_i))$: pour calculer le domaine réduit de chaque variable nous proposons d'utiliser l'algorithme *KPPV*.
2. Résoudre le *CSP* (X, D', C) avec un solveur de programmation par contraintes. Selon le type des contraintes, nous distinguons deux cas : (a) contraintes numériques : dans ce cas il suffit de spécifier les contraintes, leur résolution étant prise en charge automatiquement par un solveur de contraintes intégrés au langage de programmation choisi ; (b) contraintes qualitatives : à notre connaissance aucun solveur classique ne permet de prendre en compte le type de contraintes qualitatives que nous définissons dans notre étude de cas. Pour cela, nous proposons d'étendre le solveur classique en intégrant les fonctions d'agrégation qualitatives définies pour les degrés linguistiques.

L'architecture de $RDM2$ est schématisée par la figure 2 ci-dessous. Après connexion à la base de données multidimensionnelle *(BDM)*, l'utilisateur choisit les paramètres de la reconstruction qui sont le nom de la table de faits, la mesure contenant des données manquantes, les attributs de similarité et les attributs de partitionnement - pour ces deux derniers paramètres voir l'explication ci-dessous -. Ensuite, ces informations sont transmises au module *Partitionnement de la BDM*, qui effectue un partitionnement horizontale de la table de faits. Chaque partie sera traitée indépendamment par un solveur *CSP*, et ce dans un but d'optimisation du processus global $RDM2$. Ces parties sont lues par le module *Constructeur CSP*. Pour chaque partie, ce module définit ainsi l'ensemble des variables et des contraintes constituant le sous-problème *CSP* associé. La liste de ces problèmes est reçue par le module *Solveur CSP*. Ce module déclenche le solveur de programmation par contraintes adéquat, selon que les contraintes sont numériques ou qualitatives. Enfin, les solutions sont écrites dans la table de faits.

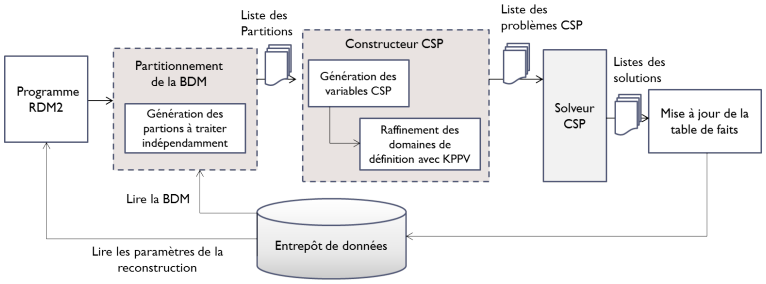

FIG. 2 – *Architecture du système.*

Attributs de similarité. La sélection des *attributs de similarité* doit être effectuée par un expert. Pour aider cet expert dans son choix, nous proposons de calculer une table de faits dénormalisée. Cette table est obtenue par la jointure de la table de faits avec ses dimensions. Le choix des attributs de dimensions peut s'appuyer sur plusieurs stratégies liées au contexte de l'étude considérée. Par exemple dans le cas de l'entrepôt *Gêne*, les attributs sélectionnés correspondent aux paramètres d'évaluation de la gêne, à savoir le mois ou la période de l'année (hiver ou été), le jour (de semaine ou de week end), le moment (jour, soir et nuit), le type de nuisance (bruit, odeur, pollution...) et son intensité et enfn, la calégorie de population.

Attributs de partitionnement. Les contraintes définies sur les données de base sont exprimées sous forme d'agrégats concernant certains attributs de dimensions. Ces attributs dits *attributs de partitionnement* sert de critère pour découper horizontalement la table de faits. Par exemple, reprenons la table de faits de la figure 1. La contrainte définie pour cette table peut être exprimée par *la gêne moyenne par moment, par zone géographique et par nuisance pour toutes catégories de population confondues doit être inférieure ou égal à importante*. Ainsi, les attributs de partitionnement de cette table sont : *Id-Temps* , *Id-Espace* et *Id-Nuisance*.

