Méthode et outil d'anonymisation des données sensibles

Feten Ben Fredj

le cnam

ÉCOLE DOCTORALE INFORMATIQUE TÉLÉCOMMUNICATIONS ET ÉLECTRONIQUE (PARIS)
CENTRE D'ETUDE ET DE RECHERCHE EN INFORMATIQUE ET COMMUNICATION

FACULTE DES SCIENCES ECONOMIQUES ET DE GESTION DE L'UNIVERSITE DE SFAX
MULTIMEDIA, INFORMATION SYSTEMS AND ADVANCED COMPUTING LABORATORY

THÈSE présentée par :

Feten BEN FREDJ

soutenue le : **3 Juillet 2017**

pour obtenir le grade de : **Docteur du Conservatoire National des Arts et Métiers**

Discipline/ Spécialité : Informatique

Méthode et outil d'anonymisation des données sensibles

THÈSE dirigée par :

PR. Isabelle Wattiau	Professeur, Cnam-Paris
DR. Nadira Lammari	Maître de Conférence-HDR, Cnam-Paris
PR. Faiez Gargouri	Professeur, Université de Sfax

RAPPORTEURS :

PR. Esma AIMEUR	Professeur, Université de Montréal
PR. Ahmed HADJ KACEM	Professeur, Université de Sfax

JURY :

PR. Régine LALEAU	Président de jury, Professeur, Paris-Est Créteil
PR. Jacky AKOKA	Professeur, Cnam-Paris
DR. Sonia AYACHI CHANNOUCHI	Maître de conférences, Institut supérieur de gestion de Sousse7

Dédicaces

*A mes parents **Touhami** et **Jalila** qui m'ont montré le chemin à suivre et m'ont toujours encouragée pour évoluer dans mes études. Que Dieu vous préserve et vous garde en bonne santé.*

*A mon très cher mari **Helmi** , ton amour, ta tendresse, ta consolation, tes sacrifices et ton affection m'ont permis de réaliser ce travail. Que Dieu te préserve pour moi et nos enfants.*

*A ma fille **Khadija** et mon fils **Mohamed Aziz**. Vous êtes la source de mon énergie et mon bonheur, que Dieu vous protège et vous préserve un avenir radieux.*

*A mon frère **Heykel**, sa femme **Nadia** et leurs enfants **Fares** et **Mehdi** ; A ma chère amie **Mouna**.*

A mes amis, mes collègues et toute ma famille ; A tous ceux que j'ai omis de citer ; Je dédie ce travail.

Remerciements

Je remercie mes directeurs de thèse, Dr. Nadira LAMMARI, Pr. Isabelle WATTIAU et Pr. Faiez GARGOURI pour avoir bien voulu m'accueillir dans leurs équipes de recherche et m'avoir initiée au métier d'enseignant chercheur. Mes remerciements s'adressent plus particulièrement à Nadira et Isabelle pour le suivi continu qu'elles m'ont accordé, pour leur présence et leurs encouragements. C'est grâce à leurs conseils et leurs orientations que j'ai pu réaliser ce travail.

Je remercie sincèrement Pr. Esma AIMEUR et Pr. Ahmed HADJ KACEM d'avoir porté autant d'intérêt à ce travail en acceptant d'être les rapporteurs de ma thèse. Je remercie également Pr. Jacky AKOKA, Dr. Sonia AYACHI CHANNOUCHI et Pr. Régine LALOU d'avoir eu l'amabilité de faire partie de mon jury de thèse.

J'adresse mes sincères remerciements à tous les membres de l'équipe ISID pour leur soutien et leurs encouragements. Je remercie plus particulièrement mes amis les doctorants : Houda, Zeineb, Amina, Lydia, Lobna, Anh, Odette, Noura, Sobhi pour toutes les discussions enrichissantes et les bons moments que nous avons partagés ensemble.

Résumé

L'anonymisation des données personnelles requiert l'utilisation d'algorithmes complexes permettant de minimiser le risque de ré-identification tout en préservant l'utilité des données. Dans cette thèse, nous décrivons une approche fondée sur les modèles qui guide le propriétaire des données dans son processus d'anonymisation. Le guidage peut être informatif ou suggestif. Il permet de choisir l'algorithme le plus pertinent en fonction des caractéristiques des données mais aussi de l'usage ultérieur des données anonymisées. Le guidage a aussi pour but de définir les bons paramètres à appliquer à l'algorithme retenu. Dans cette thèse, nous nous focalisons sur les algorithmes de généralisation de micro-données. Les connaissances liées à l'anonymisation tant théoriques qu'expérimentales sont stockées dans une ontologie.

MOTS-CLES : guidage, sécurité, ontologie, méthodologie, respect de la vie privée, anonymisation, approche guidée par les modèles.

Résumé en anglais

Personal data anonymization requires complex algorithms aiming at avoiding disclosure risk without losing data utility. In this thesis, we describe a model-driven approach guiding the data owner during the anonymization process. The guidance may be informative or suggestive. It helps the data owner in choosing the most relevant algorithm given the data characteristics and the future usage of anonymized data. The guidance process also helps in defining the best input values for the algorithms. In this thesis, we focus on generalization algorithms for micro-data. The knowledge about anonymization is composed of both theoretical aspects and experimental results. It is managed thanks to an ontology.

Keywords: guidance, security, ontology, methodology, privacy, anonymization, model-driven approach.

Table des matières

Liste des tableaux

Liste des figures

Liste des annexes

Chapitre 1 Introduction

1. Contexte et problématique de la thèse

Il est connu et reconnu que les données jouent un rôle important dans le développement de la science et de l'innovation. Elles sont aussi, pour les organismes publics et privés, un des plus importants actifs à sécuriser. En plus du fait qu'elles sont une ressource indispensable pour la production de biens et de services, elles apportent des informations cruciales pour la prise de décision, permettant ainsi aux organisations de perdurer et de se démarquer de la concurrence. La collecte et le stockage des données, en vue de leur exploitation et leur partage au sein des organismes, s'avèrent ainsi indispensables. Cependant, l'avènement du numérique conjugué à la croissance constante du nombre des innovations technologiques, a rendu ces données partageables au-delà même des frontières d'un organisme. Il a aussi contribué à sa massification. Ce phénomène s'est accentué par l'engagement des pays sur la voie de l'ouverture et du partage des données publiques, plus connue sous le nom d' « open data ». Cette situation soulève la question du risque de divulgation de données sensibles, et plus particulièrement, la question du risque de violation de la vie privée via l'utilisation de données personnelles[1].

La norme ISO/TS 25237:2008, définit l'anonymisation[2] comme «un processus qui supprime l'association entre l'ensemble de données identifiant et le sujet des données». Ceci sous-entend que, suite à un processus d'anonymisation, les données sont présentées sous une forme qui ne permet pas d'identifier les individus et dont la combinaison avec d'autres données ne devrait pas permettre de les identifier. En d'autres termes, l'anonymisation permet d'obtenir une dé-identification irréversible, contrairement à la pseudonymisation (remplacement d'un nom par un pseudonyme). Cette dernière selon [2] est un «processus par lequel les données perdent leur caractère nominatif. Elle diffère de l'anonymisation car les données restent liées à la même personne dans tous les systèmes informatiques sans que l'identité ne soit révélée.

D'autre part, en règle générale, des données collectées ne peuvent générer des connaissances et donc apporter de la valeur aux entreprises que si elles sont analysées et rapprochées d'autres données. Ce postulat est aussi valable pour des données anonymes. Par conséquent, il apparaît important qu'un processus d'anonymisation puisse aussi offrir des données anonymes de qualité aux professionnels.

Ainsi le processus d'anonymisation des données est une activité complexe dans le sens où elle est soumise à deux risques : le risque de divulgation si des données confidentielles et / ou sensibles sont accessibles par des utilisateurs non autorisés et le risque de fournir des données de faible valeur si le processus d'anonymisation

[1] Selon la Commission Européenne la valeur des données personnelles a le potentiel de croître jusqu'à 1 trillion d'euros annuellement en 2020 (pour les données des citoyens de l'Europe).

[2] Process that removes the association between the identifying data set and the data subject

utilisé réduit leur utilité. Elle exige des compétences techniques suffisantes pour choisir les techniques d'anonymisation appropriées compte tenu du contexte d'utilisation des données.

Plusieurs techniques d'anonymisation existent avec des degrés de fiabilité et des contextes d'applicabilité variables. Le contexte d'applicabilité de ces techniques est caractérisé, entre autres, par l'usage souhaité des données (tel que les tests ou la publication) et par le type de données à anonymiser (telles que micro et macro données tabulaires, données spatio-temporelles, graphes, images, textes, etc.). Le degré de fiabilité est en lien direct avec le risque de ré-identification des données anonymes. Ce risque englobe, selon [3], aussi bien le risque d'individualisation (la possibilité d'isoler un individu), de corrélation (la possibilité de relier des ensembles de données distincts concernant un même individu) que le risque d'inférence (possibilité de déduction d'information sur un individu). Cependant, face à l'évolution des technologies de l'information qui rendent possible le lien entre données de différentes sources, il est quasiment impossible de garantir une anonymisation qui offrirait un risque de ré-identification nul.

Prenant conscience de l'importance de cette problématique, la communauté de chercheurs, composée essentiellement de statisticiens et d'informaticiens, s'est focalisée sur la définition de modèles de protection de la vie privée, sur la production de techniques et d'algorithmes tentant la dé-identification de données sensibles tout en maintenant leur utilité et sur la mise en œuvre de métriques d'évaluation de résultats produits par anonymisation.

Partant de l'état de l'art effectué dans ce contexte, nous avons pu constater la variété des techniques d'anonymisation ainsi que la diversité et la complexité des algorithmes mettant en œuvre une grande partie de ces techniques. Des comparaisons de techniques sont proposées. Même si certaines sont orientées usage, pour autnat, elles ne sont pas accessibles à des éditeurs de données avec de faibles compétences dans le domaine. Les algorithmes associés aux techniques ne sont accessibles qu'à travers les publications de recherche. Leur spécification se rapproche du code de programmation. Ils sont, le plus souvent, partiellement instanciés à l'aide d'exemples. Leurs principes fondamentaux sont décrits textuellement. Par conséquent, ils ne sont compréhensibles que par des informaticiens ou des professionnels ayant des compétences en programmation.

Nous avons aussi pu constater la disponibilité d'outils d'anonymisation. Certains de ces outils sont en libre utilisation. La plupart sont opaques. Même s'ils proposent plusieurs techniques, ils mettent en œuvre, la plupart du temps, un seul algorithme par technique sans mentionner lequel. La plupart de ces outils ne fournissent pas de guidage dans le choix de la technique et de l'algorithme. Ils n'offrent pas une aide au paramétrage des algorithmes implémentés.

Enfin, nous avons pu constater l'absence de base de connaissance où le professionnel chargé de la dé-identification des données pourrait rechercher les connaissances le guidant vers une anonymisation utile et préservant au mieux la vie privée. Il n'existe pas non plus de méthode qui puisse concrétiser le processus d'anonymisation de données tout en offrant des aides à la prise de décision.

De ce constat est née notre question de recherche à laquelle nous avons souhaité répondre dans le cadre de cette thèse : Comment aider un professionnel chargé de l'anonymisation des données à choisir une technique et un algorithme d'anonymisation et comment l'aider à concrétiser son processus de dé-identification ? Cependant, la richesse de la littérature dans ce domaine, nous a amenés à nous concentrer, dans le cadre de cette thèse, aux travaux de recherche ayant trait à l'anonymisation de données confidentielles à des fins de publication, connue sous le nom de PPDP (acronyme anglais de «Privacy Preserving Data Publishing). Plus particulièrement, nous nous sommes focalisés sur le choix d'algorithmes pour une anonymisation par généralisation de micro-données (données atomiques décrivant des individus) se trouvant dans des tables relationnelles.

La généralisation est la technique la plus utilisée dans le cadre de la publication des données. Elle consiste à remplacer les données originales par des données plus générales donc moins précises. Ainsi, par exemple, la date de naissance d'un individu pourrait être remplacée par une tranche d'âge à laquelle il appartient.

2. Résumé des contributions de la thèse

En réponse à la question de recherche posée ci-avant, nous avons, dans un premier temps, extrait de la multitude de papiers de recherche les connaissances requises sur la technique de généralisation et sur ses algorithmes. Ceci nous a permis de fournir :

- Une caractérisation fine de chacun des algorithmes de généralisation qui facilite leur sélection,
- Des descriptions simplifiées des algorithmes de généralisation qui aident à leur compréhension,
- Une ontologie du domaine de l'anonymisation qui formalise toutes les connaissances ainsi extraites.

Dans un second temps, nous avons exploité toutes ces connaissances pour fournir une méthode guidant l'éditeur de données dans son processus d'anonymisation, par généralisation d'une table relationnelle contenant des micro-données confidentielles.

Dans un objectif de validation et d'expérimentation, nous avons aussi outillé notre méthode.

3. Organisation du document

La suite de ce document est organisée en deux grandes parties.

La première partie est structurée en deux chapitres. Le premier chapitre (chapitre 2) est consacré à la présentation du domaine de l'anonymisation des micros données tabulaires. Il rappelle quelques concepts directement liés à cette thèse et fournit une vue d'ensemble des techniques de l'anonymisation et des différents modèles de protection de la vie privée. Le second chapitre (chapitre 3) présente les algorithmes de généralisation les plus connus et quelques métriques d'évaluation proposées dans la littérature. Il décrit et compare les outils d'anonymisation existants en se basant sur les types de guidage utilisés. Ces comparaisons nous ont permis de dégager les limites de ces outils et des approches existantes et d'introduire nos contributions.

La seconde partie est dédiée à la description de nos contributions. Elle est composée de quatre chapitres. Le chapitre 4 détaille une description simplifiée et une typologie plus fine des algorithmes d'anonymisation par généralisation étudiés dans le chapitre 3, selon un processus d'abstraction. Il décrit ensuite une expérimentation menée pour valider les résultats de notre approche d'abstraction.

Le chapitre 5 est dédié à la conception et la mise en œuvre d'OPAM, une ontologie de domaine pour l'anonymisation de micro-données à des fins de publication. OPAM sert à capter, représenter et modéliser des connaissances qui sont exploitables, par l'approche guidée d'anonymisation de micro-données, nommée MAGGO, que nous proposons dans le chapitre 6. MAGGO sert de guide pour un professionnel dans sa prise de décision lors d'une anonymisation par généralisation de micro-données. Cependant, elle est générique afin de pouvoir être appliquée à d'autres techniques d'anonymisation de micro-données.

Le chapitre 7 décrit le prototype relatif à la plateforme de guidage d'un éditeur de données au cours de son processus d'anonymisation.

La conclusion constitue le chapitre 8 de cette thèse. Elle résume nos contributions et propose un ensemble de perspectives.

Chapitre 2 : L'anonymisation de micro-données à des fins de publication

L'avènement du numérique a entrainé la collecte massive de données. Pour exploiter pleinement l'utilité analytique de ces données, ces dernières ont besoin d'être rendues disponibles aux chercheurs et/ou aux professionnels. Cependant, ces données sont susceptibles de contenir des informations que les propriétaires ou les législateurs considèrent comme privées. Par conséquent, la publication et l'externalisation de bases de données doivent maintenir l'équilibre entre l'utilité des données et le droit au respect de la vie privée. L'anonymisation est une réponse à ce besoin car elle permet la réutilisation des données tout en protégeant la vie privée.

Notre contribution s'inscrit dans le cadre de l'anonymisation par généralisation de micro-données tabulaires à des fins de publication. Nous consacrons ce chapitre à la présentation de ce vaste domaine.

Après un rappel de quelques concepts directement liés à cette thèse, nous fournissons une vue d'ensemble de l'anonymisation, à savoir les différentes techniques applicables à l'anonymisation de micro-données. Puis nous dressons un état des lieux détaillé de la recherche dans le domaine concerné par cette thèse, c'est-à-dire l'anonymisation par généralisation.

1 Notions préliminaires

La publication des données préservant la vie privée suit généralement un processus composé de deux phases (voir **Figure 1**) : une phase de collecte et une phase de publication. Elle fait intervenir trois acteurs : un propriétaire (ou « owner »), un éditeur (ou « publisher ») et un destinataire (ou « recipient ») de données. La collecte des données, auprès des propriétaires de données, est effectuée par un éditeur de données.

Figure 1. Collecte et publication des données (Y. Xu et al. 2014)

La préservation de la vie privée dans ce contexte sous-entend la non divulgation d'informations sensibles personnelles, suite à des autorisations d'accès fournies à des destinataires. L'hypothèse émise, dans le cadre de

la publication, est que certains de ces destinataires sont susceptibles d'être des attaquants[3]. Les propriétaires des données sont considérés comme victimes dès lors que leurs données personnelles ont été divulguées sans leur accord. Ainsi, les éditeurs de données doivent gagner la confiance des propriétaires de données en empêchant l'utilisation abusive des informations sensibles qu'ils publient. Pour ce faire, ils doivent mettre en place des processus d'anonymisation qui préservent la vie privée et qui permettent tout de même de mettre des données utiles à disposition des destinataires (a priori «bienveillants»).

Par conséquent, la publication des données préservant la vie privée suppose que celles-ci ont été rendues anonymes par l'éditeur avant d'être transmises aux destinataires. Pour cela, deux scénarios de publication de données sont possibles : la publication interactive et la publication non interactive (Dwork 2008). La publication interactive consiste à dé-identifier en continu des résultats de requêtes avant transmission au destinataire (Blum et al. 2005). La publication non interactive s'emploie à fournir un ensemble de données anonymes à un destinataire (Blum, Ligett, et Roth 2008). Dans cette thèse, nous nous focalisons sur le scénario de la publication non interactive.

D'autre part, dans le cadre de la publication des données préservant la vie privée, les données mises à disposition de l'éditeur sont, le plus souvent, stockées dans une table relationnelle appelée table originale. Ces données peuvent être des micro-données ou des macro-données. Une macro-donnée, telle que le mentionne (V. Ciriani, 2007), est une donnée agrégée décrivant un ensemble d'individus (appelés répondants). Leur anonymisation a pour objectif de garantir qu'une information concernant un individu ne puisse pas être déduite à partir des macro-données.

Une micro-donnée, quant à elle, est une information de base caractérisant un individu vis-à-vis d'un attribut de la table (par exemple, le prénom d'une personne). Elle peut contribuer à l'identification d'un individu ou à sa quasi-identification. Elle peut aussi être sensible ou non sensible. Ainsi, dans une table relationnelle représentant des individus, on peut retrouver quatre groupes distincts d'attributs : l'identifiant explicite, le quasi-identifiant, le groupe d'attributs sensibles et le groupe d'attributs non sensibles que nous définissons comme suit (V. Ciriani, 2007):

Définition 2.1. Un **identifiant explicite** (IE) est un attribut ou un ensemble d'attributs qui désigne directement un individu (par exemple, un numéro de sécurité sociale, un prénom, un nom). Ce n'est pas nécessairement un identifiant au sens de la modélisation conceptuelle, puisqu'un prénom et/ou un nom peuvent être partagés par plusieurs individus. Toutefois, au sein d'un jeu de données, ce type d'information nominative peut facilement conduire à une ré-identification.

Définition 2.2. Un **quasi-identifiant** (QI) est un ensemble d'attributs pouvant être utilisé pour identifier, de façon indirecte, au moins un individu parmi ceux décrits dans la table, en liant ces attributs à des sources externes de

[3] **L'attaquant ou l'adversaire** est un terme utilisé en sécurité informatique, et notamment en cryptographie, pour désigner l'utilisateur potentiellement mal intentionné, contre lequel on met en place des obstacles pour éviter toute divulgation d'information confidentielle**Source spécifiée non valide.**

données. Par exemple {sexe, code postal, date de naissance} forme un quasi-identifiant connu dans de nombreux ensembles de données. Ils sont suffisamment discriminants pour permettre de retrouver un seul individu, même dans un très grand ensemble de données.

*Définition 2.*3. Un **attribut sensible** (AS) représente les données que les individus ne veulent généralement pas publier, comme des informations médicales ou des salaires.

Définition 2.4. Un **attribut non sensible** (ANS) est un attribut qui ne décrit pas des données sensibles et qui n'est pas constituant d'un identifiant explicite ni d'un quasi-identifiant. Il n'entre dans aucune des catégories précédentes.

Ainsi, on peut représenter une table originale T sous la forme suivante :

T(IE, QI, AS, ANS)

Par exemple, le **Tableau 1** représente une table originale qu'on souhaite rendre anonyme. L'attribut nom est un identifiant explicite, les attributs "âge" et "niveau" constituent le quasi-identifiant. L'attribut "maladie"' est un attribut sensible.

IE	**QI**		**AS**
Nom	**Age**	**Niveau**	**Maladie**
Umeko	19	Bac+2	maladie cardiaque
Jean	19	Bac+3	cancer
Alice	27	Bac+3	grippe
David	30	Bac+3	grippe
Bob	23	Bac	cancer
Dupont	23	Bac	cancer

Tableau 1. Un exemple de table originale

Pour le traitement des attributs quasi-identifiants ou sensibles, on distingue ceux qui sont continus de ceux qui sont catégoriels. Un attribut est continu dès lors qu'il est numérique et que des opérations arithmétiques peuvent lui être appliquées. Il est catégoriel lorsque son domaine est un ensemble borné de valeurs sur lesquelles aucune opération arithmétique ne peut être appliquée. Dans le cas où une relation d'ordre peut être appliquée sur ses valeurs possibles, on dira qu'il est ordinal. A titre d'exemple, les attributs «âge» et « niveau » du tableau 1 sont respectivement continu et catégoriel.

2 L'anonymisation de micro-données

Comme le mentionne article 29 data protection working party [3], l'anonymisation de données, visant la protection de la vie privée, a pour objectif d'empêcher 1) la singularisation d'un individu dans un ensemble de données, 2) le lien entre deux enregistrements (dont l'un correspond à des données propres à un individu) au sein d'un ensemble de données (ou entre deux ensembles de données distincts) et 3) la déduction d'informations dans ce jeu de données. Ainsi la pseudonymisation (c'est-à-dire la suppression des identifiants explicites et leur remplacement éventuel par des pseudonymes), qui est une pratique couramment utilisée par les organisations,

ne garantit pas la non ré-identification d'individus, notamment dans un contexte de publication de données. Plusieurs situations concrètes ont permis de le démontrer. Citons, à titre d'exemple, le cas de la compagnie America On Line (AOL) qui a publié, en août 2006, après suppression des noms d'utilisateurs, l'équivalent de 20 millions de lignes issues des journaux (« logs ») stockant les requêtes de recherche émises par plus d'un demi-million d'utilisateurs de Google (Harold Abelson). Par la suite, des journalistes du New York Times ont pu révéler l'identité d'une femme âgée de 62 ans, habitant Lilburn en Géorgie, en lançant différentes requêtes spécifiques sous Google. Le second exemple est celui de la société de location de film en ligne Netflix. En 2006, dans le cadre d'une compétition, le « Netflix prize », visant l'amélioration de son système de recommandation, Netflix a publié un jeu de données dé-identifié par pseudonymisation contenant plus de 100 millions d'évaluations de films effectuées par ses abonnés, entre décembre 1999 et 2005. On est arrivé à prouver que la plupart des abonnés peuvent être identifiés en ayant une connaissance limitée sur au plus 8 évaluations de films effectuées par ces abonnés et sur leurs dates d'évaluation (Narayanan et Shmatikov 2006).

Enfin, en 2002, Sweeney (Sweeney 2002b) a prouvé, grâce à l'achat d'un fichier d'électeurs et d'un fichier de données patients d'une société d'assurance médicale et après suppression des identifiants des patients, qu'il était possible, par une simple mise en correspondance, de ré-identifier des patients. Pour illustrer ce processus, l'exemple de tables Patient et Electeur cité dans (Sweeney 2002a) est repris dans la **Figure 2**. Dans cet exemple, l'anonymisation est effectuée en appliquant la pseudonymisation (c'est-à-dire en supprimant les identifiants explicites qui sont les noms et les numéros de sécurité sociale).

Données médicales anonymisées

SSN	Name	Race	DateOfBirth	Sex	ZIP	Marital Status	HealthProblem
		asian	9/27/1964	Female	94139	divorced	hypertension
		asian	9/30/1964	Female	94139	divorced	obesity
		asian	4/18/1964	Male	94139	married	chest pain
		asian	4/15/1964	Male	94139	married	obesity
		black	3/13/1963	Male	94138	married	hypertension
		black	3/18/1963	Male	94138	married	shortness of breath
		black	9/13/1964	Female	94141	married	shortness of breath
		black	9/7/1964	Female	94141	married	obesity
		white	5/14/1961	Male	94138	single	chest pain
		white	5/8/1961	Male	94138	single	obesity
		white	9/15/1961	Female	94142	widow	shortness of breath

Liste de votants

Name	Address	City	ZIP	DOB	Sex	Party	…
..							
Sue J. Carlson	900 Market St.	San Francisco	94142	9/15/1961	female	democrat	
…							

Figure 2. Exemple de ré-identification (Samarati 2001)

Ainsi, en rapprochant la table des données médicales anonymisées et l'extrait d'une liste de votants, on peut révéler l'identité de la seule personne de sexe féminin née le 15 septembre 1961 et vivant dans la zone de code

postal 94142. Il s'agit de Sue J. Carlson. On peut ainsi révéler ses informations privées telles que le fait qu'elle souffre de difficultés respiratoires.

3 Modèles d'attaque de micro-données publiées

La littérature relative à la protection de la vie privée dans un contexte de publication des données souligne, qu'en règle générale, un attaquant, pour accéder à l'information sensible, utilise des stratégies fondées avant tout sur sa connaissance du contexte. Dans ces stratégies, nommées aussi modèles d'attaque, l'adversaire déduit des informations sensibles sur sa victime en établissant des liens ou encore en procédant à des inférences probabilistes (voir **Figure 3**).

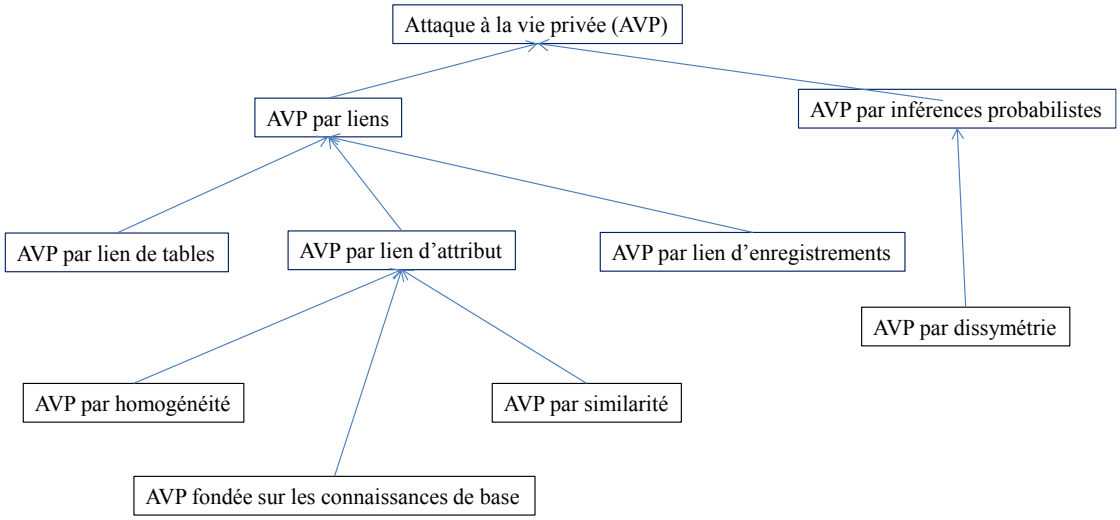


Figure 3.Taxonomie des modèles d'attaque de la vie privée

Le modèle d'attaque par liens s'applique dans le cas où l'attaquant connaît le quasi-identifiant de sa victime, c'est-à-dire la personne dont il veut obtenir les données sensibles (B. C. M. Fung et al. 2010).

Dans l'attaque par lien de tables (« table linkage »), l'adversaire ne sait pas *a priori* si les données concernant sa victime sont présentes dans la table, mais il peut le déduire à partir des informations qu'il y observe. Par exemple, soit une table publique externe E et une table T' anonymisée à partir d'une table originale T. Soit l'individu I dont on cherche à connaître les valeurs sensibles. On suppose que I appartient à un groupe contenant k individus dans T' et k' individus dans E. La probabilité que l'individu I soit présent dans T' est k/k'. Si par exemple k = 4 et k'=5, alors la probabilité de sa présence est égale à 4/5 = 0.8, ce qui est une probabilité forte.

Le second scénario, que l'on nommera *«lien d'attributs»* pour «attribute linkage», constitue une menace de ré-identification lorsque l'adversaire connaît le quasi-identifiant de la victime et qu'il peut, par simple analyse des deux tables (celle dont il dispose et celle publiée), inférer des connaissances qui le mèneront vers des ré-identifications sans pour autant savoir *a priori* si sa victime est présente ou absente de la table publiée. A titre d'exemple, si deux tables T et E anonymisées ont été publiées et que l'une des deux tables (E par exemple)

contient des données médicales telles que les maladies et qu'un attaquant connaît l'identifiant de sa victime, il pourra repérer chacun des deux groupes d'individus dans T et E dont fait partie sa victime et, par ce repérage, inférer, avec un certain degré de confiance, que sa victime souffre de l'une des maladies du groupe de E dans lequel se trouve sa victime.

Dans le scénario «lien d'attributs», le risque de ré-identification subsiste dans la mesure où l'attaquant, connaissant le QI de la victime, peut repérer le groupe d'individus partageant le même QI. Ce repérage lui donnera éventuellement la possibilité d'inférer les informations sensibles de la victime en se fondant sur les valeurs sensibles associées au groupe dans lequel se trouve la victime. Si, pour un attribut sensible, ces valeurs sont identiques (par exemple tous les individus d'un même groupe par coïncidence souffrent de la même maladie), la déduction de la donnée sensible de la victime est directe. On parle alors *d'attaque par homogénéité.*

A titre d'exemple, si l'on suppose que le quasi-identifiant d'une table «patient» est constitué du sexe, de l'âge et du code postal du patient, et, que dans cette table, toutes les femmes de 52 ans habitant le 20ème arrondissement (code postal 75020) souffrent d'un ulcère gastrique, l'adversaire pourra donc vérifier que sa victime fait partie de ce groupe et, par là-même, déduire qu'elle souffre de cette même maladie. Si maintenant, toutes les femmes de 52 ans habitant le 20ème arrondissement souffrent soit d'un ulcère gastrique ou de dyspepsie, l'adversaire pourra déduire que sa victime souffre d'une maladie de l'estomac. Ce type d'attaque, reposant sur l'analyse de la similarité sémantique des valeurs des données sensibles, est nommé *« attaque par similarité »* (similarity attack).

Un autre exemple de scénario d'attaque par «lien d'attributs» cité dans la littérature est *«l'attaque fondée sur des connaissance de base».* Dans ce type de scénario, l'attaquant dispose de connaissances suffisantes sur l'attribut sensible, lui permettant ainsi de deviner des données sensibles de sa victime une fois qu'il a repéré son appartenance à un groupe d'individus. A titre d'exemple, supposons qu'un attaquant connaît l'âge et le code postal de sa victime et que la table «patient» révèle que les patientes de 52 ans habitant Paris ont soit une maladie cardiaque soit de l'hypertension. Si l'attaquant dispose d'une connaissance selon laquelle 80% des personnes en activité sont hypertendues, sachant que sa victime est encore en activité, il pourra conclure que sa victime est très probablement hypertendue.

Dans le scénario suivant que l'on nomme *«lien d'enregistrements»*, en plus du fait que l'attaquant connaît le quasi-identifiant de la victime, il sait aussi que les informations sur la victime font partie de la table publiée. La menace est réelle lorsque, dans la table publiée, il y a très peu d'enregistrements ayant la même valeur de QI que celle de la victime.

Enfin, dans un modèle d'attaque par inférences probabilistes, l'attaquant n'établit pas de lien avec des tables, des enregistrements ou des attributs sensibles. Il s'appuie plutôt sur ses croyances probabilistes avant et après analyse de la distribution des valeurs des attributs sensibles de la table publiée. Le scénario d'attaque, fréquemment mentionné dans la littérature pour ce type de modèle, est l'attaque par dissymétrie (« skewness attack »). Dans ce scénario, l'adversaire déduit la valeur d'une donnée sensible de sa victime par comparaison de la distribution

globale des valeurs de l'attribut sensible (croyance probabiliste avant analyse des données publiées) avec la distribution des valeurs de ce même attribut sensible au sein d'un groupe d'individus de même QI (croyance probabiliste après analyse des données publiées). A titre d'exemple, si la proportion d'individus atteints d'un cancer de l'estomac dans un groupe donné est nettement plus élevée que dans l'ensemble de la population alors on peut déduire que les personnes faisant partie de ce groupe ont une forte chance d'être atteintes d'un cancer de l'estomac.

Pour contrer ces scénarios d'attaques potentielles, des modèles de protection de la vie privée ont été proposés dans la littérature. Ces modèles ont été mis en œuvre via des techniques d'anonymisation prenant en compte l'usage ultérieur des données. Ces techniques ont été, à leur tour, instanciées par des algorithmes plus ou moins performants. Les sections qui suivent présentent brièvement quelques-uns de ces modèles et les techniques afférentes.

4 Modèles de protection de la vie privée

Les efforts de recherche consacrés à la protection de la vie privée ont donné naissance à plusieurs modèles et variantes de modèles. A titre d'exemple, (B. C. M. Fung et al. 2010) recense non moins de quinze modèles. Dans cette section, nous ne citons que quelques exemples permettant de contrer les types d'attaques cités ci-avant, et parmi eux les modèles les plus référencés dans la littérature, à savoir le k-anonymat, la l-diversité et la t-proximité ainsi que la δ-présence.

4.1 Le modèle de k-anonymat

Le modèle de k-anomymat est le premier modèle proposé dans la littérature (Samarati et Sweeney 1998). Lorsqu'il est mis en œuvre, ce modèle offre l'assurance que chaque n-uplet de valeurs de quasi-identifiants apparaît au moins k fois dans la table à publier. Ainsi, une table qui répond à ce type de modèle est dite «k-anonyme». A titre d'exemple, si l'on considère que les attributs «Age» et «Education» constituent le QI du **Tableau 2** Table qui satisfait le 2-anonymat, alors cette table satisfait le 2-anonymat contrairement au **Tableau 3**. En effet, chacune des différentes valeurs du QI dans le **Tableau 2** («[19,23], Supérieur», «[19,23], Secondaire» et «[27,30], Secondaire»), appelées aussi «classes d'équivalence», apparaissent au moins deux fois dans cette table. En revanche, le **Tableau 3** ne satisfait pas le 2-anonymat puisque la classe d'équivalence «[27,30], Supérieur» contient un seul enregistrement (le dernier de cette table).

Age	Education	Maladie
[19,23]	Secondaire	Maladie cardiaque
[19,23]	Secondaire	Cancer
[27,30]	Secondaire	Grippe
[27,30]	Secondaire	Grippe
[19,23]	Supérieur	Cancer
[19,23]	Supérieur	Cancer
[19,23]	Supérieur	Cancer

Tableau 2 Table qui satisfait le 2-anonymat

Age	Education	Maladie
[19,23]	Secondaire	Maladie cardiaque
[19,23]	Secondaire	Cancer
[27,30]	Secondaire	Grippe
[27,30]	Secondaire	Grippe
[19,23]	Supérieur	Cancer
[19,23]	Supérieur	Cancer
[27,30]	Supérieur	Cancer

Tableau 3. Table qui ne satisfait pas le 2-anonymat

Ainsi, le degré de protection fourni par ce modèle est fonction du k choisi qui, de toute évidence, est strictement supérieur à 1.

Une table peut satisfaire le k-anonymat d'une façon optimale si elle est k-anonyme tout en préservant au maximum la qualité des données. Mais, (Haritsa, 2005) a montré qu'obtenir le k-anonymat optimal est très coûteux.

Cependant, le k-anonymat, se focalisant uniquement sur les attributs du QI, permet de contrer uniquement les attaques par «liaison d'enregistrements». Il n'est aucunement résistant à d'autres types d'attaques telles que celles par «liaison d'attributs» et, plus particulièrement, les «attaques par homogénéité» et «les attaques fondées sur des connaissances de base». A titre d'exemple, supposons qu'un attaquant nommé «Alice» connaisse une victime nommée «Bob». Alice, connaissant l'âge (22 ans) et le niveau d'éducation de Bob («Supérieur»), peut directement conclure, après analyse du **Tableau 2**, si bien sûr celui-ci est publié tel quel, que Bob a un cancer. En effet, les deux données de Bob (âge et niveau d'éducation) connues et rapprochées, par Alice, du QI du **Tableau 2** pourtant 2-anonyme révèlent l'appartenance de Bob à la classe d'équivalence «[19,23], Supérieur». Dans la mesure où toutes les valeurs de l'attribut sensible «maladie» sont identiques, Alice déduit que Bob est atteint de la même maladie que tous les individus de la classe dans laquelle il se trouve.

Supposons maintenant qu'Alice ait une voisine japonaise qui s'appelle Junko. Alice sait que Junko a 20 ans et a un niveau d'études secondaire. Disposant du **Tableau 2**, Alice pourra déduire que Junko fait partie de la classe d'équivalence «[19,23], Supérieur». Si l'on suppose que la probabilité qu'un Japonais soit atteint de maladie cardiaque est très faible et qu'Alice dispose de cette connaissance, alors elle pourra conclure que sa voisine est probablement atteinte d'un cancer.

En résumé, le k-anonymat, appliqué à une table, peut mener à la création de classes d'équivalences qui laissent échapper des informations sensibles en raison du manque de diversité dans les valeurs de l'attribut sensible. Pour pallier cet inconvénient, d'autres modèles ont vu le jour tels que le modèle de l-diversité proposé initialement par (Machanavajjhala et al. 2007).

4.2 Le modèle de l-diversité

Comme énoncé ci-dessus, le modèle de l-diversité permet de contrer des attaques par liaison d'attributs. Il vient renforcer le k-anonymat en évitant, dans le cas où le QI d'une victime est connu, de cibler un enregistrement d'une table publiée et donc, de ce fait, de révéler de façon directe des données sensibles de la victime. Le principe de la l-diversité défini dans (Machanavajjhala et al. 2007) est le suivant :

Définitions : Une classe d'équivalence respecte la l-diversité s'il existe au moins l valeurs «bien représentées » pour l'attribut sensible. Une table respecte la l-diversité si chacune de ses classes d'équivalence respectent la l-diversité.

Notons que, derrière cette définition de la l-diversité, se cachent plusieurs dimensions selon l'interprétation donnée à l'expression « bien représentées » et au fait que l'on puisse disposer, dans une table, d'un ou plusieurs attributs sensibles. (Machanavajjhala et al. 2007) distinguent ainsi différentes dimensions ou modèles associés à la l-diversité. Le modèle le plus simple est le modèle « distinct l-diversity » que l'on nommera « l-diversité distincte ». Ce modèle n'accorde pas d'importance au terme « bien ». Il se concentre sur le reste de la définition c'est-à-dire l'obtention de classes d'équivalences l-diverses. Ainsi, dans ce modèle, on fait en sorte que, pour un attribut sensible, il existe au moins l valeurs représentées de cet attribut sensible au sein de tout groupe d'individu partageant le même QI. A titre d'exemple, le **Tableau 4** possède la « 3-diversité distincte » (et le 4-anonymat) car chaque classe d'équivalence contient au moins trois valeurs distinctes de l'attribut 'maladie'. Cependant, le **Tableau 5** ne possède pas la « 3-diversité distincte » car la deuxième classe d'équivalence contient seulement deux valeurs distinctes de l'attribut sensible 'maladie'. Cependant, il satisfait la « 2-diversité distincte ».

Age	Education	Maladie
[19,23]	Secondaire	Maladie cardiaque
[19,23]	Secondaire	Cancer
[19,23]	Secondaire	Grippe
[19,23]	Secondaire	Grippe
[27,30]	Supérieur	Cancer
[27,30]	Supérieur	Cancer
[27,30]	Supérieur	Maladie cardiaque
[27,30]	Supérieur	Grippe

Tableau 4. Table qui satisfait la 3-diversité

Age	Education	Maladie
[19,23]	Secondaire	Maladie cardiaque
[19,23]	Secondaire	Cancer
[19,23]	Secondaire	Grippe
[19,23]	Secondaire	Grippe
[27,30]	Supérieur	Cancer
[27,30]	Supérieur	Cancer

[27,30]	Supérieur	Maladie cardiaque
[27,30]	Supérieur	Maladie cardiaque

Tableau 5. Table qui satisfait la 2-diversité

D'autres modèles plus robustes, car offrant de meilleures protections de la vie privée, sont proposés dans la littérature. Parmi ces modèles, on peut citer le modèle « entropy l-diversity que l'on nommera «l-diversité fondée sur l'entropie ». Ce modèle capte la sémantique du terme « bien » en uniformisant au mieux la distribution des valeurs sensibles de telle sorte à maximiser l'ignorance sur la distribution des valeurs sensibles dans chacune des classes d'équivalence, c'est-à-dire de telle sorte que l'entropie de la distribution des valeurs sensibles dans chacune des classes d'équivalence soit supérieure ou égale à log(l).

A titre d'exemple, le calcul de l'entropie de chacune des deux classes d'équivalence du **Tableau 4** se fait au moyen de la formule suivante :

$$\text{Entropie } (C) = -\sum_{s \in S} P(qid,s)\log(P(qid,s))$$

Où C est la classe d'équivalence, S est l'attribut sensible et P(qid,s) représente la fraction des enregistrements ayant la valeur sensible 's' dans cette classe d'équivalence C. Ainsi, cette formule nous permet de conclure que le **Tableau 5** ne satisfait pas la « 3-diversité fondée sur l'entropie » mais plutôt la «2-diversité fondée sur l'entropie » car les entropies des deux classes d'équivalence sont inférieures à log(3) (égales respectivement à log(2,85) et log(2)).

Dans l'ensemble, les différents modèles conçus sur le principe de la l-diversité contrecarrent les attaques par homogénéité et celles fondées sur les connaissances de base que le k-anonymat ne peut pas écarter. Plus la diversité des valeurs des attributs sensibles est importante, plus la menace d'atteinte à la vie privée via ce type d'attaques est amoindrie. Cependant, pour des adversaires beaucoup plus expérimentés, la l-diversité n'assure pas la protection contre les attaques par similarité et les attaques par inférences probabilistes (« probabilistic inference attacks ») dont celles par dissymétrie. Si par exemple, dans une classe d'équivalence du **Tableau 5** qui est 2-diverse, on a dix tuples et que, pour l'attribut maladie, on a un tuple avec la valeur sensible 'cancer', un autre avec une maladie cardiaque et huit autres contenant la grippe, cela n'empêchera pas l'attaquant de conclure que sa victime est atteinte de la grippe avec un degré de confiance de 80%. De même, si dans une autre classe d'équivalence de cette table, tous les tuples correspondent à des patients ayant des maladies en lien avec le cœur, cela n'empêchera pas non plus un attaquant de déduire, dans le cas où sa victime a le même QI que les individus de cette classe, que celle-ci est atteinte d'une maladie cardiaque.

Pour pallier l'incapacité de la l-diversité à contrer les attaques par dissymétrie, le principe de la « t-closeness », que nous nommerons t-proximité a été proposé dans la littérature. Nous le décrivons succinctement dans le paragraphe qui suit.

4.3 Le modèle de t-proximité

Bien qu'une table puisse être protégée par le principe de la l-diversité, il est possible pour un adversaire d'obtenir des informations au sujet d'un attribut sensible dès lors qu'il dispose d'informations sur la distribution globale de cet attribut. Pour contrer cela, la t-proximité (« t-closeness ») a été proposée par (N. Li, Li, et Venkatasubramanian 2007). Sachant qu'il est impossible d'empêcher un adversaire d'avoir des informations sensibles globales sur une population, le principe de la t-proximité a pour objectif de limiter la capacité de cet adversaire à déduire des informations sensibles sur des individus ciblés. Pour ce faire, il fait en sorte que la distribution de l'attribut sensible au sein de n'importe quelle classe d'équivalence soit proche de la distribution globale de l'attribut. En d'autres termes, il introduit le concept de distance entre ces deux distributions et propose que cette distance ne dépasse pas le seuil t. Ainsi, plus t est petit, plus la possibilité d'inférence de l'adversaire est réduite. D'où la définition suivante de ce principe proposée dans (N. Li, Li, et Venkatasubramanian 2007):

Définition 2: Une classe d'équivalence satisfait la t-proximité si la distance entre la distribution d'un attribut sensible dans cette classe et la distribution de l'attribut dans la table entière ne dépasse pas un seuil t. Une table satisfait la t-proximité si toutes ses classes d'équivalence satisfont la t-proximité.

Pour la mise en œuvre de ce principe, la distance EMD (Earth Mover's Distance) est celle privilégiée dans la littérature. Son calcul diffère selon que l'attribut sensible est catégoriel ou continu. Il est détaillé dans (Rubner, Tomasi, et Guibas 2000).

4.4 Le modèle de δ-Présence

Le modèle de δ-Présence a été mis en œuvre dans l'objectif de contrer les attaques par «lien de tables». En effet, la publication de plusieurs tables anonymes par des éditeurs différents étant possible, on ne peut exclure la possibilité de rapprochement entre elles dès lors qu'elles partagent des valeurs de QI. Certains rapprochements peuvent mener à la divulgation de données sensibles. A titre d'exemple, supposons que le **Tableau 2** sur les maladies a été publié au même titre que le **Tableau 6** sur les catégories professionnelles.

Age	Education	Nom
[19,23]	Supérieur	Malik
[19,23]	Supérieur	George
[27,30]	Supérieur	Fred
[27,30]	Supérieur	Jean
[19,23]	Supérieur	Pierre
[27,30]	Supérieur	Paul
[19,23]	Supérieur	Alice

Tableau 6. Table qui satisfait le 3-anonymat

Le **Tableau 6** révèle la tranche d'âge et le niveau d'éducation des individus. A titre d'exemple, Alice, la victime de l'attaquant, a un âge compris entre 19 et 23 ans. Elle a un niveau supérieur d'études.

En rapprochant ces deux tables, l'attaquant peut déduire qu'Alice a une probabilité de 3/4= 75% de se trouver dans le **Tableau 2** (le chiffre 3 correspond à la taille de la classe d'équivalence du QI « [19, 23], Supérieur »

dans le **Tableau 2** et le chiffre 4 correspond à celle du même QI dans le **Tableau 6**). Sachant que les individus de QI « [19, 23], Supérieur » sont tous atteints d'un cancer et qu'Alice a ce même QI, on peut déduire que la probabilité qu'Alice soit atteinte d'un cancer est de 75%.

Pour empêcher ce type d'attaque et donc éviter d'inférer la présence de tout enregistrement d'une table publiée dans une autre table publiée, (Nergiz, Atzori, et Clifton 2007) a proposé le modèle de δ-présence qui impose que la probabilité de présence d'un enregistrement soit dans un intervalle $\delta = (\delta_{min}, \delta_{max})$ prédéfini.

Ce modèle bien qu'intéressant est difficile à mettre en œuvre car il suppose que l'éditeur connaisse *a priori* la table de rapprochement que l'attaquant est susceptible d'utiliser.

5. Les techniques d'anonymisation de micro-données

Les données contenues sous forme agrégée dans des tables ont pendant longtemps constitué les sorties traditionnelles des organismes nationaux de statistique. Ainsi, la recherche sur la protection de ce type de données est la plus ancienne et la plus établie (Dalenius 1977). Elle a été essentiellement menée par la communauté des statisticiens travaillant sur le contrôle de divulgation statistique, connu sous le terme de SDC (Statistical Disclosure Control) et/ou sous celui de SDL (Statistical Disclosure Limitation).

En revanche, la protection des micro-données est plus récente. La recherche liée à la définition et la mise en œuvre des techniques d'anonymisation pour ce type de données bénéficie ainsi non seulement de l'apport de la communauté de statisticiens mais aussi de la communauté des informaticiens s'intéressant à la préservation de la vie privée à des fins de fouille de données connue sous le nom de PPDM (Privacy Preserving Data Mining) ou encore à la préservation de la vie privée à des fins de publication connue sous le nom de PPDP (Privacy Preserving Data Publishing). Le PPDM est un domaine de recherche visant à étendre les techniques traditionnelles de fouille de données de telle sorte à pouvoir manipuler des données où l'information sensible a été masquée (Aïmeur 2009) (Vaghashia et Ganatra 2015) (Verykios et al. 2004). Le PPDP étudie comment immuniser les données contre les attaques de la vie privée (Amita S., 2014) (Kiran et Kavya 2012). Une description des travaux de ces trois disciplines (SDC/SDL, PPDM, PPDP) est fournie dans plusieurs revues de la littérature dont (Hussien A.E.E.A., 2013) (B.M.Y, 2015).

Comme notre recherche ne requiert pas la présentation exhaustive de toutes les techniques recensées dans la littérature, nous nous concentrons, dans cette section, sur celles fréquemment citées. Ces dernières sont appliquées sur des attributs quasi-identifiants ou sensibles.

Nous allons donc décrire les principes généraux de quelques techniques en les illustrant au travers d'un exemple. Pour ce faire, nous exploitons le **Tableau 7**. Il représente un extrait de données originales médicales. La table est constituée de huit attributs : sexe, âge, profession, statut marital, âge, nombre de jours d'hospitalisation (JH), le taux de cholestérol et la température. Les quatre premiers attributs sont catégoriels et les autres sont continus.

Num tuple	Sexe	Ville	Profession	Statut marital	Age	JH	Cholestérol	Température
1	F	Paris	ingénieur	mariée	33	3	150	36,6
2	F	Pontoise	ingénieur	veuve	65	1	290	37,2
3	F	Paris	étudiant	célibataire	21	2	190	36,2
4	M	Nice	étudiant	célibataire	19	7	170	38,5
5	M	Paris	statisticien	marié	36	40	200	40,1
6	F	Cannes	étudiant	mariée	28	3	185	37,5
7	F	Pontoise	statisticien	divorcée	46	60	200	36,5
8	M	Pontoise	statisticien	divorcé	81	5	300	37,6
9	M	Cannes	statisticien	marié	58	10	260	36,9
10	F	Nice	professeur	veuve	63	7	290	38,2
11	M	Paris	étudiant	célibataire	19	5	190	39,3

Tableau 7. Table originale avant anonymisation (exemple extrait de (V. Ciriani, 2007))

5.1 La généralisation (Samarati 2001)

L'objectif de la généralisation est de renforcer le k-anonymat sur les micro-données publiées. Ainsi, elle permet de confondre chaque individu avec k-1 autres individus de la table publiée. En d'autres termes, elle procède à la transformation des quasi-identifiants de telle sorte à ce qu'au moins k individus aient la même valeur de QI. Intuitivement, elle procède au partitionnement des tuples selon leur valeur du QI, puis elle transforme les valeurs des QI d'un même groupe en une valeur moins spécifique de telle sorte que les tuples d'un même groupe ne puissent pas se distinguer des autres tuples du groupe à travers la valeur de leur QI. Pour réaliser la transformation de valeurs du QI, elle s'appuie sur des hiérarchies de généralisation prédéfinies.

La généralisation ne dicte pas d'exigence sur la nature des attributs du QI. Ces derniers peuvent être continus ou catégoriels. Chacun fait l'objet d'une hiérarchie contenant au moins deux niveaux. Les feuilles correspondent aux valeurs originales et constituent le niveau le plus bas, noté 0. Un lien entre une valeur de niveau n-1 et une valeur de niveau n décrit la possibilité de remplacement ou de généralisation de la valeur de niveau n-1 par la valeur de niveau n. Ainsi, la racine de la hiérarchie est la valeur la plus générale. Elle représente le plus haut niveau. A titre d'exemple, la **Figure 4** et la **Figure 5** représentent respectivement une hiérarchie de généralisation de l'attribut « ville » et une hiérarchie de généralisation de l'attribut « âge ». Le nœud "Ile de France" est au niveau 1 de la hiérarchie.

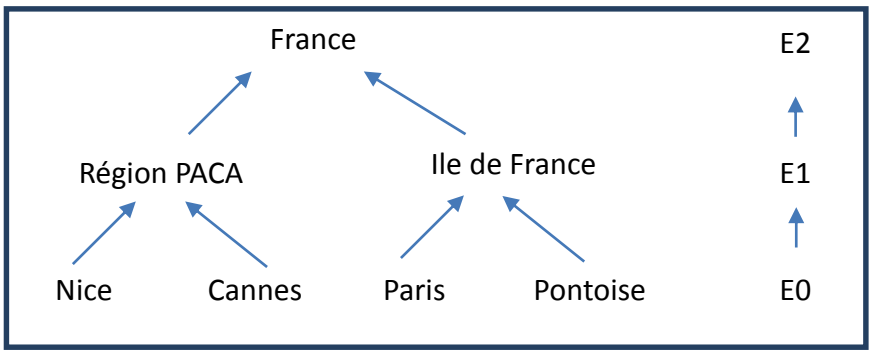


Figure 4. Hiérarchie de généralisation de l'attribut Ville

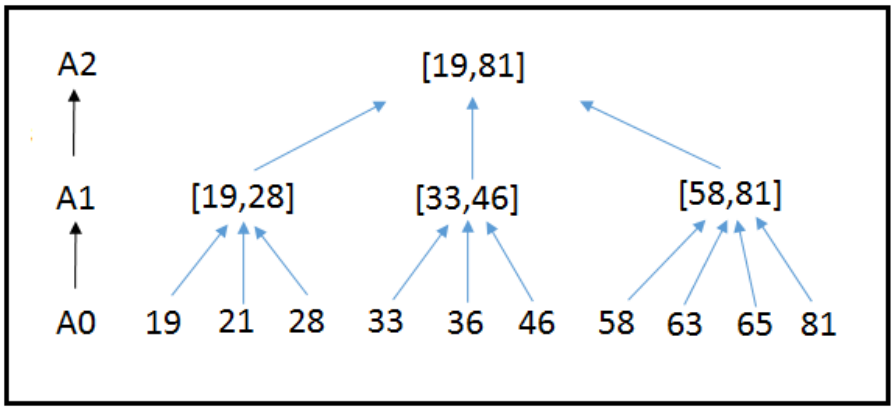


Figure 5. Hiérarchie de généralisation de l'attribut âge

Une fois la hiérarchie définie, la technique de généralisation consiste à remplacer dans la table anonyme, une valeur d'un attribut du QI de la table originale par un de ses ancêtres dans la hiérarchie de généralisation. Le niveau de généralisation appliqué peut être différent pour chacun des attributs du QI. A titre d'exemple, si l'on considère que l'âge et la ville sont des attributs du QI, on pourrait, par le procédé de généralisation, obtenir le **Tableau 8**.

Num tuple	Sexe	Ville	Profession	Statut marital	Age	JH	Choles-térol	Tempé-rature
1	F	Ile de France	ingénieur	mariée	[19,81]	3	150	36,6
2	F	Ile de France	ingénieur	veuve	[19,81]	1	290	37,2
3	F	Ile de France	étudiant	célibataire	[19,81]	2	190	36,2
4	M	Région PACA	étudiant	célibataire	[19,81]	7	170	38,5
5	M	Ile de France	statisticien	marié	[19,81]	40	200	40,1
6	F	Région PACA	étudiant	mariée	[19,81]	3	185	37,5
7	F	Ile de France	statisticien	divorcée	[19,81]	60	200	36,5
8	M	Ile de France	statisticien	divorcé	[19,81]	5	300	37,6
9	M	Région PACA	statisticien	marié	[19,81]	10	260	36,9
10	F	Région PACA	professeur	veuve	[19,81]	7	290	38,2
11	M	Ile de France	étudiant	célibataire	[19,81]	5	190	39,3

Tableau 8. Application de la technique de généralisation aux attributs ville et âge

5.2 La suppression (Lawrence H. 1980)

Cette technique génère une table anonyme où toutes les micro-données de la table originale sources d'un risque de ré-identification sont retirées. Une suppression peut concerner aussi bien la suppression de tuples dans leur totalité que la suppression de quelques données de tuples (remplacement par la valeur nulle). Dans le premier cas, on parle de suppression globale. Le second cas est nommé suppression locale. Il s'agit, dans ce cas de figure, de remplacer la donnée originale dans la table anonyme par une marque mentionnant son retrait.

Notons que la suppression globale est très souvent effectuée en complément de la généralisation, dans l'objectif de satisfaire le k-anonymat sans généraliser à outrance les données du QI.

A titre d'exemple, supposons que, dans le **Tableau 7**, les deux derniers enregistrements présentent un risque de ré-identification. Pour éviter cette dernière, la solution d'anonymisation proposée est la suppression des valeurs des attributs « sexe » et « profession » respectivement pour ces deux enregistrements (voir **Tableau 9**). Le choix des valeurs à supprimer se fonde sur un calcul qui vise à diminuer le nombre de suppressions locales. Notons qu'une combinaison de ces deux modes de suppression est possible.

Num tuple	Sexe	Ville	Profession	Statut marital	Age	JH	Choles-térol	Tempé-rature
1	F	Paris	ingénieur	mariée	33	3	150	36,6
2	F	Pontoise	ingénieur	veuve	65	1	290	37,2
3	F	Paris	étudiant	célibataire	21	2	190	36,2
4	M	Nice	étudiant	célibataire	19	7	170	38,5
5	M	Paris	statisticien	marié	36	40	200	40,1
6	F	Cannes	étudiant	mariée	28	3	185	37,5
7	F	Pontoise	statisticien	divorcée	46	60	200	36,5
8	M	Pontoise	statisticien	divorcé	81	5	300	37,6
9	M	Cannes	statisticien	marié	58	10	260	36,9
10	****	Nice	professeur	veuve	63	7	290	38,2
11	M	Paris	****	célibataire	19	5	190	39,3

Tableau 9. Application de la suppression locale au **Tableau 7**.

5.3 La micro-agrégation (Defays et Nanopoulos 1992)

La micro-agrégation fait partie de la famille de techniques de SDC. Elle est appliquée à des micro-données continues. Elle contribue au renforcement du k-anonymat en faisant en sorte que les enregistrements correspondent à des groupes d'au moins k individus appelés micro-agrégats. Pour satisfaire la confidentialité des données, les valeurs originales sont remplacées par une mesure centrale (généralement la moyenne ou la médiane) du micro-agrégat auquel elles appartiennent. Ainsi, la micro-agrégation est réalisée en deux grandes étapes : le partitionnement et l'agrégation. Au cours du partitionnement, les groupes d'enregistrements sont construits de telle sorte que les enregistrements soient voisins dans le même groupe et que leur nombre dans chaque groupe soit au moins égal à k. Cette étape doit mettre en place des groupes aussi homogènes que possible. Dans la seconde étape, un opérateur d'agrégation est calculé pour chaque groupe. Ensuite, chaque enregistrement est remplacé par la valeur agrégée calculée pour le groupe auquel il appartient.

A titre d'exemple, nous allons appliquer la micro-agrégation à l'attribut 'cholestérol' dans le **Tableau 7**. La première étape consiste à diviser les enregistrements en groupes homogènes selon l'attribut âge afin de satisfaire le 3-anonymat. Par conséquent, nous trions les enregistrements selon l'attribut âge et nous constituons des groupes dont chacun doit contenir au minimum trois enregistrements comme le montre le tableau souvant.

Num tuple	Sexe	Ville	Profession	Statut marital	Age	JH	Choles-térol	Tempé-rature

11	M	Paris	étudiant	célibataire	19	5	190	39,3
4	M	Nice	étudiant	célibataire	19	7	170	38,5
3	F	Paris	étudiant	célibataire	21	2	190	36,2
6	F	Cannes	étudiant	mariée	28	3	185	37,5
1	F	Paris	ingénieur	mariée	33	3	150	36,6
5	M	Paris	statisticien	marié	36	40	200	40,1
7	F	Pontoise	statisticien	divorcée	46	60	200	36,5
9	M	Cannes	statisticien	marié	58	10	260	36,9
2	F	Pontoise	ingénieur	veuve	65	1	290	37,2
10	F	Nice	professeur	veuve	63	7	290	38,2
8	M	Pontoise	statisticien	divorcé	81	5	300	37,6

Tableau 10. Etape de partition de la technique de micro-agrégation

Num tuple	Sexe	Ville	Profession	Statut marital	Age	JH	Choles-térol	Tempé-rature
11	M	Paris	étudiant	célibataire	19	5	190	39,3
4	M	Nice	étudiant	célibataire	19	7	170	38,5
3	F	Paris	étudiant	célibataire	21	2	190	36,2
6	F	Cannes	étudiant	mariée	28	3	185	37,5
1	F	Paris	ingénieur	mariée	33	3	150	36,6
5	M	Paris	statisticien	marié	36	40	200	40,1
7	F	Pontoise	statisticien	divorcée	46	60	200	36,5
9	M	Cannes	statisticien	marié	58	10	260	36,9
2	F	Pontoise	ingénieur	veuve	65	1	290	37,2
10	F	Nice	professeur	veuve	63	7	290	38,2
8	M	Pontoise	statisticien	divorcé	81	5	300	37,6

Tableau 10. Etape de partition de la technique de micro-agrégation

Ensuite, nous allons remplacer la valeur de l'attribut 'température' de chaque enregistrement par la moyenne du groupe comme le montre le **Tableau 11.** Etape d'agrégation de la technique de micro-agrégation

Num tuple	Sexe	Ville	Profession	Statut marital	Age	JH	Choles-térol	Tempé-rature
11	M	Paris	étudiant	célibataire	19	5	190	38
4	M	Nice	étudiant	célibataire	19	7	170	38
3	F	Paris	étudiant	célibataire	21	2	190	38
6	F	Cannes	étudiant	mariée	28	3	185	38,06
1	F	Paris	ingénieur	mariée	33	3	150	38,06
5	M	Paris	statisticien	marié	36	40	200	38,06
7	F	Pontoise	statisticien	divorcée	46	60	200	37,2
9	M	Cannes	statisticien	marié	58	10	260	37,2
2	F	Pontoise	ingénieur	veuve	65	1	290	37,2
10	F	Nice	professeur	veuve	63	7	290	37,2
8	M	Pontoise	statisticien	divorcé	81	5	300	37,2

Tableau 11. Etape d'agrégation de la technique de micro-agrégation

5.4 La technique de « bucketization » (Martin et al. 2007)

Cette technique a été proposée pour intégrer la l-diversité dans les tables anonymes. Son principe général consiste à partitionner les tuples de la table à anonymiser en segments (« buckets » en anglais) puis à procéder à la séparation des attributs sensibles de ceux qui ne le sont pas, en permutant aléatoirement les valeurs des attributs sensibles dans chaque segment.

A titre d'exemple, le **Tableau 12** est le résultat de l'application de la technique de « bucketization » sur la table originale. Les enregistrements de la table originale ont été divisés en deux segments. Ensuite, dans chaque groupe, les valeurs de l'attribut maladie des deuxième et troisième enregistrements ont été permutées. Il en est de même pour les deux derniers enregistrements.

Quasi identifiant		**Attribut sensible**
Age	**Niveau**	**Maladie**
19	Bac+2	maladie cardiaque
19	Bac+3	grippe
27	Bac+3	cancer
30	Bac+3	cancer
23	Bac	grippe
23	Bac	cancer

Tableau 12. La table « bucketisée »

Notons que cette technique, contrairement à la généralisation, maintient les valeurs originales des attributs du QI dans la table anonyme. Cependant, elle supprime les corrélations entre les attributs du QI et les attributs sensibles.

5.5 La technique « Anatomy » (Xiao et Tao 2006)

Comme la technique de « bucketization », Anatomy permet de créer des tables l-diverses. Cette technique a été proposée dans l'objectif de contrer les désavantages de la généralisation. Anatomy casse le lien entre le QI et les attributs sensibles en créant deux tables séparées à partir d'une table originale. La première contient les valeurs du QI et des attributs non-sensibles (voir **Tableau 13**). La seconde contient les données sensibles. Pour établir le lien entre les deux tables, elle fait partager aux deux tables un attribut commun qui mentionne l'appartenance d'un tuple à un groupe (voir **Tableau 14**).

De façon informelle, Anatomy partitionne la table originale en groupes en adoptant une certaine stratégie de partitionnement. Ensuite, elle affecte à chaque partition un identifiant puis crée les deux tables et ajoute à chaque tuple des deux tables l'identifiant de partition adéquat.

Age	**Niveau**	**groupe**
19	Bac+2	1
19	Bac+3	1
27	Bac+3	1
30	Bac+3	2
23	Bac	2
23	Bac	2

Tableau 13. Table des attributs QI

groupe	Maladie	fréquence
1	maladie cardiaque	1
1	cancer	1
1	grippe	1
2	grippe	1
2	cancer	2

Tableau 14. Table des attributs sensibles

5.6 La technique de "Slicing" (T. Li et al. 2012)

Cette technique, dont l'objectif est de garantir la l-diversité dans des tables anonymes, adopte un partitionnement à la fois vertical et horizontal. Le partitionnement vertical concerne les attributs et est fondé sur la corrélation entre ceux-ci. Chaque partition obtenue, que l'on nommera « partition verticale », contient les attributs fortement corrélés entre eux. Le second partitionnement concerne les tuples de la table. A l'intérieur de chaque partition obtenue, les valeurs d'un même attribut sont aléatoirement permutées (ou encore triées) afin de casser le lien entre les différents attributs d'une même partition verticale.

A titre d'exemple, supposons que les attributs « niveau d'éducation » et « maladie » sont fortement corrélés. Nous obtenons alors deux partitions verticales : la première contient l'attribut 'âge' et la seconde les attributs 'niveau d'études' et 'maladie'. De plus, si l'on divise les enregistrements de la table originale en deux groupes (partitionnement horizontal) et que l'on procède à des permutations, on obtient la table suivante :

Age	(Niveau d'études, Maladie)
19	(Bac+2, maladie cardiaque)
19	(Bac+3, grippe)
27	(Bac+3, cancer)
23	(Bac, cancer)
23	(Bac, grippe)
30	(Bac + 3, cancer)

Tableau 15. Table issue du Slicing

5.7 La permutation ou technique de "Swapping" (Dalenius et Reiss 1982)

La technique de « Swapping » est l'une des plus anciennes techniques de SDC. Elle peut être appliquée aussi bien sur un attribut du QI que sur un attribut sensible, qu'il soit continu ou catégoriel. Comme son nom l'indique, elle procède à des permutations de valeurs d'un même attribut au sein d'un sous-ensemble de tuples. A titre d'exemple, la permutation appliquée sur l'attribut Profession au sein du sous-ensemble constitué des tuples 3 et 5 donnerait la table anonyme suivante (**Tableau 16**).

Num tuple	Sexe	Ville	Profession	Statut marital	Age	JH	Choles-térol	Tempé-rature
1	F	Paris	ingénieur	mariée	33	3	150	36,6
2	F	Pontoise	ingénieur	veuve	65	1	290	37,2
3	F	Paris	statisticien	célibataire	21	2	190	36,2
4	M	Nice	étudiant	célibataire	19	7	170	38,5
5	M	Paris	étudiant	marié	36	40	200	40,1
6	F	Cannes	étudiant	mariée	28	3	185	37,5
7	F	Pontoise	statisticien	divorcée	46	60	200	36,5
8	M	Pontoise	statisticien	divorcé	81	5	300	37,6
9	M	Cannes	statisticien	marié	58	10	260	36,9
10	F	Nice	professeur	veuve	63	7	290	38,2
11	M	Paris	étudiant	célibataire	19	5	190	39,3

Tableau 16. Application du « data swapping » à l'attribut Profession

5.8 Le recodage global (Domingo-Ferrer et Torra 2001), (Domingo-Ferrer et Torra 2002)

Dans ce type de technique, le domaine de valeurs d'un attribut est partitionné en plusieurs intervalles auxquels on associe des labels. Une préférence est donnée à des intervalles de même largeur. Chaque valeur d'un attribut de la table originale est remplacée, dans la table anonyme, par le label de l'intervalle auquel appartient cet attribut.

A titre d'exemple, si l'on applique cette technique au domaine de l'attribut cholestérol, en créant trois partitions : [150,200[, [200,250[, [250,300], on obtient la table anonyme du **Tableau 17**.

Num tuple	Sexe	Ville	Profession	Statut marital	Age	JH	Choles-térol	Tempé-rature
1	F	Paris	ingénieur	mariée	33	3	[150,200[	36,6
2	F	Pontoise	ingénieur	veuve	65	1	[250,300]	37,2
3	F	Paris	étudiant	célibataire	21	2	[150,200[	36,2
4	M	Nice	étudiant	célibataire	19	7	[150,200[	38,5
5	M	Paris	statisticien	marié	36	40	[200,250[	40,1
6	F	Cannes	étudiant	mariée	28	3	[150,200[	37,5
7	F	Pontoise	statisticien	divorcée	46	60	[200,250[	36,5
8	M	Pontoise	statisticien	divorcé	81	5	[250,300]	37,6
9	M	Cannes	statisticien	marié	58	10	[250,300]	36,9
10	F	Nice	professeur	veuve	63	7	[250,300]	38,2
11	M	Paris	étudiant	célibataire	19	5	[150,200[	39,3

Tableau 17. Recodage global de l'attribut cholestérol

5.9 Les techniques de « Top Coding » et de « Bottom Coding » (Domingo-Ferrer et Torra 2001) (Domingo-Ferrer et Torra 2002)

En plus du recodage global, la littérature propose deux autres techniques de recodage : « Top coding » et « Bottom Coding ». La technique de « Top coding » consiste à reproduire dans la table anonyme toutes les données originales d'un attribut, hormis celles qui dépassent une valeur « limite supérieure » prédéfinie (nommée en anglais « top coding »). Dans ce cas, c'est la valeur « limite supérieure » qui est utilisée pour masquer la valeur originale. Inversement, la technique de « Bottom Coding » reprend dans la table anonyme toutes les données originales d'un attribut, hormis celles qui sont inférieures à une valeur « limite inférieure » prédéfinie (nommée en anglais « bottom code »). Dans ce cas, c'est la valeur « limite inférieure» qui est appliquée.

A titre d'exemple, si l'on définit la limite supérieure de l'attribut « température » comme étant égale à 38, nous obtiendrons, par application de « top coding », le **Tableau 18**.

Il est, bien sûr, évident que les deux techniques sont applicables à tout attribut continu ainsi qu'aux attributs catégoriels ordinaux.

Num tuple	Sexe	Ville	Profession	Statut marital	Age	JH	Choles-térol	Tempé-rature
1	F	Paris	ingénieur	mariée	33	3	150	36,6
2	F	Pontoise	ingénieur	veuve	65	1	290	37,2
3	F	Paris	étudiant	célibataire	21	2	190	36,2
4	M	Nice	étudiant	célibataire	19	7	170	>38
5	M	Paris	statisticien	marié	36	40	200	>38
6	F	Cannes	étudiant	mariée	28	3	185	37,5
7	F	Pontoise	statisticien	divorcée	46	60	200	36,5
8	M	Pontoise	statisticien	divorcé	81	5	300	37,6
9	M	Cannes	statisticien	marié	58	10	260	36,9
10	F	Nice	professeur	veuve	63	7	290	>38
11	M	Paris	étudiant	célibataire	19	5	190	>38

Tableau 18. Recodage plafond de l'attribut température

5.10 Le bruit aléatoire (Brand 2002)

Cette technique, nommée « Random Noise » en anglais, s'applique à un seul attribut à la fois. Elle fonctionne en ajoutant ou en multipliant chaque valeur de l'attribut à anonymiser par une valeur aléatoire. Cette dernière est choisie selon une distribution et une moyenne données. Deux types de bruit aléatoire sont connus : le bruit multiplicatif (« multiplicative noise » en anglais) et le bruit additif (« additive noise » en anglais).

Le bruit multiplicatif consiste à multiplier toutes les valeurs de la colonne à anonymiser par une valeur aléatoire ε. Le bruit additif ajoute une valeur aléatoire ε à une donnée pour en masquer la valeur exacte. Deux variantes de cette technique existent : le bruit additif non corrélé (« uncorrelated additive noise » en anglais) et le bruit

additif corrélé (« correlated additive noise » en anglais). L'objectif du bruit additif non corrélé est de préserver la moyenne et les covariances des données originales tout en perturbant les corrélations et les variances. La technique de bruit additif corrélé, quant à elle, préserve les moyennes et les corrélations des données originales. A titre d'exemple, supposons que la protection de l'attribut JH doive se concrétiser en lui appliquant la technique de bruit additif non corrélé. En se fondant sur le calcul de la moyenne et de la variance des valeurs originales de l'attribut JH, les valeurs de la variable de bruit ont été calculées (voir **Tableau 19**). Sachant que la moyenne des valeurs doit être égale à 0 et que sa variance est proportionnelle à la variance des valeurs originales, nous procédons au remplacement des valeurs originales de **Tableau 7**, ce qui nous permet d'obtenir le **Tableau 20**.

Valeur originale	Valeur aléatoire	Valeur modifiée
3	2	5
1	1	2
2	5	7
7	3	10
40	-10	30
3	8	11
60	-11	49
5	4	9
10	-3	7
7	-2	5
5	3	8

Tableau 19. Calcul du bruit additif

Num tuple	Sexe	Ville	Profession	Statut marital	Age	JH	Choles-térol	Tempé-rature
1	F	Paris	ingénieur	mariée	33	5	150	36,6
2	F	Pontoise	ingénieur	veuve	65	2	290	37,2
3	F	Paris	étudiant	célibataire	21	7	190	36,2
4	M	Nice	étudiant	célibataire	19	10	170	38,5
5	M	Paris	statisticien	marié	36	30	200	40,1
6	F	Cannes	étudiant	mariée	28	11	185	37,5
7	F	Pontoise	statisticien	divorcée	46	49	200	36,5
8	M	Pontoise	statisticien	divorcé	81	9	300	37,6
9	M	Cannes	statisticien	marié	58	7	260	36,9
10	F	Nice	professeur	veuve	63	5	290	38,2
11	M	Paris	étudiant	célibataire	19	8	190	39,3

Tableau 20. Table après application du bruit additif non corrélé

6. Synthèse

Le partage et la mise à disposition de micro-données, pour le soutien de la recherche scientifique ou encore pour satisfaire le besoin d'ouverture des données, mettent en évidence le risque de divulgation d'informations

sensibles et, par voie de conséquence, le risque d'atteinte à la vie privée. Ainsi, par exemple pour maintenir la confiance des répondants aux enquêtes, les éditeurs de micro-données sont confrontés au défi d'assurer leur confidentialité tout en produisant des données de qualité. L'anonymisation des micro-données avant leur publication est considérée comme la mesure de sécurité nécessaire pour contrer des attaques imprévisibles. Cependant, notre étude certes non exhaustive, nous a permis de constater que la réflexion des chercheurs sur les modèles de protection de la vie privée (MPVP) est guidée par la réalisation de scénarios d'attaques. De plus, il n'existe pas de MPVP permettant de contrer tout type d'attaque. Le **Tableau 21**, qui représente la synthèse de notre état de l'art sur les MPVP, confirme notre constat. Ceci explique la variété et le nombre de techniques d'anonymisation de micro-données couverts par la littérature dont on a extrait une partie décrite brièvement dans cet état de l'art.

Modèle de protection De la vie privée	Liaison d'enregistrement	Liaison d'attribut			Liaison de table
		L'attaque d'homogénéité	L'attaque de connaissance acquise	L'attaque d'inférence Probabiliste	
k-anonymat	*				
l-diversité	*	*	*		
(l,c)-diversité	*	*	*	*	
l-diversité d'entropie	*	*	*	*	
t-ferméture		*			
δ-Présence					*

Tableau 21. Les modèles de protection de la vie privée

Il existe de nombreux états de l'art sur les techniques d'anonymisation. Selon le cas, ils émanent de la communauté SDC, PPDM ou PPDP :

- ils sont tous non exhaustifs,
- ils ne proposent pas forcément les mêmes catégorisations des techniques,
- ils n'associent pas toujours une solution à un modèle d'attaque ou à un scénario d'attaque.

Nous avons donc pris l'option de présenter le principe de quelques techniques d'anonymisation de micro-données sachant qu'il existe de nombreuses variantes et que, pour chaque technique, il peut exister plusieurs algorithmes.

A titre d'exemple, pour la technique de « Swapping » de micro-données, nous avons recensé au moins quatre variantes : Random Swap, Rank Swap, C&C Swap et Target Swap. Pour la technique de généralisation, nous avons recensé au moins neuf algorithmes que nous exposons dans le prochain chapitre.

Pour les techniques présentées dans cet état de l'art, nous pouvons constater que certaines sont utilisées pour satisfaire un des modèles de protection de la vie privée citées dans cet état de l'art (**Tableau 22**). Certaines techniques sont perturbatrices, d'autres non (**Tableau 23**). On dit qu'elles sont perturbatrices si les données résultantes sont dénaturées. La généralisation, par exemple, n'est pas perturbatrice parce qu'elle diminue la précision des données mais ne provoque pas d'autre transformation. Ainsi, elle préserve l'usage des données à des fins de test ou de statistique par exemple.

De plus, les techniques citées dans cet état de l'art ne s'appliquent pas à tous les attributs, mais seulement aux attributs continus ou, au contraire, catégoriels, aux attributs sensibles ou aux attributs faisant partie du QI.

Technique	modèle de protection
Généralisation	k-anonymat
Micro-agrégation	k-anonymat
« Bucketization »	l-diversité
« Anatomy »	l-diversité
« Slicing »	k-anonymat et l-diversité

Tableau 22. Techniques d'anonymisation et MPVP cibles

Technique	perturbatrice	non perturbatrice
Généralisation		*
Suppression		*
Micro-agrégation	*	
« Bucketization »	*	
« Anatomy »		*
« Slicing »	*	
« Swapping »	*	
Recodage global		*
« Bottom coding »		*
« Top coding »		*
Bruit aléatoire	*	

Tableau 23. Types de techniques d'anonymisation

7. Conclusion

Ce chapitre nous a permis de faire un tour d'horizon succinct sur l'anonymisation des micro-données. Nous nous sommes focalisés sur la publication de micro-données. Il en ressort que le choix de techniques est guidé par le type de modèle de protection de la vie privée que l'on souhaite mettre en œuvre ainsi que par le domaine d'applicabilité des techniques. Un modèle de protection de la vie privée est défini pour contrer un ou plusieurs scénarios d'attaque. Cet état de l'art est une première étape de validation et d'instanciation du modèle ci-après (voir **Figure 6**), lequel modèle pourra être utilisé comme composant d'un outil d'aide au choix d'une technique d'anonymisation.

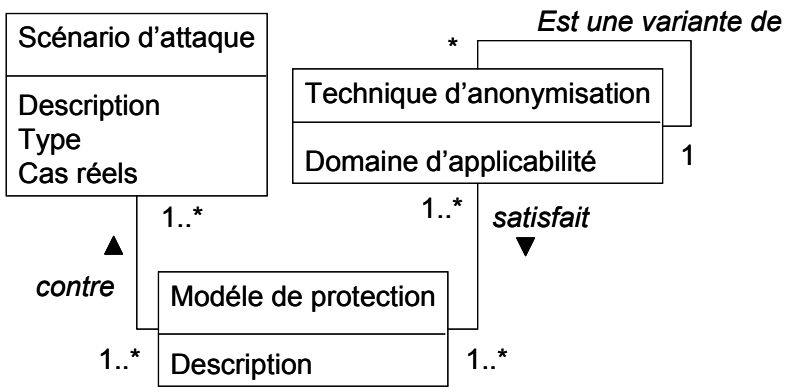


Figure 6. Le modèle des techniques d'anonymisation

Comme on l'a vu plus haut, le choix d'une technique ne suffit pas. Son application passe par l'exécution d'un algorithme. Or, l'état de l'art met en évidence, la plupart du temps, plusieurs algorithmes pour une technique. Nous avons préféré, dans le cadre de cette thèse, nous focaliser dans un premier temps sur la technique de généralisation et ses algorithmes. Le choix de cette technique est motivé par le fait que c'est la technique la plus fréquemment utilisée dans le cadre de la publication de micro-données.

Le chapitre qui suit décrit d'une façon détaillée cette technique et ses algorithmes.

Chapitre 3 Algorithmes et outils de généralisation de micro-données

Au cours de ces dernières années, une attention particulière a été accordée aussi bien par les statisticiens que par les informaticiens à la protection de la vie privée. Beaucoup de recherches ont ainsi ciblé la proposition de techniques et d'algorithmes diminuant le risque de ré-identification de données sensibles tout en maintenant leur utilité. Comme précisé dans le chapitre précédent, ces contributions se concentrent sur le contrôle statistique et/ou sur la publication des données et/ou sur la fouille de données. Dans ces trois contextes, la technique de généralisation semble être l'une des plus explorées. Elle est mise en œuvre via plusieurs algorithmes. Ces algorithmes ont non seulement l'objectif de préservation de la vie privée mais aussi celui de la qualité des données anonymisées qu'ils fournissent. Ils mettent tous en œuvre *a minima* le k-anonymat. Ils sont implémentés, pour certains d'entre eux, dans des logiciels commerciaux ou encore dans des prototypes issus de la recherche et dont l'utilisation n'est pas aussi triviale qu'on pourrait le penser.

Pour éviter que le processus de transformation des données par généralisation ne dégrade trop la précision de celles-ci et ne remette en cause, par-là, leur utilité, la plupart de ces algorithmes utilisent, au cours de leur processus, une métrique permettant d'orienter le codage des données. Ces métriques de guidage sont nommées dans (B. C. M. Fung et al. 2010) « search metrics ». Elles sont le plus souvent associées à un seul algorithme.

D'autres métriques de qualité ou d'évaluation existent, appelées « data metrics » (B. C. M. Fung et al. 2010). Elles permettent, comme leur nom l'indique, de mesurer la qualité des données de la table anonyme en la comparant à la qualité des données de la table originale.

Parmi ces deux types de métriques (de guidage ou de qualité), on distingue des métriques de compromis, des métriques pour tout usage que Fung appelle « general metrics » ou encore des métriques à usage spécifique. Une métrique pour tout usage, comme son nom l'indique, peut être utilisée quand l'on ne connaît pas l'usage des micro-données anonymisées. Une métrique de compromis, quant à elle, sert à établir un équilibre souhaité entre l'utilité des données et la préservation de la vie privée.

Dans ce chapitre, nous décrivons les algorithmes de généralisation de micro-données les plus connus. Ils sont au nombre de neuf : « μ-argus », « Datafly », l'algorithme de Samarati, « Incognito », « Bottom up généralisation », « Top Down spécialisation », « Median Mondrian », « Infogain Mondrian » et « LSD Mondrian ». Pour certains d'entre eux, la métrique de guidage associée est explicitée. Cette section est suivie par une description de quelques métriques d'évaluation proposées dans la littérature ainsi que par un état de l'art sur les outils d'anonymisation les implémentant. Une synthèse conclut ce chapitre.

