Algorithmes automatiques pour la fouille visuelle de données et la visualisation de règles d'association : application aux données aéronautiques

Gwenael Bothorel

THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par :

Institut National Polytechnique de Toulouse (INP Toulouse)

Discipline ou spécialité :

Intelligence Artificielle

Présentée et soutenue par :

M. GWENAEL BOTHOREL

le jeudi 18 décembre 2014

Titre :

ALGORITHMES AUTOMATIQUES POUR LA FOUILLE DE DONNEES VISUELLE ET LA VISUALISATION DE REGLES D'ASSOCIATION. APPLICATION AUX DONNEES AERONAUTIQUES.

Ecole doctorale :

Mathématiques, Informatique, Télécommunications de Toulouse (MITT)

Unité de recherche :

Institut de Recherche en Informatique de Toulouse (I.R.I.T.)

Directeur(s) de Thèse :

M. JEAN MARC ALLIOT

M. MATHIEU SERRURIER

Rapporteurs :

M. GILLES VENTURINI, UNIVERSITE DE TOURS

Mme MICHÈLE SEBAG, UNIVERSITE PARIS 11

Membre(s) du jury :

Mme FLORENCE SEDES, UNIVERSITE TOULOUSE 3, Président

M. ERIC BLOND, DIR DES SERVICES DE NAVIGATION AERIENNE, Membre

M. JEAN MARC ALLIOT, ECOLE NATIONALE DE L'AVIATION CIVILE, Membre

M. MATHIEU SERRURIER, UNIVERSITE TOULOUSE 3, Membre

Remerciements

Ce travail de thèse n'aurait pas été rendu possible sans l'aide et le soutien de nombreuses personnes, cotoyées tant dans le domaine professionnel que dans la vie privée. Elles m'ont apporté leurs connaissances, leur gentillesse, leurs encouragements, ou simplement leur présence, qui ont été autant d'occasions de m'aider à avancer et, parfois, à reprendre la route.

Mes remerciements sont adressés à :

- Jean-Marc Alliot et Mathieu Serrurier qui m'ont mis sur les rails de cette aventure, en me témoignant de leur confiance. Mathieu m'a suivi de près pendant ces quatre années, avec une grande disponibilité, compétence et patience, trouvant toujours le mot juste pour lever les doutes et m'encourager.
- Michèle Sebag, Gilles Venturini, Florence Sèdes, Eric Blond et Christophe Hurter qui m'ont fait l'honneur d'être rapporteurs, examinateurs et invité de mon jury de thèse.
- Christiane Dujardin et Jean-Louis Garcia, qui ont accepté de me laisser poursuivre la thèse à la DTI dans de bonnes conditions, malgré de nombreux changements dans le service.
- Christophe Hurter pour la collaboration, les conseils et les encouragements.
- L'équipe PII actuelle (Eric, François, Robert, Rémi, Pierre, Géraldine, Bruno et Patrick) et ancienne, en remontant à de nombreuses années, dont Stéphane, qui m'a enseigné les fondamentaux il y a 25 ans, François-Régis et Alexandre, qui, en plus de partager le bureau pendant 13 ans, m'ont apporté les bases d'OpenGL, Jean-Luc, Jean-Paul, les Philippe, Daniel, Sylvie, Patrick, Franck et de nombreux autres.
- L'équipe APO qui m'a accueilli.
- Les anciens collègues du CENA, qui m'ont aidé de multiples manières.
- Les membres du domaine EEI qui m'ont souvent encouragé en me demandant si je voyais le bout du tunnel.
- Jean-Paul pour tous ses coups de main précieux.
- Isabelle et Marie-Antoinette pour leur aide efficace.
- Robert et Roland pour les précieux conseils en analyse statistique.
- Serge et Bernard pour les données.
- François pour le design.
- Jean et David, mes collègues de bureau d'une année, qui ont su éclairer ma lanterne sur le contrôle aérien.
- Les sujets de mon expérimentation qui se sont prêtés au jeu.
- Mes compagnons de thèse : Caroline, Brunilde, Laureline, Olga, Maxime, Cyril, Clément, Richard, Mohammad, sans oublier les papys thésards Jean-Paul et Jean-Luc.

- Ceux qui m'ont permis de suivre des formations enrichissantes, notamment Yannick, Kim et Robert.
- Les membres de l'ABCCPR, institution matinale qui a su résister à la tempête.
- Peppino, François, Sébastien, Etienne, Nacho et Jean-Marie qui m'ont encouragé pendant tout ce temps.
- Toni qui m'a soutenu discrètement mais efficacement.
- Mes parents et amis qui s'inquiétaient régulièrement de mon sort : Paule, Michel, Andrée, et tous les autres.
- Je terminerai par mon épouse Véronique et mes enfants qui m'ont toujours soutenu et aidé, chacun à sa manière, malgré mon indisponibilité chronique.

Veritas liberabit vos
Jn 8, 32

Table des matières

Table des figures

Première partie

Introduction

Depuis quelques années, nous assistons à une explosion de la production de données, et cela dans divers domaines, qu'ils soient scientifiques, médicaux, économiques, sociaux... Des comparaisons peuvent aider à appréhender l'ampleur de ce phénomène. Eric Shmidt, alors P.D.G. de Google, déclarait en août 2010, lors de la conférence Techonomy : « Aujourd'hui, nous créons tous les deux jours autant d'information que nous en avons créée depuis l'aube de la civilisation jusqu'en 2003... C'est de l'ordre de 5 exaoctets [1] [2]. » Cette quantité est aujourd'hui à mettre en regard avec la capacité de stockage du Utah Data Center de la N.S.A. (National Security Agency). Lors de son intervention au 2012 Energy Summit, le Gouverneur Gary Herbert l'a présenté comme la première installation au Monde capable de rassembler et d'héberger un yottaoctet de données [3] [Herbert 12]. Cette manne de données numériques, mise en réseau et exploitable par des calculateurs de plus en plus puissants, procure un potentiel jusqu'à présent jamais atteint pour l'analyse, l'étude et la compréhension du Monde dans lequel nous vivons. La démocratisation des périphériques connectés, des smartphones aux tablettes, en passant par les montres et bracelets, va également dans ce sens. Il en de même, des voitures, dont les modèles, qui ont été présentés au Consumer Electronic Show de Las Vegas en janvier 2014 [4] (BMW, Audi, General Motors, Honda, Hyundai...), préfigurent le véhicule de demain.

Beaucoup de domaines s'intéressent à l'exploitation des masses de données. Il s'agit, entre autres, de la grande distribution, du cyber-marché, des assurances, des banques, des ressources humaines, de la médecine, de l'astronomie, de la météorologie, des réseaux sociaux... L'aéronautique en fait également partie. Que ce soit dans les systèmes embarqués à bord des avions ou dans les systèmes au sol, le flux de données est permanent. Classer les données aéronautiques n'est pas aisé, car cela dépend du point de vue selon lequel nous nous plaçons. Nous proposons trois catégories. La première concerne ce qui est généré par l'avion et qui n'est pas émis vers l'extérieur. Il s'agit par exemple des communications internes entre les pilotes. Ce type de données peut être exploité ultérieurement à l'extérieur de l'avion. C'est le cas des enregistrements dans les boîtes noires. Une deuxième catégorie recouvre les données émises par les systèmes de gestion du trafic aérien, ou systèmes ATM (Air Trafic Management), et qui ne remontent pas vers les systèmes bord. Une troisième catégorie concerne les données échangées entre les systèmes embarqués et les systèmes au sol. Celle-ci est appelée à se développer dans le but d'avoir un échange plus fréquent qu'actuellement, voire permanent entre l'avion et les systèmes au sol. Les moyens mis en œuvre pour retrouver les boîtes noires du crash du vol AF447, survenu le 1er juin 2009 au large du Brésil, a montré l'importance des données pour comprendre les raisons de l'accident. Ces recherches ont duré environ deux ans, pour un coût estimé de 111,6 millions d'euros. Le rapport final [BEA 12] du Bureau d'Enquêtes et Analyses préconise, dans ses recommandations de sécurité, d'étudier la possibilité d'imposer pour les avions effectuant du transport public de passagers la transmission régulière de paramètres de base, comme la position, l'altitude, la vitesse et le cap. Cette recommandation a été adressée

1. 1 exaoctet correspond à 10^6 teraoctets, soit 10^{18} octets
2. http ://www.techonomy.com
3. 1 yottaoctet correspond à 10^6 exaoctets, c'est-à-dire 10^{12} teraoctets
4. http ://www.cesweb.org

à l'EASA (European Aviation Safety Agency) [5] et à l'OACI (Organisation de l'Aviation Civile Internationale) [6], puis a été réitérée à la suite de la disparition du vol MH370 de Kuala Lumpur à Pékin, le 8 mars 2014.

La généralisation des échanges de données entre les différents systèmes aéronautiques s'inscrit, par ailleurs, dans une politique d'évolution de la gestion du trafic aérien. Dans ce contexte, le programme européen SESAR (Single European Sky ATM Research) prépare le futur système qui sera déployé dans quelques années. Il a pour objectif de moderniser et d'harmoniser ceux qui sont actuellement en service en Europe, et se présente comme le volet technologique du Ciel Unique Européen [SES 99]. Ses membres sont issus des acteurs essentiels de l'aéronautique : les industriels produisant des systèmes sol et embarqués, des avionneurs, des fournisseurs de service de contrôle aérien, etc. Ce programme est constitué d'un ensemble de projets dont le huitième, appelé SWIM (System Wide Information Management) [SJU 12] a pour objectif d'assurer un partage d'informations en temps réel entre tous les systèmes, portant sur les données relatives au vol des avions, notamment leur trajectoire, à l'espace aérien et à la météorologie. Ce projet est présenté comme l'intranet du futur système de gestion du trafic aérien. Ces exemples montrent à quel point l'information et son partage sont des points clé dans l'évolution des systèmes ATM, sachant que les prévisions de trafic européen annoncent une augmentation de 2,9% par an jusqu'à 2019 [Eurocontrol 13]. Cela entraînera un fort accroissement du volume de données échangées.

La production massive et permanente de données a donné lieu à un nouveau domaine scientifique et économique appelé *Big Data*. Cette expression semble trouver son origine dans une publication de Cox & Ellsworth en 1997 [Cox 97], dont l'introduction commence par les mots suivants : « Visualization provides an interesting challenge for computer systems : data sets are generally quite large, taxing the capacities of main memory, local disk, and even remote disk. We call this the problem of big data. » Cependant, le problème de la quantité de données se posait déjà en 1944, lorsque Rider estimait que le volume de livres contenu dans les bibliothèques des universités américaines doublait tous les 16 ans. Ainsi, la bibliothèque de Yale aurait dû faire face à approximativement 200 millions d'ouvrages en 2040 [Rider 44]. Aujourd'hui, le problème ne se pose bien évidemment plus en ces termes, mais la question reste posée, et de manière beaucoup plus cruciale. Ainsi, les prévisions de l'IDC (International Data Corporation), fournisseur spécialisé dans l'étude de marché dans les domaines des technologies de l'information et de la communication, pour 2014, envisagent une augmentation de 30% des dépenses en services et technologie liés au Big Data [Gens 13]. En 2001, une note de Meta Group a défini, dans une problématique de commerce en ligne, le modèle *3D Data Management* qu'il est nécessaire de maîtriser pour faire face à la forte croissance du volume de données [Lan 01]. Ce modèle, toujours utilisé pour décrire le Big Data, est constitué des trois approches suivantes :

- Volume des données : le commerce en ligne augmente la quantité et le spectre des données relatives aux transactions. En baissant le prix de ce service, cela augmente le

5. http ://www.easa.eu.int
6. http ://www.icao.int

nombre de clients et donc la quantité de données. De plus, les entreprises les considèrent comme faisant partie de l'actif tangible, et sont donc réticentes à s'en séparer. Il est donc nécessaire d'augmenter les capacités de stockage ;

- Vitesse des données : elle concerne le temps de réponses des sites marchands, la disponibilité de l'analyse des données, l'exécution de la transaction, la mise à jour du suivi de la commande, etc. Les systèmes doivent être de plus en plus capables de réagir en temps réel, malgré l'accroissement des flux de données ;
- Variétés des données : elles sont de différents types et proviennent de sources hétérogènes et asynchrones. Il s'agit de données textuelles, structurées ou non, de données multimédia... Privées ou publiques, elles sont émises de manière continue, cyclique ou aléatoire.

L'extraction de connaissances à partir des données (ECD), ou Knowledge Discovery in Databases (KDD), est un domaine informatique qui a pour but d'extraire de l'information et de la connaissance à partir d'une très grande quantité de données [Piatetsky-Shapiro 91b, Fayyad 96a]. Il s'agit d'un processus ayant fait l'objet de plusieurs modèles le découpant en plusieurs étapes [Clos 07], allant des données initiales jusqu'à l'exploitation de l'information qui en est retirée, en passant par des étapes de préparation et d'extraction. L'une de ces étapes est appelée fouille de données, ou Data Mining. Elle consiste à extraire des motifs reliant les données, dans le but de les exploiter par la suite.

Cette étape de fouille de données s'appuie sur des algorithmes, sur de l'exploration visuelle ou sur les deux de manière combinée. Elle produit, à partir de données initiales, des motifs qui sont ensuite exploités, par exemple sous la forme de règles d'association qui relient les données de manière ordonnée, ou sous forme de modèles, ceux-ci étant des représentations mathématiques qui procurent une vision simplifiée de systèmes ou de phénomènes complexes. Chacune des approches, algorithmique et visuelle, présente des avantages, mais aussi des limites qui peuvent s'avérer rédhibitoires. La première se présente souvent sous forme de boîte noire qui fournit des résultats, mais qui ne conduisent pas toujours à la résolution du problème, par exemple parce qu'elle ne prend pas en compte les connaissances de l'utilisateur. La seconde considère certes cette connaissance, mais il est difficile d'appréhender des jeux de données multidimensionnelles trop volumineux.

Aujourd'hui, ce problème est à envisager sous l'angle d'un domaine récent appelé le *Visual Analytics*. Ses origines remontent aux attentats du 11 septembre 2001, qui ont montré la nécessité de coordonner les efforts pour sécuriser les Etats Unis et ses ressortissants. Dans ce but, le DHS [7] a été créé en 2003. L'une de ses missions est d'être capable d'analyser l'énorme quantité de données disparates et dynamiques, afin de prévenir toute menace, de protéger les frontières et de répondre à toute attaque ou autre désastre [Thomas 05]. Étant donné que la quantité de données produites est supérieure à la capacité de les analyser, le besoin de nouvelles méthodes a alors été exprimé pour être à même de traiter des données massives, multidimensionnelles, multi-sources et produites en permanence. *Illuminating the path* [Thomas 05] est l'ouvrage fondateur de ce domaine. Il se présente comme un appel adressé

7. US Department of Homeland Security

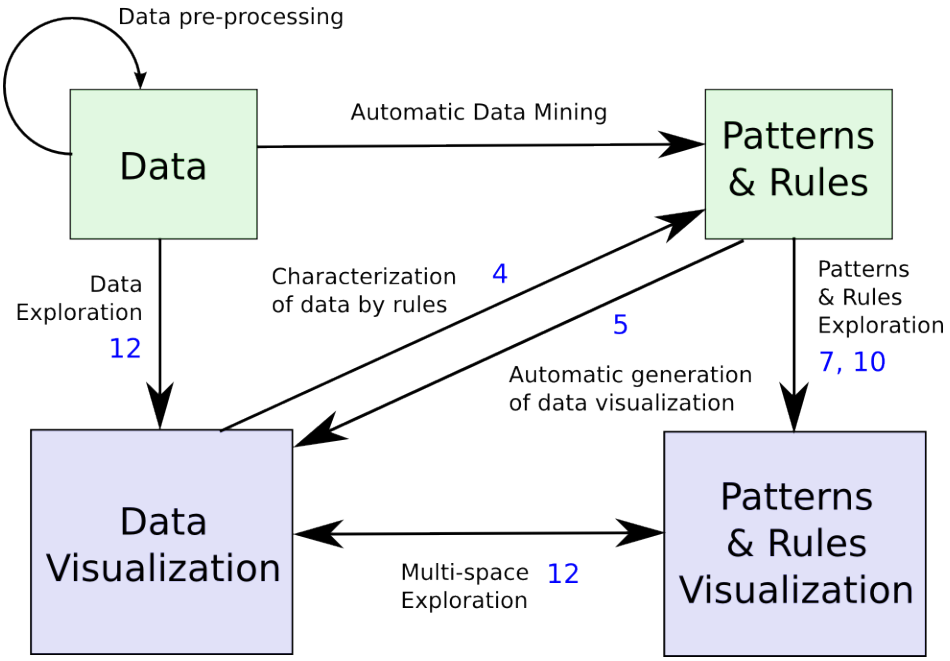


FIGURE 1 – Représentation schématique de la fouille algorithmique de données et la fouille visuelle de données.

Les valeurs situées à côté des liens sont les numéros de chapitre où ils sont explicités.

à la communauté scientifique, à transformer l'information en perspicacité (*insight*) grâce au Visual Analytics, qui est la science du raisonnement analytique facilité par les interfaces visuelles interactives [Wong 04]. Dans cette dynamique, la Commission Européenne a lancé le programme VisMaster [8] en 2008, afin de réunir la recherche académique et les industriels dans une communauté de recherche. Cela a donné lieu en 2010 à un état de l'art et à une liste d'actions et de défis à relever pour les années à venir [Keim 10a].

Notre travail de thèse, présenté dans ce mémoire, s'inscrit dans le Visual Analytics. La figure 1, qui définit le périmètre de notre recherche, est une représentation globale de la combinaison des fouilles de données algorithmiques et visuelles. Celle-ci est explorée en nous appuyant sur la collaboration entre l'homme et le système, notamment dans le cadre de la recherche de règles d'association. Ainsi, l'expertise humaine est renforcée par la capacité de calcul du système. Pour cela, un environnement multi-spatial est étudié, présentant, d'une part, les données, et, d'autre part les résultats algorithmiques. Ces deux espaces s'alimentent mutuellement par la mise en œuvre d'algorithmes enrichissant chaque espace par les informations issues de l'autre espace. La partie gauche de la figure concerne l'exploration des données. Elle est mise à profit pour piloter la génération de règles d'association. La partie droite correspond à l'exploration des motifs et des règles d'association, pour laquelle nous proposons de nouvelles formes de visualisation. A partir des règles d'association, nous proposons par ailleurs de configurer automatiquement la visualisation des données.

8. http ://www.vismaster.eu et http ://www.visual-analytics.eu

Après cette introduction, la deuxième partie du document est un état de l'art, présentant les principes et les travaux dans le domaine de la fouille de données. L'approche algorithmique procure des motifs et des règles qui sont caractérisés par des métriques appelées également mesures de qualité. Celles-ci ont une incidence sur la manière dont le motif ou la règle est utilisé dans la suite du processus. L'approche visuelle permet d'explorer les données et de détecter des motifs que l'algorithme ne peut trouver. Elle est également utilisée pour explorer une représentation des résultats des algorithmes, ce qui donne lieu à de la fouille de motifs ou de règles d'association. Des solutions mixtes combinant algorithme et visualisation sont également présentées. Elles proposent une vision plus globale de la fouille, intéressante pour le Visual Analytics.

La troisième partie présente notre problématique en nous appuyant sur l'état de l'art. Elle est déclinée en plusieurs questions, auxquelles nous essayons de répondre dans la suite du mémoire, portant notamment sur la représentation des résultats algorithmiques et sur le lien bilatéral qui peut être établi entre ceux-ci et les données.

Dans la quatrième partie, nous établissons un lien entre la fouille de données algorithmique et la fouille visuelle de données. Pour cela, nous formalisons la visualisation, ce qui permet de diriger et de contraindre la mise en œuvre des algorithmes qui deviennent ainsi pilotés par l'utilisateur. Les motifs et règles ainsi obtenus, permettent en retour, grâce à la formalisation, d'enrichir et de configurer automatiquement la visualisation des données initiales. Leur implication dans la base de données se trouve ainsi exploitée et valorisée visuellement.

Dans la cinquième partie, nous étudions et proposons de nouvelles visualisations des motifs et des règles d'association, en nous appuyant sur un outil d'exploration visuelle que nous avons développé. Il représente les motifs sous la forme d'un graphe circulaire, qui est optimisé et enrichi par des techniques de visualisation d'informations, comme le bundling. L'optimisation a fait l'objet d'une expérimentation qui sera explicitée. Les règles sont représentées à l'aide d'un scatter plot permettant la visualisation simultanée de plusieurs dimensions. Ces visualisations permettent de représenter une très grande quantité de règles et de motifs, tout en gardant la possibilité de les connaître dans le détail.

Enfin, nous présentons, dans la dernière partie, des scénarios d'usage de la plate-forme que nous avons mise en œuvre, appliqués au domaine des données aéronautiques. Ils illustrent, selon une approche centrée utilisateur, la combinaison de l'exploration des données, de la mise en œuvre des algorithmes de fouille et de l'exploration des motifs et des règles d'association. A partir d'enregistrements de données de trafic aérien, la génération de règles d'association est pilotée par l'exploration visuelle des données. Puis, les règles sont traitées pour générer automatiquement des visualisations de données.

Ce travail de thèse est réalisé dans un contexte d'exploitation de données aéronautiques qui ne sont pas publiques. Afin d'illustrer la formalisation et l'explication de la visualisation des motifs et des règles d'association, nous exploitons des bases de données publiques, utilisées couramment dans les publications du domaine de l'extraction de connaissances à partir de

données (conférences KDD, KDD Cup) [9] [10]. Cela permet d'assurer une reproductibilité des résultats.

9. http ://www.kdnuggets.com
10. http ://www.kdd.org

Deuxième partie

Etat de l’art

Introduction

Le Visual Analytics est une discipline récente qui puise son origine dans les attentats du 11 septembre 2001. S'appuyant sur plusieurs approches et domaines scientifiques, elle combine les techniques algorithmiques qui permettent d'exploiter les puissances de calculs des systèmes informatiques, et les techniques visuelles offrant à l'homme des représentations variées de jeux de données. Il permet ainsi d'aller au-delà de la fouille de données grâce à cette pluridisciplinarité.

Afin d'en comprendre les différents aspects, nous allons, dans un premier chapitre, aborder de manière globale l'extraction de connaissances à partir des données, dont une partie est la fouille de données, appelée également *Data Mining*. Dans ce cadre, l'approche algorithmique traite les données pour en extraire des motifs mettant en évidence des associations entre elles.

Une autre manière d'aborder les données est la fouille visuelle qui est un outil puissant permettant à l'utilisateur de les appréhender et d'en détecter des caractéristiques. Elle fait l'objet de la seconde grande partie de l'état de l'art. Après avoir posé les bases théoriques de la visualisation, de nombreuses méthodes et techniques seront ainsi présentées. Cependant, la visualisation ne concerne pas uniquement les données, car elle est également mise en œuvre pour représenter des résultats algorithmiques issus de l'approche décrite dans la première partie.

Les recherches dans le cadre du Visual Analytics combinent ces techniques, et proposent maintenant une approche mixte placée dans un continuum allant de l'approche algorithmique exclusive à l'approche uniquement visuelle. C'est par ce dernier point que nous achèverons cette partie.

Chapitre 1

Data Mining

1.1 L'extraction de connaissances à partir de données

1.1.1 Définition de la donnée

Avant d'aborder l'extraction de connaissances à partir de données, il convient de définir ce qu'est une donnée.

Le dictionnaire Petit Larousse [1] en donne les définitions suivantes :

- Ce qui est connu ou admis comme tel, sur lequel un raisonnement peut être fondé, qui sert de point de départ pour une recherche (surtout pluriel) : Les données actuelles de la biologie ;
- Idée fondamentale qui sert de point de départ, élément essentiel sur lequel est construit un ouvrage : Les données d'une comédie ;
- Renseignement qui sert de point d'appui (surtout pluriel) : Manquer de données pour faire une analyse approfondie ;
- Représentation conventionnelle d'une information en vue de son traitement informatique ;
- Dans un problème de mathématiques, hypothèse figurant dans l'énoncé ;
- Résultats d'observations ou d'expériences faites délibérément ou à l'occasion d'autres tâches et soumis aux méthodes statistiques.

Elle peut ainsi être définie comme une entité recueillie consécutivement à une cause et comme point de départ pour une future utilisation. Cette entité peut être de différents types, de même que la cause peut être très variée. Il peut s'agir d'une mesure, un raisonnement, une idée, une hypothèse, un renseignement, un résultat d'observation. Ce qui lie toutes ces causes et font qu'il en résulte une donnée, est l'intention de l'exploiter. En effet, peut-on dire qu'une donnée existe avant de la connaître ? La découverte de Lucy [Johanson 76] en 1974 a apporté une multitude d'informations qui ont été exploitées et archivées sous forme de données. Ainsi,

1. http ://www.larousse.com

la donnée est la conséquence de cette découverte et n'existait pas avant, car elle n'était pas recueillie. Si la découverte de ce squelette n'avait pas été exploitée, alors il n'en aurait résulté aucune donnée, si ce n'est son existence et sa position.

La donnée recueillie est archivée, transformée, exploitée... L'archivage est réalisé sous la forme de fichiers de type image, vidéo, sonore, sous forme alphanumérique, sous forme structurée comme un tableau ou une liste d'éléments, etc. Nous réduisons le cadre de notre propos aux données numériques, parce qu'elles seules concernent cette étude. Nous ne prenons donc pas en compte d'autres supports, comme le papier, la pierre, etc. Dans d'autres domaines, ceux-ci seraient cependant incontournables. Ainsi, les archives d'état civil anciennes révèlent, grâce aux caractéristiques de l'écriture manuscrite, des informations intéressantes quant au niveau social des personnes, ne serait-ce que par la manière de signer.

Dans la suite du manuscrit, dans un souci de simplification d'écriture et pour limiter les répétitions, nous utiliserons indifféremment les mots *données* et *informations*, en considérant qu'il s'agit bien de données avec le sens que nous avons défini dans ce chapitre.

1.1.2 Le processus d'Extraction de Données à partir des Connaissances

La donnée peut constituer une connaissance en tant que telle, mais peut également être traitée pour extraire de la connaissance. Jean-Paul Benzécri [Benzécri 77] écrivait en 1977 que l'analyse des données avait pour objectif de dégager de la gangue des données le pur diamant de la véridique nature. Plus tard, Witten et al. [Witten 11] ont défini la fouille de données comme l'extraction d'informations implicites, inconnues et utiles à partir des données. Cela a donné lieu à l'émergence d'une nouvelle discipline dans les années 80 appelée Knowledge Discovery in Databases (KDD) ou Extraction de Connaissances à partir des Données (ECD) [Piatetsky-Shapiro 91b, Fayyad 96a]. Gregory Piatetsky-Shapiro [2] est un pionnier dans ce domaine. Fondateur des conférences KDD [3], il a organisé le premier workshop *Knowledge Discovery in Databases* en 1989. Depuis, nombre de travaux et de publications alimentent cette communauté et en enrichissent les techniques.

La définition de l'extraction de connaissances à partir des données a été donnée en 1996 par Fayyad et al. : « KDD is the nontrivial process of identifying valid, novel, potentially useful, and ultimately understandable patterns in data [Fayyad 96b]. » L'ECD est un domaine distinct de l'apprentissage automatique (*Machine Learning*), car ce dernier met l'accent sur l'induction de modèles pour la prédiction, par exemple pour reproduire une tâche humaine ou pour adapter un comportement en fonction de résultats de calculs. L'extraction de données combine des techniques issues de disciplines variées, comme les bases de données, l'intelligence artificielle, les statistiques, les découvertes scientifiques et la visualisation [Williams 96]. Elle est donc à l'intersection de ces disciplines et va puiser dans chacune d'elles. Une de ses caractéristiques est la taille des bases de données qu'elle traite, qui atteindra le yottaoctet avec la mise en service du Utah Data Center de la N.S.A. [Herbert 12].

2. http ://www.kdnuggets.com/gps.html
3. http ://www.sigkdd.org

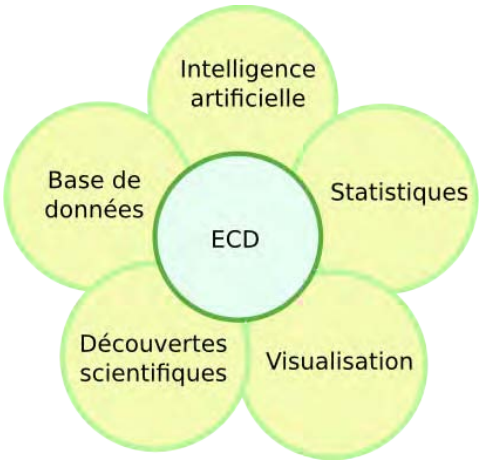


FIGURE 1.1 – L'extraction de connaissances à partir de données est à l'intersection de plusieurs disciplines.