3.3 Partitionnement des données

Pour chaque combinaison de valeurs des attributs de partitionnement, le module de partitionnement construit des parties virtuelles i.e. des listes constituées des identifiants des faits. Par exemple, le partitionnement de la table de faits *Gêne*, selon les attributs *Id-Temps*, *Id-Espace* et *Id-Nuisance*, produit trois parties, comme indiqué par les cadres de la figure 1.

3.4 Modélisation CSP

La modélisation CSP consiste à définir les problèmes CSP correspondant aux différentes parties de données. Chaque problème est constitué d'un ensemble de variables, soit X et d'une liste de contraintes, soit C. Les variables sont associées aux données manquantes. Les contraintes correspondent aux agrégats définis par l'utilisateur. Pour chaque variable X_i, est associé un domaine de définition, soit $D(X_i)$ et un domaine de définition réduit, soit $D'(X_i)$. Le domaine de définition est commun à toutes les variables et il correspond au domaine de définition de la mesure représentant les valeurs manquantes. Le domaine réduit d'une variable X_i issue d'un fait noté F_{X_i} est l'ensemble des valeurs correspondant aux faits similaires à F_{X_i}. Dans ce qui suit, nous allons appeler cet ensemble par "*valeurs possibles*".

La génération des valeurs possibles permet de restreindre le domaine de recherche des solutions et d'augmenter la précision des données estimées. Le processus de génération de valeurs possibles (cf. le programme ci-dessous) est appliqué pour chaque variable $X_i \in X$. Il est basé sur l'algorithme *KPPV* qui consiste à sélectionner à partir de la table de faits dénormalisée les *k* faits plus similaires à F_{X_i}. L'idée est que les faits ont tendance à être similaires lorsque les valeurs des attributs de dimensions tendent à être similaires.

Algorithm 1 Processus de génération des valeurs possibles

Soit $X = \{X_1, X_2, ..., X_n\}$ *la liste des variables correspondant aux valeurs manquantes*
Soit $T = \{F_1, F_2, ..., F_N\}$ *la liste des faits, Card (T)=|T|=N*
Soit $L_{X_i} = null$ *la liste des valeurs possibles de la variable* X_i
Soit F_{X_i} *le fait correspondant à la variable* X_i
$foreach(X_i \; in \; X)$ // Calculer les valeurs possibles pour chaque variable
{
  $foreach(F_j \; in \; T)$ // appel de kppv pour chaque variable
    $Similar[F_j] = Distance(F_{X_i}, F_j)$ // Calculer les similarités
  $SortDescending(Similar[])$; // Trier les distances
  $L_{X_i} = SelectKNN()$ // Sélectionner les valeurs des k faits plus similaires
}

3.5 Reconstruction des données manquantes dans la table de faits

Une fois la liste des problèmes CSP construite, le solveur CSP permet de calculer les données manquantes en résolvant les différents problèmes CSP. La dernière étape de la renconstruction des données manquantes consiste à mettre à jour la table de faits.

3.6 Exemple d'application

Le problème de la reconstruction de données manquantes correspondant à l'exemple de la section 1 peut être modélisé par le quadruplet (X, D, C, D') défini comme suit :

- $X = \{X_1, X_2\}$, tel que $X_1 et X_2$ correspondent respectivement aux valeurs des faits 5 et 7. Appelons ces valeurs par F_5 et F_7
- D correspond à l'échelle utilisée pour l'évaluation de la gêne, tel que : $D = \{négligeable,\ faible,\ considérable,\ importante,\ extrême\}$
- $C = \{C_1, C_2\}$, tel que : $C_1 :\ OMin(F_4, F_5, F_6) \leq "importante, C_2 :\ OMin(F_7, F_8, F_9) \leq "importante$
- D' définit le domaine de définition réduit des variables $X_1 et X_2$.

La réducution de domaine de définition d'une variable X_i associée à un fait F_{X_i}, consiste à chercher des faits similaires à F_{X_i} et à considérer comme valeurs possibles de X_i les valeurs correspondantes à ces nouveaux faits. La recherche des faits similaires s'appuie non seulement sur les instances de la table des faits, mais aussi sur leur jointure avec les tables de dimensions, de manière à prendre en compte les valeurs des attributs de dimensions.

