Nº 12-001-X au catalogue
ISSN 1712-5685

Techniques d'enquête

Une généralisation du paradigme de Fellegi-Holt pour la localisation automatique des erreurs

par Sander Scholtus

Date de diffusion : le 22 juin 2016

Statistique Canada Statistics Canada

Canada

Comment obtenir d'autres renseignements

Pour toute demande de renseignements au sujet de ce produit ou sur l'ensemble des données et des services de Statistique Canada, visiter notre site Web à www.statcan.gc.ca.

Vous pouvez également communiquer avec nous par :

Courriel à STATCAN.infostats-infostats.STATCAN@canada.ca

Téléphone entre 8 h 30 et 16 h 30 du lundi au vendredi aux numéros sans frais suivants :

- Service de renseignements statistiques 1-800-263-1136
- Service national d'appareils de télécommunications pour les malentendants 1-800-363-7629
- Télécopieur 1-877-287-4369

Programme des services de dépôt

- Service de renseignements 1-800-635-7943
- Télécopieur 1-800-565-7757

Normes de service à la clientèle

Statistique Canada s'engage à fournir à ses clients des services rapides, fiables et courtois. À cet égard, notre organisme s'est doté de normes de service à la clientèle que les employés observent. Pour obtenir une copie de ces normes de service, veuillez communiquer avec Statistique Canada au numéro sans frais 1-800-263-1136. Les normes de service sont aussi publiées sur le site www.statcan.gc.ca sous « Contactez-nous » > « Normes de service à la clientèle ».

Note de reconnaissance

Le succès du système statistique du Canada repose sur un partenariat bien établi entre Statistique Canada et la population du Canada, les entreprises, les administrations et les autres organismes. Sans cette collaboration et cette bonne volonté, il serait impossible de produire des statistiques exactes et actuelles.

Signes conventionnels dans les tableaux

Les signes conventionnels suivants sont employés dans les publications de Statistique Canada :

.	indisponible pour toute période de référence
..	indisponible pour une période de référence précise
...	n'ayant pas lieu de figurer
0	zéro absolu ou valeur arrondie à zéro
0^s	valeur arrondie à 0 (zéro) là où il y a une distinction importante entre le zéro absolu et la valeur arrondie
^{p}	provisoire
^{r}	révisé
x	confidentiel en vertu des dispositions de la *Loi sur la statistique*
^{E}	à utiliser avec prudence
F	trop peu fiable pour être publié
*	valeur significativement différente de l'estimation pour la catégorie de référence (p<0,05)

Une version HTML est aussi disponible.

This publication is also available in English.

Techniques d'enquête, juin 2016
Vol. 42, nº 1, pp. 1-20
Statistique Canada, nº 12-001-X au catalogue

Une généralisation du paradigme de Fellegi-Holt pour la localisation automatique des erreurs

Sander Scholtus[1]

Résumé

La vérification automatique consiste en l'utilisation d'un ordinateur pour déceler et corriger sans intervention humaine les valeurs erronées dans un ensemble de données. La plupart des méthodes de vérification automatique actuellement employées aux fins de la statistique officielle sont fondées sur les travaux fondamentaux de Fellegi et Holt (1976). La mise en application de cette méthode dans la pratique révèle des différences systématiques entre les données vérifiées manuellement et celles qui sont vérifiées de façon automatisée, car l'humain est en mesure d'effectuer des opérations de vérification complexes. L'auteur du présent article propose une généralisation du paradigme de Fellegi-Holt qui permet d'intégrer de façon naturelle une grande catégorie d'opérations de vérification. Il présente aussi un algorithme qui résout le problème généralisé de localisation des erreurs qui en découle. Il est à espérer que cette généralisation puisse améliorer la pertinence des vérifications automatiques dans la pratique et ainsi accroître l'efficience des processus de vérification des données. Certains des premiers résultats obtenus à l'aide de données synthétiques sont prometteurs à cet égard.

Mots-clés : Vérification automatique; opérations de vérification; maximum de vraisemblance; données numériques; vérifications linéaires.

1 Introduction

Les données recueillies aux fins de la production de statistiques contiennent inévitablement des erreurs. Il est donc nécessaire de mettre en place un processus de vérification des données pour déceler et corriger ces erreurs, au moins dans la mesure où elles ont un effet appréciable sur la qualité des produits statistiques (Granquist et Kovar 1997). Traditionnellement, la vérification des données se faisait manuellement, idéalement par des vérificateurs spécialisés ayant une connaissance approfondie du sujet. Pour améliorer l'efficience, la rapidité et la reproductibilité de la vérification, beaucoup d'instituts de statistique ont tenté d'automatiser certains segments du processus (Pannekoek, Scholtus et van der Loo 2013). Il en a résulté des méthodes de correction déductive des *erreurs systématiques* et des algorithmes de localisation des erreurs pour les *erreurs aléatoires* (de Waal, Pannekoek et Scholtus 2011, chapitre 1). Le présent article est axé sur la vérification automatique des erreurs aléatoires.

Les méthodes pour l'exécution de cette tâche comprennent généralement un ajustement minimal de chaque enregistrement de données en fonction de certains critères d'optimisation, afin d'en assurer la cohérence avec un ensemble déterminé de contraintes que l'on appelle *règles de vérification*, ou simplement *contrôles*. Selon l'efficacité des critères d'optimisation et la puissance des contrôles, la vérification automatique peut remplacer en partie la vérification manuelle traditionnelle. Dans les faits, la vérification automatique est presque toujours jumelée à une forme quelconque de *vérification sélective*, ce qui signifie que les erreurs ayant les répercussions les plus importantes sont traitées manuellement (Hidiroglou et Berthelot 1986; Granquist 1995, 1997; Granquist et Kovar 1997; Lawrence et McKenzie 2000; Hedlin 2003; de Waal et coll. 2011).

1. Sander Scholtus, Statistics Netherlands, Department of Process Development and Methodology, P.O. Box 24500, 2490 HA, La Haye, Pays-Bas. Courriel : sshs@cbs.nl.

La plupart des méthodes de vérification automatique actuellement utilisées pour la statistique officielle sont fondées sur le paradigme de Fellegi et Holt (1976) : pour chaque enregistrement, on trouve le plus petit sous-ensemble de variables erronées qui peuvent être imputées de sorte que l'enregistrement satisfasse aux contrôles. On peut obtenir une légère généralisation en attribuant ce qu'on appelle *poids de confiance* aux variables et en minimisant le poids total des variables imputées. Une fois résolu ce *problème de localisation des erreurs*, il faut trouver séparément de nouvelles valeurs qui conviennent pour les variables identifiées comme étant erronées. C'est ce qu'on appelle le *problème d'imputation cohérente*; voir à ce sujet de Waal et coll. (2011) et les ouvrages cités en référence. Le présent article est axé sur le problème de localisation des erreurs.

À *Statistics Netherlands*, la localisation des erreurs à l'aide du paradigme de Fellegi-Holt fait partie du processus de vérification des données relatives aux statistiques structurelles sur les entreprises (SSE) depuis plus de dix ans. Dans le cadre d'études d'évaluation, où les mêmes données sur les SSE ont été vérifiées à la fois automatiquement et manuellement, on a constaté un certain nombre de différences systématiques entre les deux processus. Bon nombre de ces différences pouvaient s'expliquer par le fait que les vérificateurs humains ont apporté certains types de correction qui ne sont pas optimaux selon le paradigme de Fellegi-Holt. Par exemple, les vérificateurs ont parfois interverti les valeurs de dépenses et de revenus associés ou transféré une partie des unités déclarées d'une variable à l'autre.

En pratique, le résultat de la vérification manuelle est généralement considéré comme étant la « norme de référence » pour évaluer la qualité de la vérification automatique. Une évaluation critique de cette hypothèse dépasse le cadre du présent article; toutefois, le lecteur intéressé pourra consulter EDIMBUS (2007, pages 34-35). On souligne simplement ici qu'en améliorant la capacité des méthodes de vérification automatique à reproduire les résultats de la vérification manuelle, on accroît leur utilité dans la pratique. Par ricochet, cela signifie que l'on peut accroître la part de la vérification automatique pour améliorer l'efficience du processus de vérification des données (Pannekoek et coll. 2013).

Dans une certaine mesure, les différences systématiques entre la vérification automatique et la vérification manuelle pourraient être éliminées par l'application judicieuse de poids de confiance. En règle générale, toutefois, les effets d'une modification des poids de confiance sur les résultats de la vérification automatique sont difficiles à prévoir. En outre, si les vérificateurs apportent un certain nombre de corrections différentes et complexes, il pourrait être impossible de toutes les modéliser sous le paradigme de Fellegi-Holt à l'aide d'un seul ensemble de poids de confiance. Une autre solution consiste à essayer de déceler les erreurs pour lesquelles on sait que le paradigme de Fellegi-Holt donne un résultat insatisfaisant dès les premières étapes du processus de vérification des données, c'est-à-dire durant la correction déductive des erreurs systématiques à l'aide de règles de correction automatique (de Waal et coll. 2011; Scholtus 2011). Cette méthode comporte toutefois des limites pratiques; elle peut notamment exiger un grand nombre de règles du type « si-alors », qui peuvent se révéler difficiles à concevoir et à tenir à jour au fil du temps (Chen, Thibaudeau et Winkler 2003). En outre, il n'est pas nécessairement aisé de trouver des règles de correction appropriées pour toutes les erreurs qui ne peuvent pas être traitées en vertu du paradigme de Fellegi-Holt.