1. Les algorithmes de généralisation de micro-données

Afin d'illustrer les neuf algorithmes, nous utilisons une table originale de laquelle a été supprimé préalablement l'identifiant des individus. Hormis l'identifiant, cette table est constituée de trois attributs sexe, code postal et niveau d'étude formant le quasi-identifiant (QI) et d'un attribut sensible appelé Salaire (**Tableau 24**). Chaque attribut du QI possède une hiérarchie de généralisation. La **Figure 7**, la **Figure 8** et la **Figure 9** représentent respectivement la hiérarchie de généralisation de l'attribut sexe, du code postal et du niveau d'étude. Dans ces figures, les niveaux de généralisation ont été identifiés par la première lettre de l'attribut correspondant à la généralisation et par un nombre mentionnant la position du niveau dans la hiérarchie. A titre d'exemple, pour la hiérarchie de la **Figure 8**, la valeur « 1305* » est au niveau « Z1 » de cette hiérarchie. Aussi, la valeur « Seconde » se trouvant dans la hiérarchie de la **Figure 9** est au niveau « E0 » de cette hiérarchie. Une description détaillée des déroulements des neuf algorithmes sur la table originale (**Tableau 24**) est présentée dans l'annexe A.

Quasi Identifiant			Attribut sensible
Sexe	**Code postal**	**Niveau d'étude**	**Salaire**
M	13050	5ième	1200
F	13051	3ième	1300
M	13050	Seconde	1200
M	13050	Seconde	1300
M	13051	1ier et 2ième cycle	1500
F	13050	1ier et 2ième cycle	1500
F	13061	1ier et 2ième cycle	1600
F	13061	Master	2000
F	13060	Master	2100
M	13061	Doctorat	3000
M	13060	Doctorat	4000
M	13061	Doctorat	4500

Tableau 24. Table originale

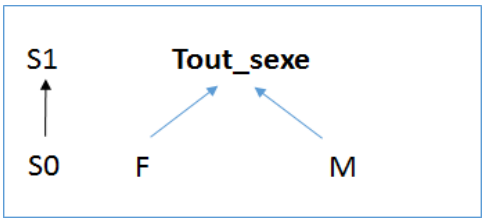


Figure 7. La hiérarchie de généralisation de l'attribut sexe

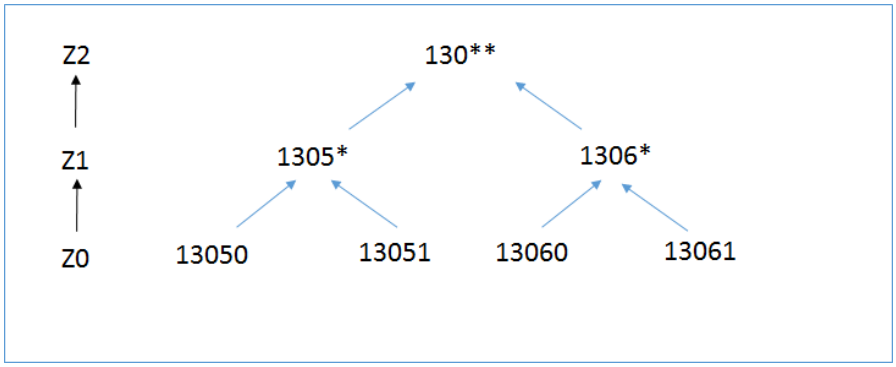


Figure 8. La hiérarchie de généralisation de l'attribut code postal

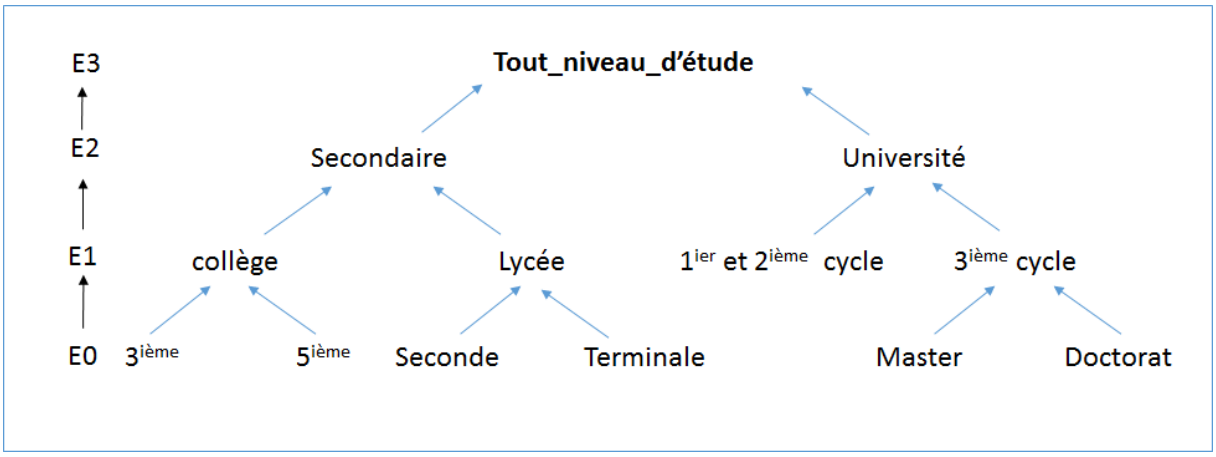


Figure 9. La hiérarchie de généralisation de l'attribut niveau d'étude

1.1. L'algorithme µ-Argus (Burton et al. 1997)

Partant d'une table originale, de la liste des attributs de son QI avec leurs hiérarchies de généralisation et de la valeur de k qui permettrait de satisfaire le degré d'anonymat souhaité, µ-Argus propose une exécution itérative du processus d'anonymisation par généralisation. A chaque itération, l'utilisateur (dans notre cas, l'éditeur de données) choisit l'attribut à généraliser. µ-Argus remplace chaque valeur de cet attribut par la valeur de son parent direct dans la hiérarchie de généralisation correspondante, vérifie la satisfaction du k-anonymat et rend compte à l'utilisateur qui a la possibilité de choisir, dans le cas où l'objectif n'est pas atteint (la table n'est pas k-anonyme), entre la poursuite du processus ou encore la suppression locale de données.

La **Figure 10** représente sa description dans (Sweeney 2002a). Dans cette figure, PT est la table à anonymiser, k est la contrainte de k-anonymat, DGH_{Ai} est la hiérarchie de généralisation de l'attribut A_i et MT la table anonyme résultante. µ-Argus construit, à partir de l'ensemble QI des attributs du quasi-identifiant, une partition en trois ensembles notés : Identifying, More et Most. L'utilisateur doit affecter, à chaque attribut, une valeur comprise entre 0 et 3. Celles-ci correspondent à "not identifying,", "identifying", "most identifying," et "more identifying" traduisant leur rôle dans la ré-identification d'individus. Les enregistrements qui ne satisfont pas le k-anonymat sont stockés dans une liste nommée "outliers" dans cette figure. Notons que µ-Argus ne teste pas toutes les combinaisons des attributs du quasi-identifiant. En effet, comme le montre l'étape 3 de cet algorithme,

seules le sont les combinaisons de deux ou trois attributs. Ces combinaisons doivent contenir au moins un attribut "identifying".

Input: Private Table PT; quasi-identifier QI = (A_1, …, A_n), disjoint subsets of QI known as *Identifying*, *More*, and *Most* where QI = *Identifying* ∪ *More* ∪ *Most*, k constraint; domain generalization hierarchies DGH_{Ai}, where i=1,…,n.
Output: MT containing a generalization of PT[QI]
Assumes: |PT| ≥ k
Method:
1. freq a frequency list containing distinct sequences of values of PT[QI], along with the number of occurrences of each sequence.
2. Generalize each A_i∈QI in freq until its assigned values satisfy k.
3. Test 2- and 3- combinations of *Identifying*, *More* and *Most* and let *outliers* store those cell combinations not having k occurrences.
4. Data holder decides whether to generalize an A_j∈QI based on *outliers* and if so, identifies the A_j to generalize. freq contains the generalized result.
5. **Repeat** steps 3 and 4 until the data holder no longer elects to generalize.
6. Automatically suppress a value having a combination in *outliers*, where precedence is given to the value occurring in the most number of combinations of *outliers*.

Figure 10. L'algorithme µ-argus

A titre d'exemple, supposons que µ-argus est appliqué au **Tableau 24** pour lequel l'utilisateur aura fourni les trois hiérarchies de généralisation représentées dans la **Figure 7**, la **Figure 8** et la **Figure 9** relatives respectivement aux attributs du QI (sexe, code postal et niveau d'étude) et contraint le résultat par k = 2. Puis, lors de l'exécution du processus d'anonymisation, il aura opté, lors de la première itération, pour l'attribut niveau d'étude, puis pour l'attribut code postal et enfin pour l'attribut sexe. Enfin, il aura terminé le processus par une suppression de la valeur de l'attribut « niveau d'étude » dans l'enregistrement qui ne satisfait pas le k-anonymat et fournira une table 2-anonyme (**Tableau 25**). Dans cette table, l'enregistrement touché par une suppression locale est marqué en rouge.

Sexe	Code Postal	Niveau d'étude
Tout-sexe	1305*	secondaire
Tout-sexe	1305*	secondaire
Tout-sexe	1305*	secondaire
Tout-sexe	1305*	secondaire
Tout-sexe	1305*	1ier et 2ième cycle
Tout-sexe	1305*	1ier et 2ième cycle
Tout-sexe	1306*	*******
Tout-sexe	1306*	3ième cycle
Tout-sexe	1306*	3ième cycle
Tout-sexe	1306*	3ième cycle
Tout-sexe	1306*	3ème cycle
Tout-sexe	1306*	3ème cycle

Tableau 25. Résultat de l'application de µ-argus sur la table originale

Le déroulement détaillé de l'algorithme sur cet exemple est fourni en Annexe A de cette thèse.

1.2. L'algorithme Datafly (Sweeney 1997)

Datafly, contrairement à μ -argus, exécute automatiquement des suppressions globales (c'est-à-dire des suppressions de tuples). Ainsi, en plus du choix de la valeur de k, il fixe, au départ, à k le nombre maximum de tuples qu'il a l'autorisation de supprimer. Cependant, afin de minimiser la perte d'information (due aussi bien à la généralisation qu'à la suppression de données), Datafly applique un processus au cours duquel, il fixe, à chacune de ses itérations, l'attribut du QI sur lequel portera la généralisation en choisissant celui qui a le plus de valeurs distinctes. En appliquant cette métrique (nommée DA pour « Distinct Attribute »), Datafly vise à réduire la perte d'information (donc à maintenir l'utilité des données) qui pourrait être engendrée par des généralisations excessives. Ainsi, Datafly généralise les valeurs de l'attribut ayant le plus grand nombre de valeurs distinctes. Puis, il prend sa décision d'arrêt ou de poursuite du processus de généralisation, selon que le nombre de tuples ne satisfaisant pas le k-anonymat est au-dessus ou en-dessous du seuil de suppressions autorisées par l'utilisateur. Dans le cas où il est au-dessus, le processus de généralisation se poursuit. Dans le cas contraire, Datafly procède à la suppression des tuples en question et marque l'arrêt de son processus.

La **Figure 11** représente sa description dans (Sweeney 2002a). Dans cet algorithme, PT est la table originale, k est la contrainte de k-anonymat, DGH_{Ai} est la hiérarchie de généralisation de l'attribut Ai et MGT est la table anonyme résultante.

Input: Private Table PT; quasi-identifier QI = $(A_1, \ldots, A_n)$,
k constraint; hierarchies DGH_{Ai}, where i=1,…,n.
Output: MGT, a generalization of PT[QI] with respect to k
Assumes: |PT| ≥ k
Method:

1. freq — a frequency list contains distinct sequences of values of PT[QI], along with the number of occurrences of each sequence.
2. **while there exists** sequences in freq occurring less than k times that account for more than k tuples **do**
   2.1. **let** A_j be attribute in freq having the most number of distinct values
   2.2. freq □ generalize the values of A_j in freq
3. freq □ suppress sequences in freq occurring less than k times.
4. freq □ enforce k requirement on suppressed tuples in freq.
5. **Return** MGT □ construct table from freq

Figure 11. L'algorithme Datafly

A titre d'exemple, à l'issue de la première itération de Datafly sur le **Tableau 24**, on suppose que l'utilisateur a fourni les trois hiérarchies de généralisation représentées dans la **Figure 7**, la **Figure 8** et la **Figure 9**, relatives respectivement aux attributs du QI sexe, code postal et niveau d'étude et a contraint le résultat en fixant k = 2. Cet algorithme a fourni la table suivante :

Sexe	Code Postal	Niveau d'étude
M	13050	Collège
F	13051	Collège
M	13050	Lycée
M	13050	Lycée
M	13051	1er et 2ème cycle
F	13050	1er et 2ème cycle
F	13061	1er et 2ème cycle
F	13061	3ème cycle
F	13060	3ème cycle
M	13061	3ème cycle
M	13060	3ème cycle
M	13061	3ème cycle

Tableau 26. Détection des enregistrements ne satisfaisant pas le k-anonymat

Cette table a été obtenue après calcul du nombre de valeurs distinctes (dans la table originale) pour chaque attribut du QI sexe, code postal et niveau d'étude. Ce nombre est respectivement de 2, 4 et 7. L'attribut niveau d'éducation, ayant le plus grand nombre, est généralisé. Etant donné que, dans la table résultant de la première itération, le nombre d'enregistrements (marqués en rouge) ne satisfaisant pas le k-anonymat (=8) est supérieur au seuil de suppressions autorisées (=2), alors l'algorithme poursuit la généralisation. La table 2-anonyme, résultat de son exécution totale, est la suivante :

Sexe	Code Postal	Niveau d'étude
M	1305*	Secondaire
M	1305*	Secondaire
M	1305*	Secondaire
M	1305*	Université
F	1305*	Université
F	1306*	Université
F	1306*	Université
F	1306*	Université
M	1306*	Université
M	1306*	Université
M	1306*	Université

Tableau 27. Résultat de l'application de Datafly sur la table originale

Les autres itérations de Datafly à partir du **Tableau 26** sont données en annexe A de cette thèse. L'auteur de cet algorithme a aussi proposé une version améliorée de Datafly, nommée MinGen dans laquelle la métrique DA a

été remplacée par la métrique MD (Minimal distortion metric) afin d'atteindre le k-anonymat avec un minimum de généralisations et de suppressions (Sweeney 2002a).

1.3. L'algorithme de Samarati (Samarati 2001)

L'algorithme de Samarati est fondé sur un treillis qui représente les combinaisons possibles des niveaux de généralisation de tous les attributs du quasi-identifiant. Plus précisément, chaque nœud, dans le treillis, contient une liste décrivant le niveau de généralisation de chaque attribut du QI (voir **Figure 12**). Chaque élément de la liste définit un niveau de généralisation d'un attribut du QI. Ainsi, chaque nœud correspond à la mise en œuvre d'une généralisation possible de la table originale. Supposons que seuls les attributs sexe et code postal composent le QI de la table originale (**Tableau 24**). La **Figure 12** représentera le treillis associé aux deux attributs. La mise en œuvre de <S1, Z1> mènera au **Tableau 28**. Toutes les valeurs de l'attribut "sexe" qui sont au niveau 0 dans la table 1 sont remplacées par leurs parents de niveau 1 dans le **Tableau 28**. La même transformation est effectuée pour les valeurs de l'attribut «code postal».

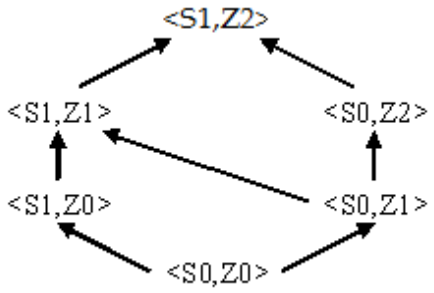


Figure 12. Treillis de généralisation des deux attributs Sexe et Code postal

sexe	code postal	Salaire
Tout_sexe	1305*	1200
Tout_sexe	1305*	1300
Tout_sexe	1305*	1200
Tout_sexe	1305*	1300
Tout_sexe	1305*	1500
Tout_sexe	1305*	1500
Tout_sexe	1306*	1600
Tout_sexe	1306*	2000
Tout_sexe	1306*	2100
Tout_sexe	1306*	3000
Tout_sexe	1306*	4000
Tout_sexe	1306*	4500

Tableau 28. Données généralisées selon <S1,Z1>

Samarati affirme que les meilleurs résultats d'anonymisation sont les nœuds décrivant une généralisation qui satisfait le k-anonymat, éventuellement avec des suppressions sans toutefois dépasser le seuil de suppression (le

nombre de suppressions autorisé). Aussi, afin de minimiser la perte d'information, les nœuds retenus doivent être proches, autant que possible, du niveau le plus bas du treillis, c'est-à-dire de l'état initial (dans notre exemple <S0,Z0>). Monter dans le treillis est nécessaire pour atteindre le k-anonymat mais, à l'inverse, plus on monte plus on perd de la précision dans les données. Ainsi, la généralisation optimale est un compromis qui peut être atteint au niveau de certains nœuds.

Afin de trouver ces nœuds « optimaux », l'algorithme agit par itération et, en considérant les nœuds du niveau h/2, h étant la hauteur de la partie inexplorée du treillis (tout le treillis est considéré à la première itération). A titre d'exemple, pour le treillis de la **Figure 12**, la hauteur h est égale à 3. Les nœuds situés au niveau h/2 sont ceux situés à la valeur entière de h/ 2 = 1.5, soit au niveau 1, c'est-à-dire les nœuds <S1,Z0> et <S0,Z1>.

Avant d'entamer la première itération, Samarati construit le treillis qui correspond à la table originale et initialise h à n/2, n étant le nombre de niveaux du treillis sachant que le nœud le plus bas dans la hiérarchie est de niveau 0. A titre d'exemple, le treillis construit sur la base des hiérarchies de généralisation de la **Figure 7**, la **Figure 8** et la **Figure 9** relatives au **Tableau 24** est le suivant (voir **Figure 13**):

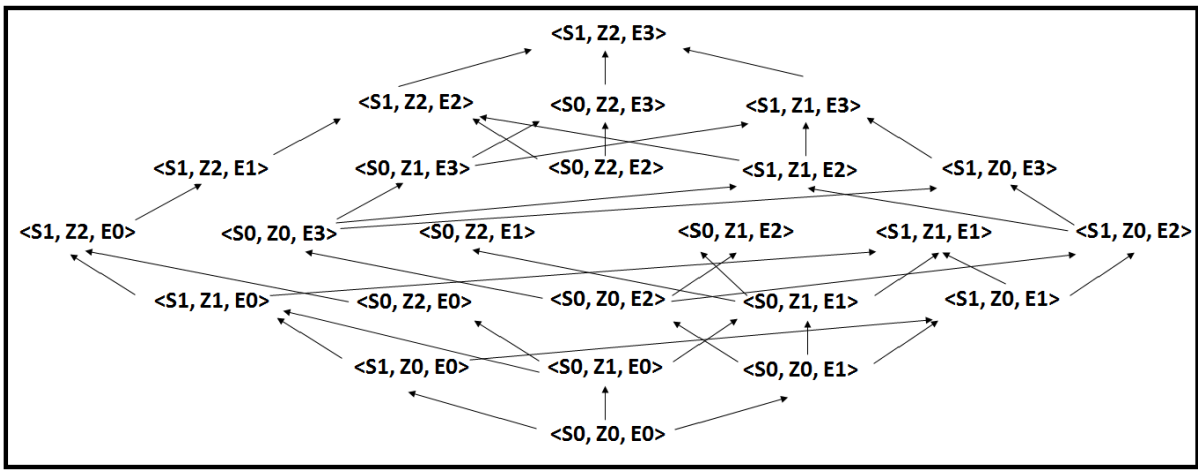


Figure 13. Treillis de généralisation de la table originale

Par la suite, chaque itération fonctionne ainsi. Si au niveau h/2, au moins un nœud permet de satisfaire le k-anonymat (par application de la généralisation proposée par le nœud sur la table en cours d'anonymisation et éventuellement de la suppression globale, tout en ne dépassant pas le seuil autorisé), Samarati stocke ce nœud ainsi que tous les nœuds de ce niveau permettant de satisfaire le k-anonymat (k est égal à 2), avec ou sans suppression globale. Ensuite, il se concentre, à l'itération suivante, sur la moitié inférieure du treillis et calcule la nouvelle valeur de h /2. Si, à ce niveau, aucun nœud ne permet de satisfaire le k-anonymat, l'algorithme cible, à l'itération suivante, la moitié supérieure du treillis pour continuer sa recherche des nœuds acceptables. Il s'arrête lorsque h est égal à 0 et restitue à l'utilisateur les derniers nœuds mémorisés.

A titre d'exemple, les nœuds recensés à la première itération de Samarati sur le **Tableau 24** moyennant le treillis de la **Figure 13** sont les nœuds de niveau 3 se trouvant dans ce treillis, c'est-à-dire <S1, Z2, E0>, <S0, Z0, E3>, <S0, Z2, E1>, <S0, Z1, E2>, <S1, Z1, E1> et <S1, Z0, E2>. Si l'on émet l'hypothèse que le nombre de

suppressions autorisées est 3, alors le nœud <S1,Z0,E2> générant la généralisation ci-dessous (voir **Tableau 29**), dans laquelle on aura supprimé le second tuple (tuple en rouge) est à mémoriser car cette dernière, après la suppression globale, satisfait le k-anonymat.

Sexe	Code Postal	Niveau d'étude
tout-sexe	13050	Secondaire
tout-sexe	13051	Secondaire
tout-sexe	13050	Secondaire
tout-sexe	13050	Secondaire
tout-sexe	13051	Université
tout-sexe	13051	Université
tout-sexe	13061	Université
tout-sexe	13061	Université
tout-sexe	13060	Université
tout-sexe	13061	Université
tout-sexe	13060	Université
tout-sexe	13061	Université

Tableau 29. La table de généralisation <S1, Z0, E2>

L'existence d'au moins un nœud au niveau 3 satisfaisant le k-anonymat oblige Samarati à itérer en ciblant la moitié inférieure du treillis de la **Figure 13**. Comme le montre le déroulement pas à pas, présenté en annexe A de cette thèse, sur notre exemple illustratif, Samarati livrera en fin d'exécution la table anonyme correspondant au nœud <S0, Z1, E1>.

La **Figure 14** est l'algorithme de Samarati tel que présenté dans (Samarati 2001). L'auteur a spécifié les entrées, les sorties et l'ensemble des instructions de l'algorithme. Ainsi, les entrées sont : la table originale T, la contrainte k de k-anonymat, le nombre maximum Maxsup de tuples à supprimer, VL le treillis et DGH les hiérarchies de généralisation. GT est la table anonymisée qui représente la sortie de l'algorithme.

Find_vector

INPUT: Table $T_i = \mathsf{PT}[QI]$ to be generalized, anonymity requirement k, suppression threshold MaxSup, lattice VL_{DT} of the distance vectors corresponding to the domain generalization hierarchy DGH_{DT}, where DT is the tuples of the domains of the quasi-identifier attributes.

OUTPUT: The distance vector *sol* of a generalized table GT_{sol} that is a k-minimal generalization of $\mathsf{PT}[QI]$ according to Definition 4.3.

METHOD: Executes a binary search on VL_{DT} based on height of vectors in VL_{DT}.

1. *low* :=0; *high*:=*height*(T, VL_{DT}); *sol* := T
2. **while** *low* < *high*
   - 2.1 *try* := $\lfloor \frac{low + high}{2} \rfloor$
   - 2.2 *Vectors* := {*vec* | *height*(*vec*, VL_{DT}) = *try*}
   - 2.3 *reach_k*:= `false`
   - 2.4 **while** *Vectors* $\neq \emptyset \wedge$ *reach_k* $\neq$ `true` **do**
     - Select and remove a vector *vec* from *Vectors*
     - **if** satisfies(*vec*,*k*,T_i,MaxSup) **then** *sol*:= *vec*; *reach_k*:= `true`
   - 2.5 **if** *reach_k* = `true` **then** *high*:= *try* **else** *low*:= *try* + 1
3. **Return** *sol*

Figure 14. L'algorithme de Samarati

Dans cet algorithme, l'auteur a utilisé plusieurs variables et fonctions. La variable *try* représente la mi-hauteur de la zone de recherche délimitée par les deux variables *low et height* (la zone de recherche initiale est le treillis entier). La variable *Vectors* est l'ensemble des nœuds qui se trouvent au niveau *try*. Enfin, la fonction *satisfies* vérifie si un nœud *vec* satisfait le k-anonymat.

1.4. L'algorithme Incognito (LeFevre, DeWitt, et Ramakrishnan 2005)

Incognito est également fondé sur un treillis. Cependant, le treillis est construit de manière itérative et incrémentale afin d'atteindre une plus grande efficacité. En d'autres termes, à la première itération, Incognito construit tous les treillis liés à un attribut du QI. Chaque treillis correspond à une hiérarchie de généralisation. Ces différents treillis sont nettoyés en supprimant tous les nœuds qui ne conduisent pas à une généralisation k-anonyme. A l'itération 2, Incognito construit, par fusion des treillis résultant de l'étape précédente, les treillis à deux attributs et, comme dans l'étape précédente, il procède à leurs nettoyages. Ce processus se poursuit de façon itérative jusqu'à ce que le treillis regroupant tous les attributs du QI soit construit et nettoyé.

A titre d'exemple, l'itération 1 de Incognito, appliquée au **Tableau 24**, fournira autant de treillis que d'attributs du QI (trois dans notre cas : un treillis pour le sexe, un autre pour le code postal et un autre pour le niveau d'étude). Tous les nœuds du premier treillis (c'est-à-dire S1 et S0) sont conservés car ils permettent de déduire une généralisation 2-anonyme. Ce qui n'est pas le cas des nœuds Z0 et E0 des deux autres treillis. Par conséquent, pour construire les treillis de l'itération 2 (c'est-à-dire le treillis fondé sur les attributs sexe et code postal, celui fondé sur l'attribut sexe et niveau d'étude ainsi que celui fondé sur les attributs code postal et niveau d'étude), Incognito ne considère que les treillis précédents satisfaisant le k-anonymat. Autrement dit, il exclut les nœuds Z0 et E0. Ainsi, pour le premier groupe d'attributs, on obtient le treillis suivant (voir **Figure 15**) :

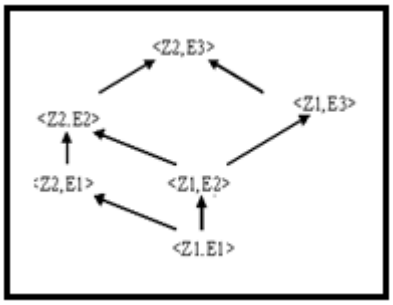
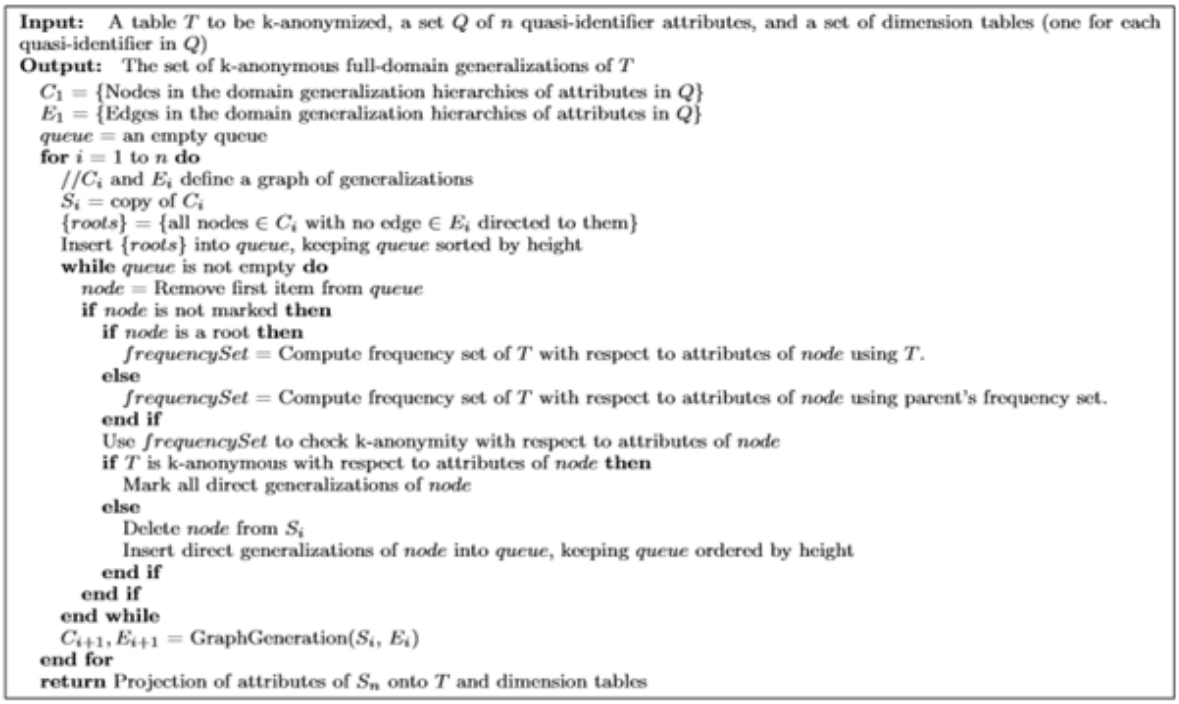


Figure 15. Treillis des attributs code postal et niveau d'études de la deuxième itération

La suite du déroulement pas à pas de cet algorithme est donnée dans l'annexe A de cette thèse.

Notons que la construction par fusion des treillis dans Incognito repose sur trois propriétés citées et prouvées dans (LeFevre, DeWitt, et Ramakrishnan 2005).

Notons aussi que, bien que plus rapide que Samarati, Incognito est de complexité exponentielle, qu'il s'agisse du temps d'exécution ou de l'espace mémoire, relativement à la taille des données (Ayala-Rivera et al. 2014). Par conséquent, l'utilisation de ces deux algorithmes sur de gros volumes de données n'est pas conseillée. La **Figure 16** correspond à l'algorithme Incognito tel que présenté dans (LeFevre, DeWitt, et Ramakrishnan 2005).



Input: A table T to be k-anonymized, a set Q of n quasi-identifier attributes, and a set of dimension tables (one for each quasi-identifier in Q)
Output: The set of k-anonymous full-domain generalizations of T

```
C1 = {Nodes in the domain generalization hierarchies of attributes in Q}
E1 = {Edges in the domain generalization hierarchies of attributes in Q}
queue = an empty queue
for i = 1 to n do
  //Ci and Ei define a graph of generalizations
  Si = copy of Ci
  {roots} = {all nodes ∈ Ci with no edge ∈ Ei directed to them}
  Insert {roots} into queue, keeping queue sorted by height
  while queue is not empty do
    node = Remove first item from queue
    if node is not marked then
      if node is a root then
        frequencySet = Compute frequency set of T with respect to attributes of node using T.
      else
        frequencySet = Compute frequency set of T with respect to attributes of node using parent's frequency set.
      end if
      Use frequencySet to check k-anonymity with respect to attributes of node
      if T is k-anonymous with respect to attributes of node then
        Mark all direct generalizations of node
      else
        Delete node from Si
        Insert direct generalizations of node into queue, keeping queue ordered by height
      end if
    end if
  end while
  Ci+1, Ei+1 = GraphGeneration(Si, Ei)
end for
return Projection of attributes of Sn onto T and dimension tables
```

Figure 16. L'algorithme Incognito

1.5. L'algorithme « Bottom up generalization »

Cet algorithme a été proposé par (Wang, Yu, et Chakraborty 2004). Il est destiné à préserver les données pour un type spécifique de traitement statistique qu'est la classification. Il parcourt les hiérarchies de généralisation, des feuilles vers la racine, comme son nom l'indique. Il exécute, itérativement, des généralisations qu'il considère comme « bonnes » dans le sens où elles préservent au mieux la qualité de la classification (c'est-à-dire qu'elles minimisent la perte d'information qui pourrait affecter la classification) tout en fournissant le k-anonymat souhaité. Chaque « bonne » généralisation est sélectionnée parmi un ensemble de généralisations candidates. Une généralisation G, notée {di}→g, est la tâche consistant à remplacer toutes les valeurs filles di de l'ensemble {di} par leur valeur parente g. G est considérée comme candidate par rapport à une table si les descendants directs d1, d2, …, di, etc., notés {di} de g dans la hiérarchie de généralisation sont également dans la table. Elle est considérée comme bonne si elle renvoie le meilleur score calculé par application de la métrique de compromis IL/AG (Information Loss/Anonymity Gain) dont le rôle est de mesurer la perte d'information concernant la classification et le gain en sécurité liés à l'anonymisation.

La formule permettant de calculer le score d'une généralisation G, noté IL/AG(G) est la suivante :

$$IL/AG(G) = \begin{cases} \dfrac{InformationLoss(G)}{AnonymityGain(G)} & if \quad AnonymityGain(G) \neq 0 \\ InformationLoss(G) & otherwise \end{cases}.$$

InformationLoss (G) correspond à la perte d'information suite à la réalisation de la généralisation G. Elle vise à garantir que le modèle de classification généré par l'ensemble de données anonymes a une efficacité

approximativement équivalente au modèle de classification généré par les données originales. Elle se définit comme suit :

$$InformationLoss(G) = Entropy(R_g) - \sum_{di} \frac{|R_{di}|}{|R_g|} Entropy(R_{di})$$

où R_g (respectivement R_{di}) représente l'ensemble des enregistrements contenant la valeur g (respectivement la valeur di).

Entropy (R_x) où $x \in \{g, di\}$ correspond à l'entropie de l'ensemble R_x.

Elle se calcule de la façon suivante :

$$Entropie(R_x) = -\sum_{cls} \frac{freq(R_x, cls)}{|R_x|} \times \log_2 \frac{freq(R_x, cls)}{|R_x|}$$

où freq(R_x, cls) représente le pourcentage d'individus de la classe labellisée cls dans R_x. Rappelons que la classification vise à classer les ensembles en catégories où chaque classe ou catégorie est labellisée.

AnonymityGain(G) correspond au gain d'anonymat qui pourrait être engendré suite à la réalisation de la généralisation G. Intuitivement, cette mesure se calcule en comparant le degré d'anonymat d'une table avant et après application d'une généralisation G. De façon formelle, elle équivaut à :

Anonymat(T, après G) - Anonymat(T, avant G)

Où :

- Anonymat(T, après G) correspond à la taille de la plus petite classe d'équivalence (la classe d'équivalence qui renferme le plus petit nombre d'individus partageant le même QI) de T après application de G,
- et Anonymat(T, avant G) correspond à taille de la plus petite classe d'équivalence de T avant application de G.

Rappelons que, dans une table de micro-données, chacune des différentes valeurs du QI constitue une classe d'équivalence.

Ainsi, à chaque itération, l'algorithme « Bottom up generalization » choisit, à partir d'un ensemble de généralisation candidates, une parmi celles qu'il qualifie de bonnes au regard de la métrique de compromis IL/AG. La généralisation sélectionnée est appliquée à la table en cours d'anonymisation. Lors de la première itération, cette table est la table originale. A l'itération i, elle correspond à celle issue de l'itération i-1. Le processus s'arrête lorsque la table en cours d'anonymisation satisfait le k-anonymat.

Pour une meilleure illustration de l'exécution de cet algorithme, nous avons modifié la table originale (**Tableau 24**) en procédant à des duplications de tuples. La table modifiée est donnée dans le **Tableau 30**.

Sexe	Code Postal	Niveau d'étude	Classe
M	13050	5ème	1200
M	13050	5ème	1200
F	13051	3ème	1300
F	13051	3ème	1300
F	13051	3ème	1300
M	13050	Seconde	1200
M	13050	Seconde	1200
F	13051	Seconde	1300
F	13051	Seconde	1300
F	13051	Seconde	1300
F	13050	1er et 2ème cycle	1500
F	13050	1er et 2ème cycle	1500
F	13050	1er et 2ème cycle	1500
F	13050	1er et 2ème cycle	1500
F	13061	1er et 2ème cycle	2000
F	13061	1er et 2ème cycle	2000
F	13060	Master	2100
F	13060	Master	2100
F	13060	Master	2100
F	13060	Master	2100
M	13061	Master	3000
M	13060	Doctorat	4000
M	13060	Doctorat	5000

Tableau 30. La table originale modifiée

L'algorithme « bottom up généralisation » commence par calculer le score des généralisations candidates se trouvant en bas des hiérarchies en appliquant la métrique de classification InformationLoss. Dans notre cas, il s'agira de déterminer le score des généralisations représentées dans la figure suivante (**Figure 17**).

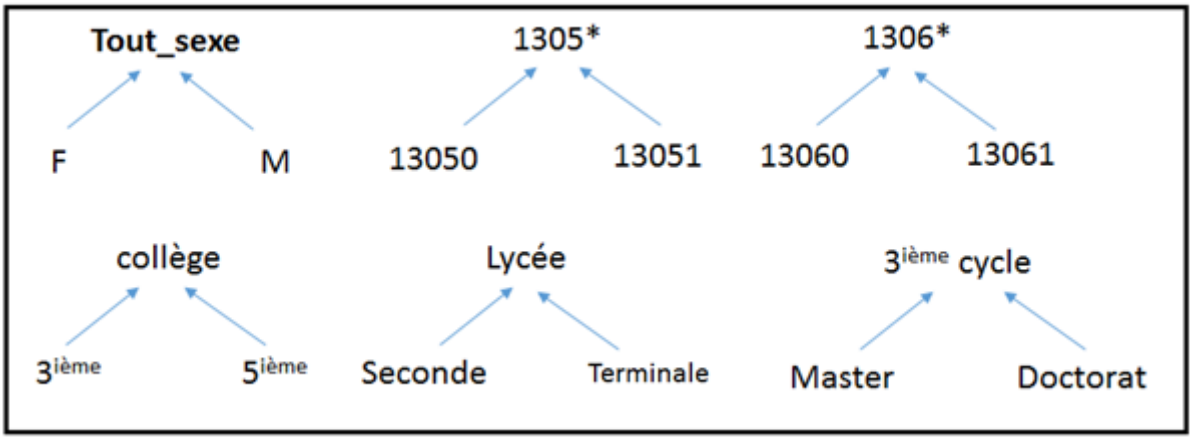


Figure 17. Les généralisations candidates de la table originale

Parmi ces généralisations, celles qui réalisent le meilleur score sont :
"{13050,13051}→1305*", "{3ème, 5ème}→ collège", "{seconde, terminale}→lycée", "{master, doctorat}→ 3ième cycle". L'une d'entre elles est choisie pour être appliquée sur la table originale. En choisissant, par exemple, la

première généralisation, nous obtenons la table ci-après (**Tableau 31**). Cette dernière, n'étant pas 2-anonyme, nous oblige à réitérer le processus afin d'anonymiser encore plus cette table.

L'annexe A présente la totalité des itérations effectuées ainsi que le résultat final obtenu.

Sexe	Code Postal	Niveau d'étude	Classe
M	1305*	5ème	1200
M	1305*	5ème	1200
F	1305*	3ème	1300
F	1305*	3ème	1300
F	1305*	3ème	1300
M	1305*	Seconde	1200
M	1305*	Seconde	1200
F	1305*	Seconde	1300
F	1305*	Seconde	1300
F	1305*	Seconde	1300
F	1305*	1er et 2ème cycle	1500
F	1305*	1er et 2ème cycle	1500
F	1305*	1er et 2ème cycle	1500
F	13050	1er et 2ème cycle	1500
F	13061	1er et 2ème cycle	2000
F	13061	1er et 2ème cycle	2000
F	13060	Master	2100
F	13060	Master	2100
F	13060	Master	2100
F	13060	Master	2100
M	13061	Master	3000
M	13060	Doctorat	4000
M	13060	Doctorat	5000

Tableau 31. Résultat de la première itération de l'algorithme « bottom up généralisation »

L'algorithme « bottom up généralisation », tel qu'il est présenté dans la littérature, est fourni dans la **Figure 18**.

Algorithm 1 The bottom-up generalization

1: **while** R does not satisfy the anonymity requirement **do**
2: **for all** generalization G **do**
3: compute $IP(G)$;
4: **end for**;
5: find the best generalization G_{best};
6: generalize R by G_{best};
7: **end while**;
8: output R;

Figure 18. L'algorithme "bottom up généralisation"

1.6. L'algorithme « Top down specialization » (B. C. Fung, Wang, et Yu 2005)

Comme la généralisation ascendante, la spécialisation descendante, communément appelée TDS ou algorithme « Top down specialization », est destinée à rendre les données propices à la classification. Cependant, contrairement à l'algorithme « bottom up généralization », TDS suit un parcours de la racine vers les feuilles des hiérarchies de généralisation.

Partant du principe qu'une généralisation maximale de toutes les valeurs de la table originale (c'est-à-dire dans laquelle toutes les valeurs sont remplacées par les valeurs des racines des hiérarchies de généralisation) permet de préserver le k-anonymat, mais affecte la qualité des données de la table résultante, l'algorithme effectue des itérations pour trouver et appliquer des spécialisations valides et bénéfiques, c'est-à-dire celles qui, non seulement, préservent le k-anonymat (contrainte de validité), mais qui génèrent aussi le moins de perte d'information, offrant ainsi une meilleure qualité pour la classification.

Soit S une spécialisation, notée aussi $a \rightarrow \{s_i\}$. S est la tâche consistant à remplacer, dans la table à anonymiser, la valeur « a » par l'une des valeurs filles « s_i » de $\{s_i\}$ se trouvant dans la hiérarchie de généralisation. Elle est considérée comme valide si elle respecte le k-anonymat et si elle renvoie le meilleur score par application de la métrique de compromis IG/AL (InformationGain/Anonymityloss). Cette métrique permet de réaliser le compromis entre le gain d'information et la perte d'anonymat dus à la spécialisation.

La formule permettant de calculer le score d'une spécialisation S, noté IG/AL(S), est la suivante :

$$\text{IG/AL(S)} = \begin{cases} \dfrac{InformationGain(S)}{AnonymityLoss(S)} & if \quad AnonymityLoss(S) \neq 0 \\ InformationGain(S) & otherwise \end{cases}.$$

InformationGain(S) correspond au gain d'information suite à la réalisation de la spécialisation S. Elle se définit comme suit :

$$InformationGain(S) = Entropy(R_a) - \sum_{si} \frac{|R_{si}|}{|R_a|} Entropy(R_a)$$

où R_a (respectivement R_{si}) représentent l'ensemble des enregistrements contenant la valeur a (respectivement la valeur si).

Entropy (R_x) où $x \in \{a, si\}$ correspond à l'entropie de l'ensemble R_x.

Elle se calcule de la façon suivante :

$$Entropie(R_x) = -\sum_{cls} \frac{freq(R_x, cls)}{|R_x|} \times \log_2 \frac{freq(R_x, cls)}{|R_x|}$$

où freq(R_x, cls) représente le pourcentage d'individus de la classe labellisée cls dans R_x. Rappelons que la classification vise à classifier les ensembles en catégories où chaque classe ou catégorie est labellisée.

AnonymityLoss (S) correspond à la perte d'anonymat qui pourrait être engendrée suite à la réalisation de la spécialisation S.

Intuitivement, cette mesure se calcule en comparant le degré d'anonymat d'une table avant et après application d'une spécialisation S. De façon formelle, elle équivaut à :

Anonymat(T, après S) - Anonymat(T, avant S)

Où :

- Anonymat(T, après S) correspond à la taille de la plus petite classe d'équivalence (la classe d'équivalence qui renferme le plus petit nombre d'individus partageant le même QI) de T après application de S,
- et Anonymat(T, avant S) correspond à la taille de la plus petite classe d'équivalence de T avant application de S.

En pratique, TDS commence par effectuer une généralisation maximale de toutes les valeurs du QI de la table originale. C'est l'étape d'initialisation du processus. A cette étape, le k-anonymat est satisfait. Ensuite, de façon itérative, en parcourant les hiérarchies de généralisation de la racine vers les feuilles, il recherche des spécialisations valides et bénéfiques parmi les spécialisations candidates et calcule leurs scores en appliquant la métrique IG/AL. Si plusieurs spécialisations valides et bénéfiques disposent d'un même meilleur score, l'une d'entre elles est sélectionnée aléatoirement pour être appliquée sur la table en cours d'anonymisation. Lors de la première itération, la table en cours d'anonymisation est celle obtenue à l'étape d'initialisation de TDS. Le processus s'arrête en l'absence d'au moins une spécialisation valide.

A titre d'exemple, en effectuant la première itération de TDS sur notre exemple illustratif représenté par le **Tableau 30**, on déduit les spécialisations valides et bénéfiques représentées dans la figure suivante (**Figure 19**) :

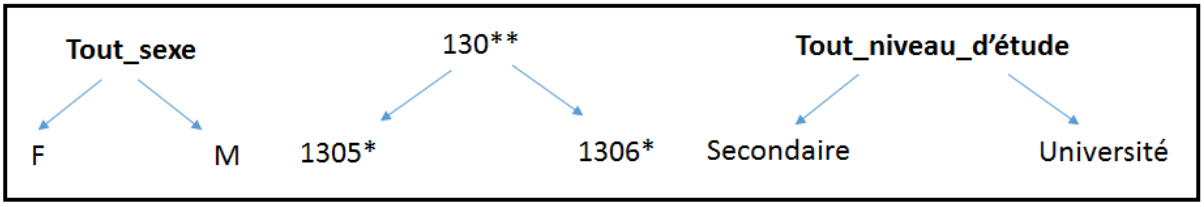


Figure 19. Les spécialisations valides du **Tableau 30**

L'une des spécialisations, parmi celles obtenant le meilleur score, sera retenue pour être appliquée sur la table originale généralisée au maximum.

Dans notre cas, il s'agit de l'unique spécialisation Tout_niveau_d'étude → {secondaire, université}. Celle-ci appliquée sur le **Tableau 32** obtenu lors de l'initialisation de TDS fournira le **Tableau 33** en cours d'anonymisation.

Sexe	Code Postal	Niveau d'étude	salaire
tout_sexe	130**	tout_niveau_d'étude	1200
tout_sexe	130**	tout_niveau_d'étude	1200
tout_sexe	130**	tout_niveau_d'étude	1300
tout_sexe	130**	tout_niveau_d'étude	1300
tout_sexe	130**	tout_niveau_d'étude	1300
tout_sexe	130**	tout_niveau_d'étude	1200
tout_sexe	130**	tout_niveau_d'étude	1200
tout_sexe	130**	tout_niveau_d'étude	1300
tout_sexe	130**	tout_niveau_d'étude	1300
tout_sexe	130**	tout_niveau_d'étude	1300
tout_sexe	130**	tout_niveau_d'étude	1500
tout_sexe	130**	tout_niveau_d'étude	1500
tout_sexe	130**	tout_niveau_d'étude	1500
tout_sexe	130**	tout_niveau_d'étude	1500
tout_sexe	130**	tout_niveau_d'étude	2000
tout_sexe	130**	tout_niveau_d'étude	2000
tout_sexe	130**	tout_niveau_d'étude	2100
tout_sexe	130**	tout_niveau_d'étude	2100
tout_sexe	130**	tout_niveau_d'étude	2100
tout_sexe	130**	tout_niveau_d'étude	2100
tout_sexe	130**	tout_niveau_d'étude	3000
tout_sexe	130**	tout_niveau_d'étude	4000
tout_sexe	130**	tout_niveau_d'étude	5000

Tableau 32. Initialisation de la table par l'algorithme TDS

Sexe	Code Postal	Niveau d'étude	salaire
tout_sexe	130**	Secondaire	1200
tout_sexe	130**	Secondaire	1200
tout_sexe	130**	Secondaire	1300
tout_sexe	130**	Secondaire	1300
tout_sexe	130**	Secondaire	1300
tout_sexe	130**	Secondaire	1200
tout_sexe	130**	Secondaire	1200
tout_sexe	130**	Université	1300
tout_sexe	130**	Université	1300

tout_sexe	130**	Université	1300
tout_sexe	130**	Université	1500
tout_sexe	130**	Université	1500
tout_sexe	130**	Université	1500
tout_sexe	130**	Université	1500
tout_sexe	130**	Université	2000
tout_sexe	130**	Université	2000
tout_sexe	130**	Université	2100
tout_sexe	130**	Université	2100
tout_sexe	130**	Université	2100
tout_sexe	130**	Université	2100
tout_sexe	130**	Université	3000
tout_sexe	130**	Université	4000
tout_sexe	130**	Université	5000

Tableau 33. Résultat après la première itération de l'algorithme TDS

Après quatre itérations, TDS fournit la table anonyme suivante (**Tableau 34**) :

Sexe	**Code Postal**	**Niveau d'étude**	**salaire**
M	130**	Secondaire	1200
M	130**	Secondaire	1200
F	130**	Secondaire	1300
F	130**	Secondaire	1300
F	130**	Secondaire	1300
M	130**	Secondaire	1200
M	130**	Secondaire	1200
F	130**	1er et 2ème cycle	1300
F	130**	1er et 2ème cycle	1300
F	130**	1er et 2ème cycle	1300
F	130**	1er et 2ème cycle	1500
F	130**	1er et 2ème cycle	1500
F	130**	1er et 2ème cycle	1500
F	130**	1er et 2ème cycle	1500
F	130**	1er et 2ème cycle	2000
F	130**	3ème cycle	2000
F	130**	3ème cycle	2100
F	130**	3ème cycle	2100
F	130**	3ème cycle	2100

F	130**	3ème cycle	2100
M	130**	3ème cycle	3000
M	130**	3ème cycle	4000
M	130**	3ème cycle	5000

Tableau 34. Résultat final de l'algorithme TDS

L'algorithme « Top down spécialisation » tel qu'il est présenté dans (B. C. Fung, Wang, et Yu 2005) est fourni à la **Figure 20**.

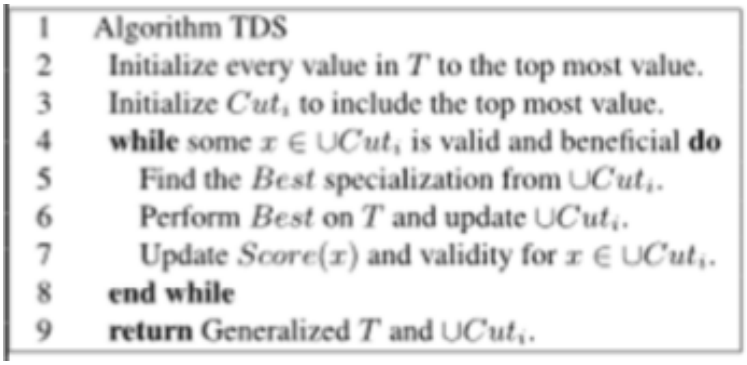


Figure 20. L'algorithme top down spécialisation

$\cup CUT_i$ représente toutes les spécialisations candidates. Lors de l'étape d'initialisation (ligne 3) $\cup CUT_i$ contient un seul enregistrement, noté CUT_i dans l'algorithme.

1.7. L'algorithme Median Mondrian (LeFevre, DeWitt, et Ramakrishnan 2006)

Comme tous les algorithmes de généralisation, le principe de Median Mondrian est de diviser l'ensemble des enregistrements contenus dans la table originale en groupes contenant au moins 'k' individus. Les personnes qui appartiennent au même groupe sont celles qui ont la même valeur du QI. Pour satisfaire cette contrainte d'anonymat, Median Mondrian représente les enregistrements dans un espace multidimensionnel où chaque dimension correspond à un attribut du QI.

La position initiale d'un enregistrement dans l'espace multidimensionnel tient compte de la valeur des attributs du QI. Le découpage de l'espace en zones permet la constitution de groupes d'individus. Il s'effectue selon la valeur médiane de la dimension correspondante. Au fur et à mesure de l'exécution de l'algorithme, l'espace est re-partitionné et les enregistrements repositionnés dans les groupes décrits par la nouvelle partition. Plus précisément, à chaque itération, l'algorithme choisit une dimension (c'est-à-dire un attribut du QI) et vérifie la possibilité de diviser un groupe en deux sous-groupes (en divisant la zone selon la valeur médiane de cette dimension). La division est possible si, dans chaque groupe résultant, il existe au moins k individus (satisfaction du k-anonymat). Chaque groupe, pour lequel la division n'est pas autorisée, est marqué. Lorsque tous les groupes sont marqués pour une dimension, l'algorithme passe à une autre dimension. Il arrête son processus de marquage lorsque toutes les dimensions ont été explorées. Ensuite, il applique les généralisations proposées, en remplaçant

les différentes valeurs d'une même zone par la valeur de leur premier parent commun. On parle de processus de recodage pour désigner cette dernière étape.

A titre d'exemple, si l'on suppose que le QI du **Tableau 24** est composé uniquement des attributs code postal et niveau d'étude, alors la répartition initiale des tuples (individus dans l'espace multidimensionnel - dans notre cas bidimensionnel) est donnée dans la **Figure 21**. Dans cette dernière, les nombres entre parenthèses indiquent chacun le nombre d'enregistrements partageant le même QI.

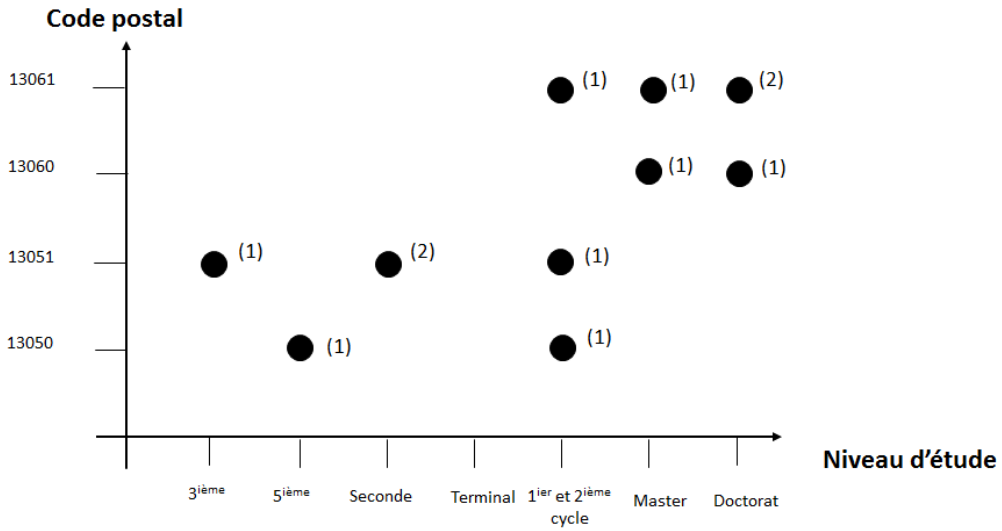


Figure 21. Représentation de la table originale dans un schéma multidimensionnel

Si l'algorithme choisit, à la première itération, le niveau d'étude comme première dimension à explorer, alors le partitionnement selon la médiane de cette dimension donnerait la répartition en groupes de la **Figure 22**.

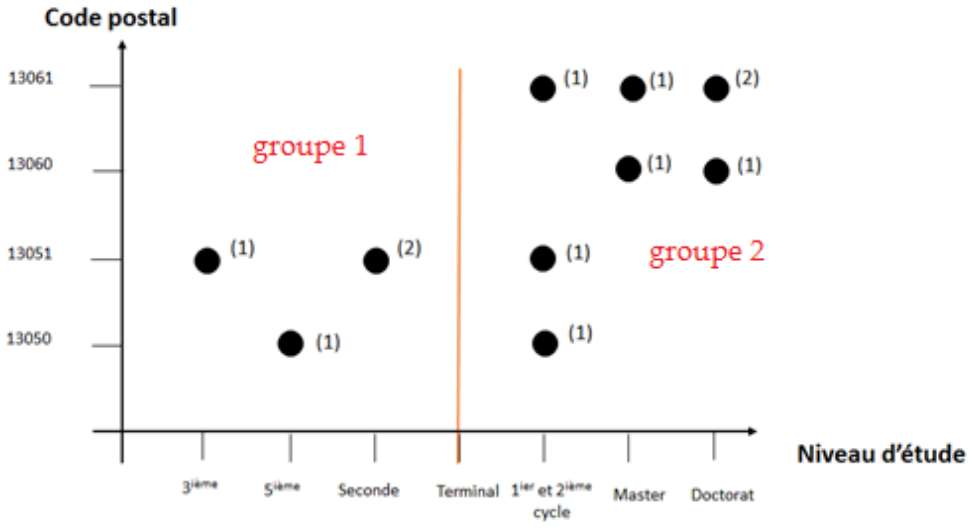


Figure 22. Le résultat de la première itération de l'algorithme Median Mondrian

En sélectionnant le groupe 1 par exemple, l'algorithme pourra procéder à un partitionnement qui donnerait le schéma de la **Figure 23**.

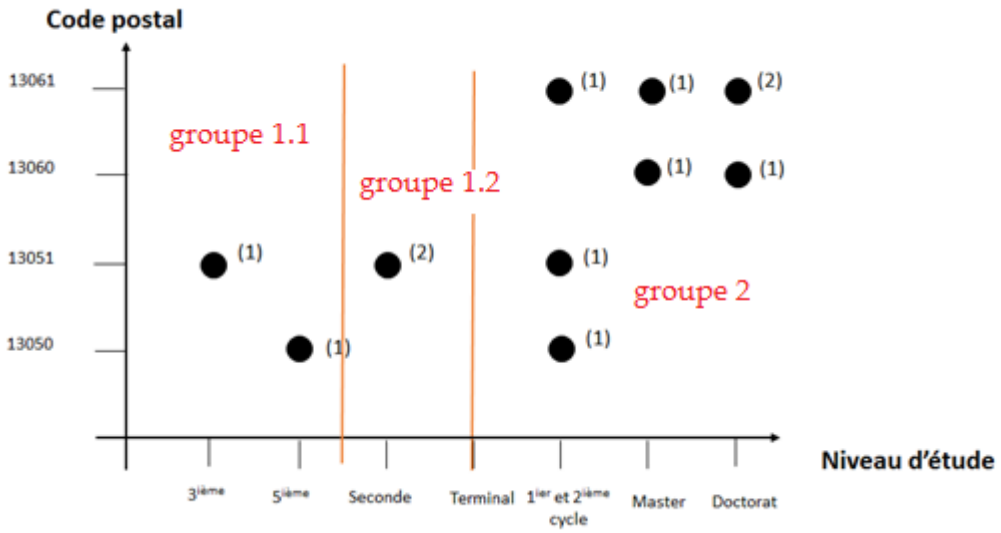


Figure 23. Résultat de la deuxième itération de l'algorithme Median Mondrian

Les groupes 1.1 et 1.2 issus de ce partitionnement ne pourront pas être re-partitionnés car cela violerait le 2-anonymat. Le partitionnement sera donc opéré pour le groupe 2.

La **Figure 24** montre le résultat final du processus de division de l'espace initial. Le **Tableau 35** est la table 2-anonyme générée à partir du recodage proposé dans la **Figure 24**.

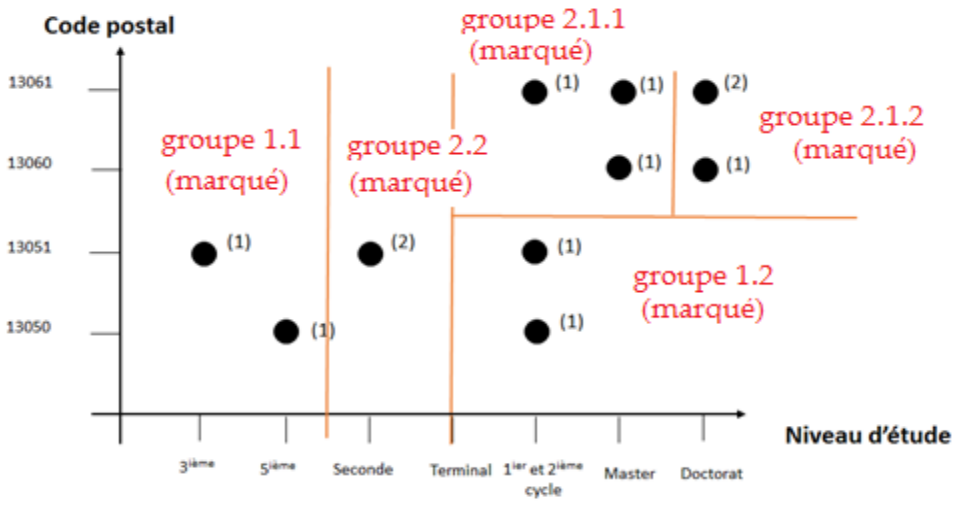


Figure 24. Résultat final de l'algorithme Median Mondrian

code postal	niveau d'étude	Salaire
1305*	collège	1200
1305*	collège	1300
13051	Seconde	1200
13051	Seconde	1300
1305*	1ier et 2ième cycle	1500

1305*	1ier et 2ième cycle	1500
1306*	université	1600
1306*	université	2000
1306*	université	2100
1306*	Doctorat	3000
1306*	Doctorat	4000
1306*	Doctorat	4500

Tableau 35. Recodage de la **Figure 24**

On remarque que les données ne sont plus toutes généralisées au même niveau. Par exemple, dans la colonne code postal, on trouve aussi bien des codes postaux que des généralisations de codes postaux.
La **Figure 25** fournit l'algorithme Median Mondrian tel que présenté dans (LeFevre, DeWitt, et Ramakrishnan 2006).

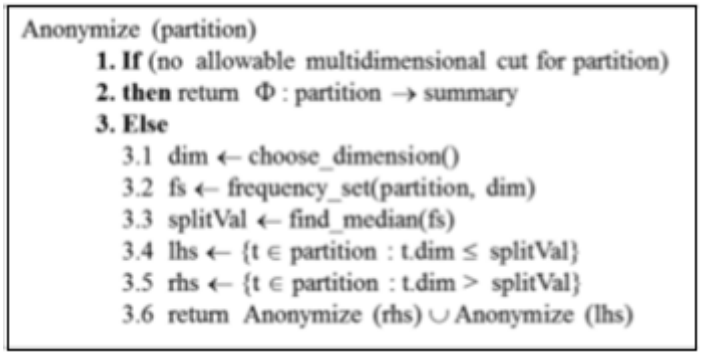


Figure 25. L'algorithme Median Mondrian

Comme le montre cette figure, l'algorithme est récursif. Lors de la première itération, "partition" contient tous les enregistrements de la table à anonymiser. La fonction "choose_dimension()" (respectivement ''frequency_set(partition, dim)'') renvoie la dimension choisie (respectivement l'ensemble «fs» des valeurs prises dans une dimension donnée "dim" au sein d'une partition donnée "partition"). La fonction find_median (fs) renvoie la valeur médiane "splitVal" de l'ensemble des valeurs «fs». "t.dim" est la valeur de la dimension "dim" pour un enregistrement "t". La fonction summary correspond à l'application de la généralisation à un ensemble de valeurs appartenant à la même partition. Elle est définie par un intervalle de valeurs dont la limite inférieure (resp. la limite supérieure) correspond à la valeur la plus petite (resp. la plus grande valeur) de la partition.

1.8. Les algorithmes « InfoGain Mondrian » et « LSD Mondrian »

Ces deux algorithmes étendent l'algorithme précédent (Median Mondrian) (LeFevre, DeWitt, et Ramakrishnan 2008) afin de préserver soit la classification pour InfoGain Mondrian, soit la régression pour LSD Mondrian (Least Square Deviance Mondrian). Pour ce faire, ils combinent le recodage multidimensionnel de Median

Mondrian avec des heuristiques de partitionnement orientées vers la classification pour Infogain Mondrian et vers la régression pour LSD Mondrian.

Intuitivement, il s'agit dans Infogain, de choisir, à chaque itération, le partitionnement qui minimise l'entropie pondérée de l'ensemble des partitions résultantes tout en préservant la contrainte d'anonymat. L'utilisation de cette métrique, selon les auteurs de cet algorithme, favorise l'obtention de partitions homogènes. La formule de calcul de l'entropie pondérée est la suivante :

$$Entropy(P,C) = \sum_{partitions\ P'} \frac{|P'|}{|P|} \sum_{c \in D_c} -p(c|P') \log p\ (c|P')$$

P est la partition courante, P' est l'ensemble des partitions résultantes pour les divisions candidates, P(c|P') est le pourcentage des enregistrements labellisés avec l'étiquette de la classe c.

Une exécution pas à pas de cet algorithme pour le **Tableau 24** est fournie en Annexe A de cette thèse.

LSD Mondrian, quant à lui, s'inspire de l'algorithme CART de construction d'un arbre de régression (Breiman et al. 1984). Par conséquent, à chaque itération, il choisit la division qui minimise la somme pondérée de MSE (Mean Squared Error) pour l'ensemble des partitions résultantes.

$$MSE(P') = \frac{1}{|P'|} \sum_{i \in P'} (r_i - \bar{r}\ (P'))2$$

$$Weighted\ MSE = \sum_{Partitions\ P'} \frac{|P'|}{|P|}\ (MSE(P'))$$

$$= \frac{1}{|P|} \sum_{Partitions\ P'} \sum_{i \in P'} (r_i - \bar{r}\ (P'))^2$$

Parce que |P| est constante pour toutes les divisions candidates, l'algorithme choisit la division qui minimise l'expression suivante :

$$Error^2(P,R) = \sum_{Partitions\ P'} \sum_{i \in P'} (r_i - \bar{r}\ (P'))^2$$

Où P est la partition courante, P' est l'ensemble des partitions résultantes pour les divisions candidates, i est un enregistrement, r_i est la valeur de l'attribut cible R de l'enregistrement i, r(P') est la moyenne des valeurs de l'attribut cible des enregistrements appartenant à P'.

De même, l'exécution pas à pas de cet algorithme, pour le **Tableau 24**, est fournie en Annexe A de cette thèse.

La **Figure 26** et la **Figure 27** fournissent respectivement les algorithmes Infogain Mondrian et LSD Mondrian tel que présentés dans (LeFevre, DeWitt, et Ramakrishnan 2008).

3.1 Single Target Classification Model

The Mondrian algorithm was recently proposed for k-anonymization using multidimensional recoding [17]. The algorithm is based on a greedy recursive partitioning of the (multidimensional) quasi-identifier domain space (see Figure 3). In order to obtain approximately uniform partition occupancy, [17] suggests recursively choosing the split attribute with the largest normalized range of values, and (for continuous or ordinal attributes) partitioning the data around the median value of the split attribute. This process is repeated until no *allowable* split remains, meaning that a particular region cannot be further divided without violating the anonymity constraint, or constraints imposed by value generalization hierarchies. We refer to this algorithm as **Median Mondrian**.

When the (set of) target mining model(s) is known, we can improve this heuristic. First consider a single target classification model, with predictor attributes $Q_1, ..., Q_d$ (also the quasi-identifier) and class label C. In this case, we propose a heuristic partitioning scheme based on information gain, which is reminiscent of decision tree construction. Intuitively, the goal of this greedy criterion is to produce homogeneous partitions of class labels.

At each recursive step, we choose the split that minimizes the weighted entropy over the set of resulting partitions (without violating the anonymity constraint). P denotes the current (recursive) tuple set, and *partitions* P' denotes the set of partitions resulting from the candidate split. $p(c|P')$ is the fraction of tuples in P' with class label $C = c$. We refer to this algorithm as **InfoGain Mondrian**.

$$Entropy(P, C) = \sum_{partitions\ P'} \frac{|P'|}{|P|} \sum_{c \in D_C} -p(c|P')logp(c|P') \quad (1)$$

InfoGain Mondrian handles continuous quasi-identifier values as they are typically handled by decision-trees, partitioning around the *threshold* value with smallest entropy (see [12]). The data is first sorted with respect to the split attribute. Then the data is scanned, and each time there is a change in class label, this *candidate threshold* is checked with respect to anonymity and entropy. In the event that no candidate threshold satisfies the anonymity constraint, the median is also checked as a default.

InfoGain Mondrian scales to large data sets through a straightforward adaptation of an existing scalable decision-tree induction scheme, such as RainForest [14].

Figure 26. L'algorithme InfoGain Mondrian

3.2 Single Target Regression Model

Similar greedy heuristics can be used when the target attribute is numeric. Specifically, we use the *mean squared error (MSE)* to measure the impurity of target attribute R within a candidate partition P'. A heuristic inspired by the CART algorithm for regression trees [7] recursively chooses the split that minimizes the weighted sum of MSEs over the set of resulting partitions. $\overline{r}(P')$ denotes the mean value of R in P'.

$$\begin{aligned} MSE(P') &= \frac{1}{|P'|}\sum_{i\in P'}(r_i-\overline{r}(P'))^2 \\ Weighted\ MSE &= \sum_{Partitions\ P'}\frac{|P'|}{|P|}(MSE(P')) \\ &= \frac{1}{|P|}\sum_{Partitions\ P'}\sum_{i\in P'}(r_i-\overline{r}(P'))^2 \end{aligned}$$

Because $|P|$ is constant for all candidate splits, the algorithm chooses the split that minimizes the following expression (without violating anonymity). We call this **Least Squared Deviance (LSD) Mondrian**. This algorithm handles continuous attributes through discretization.

$$Error^2(P,R) = \sum_{Partitions\ P'}\sum_{i\in P'}(r_i-\overline{r}(P'))^2 \qquad (2)$$

Figure 27. LSD Mondrian

1.9 Evaluation de la performance des algorithmes de généralisation

La plupart des articles décrivant les algorithmes d'anonymisation procèdent à des expérimentations menant à une évaluation de leurs performances. Cette évaluation est très souvent menée selon plusieurs axes. Le cas le plus récurrent est celui de la comparaison de l'algorithme avec un ou plusieurs autres algorithmes en se fondant sur une métrique de qualité (LeFevre, DeWitt, et Ramakrishnan 2006) (Kaur Arora, Bansal, et Sofat 2014). La sous-section qui suit présente quelques métriques d'évaluation de la qualité d'une anonymisation de micro-données par généralisation.

Un autre axe de comparaison récurrent est celui du temps d'exécution de l'algorithme, compte tenu d'un certain nombre de paramètres tels que la taille du QI, la valeur de k, la taille de la table originale, sa densité, etc. (Ayala-Rivera et al. 2014) (Babu et al. 2013) (Issa 2009).

D'autres évaluations d'algorithmes ciblent plutôt à exhiber leurs limites en termes de taille de la base par exemple. (Ayala-Rivera et al. 2014) relatent des expériences qui permettent de comparer les trois algorithmes Median Mondrian, Datafly et Incognito. Ces expériences ont montré, par exemple, que Median Mondrian et Datafly sont les algorithmes les plus rapides. Cependant, le temps d'exécution d'Incognito est dix fois supérieur au temps d'exécution des deux autres algorithmes.

2. Evaluation de la qualité d'une anonymisation de micro-données par généralisation

Comme mentionné en introduction de ce chapitre, en plus des métriques de guidage qui permettent d'orienter le codage des données, d'autres métriques existent. Il s'agit notamment de celles qui permettent d'évaluer la qualité des micro-données anonymes en la comparant à celle des données originales.

L'absence d'une mesure standard, qui serait largement acceptée par la communauté de chercheurs (Kiran et Kavya 2012), nous amène, dans cette section, à présenter les plus utilisées, sans toutefois pouvoir les lister toutes.

2.1. Métrique de complétude des données

Cette métrique trouve son utilité dans le cas où l'utilisation d'un algorithme peut générer des suppressions globales, donc une perte de données concernant certains individus. Elle mesure la complétude de la table anonyme relativement à la table originale par calcul du taux de suppressions effectuées. A titre d'exemple, si une table originale dispose de 100 tuples représentant autant d'individus et que l'anonymisation de cette table via un algorithme de généralisation a généré une table anonyme de 95 tuples, alors on pourra dire que l'anonymisation a permis de conserver 95% des données originales.

2.2 Métrique DM (Bayardo et Agrawal 2005)

Le k-anonymat, de par sa définition, oblige k individus à partager le même QI. Cette contrainte d'anonymat affecte négativement l'utilité des données. En effet, plus la taille de la classe d'équivalence est grande, plus l'utilité des données est amoindrie. Le rôle de la métrique DM (Discernability Metric) est d'informer l'éditeur des données sur la qualité des données résultant du degré de différenciation des individus. Ainsi, cette métrique affecte, à chaque tuple t, dans la table anonyme une pénalité déterminée par la taille de la classe d'équivalence représentant t et fait la somme des pénalités calculées.

Soit E l'ensemble des classes d'équivalence d'une table k-anonymisée. E_i est une des classes d'équivalence de E. La métrique DM peut être exprimée plus formellement de la façon suivante:

$$DM = \sum_{Ei} |Ei|^2$$

A titre d'exemple, considérons la table originale disposant de 10 classes d'équivalence (**Tableau 24**). Sur les 10 classes, 8 sont composées d'un seul tuple et 2 de 2 tuples ; ce qui donne une valeur de DM pour cette table de 16 ($=8*1^2+ 2*2^2$). Si aucun tuple ne partageait la même valeur du QI avec un autre, DM aurait été de 12. Si tous les tuples partageaient la même valeur de QI, DM vaudrait 144 (=12*12). Le degré de non différenciation entre les tuples de la table résultante de l'application de l'algorithme Datafly (**Tableau 36**) est égal à 31 ($=3*3^2+ 2^2$) donc plus élevé que celui de la table originale.

Sexe	Code Postal	Niveau d'étude
M	1305*	Secondaire
M	1305*	Secondaire
M	1305*	Secondaire
F	1305*	Université
F	1305*	Université
F	1306*	Université
F	1306*	Université
F	1306*	Université
M	1306*	Université
M	1306*	Université
M	1306*	Université

Tableau 36. Résultat de Datafly après suppression de tuple

2.3 Métrique CAVG (LeFevre, DeWitt, et Ramakrishnan 2006)(normalized average equivalence class size metric).

Dans une table k-anonyme, la taille des classes d'équivalence est supérieure ou égale à k. Par conséquent, la qualité des données est moindre si la taille de tout ou partie des classes d'équivalence dépasse de beaucoup la valeur k. Ainsi, procéder à une k-anonymisation tout en essayant d'approcher le cas idéal où chaque enregistrement de la table originale est généralisé dans une classe d'équivalence à k enregistrements minimiserait l'effet négatif de l'anonymisation sur la qualité des données. Pour mesurer cet effet sur la qualité des données, on peut appliquer la métrique C_{avg} (normalized average equivalence class size metric) sur la table anonyme. Sa formule de calcul est la suivante :

$$C_{AVG}(T') = (\frac{|T|}{|E|}) / k$$

Où T' est la table anonyme correspondante à la table originale T , k est le seuil de k-anonymat, |T| (respectivement |E|) est le nombre total d'enregistrements de T (respectivement le nombre total de classes d'équivalence).

Dans le cas idéal, si le nombre d'enregistrements de chaque classe d'équivalence d'une table anonyme T' est k, cette métrique prendrait la valeur 1 pour cette table.

Pour illustrer cette métrique, considérons le tableau 36 ci-dessus. Cette table 2-anonymisée a 4 classes d'équivalence et possède 12 enregistrements. Son score C_{AVG} est de (12/4)/2= 1,5.

2.4 La métrique de précision PREC

Partant des postulats selon lesquels plus la hiérarchie associée à la donnée est profonde, plus elle est généralisable et plus la donnée est généralisée, plus on perd en précision, Sweeney dans (Sweeney 2002a) a introduit une métrique afin de mesurer la perte d'information due à la profondeur de la hiérarchie de généralisation. Ainsi, pour une table anonymisée par généralisation, la perte en précision est la moyenne des pertes induites par la généralisation de tous les attributs du QI. Pour un attribut du QI, la perte en précision se calcule en appliquant le ratio entre le nombre d'étapes de généralisation appliquées et le nombre d'étapes de généralisations possibles (le nombre de niveaux dans la hiérarchie).

$$PREC\ (T') = 1 - \sum_{j=1}^{|QI|} \sum_{i=1}^{|T|} \frac{h}{|VGHA_j|} / (|T| * |QI|)$$

Où $|QI|$ est le nombre des attributs du QI, $|T|$ est le nombre total d'enregistrements de la table T, h représente la hauteur de la hiérarchie de généralisation de l'attribut Aj et de l'enregistrement i après la généralisation et $VGHA_j$ est la hauteur totale de la hiérarchie de généralisation de l'attribut A_j. Plus le score obtenu pour l'ensemble du QI est petit, moindre est la perte en précision pour la table anonyme.

2.5 La métrique GenILoss (Iyengar 2002) (Ayala-Rivera et al. 2014)

GenIloss (Generalized Information Loss) est fondée sur l'hypothèse selon laquelle si un attribut du QI prend ses valeurs dans une grande plage de valeurs, alors, après anonymisation, les valeurs sont moins précises que celles qui généraliseraient une petite plage. Ainsi le score de cette métrique pour une table originale serait à 1 pour exprimer la non-perte d'information. Une table anonyme, où les attributs du QI ont été généralisés au maximum aurait un score, pour GenILoss, de 0 pour exprimer la perte d'information maximale.

Pour permettre aussi la mise en œuvre de cette métrique sur des QI constitués d'attributs catégoriels, on associe des valeurs numériques aux valeurs de cette catégorie d'attributs. On peut ainsi, comme pour les attributs continus, borner l'ensemble des valeurs entre une limite inférieure et une limite supérieure. A titre d'exemple, si l'on considère l'attribut niveau d'étude, on pourrait attribuer la valeur numérique 1 à la valeur catégorielle 5ème, 2 à la valeur 3ème, 3 à la valeur seconde et 4 à la valeur terminale. Les limites inférieure et supérieure sont alors respectivement 1 et 4. Pour l'attribut âge qui est un attribut continu, les limites inférieure et supérieure sont respectivement 19 et 30.

La perte d'information globale d'une table anonyme T', calculée selon GenIloss, est notée GenIloss(T'). Elle peut être obtenue à l'aide de la formule suivante :

$$GenOLoss\ (T') = \frac{1}{|T| * |QI|} + \sum_{i-1}^{|QI|} \sum_{j=1}^{|T|} \frac{U_{ij} - L_{ij}}{U_i - L_i}$$

où T est la table originale, $|QI|$ est le nombre d'attributs du QI, |T| le nombre d'enregistrements dans T, Ui et Li sont respectivement les limites supérieure et inférieure de l'attribut Ai, Uij et Lij sont respectivement les limites supérieure et inférieure de la valeur de l'enregistrement j selon l'attribut Ai dans la table T'.

A titre d'exemple, considérons le **Tableau 37**. Pour cette table |T| = 5 et |QI| = 2. Selon les hiérarchies de généralisation qui se trouvent dans **Figure 4** et **Figure 5**, la limite inférieure et supérieure pour l'attribut âge du Qi sont respectivement 19 et 81. Si l'on adopte la normalisation ci-dessus pour l'attribut ville, les valeurs 1 et 4 constitueront respectivement la limite inférieure et supérieure de cet attribut. La valeur généralisée [19,23] de l'attribut âge a comme limites inférieure et supérieure respectivement les valeurs 19 et 28. La valeur généralisée Ile de France de l'attribut niveau d'étude a 3 et 4 comme limites inférieure et supérieure.

GenLoss (**Tableau 37**) = (1/ 5*2) * ((28-19/81-19) + (28-19/81-19) + (46-33/81-19) + (46-33/81-19) + (81-58/81-19) + (2-1/4-1) + (2-1/4-1) + (4-3/4-1) + (4-3/4-1) + (4-3/4-1)) = 0.1 * (0.14 +0.14+ 0.2 + 0.2+ 0.36 + 0.37 + 0.5 + 0.5+ 0.5+ 0.5+ 0.5) = 0.34

Age	Ville
[19,28]	Région PACA
[19,28]	Région PACA
[33,46]	Ile de France
[33,46]	Ile de France
[58,81]	Ile de France

Tableau 37. Exemple d'une table généralisée

2.6 La métrique de classification (Iyengar 2002)

La métrique de classification CM est définie comme la somme des pénalités des enregistrements, normée par le nombre total de lignes N. En effet, un enregistrement est pénalisé s'il est supprimé ou si son label classe(e) ne représente pas le label majoritaire de sa classe d'équivalence majorité(G(e)) comme le montrent les formules suivantes :

$$CM = \frac{\sum_{tous\ les\ enregistrements} pénalité\ (enregistrement\ e)}{N}$$

$$penalité\ (enregsitrement\ e) = \begin{cases} 1 \ \ si\ e\ est\ supprimé \\ 1\ si\ classe(e) \ \neq majorité\big(G(e)\big) \\ 0 \quad si\ non \end{cases}$$

A titre d'exemple, le **Tableau 38** contient 10 enregistrements et trois classes d'équivalence. Supposons qu'il existe deux classes pour l'attribut cible salaire, qui sont x et y (x si la valeur est inférieure ou égale à 2000 et y si la valeur est supérieure à 2000). Pour la première classe d'équivalence contenant les 4 premiers enregistrements, tous les enregistrements sont labellisés par la même classe x sauf le deuxième enregistrement qui a été supprimé. La classe majoritaire pour la deuxième classe d'équivalence qui contient les enregistrements numéros 4, 5 et 6 est y. L'enregistrement numéro 5 est labellisé par la classe y. Finalement, aucun enregistrement dans la troisième classe d'équivalence n'est pénalisé. CM (**Tableau 38**) = 2/10 = 0.2

Sexe	Code Postal	Niveau d'étude	Salaire
M	1305*	Secondaire	1200
M	1305*	Secondaire	1200
M	1305*	Secondaire	1300
F	1306*	Université	1600
F	1306*	Université	2000
F	1306*	Université	2100
M	1306*	Université	3000
M	1306*	Université	4000
M	1306*	Université	4500

Tableau 38. Exemple d'une table anonyme

3. Les outils d'anonymisation de micro-données par généralisation

Plusieurs outils d'anonymisation existent. Certains sont spécifiques aux données de test, d'autres ont pour but de fournir des données qui vont être analysées et explorées. Dans cette thèse, nous nous concentrons sur le deuxième type d'outil et spécifiquement sur ceux qui implémentent la technique de généralisation. Nous présentons dans cette section les six outils les plus connus en présentant un cas d'utilisation de chacun.

3.1 μ-argus

μ -argus (Anti-Re-identification General Utility System) a été développé par l'Agence Nationale des Statistiques des Pays-Bas dans le cadre du projet européen CASC (Computational Aspects of Statistical Confidentiality [5] puis étendu dans le second projet européen CENEX-SDC [4].

Parmi les techniques qu'il met en œuvre, on peut citer le recodage global, la suppression locale, la micro-agrégation, la permutation ainsi que les techniques de « Top Coding » et « Bottom Coding ».