Table de faits dénormalisée

Faits	Nom IRIS	Mois	Semaine ou Weekend	Moment de la journée	Type Nuisance	Intensité	Nom Catégorie	Degré de gêne
1	Zone 1	Janvier	Semaine	Jour	Bruit	3	Catégorie 1	Considérable
2	Zone 1	Janvier	Semaine	Jour	Bruit	3	Catégorie 2	Négligeable
3	Zone 1	Janvier	Semaine	Jour	Bruit	3	Catégorie 3	Extrême
4	Zone 1	Janvier	Semaine	Soir	Bruit	2	Catégorie 1	Négligeable
5	Zone 1	Janvier	Semaine	Soir	Bruit	?	Catégorie 2	?
6	Zone 1	Janvier	Semaine	Soir	Bruit	2	Catégorie 3	Extrême
7	Zone 1	Janvier	Semaine	Jour	Bruit	?	Catégorie 1	?
8	Zone 1	Janvier	Semaine	Jour	Bruit	2	Catégorie 2	faible
9	Zone 1	Janvier	Semaine	Jour	Bruit	3	Catégorie 3	Considérable

FIG. 3 – *Extrait de la table de faits Gêne dénormalisée.*

La figure 3 présente la table de faits dénormalisée, où chaque fait contient également des valeurs provenant des n-uplets de dimensions correspondantes. Par exemple, (*Mois*, *Semaine Ou Weeked*, *Moment De La journée*) pour la dimension Temps, (*Nom Zone*) pour la dimension Espace, (*Type Nuisance*, *Intensité Nuisance*) pour la dimensions Nuisance et (*Nom De La Catégorie*) pour la dimension Catégorie de population. Sans cette information, l'estimation aurait été calculée seulement en fonction de la valeur des clés étrangères des dimensions, ce qui ne permet qu'une comparaison exacte et non par similarité sur les propriété du fait en question. Par exemple, le fait 7 a des valeurs manquantes, mais on peut observer qu'il est similaire au fait 1. Si l'on se base uniquement sur la combinaison d'identifiants de dimensions, ils n'auraient pas été considérés similaires, vu que le fait 1 est représenté par $< 1, 27, 102, 23, 2 >$ et le fait 7 par $< 30, 104, 102, 14 >$. Pourtant, les deux correspondent à l'exposition de la catégorie 1 à une nuisance de type bruit, pendant la journée en semaine, avec des zones et des mois similaires. Ainsi, le fait 5 est similaire aux faits 2 et 8, et le fait 7 est similaire au fait 1. Par conséquent, les domaines réduits des variables X_1 et X_2 sont respectivement : $D'(X_1) = \{négligeable,\ faible\}$, $D'(X_2) = \{considérable\}$. Les valeurs finales de X_1 et X_2 sont calculées par le solveur CSP. Il associe la variable X_1 avec le degré *négligeable* ou le degré *faible* et la variable X_2 avec le degré *considérable*.

La section suivante présente une évaluation de notre modèle. Cependant, pour des raisons de limitation du nombre de pages nous présentons dans cet article uniquement le cas qualitatif.

4 Evaluation expérimentale

Pour valider notre modèle, nous avons utilisé une table de faits avec une seule mesure qualitative de 25 degrés linguistiques. Partant de notre exemple de cas réel, nous avons généré des données synthétiques afin de contrôler les paramètres d'évaluation et la taille des données. Les différentes tailles de données, peuvent aller de 5000 à 1 Million faits. Pour chaque taille, nous avons généré des tables avec des pourcentages de données manquantes de $10\%, 20\%, 30\%, 40\%, 50\%, 60\%, 70\%, 80\%$. Pour simuler ces pourcentages de données manquantes tout en permettant l'évaluation de leur estimation, nous avons procédé par masquage des données réelles ou initiales correspondantes. Les principaux critères d'évaluation que nous considérons ici sont la précision et le temps d'exécution de notre programme de reconstruction de données manquantes. Pour calculer la précision de la reconstruction des données manquantes nous avons utilisé la mesure dite *-Relative Absolute Derivation- (RAD)* (Ribeiro et al., 2011). Cette mesure correspondant au taux d'erreur entre les données réelles et les données estimées.