L'extraction de connaissances à partir de données est un processus interactif et itératif, contenant plusieurs niveaux de décisions de la part de l'utilisateur [Fayyad 96a]. Il est constitué des neuf étapes suivantes [Fayyad 96a, Clos 07] (Figure 1.2) :

1. Développer une compréhension du domaine d'application et identifier le but du processus ECD du point de vue de l'utilisateur ;
2. Sélectionner un jeu de données sur lequel l'extraction va être réalisée ;
3. Nettoyer et prétraiter les données. Cela concerne par exemple l'extraction du bruit et la mise en œuvre de stratégies dans le cas de données manquantes ;
4. Réduire et projeter les données, en identifiant des caractéristiques pour les représenter en fonction du but de la tâche. Le nombre de variables peut ainsi être fortement diminué ;
5. Faire correspondre les buts de l'extraction avec une méthode de fouille de données particulière, comme la classification, le regroupement (clustering) ;
6. Analyser de manière exploratoire et choisir l'algorithme de fouille de données et de la méthode de sélection ;
7. Réaliser la **fouille de données**. Il s'agit de rechercher des motifs intéressants ;
8. **Interpréter les motifs trouvés**. Cette étape comprend également la **visualisation des motifs** ;
9. Valoriser la connaissance acquise, en l'utilisant directement, ou en l'intégrant dans un autre système pour un futur processus.

Ce processus est itératif, de manière globale ou entre différentes étapes, comme indiqué sur la figure 1.2.

La notion de fouille de données ou Data Mining varie selon la littérature. La définition de ce concept peut aller de l'extraction de motifs jusqu'au processus global d'Extraction de Connaissances à partir des Données. Dans ce mémoire, nous considérons qu'il ne porte que sur l'extraction de motifs.

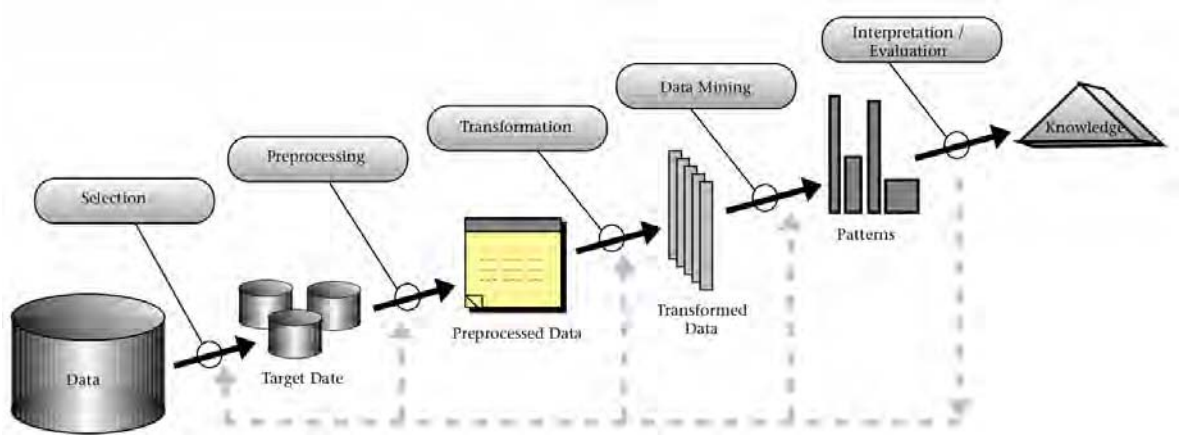


FIGURE 1.2 – Les neuf étapes de l'Extraction de Connaissances à partir de Données [Fayyad 96a].

1.2 Apprentissage supervisé et apprentissage non supervisé

Les techniques d'apprentissage font l'objet d'une riche littérature. Nous renvoyons le lecteur vers des ouvrages de référence, comme *Apprentissage artificiel : Concepts et algorithmes* [Cornuéjols 10] ou *Machine Learning* [Mitchell 97].

1.2.1 Apprentissage supervisé

Il consiste à élaborer un apprentissage automatique à partir d'un ensemble de données de référence appartenant à des classes déterminées. Pour cela, les attributs de chaque donnée sont analysés et servent à établir une description des classes. A partir de cette description, il devient possible de trouver la classe d'appartenance de toute autre donnée. Ce type d'apprentissage concerne par exemple les arbres de décision et les réseaux de neurones. Les premiers présentent une situation sous la forme d'un arbre dont les nœuds correspondent à des résultats de décisions. Les seconds s'inspirent du comportement des neurones lors de la transmission d'informations entre eux, chaque neurone étant connecté à plusieurs autres en amont et en aval. La transmission ou non de l'information résultante d'un neurone vers les suivants dépend d'une fonction d'activation.

1.2.2 Apprentissage non supervisé

Dans ce type d'apprentissage, les classes ne sont pas connues d'avance et les informations sont extraites directement des données. Parmi les méthodes usuelles, se trouvent le clustering et la recherche de motifs fréquents. Le clustering consiste à partitionner les données, et à les regrouper en classes, à partir de critères d'homogénéité qui permettent de définir une distance. Grâce à celle-ci, des groupes de données sont établis, en minimisant les distances intra-groupes et en maximisant les distances inter-groupes. Les motifs fréquents ou *itemsets* caractérisent des

associations entre des attributs de données. Leur recherche sera abordée dans le paragraphe 1.5.1.

1.3 Le Data Mining

La fouille de données est une étape du processus d'Extraction de Connaissances à partir des Données. Son but est de rechercher des motifs reliant les données entre elles, sachant qu'ils doivent être intéressants, et, pour cela, nouveaux, utiles et non triviaux [Frawley 92]. L'image régulièrement utilisée est celle de la montagne au sein de laquelle sont enfouies des pépites. La fouille de données consiste donc à les extraire malgré l'immensité de la montagne. Une anecdote, connue dans le monde de la fouille de données, serait à l'origine de la réussite de la chaîne de grande distribution Walmart. En exploitant les données issues des achats des clients, un lien a été détecté entre la vente en soirée des packs de bière et celle des couches culottes. Il s'expliquait par les pères de famille qui venaient acheter simultanément la bière et les couches. Après cette constatation, les rayonnages de bière et de couches ont été rapprochés, et cela a entraîné un impact sensible sur le chiffre d'affaire.

Initialement, deux communautés abordaient la fouille de données de manière différente. D'une part se trouvaient les partisans de la visualisation d'information, dont l'objectif était de donner à l'utilisateur une vue générale des données, tout en permettant une vue détaillée. D'autre part, les défenseurs de l'approche algorithmique arguaient de la suffisance des méthodes statistiques et d'apprentissage pour trouver des motifs intéressants. Aujourd'hui, même si ces deux philosophies existent toujours, une troisième approche a vu le jour par la recherche de motifs en combinant l'approche visuelle et l'approche algorithmique. Shneiderman [Shneiderman 01] a abordé cette question en mettant en regard ces approches, dans la lignée de Tukey [Tukey 65] et Westphal & Blaxton [Westphal 98]. Il en a déduit quatre recommandations pour l'élaboration de futurs systèmes de recherche de connaissance :

- Intégrer la Data Mining et la visualisation d'informations pour élaborer de nouveaux outils. En ajoutant la visualisation au processus de Data Mining, l'utilisateur développe une meilleure compréhension de ses données. Réciproquement, en ajoutant le Data Mining à la visualisation, l'utilisateur peut spécifier ce qu'il recherche ;
- Permettre à l'utilisateur de spécifier ce qu'il cherche et ce qu'il estime intéressant. En lui permettant de contraindre et de diriger ses outils, ils peuvent être plus efficients ;
- Reconnaître que l'utilisateur est situé dans un contexte social. Il ne travaille pas seul et a donc besoin d'échanger des données et de les présenter ;
- Respecter la responsabilité de l'homme dans la conception des futurs outils. S'ils sont bien conçus, il parviendra à les utiliser et sera d'autant plus performant.

Cette fouille de donnée est réalisée de manière automatique, de manière manuelle, ou par une combinaison de ces deux approches.

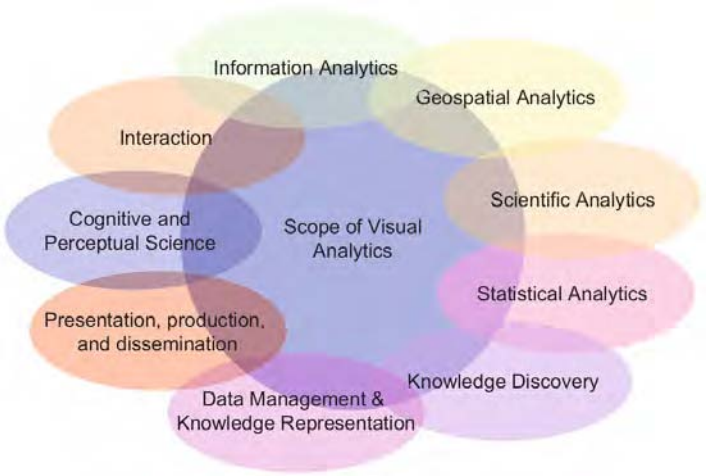


FIGURE 1.3 – Etendue pluridisciplinaire du Visual Analytics [Keim 08b].

1.4 Le Visual Analytics

Le Visual Analytics a été défini comme étant la science du raisonnement analytique facilité par les interfaces visuelles interactives [Wong 04]. Keim et al. [Keim 10b] complètent cette définition en le présentant comme une combinaison de techniques d'analyse automatique et de visualisations interactives, pour une compréhension, un raisonnement et une prise de décision efficaces, sur la base d'une très grande quantité de données complexes. Le but est de transformer une surcharge d'informations en opportunité [Keim 10a]. En effet, comme la visualisation d'informations a évolué en changeant notre vision des bases des données, cela en fonction des progrès technologiques liés aux capacités graphiques et de traitement des systèmes, il s'agit de rendre le traitement des données transparent pour l'utilisateur. La visualisation des processus fournit un moyen de les examiner tout au long de leur déroulement, plutôt que d'avoir à traiter des résultats. Pour cela, le Visual Analytics cherche à combiner les qualités de l'homme et celles de la machine. La visualisation devient ainsi un moyen au service d'un processus semi-automatique, considérant que l'utilisateur garde toujours le pouvoir de décision afin de pouvoir l'orienter. Ce domaine pluridisciplinaire met en commun des recherches issues de différents horizons, telles que la visualisation, le Data Mining, la gestion de données, la fusion de données, les statistiques, les facteurs humains, etc. La figure 1.3 illustre son étendue, les recouvrements de disciplines montrant les liens qui sont réalisés entre elles. Comme le Visual Analytics peut être considéré comme un domaine intégrant plusieurs disciplines, celles-ci y contribuent avec leurs propres outils et modèles, et deviennent ainsi interdépendantes du fait de ses besoins [Keim 08a].

Le processus du Visual Analytics, défini par Keim et al. [Keim 08b, Keim 10a], est schématisé par la figure 1.4. Les étapes sont représentées par des ovales et les transitions par des flèches. Ce processus est caractérisé par les interactions entre les données, les visualisations, les modèles et les utilisateurs.

La transformation des données consiste à les prétraiter, car elles sont hétérogènes et issues de sources variées. Elles sont générées selon différentes présentations exploitables par la suite. Après cette transformation, l'utilisateur choisit entre l'approche automatique, qui va traiter les données à l'aide d'algorithmes, et l'approche visuelle, qui explore les données à l'aide d'outils de visualisation.

L'approche automatique produit des modèles ou des motifs. Un modèle [Bertini 09] est une représentation mathématique ou logique d'un système d'entités, de phénomènes ou de processus. Il correspond à une vue simplifiée abstraite de réalité complexe, qui facilite le raisonnement humain, et peut être simulé, visualisé et manipulé. Un motif [Bertini 09] est constitué d'événements ou d'informations récurrents, qui interviennent, simultanément ou non, de manière prédictible. Une fois généré, il est ensuite évalué et raffiné, éventuellement en agissant au niveau des données. Grâce à la visualisation, l'utilisateur interagit sur les algorithmes en modifiant leurs paramètres ou en sélectionnant l'algorithme le plus approprié à la tâche. Elle permet également de visualiser les modèles. En alternant entre l'utilisation des algorithmes et la visualisation des modèles, l'utilisateur peut ainsi affiner progressivement son étude, cela de manière itérative, jusqu'à l'aboutissement d'une solution satisfaisante. Dans le cas de l'approche visuelle, les données sont explorées à l'aide d'outils de visualisation, assistés par des opérateurs d'interaction, comme le zoom, et par des représentations de différentes sortes.

Il ressort du processus du Visual Analytics que la connaissance émerge des algorithmes, de la visualisation, de leur action combinée, et de l'expertise humaine. Cela a été résumé par Keim et al. [Keim 06] de la manière suivante : « Analyse First, show the important, zoom, filter and analyse further, details on demand ». La suite de l'état de l'art est réalisée selon cette vision du Visual Analytics. Nous allons d'abord aborder l'approche algorithmique, avec le Data Mining, permettant d'extraire des motifs fréquents et des règles d'association. Puis nous aborderons l'approche visuelle avec le Visual Data Mining, permettant, d'une part, de visualiser les données, et d'autre part de visualiser le résultat des algorithmes de Data Mining.

1.5 Approche algorithmique de l'extraction de règles d'association

L'approche algorithmique a pour objectif l'extraction de règles d'association. A partir d'une base de transactions, elle recherche, dans un premier temps, les motifs fréquents satisfaisant des conditions de seuil. En fonction de la quantité de données et des buts recherchés, de nombreux algorithmes sont étudiés, dont quelques uns sont présentés dans ce document. A partir de ces motifs, des règles d'association sont extraites, permettant de réaliser des liens entres les items qui les composent. Elles sont caractérisées par des mesures d'intérêt, mettant en exergue certains aspects. Celles-ci étant nombreuses, quelques unes sont présentées dans ce chapitre.

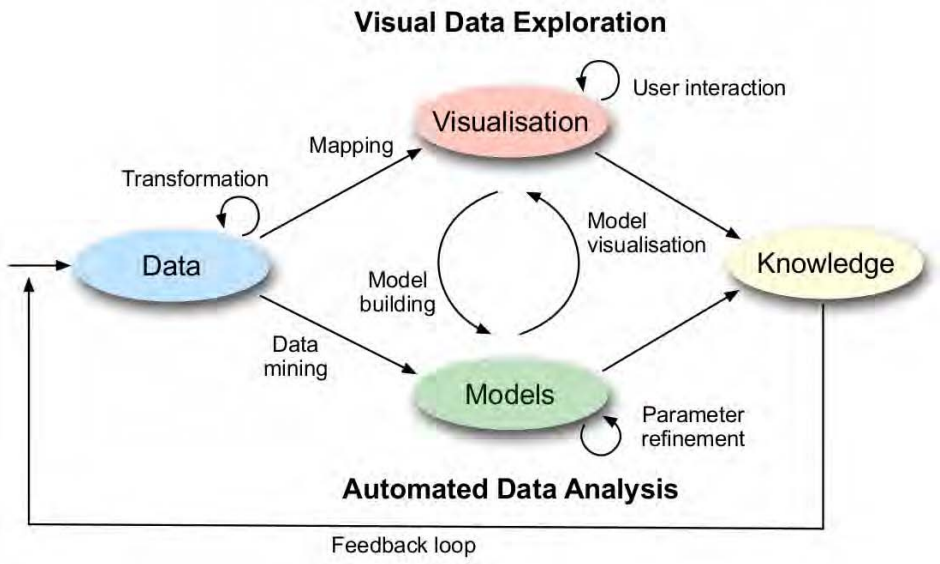


FIGURE 1.4 – Le processus du Visual Analytics [Keim 08b, Keim 10a].

1.5.1 Itemset

Principe et définition

Les données, appelées également *transactions*, sont composées d'attributs. Leur ensemble constitue la base de données.

Nous appelons $\mathcal{A} = \{A_1, A_2, ..., A_m\}$ l'ensemble des m attributs possibles d'une base. Ils peuvent être regroupés de multiples manières pour former des *itemsets* définis de la manière suivante :

Définition 1 *(itemset) Un itemset $\mathcal{I}$ est un sous-ensemble d'attributs :*

$$\mathcal{I} = \{I_1, I_2, ..., I_n\},\ \mathcal{I} \subset \mathcal{A}$$

Il est également noté k-itemset pour préciser son cardinal k, qui est appelé son *ordre*. Ainsi, un 1-itemset, c'est-à-dire un itemset d'ordre 1, correspond à un seul attribut.

Définition 2 *(support [Agrawal 93]) Le support $s(\mathcal{I})$ d'un itemset $\mathcal{I}$ est le nombre d'occurrences de celui-ci dans la base de données.*

Constituant la mesure principale caractérisant un itemset, il est exprimé, soit par le nombre d'occurrences, soit, ce qui est le plus fréquent, en indiquant sa fréquence d'apparition. En d'autres termes, il donne une mesure de la généralisation de l'itemset dans la base. Par exemple, un itemset ayant une fréquence égale à $0,20$ apparaît dans 20% des enregistrements de la base.

Définition 3 *(itemset fréquent) Un itemset est dit fréquent si son support est supérieur à un seuil donné.*

Cette notion permet de filtrer les itemsets, pour ne garder que les plus intéressants dans le processus de fouille de données.

Les algorithmes Apriori et AprioriTid

En 1994, Agrawal & Srikant [Agrawal 94a] ont proposé deux algorithmes pour calculer les itemsets fréquents, qui ont été fondateurs dans le domaine de la fouille de données algorithmique. Les itemsets sont préalablement agencés selon un ordre lexicographique [4]. Le premier, *Apriori*, consiste, en incrémentant successivement l'ordre k, dans un premier temps, à construire, à partir de cet ordre k, les itemsets d'ordre $k+1$, puis, dans un second temps, à ne retenir que les $(k+1)$-itemsets fréquents. Cela se fait en testant chaque $(k+1)$-itemset avec toutes les transactions de la base. Le second algorithme, *AprioriTid*, améliore les performances d'Apriori, car les $(k+1)$-itemsets ne sont plus testés avec toutes les transactions de la base, mais avec un sous-ensemble de celle-ci, qui diminue au fur et à mesure que des transactions ne sont plus concernées par les itemsets. Ce second algorithme ne nécessite de lire la base de données qu'une seule fois, mais il est nécessaire qu'elle tienne intégralement en mémoire.

La complexité d'Apriori est $\mathcal{O}(nm2^m)$, où n est le nombre de transactions, et m le nombre d'attributs. Elle montre que la durée d'exécution de l'algorithme croît linéairement avec le nombre de transactions, et exponentiellement avec le nombre d'attributs.

Pour calculer les $(k+1)$-itemsets à partir des k-itemsets, les propriétés suivantes, issues de l'antimonotonicité du support [Cornuéjols 10] sont utilisées :

- Si un itemset est fréquent, alors tous ses sous-sets le sont également ;
- Si un itemset n'est pas fréquent, alors ses supersets ne le sont pas également.

L'antimonotonicité du support vient de ce que sa relation d'ordre est inversée par rapport à l'inclusion des itemsets entre eux. En effet, le support d'un k-itemset est supérieur ou égal au support d'un $(k+1)$-itemset le contenant. Ce $(k+1)$-itemset est appelé *superset* du k-itemset, qui est un de ses *sous-sets*. La littérature parle également d'itemset plus *général* au sujet d'un superset, et d'itemset plus *spécifique* pour un sous-set. Ainsi, alors que la valeur de k augmente, le support des itemsets décroît.

Pour construire un $(k+1)$-itemset, une combinaison est réalisée entre deux k-itemsets qui ne diffèrent que d'un seul élément. Par exemple, le 2-itemset $\{a,b\}$ est la combinaison des 1-itemsets $\{a\}$ et $\{b\}$. Le 4-itemset $\{a,b,c,d\}$ est la combinaison des 3-itemsets $\{a,b,c\}$ et $\{a,b,d\}$, mais également des 3-itemsets $\{a,b,d\}$ et $\{b,c,d\}$, ou $\{a,b,d\}$ et $\{a,c,d\}$, etc. Cette méthode de construction assure, grâce à l'antimonotonicité, que tous les k-itemsets fréquents seront extraits étant donné un support minimum et éventuellement une valeur maximale de k.

4. Relation d'ordre entre deux ensembles d'éléments ordonnés

Caractéristiques des itemsets

En fonction de leur support, les itemsets sont dotés de caractéristiques :

- Un itemset est dit *clos* ou *fermé* [Pasquier 99a] si aucun de ses supersets n'a de support identique au sien. Cela signifie que tous ses supersets ont un support inférieur ;
- Un itemset est dit *maximal* si aucun de ses supersets n'est fréquent ;
- Un itemset est dit *générateur* si tous ses sous-sets ont un support supérieur.

A partir de ces définitions, Bayardo & Agrawal [Bayardo 99] ont introduit la notion de *bordure* (*Border*) pour caractériser la séparation entre les itemsets fréquents et les itemsets non fréquents :

- Une bordure *positive* est l'ensemble des itemsets fréquents maximaux, c'est-à-dire qu'ils n'ont pas de supersets fréquents ;
- Une bordure *négative* est l'ensemble des itemsets non fréquents, dont les sous-sets de premier ordre inférieur sont des itemsets fréquents.

La bordure sépare donc l'ensemble des itemsets en deux sous-ensembles, celui des itemsets fréquents, et celui des itemsets non fréquents. Elle est exploitée par des algorithmes de recherche de règles d'association.

Amélioration de la recherche d'itemsets fréquents

Plusieurs algorithmes proposent d'améliorer la recherche d'itemsets fréquents. Ils répondent à des besoins de performance, ou de rapidité d'exécution, et prennent également en compte, par exemple, la possibilité de pouvoir charger intégralement la base en mémoire ou non, ou la possibilité de traiter de très longs itemsets, comme c'est le cas dans le domaine de la biologie. Nous en donnons une courte liste qui est loin d'être exhaustive.

FP-Growth. FP-Growth [Han 00, Han 04] s'appuie sur une structure de données compactes appelée *FP-tree* (Frequent Pattern Tree) créée à partir des itemsets fréquents et des transactions. Le FP-Tree est constitué d'une racine nulle, à partir de laquelle sont structurées les transactions sous la forme d'un arbre dont les nœuds correspondent aux 1-itemsets. Par ailleurs, un tableau indexe ces derniers en les triant par leur support, à partir desquels sont reliés, sous la forme d'une liste chaînée, les nœuds correspondant à chaque instance de l'item. Le parcours de l'arborescence permet ensuite de trouver les itemsets fréquents.

Partition. Partition a été proposé par Savasere et al. [Savasere 95]. La base étant divisée en partitions pouvant tenir en mémoire, dans une première phase, les itemsets fréquents sont calculés dans chacune d'elles. A l'issue, ils sont fusionnés pour élaborer des supersets potentiellement fréquents. Dans une seconde phase, les supports de ces itemsets sont calculés. Ce calcul est effectué de manière simple en utilisant les *tidlists*. Une tidlist est un ensemble constitué d'un itemset et de l'ensemble des identifiants uniques des transactions qui contiennent

cet itemset. Pour calculer le support de la réunion de deux itemsets, il suffit de considérer l'intersection des ensembles de transactions contenus dans leurs tidlists respectives. Le cardinal de cette intersection est alors le support de cette réunion.

Eclat. Eclat [Zaki 97] utilise également les tidlists. La recherche des itemsets fréquents est réalisée en profondeur et s'arrête dès qu'il n'y a plus de k-itemsets satisfaisant le support minimum. Cette méthode est très rapide, mais elle nécessite une capacité mémoire qui peut être grande, en fonction du nombre de transactions.

Echantillonnage de la base (*Sampling*). Toivonen [Toivonen 96] propose de considérer un échantillon aléatoire de la base, afin d'en extraire un sous-ensemble d'itemsets fréquents et sa bordure négative. Pour cela, il prend en compte un support inférieur au support minimum défini par l'utilisateur. Le support des itemsets du sous-ensemble est ensuite calculé, ainsi que celui des itemsets de la bordure. Si la bordure ne contient pas d'itemsets fréquents, alors l'ensemble des itemsets fréquents est correct et complet. Sinon, il faut reprendre le traitement depuis le début, afin d'obtenir tous les itemsets fréquents.

DIC. Le Dynamic Itemset Counting [Brin 97b] a pour objectif de diminuer le nombre de passes dans la base de donnée, par comparaison avec Apriori qui nécessite de balayer la base pour chaque ordre d'itemsets. Pour cela, elle est divisée en plusieurs parties séparées par un point de contrôle. Durant chaque passe, à chaque point de contrôle, les k-itemsets dont le support est supérieur au support minimum, sont utilisés pour construire les $(k+1)$-itemsets. Le support de ces derniers commence alors à être calculé durant cette même phase. Ainsi, il n'est plus nécessaire de parcourir autant de fois la base que dans Apriori. Cet algorithme demande une capacité mémoire suffisante pour être capable de traiter des itemsets de différents ordres simultanément.

Close et A-Close. Bayardo & Agrawal [Bayardo 99] montrent que les meilleures règles d'association sont extraites à partir des itemsets présents sur la bordure. Pour l'obtenir, Pasquier et al. ont proposé les algorithmes *Close* [Pasquier 99b] puis *A-Close* [Pasquier 99a] qui permettent de trouver, de manière itérative, les itemsets fréquents clos. Pour cela, ils s'appuient sur leurs itemsets générateurs. A partir de cet ensemble, il est alors possible de déduire tous les itemsets fréquents.

MaxMiner. De la même manière que l'algorithme précédent cherche les itemsets clos fréquents, MaxMiner [Bayardo 98] s'intéresse aux itemsets maximaux. A partir de cet ensemble, il est aisé de déduire tous les itemsets fréquents. Durant son exécution, MaxMiner cherche à trouver le plus tôt possible des longs itemsets fréquents pour réduire ensuite son champ de recherche. En effet, quand un long itemset maximum est découvert, alors il est inutile de poursuivre le processus avec ses sous-items, car, d'après le principe d'antimonotonicité,

tous ses sous-items sont fréquents. D'autres algorithmes exploitant l'ensemble des itemsets maximaux ont été étudiés comme, MaxEclat [Zaki 97] ou MaxClique [Zaki 97].

SSDM. Avec un autre type d'approche, Escovar et al. [Escovar 05] ont proposé SSDM (Semantically Similar Data Mining Algorithm) qui utilise de la logique floue pour extraire des itemsets fréquents. Pour ce faire, l'utilisateur doit préalablement remplir une table de similarité indiquant les items faisant partie de mêmes catégories. Par ailleurs, en plus du support, il est nécessaire de fixer un minimum de similarité qui permet à l'algorithme de choisir si deux items sont similaires ou non.

1.5.2 Règle d'association

Une règle d'association [Agrawal 93, Agrawal 94a] est définie, à partir d'un itemset $\mathcal{I}$, par la relation :

$$X \Rightarrow Y$$

où $X \cup Y = \mathcal{I}$ et $X \cap Y = \emptyset$.

Cela peut se traduire par : « Si X est présent dans la transaction, alors Y l'est également ». Notons que X et Y peuvent être composés de plusieurs attributs, mais un attribut ne peut pas figurer simultanément dans les deux parties de la règle. La partie gauche de la règle s'appelle la *prémisse* ou l'*antécédent* ou le *corps*. La partie droite s'appelle la *conclusion* ou le *conséquent* ou la *tête*.

Définition 4 *(confiance [Agrawal 93]) La confiance c de la règle d'association $X \Rightarrow Y$ est définie par :*

$$c(X \Rightarrow Y) = \frac{s(X \cup Y)}{s(X)}$$

S'agissant de la mesure de base caractérisant une règle d'association, elle peut être assimilée à la probabilité conditionnelle $P(Y|X)$, c'est-à-dire la probabilité d'avoir Y sachant X. La confiance peut donc être écrite de la manière suivante :

$$c(X \Rightarrow Y) = \frac{P(XY)}{P(X)}$$

Elle indique la validité de la règle. Pasquier et al. [Pasquier 99a] définissent la notion de *règle valide* si sa confiance est supérieure à un seuil. Elle est *approximative* si elle est inférieure à 1, et *exacte* si elle est égale à 1. Dans ce dernier cas, $P(XY)$ est égal à $P(X)$, ce qui signifie que si X est présent dans les transactions, alors Y l'est également. En d'autres termes, l'ensemble des transactions contenant X est inclus dans l'ensemble des transactions contenant Y.

Pour illustrer le support et la confiance, nous considérons la règle $(pain, saucisse) \Rightarrow hotdog$, pour laquelle ils sont égaux respectivement à 0,6 et 0,8. Les trois attributs *pain*, *saucisse* et *hot dog* sont ainsi présents simultanément dans 60% de la base de données, et 80% des transactions, contenant les attributs *pain* et *saucisse*, contiennent également l'attribut *hotdog*.

L'algorithme de recherche de règles d'association le plus connu est communément assimilé à Apriori, bien que ce dernier n'extraie que des itemsets fréquents. Il a cependant été présenté avec Apriori [Agrawal 94a], en reprenant une note de recherche [Agrawal 94b]. Agrawal & Srikant [Agrawal 94a] présentent la recherche de règles d'association en deux étapes :

- Les itemsets, dont le support est supérieur au support seuil, sont d'abord calculés à l'aide d'Apriori ou AprioriTid ;
- Pour chaque itemset retenu, toutes les règles d'association possibles sont calculées. Seules sont gardées celles dont la confiance est supérieure à la confiance seuil.

1.5.3 Mesures de qualité des règles d'association

Afin d'extraire les règles d'association d'une base de données, il est nécessaire de fixer le support seuil, pour déterminer les itemsets fréquents, ainsi que la confiance seuil, pour trouver les règles valides. En fonction du besoin de l'utilisateur, la taille maximale de la règle peut être fixée, ainsi que la taille de la prémisse ou de la conclusion. Cependant, le support et la confiance ne sont pas toujours suffisants pour trouver des règles pertinentes [Lallich 04]. En effet, en fonction des valeurs seuil, les algorithmes peuvent générer un nombre de règles très important, qui peut, dans certaines situations de seuil trop bas, dépasser le nombre de transactions initiales. De même, si le minimum est trop élevé, alors des règles intéressantes à faibles supports peuvent ne pas être détectées. De plus, ces deux mesures sont souvent insuffisantes pour prouver l'intérêt d'une règle, parce qu'elles ne prennent pas en compte $P(Y)$ ni les contre-exemples $P(X\overline{Y})$. Par exemple, si $c(X \Rightarrow Y) = P(Y)$, cela signifie que X et Y sont indépendants, parce que $P(X)P(Y) = P(XY)$. Cette règle n'est donc d'aucun intérêt, même si le support et la confiance sont élevés.