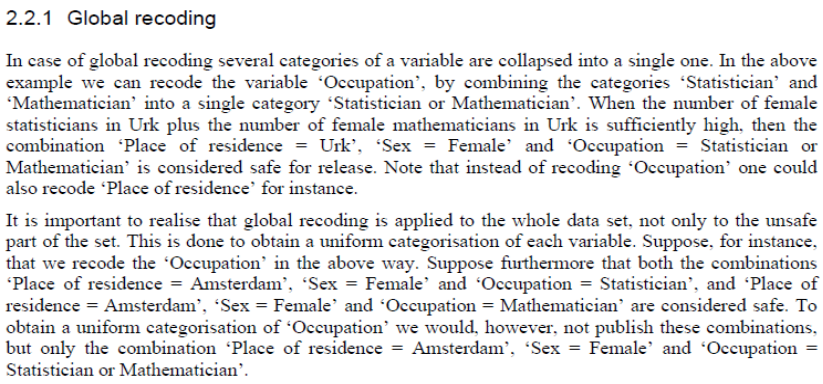
Une session d'utilisation de μ _argus est un enchaînement de tâches où :

1. l'utilisateur fournit les informations d'accès aux données originales (nom du fichier les contenant ainsi que le chemin d'accès à ce fichier) et procède à une première phase de qualification de ses données en saisissant pour chaque attribut, son type (catégoriel ou continu), son degré de sensibilité, son appartenance ou non au quasi-identifiant ainsi que le seuil de risque de ré-identification autorisé.
2. L'utilisateur poursuit la qualification de ses données en fournissant au système, pour un enregistrement, la probabilité maximale tolérée pour sa ré-identification. Le système évalue le taux de risque global pour tous les enregistrements des données originales, ce qui lui permet de rendre compte du nombre d'enregistrements à fort risque selon la probabilité maximale spécifiée par l'utilisateur.
3. L'utilisateur choisit une technique, spécifie les valeurs de ses paramètres en entrée et demande au système à ce qu'elle soit exécutée sur les données.

Les tâches 2 et 3 peuvent être répétées à chaque application d'une technique jusqu'à ce que l'utilisateur soit satisfait de l'anonymisation exécutée.

Afin d'aider l'éditeur des données dans la conduite de son processus d'anonymisation, un manuel d'utilisation [5] de l'outil sous le format pdf est fourni. Ce manuel contient :

- un rappel de quelques notions de base du domaine (perte d'information, risque de ré-identification, attributs identifiants, quasi-identifiants, etc.),
- une explication textuelle de chaque technique (à titre d'exemple, la **Figure 28** donne l'explication fournie pour la technique de recodage global),
- des références bibliographiques décrivant des algorithmes pour certaines techniques telles que la micro-agrégation,
- un organigramme décrivant brièvement les étapes du processus d'anonymisation,
- une description des fonctionnalités de l'outil.



2.2.1 Global recoding

In case of global recoding several categories of a variable are collapsed into a single one. In the above example we can recode the variable 'Occupation', by combining the categories 'Statistician' and 'Mathematician' into a single category 'Statistician or Mathematician'. When the number of female statisticians in Urk plus the number of female mathematicians in Urk is sufficiently high, then the combination 'Place of residence = Urk', 'Sex = Female' and 'Occupation = Statistician or Mathematician' is considered safe for release. Note that instead of recoding 'Occupation' one could also recode 'Place of residence' for instance.

It is important to realise that global recoding is applied to the whole data set, not only to the unsafe part of the set. This is done to obtain a uniform categorisation of each variable. Suppose, for instance, that we recode the 'Occupation' in the above way. Suppose furthermore that both the combinations 'Place of residence = Amsterdam', 'Sex = Female' and 'Occupation = Statistician', and 'Place of residence = Amsterdam', 'Sex = Female' and 'Occupation = Mathematician' are considered safe. To obtain a uniform categorisation of 'Occupation' we would, however, not publish these combinations, but only the combination 'Place of residence = Amsterdam', 'Sex = Female' and 'Occupation = Statistician or Mathematician'.

Figure 28. Description du recodage global dans le manuel μ-argus [5] page 12

A titre indicatif, dans ce manuel, il est recommandé, après l'évaluation du risque global, d'utiliser la suppression locale ou le recodage global. Cependant, précéder la suppression locale d'un recodage global est fortement conseillé.

3.2 L'outil CAT

Comme μ-argus, l'outil CAT (Cornell Anonymization Toolkit) est un logiciel libre (Open source). Il a été développé à l'université de Cornell aux Etats Unis (Xiao, Wang, et Gehrke 2009). Il met en œuvre les modèles de protection de la vie privée l-diversité (l-diversity) et t-proximité (t-closeness) via la technique de généralisation. Son processus d'anonymisation, tel que décrit dans le manuel d'utilisation [http://osdn.net/projects/sfnet_anony-toolkit/downloads/Documents/cat-mannual-1.0.PDF/] est un ensemble de phases, où :

1) L'utilisateur introduit, sous forme de fichiers textes, les données nécessaires pour l'anonymisation :

- le nombre d'attributs, le nombre d'enregistrements,

- la liste des attributs avec, pour chacun, son type, son domaine de valeurs, sa qualité : composant du quasi-identifiant ou sensible ou non sensible,
- le jeu de données,
- la hiérarchie de généralisation,

2) l'utilisateur choisit le modèle de protection de la vie privée (l-diversité ou t-proximité) et renseigne les paramètres associés. A titre d'exemple, dans le cas de la l-diversité, il aura à fournir la valeur du paramètre l ainsi que la connaissance que pourrait avoir l'adversaire sur certains attributs du QI,

3) le système exécute l'anonymisation par généralisation en tenant compte des paramètres saisis,

4) le système évalue, à la demande de l'utilisateur, le résultat de l'anonymisation d'un point de vue sécurité et utilité. Ainsi, il met à sa disposition, pour analyse, un histogramme de distribution du risque de ré-identification pour l'ensemble des tuples. Pour permettre à l'utilisateur d'évaluer l'utilité des données, le système affiche les deux graphes de densité associés respectivement aux deux tables de contingence, montrant les corrélations entre paires d'attributs avant et après anonymisation. Ces deux graphes permettent à l'utilisateur d'avoir une idée sur l'effet de l'anonymisation sur les données. En règle générale, plus les graphes sont similaires plus les données sont utiles ;

5) le système donne la possibilité à l'utilisateur de réitérer le processus soit en changeant les paramètres en entrée de l'anonymisation soit en procédant à des suppressions globales selon le seuil de risque toléré préalablement saisi (**Figure 29**).

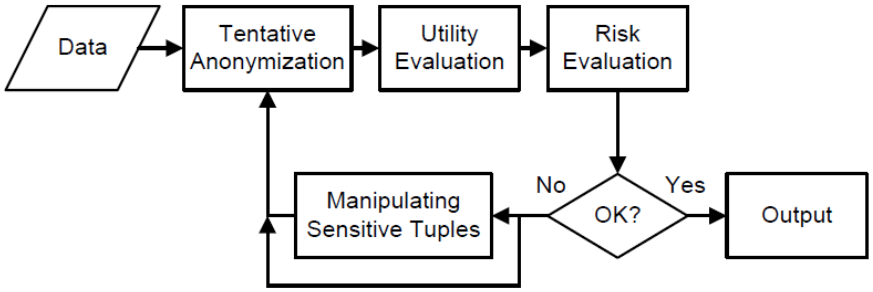


Figure 29. Processus d'anonymisation de CAT (Xiao, Wang, et Gehrke 2009)

Enfin, notons que dans le manuel d'utilisation de CAT, il est mentionné que la technique de généralisation est mise en œuvre dans l'outil à l'aide de l'algorithme de Samarati. Cependant, selon (Xiao, Wang, et Gehrke 2009), CAT utilise l'algorithme « Incognito ».

3.3 L'outil TIAMAT

Dans TIAMAT (Tool for Interactive Analysis of Microdata Anonymization Techniques) (Dai et al. 2009), deux algorithmes de généralisation du QI sont mis en œuvre : le Median Mondrian (LeFevre, DeWitt, et Ramakrishnan 2006) et l'algorithme k-Member (Dai et al. 2009)

Via son interface, TIAMAT aide l'utilisateur à définir le QI de sa table originale en calculant des statistiques sur les valeurs des attributs. Il lui permet aussi de comparer, moyennant son interface graphique, les résultats d'exécution de Median Mondrian et k-Member sur les données originales en évaluant la perte d'information. Pour ce faire, il lui propose de choisir une métrique de qualité parmi les deux dont il dispose.

TIAMAT est pour l'instant au stade de prototype. Il ne dispose pas de manuel d'utilisation, ni d'interface d'aide à la compréhension des deux processus qu'il supporte (processus de choix de QI et processus d'évaluation de la qualité d'une anonymisation).

3.4 L'outil SECRETA

L'outil SECRETA (System for Evaluation and Comparing RElational and Transaction Anonymization algorithms) (Poulis et al. 2014) est aussi un prototype. Il intègre neuf algorithmes dont quatre pour les tables relationnelles (Incognito, Cluster, Top-down and Full subtree bottom-up). Ces algorithmes implémentent la technique de généralisation. Les cinq autres algorithmes sont conçus pour des jeux de données de transactions. Ces derniers, souvent utilisés pour des études de marketing par exemple, regroupent des informations diverses sur le comportement ou les activités des personnes (telles que leurs habitudes d'achats par exemple).

SECRETA propose trois modules :

- Le premier module permet de sélectionner les données originales, de les mettre à jour (supprimer ou ajouter des données), de les décrire en indiquant les attributs de transaction et de visualiser les fréquences des valeurs de chaque attribut sous forme d'histogrammes.

- Le second module est un module de paramétrage, d'exécution et d'évaluation d'algorithmes. Il permet donc de construire une signature, de l'exécuter et de l'évaluer. Une signature dans SECRETA est une combinaison d'au plus deux algorithmes telle que chacun est appliqué sur un type d'attribut (de transaction ou autre) avec certaines valeurs des paramètres. L'évaluation est faite selon trois axes : la fréquence des nouvelles valeurs généralisées du QI, l'efficience (le temps d'exécution de chaque étape) et la perte d'information. Pour cette dernière, SECRETA met en œuvre la métrique d'utilité ARE (Average Relative Error) (J. Xu et al. 2006), dont l'objectif est de mesurer la qualité des données anonymisées. Ce module offre aussi la possibilité de faire varier les valeurs des paramètres des deux algorithmes afin de l'aider à prendre une décision sur la base d'une comparaison des évaluations produites. A titre d'exemple, l'utilisateur peut faire varier k entre les valeurs 20 et 30 avec un pas de 5. Dans ce cas, SECRETA évalue les algorithmes avec les trois valeurs : k=20, k=25, k=30.

- Le dernier module permet de choisir plusieurs signatures (donc plusieurs algorithmes par type d'attribut) et de les comparer selon les trois axes cités ci-avant.

Notons que SECRETA contient un tutoriel [6] composé d'une description de l'outil, de captures d'écrans commentées, d'une vidéo et d'un lien vers un article décrivant les algorithmes utilisés.

3.5 L'outil PARAT

Contrairement aux outils précédemment décrits, l'outil PARAT (Privacy Analytics Risk Assessment Tool) est un outil commercialisé. Il est distribué par la société Privacy Analytics Inc. [7]. Il est dédié à l'anonymisation de données structurées et non structurées. Pour les données structurées, qui font l'objet de notre présente étude, il implémente la technique de généralisation et la pseudonymisation. Comme pour tous les outils commercialisés, les algorithmes associés à cette technique ne sont pas dévoilés. Toujours pour les données structurées, PARAT [8], offre un certain nombre de fonctionnalités dont la mesure du risque de ré-identification que l'éditeur de données pourra utiliser dans le cas où il a la connaissance des destinataires de ses données. Il permet aussi de guider l'utilisateur dans son processus d'anonymisation via une interface graphique en forme de pipeline. Ce dernier est un schéma de processus composé de nœuds parallèles ou séquentiels, chacun représentant une opération. Un clic sur un nœud affiche une interface montrant comment exécuter l'opération associée. PARAT permet à l'utilisateur de vérifier s'il a accompli toutes les opérations qui constituent le processus d'anonymisation.

PARAT offre une fonctionnalité de configuration d'une anonymisation à l'aide de laquelle :

- on peut sélectionner et décrire les attributs à anonymiser (types d'attribut : text, dateTime, identifiant, quasi-identifiant).
- On peut visualiser le taux de risque pour un QI, le pourcentage d'enregistrements à risque, le taux de risque pour chaque attribut.
- On peut choisir, pour chaque QI, une technique à appliquer (soit une technique d'anonymisation ou une pseudonymisation).
- On peut visualiser le risque de ré-identification après chaque anonymisation, comme on peut afficher les données anonymes face aux données originales afin de les comparer.

3.6 ARX Data Anonymization Tool [9]

ARX est un logiciel « open source » pour l'anonymisation des données sensibles. Il utilise un unique algorithme baptisé 'flash' [9] qui permet de construire un treillis de généralisation (selon le même principe que l'algorithme Incognito ou Samarati). Il est composé de trois grandes phases précédées par l'importation des données originales et suivies par l'exportation des données anonymisées. Ces trois phases sont : la configuration, l'exploration et l'analyse (**Figure 30**).

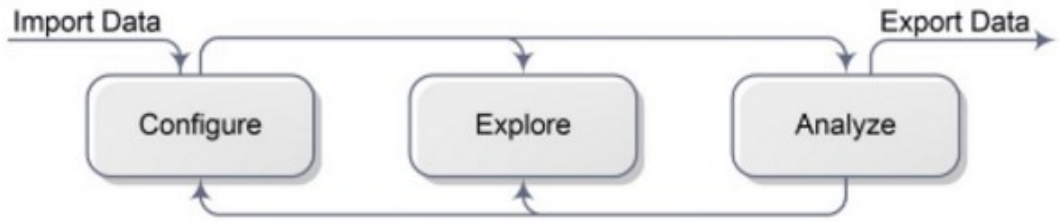


Figure 30. Processus d'anonymisation d'ARX

1. **La configuration** : Cette phase permet de spécifier plusieurs paramètres qui vont permettre par la suite d'appliquer l'anonymisation. En effet, l'utilisateur doit spécifier :
    - le type de chaque attribut : (identifiant, quasi-identifiant, sensible ou non sensible) et (entier ou chaine de caractère),
    - les hiérarchies de généralisation de chaque attribut quasi-identifiant,
    - le modèle de protection de la vie privée (k-anonymat, l-diversité, t-proximité et t-présence) et les valeurs de ses paramètres d'entrée,
    - le seuil de suppression,
    - la métrique d'utilité permettant de trier les résultats de l'anonymisation,
    - la fonction d'agrégation,
    - le poids de chaque attribut quasi-identifiant,
    - la préférence entre généralisation et suppression.

2. **L'exploration** : ARX permet d'afficher le treillis de généralisation construit par l'algorithme 'flash'. Chaque nœud de ce treillis représente une anonymisation possible. Il a une couleur ayant une signification : Si la couleur est verte, alors le nœud satisfait le modèle de protection de la vie privée déjà choisi ; si la couleur est rouge, alors le nœud ne satisfait pas le modèle ; enfin, si la couleur est orange, alors la solution est optimale selon la métrique d'utilité choisie lors de la phase de la configuration. La phase d'exploration permet aussi de trier les nœuds en affichant la valeur de la perte d'information associée, afin d'aider l'utilisateur à choisir une ou plusieurs solutions qui vont être analysées par la suite.
3. **L'analyse** : Cette phase permet à l'utilisateur d'évaluer un nœud choisi dans la phase précédente en termes de sécurité et d'utilité. Cette analyse permet de connaître les propriétés de la table anonymisée (le nombre de classes d'équivalence et la valeur de la métrique d'utilité) et de comparer les données originales par rapport aux données anonymisées selon les fréquences de chaque valeur de chaque attribut, la distribution des attributs et les tables de contingence. La phase d'analyse permet aussi d'évaluer le risque de chaque enregistrement, de chaque attribut quasi-identifiant et de la totalité des données.

Le manuel d'utilisation d'ARX se trouve sur son site officiel (http://arx.deidentifier.org/overview/, s.d.). Il décrit toutes les fonctionnalités de l'outil à l'aide de textes et de vidéos. Il décrit aussi l'algorithme flash sous forme de lignes Java comme le montre la **Figure 31**.

```
1  PriorityQueue pqueue = new PriorityQueue();
2  for (int i = 0; i < lattice.height; i++) {
3     for (Node node : sort(level[i])) {
4        if (!node.isTagged()) {
5           pqueue.add(node);
6           while (!pqueue.isEmpty()) {
7              Node head = pqueue.poll();
8              if (!head.isTagged()) {
9                 check(findPath(head), pqueue);
10             }
11          }
12       }
13    }
14 }
```

Figure 31. L'algorithme 'flash'

4. Synthèse sur les algorithmes de généralisation de micro-données

La technique de généralisation de micro-données fait partie d'un ensemble vaste de techniques d'anonymisation de ce type de données dont quelques-unes ont été présentées dans le chapitre précèdent. De nombreuses études comparatives de ces techniques existent (B. C. M. Fung et al. 2010) (Matthews et Harel 2011) Certaines d'entre elles sont orientées usage. Elles effectuent une analyse de ces techniques en mettant en évidence leurs avantages et inconvénients afin de proposer d'autres directions de recherche. D'autres études sont purement techniques et donc inaccessibles à des éditeurs ayant de faibles compétences dans le domaine de l'anonymisation.

Face à cette variété de techniques, nous nous sommes focalisés sur la technique de généralisation de micro-données dont la mise en œuvre est faite au travers de plusieurs algorithmes dont les plus cités dans la littérature ont été présentés dans la section 1 de ce chapitre. Dans la plupart des articles de recherche étudiés, ces algorithmes sont présentés d'une manière proche de la programmation. Ils sont le plus souvent partiellement instanciés via un exemple. Leurs principes fondamentaux sont succinctement décrits. Leur description est aussi éparpillée dans le lot non négligeable des publications les concernant. La plupart des travaux de recherche les décrivent dans un objectif d'exhiber leur efficacité. Ainsi, ils sont compréhensibles uniquement par des informaticiens ou des personnes ayant des compétences importantes en programmation.

Notre étude comparative issue de notre synthèse de l'état de l'art sur les neuf algorithmes de généralisation de micro-données a pour objectif, d'une part, (1) de réunir toute la connaissance actuellement intégrée dans la myriade de documents de recherche afin de la rendre disponible sous forme d'une base de connaissances,

accessible à des éditeurs de données et (2) d'autre part, de contribuer concrètement au choix d'algorithmes de généralisation.

Au vu de la littérature associée, notre étude comparative a été menée selon deux directions. La première direction a consisté à fournir une caractérisation fine des algorithmes. La seconde direction a consisté à réunir et à analyser des résultats d'expérimentation d'algorithmes par des spécialistes du domaine afin d'en extraire les critères influençant leur performance.

4.1 Caractérisation des algorithmes de généralisation

Notre étude comparative des algorithmes a fait ressortir à la fois des ressemblances et des différences entre ces algorithmes. Ces dernières peuvent concerner aussi bien les prérequis à leur exécution que leurs entrées, leurs processus et leurs sorties.

Pour ce qui est des pré-requis, nous avons pu constater que la plupart des algorithmes sont applicables à des quasi-identifiants constitués d'attributs continus et/ou catégoriels. Certains d'entre ceux, dont l'objectif est la préservation de l'utilité des données pour des besoins de classification ou de régression, tels que le LSD Mondrian et InfoGain Mondrian (Lefèvre, 2006b), exigent une corrélation entre plusieurs attributs cibles.

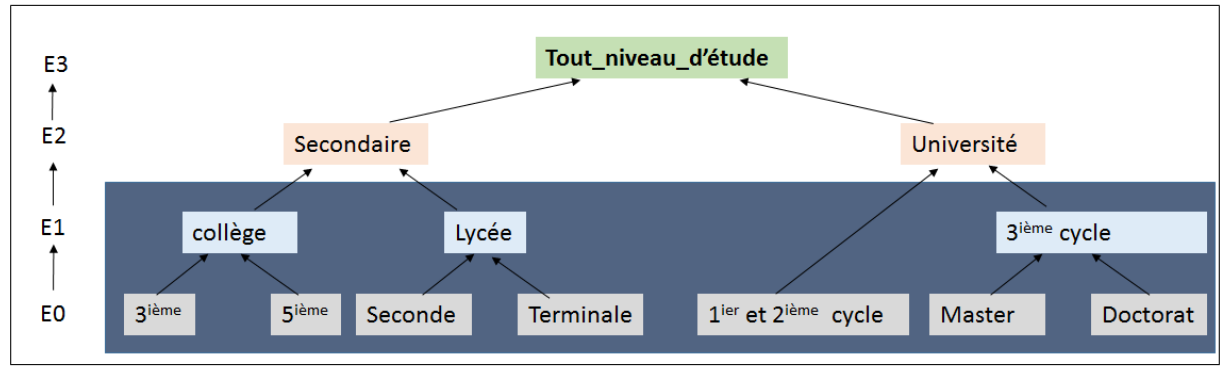
Tous les algorithmes de généralisation requièrent des paramètres d'entrée. Il faut notamment définir, pour chaque algorithme, au moins la valeur de k (propre au k-anonymat), les attributs qui constituent le QI et enfin, le cas échéant, les hiérarchies de généralisation. Notons que certains algorithmes, tels TDS, n'exigent pas de hiérarchie de généralisation pour les attributs continus. Ils proposent de la construire automatiquement. En outre, pour les algorithmes qui intègrent la suppression d'enregistrements, le nombre de suppressions autorisé (MaxSup) est également un paramètre d'entrée. Enfin, tous les algorithmes qui préservent la qualité des données relatives à un type spécifique d'analyse statistique, telle que la classification et la régression, exigent la déclaration d'au moins un attribut cible.

Tous les algorithmes étudiés, hormis μ_argus qui nécessite une interaction avec l'utilisateur à chaque étape de généralisation, effectuent un processus d'anonymisation automatique. La plupart de ces processus sont itératifs. Cependant ils peuvent se distinguer par le fait qu'ils sont guidés ou non par des métriques, fondés ou non sur des heuristiques. En outre, certains d'entre eux sont des processus ascendants dans le sens où ils construisent des petits groupes d'enregistrements qui sont ensuite fusionnés itérativement jusqu'à ce que chaque groupe préserve le k-anonymat (contienne au moins k enregistrements). D'autres processus sont plutôt descendants (« top down ») c'est-à-dire qu'ils commencent par un seul groupe contenant tous les enregistrements et divisent, itérativement, chaque groupe en deux sous-groupes, tout en préservant le k-anonymat.

Enfin, les algorithmes de généralisation ne fournissent pas le même type de résultat. Certains algorithmes fournissent une seule table anonymisée, tandis que d'autres fournissent plusieurs tables alternatives. Certains fournissent une solution optimale selon le k-anonymat (voir section 4.1 dans le chapitre 2), mais, selon (Benjamin, 2010), ils sont limités à un ensemble de données de petite taille. D'autres, fondés sur des heuristiques, ne garantissent pas l'optimalité.

De plus, les généralisations générées par ces algorithmes sont qualifiées, dans la littérature, soit de type domaine complet (« full domain generalization ») , soit de type sous-arbre (« subtree generalization ») ou encore de type multidimensionnelle (« multidimensional generalization »).

Une généralisation de type domaine complet est une généralisation qui aboutit à une table anonyme au sein de laquelle toutes les données originales d'un attribut constituant le quasi-identifiant ont été généralisées en valeurs appartenant à un même niveau de la hiérarchie de généralisation. Par exemple, dans la **Figure 32**, toutes les valeurs de l'attribut niveau d'étude sont généralisées au niveau 2 de la hiérarchie de généralisation (secondaire, université)



niveau d'étude
5ième
3ième
Seconde
Terminale
1ier et 2ième cycle
1ier et 2ième cycle
1ier et 2ième cycle
Master
Master
Doctorat
Doctorat
Doctorat

Données originales

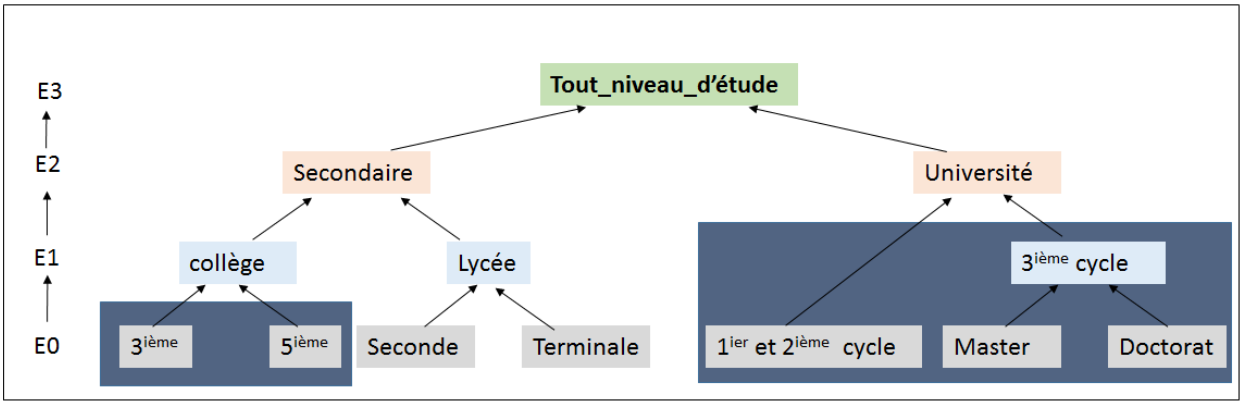
niveau d'étude
Secondaire
Secondaire
Secondaire
Secondaire
Université
Université
Université
Université
Université
Université
Université
Université

Données anonymisées

Figure 32.Exemple d'une généralisation de type domaine complet

La généralisation de type sous-arbre génère une table anonyme dans laquelles les données généralisées correspondent à des données originales partageant un même ancêtre dans la hiérarchie de généralisation. C'est une généralisation moins stricte que la généralisation de type « domaine complet » dans le sens où généralisation elle exige uniquement le partage d'un même ancêtre et non forcément le même niveau de généralisation pour toutes les données.

A titre d'exemple, la **Figure 33** montre que les valeurs 3ème et 5ème sont remplacées par leur ancêtre commun qui est collège. De même, les valeurs 1er et 2ème cycle, Master et Doctorat sont remplacées par la valeur Université.



niveau d'étude
5ième
3ième
Seconde
Terminale
1ier et 2ième cycle
1ier et 2ième cycle
1ier et 2ième cycle
Master
Master
Doctorat
Doctorat
Doctorat

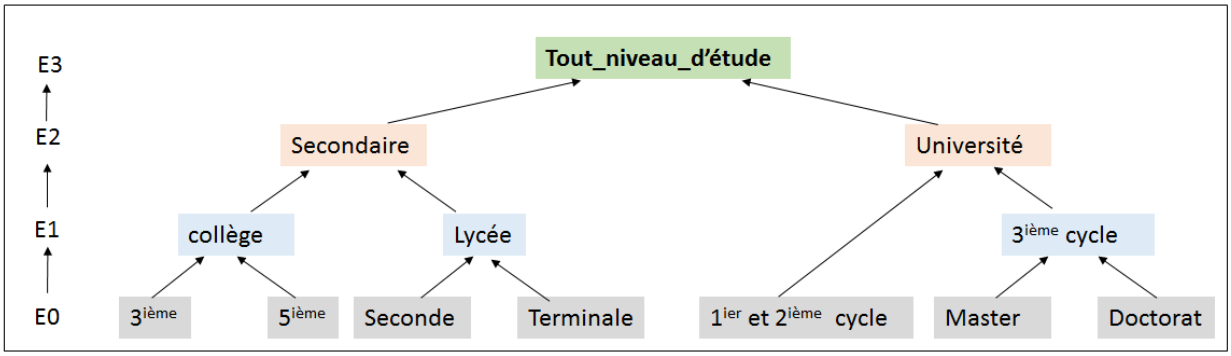
Données originales

niveau d'étude
Collège
Collège
Seconde
Terminale
Université
Université
Université
Université
Université
Université
Université
Université

Données anonymisées

Figure 33. Exemple d'une généralisation de type sous-arbre

Enfin, dans le cas d'une généralisation multidimensionnelle, deux données identiques dans la table originale peuvent être généralisées par des données différentes de la hiérarchie. A titre d'exemple, une généralisation multidimensionnelle de la table originale (**Tableau 24**) pourrait fournir la table suivante dans laquelle la donnée originale « 1er et 2ème cycle » a été tantôt reproduite telle qu'elle, tantôt généralisée à université.



niveau d'étude
5ième
3ième
Seconde
Seconde
1ier et 2ième cycle
1ier et 2ième cycle
1ier et 2ième cycle
Master
Master
Doctorat
Doctorat
Doctorat

Données originales

niveau d'étude
collège
collège
Seconde
Seconde
Université
Université
1ier et 2ième cycle
1ier et 2ième cycle
Université
Université
Doctorat
Doctorat

Données anonymisées

Figure 34. Exemple d'une généralisation de type multidimensionnelle

Une dernière constatation faite sur les résultats d'anonymisation par généralisation concerne leur usage. En effet, les données anonymes issues de certains algorithmes de généralisation sont produites pour des usages spécifiques. C'est le cas, par exemple, de TDS et Infogain Mondrian qui produisent des données pour des tâches de classification.

Le **Tableau 39** résume notre caractérisation des neuf algorithmes étudiés.

		Samarati	Incognito	Datafly	µ-argus	Bottom up	TDS	Media Mondrian	Info gain Mondrian	LSD Mondrian
Prérequis	**Limité au petite BD**	OUI		NON						
	Au moins deux attributs corrélés	Non Applicable							Vérifié	
Paramètre d'entrée	**k et les attributs QI**	OUI								
	Hiérarchie de généralisation	OUI						NON		
	Nombre maximal de suppression	OUI	Non Applicable	OUI	Non Applicable					
	Attributs cibles	Non Applicable				Un		Non Applicable	Au moins un	
Processus	**Degré d'automatisation**	Automatique			semi-automatique	Automatique				
	Guidé par métrique	NON		OUI				NON	OUI	
	Heuristique	NON						OUI		
	Ascendant/ Descendant	Ascendant					Descendant			
Sortie	**Multiplicité**	au moins une table		Une table de sortie						
	Qualité	k-anonymat optimal		k-anonymat pas nécessairement optimal						
	Type de généralisation	domaine complet				sous arbre		Multidimentionelle		
	scénario d'usage	tout scénario				classification		tout scénario	classification	régression

Tableau 39. Comparaison des algorithmes de généralisation

4.2 Etude des résultats d'expérimentation des algorithmes

Plusieurs articles décrivent des résultats d'expérimentation donnant lieu à des évaluations d'algorithmes en termes de qualité et de temps d'exécution (Ayala-Rivera et al. 2014) (B. C. Fung, Wang, et Yu 2005) (LeFevre, DeWitt, et Ramakrishnan 2008). Ces deux critères varient la plupart du temps selon un certain nombre de paramètres en entrée tels que la valeur de k, la taille du QI, la distribution de la base de données originale, etc. En ce qui concerne l'évaluation de la qualité, les auteurs la mesurent selon différents critères tels que la complétude ou la précision, en utilisant différents métriques pour un même critère.

A titre d'exemple, le **Tableau 40**, le **Tableau 41** et le **Tableau 42** synthétisent quelques évaluations recueillies dans la littérature pour les algorithmes Median Mondrian et Datafly.

On peut remarquer que la précision mesurée à l'aide de la métrique C_{AVG} pour l'algorithme Median Mondrian pour une même valeur de k et le même nombre d'attributs constituant le QI, est différente selon le type de distribution des données originales. Cette valeur vaut 100 pour des données de distribution uniforme et 5 pour des données de distribution dense (Ayala-Rivera et al. 2014) .

Critère d'évaluation	métrique	algorithme	input 'k'	nombre QI	distribution	Valeur
Précision	DM	dataFly	5	3	dense	**500 000 000**
Précision	DM	dataFly	10	3	dense	**500 000 000**
Précision	DM	dataFly	30	3	dense	**500 000 000**
Précision	DM	dataFly	50	3	dense	**500 000 000**
Précision	DM	dataFly	100	3	dense	**500 000 000**
Précision	DM	dataFly	200	3	dense	**500 000 000**
Précision	DM	dataFly	500	3	dense	**500 000 000**
Précision	DM	dataFly	5	3	uniforme	**50 000 000**
Précision	DM	dataFly	10	3	uniforme	**50 000 000**
Précision	DM	dataFly	30	3	uniforme	**50 000 000**
Précision	DM	dataFly	50	3	uniforme	**50 000 000**
Précision	DM	dataFly	100	3	uniforme	**100 000 000**
Précision	DM	dataFly	200	3	uniforme	**100 000 000**
Précision	DM	dataFly	500	3	uniforme	**100 000 000**
Précision	DM	Mondrian	5	3	dense	**50 000 000**
Précision	DM	Mondrian	10	3	dense	**50 000 000**
Précision	DM	Mondrian	30	3	dense	**50 000 000**
Précision	DM	Mondrian	50	3	dense	**50 000 000**
Précision	DM	Mondrian	100	3	dense	**50 000 000**
Précision	DM	Mondrian	200	3	dense	**50 000 000**
Précision	DM	Mondrian	500	3	dense	**50 000 000**
Précision	DM	Mondrian	5	3	uniforme	**1 000 000**
Précision	DM	Mondrian	10	3	uniforme	**1 000 000**
Précision	DM	Mondrian	30	3	uniforme	**1 000 000**
Précision	DM	Mondrian	50	3	uniforme	**5 000 000**
Précision	DM	Mondrian	100	3	uniforme	**5 000 000**
Précision	DM	Mondrian	200	3	uniforme	**10 000 000**
Précision	DM	Mondrian	500	3	uniforme	**100 000 000**

Tableau 40. Précision des algorithmes Median Mondrian et Datafly (métrique DM)

Critère d'évaluation	métrique	algorithme	input 'k'	nombre QI	distribution	Valeur
Précision	CAVG	dataFly	5	3	dense	**1 000**
Précision	CAVG	dataFly	10	3	dense	**500**
Précision	CAVG	dataFly	30	3	dense	**100**
Précision	CAVG	dataFly	50	3	dense	**500**
Précision	CAVG	dataFly	100	3	dense	**100**
Précision	CAVG	dataFly	200	3	dense	**50**
Précision	CAVG	dataFly	500	3	dense	**50**
Précision	CAVG	dataFly	5	3	uniforme	**50**
Précision	CAVG	dataFly	10	3	uniforme	**50**
Précision	CAVG	dataFly	30	3	uniforme	**50**
Précision	CAVG	dataFly	50	3	uniforme	**10**
Précision	CAVG	dataFly	100	3	uniforme	**50**
Précision	CAVG	dataFly	200	3	uniforme	**50**
Précision	CAVG	dataFly	500	3	uniforme	**10**
Précision	CAVG	Mondrian	5	3	dense	**100**
Précision	CAVG	Mondrian	10	3	dense	**100**
Précision	CAVG	Mondrian	30	3	dense	**50**
Précision	CAVG	Mondrian	50	3	dense	**10**
Précision	CAVG	Mondrian	100	3	dense	**10**
Précision	CAVG	Mondrian	200	3	dense	**5**
Précision	CAVG	Mondrian	500	3	dense	**5**
Précision	CAVG	Mondrian	5	3	uniforme	**5**
Précision	CAVG	Mondrian	10	3	uniforme	**5**
Précision	CAVG	Mondrian	30	3	uniforme	**5**
Précision	CAVG	Mondrian	50	3	uniforme	**5**
Précision	CAVG	Mondrian	100	3	uniforme	**5**
Précision	CAVG	Mondrian	200	3	uniforme	**5**
Précision	CAVG	Mondrian	500	3	uniforme	**5**

Tableau 41. Précision des algorithmes Median Mondrian et Datafly (métrique CAVG)

Critère d'évaluation	métrique	algorithme	input 'k'	nombre QI	distribution	Valeur
temps	Seconde	dataFly	5	3	dense	**40**
temps	Seconde	dataFly	10	3	dense	**40**
temps	Seconde	dataFly	30	3	dense	**40**
temps	Seconde	dataFly	50	3	dense	**40**
temps	Seconde	dataFly	100	3	dense	**40**
temps	Seconde	dataFly	200	3	dense	**40**
temps	Seconde	dataFly	500	3	dense	**40**
temps	Seconde	dataFly	5	3	uniforme	**100**
temps	Seconde	dataFly	10	3	uniforme	**100**
temps	Seconde	dataFly	30	3	uniforme	**100**
temps	Seconde	dataFly	50	3	uniforme	**100**
temps	Seconde	dataFly	100	3	uniforme	**100**
temps	Seconde	dataFly	200	3	uniforme	**100**
temps	Seconde	dataFly	500	3	uniforme	**100**
temps	Seconde	Mondrian	5	3	dense	**10**
temps	Seconde	Mondrian	10	3	dense	**10**
temps	Seconde	Mondrian	30	3	dense	**10**
temps	Seconde	Mondrian	50	3	dense	**10**
temps	Seconde	Mondrian	100	3	dense	**10**
temps	Seconde	Mondrian	200	3	dense	**10**
temps	Seconde	Mondrian	500	3	dense	**10**
temps	Seconde	Mondrian	5	3	uniforme	**200**
temps	Seconde	Mondrian	10	3	uniforme	**100**
temps	Seconde	Mondrian	30	3	uniforme	**100**
temps	Seconde	Mondrian	50	3	uniforme	**80**
temps	Seconde	Mondrian	100	3	uniforme	**50**
temps	Seconde	Mondrian	200	3	uniforme	**10**
temps	Seconde	Mondrian	500	3	uniforme	**8**

Tableau 42. Temps d'exécution des algorithmes Median Mondrian et Datafly

Notons que, certains articles, au vu des expérimentations et de la complexité de certains algorithmes, fournissent quelques recommandations quant à l'utilisation de ces algorithmes. A titre d'exemple, dans (Ayala-Rivera et al. 2014), les auteurs montrent que, bien que l'algorithme de Samarati soit plus rapide que l'algorithme Incognito, leur complexité est, dans les deux cas, de nature exponentielle en temps d'exécution et en capacité mémoire, en

fonction de la taille de la base de données originales. D'ailleurs, les auteurs de (B. C. M. Fung et al. 2010)recommandent de ne pas utiliser ces deux algorithmes sur de gros volumes de données sans pour autant définir leurs seuils d'inapplicabilité.

Notons aussi, qu'en plus de la sécurisation, de l'utilité et la complétude des données, l'anonymisation, utilisée dans un contexte autre que la publication, peut avoir à satisfaire d'autres besoins non fonctionnels. A titre d'exemple, une anonymisation en vue de l'externalisation des données à des fins de test doit sauvegarder les liens sémantiques entre les données comme les contraintes d'intégrité référentielle. De même, une anonymisation en vue d'une utilisation statistique doit préserver certaines propriétés statistiques des données originales. A titre d'exemple, le degré de corrélation et la distribution des données sont des propriétés statistiques que doivent préserver les techniques d'anonymisation du SDC.

En résumé, nous avons hiérarchisé à la **Figure 35** les objectifs d'une anonymisation de jeux de données, recensés au travers de l'état de l'art.

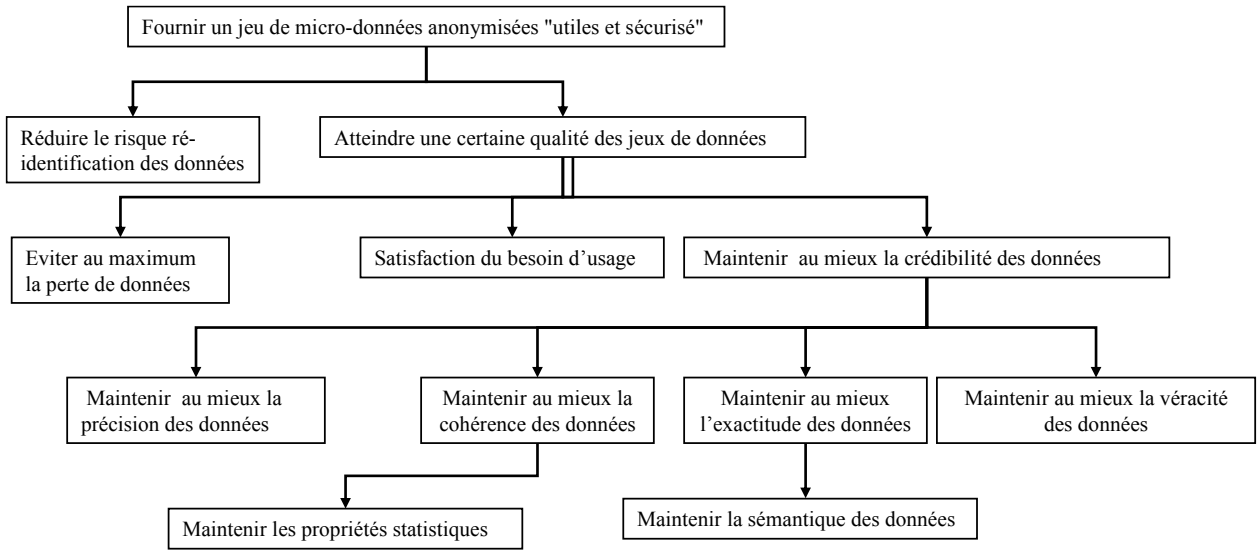


Figure 35. Hiérarchie des buts d'un processus d'anonymisation

Les algorithmes d'anonymisation par généralisation de micro-données sauvegardent de façon implicite la cohérence, l'exactitude et la véracité des données de par le principe de la généralisation qu'ils mettent en œuvre. Cependant, l'anonymisation par généralisation est, comme mentionné ci-avant, fortement liée à l'algorithme choisi.

La réduction du risque de ré-identification sous-entend l'identification préalable des risques et la définition du modèle de protection adéquat. Dans le cas où le k-anonymat a été choisi, les algorithmes de généralisation peuvent s'appliquer. Ceci signifie que le modèle de respect de la vie privée est une donnée préalable à toute anonymisation. Dans le cas où le k-anonymat a été choisi, les algorithmes de généralisation peuvent être évalués sur un jeu de micro-données selon les points de vue de sécurité, de précision et de complétude. Cette dernière pourrait, comme on a pu le voir au travers des différents algorithmes de généralisation, la plupart du temps, être contrôlée par l'éditeur de données qui aura la charge de fournir un plafond de suppression de données autorisé.

Le choix d'algorithmes répondant aux objectifs de sécurité et de qualité peut, bien sûr, être contraint par les performances offertes, telles que le temps d'exécution, et par le degré d'interactivité. Un éditeur de données dont les connaissances sont limitées n'aura sans doute pas l'envie d'interagir avec le processus lors de son exécution. La comparaison des algorithmes étant une tâche très difficile en soi du fait de la variabilité des métriques associées aux critères d'évaluation, il peut s'avérer intéressant de conserver les évaluations d'algorithmes, fournies dans les publications, comme base d'exemples pour un apprentissage supervisé sur les algorithmes d'anonymisation. Chaque exemple d'évaluation est ainsi constitué de paramètres d'évaluation ainsi que de l'évaluation proprement dite. Dans le cadre d'une k-anomymisation par généralisation de micro-données, l'ensemble des paramètres est composé de l'algorithme servant à l'anonymisation, du critère d'évaluation de l'algorithme, de la métrique d'évaluation associée au critère, de la valeur de k, du nombre d'attributs constituant le QI ainsi que de la distribution des données. Ainsi, les données d'évaluation peuvent être représentées selon un modèle multidimensionnel où les dimensions sont les paramètres (voir **Figure 36**).

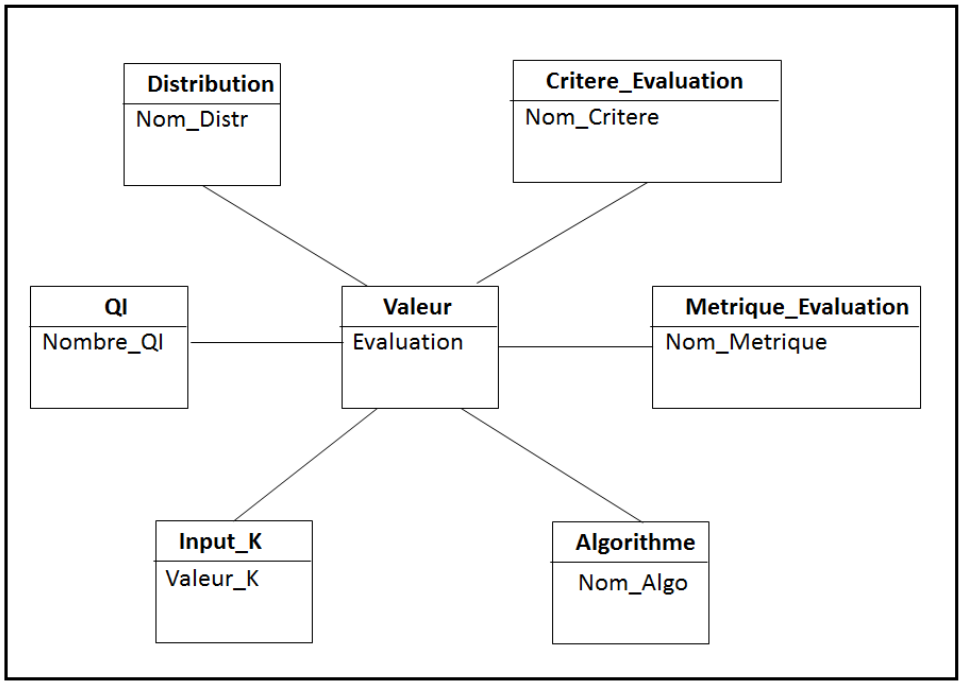


Figure 36. Le modèle d'évaluation

5. Synthèse sur les outils dédiés à la généralisation de micro-données

La complexité du processus d'anonymisation rend indispensable l'utilisation d'un outil intelligent permettant de guider un éditeur dans son processus de choix d'algorithmes.

Le concept de guidage dans les outils interactifs a été étudié par la communauté des systèmes d'information depuis des décennies. Dans (Morana et al. 2014), une typologie de ce concept, qui étend les précédentes, en

particulier celle de (Silver 2006), est proposée. Celle-ci a pour objectif de fournir un ensemble de caractéristiques permettant de décrire aussi bien le guidage décisionnel que le guidage explicatif.

Comme le montre le **Tableau 43**, cette typologie regroupe six descripteurs (nommés « catégories ») avec, pour chacun, un ensemble de valeurs permises (nommées « characteristics »).

Le guidage d'un outil peut faciliter le choix d'une fonctionnalité ou l'utilisation de cette fonctionnalité. C'est cette alternative que distingue le premier descripteur.

Le second descripteur permet de qualifier le guidage de suggestif, semi-suggestif ou encore d'informatif. Le guidage informatif fournit des explications pertinentes qui clarifient le choix de l'utilisateur sans pour autant lui suggérer comment agir. A l'opposé, le guidage suggestif fait des recommandations explicites à l'utilisateur sur la façon de construire son appréciation. Le guidage quasi-suggestif ne donne pas de façon explicite des recommandations. Cependant, il aide à inférer des recommandations. A titre indicatif, (Parikh, Fazlollahi, et Verma 2001) a constaté que le guidage suggestif est plus efficace dans l'amélioration de la qualité de la décision. Il augmente la satisfaction des utilisateurs et réduit le temps de décision. Le guidage informatif, quant à lui, est plus efficace pour aider l'utilisateur dans son apprentissage de l'outil.

Le mode de guidage peut être prédéfini, dynamique ou participatif. Le guidage prédéfini est statique dans le sens où il est préparé par le concepteur de l'outil. Le guidage dynamique génère un guidage à la volée. Ces deux modes de guidage ne requièrent pas des entrées de l'utilisateur. A l'inverse, le guidage participatif est sollicité par l'utilisateur à chaque information demandée.

De plus les guidages peuvent être distingués selon leur style d'invocation : invoqué par l'utilisateur, non invoqué ou intelligent. Les deux premiers styles dépendent de la conception de l'interface du système. Le dernier type évoque un guidage fourni en fonction du comportement de l'utilisateur observé par le système.

Le sixième descripteur qualifie le guidage selon le moment de sa délivrance. Il distingue ainsi le guidage prospectif qui est fourni avant une activité du guidage simultané qui est dispensé au cours d'une activité ou du guidage rétrospectif qui est livré après une activité. (Shen et al. 2012) a montré que la performance des décideurs augmente avec le guidage prospectif.

L'avant-dernier descripteur représente le format de la présentation du guidage : texte, image, animation (y compris les vidéos) et audio. Ces caractéristiques ne sont pas mutuellement exclusives et peuvent être combinées.

Categories	Characteristics			
target	Choosing functionnal capabilities		using functional capabilities	
Directivity	Suggestif		informatif	
Mode	Predifined	dynamic	participative	
Invocation	Automatic	user invoked	intelligent	
timing	Prospective	concurrent	retrospective	
format	Text	image	animation	audio
intention	Clarification	knowledge	learning	recommending
Audience	Novices		experts	

Tableau 43. Une typologie du guidage

Les systèmes de guidage sont destinés à être utilisés pour la prise de décision ou pour l'explication. Ainsi, quatre intentions sont possibles lors de leur délivrance : la clarification d'une anomalie perçue, la fourniture de connaissances supplémentaires, la facilitation de l'apprentissage et la recommandation. Généralement, les experts utilisent le guidage dans l'intention de résoudre des anomalies. Les novices l'utiliseront pour l'apprentissage.

Pour nous permettre d'évaluer de façon précise le guidage offert par les outils étudiés dans notre état de l'art, nous avons procédé à une analyse de ces outils afin d'identifier les tâches associées à leur processus d'anonymisation et de les qualifier, le cas échéant, le guidage selon la typologie présentée ci-avant.

Notre étude de l'état de l'art nous a permis de constater, le cas échéant en menant un processus de rétro-conception, que tous les outils suivent les mêmes étapes du processus d'anonymisation. Ce dernier est décrit à l'aide d'un diagramme d'activité UML à la **Figure 37**. Comme le montre cette figure, les trois premières étapes offrent à l'utilisateur la possibilité de successivement charger le jeu de micro-données, spécifier le type de chaque attribut (identifiant direct, quasi-identifiant ou sensible) et de fournir le seuil de risque de ré-identification autorisé.

Les étapes suivantes participent au choix d'algorithme ou de technique, au paramétrage de ces algorithmes ou techniques ainsi qu'à leur évaluation. Ces étapes peuvent se répéter jusqu'à la satisfaction de l'utilisateur. Certains outils offrent la possibilité de définir plusieurs signatures par algorithme ou technique, de comparer leur évaluation, de choisir puis d'itérer.

Lors de l'étape G5, l'utilisateur doit analyser les valeurs résultant des évaluations des bases de données anonymisées selon chaque critère. A titre d'exemple, il doit analyser le nombre de secondes pour le critère 'temps d'exécution' et les mesures de risque pour le critère 'risque de ré-identification' pour chaque base de données anonymisée. Puis, il doit les comparer pour choisir la meilleure selon ses préférences et ses besoins. Finalement, il exporte la base de données anonymisée qui satisfait au mieux ses préférences et ses besoins.

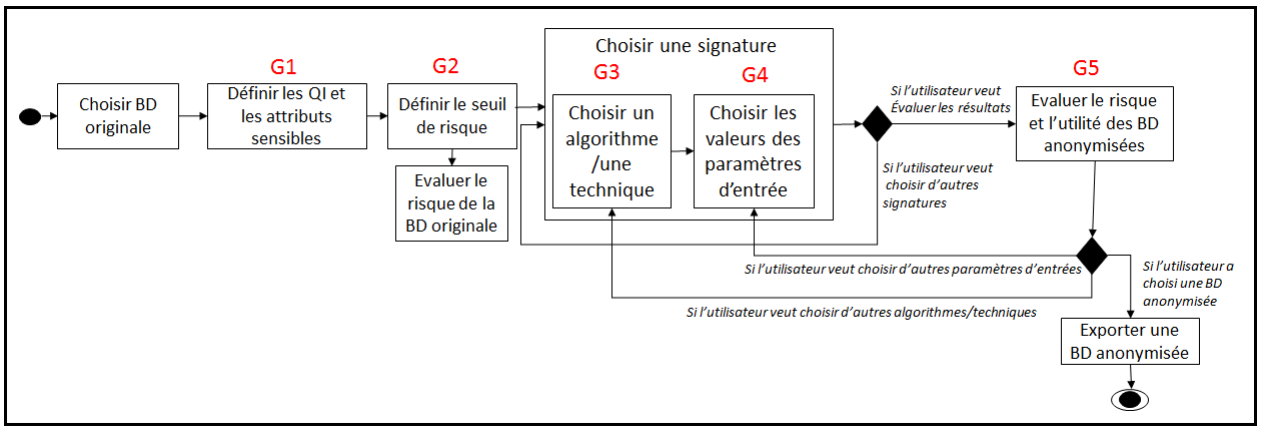


Figure 37. Les étapes du processus d'anonymisation

Outil	Type de guidage					
Mu_argus tool	**Etape du Processus:**	**G5: Evaluer l'utilité Et le risque**				
	target :	using functional				
	audience :	expert				
	directivity:	infomormatif				
	mode:	participatif				
	invocation:	user invoked				
	timing:	retrospectif				
	format:	image				
CAT	**Etape du Processus:**	**G5: Evaluer l'utilité Et le risque**				
	target :	using functional				
	audience:	expert				
	directivity:	infomormatif				
	mode:	participatif				
	invocation:	user invoked				
	timing:	retrospectif				
	format:	image				
TIAMAT	**Etape du Processus:**	**G1: Choisir les Attributs de QI**	**Etape du Processus:**	**G5: Evaluer l'utilité Et le risque**		
	target :	using functionnal	**target :**	using functional		
	audience:	Novices	**audience:**	expert		
	directivity:	suggestif	**directivity:**	infomormatif		
	mode:	dynamic	**mode:**	participatif		
	invocation:	user invoked	**invocation:**	user invoked		
	timing:	prospective	**timing:**	retrospectif		
	format:	texte	**format:**	image		
Secreta	**Etape du Processus:**	**G5: Evaluer l'utilité Et le risque**				
	target :	using functional				
	audience:	expert				
	directivity:	infomormatif				
	mode:	participatif				
	invocation:	user invoked				
	timing:	retrospectif				
	format:	image				
PARAT	**Etape du Processus:**	**G2: Définir le seuil Du risque**	**Etape du Processus:**	**G5: Evaluer l'utilité Et le risque**	**Etape du Processus:**	**G6: Mener le processus D'aninymisation**
	target :	novices	**target :**	using functional	**target :**	choosing functional
	audience:	novices	**audience:**	expert	**audience:**	novices and expert
	directivity:	suggestif	**directivity:**	infomormatif	**directivity:**	suggetsif
	mode:	participative	**mode:**	participatif	**mode:**	Predifined
	invocation:	user invoked	**invocation:**	user invoked	**invocation:**	automatic
	timing:	prospective	**timing:**	retrospectif	**timing:**	concurent
	format:	texte	**format:**	image	**format:**	animattion

Tableau 44. Types de guidage dans les outils d'anonymisation

L'étude de ces différentes tâches nous a permis de constater que :

- Chaque outil propose un tutoriel décrivant les connaissances du domaine et parfois certains conseils pour mener à bien son processus d'anonymisation. Cependant, ces descriptions ne sont pas, la plupart du temps, explicites ou faciles à comprendre.
- Aucun guidage dans le choix des algorithmes ou des techniques n'est offert par les outils étudiés.
- Aucun guidage dans le paramétrage des algorithmes ou des techniques n'est proposé.
- L'outil PARAT est le seul outil à offrir une interface interactive permettant à l'utilisateur de suivre les étapes du processus d'anonymisation.
- L'outil TIAMAT est le seul à offrir un guidage pour la détermination des constituants d'un QI.
- L'outil PARAT est le seul à offrir un guidage dans le choix du seuil de risque toléré. Il propose un taux selon un contexte fourni par l'utilisateur.
- Tous les outils offrent un guidage semblable pour l'évaluation des signatures (**Tableau 44**). En effet, ils proposent un guidage informatif afin d'aider l'utilisateur à choisir la signature qui correspond le plus à ses besoins et ses préférences.

6. Conclusion

Ce chapitre a présenté les algorithmes de généralisation les plus connus et quelques métriques d'évaluation proposées dans la littérature. Nous avons décrit et comparé les outils d'anonymisation existants nous fondant notamment sur le type de guidage proposé. Ces comparaisons nous ont permis de dégager les limites de ces outils et des approches existantes. En effet, même si certains outils offre un guidage au moment des évaluations des bases de données anonymisées, ils ne sont pas accessibles à des éditeurs de données avec de faibles compétences dans le domaine d'anonymisation. De plus, les tutoriaux proposés par les outils, contiennent des descriptions (ou des références à des travaux de recherche) des techniques et/ou des algorithmes qui ne sont pas, la plupart du temps, explicites ou faciles à comprendre notamment par des personnes qui n'ont pas des compétences en programmation. Le chapitre suivant propose des descriptions simplifiées qui rend ces techniques et ces algorithmes plus intelligibles

Chapitre 4 Notre proposition de guidage informatif pour la technique de généralisation et ses algorithmes

L'anonymisation est une tâche complexe car sujette à deux risques : le risque de divulgation de données sensibles à des personnes non autorisées et le risque de livraison de données à faible valeur dans le cas où le processus de masquage réduit trop leur utilité. Une anonymisation maîtrisée nécessite avant tout une compréhension des techniques et algorithmes permettant de la réaliser. La généralisation est une technique assez simple à appréhender. Cependant, les algorithmes permettant de la mettre en œuvre sont, de par leur variété, leur complexité ainsi que leur présentation dans la littérature, difficiles à comprendre. Pourtant ils sont utilisés par des personnes qui peuvent ne posséder que peu de connaissances dans le domaine de l'anonymisation et/ou dont les compétences en programmation sont limitées.

Dans ce chapitre nous proposons une description simplifiée et une typologie plus fine des algorithmes d'anonymisation par généralisation étudiés dans le chapitre 3.

Ces deux types d'artefacts sont obtenus en suivant une approche d'abstraction que nous explicitons.

Ainsi, ce chapitre est structuré en deux sections. La première détaille le processus d'abstraction que nous avons mis en œuvre et fournit ses différents résultats. La seconde section décrit l'expérimentation menée pour valider les résultats de notre approche d'abstraction.

1. Intérêt de l'abstraction des algorithmes d'anonymisation

Selon Miller (Miller 1967), lors d'un traitement cognitif, la mémoire temporaire de l'humain est limitée en nombre d'items qu'il peut appréhender simultanément. Cet état de fait a été soutenu par plusieurs chercheurs dont Návrat et al. dans (Návrat et Filkorn 2005). Ces derniers affirment qu'un développeur ne peut traiter plus de quelques concepts et leurs liens simultanément. En réponse à cette limitation cognitive humaine conjuguée à la complexité des logiciels, l'abstraction, initialement utilisée dans d'autres disciplines, a fait l'objet de beaucoup de travaux de recherche en ingénierie des logiciels.

Kramer et Hazzen définissent l'abstraction de la façon suivante :

> *« Abstraction is a cognitive means by which engineers, mathematicians and others deal with complexity. It covers both aspects of removing detail as well as the identification of generalisations or common features »* (Kramer et Hazzan 2006).

Comme le souligne Kramer dans (Kramer 2007) et Tsui et al. dans (Tsui et al. 2011), le processus d'élimination de détail mentionné dans cette définition est réalisé dans un objectif de simplification et de capture de l'attention, en ignorant des éléments pour ne considérer que certaines propriétés d'objets complexes. Le processus de généralisation consiste en l'explicitation des caractéristiques communes à partir d'exemples spécifiques. A titre

d'exemple, la **Figure 38** est une abstraction d'un oiseau. On n'y représente pas tous les détails de l'oiseau. On fait abstraction de ce qui n'est pas important. On y représente les éléments communs à tous les oiseaux (bec, pattes, etc.) généralisant ainsi le concept d'oiseau.

Figure 38. Abstraction d'un oiseau

Selon (Tsui et al. 2011), l'une des raisons fondamentales pour s'engager dans une tâche d'abstraction dans l'analyse, la conception et le développement de logiciels est de réduire la complexité à un certain niveau afin que les aspects «pertinents» des exigences, de la conception et du développement puissent être facilement articulés et compris. Wagner et al. dans (Wagner et Deissenboeck 2008) confirment aussi l'apport de l'abstraction dans la réduction de la complexité des logiciels et ajoutent, comme autre avantage, celui de l'aide à sa compréhension. De plus, comme le soulignent Kramer et al. (Kramer et Hazzan 2006), la simplification attendue d'un processus d'abstraction, qui consisterait à garder ce qui est pertinent après l'avoir distingué de ce qui ne l'est pas, dépend de l'objectif attendu de cette abstraction.

Appliquée à notre contexte, l'abstraction est un outil cognitif qui permet de maîtriser la complexité des algorithmes de généralisation. Nous projetons de décrire les différentes étapes associées aux algorithmes de généralisation aussi simplement que possible afin de faciliter leur compréhension, leur appropriation et leur adoption. Aussi, nous souhaitons exhiber leurs exigences afin de les définir comme méta-données. Une fois l'abstraction obtenue, elle pourra servir comme outil de guidage informatif dans un processus d'anonymisation automatisé.

2. Notre processus d'abstraction des algorithmes d'anonymisation

La mise en œuvre d'un processus d'abstraction introduit inévitablement des niveaux. Comme le souligne Wing (Wing 2008), en informatique, on manipule simultanément au moins deux niveaux d'abstraction (le plus souvent plus de deux) : le niveau d'intérêt et le niveau en dessous ou encore le niveau d'intérêt et le niveau au-dessus. Dans le cadre de notre problématique, à partir d'un algorithme (niveau en dessous), nous projetons, via la tâche d'abstraction, de déduire le niveau d'intérêt qui correspond à son abstraction.

Il existe plusieurs façons de réaliser une abstraction. Pour déduire ce niveau d'intérêt, notre proposition est de mener un processus d'abstraction par paramétrage. Ce processus est défini dans (Návrat et Filkorn 2005) et (Abbott et Sun 2008) de la façon suivante :

> "*abstraction by parameterization extracts an essential core of some computational elements and reifies them as a named element of its own, leaving parameters to be filled in when the abstraction is instantiated".*

Comme le mentionne Liskov (Liskov et Guttag 2000), dans ce type de processus, l'abstraction est définie en termes de paramètres formels. L'identité des données réelles n'est pas pertinente, mais la présence, le nombre et le type des données sont pertinents. Ainsi, le paramétrage généralise les abstractions, les rendant utiles dans plus d'une situation.

Enfin, afin de parvenir à une compréhension globale de ces algorithmes, notre seconde proposition est de mener, à la suite du processus de paramétrage, un processus inductif, également appelé généralisation dans (Návrat et Filkorn 2005). Ce processus permettra une présentation de tous les algorithmes d'une technique à l'aide d'une description commune.

Les deux processus d'abstraction sont présentés dans les deux sous-sections ci-après. Nous avons appliqué ces deux processus à nos neuf algorithmes d'anonymisation par généralisation. Les deux sections qui suivent présentent cette application ainsi que les abstractions obtenues.

2.1 Nos abstractions par paramétrage des algorithmes de généralisation

Comme on peut le constater dans le chapitre 3 de cette thèse, la présentation des neuf algorithmes de généralisation dans les articles où ils apparaissent est plutôt proche de la programmation. La plupart du temps, ils sont partiellement expliqués au travers d'une instanciation d'un jeu de données. Leurs principes fondamentaux sont décrits textuellement. Ces descriptions sont, à notre avis, dédiées à des professionnels ayant de bonnes compétences en programmation.

C'est pourquoi nous avons jugé nécessaire, pour chacun de ces algorithmes, de fournir aux éditeurs de données une autre représentation qui soit suffisamment intelligible pour des personnes sans compétences en programmation. Pour obtenir une description intelligible, nous avons procédé à une abstraction par paramétrage au cours de laquelle nous avons éliminé toutes les informations non pertinentes et parfois ajouté de l'information afin de faciliter leur compréhension.

Dans la mesure où un algorithme est un artefact dynamique, nous avons choisi de le présenter via un diagramme épousant la notation BPMN (Business Process Modeling Notation). Cette dernière est bien connue et relativement facile à lire. De plus, elle est très utile pour la compréhension de la logique des problèmes complexes.

Ce processus d'abstraction par paramétrage, appliqué aux neuf algorithmes de généralisation, a donné les neuf abstractions présentées de la **Figure 39** à la **Figure 47**.

Par exemple, le processus de l'abstraction de l'algorithme Datafly utilise des phrases simples et compréhensibles (**Figure 39**). En effet Datafly généralise un attribut à la fois. A chaque itération, il sélectionne (comme mentionné dans l'étape 2) celui qui a le meilleur score (le plus grand nombre de valeurs distinctes dans la table courante). L'exécution s'arrête lorsque le k-anonymat est satisfait.

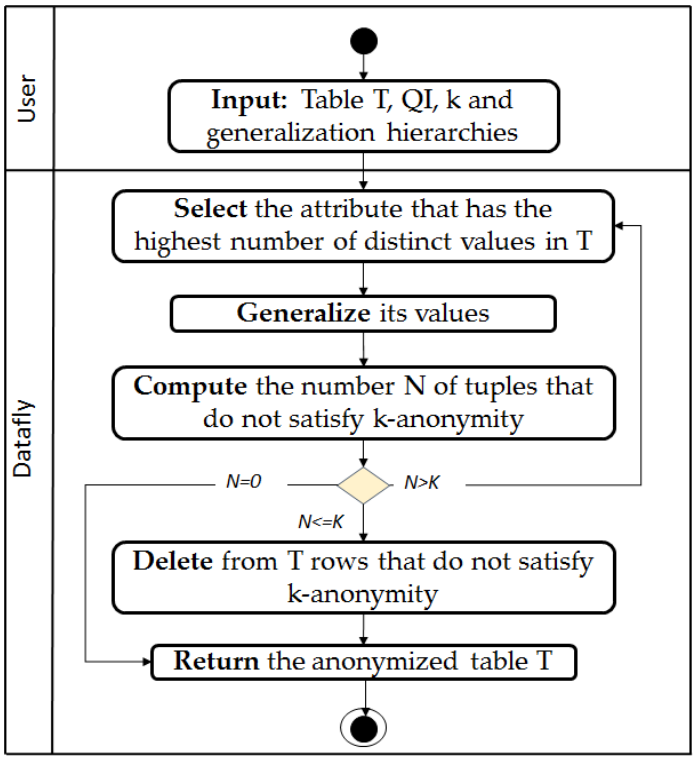


Figure 39. Abstraction de l'algorithme Datafly

Le résultat de l'abstraction de µ-argus est représenté dans la **Figure 40**. Dans µ_argus, à chaque itération, l'utilisateur choisit un attribut du QI (étape 2) que le système doit généraliser (étape 3). Ensuite, le système doit calculer et sélectionner les enregistrements qui ne satisfont pas le k-anonymat (étape 4). Puis l'utilisateur a deux choix : soit il continue la généralisation en sélectionnant un attribut, soit il s'arrête. Dans ce dernier cas, le système doit appliquer la suppression locale et fournir la table anonymisée.

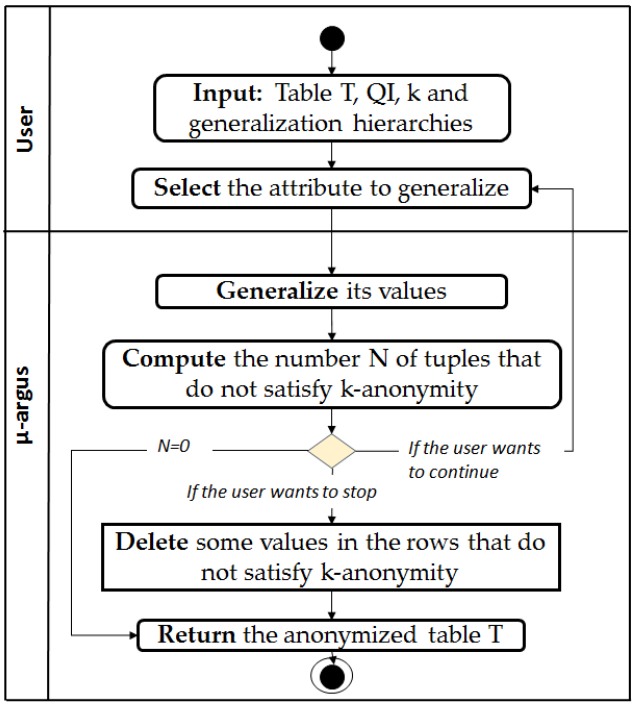


Figure 40. Abstraction de l'algorithme µ-argus

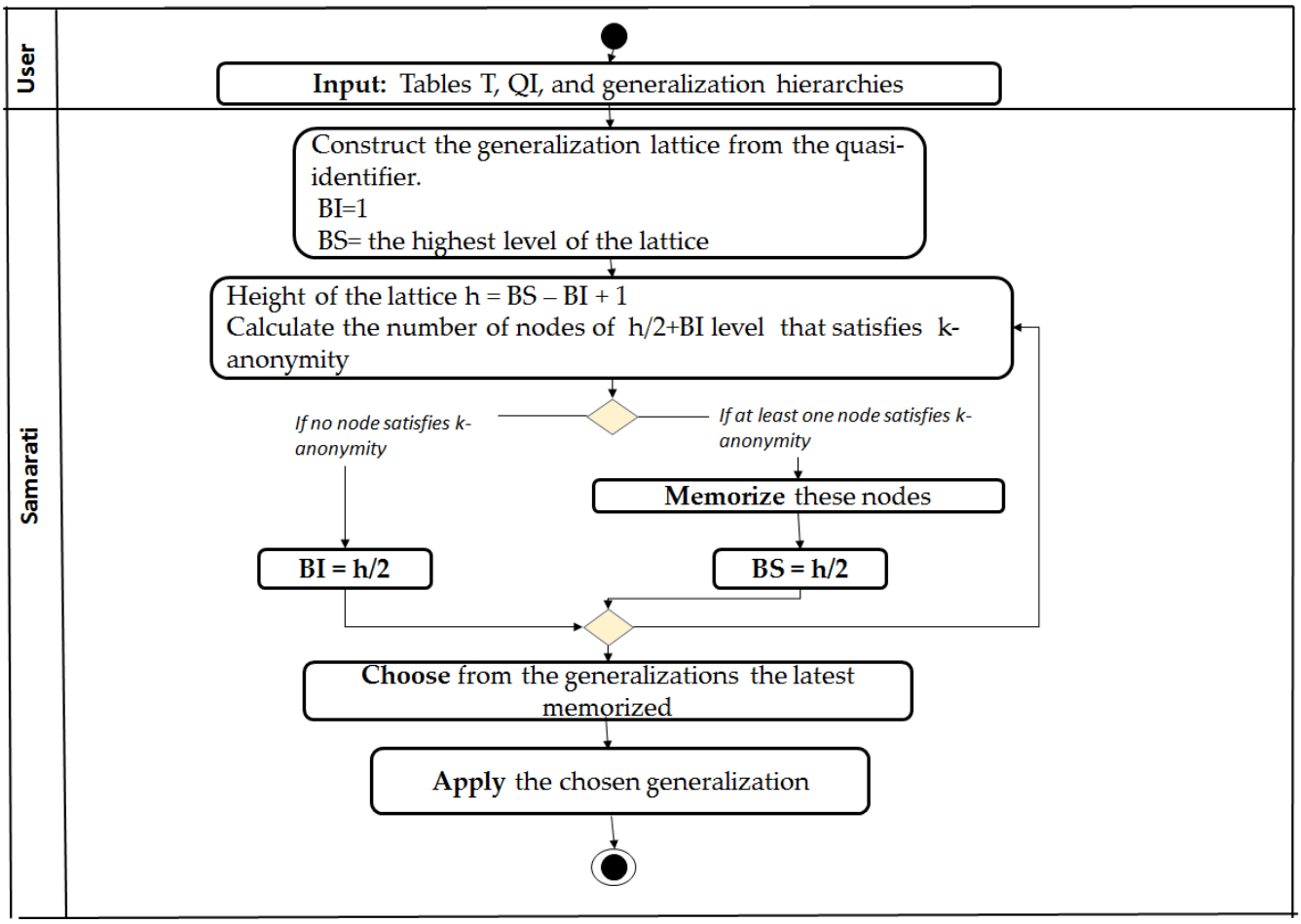


Figure 41. Abstraction de l'algorithme Samarati

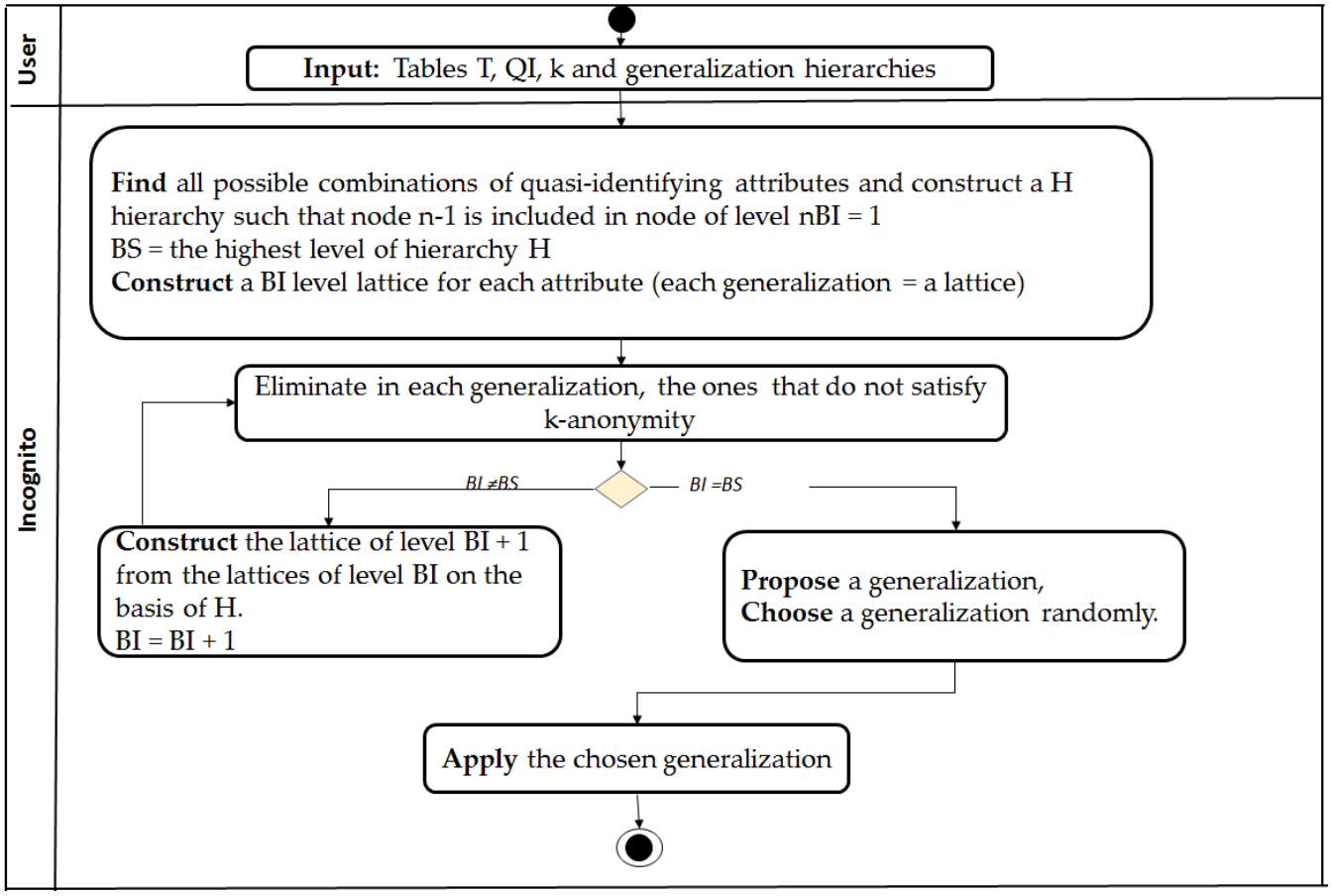


Figure 42. Abstraction de l'algorithme Incognito

Nous avons appliqué l'abstraction à l'algorithme de généralisation « bottom up » et nous avons obtenu un diagramme d'activité clair et simple comme le montre la **Figure 43** en éliminant toutes les boucles (sauf une). En effet, l'utilisateur doit spécifier la table à anonymiser T, la contrainte k, l'ensemble QI et leurs hiérarchies de généralisation et enfin les attributs cibles que nous avons nommés contraintes de classification. L'algorithme sélectionne la meilleure généralisation ayant le meilleur score et l'applique. Enfin, si tous les enregistrements satisfont le k-anonymat, l'algorithme s'arrête, si non, il passe à l'étape 2.

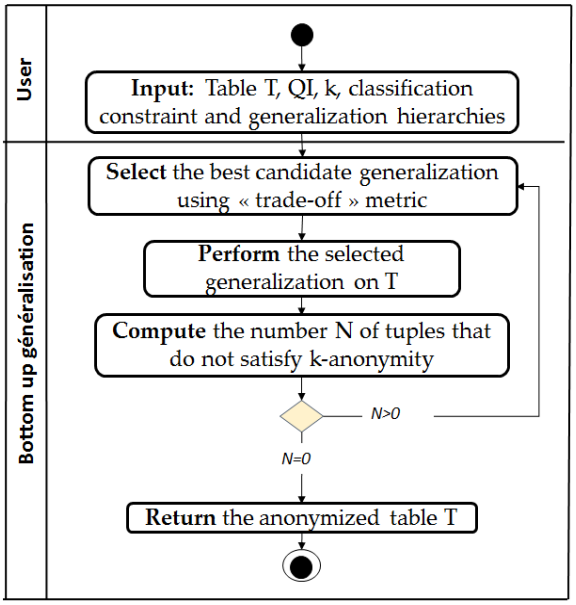
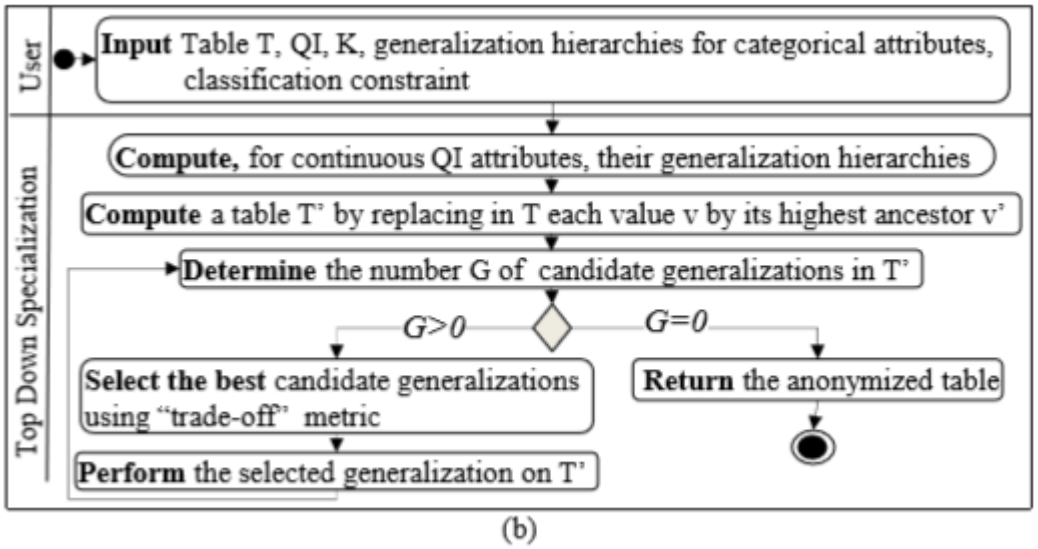


Figure 43. Abstraction de l'algorithme « Bottom up generalization »

Le processus d'abstraction de cet algorithme conduit au modèle présenté dans la **Figure 44**. Grâce à cette abstraction, nous exhibons le fait que TDS commence à partir d'une table T' qui représente le plus haut niveau de généralisation, en donnant la priorité à la sécurité au détriment de la qualité. Puis, afin de trouver le meilleur compromis entre la qualité et la sécurité, TDS tente de se rapprocher des valeurs originales de T.



(b)

Figure 44. Abstraction de l'algorithme TDS

Nous avons fait une abstraction de cet algorithme afin de faciliter sa compréhension tout en évitant les fonctions récursives complexes et les structures de données. Notre abstraction de Median Mondrian a conduit au diagramme d'activités de la **Figure 45**.

A chaque itération, l'algorithme choisit une dimension et vérifie la possibilité de diviser un groupe en deux sous-groupes (c'est-à-dire découper la zone selon la valeur médiane de cette dimension). Un groupe peut être divisé en deux sous-groupes si chaque sous-groupe résultant contient au moins k individus. Si la division

n'est pas possible, le groupe correspondant est marqué. Le processus de séparation passe à une autre dimension lorsque tous les groupes sont marqués pour la dimension actuelle. Il s'arrête quand toutes les dimensions ont été explorées. Ensuite, l'algorithme effectue des généralisations (appelé processus de recodage), en remplaçant les différentes valeurs dans la même zone avec la valeur de leur premier parent commun dans la hiérarchie de généralisation.