$RAD = \frac{1}{n}\sum_{i=1}^{n}\frac{|x_R^i - x_E^i|}{x_R^i}$, tel que :

x_R^i est la valeur réelle, x_E^i est la valeur estimée et n est le nombre de données manquantes.

Cette mesure est valable pour les données numériques et les données qualitatives ordinales comme dans notre étude de cas, où les données sont des degrés linguistiques. La différence entre deux degrés correspond au nombre de degrés qui les séparent. Par exemple, dans l'exemple précédent, si la valeur réelle correspondant à X_2 est *considérable* et si le solveur CSP lui associé le degré *négligeable*, alors la différence est égale à 2 car il y a deux degrés qui séparent *considérable* et *négligeable*.

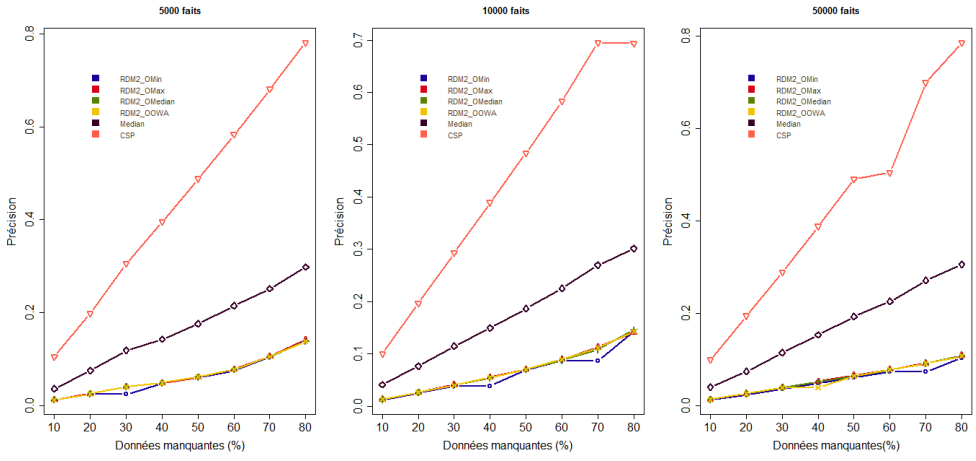

FIG. 4 – *La précision en focntion de la méthode de reconstruction.*

La figure 4 montre que, par la combinaison de *KPPV* et de *CSP*, notre modèle améliore nettement la précision des résultats comparé à la reconstruction qui s'appuie uniquement sur *CSP*. La même figure 4 montre que notre modèle permet de reconstruire les données manquantes avec une précision beaucoup plus importante que la moyenne (la médiane pour le cas qualitatif) qui est la méthode la plus couramment utilisée pour le traitement de données manquantes dans le contexte des base de données multidimensionnelles. Cela est valable quelque soit la moyenne qualitative utilisée pour le calcul des agrégats qualitatifs (*OMin*, *OMax*, *OMedian*, *OOWA*). Le temps d'exécution en fonction de la taille des données est montré par la figure 5. Le partitionnement de la table de faits permet des temps d'exécution raisonnables, même lorsque la taille des données et les pourcentages des données manquantes sont très importants.