Dans le présent article, on propose une autre approche : une nouvelle définition du problème de localisation des erreurs qui tient compte de la possibilité qu'une erreur puisse toucher plus d'une variable à la fois. On montre que ce problème contient la localisation des erreurs en vertu du paradigme original de Fellegi-Holt comme un cas particulier. Le présent article porte principalement sur les données numériques

et les règles de vérification linéaires; un élargissement possible aux données catégoriques et mixtes est présenté brièvement à la section 8.

Le reste de l'article se présente comme suit. La section 2 passe brièvement en revue les travaux antérieurs pertinents dans le domaine. À la section 3, on présente et on illustre le concept des opérations de vérification. Le nouveau problème de localisation des erreurs est formulé en termes de ces opérations à la section 4. La section 5 énonce une généralisation d'une méthode existante pour trouver des solutions au problème de localisation des erreurs fondé sur le paradigme de Fellegi-Holt, et le résultat est utilisé à la section 6 pour construire un algorithme possible pour la résolution du nouveau problème. Une étude par simulations de petite envergure est présentée à la section 7. Enfin, à la section 8, on énonce certaines conclusions et on formule des questions pour approfondir la recherche.

2 Contexte et travaux connexes

Soit $\mathbf{x} = (x_1, \ldots, x_p)' \in \mathbb{R}^p$ un enregistrement de p variables numériques. Supposons que cet enregistrement doive satisfaire à k règles de vérification, se présentant sous la forme du système d'(in)égalités linéaires suivant :

$$\mathbf{A}\mathbf{x} + \mathbf{b} \odot \mathbf{0}, \tag{2.1}$$

où $\mathbf{A} = (a_{rj})$ est une matrice $k \times p$ de coefficients et $\mathbf{b} = (b_1, \ldots, b_k)'$ est un vecteur de constantes. Ici comme ailleurs, $\mathbf{0}$ représente un vecteur de zéros de longueur appropriée; de même, $\odot$ représente un vecteur symbolique d'opérateurs de l'ensemble $\{\geq, \leq, =\}$.

Pour un enregistrement donné $\mathbf{x}$ qui ne satisfait pas à toutes les règles de vérification énoncées en (2.1), le problème de localisation des erreurs fondée sur le paradigme de Fellegi-Holt consiste à trouver la valeur minimale de

$$\sum_{j=1}^{p} w_j \delta_j, \tag{2.2}$$

où $w_j > 0$ est le poids de confiance de la variable x_j et $\delta_j \in \{0,1\}$, à condition qu'on puisse assurer la cohérence de l'enregistrement original avec les règles de vérification en imputant uniquement les variables x_j pour lesquelles $\delta_j = 1$ (de Waal et coll. 2011, page 66).

Fellegi et Holt (1976) ont aussi proposé une méthode de résolution du problème de localisation des erreurs ci-dessus fondée sur la production d'un ensemble suffisant de *vérifications implicites* (voir ci-dessous). Malheureusement, cette méthode exige souvent un très grand nombre de vérifications implicites. Au cours des dernières décennies, divers algorithmes spécialisés ont été élaborés pour le problème de localisation des erreurs, notamment par Schaffer (1987), Garfinkel, Kunnathur et Liepins (1988), Kovar et Whitridge (1990), Ragsdale et McKeown (1996), de Waal (2003), de Waal et Quere (2003), Riera-Ledesma et Salazar-González (2003, 2007), Bruni (2004), ainsi que de Jonge et van der Loo (2014). Les premiers algorithmes visaient principalement à renforcer la méthode originale de Fellegi et Holt (1976) en réduisant le nombre de vérifications implicites requises. Les algorithmes plus récents reposent sur le fait que le

problème de localisation des erreurs peut être rédigé sous forme de problème de programmation mixte en nombres entiers, ce qui permet l'application de techniques d'optimisation normalisées. Voir aussi de Waal et Coutinho (2005) ou de Waal et coll. (2011) pour une vue d'ensemble et une comparaison des divers algorithmes de localisation des erreurs.

Les vérifications implicites sont des contraintes qui découlent logiquement des règles de vérification originales (2.1). Dans le contexte qui nous occupe (données numériques, vérifications linéaires), toutes les vérifications implicites pertinentes peuvent être générées par une technique appelée *élimination de Fourier-Motzkin* (élimination FM; voir Williams 1986). L'élimination FM transforme un système de contraintes linéaires à p variables en un système de contraintes linéaires implicites à au plus $p-1$ variables; ainsi, au moins une des variables originales est éliminée. Pour les détails mathématiques, consultez l'annexe.

L'élimination FM est assortie de la propriété fondamentale suivante : le système de contraintes implicites est satisfait par les valeurs des variables non éliminées si et seulement s'il existe une valeur pour la variable éliminée qui, prise avec les autres valeurs, satisfait au système original de contraintes. Dans la localisation des erreurs en vertu du paradigme de Fellegi-Holt, on peut, en appliquant à répétition cette propriété fondamentale, vérifier si une combinaison particulière de variables peut être imputée pour obtenir un enregistrement cohérent, compte tenu des valeurs originales des autres variables. L'algorithme de localisation des erreurs de de Waal et Quere (2003) illustre bien cette utilisation de l'élimination FM.

Pour conclure cette section, il est intéressant d'examiner brièvement l'interprétation statistique du problème de localisation des erreurs. En fait, Fellegi et Holt (1976) n'ont fourni aucun argument statistique formel pour expliquer leur paradigme de localisation automatique des erreurs. Leur raisonnement était plutôt intuitif :

> *« Les données de chaque enregistrement doivent être corrigées afin de satisfaire à toutes les règles de vérification en changeant le moins d'éléments de données (champs) possible. Nous sommes d'avis que cette méthode respecte l'idée de garder telles quelles le plus grand nombre possible des données originales, compte tenu des contraintes des règles de vérification, et donc de modifier le moins de données possible. Parallèlement, si les erreurs sont relativement rares, il semble plus probable que l'on puisse identifier les champs réellement erronés. » (Fellegi et Holt 1976, page 18).* [Traduction]

Liepins (1980) ainsi que Liepins, Garfinkel et Kunnathur (1982), en se fondant sur les résultats antérieurs de Naus, Johnson et Montalvo (1972), ont formulé un argument statistique pour minimiser le nombre pondéré de variables imputées. Supposons que les erreurs se produisent selon un processus stochastique, chaque variable x_j étant erronée selon une probabilité p_j qui ne dépend pas de sa valeur réelle et les erreurs étant indépendantes d'une variable à l'autre. Supposons en outre que les poids de confiance sont définis comme suit :

$$w_j = -\log\left(\frac{p_j}{1-p_j}\right). \tag{2.3}$$

On peut alors montrer que la minimisation de l'expression (2.2) correspond approximativement à la maximisation de la vraisemblance de l'enregistrement exempt d'erreur non observé. Soulignons que ces auteurs supposent tacitement qu'une erreur affecte toujours une seule variable à la fois.

D'autres méthodes de localisation des erreurs reposant plus directement sur des modèles statistiques ont été proposées, notamment par Little et Smith (1987) et par Ghosh-Dastidar et Schafer (2006). Ces méthodes ont recours à des techniques de détection des valeurs aberrantes et exigent un modèle explicite pour les données réelles. Malheureusement, elles ne peuvent pas tenir compte de façon directe des règles de vérification comme celle qui est illustrée en (2.1).

3 Opérations de vérification

Poursuivons la notation de la section 2 en définissant une *opération de vérification* g comme une fonction affine de forme générale

$$g(\mathbf{x}) = \mathbf{T}\mathbf{x} + \mathbf{S}\boldsymbol{\alpha} + \mathbf{c}, \tag{3.1}$$

où $\mathbf{T}$ et $\mathbf{S}$ sont des matrices de coefficients connues de dimensions $p \times p$ et $p \times m$, respectivement, $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_m)'$ est un vecteur de paramètres libres qui peuvent se produire dans g, et $\mathbf{c}$ est un vecteur de p constantes connues. Dans le cas particulier où g ne comprend aucun paramètre libre $(m = 0)$, le second terme de l'équation (3.1) disparaît. Il est parfois utile d'imposer une ou plusieurs contraintes linéaires aux paramètres libres de g :

$$\mathbf{R}\boldsymbol{\alpha} + \mathbf{d} \odot \mathbf{0}, \tag{3.2}$$

où $\mathbf{R}$ est une matrice connue et $\mathbf{d}$, un vecteur de constantes connu. (Remarque : La notation matricielle-vectorielle est utilisée tout au long de l'article parce qu'elle permet de décrire avec concision les résultats; le recours à des matrices pour représenter les règles et les opérations de vérification ne constitue toutefois probablement pas le meilleur moyen de traiter ces résultats par ordinateur.)