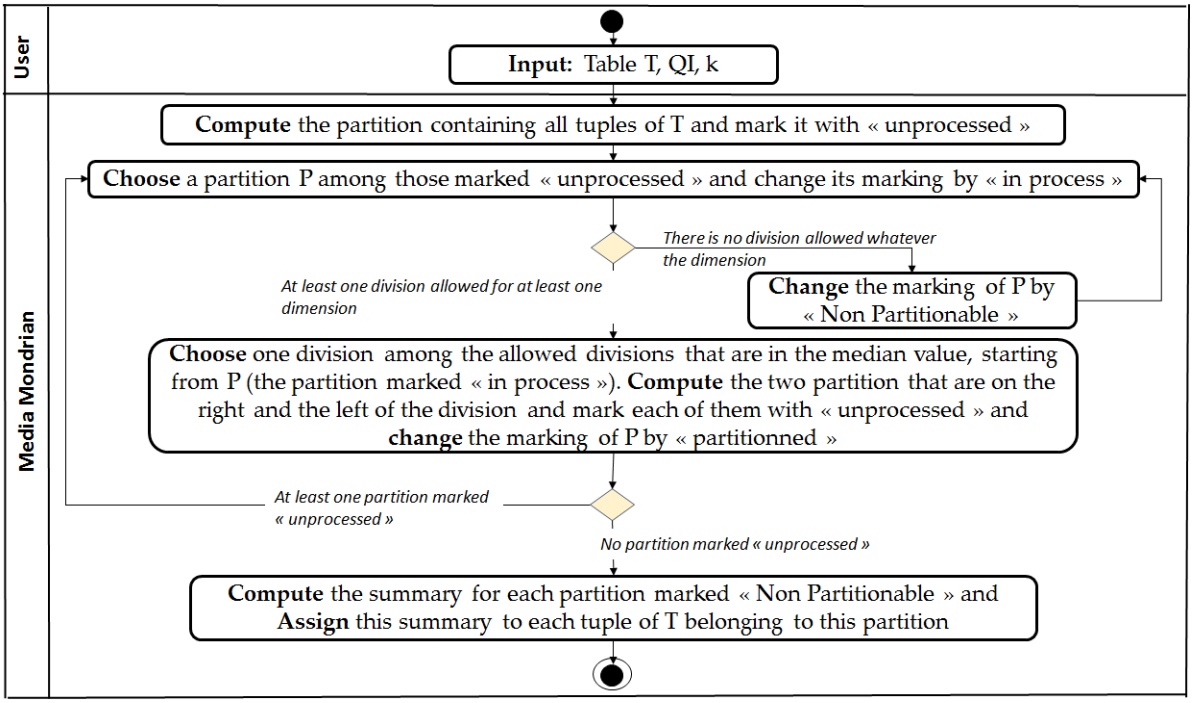


Figure 45. Abstraction de l'algorithme Median Mondrian

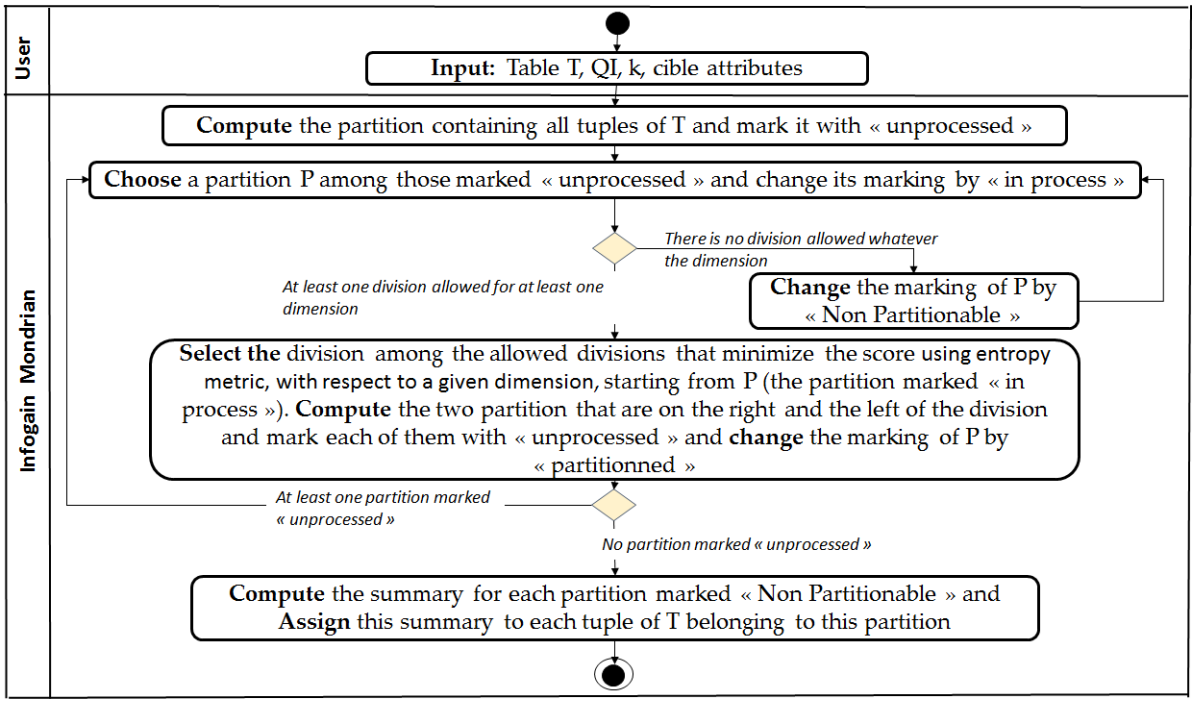


Figure 46. Abstraction de l'algorithme InfoGain Mondrian

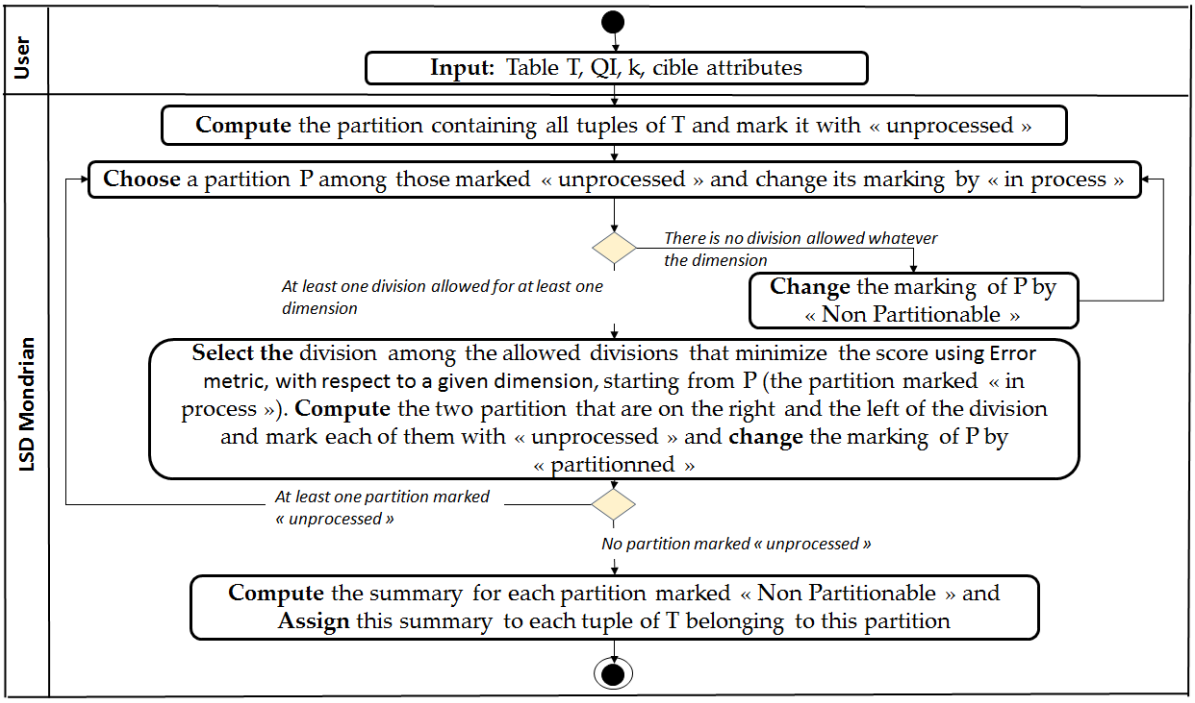


Figure 47. Abstraction de l'algorithme LSD Mondrian

2.2 Notre processus de généralisation appliqué aux abstractions d'algorithmes

Nous avons, jusque-là, appliqué l'abstraction par paramétrage à neuf algorithmes qui implémentent tous la technique de généralisation. L'abstraction de chaque algorithme nous a permis d'exhiber des similitudes et de regrouper les algorithmes similaires dans des catégories. Nous avons aussi défini, pour chaque catégorie, une représentation abstraite. Ainsi, nous avons défini un mappage de un à plusieurs entre cette représentation et les différentes abstractions d'algorithmes appartenant à cette catégorie.

Nous avons d'abord regroupé les algorithmes en catégories selon leurs principes de fonctionnement et selon les types de généralisation résultants. Ces deux critères de généralisation d'abstraction nous ont conduits à identifier trois catégories :

- La catégorie rassemblant les algorithmes dont le recodage[4] est fondé sur la notion d'espace multidimensionnel. Le recodage est exécuté sur la base d'un partitionnement de cet espace.
- La catégorie regroupant les algorithmes dont le recodage est fondé sur la notion de treillis. Le recodage est effectué une fois que la généralisation est sélectionnée dans le treillis.
- Enfin, la catégorie comportant les algorithmes dans lesquels le recodage est appliqué pas à pas, de façon itérative, sur la table à anonymiser.

On symbolisera respectivement chacune de ces techniques par :

[4] Le terme recodage est souvent utilisé pour définir la transformation, par généralisation, requise sur les données

- MRT pour « Multidimentional Recoding Technique »,
- LRT pour «Lattice Recoding Technique»,
- TRT pour «Table Recoding Technique».

Ainsi les algorithmes Median Mondrian, Infogain Mondrian et LSD Mondrian font partie de la catégorie MRT (**Figure 48**). Les algorithmes Samarati et Incognito appartiennent à la catégorie LRT. Enfin, la catégorie TRT comprend Datafly, µ-ARGUS, ainsi que les algorithmes « Bottom up generalization » et TDS.

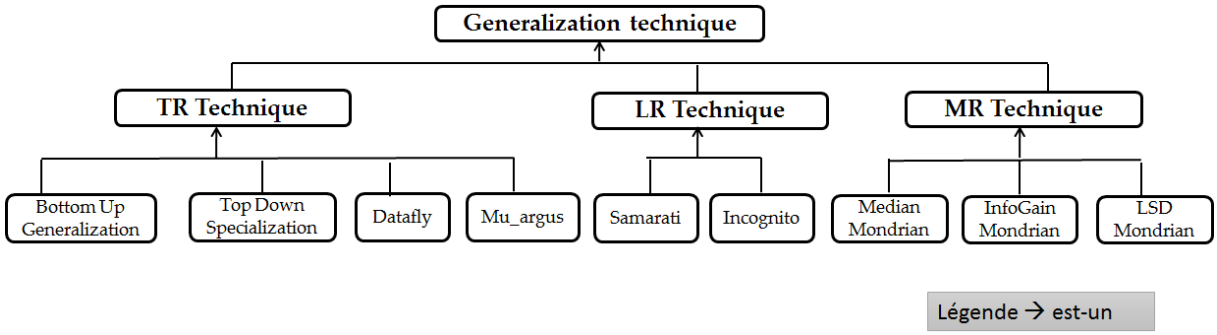


Figure 48. Taxonomie des algorithmes de généralisation

Les algorithmes de la catégorie MRT génèrent des généralisations multidimensionnelles. Leur spécificité est qu'ils considèrent simultanément l'ensemble des attributs du QI en les plaçant dans un même espace multidimensionnel. Ils procèdent au partitionnement de cet espace de façon itérative jusqu'à ce que chaque partition puisse regrouper au moins k enregistrements. Les algorithmes des deux autres catégories ne génèrent pas de généralisations multidimensionnelles. En effet, ils manipulent, à chaque itération, un seul attribut du QI à la fois.

Pour faciliter la compréhension de ces catégories, nous avons aussi produit une abstraction pour chaque catégorie. Pour ce faire, il était nécessaire d'homogénéiser les abstractions des algorithmes, ce qui a nécessité d'augmenter le paramétrage des algorithmes et des techniques de généralisation (TRT, LRT et MRT).

Comme exemple d'homogénéisation des algorithmes de la catégorie TRT, nous pouvons citer l'introduction d'un paramètre permettant d'interdire ou d'autoriser les suppressions. Pour les algorithmes n'autorisant pas de suppression, ce paramètre est positionné à la valeur « faux ». Dans le cas contraire, il est positionné à la valeur « vrai ». Vu que la suppression peut être locale ou globale, le concept de « appropriate suppression » (suppression appropriée) a été introduit afin de distinguer les deux types de suppression.

Nous avons aussi procédé à une reformulation des instructions en définissant de nouveaux concepts. A titre d'exemple, dans la mesure où dans l'abstraction de TDS, nous avions introduit le concept de « candidate generalization » (généralisation candidate) et le concept de "best candidate generalization » (meilleure généralisation candidate), nous avons transformé l'abstraction de Datafly en introduisant le concept de

«candidate attribute » (attribut candidat à la généralisation) et le concept de « best candidate attribute » (meilleur attribut candidat).

Un autre exemple d'homogénéisation a concerné les métriques utilisées dans chaque algorithme pour la sélection de la meilleure généralisation. Le paramètre « appropriate metric » (métrique appropriée) a été introduit. Ce paramètre, prend, à titre d'exemple, la valeur « trade-off metric » pour « métrique de compromis » dans le cas de l'algorithme TDS et la valeur « distinct metric » pour l'algorithme Datafly.

Le troisième exemple d'homogénéisation a trait aux paramètres en entrée des algorithmes de généralisation de TRT. Dans la mesure où certains de ces paramètres sont fournis par l'utilisateur tandis que d'autres sont calculés préalablement par certains algorithmes, il a fallu, dans un souci de compréhension de ces algorithmes, les distinguer. Pour ce faire le concept de « parameters » (paramètres) » et « other parameters » (autres paramètres) ont été introduits. Le premier concept est instancié par les paramètres fournis par l'utilisateur, le second est renseigné par les paramètres calculés par l'algorithme.

Cette homogénéisation a donné lieu à une ré-écriture des quatre abstractions de TRT (voir **Figure 49** à **Figure 52**) et à la génération d'une abstraction pour cette catégorie (voir **Figure 53**).

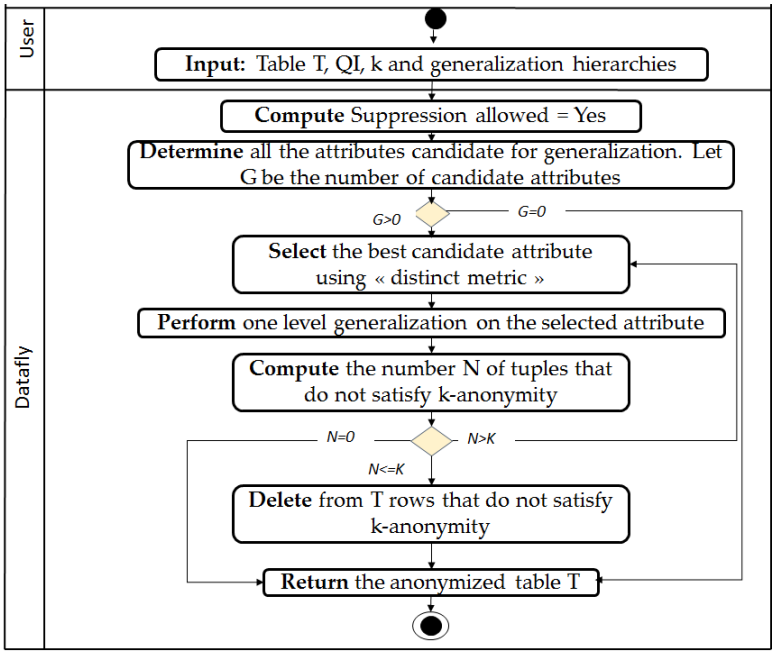


Figure 49. Abstraction homogénéisée de l'algorithme Datafly

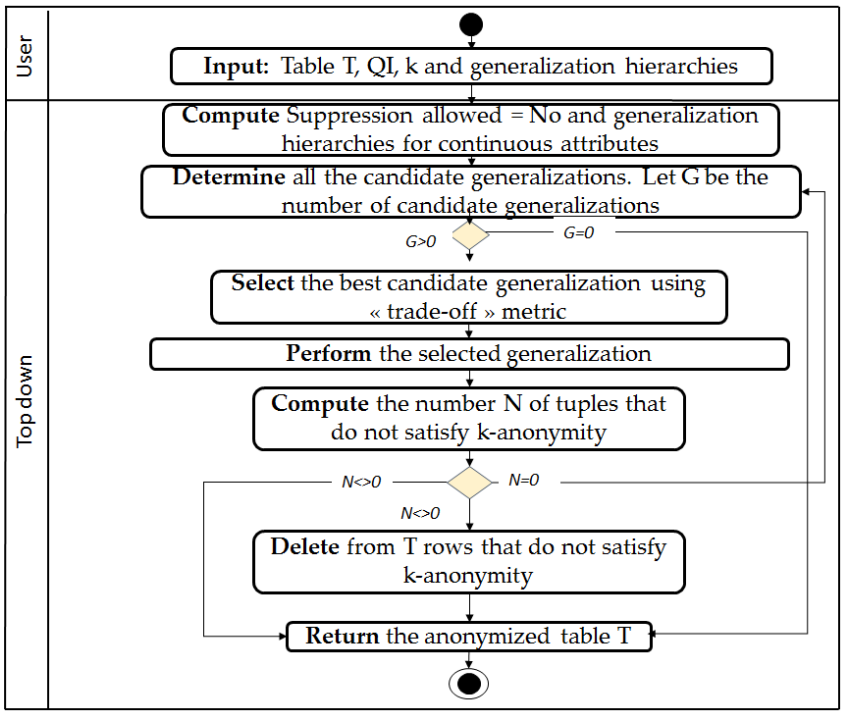


Figure 50. Abstraction homogénéisée de l'algorithme TDS

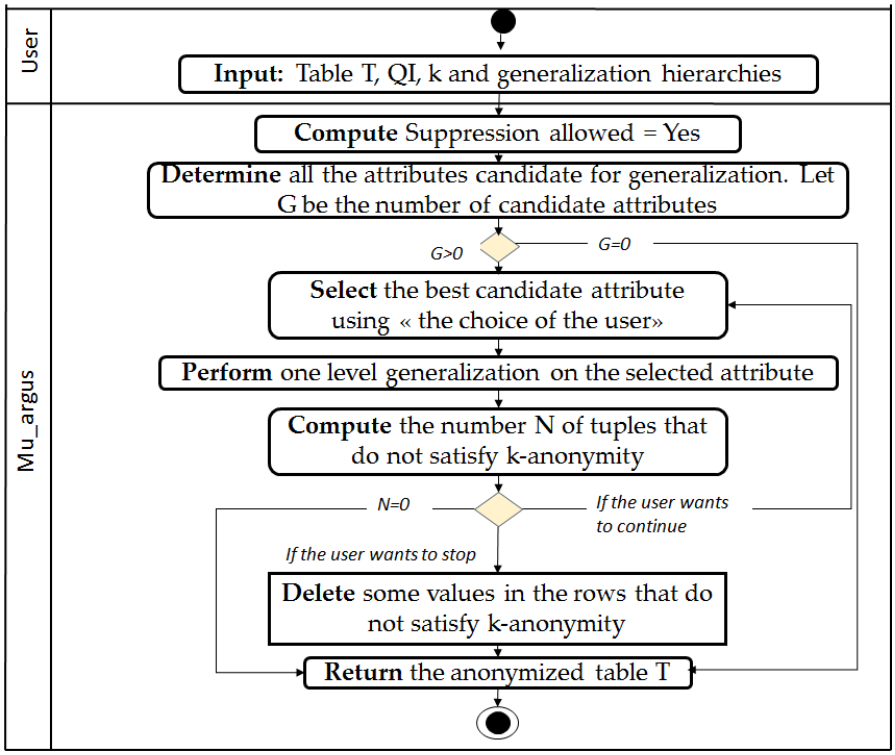


Figure 51. Abstraction homogénéisée de l'algorithme μ_argus

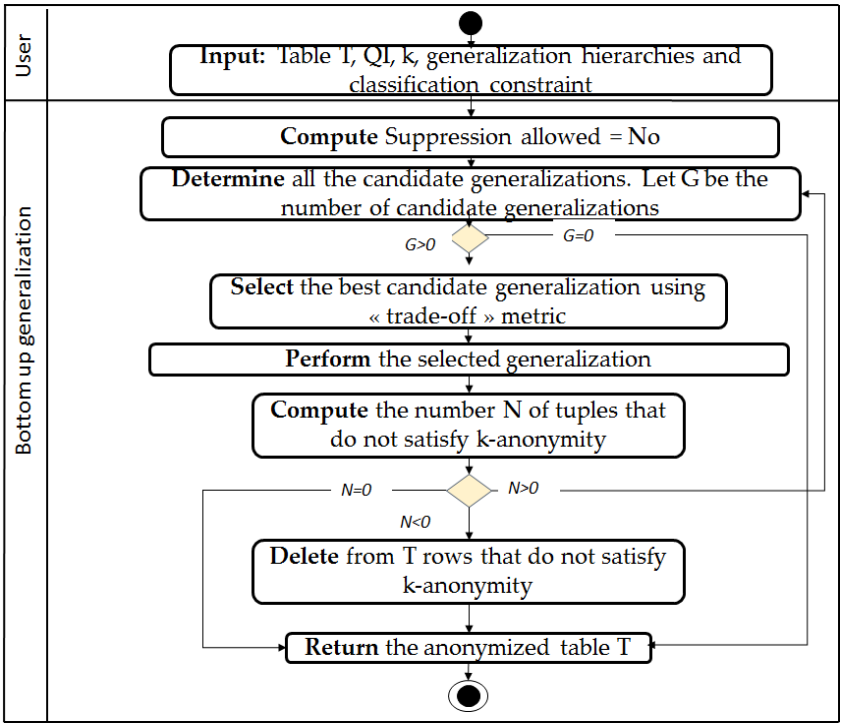


Figure 52. Abstraction homogénéisée de l'algorithme Bottom up generalization

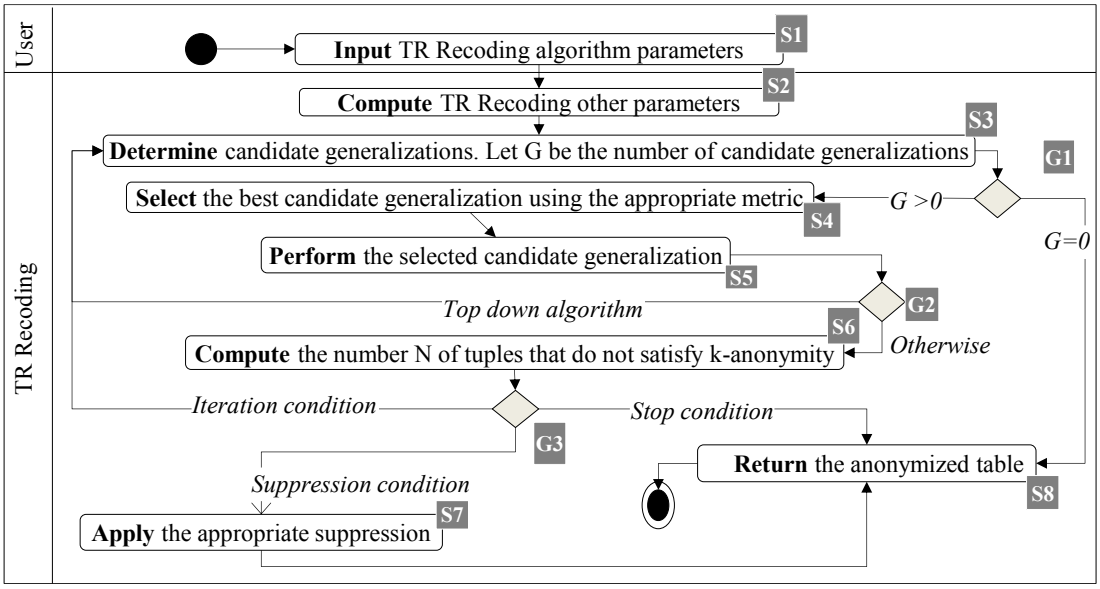


Figure 53. Abstraction de la technique TRT

Dans la **Figure 53**, une étiquette Si (respectivement une étiquette Gi) représente une étape de l'algorithme (respectivement une condition de branchement) dans cette abstraction.

Ainsi, l'abstraction de la **Figure 53** peut être instanciée afin d'obtenir les quatre algorithmes de cette catégorie grâce à un paramétrage correct. Cette instanciation est détaillée dans la **Table 1**.

Table 1. Instanciation de la **Figure 53** pour les quatre algorithmes de TRT

TR Technique		Instantiation of the parameters			
Step/Gateway number	**Parameter**	**Datafly**	**µ-argus**	**Top Down Specialization**	**Bottom up Generalization**
S1	**User inputs**	Table T, QI, k and hierarchies for categorical attributes of the QI			
		Hierarchies for continuous attributes of the QI	Hierarchies for continuous attributes of the QI	Classification constraint	
			Maximum number of suppressions		
S 2	**Other parameters**	Suppression grant		Hierarchies for continuous attributes of the QI	
				Table T'	
S3	**Candidate generalization**	A candidate attribute which values in T aren't in the top of the generalization hierarchy		A valid and beneficial value v' of an attribute in T' which is not a leaf of the generalization hierarchy	The set of values which are not in the top of the generalization hierarchy but share the same ancestors v' in T'
	Appropriate metric	Distinct metric	user choice	tradeoff metric	
S5	**all S5**	Each value v of the selected candidate attribute is updated on T into the value just		Replace v' in T' by its children which it also the	Replace in T' each value of the select set by v'
G3	**Stop condition**	N=0	N=0	Not applicable	N=0
	Iteration condition	N>K	N≠0 and the user choose	Not applicable	N≠0
	Suppression condition	N≤K and suppression granted	Not applicable	Not applicable	N≠0 and suppression granted and the user choose
S8	**Appropriate suppression**	row deletion	Not applicable	Not applicable	values deletion
S9	**Anonymized Table**	Table T	Table T	Table T	Table T

Suivant cette logique ascendante, nous avons effectué le même effort d'abstraction pour homogénéiser les algorithmes au sein de chacun des deux autres types de recodage, en l'occurrence MRT et LRT. L'annexe B fournit le résultat de cette homogénéisation ainsi que les abstractions de MRT (**Figure 54**) et LRT (**Figure 57**). Ceci nous a permis de générer aussi une abstraction de la technique de généralisation (**Figure 58**(a)). Cette abstraction identifie six étapes. La première étape donne la possibilité à l'utilisateur d'introduire les paramètres associés à la technique. Certains de ces paramètres sont facultatifs car non applicables à tous les algorithmes de généralisation. La seconde étape comprend des éléments facultatifs et des éléments automatiques car elle est relative au calcul de paramètres nécessaires à certains algorithmes de généralisation tels que les algorithmes de la technique TRT. La seconde étape permet de choisir la généralisation à exécuter, la troisième l'exécute. L'avant dernière étape vérifie le k-anonymat et exécute éventuellement des suppressions.

Une instanciation de cette abstraction pour la sous-technique TRT est fournie dans la **Figure 58**(b). Les autres instanciations pour les deux autres sous-techniques sont en annexe B de cette thèse (**Table 2** et **Table 3**).

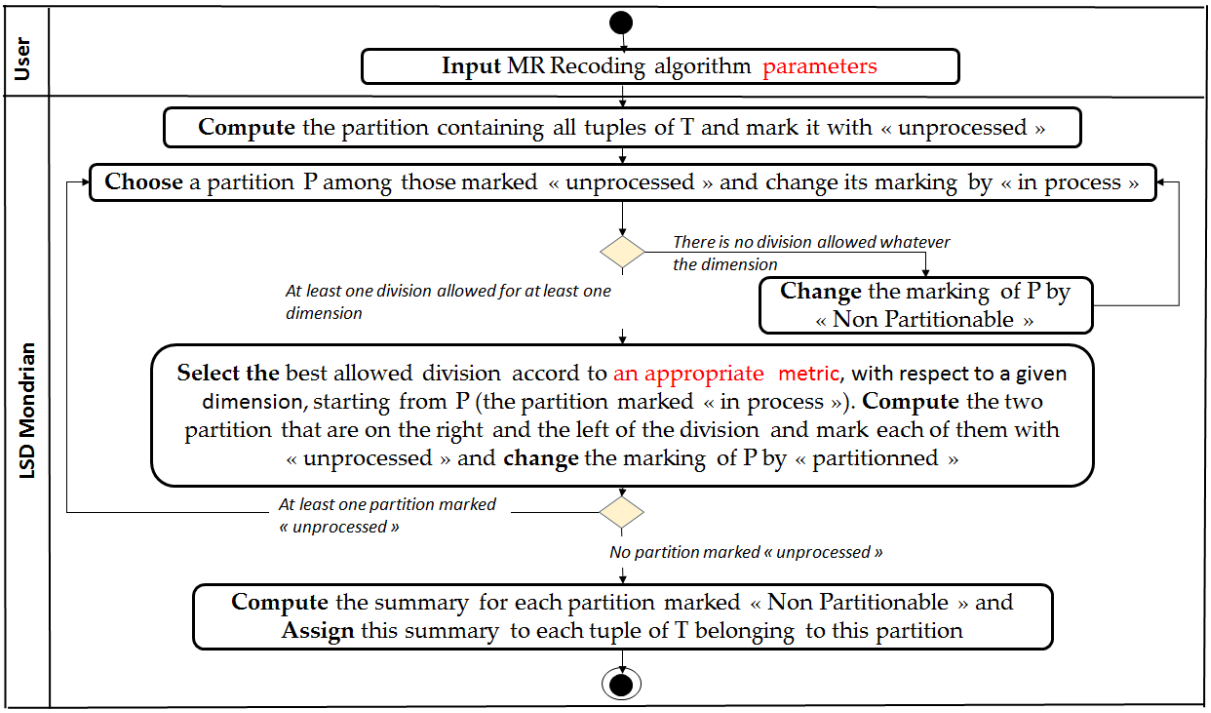


Figure 54. Abstraction de la MR technique

Table 2. Instanciation de la **Figure 54** pour les trois algorithmes de MRT

MR Technique parameterization	Instantiation of parameterization components		
	Median Mondrian	InfoGain Mondrian	LSD Mondrian
Parameters	Table T, QI, k , generalization hierarchies		
		List of target attributes	
Appropriate metric	Median	Entropy metric	Error metric

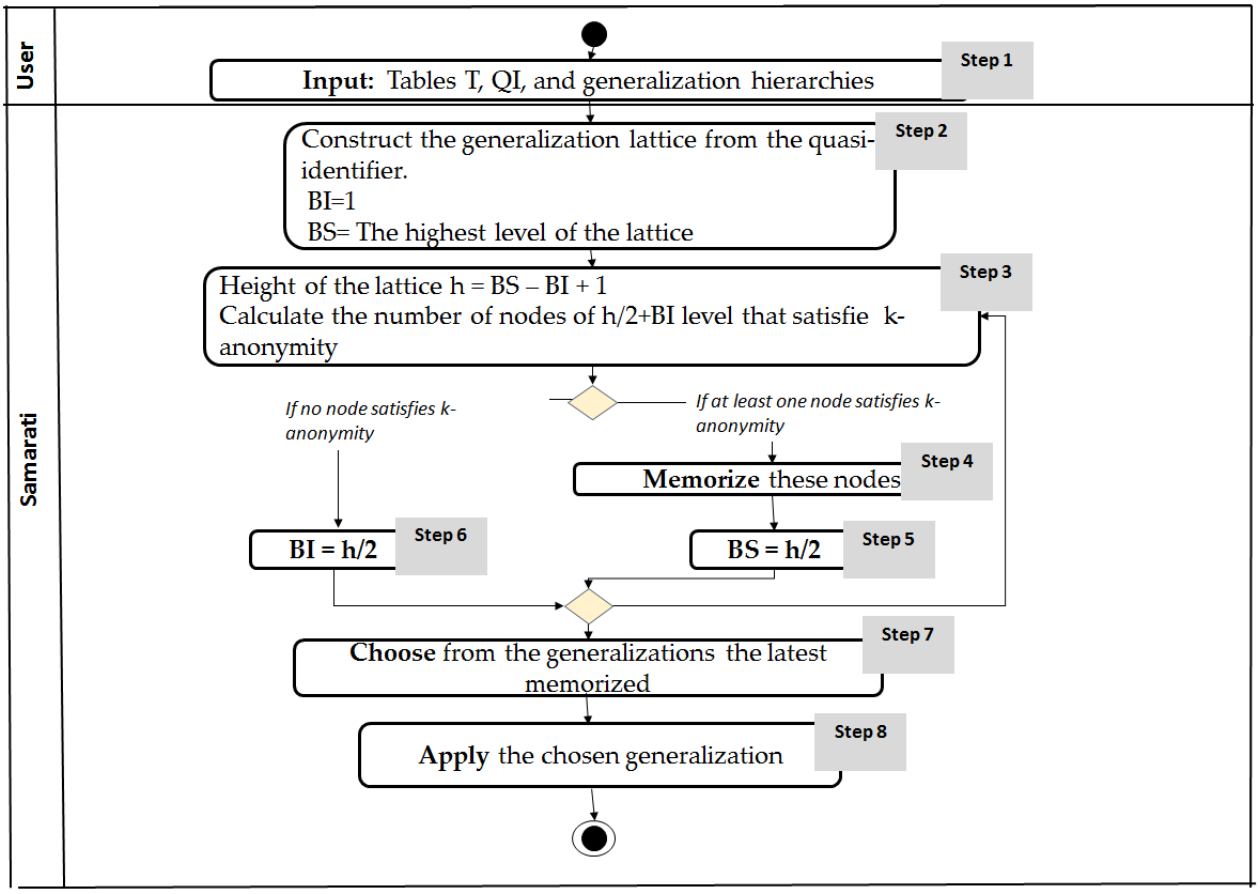


Figure 55. Abstraction homogénéisée de l'algorithme de Samarati

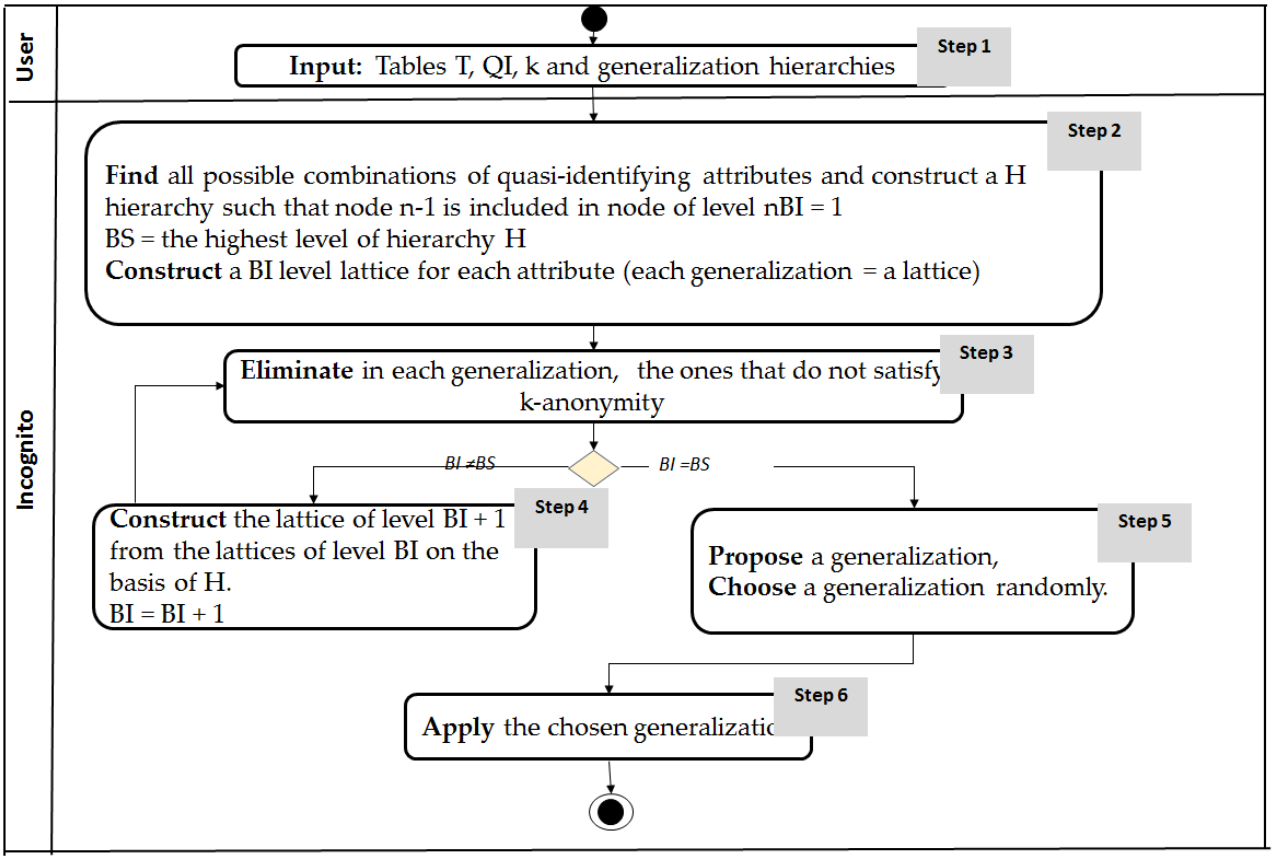


Figure 56. Abstraction homogénéisée de l'algorithme Incognito

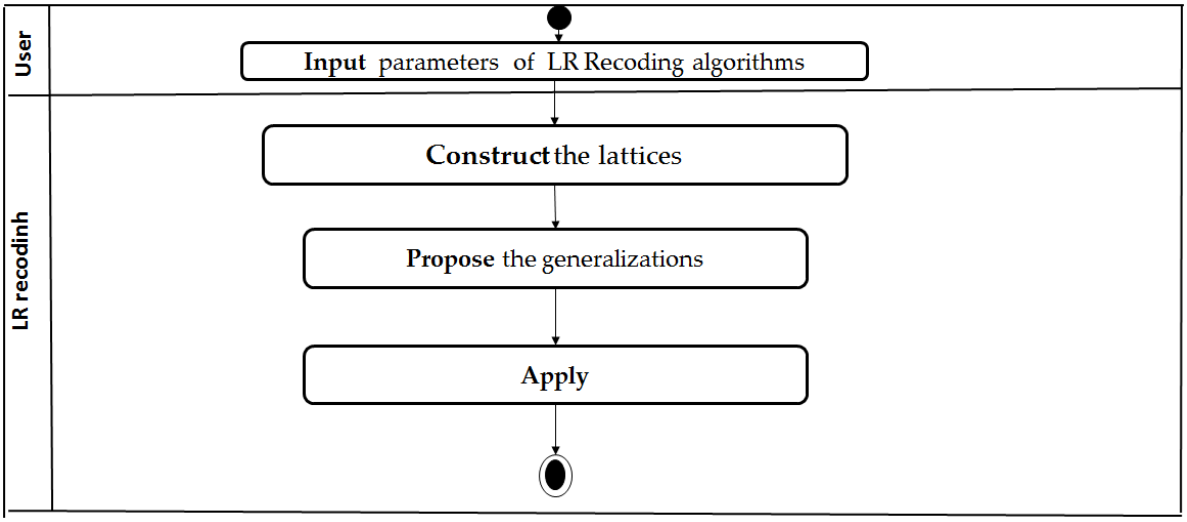
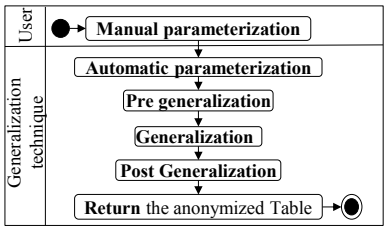


Figure 57. Abstraction de la LR Technique

Table 3.Instanciation de la **Figure 57** pour les deux algorithmes de la LR technique

Les composants du paramétrage	Instanciation de LR	
	Samarati	Incognito
Les inputs des algorithmes de LR Technique	Etape 1	Etape 1
Construire le treuills	technique	Etape 2
Proposer les généralisations	Etapes 3, 4, 5, 6 et 7	Etapes 3, 4 et 5
Appliquer	Etape 8	Etape 6



(a)

Parametrization components	Instantiation for TR recoding
Manual parametrization	*step 1*
Automatic parametrization	*step 2*
Pre generalisation	*step 3 and 4*
generalization	*step 5*
post generalization	*step 6 and 7*

(b)

Figure 58. Abstraction de la technique de généralisation et son instanciation pour TRT

3. Validation des abstractions

Afin d'évaluer l'utilité de nos abstractions, nous avons effectué une expérience nous permettant de les comparer à celles fournies dans la littérature. Nous avons choisi de faire subir une expérimentation à un ensemble de testeurs composé de personnes ayant des compétences en programmation. Ainsi, si les utilisateurs ayant des compétences en programmation préfèrent utiliser notre abstraction, combien plus des éditeurs de données avec moins de compétences en programmation seront à l'aise avec elle. Nous avons aussi souhaité évaluer aussi bien

l'utilité perçue que l'utilité objective de nos abstractions. L'utilité perçue est capturée au moyen de questions sur la façon dont les participants peuvent comprendre la logique sous-jacente de chaque algorithme. L'utilité objective est mesurée grâce à l'exactitude des résultats obtenus par les participants après exécution des abstractions.

Nous avons mené notre comparaison pour deux algorithmes, l'un considéré, de notre point de vue, parmi les plus faciles à appréhender et l'autre parmi les plus difficiles. Notre choix s'est porté ainsi sur Datafly et Median Mondrian.

Douze participants ont été recrutés. Ils étaient tous soit des étudiants doctorants, soit des chercheurs en informatique. Par conséquent, ils étaient tous familiers avec l'algorithmique et la programmation. Cependant, ils n'avaient pas de connaissances sur les algorithmes d'anonymisation. Pour éviter toute interprétation biaisée des résultats, nous avons fourni nos abstractions sous forme textuelle et non sous forme de diagrammes.

L'expérience a duré quatre heures environ au cours desquelles, dans un premier temps, les participants ont renseigné un questionnaire sur leur niveau de connaissances des techniques d'anonymisation et sur leurs compétences en programmation. Pour le premier point, ils devaient évaluer leur niveau en utilisant une échelle de 1 à 10. Pour le second point, ils devaient mentionner s'ils avaient des compétences anciennes, récentes ou très récentes en programmation ; ceci nous a permis d'associer à chacun d'eux un profil : 1 pour participant à compétences anciennes, 2 pour participant à compétences récentes, et 3 pour participant à compétences très récentes.

Une fois le questionnaire rempli, tous les participants ont reçu :

- une brève présentation orale de l'anonymisation en mettant l'accent sur la technique de généralisation,
- une copie des diapositives,
- des feuilles de papier vierges pour prendre des notes,
- et un jeu de données à anonymiser (une petite table à anonymiser).

Dans un troisième temps, en nous appuyant sur leurs profils de programmation, nous avons réparti les participants en deux groupes homogènes. Chaque participant d'un même groupe a reçu les mêmes algorithmes d'anonymisation pour une exécution manuelle sans limitation de temps. Nous avons proposé :

- pour le premier groupe, notre abstraction de Datafly (**Figure 39**), puis, successivement, Datafly puis Median Mondrian sous leur forme publiée dans la littérature et reprise respectivement dans la **Figure 39** et la **Figure 45** du chapitre 3.
- pour le second groupe, notre abstraction de Median Mondrian (**Figure 45**), suivie successivement de Median Mondrian puis Datafly sous leur forme publiée dans la littérature et reprise respectivement dans la **Figure 11** et la **Figure 25**.

Aux algorithmes proposés dans la littérature, nous avons joint les explications proposées par leurs auteurs. L'annexe B rassemble tous les documents fournis aux participants. En particulier, les participants n'étaient pas informés du fait qu'ils avaient le même algorithme sous deux formes différentes.

Lorsqu'un participant rencontrait un problème lors de l'exécution d'un algorithme, il était invité à indiquer sur sa feuille la raison de son blocage.

Une fois les copies rendues, chaque participant d'un groupe a été invité à répondre aux questions figurant sur l'un des deux formulaires décrits en **Figure 60** et **Figure 61**. Ces questions avaient pour objectif :

- dans un premier temps de s'assurer que la présentation initiale, au début de l'expérience, était jugée suffisante pour les participants,
- dans un second temps, de déceler si le participant a été en mesure de reconnaître des abstractions différentes du même algorithme et, si tel était le cas, de demander si notre abstraction l'a aidé à dérouler l'abstraction suivante (c'est-à-dire la description algorithmique de la littérature).

Les participants devaient également attribuer à chaque algorithme, sur une échelle de 0 à 10, un niveau de difficulté (voir **Figure 60** et **Figure 61**).

QUESTIONNAIRE GROUPE 1 et 2 AVANT L'EXPERIENCE)

Evaluez votre niveau d'expertise en algorithmique sur une échelle de 0 à 10

Evaluez votre niveau d'expertise en techniques d'anonymisation sur une échelle de 0 à 10

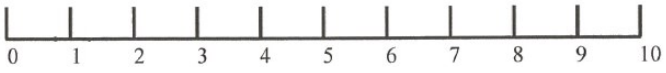


Figure 59. Questionnaire rempli avant l'expérience

QUESTIONNAIRE GROUPE 1 APRES L'EXPERIENCE

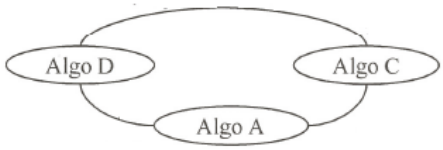
La présentation vous a-t-elle suffi pour l'exécution des algorithmes? **Oui** □ **Non** □

Si votre réponse est « non » quelles informations vous ont manqué ?

……
……
……..

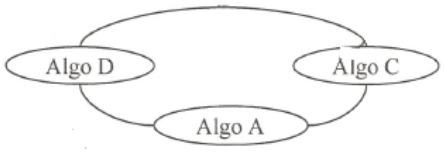
Vous avez exécuté sur un même exemple trois algorithmes. Orientez et annoter les arêtes du graphe ci-après par l'une des annotations :

1: « ressemble à » **ou** **2: « est différent de »** **ou** **3: «est le même que»**



Vous avez exécuté sur un même exemple trois algorithmes. Orientez et annoter les arêtes du graphe ci-contre par l'une des annotations :

1: «m'a aidé à comprendre» **ou** **2: «ne m'a pas aidé à comprendre»**



Associez à chacun de ces trois algorithmes un niveau de difficulté de compréhension sur une échelle de 0 à 10 :

ALGO A : □ ALGO D □ ALGO C : □

Que pensez vous de chacun des algorithmes que vous avez exécuté ?

……
……
……..

Figure 60. Questionnaire pour le groupe 1 rempli après l'expérience

QUESTIONNAIRE GROUPE 2 APRES L'EXPERIENCE

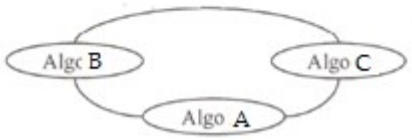
La présentation vous a-t-elle suffi pour l'exécution des algorithmes? **Oui** ☐ **Non** ☐

Si votre réponse est « non » quelles informations vous ont manqué ?

.....
.....
.....

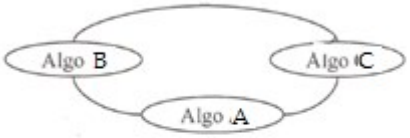
Vous avez exécuté sur un même exemple trois algorithmes. Orientez et annoter les arêtes du graphe ci-après par des annotations :

1: « ressemble à » **ou** **2: « est différent de »** **ou** **3: «est le même que»**



Vous avez exécuté sur un même exemple trois algorithmes. Orientez et annoter les arêtes du graphe ci-contre par l'une des annotations :

1: «m'a aidé à comprendre» **ou** **2: «ne m'a pas aidé à comprendre»**



Associez à chacun de ces trois algorithmes un niveau de difficulté de compréhension sur une échelle de 0 à 10 :

ALGO A : ☐ ALGO B ☐ ALGO C : ☐

Que pensez vous de chacun des algorithmes que vous avez exécuté ?

.....
.....
.....

Figure 61. Questionnaire pour le groupe 2 rempli après l'expérience

Une fois l'expérience terminée, nous avons analysé les informations collectées. Chacun des douze participants a exécuté les trois algorithmes. Nous avons rejeté trois exécutions illisibles sur les 36. Pour chaque algorithme, les exécutions lisibles ont été regroupées en trois classes. La première (resp. la seconde) a rassemblé toutes les exécutions correctes (resp. toutes les exécutions partiellement correctes). La dernière classe contenait toutes les exécutions erronées. Pour les exécutions partielles, nous avons déduit, à partir des commentaires des participants, trois raisons de blocage. La première raison est liée à l'interprétation de l'instruction de 'WHILE' dans Datafly original (**Figure 11** du chapitre 3). La deuxième raison était la compréhension de la structure de données "freq" dans ce même algorithme. Enfin, la troisième raison est la double récursivité du Median Mondrian sous sa forme originale (**Figure 25** du chapitre 3).

Le **Tableau 45** résume les résultats de notre expérience.

	Abstraction de Datafly			Datafly d'origine			Abstraction de Median Mondrian			Median Mondrian d'origine		
Classe	**Erroné**	**Correct**	**Partiel**	**Erroné**	**Correct**	**Partiel**	**Erroné**	**Correct**	**Partiel**	**Erroné**	**Correct**	**Partiel**
Profile 1	20%	0%	0%	20%	0%	0%	20%	20%	0%	20%	0%	10%
Profile 2	0%	40%	0%	0%	20%	20%	0%	20%	0%	0%	20%	10%
Profile 3	0%	40%	0%	0%	20%	20%	0%	40%	0%	0%	40%	0%

Tableau 45. Synthèse des données collectées à partir de l'expérience

Pour chaque algorithme et chaque profil, les pourcentages d'exécutions erronées, correctes et partiellement correctes sont mentionnés. Aucun des participants n'a été confronté à des difficultés bloquantes qui l'auraient amené à arrêter sa tâche. En outre, seuls ceux qui avaient des anciennes connaissances ont obtenu des exécutions erronées et ils sont peu nombreux.

De plus, seuls ceux qui ont commencé par l'exécution de l'abstraction de Datafly ont proposé une exécution correcte du Datafly d'origine. Pourtant, Datafly est, selon nous, un algorithme très simple. La même observation a émergé pour Median Mondrian.

Les participants ont été unanimes sur le fait que notre abstraction les a aidés à comprendre les algorithmes originaux. En outre, ils ont mentionné que les algorithmes originaux sont plus difficiles à comprendre que nos abstractions. Tous les participants qui ont eu un blocage dans le Datafly d'origine (40%) n'ont pas eu à exécuter notre abstraction dans cette expérience. Il en est de même pour ceux qui ont commencé par l'exécution de Median Mondrian sous sa forme d'origine.

Pour résumer, cette analyse a révélé que le manque d'intelligibilité des algorithmes originaux impacte tous les participants quelles que soient leurs compétences en programmation.

La taille limitée de notre groupe de participants et la restriction de l'étude à seulement deux algorithmes peuvent limiter la validité de notre expérimentation. Cependant, cette première analyse nous encourage à persévérer dans cet effort d'abstraction.

4. Conclusion

Notre analyse inductive des articles référençant les neuf algorithmes de généralisation de micro-données nous a permis dans un premier temps de les caractériser (section 5.1 du chapitre 3) puis, dans un second temps, de fournir des représentations simplifiées de ces algorithmes. Ces représentations simplifiées ont été obtenues suite à l'application d'un processus d'abstraction par paramétrage. Elles sont, à notre avis, beaucoup plus intelligibles, notamment pour des personnes sans compétence en programmation.

L'abstraction de ces algorithmes nous a permis d'identifier des similitudes et de procéder à leur regroupement au sein de catégories de sous-techniques identifiées, via un processus de généralisation d'abstractions. A ces sous-techniques dont le nombre est de trois, nous avons aussi associé des abstractions. Un dernier effort

d'abstraction par généralisation nous a permis de fournir une description abstraite de la technique de généralisation proprement dite.

Nous avons ainsi construit une taxonomie aidant un novice à comprendre comment tous ces algorithmes d'anonymisation par généralisation de micro-données fonctionnent et comment ils peuvent être différenciés.

La validation de l'utilité d'une abstraction d'un algorithme a été conduite au travers d'une expérimentation contrôlée, effectuée sur un panel de participants disposant de compétences variées en programmation. Les résultats de cette expérimentation sont encourageants. Ils nous invitent à poursuivre nos efforts d'abstraction à l'ensemble des algorithmes d'anonymisation de micro-données.

Toutes les abstractions déduites peuvent faire partie d'une base de connaissances qui peut être rendue disponible, via un guidage informatif, au sein d'un processus d'anonymisation informatisé. Cette base de connaissance, objet de notre recherche future, renfermera une famille de « patterns » documentant aussi bien les techniques que les algorithmes quant à leurs entrées, leurs sorties, leurs processus et éventuellement des exemples illustratifs. Les processus y seront décrits aussi bien à l'aide des algorithmes existant dans la littérature qu'à travers nos abstractions.

Les abstractions déduites peuvent aussi être utilisées dans le cadre du e-apprentissage (e-learning en anglais). Ils peuvent servir de base dans le développement de tutoriels pour le transfert de compétence en anonymisation. Ces derniers peuvent être contextualisés selon le niveau d'expertise des utilisateurs potentiels.

La taxonomie proposée dans ce chapitre, ainsi que la caractérisation des algorithmes d'anonymisation de micro-données par généralisation, constitue un point de départ à la conception de l'ontologie d'anonymisation que nous décrivons dans le chapitre qui suit. Cette ontologie est exploitée par l'approche d'anonymisation présentée dans le chapitre 6.

Chapitre 5 Construction de l'Ontologie Pour l'Anonymisation de Micro-données (OPAM)

Face à la complexité des processus d'anonymisation, aux risques qu'encourent les entreprises si ces processus sont mal menés et à l'absence de ressources sémantiques où les éditeurs de données peuvent trouver les connaissances qui leur manquent, nous proposons, dans le cadre de cette thèse, d'extraire et de formaliser les connaissances relatives aux techniques d'anonymisation qui sont actuellement contenues dans la myriade de travaux de recherche sur ce sujet afin de les rendre disponibles via une ontologie de domaine. Sa mise à disposition permettra :

- à un expert, de l'enrichir,
- à un éditeur, d'éclairer son jugement sur un algorithme ou une technique d'anonymisation et pourquoi pas d'adopter de nouvelles techniques,
- et enfin, à une entreprise, de minimiser la perte de compétences qui peut survenir lorsqu'un de ses employés qualifiés la quitte.

Elle constitue aussi une source d'information exploitée dans le processus d'anonymisation que nous proposons dans le cadre de cette thèse et dont la description détaillée fait l'objet du chapitre suivant.
Le présent chapitre, après un état de l'art sur les méthodes de construction d'ontologies de domaine, justifie et décrit le processus proposé pour la conception de notre ontologie d'anonymisation de micro-données. La suite du chapitre présente la mise en œuvre de notre ontologie.

1. Etat de l'art sur la construction d'ontologies de domaine

Le terme ontologie est utilisé avec des sens différents dans différents domaines (Staab et Studer 2009). Il est issu du domaine de la philosophie, où il signifie « explication systématique de l'existence ». Dans le domaine de l'informatique, ce terme, initialement adapté par Gruber (Gruber et others 1993) comme « une spécification explicite d'une conceptualisation », a vu son sens évoluer. La définition, qui devient prédominante et que nous adoptons dans le cadre de cette thèse, est celle proposée par Studer et al. (Studer, Benjamins, et Fensel 1998) : « une ontologie est une spécification explicite et formelle d'une conceptualisation partagée ». Dans l'explicitation de leur définition, ces mêmes auteurs traduisent le terme « conceptualisation » par «un modèle abstrait dans lequel ont été identifiés les concepts pertinents ». Ils utilisent l'adjectif «formelle» pour insister sur le fait que la spécification doit être lisible, donc interprétable, par les machines et le qualificatif «explicite» pour insister sur le fait que les concepts utilisés ainsi que les contraintes associées à cet usage doivent être

explicitement définies. La notion de partage met l'accent sur le fait qu'une ontologie capture des connaissances consensuelles (c'est-à-dire des connaissances acceptées par des groupes de personnes).

Les types d'ontologies les plus couramment utilisés sont, selon Gomez et al. (Gómez-Pérez et Benjamins 1999) :

- les ontologies, pour la représentation des connaissances ou encore pour la construction d'ontologies, qui regroupent les primitives pour la description d'ontologies,
- les ontologies de haut niveau qui décrivent des concepts généraux tels que le temps, l'espace, la matière, les objets, les événements, les actions, etc.
- les ontologies de domaine qui rassemblent des concepts propres à un domaine spécifique,
- les ontologies de tâche qui conceptualisent des tâches spécifiques à un domaine,
- enfin les ontologies d'application qui contiennent des concepts dépendant à la fois d'un domaine et d'une tâche.

La catégorisation de Gomez est faite selon l'objet de conceptualisation concerné par l'ontologie. D'autres catégorisations existent. Certaines sont décrites dans (Psyché, Mendes, et Bourdeau 2003). La catégorisation présentée ci-dessus permet de situer le type d'ontologie ciblé dans cette thèse. En effet, dans le cadre de notre recherche, nous produisons une ontologie du domaine de l'anonymisation de micro-données.

La formalisation des connaissances dans une ontologie de domaine est faite à l'aide de quatre types de composants principaux (Gómez-Pérez et Benjamins 1999) : les concepts, les relations, les axiomes et les instances. Les concepts représentent un ensemble d'entités ou d'objets du domaine. Les relations traduisent les associations entre concepts. Dans une ontologie de domaine, au minimum est représentée la relation de généralisation/spécialisation. Tout autre type de lien peut être ajouté pour spécifier plus finement le domaine. Parmi ces liens, on peut citer le lien associatif, le lien « fait partie de », « est composé de », etc. Au même titre que les concepts, certaines relations peuvent disposer de propriétés pour capturer au mieux les connaissances. On peut, par exemple, énoncer qu'une relation est optionnellement disponible pour un concept donné. Les axiomes sont des assertions censées fournir de l'aide au raisonnement. Parmi ces axiomes, on peut considérer ceux servant à contraindre les valeurs des propriétés des concepts. Enfin, les instances constituent la définition extensionnelle de l'ontologie. Ils sont les objets, les entités qu'est censé représenter le concept.

La conception d'ontologies est une activité laborieuse qui a suscité l'intérêt de plusieurs chercheurs dès les années 1990. Plusieurs méthodes ont été construites afin de transformer l'art de construction d'une ontologie en une activité d'ingénierie. Ces méthodes comprennent essentiellement des lignes directrices pour la construction d'une ontologie. Bien que nombreuses, la plupart des méthodes ne gagnent pas en popularité, en termes d'utilisation. A titre d'information, selon l'enquête menée par Cordoso en 2007, 60% des personnes questionnées affirment ne pas avoir utilisé de méthodologie pour la construction de leurs ontologies (Cardoso et Sheth 2006). Dans les paragraphes qui suivent, nous décrivons, succinctement, quelques méthodes citées dans la littérature. Nous avons retenu celles qui nous semblent les plus connues.

1.1 La méthode « TOVE » (Gruninger et Fox 1995)

TOVE (TOronto Virtual Enterprise) a été l'une des premières méthodes de construction d'ontologies pour le domaine des affaires. TOVE a montré comment formaliser des ontologies à partir des connaissances du domaine étudié. TOVE se concentre d'abord sur l'extraction des exigences de l'ontologie qui sont spécifiées sous la forme de questions informelles auxquelles l'ontologie doit permettre de répondre[5]. Ces questions sont définies sur la base de scénarii préalablement identifiés. Ces derniers motivent le besoin d'une ontologie. Ils sont accompagnés d'un ensemble de solutions intuitives permettant de répondre aux problèmes posés par les scénarii.

La deuxième étape consiste à définir les composants de l'ontologie (concepts, propriétés, relations) et à spécifier leurs définitions et contraintes. Ces composants sont extraits des réponses aux questions informelles.

La description informelle de l'ontologie est ensuite transformée en logique du premier ordre qui est un des langages formels d'expression des ontologies. Il en est de même pour les questions informelles. Ces dernières sont formalisées. Elles guident la définition d'axiomes associés à l'ontologie.

Une évaluation de l'ontologie en termes de complétude (capacité à répondre à toutes les questions formelles posées) est faite à l'issue de sa construction.

1.2 La méthodologie "METHONDOLOGY" (Fernández-López 1999)

Dans METHONDOLOGY, le processus de construction de l'ontologie combine l'ingénierie des connaissances et le génie logiciel. La construction de l'ontologie est faite à partir de zéro et son cycle de vie est constitué de cinq grandes étapes :

- *La spécification* : l'identification des objectifs et des utilisateurs finaux de l'ontologie visée.
- *L'acquisition des connaissances* : L'extraction des connaissances du domaine à partir de plusieurs sources : les experts, les livres, des tableaux, des articles, etc.
- *La conceptualisation* : la structuration des connaissances du domaine.
- *La formalisation :* la traduction automatique du modèle conceptuel en un modèle formel.
- *La mise en œuvre* : l'écriture du modèle formel à l'aide d'un langage de mise en œuvre.

Des directives ont été proposées sous la forme de tableaux prédéfinis afin de faciliter l'acquisition et la conceptualisation des connaissances.

1.3 La méthodologie “DILIGENT”

Cette méthode est destinée à soutenir les experts du domaine dans un contexte où l’ontologie va être composée de plusieurs ontologies locales. Cette méthodologie se focalise sur l’ingénierie ontologique collaborative et distribuée. Son processus de développement de l'ontologie est constitué des activités suivantes:

1. *La construction* : cette activité correspond à la construction d’une ontologie initiale.
2. *L´adaptation locale* : une fois l'ontologie initiale mise à disposition, les utilisateurs peuvent commencer à l'utiliser et l’adapter localement selon leurs propres besoins.
3. *L´analyse* : un groupe d’experts analyse les ontologies locales et les requêtes des utilisateurs et essaie d’identifier les similitudes entre leurs ontologies.
4. *La révision* : le groupe d’experts doit régulièrement réviser l'ontologie partagée, de sorte que les parties des ontologies qui se chevauchent ne se contredisent pas.
5. *La mise à jour locale* : une fois qu’une nouvelle version de l'ontologie partagée est mise à disposition, les utilisateurs peuvent mettre à jour leurs propres ontologies locales afin de mieux utiliser les connaissances représentées dans la nouvelle version.

1.4 Le cadre méthodologique NeOn (Suárez-Figueroa, Gómez-Pérez, et Fernández-López 2012)

Neon propose un cadre méthodologique de construction d’ontologie dans lequel on identifie cinq modèles de développement d’ontologies (quatre en cascade et un itératif) et neuf scénarios intervenant pour la conception d’ontologies.

Les neuf scénarios identifiés sont considérés comme étant les plus plausibles (voir **Figure 62**).

La phase de spécification des besoins est la phase préalable à tous les scénarios. Elle permet de définir le cahier des charges associé à l’ontologie. Ce dernier renseigne sur l’objectif de l’ontologie, son domaine, le langage à utiliser lors de sa mise en œuvre, la cible (groupe) à qui elle est destinée, ses utilisations potentielles ainsi que l’ensemble de ses exigences exprimées sous forme de questions informelles auxquelles elle doit répondre. Une fois cette phase finalisée, une recherche et une étude des ressources (ontologiques et non ontologiques) disponibles favorisant l’élaboration de l’ontologie, s’opèrent. Cette phase de recherche et d’étude constitue la phase maîtresse dans le processus d’élaboration de l’ontologie car, à son issue, au moins un parmi les neuf scénarios de construction d’ontologie est choisi et l’activité de gestion du projet démarre. Cette dernière comprend entre autres la définition des ressources humaines nécessaires à la construction de l’ontologie.

Les phases de conceptualisation, de formalisation et de mise en œuvre sont communes à tous les scénarios. Elles peuvent être effectuées moyennant d'autres méthodes telles que METHONDOLOGY.

La phase de conceptualisation permet l’obtention d’une description conceptuelle (sous forme de modèle) de l’ontologie. Le modèle conceptuel résultant est traduit, lors de la phase de formalisation, en un modèle logique nommé, dans Neon, «semi-computable model». L’implémentation consiste à mettre en œuvre le modèle logique à l’aide d’un langage ontologique.

Les neuf scénarios candidats proposés dans Neon sont décrits ci-après. Les huit premiers sont des scénarios préparant la phase de conceptualisation pour une ontologie mono-lingue. Le neuvième scénario favorise la prise en compte du multi-linguisme.

- **Scénario 1 :** Ce scénario est exécuté en l'absence de ressources, ontologiques ou non ontologiques. La construction de l'ontologie se fait, dans ce cas, «à partir de rien » (« from scratch »). Elle débute par l'activité d'acquisition des connaissances. Cette dernière se poursuit, en continu, tout au long du processus de construction. Une fois les connaissances acquises, elles sont conceptualisées (activité de conceptualisation).
- **Scénario 2** : *Réutilisation et réingénierie de ressources non-ontologiques.* Dans ce scénario, des ressources non ontologiques sont exploitées. Un processus de réingénierie permettant leur transformation en ontologies est mis en œuvre. Ce processus précède bien sûr l'activité de conceptualisation.

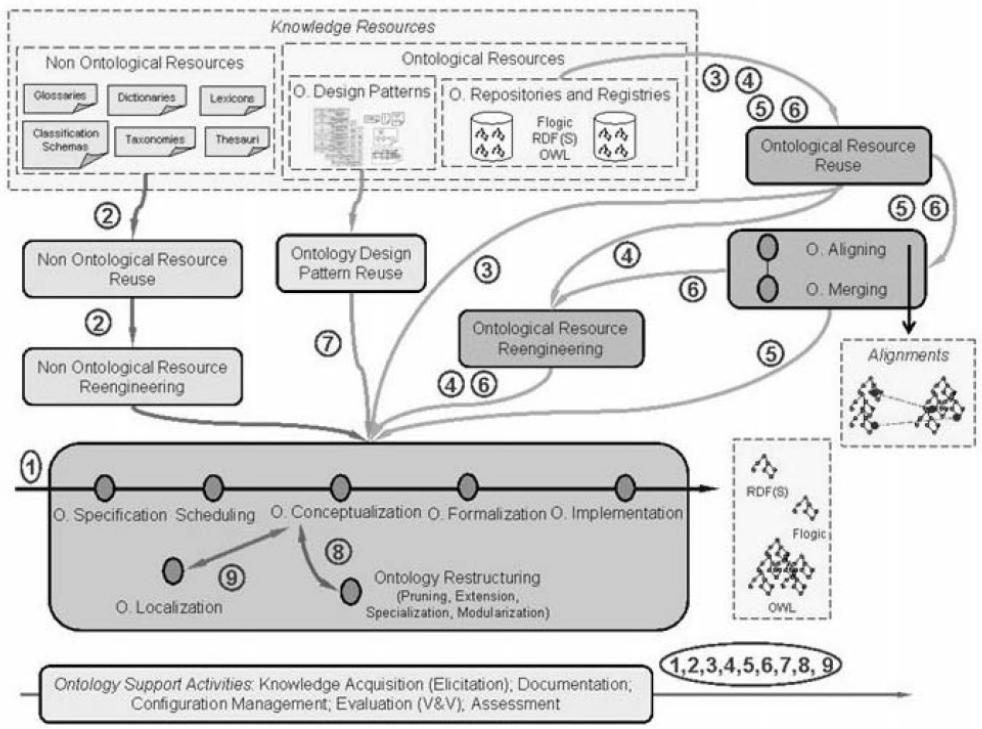


Figure 62. Les neuf scénarios de la méthodologie Neon (Suárez-Figueroa, Gómez-Pérez, et Fernández-López 2012)

- **Scénario 3 :** *Réutilisation de ressources ontologiques*. Dans ce scénario, des ressources ontologiques couvrant partiellement ou totalement le domaine de l'ontologie visée sont disponibles. Une étude des ressources est alors menée afin d'extraire les portions répondant au cahier des charges établi. Le résultat de cette étude peut mener à une fusion ou intégration de portions de ressources candidates.
- **Scénario 4 :** *Réutilisation et réingénierie de ressources ontologiques***.** Ce scénario est appliqué en cas de disponibilité d'une ressource ontologique relevant du même domaine que celui de l'ontologie cible et qui

répondrait aux exigences du cahier des charges moyennant quelques changements tels que sa traduction dans la langue cible ou encore sa transformation, par le biais d'un processus de réingénierie, afin qu'elles répondent aux contraintes d'implémentation énoncées dans le cahier des charges.

- **Scénario 5 :** *Réutilisation et fusion des ressources ontologiques*. Ce scénario est sélectionné dans le cas où plusieurs ressources ontologiques couvrant le domaine ciblé sont disponibles. Dans ce cas, l'ontologie cible conceptualisée sera issue de la fusion des ontologies sources. Cette fusion s'opère après un appariement de concepts se trouvant dans les différentes ontologies sources.
- **Scénario 6 :** *Réutilisation, fusion et réingénierie de ressources ontologiques*. Ce scénario est un cas particulier du scénario 5. Il s'applique lorsqu'une réutilisation de ressources ontologiques couvrant le domaine ciblé est possible mais qu'elle nécessite, avant la fusion, la réingénierie d'au moins une des ressources trouvées.
- **Scénario 7 :** *Réutilisation de « patterns » de conception d'ontologies*. Ce scénario consiste à s'appuyer sur des « patterns » de conception d'ontologies disponibles sur Internet[6] pour la construction de l'ontologie.
- **Scénario 8 :** *Restructuration des ressources ontologiques***.** Ce scénario suppose l'existence d'une ressource ontologique candidate que l'on peut restructurer pour la rendre conforme au cahier des charges. Par exemple, on peut citer la transformation de l'ontologie source en ontologie modulaire, son élagage par suppression de liens non requis, son enrichissement par d'autres concepts et liens, etc.
- **Scénario 9 :** *Localisation de ressources ontologiques*. Ce scénario est applicable dans le cas où le cahier des charges prévoit l'utilisation de l'ontologie dans un contexte multilingue. Il permet de transformer l'ontologie en une ontologie multilingue.

1.5 Discussion

L'ingénierie ontologique est une discipline qui étudie les principes, les méthodes et les outils pour créer et maintenir des ontologies. La plupart des méthodologies existantes se focalisent sur un scénario de construction d'ontologies. Neon fait exception en suggérant une variété de chemins menant à la construction d'ontologies. Toutes les méthodes « mono-scénarios », excepté METHONDOLOGY, offrent des lignes directrices couvrant partiellement les étapes de développement d'une ontologie. A titre d'exemple, TOVE se concentre sur les questions informelles alors que DILIGENT se focalise sur les étapes de collaboration.

[6] http://ontologydesignpatterns.org/wiki/Main_Page

2. Notre approche de construction d'OPAM

Une ontologie de domaine capte, représente et modélise les connaissances d'un domaine. Sa construction à partir de rien, est une activité consommatrice d'effort. Cependant, face à l'absence d'ontologies, même embryonnaires dans ce domaine, nous avons planifié de la construire en nous inspirant du scénario 1 de Néon.

Comme mentionné précédemment, la construction d'une ontologie doit obéir à un cahier des charges dans lequel sont spécifiés, *a minima*, ses objectifs et ses exigences. Les objectifs auxquels doit répondre OPAM peuvent être déduits de la question de recherche à laquelle nous souhaitons contribuer et que nous avons formulée comme suit, en page 4 de l'introduction générale de cette thèse :

> «Comment aider un professionnel chargé de l'anonymisation des données à choisir une technique et un algorithme d'anonymisation et comment l'aider à concrétiser son processus de dé-identification ?»

En d'autres termes, l'objectif d'OPAM est d'aider un professionnel chargé de l'anonymisation de micro-données (à des fins de publication), via les connaissances qu'elle met à sa disposition, à choisir un procédé de brouillage[7]. Quant aux exigences, nous les exprimons, comme dans la plupart des méthodes connues, sous la forme de questions informelles auxquelles elle doit répondre.

- Quels sont les procédés d'anonymisation de micro-données disponibles ?
- Qu'est ce qui caractérise les procédés existants ?
- Quel est le contexte d'utilisation des procédés existants ?
- Qu'est ce qui permettrait de déduire qu'un procédé appliqué à un jeu de micro-données est celui qui convient ?
- Dispose-t-on de statistiques d'évaluation des procédés existants qui pourraient aider dans le choix d'un procédé pour un jeu de micro-données donné ?

Les réponses à ces questions constituent l'information qu'offrira OPAM pour le choix de procédé.

Les connaissances requises, que renfermera notre ontologie, ne sont disponibles que dans des articles de recherche. Le volume de ces derniers étant considérable, nous avons choisi un modèle de développement incrémentiel nous permettant de fournir notre ontologie de domaine par incrément plutôt qu'en un seul lot. Comme le montre la **Figure 63**, lors du premier incrément, nous menons une activité d'acquisition de connaissances pour une technique et ses algorithmes afin de collecter un noyau de concepts et de relations de base que nous représentons au sein d'un diagramme de classe UML. Notre ontologie ainsi conceptualisée, nous procédons à la définition de son extension. L'artefact résultant de cette activité est un diagramme d'objets UML.

[7] On parle indifféremment de brouillage, anonymisation, dé-identification

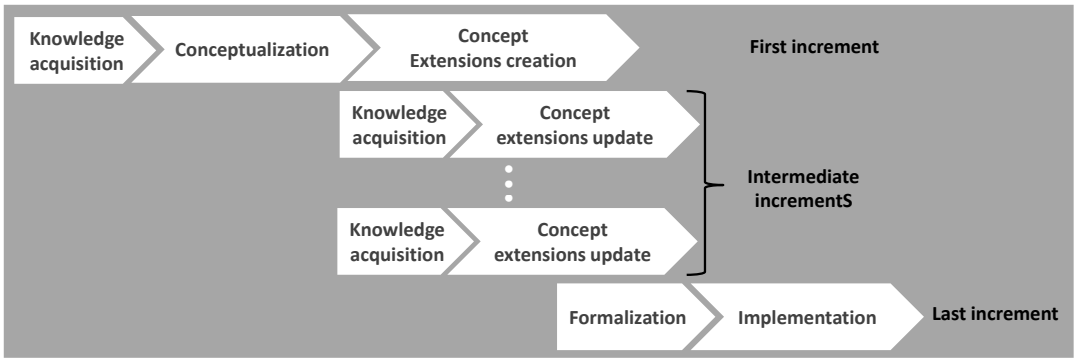


Figure 63. Le processus de la construction d'OPAM

Nous avons choisi d'initier notre processus de construction d'OPAM en nous focalisant, pour le premier incrément, sur l'anonymisation par généralisation de micro-données. Nous avons veillé à ce que la conceptualisation produite soit générique. Chaque incrément est une exécution de l'activité d'acquisition des connaissances associées à une technique non encore explorée. Cette activité est suivie de l'enrichissement de l'extension produite dans l'incrément précédent.

Une fois toutes les techniques d'anonymisation de micro-données explorées, nous enchaînons sur les activités de formalisation et de mise en œuvre de notre ontologie. Ces deux activités constituent le dernier incrément dans la construction de notre ontologie OPAM. L'activité de formalisation d'OPAM consiste 1) à transformer le diagramme des classes qui conceptualise notre ontologie en un modèle OWL et 2) à instancier le modèle OWL sur la base des objets collectés dans le diagramme objet UML relatif à OPAM. La mise en œuvre d'OPAM consiste en son implémentation dans un système de stockage.

Les deux sections qui suivent décrivent la phase d'acquisition des connaissances à appliquer à chaque incrément ainsi que son instanciation (phase de conceptualisation), au premier incrément, pour la technique de généralisation.

3. Phase d'acquisition des connaissances

Comme énoncé précédemment, les seules ressources disponibles permettant d'offrir une conceptualisation d'OPAM sont des ressources non ontologiques. Ces dernières doivent nous permettre de répondre aux questions informelles du cahier des charges d'OPAM. Compte tenu de notre connaissance sur l'anonymisation de micro-données acquise suite à l'étude de l'état de l'art du domaine, l'ensemble des questions informelles peut être reformulé en l'ensemble suivant de questions plus précises car adoptant le vocabulaire du domaine :

1. Quelles sont les techniques disponibles pour l'anonymisation de micro-données ? En quoi consistent-elles ?
2. Quels sont les algorithmes implémentant ces techniques ? En quoi consistent-ils ?
3. Qu'est ce qui caractérise les techniques et les algorithmes d'anonymisation de micro-données ?
4. Quel est le contexte d'utilisation des différents algorithmes disponibles ?
5. Qu'est ce qui permettrait de déduire qu'un algorithme appliqué à un jeu de micro-données est celui qui convient le mieux ?

6. Dispose-t-on d'évaluations servant de guide dans le choix d'algorithmes ?

Pour répondre aux questions 2 à 6, nous proposons, pour chaque incrément relatif à une technique d'anonymisation de micro-données, de mener cette phase d'acquisition de connaissance selon trois sous-phases (voir **Figure 64**).

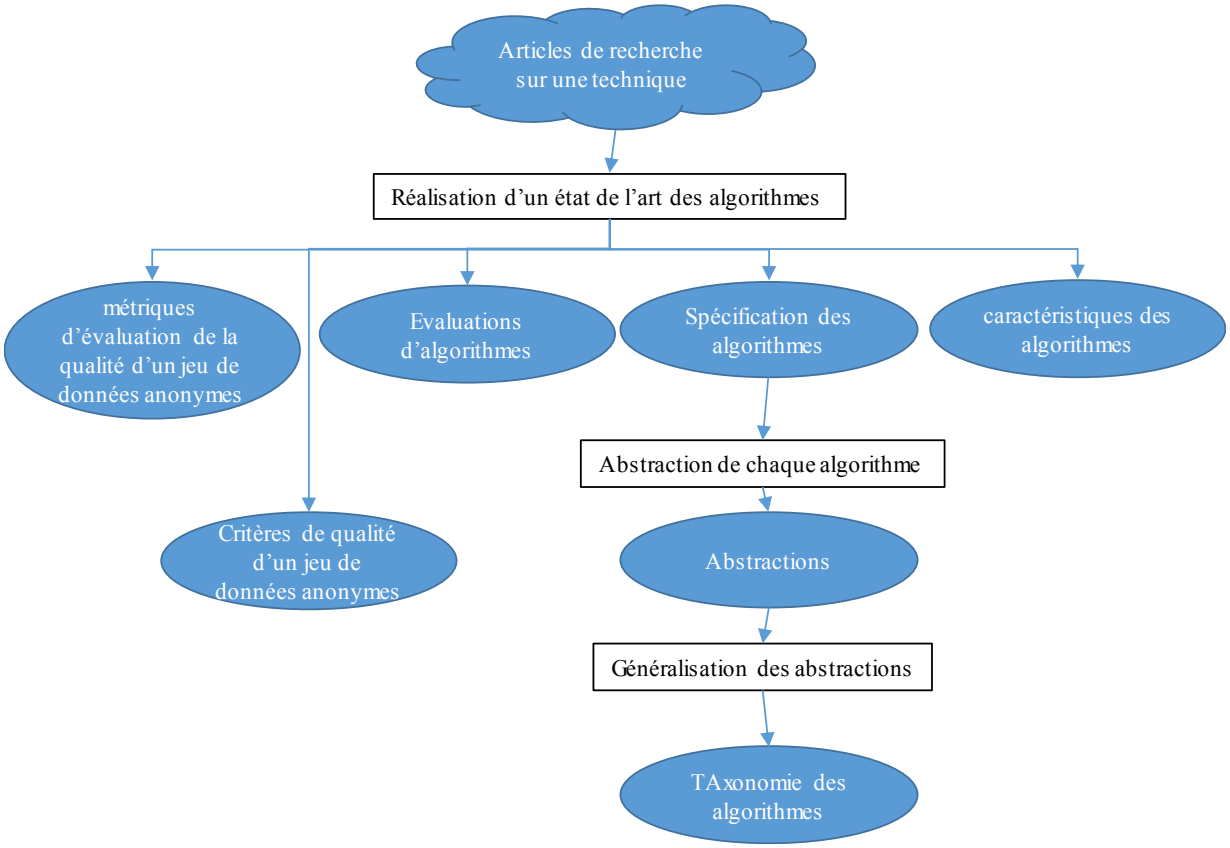


Figure 64. Les trois sous-phases de l'acquisition de connaissances

La première consiste à faire un état de l'art des algorithmes de la technique cible afin de les caractériser, d'extraire leur spécification, de recenser toutes les évaluations jusqu'à présent publiées sur ces algorithmes et d'obtenir des critères de qualité permettant d'évaluer un jeu de données anonymes ainsi que les métriques permettant de les mesurer.

La caractérisation des algorithmes recensés facilite la compréhension des différences subtiles qui existent entre eux et, par conséquent, rend la tâche de sélection d'algorithmes plus simple.

La seconde sous-phase d'acquisition des connaissances consiste à fournir les abstractions de chaque algorithme. Ces abstractions facilitent leur compréhension par des professionnels. La dernière sous-phase permet la génération d'une taxonomie de ces algorithmes afin d'expliciter leurs différences et ressemblances. Cette génération peut se faire en appliquant un processus de généralisation sur les abstractions d'algorithme. Chaque constituant de la taxonomie est ainsi décrit par une abstraction.

La généralisation de micro-données étant l'une des techniques les plus utilisées pour la publication de données, nous l'avons considéré dans le premier incrément du processus d'OPAM. Nous avons, par conséquent, exploité les contributions des chapitres 3 et 4 pour répondre aux questions informelles 2 à 6 et construire le noyau de

concepts et de relations d'OPAM que nous avons structuré sous la forme d'un diagramme de classes UML. La section qui suit décrit cette phase de conceptualisation. La réponse à la question 1 sera complète dès lors que toutes les techniques auront été étudiées. Cependant, un début de réponse est fourni dans la synthèse du chapitre 2 de cette thèse.

4. Phase de conceptualisation

Récolter les connaissances facilitant la compréhension de ce que fait chaque algorithme d'anonymisation de micro-données ainsi que son fonctionnement est l'objectif principal de la phase précédente. Chaque artefact produit rassemble des connaissances qui sont traduites, au cours de la phase de conceptualisation, en concepts et relations avant d'être structurées au sein d'un diagramme de classes UML. La fusion de ces diagrammes constitue le modèle conceptuel de notre ontologie. Nous avons choisi une conceptualisation d'OPAM à l'aide de ce type de diagramme en raison de ses avantages largement reconnus pour cette phase de construction d'ontologies (Pinet, Roussey, Brun & Vigier, 2009). Il est d'ailleurs largement utilisé par les concepteurs d'ontologies. En outre, s'agissant d'une représentation semi-formelle, les utilisateurs sont susceptibles de se familiariser davantage avec ce modèle qu'avec OWL dont la représentation est purement textuelle. Ainsi, il est plus pertinent pour la vérification de la portée ontologique.

A l'issue de la sous-phase «réalisation d'un état de l'art des algorithmes», nous avons opté pour une caractérisation des algorithmes reposant sur la description de leurs quatre constituants : leurs prérequis, leurs entrées, leurs sorties ainsi que leur processus. Cette caractérisation nous paraît la plus générique car elle est applicable à tous les algorithmes quelle que soit la technique. De plus, elle donne une première idée sur son fonctionnement. Comme le montre le graphe de la **Figure 65**, les premiers concepts et relations à prendre en compte dans notre ontologie sont ainsi recensés. Dans ce graphe, le lien « owns »/« is owned by » est étiqueté par un label indiquant la valeur prise par une caractéristique (d'un algorithme) de type entrée, sortie ou processus. La **Figure 66** est une instanciation du graphe de la **Figure 65** pour l'algorithme Datafly.

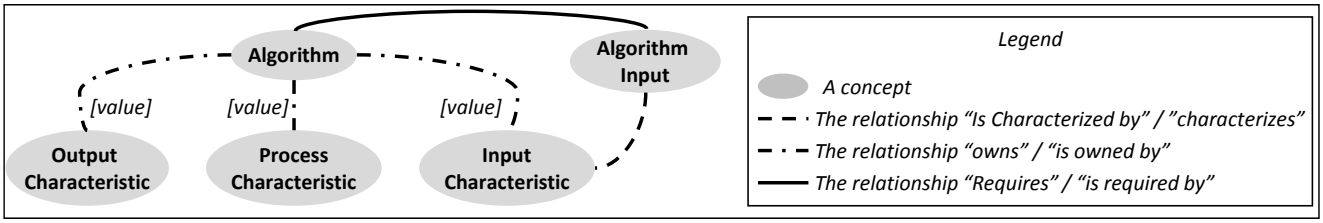


Figure 65. Les concepts et les relations issus de la caractérisation des algorithmes

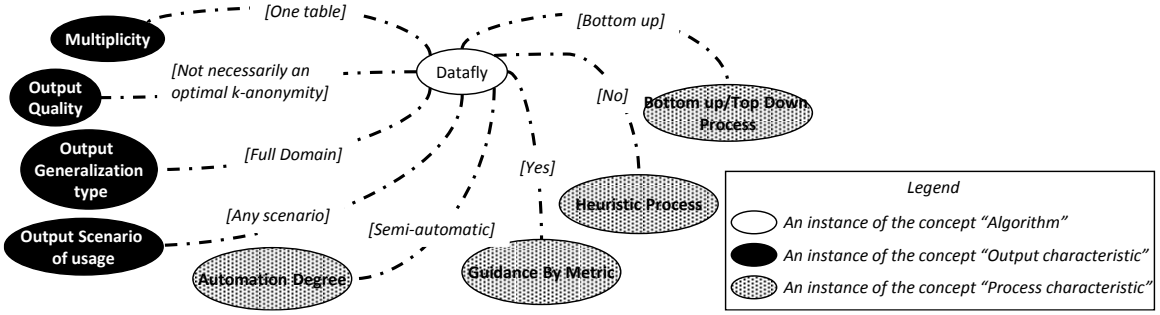


Figure 66. Instanciation partielle des concepts présentés à la **Figure 65**

Ainsi, la **Figure 67** fournit la portion du diagramme de classes UML structurant ces concepts et relations. Un algorithme décrit par une spécification requiert des entrées (« inputs »). Il possède (« owns ») un certain nombre de caractéristiques décrivant ses entrées, ses sorties mais aussi son processus.