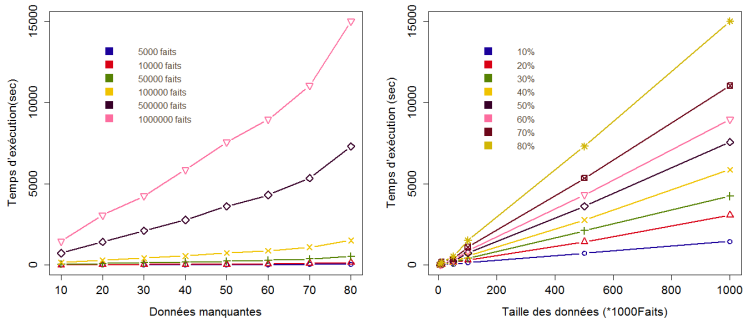

FIG. 5 – *Le temps d'exécution en fonction de la taille des données.*

5 Conclusion

Dans cet article, nous avons présenté un modèle de reconstruction de données manquantes dans le contexte des entrepôts de données. Ce modèle consiste à combiner la programmation par contraintes et une technique d'apprentissage automatique, à savoir l'algorithme des k-plus proches voisins. La programmation par contraintes permet d'inférer les données manquantes à partir des contraintes sommaires définies sur les données de base. La méthode des k-plus proches voisins permet d'augmenter la précision de cette inférence. Nous avons montré que notre modèle de reconstruction de données manquantes peut aussi bien s'appliquer aux entrepôts de données classiques qu'aux entrepôts qualitatifs.

Références

Amanzougarene, F., M. Chachoua, et K. Zeitouni (2012). Qualitative representation of building sites annoyance. In *Proceedings of the 2012 ACM Workshop on City Data Management Workshop*.

Antova, L., C. Koch, et D. Olteanu (2007). From complete to incomplete information and back. In *Proceedings of the 2007 ACM SIGMOD International Conference on Management of Data*, New York, NY, USA.

Arasu, A., R. Kaushik, et J. Li (2011). Data generation using declarative constraints. In *Proceedings of the 2011 ACM SIGMOD International Conference on Management of Data*, New York, NY, USA.

Barbará, D. et M. Sullivan (1997). Quasi-cubes : Exploiting approximations in multidimensional databases. *SIGMOD*.

Dyreson, C. (1997). Using an incomplete data cube as a summary data sieve. In *In IEEE Data Eng. Bull*, pp. 19–26.

Dyreson, C. E., T. B. Pedersen, et C. S. Jensen (2003). Incomplete information in multidimensional databases. In M. Rafanelli (Ed.), *Multidimensional Databases : Problems and Solutions*, Chapter Incomplete, pp. 282–309. Hershey, PA, USA : IGI Global.

Feng, H., N. Lumineau, M.-S. Hacid, et R. Domps (2012). Hierarchy-based update propagation in decision support systems. In *Proceedings of the 17th International Conference on Database Systems for Advanced Applications - Volume Part II*, Berlin, Heidelberg.

Hruschka, E. R., E. R. Hruschka, et N. F. F. Ebecken (2004). Towards efficient imputation by nearest-neighbors : A clustering-based approach. In *Proceedings of the 17th Australian Joint Conference on Advances in Artificial Intelligence*, Berlin, Heidelberg.

Jensen, C. S., A. Kligys, T. B. Pedersen, et I. Timko (2004). Multidimensional data modeling for location-based services. *The VLDB Journal The International Journal on Very Large Data Bases*.

Meliou, A., W. Gatterbauer, et D. Suciu. (2011). Reverse data management. *VLDB 4*(12).

Ribeiro, L., R. R. Goldschmidt, et M. C. Cavalcanti (2011). Complementing data in the etl process. In *Proceedings of the 13th International Conference on Data Warehousing and Knowledge Discovery*, DaWaK'11, pp. 112–123.

Timko, I., C. E. Dyreson, et T. B. Pedersen (2005). Probabilistic data modeling and querying for location-based data warehouses. In *IN PROCEEDINGS OF 17TH INTERNATIONAL SCIENTIFIC AND STATISTICAL DATABASE MANAGEMENT CONFERENCE (SSDBM*, pp. 273–282.

Wu, X. et D. Barbará (2002). Learning missing values from summary constraints. *ACM SIGKDD Explorations Newsletter*.

Summary

The problem of missing data is often encountered in large databases. In this paper, we present a new model to reconstruct missing values in data warehouses. Our model consists in combining the constraint programming and a technique of machine learning, namely the k nearest neighbor algorithm. The constraint programming allows learning the missing values from summary constraints defined on basic data. The k nearest neighbor technique improves the learning precision. In addition to its application in classical data warehouses, our model adapts to qualitative data warehouses, as in the annoyances analysis of urban building sites.