Comme premier exemple, prenons l'opération qui remplace l'une des valeurs originales de $\mathbf{x}$ par une nouvelle valeur arbitraire (imputation), que nous appellerons *opération FH*, vu son rôle central dans la vérification automatique fondée sur le paradigme de Fellegi-Holt. Soit $\mathbf{I}$ la matrice identité $p \times p$ et $\mathbf{e}_i$ le i^{e} vecteur de base canonique de $\mathbb{R}^p$. L'opération FH qui impute la variable x_j est donnée par (3.1) où $\mathbf{T} = \mathbf{I} - \mathbf{e}_j\mathbf{e}_j'$, $\mathbf{S} = \mathbf{e}_j$ et $\mathbf{c} = \mathbf{0}$. On obtient : $g(\mathbf{x}) = \mathbf{x} + \mathbf{e}_j(\alpha - x_j) = (x_1, \ldots, x_{j-1}, \alpha, x_{j+1}, \ldots, x_p)'$, où $\alpha \in \mathbb{R}$, un paramètre libre représentant la valeur imputée. Soulignons que pour un enregistrement de p variables, on peut définir p opérations FH distinctes.

Pour mieux illustrer le concept d'opération de vérification, d'autres exemples sont présentés ci-dessous. Pour faciliter la notation, ces exemples sont limités au cas où $p = 3$.

- Opération de vérification qui change le signe d'une des variables :

$$g\left(\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}\right) = \begin{pmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} -x_1 \\ x_2 \\ x_3 \end{pmatrix}.$$

- Opération de vérification qui intervertit les valeurs de deux éléments adjacents :

$$g\left(\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}\right) = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} x_2 \\ x_1 \\ x_3 \end{pmatrix}.$$

- Opération de vérification qui transfère un nombre d'unités d'un élément à un autre, où le nombre d'unités transférées peut équivaloir à au plus K unités dans un sens ou dans l'autre :

$$g\left(\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}\right) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} + \begin{pmatrix} 1 \\ 0 \\ -1 \end{pmatrix}\alpha + \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} x_1 + \alpha \\ x_2 \\ x_3 - \alpha \end{pmatrix},$$

  où la contrainte suivante s'applique : $-K \leq \alpha \leq K.$

- Opération de vérification qui impute deux variables simultanément selon un ratio fixe :

$$g\left(\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}\right) = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} + \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{pmatrix}\begin{pmatrix} \alpha_1 \\ \alpha_2 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ x_3 \end{pmatrix},$$

  où la contrainte suivante s'applique : $\boldsymbol{\alpha} = (\alpha_1, \alpha_2)'$ satisfait $10\alpha_1 - \alpha_2 = 0$.

Intuitivement, une opération de vérification est censée « renverser les effets » d'un type particulier d'erreur qui aurait pu se produire dans les données observées, c'est-à-dire que si l'erreur associée à l'opération de vérification g s'est réellement produite dans l'enregistrement $\mathbf{x}$ observé, alors $g(\mathbf{x})$ correspond à l'enregistrement que l'on aurait observé si cette erreur ne s'était pas produite. De façon un peu plus formelle, on présume ici que les erreurs qui surviennent dans les données peuvent être modélisées par un « processus de génération d'erreurs » stochastique $\mathcal{E}$, et que chaque opération de vérification joue le rôle de « correcteur » d'une erreur particulière qui peut se produire dans $\mathcal{E}$ (voir la remarque n° 4 à la section suivante).

Si l'opération de vérification g contient des paramètres libres, l'enregistrement $g(\mathbf{x})$ pourrait ne pas être déterminé de façon unique même lorsque les restrictions (2.1) et (3.2) sont prises en compte. Dans ce cas, il faut « imputer » des valeurs pour les paramètres libres de l'opération de vérification, ce qui signifie que certaines des variables de $\mathbf{x}$ sont imputées au moyen de la transformation affine donnée par (3.1). Comme pour la vérification traditionnelle reposant sur le paradigme de Fellegi-Holt, la recherche des « imputations » appropriées pour les paramètres libres n'est pas considérée ici comme faisant partie du problème de localisation des erreurs. En revanche, si g ne contient aucun paramètre libre, les valeurs imputées dans $g(\mathbf{x})$ découlent directement de l'opération de vérification elle-même et la distinction entre la localisation des erreurs et l'imputation devient floue.

Pour n'importe quelle application particulière, seul un petit sous-ensemble d'opérations de vérification possibles de la forme donnée en (3.1) pourrait être interprété de façon considérablement significative, au sens où l'on sait que les types d'erreur associés peuvent se produire. Dans ce qui suit, on présume qu'un

ensemble fini d'opérations de vérification spécifiques de la forme donnée en (3.1) a été déterminé comme étant pertinent pour une application particulière. Cet ensemble correspond aux *opérations de vérification autorisées* pour cette application. Des suggestions sur la façon de bâtir cet ensemble sont présentées à la section 8.

4 Un problème généralisé de localisation des erreurs

Soit $\mathcal{G}$ un ensemble fini d'opérations de vérification autorisées pour une application donnée de vérification automatique. De façon informelle, il est proposé ici de généraliser le problème de localisation des erreurs de Fellegi et Holt (1976) en remplaçant l'énoncé « le plus petit sous-ensemble de variables qui peuvent être imputées pour assurer la cohérence de l'enregistrement » par « la plus courte séquence d'opérations de vérification autorisées qui peut être appliquée pour assurer la cohérence de l'enregistrement ». Pour donner une définition formelle de ce problème généralisé de localisation des erreurs, il faut introduire de nouveaux éléments de notation et quelques concepts.

Supposons une séquence de points $\mathbf{x} = \mathbf{x}_0, \mathbf{x}_1, \ldots, \mathbf{x}_t = \mathbf{y}$ appartenant à $\mathbb{R}^p$. Un *chemin* allant de $\mathbf{x}$ à $\mathbf{y}$ s'entend d'une séquence d'opérations de vérification *distinctes* $g_1, \ldots, g_t \in \mathcal{G}$ de sorte que $\mathbf{x}_n = g_n(\mathbf{x}_{n-1})$ pour tout $n \in \{1, \ldots, t\}$. (Remarque : Si g_n contient des paramètres libres, il faut interpréter cette égalité comme « il existe des valeurs des paramètres réalisables faisant en sorte que g_n établisse une correspondance entre $\mathbf{x}_{n-1}$ et $\mathbf{x}_n$ ».) Un chemin est désigné par $P = [g_1, \ldots, g_t]$. L'ensemble de tous les chemins possibles allant de $\mathbf{x}$ à $\mathbf{y}$ est désigné par $\mathcal{P}(\mathbf{x}, \mathbf{y})$. Cet ensemble peut être vide. Plus loin, on utilise $\mathcal{P}(\mathbf{x}; G)$ pour désigner, pour un sous-ensemble donné $G \subseteq \mathcal{G}$, l'ensemble de tous les chemins partant de $\mathbf{x}$ et correspondant aux opérations de vérification de G dans un certain ordre (sans spécifier les paramètres libres); si G contient t éléments, $\mathcal{P}(\mathbf{x}; G)$ contient $t!$ chemins.

Pour chaque opération de vérification $g \in \mathcal{G}$, on peut associer un poids $w_g > 0$ représentant le coût de l'application de l'opération de vérification g. Plus particulièrement, le poids d'une opération FH doit être égal au poids de confiance de la variable qu'elle impute. La *longueur* d'un chemin $P = [g_1, \ldots, g_t]$ peut donc être définie comme la somme des poids des opérations de vérification qui la constitue : $\ell(P) = \sum_{n=1}^{t} w_{g_n}$, où, par convention, le chemin vide a une longueur de zéro. La *distance* de $\mathbf{x}$ à $\mathbf{y}$ s'entend de la longueur du chemin le plus court reliant $\mathbf{x}$ à $\mathbf{y}$:

$$d(\mathbf{x}, \mathbf{y}) = \begin{cases} \min\{\ell(P) | P \in \mathcal{P}(\mathbf{x}, \mathbf{y})\} & \text{si } \mathcal{P}(\mathbf{x}, \mathbf{y}) \neq \varnothing, \\ \infty & \text{sinon.} \end{cases}$$

En règle générale, $d(\mathbf{x}, \mathbf{y})$ satisfait aux axiomes types d'un espace métrique, *sauf* qu'elle ne doit pas nécessairement être symétrique pour $\mathbf{x}$ et $\mathbf{y}$; il s'agit plutôt de ce qu'on appelle un *espace quasimétrique* (Scholtus 2014). En conséquence, $d(\mathbf{x}, \mathbf{y})$ représente « la distance de $\mathbf{x}$ à $\mathbf{y}$ » plutôt que « la distance entre $\mathbf{x}$ et $\mathbf{y}$ ».

La distance de $\mathbf{x}$ à n'importe quel sous-ensemble fermé non vide $D \subseteq \mathbb{R}^p$ s'entend de la distance jusqu'au plus proche $\mathbf{y} \in D$: $d(\mathbf{x}, D) = \min\{d(\mathbf{x}, \mathbf{y}) | \mathbf{y} \in D\}$. Aux fins de la localisation des erreurs, le

sous-ensemble fermé non vide de $\mathbb{R}^p$ présentant un intérêt particulier est l'ensemble D_0 de tous les points qui satisfont à (2.1).