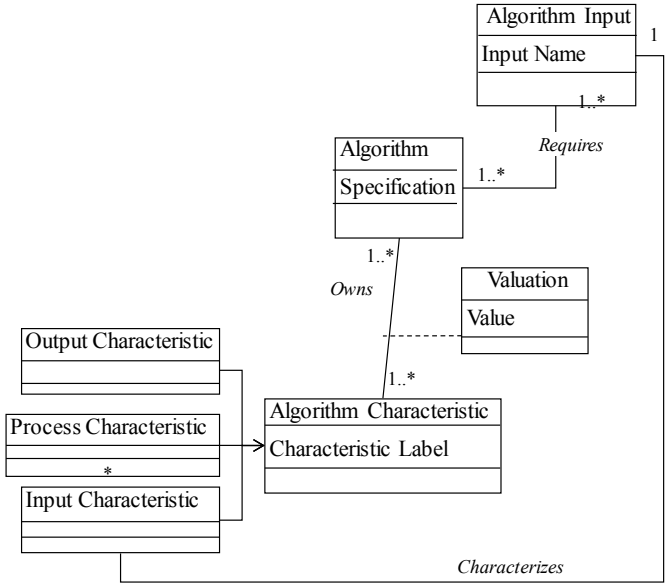


Figure 67. Les concepts et relations décrivant les algorithmes de généralisation (extrait)

La **Table 3** fournit l'extension des concepts de la **Figure 67** obtenue à l'issue du premier incrément, c'est-à-dire à la fin de l'étude de la technique d'anonymisation par généralisation.

Table 3. Ensembles d'extensions de concepts

Concept	Extension
Algorithm	{Samarati, Incognito, Datafly, Bottom up generalization, TDS, Median Mondrian, LSD Mondrian}
Algorithm Input	{Relational database, k, QI, set of generalization Hierarchy for attributes of the QI, classification constraint, Target attributes}
Input characteristic	{size data set, Number of correlated attributes, Number of allowed suppression}
Process Characteristic	{automation Degree, Bottom up/top down process, Guidance by metric, type of process}
Output Characteristic	{Multiplicity of outputs, Quality of outputs, Type of generalization applied to the outputs, Scenario of usage of the outputs}

De même, la sous-phase de «réalisation d'un état de l'art des algorithmes» nous permet d'obtenir une spécification des algorithmes relevant d'une technique. Ces spécifications sont, par la suite, abstraites puis généralisées. Une abstraction résultante associée à un algorithme aura permis de comprendre, dans le détail, son fonctionnement. L'abstraction par généralisation nous a permis de mettre en évidence les ressemblances et différences entre algorithmes. Elle fournit une taxonomie faisant ressortir éventuellement la décomposition de la technique en sous-techniques, une abstraction pour la technique étudiée et une abstraction pour chaque sous-technique découverte.

Ces deux sous-phases appliquées, au premier incrément, à la technique de généralisation, exhibent les concepts et relations représentés par le graphe de la **Figure 68**.

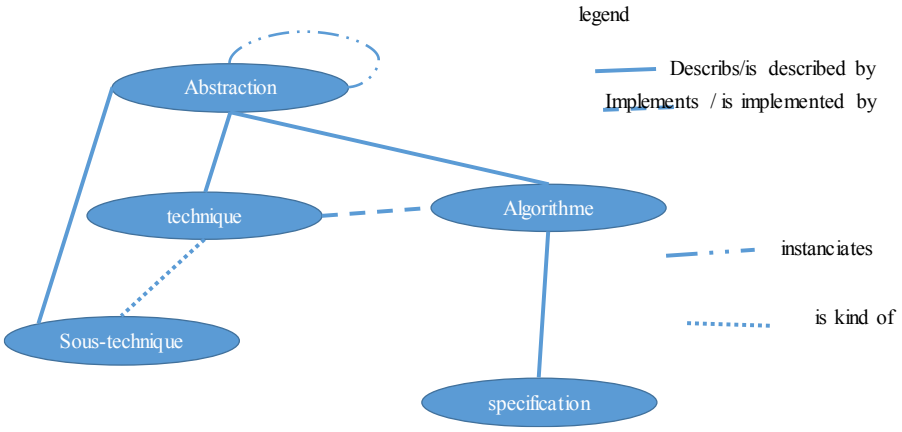


Figure 68. Les concepts et les relations issus de la phase d'abstraction

Ainsi, la **Figure 69** fournit la portion du diagramme de classes UML structurant ces concepts et relations. Le processus d'anonymisation, qu'il s'agisse de l'application d'une technique ou du déroulement d'un algorithme, peut être décrit au moyen d'une abstraction.

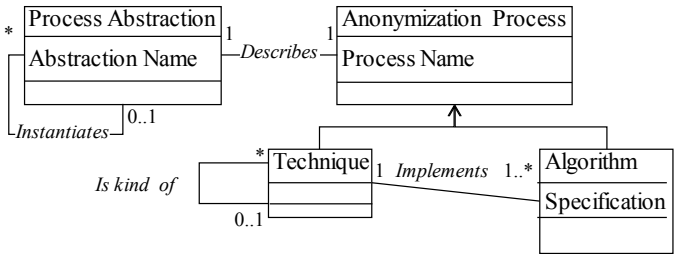


Figure 69. Diagramme UML du processus d'anonymisation (extrait)

La **Table 4** fournit l'extension des concepts de la **Figure 69**, obtenue à l'issue du premier incrément c'est-à-dire à la fin de l'étude de la technique d'anonymisation par généralisation.

Table 4. Instanciation du diagramme UML pour l'anonymisation par généralisation

Concept	Extension
Technique	generalization
Algorithm	{Samarati, Incognito, Datafly, µ-argus, median Mondrian, LSD Mondrian, Info Gain Mondrian, TDS, bottom up generalization}
Process abstraction	**Figure 39**, **Figure 40**, **Figure 41**, **Figure 42**, **Figure 43**, **Figure 44**, **Figure 45**, **Figure 46** et **Figure 47**

L'objectif ultime d'une anonymisation est de fournir un jeu de données à la fois utile et sécurisé. En se posant la question du « comment», on a pu identifier, par raffinement, l'ensemble des sous-objectifs qui permettront d'atteindre cet objectif ultime. Ainsi, nous avons fourni, dans notre synthèse du chapitre 3, une hiérarchie de buts (**Figure 35**) de laquelle nous extrayons des concepts et relations que nous représentons sous forme de graphe dans la **Figure 70**. Ainsi, les objectifs se trouvant aux feuilles de l'arborescence, que nous nommons « besoins d'anonymisation » peuvent être vus comme des propriétés qu'un professionnel désire maintenir ou obtenir dans son jeu de données après anonymisation. Ces propriétés, comme on a pu le voir aussi dans le chapitre 3 de cette thèse, sont implicitement satisfaites par certaines techniques, de par leur principe, et par conséquent par les algorithmes qui implémentent ces techniques. Pour d'autres techniques, la satisfaction dépend à la fois du jeu de données fourni et de l'algorithme appliqué sur ces données. Dans ce cas, on peut évaluer leur degré de satisfaction, suite à l'application de cet algorithme sur un jeu de micro-données, à l'aide de métriques. Le professionnel peut alors imposer un seuil de satisfaction à atteindre. Ce seuil constitue, dans ce cas, une contrainte imposée au procédé d'anonymisation.

Notre état de l'art a mis en évidence l'existence de plusieurs métriques pour une même propriété.

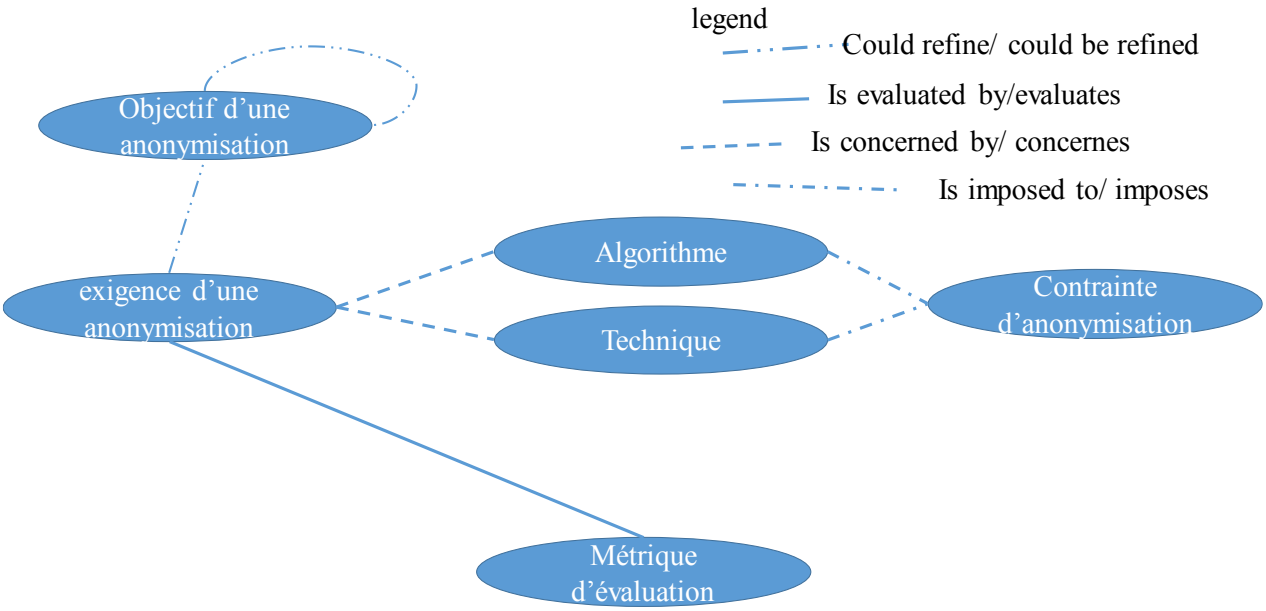


Figure 70. Relations entre les concepts de l'anonymisation

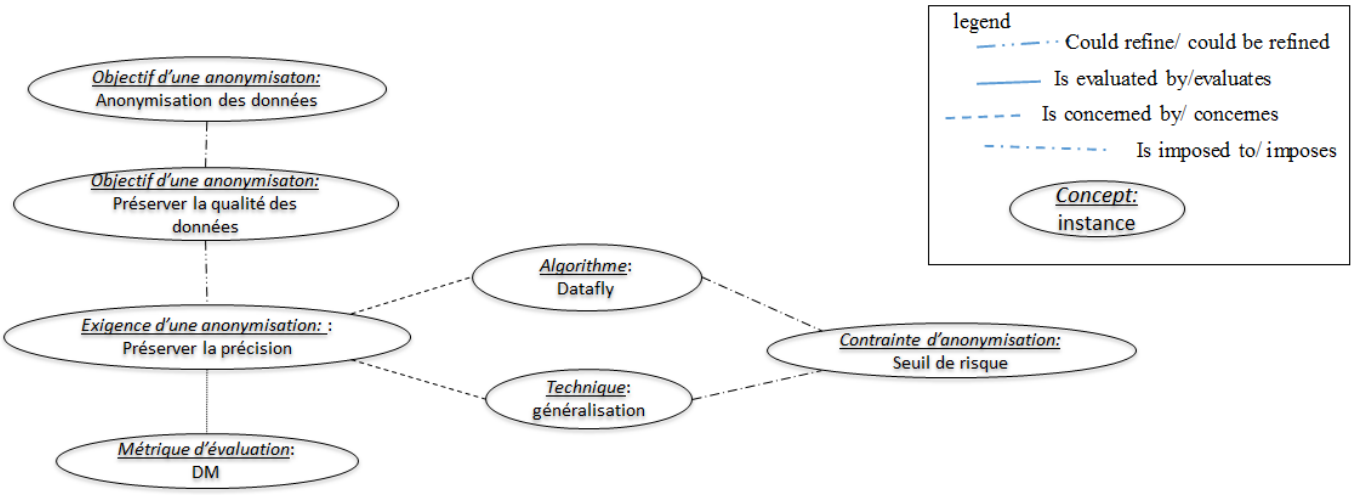


Figure 71 est une instanciation de ce graphe pour l'anonymisation par généralisation.

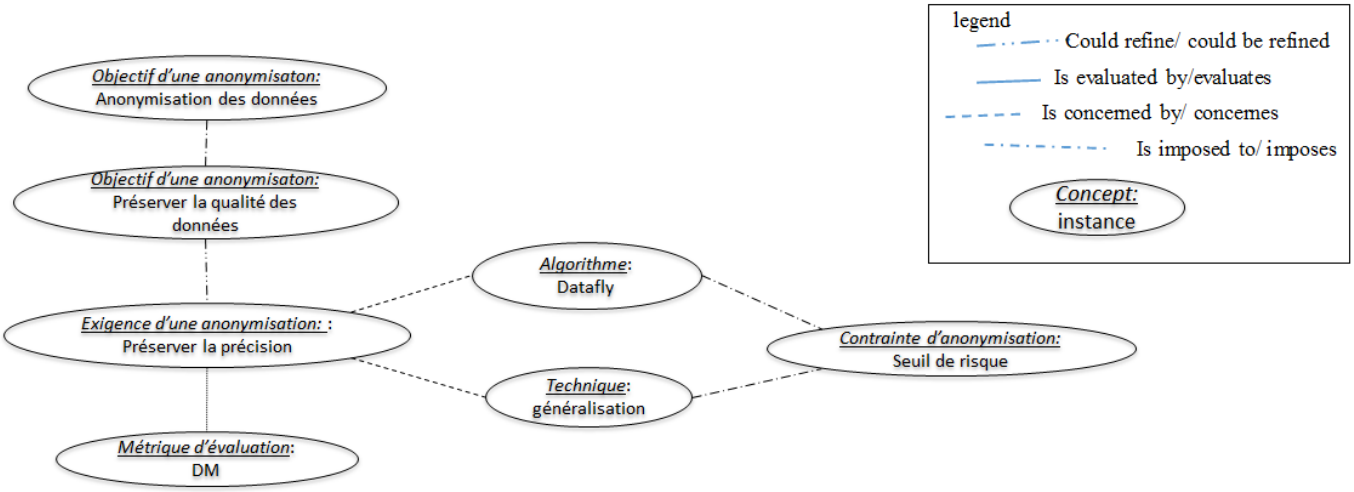


Figure 71. Instanciation partielle des concepts présentés à la **Figure 70**

A partir de ce recueil de concepts et de relations, nous pouvons construire la portion de diagramme UML suivant (voir **Figure 72**).

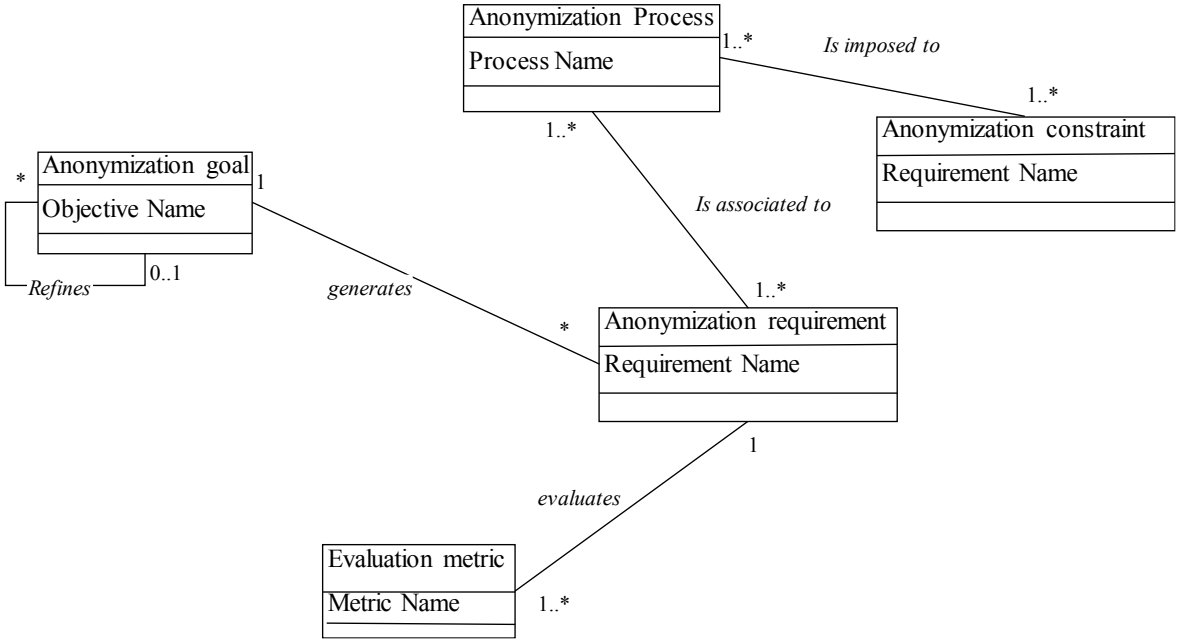


Figure 72. Diagramme UML du processus d'anonymisation

Le dernier artefact produit par notre état de l'art et qui constitue la réponse à la dernière question du cahier des charges d'OPAM[8] est le recueil de toutes les données issues d'évaluations expérimentales jusqu'à présent publiées sur les algorithmes d'anonymisation de micro-données. Ces données ont été recueillies pour les algorithmes de généralisation. Une abstraction sous forme de modèle multidimensionnel a été fournie au chapitre 3 de cette thèse (**Figure 36**). Notre étude de l'état de l'art relatif à d'autres techniques d'anonymisation nous permet de confirmer l'existence d'évaluations expérimentales pour d'autres algorithmes implémentant des techniques autres que la généralisation. Chaque donnée expérimentale fournie correspond à une évaluation d'un algorithme donné vis-à-vis d'un critère donné que nous avons nommé ci-avant « exigence d'anonymisation ». Cette évaluation est faite par application d'une métrique spécifique associée à cette exigence. Elle est contrainte par les valeurs que peuvent prendre certains paramètres d'entrée de l'algorithme visé et par les valeurs que peuvent prendre certains éléments descriptifs d'un jeu de données. Par conséquent, pour favoriser la généricité dans l'expression de données expérimentales, nous proposons d'intégrer dans OPAM la liste des concepts et des relations décrites dans la **Figure 73**.

[8] « Dispose-t-on de statistiques d'évaluation des procédés existants qui pourraient aider dans le choix d'un procédé pour un jeu de micro-données donné ? »

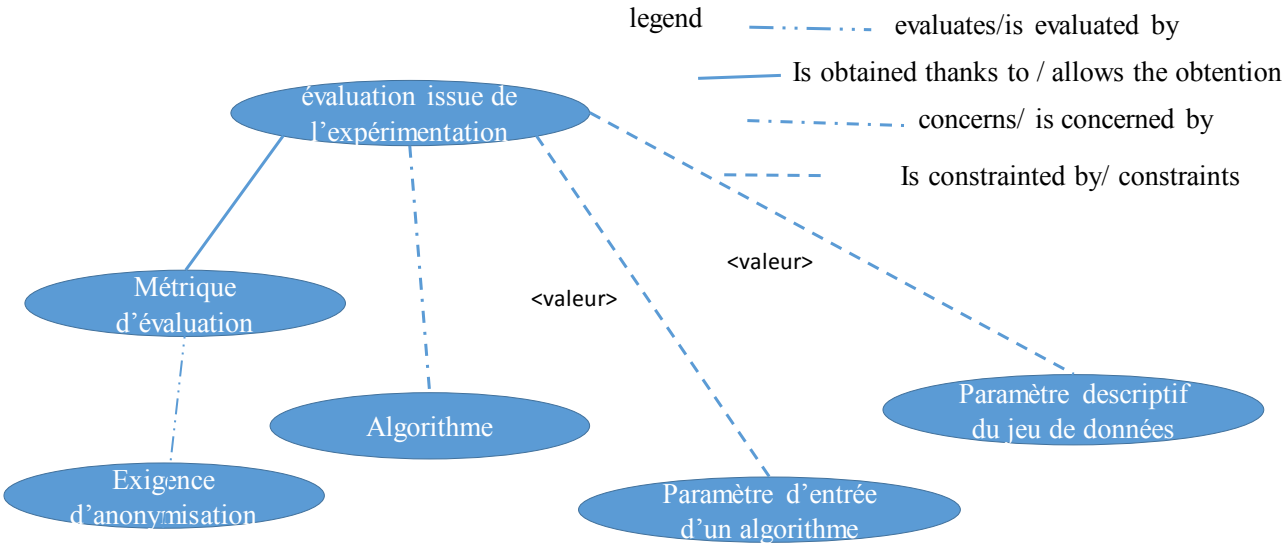


Figure 73. Les concepts et les relations extraites des évaluations expérimentales d'algorithmes

La **Figure 75** présente une instanciation du graphe de la **Figure 73** pour la donnée d'évaluation expérimentale extraite du **Tableau 40** du chapitre 3. Cette évaluation concerne l'algorithme Datafly. L'exigence d'anonymisation par rapport à laquelle il a été évalué est la « précision » moyennant la métrique DM. La précision selon DM pour Datafly est de 500 000 000 lorsque le paramètre k de Datafly est 5 et que le jeu de donnée est dense et dispose d'un QI comportant 3 attributs.

Critère d'évaluation	métrique	algorithme	input 'k'	nombre QI	distribution	Valeur
Précision	DM	dataFly	5	3	dense	500 000 000

Figure 74. Données d'évaluation extraites du **Tableau 40**

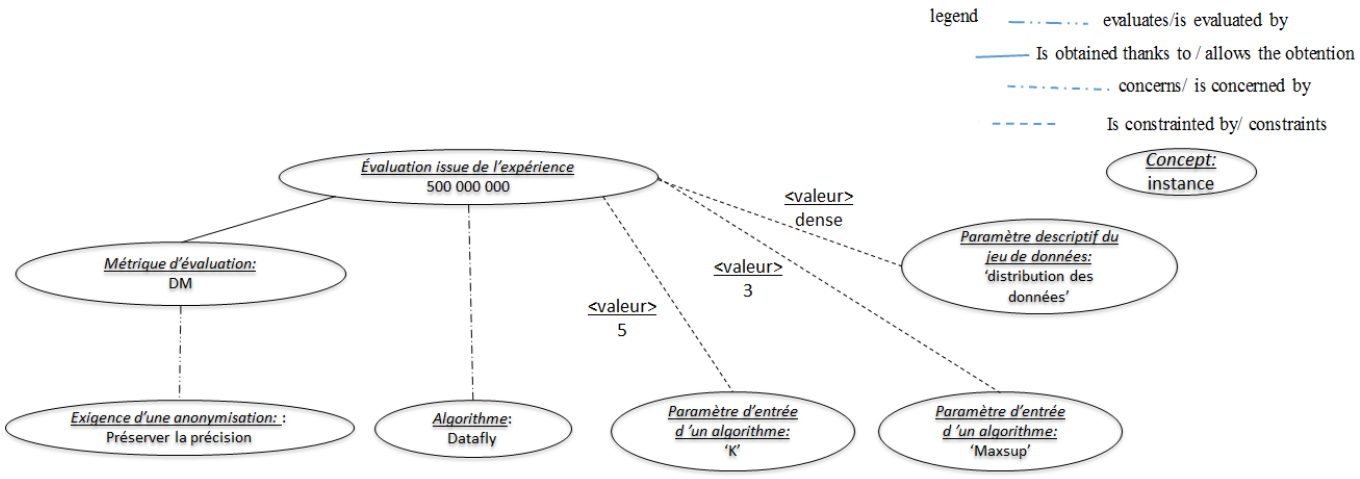


Figure 75. Instanciation du graphe de la **Figure 73** pour la donnée de la **Figure 74**

La portion de diagramme UML structurant les concepts et relations représentés par le graphe de la **Figure 73** est le suivant (Figure 76).

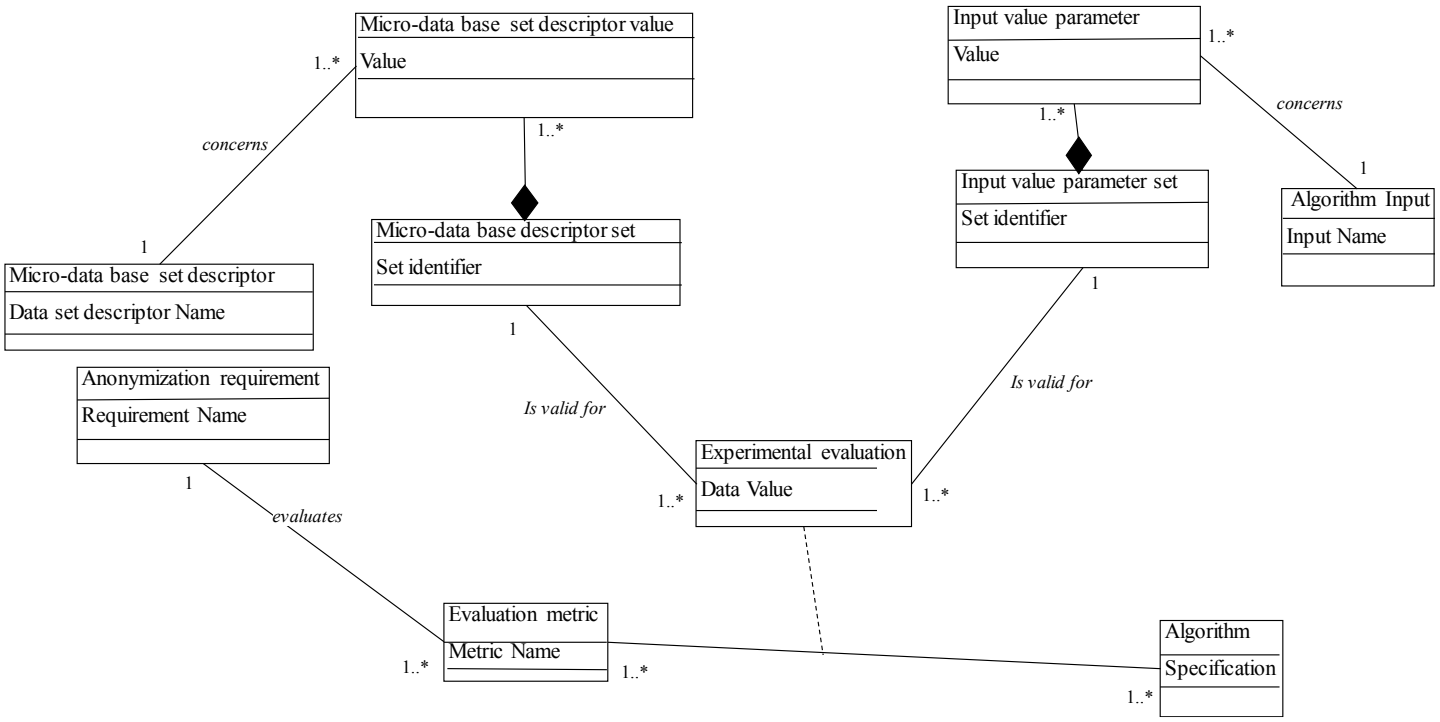


Figure 76. Diagramme UML des données expérimentales relatives à l'anonymisation

Une fois les portions de modèles conceptuels relatifs aux artefacts produits et analysés pour l'extraction des concepts, nous procédons à leur fusion. La **Figure 77** fournit le modèle conceptuel global d'OPAM après fusion.

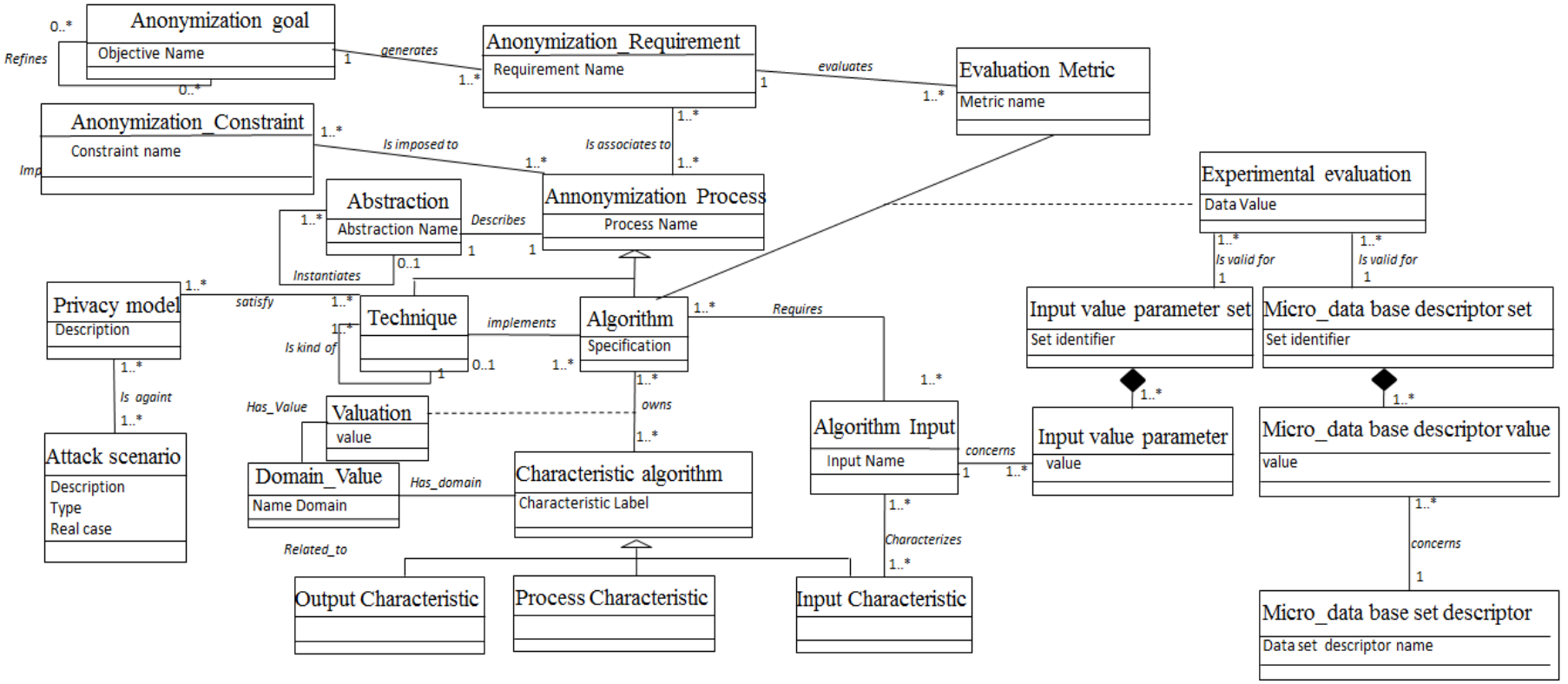


Figure 77. Modèle conceptuel global d'OPAM après fusion

Ce modèle, dans un souci de complétude, a été enrichi de la portion de diagramme UML issue de notre synthèse du chapitre 2. Cette dernière nous a permis de réunir les informations concernant les modèles de protection de la vie privée qu'est censée satisfaire une technique. Pour chacun des modèles de protection, sont recensés des scénarios d'attaque.

5. Formalisation et implémentation d'OPAM

La transition de la conceptualisation vers la formalisation d'OPAM peut être effectuée en s'appuyant sur les règles de transformation proposées par l'OMG dans l'ODM (OMG, 2009). Ces règles de transformations sont mises en œuvre dans l'environnement UML « Papyrus ». Ce dernier repose sur le cadriciel EMF (Steinberg et al. 2008) qui fournit un ensemble de composants dédiés à la création et la manipulation de modèles, ainsi que des générateurs de code fondés sur ceux-ci.

L'ontologie a été créée, sous le format OWL/XML en utilisant le projet ATL nommé « UML2OWL » de Papyrus. Ce projet exploite deux procédés de transformation disponibles: le procédé M2M (modèle vers modèle) et le procédé M2T (modèle vers texte) (Juan 2009). Ces deux procédés adoptent les principes de l'IDM (Ingénierie Dirigée par les Modèles).

Dans UML2OWL, deux transformations successives sont mises en œuvre (voir **Figure 78**). La première est une transformation M2M qui, partant d'une instanciation du méta-modèle UML, fournit une instanciation du méta-modèle OWL. Dans notre cas, l'instanciation du méta-modèle source correspond à notre conceptualisation d'OPAM. La seconde transformation permet de générer le document OWL.XML correspondant à notre ontologie formalisée.

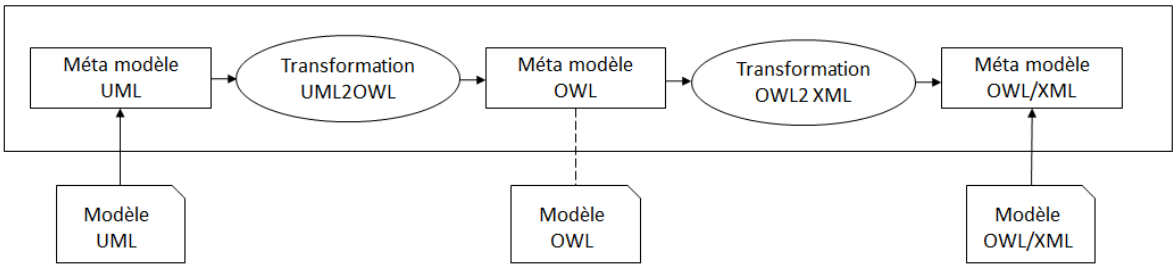


Figure 78. Schéma de transformation d'UML à OWL/XML

La **Figure 79** donne un extrait du document OWL/XML associé à OPAM.

```
<?xml version = '1.0' encoding = 'ISO-8859-1' ?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#" xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema
  <owl:Ontology rdf:about="AnoOontology"/>
  <owl:Class rdf:ID="Anonymization_Process">
    <rdfs:label>Anonymization_Process</rdfs:label>
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty rdf:resource="#Anonymization_Process.Anon_Pros_Label"/>
        <owl:cardinality  rdf:datatype="">1</owl:cardinality >
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty rdf:resource="#Anonymization_Process.dB_caracteristic"/>
        <owl:cardinality  rdf:datatype="">1</owl:cardinality >
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty rdf:resource="#Anonymization_Process.imposed_in"/>
        <owl:minCardinality  rdf:datatype="">1</owl:minCardinality >
      </owl:Restriction>
    </rdfs:subClassOf>
```

Figure 79. Extrait de l'ontologie OTA en OWL

Nous avons stocké OPAM dans le triple store SESAME. En effet, un triple store est une base de données spécialement conçue pour le stockage et la récupération des données RDF (Resource Description Framework) ou OWL à travers des requêtes sémantiques. Contrairement à une base de données relationnelle, un triple store ne stocke qu'un seul type de données : le triplet. Ce dernier est une entité de données composée de sujet-prédicat-objet.

SESAME est un triple store « open source ». Il a été créé dans le cadre d'un projet web sémantique «On-To-Knowledge», qui a duré de 1999 à 2002. Il s'agit d'une bibliothèque qui peut être intégrée dans n'importe quelle application Java.

SESAME propose une boîte à outils entièrement modulaire et des API Java qui peuvent être connectées à tous ses modules. En effet, il contient une API simple pour les analyseurs RDF, une couche de stockage et d'inférence, une API de référentiel pour la gestion des données RDF et un serveur HTTP pour accéder aux référentiels SESAME via le protocole HTTP. Le langage de requête utilisé dans SESAME est SPARQL.

6. Conclusion

Ce chapitre a été dédié à la conception et la mise en œuvre d'OPAM, une ontologie de domaine pour l'anonymisation de micro-données à des fins de publication.

OPAM sert à capter, représenter et modéliser des connaissances de telle sorte à ce que ces dernières soient non seulement comprises mais aussi interprétées aussi bien par les humains que par les machines.

Ces connaissances peuvent aider un professionnel, chargé de l'anonymisation de micro-données, à choisir un algorithme d'anonymisation et à concrétiser son processus de dé-identification. Elles constituent les réponses à des questions qu'il peut se poser lorsqu'il s'engage dans un processus d'anonymisation.

Ces connaissances sont aussi exploitables, comme le montre la suite, par l'approche guidée d'anonymisation de micro-données que nous proposons dans le cadre de cette thèse.

OPAM, dans son état actuel, ne couvre que les connaissances associées à la technique d'anonymisation et à ses algorithmes. Néanmoins, l'approche incrémentale de construction et de peuplement d'OPAM proposée dans ce chapitre garantit son évolutivité pour prendre en compte les autres techniques.

Chapitre 6 Approche guidée pour l'anonymisation de micro-données

L'anonymisation est un processus complexe. Sa complexité est due au fait qu'elle tente de satisfaire deux objectifs contradictoires que sont : l'utilité des données (c'est-à-dire leur qualité) et leur sécurité (c'est-à-dire leur confidentialité). Par conséquent, les éditeurs de données sont toujours à la recherche d'une solution qui réponde au mieux à la confidentialité et à l'utilité de leurs données. Leur solution émerge après des prises de décision à différentes phases du déroulement de leur tâche. En effet, ils sont amenés, entre autres, à sélectionner un algorithme d'anonymisation, à opter pour un paramétrage adéquat de cet algorithme et à juger de la qualité du rendu après application du procédé ainsi paramétré. Ils sont donc engagés dans un processus de décision qui s'appuie sur leur connaissance du domaine.

Les outils existants, comme on a pu le constater dans notre état de l'art, de par, d'une part leur opacité et, d'autre part, leur manque de guidage suggestif dans le choix et le paramétrage d'algorithmes, ne dé-complexifient pas cette activité pour un professionnel ayant une faible expertise dans le domaine. D'un point de vue académique, nous avons aussi constaté l'absence d'approches guidées pour l'anonymisation bien que la littérature abonde d'articles de recherche sur les algorithmes d'anonymisation.

Ces constats ont motivé notre démarche de mise à disposition des professionnels d'une ontologie de domaine pour l'anonymisation de micro-données ainsi que d'une approche guidée s'appuyant sur cette ontologie. L'ontologie produite au chapitre précédent et que nous avons nommée OPAM, permet de capitaliser les connaissances du domaine. Cependant, elle ne stocke qu'une portion d'expertise du domaine. En effet OPAM, n'a été, pour l'instant, instanciée que par les connaissances récoltées sur la technique de généralisation de micro-données. Par conséquent, l'approche, que nous proposons dans ce chapitre et que nous nommons MAGGO (Méthodologie pour une Anomymisation par Généralisation Guidée par une ontologie), sert de guide pour un professionnel dans sa prise de décision lors d'une anonymisation par généralisation de micro-données. Cela n'enlève rien à la généricité de notre approche. En effet, elle peut être instanciée pour une autre technique.

Ce chapitre présente notre approche MAGGO.

1. Présentation générale de MAGGO

L'anonymisation de données est une des mesures de sécurité qui peuvent être préconisées dans le cadre de la protection de la vie privée. Dès lors que cette mesure est décidée, le responsable de l'anonymisation doit concevoir et exécuter un processus de brouillage. Au-delà du fait qu'il doit a) repérer les données identifiantes, quasi-identifiantes et sensibles, proposer des techniques appropriées avec une orchestration adéquate, etc. , il doit aussi, pour chaque technique, identifier l'algorithme à appliquer, proposer un paramétrage reflétant ses

besoins et évaluer la qualité des données en termes d'utilité et de sécurité en se conformant au cahier des charges de l'anonymisation. Ce processus est compliqué car il comprend plusieurs points de décisions clés dont la qualité affecte le résultat final. Il exige du responsable de l'anonymisation, en l'absence d'aide cognitive, une grande maîtrise du domaine. Offrir une telle aide sur la totalité du processus demanderait des efforts considérables compte tenu de la variété des données susceptibles d'être brouillées (micro-données, données liées, données géographiques, etc) et de la diversité des techniques existantes et des algorithmes implémentant ces techniques. Au cours de notre recherche, nous avons souhaité contribué modestement en ne considérant qu'une portion du processus d'anonymisation, une technique et un type de donnée. En effet, nous offrons une approche guidée permettant, compte tenu d'un contexte d'anonymisation (cahier des charges) de choisir l'algorithme de généralisation de micro-données le plus proche des exigences du cahier des charges et de l'exécuter. La notion de «plus proche» est liée au fait qu'une anonymisation parfaite, c'est-à-dire satisfaisant un très haut niveau d'exigence de sécurité et d'utilité en même temps, n'existe pas. Un compromis entre ces deux exigences est à prévoir du fait de leur caractère contradictoire.

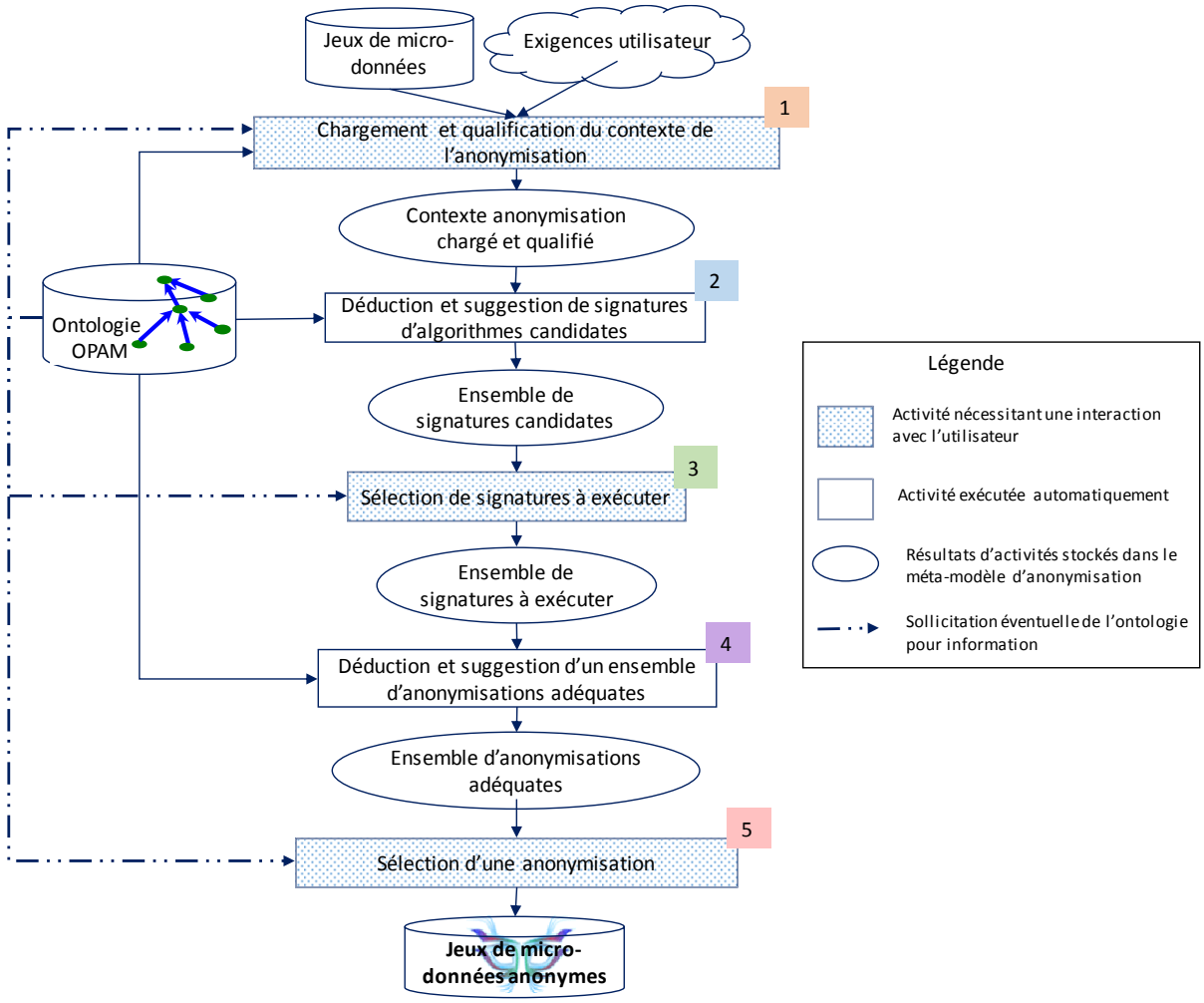


Figure 80. Les grandes étapes de MAGGO

MAGGO telle que représentée dans la **Figure 80** est un enchaînement de 5 étapes.

La première étape de MAGGO permet de spécifier le cahier des charges de l'anonymisation à réaliser. Le contexte de l'anonymisation sur lequel se fonde le processus pour fournir une solution de brouillage adéquate est alors décrit. Cette description est faite conjointement avec l'utilisateur qui aura à fournir son jeu de données et à préciser ses attentes.

La seconde étape offre, à l'utilisateur, une aide au choix et au paramétrage d'algorithmes d'anonymisation par généralisation. Elle lui propose un ensemble de signatures d'algorithmes candidates (algorithmes candidats avec pour chacun les valeurs de paramètres qu'il pourrait accepter en entrée) compte tenu du cahier des charges établi. De cet ensemble, l'utilisateur pourra sélectionner lors de l'étape suivante un sous-ensemble qu'il souhaiterait faire exécuter, par le processus, sur le jeu de données original afin de décider de la solution d'anonymisation à retenir. Ce sous-ensemble exécuté, à l'avant dernière étape, sur le jeu de données fourni initialement, produit des solutions que le processus évalue d'un point de vue sécurité et qualité. Les évaluations fournies à l'utilisateur lui permettront de retenir une solution parmi celles évaluées. Elles reposent sur des métriques disponibles dans OPAM.

Pour aider l'utilisateur dans la spécification du contexte de son anonymisation, dans la sélection de signatures et de solutions d'anonymisation, le processus met à la disposition de l'utilisateur des connaissances complémentaires nécessaires le rendant apte à participer à la prise de décision.

Ainsi, à chacune de ses étapes, MAGGO fait intervenir des connaissances expertes au vu d'un guidage suggestif ou informatif (**Figure 81**). Ces connaissances sont rendues disponibles via OPAM.

Etapes Approche d'anonymisation	**Guidage**
Chargment et qualification du contexte de l'anonymisation	informatif
Déduction et suggestion d'un ensemble de signatures candidates	**suggestif**
Sélection de signatures à exécuter	informatif
Déduction et suggestion d'un ensemble d'anonymisations adéquates	**suggestif**
Sélection d'une anonymisation	informatif

Figure 81. Type de guidage à chaque étape de l'approche

Le guidage de MAGGO peut être qualifié à la fois d'incrémental et d'interactif. Il est incrémental dans le sens où il est introduit à différents points de décisions clés tout au long du processus. Il est interactif car il fait participer l'utilisateur dans la prise de décision.

La notion de méta-modèle joue un rôle central dans notre approche. En effet, alors que l'ontologie met à la disposition de l'utilisateur et de l'approche les connaissances nécessaires à l'anonymisation, le méta-modèle d'anonymisation réunit les abstractions conceptuelles des artefacts cibles et sources de notre approche. Il est représenté à la **Figure 82**.

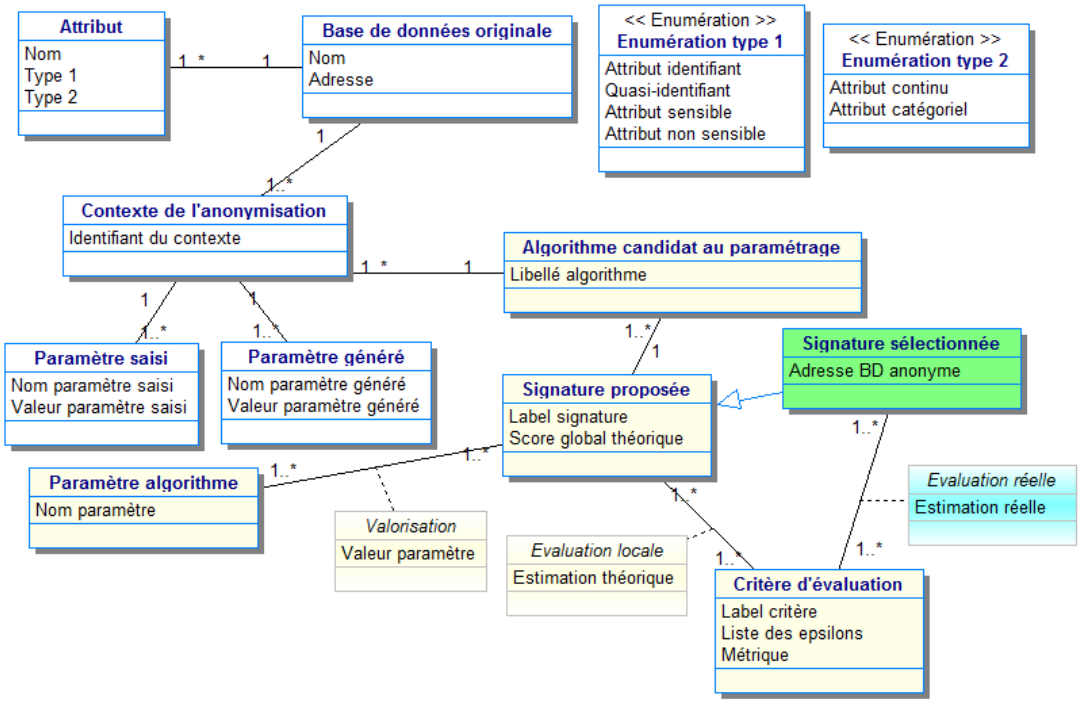


Figure 82. Le méta modèle du processus d'anonymisation

Ainsi, l'exécution de la première phase de notre approche, pour une anonymisation par généralisation d'un jeu de données original donné, permet d'instancier notre méta-modèle par les données relatives au cahier des charges de l'utilisateur. Cette instanciation correspond à une description du contexte de l'anonymisation ainsi qu'à sa qualification. Un enrichissement du modèle par des données complémentaires issues de chacune des différentes phases est effectué à chacune des phases de MAGGO. Dans la **Figure 82** représentant ce méta-modèle, les concepts concernés par une phase de MAGGO sont représentés par une même couleur.

De plus, MAGGO exploite, pour l'enrichissement du méta-modèle, deux techniques statistiques : la technique Analytical Hierarchy Process (AHP) et la technique de régression. La première est une technique d'aide à la décision multicritère utilisée par MAGGO, aux étapes 2 et 4 pour évaluer les résultats qu'elle soumet à l'utilisateur pour une éventuelle sélection. La seconde technique est une technique que MAGGO utilise sous la forme d'un apprentissage supervisé afin de prédire la valeur d'un critère d'utilité ou de sécurité compte tenu de données expérimentales disponibles dans OPAM. Ces deux techniques ne sont pas les seules techniques disponibles. Il n'est pas exclu de les remplacer par d'autres dès lors que ces dernières répondent au besoin de l'étape.

Avant de présenter de façon détaillée les étapes 1, 2 et 4 de MAGGO avec à chaque fois la partie du méta-modèle alimentée ou sollicitée, nous rappelons brièvement, dans la section qui suit, le principe de chacune des deux techniques statistiques utilisées.

2. Rappels sur l'approche AHP et la régression statistique

2.1 L'approche AHP

AHP est une approche pour le guidage de processus complexes de prise de décision à critères multiples. Ce dernier, très demandé par les organisations, relève du domaine de l'aide à la décision (roy 1999) et plus particulièrement d'un de ses sous-domaines, communément connu sous le nom de MCDM (Multiple criteria decision making). Ce sous-domaine fait référence à la prise de décision en présence de critères multiples, généralement contradictoires.

Le MCDM permet, pour un problème pour lequel un choix de solution est attendu, d'évaluer plusieurs alternatives selon un ensemble de critères. Une alternative nommée aussi action constitue l'objet de la décision ou ce sur quoi porte l'aide à la décision (Roy, 1985). L'idée de base est de considérer tous les critères entrant en compte dans la prise de décision, de leur attribuer un poids lié à leur importance relative, de noter chaque alternative (action) par rapport à tous les critères et finalement d'agréger ces résultats (Mena 2000). Cette agrégation vise à obtenir une évaluation globale des actions permettant d'aboutir à une recommandation finale.

Plusieurs techniques offrant des aides pour le MCDM sont proposées dans la littérature. Une revue de ces techniques et de leur application est fournie dans (Mardani et al. 2015). Elles peuvent être classées en techniques compensatoires ou non-compensatoires. Les techniques non compensatoires ne permettent pas de compromis entre critères. Une valeur défavorable pour un critère ne peut pas être compensée par une valeur favorable pour d'autres critères. Les techniques compensatoires, dont fait partie AHP, permettent, quant à elles, d'établir des compromis entre la bonne performance d'un critère et la mauvaise performance d'un autre (Mrinmoy 2015). Plus précisément, AHP fait partie de la sous-catégorie de méthodes compensatoires fondées sur le calcul de score. Cette dernière en compte quatre. Elles sont toutes décrites dans (L. Xu et Yang 2001).

AHP, au même titre que les autres méthodes fondées sur le calcul de score, est caractérisée par le fait qu'elle procède à la sélection et l'évaluation d'une alternative sur la base de son utilité (c'est-à-dire sur la base de la préférence du preneur de décision). Elle a été développée par le mathématicien Thomas L. Saaty (Saaty, 1986,1990). Elle est mise en œuvre dans une grande variété d'applications académiques et professionnelles.

Pour prendre une décision sur un problème identifié, compte tenu d'un ensemble d'alternatives dont l'évaluation repose sur un ensemble de critères, AHP propose de structurer, dans un premier temps, le problème de décision en trois niveaux hiérarchiques. Le niveau le plus haut représente l'objectif de l'analyse (c'est-à-dire le problème de décision pour lequel AHP doit fournir une recommandation). Le niveau intermédiaire représente une hiérarchisation des critères de choix des objectifs. Le niveau le plus bas représente les alternatives ou actions, c'est-à-dire ce sur quoi porte l'aide à la décision. Chaque alternative est soumise à une évaluation selon les différents critères énoncés au niveau intermédiaire. Cette structuration aboutit à une hiérarchie sur laquelle va reposer l'analyse multicritère. Elle met en évidence les critères qui auront un impact sur la recommandation finale. La **Figure 83** est un exemple de hiérarchie liée à un problème d'achat de voiture pour lequel une famille

nommée Jones souhaite se positionner compte tenu des six modèles de voiture qu'elle considère. Son choix d'un modèle doit tenir compte des critères énoncés dans la hiérarchie. Certains de ces critères, tels que la capacité, sont décomposables en sous-critères. AHP, muni de cette hiérarchie et des préférences de la famille, peut recommander cette famille pour l'achat d'une parmi les six voitures. La voiture recommandée sera celle qui concilie au mieux tous les critères énoncés dans la hiérarchie.

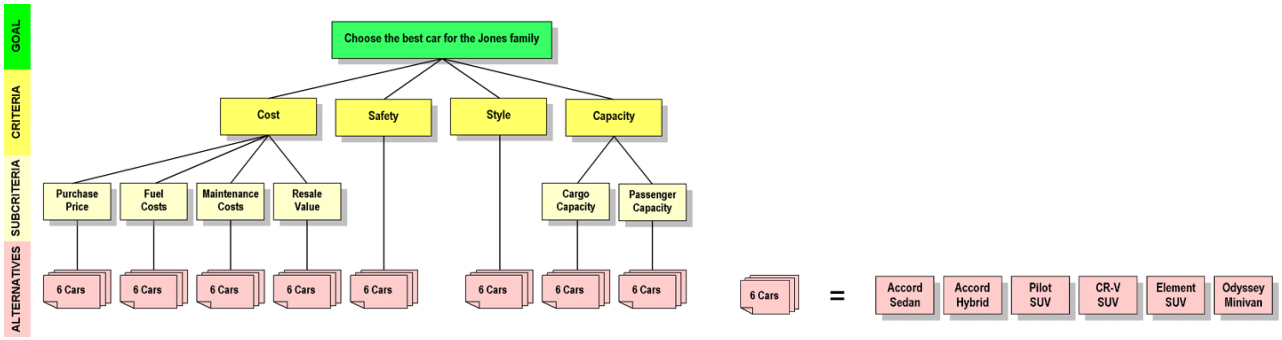


Figure 83. Exemple d'une hiérarchie AHP extrait de [10]

Une fois cette hiérarchie construite, AHP fait participer les décideurs dans le processus d'analyse. Elle émet l'hypothèse que l'humain est plus capable de faire des jugements relatifs que des jugements absolus (Kiker et al. 2005), surtout en présence de critères intangibles (c'est-à-dire des critères qualitatifs pour lesquels le décideur n'a pas de mesure lui permettant de le guider dans son classement des alternatives). Par conséquent, elle demande au décideur de fournir son jugement sur l'importance relative des éléments de la hiérarchie, c'est-à-dire tous les critères, sous-critères et alternatives. Ainsi, il est amené à comparer tout couple d'éléments de même niveau hiérarchique relativement à leur ascendant direct. Dans le vocabulaire du domaine, ce type de comparaison est nommé « comparaison par paires ». A titre d'exemple, la famille Jones aura, du point de vue du critère « prix», à comparer l'importance du sous-critère «prix de la voiture» avec l'importance du sous-critère «prix du fuel». Elle aura aussi à comparer, du point de vue de l'objectif «achat de la meilleure voiture», l'importance du critère «prix» avec l'importance du critère «sûreté». De même, elle aura à comparer, du point de vue du sous-critère « prix du fuel », l'alternative d'achat d'une voiture « Accord Sedan » avec l'alternative d'achat d'une voiture « Accord hybrid ». Pour aider le décideur dans son jugement, une échelle sémantique de valeurs, décrite dans la **Figure 84**, a été proposée dans (Saaty 2004).

Intensity of Importance	*Definition*	*Explanation*
1	Equal Importance	Two activities contribute equally to the objective
2	Weak or slight	
3	Moderate importance	Experience and judgement slightly favour one activity over another
4	Moderate plus	
5	Strong importance	Experience and judgement strongly favour one activity over another
6	Strong plus	
7	Very strong or demonstrated importance	An activity is favoured very strongly over another; its dominance demonstrated in practice
8	Very, very strong	
9	Extreme importance	The evidence favouring one activity over another is of the highest possible order of affirmation
Reciprocals of above	If activity *i* has one of the above non-zero numbers assigned to it when compared with activity *j*, then *j* has the reciprocal value when compared with *i*	A reasonable assumption
1.1–1.9	If the activities are very close	May be difficult to assign the best value but when compared with other contrasting activities the size of the small numbers would not be too noticeable, yet they can still indicate the relative importance of the activities.

Figure 84. L'échelle de comparaison sémantique de la méthode AHP

A l'issue de ces comparaisons par paires, des matrices de jugements sont établies et exploitées par AHP dans sa dernière phase. A titre d'exemple, le tableau (b) de la **Figure 85** fournit la matrice de jugements établie par AHP pour l'évaluation de la famille Jones selon les critères Cost, Safety, Style et Capacity. Cette évaluation est représentée dans le tableau (a) de la **Figure 85**.

Critère		Plus importan	Intensité
A	B		
Cost	Safety	A	3
Cost	Style	A	7
Cost	Cpacity	A	3
Safety	Style	A	9
Safety	Capacity	A	1
Style	Capacity	B	7

(a)

	Cost	Safety	Style	Capacity
Cost	1	3	7	3
Safety	0,33	1	9	1
Style	0,14	0,11	1	0,14
Capacity	0,33	1	7	1

(b)

Figure 85. Comparaison des critères

A titre d'exemple, dans le tableau (a), le critère Cost est évalué comme plus important que le critère Safety avec une intensité de 3. La matrice de jugement reprend cette évaluation dans la case se trouvant dans la ligne représentant Cost et la colonne représentant Safety. Dans la case dont la ligne représente Safety et la colonne représente Cost, la valeur 1/3 est inscrite.

A partir de cette matrice de jugement, AHP évalue le poids (score) à affecter à chaque élément de la hiérarchie. Son calcul est fondé sur la notion mathématique de vecteur propre (FINANCIER 2015).

Un poids représente, pour un critère, son importance dans l'environnement de la décision et, pour une alternative, sa capacité à atteindre l'objectif. La somme des poids des descendants d'un élément de la hiérarchie est égale au poids de leur ascendant direct. A titre d'exemple, la **Figure 86** fournit les scores calculés par AHP pour les quatre critères ci-dessus compte tenu de la matrice de jugements du tableau (b) de la **Figure 85** .

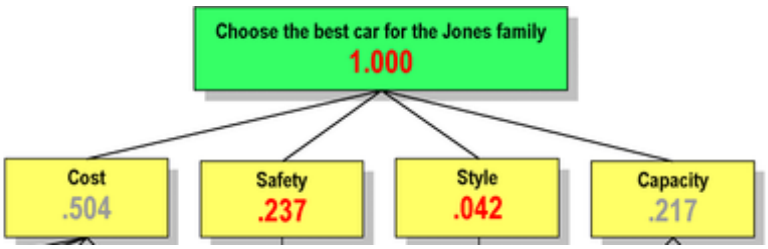


Figure 86. Les scores des critères calculés par AHP

Remarque : Une cohérence parfaite de l'ensemble des comparaisons par paires n'est pas toujours garantie dans la pratique. En effet, lorsque de nombreuses comparaisons sont effectuées, certaines incohérences peuvent surgir. A titre d'exemple, si la famille Jones avait évalué le critère Cost à égale importance avec le critère Capacity, on serait face à une incohérence puisque Cost est plus important que Safety et que Safety est d'égale importance par rapport à Capacity. Pour pallier cet inconvénient, AHP dispose d'une technique de vérification de la cohérence du décideur dans ses jugements. Cette technique calcule l'indice de cohérence IR qu'elle compare à un seuil minimum pour juger de l'acceptabilité des jugements du décideur. (Saaty, 1980) suggère de ne pas dépasser un seuil de cohérence de 10%.

2.2 La régression

L'apprentissage automatique désigne un ensemble de méthodes et d'algorithmes permettant d'extraire de l'information pertinente à partir des données ou d'apprendre un comportement à partir de l'observation d'un phénomène. En général, ce processus est associé à la possibilité de mesurer, d'une certaine façon, la qualité et la précision des résultats. L'apprentissage comprend deux grandes branches : l'apprentissage supervisé et l'apprentissage non supervisé. Dans le cas de l'apprentissage supervisé, la finalité est de déterminer une nouvelle sortie Y à partir d'une nouvelle entrée X. En effet, la base de données d'apprentissage est un ensemble de couples entrée-sortie $(x_n, y_n)_{1 \leq n \leq N}$ avec $x_n \in X$. L'objectif est de modéliser le lien entre X et Y pour faire des prévisions : connaissant X, on prédit Y. Les y_n sont les variables à expliquer et les x_n sont les variables explicatives. Lorsque les y_n prennent des valeurs discrètes, il s'agit d'un problème de classification, tandis que les y_n à valeurs réelles nous placent dans le cadre de la régression.

En apprentissage non supervisé, en revanche, il n'y a pas de sortie. Il s'agit alors de construire un modèle permettant de représenter au mieux les observations x_n de manière à la fois précise et compacte. Il s'agit pour

une méthode de diviser un groupe hétérogène de données, en sous-groupes de telle manière que les données considérées comme les plus similaires soient associées au sein d'un groupe homogène et qu'au contraire les données considérées comme différentes se retrouvent dans d'autres groupes distincts, l'objectif étant de permettre une extraction de connaissance organisée à partir de ces données.

Le but de notre approche est d'estimer la valeur d'un algorithme d'anonymisation par rapport à un critère selon les valeurs de ses inputs et le type de la distribution des données originales (uniforme ou dense). Par conséquent, nous nous plaçons dans le cadre des méthodes de régression. Plusieurs modèles de régression existent, notamment les réseaux de neurones, les règles d'association, les arbres de décisions et les « sequence mining ». Puisque nous disposons de jeux de données simples pour chaque algorithme par rapport à un critère et de variables explicatives mixtes (numérique et nominal), nous choisissons le modèle des arbres de décision pour la régression.

En effet, les arbres de régression sont l'une des structures de données majeures de l'apprentissage automatique (Breiman et al. 1984). Leur fonctionnement repose sur des heuristiques qui donnent des résultats remarquables en pratique. Ils permettent de construire des modèles de prédiction à partir des données. Les modèles sont obtenus par le partitionnement récursif de l'espace de données et l'adaptation d'un modèle de prédiction simple à l'intérieur de chaque partition. Par conséquent, le partitionnement peut être représenté graphiquement sous la forme d'un arbre de décision. Leur structure arborescente les rend également lisibles par un être humain, contrairement à d'autres approches où le prédicteur construit est une « boîte noire ». Ils sont conçus pour des variables à expliquer qui prennent des valeurs continues, avec une erreur de prédiction typiquement mesurée par la différence des carrés entre les valeurs observées et les valeurs prédites.

Parmi les avantages des arbres de décision, citons (Breiman et al. 1984) :

- la prise en compte simultanée de variables qualitatives et quantitatives,
- le fait qu'ils sont implémentés par des algorithmes très rapides tant en phase de construction des arbres que lors de la prédiction des nouveaux cas (un seul chemin est parcouru),
- le peu d'influence des données erronées,
- l'absence d'hypothèse concernant les données (modèle non-paramétrique),
- le fait que l'arbre de décision est un ensemble de règles logiques aisément interprétables qui permettent une meilleure compréhension du problème étudié.

3. Etape de chargement et qualification du contexte de l'anonymisation

Une anonymisation vise la prévention contre des attaques potentielles portant atteinte à la vie privée. Sa mise en œuvre nécessite, avant tout, la sélection d'une ou plusieurs techniques qui mettent en œuvre le modèle de protection censé contrer ces attaques. Pour le modèle de protection visé et la technique correspondante sélectionnée, se pose le problème de choix d'algorithmes pour mettre en œuvre l'anonymisation qui répond aux

attentes de son initiateur. Ces dernières constituent l'ensemble des exigences que doit satisfaire l'anonymisation. A ce titre, on peut considérer deux catégories d'exigence pour la réalisation d'une anonymisation de micro-données. La première catégorie rassemble les exigences indépendantes de la technique, c'est-à-dire l'usage projeté des données anonymes, le seuil de risque de ré-identification toléré, le taux de suppression à ne pas dépasser ainsi que la qualité minimale exigée. Cette dernière difficilement quantifiable pourrait être exprimée par l'importance relative accordée aux critères de qualité des données anonymes. La deuxième catégorie regroupe des exigences dépendantes de la technique choisie et qui influent sur le choix d'algorithme. Dans le cas de la technique de généralisation, le type de généralisation souhaité pourrait constituer une exigence spécifique à la technique de généralisation. A titre d'exemple, une anonymisation par généralisation pourrait être demandée pour un besoin de classification des données anonymes, tout en exigeant de ne pas l'accepter si elle engendre un risque de ré-identification de plus de 10% et un taux de suppression de plus de 5%. Le demandeur pourrait aussi préciser qu'il accorde plus d'importance à la sécurité qu'à la complétude des données anonymes. Enfin, il pourrait opter pour une généralisation multidimensionnelle (deux données identiques dans la table d'origine pourront être généralisées par des données différentes de la hiérarchie de généralisation). Quand bien même ces informations sont rendues disponibles, elles ne suffisent pas pour sélectionner des algorithmes adéquats. En effet, comme on a pu le constater dans notre état de l'art sur l'anonymisation par généralisation, le choix des algorithmes repose sur des données descriptives de la base. Ces dernières, si elles ne peuvent pas être déduites automatiquement, doivent être fournies par le demandeur. A cet effet, on peut citer la qualité des attributs (identifiant/quasi-identifiant/sensible/non sensible, catégoriel/continu). De plus, certaines de ces données descriptives sont nécessaires et cela quelle que soit la technique. D'autres sont plutôt spécifiques à une technique. A titre d'exemple, la liste des attributs constituant le quasi-identifiant est nécessaire quelle que soit la technique à utiliser. Cependant le type de distribution des données peut entrer en ligne de compte dans la sélection d'algorithmes propres à certaines techniques dont la généralisation.

En résumé, dans un souci de généricité, le contexte d'une anonymisation sollicitée par un utilisateur pour ses micro-données, est construit en deux temps (voir **Figure 87**).

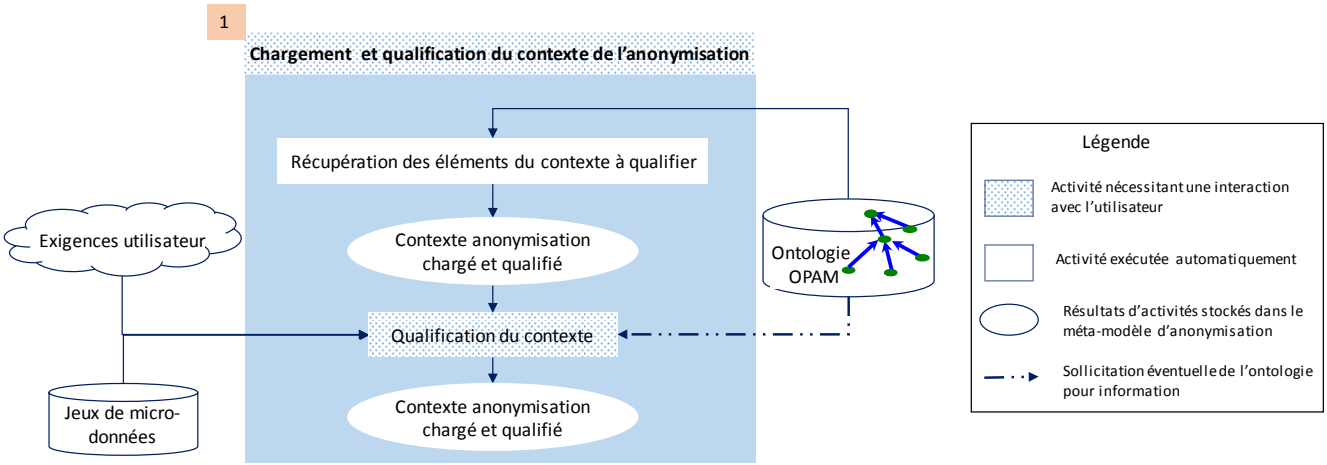


Figure 87. Chargement et qualification du contexte de l'anonymisation

Dans un premier temps, MAGGO construit le contexte à qualifier en récupérant dans l'ontologie ses paramètres, c'est-à-dire les types d'exigences utilisateurs à renseigner ainsi que les types de données descriptives à connaître pour le type d'anonymisation sollicitée. Le sous-schéma d'OPAM sollicité par MAGGO est celui de la **Figure 88**.

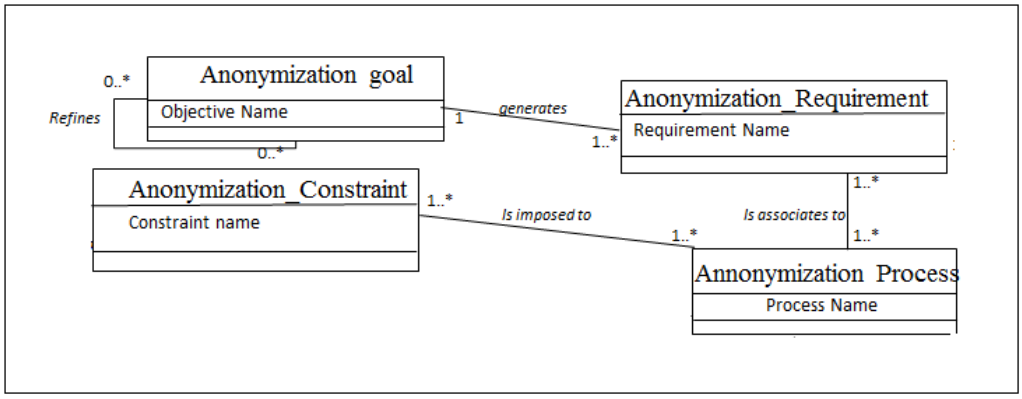


Figure 88. Sous-schéma d'OPAM lié au contexte d'anonymisation

Certains de ces paramètres, rappelons-le, sont spécifiques à une technique. A titre d'exemple, dans le cas d'une anonymisation par généralisation, notre approche MAGGO, après interrogation de l'ontologie OPAM, construira le contexte d'anonymisation par généralisation. Ce contexte est constitué des paramètres de contexte décrits dans le **Tableau 46**.

	Paramètres du contexte d'anonymisation de micro-données par généralisation
Paramètres fournis par l'utilisateur	Seuil de risque toléré
	Taux de suppression autorisé
	Besoin d'usage
	Jeu de micro-données original
	Propriétés de qualité attendues
	Importance relative des propriétés de qualité
Paramètres pouvant être déduits automatiquement	Attributs du QI
	Attributs identifiants
	Attributs sensibles
	Nature de chaque attribut du QI : catégoriel ou continu
	Type de généralisation attendu
	Distribution des données
	taille du jeu de micro-données
	K : taille maximale des classes d'équivalence de QI anonymes
	MaxSup

Tableau 46. Les paramètres du contexte d'anonymisation de micro-données par généralisation

Ces paramètres de contexte, récupérés dans le méta-modèle d'anonymisation, seront renseignés, dans la seconde phase de l'étape de « chargement et qualification du contexte », pour certains par l'utilisateur car ils sont relatifs à ses exigences. A l'exception de k et MaxSup, tous les paramètres sont déductibles de l'analyse des jeux de données. A l'heure actuelle, MAGGO n'est pas en mesure de le faire. Nous comptons, dans l'avenir, intégrer dans MAGGO des composants d'approches pour réaliser automatiquement ce type d'extraction.

Le paramètre MaxSup est calculé à partir de la taille du jeu de données et du taux de suppression autorisé par l'utilisateur en appliquant la formule suivante :

*MaxSup = Taille du jeu de microèdonnées * taux de suppression autorisé*

L'attribut k fait référence au modèle de protection de la vie privée supporté par la technique de généralisation. Il correspond à la taille minimale des classes d'équivalence de QI anonymes pouvant être générées par généralisation. Pour calculer k, nous optons, dans le cadre de cette thèse, pour l'utilisation de la formule suivante proposée dans l'outil PARAT[7] :

k= 100/ taux de risque de réidentification

Cette formule exprime le fait que le taux de risque de ré-identification est inversement proportionnel à k. En d'autres termes, plus k est petit, plus le risque de ré-identification est grand. En effet, dans un jeu de micro-données original, chaque valeur de QI étant différente, le besoin d'anonymisation s'est justifié par le fait que le risque de ré-identification est de 100%. Dès lors que des individus sont regroupés autour d'un QI anonyme commun, le risque est amoindri.

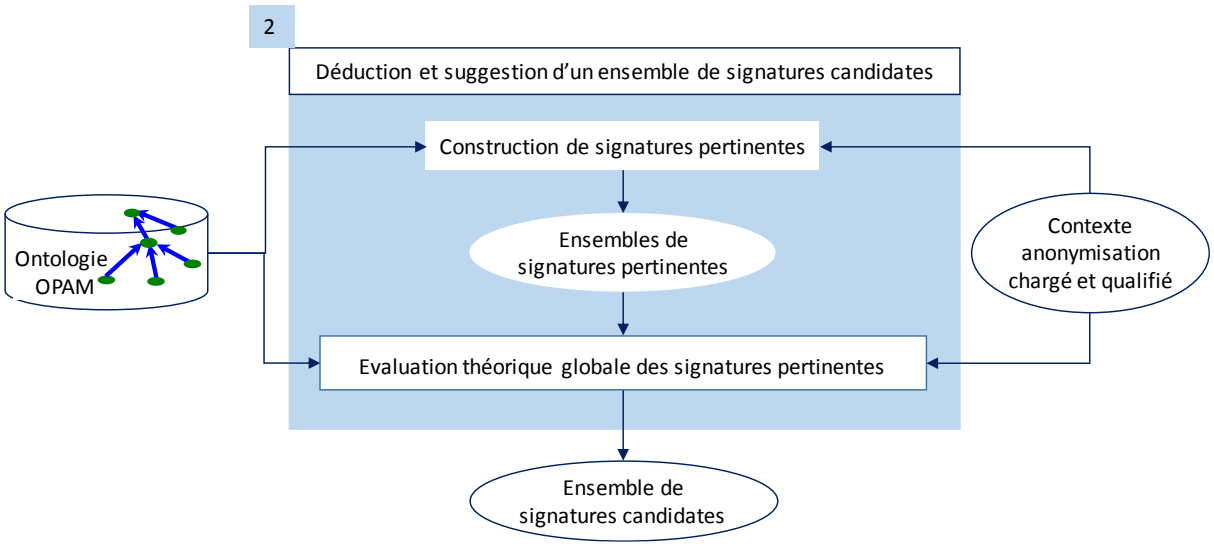
Une fois le contexte d'anonymisation renseigné, MAGGO suggère à l'utilisateur, dans sa seconde étape, sous forme de signatures, un ensemble potentiel d'algorithmes paramétrés susceptibles de satisfaire à ses exigences.

4. Etape de déduction et suggestion de signatures d'algorithmes candidates

Le jeu de données brouillé renvoyé par application d'une technique d'anonymisation dépend fortement de la signature d'algorithme[9] exécutée sur le jeu de données original. Par conséquent, considérant le fait que :

1. les techniques d'anonymisation peuvent être mises en œuvre par au moins un algorithme
2. et que chaque algorithme renvoie un jeu de données anonyme dépendant des valeurs de ses paramètres en entrée,

la construction, l'évaluation et la proposition, à l'utilisateur, de signatures d'algorithmes se rapprochant le plus de ses exigences de qualité, s'avèrent inévitables. C'est ce que se propose de faire cette étape de MAGGO (voir **Figure 89**).



[9] Nous définissons, une signature d'un algorithme par le nom de cet algorithme suivi de la liste des valeurs de ses paramètres en entrée.

Figure 89. Déduction et suggestion d'un ensemble de signatures candidates

Comme le montre la **Figure 89**, la première phase de cette étape consiste à construire des signatures pertinentes. Il s'agira, pour cela, de récupérer, dans un premier temps, parmi l'ensemble des algorithmes de la technique visée, ceux applicables au contexte de l'anonymisation et de les doter de valeurs de paramètres conformes aux contraintes de l'anonymisation spécifiées dans le contexte. La seconde phase a pour objectif de proposer à l'utilisateur, parmi les signatures pertinentes, celles offrant le meilleur score en termes de concordance avec ses exigences de qualité.

Les paragraphes qui suivent détaillent chacune de ces phases

4.1 Construction des signatures pertinentes

Le ciblage des algorithmes applicables au contexte doit précéder leur paramétrage. Cette phase concrétise cette activité en exploitant les paramètres de contexte influençant la sélection des algorithmes. A titre d'exemple, pour une anonymisation par généralisation, si l'utilisateur n'a pas d'exigence sur le type de généralisation à obtenir alors, de ce point de vue, tous les algorithmes de généralisation sont candidats au paramétrage. En revanche, si son souhait est d'obtenir des généralisations multidimensionnelles, alors, cet ensemble se restreint aux algorithmes fournissant ce type de généralisation tels que le « Median Mondrian ».

Pour effectuer ce filtrage d'algorithmes selon un contexte d'anonymisation, l'ontologie OPAM est, bien évidemment, exploitée car elle dispose des connaissances permettant de confronter les exigences des algorithmes aux exigences de l'anonymisation. Ces connaissances sont celles se trouvant dans la partie du sous-schéma d'OPAM représenté dans la **Figure 90**.

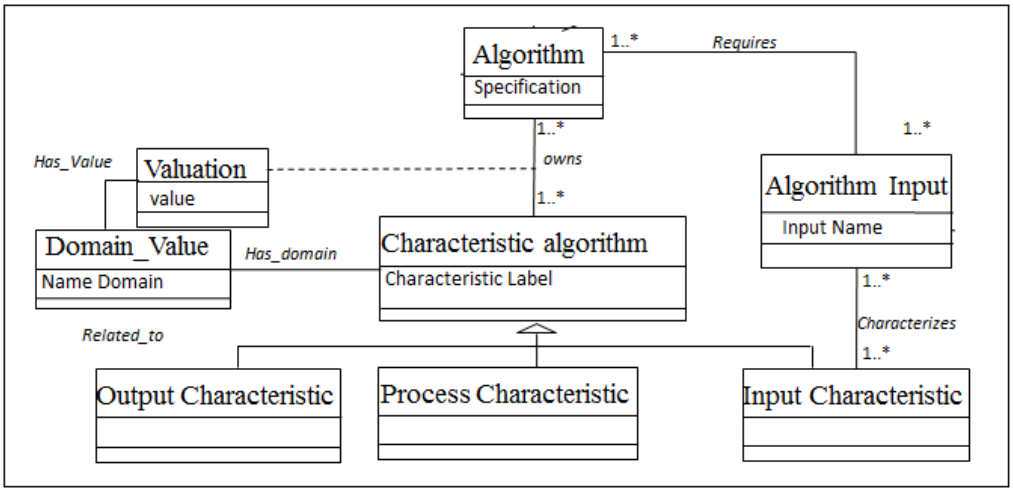


Figure 90. Sous-schéma d'OPAM lié à la construction des signatures pertinentes

Les algorithmes sélectionnés permettent bien sûr d'instancier le méta modèle de l'anonymisation (voir **Figure 91**). Cette instanciation est complétée, pour chaque algorithme sélectionné, par l'ensemble des combinaisons

possibles de valeurs de paramètres pouvant lui être affecté. Chaque algorithme sélectionné, couplé avec chaque combinaison de valeurs de paramètres possible, constitue une signature pertinente.

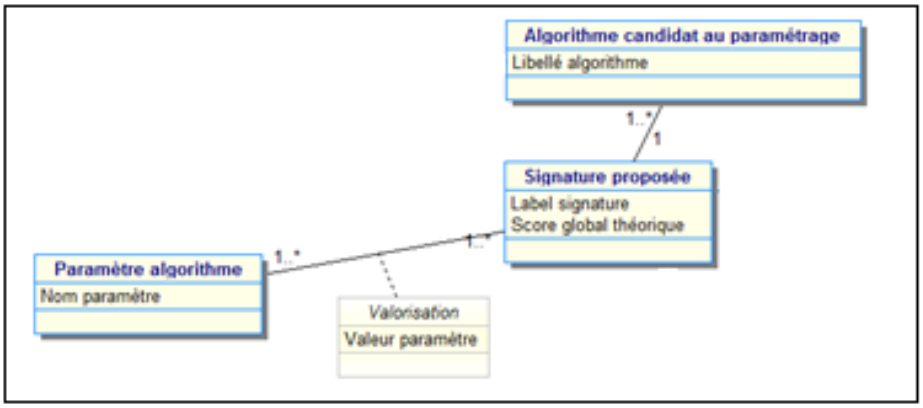


Figure 91. Partie du méta modèle lié à la construction des signatures pertinentes

Les paramètres des algorithmes pouvant être vus comme des contraintes d'anonymisation, il s'agit tout simplement d'octroyer au paramètre de l'algorithme, la valeur du paramètre de contexte généré suite à la prise en compte de la contrainte d'anonymisation imposée par l'utilisateur. A titre d'exemple, dans le cas d'une anonymisation par généralisation, l'utilisateur émet des contraintes qui sont le taux de risque de ré-identification souhaité et le taux de suppression toléré. Ces deux contraintes génèrent dans le contexte de l'anonymisation une valeur pour k et MaxSup. Ces deux valeurs combinées avec chaque algorithme retenu, constituent une signature pertinente.