Il est donc nécessaire de caractériser les règles d'association par des mesures supplémentaires dites de *qualités* ou d'*intérêt*. Celles-ci sont nombreuses et ont fait l'objet de multiples publications [Geng 06, Guillet 07, Vaillant 05].

Piatetsky-Shapiro [Piatetsky-Shapiro 91a] a défini la notion de bonne mesure, en fonction de sa valeur par rapport à 0 [Lallich 04]. Ainsi, une bonne mesure est :

- Nulle dans le cas de l'indépendance ;
- Positive en cas d'attraction entre X et Y, c'est-à-dire dépendance positive : $P(XY) > P(X)P(Y)$;
- Négative en cas de répulsion entre X et Y, c'est-à-dire dépendance négative : $P(XY) < P(X)P(Y)$.

Lift [Brin 97a]

La mesure la plus connue est le *lift*, défini par :

$$l(X \Rightarrow Y) = \frac{P(XY)}{P(X)P(Y)}$$

Il indique la dépendance entre la prémisse et la conclusion. S'il est inférieur à 1, alors la règle est considérée sans intérêt. S'il est égal à 1, alors, comme $P(XY) = P(X)P(Y)$, X et Y sont indépendants, c'est-à-dire que la présence de l'un n'apporte rien à la présence de l'autre. Puis, plus il est élevé, plus un lien entre X et Y est probable. Ainsi, si le lift est égal à 3, cela signifie que $P(Y/X) = 3P(Y)$ et que $P(X/Y) = 3P(X)$, c'est-à-dire que si nous avons X, la probabilité d'avoir Y est trois fois plus grande que la probabilité d'avoir Y en général. Il en est de même en inversant X et Y. Il s'agit donc d'un indicateur de pertinence de la règle.

Corrélation linéaire de Pearson [Pearson 96]

Elle est définie par :

$$r(X,Y) = \frac{P(XY) - P(X)P(Y)}{\sqrt{P(X)P(\overline{X})P(Y)P(\overline{Y})}}$$

Elle permet de mesurer la force de la liaison entre X et Y. Si elle est nulle, alors cela signifie que X et Y sont indépendants. Une valeur positive forte indique que X et Y sont corrélés. Une valeur négative forte indique que X et Y sont corrélés négativement, c'est-à-dire que X et $\overline{Y}$ sont corrélés.

Loevinger [Loevinger 47]

Elle est définie par :

$$LO(X \Rightarrow Y) = 1 - \frac{P(X\overline{Y})}{P(X)P(\overline{Y})} = \frac{P(Y/X) - P(Y)}{P(\overline{Y})}$$

Cette mesure est considérée comme un indice d'écart à l'indépendance et prend la valeur nulle en cas d'indépendance. Elle augmente au fur et à mesure que le nombre de contre-exemples diminue, c'est-à-dire quand $P(X\overline{Y})$ décroît, pour atteindre la valeur 1 quand il n'y en a plus. Elle décroît avec le support, et permet de rejeter des règles peu intéressantes, malgré une confiance élevée.

Confiance centrée

Elle est définie par :

$$CC(X \Rightarrow Y) = c(X \Rightarrow Y) - P(Y) = P(Y/X) - P(Y)$$

Dans le cas de l'indépendance, la confiance est égale à $P(Y)$. En la recentrant par rapport à $P(Y)$, la confiance centrée devient alors nulle à l'indépendance, ce qui est vrai quelle que soit la probabilité de Y.

Conviction [Brin 97b]

Elle est définie par :

$$CO(X \Rightarrow Y) = \frac{P(X)P(\overline{Y})}{P(X\overline{Y})} = \frac{1-P(Y)}{1-c(X \Rightarrow Y)}$$

Brin et al. [Brin 97b] ont créé cette mesure, car le lift ne mesure qu'une cooccurrence de X et Y et pas une implication. En effet, $l(X \Rightarrow Y) = l(Y \Rightarrow X)$. La conviction est une mesure d'écart à l'indépendance où elle est égale à 1. Dans le cas où $X \Rightarrow Y$ est toujours vérifié, alors $P(X\overline{Y})$ est nulle et la conviction est infinie. Elle mesure donc bien l'implication. De plus, elle est un indicateur du nombre de contre-exemples d'une règle, car, s'il augmente, alors la conviction diminue.

Sebag et Schoenauer [Sebag 88]

Elle est définie par :

$$SS(X \Rightarrow Y) = \frac{P(XY)}{P(X\overline{Y})} = \frac{c(X \Rightarrow Y)}{1-c(X \Rightarrow Y)}$$

Comme dans le cas de la conviction, quand le nombre de contre-exemples augmente, sa valeur diminue. Si SS est égale à 3, alors $P(XY) = 3P(X\overline{Y})$, ce qui signifie qu'en cas de présence de X, le nombre de chances d'avoir Y est 3 fois plus élevé que le nombre de chance de ne pas l'avoir. Autrement dit, si X est présent, alors la quantité de chances d'avoir Y est de 75%.

A l'indépendance, la mesure est égale à $\frac{P(Y)}{P(\overline{Y})}$.

Piatetsky-Shapiro [Piatetsky-Shapiro 91a]

Cette mesure, appelée *Rule Interest* par Piatetsky-Shapiro, est définie par :

$$PS(X \Rightarrow Y) = n(P(XY) - P(X)P(Y)) = nP(X)(c(X \Rightarrow Y) - P(Y))$$

A l'indépendance, sa valeur est nulle. Comme le lift, il s'agit d'une mesure symétrique qui est donc la même que pour la règle $Y \Rightarrow X$.

Multiplicateur de cote [Lallich 04]

Il est défini par :

$$MC(X \Rightarrow Y) = \frac{P(XY)P(\overline{Y})}{P(X\overline{Y})P(Y)}$$

Cette mesure d'écart à l'indépendance est une variante de celle de Sebag & Schoenauer :

$$MC(X \Rightarrow Y) = \frac{P(\overline{Y})}{P(Y)} SS(X \Rightarrow Y)$$

Cela lui permet d'être égale à 1 à l'indépendance.

Elle peut également s'exprimer en fonction du lift et de la conviction :

$$MC(X \Rightarrow Y) = l(X \Rightarrow Y)CO(X \Rightarrow Y)$$

Zhang [Zhang 00]

Elle est définie par :

$$ZH(X \Rightarrow Y) = \frac{P(XY) - P(X)P(Y)}{Max\{P(XY)P(\overline{Y}); P(Y)P(X\overline{Y})\}}$$

Elle est nulle à l'indépendance.

Surprise [Azé 02]

Elle est définie par :

$$SU(X \Rightarrow Y) = \frac{P(XY) - P(X\overline{Y})}{P(Y)}$$

A l'indépendance, elle est égale à $-2P(X) - \frac{P(X)}{P(Y)}$.

Pearl [Pearl 88]

Le mesure est définie par :

$$PE(X \Rightarrow Y) = P(X)|P(Y/X) - P(Y)| = P(XY) \pm P(X)P(Y)$$

A l'indépendance, elle est nulle. Elle rappelle la mesure de Piatetsky-Shapiro, mais ne prend pas en compte l'effectif de la base et ne différencie pas l'attraction et la répulsion.

J-mesure [Goodman 88]

Elle est définie par :

$$JM(X \Rightarrow Y) = P(XY)log(\frac{P(XY)}{P(X)P(Y)}) + P(X\overline{Y})log(\frac{P(X\overline{Y})}{P(X)P(\overline{Y})})$$

A l'indépendance, elle est nulle. Elle prend en compte la généralité de la règle et sa capacité de prédiction [Lallich 04]. Son évolution en fonction des contre-exemples ne correspond pas à une fonction monotone. Donc il n'est pas aisé de déterminer si la règle est de meilleure qualité que son contre-exemple [Gras 10]. Une autre particularité de cette mesure est qu'elle a la même valeur pour la règle et pour son contre-exemple.

Implication [Lerman 81]

Elle est définie par :

$$IM(X \Rightarrow Y) = \sqrt{n}\frac{P(X\overline{Y}) - P(X)P(\overline{Y})}{\sqrt{P(X)P(\overline{Y})}}$$

A l'indépendance, elle est nulle. Cette mesure est utile pour étudier les contre-exemples, car elle augmente au fur et à mesure que leur nombre augmente.

Intensité d'implication [Gras 79]

Elle est définie par :

$$II(X \Rightarrow Y) = P[Poisson(nP(X)P(\overline{Y}) \geq nP(X\overline{Y})]$$

L'implication d'intensité mesure la surprise statistique de trouver la règle.

1.5.4 Conclusion sur l'approche algorithmique

Nous venons d'aborder l'approche algorithmique de la fouille de données qui consiste à extraire, d'une part les motifs fréquents, et, d'autre part, les règles d'association à partir de ces motifs. Ces motifs, ou itemsets, sont caractérisés principalement par le support qui est leur nombre d'occurrences dans la base de données.

Le nombre d'items, ou d'attributs par transaction, est dimensionnant pour la recherche des itemsets. En effet, à partir de n items, il est possible d'extraire $2^n - 1$ itemsets, allant des n 1-itemsets à l'unique n-itemset. Pour une base de transactions à 100 items, chacune d'elles donne donc lieu à $2^{100} - 1$ itemsets, soit $1,27 \times 10^{30}\ldots$ Le domaine de la biologie a déjà été évoqué. Il est amené à traiter des itemsets d'ordre élevé. Par ailleurs, le nombre de transactions dans la base de données peut être également très important, comme c'est le cas

du Grand Collisionneur de hadrons du CERN qui produit 15 pétaoctets de données par année, ou dans le domaine des réseaux sociaux, avec des données également très nombreuses qui arrivent en flot continu.

Les deux paramètres que sont le nombre d'attributs et le nombre de transactions donnent lieu à de nombreux travaux et publications. En effet, dans un but d'optimisation, il est nécessaire de réduire le nombre de lectures de la base de données. Dans d'autres situations, il peut être intéressant, voire nécessaire, de pouvoir contenir toutes les données en mémoire, pour un traitement rapide, comme avec Eclat [Zaki 97]. Afin d'optimiser cette recherche, il est alors nécessaire d'obtenir la bordure qui sépare les itemsets fréquents des itemsets non fréquents. Des algorithmes comme A-Close [Pasquier 99a] ou MaxMiner [Bayardo 98] ont été conçus dans ce but. Une fois celle-ci obtenue, il est alors rapide d'obtenir tous les itemsets fréquents.

Les règles d'association sont obtenues à partir des itemsets fréquents et sont caractérisées par la confiance, qui sert à en faire une première sélection. Elles sont écrites sous la forme $X \rightarrow Y$ ou $X \Rightarrow Y$. Cette seconde écriture suggère qu'il s'agit d'une relation d'implication de X vers Y, alors que les mesures d'intérêt montrent que ce n'est pas toujours le cas. Ainsi, le lift [Brin 97a] et le mesure de Pearl [Pearl 88] sont des mesures de qualité symétriques, car elles sont identiques pour les règles $X \Rightarrow Y$ et $Y \Rightarrow X$. L'importance donnée à cette notion d'implication conditionne donc le choix des mesures d'intérêt.

Un autre paramètre permettant de choisir la mesure est son comportement devant les contre-exemples caractérisés par $X\overline{Y}$. Selon la mesure, sa courbe en fonction des contre-exemples peut avoir plusieurs aspects. Elle peut être décroissante avec le nombre de contre-exemples, comme l'implication [Lerman 81] et Piatetsky-Shapiro [Piatetsky-Shapiro 91a]. Sa décroissance peut être très rapide dès l'apparition des contre-exemples, comme c'est le cas de la conviction [Brin 97b]. Avec la J-mesure [Goodman 88], il est plus délicat de considérer son comportement devant les contre-exemples, car son évolution n'est pas monotone. D'autres critères peuvent également être pris en compte par l'utilisateur, comme l'effectif, avec Piatetsky-Shapiro [Piatetsky-Shapiro 91a], ou la bonne mesure [Piatetsky-Shapiro 91a].

Cet état de l'art succinct sur l'approche algorithmique donne une idée de la diversité des techniques et mesures pour extraire des motifs fréquents et des règles d'association, et pour les caractériser. Les valeurs seuil et les métriques sont autant de paramètres qu'il peut être intéressant de maîtriser et d'utiliser dans la recherche de solutions à un problème donné. Pour cela, l'approche visuelle est un atout pour piloter les algorithmes. En effet, en plus d'être une technique à part entière pour explorer les données, elle est également un complément précieux de l'approche algorithmique. Ce sont ces points que nous allons aborder dans la suite de cette partie, avec le Visual Data Mining.

Chapitre 2

Approche théorique de la visualisation

2.1 La perception visuelle

La visualisation d'informations fait appel à la perception visuelle qui ne s'appuie que sur un seul sens : la vue. Elle est primordiale dans le processus d'acquisition d'information, parce qu'elle constitue le sens le plus utilisé. L'information visuelle pénètre dans l'œil par la pupille, puis traverse le cristallin, pour atteindre, en traversant la choroïde, la rétine au fond de l'œil, d'où part ensuite le nerf optique qui la transmet au cerveau. Elle est filtrée en fonction de paramètres, tels que l'acuité de l'œil ou le champ visuel, puis son traitement est finalement réalisé par le cerveau. Dans une étude récente ayant pour objectif de trouver la durée minimale nécessaire pour percevoir et comprendre une image projetée, Potter et al. ont montré que 13 millisecondes sont suffisantes [Potter 13]. Cela montre que la vue est un sens particulièrement efficient pour acquérir une information de manière rapide. Cependant, ce n'est pas systématique, car il peut exister des situations dans lesquelles cela s'avère difficile, notamment quand la perception pré-attentive, que nous allons aborder plus bas, n'est pas possible.

Afin d'appréhender la perception visuelle, ce chapitre présente des concepts qui aident à comprendre les mécanismes mis en œuvre quand il s'agit de transmettre une information, de manière visuelle, à un individu.

2.1.1 La perception pré-attentive

La perception pré-attentive d'une information visuelle est réalisée de manière automatique en détectant des caractéristiques d'objets visualisés [Treisman 85]. Grâce à elle, des formes simples peuvent émerger d'une visualisation. Des tâches pouvant être réalisées sur des représentations visuelles de plusieurs éléments en moins de 200-250 millisecondes sont considérées comme pré-attentives [Healey 12]. Comme les saccades oculaires durent au moins

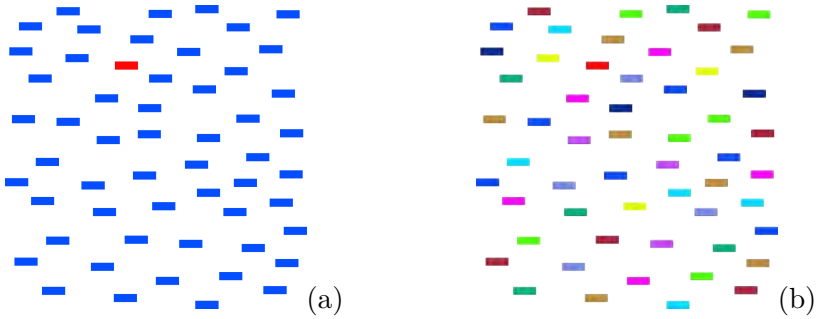


FIGURE 2.1 – Perception pré-attentive de la couleur.
(a) : la détection du trait rouge est immédiate et ne demande aucun effort. (b) : quand plusieurs couleurs sont présentes, la détection du trait fait appel à une tâche de recherche.

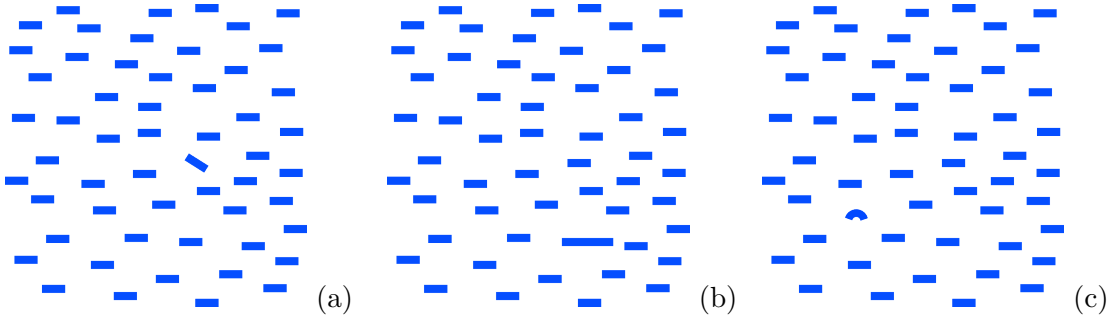


FIGURE 2.2 – Autres caractéristiques de la perception pré-attentive. (a) : orientation. (b) : longueur. (c) : courbure.

200 millisecondes, la tâche peut être réalisée d'un seul coup d'œil et donc sans effort. Ainsi, la détection du trait rouge dans la figure 2.1 (a) est immédiate et sans difficulté. La perception pré-attentive est cependant plus délicate si le nombre de couleurs devient trop important. Trouver le trait rouge de la figure 2.2 (b) requiert plus de temps, car l'utilisateur doit le chercher, et donc exercer une activité cognitive.

La perception pré-attentive s'appuie également sur d'autres caractéristiques, comme l'orientation, la longueur, la taille, la courbure et la densité [Healey 90]. La figure 2.2 en montre quelques exemples.

2.1.2 La théorie de la Gestalt

La théorie de la Gestalt est une émanation de l'Ecole de Psychologie Gestalt, fondée en 1912 pour étudier la manière dont sont perçues les formes [Koffka 35]. En effet, le mot allemand *Gestalt* signifie forme. Leur approche se plaçait sur le terrain de la psychologie et avait pour objectif de montrer que le comportement humain ne pouvait pas être considéré comme une suite de phénomènes élémentaires de type *stimulus* $\rightarrow$ *réponse*. Le travail des fondateurs de cette école est toujours d'actualité parce qu'elle fournit une description claire de phénomènes perceptuels simples [Ware 00]. Elle considère que la perception visuelle est

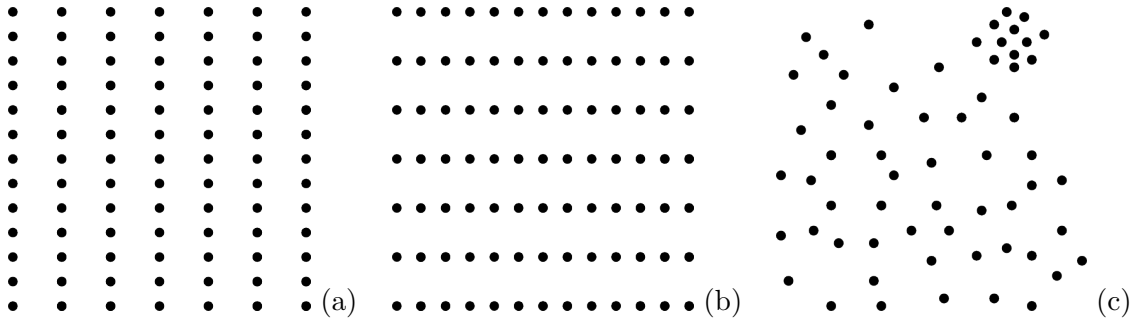


FIGURE 2.3 – La loi de proximité.
(a) et (b) : un rapprochement des points donne une impression de lignes verticales ou horizontales. (c) : la zone supérieure droite, ayant une densité homogène différente du reste de l'image, est perçue comme un groupe de points distincts.

structurée par le cerveau, en ce sens qu'elle n'est pas isolée. Le cerveau considère d'abord le tout qui est supérieur à la somme des parties [Masquelier 12]. Cela se traduit par la recherche de formes dans une information visuelle.

Les théoriciens de la Gestalt ont édicté une série de lois, ou propriétés, qui décrivent la manière dont nous percevons les formes [Ware 00], et ainsi permettent de comprendre comment sont perçues des informations présentées dans un outil d'exploration visuelle. Grâce aux lois de la Gestalt, la structure générale d'une visualisation de données, ainsi que ses particularités, peuvent être appréhendées grâce à la détection de motifs ou de groupes, d'éléments isolés et de tendances.

La loi de proximité (Proximity)

Les objets qui sont proches ont tendance à être perçus comme étant groupés. La figure 2.3 montre la différence de perception de lignes de points selon que ceux-ci sont plus rapprochés verticalement ou horizontalement. Cette proximité est exploitée dans le principe de concentration spatiale [Slocum 83] en vertu duquel les zones de points de densités homogènes sont considérées comme des groupes visuellement distincts des autres points.

La loi de similarité (Similarity)

Les attributs des objets constituant l'image déterminent la manière dont ils sont regroupés. Ainsi, des éléments ayant les mêmes attributs sont perçus comme faisant partie d'un même groupe. Ces attributs peuvent être la couleur, la forme ou l'orientation, comme illustré dans la figure 2.4, où la lettre H est aisément détectable. Ainsi, des éléments semblables ont tendance à être groupés.

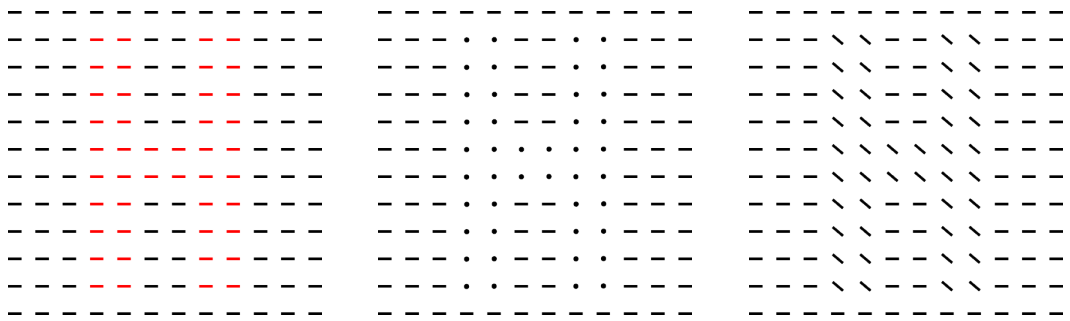


FIGURE 2.4 – La loi de similarité.
Des objets ayant les mêmes attributs sont perçus comme faisant partie des mêmes groupes. Exemple avec la couleur, la forme et l'orientation.

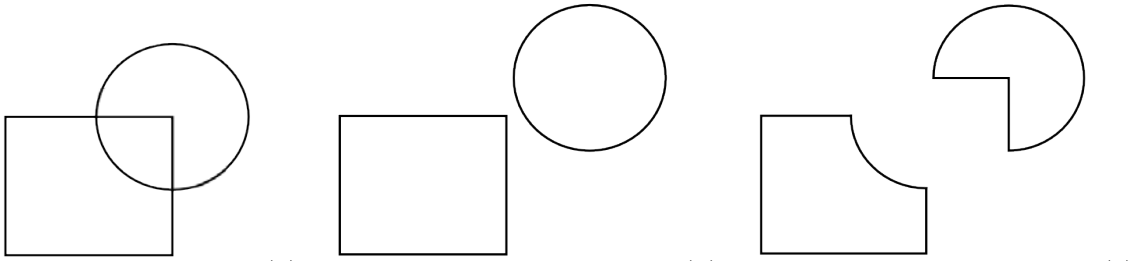


FIGURE 2.5 – La loi de continuité.
(a)(b) : il semble naturel que cela représente un cercle recouvrant un rectangle. (c) : cette possibilité n'est pas celle qui est interprétée a priori.

La loi de continuité (Continuity)

Il est plus facile de lier des éléments contigus ou rapprochés ayant des jonctions continues et lisses, plutôt que des éléments ayant des changements de direction abrupts. Ils sont alors perçus comme étant groupés, car ils contiennent des tracés dans le même prolongement. Ainsi, dans un diagramme constitué de nœuds reliés par un réseau, il est plus facile d'identifier les extrémités si le tracé est courbe plutôt que constitué d'une ligne brisée.

La loi de symétrie (Symmetry)

La symétrie (Figure 2.6) est un principe d'organisation très efficace. Des paires d'éléments présentant une symétrie sont perçus comme faisant partie d'un même groupe. S'ils sont dissymétriques, ils sont considérés comme étant séparés.

La loi de fermeture (Closure)

La perception humaine a tendance à fermer les contours qui ne le sont pas, ou à compléter des vides, afin que le cerveau ne considère qu'une forme complète. Ainsi, sur la figure 2.7 (a),

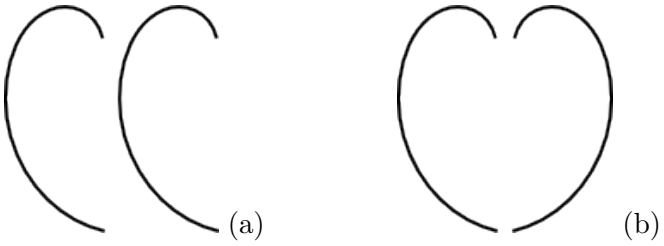


FIGURE 2.6 – La loi de symétrie.
(a) : les deux éléments sont identiques et sont perçus séparément. (b) : un élément a fait l'objet d'une symétrie par rapport à l'axe vertical. Une forme générale émerge alors de ce groupe.

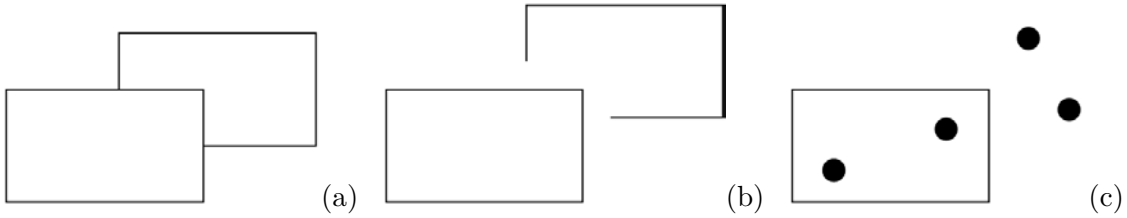


FIGURE 2.7 – La loi de fermeture.
(a) : cette figure suggère la présence de deux rectangles, dont l'un masque partiellement l'autre. (b) : le rectangle partiel n'est pas naturellement perçu. (c) : le rectangle contenant les deux disques de gauche suggère qu'ils sont perçus comme faisant partie d'un même groupe, alors que les deux disques de droite sont plus proches l'un de l'autre et sont perçus comme étant séparés.

l'observateur voit deux rectangles, alors que l'un des deux est incomplet (b). De plus, des objets étant contenus dans un autre objet les englobant, sont considérés comme faisant partie d'un même groupe. Un simple cadre, comme un rectangle, tracé autour de plusieurs objets, permet donc de les relier visuellement.

La loi de la taille relative (Relative Size)

Des composants plus petits ont tendance à être perçus comme des objets, alors que des composants plus grands peuvent être perçus comme faisant partie du fond de l'image (Cf. Figure 2.8).

La loi de l'image et du fond (Figure and Ground)

L'image est considérée comme étant ce qui est au premier plan, alors que le fond est en arrière-plan. Cette opposition entre l'image et le fond permet de les séparer et de les opposer. Ainsi, sur la figure 2.9, une forme noire est perçue devant un fond gris. Cependant, l'objet pourrait être la partie grise trouée, et le fond serait en noir.

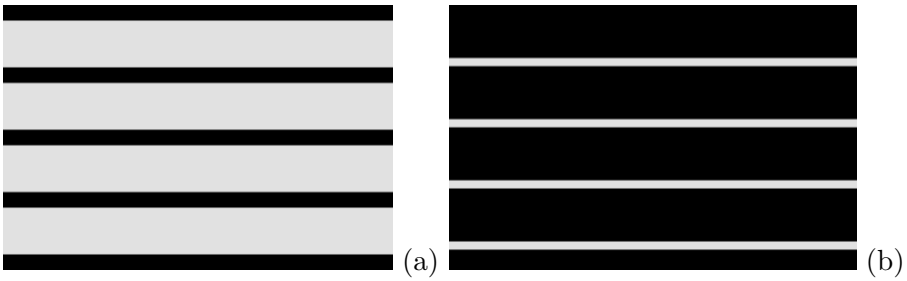


FIGURE 2.8 – La loi de la taille relative.
(a) : cette image suggère qu'il s'agit de bandes noires sur un fond gris, alors que (b) suggère l'inverse. Les lignes de plus faible épaisseur sont perçues comme des objets, et les autres comme faisant partie du fond.

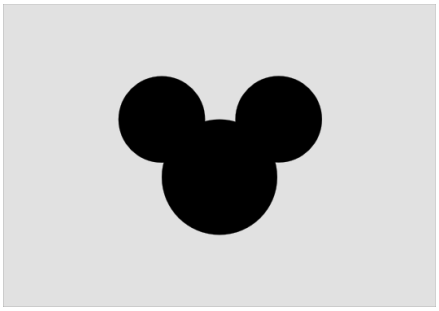


FIGURE 2.9 – La loi de l'image et du fond.
Le dessin suggère un personnage célèbre sur fond gris.