On peut maintenant formuler le problème généralisé de localisation des erreurs.

Problème. Supposons un ensemble donné d'enregistrements cohérents D_0, un ensemble donné d'opérations de vérification autorisées $\mathcal{G}$ et un enregistrement donné $\mathbf{x}$. Si $d(\mathbf{x}, D_0) = \infty$, alors le problème de localisation des erreurs pour $\mathbf{x}$ est irréalisable. Sinon, n'importe quel chemin le plus court menant à un enregistrement $\mathbf{y} \in D_0$ de sorte que $d(\mathbf{x}, \mathbf{y}) < \infty$ correspond à une *solution réalisable* au problème de localisation des erreurs pour $\mathbf{x}$. Une solution réalisable est dite *optimale* si elle produit un enregistrement $\mathbf{x}^* \in D_0$ faisant en sorte que

$$d(\mathbf{x}, \mathbf{x}^*) = d(\mathbf{x}, D_0). \tag{4.1}$$

Le problème généralisé de la localisation des erreurs consiste donc officiellement à trouver un chemin optimal d'opérations de vérification.

Remarque n° 1. En règle générale, il peut y avoir un très grand nombre d'enregistrements $\mathbf{x}^*$ dans D_0 qui satisfont à (4.1) et qui peuvent être atteints par le même chemin d'opérations de vérification. Pour résoudre le problème de localisation des erreurs, il suffit de trouver un chemin optimal. La construction d'un enregistrement associé $\mathbf{x}^* \in D_0$ peut donc être considérée comme une généralisation du problème d'imputation cohérente (voir la discussion sur l'imputation à la fin de la section 3).

Remarque n° 2. Le problème de localisation des erreurs ci-dessus est irréalisable pour les enregistrements qui ne peuvent être mis en correspondance dans D_0 par aucune combinaison d'opérations de vérification distinctes de $\mathcal{G}$. Pour éviter cette situation, il faut définir un ensemble $\mathcal{G}$ suffisamment vaste pour que $d(\mathbf{x}, D_0) < \infty$ pour tout $\mathbf{x} \in \mathbb{R}^p$. Dans ce qui suit, on présume tacitement que $\mathcal{G}$ a cette propriété. Un moyen simple d'y arriver, mais pas nécessairement le seul, est de s'assurer que $\mathcal{G}$ contient au moins toutes les opérations FH. Cela est suffisant parce que deux points quelconques de $\mathbb{R}^p$ peuvent toujours être reliés par un chemin qui concatène les opérations FH associées aux coordonnées qui les différencient.

Remarque n° 3. Il n'est pas difficile de voir que le problème de localisation des erreurs ci-dessus réduit le problème original de Fellegi et Holt (1976) au cas particulier où $\mathcal{G}$ contient seulement les opérations FH.

Remarque n° 4. Comme pour le problème de localisation des erreurs original fondé sur le paradigme de Fellegi-Holt, on peut montrer que, en vertu de certaines hypothèses, la minimisation de $d(\mathbf{x}, \mathbf{y})$ pour tout $\mathbf{y} \in D_0$ pour un enregistrement observé donné $\mathbf{x}$ équivaut approximativement à la maximisation de la vraisemblance de l'enregistrement exempt d'erreur non observé associé. L'argument est sensiblement le même que celui de Kruskal (1983, pages 38-39) pour la distance dite de Levenshtein dans le contexte de l'appariement approximatif de chaînes. Pour cela, il faut d'abord que toutes les vérifications (2.1) soient des vérifications avec rejet, c'est-à-dire des vérifications auxquelles seules les valeurs erronées échouent. De plus, il faut présumer que le « processus de génération d'erreurs » stochastique $\mathcal{E}$ dont il est question à la section 3 a les propriétés suivantes :

- Il existe une correspondance biunivoque entre l'ensemble des erreurs qui peuvent se produire en vertu de $\mathcal{E}$ et l'ensemble des opérations de vérification autorisées $\mathcal{G}$ qui corrigent les erreurs.
- Les erreurs dans $\mathcal{E}$ se produisent de façon indépendante les unes des autres.
- L'erreur correspondant à l'opération g se produit selon une probabilité connue p_g.

Enfin, comme pour (2.3), les poids w_g doivent être choisis comme suit :

$$w_g = -\log\left(\frac{p_g}{1-p_g}\right). \tag{4.2}$$

En vertu de ces hypothèses, Scholtus (2014) a adapté l'argument de Kruskal (1983) pour montrer que la solution optimale au problème de localisation des erreurs (4.1) peut être justifiée comme un estimateur approximatif du maximum de vraisemblance. [Remarque : Le calcul présenté par Scholtus (2014) suppose en outre que $p_g \ll 1$ pour toutes les valeurs, auquel cas $w_g \approx -\log p_g$. Cette hypothèse n'est pas nécessaire; voir Liepins (1980).]

5 Vérifications implicites pour les opérations de vérification générales

Dans la présente section, on dérive un résultat qui détermine si un chemin donné d'opérations de vérification de la forme (3.1) peut être utilisé pour assurer la cohérence d'un enregistrement particulier avec un système de règles de vérification donné (c'est-à-dire s'il correspond à une solution réalisable au problème de localisation des erreurs). Pour obtenir ce résultat, on fait appel à la technique d'élimination FM présentée à la section 2.

Soit $\mathbf{x}$ un enregistrement donné et $\mathbf{y}_t$ un enregistrement quelconque obtenu en appliquant séquentiellement les opérations de vérification $g_1,\ldots,g_t$ à $\mathbf{x}$:

$$\mathbf{y}_t = g_t \circ g_{t-1} \circ \cdots \circ g_1(\mathbf{x}). \tag{5.1}$$

Écrivons $g_n(\mathbf{x}) = \mathbf{T}_n\mathbf{x} + \mathbf{S}_n\boldsymbol{\alpha}_n + \mathbf{c}_n$, pour $n \in \{1,\ldots,t\}$. De l'équation (5.1) ci-dessus, il découle par induction que :

$$\begin{aligned}\mathbf{y}_1 &= \mathbf{T}_1\mathbf{x} + \mathbf{S}_1\boldsymbol{\alpha}_1 + \mathbf{c}_1,\\ \mathbf{y}_2 &= \mathbf{T}_2\mathbf{T}_1\mathbf{x} + \mathbf{S}_2\boldsymbol{\alpha}_2 + \mathbf{c}_2 + \mathbf{T}_2(\mathbf{S}_1\boldsymbol{\alpha}_1 + \mathbf{c}_1),\end{aligned}$$

et, en général,

$$\mathbf{y}_t = \mathbf{T}_t\cdots\mathbf{T}_1\mathbf{x} + \mathbf{S}_t\boldsymbol{\alpha}_t + \mathbf{c}_t + \sum_{n=2}^{t}\mathbf{T}_t\cdots\mathbf{T}_n(\mathbf{S}_{n-1}\boldsymbol{\alpha}_{n-1} + \mathbf{c}_{n-1}), \tag{5.2}$$

où la somme pour tous les n est nulle lorsque $t=1$. En outre, tous les termes comprenant $\mathbf{S}_n\boldsymbol{\alpha}_n$ disparaissent lorsque g_n ne contient aucun paramètre libre.

Le chemin des opérations de vérification $P = [g_1,\ldots,g_t]$ peut être appliqué à $\mathbf{x}$ pour obtenir un enregistrement cohérent avec les règles de vérification énoncées en (2.1) si et seulement s'il existe une valeur $\mathbf{y}_t$ de la forme (5.2) qui satisfait $\mathbf{A}\mathbf{y}_t + \mathbf{b} \odot \mathbf{0}$ et toutes les restrictions supplémentaires pertinentes de la forme (3.2) s'appliquant à $\boldsymbol{\alpha}_1,\ldots,\boldsymbol{\alpha}_t$. À l'aide de (5.2), on peut écrire $\mathbf{A}\mathbf{y}_t + \mathbf{b} \odot \mathbf{0}$ comme suit :

$$(\mathbf{A}\mathbf{T}_t\cdots\mathbf{T}_1)\mathbf{x} + (\mathbf{A}\mathbf{S}_t)\boldsymbol{\alpha}_t + \sum_{n=2}^{t}(\mathbf{A}\mathbf{T}_t\cdots\mathbf{T}_n\mathbf{S}_{n-1})\boldsymbol{\alpha}_{n-1} + \mathbf{b}_t \odot \mathbf{0}, \tag{5.3}$$

où $\mathbf{b}_t = \mathbf{b} + \mathbf{A}\mathbf{c}_t + \sum_{n=2}^{t}\mathbf{A}\mathbf{T}_t\cdots\mathbf{T}_n\mathbf{c}_{n-1}$ correspond à un vecteur de constantes.