4.2 Evaluation théorique globale des signatures pertinentes

Cette phase vise à fournir à l'utilisateur les signatures se rapprochant le plus de ses exigences de qualité et de sécurité. C'est un processus décisionnel multicritère pour lequel nous proposons d'appliquer la méthode AHP présentée plus haut dans ce chapitre. Cette dernière, sur la base des comparaisons par paires, détermine le score global de chacune des signatures afin de retenir les mieux classées. Cette phase peut décider de fournir à l'utilisateur les trois signatures pertinentes de score le plus élevé.

La hiérarchie à fournir à AHP a pour premier niveau l'objectif de cette étape. Son niveau intermédiaire correspond à la hiérarchie des exigences emmagasinée dans OPAM et représentée conceptuellement dans la **Figure 92**. Son dernier niveau, c'est-à-dire les feuilles de cette hiérarchie, regroupe les signatures pertinentes à évaluer. A titre d'exemple, la hiérarchie construite par cette phase pour une anonymisation par généralisation à des fins de classification est schématisée à la **Figure 93**.

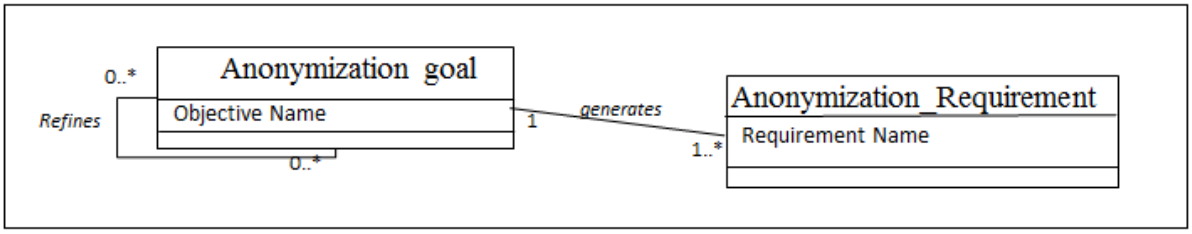


Figure 92. Sous schéma d'OPAM pour la hiérarchie des exigences

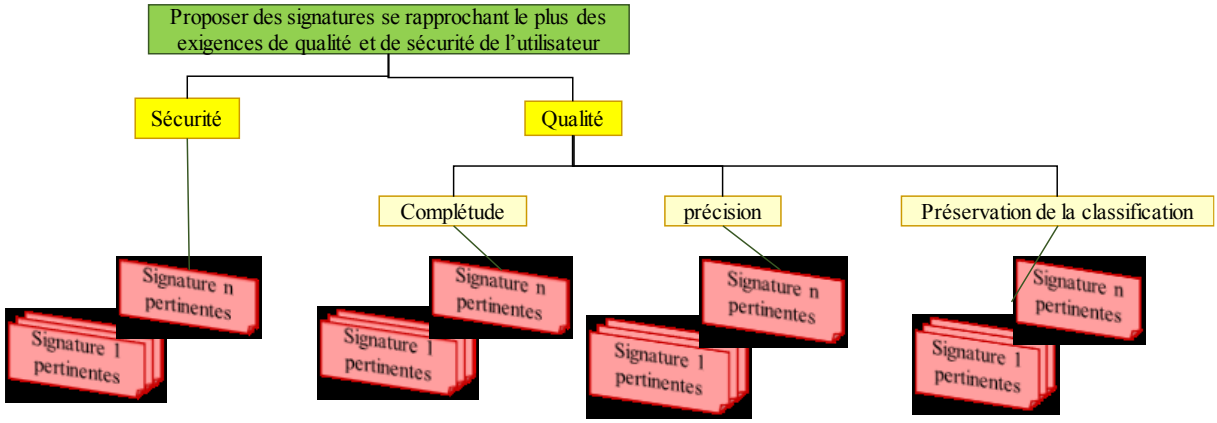


Figure 93. Hiérarchie des exigences pour une anonymisation par généralisation

Une fois la hiérarchie construite, les jugements sur l'importance relative des éléments de cette hiérarchie sont déterminés. Pour ce qui est des jugements entre les éléments du niveau intermédiaire de la hiérarchie (les critères), ceux-là représentent l'ensemble des jugements fournis par l'utilisateur et spécifiés dans le contexte de l'anonymisation. Les jugements sur l'importance relative des signatures sont, quant à eux, déterminés de façon automatique après une évaluation approximative de chaque signature selon un critère donné. Cette évaluation, que l'on nomme « évaluation théorique locale », est déduite des expérimentations faites par les experts en anonymisation. Ces dernières, rappelons-le, sont emmagasinées dans OPAM (voir **Figure 94**).

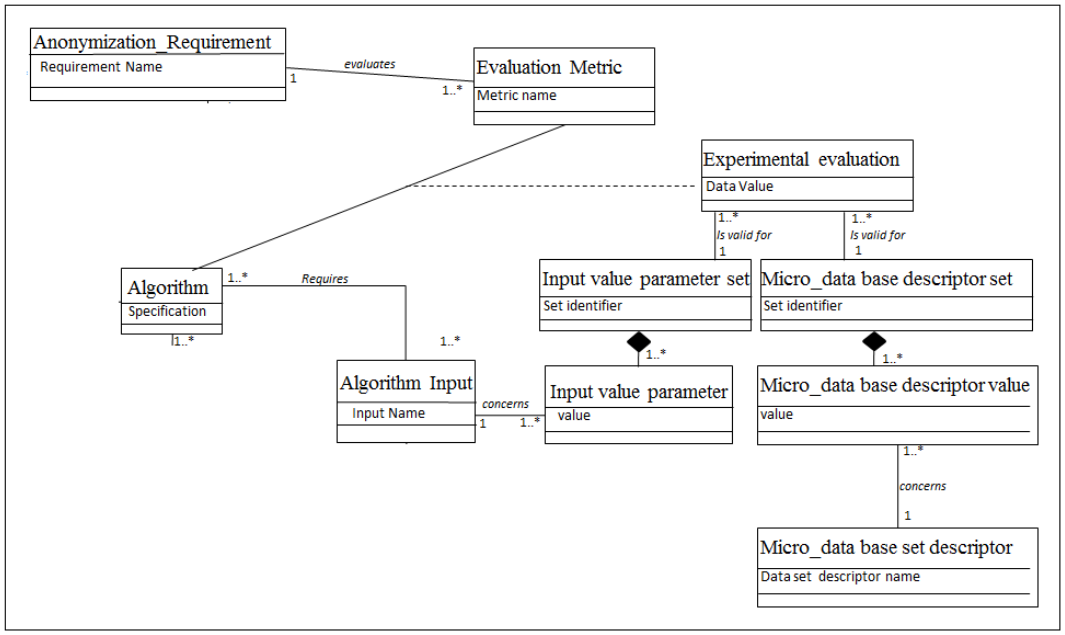


Figure 94. Sous-schéma d'OPAM lié à la construction des signatures pertinentes

La mesure de l'importance relative de chaque signature est aussi déterminée automatiquement. Elle est fondée sur leur évaluation locale et sur une échelle de comparaison disponible dans MAGGO.

Les paragraphes qui suivent décrivent respectivement le processus d'évaluation local et global (le score) d'une signature.

4.2.1 Evaluation théorique locale des signatures pertinentes

Plusieurs évaluations théoriques d'algorithmes d'anonymisation de micro-données sont disponibles dans la littérature. Chaque évaluation fournit la qualité d'un jeu de données anonyme vis-à-vis d'un critère de qualité (sécurité, précision, complétude, etc.) compte tenu d'une signature d'algorithme donnée et de caractéristiques spécifiques au jeu de données originales. Le critère en question est mesuré à l'aide d'une métrique appropriée spécifiée dans l'évaluation. Dès lors qu'une évaluation théorique, coïncidant avec une signature pertinente et avec les caractéristiques du jeu de données original spécifiées dans le contexte d'anonymisation, est indisponible, une technique d'apprentissage supervisée peut être mise en place afin de prédire la qualité de cette signature vis-à-vis d'un critère. Nous avons choisi la régression car elle se prête bien à notre problématique. La variable à expliquer est le critère de qualité à mesurer. Les variables explicatives sont les différents éléments de contexte influençant la variable cible (variable à expliquer). Le jeu d'exemples d'entrainement du modèle de régression est extrait de notre ontologie OPAM. Un exemple est constitué d'une entrée et d'une sortie. L'entrée est un ensemble de valeurs, chacune d'elle représentant une variable explicative. Ainsi, à titre d'exemple, pour une anonymisation par généralisation à des fins de classification, nous disposons de quatre jeux d'exemples : un par critère constituant une feuille du niveau intermédiaire de la hiérarchie AHP (sécurité, complétude, précision, préservation de la classification) décrite à la **Figure 93**. Tous les jeux d'exemples auront la même entrée : une valeur pour k, une valeur pour le nombre d'attributs constituant le QI, une valeur pour la distribution du jeu de

micro-données original. En revanche, ces jeux d'exemples se distinguent par la sortie qui correspond à la mesure du critère cible.

Après évaluation de chaque signature, le méta-modèle est enrichi.

4.2.2 Mesure de l'importance relative des signatures

Une fois les évaluations locales des différentes signatures effectuées, il s'agit de procéder à des comparaisons par paires de signatures afin de déduire l'importance relative des signatures vis-à-vis de chaque critère. Pour ce faire, nous nous inspirons de l'échelle sémantique de Saaty (Saaty 2004) afin de permettre une comparaison automatique par paires de signatures et de livrer à AHP la matrice de comparaison des signatures pertinentes.

Si l'on considère deux couples (E_{Sj}^{Ci}, $E_{Sj'}^{Ci}$) où E_{Sj}^{Ci} (respectivement $E_{Sj'}^{Ci}$) représente l'évaluation locale de la signature Sj (respectivement la signature Sj') pour le critère Ci, nous construisons la table d'échelle sémantique de la façon suivante :

Intensité	Signification	Interprétation formelle de la signification
(Sj, Sj', 1)	Sj et Sj' sont d'égale qualité vis-à-vis du critère Ci	$E_{Sj}^{Ci} - E_{Sj'}^{Ci} \leq \varepsilon_1$
(Sj, Sj', 2)	Sj et Sj' sont entre d'égale qualité et légèrement meilleure vis-à-vis du critère Ci	$\varepsilon_1 < E_{Sj}^{Ci} - E_{Sj'}^{Ci} \leq \varepsilon_2$
(Sj, Sj', 3)	Sj est d'une qualité légèrement meilleure que celle de Sj' vis-à-vis du critère Ci	$\varepsilon_2 < E_{Sj}^{Ci} - E_{Sj'}^{Ci} \leq \varepsilon_3$
(Sj, Sj', 4)	Sj est d'une qualité légèrement meilleure que celle de Sj' vis-à-vis du critère Ci	$\varepsilon_3 < E_{Sj}^{Ci} - E_{Sj'}^{Ci} \leq \varepsilon_4$
(Sj, Sj', 5)	Sj est d'une qualité meilleure que celle de Sj' vis-à-vis du critère Ci	$\varepsilon_4 < E_{Sj}^{Ci} - E_{Sj'}^{Ci} \leq \varepsilon_5$
(Sj, Sj', 6)	Sj est d'une qualité meilleure que celle de Sj' vis-à-vis du critère Ci	$\varepsilon_5 < E_{Sj}^{Ci} - E_{Sj'}^{Ci} \leq \varepsilon_6$
(Sj, Sj', 7)	Sj est d'une qualité nettement meilleure que celle de Sj' vis-à-vis du critère Ci	$\varepsilon_6 < E_{Sj}^{Ci} - E_{Sj'}^{Ci} \leq \varepsilon_7$
(Sj, Sj', 8)	Sj est d'une qualité nettement meilleure que celle de Sj' vis-à-vis du critère Ci	$\varepsilon_7 < E_{Sj}^{Ci} - E_{Sj'}^{Ci} \leq \varepsilon_8$
(Sj, Sj', 9)	Sj est d'une qualité très nettement meilleure que celle de Sj' vis-à-vis du critère Ci	$\varepsilon_8 < E_{Sj}^{Ci} - E_{Sj'}^{Ci} \leq \varepsilon_9$

Tableau 47. Passage des évaluations des signatures à leurs comparaisons deux par deux

Dans cette table servant de comparaison par paires de signatures, $\varepsilon_1 < \varepsilon_2 < \varepsilon_3 < \varepsilon_4 < \varepsilon_5$. Ces valeurs sont définies par l'approche pour chaque critère de qualité.

Par exemple, supposons cinq signatures ont été évaluées selon le critère besoin d'usage classification (**Tableau 48**). Pour passer à leurs comparaisons deux par deux (**Tableau 50**), nous appliquons notre jugement sur ce critère (**Tableau 49**). Cependant, ce jugement se base sur les évaluations des algorithmes qui se trouvent dans la littérature et reste subjectif. Il peut être mis à jour par un expert du domaine. Par ailleurs, cette connaissance aide un novice dans le domaine à pouvoir comparer entre les signatures comme étant un expert.

Signature	Algorithme	input 'k'	input 'Maxsup'	Besoin d'usage 'Classification '
Sig 1	**Datafly**	**10**	**150**	0,54
Sig 2	**Datafly**	**10**	**150**	0,54
Sig 3	**Datafly**	**12**	**200**	0,61
Sig 4	**Datafly**	**12**	**200**	0,61
Sig 5	**Mondrian**	**10**	**0**	0,65

Tableau 48. Evaluation des signatures selon le critère besoin d'usage 'classification'

différence et intensité	
< =2 %	1
Entre 3 % et 5 %	2
Entre 6 % et 8%	3
Entre 9 % et 10 %	4
Entre 11% et 12 %	5
Entre 13 % et 14 %	6
Entre 15 % et 16 %	7
Entre 17 % et 19%	8
>=20 %	9

Tableau 49. Notre jugement sur le critère besoin d'usage 'classification'

A vs. B	Besoin d'usage 'classification'		le plus élevé	différence	intensité
	A	B			
Sig 1 vs. Sig 2	0,54	0,54	A et B	0%	1
Sig 1 vs. Sig 3	0,54	0,61	B	7%	3
Sig 1 vs. Sig 4	0,54	0,61	B	7%	3
Sig 1 vs. Sig 5	0,54	0,65	B	11%	5
Sig 2 vs. Sig 3	0,54	0,61	B	7%	3
Sig 2 vs. Sig 4	0,54	0,61	B	7%	3
Sig 2 vs. Sig 5	0,54	0,65	B	12%	5
Sig 3 vs. Sig 4	0,61	0,61	A et B	0%	1
Sig 3 vs. Sig 5	0,61	0,65	B	4%	2
Sig 4 vs. Sig 5	0,61	0,65	B	4%	2

Tableau 50. Comparaison deux par deux des signatures selon le critère 'classification'

Une fois la comparaison par paires effectuée, AHP se charge de fournir le score global de chaque signature pertinente, ce qui permet de classer ces signatures et de proposer à l'utilisateur, dans l'étape 3 de MAGGO, les signatures qui ont le meilleur score. Ce dernier a la possibilité de choisir une ou plusieurs signatures à exécuter sur son jeu de micro-données. L'exécution de ces signatures fait l'objet de l'étape 4 de MAGGO. Dans cette étape, un jeu de données anonyme est livré pour toutes les signatures pertinentes, de score le plus élevé, choisies par l'utilisateur. Pour guider l'utilisateur dans son choix de jeux de données anonymes, différentes évaluations cette fois-ci réelles, sont effectuées. Chaque évaluation permet de positionner le jeu anonyme vis-à-vis d'une exigence de qualité attendue.

5. Conclusion

Les professionnels sont confrontés à deux difficultés majeures lors d'un processus d'anonymisation. La première concerne le choix de l'algorithme adéquat au contexte de l'anonymisation. La seconde est son paramétrage de telle sorte qu'il délivre un jeu de données sécurisé et utile. Au moyen de notre approche guidée, nous levons ces difficultés en rendant ces deux tâches automatiques. Pour gagner en automatisme, nous exploitons l'ontologie OPAM que nous avons préalablement construite. Cette ontologie peut aussi être consultée par l'utilisateur afin de recueillir les connaissances nécessaires lui permettant de décrire le contexte d'anonymisation et de réagir à chaque fois qu'il est sollicité par l'approche.

De plus, une anonymisation idéale qui produit des données à la fois sécurisées et de qualité n'existant pas, un compromis entre ces deux objectifs peut être atteint, via notre approche tout en tenant compte des attentes de l'utilisateur.

Enfin, dans la conception de MAGGO, nous nous sommes efforcés de la rendre la plus générique possible afin qu'elle puisse être appliquée à d'autres techniques d'anonymisation de micro-données. Nous avons aussi tenu compte de son évolutivité et de son implémentation incrémentale. Pour ce faire, nous avons opté pour une conception dirigée par les modèles.

Chapitre 7 Conception, mise en œuvre et évaluation de MAGGO

Un système d'aide à la décision permet aux utilisateurs à prendre les meilleures décisions et d'éviter les mauvaises. Il peut aussi aider l'utilisateur à prendre des décisions plus rapidement ou avec moins d'informations et de connaissances.

MAGGO a pour but d'aider un éditeur de données à prendre les décisions concernant le choix des algorithmes d'anonymisation qui vont être appliqués sur une base de données originales selon un contexte donné.

Ce chapitre présente la conception, la mise en œuvre et l'évaluation de MAGGO. Il aborde les aspects les plus importants du développement de la plate-forme implémentant MAGGO, dédiée au guidage d'un éditeur de données au cours de son processus d'anonymisation.

La réalisation repose sur EMF (Eclipse Modeling Framework), un cadriciel qui permet de réduire, de manipuler et de transformer les méta-modèles.

Le chapitre est structuré comme suit : la première section donne une vue d'ensemble de la plateforme. Quelques aspects techniques liés au développement des modules constituant l'architecture du prototype sont présentés dans la deuxième section. Un exemple des interfaces utilisateurs sont présentés pour illustrer l'exécution de la plateforme. La dernière section présente l'évaluation de l'approche MAGGO.

1. Présentation du prototype

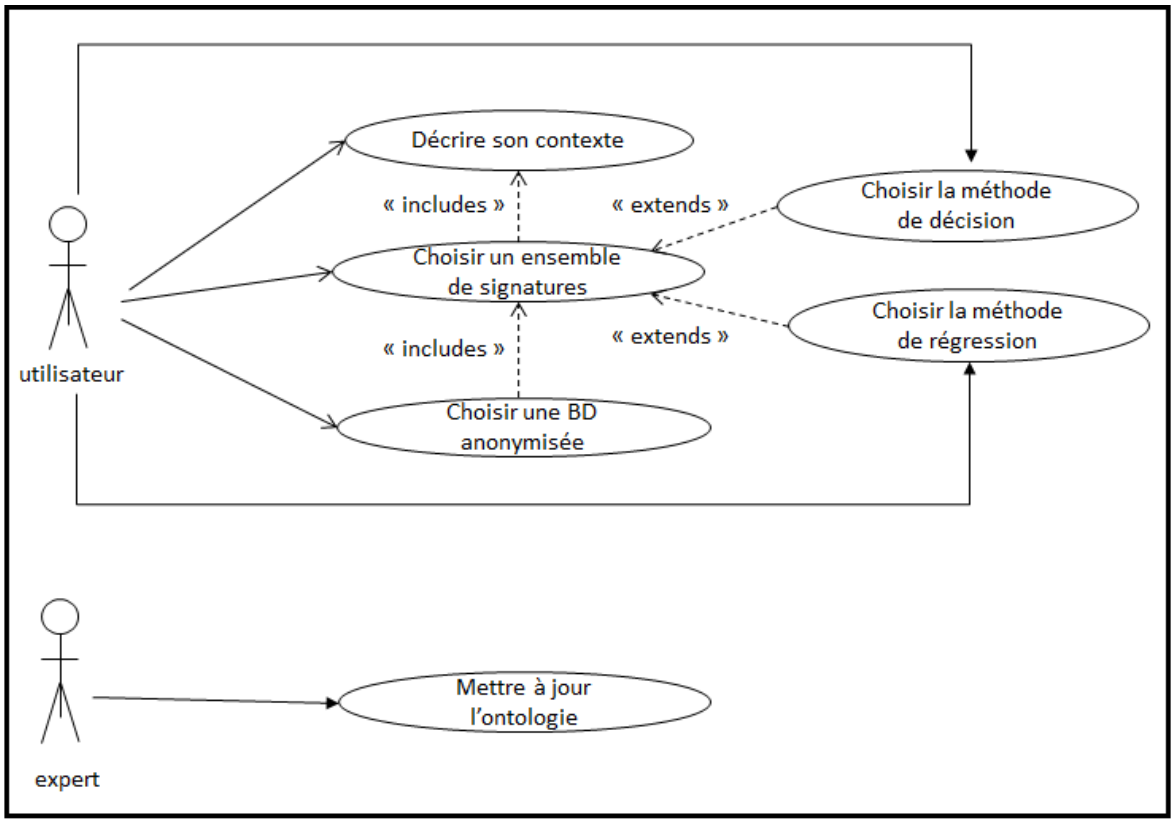


Figure 95**.** Cas d'utilisation de la plateforme

Les scénarios associés aux cas d'utilisation mentionnés dans la **Figure** 95 et qui ont fait l'objet de développement, sont liés aux deux acteurs qui sont l'utilisateur de la plateforme et l'expert dans le domaine de l'anonymisation. Ces scénarios sont :

- Le scénario nominal du cas « décrire son contexte »
- Le scénario nominal du cas « choisir un ensemble de signatures» qui inclut les cas « choisir la méthode de décision» et « choisir la méthode de régression »
- Le scénario nominal du cas « exporter une base de données anonymisée»
- Le scénario nominal du cas « mettre à jour l'ontologie»

1.1 Principales composantes de la plateforme

L'architecture logicielle associée à cette version du prototype est fourni dans la **Figure** 96. Dans cette architecture sont distinguées :

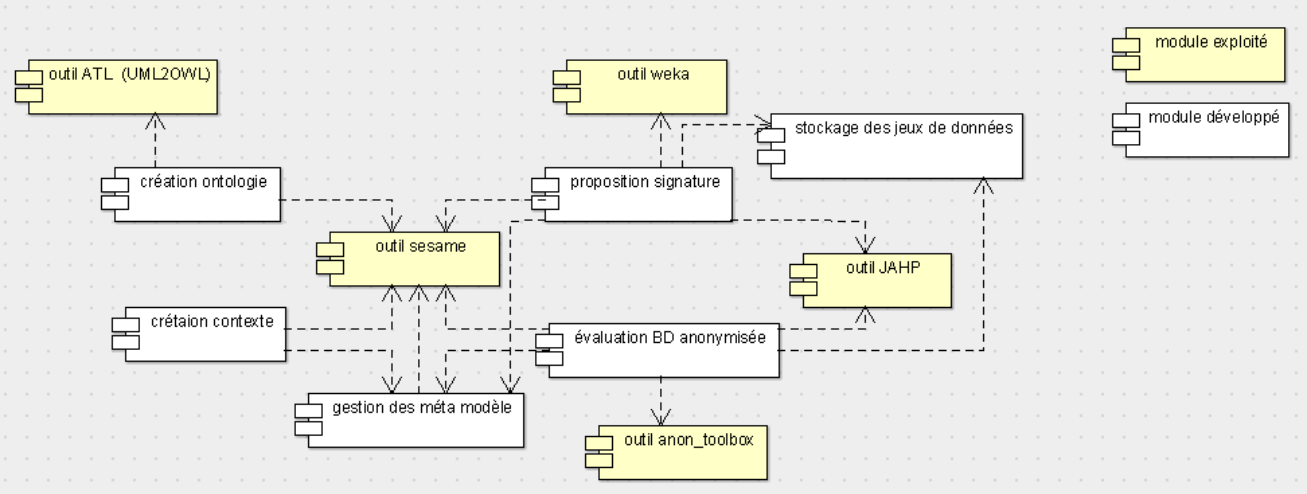


Figure 96**.** Architecture logicielle

- Les modules internes à la plateforme : création ontologie, création contexte, proposition signature, gestion des métas modèles, stockage des jeux de données et évaluation BD anonymisée.
- De ceux qui sont sollicités : outil ATL (UML2OWL), outil weka, JAHP, sesame et anon-toolbox

1.2 Les outils et les technologies utilisées

Cette section présente les technologies utilisées dans la plateforme (l'outil ATL UML2OWL a été déjà présenté dans la section 5 du chapitre 5.

1.2.1 Le triple store sesame

Un TripleStore est une base de données spécialement conçue pour le stockage et la récupération des données RDF (Resource Description Framework) ou OWL à travers des requêtes sémantiques. Contrairement à une base de données relationnelle, un triple store ne stocke qu'un seul type de données : le triplet. Ce dernier est une entité de données composée de sujet-prédicat-objet.

Sesame est un Open-source triple store. Il a été créé dans le cadre d'un projet web sémantique «On-To-Knowledge», qui a duré de 1999 à 2002. Il s'agit d'une bibliothèque qui peut être intégrée dans n'importe quelle application Java.

Sesame propose une boîte à outils entièrement modulaire et des API java qui peuvent être connectées à tous ses modules. En effet, il contient une API simple pour les analyseurs RDF, une couche de stockage et d'inférence, une API de référentiel pour la gestion des données RDF et un serveur HTTP pour accéder aux référentiels Sesame via le protocol HTTP. Le langage de requête utilisé dans Sesame est SPARQL.

1.2.2 API Weka

Weka (Waikato Environment for Knowledge Analysis) est une plateforme qui permet l'application de l'apprentissage automatique. Elle contient une collection d'algorithmes pour les tâches d'exploration des données qui peuvent être appliquées directement ou appelés à partir des API Java. Elle supporte des outils pour le

prétraitement des données, la classification, la régression, le regroupement, la visualisation, les fonctions de sélection et les règles d'association.
Toutes les techniques de Weka reposent sur l'hypothèse que les données sont rassemblées dans un seul fichier, avec un nombre fixe d'attributs de type numérique ou nominal.

1.2.3 JAHP

- JAHP Java Analytical Hierrachy Process est une API java, conçue afin de permettre aux utilisateurs de prendre des décisions personnelles. Elle est basée sur des interfaces Swing, et aide à obtenir des décisions plus rapides et plus justifiables avec une approche de décision structurée et basée sur AHP. En effet, elle offre une interface graphique qui permet à un utilisateur de construire la hiérarchie des critères, spécifier les alternatives et de comparer les nœuds deux à deux par rapport à leur ancêtre dans la hiérarchie. JAHP contient deux principales packages :

- le package 'view' est composé d'un ensemble d'interfaces Swing
- le package 'Model' est composé par des classes java qui permettent de traiter les données récupérées à partir du package 'view'.

1.2.4 UTD Anonymization Toolbox

C'est une plateforme open source en langage java implémentée par des chercheurs et contient divers algorithmes qui peuvent être appliqués directement sur un jeu de données ou peuvent être utilisés à l'intérieur d'autres applications. Le SGBD est de type sqlite.
Elle contient six algorithmes d'anonymisation et supporte trois modèles de protection de la vie privée: Datafly, Mondrian multidimensionnel, Incognito avec k-anonymat, Incognito avec l-diversity, Incognito avec t-proximité et Anatomie.

1.3 Description des modules

Dans cette partie nous décrivons les différents modules de l'architecture (voir **Figure** 96). Le module gestion méta-modèles gère les métas modèles et leurs instanciations. Le module stockages de jeu de données stocke les données (les bases de données originales et anonymisées) et gère leurs métadonnées. Le module création ontologie a été déjà présenté dans la section 5 du chapitre 5. Dans ce qui suit, nous allons présenter chacun des modules développés restants.

1.3.1 Description du module création contexte

Dans la conception de MAGGO, nous nous sommes efforcé de la rendre la plus évolutive et incrémentale ~~que~~ possible. Nous proposons par ailleurs que la phase du chargement du contexte soit faite une seule foi. Cependant, la qualification du contexte par un utilisateur sera mise en œuvre au fur et à mesure que les autres étapes de MAGGO. En d'autres termes, au début de chaque sous étape de MAGGO, l'utilisateur spécifie une partie de son

contexte. Un méta modèle de contexte est alors proposé stockant les éléments du contexte nécessaires aux différentes étapes du processus d'anonymisation (voir **Figure** 97). Ainsi, ces éléments seront extraits à partir de l'ontologie (**Figure 88** dans le chapitre 6). L'élément du contexte le plus important est la base de données originale représentée par le concept '*données_à_anonymiser'*. Les autres éléments sont extraits et stockés selon leur fonction dans les étapes de l'approche. Nous présentons dans ce qui suit les concepts qui se trouvent dans le méta-modèle de contexte (**Figure** 97):

- '*Critères_détermination_signature'*: sont les éléments du contexte qui sont nécessaires pour la génération des valeurs des paramètres d'entrée.
- '*Critères_filtrage'* : sont les éléments du contexte qui sont nécessaires pour la sélection ou le filtrage des algorithmes qui peuvent être appliqués au cours d'un processus d'anonymisation.
- '*Critères_évaluation_théorique*': sont les éléments du contexte qui sont nécessaires pour les évaluations locales théoriques.
- '*Préférences_utilisateur*': la hiérarchie des buts du processus d'anonymisation est un élément du contexte. Un but est représenté par le concept '*préférence_utilisateur'* et la relation entre un but et son sous but est représentée par le lien réflexif '*ascendant*'. Grâce à cette hiérarchie, l'utilisateur peut exprimer ses préférences par rapport à ces buts.
- Une fois que le méta modèle contexte est instancié, il sera stocké dans le module gestion méta modèle.

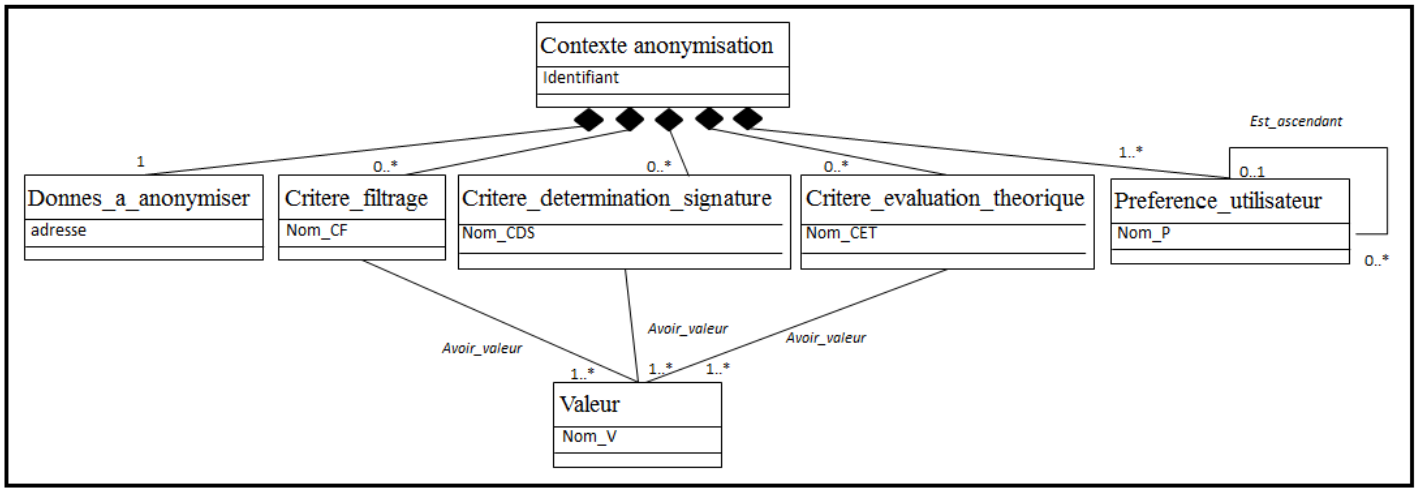


Figure 97**.** Le méta-modèle de contexte

1.3.2 Description du Module proposition des signatures

Afin d'avoir une plate-forme générique, nous avons construit sept sous modules dans le module 'proposition des signatures' (**Figure** 98) qui sont présentés un par un dans ce qui suit :

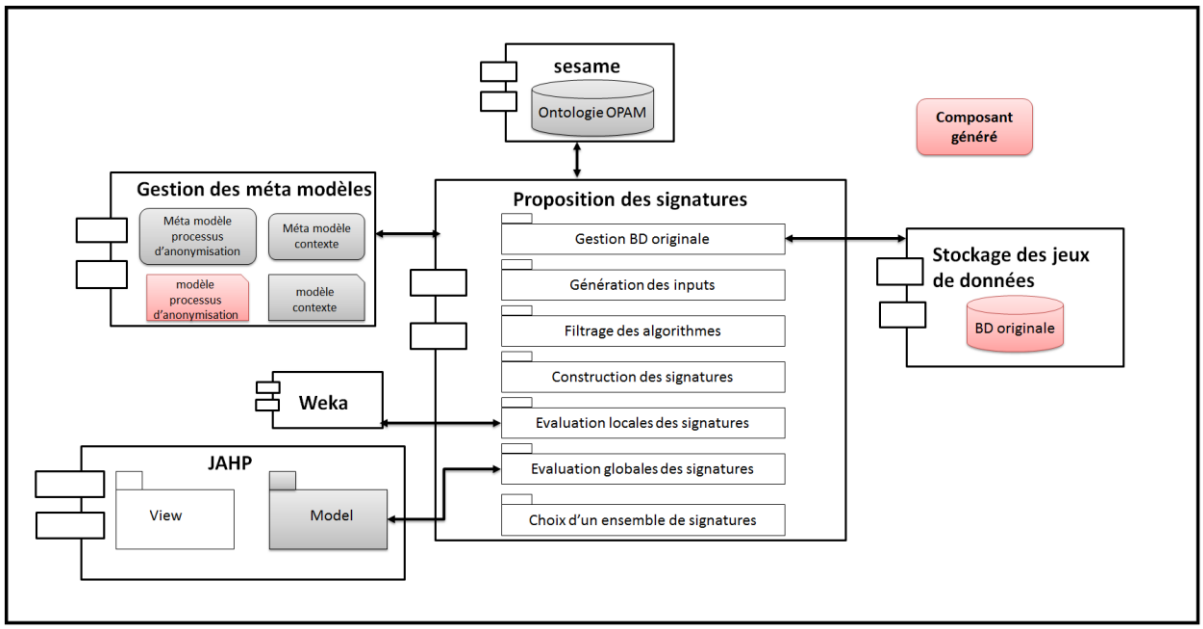


Figure 98**.** Les interactions du module 'proposition des signatures' avec les autres modules

- *Gestion des bases de données originales*

Ce module permet à l'éditeur de données de charger sa base de données originale, d'analyser ses données et de spécifier les types des attributs (numérique ou nominal, ID ou QID ou sensible). La base de données chargée et les métas données spécifiées seront stockées dans le module 'stockage de jeu de données'.

- *Génération des inputs*

Ce module récupère les instances du concept '*critères_détermination_signature'* à partir du méta modèle contexte déjà instancié dans le module gestion du contexte, les affiche dans une interface graphique pour que l'éditeur de données puisse spécifier leurs valeurs (voir **Figure** 104). Il récupère les fonctions qui permettent de générer des inputs à partir de l'ontologie, ainsi, il stocke les valeurs des '*critères_détermination_signature'* spécifiés par l'éditeur de données ainsi que les valeurs des inputs générées automatiquement dans le méta modèle du processus d'anonymisation.

- *Filtrage des algorithmes*

Ce module récupère les instances du concept '*critères_filtrage'* à partir du méta modèle contexte, il récupère leurs valeurs à partir du module '*stockage de jeu de données'*. Il récupère aussi l'ensemble des algorithmes et leurs domaines d'application à partir de l'ontologie, il exécute l'algorithme de filtrage (voir section dans le chapitre 6), ainsi, il stocke les valeurs des '*critères_filtrage'* et l'ensemble des algorithmes sélectionnés dans le méta modèle du processus d'anonymisation.

- *Construction des signatures*

Ce module récupère les valeurs des inputs et les algorithmes sélectionnés à partir du méta modèle du processus d'anonymisation, applique l'algorithme de construction des signatures (voir section dans le chapitre 6). Enfin, il met à jour ce méta modèle en stockant les signatures générées automatiquement.

- *Evaluation locales des signatures*

Ce module récupère les instances des concepts '*critères_évaluation_théorique'* et les '*préférences_utilisateur'* (afin de spécifier les critères feuilles dans la hiérarchie des buts) à partir du méta modèle contexte. Il récupère aussi l'ensemble des signatures sélectionnées à partir du méta modèle de processus d'anonymisation. Finalement, il récupère des évaluations à partir de l'ontologie et les stocke dans un fichier csv nommé données_évalutaion.csv qui représente le modèle multidimensionnel (voir chapitre 3).

Un ensemble de fichier arff est construit à partir du fichier données_evaluation.csv. Chacun de ces fichiers représente l'ensemble des lignes ayant les mêmes valeurs de critère et les mêmes valeurs d'algorithme nommé par Critère_Algo.arff (voir **Figure** 99). En appliquant la méthode l'arbre qui se trouve dans le module Weka sur chacun de ces fichiers, nous obtenons un modèle de régression pour chaque algorithme et chaque critère.

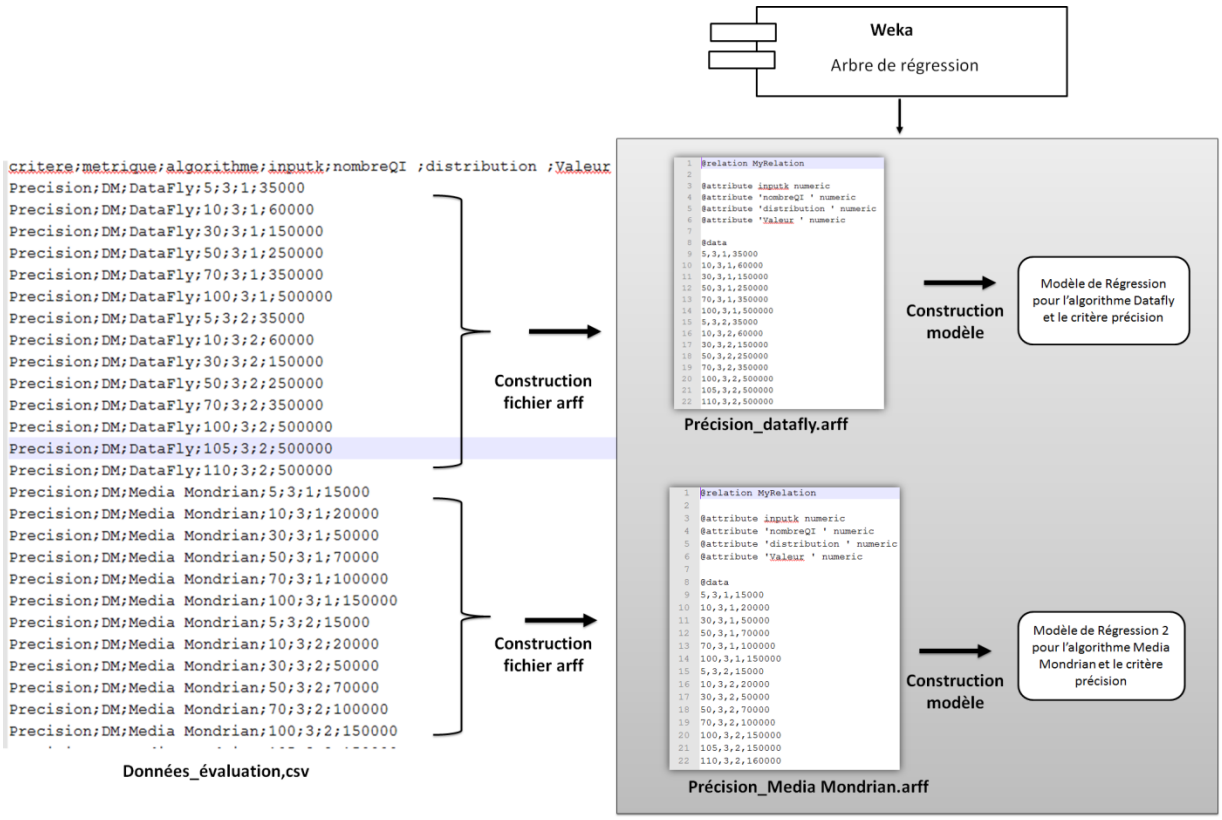


Figure 99**.** Construction des fichiers arff

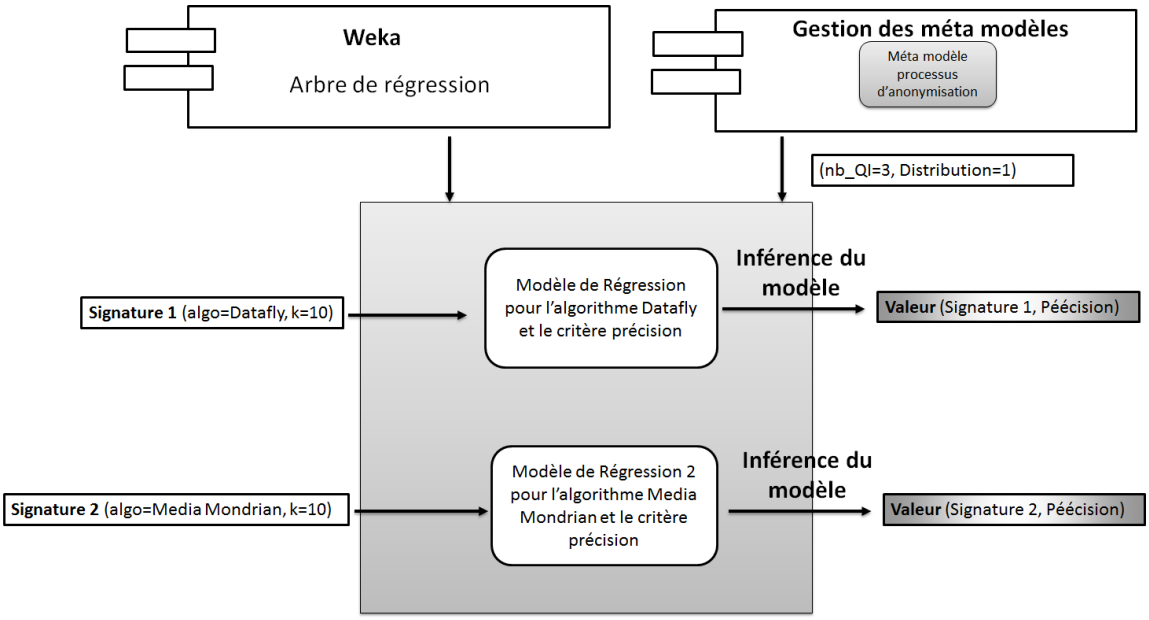


Figure 100**.** L'inférence des modèles de régression

Afin de déterminer la valeur de chaque signature par rapport à un critère, il faut tout d'abord chercher le modèle de régression correspond à l'algorithme de la signature et le critère. Ensuite, en entrant la valeur de k de la signature, le nombre de QI et la distribution des données, la valeur est générée par l'inférence du modèle suivant la méthode de régression. (voir **Figure** 100). Ces valeurs sont enregistrées dans le méta modèle processus d'anonymisation.

- *Evaluation globales des signatures*

Ce module récupère l'ensemble des signatures et leurs comparaisons à partir du méta modèle du processus d'anonymisation, les comparaisons des critères générées par l'éditeur des données, les traite en appliquant des fonctions dans le package model de JAHP qui génère les scores des signatures

1.3.3 Description du module 'évaluation des bases de données anonymisées'

Le module 'évaluation des bases de données anonymisées' est composé de quatre sous modules : exécution des signatures, évaluations locales des BD anonymisées, évaluation globales des BD anonymisées et choix d'une BD anonymisées (voir **Figure 101**).

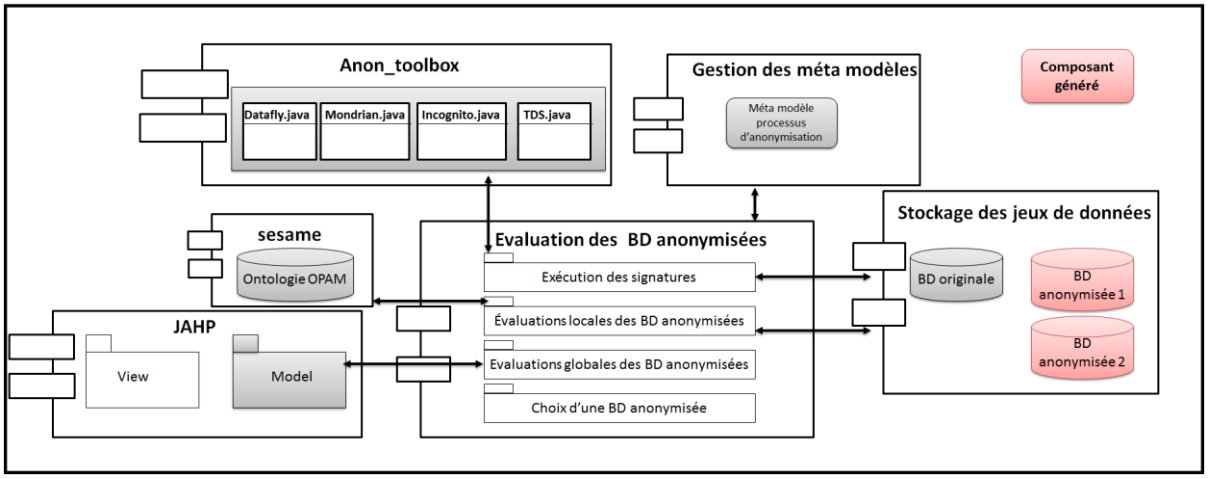


Figure 101. Les interactions du module 'Evaluation BD anonymisées' avec les autres modules

- *Exécution des signatures*

Ce module récupère l'adresse de la BD originale et l'ensemble des signatures à partir du méta modèle de processus d'anonymisation. Ensuite, il exécute les algorithmes (se trouvant dans le module Anon_Toolbox sous forme de classe java et qui correspondent aux signatures récupérées) sur la BD originale. Enfin, il génère des BD anonymisées et les stocke dans le module stockage des jeux de données.

- *Evaluations locales des BD anonymisées*

Ce module récupère les adresses des BDs anonymisées et la hiérarchie des critères à partir du méta modèle de processus d'anonymisation. Ensuite, il extrait les métriques d'évaluation à partir de l'ontologie OPAM. Finalement, il évalue chaque BD anonymisée selon chaque critère en exécutant la métrique correspondante.

- *Evaluation globales des BD anonymisées*

Ce module se fonctionne de la même façon que le package '*évaluation globale des signatures*' qui se trouve dans le module '*Proposition des signatures'* à la différence que les préférences utilisateurs sont déjà stockés dans le méta modèle de processus d'anonymisation et les évaluations sont faites sur les BD anonymisées et non sur les signatures.

- *Choix d'une BD anonymisée*

Ce module récupère toutes les définitions set les abstractions à partir de l'ontologie, les évaluations des BDs anonymisées à partir du méta modèle du processus d'anonymisation et affiche une interface graphique à l'éditeur de données afin de lui permettre d'exporter celle qui lui semble la meilleure.

2. Les interfaces de la plateforme

Dans ce qui suit nous présentons les interfaces qui assurent le déroulement de notre application selon un contexte donné d'un processus d'anonymisation. On suppose qu'un éditeur de données a chargé une base de données

originale contenant 1000 enregistrements, trois attributs constituants le QI (âge, niveau d'étude et sexe) et un attribut sensible salaire. La distribution de ces données est uniforme. Les métas données de la base de données sont représentées dans l'annexe C. On suppose ainsi que le contexte est caractérisé comme suit (**Figure** 102). Le risque maximum toléré est 20%. De même, on admet que l'on ne peut supprimer plus de 20% des données. De plus, la table à anonymiser est de grande taille (1000 enregistremts). L'usage des données anonymisées est la classification selon l'attribut cible salaire. L'utilisateur accorde autant d'importance à l'utilité des données qu'au respect de la vie privée. La qualité de la précision des données est un peu plus préférable que la qualité du besoin d'usage et beaucoup plus préférable que la complétude. Finalement, la qualité de besoin d'usage est beaucoup plus préférable que la complétude.

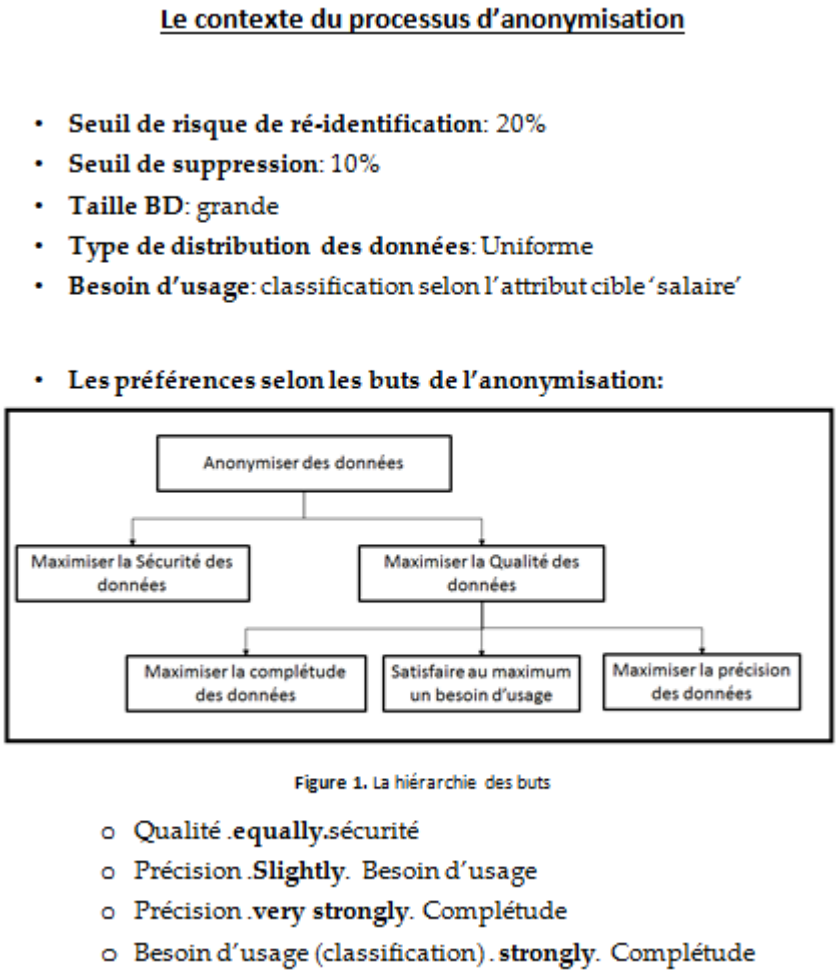


Figure 102. Le contexte proposé pour le processus d'anonymisation

Une fois que les données sont chargées, l'utilisateur peut les analyser à travers l'interface 'BD originale' (**Figure** 103).

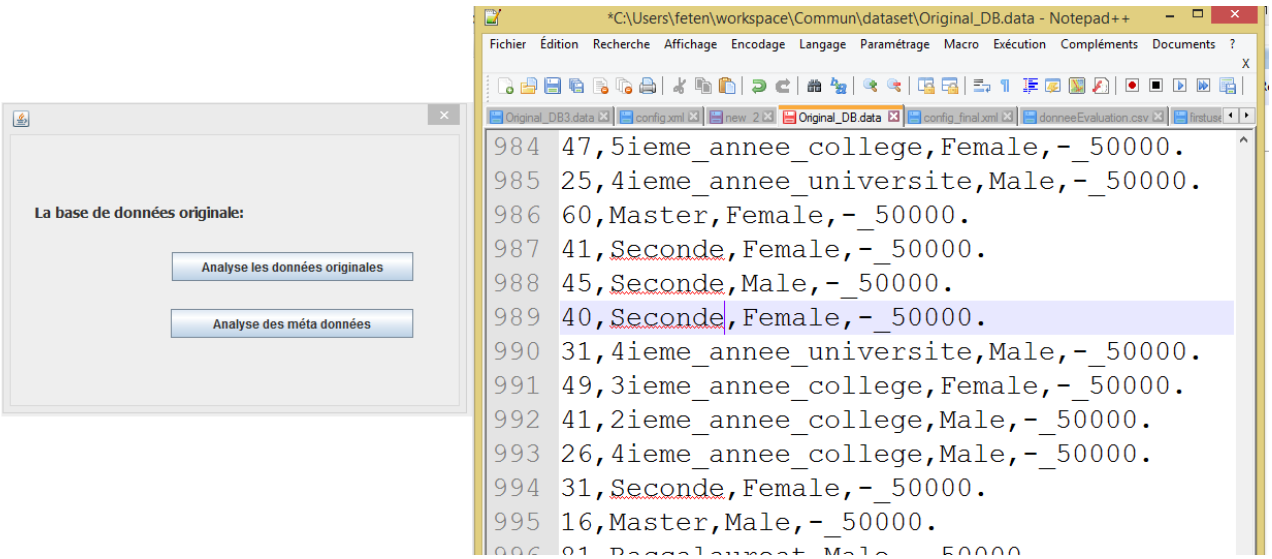


Figure 103**.** Interface ‘BD originale’

Ensuite, l’utilisateur doit spécifier son contexte en commençant par la saisie de ses contraintes (seuil de risque de ré-identification et seuil de suppression des données autorisé) à travers l’interface ‘contraintes d’anonymisation’ (**Figure** 104). Elle contient des boutons ‘aide’ qui permettent de fournir à l’utilisateur les informations nécessaires afin de l’aider à spécifier ses contraintes d’anonymisation. Si par exemple, il appuie sur le premier bouton aide, alors, la définition de l’input k est affichée.

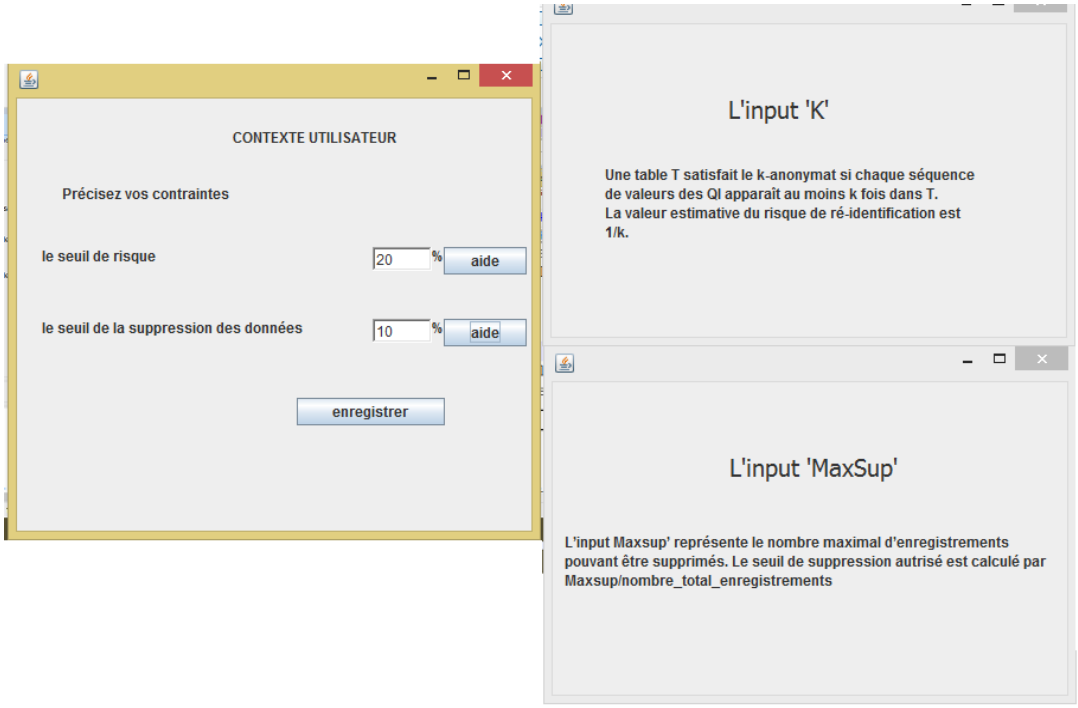


Figure 104**.** Interface ‘contraintes d’anonymisation’

L'outil génère par la suite les valeurs des paramètres d'entrée des algorithmes d'anonymisation et les affiche à l'utilisateur (**Figure 105**).

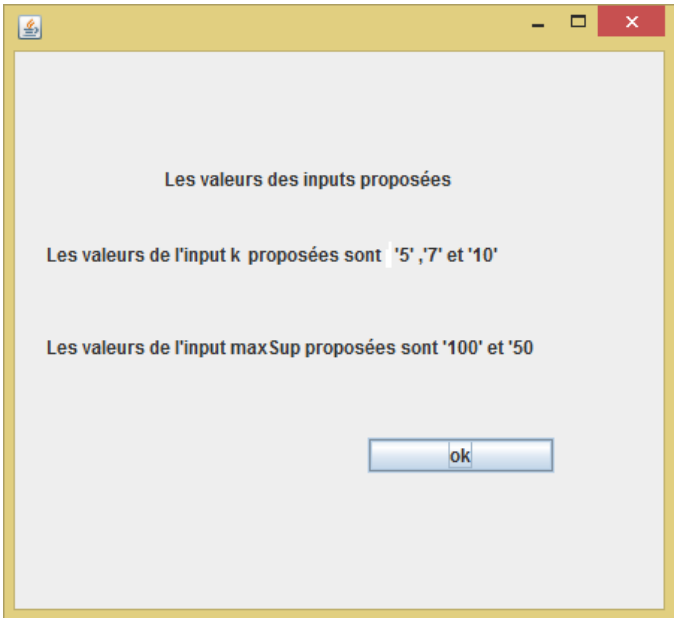


Figure 105. Interface 'les valeurs des inputs proposées'

Puis, l'utilisateur doit spécifier ses critères de filtrage (la taille de sa base de données et son besoin d'usage) à travers 'interface critères de filtrage' (**Figure** 106). L'algorithme de Samarati ne peut pas être appliqué à une table de cette taille, car il est trop gourmand en temps de réponse. Cette information fait partie des connaissances contenues dans l'ontologie. Supposons donc que seuls les algorithmes Datafly, Median Mondrian et TDS remplissent les contraintes et peuvent donc être proposés à l'utilisateur.

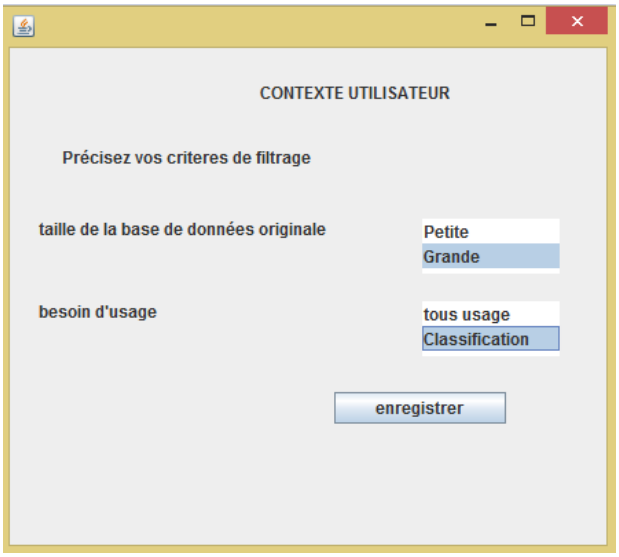


Figure 106**.** Interface 'critères de filtrage'

L'outil pourrait maintenant construire les signatures en combinant les valeurs des paramètres d'entrées (k et maxSup) avec les algorithmes selectionnés. Douze signatures sont générées et affichées dans la **Figure** 109.
Les critères des évaluations théoriques ou locales (la distribution des données et le nombre des attributs QI) peuvent être déduites automatiquement, cependant, à priori, ils doivent être spécifiés par l'utilisateur à travers l'interface 'critères d'évaluations théoriques' (**Figure** 107). Ces valeurs permettent à l'outil d'estimer les valeurs des signatures par rapport à chaque critère d'anonymisation en appliquant l'arbre de régression.

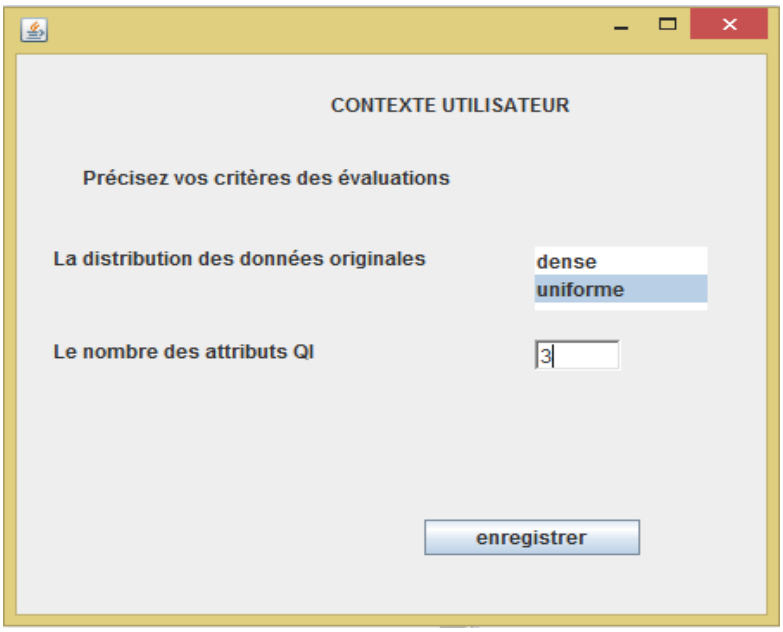


Figure 107**.** Interface 'critères d'évaluations théoriques'

Finalement, l'utilisateur doit spécifier ses préférences à travers l'interface des 'préférences de l'utilisateur' (**Figure** 108). A chaque fois qu'il sélectionne un critère dans la hiérarchie des critères (qui se trouve dans la partie gauche), affiche instantanément ses descendants et lui permet de les comparer deux à deux selon l'échelle sémantique de AHP. Elle permet aussi d'afficher instantanément le ratio d'inconsistance ainsi que les poids des critères.

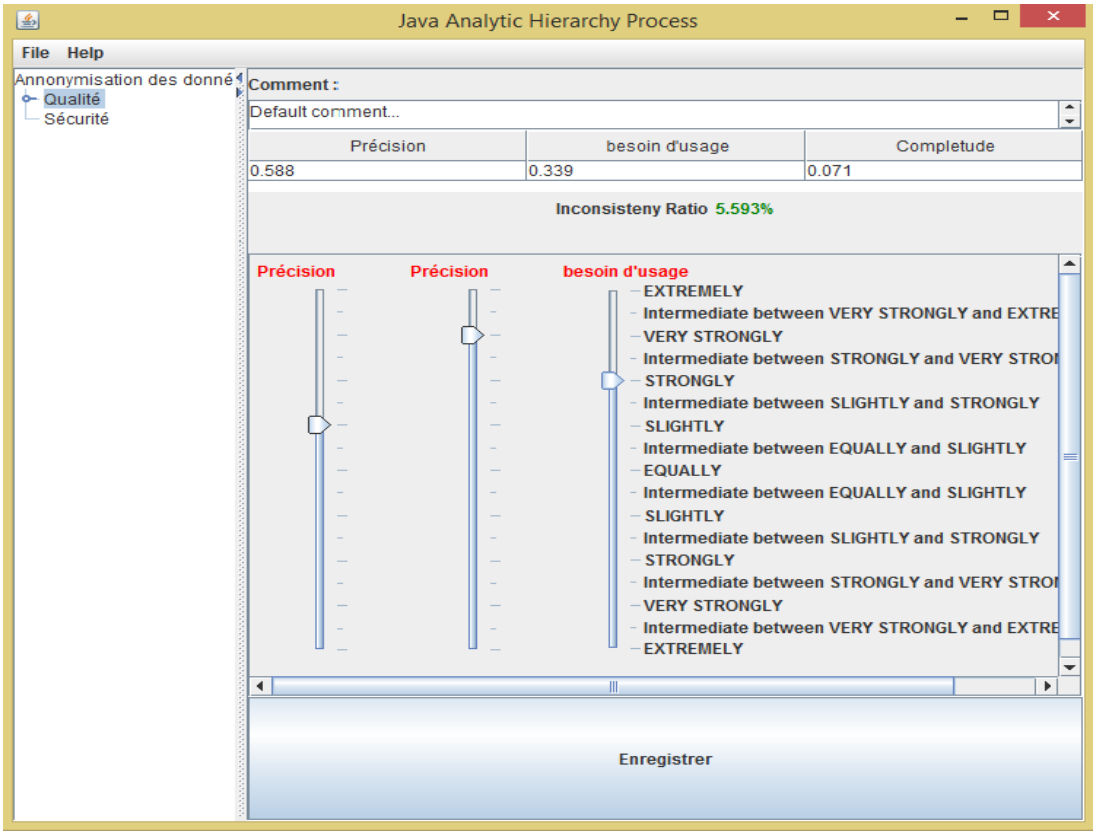


Figure 108**.** Interface ‘Préférences de l’utilisateur’

L’interface ‘les évaluations des signatures’ (**Figure** 109) affiche les évaluations théoriques des signatures selon les critères d’anonymisation. Elle fournit un guidage informatif (en utilisant les boutons qui sont en bas de l’interface) représenté par des descriptions simples des algorithmes proposés (**Figure 110**) et des définitions des critères d’anonymisation (**Figure 111**) afin d’aider l’éditeur de données à analyser les évaluations des signatures. Elle affiche aussi les scores finaux des signatures qui représentent la pertinence de chacune par rapport au contexte du processus d’anonymisation. Ces scores aident l’éditeur de données à comparer les signatures et d’en choisir un sous ensemble afin de les exécuter sur la base de données originale. L’utilisateur sélectionne un sous ensemble de signatures pour les exécuter.

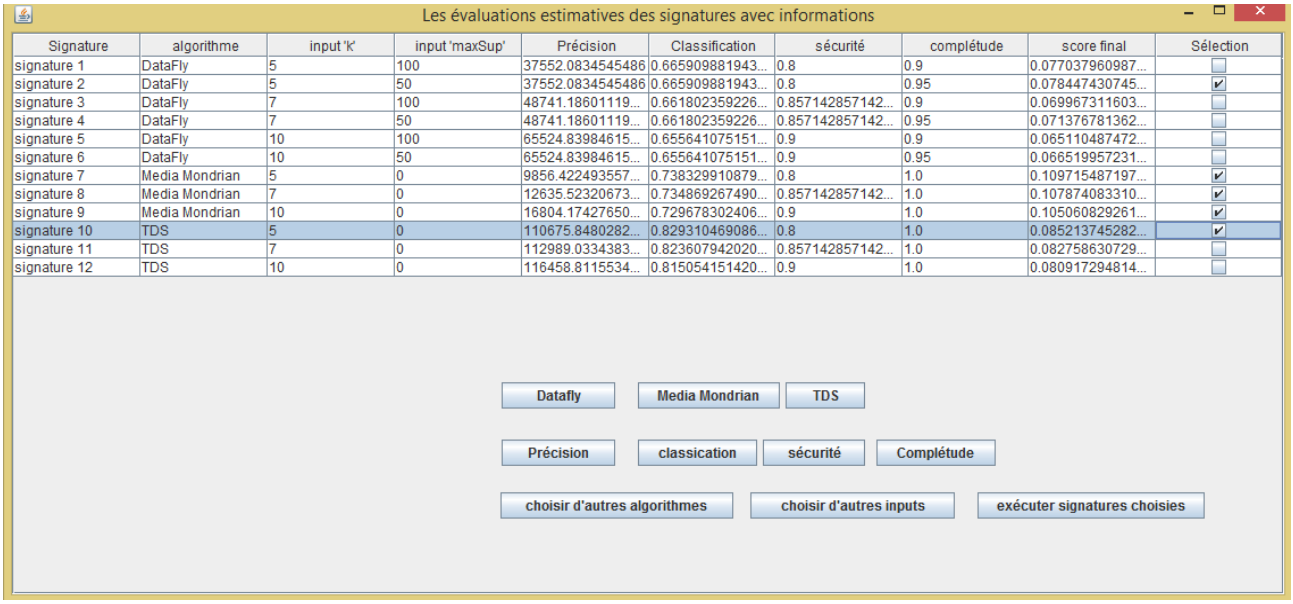


Figure 109**.** Interface ‘Les évaluations des signatures’

L’interface ‘description de l’algorithme Datafly’ (**Figure 110**) s’affiche quand l’utilisateur appuie sur le bouton ‘Datafly’. Ceci est un exemple de description des algorithmes d’anonymisation. Le bouton fonctionnement de Datafly permet d’afficher l’abstraction de l’algorithme avec un exemple de son déroulement.

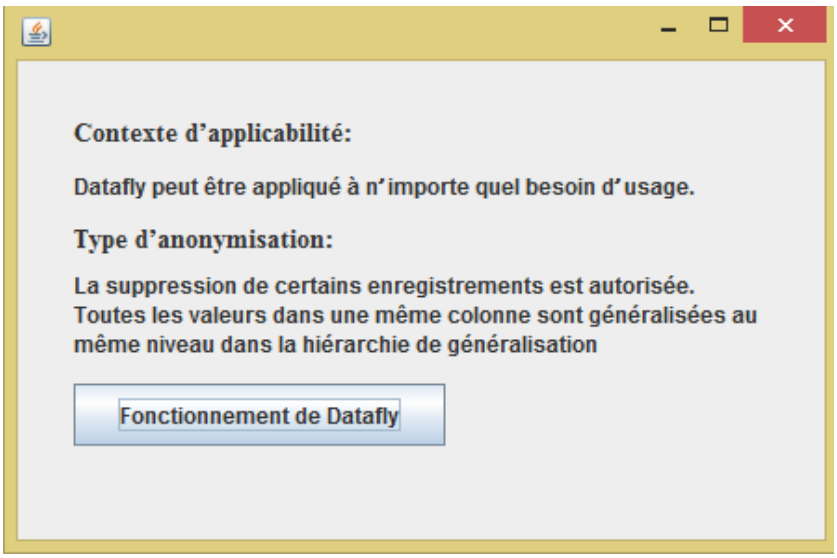


Figure 110. Interface ‘description de l’algorithme Datafly’

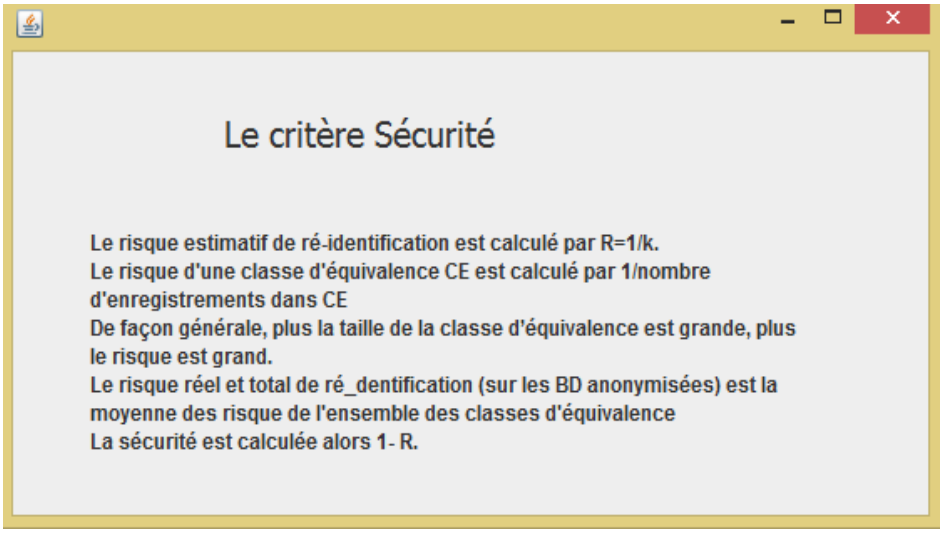


Figure 111. Définition du critère 'sécurité'

La dernière étape exécute les signatures sélectionnées et les évalue en utilisant des métriques d'évaluations. L'interface les évaluations des BD anonymisées (**Figure 112**) affiche les scores et permet à l'utilisateur d'exporter la BD anonymisée choisie. Elle permet aussi d'analyser les données anonymisée lorsqu'il clique sur la ligne correspondante (**Figure 113**).

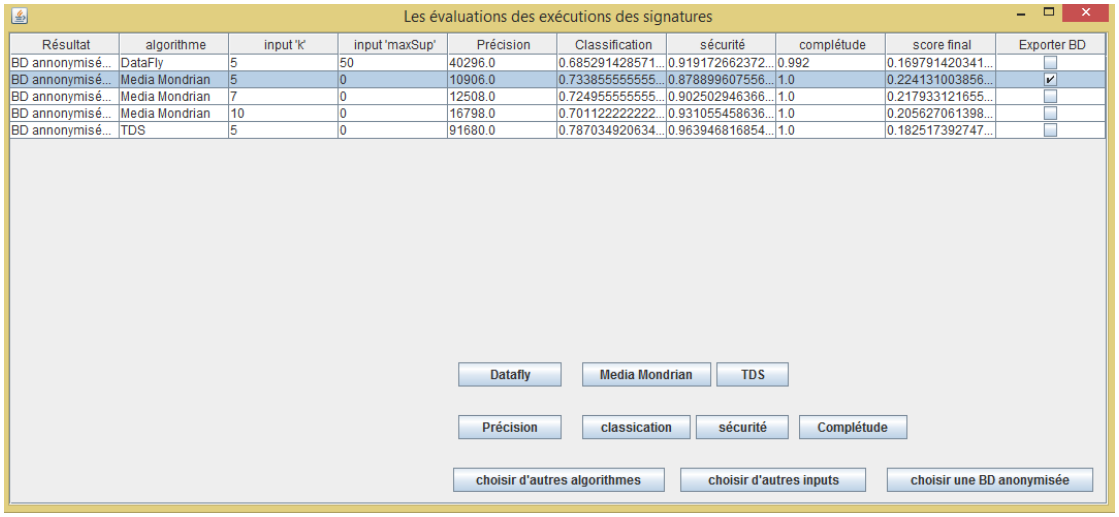


Figure 112. Interface 'Les évaluations des BD anonymisées'

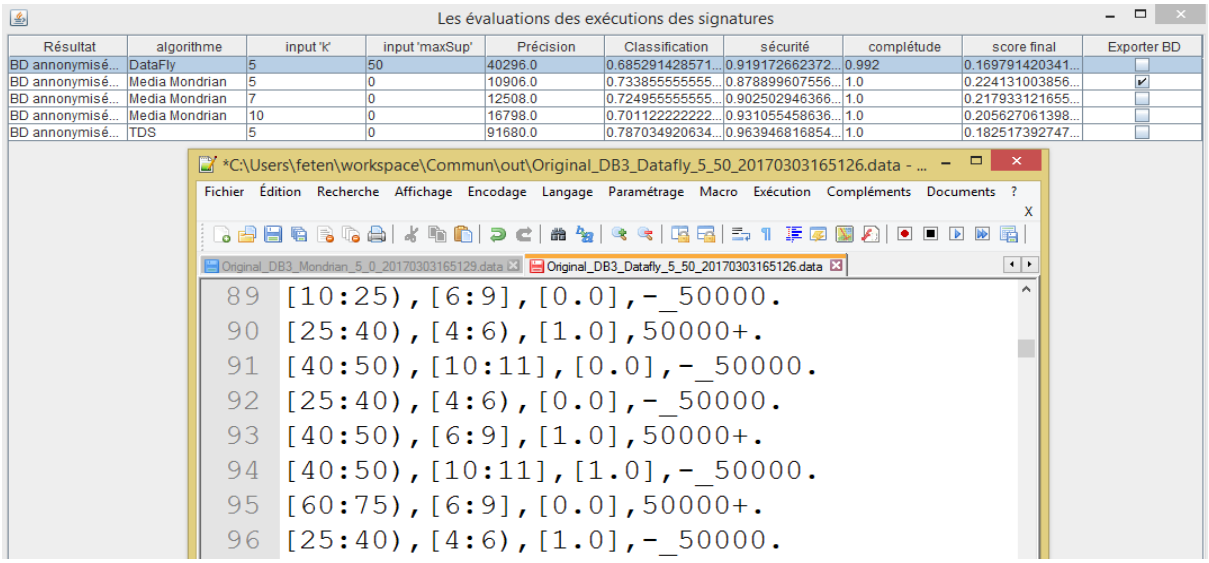


Figure 113. Analyse des données anonymisées

3. Evaluation de l'approche

Toute nouvelle approche nécessite une évaluation auprès de ces utilisateurs finaux afin d'évaluer son utilisabilité et son efficacité. MAGGO vise à aider les éditeurs de données à prendre des décisions concernant le choix des algorithmes d'anonymisation appliqués sur une base de données originales selon un contexte donné. Afin d'évaluer l'effet de cette aide à la décision sur l'utilisateur, nous avons réalisé une expérience contrôlée selon un modèle d'utilisabilité inspiré par ceux qui se trouvent dans la littérature.

3.1 Modèle d'utilisabilité

Nous avons extrait de la littérature les attributs d'évaluation les plus fréquemment utilisés dans les modèles d'utilisabilité. Ainsi, l'efficience, l'efficacité, l'apprentissage et la satisfaction sont les attributs les plus recensés (**Figure** 114).

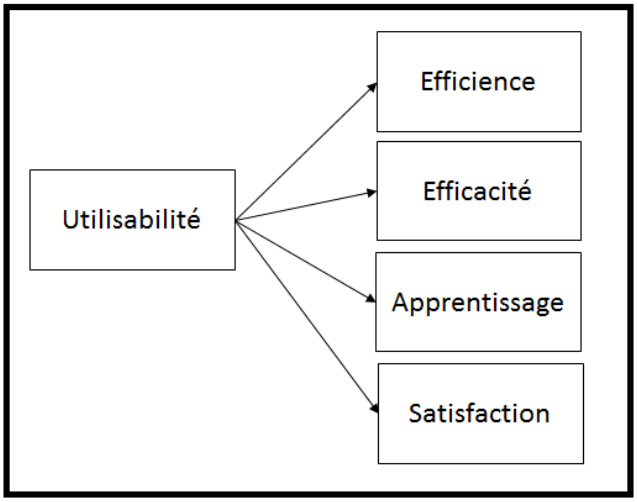


Figure 114**.** Modèle d'utilisabilité

Dans ce qui suit, nous allons définir les attributs d'évaluation:

- **Efficacité (utilité)**: L'efficacité focalise sur les résultats. Elle représente la capacité d'un système à exécuter avec succès la tâche spécifiée. Cet attribut reflète alors la qualité des buts à atteindre en d'autres termes, la qualité de la décision.
- **Efficience:** L'efficience est la capacité de l'utilisateur à accomplir ses tâches avec rapidité et précision. Cet attribut reflète la productivité d'un utilisateur pendant qu'il déroule l'application. L'efficience peut être mesurée de plusieurs manières, par exemple le temps passé ou le nombre de touches nécessaires pour accomplir une tâche donnée.
- **Apprentissage:** dans le domaine du logiciel, l'apprentissage représente la capacité d'apprendre les fonctionnalités du système, alors que dans le domaine de guidage décisionnel, il correspond à l'apprentissage des connaissances du domaine.
- **Satisfaction (attitude):** c'est le degré d'acceptation d'un système par les utilisateurs par rapport à leurs niveaux d'inconfort, de la fatigue, de la frustration et de l'effort personnel.

3.2 Procédure

Afin d'évaluer notre contribution, nous avons effectué une expérience auprès d'un groupe de participants. L'objectif était d'évaluer chaque type de guidage (guidage informatif et guidage suggestif) selon le modèle d'utilisabilité présenté ci-dessus. Tous les participants doivent effectuer la même tâche de décision dans un environnement contrôlé.

Seize participants ont été recrutés. Ils étaient tous soit des étudiants doctorants, soit des chercheurs en informatique et ne possèdent ni des expériences, ni des connaissances dans le domaine de l'anonymisation. Par conséquent, nous considérons que tous les participants possèdent le même profil tant dans le domaine informatique que dans le domaine de l'anonymisation.

Nous avons construit quatre versions d'outil. Chaque version est basée sur un type de guidage différent:

- *Version avec guidage Informatif prédéfini* (outils existant) : ce guidage ressemble à celui qui se trouve dans les outils actuels de l'anonymisation, lesquels offrent à leurs utilisateurs une aide sous forme d'un tutorial. les connaissances du domaine sont généralement décrites de façon très formelle ou font référence à des articles de recherche. Les étapes du processus d'anonymisation sont comme suit: l'utilisateur choisit un ou plusieurs algorithmes parmi un ensemble d'algorithmes → il fournit les valeurs des paramètres d'entrée → il exécute les signatures qu'il a construites → il visualise les résultats → il choisit une BD anonymisée.
- *Version avec guidage informatif dynamique* : les étapes d'anonymisation pour ce groupe ressemble à celles de version précédente. Cependant, l'aide est fournie au cours des tâches d'anonymisation de l'utilisateur (guidage simultané). Par exemple, au moment du choix des algorithmes, l'outil offre des descriptions de ces algorithmes. Les connaissances du domaine sont présentées sous une forme simple et facile à comprendre

par l'utilisateur, quel que soit son niveau d'expertise en anonymisation et ses compétences en programmation. Les étapes du processus d'anonymisation sont comme suit : l'utilisateur choisit un ou plusieurs algorithmes parmi un ensemble d'algorithmes (le système fournit les abstractions de chacun)→ il fournit les valeurs des paramètres d'entrée (le système fournit les définitions de chacun) → il exécute les signatures qu'il a construit → il visualise les résultats (le système fournit les définitions des critères et des métriques d'évaluation) → il choisit une BD anonymisée.

- *Version avec guidage suggestif* : les étapes du processus d'anonymisation de ce groupe sont différentes à celles qui se trouvent dans les deux groupes précédents. Elles sont comme suit : l'utilisateur fournit son contexte→ il visualise les résultats estimatifs des signatures proposées et triées selon son adéquation au contexte →il choisit une ou plusieurs signatures à exécuter →il visualise les résultats réels des BD anonymisées résultantes des signatures, triées selon son adéquation au contexte → il choisit une BD anonymisée. Le même tutorial du premier groupe est fourni pour représenter les connaissances du domaine.
- *Version avec guidage informatif dynamique et suggestif* (MAGGO): l'outil offre pour ce groupe le guidage informatif et simultané proposé dans le premier groupe et le guidage suggestif proposé dans le troisième groupe. De ce fait, les étapes du processus d'anonymisation sont comme suit : l'utilisateur fournit son contexte→ il visualise les résultats estimatifs des signatures proposées et triées selon son adéquation au contexte (le système fournit les abstractions des algorithmes, les définitions des paramètres d'entrée et les définitions des critères et des métriques d'évaluation) →il choisit une ou plusieurs signatures à exécuter →il visualise les résultats réels des BD anonymisées résultantes des signatures, triées selon son adéquation au contexte (le système fournit les abstractions des algorithmes, les définitions des paramètres d'entrée et les définitions des critères et des métriques d'évaluation) → il choisit une BD anonymisée. Le même tutorial du premier groupe est fourni pour représenter les connaissances du domaine.