La loi du destin commun (Common Fate)

Cette loi s'applique aux éléments en mouvement. De tels objets, ayant des caractéristiques cinétiques identiques ou très proches, comme la vitesse et la direction, sont perçus comme faisant partie d'un même groupe.

2.2 La sémiologie graphique

La sémiologie graphique [Bertin 67], publiée par Jacques Bertin en 1967, apporte une réponse objective aux deux questions suivantes : dans quel cas faut-il faire un dessin ? Quel dessin faut-il faire ? Elle définit les propriétés spécifiques de la représentation graphique par rapport aux autres systèmes de signes et permet de déterminer, dans chaque cas, la meilleure transcription d'une information. Elle est née dans un contexte où la cartographie était un facteur clé dans la transmission et le maintien de la connaissance. Cette époque est également celle de l'essor de la géographie quantitative qui a vu une réorientation de cette discipline en lien avec les mathématiques et l'arrivée des calculateurs [Pumain 02].

L'approche de Jacques Bertin s'appuie d'une part sur la perception visuelle, et, d'autre part, sur les différents types de significations que l'homme peut attribuer à un signe. La signification peut ainsi être :

- monosémique : une seule interprétation est possible ;
- polysémique : plusieurs interprétations sont possibles ;
- pansémique : toute interprétation est possible.

La sémiologie graphique est un système monosémique, car la connaissance des signes doit être un préalable à l'interprétation de leur agencement dans un graphique. En d'autres termes, avant de comprendre un graphique, il est nécessaire d'en connaître la légende et la signification des signes qu'il contient (Cf. Figure 2.10).

Bertin définit sept variables visuelles en établissant leurs propriétés perceptives, afin de de pouvoir transmettre de l'information par un graphique, qui sera lisible et interprétable. En s'appuyant sur leurs propriétés issues de la perception et de la théorie de la Gestalt, des particularités peuvent apparaître dans un graphique, comme des corrélations entre éléments, par des regroupements, ou des oppositions, l'isolement d'autres éléments, etc. Les variables visuelles sont les suivantes :

- La position ;
- La taille ;
- L'orientation ;
- La forme ;
- La valeur ;
- Le gain ;
- La couleur.

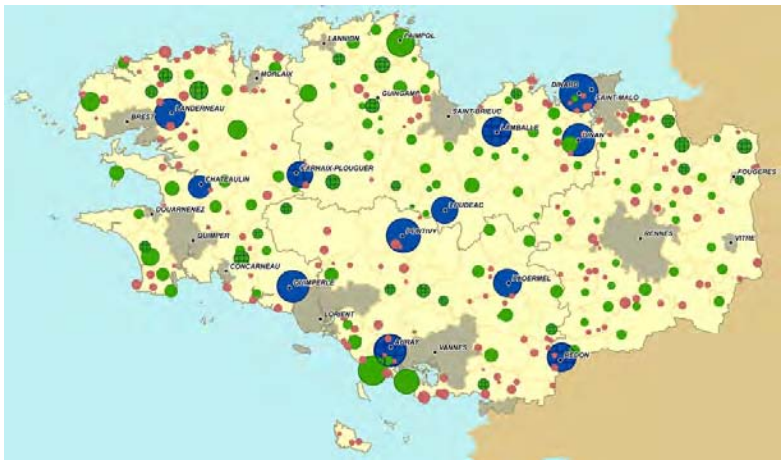


FIGURE 2.10 – Une carte inexploitable.
Elle ne relève pas d'un système monosémique : la légende étant absente, il n'est pas possible d'en comprendre la signification.

Elles sont à considérer dans un contexte de cartographie ou de photographie et sont utilisées dans trois types d'implantation dans le plan, qui sont le point, la ligne et la zone.

Le graphique présente des relations entre les différents éléments qui le composent. La perception des variables visuelles correspond à quatre niveaux d'organisation :

- Le *niveau associatif* ($\equiv$). Il correspond aux variables nominales, ou qualitatives, dites de séparation, en ce sens qu'elle permettent de séparer les éléments entre eux, indépendamment des autres variables. Elles ont une visibilité constante. Par exemple, la forme est associative car trouver des losanges dans une image peut être fait quelle que soit la couleur ou leur orientation. Si la variable n'est pas associative, alors elle est dissociative ($\not\equiv$) et a une visibilité variable. Il s'agit de la taille et de la valeur ;
- Le *niveau sélectif* ($\neq$). Il permet d'isoler sans effort tous les éléments d'une même catégorie nominale. Il s'agit par exemple de la catégorie des éléments rouges ;
- Le *niveau de l'ordre* ($\mathbf{O}$). Il concerne les éléments pouvant être ordonnés, c'est-à-dire qu'il est possible de dire quel tel élément est plus ou moins que tel autre, selon un critère défini, comme une mesure ou une appréciation ;
- Le *niveau quantitatif* ($\mathbf{Q}$). Il est associé à une métrique. Un élément est donc caractérisé par une valeur comptable. Par exemple, ce billet vaut 20 Euros, ou la surface de ce champs est de 33 hectares. La notion de distance est associée à ce niveau.

Ces niveaux sont emboîtés. Ainsi, le quantitatif est ordonné et qualitatif, ce qui est ordonné est qualitatif, et ce qui est qualitatif est ordonnable. La figure 2.11 présente les variables visuelles, ainsi que leur niveau d'organisation. Elles sont divisées en deux parties. D'une part, les variables de l'image permettent de l'élaborer, notamment en représentant des formes. Le second groupe est constitué de variables de séparation des images, qui sont associatives. Elles servent essentiellement à séparer les éléments de l'image et interviennent en complément des

autres variables pour l'enrichir.

La variable répondant à tous les niveaux d'organisation est la position dans le plan, ce qui lui procure le plus grand pouvoir d'expression graphique. Grâce aux nouvelles technologies, les représentations en trois dimensions ont introduit une nouvelle valeur spatiale, qui peut être représentée par projection de la vue en 3D dans le plan, ou par des dispositifs, comme des lunettes ou des casques de réalité virtuelle. Cette nouvelle dimension présente l'avantage de pouvoir offrir une plus grande expression graphique, mais des effets, comme l'occultation, sont à prendre en compte et peuvent se révéler pénalisants dans une visualisation.

La couleur n'est pas une variable ordonnée. En effet, il n'est pas possible de dire que le rouge est plus que le jaune. Cependant, en utilisant un gradient de couleur pour caractériser une valeur, par exemple du jaune au bleu, elle devient ordonnée, car il est alors possible de dire que telle valeur de couleur correspond à une valeur numérique supérieure ou inférieure à telle autre. Le gradient de couleur est à rapprocher de la variable visuelle *valeur*, qui, en cartographie, varie du blanc au noir, en passant par des niveaux de gris. De plus, dans le cadre de l'affichage sur écran, la valeur serait également à rapprocher de la luminosité, le grain de la texture et la couleur des composantes *teinte* et *saturation*.

Une nouvelle composante, inexistante sur une carte imprimée mais présente et fortement exploitée dans les systèmes de visualisations informatiques actuels, est la transparence, qui a été ultérieurement ajoutée à la liste des variables visuelles par Wilkinson [Wilkinson 99]. Elle est à rapprocher de la valeur, mais constitue une variable visuelle difficilement exploitable, car ses variations ne sont pas toujours détectables. De plus, un objet transparent peut laisser apparaître un autre objet en arrière-plan, ce qui entraîne un effet combinatoire entre eux, et ainsi fausser la perception de la transparence. La valeur de la transparence est l'alpha, qui correspond plus précisément à l'opacité. En effet, une valeur alpha nulle indique une transparence maximale, alors qu'un alpha maximum signifie que la transparence est nulle.

En plus de son niveau d'organisation, une variable visuelle est caractérisée par sa *longueur*. Il s'agit du nombre d'éléments différentiables qu'elle permet d'identifier. Elle est parfois confondue avec l'*étendue* d'une variable quantitative qui est le rapport entre la valeur la plus grande et la valeur la plus petite. Longueur et étendue procurent une échelle de perception de la variable visuelle.

2.3 La caractérisation des visualisations

Afin de comprendre comment la visualisation va être exploitée dans l'élaboration des règles d'association, nous commençons par la caractériser. Cette caractérisation sera également exploitée pour générer des visualisations à partir des règles. Pour cela, nous nous appuyons sur la perception visuelle, la sémiologie graphique [Bertin 67] et la caractérisation des visualisations à l'aide du modèle de Card & Mackinlay [Card 97]. Ceux-ci sont partis de la sémiologie graphique, étendue par Mackinlay [Mackinlay 86], et ont mis au point un modèle considérant qu'une visualisation est constituée de :

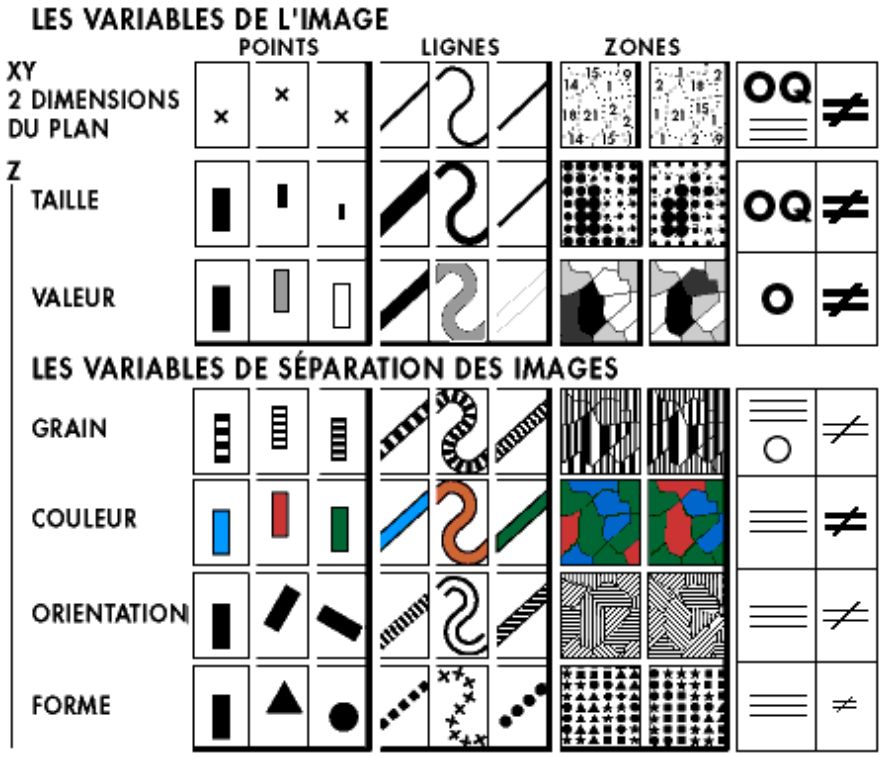


FIGURE 2.11 – Les variables visuelles de la sémiologie graphique, et leurs niveaux d'organisation.

- un ensemble de signes (*marks*), qui peuvent être des points, des lignes, des zones, des surfaces ou des volumes ;
- une position dans l'espace et dans le temps. Il s'agit de X et Y dans le plan, et X, Y, Z dans l'espace. A cela est ajouté le temps T ;
- un ensemble de variables rétiniennes, qui sont la couleur, la taille, la forme, le niveau de gris, l'orientation et la texture ;
- des propriétés de connexion ;
- un contenant (*enclosure*). Il prend en compte la notion d'encapsulation ;
- des propriétés de perception contrôlée. Elles permettent de préciser si l'acquisition de la donnée est traitée par le système perceptif, donc à faible charge cognitive éventuellement nulle, ou si elle est contrôlée. Dans ce cas, cela requiert une forte charge cognitive, comme lors de la lecture de texte.

Afin de pouvoir réaliser des comparaisons entre les visualisations, Card & Mackinlay utilisent un tableau dans lequel sont indiquées les propriétés précitées (Cf. Tableau 2.1), les valeurs utilisées étant présentées dans le tableau 2.2. Chaque ligne correspondant à une donnée en entrée, il contient trois groupes principaux de colonnes :

- les données. La colonne D contient les données initiales qui sont de type nominal, ordonné ou quantitatif. Par une fonction de transformation F, ces données donnent lieu à un sous-ensemble de données F' ;

Data				Automatic perception							Controlled perception
Variable	D	F	D'	X	Y	Z	T	R	-	[ ]	CP

TABLE 2.1 – Le modèle de Card & Mackinlay

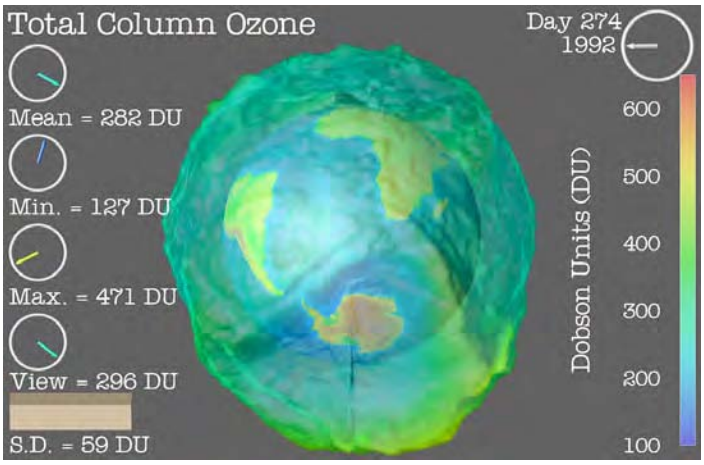


FIGURE 2.12 – Visualisation de la couche d'ozone (d'après [Card 97]).

- les variables de perception automatique, correspondant à la perception visuelle. Elles émanent de la sémiologie graphique [Bertin 67]. Les liens, ou connexions entre les éléments graphiques, sont précisés dans la colonne '-'. Les données sont souvent reliées par des lignes (L). L'encapsulation [ ] correspond par exemple au cas du Treemap [Johnson 91] qui est une représentation de données hiérarchiques sous la forme d'un plan récursif (Cf. Chapitre 3.1.2) ;
- la perception contrôlée.

Ce modèle a été illustré avec plusieurs types de visualisation, comme la visualisation scientifique, le scattergraph multidimensionnel, les tables multidimensionnelles, les paysages et les espaces d'informations, les arbres, etc. L'illustration de la figure 2.12 est intéressante pour notre contexte de données aéronautiques, parce qu'elle s'appuie sur des données spatiales et géographiques. Elle présente une animation de la couche d'ozone inspirée des travaux de Lloyd Treinish [Treinish 94] [1]. Le tableau 2.3 contient les coordonnées géographiques de type QX et QY, ce qui signifie que ce sont des données quantitatives spatiales. Comme il s'agit de coordonnées, QX et QY sont alors respectivement notés QXlon et QYlat. La notation, pour signifier que $QXlon$ est assigné à la variable visuelle X sous la forme d'un point, est noté $QXlon \rightarrow X : P$. De même, il est noté pour la latitude $QYlat \rightarrow Y : P$, et pour la hauteur $QZ \rightarrow Z : P$. La densité d'ozone est assignée à la variable rétinienne de type couleur qui est ici ordonnée, étant donné le gradient présenté dans la partie droite de l'image. Le tableau permet de faire, de manière simple, le lien entre la donnée et sa représentation dans une visualisation. Cependant, pour une visualisation dont l'affectation des variables visuelles peut

1. http ://www.research.ibm.com/people/l/lloydt

Symbole	Signification
D	Types de données : • N : nominale • O : ordonnée • Q : quantitative • QX : quantitative et intrinsèquement spatiale • QXlon : géographique • NxN : nominale, avec la même variable sur deux axes
F	Filtre ou fonction d'encodage de la donnée : • f : filtre non spécifié • fs : tri • mds : échelle multidimensionnelle • > : filtre de réduction de données, à l'aide des curseurs glissants ou des menus • sl : curseur glissant
D'	Type de données modifié
XYZT	Position spatio-temporelle
*	Utilisation non sémantique de l'espace-temps
R	Propriétés rétiniennes : • C : couleur • S : taille
—	Connexion
[ ]	Encapsulation
CP	Perception contrôlée
P,L,A,S,V	Types de signes : • P : point • L : ligne • S : surface • A : zone • V : volume

TABLE 2.2 – Les valeurs utilisées dans le tableau de Card & Mackinlay

Data				Automatic perception							Controlled perception
Variable	D	F	D'	X	Y	Z	T	R	-	[ ]	CP
Lon	QXlon	f	QXlon	P							
Lat	QYlat	f	QYlat		P						
Height	QZ	f	QZ			P					
Ozone	Q		O					C			

TABLE 2.3 – Table correspondant à la visualisation de la couche d'ozone de la figure 2.12.

changer, il devient nécessaire d'utiliser d'autres tableaux pour ces nouvelles affectations. Par exemple, dans une vue représentant une projection dans le plan, d'un ensemble de données tridimensionnelles, chaque projection, dans les plans XY, XZ et YZ, nécessite un tableau différent. De plus, il comporte une fonction de transfert f qui est insuffisamment prise en compte dans le modèle. Il peut s'agir par exemple de données sélectionnées à l'aide d'un opérateur dans une vue selon un tableau, puis à nouveau sélectionnées dans une autre vue correspondant à un autre tableau, à partir de laquelle vont être réalisées des opérations de type zoom et excentrement.

Chapitre 3

Visual Data Mining

3.1 Fouille visuelle des données

L'objectif de la fouille visuelle des données, ou *Visual Data Mining*, est d'aider l'utilisateur à les appréhender, pour détecter des informations intéressantes, et pour en acquérir une connaissance profonde [Simoff 08]. Elle fait appel à deux notions : la visualisation des informations et leur manipulation. Elle relève donc du domaine de l'InfoVis (*Information Visualization*), défini par Card et al. [Card 99] comme l'utilisation de représentations visuelles de données abstraites, interactives et assistées par ordinateur pour amplifier la cognition, cette dernière étant considérée comme l'acquisition ou l'utilisation de connaissances.

Les premières représentations de données semblent remonter à William Playfair, ingénieur et économiste écossais, qui a introduit de nouveaux types de graphiques dès 1785 [Playfair 86] [1]. Pour un autre ouvrage paru en 1801 [Playfair 01], il lui a été demandé de trouver une méthode pour « faire servir » les statistiques. Il l'a ainsi rédigé dans le but d'être « à la portée de presque toutes les classes de la société », en considérant que « le plus sûr moyen de frapper l'esprit, est de parler aux yeux » [Playfair 02]. La figure 3.1, extraite de la traduction française [Playfair 02] de cet ouvrage, contient un diagramme circulaire dont la taille est proportionnelle à une valeur représentée. Bien plus tard, des travaux, comme ceux de Bertin [Bertin 67, Bertin 83], Tufte [Tufte 83, Tufte 90], Tukey [Tukey 77] et Cleveland & MacGill [Cleveland 88] ont permis d'asseoir les bases de la visualisation d'informations. Bertin est le père de la sémiologie graphique [Bertin 67, Bertin 83] (Cf. Chapitre 2.2). Tufte a étudié les visualisations scientifiques et abstraites, notamment en communication visuelle de l'information. Il a introduit par exemple la notion de densité de données d'un graphique (*Data density of a graphic*) [Tufte 83]. Tukey [Tukey 77] a introduit le concept EDA (Exploratory Data Analysis) en caractérisant les ensembles de données statistiques par des méthodes visuelles. Ses travaux ont contribué à l'élaboration de graphiques standard présents dans beaucoup de logiciels grand public comme Microsoft Excel [Oleg Sindiy 13]. Cleveland [Cleveland 85, Cleveland 93]

1. La première version auto-éditée date de 1785. La seconde, généralement référencée, fut éditée par James Corry un an plus tard.

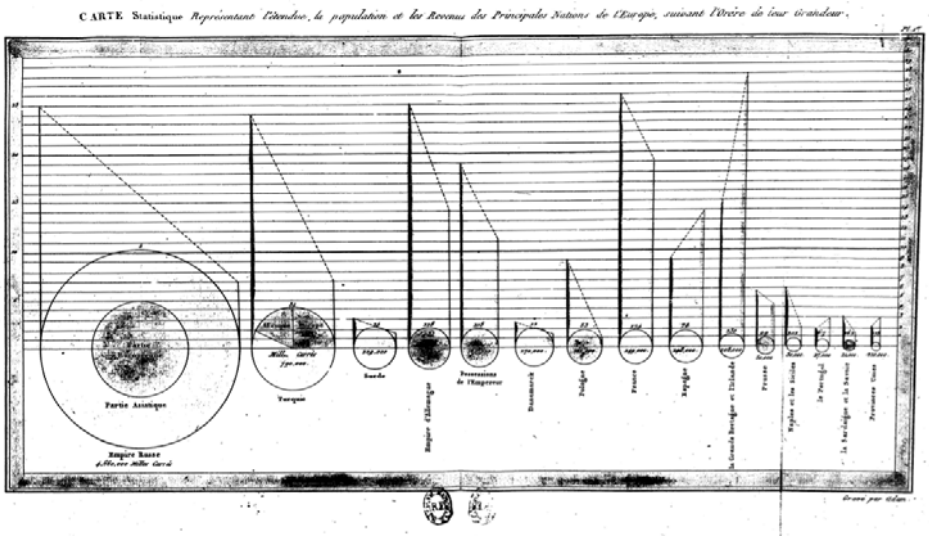


FIGURE 3.1 – Carte statistique de William Playfair [Playfair 01].
Cette carte, traduite par Donnant [Playfair 02], montre l'étendue, la population et les revenus des principales nations d'Europe. L'étendue de chaque pays est représentée par un cercle qui lui est proportionnel.

a, quant à lui, étudié les méthodes de visualisations de données.

Le domaine de la visualisation d'informations est récent. Son essor est dû essentiellement aux progrès technologiques, par les possibilités d'affichage et de performances des calculateurs, que ce soit par la puissance des CPU, mais également par celle des GPU des cartes graphiques auxquelles est maintenant dévolue une partie des traitements. Cela est abondamment exploité dans l'industrie du jeu vidéo. Cependant, la visualisation des données existait déjà dans les domaines de la cartographie et de la visualisation scientifique. Une distinction est faite entre la visualisation d'informations et la visualisation scientifique ou *SciVis* (*Scientific Visualization*), qui est une discipline beaucoup plus ancienne. Celle-ci s'applique aux données scientifiques basées sur des phénomènes ou des objets physiques, souvent à caractère spatio-temporel, par opposition aux données abstraites, sans références spatiales, qui sont plus l'apanage de l'InfoVis, comme des données commerciales, bancaires, ou issues de réseaux sociaux.

Nous avons évoqué, dans le chapitre 1.4, le rôle grandissant que prend le Visual Analytics. Ce domaine est plus que de la visualisation [Keim 08a], car il se présente comme une approche globale pour l'aide à la décision, combinant la visualisation, les facteurs humains et l'analyse des données. Ainsi, même si nous abordons la fouille visuelle de données selon l'angle de l'InfoVis, nous ferons régulièrement référence à des travaux issus du Visual Analytics qui apportent une ouverture très enrichissante et prometteuse par sa pluridisciplinarité.

Selon Shneiderman [Shneiderman 96], le mantra de la recherche d'information visuelle est résumé en ces termes : « Overview first, zoom and filter, then details on demand. » Ainsi, l'utilisateur doit avoir en premier lieu une vision globale de la visualisation, afin d'en dégager la structure, les tendances, les caractéristiques essentielles et les zones de données intéressantes

et celles qui ne le sont pas. Pour poursuivre son exploration, il est ensuite amené à étudier certains aspects dans le détail. Selon Keim et al. [Keim 03, Keim 04], durant ce processus, il est important de garder une vision globale de l'espace de données, même quand l'utilisateur en étudie une partie. Cela peut se faire par une distorsion de la vue générale, afin de se focaliser sur une partie, ou en dédiant la plus grande surface d'affichage à la partie des données à étudier, tout en atténuant l'espace pour la vue générale. La présentation visuelle des données fait également appel à l'intuition et suggère à l'utilisateur les directions possibles pour la poursuite de leur étude [Shneiderman 01].

La fouille visuelle des données peut être réalisée selon trois types d'approche permettant d'intégrer l'homme dans la boucle [Keim 04] :

- Visualisation préalable (*Preceding Visualization = PV*). Les données sont explorées visuellement avant la mise en œuvre d'un algorithme. Cela permet de découvrir des motifs intéressants ;
- Visualisation a posteriori (*Subsequent Vizualisation = SV*). Un algorithme est préalablement utilisé afin d'extraire des motifs. Ceux-ci sont ensuite visualisés pour être analysés par l'utilisateur. En fonction de cette visualisation, l'utilisateur peut être amené à modifier le paramétrage de l'algorithme pour l'exécuter une nouvelle fois ;
- Visualisation étroitement intégrée (*Tightly Integrated Visualization = TIV*). Un algorithme analyse les données, mais ne donne pas le résultat final. Les résultats intermédiaires sont cependant visualisés et permettent à l'utilisateur de détecter des motifs intéressants, en fonction de son domaine de connaissance. Comme un algorithme ne peut pas convenir à toutes les situations, son choix est réalisé par l'utilisateur, et les résultats sont ainsi adaptés à son domaine. Ce processus peut être réitéré jusqu'à obtention d'un résultat.

Ces approches ont été reprises et développées dans le cadre du Visual Analytics, par Bertini & Lalanne [Bertini 09], qui présentent les notions de visualisation améliorée (*Enhanced Visualization*), de fouille améliorée (*Enhanced Mining*) et de visualisation et fouille intégrées (*Integrated Visualization & Mining*). Dans ce troisième cas, ils présentent deux modèles d'intégration. Avec le premier, appelé *White Box*, l'homme et la machine coopèrent durant toutes les étapes de l'élaboration du modèle. L'utilisateur peut ainsi orienter le processus, notamment grâce aux visualisations. Le second, est appelé *Black Box*. Avec celui-ci, l'algorithme est considéré comme une boîte noire, et il n'est possible que de le paramétrer, pour en voir les résultats sur un outil de visualisation.

Dans la suite de cette partie, nous allons aborder la fouille visuelle des données, selon deux points de vue. Tout d'abord, nous considérerons les données en tant que telles. Puis nous étudierons la visualisation des résultats des algorithmes, c'est-à-dire des itemsets et des règles d'association.

La visualisation d'informations et les techniques de Visual Data Mining sont basées sur le type de données visualisées, les techniques de visualisation et les techniques d'interaction [Keim 03]. Elles ont fait l'objet d'une classification [Keim 03, Keim 04]. Selon Keim, afin que l'exploration visuelle soit efficace, il est important d'inclure l'homme dans ce processus, pour

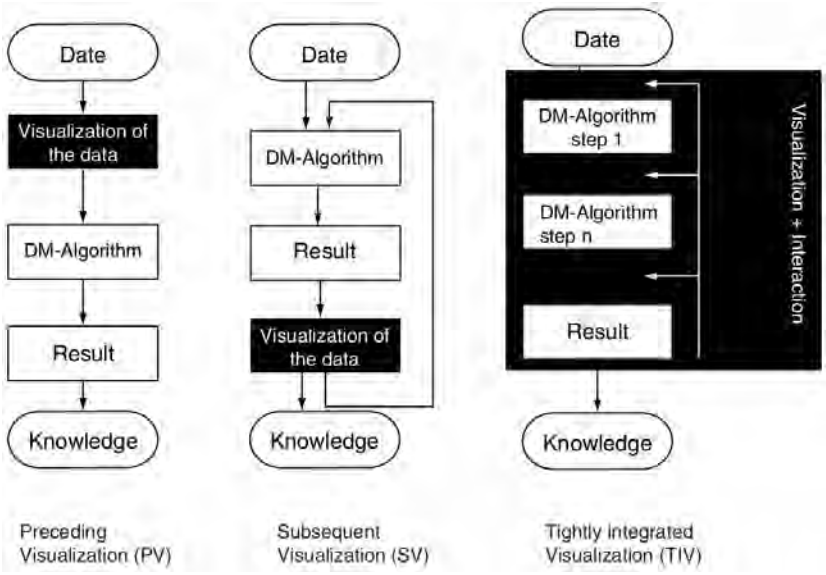


FIGURE 3.2 – Trois types d'approche pour la fouille visuelle des données [Keim 04].

combiner la flexibilité, la créativité et sa connaissance, avec la capacité de stockage et de calcul des systèmes informatiques. Ce concept constitue un des éléments clé du Visual Analytics. Le but est donc de présenter les données de telle manière que l'utilisateur puisse les explorer et les manipuler pour en tirer des conclusions et éventuellement formuler de nouvelles hypothèses. Les avantages qu'il expose par rapport à la fouille automatique des données sont les suivants :

- L'exploration visuelle peut être réalisée sur des données non-homogènes et bruitées ;
- Elle est intuitive et ne requiert aucune connaissance mathématique ou statistique complexe ;
- La visualisation peut procurer une vue générale et qualitative des données, permettant d'en isoler une partie afin de poursuivre l'analyse.

De plus, l'exploration visuelle peut être rapide et efficace, particulièrement quand les algorithmes échouent, et procure une plus grande confiance dans les résultats trouvés.

Une classification des techniques de Visual Data Mining a été proposée par Keim & Ward [Keim 03], selon trois critères (Figure 3.3) :

- Le type de données à visualiser ;
- La technique de visualisation ;
- La technique d'interaction avec les données.

Cette classification, que nous allons étudier dans la suite de ce chapitre, est dite orthogonale, en ce sens que n'importe quelle technique issue d'un critère peut être combinée à n'importe quelle autre issue d'un autre critère.