Fait intéressant, (5.3) et les restrictions supplémentaires possibles de la forme (3.2) constituent un système linéaire de la forme (2.1) appliqué à l'enregistrement élargi $(\mathbf{x}', \boldsymbol{\alpha}_1', \ldots, \boldsymbol{\alpha}_t')'$. En conséquence, l'élimination FM peut servir à retirer tous les paramètres libres du système; on obtient alors un système de contraintes implicites pour $\mathbf{x}$. De plus, l'application répétée de cette propriété fondamentale de l'élimination FM établit que $\mathbf{x}$ satisfait au système de règles de vérification implicites si et seulement s'il existe des valeurs pour les paramètres $\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_t$ qui, avec $\mathbf{x}$, satisfont (5.3) et (3.2). Il s'ensuit que le chemin des opérations de vérification $P = [g_1, \ldots, g_t]$ peut déboucher sur un enregistrement cohérent pour $\mathbf{x}$ si et seulement si $\mathbf{x}$ satisfait le système de règles de vérification implicites obtenues par l'élimination de $\boldsymbol{\alpha}_1, \ldots, \boldsymbol{\alpha}_t$ de (5.3) et, s'il y a lieu, les restrictions supplémentaires de la forme (3.2).

Exemple. Supposons les règles de vérification suivantes dans x_1 et x_2 :

$$x_1 \geq 0, \tag{5.4}$$

$$x_2 \geq 0, \tag{5.5}$$

$$x_1 + x_2 \leq 5. \tag{5.6}$$

Soit g l'opération de vérification qui transfère un nombre d'au plus quatre unités entre x_1 et x_2, dans l'une ou l'autre direction : $g((x_1, x_2)') = (x_1 + \alpha, x_2 - \alpha)'$ où $-4 \leq \alpha \leq 4$. Pour cette opération de vérification unique, le système des règles de vérification transformées (5.3) est le suivant :

$$x_1 + \alpha \geq 0, \tag{5.7}$$

$$x_2 - \alpha \geq 0, \tag{5.8}$$

$$x_1 + x_2 \leq 5. \tag{5.9}$$

On ajoute aussi les restrictions suivantes à la forme (3.2) pour α :

$$\alpha \geq -4, \tag{5.10}$$

$$\alpha \leq 4. \tag{5.11}$$

On obtient cinq contraintes linéaires (5.7)−(5.11) pour x_1, x_2 et α, desquelles α peut être retirée par élimination FM pour obtenir :

$$x_1 \geq -4, \tag{5.12}$$

$$x_2 \geq -4, \tag{5.13}$$

$$x_1 + x_2 \geq 0, \tag{5.14}$$

$$x_1 + x_2 \leq 5. \tag{5.15}$$

En théorie, tout enregistrement $(x_1, x_2)'$ qui satisfait (5.12)−(5.15) peut être rendu cohérent avec les règles de vérification originales (5.4)−(5.6) en transférant un certain nombre d'unités $-4 \leq \alpha \leq 4$ entre x_1 et x_2. L'enregistrement $(x_1, x_2)' = (-2, 3)'$ donné en exemple n'est pas cohérent avec les règles de vérification originales (5.4)−(5.6), mais satisfait (5.12)−(5.15). Cela signifie qu'on peut rendre l'enregistrement cohérent avec les règles de vérification originales en appliquant g. On peut facilement constater que cette affirmation est vraie; n'importe quelle valeur $2 \leq \alpha \leq 3$ fera l'affaire.

Il est intéressant de souligner que, dans le cas particulier où P correspond à l'opération FH unique qui impute x_j, on obtient le système transformé de règles de vérification (5.3) en remplaçant chaque occurrence de x_j des règles de vérification originales par un paramètre non restreint α. L'élimination de α de (5.3)

équivaut dans ce cas à l'élimination directe de x_j des règles de vérification originales. En ce sens, le résultat ci-dessus généralise la propriété fondamentale de l'élimination FM pour les opérations FH à toutes les opérations de vérification de la forme (3.1).

En général, l'ensemble d'enregistrements défini par l'expression (5.2) dépend de l'ordre d'exécution des opérations de vérification. Ainsi, deux chemins composés du même ensemble d'opérations de vérification exécutées dans un ordre différent ne donnent pas nécessairement la même solution au problème de localisation des erreurs. À cet égard, les opérations de vérification générales diffèrent des opérations FH (Scholtus 2014).

6 Algorithme de localisation des erreurs

La section qui suit propose un algorithme relativement simple pour résoudre le problème de localisation des erreurs de la section 4 à l'aide du résultat théorique obtenu à la section 5.

Étape 0.	Soit $\mathbf{x}$ un enregistrement donné et $\mathcal{G}$ un ensemble donné d'opérations de vérification autorisées. Initialiser : $\mathcal{L} := \emptyset$; $\mathcal{B}_0 := \{\emptyset\}$; $W := \infty$; et $t := 1$.
Étape 1.	Déterminer tous les sous-ensembles $G \subseteq \mathcal{G}$ de cardinalité t qui satisfont aux conditions suivantes : 1. Chaque sous-ensemble de $t-1$ éléments de G appartient à $\mathcal{B}_{t-1}$. 2. Il est vérifié que $\sum_{g \in G} w_g \leq W$.
Étape 2.	Pour chaque G obtenu à l'étape 1, construire $\mathcal{P}(\mathbf{x}; G)$ et, pour chaque chemin $P \in \mathcal{P}(\mathbf{x}; G)$, déterminer s'il est possible d'obtenir un enregistrement cohérent. Dans l'affirmative : • si $\ell(P) < W$, définir $\mathcal{L} := \{P\}$ et $W := \ell(P)$; • si $\ell(P) = W$, définir $\mathcal{L} := \mathcal{L} \cup \{P\}$. Si *aucun* des chemins $P \in \mathcal{P}(\mathbf{x}; G)$ ne permet d'obtenir un enregistrement cohérent, ajouter G à $\mathcal{B}_t$.
Étape 3.	Si $t < R$ et $\mathcal{B}_t \neq \emptyset$, définir $t := t + 1$ et revenir à l'étape 1.

Figure 6.1 Algorithme pour trouver tous les chemins optimaux des opérations de vérification relatives au problème (4.1).

Dans le cadre des applications pratiques de localisation des erreurs aux fins de la statistique officielle, il arrive souvent que les enregistrements contiennent plus de 100 variables. Pour formuler un problème dont les calculs sont réalisables, les applications actuelles de vérification automatique fondée sur le paradigme de Fellegi-Holt définissent généralement une borne supérieure M au nombre de variables qui peuvent être imputées dans un même enregistrement (par exemple $M = 12$ ou $M = 15$). de Waal et Coutinho (2005) soutiennent que l'introduction d'une telle borne supérieure est raisonnable parce qu'un enregistrement qui exige plus de quinze imputations, par exemple, ne devrait pas être considéré admissible à la vérification automatique de toute manière. Selon cette convention, on peut aussi fixer une borne supérieure R au nombre d'opérations de vérification distinctes qui peuvent être appliquées à un même enregistrement. Même en appliquant cette restriction supplémentaire, l'espace de recherche des solutions possibles à (4.1) est

généralement trop vaste dans la pratique pour trouver une solution optimale au moyen d'une recherche exhaustive.

La figure 6.1 présente un résumé de l'algorithme de localisation des erreurs proposé. La formulation de base est inspirée de l'*algorithme apriori* établi par Agrawal et Srikant (1994) pour l'exploration de données. L'exécution de l'algorithme donne un ensemble $\mathcal{L}$ contenant tous les chemins d'opérations de vérification autorisées qui correspondent à une solution optimale à (4.1), ainsi que la longueur du chemin optimal W. [Remarque : Un problème de localisation des erreurs peut avoir plusieurs solutions optimales, et il peut être utile de les trouver toutes (Giles 1988; de Waal et coll. 2011, pages 66-67).]

Après la définition initiale à l'étape 0, l'algorithme passe par les étapes 1, 2 et 3 au plus R fois. À l'étape 1 de l'algorithme, l'espace de recherche est limité par ce qui suit : si G comprend un sous-ensemble approprié $H \subset G$ pour lequel $\mathcal{P}(\mathbf{x}; H)$ contient un chemin menant à un enregistrement cohérent, alors $\mathcal{P}(\mathbf{x}; G)$ ne peut contenir que des solutions sous-optimales. Ainsi, tout ensemble G contenant un tel sous-ensemble peut être ignoré par l'algorithme. De même, G peut aussi être ignoré lorsque le poids total des opérations de vérification contenues dans G est supérieur à la longueur du chemin de la meilleure solution réalisable déjà trouvée.

Durant la t^{e} itération, le nombre de sous-ensembles G obtenus à l'étape 1 de l'algorithme est égal à $\binom{N}{t}$. Pour chacun de ces sous-ensembles, les conditions de l'étape 1 doivent être vérifiées. Si un sous-ensemble de G réussit les vérifications, à l'étape 2 tous les chemins $t!$ de $\mathcal{P}(\mathbf{x}; G)$ sont évalués selon la théorie exposée à la section 5. L'algorithme apriori repose sur le principe suivant : à mesure que t augmente, la majorité des sous-ensembles échouent aux vérifications de la première étape, de sorte que le nombre total de calculs à effectuer demeure limité. Dans le contexte de l'exploration de données, ce comportement souhaitable a effectivement été observé dans les faits. La question de savoir s'il se produit aussi dans le contexte de la localisation des erreurs reste à déterminer.