Le but de ces versions est de comparer : d'une part l'outil MAGGO qui est pleinement mise en œuvre dans la dernière version, par rapport aux outils existants qui sont présents dans la première version. D'autre part, le guidage informatif dynamique qui est proposé dans la deuxième version s'oppose au guidage suggestif présent dans la troisième version. Chaque participant a été assigné au hasard à l'une des quatre versions de l'outil (**Figure** 115); chaque version a été déroulée par quatre participants.

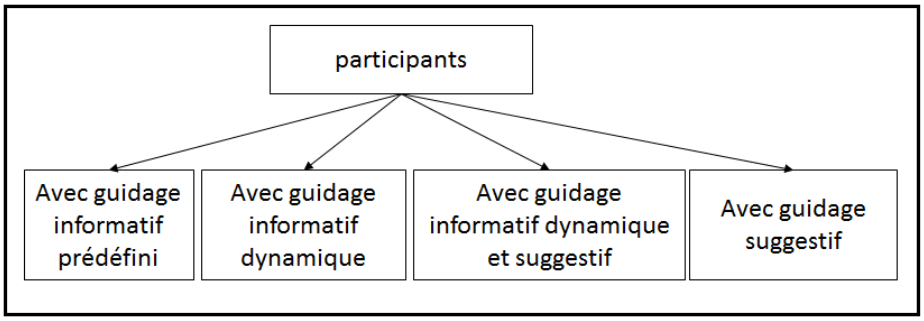


Figure 115**.** Les groupes des participants

Le contexte d'anonymisation décrit dans la section 3 de ce chapitre sera proposé à chaque participant. L'expérience est individuelle et dure à peu près entre une heure et demi et deux heures. Dans un premier temps, chaque participant reçoit une brève présentation orale de l'anonymisation en mettant l'accent sur la technique de généralisation. Ensuite, les données originales et leurs métas donnés (voir annexe C). Au cours de l'analyse des métas données, le participant est informé du fait que ceux-ci sont des paramètres d'entrées qu'un utilisateur doit spécifier. Cependant, pour l'expérience, nous les imposons puisque ce n'est pas le but de notre évaluation. Finalement, chaque participant a reçu le contexte d'anonymisation décrit dans la **Figure** 102. La présentation, les données originales, leurs métas donnés et le contexte sont identiques pour tous les participants.

Chaque participant est invité ensuite à anonymiser les données originales en choisissant la meilleur base de données anonymisée selon le contexte proposé. Il est observé par un chercheur. Le participant est encouragé à exprimer ses pensées à haute voix. L'horaire du début de ce processus d'anonymisation, l'horaire de fin, ses commentaires, ses efforts cognitifs et la base de données anonymisée choisie sont notés manuellement par le chercheur qui observe le participant.

Une fois le processus d'anonymisation finalisé, le participant est invité à répondre à un QCM composé de quinze questions afin d'évaluer son apprentissage des connaissances (voir annexe C). Le participant doit également évaluer, sur une échelle de 1 à 10, son niveau de satisfaction par rapport au guidage de l'outil. Afin d'éviter des résultats erronés, nous avons présenté les trois autres versions de l'outil à chaque participant avant d'évaluer sa satisfaction.

3.3 Résultats

Une fois l'expérience terminée, nous avons analysé les informations collectées représentées dans le tableau suivant :

groupe	efficacité (Qualité de la décision en %)	temps total (minutes)	Effort cognitif	Efficience de l'utilisateur (efficacité/temps total)	Efficience humaine (efficacité/ Effort)	Satisfaction (de 1 à 10)	Apprentissage (de 1 à 15)
Informatif prédéfini	45	75	4	0,6	11,25	4	6,5
Informatif dynamique et suggestif	98	35	0,75	2,8	130	9	11,5
Informatif dynamique	69	40	2,5	1,7	27	7	14
suggestif	98	20	1	4,9	98	8,5	9,5

Tableau 51**.** Synthèse des données collectées à partir de l'expérience

- **Efficacité:** est évaluée à travers la qualité de la décision des participants. Nous avons effectué plusieurs tests afin d'évaluer une dizaine de bases de données anonymisées selon le contexte proposé et nous avons les comparer selon la plateforme JAHP. L'alternative optimale est celle qui possède le meilleur score étant l'algorithme Median Mondrian avec la valeur 5 pour le paramètre d'entrée 'k'. Pour évaluer la qualité de décision pour chaque participant, nous comparons la solution choisie par rapport à la solution optimale selon les scores fournis par JAHP. La qualité de la décision est évaluée par : 1 - la différence entre les scores des

deux solutions. Par exemple, si la qualité de la décision est égale à 1 alors, l'utilisateur a choisi la solution optimale. Le **Tableau** 51 représente la moyenne de la qualité de décision pour chaque groupe. Par conséquent, nous observons que la qualité des décisions prises à l'aide du guidage informatif prédéfini (0,45) était inférieure à celle qui était à l'aide du guidage informatif dynamique et suggestif (0,98). La qualité de la décision avec le guidage suggestif (0,98) était plus élevée que celle avec le guidage informatif (0,69).

- **Satisfaction:** elle est évaluée sur une échelle de 1 à 10. Le **Tableau** 51 montre la moyenne de la satisfaction des participants dans chaque groupe. Les participants qui reçoivent le guidage dynamique et suggestif (9) sont beaucoup plus satisfaits que ceux qui reçoivent le guidage informatif prédéfini (4). Les participants qui reçoivent du guidage suggestif (9) sont légèrement satisfaits que ceux qui reçoivent le guidage informatifs (8.5).
- **Apprentissage:** le **Tableau** 51 présente la moyenne des réponses justes des QCM des participants dans chaque groupe. Les participants qui reçoivent le guidage informatif dynamique et suggestif (11,5) ont appris mieux que ceux qui reçoivent le guidage prédéfini (6,5). De même pour les participants qui reçoivent un guidage informatif dynamique (14), ont appris mieux que ceux qui reçoivent le guidage suggestif (9.5).
- **Efficience:** nous distinguons dans la littérature trois types de métrique pour l'évaluation de l'efficience à savoir : l'efficience de l'utilisateur, l'efficience humaine et l'efficience de l'entreprise qui dépendent respectivement du temps passé pour accomplir une tâche, l'effort de l'utilisateur et du coût total (qui est la somme de la main d'œuvre, du coût des ressources et du coût des formations). Nous nous intéressons aux deux premiers types de métrique. L'efficience de l'utilisateur est calculée par la qualité de la décision divisée par le temps total de la tâche à accomplir. D'après le **Tableau** 51 l'efficience de l'utilisateur des participants qui reçoivent le guidage informatif et suggestif (2,8) et nettement mieux que celle des participants qui reçoivent le guidage informatif prédéfini (0.6). De même, l'efficience de l'utilisateur des participants qui reçoivent le guidage suggestif (4,9) est nettement mieux que celle des participants qui reçoivent le guidage informatif (1,7). Concernant l'efficience humaine, nous avons capté quatre points d'effort d'un utilisateur au cours de son processus d'anonymisation à savoir : le choix des algorithmes, le choix des valeurs des paramètres d'entrée, L'analyses des valeurs des bases de données anonymisées selon les critères d'anonymisation et enfin, la comparaison des bases de données anonymisées ligne par ligne selon la hiérarchie des buts et selon ses préférences. Les comparaisons des groupes selon l'efficience humaine est semblable à celle de l'efficience utilisateur.

Au cours de nos observations des participants, nous avons capté deux différents types de profils ayant reçu le guidage suggestion : ceux qui ont confiance au système et ceux qui ne l'ont pas. Nous avons alors décidé d'analyser les résultats des deux groupes qui ont reçu le guidage suggestif et le guidage suggestif et informatif dynamique selon les deux profils.

groupe	efficacité (Qualité de la décision en %)	temps total (minutes)	Effort cognitif	Efficience de l'utilisateur (efficacité/temps total)	Efficience humaine (efficacité/ Effort)	Satisfaction (de 1 à 10)	Apprentissage (de 1 à 15)
Participants ont confiance à l'outil	98	12	0,33	8,16	296	8,33	10,8
Participants n'ont pas confiance à l'outil	98	43	2,33	2,27	42	9	11,33

Tableau 52**.** Synthèse des données collectées des participants qui ont reçu le guidage suggestif

Le **Tableau** 52 montre que les participants qui n'ont pas confiance à l'outil ont fait des efforts cognitifs et ont passé plus de temps afin de choisir la meilleure solution, à l'inverse des participants qui ont confiance à l'outil. Ces participants demandent des explications sur valeurs des paramètres d'entrée proposées et les scores affichés des signatures. Les évaluations de la satisfaction et de l'apprentissage des deux groupes sont presque égales.

4. Conclusion

Ce chapitre est consacré à la présentation de quelques aspects liés à la mise en œuvre et à l'évaluation de MAGGO. La première version de la plateforme vise à satisfaire quelques exigences fonctionnelles décrites dans le diagramme de cas de UML. Cette plateforme a été développée par des modules afin d'assurer son évolutivité. Les résultats de l'évaluation ont montré que le guidage fournit par MAGGO est plus utile que le guidage fourni par les outils existants. Premièrement, MAGGO améliore la qualité des décisions, augmente la satisfaction des utilisateurs, aide l'utilisateur à apprendre davantage sur le domaine d'anonymisation, et raccourcit le temps consacré à la prise de décision. Deuxièmement, le guidage informatif dynamique proposé par MAGGO permet à l'utilisateur d'apprendre d'avantage tandis que le guidage suggestif améliore la qualité de la décision, raccourci le temps de décision et diminue l'effort cognitif de l'utilisateur.

Il est à signaler aussi qu'au cours de cette expérience, nous avons détecté un facteur important qui augmente l'efficience de l'utilisateur et l'efficience humaine du guidage suggestif. Ce facteur est la confiance d'un utilisateur envers l'outil.

A partir de ces constats, nous proposons d'améliorer notre plateforme dans sa future version qui permet d'assurer la confiance de ses utilisateurs. Ceci pourrait être réalisé en offrant à l'utilisateur un guidage informatif qui permet à la fois d'améliorer son apprentissage, de lui argumenter et de lui expliquer les propositions fournies par le guidage suggestif.

Chapitre 8 conclusion

La donnée est une source d'avantage concurrentiel. Elle est la clé de la performance des organisations. Les données sont dorénavant volumineuses et partagées au-delà des frontières même d'une organisation. Cependant, ces collections de données contiennent des données personnelles (le salaire, l'état de santé, la religion sont des données personnelles). La protection de la vie privée est un droit fondamental dont la définition doit être adaptée à l'ère numérique. Les propriétaires de données se trouvent alors dans une situation qui exige de satisfaire deux buts contradictoires : respecter la confidentialité des données et, en même temps, préserver leur utilité.

Plusieurs techniques et algorithmes d'anonymisation des données sont proposés, lesquels modifient les données originales afin de minimiser les risques de ré-identification tout en sauvegardant autant que possible l'utilité de ces données. De plus, il est impossible de proposer un seul algorithme qui s'adapte à tous les contextes et qui donne le meilleur résultat à chaque fois. D'après notre analyse des travaux existants, nous avons établi que le choix du « bon » algorithme dépend d'un certain nombre de paramètres de contexte tels que les caractéristiques de la base de données, le besoin de l'anonymisation, etc. Plusieurs outils d'anonymisation existent afin d'offrir à l'éditeur de données la possibilité d'appliquer ces techniques d'anonymisation.

Notre étude sur les outils actuels nous a permis de constater qu'ils sont opaques dans la description des algorithmes qui implémentent les techniques et, même quand ces outils offrent des descriptions de leurs algorithmes, celles-ci ne sont pas compréhensibles par les éditeurs de données ayant des faibles compétences en programmation et en algorithmique. Il est donc nécessaire de fournir des descriptions claires afin de faciliter le choix des algorithmes.

Le décalage entre le petit nombre d'algorithmes d'anonymisation implémentés dans les outils et l'ensemble de ceux publiés dans la littérature, nous a amenés, après un état de l'art détaillé, à proposer une structure générique qui permettrait de stocker toutes les techniques d'anonymisation.

Enfin, au cours d'un processus d'anonymisation, l'éditeur de données doit choisir le meilleur algorithme qui s'adapte le mieux à son contexte. L'aide des outils actuels consiste à exécuter les algorithmes et présenter une évaluation des jeux de données anonymisées qui en résultent. Si l'éditeur de données est insatisfait, l'outil lui offre la possibilité d'ajuster les paramètres des algorithmes. L'absence d'une aide, avant même le choix des algorithmes et des valeurs des paramètres, nous a amenés à proposer une approche complète de guidage au cours des différentes étapes du processus d'anonymisation.

Pour résumer, nos contributions de recherche sont de plusieurs ordres. La première contribution concerne le travail de caractérisation des algorithmes de généralisation qui nous a permis d'identifier des paramètres de comparaison entre ces algorithmes. Ces paramètres ont constitué pour nous une source d'inspiration pour construire les critères d'évaluation de ces algorithmes qui doivent contribuer à leur sélection.

La deuxième contribution méthodologique est la proposition d'un guidage informatif via des représentations simplifiées des algorithmes de généralisation, obtenues suite à l'application d'un processus d'abstraction par paramétrage. Ces représentations sont beaucoup plus intelligibles, notamment pour des personnes sans compétence en programmation. L'abstraction de ces algorithmes nous a permis d'identifier des similarités et de procéder à leur regroupement au sein de trois catégories de techniques auxquelles nous avons aussi associé des abstractions. Un dernier effort d'abstraction par généralisation nous a permis de fournir une description abstraite de la technique de généralisation elle-même.

La troisième contribution est une ontologie de domaine (OPAM) offrant des définitions rigoureuses et formelles des concepts mis en jeu dans le domaine de l'anonymisation. Elle représente une source d'information évolutive, exploitée par le processus d'anonymisation.

La quatrième contribution est une approche guidée d'anonymisation exploitant l'ontologie OPAM. Cette approche, que nous nommons MAGGO (Méthodologie pour une Anonymisation par Généralisation Guidée par une Ontologie), sert de guide pour un professionnel dans sa prise de décision lors du choix et du paramétrage de l'algorithme adéquat au contexte de l'anonymisation. Elle propose un guidage informatif exploitant les descriptions simplifiées des algorithmes et un guidage suggestif en se fondant sur une méthode de régression statistique et une méthode d'aide multicritère à la décision qui suggère un ensemble d'algorithmes évalués selon leur adéquation au contexte.

La dernière contribution est d'ordre pratique. Elle concerne la mise en œuvre d'une plateforme dédiée au guidage d'un éditeur de données au cours de son processus d'anonymisation.

Perspectives de recherche

Notre travail peut être poursuivi dans au moins trois axes qu'on peut classer par ordre de priorité :

Axe 1: Enrichir l'ontologie OPAM par d'autres techniques d'anonymisation

Axe 2: Améliorer notre approche de guidage de processus d'anonymisation en proposant des combinaisons d'algorithmes appartenant à plusieurs techniques selon un contexte. Une autre amélioration concerne le choix des méthodes de régression et d'aide à la décision.

Axe 3: Amélioration du guidage suggestif en donnant des arguments afin de justifier les scores attribués aux algorithmes proposés.

Des perspectives d'ordre pratique peuvent être menées en parallèle avec les perspectives de recherche citées plus haut. Ces perspectives concernent le développement d'un module permettant d'offrir à un expert du domaine la possibilité d'enrichir l'ontologie OPAM.

Chapitre 9 Liste des publications de cette thèse

F. Ben Fredj, N. Lammari, I. Comyn-Wattiau. "Approche guidée pour l'anonymisation de base de données", soumis à INFORSID 2017, Juin 2017, Toulouse, France,

F. Ben Fredj, N. Lammari, I. Comyn-Wattiau. "L'anonymisation des données par généralisation : un arbre de décision", Titre du livre: "Ingénierie et management des systèmes d'information", December 2016, Cepadues, pp. 227-241, (isbn: 978.2.36493.573.0)

F. Ben Fredj, N. Lammari, I. Comyn-Wattiau. "Abstracting Anonymization Techniques: A Prerequisite for Selecting a Generalization Algorithm", KES 2015 (19th International Conference on Knowledge Based and Intelligent Information & Engineering Systems), September 2015, pp.en cours d'édition, Singapore,

F. Ben Fredj, N. Lammari, I. Comyn-Wattiau. "Building an Ontology to Capitalize and Share Knowledge on Anonymization Techniques", ECKM 2015 (16th European Conference on Knowledge Management), September 2015, pp.en cours d'edition, Udine, Italy,

F. Ben Fredj, N. Lammari, I. Comyn-Wattiau. "Characterizing Generalization Algorithms-First Guidelines for Data Publishers", KMIS 2014- International Conference on Knowledge Management and Information Sharing, October 2014, pp.pp, Rome, Italy,

Bibliographie

Abbott, Russ, et Chengyu Sun. 2008. « Abstraction abstracted ». In *Proceedings of the 2nd International Workshop on the Role of Abstraction in Software Engineering*, 23–30. ACM. http://dl.acm.org/citation.cfm?id=1370171.

Aïmeur, Esma. 2009. « Data Mining and Privacy ». In Encyclopedia of Data Warehousing and Mining, Second Edition, 388–393. IGI Global. http://www.igi-global.com/chapter/encyclopedia-data-warehousing-mining-second/10849.

Ayala-Rivera, Vanessa, Patrick McDonagh, Thomas Cerqueus, et Liam Murphy. 2014. « A Systematic Comparison and Evaluation of k-Anonymization Algorithms for Practitioners. » *Trans. Data Privacy* 7 (3): 337–370.

Babu, Korra Sathya, Nithin Reddy, Nitesh Kumar, Mark Elliot, et Sanjay Kumar Jena. 2013. « Achieving k-anonymity Using Improved Greedy Heuristics for Very Large Relational Databases. » *Trans. Data Privacy* 6 (1): 1–17.

Bayardo, Roberto J., et Rakesh Agrawal. 2005. « Data privacy through optimal k-anonymization ». In *Data Engineering, 2005. ICDE 2005. Proceedings. 21st International Conference on*, 217–228. IEEE. http://ieeexplore.ieee.org/abstract/document/1410124/

.

Blum, Avrim, Cynthia Dwork, Frank McSherry, et Kobbi Nissim. 2005. « Practical privacy: the SuLQ framework ». In *Proceedings of the twenty-fourth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, 128–138. ACM. http://dl.acm.org/citation.cfm?id=1065184.

Blum, Avrim, Katrina Ligett, et Aaron Roth. 2008. « A learning theory approach to non-interactive database privacy. » *the 40th Annual ACM Symposium on Theory of Computing (STOC)*, 609–618.

Brand, Ruth. 2002. « Microdata Protection through Noise Addition ». *Inference Control in Statistical Databases, From Theory to Practice*, 97-116.

Breiman, Leo, Jerome Friedman, Charles J. Stone, et R.A. Olshen. 1984. *Classification and Regression Trees*. Wadsworth Statistics/Probability.

Burton, Robert, Anco J. Hundepool, Leon CRJ Willenborg, Lawrence H. Nitz, et Karl E. Kim. 1997. « Record Linkage ». In *Record Linkage Techniques-1997: Proceedings of an International Workshop and Exposition, March 20-21, 1997, Arlington, Va*, 139. National Academies.

Cardoso, Jorge, et A. Sheth, éd. 2006. *Semantic Web services, processes and applications*. Semantic web and beyond 3. New York, NY: Springer.

Dai, Chenyun, Gabriel Ghinita, Elisa Bertino, Ji-Won Byun, et Ninghui Li. 2009. « TIAMAT: a tool for interactive analysis of microdata anonymization techniques ». *Proceedings of the VLDB Endowment* 2 (2): 1618–1621.

Dalenius, Tore. 1977. « Towards a methodology for statistical disclosure control ». Statistisk Tidskrift.

Dalenius, Tore, et Steven P. Reiss. 1982. « Data-Swapping: A Technique for Disclosure Control ». *Journal of Statistical Planning and Inference* 6 (1): 73-85. doi:10.1016/0378-3758(82)90058-1.

Defays, D., et P. Nanopoulos. 1992. « Panels of enterprises and confidentiality: the small aggregates method ». *Design and Analysis of Longitudinal Surveys*.

Domingo-Ferrer, Josep, et Vicenc Torra. 2001. « A quantitative comparison of disclosure control methods for microdata ». *Confidentiality, disclosure and data access: theory and practical applications for statistical agencies*, 111–134.

Domingo-Ferrer, Josep, et Vicenç Torra. 2002. « Distance-based and probabilistic record linkage for re-identification of records with categorical variables ». *Butlletí de lACIA, Associació Catalana dIntelligència Artificial*, 243–250.

Dwork, Cynthia. 2008. « Differential privacy: A survey of results ». In *International Conference on Theory and Applications of Models of Computation*, 1–19. Springer. http://link.springer.com/chapter/10.1007/978-3-540-79228-4_1.

FINANCIER, DECISION POUR LE SECTEUR. 2015. « EQUI SE P UNIVERSITE PAUL SABATIER DE TOULOUSE ». https://www.irit.fr/publis/SMAC/DOCUMENTS/PUBLIS/RAKOTOARIVELO_REPORT1_Juin_2015.pdf.

Fung, Benjamin C. M., Ke Wang, Rui Chen, et Philip S. Yu. 2010. « Privacy-Preserving Data Publishing: A Survey of Recent Developments ». *ACM Computing Surveys* 42 (4): 1-53. doi:10.1145/1749603.1749605.

Fung, Benjamin CM, Ke Wang, et Philip S. Yu. 2005. « Top-down specialization for information and privacy preservation ». In *Data Engineering, 2005. ICDE 2005. Proceedings. 21st International Conference on*, 205–216. IEEE. http://ieeexplore.ieee.org/abstract/document/1410123/.

Gómez-Pérez, Asunción, et Richard Benjamins. 1999. « Overview of knowledge sharing and reuse components: Ontologies and problem-solving methods ». In . IJCAI and the Scandinavian AI Societies. CEUR Workshop Proceedings. http://oa.upm.es/6468/1/Overview_of_Knowledge.pdf.

Gruber, Thomas R., et others. 1993. « A translation approach to portable ontology specifications ». *Knowledge acquisition* 5 (2): 199–220.

Gr□uninger, Michael, et Mark S. Fox. 1995. « Methodology for the Design and Evaluation of Ontologies ». *Workshop on Basic Ontological Issues in Knowledge Sharing*.

Issa, Romeo. 2009. « Satisfying K-Anonymity: New Algorithm and Empirical Evaluation ». Carleton UNIVERSITY Ottawa.

Iyengar, Vijay S. 2002. « Transforming data to satisfy privacy constraints ». In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, 279–288. ACM. http://dl.acm.org/citation.cfm?id=775089.

Juan, Vara Mesa. 2009. « M2DAT: a Technical Solution for Model-Driven Development of Web Information Systems ».

Kaur Arora, Dilpreet, Divya Bansal, et Sanjeev Sofat. 2014. « Comparative Analysis of Anonymization Techniques ». *International Journal of Electronic and Electrical Engineering.*

Kiker, Gregory A., Todd S. Bridges, Arun Varghese, Thomas P. Seager, et Igor Linkov. 2005. « Application of multicriteria decision analysis in environmental decision making ». *Integrated environmental assessment and management* 1 (2): 95–108.

Kiran, P., et N. P. Kavya. 2012. « A Survey on Methods, Attacks and Metric for Privacy Preserving Data Publishing ». *International Journal of Computer Applications* 53 (18). http://search.proquest.com/openview/40944937a3f6257e1e9c8d7af4bded49/1?pq-origsite=gscholar&cbl=136216.

Kramer, Jeff. 2007. « Is abstraction the key to computing? » *Communications of the ACM* 50 (4): 36–42.

Kramer, Jeff, et Orit Hazzan. 2006. « The role of abstraction in software engineering ». In *Proceedings of the 28th international conference on Software engineering*, 1017–1018. ACM. http://dl.acm.org/citation.cfm?id=1134481.

Lawrence H., Cox. 1980. « Suppression methodology and statistical disclosure analysis ». *Journal of the American Statistical Association*.

LeFevre, Kristen, David J. DeWitt, et Raghu Ramakrishnan. 2005. « Incognito: Efficient full-domain k-anonymity ». In *Proceedings of the 2005 ACM SIGMOD international conference on Management of data*, 49–60. ACM. http://dl.acm.org/citation.cfm?id=1066164.

LeFevre Kristen, David J. DeWitt, Raghu Ramakrishnan. 2006. « Mondrian multidimensional k-anonymity ». In *Data Engineering, 2006. ICDE'06. Proceedings of the 22nd International Conference on*, 25–25. IEEE. http://ieeexplore.ieee.org/abstract/document/1617393/.

LeFevre Kristen, David J. DeWitt, Raghu Ramakrishnan. 2008. « Workload-Aware Anonymization Techniques for Large-Scale Datasets ». *ACM Transactions on Database Systems* 33 (3): 1-47. doi:10.1145/1386118.1386123.

Li, Ninghui, Tiancheng Li, et Suresh Venkatasubramanian. 2007. « t-closeness: Privacy beyond k-anonymity and l-diversity ». In *Data Engineering, 2007. ICDE 2007. IEEE 23rd International Conference on*, 106–115. IEEE. http://ieeexplore.ieee.org/abstract/document/4221659/.

Li, Tiancheng, Ninghui Li, Jian Zhang, et Ian Molloy. 2012. « Slicing: A New Approach for Privacy Preserving Data Publishing ». *IEEE Transactions on Knowledge and Data Engineering* 24 (3): 561-74. doi:10.1109/TKDE.2010.236.

Liskov, Barbara, et John Guttag. 2000. *Program Development in Java: Abstraction, Specification, and Object-Oriented Design*. Addison-Wesley Longman Publishing.

Machanavajjhala, Ashwin, Daniel Kifer, Johannes Gehrke, et Muthuramakrishnan Venkitasubramaniam. 2007. « l-diversity: Privacy beyond k-anonymity ». *ACM Transactions on Knowledge Discovery from Data (TKDD)* 1 (1): 3.

Mardani, Abbas, Ahmad Jusoh, Khalil MD Nor, Zainab Khalifah, Norhayati Zakwan, et Alireza Valipour. 2015. « Multiple Criteria Decision-Making Techniques and Their Applications – a Review of the Literature from 2000 to 2014 ». *Economic Research-Ekonomska Istraživanja* 28 (1): 516-71. doi:10.1080/1331677X.2015.1075139.

Martin, David J., Daniel Kifer, Ashwin Machanavajjhala, Johannes Gehrke, et Joseph Y. Halpern. 2007. « Worst-case background knowledge for privacy-preserving data publishing ». In *Data Engineering, 2007. ICDE 2007. IEEE 23rd International Conference on*, 126–135. IEEE. http://ieeexplore.ieee.org/abstract/document/4221661/.

Matthews, Gregory J., et Ofer Harel. 2011. « Data Confidentiality: A Review of Methods for Statistical Disclosure Limitation and Methods for Assessing Privacy ». *Statistics Surveys* 5 (0): 1-29. doi:10.1214/11-SS074.

Mena, S. Ben. 2000. « Introduction aux méthodes multicritères d'aide à la décision ». *Biotechnologie, Agronomie, Société et Environnement* 4 (2): 83–93.

Miller, George A. 1967. « The magical number seven, plus-or-minus two, some limits to our capacity for processing information ». *Brain Physiology and Psychology. Buttenvorths: London*, 175–200.

Morana, Stefan, Silvia Schacht, Ansgar Scherp, et Alexander Maedche. 2014. « Conceptualization and Typology of Guidance in Information Systems ». *Working Paper Series in Information Systems* 7. https://ub-madoc.bib.uni-mannheim.de/36876/.

Mrinmoy, Majunder. 2015. « Multi Criteria Decision Making ». In *, Impact of Urbanization on Water Shortage in Face of Climatic Aberrations*, SpringerBriefs in Water Science and Technology, 35-47.

Narayanan, Arvind, et Vitaly Shmatikov. 2006. « How to break anonymity of the netflix prize dataset ». *arXiv preprint cs/0610105*. https://arxiv.org/abs/cs/0610105.

Návrat, Pavol, et Roman Filkorn. 2005. « A Note on the Role of Abstraction and Generality in Software Development ». *Journal of Computer Science* 1 (1): 98–102.

Nergiz, Mehmet Ercan, Maurizio Atzori, et Chris Clifton. 2007. « Hiding the presence of individuals from shared databases ». In *Proceedings of the 2007 ACM SIGMOD international conference on Management of data*, 665–676. ACM. http://dl.acm.org/citation.cfm?id=1247554.

Parikh, Mihir, Bijan Fazlollahi, et Sameer Verma. 2001. « The effectiveness of decisional guidance: an empirical evaluation ». *Decision Sciences* 32 (2): 303–332.

Poulis, Giorgos, Aris Gkoulalas-Divanis, Grigorios Loukides, Spiros Skiadopoulos, et Christos Tryfonopoulos. 2014. « SECRETA: a system for evaluating and comparing relational and transaction anonymization algorithms ». http://orca.cf.ac.uk/id/eprint/59440.

Psyché, Valéry, Olavo Mendes, et Jacqueline Bourdeau. 2003. « Apport de l'ingénierie ontologique aux environnements de formation à distance ». *Sciences et Technologies de l'Information et de la Communication pour l'Éducation et la Formation (STICEF)* 10: 89–126.

roy, Bernard. 1999. « Decision-Aiding Today: What Should We Expect? » In *Multicriteria Decision Making: Advances in MCDM Models, Algorithms, Theory and applications.*, 1999 springer Science+Business Media, 1--35. Tomas Gal,Theo Stewart,Thomas Hanne.

Rubner, Yossi, Carlo Tomasi, et Leonidas J. Guibas. 2000. « The earth mover's distance as a metric for image retrieval ». *International journal of computer vision* 40 (2): 99–121.

Saaty, Thomas L. 2004. « Decision making—the analytic hierarchy and network processes (AHP/ANP) ». *Journal of systems science and systems engineering* 13 (1): 1–35.

Samarati, Pierangela. 2001. « Protecting respondents identities in microdata release ». *IEEE transactions on Knowledge and Data Engineering* 13 (6): 1010–1027.

Samarati, Pierangela, et Latanya Sweeney. 1998. « Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression ». Technical report, SRI International. http://epic.org/privacy/reidentification/Samarati_Sweeney_paper.pdf.

Shen, Milton, Melody Carswell, Radhika Santhanam, et Kyle Bailey. 2012. « Emergency Management Information Systems: Could Decision Makers Be Supported in Choosing Display Formats? » *Decision Support Systems* 52 (2): 318-30. doi:10.1016/j.dss.2011.08.008.

Silver, Mark S. 2006. « Broadening the Scope ». *Human-Computer Interaction and Management Information Systems: Foundations*, 90.

Staab, Steffen, et Rudi Studer, éd. 2009. *Handbook on Ontologies*. Berlin, Heidelberg: Springer Berlin Heidelberg. doi:10.1007/978-3-540-92673-3.

Steinberg, Dave, Frank Budinsky, Ed Merks, et Marcelo Paternostro. 2008. *EMF: Eclipse Modeling Framework*. Pearson Education.

Studer, Rudi, V. Richard Benjamins, et Dieter Fensel. 1998. « Knowledge engineering: principles and methods ». *Data & knowledge engineering* 25 (1-2): 161–197.

Suárez-Figueroa, Mari Carmen, Asunción Gómez-Pérez, et Mariano Fernández-López. 2012. « The NeOn Methodology for Ontology Engineering ». In *Ontology Engineering in a Networked World*, édité par Mari Carmen Suárez-Figueroa, Asunción Gómez-Pérez, Enrico Motta, et Aldo Gangemi, 9-34. Berlin, Heidelberg: Springer Berlin Heidelberg. doi:10.1007/978-3-642-24794-1_2.

Sweeney, Latanya. 1997. « Datafly: a System for Providing Anonymity in Medical Data ». *the Eleventh International Conference on Database Securty XI: Status and Prospects*, 356-81.

Sweeney, Latanya. 2002a. « Achieving k-anonymity privacy protection using generalization and suppression ». *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 10 (05): 571–588.

Sweeney, Latanya. 2002b. « k-anonymity: A model for protecting privacy ». *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 10 (05): 557–570.

Tsui, Frank, Abdolrashid Gharaat, Sheryl Duggins, et Edward Jung. 2011. « Measuring Levels of Abstraction in Software Development. » In *SEKE*, 466–469. https://www.researchgate.net/profile/Frank_Tsui/publication/221390920_Measuring_Levels_of_Abstraction_in_Software_Development/links/00b7d5193975f0058f000000.pdf.

Vaghashia, Hina, et Amit Ganatra. 2015. « A survey: privacy preservation techniques in data mining ». *International Journal of Computer Applications* 119 (4). http://search.proquest.com/openview/b16452866259def957dd143cd99d98f6/1?pq-origsite=gscholar&cbl=136216.

Verykios, Vassilios S., Elisa Bertino, Igor Nai Fovino, Loredana Parasiliti Provenza, Yucel Saygin, et Yannis Theodoridis. 2004. « State-of-the-art in privacy preserving data mining ». *ACM Sigmod Record* 33 (1): 50–57.

Wagner, Stefan, et Florian Deissenboeck. 2008. « Abstractness, specificity, and complexity in software design ». In *Proceedings of the 2nd International Workshop on the Role of Abstraction in Software Engineering*, 35–42. ACM. http://dl.acm.org/citation.cfm?id=1370173.

Wang, Ke, Philip S. Yu, et Sourav Chakraborty. 2004. « Bottom-up generalization: A data mining solution to privacy protection ». In *Data Mining, 2004. ICDM'04. Fourth IEEE International Conference on*, 249–256. IEEE. http://ieeexplore.ieee.org/abstract/document/1410291/.

Wing, J. M. 2008. « Computational Thinking and Thinking about Computing ». *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* 366 (1881): 3717-25. doi:10.1098/rsta.2008.0118.

Xiao, Xiaokui, et Yufei Tao. 2006. « Anatomy: Simple and effective privacy preservation ». In *Proceedings of the 32nd international conference on Very large data bases*, 139–150. VLDB Endowment. http://dl.acm.org/citation.cfm?id=1164141.

Xiao, Xiaokui, Guozhang Wang, et Johannes Gehrke. 2009. « Interactive anonymization of sensitive data ». In *Proceedings of the 2009 ACM SIGMOD International Conference on Management of data*, 1051–1054. ACM. http://dl.acm.org/citation.cfm?id=1559979.

Xu, Jian, Wei Wang, Jian Pei, Xiaoyuan Wang, Baile Shi, et Ada Wai-Chee Fu. 2006. « Utility-based anonymization using local recoding ». In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, 785–790. ACM. http://dl.acm.org/citation.cfm?id=1150504.

Xu, Ling, et Jian-Bo Yang. 2001. *Introduction to multi-criteria decision making and the evidential reasoning approach*. Manchester School of Management. http://www.academia.edu/download/31563947/Introduction_to_Multi-Criteria_Decision_Making_evidential_reasoning_approach.pdf.

Xu, Yang, Tinghuai Ma, Meili Tang, et Wei Tian. 2014. « A Survey of Privacy Preserving Data Publishing using Generalization and Suppression ». *Applied Mathematics & Information Sciences* 8 (3): 1103-16. doi:10.12785/amis/080321.

[1] http://ec.europa.eu/justice/data-protection/files/4_strenghten_2016_en.pdf

[2] https://www.iso.org/fr/news/2009/03/Ref1209.html

[3]http://ec.europa.eu/justice/data-protection/article-29/documentation/opinion-recommendation/files/2014/wp216_en.pdf

[4] http://neon.vb.cbs.nl/casc/mu.htm

[5]http://neon.vb.cbs.nl/casc/Software/MUmanual5.1.pdf

[6] http://users.uop.gr/~poulis/SECRETA/index.html

[7] http://www.privacyanalytics.ca/software/parat/

[8] https://www.nahdo.org/sites/nahdo.org/files/conference_sessions/PrivacyAnalytics.pdf

[9] http://arx.deidentifier.org/development/algorithms/

[10] https://en.wikipedia.org/wiki/Analytic_hierarchy_process_%E2%80%93_car_example

Annexes

Annexe A Exécution des algorithmes de généralisation

Dans cette annexe, nous allons appliquer les neufs algorithmes de généralisation les plus importants sur la table originale représentée dans le troisième chapitres **Tableau 24** en utilisant les hiérarchies de généralisation des attributs QI (**Figure 7**, **Figure 8** et **Figure 9**). Parfois la table originale est légèrement modifiée afin de faciliter l'exécution d'un algorithme de généralisation. On affecte la valeur de 2 pour la valeur du paramètre d'entrée 'k' et 2 pour 'maxSup'. Finalement, les enregistrements qui ne satisfont pas le k-anonymat sont marqués en rouge.

L'algorithme µ-argus

µ-Argus est un algorithme itératif. A chaque itération :

3. Le « data publisher » choisit l'attribut qui va être généralisé.
4. µ-Argus remplace chaque valeur de cet attribut par la valeur de son parent direct dans la hiérarchie de généralisation correspondante et affiche au « data publisher » les enregistrements qui ne satisfont pas le k-anonymat appelés (outliers).
5. µ-Argus vérifie si la table résultante satisfait le k-anonymat.
6. Enfin, le « data publisher » choisit entre la poursuite du processus de généralisation ou l'application de la suppression locale des enregistrements qui ne satisfont pas le k-anonymat.

Dans ce qui suit, nous allons appliquer *µ-argus* sur la table T afin de satisfaire k=2.

Itération1 :

Etape 1 : Le « data publisher » décide de généraliser l'attribut niveau d'étude

Etape 2 : L'algorithme remplace la valeur de chaque attribut niveau d'étude par son parent direct dans la hiérarchie de généralisation. Huit enregistrements ne satisfont pas le k-anonymat.

sexe	code postal	niveau d'étude
M	13050	collège
F	13051	collége
M	13050	lycée
M	13050	lycée
M	13051	1ier et 2ième cycle
F	13050	1ier et 2ième cycle
F	13061	1ier et 2ième cycle
F	13061	3ième cycle
F	13060	3ième cycle

M	13061	3ième cycle
M	13060	3ième cycle
M	13061	3ième cycle

Itération 2 :

Etape 1: En se basant sur les outliers, le data publisher décide de généraliser par exemple le code postal.

Etape 2: l'algorithme généralise l'attribut code postal. Cinq enregistrements ne satisfont pas le k-anonymat.

Sexe	Code Postal	Niveau d'étude
M	1305*	collège
F	1305*	collège
M	1305*	lycée
M	1305*	lycée
M	1305*	1i^{er} et 2^{ième} cycle
F	1305*	1i^{er} et 2^{ième} cycle
F	1306*	1i^{er} et 2^{ième} cycle
F	1306*	3ième cycle
F	1306*	3ième cycle
M	1306*	3ième cycle
M	1306*	3ième cycle
M	1306*	3ième cycle

Etape 3 : La table ne satisfait pas k-anonymat puisqu'il existe encore cinq outliers.

Etape 4 : Le data publisher décide donc de continuer la généralisation

Itération 3 :

Etape 1: En se basant sur les outliers, le data publisher décide de généraliser par exemple l'attribut sexe.

Etape 2: l'algorithme généralise l'attribut sexe. Un seul enregistrement ne satisfait pas le k-anonymat.

Sexe	Code Postal	Niveau d'étude
Tout-sexe	1305*	collège
Tout-sexe	1305*	collège
Tout-sexe	1305*	lycée
Tout-sexe	1305*	lycée
Tout-sexe	1305*	1i^{er} et 2^{ième} cycle
Tout-sexe	1305*	1i^{er} et 2^{ième} cycle
Tout-sexe	1306*	1i^{er} et 2^{ième} cycle
Tout-sexe	1306*	3ième cycle
Tout-sexe	1306*	3ième cycle
Tout-sexe	1306*	3ième cycle

Tout-sexe	1306*	3ième cycle
Tout-sexe	1306*	3ième cycle

Etape 3 : La table ne satisfait pas k-anonymat puisqu'il existe encore 1 outlier.

Etape 4 : Le data publisher décide d'arrêter la généralisation et l'algorithme applique la suppression locale en supprimant la valeur de l'attribut niveau d'étude dans l'enregistrement qui ne satisfait pas k-anonymat.

Sexe	Code Postal	Niveau d'étude
Tout-sexe	1305*	collège
Tout-sexe	1305*	collège
Tout-sexe	1305*	lycée
Tout-sexe	1305*	lycée
Tout-sexe	1305*	1i^{er} et 2^{ième} cycle
Tout-sexe	1305*	1i^{er} et 2^{ième} cycle
Tout-sexe	1306*	*******
Tout-sexe	1306*	3ième cycle
Tout-sexe	1306*	3ième cycle
Tout-sexe	1306*	3ième cycle
Tout-sexe	1306*	3ième cycle
Tout-sexe	1306*	3ième cycle

L'algorithme Datafly

A chaque itération, Datafly :

1. Calcule le nombre de valeurs distinctes de chaque attribut du QI
2. Généralise les valeurs de l'attribut ayant le plus grand nombre de valeurs distinctes.
3. Calcule le nombre d'enregistrements qui ne satisfont pas le k-anonymat. Si ce nombre est inférieur à MaxSup, alors ces enregistrements sont supprimés de la table et l'algorithme s'arrête. Sinon, l'algorithme effectue une autre itération de généralisation.

Itération1 :

Etape 1 : Datafly calcule le nombre de valeurs distinctes dans la table originale. Les attributs sexe, code postal et niveau d'étude ont respectivement 2, 4 et 7 valeurs distinctes.

sexe	code postal	niveau d'étude
M	13050	5ième
F	13051	3ième
M	13050	Seconde
M	13050	Seconde
M	13051	1ier et 2ième cycle
F	13050	1ier et 2ième cycle
F	13061	1ier et 2ième cycle

F	13061	Master
F	13060	Master
M	13061	Doctorat
M	13060	Doctorat
M	13061	Doctorat
2	**4**	**7**

Etape 2: l'attribut qui a le plus grand nombre de valeurs distinctes est 'niveau d'éducation'. Datafly généralise cet attribut.

sexe	code postal	niveau d'étude
M	13050	collège
F	13051	collège
M	13050	lycée
M	13050	lycée
M	13051	1ier et 2ième cycle
F	13050	1ier et 2ième cycle
F	13061	1ier et 2ième cycle
F	13061	3ième cycle
F	13060	3ième cycle
M	13061	3ième cycle
M	13060	3ième cycle
M	13061	3ième cycle

Etape 3: Huit enregistrements ne satisfont pas le k-anonymat et 8 > 2. Alors Datafly passe à une autre itération.

Itération 2:

Etape 1: Les attributs sexe, code postal et niveau d'étude ont respectivement 2, 4 et 4 valeurs distinctes.

Etape 2: Les attributs code postal et niveau d'étude ont le plus grand nombre des valeurs distinctes. Datafly généralise l'attribut code postal.

Sexe	Code Postal	Niveau d'étude
M	1305*	collège
F	1305*	collège
M	1305*	Lycée
M	1305*	Lycée
M	1305*	1ier et 2ième cycle
F	1305*	1ier et 2ième cycle
F	1306*	1ier et 2ième cycle
F	1306*	3ième cycle
F	1306*	3ième cycle
M	1306*	3ième cycle
M	1306*	3ième cycle

M	1306*	3ième cycle

Etape 3: Cinq enregistrements ne satisfont pas le k-anonymat et 5> 2. Alors Datafly passe à une autre itération.

Itération 3:

Etape 1: Les attributs sexe, code postal et niveau d'étude ont respectivement 2, 2 et 4 de valeurs distinctes.

Etape 2: L'attribut code postal a le plus grand nombre des valeurs distinctes. Datafly généralise alors cet attribut.

Sexe	Code Postal	Niveau d'étude
M	1305*	Secondaire
F	1305*	Secondaire
M	1305*	Secondaire
M	1305*	Secondaire
M	1305*	Université
F	1305*	Université
F	1306*	Université
F	1306*	Université
F	1306*	Université
M	1306*	Université
M	1306*	Université
M	1306*	Université

Etape 3: un seul enregistrement ne satisfait pas le k-anonymat et 1<2 . Alors datafly supprime cet enregistrement.

Sexe	Code Postal	Niveau d'étude
M	1305*	Secondaire
M	1305*	Secondaire
M	1305*	Secondaire
M	1305*	Université
F	1305*	Université
F	1306*	Université
F	1306*	Université
F	1306*	Université
M	1306*	Université
M	1306*	Université
M	1306*	Université

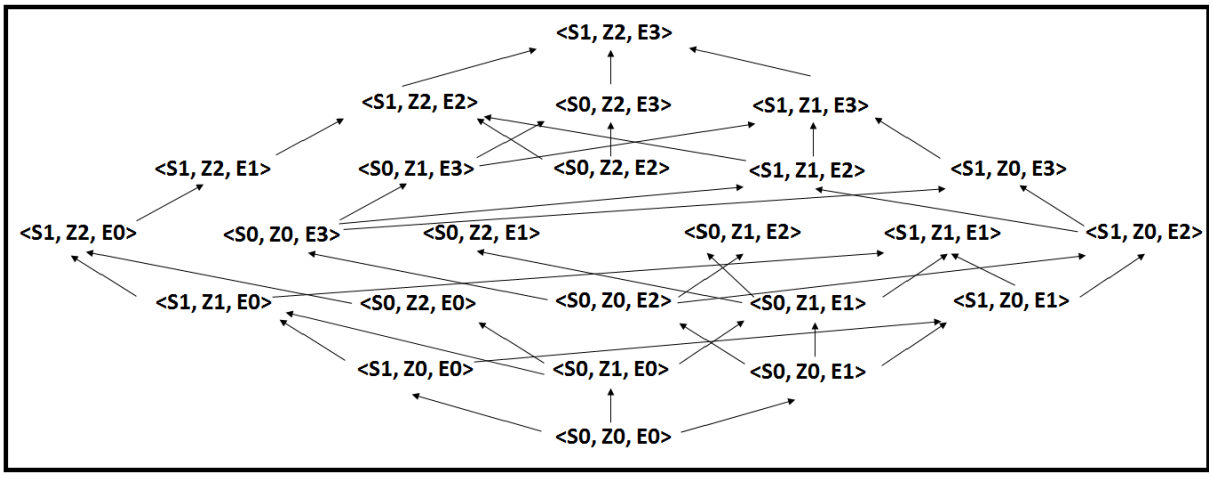
3. L'algorithme Samarati

Samarati est composé de deux grandes parties : la première partie construit le treillis qui correspond à la table originale et initialise h = h/2. Ensuite, la deuxième partie est composée par plusieurs itérations dont chacune fonctionne comme suit:

1. Sélectionne les nœuds au niveau h/2.
2. S'il existe au moins un nœud qui satisfait le k-anonymat alors Samarati stocke tous les nœuds au niveau h/2 et il se concentre sur la moitié inférieure du treillis , si non, il cible la moitié supérieure du treillis.
3. Il calcule la valeur de h=h/2. Si h égale à 0 alors l'algorithme s'arrête et les nœuds stockés sont les solutions proposées au data publisher. si non, l'algorithme passe à une autre itération.

Partie I

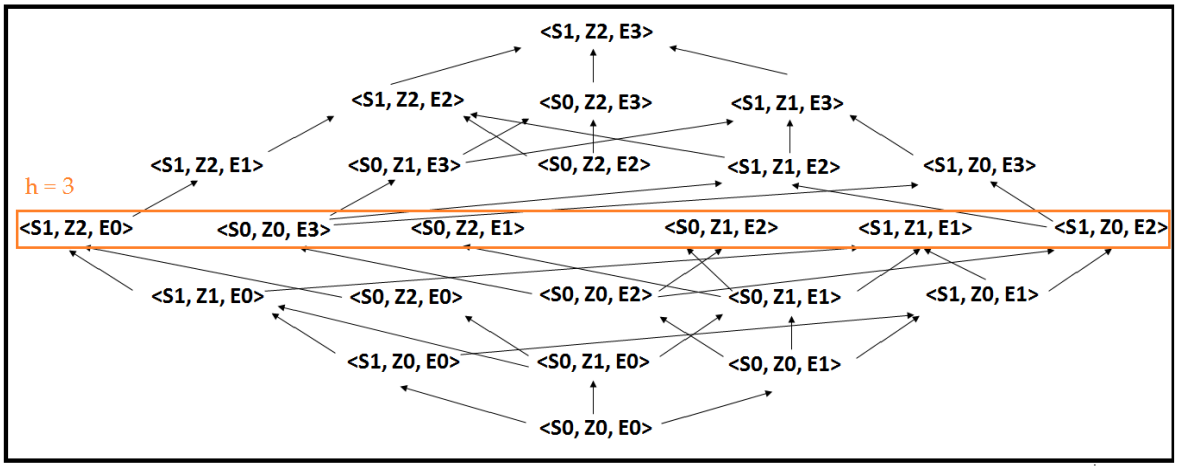
Samarati construit le treillis correspond à la table originale qui est composée de trois attributs QI. On suppose que SupMax=3 et k=2. La hauteur du treillis est h=6, alors h= h/2 =3.



Partie II

Itération 1

Etape 1: les nœuds qui se trouvent au niveau h=3 sont <S1,Z2,E0>, <S0,Z0,E3>, <S0,Z2,E1>, <S0,Z1,E2>, <S1,Z1,E1> et <S1,Z0,E2>



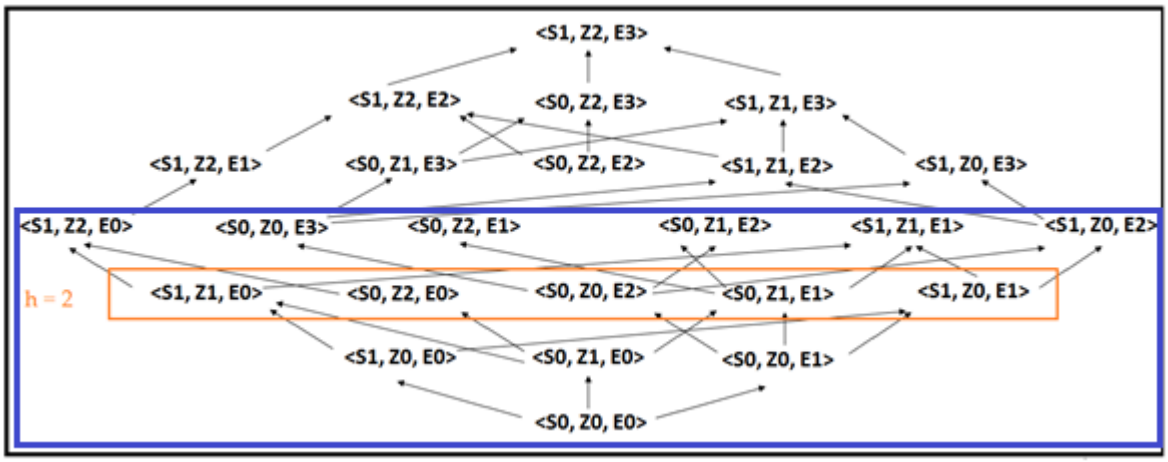
Etape 2 : un seul enregistrement ne satisfait pas le k-anonymat Dans le nœud <S1,Z0,E2> qui est représenté par la table ci-dessous. 1 est inférieur à maxSup, alors <S1,Z0,E2> satisfait le k-anonymat. Samarati cible la moitié inférieure du treillis.

Sexe	Code Postal	Niveau d'étude
tout-sexe	13050	Secondaire
tout-sexe	13051	Secondaire
tout-sexe	13050	Secondaire
tout-sexe	13050	Secondaire
tout-sexe	13051	Université
tout-sexe	13050	Université
tout-sexe	13061	Université
tout-sexe	13061	Université
tout-sexe	13060	Université
tout-sexe	13061	Université
tout-sexe	13060	Université
tout-sexe	13061	Université

Etape 3 : h est égale à trois qui est différent de 0, alors l'algorithme passe à une autre itération.

Itération 2

Etape 1: le nouveau treillis ciblé est dans le cadre bleu. Sa hauteur est h = 3. Les nœuds à niveau h/2=1.5 soit 2. Les nœuds au niveau h/2 sont <S1,Z1,E0>, <S0,Z2,E0>, <S0,Z0,E2>, <S0,Z1,E1> et <S1,Z0,E1>.

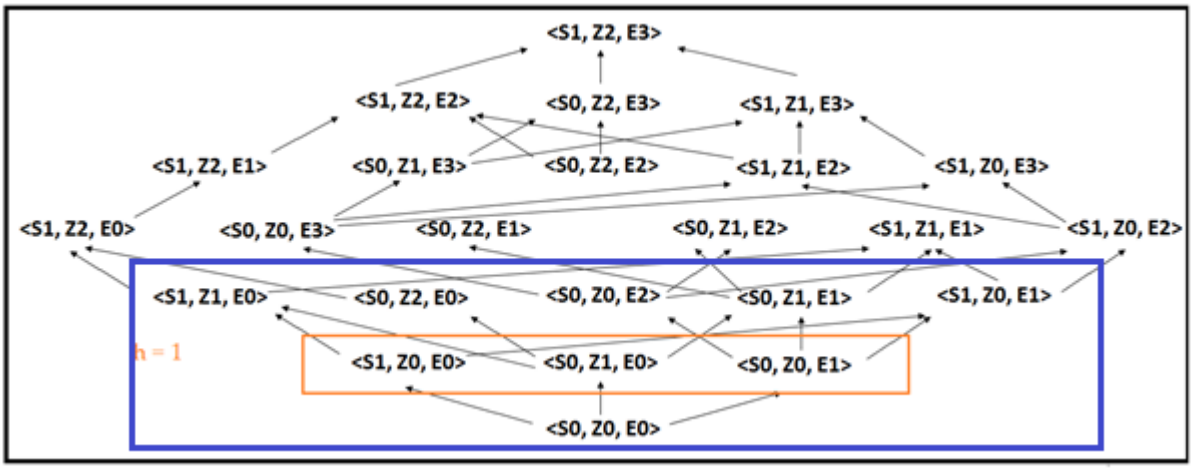


Etape 2 : le noeud <S0.Z1. E1> satisfait le k-anonymat. Samarati cible la moitié inférieure du treillis avec une hauteur de h = 2.

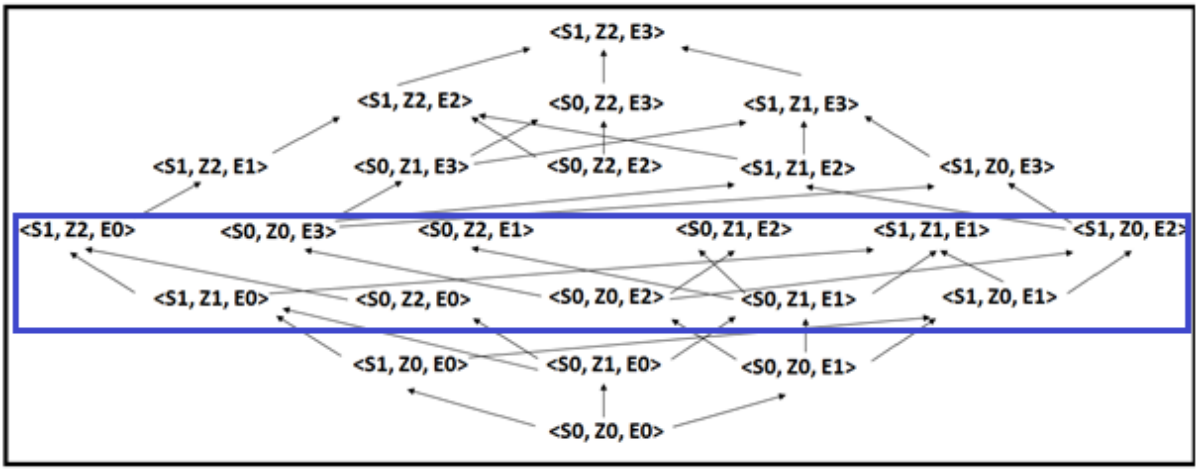
Etape 3 : h est égale à deux qui est différent de 0, alors l'algorithme passe à une autre itération.

Itération 3

Etape 1: le nouveau treillis ciblé est dans le cadre bleu. Sa hauteur est h = 2. Les nœuds à niveau h/2=1. Les nœuds au niveau h/2 sont <S1,Z0,E0>, <S0,Z1,E0> et <S0,Z0,E1>.



Etape 2 : Aucun nœud ne satisfait le k-anonymat. Samarati cible la moitié supérieure du treillis. Avec une hauteur de 1. h/2 = 0.5



Etape 2 : h/2 = 0. Alors Samarati s'arrête et les solutions sont les nœuds qui satisfont le k-anonymat stockés dans la deuxième itération.

L'algorithme Incognito

Incognito se base sur trois propriétés :

- La propriété de généralisation : Soit T une table, et soit P et Q des ensembles d'attributs de T tel que l'ensemble des attributs de P est plus général que Q. Si T est k-anonyme par rapport à P, T est également k-anonyme par rapport à Q. Par exemple, si <S0,Z1,E2> satisfait le k-anonymat et <S1,Z1,E2>, <S0,Z2,E2> et <S0,Z1,E3> sont plus générales que <S0,Z1,E2>, alors, <S1,Z1,E2>, <S0,Z2,E2> et <S0,Z1,E3> satisfont aussi le k-anonymat.

- la propriété des sous-ensembles : si une table satisfait k-anonymat avec deux attributs <X,Y>, alors, le satisfait aussi pour chacun de ces deux attributs <X> et <Y>. De même, si elle ne satisfait pas le k-anonymat pour les deux attributs <X,Y>, alors elle ne peut pas le satisfaire avec <X,Y,Z>. Par exemple, si <Z1,E1> satisfait le k-anonymat, alors, <Z1> et <E1> satisfont le k-anonymat. Si <S0,Z0> ne satisfait pas le k-anonymat, alors <S0, Z0,E1> ne le satisfait pas.

- La propriété de Rollup: Soit T une table, et soit P et Q des ensembles d'attributs de T tel que l'ensemble des attributs de P est plus général que Q. Si nous avons F1, l'ensemble de fréquences de T par rapport à P, alors nous pouvons générer chaque comptage à f2, l'ensemble des fréquences de T par rapport à Q, en additionnant l'ensemble des chiffres en f1 à chaque ensemble de valeur de f2. Par exemple, dans <E0> on a 2 masters et 3 doctorats. Au niveau <E1>, au lieu de recalculer le nombre de grad, on fait la somme de 2 et 3. Ainsi le nombre de grad est 2+3=5.

Incognito commence dans une première partie par l'initialisation de l'entier i qui représente le nombre d'attribut QI. Ensuite, dans chaque itération, i fonctionne comme suit :

1. Construit le treillis avec i attribut en faisant la jointure des treillis de l'itération précédente (excepté pour l'itération 1 dont les treillis qui sont construits en utilisant les hiérarchies de généralisation).
2. Supprimer les nœuds qui ne satisfont pas k-anonymat.
3. Vérifier si i égale au nombre total des attributs QI. Si oui, l'algorithme s'arrête si non, il passe à une autre itération.

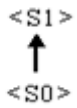
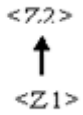
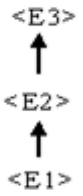
Dans ce qui suit, nous appliquons Incognito sur la table originale T.

Itération 1

Etape 1: construction des treillis avec des attributs singuliers (i=1)

Etape 2: Après la vérification des nœuds composés par un seul attribut, nous constatons que les nœuds <Z0> et <E0> ne satisfont pas le k-anonymat. Ainsi, ces nœuds sont supprimés des treillis.

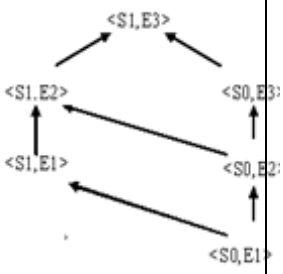
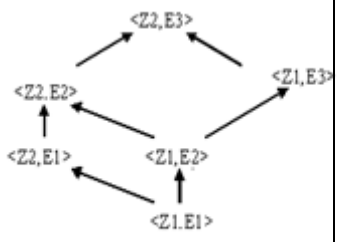
	<sexe>	**<Code postal>**	**<Niveau d'étude>**
Etape 1 : Construction des treillis	<S1> ↑ <S0>	<Z2> ↑ <Z1> ↑ <Z0>	<E3> ↑ <E2> ↑ <E1> ↑ <E0>

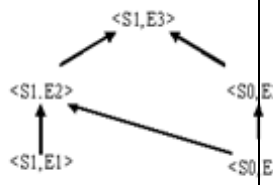
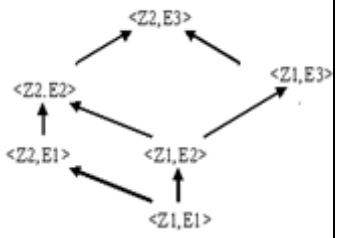
Etape 2 : Suppression des nœuds qui ne satisfont pas le k-anonymat			

Itération 2

Etape 1: constructions des treillis avec deux attributs (i=2).

Etape 2: le nœud <S0, E1> ne satisfait pas le k-anonymat, alors, il sera supprimé du treillis.

	<sexe, niveau d'étude>	**<sexe, code postal>**	**<code postal, niveau d'étude>**
Etape 1 : Construction des treillis			

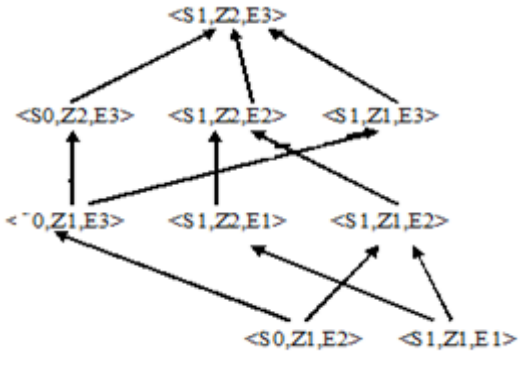
Etape 2 : Suppression des nœuds qui ne satisfont pas le k-anonymat			

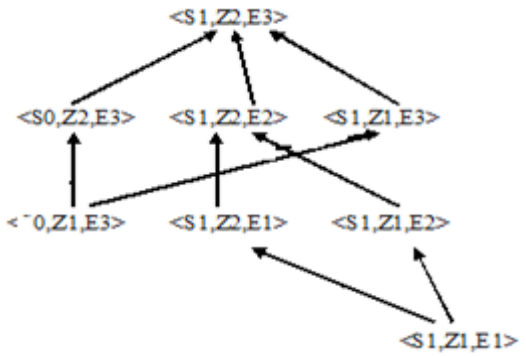
Itération 3

Etape 1: constructions des treillis avec trois attributs (i=3). Le treillis est construit selon les trois treillis précédents en se basant sur la propriété des sous-ensembles.

Etape 2: Seul le nœud <S0,Z1,E2> ne satisfait pas le k-anonymat. Pour cela, ce nœud est supprimé dans la deuxième étape. Le treillis de cette étape représente toutes les solutions possibles de la table T.

Finalement, les nœuds <S1,Z1,E1>, <S1,Z2,E1>, <S1,Z1,E2>, <S0,Z1,E3>, <S0,Z2,E3>, <S1,Z2,E2>, <S1,Z2,E3> et <S1,Z2,E3> sont les solutions proposées.

	<sexe, niveau d'étude, niveau d'étude>
Etape 1 : Construction des treillis	

Etape 2 : Suppression des nœuds qui ne satisfont pas le k-anonymat	

L'algorithme Généralisation ascendante (« Bottom up generalization »)

A chaque itération, la généralisation ascendante fonctionne comme suit :

1. Sélectionner les généralisations candidates et les généralisations critiques.
2. S'il existe des généralisations critiques, calculer le score de chaque. Si non calculer le score de chaque généralisation candidate.
3. Choisir la meilleure généralisation selon son score et l'effectuer.
4. Vérifier si la table satisfait le k-anonymat : si oui , l'algorithme s'arrête, si non, il passe à l'itération suivante.

Afin de faciliter l'application de l'algorithme de généralisation la généralisation ascendante, nous proposons de doubler la table originale. En effet, la colonne classe contient la fréquence de la classe des enregistrements. La classe **Y** pour ceux qui ont un salaire < 2000 et la classe **N** pour ceux ayant un salaire>= 2000. Par exemple, 2Y1N signifie qu'il existe deux enregistrements appartenant à la classe Y et un seul appartient à la classe N.

Sexe	Code Postal	Niveau d'éducation	Classe
M	13050	5ième	2Y0N
F	13051	3ième	3Y0N
M	13050	Seconde	2Y0N
F	13051	Seconde	3Y0N
F	13050	1ier et 2ième cycle	4Y0N
F	13061	1ier et 2ième cycle	1Y0N
F	13061	1ier et 2ième cycle	0Y1N
F	13060	Master	0Y4N
M	13061	Master	0Y1N
M	13060	Doctorat	0Y2N

Itération 1:

Etape 1: Les généralisations candidates sont: "{F,M}→tout-sexe", "{13050,13051}→1305*", "{13060,13061}→ 1306*", "{3ième, 5ième }→ collège", "{seconde,terminal}→lycée", "{master,doctorat}→ 3ième cycle"

Etape 2:

Calcul du score de la généralisation « {F,M}→any-sexe »

Sexe	classe	occurrence
F	11Y5N	16
M	4Y3N	7

I(F)= - (11/16 $\log_2$ 11/16) - (5/16 $\log_2$5/16) = 0,25 + 0,36 = 0,61

I(M)= - (4/7 $\log_2$ 4/7) - (3/7 $\log_2$3/7) = 0,55 + 0,36 = 0,91

I(tout-sex)= - (15/23 $\log_2$ 15/23) - (8/23 $\log_2$8/23) = 0,27 + 0,36 = 0,63

InfoLoss (tout-sex) = I(tout-sex) – ((16/23 * I (F)) + (7/23 * I (M)))= 0,63 – (0,7*0,61 + 0,3*0,91) = 0,46

Calcul du score de la généralisation « {13050,13051}→1305*»

Code postal	classe	occurrence
13050	8Y0N	8
13051	6Y0N	6
1305*	14Y0N	14

I (13050) = 0

I (13051) = 0

I (1305*) = 0

InfoLoss (1305*) = 0

Calcul du score de la généralisation « {13060,13061}→ 1306*»

Code postal	classe	occurrence
13060	0Y5N	5
13061	1Y2N	3
1306*	1Y7N	8

I (13060) = - 0/5 * log2 (0/5) – 5/5 * log2 (5/5)= 0

I (13061) = - 1/3 * log2 (1/3) - 2/3 * log2 (2/3)= 0.36 + 0,26 = 0,62

I (1306*) = - 1/8 * log2 (1/8) - 7/8 * log2 (7/8) = 0,25 + 0.11 = 0,36

InfoLoss (1306*) = I (1306*) – ((5/8 * I (13060))+ (3/8 * I (13061)) = 0,36 – (0,37 * 0,62) = 0,13

Calcul du score de la généralisation «{3ième, 5ième }→ collège»

Niveau d'étude	classe	occurrence
3ième	2Y0N	2
5ième	3Y0N	3

collège	5Y0N	5

I ($3^{ième}$) = 0

I ($5^{ième}$) = 0

I (collège) = 0

InfoLoss (collège) = 0

Calcul du score de la généralisation «{seconde, terminal}→lycée»

Sexe	classe	occurrence
seconde	2Y0N	2
lycée	2Y0N	2

I (seconde) = 0

I (lycée) = 0

InfoLoss (lycée) = 0

Calcul du score de la généralisation «{master, doctorat}→ $3^{ième}$ cycle»

Sexe	classe	occurrence
master	0Y5N	5
doctorate	0Y3N	3
grad school	0Y8N	8

I (master) = 0

I (doctorat) = 0

I ($3^{ième}$ cycle) = 0

InfoLoss ($3^{ième}$ cycle) = 0

Etape 3 : Les généralisations ayant le minimum de score sont: "{13050,13051}→1305*", "{$3^{ième}$, $5^{ième}$ }→ collège", "{seconde,terminal}→lycée", "{master,doctorat}→ $3^{ième}$ cycle". Nous choisissons aléatoirement la généralisation "{13050,13051}→1305*".

Sexe	Code Postal	Niveau d'éducation	Classe
M	1305*	5ième	2Y0N
F	1305*	3ième	3Y0N
M	1305*	Seconde	2Y0N
F	1305*	Seconde	3Y0N
F	1305*	1ier et 2ième cycle	4Y0N
F	13061	1ier et 2ième cycle	1Y0N

F	13061	1ier et 2ième cycle	0Y1N
F	13060	Master	0Y4N
M	13061	Master	0Y1N
M	13060	Doctorat	0Y2N

Etape 4 : Cette table ne satisfait pas le k- anonymat. L'algorithme passe à l'itération suivante.

Itération 2

Etape 1: Les généralisations candidates sont: "{F,M}→tout-sexe", "{13060,13061}→ 1306*", "{$3^{ième}$, $5^{ième}$}→ collège", "{seconde,terminal}→lycée", "{master,doctorat}→ $3^{ième}$ cycle"

Etape 2:

- **Score de** "{F,M}→tout-sexe" = 0.63
- **Score de** "{13060,13061}→ 1306*" =0.13
- **Score de** "{$3^{ième}$, $5^{ième}$ }→ collège" = 0
- **Score de** "{seconde,terminal}→lycée" = 0
- **Score de** "{master,doctorat}→ $3^{ième}$ cycle" = 0

Etape 3 : Les généralisations ayant le minimum de score sont: "{$3^{ième}$, $5^{ième}$}→ collège", "{seconde,terminal}→lycée", "{master,doctorat}→ $3^{ième}$ cycle". Nous choisissons aléatoirement la généralisation "{$3^{ième}$, $5^{ième}$ }→ collège".

Sexe	Code Postal	Niveau d'éducation	Classe
M	1305*	collège	2Y0N
F	1305*	collège	3Y0N
M	1305*	Seconde	2Y0N
F	1305*	Seconde	3Y0N
F	1305*	1ier et 2ième cycle	4Y0N
F	13061	1ier et 2ième cycle	1Y0N
F	13061	1ier et 2ième cycle	0Y1N
F	13060	Master	0Y4N
M	13061	Master	0Y1N
M	13060	Doctorat	0Y2N

Etape 4 : Cette table ne satisfait pas le k- anonymat. L'algorithme passe à l'itération suivante.

Itération 3

Etape 1: Les généralisations candidates sont: "{F,M}→tout-sexe", "{13060,13061}→ 1306*", "{seconde,terminal}→lycée", "{master,doctorat}→ $3^{ième}$ cycle"

Etape 2:

- **Score de** "{F,M}→tout-sexe" = 0.63
- **Score de** "{13060,13061}→ 1306*" =0.13

- **Score de** "{seconde,terminal}→lycée" = 0
- **Score de** "{master,doctorat}→ 3ième cycle" = 0

Etape 3: Les généralisations ayant le minimum de score sont: "{seconde,terminal}→lycée", "{master,doctorat}→ 3ième cycle". Nous choisissons aléatoirement la généralisation "{master,doctorat}→ 3ième cycle".

Sexe	Code Postal	Niveau d'éducation	Classe
M	1305*	collège	2Y0N
F	1305*	collège	3Y0N
M	1305*	Seconde	2Y0N
F	1305*	Seconde	3Y0N
F	1305*	1ier et 2ième cycle	4Y0N
F	13061	1ier et 2ième cycle	1Y0N
F	13061	1ier et 2ième cycle	0Y1N
F	13060	3ième cycle	0Y4N
M	13061	3ième cycle	0Y1N
M	13060	3ième cycle	0Y2N

Etape 4 : Cette table ne satisfait pas le k- anonymat. L'algorithme passe à l'itération suivante.

Itération 4

Etape 1: Les généralisations candidates sont: "{F,M}→tout-sexe", "{13060,13061}→ 1306*", "{seconde,terminal}→lycée".

Etape 2:

- **Score de** "{F,M}→tout-sexe" = 0.63
- **Score de** "{13060,13061}→ 1306*" =0.13
- **Score de** "{seconde,terminal}→lycée" = 0

Etape 3 : La généralisation ayant le minimum de score est "{seconde,terminal}→lycée"

Sexe	Code Postal	Niveau d'éducation	Classe
M	1305*	collège	2Y0N
F	1305*	collège	3Y0N
M	1305*	lycée	2Y0N
F	1305*	lycée	3Y0N
F	1305*	1ier et 2ième cycle	4Y0N
F	13061	1ier et 2ième cycle	1Y0N
F	13061	1ier et 2ième cycle	0Y1N
F	13060	3ième cycle	0Y4N
M	13061	3ième cycle	0Y1N

M	13060	3ième cycle	0Y2N

Etape 4 : Cette table ne satisfait pas le k- anonymat. L'algorithme passe à l'itération suivante.

Itération 5

Etape 1: Les généralisations candidates sont: "{F,M}→tout-sexe", "{13060,13061}→ 1306*", "{collège, lycée}→ secondaire", "{1ier et 2ième cycle, 3ième cycle }→ université".

Etape 2:

- **Score de** "{F,M}→tout-sexe" = 0.63
- **Score de** "{13060,13061}→ 1306*" =0.13
- **Score de** "{collège, lycée}→ secondaire"= 0
- **Score de** "{1ier et 2ième cycle, 3ième cycle }→ université" = 0,68

Etape 3 : La généralisation ayant le minimum de score est "{collège, lycée}→ secondaire"

Sexe	Code Postal	Niveau d'éducation	Classe
M	1305*	secondaire	2Y0N
F	1305*	secondaire	3Y0N
M	1305*	secondaire	2Y0N
F	1305*	secondaire	3Y0N
F	1305*	1ier et 2ième cycle	4Y0N
F	13061	1ier et 2ième cycle	1Y0N
F	13061	1ier et 2ième cycle	0Y1N
F	13060	3ième cycle	0Y4N
M	13061	3ième cycle	0Y1N
M	13060	3ième cycle	0Y2N

Etape 4 : Cette table ne satisfait pas le k- anonymat. L'algorithme passe à l'itération suivante.

Itération 6

Etape 1: Les généralisations candidates sont: "{F,M}→tout-sexe", "{13060,13061}→ 1306*", "{1ier et 2ième cycle, 3ième cycle }→ université".

Etape 2:

- **Score de** "{F,M}→tout-sexe" = 0.63
- **Score de** "{13060,13061}→ 1306*" =0.13
- **Score de** "{1ier et 2ième cycle, 3ième cycle }→ université" = 0,68

Etape 3 : La généralisation ayant le minimum de score est "{13060,13061}→ 1306*"

Sexe	Code Postal	Niveau d'éducation	Classe
M	1305*	secondaire	2Y0N
F	1305*	secondaire	3Y0N
M	1305*	secondaire	2Y0N
F	1305*	secondaire	3Y0N
F	1305*	1ier et 2ième cycle	4Y0N
F	1306*	1ier et 2ième cycle	1Y0N
F	1306*	1ier et 2ième cycle	0Y1N
F	1306*	3ième cycle	0Y4N
M	1306*	3ième cycle	0Y1N
M	1306*	3ième cycle	0Y2N

Etape 4 : Cette table ne satisfait pas le k- anonymat. L'algorithme passe à l'itération suivante.

Itération 7

Etape 1: Les généralisations candidates sont: "{F,M}→tout-sexe", {1305*,1306*}→130** et "{1^{ier} et 2^{ième} cycle, 3^{ième} cycle }→ université".

Etape 2:

- **La seule généralisation critique est** {1305*,1306*}→130**.

Etape 3 : La généralisation ayant le minimum de score est "{1305*,1306*}→130**"

Sexe	Code Postal	Niveau d'éducation	Classe
M	130**	secondaire	2Y0N
F	130**	secondaire	3Y0N
M	130**	secondaire	2Y0N
F	130**	secondaire	3Y0N
F	130**	1ier et 2ième cycle	4Y0N
F	130**	1ier et 2ième cycle	1Y0N
F	130**	1ier et 2ième cycle	0Y1N
F	130**	3ième cycle	0Y4N
M	130**	3ième cycle	0Y1N
M	130**	3ième cycle	0Y2N

Etape 4 : Cette table satisfait le k- anonymat. L'algorithme s'arrête.

L'algorithme Spécialisation descendante (« Top down specialization »)

La première partie de TDS permet d'effectuer une généralisation maximale de toutes les valeurs de la table originale. Ensuite, chaque itération se fonctionne comme suit :

1. Vérifier s'il existe des spécialisations valides et bénéfiques parmi les spécialisations candidates. Si oui, l'algorithme passe à l'étape suivante, si non, il s'arrête.

2. Calculer le score de chaque spécialisation.
3. Choisir la spécialisation ayant le meilleur score.

Sexe	Code Postal	Niveau d'étude	Classe
M	13050	3ième	2Y0N
F	13051	5ième	3Y0N
M	13050	Seconde	2Y0N
F	13051	1ier et 2ième cycle	3Y0N
F	13050	1ier et 2ième cycle	4Y0N
F	13061	1ier et 2ième cycle	1Y0N
F	13061	Master	0Y1N
F	13060	Master	0Y4N
M	13061	Doctorat	0Y1N
M	13060	Doctorat	0Y2N

Partie 1

Sexe	code postal	niveau d'éducation	classe	occurrence
tout-sex	130**	tout-education	15Y8N	23

Partie 2

<u>Itération1</u>

Etape 1: les spécialisations valides et bénéfiques sont : "tout-sexe→ {F,M}", "130**→ {1306*,1305*}", "tout-education → {université, secondaire}"

Etape 2 :

- **Calcul du score de la spécialisation** "tout-sexe→ {F,M}"

Si on applique "tout-sexe→ {F,M}", le résultat sera la table suivante:

Sexe	code postal	niveau d'éducation	classe	occurrence
F	130**	tout-éducation	11Y5N	16
M	130**	tout-éducation	4Y3N	7

$I(F) = -(11/16 \log_2 11/16) - (5/16 \log_2 5/16) = 0,25 + 0,36 = 0,61$

$I(M) = -(4/7 \log_2 4/7) - (3/7 \log_2 3/7) = 0,55 + 0,36 = 0,91$

$I(\text{tout-sex}) = -(15/23 \log_2 15/23) - (8/23 \log_2 8/23) = 0,27 + 0,36 = 0,63$

MCgain (tout-sex) = I(tout-sex) – ((16/23 * I (F)) + (7/23 * I (M)))= 0,63 – (0,7*0,61 + 0,3*0,91) = 0,46

AnonymityLoss (tout-sex) = 23 – 7 = 16

Score (tout-sexe→ {F,M}) =0,46/16 = 0,028

- **Calcul du score de la spécialisation** "130**→ {1305*,1306*}"

Si on applique la spécialisation "130**→ {1305*,1306*}", le résultat sera la table suivante:

Sexe	code postal	niveau d'étude	classe	occurrence
tout-sexe	1305*	tout-éducation	14Y0N	14
tout-sexe	1306*	tout-éducation	1Y8N	9

I(1305*)= - (14/14 $\log_2$ 14/14) = 0

I(1306*)= - (1/9 $\log_2$ 1/9) - (8/9$\log_2$8/9) = 0,24 + 0,1 = 0,34

I(130**)= - (15/23 $\log_2$ 15/23) - (8/23 $\log_2$8/23) = 0,27 + 0,36 = 0,63

MCgain (130**) = I(130**) – ((14/23 * I (1305*)) + (9/23 * I (1306*)))= 0,63 – (0,39*0,34) = 0,63- 0,13 = 0,49

AnonymityLoss (130**) = 23 – 10 = 13

Score (130**→ {1305*,1306*}) =0,49/13 = 0,03

- **Calcul du score de la spécialisation** "tout-éducation→ { Secondary, Université }"

Si on applique la spécialisation "tout-éducation→ { Secondary, Université }", le résultat sera la table suivante:

Sexe	code postal	niveau d'étude	classe	occurrence
tout-sexe	130**	Secondaire	7Y0N	7
tout-sexe	130**	Université	8Y8N	16

I(secondaire)= - (7/7 $\log_2$ 7/7) = 0

I(Université)= - (8/16 $\log_2$ 8/16) - (8/16 $\log_2$ 8/16) = 0,35 + 0,35 = 0,7

I(tout-education)= - (15/23 $\log_2$ 15/23) - (8/23 $\log_2$8/23) = 0,27 + 0,36 = 0,63

MCgain (tout-education)= I(tout-education) – ((7/23 * I (Secondaire)) + (16/23 * I (université))) = 0,63 -

AnonymityLoss(tout-education)= 23 – 8 = 15

Score (tout-éducation→ { Secondaire, Université }) =0,63/15 = 0,042

Etape 3 : la spécialisation qui a le meilleur score est tout-education → {secondaire, université}

Sexe	code postal	niveau d'étude	classe	occurrence
tout-sexe	130**	Secondaire	7Y0N	7
tout-sexe	130**	Université	8Y8N	16

Itération 2

Etape 1: les spécialisations valides et bénéfiques sont : "tout-sexe→ {F,M}", "130**→ {1306*,1305*}", université→ {1^{ier} et 2^{ième} cycle, 1^{ier} et 3^{ième} cycle }.

Etape 2 :

- Score (université→ {1ier et 2ième cycle, 1ier et 3ième cycle }) =0,7
- Score (130**→ {1306*,1305*}) =0,49
- Score (tout-sexe→ {F,M}) = 0,115

Etape 3 : la spécialisation qui a le meilleur score est université→ {1ier et 2ième cycle, 3ième cycle }

Sexe	code postal	niveau d'éducation	classe	occurrence
tout-sexe	130**	Secondaire	7Y0N	7
tout-sexe	130**	1ier et 2ième cycle	8Y0N	8
tout-sexe	130**	3ième cycle	0Y8N	8

Itération 3

Etape 1: la spécialisation valide et bénéfique est "tout-sexe→ {F,M}".