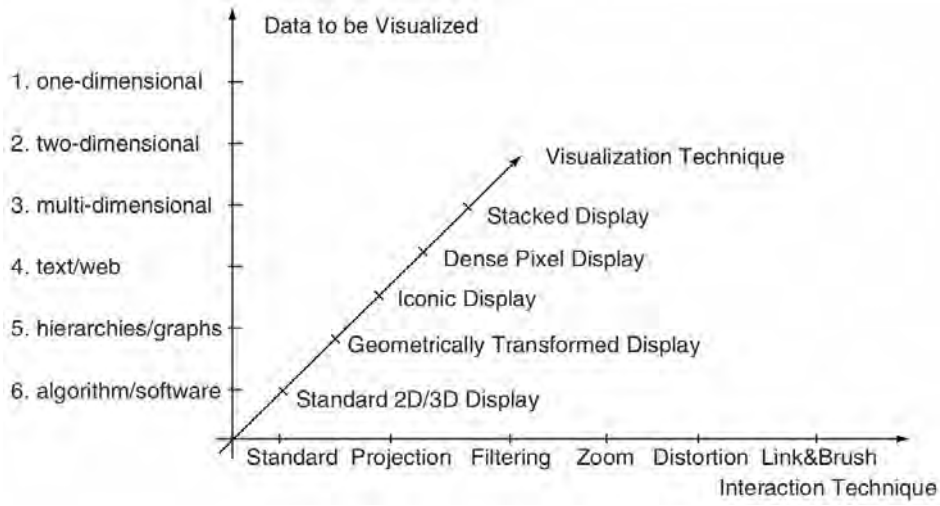


FIGURE 3.3 – Classification orthogonale des techniques de Visual Data Mining [Keim 03].

3.1.1 Les types de données

Nous avons abordé la notion de donnée dans le chapitre 1.1.1. Le type de donnée traité dans la visualisation d'information correspond à des enregistrements, chacun souvent constitué de multiples attributs. Le nombre d'attributs constitue alors le nombre de dimensions de la donnée.

Données monodimensionnelles

Il s'agit typiquement des données temporelles, pour lesquelles, à chaque point de temps, correspond une ou plusieurs données. Un autre exemple est l'annuaire contenant une liste de personnes, ayant un seul numéro de téléphone.

Données bidimensionnelles

Il s'agit par exemple des données géographiques, les deux dimensions étant la longitude et la latitude. La figure 3.4 montre la trajectoire du circuit Paul Ricard, où chaque point est défini par des coordonnées dans le plan.

Données multidimensionnelles

Hormis les données tridimensionnelles, qui peuvent être présentées selon une visualisation spatiale 3D, il s'agit ici des données contenant plus de trois attributs, leur nombre allant jusqu'à plusieurs dizaines voire centaines. Il s'agit alors de trouver des techniques permettant de visualiser plusieurs attributs à la fois. L'une d'entre elles est l'utilisation de coordonnées parallèles.

FIGURE 3.4 – Exemple de visualisation de données bidimensionnelles.
Le tracé du circuit Paul Ricard.

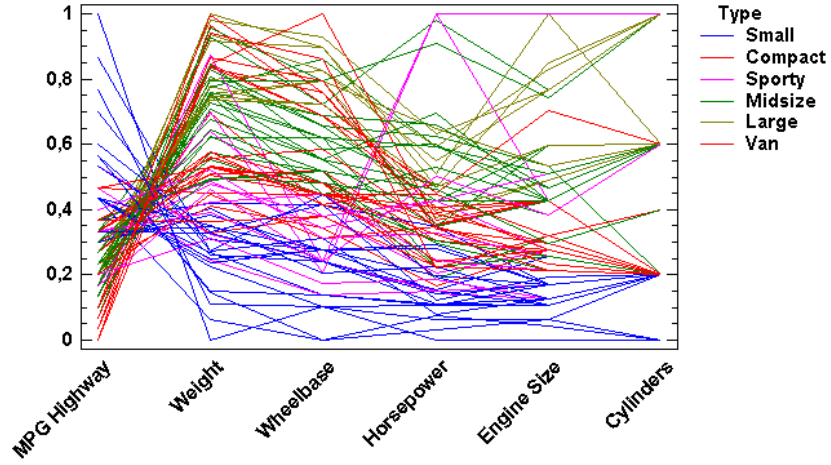


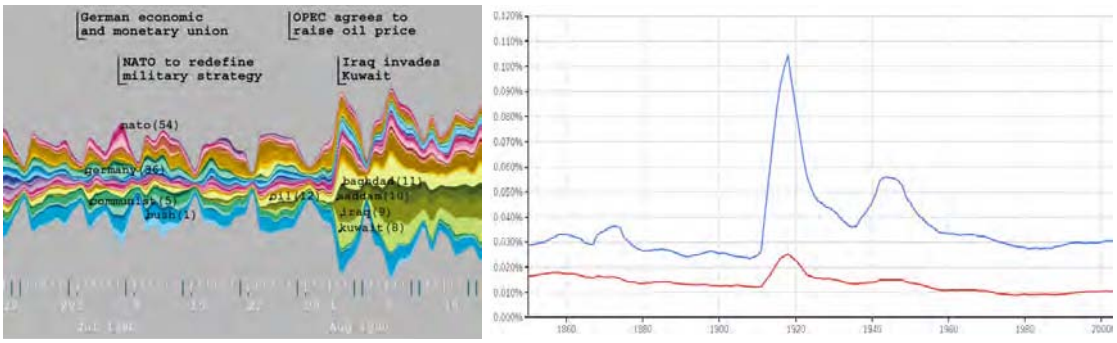
FIGURE 3.5 – Exemple de visualisation multidimensionnelle avec les coordonnées parallèles.

Les coordonnées parallèles ont pour but de visualiser un nombre important de données, en reliant des axes parallèles, chacun représentant une dimension d'un espace multidimensionnel. Ainsi, la présentation d'un espace multidimensionnel est ramené à une représentation bidimensionnelle. Ce concept a fait l'objet d'un première publication en 1885 par Maurice d'Ocagne [d'Ocagne 85], puis il a été réintroduit par Inselberg [Inselberg 81] un siècle plus tard. La figure 3.5 montre des caractéristiques de types de véhicules selon six attributs [2]. Chaque attribut est considéré sous une forme normalisée pour permettre de garder la même valeur visualisée. Pour chaque véhicule, une ligne brisée relie successivement les axes verticaux parallèles à la hauteur correspondant à la valeur.

Données textuelles

Malgré l'importance croissante de l'image dans notre quotidien, la donnée textuelle est restée très présente. En effet, il n'est pas toujours possible de représenter un texte sous forme graphique ou chiffrée. En considérant par exemple un discours, sa structure peut être dessinée à l'aide d'un diagramme heuristique [3], ou des idées peuvent être illustrées par des

2. www.statgraphics.fr
3. www.thinkbuzan.com



(a) (b)

FIGURE 3.6 – Visualisation de l'évolution de l'occurrence de mots en fonction du temps. (a) : à partir d'une base de documents, avec ThemeRiver [Havre 00]. (b) : avec Ngram Viewer.

images, des photos ou des graphes. Mais le discours en tant que tel ne peut pas être reproduit graphiquement. Afin d'obtenir une visualisation à partir de textes, il est nécessaire de procéder à une transformation. Ainsi, dans ThemeRiver [Havre 00], Havre et al. visualisent l'évolution du nombre d'occurrences de mots, à partir d'une grande quantité de documents (Cf. Figure 3.6 (a)). Une bande colorée épaisse signifie que le mot correspondant est fortement utilisé durant la période correspondante. Fort de ses 30 millions de livres numérisés [4], Google a lancé en 2010 le service Ngram Viewer [5] reprenant le principe de ThemeRiver. Il permet de visualiser l'évolution de l'apparition d'un mot dans les ouvrages, au cours du temps. La figure 3.6 (b) montre ainsi l'évolution des mots *paix* (rouge) et *guerre* (bleu). Il est facile d'y détecter une plus forte apparition du mot *guerre* à l'occasion de la Guerre de 1870 et des Guerres Mondiales.

Hiérarchies et graphes

Les liens entre les données peuvent être ordonnés et hiérarchiques. Ils sont souvent représentés par des graphes, caractérisés par des nœuds reliés par des arêtes. Il s'agit d'une problématique à part entière, qui donne lieu à des conférences spécifiques annuelles, telles que Graph Drawing [6].

Algorithmes et programmes

Les algorithmes et les programmes constituent une autre catégorie de données, dont la visualisation peut permettre d'en faciliter la compréhension, en visualisant la structure ou en montrant les liens entre les différentes parties. Egalement utilisée pour le debug de programmes, cette technique fait l'objet de multiples recherches, comme CodeCity [Wettel 11] ou sv3D [Marcus 03]. Sur la figure 3.7 (b), le programme mailbox.cpp est illustré avec sv3D. Chaque

4. En avril 2013
5. http ://fr.wikipedia.org/wiki/NgramViewer
6. http ://www.graphdrawing.org

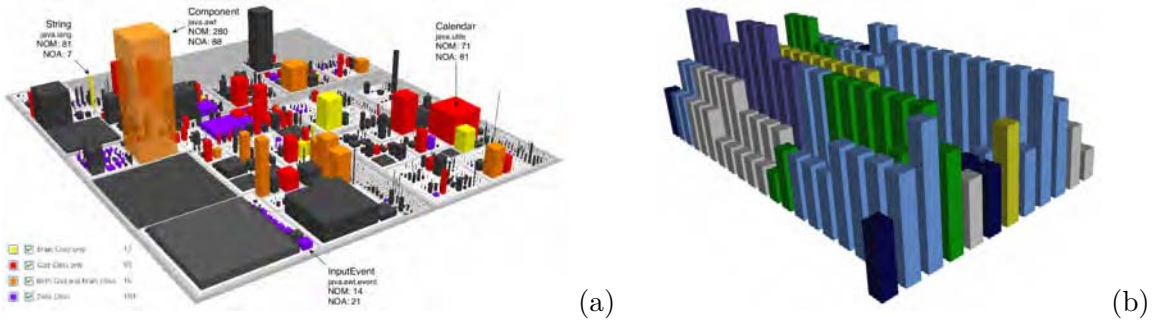


FIGURE 3.7 – Visualisation de logiciels.
(a) une partie du JDK 1.5, avec CodeCity [Wettel 11]. (b) : mailbox.cpp par sv3D [Marcus 03].

ligne de code est représentée par un cylindre. La couleur correspond au type de mots clé (*if*, *then*, *else*, *for*, etc.) et la hauteur au niveau d'imbrication.

3.1.2 Les techniques de visualisation

De nombreuses techniques de visualisation sont étudiées et utilisées dans les logiciels. Il s'agit de présentations de données multidimensionnelles, faisant l'objet de transformations géométriques, comme les coordonnées parallèles, de visualisations à base d'icônes, d'affichages empilés, comme les Treemaps, etc. Nous en présentons quelques unes dans ce paragraphe.

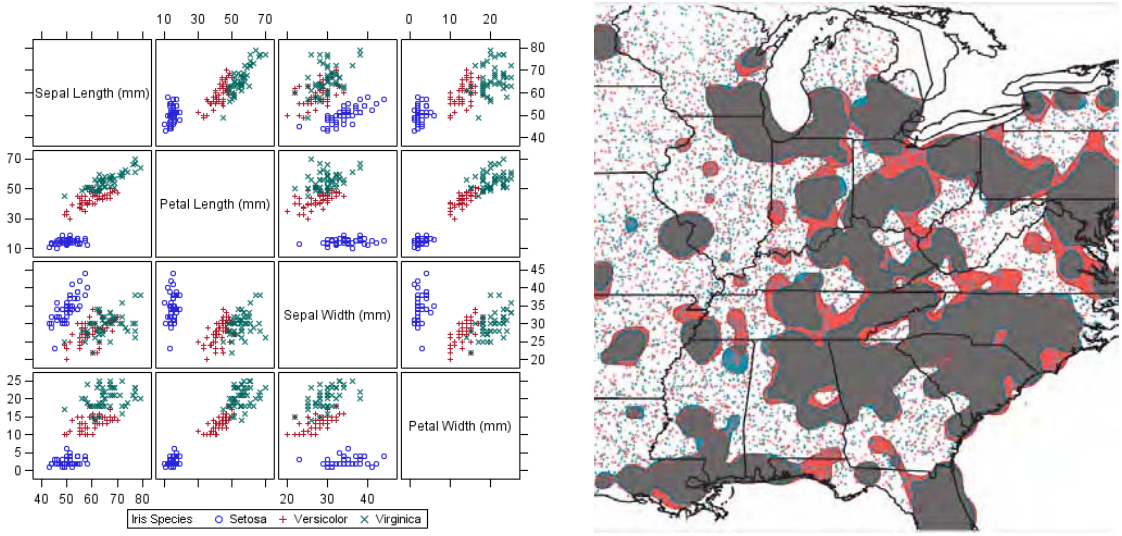
Visualisations géométriquement transformées (Geometrically-Transformed Displays)

Elles présentent des données multidimensionnelles, après leur avoir appliqué des fonctions de transformation. Il s'agit par exemple des coordonnées parallèles abordées précédemment. Les représentations de type scatter plot [7] visualisent deux ou trois attributs de variables multidimensionnelles, selon que l'on représente un espace respectivement 2D ou 3D. Une telle représentation sera étudiée plus loin dans le chapitre 3.1.4. Un dérivé de ce concept est la matrice scatter plot [Carr 86], permettant d'étudier, au sein d'une seule matrice, la corrélation entre plusieurs variables prises deux-à-deux. La colonne et la ligne de la case correspondent à une paire de variables. Le splatterplot [Mayorga 13] reprend le principe du scatter plot ou de la matrice scatter plot, mais, au lieu de dessiner les points, ceux-ci sont agrégés dans les régions denses, en surfaces colorées et fermées, tout en gardant l'information de densité et en autorisant la visualisation détaillée des données par un zoom (Cf. Figure 3.8 (b)).

Affichage d'icônes (Iconic Displays)

Au lieu d'afficher des points, comme dans le cas du scatter plot, ceux-ci sont remplacés par des icônes, ou glyphes, dont les attributs graphiques sont associés à une combinaison

7. Représentation de points dans le plan ou l'espace.



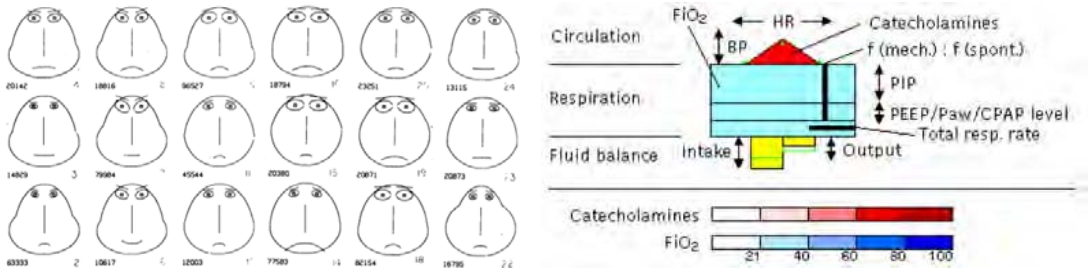
(a) (b)

FIGURE 3.8 – Matrices scatter plot et splatterplot.
(a) : la matrice permet d'étudier la corrélation entre les variables prises deux à deux [8]. (b) : le splatterplot regroupe les zones de fortes densités en surfaces fermées [Mayorga 13].

de dimensions. Ainsi, un glyphe est un objet graphique conçu pour représenter des données multidimensionnelles [Ware 00]. Cela permet de concentrer en un espace réduit plusieurs dimensions d'une donnée. Cette technique fait appel à la perception pré-attentive (Cf. Chapitre 2.1.1). Les visages de Chernoff [Chernoff 73] en sont l'exemple le plus connu. Ils présentent une donnée ayant jusqu'à 18 dimensions sur un visage, en associant une dimension à une caractéristique de celui-ci, comme la taille globale, la taille des yeux, la longueur du nez... En positionnant ces visages, par exemple dans le plan, cela permet de rajouter deux dimensions (Cf. Figure 3.9 (a)). Horn et al. [Horn 98] reprennent cette technique de visualisation à l'aide de glyphes, pour représenter des paramètres physiologiques dans le cadre d'une surveillance de patients en service de néonatologie. Ainsi, 15 paramètres sont codés graphiquement par des formes simples (rectangles, triangles, traits...), des tailles, et des couleurs à l'aide de l'outil VIE-VISU. De plus, ces formes sont concaténées dans des animations, afin de d'avoir une connaissance de l'évolution des paramètres dans le temps (Cf. Figure 3.9 (b)).

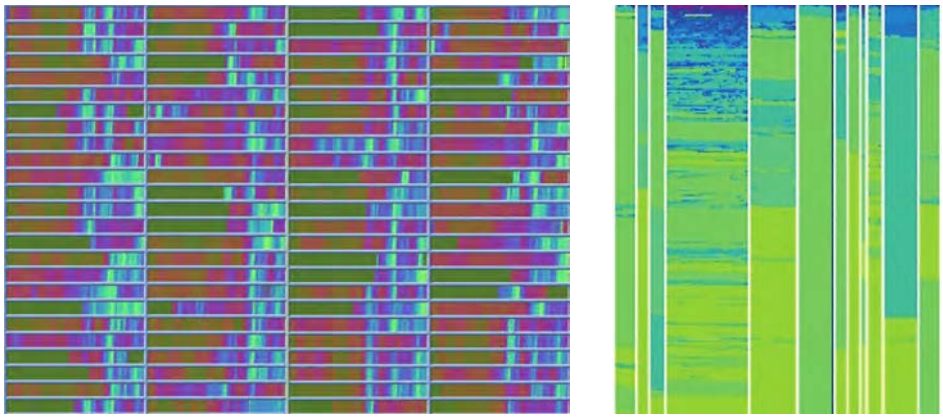
Affichages à forte densité de pixels (Dense Pixels Displays)

Le principe consiste à associer chaque dimension à un pixel coloré, et de regrouper les pixels correspondant à la même dimension [Keim 95]. Le recouvrement d'informations est donc exclu, car chaque pixel ne correspond qu'à une seule dimension. Le nombre de pixels nécessaires pour ce type de visualisation est facilement calculable. L'affichage de n données, ayant chacune m attributs, recouvre donc une surface de $n \times m$ pixels. L'encombrement de l'affichage d'un grand nombre de données est ainsi optimisé, et il n'y a pas de risque de recouvrement, ce qui n'est pas le cas de l'affichage des glyphes, abordé ci-dessus. De plus, grâce à la perception



(a) (b)

FIGURE 3.9 – Visualisation d'objets multidimensionnels par des glyphes.
(a) : les visages de Chernoff [Chernoff 73] permettent de représenter jusqu'à 18 dimensions.
(b) : VIE-VISU [Horn 98] représente, en un seul objet, 15 attributs correspondant à des paramètres physiologiques.



(a) (b)

FIGURE 3.10 – (a) : Dense Pixel Display [Keim 03]. (b) : Pixel Bar Chart [Keim 02].

pré-attentive, les régularités ou irrégularités sont facilement perçues (Cf. Chapitre 2.1.1). Avec le Pixel Bar Chart, [Keim 02], Keim et al. reprennent ce principe en codant les données, non plus en pixels, mais en barres verticales dont les caractéristiques graphiques représentent des attributs de données.

Affichages empilés (Stacked Displays)

Il s'agit de présenter les données partionnées dans une structure hiérarchique, en encapsulant, de manière itérative, une coordonnée dans une autre. Un exemple classique est la Treemap [Johnson 91] qui visualise des données hiérarchiques dans un espace déterminé. Les données sont représentées par des rectangles dont la surface et la couleur correspondent à des attributs des données. Ces rectangles remplissent la fenêtre d'affichage. Chacun est ensuite divisé en plusieurs sous-rectangles qui le remplissent, et qui correspondent à des sous-catégories de celui-ci. Ce processus peut être réitéré autant de fois que nécessaire, en fonction des données et de leur hiérarchie. La figure 3.11 (a) montre l'évolution du marché boursier américain pendant

(a) (b)

FIGURE 3.11 – Exemples de visualisations empilées.
(a) : Treemap de l'évolution du marché boursier des Etats-Unis en 2013. (b) : Cam Tree [Robertson 91]

une année [9]. Il est composé de secteurs, tels que *Health Care*, *Consumer Services*, *Technology*... Un rectangle correspond à un titre, dont la couleur indique son évolution. Les tons de rouge correspondent à une baisse, les verts à une hausse, et les blancs à une globale stabilité. Le Cone Tree [Robertson 91] est un arbre tridimensionnel, dans lequel une sous-partie d'un nœud est située dans un volume conique ou cylindrique sous celui-ci. Deux nœuds de même niveau hiérarchique se trouvent dans le même plan. Le Cam Tree [Robertson 91] en est une variante, dans laquelle, l'arbre est horizontal (Cf. Figure 3.11 (b)).

Table Lens

Présenté par Rao & Card [Rao 94], la Table Lens est une technique de visualisation d'une grande quantité de données tabulées. Les valeurs des cellules du tableau sont représentées graphiquement par des petites barres horizontales, ce qui permet de les compresser verticalement, et ainsi de représenter beaucoup de données. De outils permettent de visualiser des groupes de barres sous forme explicite, et ainsi de connaître les valeurs des cellules. L'outil MiDAVisT [Johansson 09], présenté au chapitre 3.3.3, contient une Table Lens dans sa partie supérieure droite (Figure 3.39).

Techniques de visualisation hybrides

De nouvelles méthodes de visualisation sont régulièrement présentées, mettant en œuvre plusieurs techniques simultanément. Des états de l'art dans le domaine du Visual Analytics [Sun 13, Keim 10a, Mittelstadt 12] permettent d'avoir une connaissance de quelques uns de

9. http ://www.marketwatch.com/tools/stockresearch/marketmap

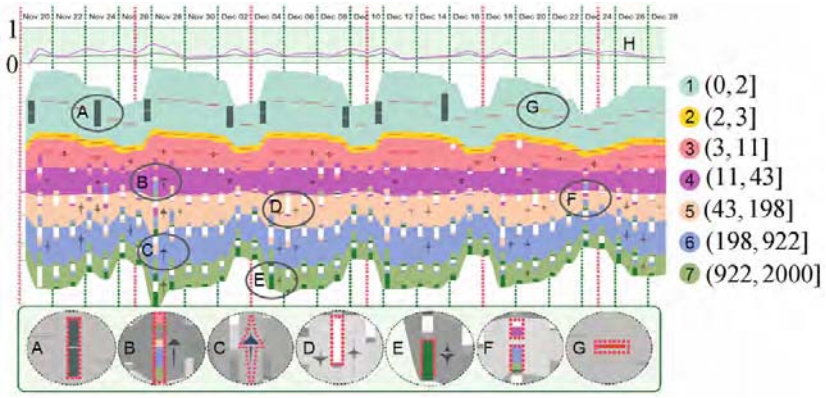


FIGURE 3.12 – Exploration de données temporelles avec RankExplorer [Shi 12].

ces travaux. Nous en présentons deux, pour illustrer la richesse de cette intégration.

Présenté par Shi et al. [Shi 12], RankExplorer est un outil d'analyse de données temporelles combinant ThemeRiver [Havre 00], des barres de couleur et des icônes (Cf. Figure 3.12). Il permet d'étudier les variations dans le temps de grandes séries de données. Les données sont subdivisées en segments, dont les variations temporelles sont visualisées dans un outil de type ThemeRiver (Cf. Chapitre 3.1.1). Des barres verticales et des icônes sont insérées dans ThemeRiver pour visualiser les changements intrinsèques à chaque thème, et ceux qui interviennent entre les thèmes. Des interactions sont mises en œuvre, comme la sélection, le filtrage et le zoom. Il est également possible de visualiser sous forme textuelle, le contenu et le nombre d'occurrences des items contenus dans une sélection.

La visualisation de données spatio-temporelles fait l'objet de nombreuses études. Dans le domaine du contrôle aérien, il s'agit de l'outil principal avec l'image radar. Tominksi et al. [Tominski 12] proposent de visualiser des trajectoires selon une approche hybride faisant cohabiter des vues 2D, qui sont des trajectoires de voitures à San Francisco, et des vues 3D, qui correspondent à l'empilement vertical de ces trajectoires pour plusieurs voitures (Cf. Figure 3.13 (gauche)). Un code couleur indique la vitesse des voitures. Des outils d'interaction, comme la sélection d'intervalles de vitesse, le déplacement vertical de la carte, pour particulariser une voiture, ou la sélection circulaire de zones géographiques sur le plan de base, permettent de naviguer dans cette représentation, de manière interactive. Ainsi, la sélection d'une zone de la carte permet de visualiser le trafic hebdomadaire dans cette zone, à l'aide d'une loupe temporelle (*Time Lens*) (Cf. Figure 3.13 (droite)).

3.1.3 Les techniques d'interaction

Il s'agit de la troisième dimension de la taxonomie de Keim & Ward [Keim 03]. L'idée est de mettre en œuvre des opérateurs, afin de faciliter l'exploration visuelle des données, selon les besoins de l'utilisateur.

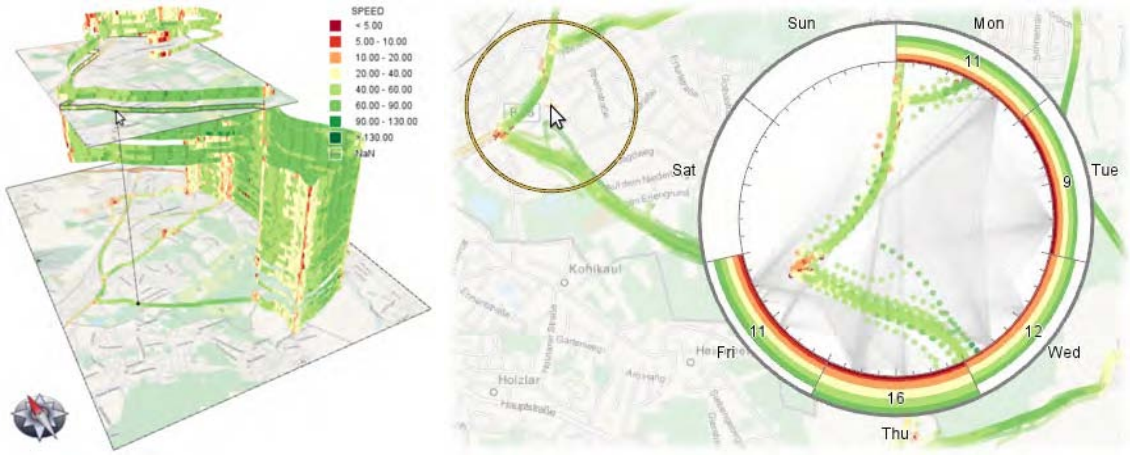


FIGURE 3.13 – Empilement 3D de trajectoires 2D, représentant le trafic dans un quartier de San Francisco [Tominski 12].

Projection dynamique

Le principe consiste à modifier dynamiquement la projection des données afin de les explorer de manière optimale. La projection peut être aléatoire, manuelle, pré-calculée ou pilotée par les données.

Filtrage interactif

Dans une activité d'exploration de données, il est souvent nécessaire d'en sélectionner un sous-ensemble, de manière interactive, sur lequel va désormais se focaliser l'exploration. Par la suite, les données peuvent être présentées différemment. C'est le cas par exemple de la Mole View [Hurter 11], qui est une loupe sémantique de présentation de données, contenues dans son périmètre, en appliquant des filtres. De plus, les données, toujours situées dans ce périmètre, et qui ne satisfont pas les filtres, sont repoussées dynamiquement vers le bord de la loupe. La figure 3.14 présente des trajectoires d'avions au-dessus du territoire français. Un filtre, fixant un intervalle d'altitudes, a été activé. Les trajectoires des avions situées dans cet intervalle sont maintenues dans le périmètre, alors que les autres sont repoussées vers l'extérieur.