Il est possible d'améliorer l'algorithme si l'on observe que l'ordre dans lequel les opérations de vérification sont exécutées n'a pas toujours d'importance. Il arrive que deux chemins de $\mathcal{P}(\mathbf{x}; G)$ soient *équivalents*, c'est-à-dire qu'un enregistrement qu'on peut atteindre à partir de $\mathbf{x}$ en empruntant le premier chemin peut aussi être atteint par le second chemin, et vice versa. Cette propriété définit une relation d'équivalence dans $\mathcal{P}(\mathbf{x}; G)$. Soit $\tilde{\mathcal{P}}(\mathbf{x}; G)$ un ensemble contenant un représentant de chaque catégorie d'équivalence de $\mathcal{P}(\mathbf{x}; G)$ en vertu de cette relation. Il est clair que l'algorithme de la figure 6.1 demeure correct si à l'étape 2 la recherche est limitée à $\tilde{\mathcal{P}}(\mathbf{x}; G)$ plutôt qu'à $\mathcal{P}(\mathbf{x}; G)$. Scholtus (2014) présente une méthode simple pour construire $\tilde{\mathcal{P}}(\mathbf{x}; G)$ à partir de $\mathcal{P}(\mathbf{x}; G)$.

Un exemple détaillé illustrant l'algorithme ci-dessus est présenté dans Scholtus (2014).

7 Étude par simulations

Pour mettre à l'essai l'utilité potentielle de la nouvelle méthode de localisation des erreurs, on a mené une étude par simulations de petite envergure dans l'environnement R pour calcul statistique (R Development Core Team 2015). Une mise en œuvre prototype de l'algorithme de la figure 6.1 a été créée dans R. Dans le cadre de cet exercice, on a largement utilisé la fonctionnalité de vérification automatique

fondée sur le paradigme de Fellegi-Holt du progiciel `editrules` (van der Loo et de Jonge 2012; de Jonge et van der Loo 2014). Le programme n'était pas optimisé pour assurer l'efficacité du calcul, mais il s'est révélé suffisamment rapide pour les problèmes de localisation des erreurs d'envergure relativement petite de l'étude par simulations. (Remarque : L'auteur peut fournir le code R utilisé sur demande.)

L'étude par simulations a été réalisée à l'aide d'enregistrements contenant cinq variables numériques qui devaient satisfaire les neuf règles de vérification linéaires suivantes :

$$\begin{aligned} x_1 + x_2 &= x_3, \\ x_3 - x_4 &= x_5, \\ x_j &\geq 0, \qquad j \in \{1,2,3,4\}, \\ x_1 &\geq x_2, \\ x_5 &\geq -0{,}1x_3, \\ x_5 &\leq 0{,}5x_3. \end{aligned}$$

On trouve généralement ce genre de règles de vérification pour les SSE, dans le cadre d'un ensemble de règles de vérification beaucoup plus vaste (Scholtus 2014).

Un ensemble aléatoire de données exempt d'erreurs contenant 2 000 enregistrements a été bâti à partir d'une distribution normale multivariée (à l'aide du progiciel `mvtnorm`) selon les paramètres suivants :

$$\boldsymbol{\mu} = \begin{pmatrix} 500 \\ 250 \\ 750 \\ 600 \\ 150 \end{pmatrix} \quad \text{et} \quad \boldsymbol{\Sigma} = \begin{pmatrix} 10\,000 & -1\,250 & 8\,750 & 7\,500 & 1\,250 \\ -1\,250 & 5\,000 & 3\,750 & 4\,000 & -250 \\ 8\,750 & 3\,750 & 12\,500 & 11\,500 & 1\,000 \\ 7\,500 & 4\,000 & 11\,500 & 11\,750 & -250 \\ 1\,250 & -250 & 1\,000 & -250 & 1\,250 \end{pmatrix}.$$

Seuls les enregistrements satisfaisant à toutes les règles de vérification susmentionnées ont été inclus dans l'ensemble de données. Soulignons que $\boldsymbol{\Sigma}$ est une matrice singulière de covariances comprenant les deux règles de vérification fondées sur une égalité. Techniquement, les données obtenues suivent une distribution normale singulière multivariée tronquée; voir de Waal et coll. (2011, pages 318ff) ou Tempelman (2007).

Les neuf opérations de vérification autorisées retenues dans le cadre de l'étude sont présentées au tableau 7.1. Soulignons que les cinq premières lignes correspondent aux opérations FH pour cet ensemble de données. Comme il est précisé dans le tableau, chaque opération de vérification est associée à un type d'erreur. Un ensemble de données synthétiques à vérifier a été créé par l'ajout aléatoire d'erreurs de ces types à l'ensemble de données exempt d'erreur susmentionné. La probabilité de chaque type d'erreur est indiquée dans la quatrième colonne du tableau 7.1. Le poids « idéal » qui y est associé selon (4.2) est précisé dans la dernière colonne.

Pour restreindre l'ampleur des calculs à effectuer, seuls les enregistrements exigeant trois opérations de vérification ou moins ont été pris en compte. Les enregistrements ne contenant aucune erreur ont aussi été retirés. Il restait donc 1 025 enregistrements à vérifier, chacun contenant une, deux ou trois des erreurs énumérées au tableau 7.1.

Tableau 7.1
Opérations de vérification autorisées aux fins de l'étude par simulations

nom	opération	type d'erreur associé	p_g	w_g
FH1	imputer x_1	valeur erronée de x_1	0,10	2,20
FH2	imputer x_2	valeur erronée de x_2	0,08	2,44
FH3	imputer x_3	valeur erronée de x_3	0,06	2,75
FH4	imputer x_4	valeur erronée de x_4	0,04	3,18
FH5	imputer x_5	valeur erronée de x_5	0,02	3,89
IC34	intervertir x_3 et x_4	valeurs réelles de x_3 et x_4 interverties	0,07	2,59
TF21	transférer une partie de x_2 à x_1	partie de la valeur réelle de x_1 déclarée comme faisant partie de x_2	0,09	2,31
CS4	changer le signe de x_4	erreur de signe dans x_4	0,11	2,09
CS5	changer le signe de x_5	erreur de signe dans x_5	0,13	1,90

Plusieurs méthodes de localisation des erreurs ont été appliquées à l'ensemble de données. On a tout d'abord utilisé la méthode de localisation des erreurs fondée sur le paradigme de Fellegi-Holt (c'est-à-dire à l'aide des opérations de vérification FH1–FH5 uniquement) et sur le nouveau paradigme (c'est-à-dire à l'aide de toutes les opérations de vérification du tableau 7.1). Les deux méthodes ont été mises à l'essai une fois à l'aide des poids « idéaux » indiqués dans le tableau 7.1 et une fois à l'aide de poids tous fixés à 1 (« aucun poids »). Ce dernier cas simule une situation où les opérations de vérification pertinentes sont connues, mais pas leurs fréquences respectives. Enfin, pour vérifier la robustesse de la nouvelle méthode de localisation des erreurs en cas de manque d'information à propos des opérations de vérification pertinentes, la méthode a aussi été appliquée en retirant l'une des opérations non-FH du tableau 7.1 de l'ensemble des opérations de vérification autorisées.

La qualité de la localisation des erreurs a été évaluée de deux façons. Tout d'abord, on a évalué dans quelle mesure les chemins optimaux des opérations de vérification trouvés par l'algorithme correspondaient à la distribution réelle des erreurs, en utilisant le tableau de contingences ci-dessous pour toutes les $1\,025 \times 9 = 9\,225$ combinaisons possibles des enregistrements et des opérations de vérification :

Tableau 7.2
Tableau de contingences des erreurs et des opérations de vérification suggérées par l'algorithme

	opération de vérification suggérée	opération de vérification non suggérée
l'erreur associée s'est produite	VP	FN
l'erreur associée ne s'est pas produite	FP	VN

À partir de ce tableau, on a calculé des indicateurs mesurant la proportion de faux négatifs (FN), de faux positifs (FP) et de l'ensemble des mauvaises décisions, respectivement :

$$\alpha = \frac{FN}{VP + FN}; \quad \beta = \frac{FP}{FP + VN}; \quad \delta = \frac{FN + FP}{VP + FN + FP + VN}$$

Des indicateurs similaires sont présentés dans de Waal et coll. (2011, pages 410-411). On a aussi calculé $\overline{\rho} = 1 - \rho$, où ρ correspond à la fraction des enregistrements de l'ensemble de données pour lesquels l'algorithme de localisation des erreurs a trouvé exactement la bonne solution. Un bon algorithme de localisation des erreurs devrait donner des notes faibles pour les quatre indicateurs.

Il importe de souligner que les indicateurs de qualité ci-dessus désavantagent la méthode originale de Fellegi-Holt, qui ne fait pas appel à toutes les opérations de vérification énumérées au tableau 7.1. On a donc aussi calculé un deuxième ensemble d'indicateurs de qualité α, β, δ et $\overline{\rho}$ portant sur les valeurs erronées plutôt que sur les opérations de vérification. Dans ce cas, α mesure la proportion des valeurs de l'ensemble de données comportant des erreurs, mais non modifiées par la solution optimale au problème de localisation des erreurs, et de même pour les autres mesures.