Etape 2:

- Score (tout-sexe→ {F,M}) = 0,115

Etape 3 : la spécialisation qui a le meilleur score est tout-sexe→ {F,M}

Sexe	code postal	niveau d'étude	classe	occurrence
F	130**	Secondaire	3Y0N	3
M	130**	Secondaire	4Y0N	4
F	130**	1ier et 2ième cycle	8Y0N	8
F	130**	3ième cycle	0Y5N	5
M	130**	3ième cycle	0Y3N	3

Itération 4

Etape 1: Il n'y a aucune spécialisation valide et bénéfique. La table proposée au data publisher est la suivante :

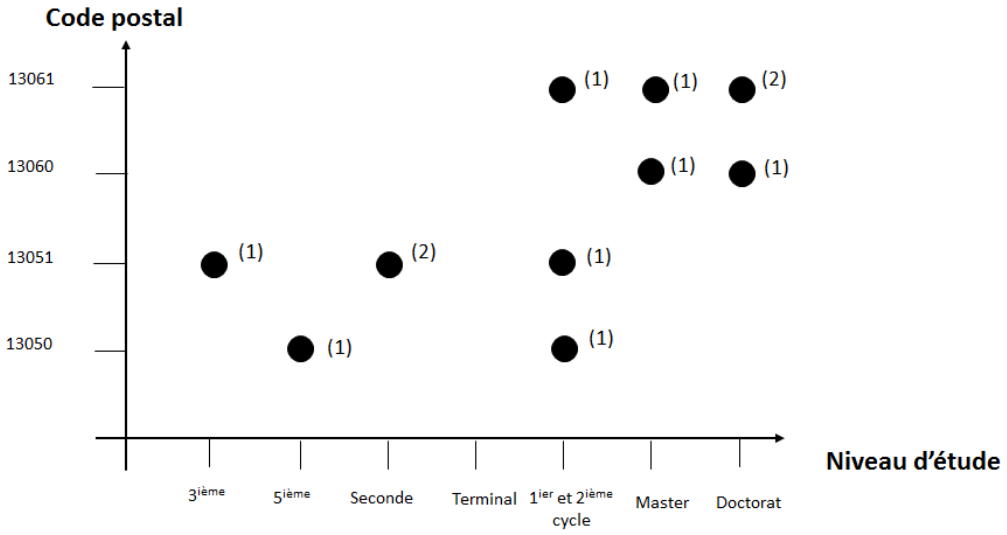
Sexe	code postal	niveau d'étude	classe	occurrence
F	130**	Secondaire	3Y0N	3
M	130**	Secondaire	4Y0N	4
F	130**	1ier et 2ième cycle	8Y0N	8
F	130**	3ième cycle	0Y5N	5
M	130**	3ième cycle	0Y3N	3

L'algorithme de « Median Mondrian »

A chaque itération

1. Choisit un groupe non marqué.
2. vérifie la possibilité de diviser un groupe en deux sous-groupes selon une dimension choisie (en divisant la zone selon la valeur médiane de cette dimension). Si la division n'est pas possible, alors, l'algorithme marque le groupe choisi, sinon, il applique la division.
3. Vérifie s'il existe encore des groupes non marqués. Si oui, l'algorithme passe l'étape 1 si non à l'étape 4.
4. Faire le recodage en remplaçant les différentes valeurs d'une même zone par la valeur de leur premier parent commun

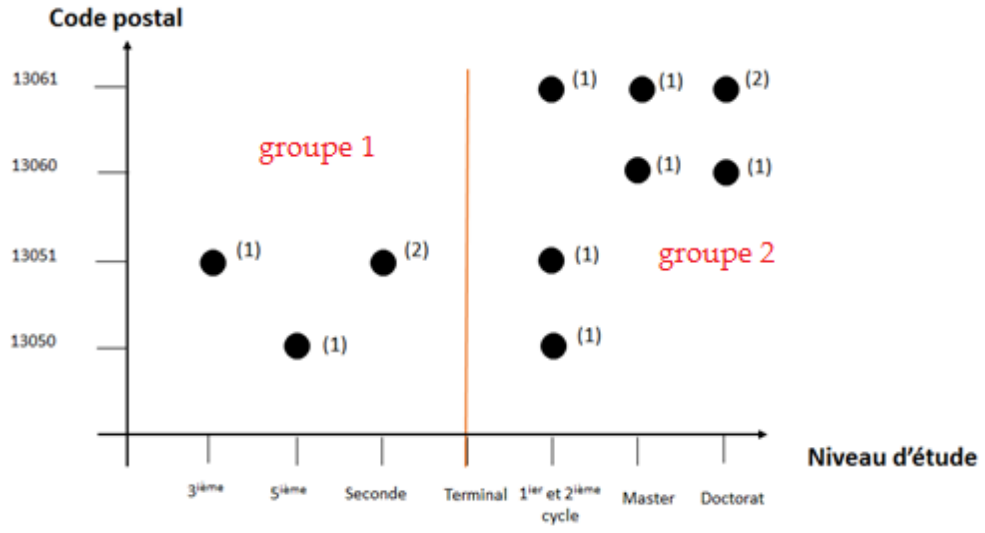
Nous avons choisi deux attributs QI (code postal et niveau d'étude) afin de simplifier les divisions. La table originale est représentée comme suit (les nombres entre parenthèses indique le nombre de duplicata des enregistrements :



Itération 1

Etape 1: l'ensemble de la table est choisi

Etape 2: le groupe peut être divisé en médian selon la dimension niveau d'étude. Groupe 1 et groupe 2 sont construit à partir de cette division.

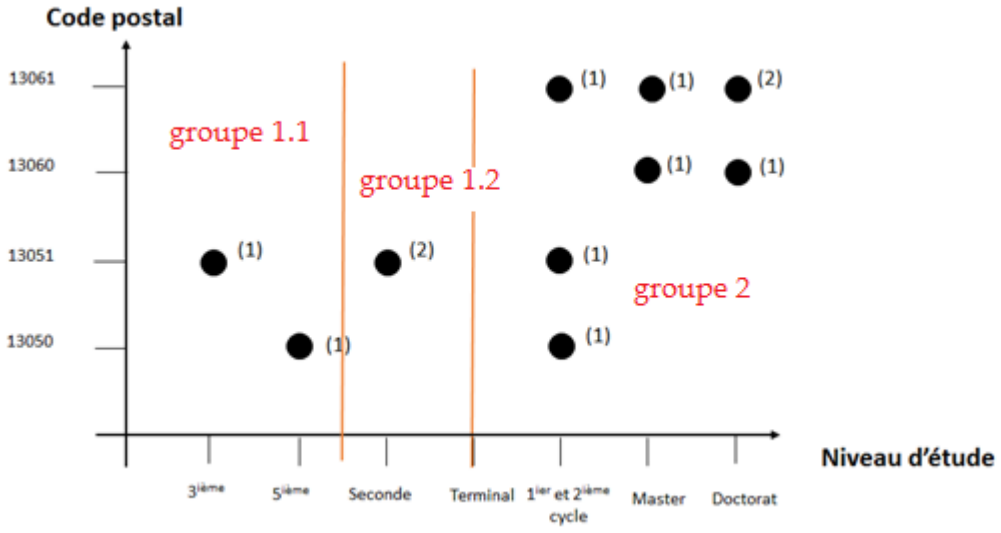


Etape 3 : Groupe 1 et groupe 2 ne sont pas marqués.

Itération 2

Etape 1: Groupe 1 est choisi

Etape 2: le groupe peut être divisé en médian selon la dimension niveau d'étude. Groupe 1.1 et groupe 1.2 sont construit à partir de cette division.

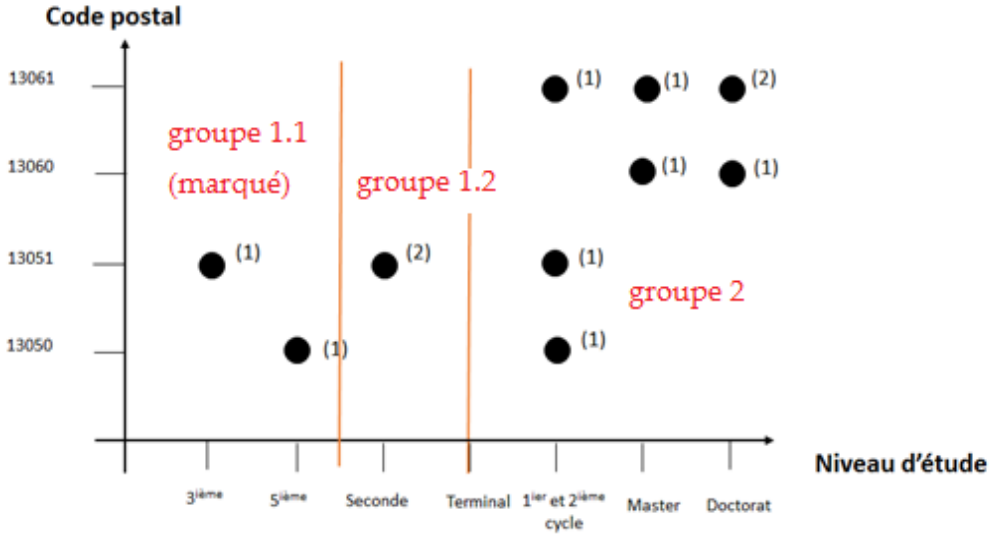


Etape 3 : Groupe 1.1, groupe 1.2 et groupe 2 ne sont pas marqués.

Itération 3

Etape 1: Groupe 1.1 est choisi

Etape 2: le groupe 1.1 ne peut pas être divisé en médian. L'algorithme le marque.

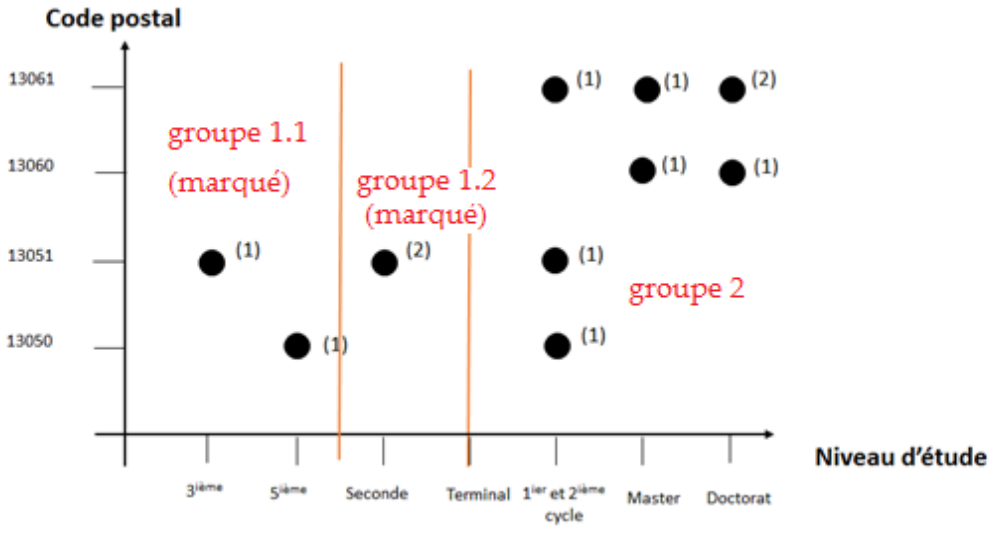


Etape 3 : Groupe 1.2 et groupe 2 ne sont pas marqués.

Itération 4

Etape 1: Groupe 1.2 est choisi

Etape 2: le groupe 1.2 ne peut pas être divisé en médian. L'algorithme le marque.

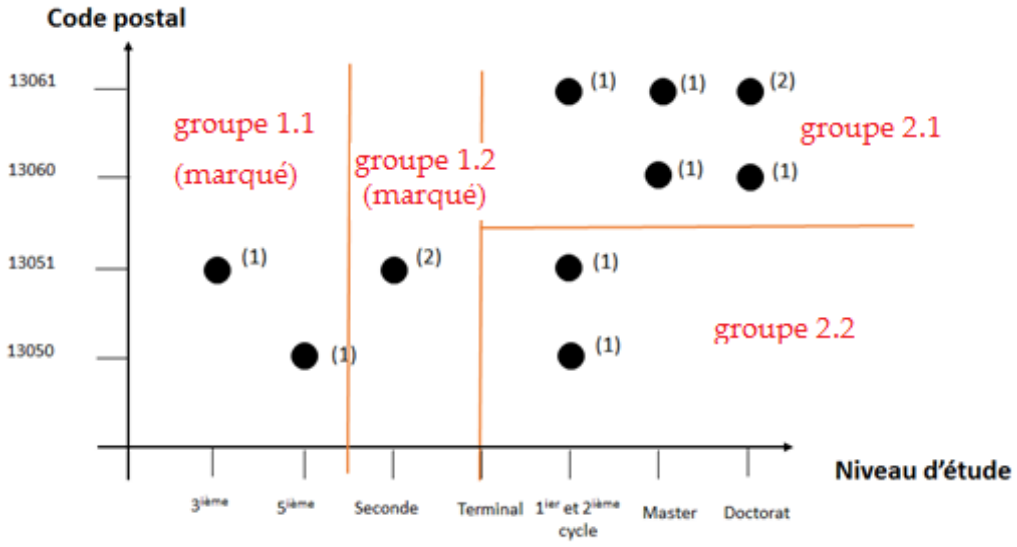


Etape 3 : groupe 2 n'est pas marqué.

Itération 4

Etape 1: Groupe 2 est choisi

Etape 2: le groupe 1.2 peut être divisé en médian selon la dimension code postal. Les groupe 2.1 et 2.2 sont construits à partir de cette division.

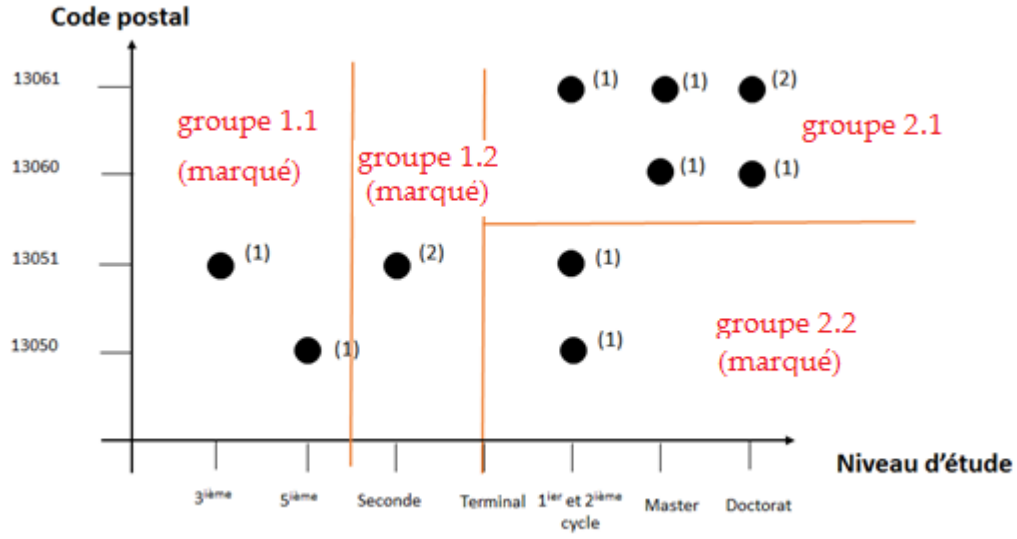


Etape 3 : Groupe 2.1 et groupe 2.2 ne sont pas marqués.

Itération 5

Etape 1: Groupe 2.2 est choisi

Etape 2: le groupe 2.2 ne peut pas être divisé en médian. L'algorithme le marque.

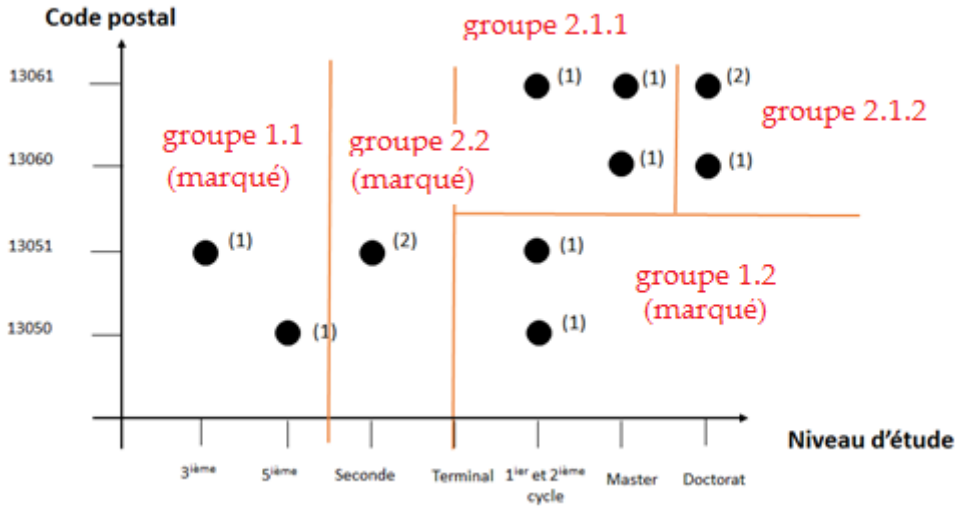


Etape 3 : Groupe 2.1 n'est pas marqué.

Itération 6

Etape 1: Groupe 2.1 est choisi

Etape 2: le groupe 2.2 peut être divisé en médian selon la dimension niveau d'étude. Les groupes 2.1.1 et 2.1.2 sont construits à partir de cette division.

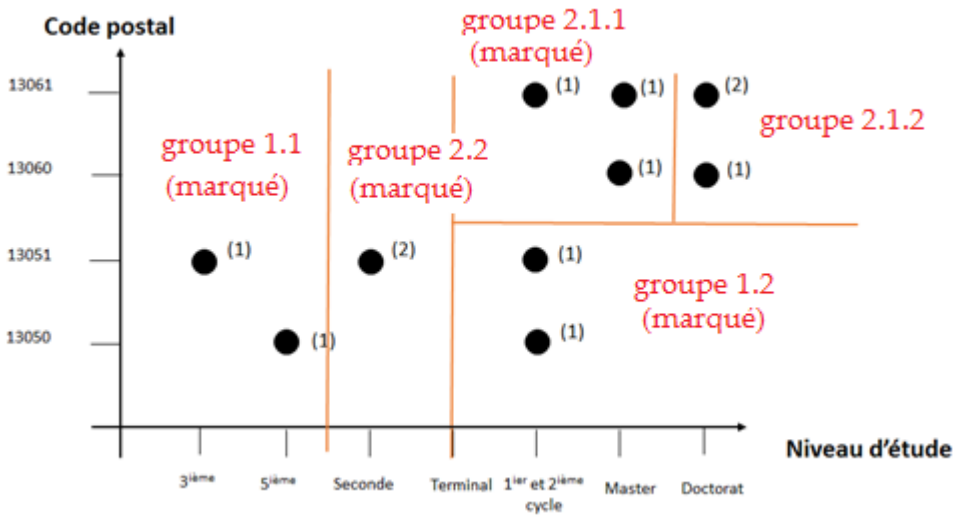


Etape 3 : Les groupes 2.1.1 et 2.1.2 ne sont pas marqués.

Itération 7

Etape 1: Groupe 2.1.1 est choisi

Etape 2: le groupe 2.1.1 ne peut pas être divisé en médian. L'algorithme le marque.

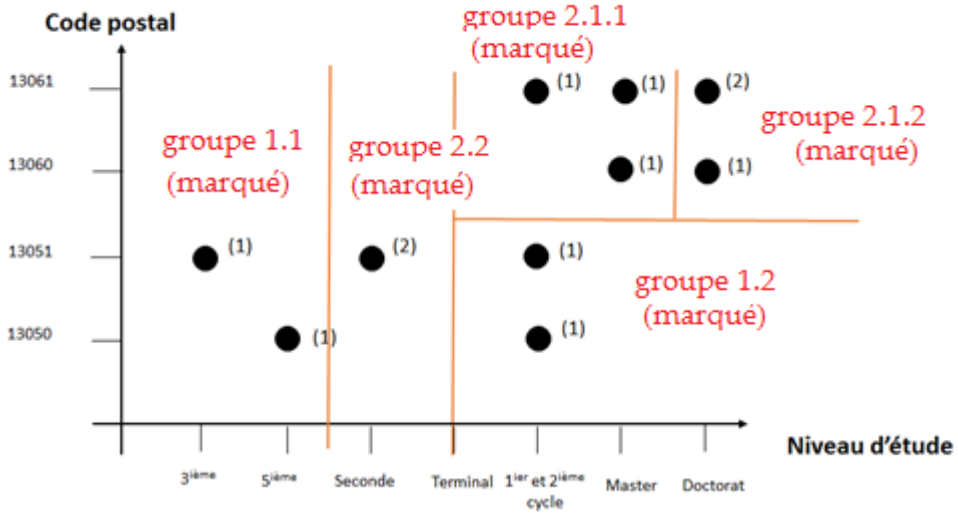


Etape 3 : Groupe 2.1.2 n'est pas marqué.

Itération 8

Etape 1: Groupe 2.1.2 est choisi

Etape 2: le groupe 2.1.2 ne peut pas être divisé en médian. L'algorithme le marque.



Etape 3 : Tous les groupes sont marqués.

Etape 4 : Le résultat du recodage est la table suivante

code postal	niveau d'étude	Salaire
1305*	collège	1200
1305*	collège	1300
13051	Seconde	1200
13051	Seconde	1300
1305*	1ier et 2ième cycle	1500
1305*	1ier et 2ième cycle	1500
1306*	université	1600
1306*	université	2000
1306*	université	2100
1306*	Doctorat	3000
1306*	Doctorat	4000
1306*	Doctorat	4500

L'algorithme InfoGain Mondrian

A chaque itération

1. Choisit un groupe non marqué.
2. vérifie la possibilité de diviser un groupe en deux sous-groupes selon une dimension choisie. Si la division n'est pas possible, alors, l'algorithme marque le groupe choisi, si non, il applique la division ayant le meilleur score selon la métrique d'entropie parmi les divisions candidates.

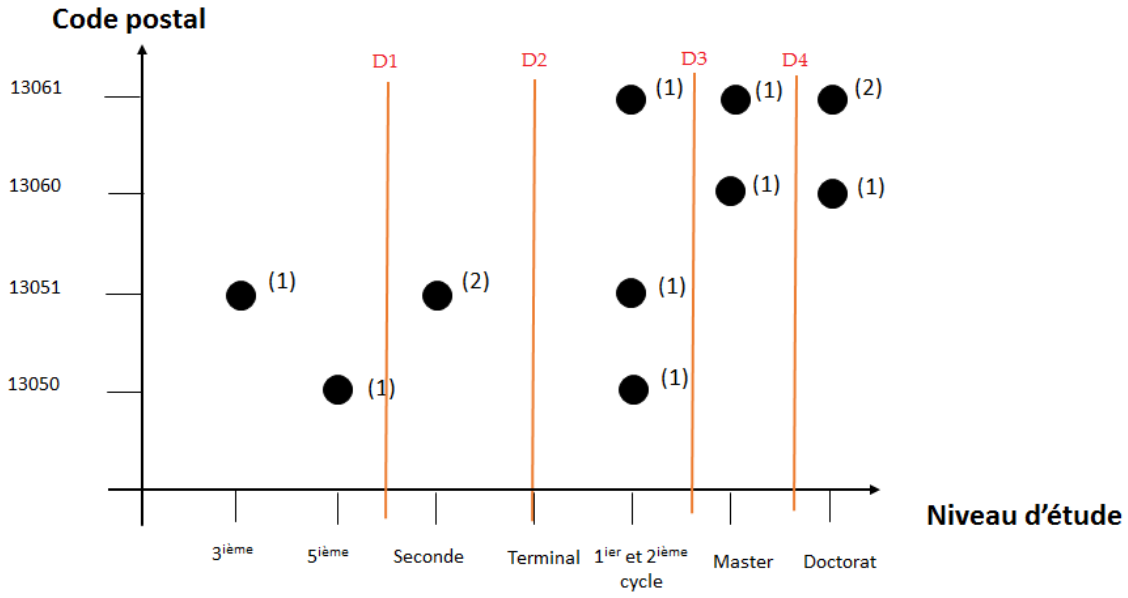
3. Vérifie s'il existe encore des groupes non marqués. Si oui, l'algorithme passe l'étape 1 si non à l'étape 4
4. Faire le recodage en remplaçant les différentes valeurs d'une même zone par la valeur de leur premier parent commun

Itération 1

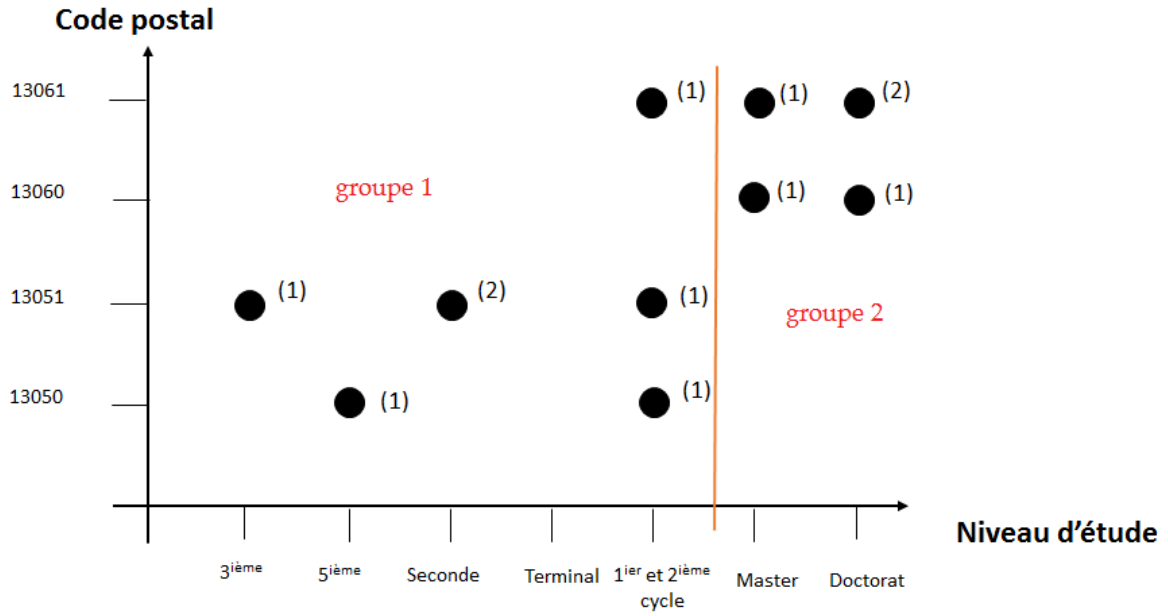
Etape 1: l'ensemble de la table est choisi.

Etape 2: selon la dimension niveau d'étude, il y a quatre divisions candidates (D1, D2, D3 et D4)

- **Score D1** = 2/12 (-2/2log2/2) + 10/12 (-5/10 log5/10 - 5/10 log5/10)= 0.25
- **Score D2** = 4/12 (-4/4log4/4) + 8/12 (-3/8 log3/8 - 5/8 log5/8)= 0.01
- **Score D3** = 7/12 (-7/7log7/7) + 5/12 (-5/5 log5/5)= 0
- **Score D4** = 3/12 (-3/3log3/3) + 9/12 (-2/9 log2/9- 7/9 log7/9)= 0.165



La meilleure division est d3. Groupe 1 et groupe 2 sont construits à partir de cette division

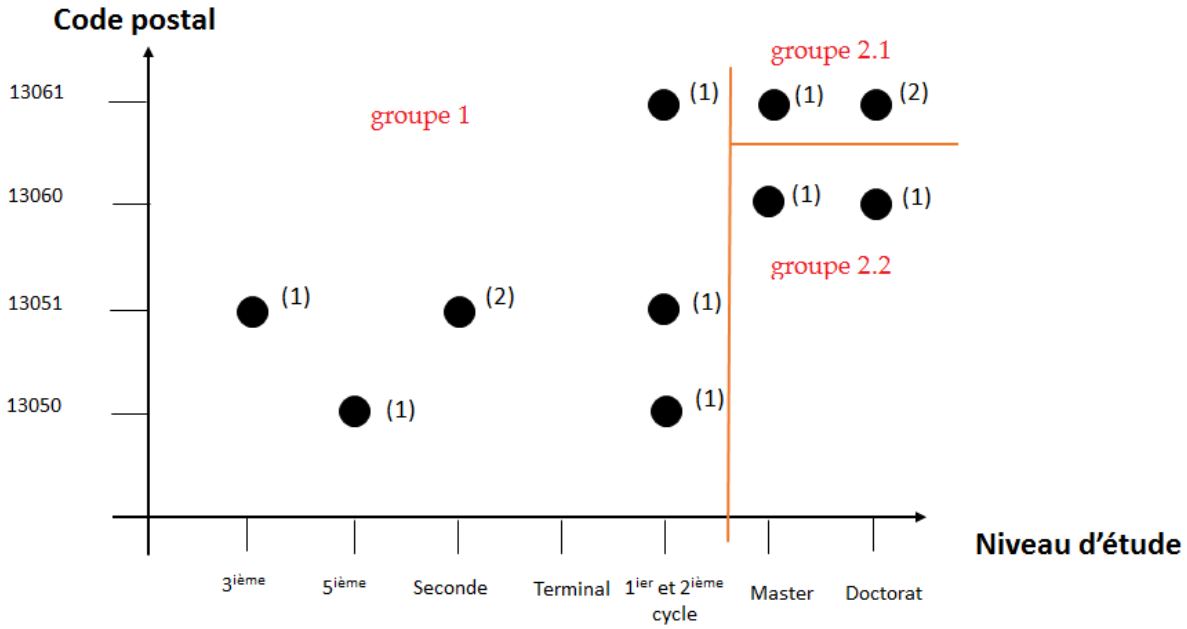


Etape 3: groupe 1 et 2 ne sont pas marqués

Itération 2

Etape 1: Le groupe 2 est choisi.

Etape 2 : selon la dimension code postal, une seule division candidate. Cette dernière est appliquée. Les groupes 2.1 et 2.2 sont construits à partir de cette division.

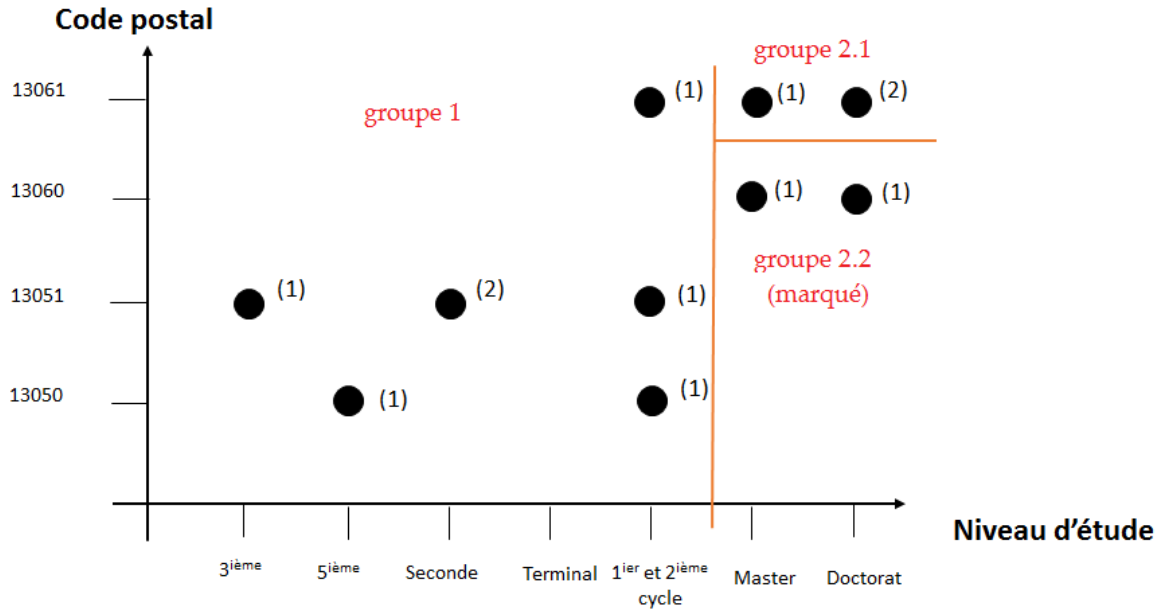


Etape 3: groupe 1, groupe 2.1 et groupe 2.2 ne sont pas marqués.

Itération 3

Etape 1: Le groupe 2.2 est choisi.

Etape 2 : aucune division n'est candidate. L'algorithme le marque.

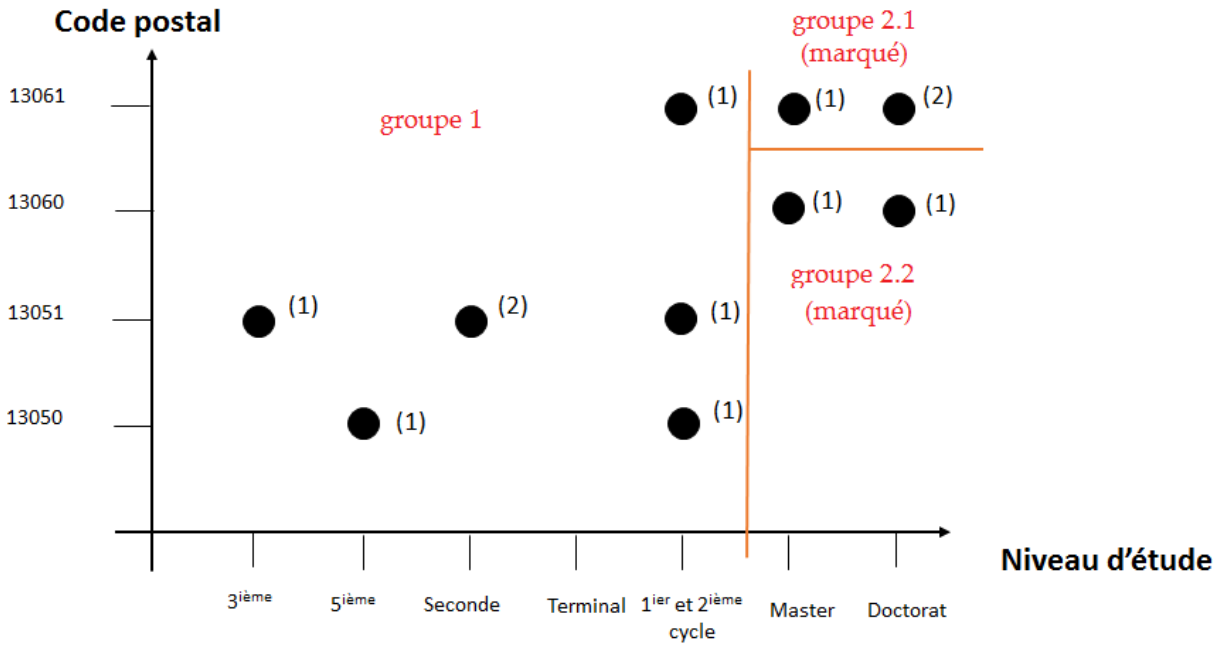


Etape 3: groupe 1 et groupe 2.1 sont pas marqués.

Itération 4

Etape 1: Le groupe 2.1 est choisi.

Etape 2 : aucune division n'est candidate. L'algorithme le marque.

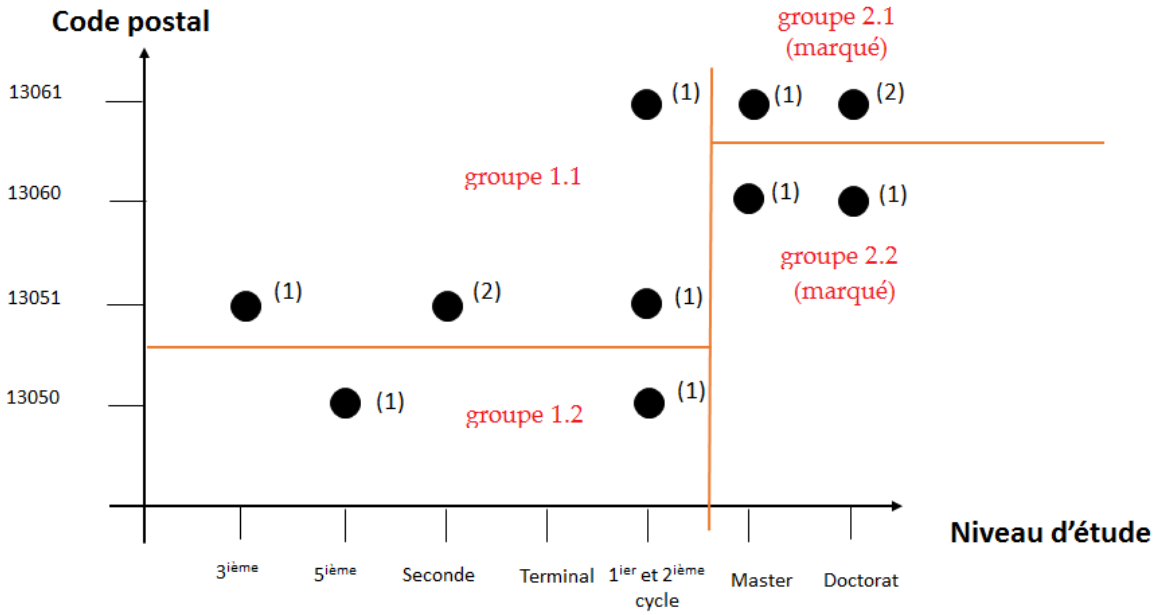


Etape 3: groupe 1 sont pas marqués.

Itération 5

Etape 1: Le groupe 1 est choisi.

Etape 2 : selon la dimension code postal, une seule division est candidate. Cette dernière est appliquée. Les groupes 1.1 et 1.2 sont construits à partir de cette division.

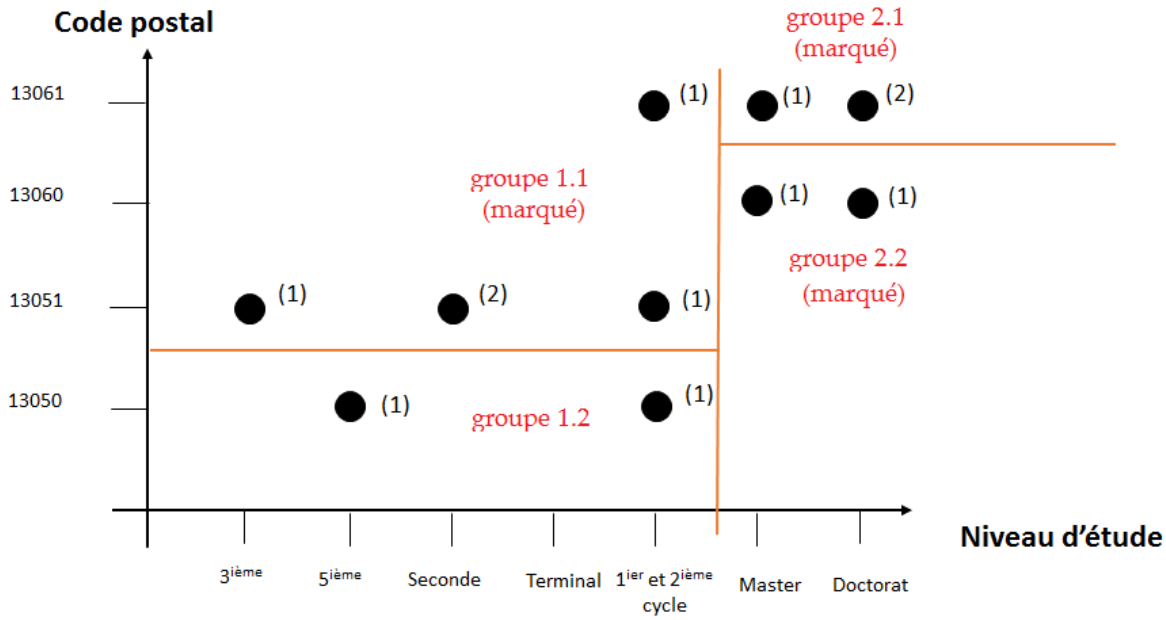


Etape 3: groupe 1.1 et groupe 1.2 ne sont pas marqués.

Itération 5

Etape 1: Le groupe 1.1 est choisi.

Etape 2 : aucune division n'est candidate. L'algorithme le marque.

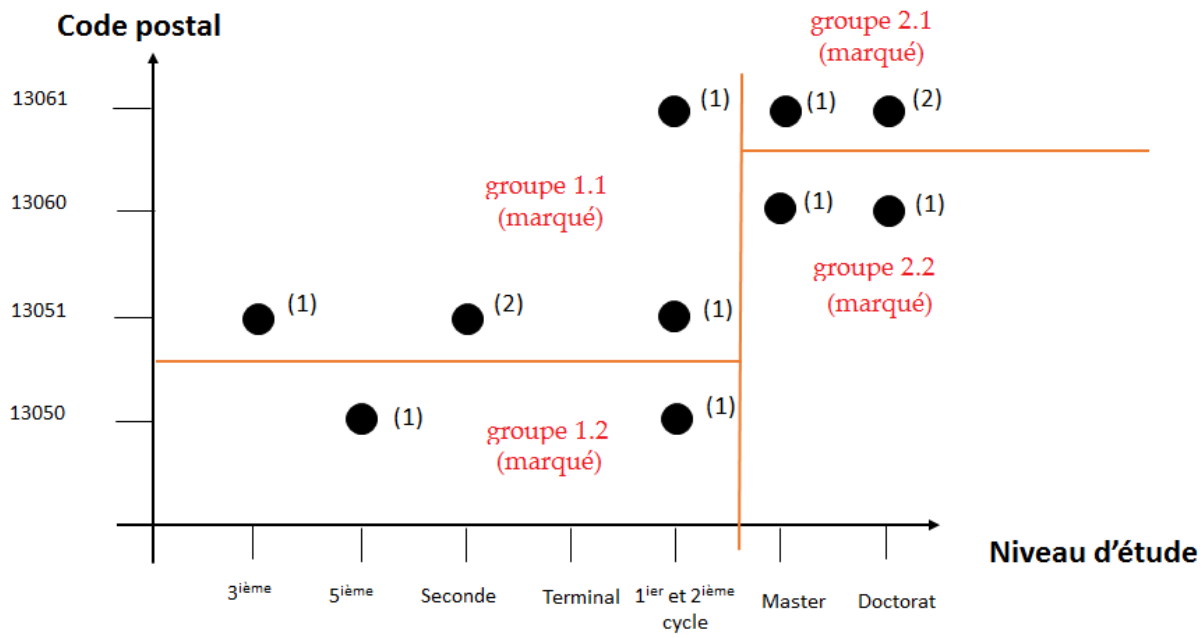


Etape 3: groupe 1.2 n'est pas marqué.

Itération 6

Etape 1: Le groupe 1.2 est choisi.

Etape 2 : aucune division n'est candidate. L'algorithme le marque.



Etape 3: tous les groupes sont marqués

Etape 4 : Le résultat du recodage est la table suivante **:**

code postal	niveau d'étude	Salaire
13051	Tout_niveau_d'étude	1200
13050	Tout_niveau_d'étude	1300
13051	Tout_niveau_d'étude	1200
13051	Tout_niveau_d'étude	1300
13051	Tout_niveau_d'étude	1500
13050	Tout_niveau_d'étude	1500
13061	Tout_niveau_d'étude	1600
13061	3ième cycle	2000
13060	3ième cycle	2100
13061	3ième cycle	3000
13060	3ième cycle	4000
13061	3ième cycle	4500

L'algorithme Least Squared Deviance (LSD) Mondrian

A chaque itération

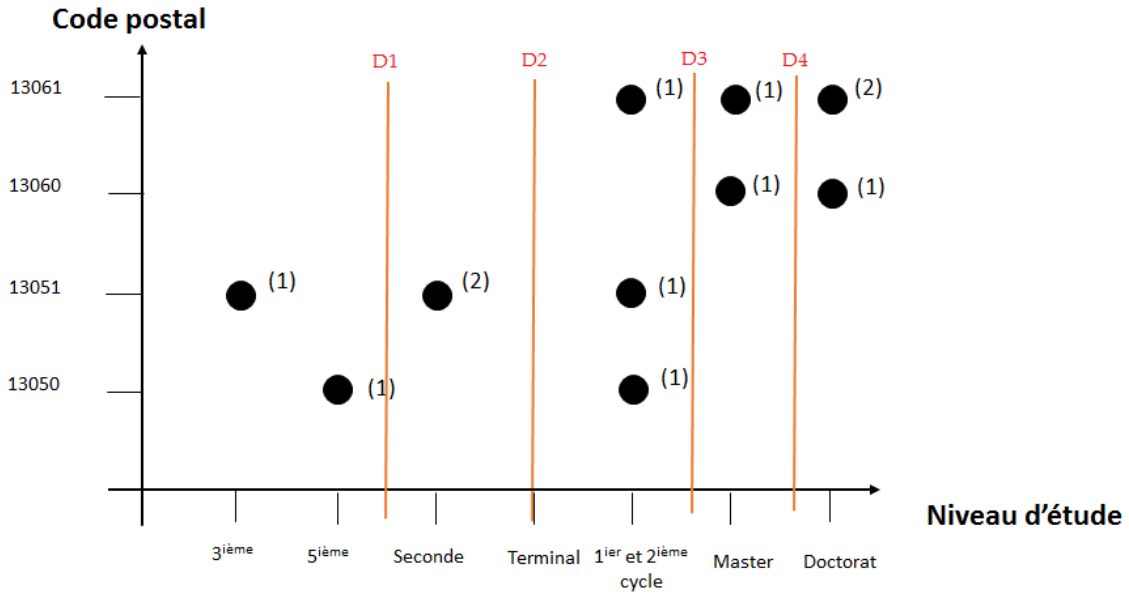
1. Choisit un groupe non marqué.
2. vérifie la possibilité de diviser un groupe en deux sous-groupes selon une dimension choisie. Si la division n'est pas possible, alors, l'algorithme marque le groupe choisi, si non, il applique la division ayant le meilleur score selon la métrique de MSE parmi les divisions candidates.

3. Vérifie s'il existe encore des groupes non marqués. Si oui, l'algorithme passe l'étape 1 si non à l'étape 4.
4. Faire le recodage en remplaçant les différentes valeurs d'une même zone par la valeur de leur premier parent commun.

Itération 1

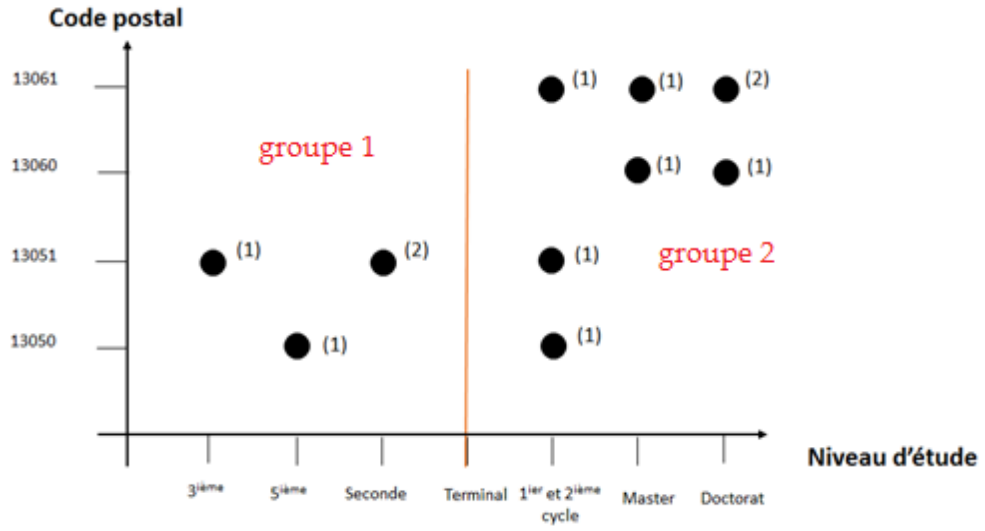
Etape 1: l'ensemble de la table est choisi.

Etape 2: selon la dimension niveau d'étude, il y a quatre divisions candidates (D1, D2, D3 et D4)



- **Score (D1)=** [(1200-1250)+(1300-1250)]+ [(1200-2270)+(1300-2270)+(1500-2270)+(1500-2270)+(1600-2270)+(2000-2270)+(2100-2270)+(3000-2270)+(4000-2270)+(4500-2270)] = 50.
- **Score (D2)=** [(1200-1250)+(1300-1250)+(1200-1250)+(1300-1250)]+ [(1500-2525)+(1500-2525)+(1600-2525)+(2000-2525)+(2100-2525)+(3000-2525)+(4000-2525)+(4500-2525)] = 0
- **Score (D3)=** [(1200-1370)+(1300-1370)+(1200-1370)+(1300-1370) +(1500-1370)+(1500-1370)+(1600-1370)]+ [(2000-3120)+(2100-3120)+(3000-3120)+(4000-3120)+(4500-3120)] = 10
- **Score (D4)=** [(1200-1520)+(1300-1520)+(1200-1520)+(1300-1520) +(1500-1520)+(1500-1520)+(1600-1520) + (2000-1520)+(2100-1520)]+ [(3000-3830)+(4000-3830)+(4500-3830)] = 30

La meilleure division est D2. Groupe 1 et goupe 2 sont construits à partir de cette division



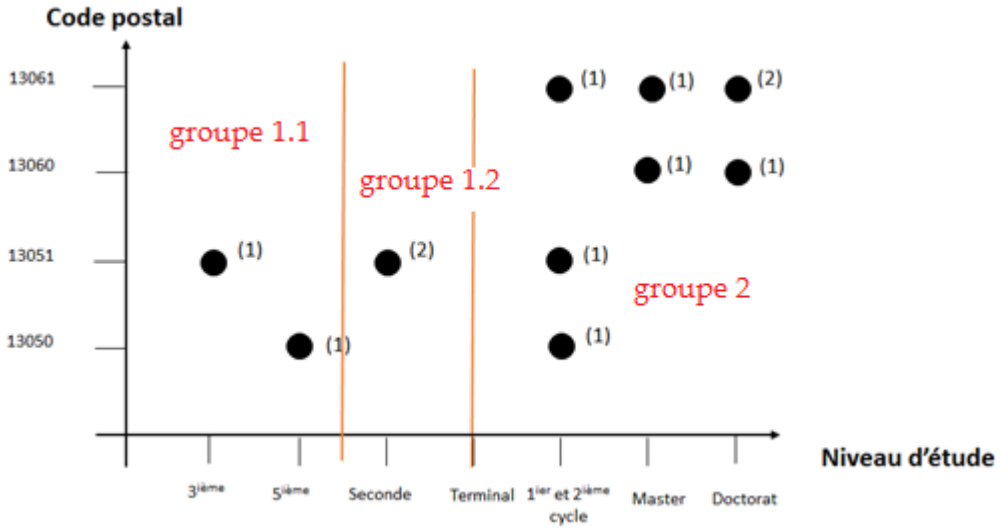
Etape 3 : Groupe 1 et groupe 2 ne sont pas marqués.

Itération 2

Etape 1: Groupe 1 est choisi

Etape 2: selon dimension niveau d'étude, une seule division candidate. Groupe 1.1 et groupe 1.2 sont construit à partir de cette division.

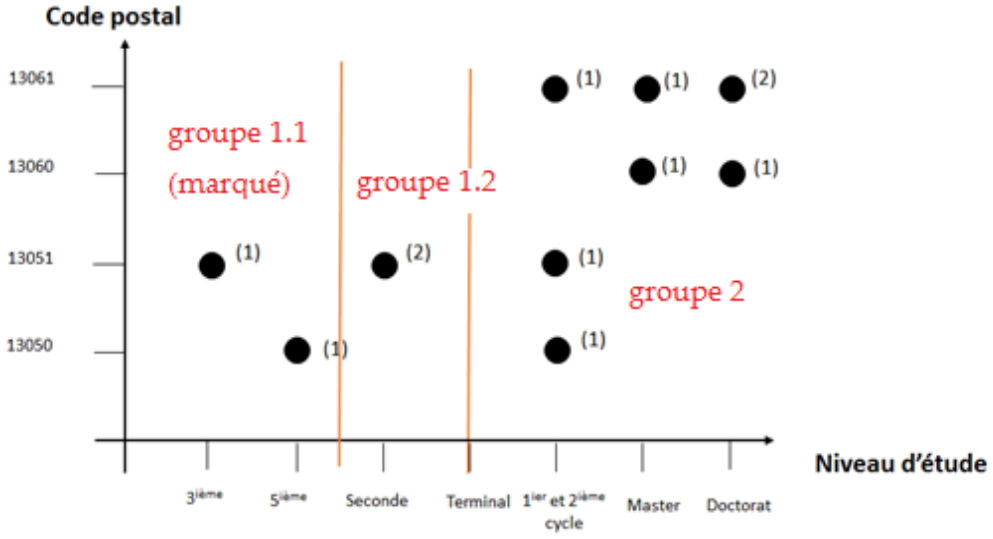
Etape 3 : Groupe 1.1, groupe 1.2 et groupe 2 ne sont pas marqués.



Itération 3

Etape 1: Groupe 1.1 est choisi

Etape 2 : Il n'existe aucune division candidate. L'algorithme le marque.

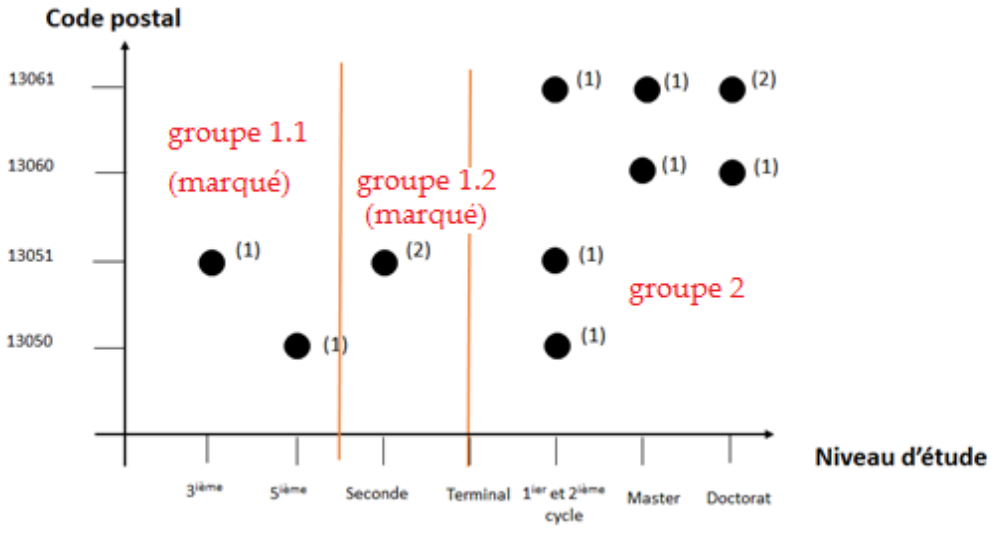


Etape 3: Groupe 1.2 et groupe 2 ne sont pas marqués.

Itération 4

Etape 1: Groupe 1.2 est choisi

Etape 2: Il n'existe aucune division candidate. L'algorithme le marque.

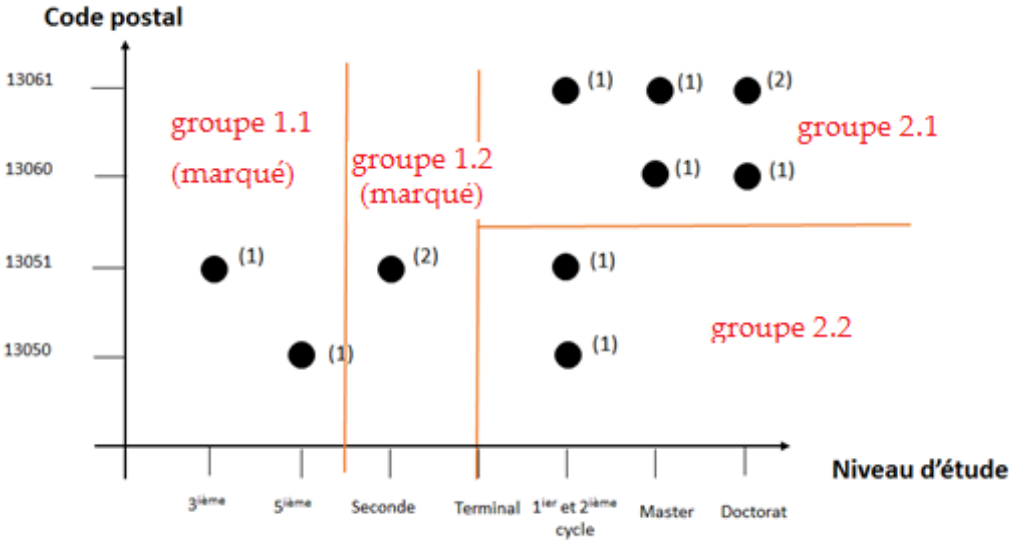


Etape 3 : groupe 2 n'est pas marqué.

Itération 4

Etape 1: Groupe 2 est choisi

Etape 2: selon la dimension code postal, une seule division candidate. Les groupes 2.1 et 2.2 sont construits à partir de cette division.

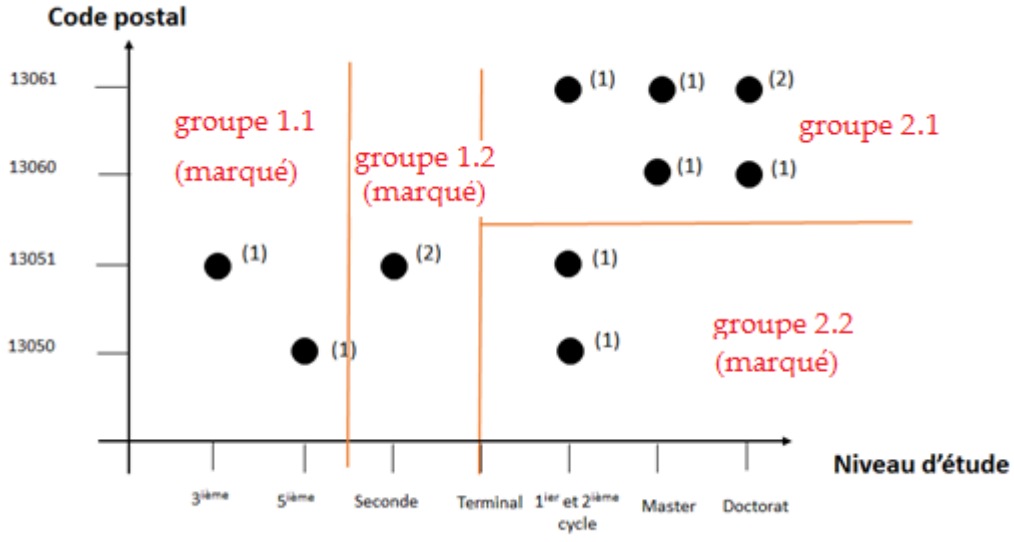


Etape 3 : Groupe 2.1 et groupe 2.2 ne sont pas marqués.

Itération 5

Etape 1: Groupe 2.2 est choisi

Etape 2: Il n'existe aucune division candidate. L'algorithme le marque.

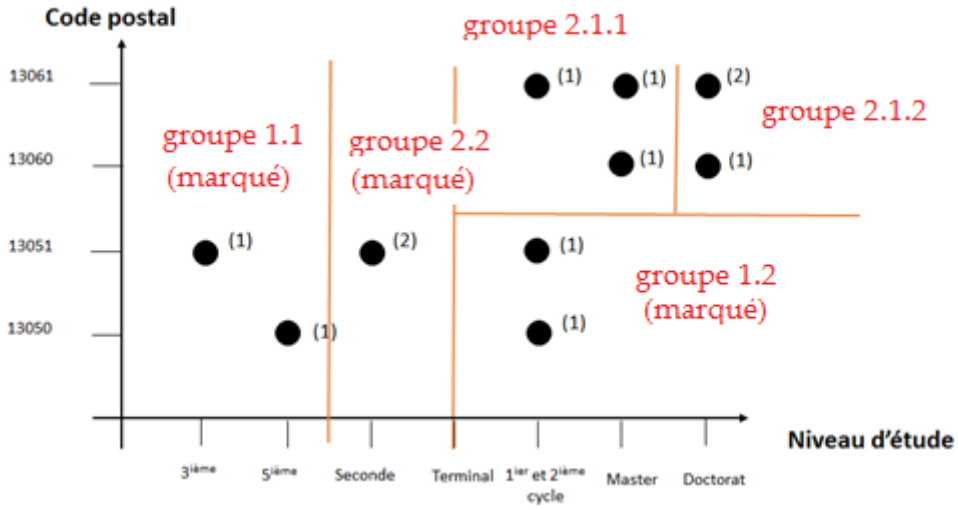


Etape 3 : Groupe 2.1 n'est pas marqué.

Itération 6

Etape 1: Groupe 2.1 est choisi

Etape 2: selon dimension niveau d'étude, une seule division candidate. Les groupes 2.1.1 et 2.1.2 sont construits à partir de cette division.

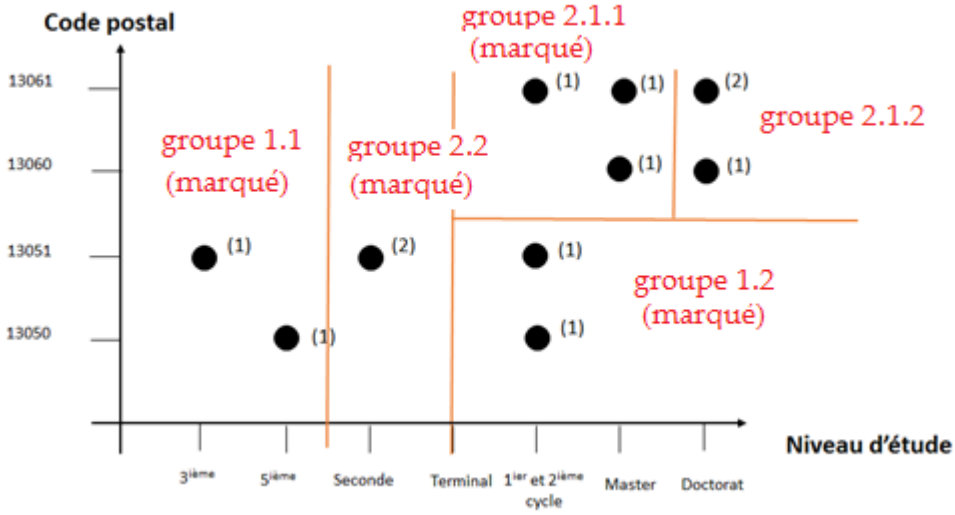


Etape 3 : Les groupes 2.1.1 et 2.1.2 ne sont pas marqués.

Itération 7

Etape 1: Groupe 2.1.1 est choisi

Etape 2: Il n'existe aucune division candidate. L'algorithme le marque.

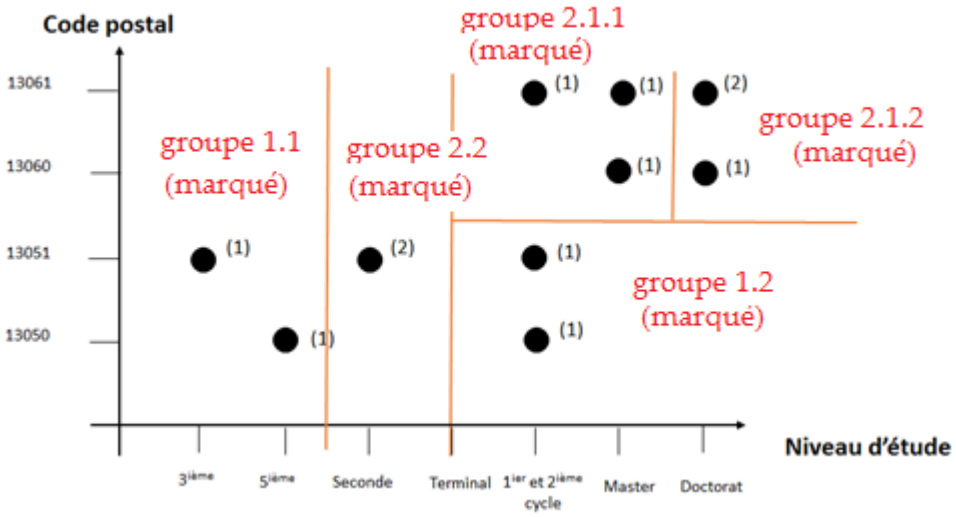


Etape 3 : Groupe 2.1.2 n'est pas marqué.

Itération 8

Etape 1: Groupe 2.1.2 est choisi

Etape 2: Il n'existe aucune division candidate. L'algorithme le marque.



Etape 3 : Tous les groupes sont marqués.

Etape 4 : Le résultat du recodage est la table suivante **:**

code postal	niveau d'étude	Salaire
1305*	collège	1200
1305*	collège	1300
13051	Seconde	1200
13051	Seconde	1300
1305*	1ier et 2ième cycle	1500
1305*	1ier et 2ième cycle	1500
1306*	université	1600
1306*	université	2000
1306*	université	2100
1306*	Doctorat	3000
1306*	Doctorat	4000
1306*	Doctorat	4500

Annexe B Evaluation des abstractions des algorithmes de généralisation

FORMULAIRE 1

Soit:

PT : Original table.
|**PT**| : number of tuples in PT.
PT[QI] : PT table that contains only the QI attributes

Enoncé «Algorithm A» :

Input: Private Table **PT**; quasi-identifier QI = $(A_1, \ldots, A_n)$,
k constraint; hierarchies DGH_{Ai}, where i=1,…,n.
Output: MGT, a generalization of PT[QI] with respect to k
Assumes: |PT| ≥ k
Method:

1. freq ← a frequency list contains distinct sequences of values of PT[QI], along with the number of occurrences of each sequence.
2. **while there exists** sequences in freq occurring less than k times that account for more than k tuples **do**
   2.1. **let** A_j be attribute in freq having the most number of distinct values
   2.2. freq ← generalize the values of A_j in freq
3. freq ← suppress sequences in freq occurring less than k times.
4. freq ← enforce k requirement on suppressed tuples in freq.
5. **Return** MGT ← construct table from freq

Quelques informations complémentaires

Step 1 constructs freq, which is a frequency list containing distinct sequences of values from PT[QI], along with the number of occurrences of each sequence. Each sequence in freq represents one or more tuples in a table.

Step 2.1 uses a heuristic to guide generalization: The attribute having the most number of distinct values in freq is generalized.

Generalization continues until there remains k or fewer tuples having distinct sequences in freq.

Step 3 suppresses any sequences of freq occurring less than k times.

Complementary suppression is performed in step 4 so that the number of suppressed tuples satisfies the k requirement.

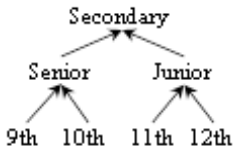
Finally, step 5 produces a table MGT, based on freq such that the values stored as a sequence in freq appear as tuple(s) in MGT replicated in accordance to the stored frequency

Exécution de «Algorithm A» sur la table suivante :

Name	Age	Education	Disease
Alice	27	9^{th}	Diabetes
Jean	30	9^{th}	Cancer
Ines	27	10^{th}	Flu
David	23	11^{th}	Flu
Bob	30	11^{th}	Cancer
Dupont	32	11^{th}	Cancer
Adam	19	12^{th}	Flu
Bryan	35	12^{th}	Diabetes

Avec :

- k = 2
- QI = {Age, Education}
- Ainsi que les hiérarchies de généralisations suivantes :



La hiérarchie de généralisation de l'attribut «Education»

La hiérarchie de généralisation de l'attribut «Age»

1. INSCRIVEZ L'HORAIRE DE DEBUT DE DEROULEMENT DE CET ALGORITHME :

H	Mn

2. DONNER LES RESULTATS DES DIFFERENTES ETAPES DE «ALGORITHME C»

Résultat(s) Itération 1:

.

.

.

3. INSCRIVEZ L'HORAIRE DE FIN DE DEROULEMENT DE CET ALGORITHME :

H	Mn

FORMULAIRE 2

Soit:

T : Original table.

Enoncé «Algorithme B» :

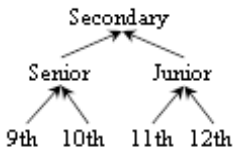
1. WHILE the number of tuples that do not satisfy k-anonymity is greater then k **DO**
  1.1 Select the attribute that has the highest number of distinct values in T
  1.2. Generalize its values.
  1.3. Compute the number of tuples that do not satisfy k-anonymity
2. IF the number of tuples that do not satisfy k-anonymity is less then or equal to k **THEN** delete from T rows that do not satisfy k-anonymity.
3. Return the anonymized table T.

Exécution de «Algorithm B» sur la table suivante :

Name	Age	Education	Disease
Alice	27	9^{th}	Diabetes
Jean	30	9^{th}	Cancer
Ines	27	10^{th}	Flu
David	23	11^{th}	Flu
Bob	30	11^{th}	Cancer
Dupont	32	11^{th}	Cancer
Adam	19	12^{th}	Flu
Bryan	35	12^{th}	Diabetes

Avec :

- k = 2
- QI = {Age, Education}
- Ainsi que les hiérarchies de généralisations suivantes :



La hiérarchie de généralisation de l'attribut «Education»

La hiérarchie de généralisation de l'attribut «Age»

1.	INSCRIVEZ L'HORAIRE DE DEBUT DE DEROULEMENT DE CET ALGORITHME :

H	Mn

2.	DONNER LES RESULTATS DES DIFFERENTES ETAPES DE «ALGORITHME C»

Résultat(s) itération 1 de l'étape 1:

.

.

.

3.	INSCRIVEZ L'HORAIRE DE FIN DE DEROULEMENT DE CET ALGORITHME :

H	Mn

FORMULAIRE 3

Soit:

Partition : A set of tuple of a table to anonymize. .

Enoncé «Algorithm C» :

```
Anonymize (partition)
    1. If (no allowable multidimensional cut for partition)
    2. then return Φ : partition → summary
    3. Else
       3.1 dim ← choose_dimension()
       3.2 fs ← frequency_set(partition, dim)
       3.3 splitVal ← find_median(fs)
       3.4 lhs ← {t ∈ partition : t.dim ≤ splitVal}
       3.5 rhs ← {t ∈ partition : t.dim > splitVal}
       3.6 return Anonymize (rhs) ∪ Anonymize (lhs)
```

Quelques informations complémentaires

- For the first iteration «partition» represents all the tuples of the table to anonymize
- The function choose_dimension() returns the chosen dimension
- The function frequency_set(partition, dim) returns the set «fs» of values taken by a given dimension «dim» in a given partition «partition»
- The function find_median(fs) returns the value "splitVAl" of the median
- t.dim is the value of a given dimension «dim» in given tuple «t»
- Use for summary the function RANGE

Exécution de «Algorithm C» sur la table suivante :

Name	Age	Education	Disease
Alice	27	9^{th}	Diabetes
Jean	30	9^{th}	Cancer
Ines	27	10^{th}	Flu
David	23	11^{th}	Flu
Bob	30	11^{th}	Cancer
Dupont	32	11^{th}	Cancer
Adam	19	12^{th}	Flu
Bryan	35	12^{th}	Diabetes

Avec :

k = 2

QI = {Age, Education}

4. INSCRIVEZ L'HORAIRE DE DEBUT DE DEROULEMENT DE CET ALGORITHME :

H	Mn

5. DONNER LES RESULTATS DES DIFFERENTES ETAPES DE «ALGORITHME C»

Résultat(s) Itération 1:

.

.

.

6. INSCRIVEZ L'HORAIRE DE FIN DE DEROULEMENT DE CET ALGORITHME :

H	Mn

FORMULAIRE 4

Soit:

Partition : A set of tuple of a table to anonymize. .

Enoncé «Algorithm D» :

1. **Compute the partition containing all tuples of the table to anonymize and mark it with «unprocessed»**
2. **While there exists a partition marked «unprocessed» Do**
   - **2.1 Choose a partition P among those marked «unprocessed» and change its marking by «in process»**
   - **2.2 If (there exists at least one authorized using the median with respect to a given dimension) Then**
     - **2.2.1 Then choose among those authorized a division**
     - **2.2.2 Compute, from P (the partition marked «in process»), the two partitions that are to the right and left of the division and mark each of them with «unprocessed»**
     - **2.2.3 Change the marking of P by «Partitioned»**
   - **Else 2.2.4 Change the marking of P by « Non Partitionnable»**
3. **For each partition marked «Non Partitionnable» Do**
   - **3.1 compute summary**

Quelques informations complémentaires

- Use for summary the function RANGE

Exécution de «Algorithm D» sur la table suivante :

Name	Age	Education	Disease
Alice	27	9^{th}	Diabetes
Jean	30	9^{th}	Cancer
Ines	27	10^{th}	Flu
David	23	11^{th}	Flu
Bob	30	11^{th}	Cancer
Dupont	32	11^{th}	Cancer
Adam	19	12^{th}	Flu
Bryan	35	12^{th}	Diabetes

Avec :

k = 2

QI = {Age, Education}

1. INSCRIVEZ L'HORAIRE DE DEBUT DE DEROULEMENT DE CET ALGORITHME :

H	Mn

2. DONNER LES RESULTATS DES DIFFERENTES ETAPES DE «ALGORITHME C»

Résultat(s) Itération 1:

.

.

.

3. INSCRIVEZ L'HORAIRE DE FIN DE DEROULEMENT DE CET ALGORITHME :

H	Mn

Annexe C Evaluation de l'approche

1. Les métadonnées

La base de données 'Origianl_DB' contient :

- 1000 enregistrements
- 4 attributs :
  - **Age** (type numérique et quasi-identifiant)
  - **Niveau d'étude** (type catégoriel et quasi-identifiant)
  - **Sexe** (type catégoriel et quasi-identifiant)
  - **Salaire** (type catégoriel et attribut sensible)

Les hiérarchies de généralisation des attributs quasi-identifiants sont présentées comme suit :

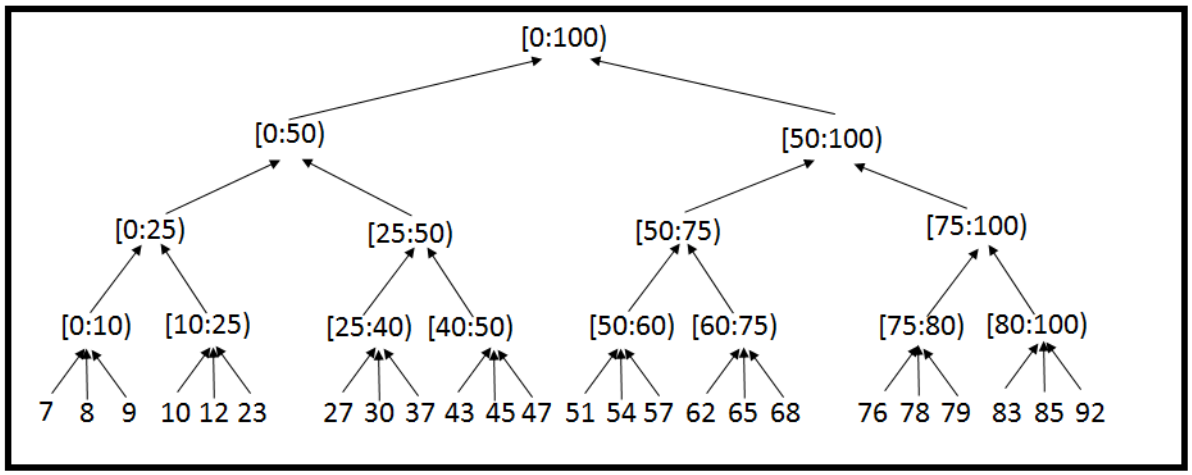


Figure 116. La hiérarchie de généralisation de l'attribut âge

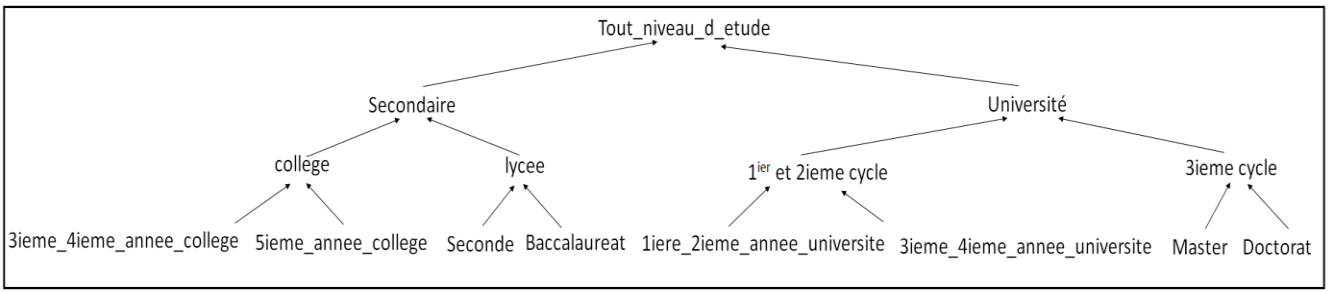


Figure 117. La hiérarchie de généralisation de l'attribut niveau d'étude

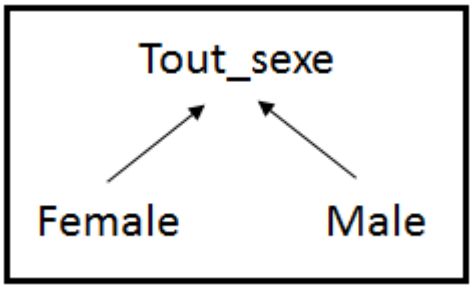


Figure 118. La hiérarchie de généralisation de l'attribut sexe

L'outil utilise une fonction de mappage afin d'appliquer la généralisation à des attributs catégoriels et les remplacer par des intervalles. Chaque catégorie est remplacée, si les valeurs sont les feuilles de la hiérarchie de généralisation par un entier, si non, par un intervalle.

Les correspondances des valeurs de domaine de l'attribut niveau d'étude

La valeur 'tout_niveau_d_etude' est remplacée par [0 : 11].

La valeur 'Secondaire' est remplacée par [0 : 6).

La valeur 'college' est remplacée par [0 : 4).

La valeur '3ieme_annee_college' est remplacée par '1'.

La valeur '4ieme_annee_college' est remplacée par '2'.

La valeur '5ieme_annee_college' est remplacée par '3'.

La valeur 'lycee' est remplacée par [4 : 6).

La valeur 'Seconde est remplacée par '4'.

La valeur 'Baccalauréat' est remplacée par '5'.

La valeur 'Université est remplacée par [6 : 11].

La valeur ''1ier et 2 ieme cycle'' est remplacée par [6 : 9]

La valeur '1iere_annee_universite' est remplacée par '6'.

La valeur '2iem_annee_universite' est remplacée par '7'.

La valeur '3iem_annee_universite' est remplacée par '8'.

La valeur '4iem_annee_universite' est remplacée par '9'.

La valeur ‘3 ieme cycle’ est remplacée par [10 : 11].

La valeur ‘Master’ est remplacée par '10'.

La valeur ‘Doctorat’ est remplacée par '11'.

Le mappage des valeurs de domaine de l’attribut sexe

La valeur ‘tout_sexe’ est remplacée par [0 : 1].

La valeur ‘Female’ est remplacée par '0'.

La valeur ‘Male’ est remplacée par '1'.

2. Questionnaire

Questionnaire

Type de guidage :

Nom et Prénom :

Heure du début :

Heure de fin :

Solution choisie :

Observations :

Pour chaque question, cochez la bonne réponse :

1. La suppression des données est autorisée dans chaque algorithme d'anonymisation

VRAI ☐ **FAUX** ☐ **Je ne sais pas** ☐

2. Tous les algorithmes peuvent être utilisés pour n'importe quel besoin d'usage de fouille de données (par exemple la classification ou la régression).

VRAI ☐ **FAUX** ☐ **Je ne sais pas** ☐

3. Dans un jeu de données anonymisées, toutes les valeurs qui se trouvent dans la même colonne d'un attribut sont généralisées à différents niveaux de la hiérarchie de généralisation de cet attribut.

VRAI ☐ **FAUX** ☐ **Je ne sais pas** ☐

4. Dans un jeu de données anonymisées, toutes les valeurs qui se trouvent dans la même colonne d'un attribut sont généralisées au même niveau de la hiérarchie de généralisation de cet attribut.

VRAI ☐ **FAUX** ☐ **Je ne sais pas** ☐

5. Un algorithme d'anonymisation utilise une métrique de compromis entre qualité et sécurité afin de choisir la meilleure généralisation.

VRAI ☐ **FAUX** ☐ **Je ne sais pas** ☐

6. La valeur du paramètre d'entrée 'K' permet d'estimer le compromis entre la sécurité et la qualité des données.

VRAI ☐ **FAUX** ☐ **Je ne sais pas** ☐

7. Il existe un lien entre la valeur du paramètre d'entrée 'K' et le seuil derisque des données anonymisées.

VRAI ☐ **FAUX** ☐ **Je ne sais pas** ☐

8. Plus la valeur de 'K' est grande, plus le risque est grand.

VRAI ☐ **FAUX** ☐ **Je ne sais pas** ☐

9. La valeur du paramètre d'entrée 'MaxSup' représente le nombre maximal des valeurs à pouvoir supprimer dans une table anonymée.

VRAI ☐ **FAUX** ☐ **Je ne sais pas** ☐

10. La valeur du paramètre d'entrée 'MaxSup' représente le nombre maximal des enregistrements à pouvoi supprimer dans une table anonymée.

VRAI ☐ **FAUX** ☐ **Je ne sais pas** ☐

11. Le risque d'une Base de Données anonymisée est la moyenne des risques de l'ensemble des classes d'équivalence.

VRAI ☐ **FAUX** ☐ **Je ne sais pas** ☐

12. Le risque d'une Base de Données anonymisée est la somme des risques de l'ensemble des classes d'équivalence.

VRAI ☐ **FAUX** ☐ **Je ne sais pas** ☐

13. Le risque d'une Base de Données anonymisée est le maximum des risques de l'ensemble des classes d'équivalence.

VRAI ☐ **FAUX** ☐ **Je ne sais pas** ☐

14. Selon la métrique de discernabilité (la métrique qui mesure la précision des données), moins les classes d'équivalence contiennent des enregistrements, moins les données sont précises.

VRAI ☐ **FAUX** ☐ **Je ne sais pas** ☐

Combien êtes satisfait du guidage de l'outil ?

1	2	3	4	5	6	7	8	9	10

Feten BEN FREDJ

Méthode et outil d'anonymisation des données sensibles

Résumé

L'anonymisation des données personnelles requiert l'utilisation d'algorithmes complexes permettant de minimiser le risque de ré-identification tout en préservant l'utilité des données. Dans cette thèse, nous décrivons une approche fondée sur les modèles qui guide le propriétaire des données dans son processus d'anonymisation. Le guidage peut être informatif ou suggestif. Il permet de choisir l'algorithme le plus pertinent en fonction des caractéristiques des données mais aussi de l'usage ultérieur des données anonymisées. Le guidage a aussi pour but de définir les bons paramètres à appliquer à l'algorithme retenu. Dans cette thèse, nous nous focalisons sur les algorithmes de généralisation de micro-données. Les connaissances liées à l'anonymisation tant théoriques qu'expérimentales sont stockées dans une ontologie.

MOTS-CLES : guidage, sécurité, ontologie, méthodologie, respect de la vie privée, anonymisation, approche guidée par

Résumé en anglais

Personel data anonymization requires complex algorithms aiming at avoiding disclosure risk without losing data utility. In this thesis, we describe a model-driven approach guiding the data owner during the anonymization process. The guidance may be informative or suggestive. It helps the data owner in choosing the most relevant algorithm given the data characteristics and the future usage of anonymized data. The guidance process also helps in defining the best input values for the algorithms. In this thesis, we focus on generalization algorithms for micro-data. The knowledge about anonymization is composed of both theoretical aspects and experimental results. It is managed thanks to an ontology

KEYWORDS: guidance, security, ontology, methodology, privacy, anonymization, model-driven