Zoom

Cette technique est largement répandue, notamment grâce à l'avènement de la souris à molette qui offre des interactions faciles et intuitives. Elle procure une vue globale des données, qui sont compressées quand elles sont nombreuses, et une concentration rapide sur une partie en l'agrandissant. Dans un premier temps, la vue globale permet d'avoir une vision de la structure des données, des zones d'intérêt, des motifs et des particularités. Dans un second temps, le zoom permet de ne considérer qu'une partie de cette vue. Selon le niveau de zoom, des applications présentent les données avec plus ou moins de détails, afin de ne pas surcharger

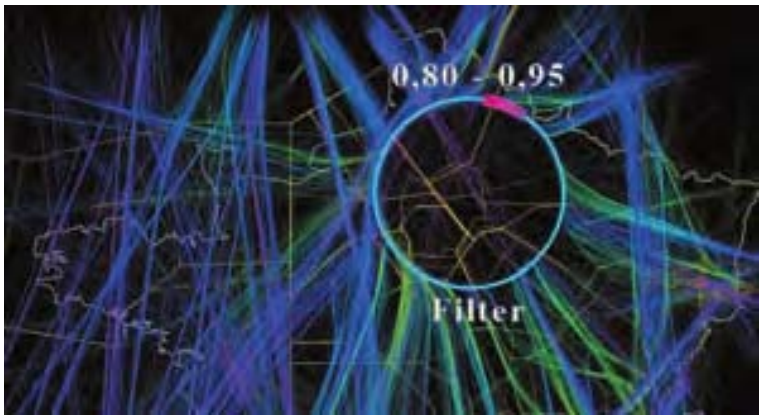
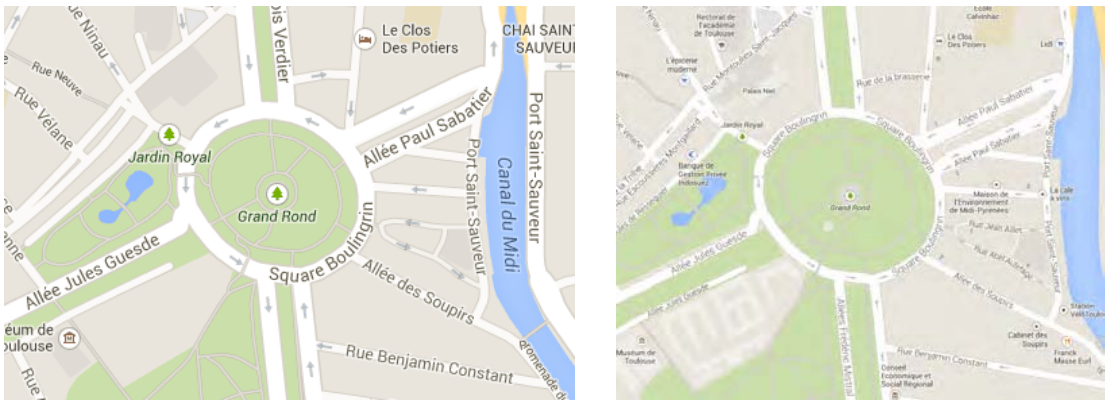


FIGURE 3.14 – Filtrage interactif des données avec la Mole View [Hurter 11].



(a) (b)

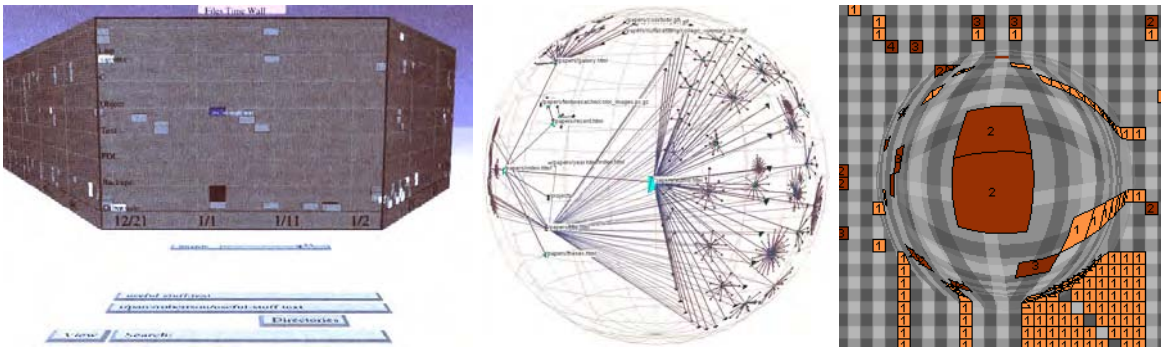
FIGURE 3.15 – Le quartier du Grand Rond à Toulouse.
Selon le niveau de zoom, l'image est plus ou moins détaillée.

la visualisation. L'image 3.15 [10] illustre ce point. Plus le zoom augmente, plus la carte contient des détails. Par exemple, la Banque Indosuez n'apparaît que sur la vue (b) qui est plus zoomée.

Distorsion

La distorsion déforme localement une partie de la vue initiale, pour l'étudier avec plus de détails, tout en en gardant une vision globale moins détaillée (Cf. Figure 3.16). Avec les murs en perspective (a) [Mackinlay 91], des objets graphiques sont positionnés sur différents plans, assimilés à des murs, avec des effets de perspective. Les arbres hyperboliques (b) [Lamping 95, Munzner 97] permettent une déformation progressive d'un arbre, afin d'en étudier une partie, tout en gardant la visualisation de l'arbre dans son intégralité. Pour obtenir un effet similaire, la vue Fisheye (c) [Furnas 86] déforme de manière concentrique l'image initiale autour d'un point d'intérêt.

10. https ://maps.google.fr



(a) (b) (c)

FIGURE 3.16 – Trois exemples de techniques de distorsion.
(a) : Perspective Wall [Mackinlay 91]. (b) : arbre hyperbolique 3D [Munzner 97]. (c) :FishEye View [Fekete 04].

Brushing et linking

Le brushing [Becker 87, Martin 95] consiste à sélectionner, par un effet de brosse ou de pinceau, une partie des données à l'aide d'un dispositif comme la souris. Il est réalisé à la manière d'une brosse, en sélectionnant les données par leur survol à l'aide d'un disque centré sur le curseur de la souris, ou par délimitation d'une zone rectangulaire à la souris, en déplaçant le curseur d'un angle du rectangle de sélection vers l'angle opposé. Cette sélection fait alors l'objet d'un traitement, comme la suppression des points qu'elle contient, leur particularisation, leur linking, etc. Le linking consiste à relier des ensembles de données présentes dans plusieurs visualisations. Grâce à lui, des changements opérés sur une visualisation, par exemple par brushing, sont répercutés sur ces mêmes données présentes dans les autres visualisations. Concernant les vues multiples et coordonnées (CMV), un état de l'art récent a été réalisé par Roberts [Roberts 07]. Un exemple de brushing et linking est donné à la figure 3.17, extraite de [Symanzik 98]. Il montre la connexion entre deux instances de l'application XGobi [Swayne 91], outil de représentation graphique de données statistiques et ViRGIS [Szabo 95], outil de visualisation tridimensionnelle de données terrain. ViRGIS, dans la partie supérieure, montre la Suisse. XGobi, en bas à gauche montre un scatter plot illustrant la population et les langues parlées, et, en bas à droite, montre les langues parlées. Sur cette dernière application, la langue italienne a été sélectionnée en blanc. Grâce au linking, cette donnée est répercutée sur l'autre Xgobi. Sur la vue terrain de ViRGIS, les villes du Sud-Est sont alors particularisées. Ainsi, le brushing sur une vue permet de particulariser les données correspondantes des autres vues.

3.1.4 FromDaDy : un outil de Visual Data Mining

Les outils de Visual Data Mining doivent utiliser un langage de description des données. Le modèle de Card & Mackinlay est mis en œuvre dans FromDaDy (FROM DAta to DisplaY) [Hurter 09], dont le but est l'exploration visuelle d'un grand volume de données (Cf. Figure 3.18).

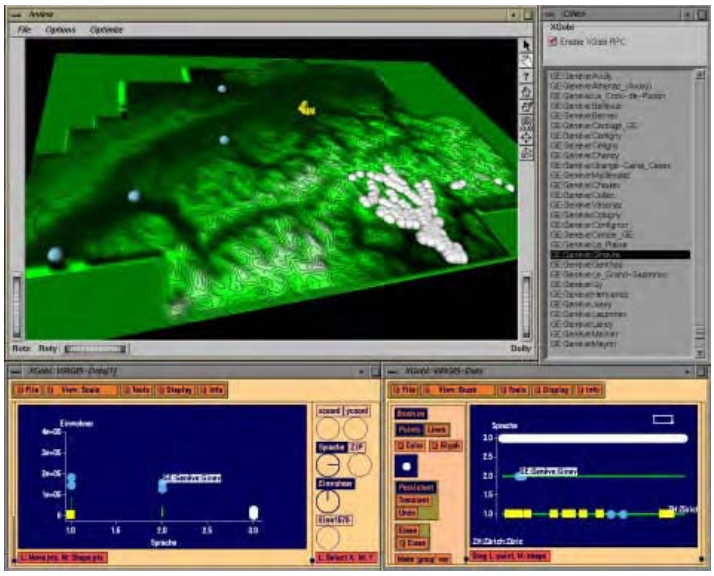


FIGURE 3.17 – Exemple de brushing et linking.
ViRGIS (haut) et XGobi (bas) [Symanzik 98]. Les données associées à la langue italienne sont particularisées en blanc.

FromDaDy utilise un simple paradigme pour explorer une base de données multidimensionnelle, basé sur une vue scatter plot. La manipulation des données est réalisée à l'aide d'opérateurs de sélections de type brushing (Cf. supra), et de pick & drop [Rekimoto 97] (prendre et déposer) pour configurer rapidement l'IHM. Par la personnalisation de l'interface à l'aide de ces interactions simples, l'utilisateur peut filtrer les données, les extraire ou les combiner, le tout étant réalisé de manière visuelle et incrémentale. La figure 3.19 illustre les étapes de sélection pour construire une visualisation. De manière simple, l'utilisateur peut appliquer des opérateurs booléens, tels que le AND, OR, NOT et XOR. Cette manipulation a été récemment enrichie par le Color Tunneling [Hurter 14]. Il s'agit d'un ensemble de techniques d'interactions pour l'exploration et la sélection de données multidimensionnelles.

FromDaDy met en œuvre des fonctionnalités avancées, telles que les cartes d'accumulation [Hurter 10] (Cf. Figure 3.21), avec la technique *Kernel Density Estimation* [Silverman 86]. Elle consiste à tracer un kernel en chaque point, c'est-à-dire une petite bosse, pour obtenir un kernel résultant (*kernel estimator*) qui est l'accumulation de tous les kernels (Cf. Figure 3.20). Ainsi, pour une zone donnée, plus les kernels sont accumulés, plus le résultant est élevé. Dans FromDaDy, l'accumulation est réalisée par la carte graphique, en dessinant, dans une texture, des kernels monochromes ayant une forme de disque, et dont la valeur de la couleur décroît du centre vers la périphérie. La superposition des kernels augmentant la valeur de la couleur, la texture résultante fait ressortir l'accumulation par des valeurs plus ou moins élevées. Cette technique permet de tirer parti de la superposition de points dans des zones denses de l'image. Elle est renforcée par des effets d'ombrage, ou *shading*, qui procurent une sensation de relief (Cf. Figure 3.21).

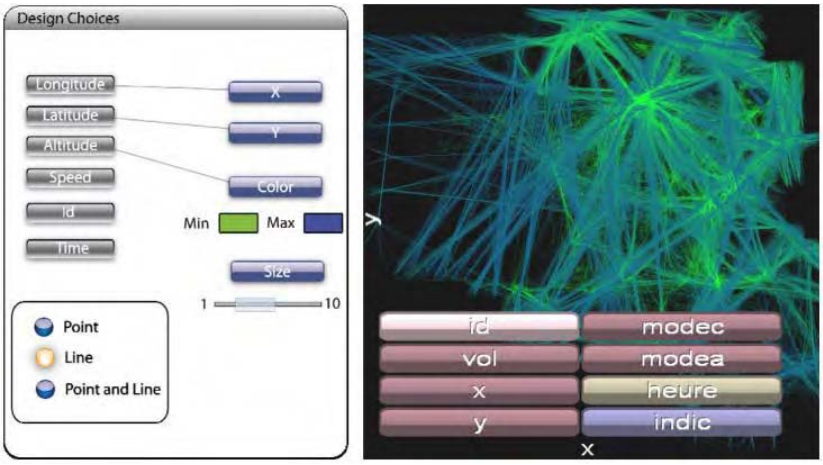


FIGURE 3.18 – FromDaDy : vue générale de l'interface [Hurter 09].

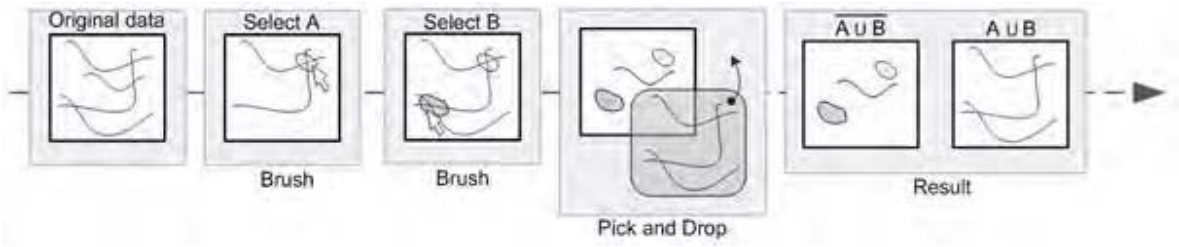


FIGURE 3.19 – FromDaDy met en œuvre des opérateurs booléens pour manipuler les vues.

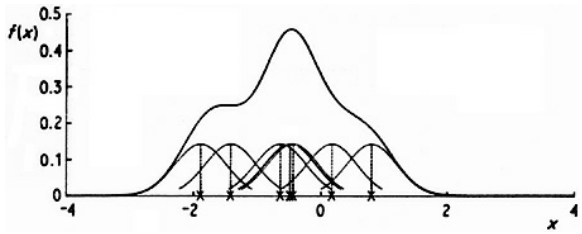


FIGURE 3.20 – Principe de l'accumulation avec la technique *KDE* [Silverman 86].
Le kernel résultant est la somme de tous les kernels.

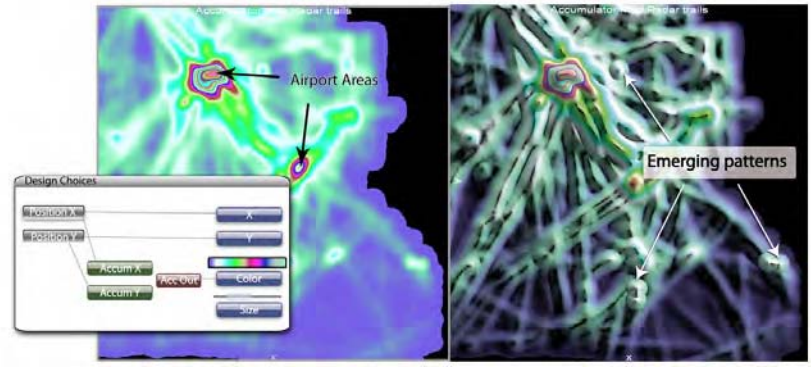


FIGURE 3.21 – Mise en œuvre des cartes d'accumulation dans FromDaDy.
Gauche : sans shading. Droite : avec shading.

Le Visual Data Mining prend en compte l'expertise de l'utilisateur. Comme elle conditionne sa recherche, il configure l'IHM pour trouver de l'information, dans le but de répondre à une question. Ainsi, ce type d'outil est plus utilisé pour valider des hypothèses que pour rechercher de la connaissance. S'il est configurable comme FromDaDy, les réglages sont différents en fonction de la question posée. Mais le volume de données et le nombre d'attributs peuvent se révéler être des obstacles. En effet, la visualisation peut devenir rapidement trop chargée pour extraire de l'information. De plus, comme les attributs correspondent à des dimensions, il s'avère difficile d'en gérer plusieurs. Un autre problème des outils de Visual Data Mining est la configuration de la visualisation. Ils montrent les données dans des vues scatter plot, des graphes, des vues de type fisheye, ou des barres 2D ou 3D [Couturier 07b]. Cela peut être également une combinaison de techniques de présentation de données. Mais un spécialiste peut avoir besoin d'un autre type de présentation que celui proposé par l'outil dont il dispose. Il lui est alors nécessaire de choisir un autre outil plus adapté à son besoin.

Après avoir abordé les outils de visualisation des données, la suite de ce chapitre va traiter la visualisation des résultats algorithmiques, c'est-à-dire des itemsets et des règles d'association.

3.2 Visualisation des résultats des algorithmes de fouille de données

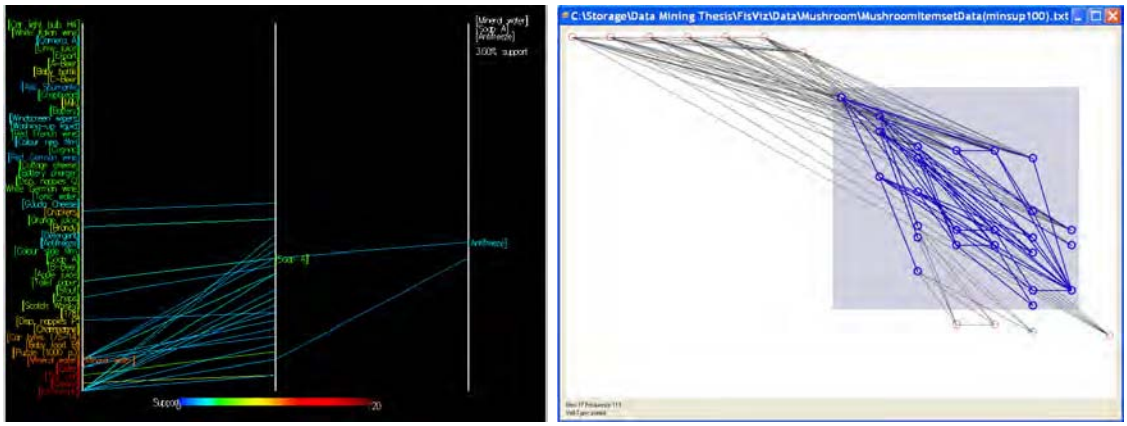
3.2.1 Visualisation des itemsets

Il existe peu d'études sur les outils de visualisation des itemsets. Glatz et al. [Glatz 12] affichent des patterns d'extraction de trafic dans des hypergraphes . D'autres études montrent l'intérêt de présenter la totalité des itemsets fréquents en les visualisant dans des graphes. En effet, en montrant simultanément les données et les liens entre elles, cela procure une vision générale des données. De plus, comme un élément est généralement impliqué dans plusieurs itemsets, ceux-ci peuvent avoir différentes tailles. Une représentation sous forme de graphes est ainsi bien adaptée pour afficher les itemsets fréquents.

Yang [Yang 03] les dispose avec des coordonnées parallèles (Cf. Chapitre 3.1). L'axe vertical, dupliqué plusieurs fois, contient tous les items, arrangés par groupes et en fonction de leur support. La duplication a lieu autant de fois qu'une règle peut contenir d'items. Ainsi, les nœuds sont reliés entre eux pour construire graphiquement les itemsets (Cf. Figure 3.22 (a)).

Avec V-Miner [Zhao 04], Zhao et al. utilisent également des coordonnées parallèles, mais de manière différente (Cf. Figure 3.23). Ils prennent en compte l'ordre selon lequel les données ont été générées. Chaque ligne verticale représente une donnée. Au-dessus est présenté un graphe de tendance (*Trend Figure*) indiquant l'implication de la donnée dans la séquence d'enregistrement.

FIsViz [Leung 08a] visualise les itemsets dans un espace 2D, en les reliant par à des liens (Cf. Figure 3.22 (b)). La composante verticale représente la valeur du support. Les 1-itemsets sont dispersés horizontalement afin de séparer les itemsets finaux sur la vue. WiFIsViz [Leung 08b]



(a) (b)

FIGURE 3.22 – Visualiseurs linéaires d'itemsets fréquents.
(a) : avec les Coordonnées Parallèles, les lignes verticales contiennent le même arrangement d'attributs. Pour obtenir un itemset, les attributs sont reliés d'une ligne verticale à la suivante [Yang 03]. (b) : FIsViz présente les itemsets fréquents en fonction de leur support qui est codé par la valeur verticale [Leung 08a].

visualise les itemsets dans le même espace 2D que FIsViz (Cf. Figure 3.24 (a)). Mais ce graphe est une vue générale des itemsets. Il les compresse en partageant des informations communes et les projette sur une ligne. Il peut également présenter les itemsets sur une vue détaillée. Dans la suite de ses travaux, Leung présente FPViz [Leung 09] qui n'affiche que la vue générale des itemsets. Avec Fp-Viz [Keim 05] [11], Keim et al. proposent une vue radiale et hiérarchique des itemsets fréquents. Après leur calcul par l'algorithme FP-Growth [Han 00] (Cf. Chapitre 1.5.1), ils sont visualisés dans une suite ordonnée de secteurs angulaires (Cf. Figure 3.24 (b)). Dans la partie basse de l'IHM, un bouton glissant permet d'ajuster le support minimum, et la vue est réajustée en fonction de ce réglage. Il a un effet sur la couleur qui est utilisée pour visualiser le support de chaque itemset.

Ertek & Demiriz [Ertek 06] présentent les itemsets fréquents dans un graphe (Cf. Figure 3.25). Les cercles blancs représentent les items, et les colorés correspondent aux itemsets dont l'ordre est codé par la couleur. La taille du cercle indique le support. Les cercles sont reliés entre eux par des flèches qui procurent un sens de lecture et permettent de comprendre la construction des itemsets. Par exemple, le 2-itemset *F01* en haut à gauche (bleu clair, car d'ordre 2) est construit à partir des items *110* et *38*. Un outil de sélection, appliqué à des items ou des itemsets, permet de faire ressortir graphiquement les cercles qui leur sont reliés, cela en atténuant les autres (Figure 3.25 (b))

Singh et Garg [Vijender Singh 11] ont présenté une étude sur les outils de recherche des itemsets fréquents. Ils montrent une importance croissante de ce type d'approche. Mais au fur et à mesure que la quantité de données augmente, les vues sont de plus en plus encombrées et de nouvelles approches sont nécessaires pour aborder ce problème.

11. Fp-Viz [Keim 05] and FpViz [Leung 09] sont deux études différentes.

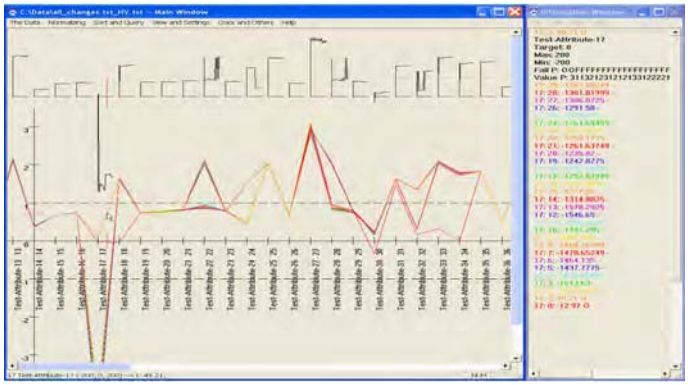
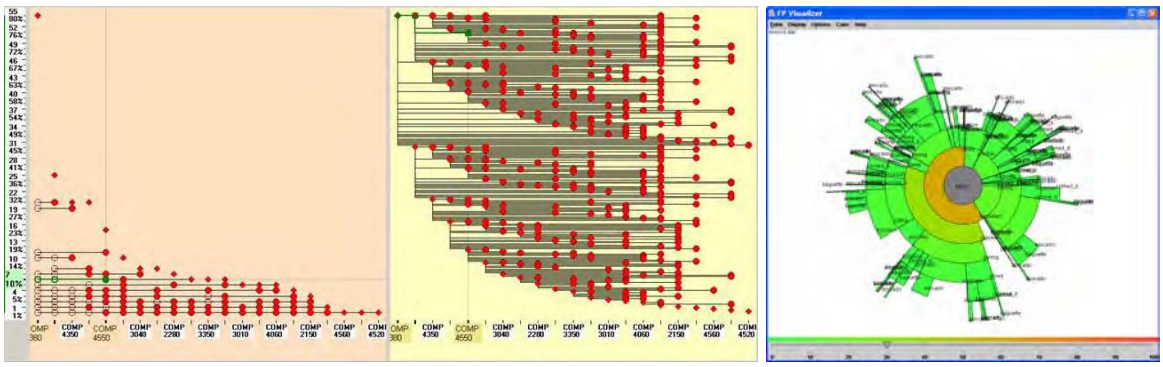


FIGURE 3.23 – La représentation de plusieurs instances d'un itemset avec V-Miner [Zhao 04]. En utilisant les coordonnées parallèles et des graphes de tendance, cette représentation visualise la séquence d'enregistrement des données.



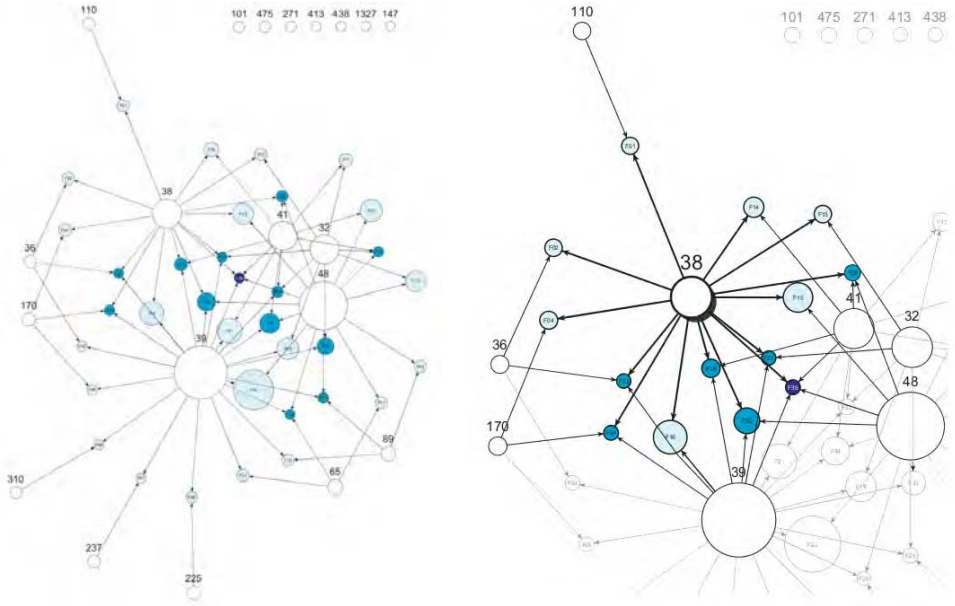
(a) (b)

FIGURE 3.24 – Visualiseurs d'itemsets fréquents.
(a) : WiFIsViz montre les itemsets dans une vue globale (gauche) ou détaillée (droite) [Leung 08b]. (b) : FP-Viz montre les itemsets fréquents selon une visualisation hiérarchique radiale. la couleur de chaque item correspond à son support [Keim 05].

Après avoir étudié la visualisation des motifs fréquents, le paragraphe suivant traite celle des règles d'association.

3.2.2 Visualisation des règles d'association

Le chapitre 1.5 a présenté l'approche algorithmique de recherche de règles d'association comme avantageuse par son caractère exhaustif, grâce auquel la totalité des règles, qui satisfont des contraintes sur un ensemble de métriques, est trouvée. Cependant, le nombre de règles extraites peut parfois être plus important que la masse de données initiale. En effet, si la valeur seuil du support est trop basse, alors le nombre d'itemsets fréquents augmente. Les règles, résultant d'une combinatoire entre les éléments des itemsets, leur quantité augmente également, cela de manière très rapide. La raison en est que la découverte des règles d'association est un



(a) (b)

FIGURE 3.25 – Graphe d'itemsets fréquents par Ertek & Demiriz [Ertek 06].
(a) : le graphe entier. (b) : un item ayant été sélectionné, les items et itemsets reliés sont particularisés.

processus non supervisé, dans lequel l'utilisateur n'expose pas ses objectifs, ce qui n'impose aucune limitation, si ce n'est celle des seuils [Blanchard 03]. La problématique de fouille dans une masse de données, pour en extraire les motifs intéressants, est ainsi reportée vers une problématique de fouille dans une masse de règles. Il s'agit maintenant d'explorer ces règles et d'identifier leurs sous-ensembles les plus pertinents.

Pour cela, le rôle de l'utilisateur est prépondérant, car il apporte une subjectivité et une expertise permettant de rendre la fouille de règles pertinente. Frawley et al. [Frawley 92] ont été pionniers dans ce domaine, en mettant en exergue, dès le début des années 90, la prépondérance à venir de ce rôle. Ils annonçaient que la meilleure opportunité pour la découverte de connaissance serait sans doute la mise en œuvre de systèmes interactifs. Cette approche combine le meilleur de l'homme et de la machine : utiliser le jugement humain, mais se fier à la machine pour effectuer la recherche et passer au crible les données. Les études, mettant l'expertise humaine dans la boucle de découverte de connaissance à partir des données, sont assez nombreuses et passent par des systèmes de visualisation. Il s'agit alors de systèmes anthropocentrés ou de recherche anthropocentrée de règles d'associations [Chevrin 07, Couturier 04, Kuntz 06]. Cette approche considère que l'utilisateur a une stratégie pour explorer les règles. Au lieu de les appréhender dans leur globalité, il oriente progressivement sa recherche vers une sous-partie des règles visualisées.

La huitième étape du processus d'extraction de connaissance à partir des données consiste à interpréter les motifs trouvés, mais aussi à en évaluer la qualité [Piatetsky-Shapiro 91b,

Fayyad 96a]. Pour ce faire, les mesures de qualités des règles sont exploitées afin de pouvoir les évaluer, de quantifier leur pertinence, et de les comparer (Cf. Chapitre 1.5.3). L'approche consiste à :

- Représenter les règles sous forme graphique. Leurs attributs visuels représentent leurs caractéristiques. En effet, l'approche graphique est un outil précieux pour visualiser une grande masse de données, qui sont dans ce cas, des règles (Cf. Chapitre 3.1) ;
- Disposer d'outils d'interaction pour pouvoir explorer la visualisation des règles. Cela passe par des visualisations éventuellement différentes, des opérateurs de sélection et de tri ;
- Présenter de manière plus ou moins explicite le contenu de la règle. Il peut être utile de connaître l'intitulé exact de la règle sous la forme *prémisse* $\Rightarrow$ *conclusion*, ou sous une forme qui explicite la prémisse et la conclusion.