Le tableau 7.3 présente les résultats de l'étude par simulations pour les deux ensembles d'indicateurs de qualité. Dans les deux cas, on constate une amélioration notable de la qualité des résultats de la localisation des erreurs de la méthode faisant appel à toutes les opérations de vérification, comparativement à la méthode utilisant uniquement les opérations FH. En outre, le fait d'omettre une seule opération de vérification pertinente de l'ensemble des opérations de vérification autorisées compromettait la qualité de la localisation des erreurs. Dans certains cas, cet effet était assez important – particulièrement en ce qui concerne les opérations de vérification utilisées –, mais les résultats de la nouvelle méthode de localisation des erreurs demeurent considérablement supérieurs à ceux de la méthode de Fellegi-Holt. Contrairement aux attentes, le fait de ne pas utiliser des poids de confiance différents a contribué à améliorer légèrement la qualité des résultats de la localisation des erreurs pour cet ensemble de données selon la méthode de Fellegi-Holt (pour les deux ensembles d'indicateurs) et aussi, dans une certaine mesure, selon la nouvelle méthode (second ensemble d'indicateurs seulement). Enfin, il semble que l'utilisation de toutes les opérations de vérification ait contribué à accroître le temps de calcul nécessaire par rapport à l'utilisation des opérations FH uniquement, mais pas de façon spectaculaire.

Tableau 7.3
Qualité de la localisation des erreurs en fonction des opérations de vérification utilisées et des valeurs erronées recensées; temps de calcul requis

	indicateurs de qualité (opérations de vérification)				indicateurs de qualité (valeurs erronées)				
méthode	α	β	δ	$\overline{\rho}$	α	β	δ	$\overline{\rho}$	**temps***
Fellegi-Holt (avec poids)	74 %	12 %	23 %	80 %	19 %	10 %	13 %	32 %	46
Fellegi-Holt (sans poids)	70 %	12 %	21 %	74 %	13 %	8 %	9 %	24 %	33
toutes les opérations (avec poids)	14 %	3 %	5 %	24 %	10 %	5 %	7 %	17 %	98
sauf IC34	29 %	5 %	9 %	35 %	15 %	9 %	11 %	29 %	113
sauf TF21	34 %	5 %	10 %	37 %	10 %	5 %	7 %	18 %	80
sauf CS4	28 %	6 %	9 %	39 %	10 %	5 %	7 %	17 %	80
sauf CS5	35 %	7 %	10 %	47 %	11 %	6 %	7 %	18 %	82
toutes les opérations (sans poids)	27 %	5 %	8 %	36 %	6 %	4 %	5 %	13 %	99

* Temps total de calcul (en secondes) sur un ordinateur portatif doté d'un processeur à 2,5 GHz sous Windows 7.

8 Conclusion

Le présent article propose une nouvelle formulation du problème de localisation des erreurs dans le contexte de la vérification automatique. On suggère de trouver le nombre minimal (pondéré) d'opérations

de vérification nécessaires pour assurer la cohérence d'un enregistrement observé avec les règles de vérification. Le nouveau problème de localisation des erreurs peut être considéré comme une généralisation du problème proposé dans l'article fondamental de Fellegi et Holt (1976), parce que l'opération qui impute une nouvelle valeur à une seule variable à la fois constitue un important cas particulier d'une opération de vérification.

L'objectif principal était de mettre au point la théorie mathématique sur laquelle repose le nouveau problème de localisation des erreurs. Il ressort que l'élimination FM, une technique utilisée par le passé pour résoudre le problème de localisation des erreurs fondé sur le paradigme de Fellegi-Holt, peut aussi être appliquée dans le contexte du nouveau problème (voir la section 5). Néanmoins, la résolution du problème de localisation des erreurs demeure une tâche difficile du point de vue du calcul, du moins pour les quantités de variables, de règles de vérification et d'opérations de vérification qui entrent en jeu dans les applications pratiques au sein des instituts de statistique. Un algorithme de localisation des erreurs possible est proposé à la section 6. D'autres algorithmes plus efficients pourraient et devraient probablement être mis au point. Comme pour l'élimination FM, il pourrait être possible d'adapter d'autres idées mises en œuvre pour résoudre le problème fondé sur le paradigme de Fellegi-Holt au problème général étudié ici.

Le présent article ne porte que sur les données numériques et les règles de vérification linéaires. Le paradigme original de Fellegi-Holt a aussi été appliqué à des données catégoriques et mixtes. Plusieurs auteurs, dont Bruni (2004) et de Jonge et van der Loo (2014), ont montré qu'une grande catégorie de règles de vérification s'appliquant à des données mixtes peuvent être reformulées en fonction de données numériques et de règles de vérification linéaires, sous réserve de la restriction supplémentaire que certaines variables doivent avoir une valeur entière. En principe, cela signifie que les résultats présentés dans l'article pourraient aussi s'appliquer à des données mixtes. Pour tenir compte du fait que certaines variables ont une valeur entière, on pourrait utiliser l'élargissement de l'élimination FM aux nombres entiers proposé par Pugh (1992); voir aussi de Waal et coll. (2011) pour en savoir davantage à propos de cette technique d'élimination élargie dans le contexte de la localisation des erreurs fondée sur le paradigme de Fellegi-Holt. Il reste à déterminer si les calculs nécessaires en vertu de cette approche sont réalisables.

La remarque n° 4 de la section 4 laisse entrevoir une analogie entre la localisation des erreurs dans les microdonnées statistiques et le domaine de l'appariement approximatif de chaînes. Dans l'appariement approximatif de chaînes, des chaînes de caractères sont comparées en vertu de l'hypothèse qu'elles pourraient avoir été partiellement corrompues (Navarro 2001). Diverses fonctions de distance ont été proposées pour cette tâche. La distance de Hamming, qui correspond au nombre de positions où deux chaînes diffèrent, peut être considérée comme analogue à la fonction cible fondée sur le paradigme de Fellegi-Holt (2.2). Le problème généralisé de localisation des erreurs défini dans le présent article peut quant à lui être considéré comme la contrepartie de l'utilisation de la distance de Levenshtein, ou « distance d'édition », pour l'appariement approximatif de chaînes. Il pourrait être intéressant d'explorer plus avant cette analogie. Plus particulièrement, des algorithmes efficients ont été mis au point pour calculer les distances d'édition entre deux chaînes; il pourrait être possible d'appliquer certaines idées sous-jacentes au problème généralisé de localisation des erreurs.

Le nouvel algorithme de localisation des erreurs a été appliqué avec succès à un petit ensemble synthétique de données (section 7). Globalement, les résultats de l'étude par simulations indiquent que la nouvelle méthode de localisation des erreurs pourrait améliorer considérablement la qualité de la vérification automatique par rapport à la méthode actuellement mise en œuvre. Il faut toutefois disposer

d'information suffisante pour déterminer toutes – ou à tout le moins la majorité – des opérations de vérification pertinentes pour une application particulière. Les gains possibles en termes de qualité de la localisation des erreurs doivent aussi être pondérés dans la pratique par rapport aux exigences supérieures de calcul du problème généralisé de localisation des erreurs.

Les SSE constituent un candidat parfait pour l'application de cette nouvelle méthode dans la pratique. Toutefois, des recherches plus poussées sont nécessaires avant que la méthode puisse être utilisée dans un contexte de production courante. Pour appliquer la méthode dans un contexte particulier, il faut d'abord préciser les opérations de vérification pertinentes. Idéalement, chaque opération de vérification doit correspondre à une combinaison de modifications aux données que les vérificateurs humains considèrent comme une correction d'une erreur en particulier. De plus, un ensemble approprié de poids w_g doit être déterminé pour ces opérations de vérification. Pour ce faire, il faut disposer d'information sur les fréquences relatives des types de modification les plus courants durant la vérification manuelle. Ces deux aspects pourraient être déterminés à partir des données historiques avant et après la vérification manuelle, des instructions de vérification et des autres sources de référence utilisées par les vérificateurs, ainsi qu'à partir d'entrevues avec des vérificateurs et des superviseurs des opérations de vérification.

Sur un plan plus fondamental, il faut encore répondre à la question de la démarcation entre les méthodes de correction déductives et la vérification automatique en vertu du nouveau problème de localisation des erreurs. En principe, bon nombre de types d'erreur connus pourraient être résolus soit par des règles de correction automatique, soit par la localisation des erreurs au moyen d'opérations de vérification. Chaque méthode présente ses propres avantages et inconvénients (Scholtus 2014). Il est probable qu'un compromis entre les deux donnera les meilleurs résultats, certaines erreurs étant traitées de façon déductive et d'autres, au moyen d'opérations de vérification. La meilleure façon d'établir un tel compromis dans la pratique demeure toutefois difficile à déterminer.

En fin de compte, la nouvelle méthode proposée dans l'article vise à accroître l'utilité de la vérification automatique dans la pratique. Les résultats obtenus à ce jour sont prometteurs.

Remerciements

Les opinions exprimées dans le présent article sont celles de l'auteur et ne reflètent pas forcément les politiques de *Statistics Netherlands*. L'auteur tient à remercier Jeroen Pannekoek, Ton de Waal et Mark van der Loo pour leurs commentaires à propos des premières versions de l'article, ainsi que le rédacteur en chef adjoint et deux évaluateurs anonymes.