Plusieurs techniques de visualisation et d'explorations des règles d'association ont été étudiées, et continuent de l'être. Bruzzese & Davino [Bruzzese 08] en ont fait récemment un état de l'art. Les techniques sont essentiellement des scatter plots, des représentations matricielles, des graphes, des mosaïques et des coordonnées parallèles. Il s'agit en général de règles de type *one-to-one*, c'est-à-dire, que la prémisse et la conclusion n'ont qu'un seul terme, ou des règles de type *many-to-one*, c'est-à-dire que la prémisse peut contenir plusieurs items, alors que la conclusion n'en a qu'un seul.

Nous allons aborder ces différentes techniques dans la suite de ce chapitre en reprenant et complétant la classification de Bruzzese & Davino.

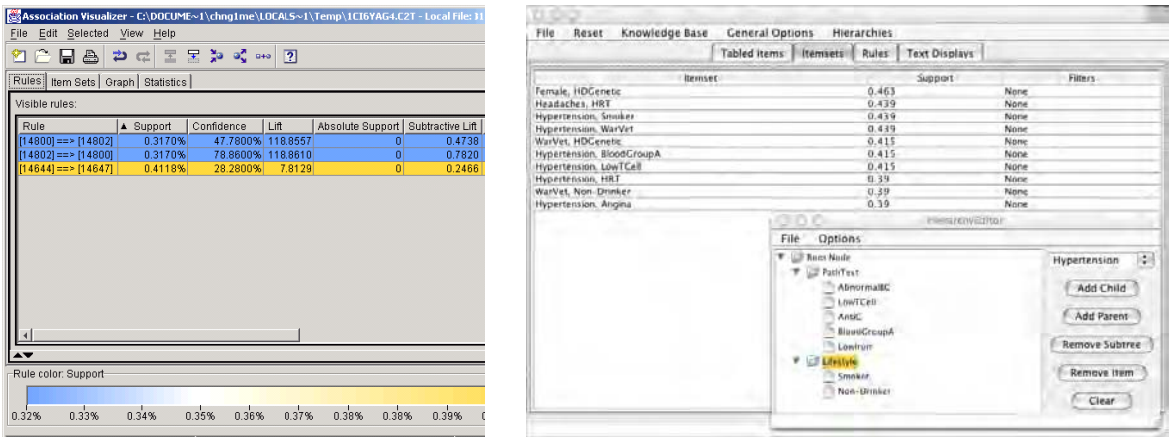
Visualisation textuelle des règles d'association dans des tables

Cette méthode est la plus immédiate. Elle consiste à afficher la liste des règles d'association sous forme de tableaux. L'intérêt est de pouvoir réaliser facilement des opérations de tris selon des critères désirés, tels que le support, la confiance, etc. Le principal inconvénient est la difficulté d'avoir une vue globale des règles et de leurs liens. De plus, il est parfois difficile d'avoir une explicitation des items qui les composent.

Avec WebSphere Commerce Analyzer d'IBM [12], chaque ligne contient une règle et ses caractéristiques : la prémisse, la conclusion et les valeurs des mesures (Cf. Figure 3.26 (a)). De plus, le support est codé par la couleur respectant un gradient explicité en bas de la fenêtre. Avec IRSetNav [Hogan 04], Fule & Roddick proposent une palette d'outils pour créer et explorer les règles. Ils présélectionnent d'abord les données en fonction de critères, puis ils calculent les règles en utilisant des métriques objectives, ne requérant pas de connaissances du domaine, et subjectives, faisant appel à l'expertise. L'ensemble des règles est ensuite simplifié, puis visualisé (Cf. Figure 3.26 (b)).

Des langages de requêtes ont été élaborés pour créer et exploiter les règles d'association. Ainsi, dans [Braga 02], les règles sont construites à partir de données au format XML. Avec

12. http ://www-01.ibm.com/software/websphere



(a) (b)

FIGURE 3.26 – Représentations textuelles de règles d'associations.
(a) : WebSphere Commerce Analyzer d'IBM. (b) : IRSetNav [Hogan 04].

MINE RULE [Meo 98], le langage SQL a été étendu pour adapter les requêtes à l'exploration de règles d'association.

Matrices à deux et trois dimensions

Le principe des matrices à deux dimensions est de représenter les règles à la croisée des lignes et des colonnes, qui représentent en général le support et la confiance. Le contenu de la cellule est codé par la couleur, ou par de l'information caractérisant la règle. Dans les matrices 3D, la troisième dimension sert à représenter une métrique associée à la règle.

Wong et al. [Wong 99b] proposent un outil de visualisation de règles basé sur une représentation matricielle tridimensionnelle (Cf. Figure 3.27). Il s'agit de règles de type *many-to-one*, les prémisses étant en bleu et les conclusions en rouge. Les lignes représentent les attributs et les colonnes représentent les règles. A l'arrière-plan de la vue sont représentés le support en bleu, et la confiance en vert, sous la forme de diagrammes à barres. Les tests ont montré que cette visualisation peut montrer jusqu'à quelques centaines de règles d'association. Un avantage de cette approche est qu'il n'y a pas de limites dans le nombre d'attributs et de règles. Cependant, les problèmes d'occultations inhérents à une représentation tridimensionnelle sont inévitables (Cf. Chapitre 2.2).

Hahsler et al. [Hahsler 11] proposent également une visualisation matricielle en 2D ou 3D, ainsi qu'une représentation matricielle groupée (Cf. Figure 3.28). Dans cette dernière, les prémisses correspondent aux colonnes, et les conclusions aux lignes. Les prémisses ne sont pas des items, mais des groupes d'items classés en catégories. Ainsi, sur la figure 3.28, le point supérieur gauche de la matrice correspond à trois règles dont les prémisses sont dans le groupe *Instant blood products* qui contient trois items, et dont la conclusion est *Hamburger meat*. La taille du point est fonction des supports et le niveau de gris est fonction du lift, sachant qu'une autre mesure d'intérêt peut être utilisée. La fonction peut être la moyenne, la valeur

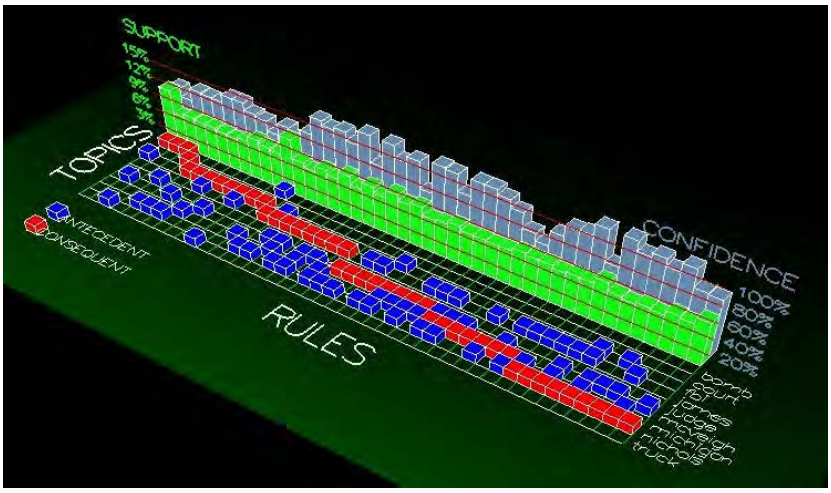


FIGURE 3.27 – Représentation de règles d'associations selon Wong et al. [Wong 99b].

maximale, la minimale ou la valeur médiane. Par ailleurs, il est possible de détailler une partie de la matrice en sélectionnant une colonne. Cela correspond à un effet de zoom.

Couturier et al. [Couturier 06] proposent une approche hybride pour améliorer la performance de l'utilisateur d'une matrice 2D, en exploitant un effet Fisheye View (Cf. Chapitre 3.1.3). Le principe consiste à déformer l'image localement à partir d'une zone qui est agrandie, et à diminuer la déformation au fur et à mesure que l'on s'éloigne de cette zone. La figure 3.29 illustre ce point, avec la toolkit InfoVis [Fekete 04] (a) et LARM [Couturier 05] (b).

Avec CbVAR (Clustering-based Visualizer of Association Rules) [Couturier 07a], Couturier et al. regroupent les règles en clusters dans un premier temps, puis visualisent ces groupes dans une matrice bidimensionnelle (Cf. Figure 3.30). En sélectionnant un cluster, son contenu est visualisé dans une vue tridimensionnelle. Le but est de proposer simultanément une vue globale et une vue détaillée des règles. En effet, les vues globales sont rapidement inexploitables, et les vues détaillées ne montrent pas la totalité des règles.

D'autres études portent sur les matrices 2D ou 3D, comme CrystalClear [Kian Huat 02] ou MineSet [13]. La représentation matricielle présente en général des règles ayant deux items, ou n'ayant qu'un seul item dans la conclusion.

Visualisations tridimensionnelles

Avec Hao et al. [Hao 01], la visualisation des règles d'association est réalisée sous la forme de sphères reliées par des segments, chacune correspondant à un item. La configuration globale de la visualisation provient d'un équilibre entre les sphères, issu d'un calcul de distance lié à leur support. Blanchard et al. [Blanchard 07] présentent, avec le *Rule Focusing*, une technique consistant à affiner l'exploration visuelle des règles, en partant de leur représentation globale et en tendant vers une représentation de plus en plus locale. Chaque itération est réalisée

13. http ://www.algorithmic-solutions.com/leda/projects/mineset.htm

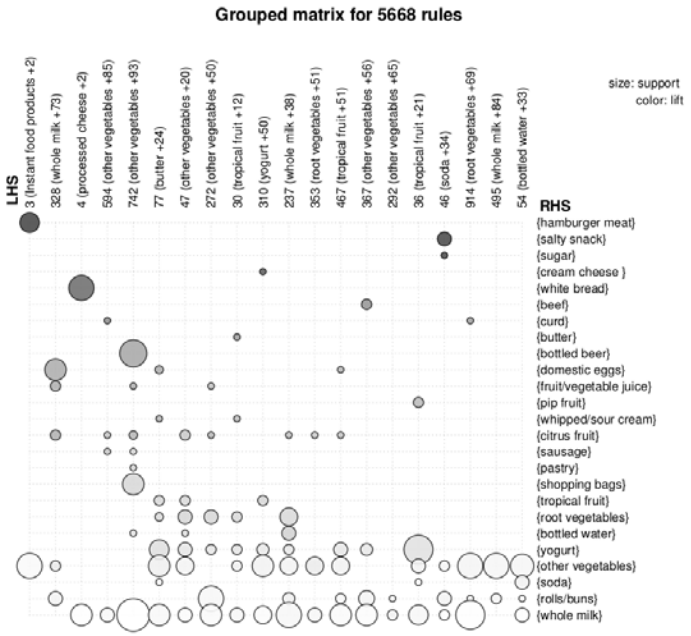
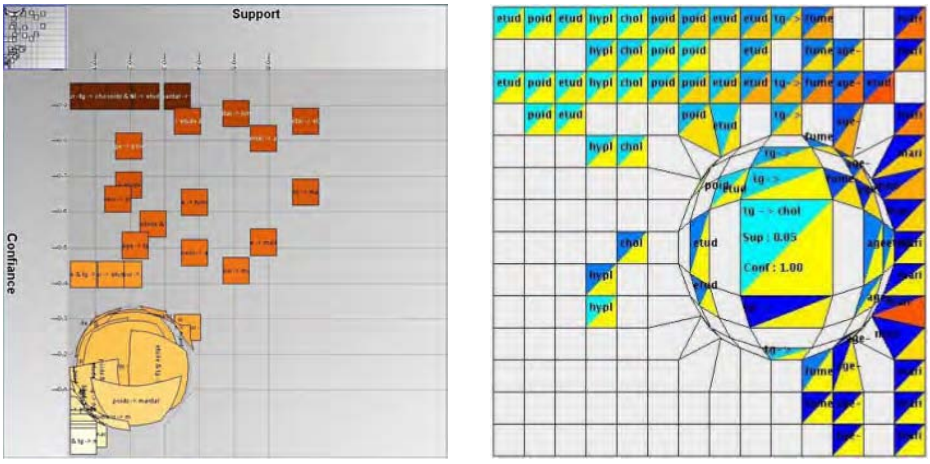


FIGURE 3.28 – Représentation de règles d'associations dans des matrices groupées.



(a) (b)

FIGURE 3.29 – Approche hybride à base d'effet Fisheye View pour améliorer la lisibilité des règles d'association sur des matrices 2D [Couturier 06].

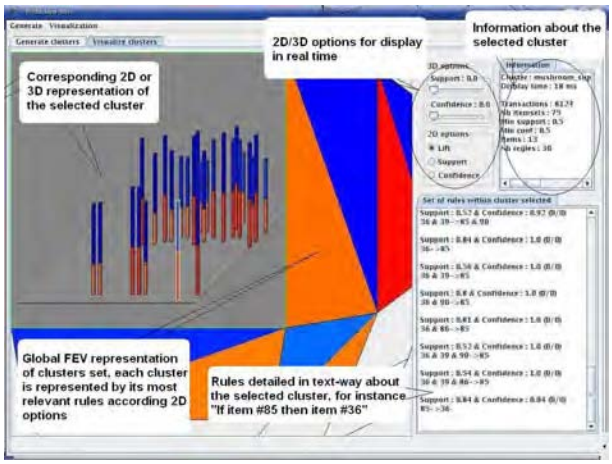


FIGURE 3.30 – Avec CbVAR [Couturier 07a], Couturier et al. présentent des clusters de règles dans une matrice 2D, et le contenu des clusters dans une vue 3D.

à l'aide d'un opérateur de sélection, qui permet de ne considérer qu'une sous-partie des règles visualisées. Ils ont illustré ce principe dans ARVis [Blanchard 03, Blanchard 07] qui est une représentation tridimensionnelle de l'espace des règles, selon une métaphore de paysage semi-circulaire (Cf. Figure 3.31). En effet, la représentation de données dans un paysage facilite l'acquisition de données spatiales [Chen 04]. Chaque règle est représentée par un objet constitué de trois parties :

- Le cône. Sa hauteur représente la confiance ;
- La sphère. Sa taille représente le support ;
- L'étiquette fournit des informations sur le nom de la règle, et les valeurs des mesures qui lui sont associées : le support, la confiance et l'intensité d'implication.

La couleur de l'objet représente de manière redondante la confiance moyenne et l'intensité d'implication entropique moyenne. La plage de valeurs des trois mesures visualisées peut être restreinte en leur fixant un minimum et un maximum. Cela a pour conséquence de faire apparaître ou disparaître les règles dans le paysage. De plus, le positionnement de la règle, sur un quart de sphère transparente intégrée dans le paysage, illustre une mesure d'intérêt qui est l'intensité d'implication entropique. La visualisation en 3D, ainsi que la couleur des objets, permettent de valoriser les règles les plus importantes au premier plan, alors que les moins importantes sont repoussées vers l'arrière-plan, avec des couleurs plus atténuées. Ainsi, une sphère rouge au premier plan, placée sur un grand cône, représente une règle dont le support, la confiance et l'intensité d'implication sont élevés. Les règles pour lesquelles ces valeurs sont les plus basses, sont au fond du paysage et colorées en bleu. Cette configuration (choix des couleurs, et mesures de qualité) peut cependant être adaptée par l'utilisateur.

Cette approche permet de comparer jusqu'à 250 règles d'association visualisées par leur position et les attributs graphiques des objets les représentant. En cliquant sur une règle, la vue effectue un zoom sur celle-ci et propose de poursuivre la navigation en fonction de critères de voisinage, qui sont par exemple *avoir les mêmes items dans la règle*, *avoir la même*

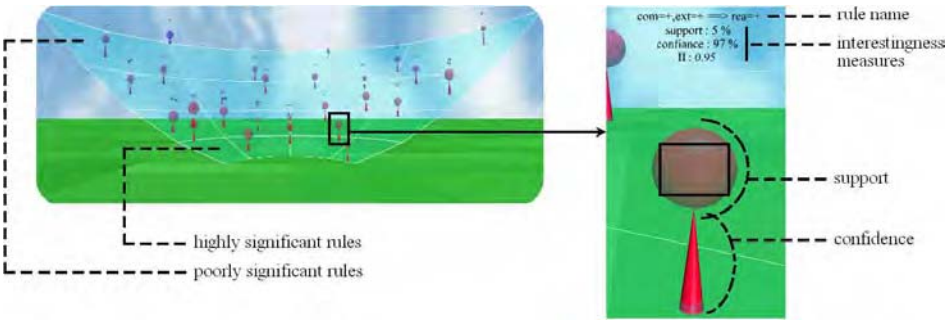


FIGURE 3.31 – L'outil ARVis de visualisation des règles d'association.

prémisse, *avoir la même conclusion*. En sélectionnant ce critère, la vue propose un nouveau sous-ensemble de règles.

Dans les visualisations 3D des règles, il est intéressant de mentionner les travaux récents sur la réalité virtuelle de Ben-Said Guefrech [Ben Said Guefrech 12], qui manipule une représentation moléculaire de règles avec l'outil IUCAREVis (Interactive User-Centered Association Rules Exploration and Visualization).

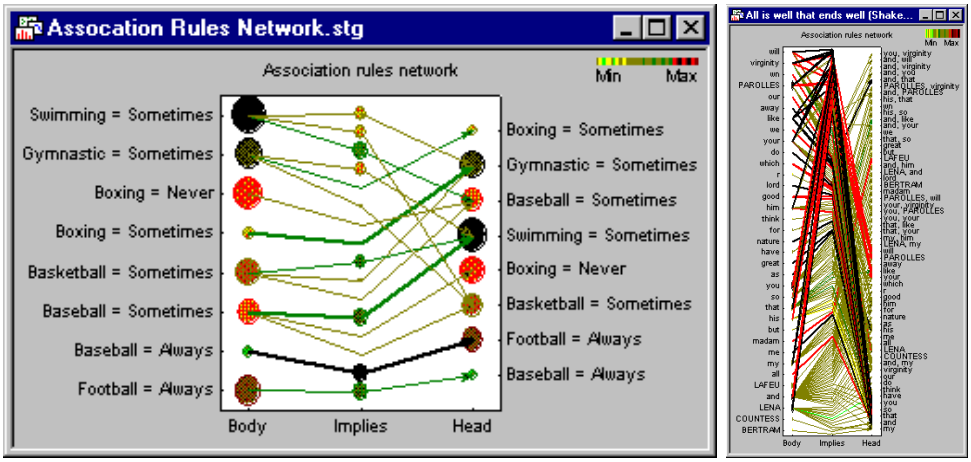
Graphes

Une technique courante de représentation de règles d'association est l'utilisation de graphes. Les nœuds représentent les attributs, et les liens représentent les associations. L'avantage principal est d'offrir une vue générale des règles grâce à la hiérarchisation. Cependant, le graphe devient rapidement inexploitable dès que le nombre d'informations à afficher augmente. L'inconvénient est donc que cela n'est exploitable qu'avec peu de règles et d'attributs. Le logiciel Statistica [14] propose ce type de visualisation (Cf. Figure 3.32). Les prémisses sont positionnées à gauche et les conclusions à droite. La taille et la couleur des cercles représentent, respectivement de gauche à droite, le support de la prémisse, de la règle et de la conclusion. L'épaisseur des liens indique la confiance. Une représentation 3D de ce réseau ajoute, sur un axe vertical, la représentation de la confiance. Comme le montre la figure 3.32, cette visualisation est rapidement saturée avec l'augmentation du nombre d'items et de règles.

Bruzzese & Buono [Bruzzese 04] présentent un graphe dans lequel, en sélectionnant une règle, celles qui lui sont apparentées sont particularisées par un point coloré différemment (Cf. Figure 3.33). La couleur indique si cette règle a des antécédents ou des successeurs, c'est-à-dire si son itemset a des supersets ou des sous-sets donnant lieu à des règles. Les problèmes d'occlusion sont gérés par une visualisation animée de la vue.

Ertek & Demiriz [Ertek 06] présentent les règles dans un graphe hiérarchique (Cf. Figure 3.34). Les items sont représentés par les cercles blancs. Les règles sont colorées en fonction de la confiance, selon un gradient de jaune à rouge. La taille des cercles correspond au support. La construction des règles et les liens entre elles sont lisibles par les segments qui les relient.

14. http ://www.statsoft.com



(a) (b)

FIGURE 3.32 – La représentation de règles d'association en réseau dans Statistica. L'image (b) montre les limites de ce type de visualisation quand le nombre d'items et de règles augmente.

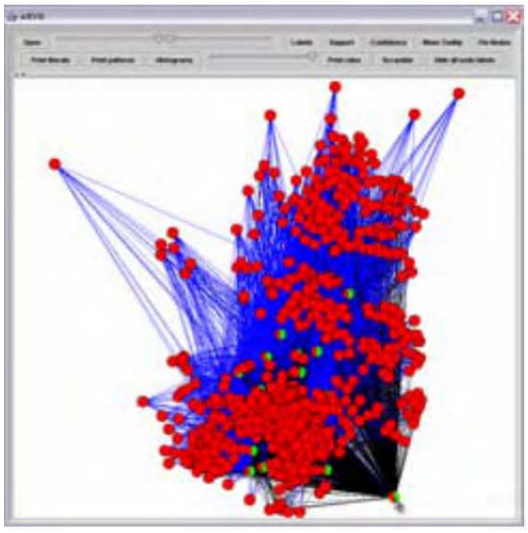


FIGURE 3.33 – Représentation de 9785 règles d'association en graphe dans [Bruzzese 04]. A partir de la sélection de l'item en bas à droite, les items qui lui sont apparentés sont particularisés. La couleur noire des traits indique une confiance égale à 100%.

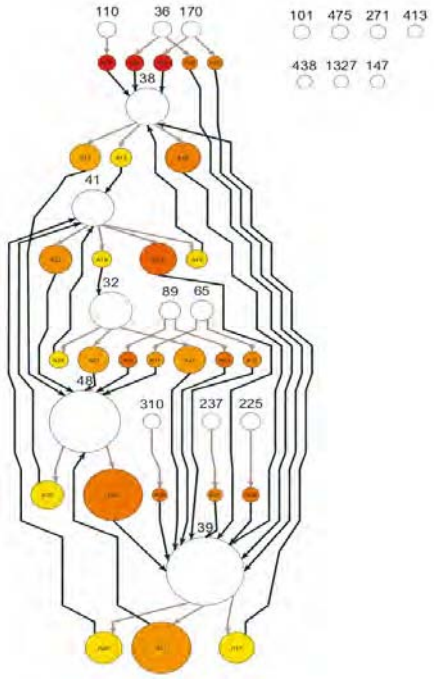


FIGURE 3.34 – Graphe hiérarchique de règles d'association par Ertek & Demiriz [Ertek 06].

Une flèche grise indique un lien de type prémisse, et une flèche noire indique un lien de type conclusion. Par exemple, sur la figure 3.34, la règle *A01* $(110 \Rightarrow 38)$ correspond au disque rouge en haut à gauche. Sa couleur élevée signifie que sa confiance est également élevée. Cette méthode est intéressante car elle montre de manière hiérarchique la construction des règles. De plus, le type de graphe limite l'enchevêtrement, ce qui procure une lisibilité, tant qu'il n'y a pas trop de cercles. Un inconvénient réside dans l'éloignement entre des items liés entre eux, et dans la limitation à des règles de type *one-to-one*.

Le TwoKey Plot

Le TwoKey Plot [Unwin 01] est une visualisation de type scatter plot, présentant la totalité des règles, dans laquelle le support est représenté par l'axe des abscisses et l'ordonnée par l'axe des ordonnées (Cf. Figure 3.35). Chaque point représente une règle, et il est possible de les relier graphiquement par la densité des points et des liens. La densité de couleur indique son ordre. La vue d'ensemble permet d'avoir une vision globale de l'ensemble des règles, de leur support et de leur confiance. Ainsi, des groupes de règles, ou des règles isolées sont facilement détectables. Des outils d'interaction sont proposés pour l'exploration des règles, tels que la requête, la sélection et le zoom. Un inconvénient est qu'il n'est pas aisé d'étudier une règle en détail.

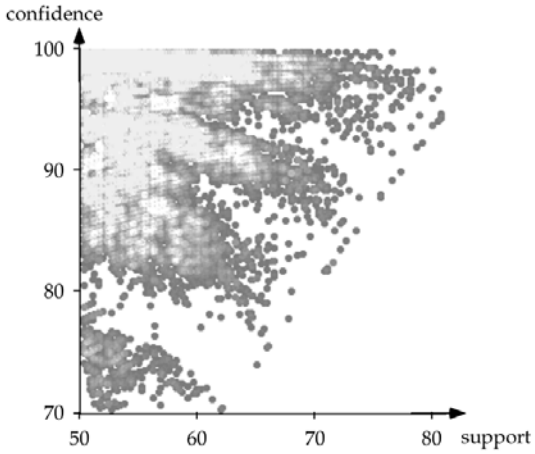


FIGURE 3.35 – TwoKey plot [Unwin 01].
Les règles sont représentées dans un espace 2D, en fonction de la confiance et du support.

Diagramme à deux étages (*Double-Decker Plot*)

Le diagramme en mosaïque (*MosaicPlot* [Hofmann 00a]) est une représentation d'un tableau de contingence, c'est-à-dire que chaque case est le résultat d'un comptage. Sa surface est proportionnelle à son effectif. Hofman et al. [Hofmann 00b] en présentent une variante, avec les Double Decker Plots (Cf. Figure 3.36). Le but est de représenter une règle d'association et les règles qui lui sont associées, à partir de tous les attributs de l'itemset les composant. Le support est représenté par la surface de la case, alors que la confiance est représentée par la quantité de rouge dans chaque case. Ainsi, sur cette figure, la confiance de la règle $Heineken, Coke, Chicken \Rightarrow Sardines$ est presque égale à 100%, alors que toutes les autres règles ont une confiance inférieure à 40%. L'avantage de cette approche est qu'elle permet de comparer toutes les règles issues d'un même itemset. De plus, elle ne se limite pas à des prémisses d'un ou de deux éléments. Cependant elle ne permet pas de visualiser la totalité des règles générées.

Coordonnées parallèles

Nous avons vu dans le chapitre 3.2.1 que Yang [Yang 03] visualise des itemsets fréquents avec des coordonnées parallèles. Il reprend ce principe pour les règles d'association. Le support est représenté par l'épaisseur de la ligne et la confiance par la couleur. La figure 3.37 (a) montre des règles dont la prémisse contient deux items, et la conclusion un seul item. Un avantage de cette représentation, est qu'il n'y a pas de limite dans le nombre d'items des prémisses et des conclusions. Cependant, l'inconvénient est que la visualisation devient rapidement illisible quand le nombre de règles et d'itemsets augmente. Il a amélioré ce concept [Yang 05] en utilisant des courbes de Béziers au lieu de segments et en rassemblant les items dans des groupes, afin d'améliorer la lisibilité du graphe (Cf. Figure 3.37 (b)). La même représentation du support et de la confiance a été maintenue.

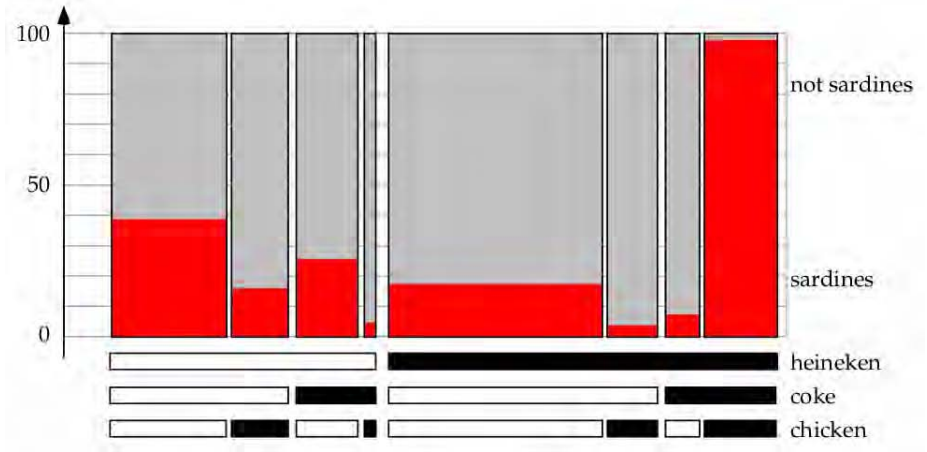
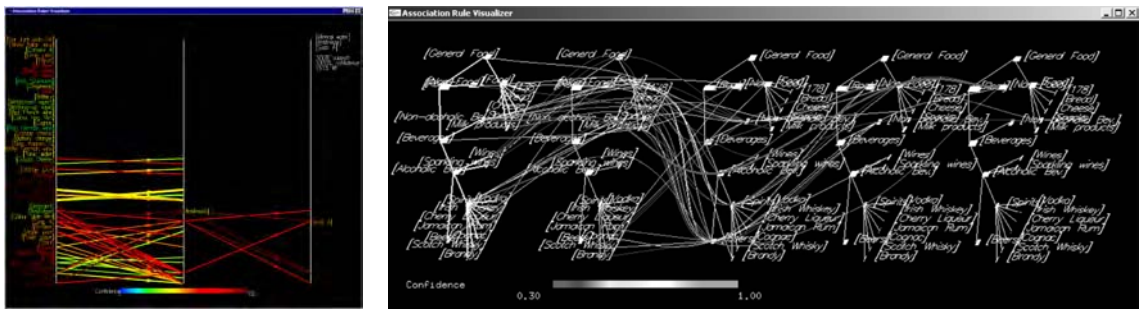


FIGURE 3.36 – Visualisation d'une règle avec Double-Decker Plot [Hofmann 00b].
Les barres inférieures montrent les attributs. La surface des cases correspond au support, la confiance est représentée par la proportion de rouge dans une case.