Annexe

Élimination de Fourier-Motzkin

Soit un système de contraintes linéaires (2.1) et x_f, la variable à éliminer. Supposons d'abord que x_f ne participe qu'à des inégalités. Pour faciliter l'explication, supposons que les règles de vérification sont normalisées de sorte que toutes les inégalités utilisent l'opérateur $\geq$. La méthode d'élimination FM considère toutes les paires (r,s) d'inégalités pour lesquelles les coefficients de x_f ont des signes opposés, c'est-à-dire $a_{rf}a_{sf} < 0.$ Supposons, sans perte de généralité, que $a_{rf} < 0$ et $a_{sf} > 0.$ À partir de la paire

originale de règles de vérification, on dérive la contrainte implicite suivante :

$$\sum_{j=1}^{p} a_j^* x_j + b^* \geq 0, \tag{A.1}$$

où $a_j^* = a_{sf} a_{rj} - a_{rf} a_{sj}$ et $b^* = a_{sf} b_r - a_{rf} b_s$. Soulignons que $a_f^* = 0,$ de sorte que x_f ne participe pas à (A.1). Une inégalité de la forme (A.1) est dérivée de chacune des paires (r, s) susmentionnées. La totalité du système de contraintes résultant de l'élimination FM est maintenant composée de ces contraintes dérivées, ainsi que de toutes les contraintes originales dans lesquelles x_f n'intervient pas.

S'il y a des égalités linéaires où intervient x_f, on pourrait appliquer la technique ci-dessus après avoir remplacé chaque égalité linéaire par deux inégalités linéaires équivalentes. de Waal et Quere (2003) ont proposé une autre solution plus efficiente pour ce cas. Supposons que la r^{e} contrainte en (2.1) soit une égalité dans laquelle intervient x_f. On peut réécrire cette contrainte comme suit :

$$x_f = \frac{-1}{a_{rf}} \left(b_r + \sum_{j \neq f} a_{rj} x_j \right). \tag{A.2}$$

En remplaçant x_f par le terme de droite de l'équation (A.2) pour toutes les autres contraintes, on obtient de nouveau un système implicite de contraintes dans lesquelles x_f n'intervient pas et qui peuvent être réécrites comme en (2.1).

Pour consulter une preuve que l'élimination FM possède la propriété fondamentale énoncée à la section 2, voir entre autres de Waal et coll. (2011, pages 69-70).

Bibliographie

Agrawal, R., et Srikant, R. (1994). *Fast Algorithms for Mining Association Rules*. Rapport technique, IBM Almaden Research Center, San José, Californie.

Bruni, R. (2004). Discrete models for data imputation. *Discrete Applied Mathematics*, 144, 59-69.

Chen, B., Thibaudeau, Y. et Winkler, W.E. (2003). *A Comparison Study of ACS If-Then-Else, NIM, DISCRETE Edit and Imputation Systems Using ACS Data*. Document de travail n° 7, UN/ECE Work Session on Statistical Data Editing, Madrid.

de Jonge, E., et van der Loo, M. (2014). *Error Localization as a Mixed Integer Problem with the Editrules Package*. Document de discussion 2014-07, Statistics Netherlands, La Haye. Disponible au : http://www.cbs.nl.

de Waal, T. (2003). Résolution du problème de localisation des erreurs par la génération de sommets. *Techniques d'enquête*, 29, 1, 81-90.

de Waal, T., et Coutinho, W. (2005). Automatic editing for business surveys: An assessment for selected algorithms. *Revue Internationale de Statistique*, 73, 73-102.

de Waal, T., et Quere, R. (2003). A fast and simple algorithm for automatic editing of mixed data. *Journal of Official Statistics*, 19, 383-402.

de Waal, T., Pannekoek, J. et Scholtus, S. (2011). *Handbook of Statistical Data Editing and Imputation*. Hoboken, New Jersey : John Wiley & Sons, Inc.

EDIMBUS (2007). *Recommended Practices for Editing and Imputation in Cross-Sectional Business Surveys*. Manuel préparé par ISTAT, Statistics Netherlands, et SFSO.

Fellegi, I.P., et Holt, D. (1976). A systematic approach to automatic edit and imputation. *Journal of the American Statistical Association*, 71, 17-35.

Garfinkel, R.S., Kunnathur, A.S. et Liepins, G.E. (1988). Error localization for erroneous data: Continuous data, linear constraints. *SIAM Journal on Scientific and Statistical Computing*, 9, 922-931.

Ghosh-Dastidar, B., et Schafer, J.L. (2006). Outlier detection and editing procedures for continuous multivariate data. *Journal of Official Statistics*, 22, 487-506.

Giles, P. (1988). A model for generalized edit and imputation of survey data. *The Canadian Journal of Statistics*, 16, 57-73.

Granquist, L. (1995). Improving the traditional editing process. Dans *Business Survey Methods*, (Éds., B.G. Cox, D.A. Binder, B.N. Chinnappa, A. Christianson, M.J. Colledge et P.S. Kott), John Wiley & Sons, Inc., 385-401.

Granquist, L. (1997). The new view on editing. *Revue Internationale de Statistique*, 65, 381-387.

Granquist, L., et Kovar, J. (1997). Editing of survey data: How much is enough? Dans *Survey Measurement and Process Quality*, (Éds., L.E. Lyberg, P. Biemer, M. Collins, E.D. de Leeuw, C. Dippo, N. Schwartz et D. Trewin), John Wiley & Sons, Inc., 415-435.

Hedlin, D. (2003). Score functions to reduce business survey editing at the U.K. Office for National Statistics. *Journal of Official Statistics*, 19, 177-199.

Hidiroglou, M.A., et Berthelot, J.-M. (1986). Contrôle statistique et imputation dans les enquêtes-entreprises périodiques. *Techniques d'enquête*, 12, 1, 79-89.

Kovar, J., et Whitridge, P. (1990). Generalized edit and imputation system; Overview and applications. *Revista Brasileira de Estadistica*, 51, 85-100.

Kruskal, J.B. (1983). An overview of sequence comparison. Dans *Time Warps, String Edits, and Macromolecules: The Theory and Practice of Sequence Comparison*, (Éds., D. Sankoff et J.B. Kruskal), Addison-Wesley, 1-44.

Lawrence, D., et McKenzie, R. (2000). The general application of significance editing. *Journal of Official Statistics*, 16, 243-253.

Liepins, G.E. (1980). *A Rigorous, Systematic Approach to Automatic Data Editing and its Statistical Basis*. Rapport ORNL/TM-7126, Oak Ridge National Laboratory.

Liepins, G.E., Garfinkel, R.S. et Kunnathur, A.S. (1982). Error localization for erroneous data: A survey. *TIMS/Studies in the Management Sciences*, 19, 205-219.

Little, R.J.A., et Smith, P.J. (1987). Editing and imputation of quantitative survey data. *Journal of the American Statistical Association*, 82, 58-68.

Naus, J.I., Johnson, T.G. et Montalvo, R. (1972). A probabilistic model for identifying errors in data editing. *Journal of the American Statistical Association*, 67, 943-950.

Navarro, G. (2001). A guided tour to approximate string matching. *ACM Computing Surveys*, 33, 31-88.

Pannekoek, J., Scholtus, S. et van der Loo, M. (2013). Automated and manual data editing: A view on process design and methodology. *Journal of Official Statistics*, 29, 511-537.

Pugh, W. (1992). The omega test: A fast and practical integer programming algorithm for data dependence analysis. *Communications of the ACM*, 35, 102-114.

R Development Core Team (2015). *R: A Language and Environment for Statistical Computing*. Vienne, Autriche : R Foundation for Statistical Computing. URL: http://www.R-project.org/.

Ragsdale, C.T., et McKeown, P.G. (1996). On solving the continuous data editing problem. *Computers & Operations Research*, 23, 263-273.

Riera-Ledesma, J., et Salazar-González, J.J. (2003). *New Algorithms for the Editing and Imputation Problem*. Document de travail n^{o} 5, UN/ECE Work Session on Statistical Data Editing, Madrid.

Riera-Ledesma, J., et Salazar-González, J.J. (2007). A branch-and-cut algorithm for the continuous error localization problem in data cleaning. *Computers & Operations Research*, 34, 2790-2804.

Schaffer, J. (1987). Procedure for solving the data-editing problem with both continuous and discrete data types. *Naval Research Logistics*, 34, 879-890.

Scholtus, S. (2011). Algorithms for correcting sign errors and rounding errors in business survey data. *Journal of Official Statistics*, 27, 467-490.

Scholtus, S. (2014). *Error Localisation using General Edit Operations*. Document de discussion 2014-14, Statistics Netherlands, La Haye. Disponible au : http://www.cbs.nl.

Tempelman, D.C.G. (2007). *Imputation of Restricted Data*. Thèse de doctorat, University of Groningen. Disponible au : http://www.cbs.nl.

van der Loo, M., et de Jonge, E. (2012). *Automatic Data Editing with Open Source R*. Document de travail n^{o} 33, UN/ECE Work Session on Statistical Data Editing, Oslo.

Williams, H.P. (1986). Fourier's method of linear programming and its dual. *The American Mathematical Monthly*, 93, 681-695.