(a) (b)

FIGURE 3.37 – Représentation de règles avec des coordonnées parallèles.
(a) : les items sont reliés par des segments [Yang 03]. (b) : ils sont regroupés par classe, et sont reliés par des courbes de Béziers [Yang 05].

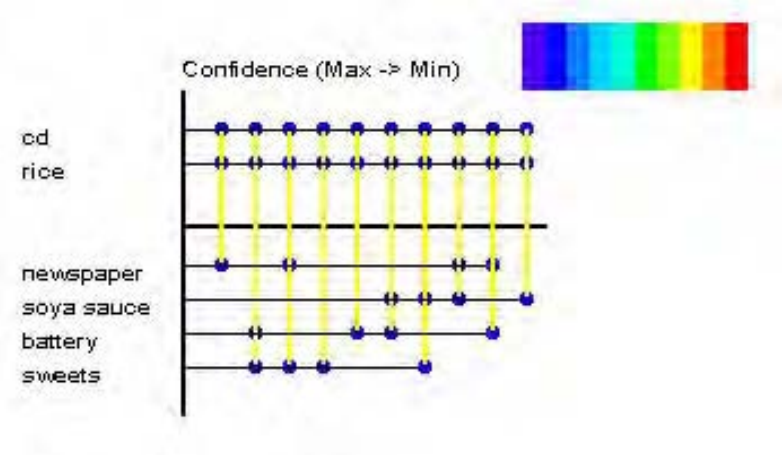


FIGURE 3.38 – La représentation de règles d'association avec VisAR [Techapichetvanich 05]. Les règles sont représentées par des traits verticaux reliant les items. Le trait horizontal épais indique la séparation entre la prémisse et la conclusion.

L'approche VisAR

Cette approche [Techapichetvanich 05] ne rentre dans aucune catégorie précitée (Cf. Figure 3.38). Les items sont représentés par des lignes horizontales. Des lignes verticales, représentant les règles, coupent ces lignes. Les points d'intersections qui sont particularisés indiquent, d'une part les items impliqués dans les règles, et, d'autre part, s'ils se trouvent dans la prémisse ou dans la conclusion. Sur la figure 3.38, les items *cd* et *rice* ont été sélectionnés. VisAR montre les règles dans lesquelles ils constituent la prémisse. La couleur indique la confiance ou le support.

Les avantages de VisAR résident dans la possibilité de visualiser les règles quel que soit le nombre d'items dans la prémisse et dans la conclusion. De plus, il n'y a pas de limites dans le nombre d'items et dans le nombre de règles visualisées. Afficher les règles selon des traits verticaux parallèles, empêche tout risque d'illisibilité due à un enchevêtrement de segments. Un inconvénient est la nécessité de sélectionner d'abord les items que l'on souhaite faire apparaître dans la prémisse. VisAR ne visualise donc pas toutes les règles d'association simultanément.

3.3 Approche mixte

Après avoir abordé la visualisation des données (Cf. Chapitre 3.1) puis celle des résultats algorithmiques (Cf. Chapitre 3.2), l'approche mixte correspond à la visualisation améliorée, la fouille améliorée et la visualisation et la fouille intégrées [Bertini 09]. Elle se situe, au sein d'un continuum présenté par Bertini & Lalanne [Bertini 09], entre l'exploration exclusive des données, sans assistance d'algorithmes, et l'analyse uniquement automatique des données, sans l'aide d'aucune visualisation. Aujourd'hui, cette approche est fondamentale dans le Visual Analytics, parce qu'elle répond parfaitement à ses objectifs d'exploiter de manière optimale

les qualités de l'homme et celles de la machine. Des travaux précurseurs, dont quelques uns sont cités ci-dessous, avaient pressenti cette nécessité. Plus récemment, de nombreuses études montrent cette recherche d'intégration entre la visualisation et l'algorithme, tout en gardant, au cœur du processus, l'opérateur humain. L'intégration est notamment réalisée par la présence simultanée, ou au choix, de multiples techniques de visualisations (Cf. Chapitre 3.1.2). Après un court survol des travaux précurseurs, nous abordons dans ce chapitre quelques études illustrant cette intégration.

3.3.1 Quelques travaux précurseurs

Ils remontent à Tukey [Tukey 77] qui, dès 1977, a énoncé le besoin de passer de la représentation graphique de résultats à l'exploration visuelle interactive des données et des résultats. Cela a donné l'EDA (Exploratory Data Analysis), dont le but est de découvrir des motifs dans les données, grâce à des techniques de visualisation. En 1986, Mackinlay [Mackinlay 86] a proposé APT, un outil de représentation de graphiques à base de règles, pour automatiser le processus de visualisation. En 1999, Wong [Wong 99a] considère qu'un système de fouille visuelle de données ne doit pas exiger de connaissances de la part de l'utilisateur, mais doit le guider pour trouver des conclusions au travers du processus de fouille de données. Plus tard, Keim et al. [Keim 04] ont présenté trois types d'approche permettant d'intégrer l'homme dans la boucle (Cf. Chapitre 3.1 et Figure 3.2).

3.3.2 ViA, un assistant de visualisation

Healey et al. [Healey 08] ont réalisé ViA (Visualization Assistant) qui partage le processus de visualisation entre le système et l'utilisateur. Ce dernier renseigne préalablement le système sur l'importance et le type des attributs, ainsi que sur le type de tâche qu'il souhaite entreprendre. Par exemple, cela peut être une recherche sur l'évolution d'une zone de la visualisation dans le temps ou la détection de frontières entre différentes zones. Puis le système réalise des appariements entre les attributs des données et les variables visuelles. Chaque appariement est évalué, pour donner lieu à une visualisation qui est proposée à l'utilisateur. Il lui est ensuite possible de suggérer un nouveau paramétrage, qui provoquera un nouveau cycle et ajustera le paramétrage des visualisations.

3.3.3 MIDAVisT : explorateur visuel d'ensembles de données mixtes

Avec MIDAVisT [Johansson 09], Johansson présente un outil interactif pour l'analyse de données mixtes, c'est-à-dire numériques et nominales. Elle élabore, dans un premier temps, une catégorisation des données, grâce à une approche combinée, d'une part, algorithmique, et, d'autre part, adaptée par l'utilisateur à l'aide de la visualisation des données. En effet, ayant une connaissance de la base, ce que l'algorithme n'a pas, il peut faire des choix plus pertinents. Dans un second temps, l'outil de visualisation reprend des techniques présentées dans le chapitre 3.1.2, c'est-à-dire les coordonnées parallèles, les matrices scatter plot et les

FIGURE 3.39 – MiDAVisT [Johansson 09].

Table Lens. Ces trois visualisations, simultanées et coordonnées, permettent ainsi d'explorer de manière interactive les données.

3.3.4 Miner3D

Miner3D [15] est un produit commercial, distribué actuellement dans la version 7, décliné en plusieurs versions (Cf. Figure 3.40). Il combine la visualisation de données et la mise en œuvre d'algorithmes. Les techniques de visualisations sont multiples, comme le scatter plot, les matrices 3D, les matrices scatter plot, les lignes, les surfaces et les cartes de chaleur (*heatmap*). Cette dernière technique consiste à représenter, avec un gradient de couleur du bleu au jaune, puis au rouge, l'intensité des points relativement à la valeur maximale des données, pour une dimension donnée. Pour assister l'utilisateur, plusieurs sortes de filtres sont utilisables, comme les plages de valeur ou des données textuelles. Il est également assisté par des outils algorithmiques permettant, entre autres, le partitionnement en classes ou *clustering* (Cf. Chapitre 1.2.2), et l'analyse en composante principale pour réduire le nombre de dimensions. La version professionnelle propose des fonctionnalités adaptées aux domaines pharmaceutiques et biochimiques pour l'étude des molécules.

3.3.5 Text Mining et visualisation

Don et al. [Don 07] proposent une intégration de la visualisation dans la fouille de données textuelles (*Text Mining*). A partir d'une base de documents, le système détecte des répétitions

15. www.miner3D.com

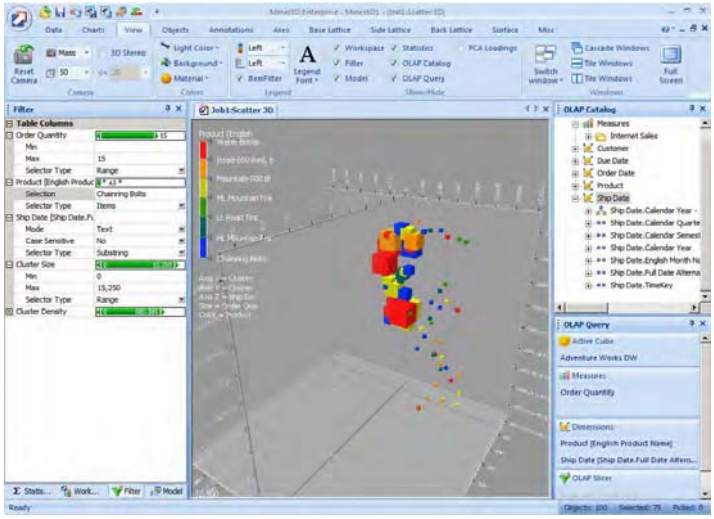


FIGURE 3.40 – L'outil commercial Miner3D.

de mots ou d'expressions, ou des similarités. Les résultats sont visualisés selon plusieurs techniques. Dans la partie inférieure (*Document Overview*) de la figure 3.41, les documents de la collection sont représentés sous forme de rectangles verticaux, dont chaque ligne colorée correspond à un paragraphe. En sélectionnant des mots dans la liste inférieure gauche, leurs occurrences sont visualisées, pour chaque paragraphe de chaque document, dans le *Document Overview*, selon un code de couleur permettant d'illustrer si ces occurrences sont importantes ou pas. Il est ainsi aisé de détecter visuellement la présence simultanée de plusieurs mots au sein d'un même paragraphe. En sélectionnant une ligne, le paragraphe correspondant est présenté dans la partie droite de l'outil.

Un graphique, donnant le nombre d'occurrences en fonction des documents, est représenté dans la partie supérieure de l'outil. Sur la gauche de celle-ci, il est possible de rechercher librement des mots qui n'ont pas été mis en valeur par l'algorithme initial, de trier les motifs par nombre d'occurrences et par nombre d'items. La partie *Trends* est constituée d'icônes permettant de détecter des profils dans l'évolution d'un mot, que ce soit au sein de l'ensemble des documents de la base, ou uniquement au sein d'un document.

3.3.6 Assistant utilisateur pour le paramétrage de la visualisation

Guettala et al. [Tahir Guettala 12] proposent un assistant utilisateur pour guider l'utilisateur dans le choix et le paramétrage de la visualisation (Cf. Figure 3.42). Cela se fait en deux étapes. Dans un premier temps, le système propose d'abord plusieurs appariements entre la base de données et les visualisations. Il s'appuie sur les objectifs de l'utilisateur, les caractéristiques des données, et les caractéristiques et l'importance des variables visuelles selon Bertin [Bertin 67] et Card et al. [Card 99]. Les données sont constituées d'attributs, chacun étant caractérisé par un type, une valeur et un degré d'importance fixé par l'utilisateur (vue 2D ou 3D, classes ou vue d'ensemble) ou de manière automatique. Les appariements sont

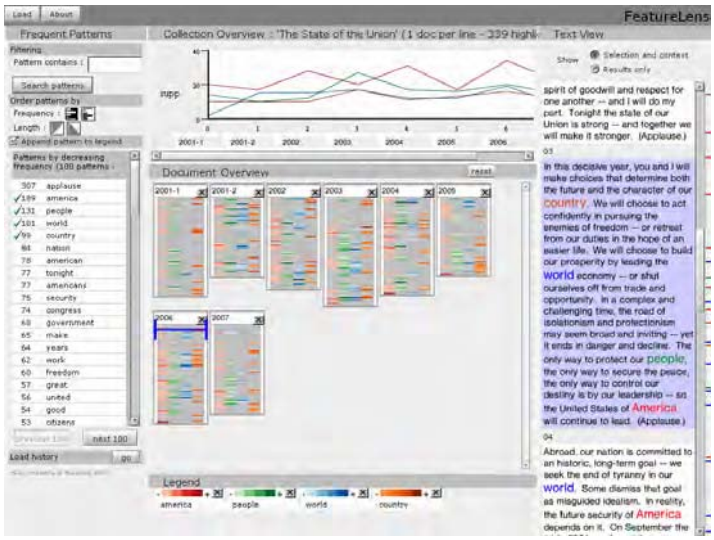


FIGURE 3.41 – FeatureLens : intégration du Text Mining et de la visualisation [Don 07].

réalisés à partir d'une base de connaissances sur les visualisations et sur la perception visuelle. A l'issue du traitement, huit vues sont proposées par ordre d'importance en fonction d'une valeur d'appariement. Puis l'utilisateur peut changer l'importance des variables visuelles, ce qui a un effet dans le résultat de l'appariement qui est à nouveau calculé.

3.4 Conclusion sur le Visual Data Mining

La fouille visuelle des données fait appel à leur visualisation et à l'interaction, ceci afin de détecter des motifs mettant en relation ces données. Elle s'appuie sur la visualisation d'informations, ou InfoVis, qui est un domaine permettant de représenter des données, en s'appuyant sur la perception visuelle et la sémiologie graphique [Bertin 67]. De nombreux travaux ont posé les bases de la visualisation moderne d'informations, dont émane l'InfoVis. Nous l'avons présentée en nous basant sur la classification orthogonale des techniques de Visual Data Mining de Keim & Ward [Keim 03]. Celle-ci repose sur trois critères : les types de données, les techniques de visualisation et les techniques d'interaction. Ces critères ont été illustrés en s'appuyant sur divers travaux, tant dans la visualisation d'informations que dans la visualisation des résultats de algorithmes.

Cependant, la visualisation n'est pas toujours suffisante, comme dans le cas de l'approche algorithmique (Cf. Chapitre 1.5.4). Pour exploiter au mieux l'apport de l'homme et celui de la machine, le Visual Analytics propose de mettre l'opérateur au cœur du système, afin de pourvoir manipuler les données, paramétrer, voire piloter, les algorithmes, et ainsi gagner une meilleure perspicacité et une meilleure connaissance des données. Quelques exemples d'approche mixte, préconisée par ce domaine, ont été présentés, afin d'illustrer, d'une part, comment des techniques de visualisations peuvent être intégrées au sein d'un même outil, et, d'autre part, comment peut être réalisée l'intégration des approches algorithmiques et

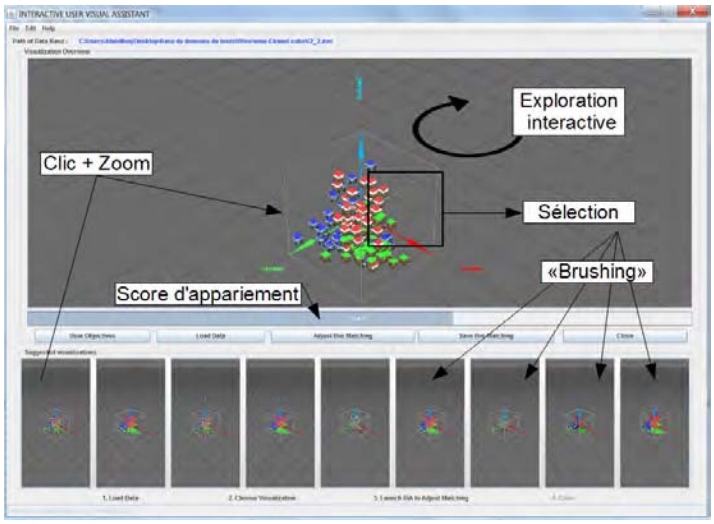


FIGURE 3.42 – Assistant utilisateur pour le choix et le paramétrage de la visualisation.

visuelles, chacune apportant à l'autre des possibilités impossibles à mettre en œuvre sans cette intégration.

Cet état de l'art essaye de montrer ainsi l'importance du lien à établir entre l'approche algorithmique et la visualisation des données, en permettant à l'homme et au système de combiner au mieux leurs capacités. Pour l'utilisateur, il s'agit, entre autres, de ses connaissances et de se créativité, et, pour le système, de sa capacité de stockage et de calcul. C'est ce type d'approche que nous avons adopté dans le cadre de ce travail de thèse, et que nous allons présenter dans la suite de ce document. Nous présenterons d'abord la manière dont nous pilotons l'algorithme de Data Mining à partir de l'exploration visuelle des données. Des motifs et des règles d'association sont ainsi élaborés, pour lesquels nous proposons des formes de visualisation. Puis, à partir de celles-ci, nous présentons un processus inverse, qui, à partir de ces visualisations de résultats algorithmiques, enrichit l'exploration des données initiales, et la configure automatiquement.

Troisième partie

Problématique

Ce travail de thèse s'inscrit dans un contexte d'expansion de la production de données dans divers domaines, comme la santé, les réseaux sociaux, le marketing... Les évolutions en cours et annoncées dans la gestion du trafic aérien, autant du côté des exploitants que de celui des fournisseurs de services de contrôle, montrent que ce domaine doit également faire face à cette tendance. De plus, cette production croissante, non seulement due à une augmentation du trafic, est nécessaire, parce qu'elle répond aux besoins et aux exigences des nouveaux systèmes. Les besoins sont par exemple ceux qui ont été mis en exergue lors d'événements récents, comme la disparition du vol MH370 en mars 2014. La nécessité d'accroître le volume de données échangées entre les systèmes bord et les systèmes sol a été exprimée, notamment par les organismes français, et entérinée par l'OACI. Les exigences sont celles des futurs systèmes de gestion du trafic aérien, qui nécessitent un échange permanent de données pour fournir une connaissance, partagée entre les différents acteurs, la plus précise possible, afin d'optimiser les flux.

Les données aéronautiques constituent le contexte de ce travail de recherche en Visual Analytics. Ce domaine émergent, de l'Extraction des Connaissances à partir des Données, établit le lien entre la recherche d'informations par le biais d'algorithmes, et la fouille visuelle des données. Ces deux approches sont complémentaires parce qu'elles fournissent simultanément la puissance de la machine et l'expertise humaine. L'état de l'art a présenté les recherches dans les méthodes algorithmiques, en particulier dans l'extraction de règles d'associations, et celles dans la fouille visuelle des données. Cette dernière a été étendue à l'exploration visuelle des résultats algorithmiques.

Notre travail a porté sur l'approche mixte qui combine l'algorithme et la visualisation, afin de proposer des éléments de réponses aux défis soulevés par le Visual Analytics et VisMaster. Ainsi, notre problématique est d'explorer des voies pour améliorer la collaboration entre l'homme et le système, en combinant leurs apports respectifs, tout en laissant à l'humain la maîtrise globale du processus d'extraction de données.

L'opérateur humain dirige le processus global d'extraction de connaissances, car, grâce à son expertise, sa compréhension de la question posée lors d'une recherche d'informations, et sa connaissance des données, il est le mieux placé pour appréhender celles-ci. Le potentiel d'expression graphique que confère une représentation visuelle de l'information nous fait privilégier l'exploration visuelle des informations pour une première approche des données. Celle-ci, quand elle est configurable, reflète l'intérêt de l'utilisateur par la manière dont les données sont représentées. Cela constitue le point de départ de notre recherche. Par ailleurs, dans les recherches en Data Mining, l'approche mixte est actuellement peu exploitée. Cependant, à l'issue de l'utilisation d'algorithmes, il serait intéressant d'en faire le lien avec une visualisation des données, parce qu'elle est plus proche de l'information brute initiale. Enfin, une étude de données peut posséder un caractère collaboratif, par l'intervention éventuelle de plusieurs personnes agissant à différentes étapes du processus, comme lors du prétraitement, de l'exploration des données, ou de l'interprétation des résultats. C'est pourquoi, cet aspect a fait partie du périmètre de notre recherche.

Ainsi, pour répondre à la problématique, nous la reformulons sous la forme de quatre

interrogations :

- Comment l'exploration visuelle des données peut-elle être exploitée pour piloter un algorithme de fouille de données ?
- Quelles seraient de nouvelles manières de présenter des résultats algorithmiques, en mettant en valeur les mesures qui les caractérisent ?
- Comment combiner la visualisation des données et les résultats algorithmiques ?
- Dans quelle mesure cette combinaison peut être mise en œuvre dans un système ouvert et évolutif ?

Afin de répondre à ces questions, notre recherche a porté sur plusieurs axes :

- Le pilotage de l'algorithme de fouille de données à partir de l'exploration visuelle. Pour cela, nous considérons une représentation multidimensionnelle et interactive des données ;
- L'exploration des résultats algorithmiques, d'une part, en exploitant leurs caractéristiques, comme les mesures de qualité, et d'autre part en considérant la mise à l'échelle. Nous étudions de nouvelles méthodes d'exploration visuelle des itemsets et des règles d'association ;
- L'enrichissement et le pilotage de la visualisation des données par les résultats algorithmiques ;
- La mise en œuvre d'une plate-forme collaborative et évolutive.

Ces différents axes sont traités dans les parties suivantes du mémoire. Dans un premier temps, une approche théorique établit un premier lien de la visualisation des données vers les résultats algorithmiques (Cf. Chapitre 4), puis un second lien de ces résultats vers la visualisation initiale, pour l'enrichir ou la paramétrer automatiquement (Cf. Chapitre 5). Ensuite, nous proposons des visualisations de résultats de fouilles des données, pour les itemsets (Cf. Chapitres 7 à 9) et pour les règles (Cf. Chapitre 10). Enfin, une plate-forme est présentée au chapitre 12, intégrant l'exploration interactive des données et des résultats algorithmiques, ainsi que la mise en œuvre des algorithmes. Sa mise en œuvre est illustrée, dans deux scénarios au chapitre 14, par l'exploitation de données aéronautiques.

Tout au long de cette recherche, le rôle central et décisionnel de l'utilisateur est pris en compte, en lui procurant, à chaque étape du processus, des possibilités de réaliser des choix influant sur les étapes suivantes.

Quatrième partie

Liens entre la fouille visuelle de données et la fouille automatique de données

Introduction

Dans cette partie théorique, nous établissons un lien bilatéral entre les fouilles de données algorithmiques et visuelles. Cette approche commence par une exploration visuelle des données, qui va piloter l'algorithme. En nous appuyant sur l'état de l'art, notamment dans les domaines de la perception, de la sémiologie graphique et de la caractérisation des visualisations, nous formalisons celle-ci en partant des données initiales, pour aboutir aux données finales qui vont être prises en compte dans l'algorithme de fouille de données. Après un prétraitement, les données initiales sont explorées visuellement par l'utilisateur, grâce à l'affectation de leurs attributs à des variables visuelles. L'exploration permet de n'en garder qu'une partie dans la vue, par exemple par un effet de zoom, ou par des filtrages, ce qui exclut les données qui ne sont pas visualisées. De plus, des opérateurs de sélection limitent encore plus le nombre de données retenues. A l'issue de cette phase exploratoire, une partie de la base initiale, ainsi obtenue, constitue l'ensemble des données qui seront traitées. Comme la configuration de l'outil de visualisation est dépendante du besoin de l'utilisateur, la manière de présenter les données, que ce soit par les choix d'affichage des attributs, les filtrages et le réglage de la visualisation, dépendent de l'importance qu'il donne à ces données. Ces choix conditionnent ensuite l'algorithme qui prend en compte le résultat de l'exploration visuelle. Ainsi, il est piloté par les données, et donc par l'utilisateur ainsi que par son expertise.

Nous considérons que la base de données est hétérogène, c'est-à-dire qu'elles peuvent être de plusieurs types : nominales, entières ou décimales. Pour les besoins de l'algorithme de fouille, il est donc nécessaire de les catégoriser. Un algorithme de classification non-supervisée permet ainsi de les partitionner en différentes classes qui seront les items en entrée de l'algorithme. Comme le système ne dispose pas de l'expertise de l'utilisateur, celui-ci peut ensuite intervenir sur ces classes, afin de les rendre plus pertinentes, en fonction de sa problématique.

Dans un troisième temps, l'algorithme, choisi par l'utilisateur, produit des motifs, c'est-à-dire des groupes d'attributs de données présents simultanément dans les transactions. A partir de ces motifs, appelés itemsets, des règles d'association sont calculées. Les itemsets et les règles sont caractérisés par des mesures qui permettent de les valoriser et de les exploiter dans la suite du processus, ou au contraire de les ignorer, cela relevant du choix de l'utilisateur. Les résultats des algorithmes sont exploitables par une exploration, ou en les associant au paramétrage de la visualisation des données, ce qui permet de l'enrichir.

Un processus inverse, allant des règles d'association vers la visualisation, est présenté dans

le second chapitre de cette partie. Il consiste, d'une part, à générer automatiquement une visualisation à partir d'un ensemble de règles choisies par l'utilisateur, et, d'autre part, à générer automatiquement plusieurs visualisations à partir de l'ensemble des règles d'association. Pour cela, les mesures qui leur sont associées sont exploitées. En enchaînant ce processus avec le premier, allant de la visualisation vers les règles, il est alors possible d'effectuer des allers-retours itératifs entre les données et les règles.

Le lien de la visualisation des données vers les règles d'association, ainsi que l'enrichissement de la visualisation par les résultats algorithmiques, ont fait l'objet de publications aux conférences SCCG 2013 [Bothorel 13a] et ISIATM 2013 [Bothorel 13b]. Ils seront illustrés, en s'appuyant sur des données aéronautiques, dans la partie VI.

Chapitre 4

Pilotage de l'algorithme de fouille de données par la visualisation

Ce chapitre présente le pilotage de l'algorithme de fouille de données par la visualisation, qui est le résultat de choix effectués par l'utilisateur.

La démarche que nous présentons est constituée des étapes suivantes :

- Formalisation de la visualisation, pour faire le lien entre les données de la base et celles qui sont visualisables ;
- Sélection des données qui seront traitées par l'algorithme. Elles sont le résultat, par l'exploration visuelle, de filtrages, de zooms, d'excentrements et de sélections ;
- Discrétisation des données, pour obtenir des classes ;
- Calcul des itemsets fréquents ;
- Extraction des règles d'association, en contraignant leur structure en fonction de l'importance des variables visuelles, qui suggèrent un ordre dans les attributs.

A partir des résultats algorithmiques, nous proposons, dans le dernier paragraphe, une méthode d'illustration des règles par les données, en enrichissant ces dernières par des mesures associées aux résultats.

Dans la formalisation, nous considérerons les variables visuelles de position X, Y et Z ce dernier dans le cas d'une visualisation tridimensionnelle, la taille du point S, la couleur C et l'alpha A (Cf. Chapitre 2.2). Les ensembles suivants seront par ailleurs introduits :

- $\mathcal{A}$ est l'ensemble des m attributs possibles ;
- $\mathcal{X}$ est l'espace de tous les vecteurs possibles construits à partir des attributs ;
- $\mathcal{D}$ est la base de données, incluse dans $\mathcal{X}$; elle contient n tuples ;
- $\mathcal{V}$ est l'ensemble des q variables visuelles ;
- $\mathcal{P}$ est l'ensemble des points visualisables.

4.1 Formalisation des visualisations

Le modèle de Card & Mackinlay [Card 97] décrit essentiellement la correspondance entre les attributs et les variables visuelles, et l'organisation de la visualisation (cf. Chapitre 2.3). Le processus, qui part des données pour arriver à ce qui est effectivement visualisé, correspondant à la notion de filtre qui produit les données D'. C'est pourquoi, nous proposons une formalisation des données et des variables visuelles, pour établir le lien entre les données initiales et leur visualisation. Nous appelons données initiales les données, non pas brutes, issues par exemple des enregistrements, mais celles qui sont issues des prétraitements. Nous nous plaçons par ailleurs dans la situation de notre outil de visualisation de type scatter plot, sachant que des données peuvent être reliées entre elles par des lignes (Cf. Chapitre 12.3). Durant la présentation de la formalisation, nous illustrerons notre propos à l'aide d'un exemple de données caractérisant des voitures.

Une base de données, ou de transactions, est un ensemble de vecteurs constitués d'attributs issus d'un espace d'attributs. Chaque attribut A_i de l'ensemble $\mathcal{A}$ peut être numérique (entier ou décimal) ou nominal (nom...) En fonction de son type, un attribut peut être ordonné ou non. L'espace $\mathcal{X}$ est l'espace de tous les vecteurs possibles construits à partir des attributs. Etant donné $x \in \mathcal{X}$, nous avons $x =< a_1, a_2, ..., a_m >$, où a_j est la valeur de l'attribut A_j. Une base de données $\mathcal{D}$ est un sous-ensemble de $\mathcal{X}$: $\mathcal{D} = \{x_1, x_2, ..., x_n\}$, où $\forall i, x_i \in \mathcal{X}$ et n est le nombre de tuples. Une donnée $x_i =< a_{i1}, a_{i2}, ..., a_{im} >$ est un vecteur, où a_{ij} est la valeur de l'attribut A_j pour la donnée i. La base $\mathcal{D}$ peut être éventuellement réduite en fonction de critères de filtrages qui restreignent certaines valeurs d'attributs. Cependant, le filtrage pourrait être également réalisé par une interaction. Dans ce cas, il s'agirait d'une interaction supplémentaire qui serait complémentaire avec la sélection.

Dans le cas de notre exemple de voitures, $\mathcal{A}$ contient les attributs *marque*, *modèle*, *cylindrée*, *puissance*, *couleur*, *nombre de places*... $\mathcal{X}$ contient toutes les combinaisons possibles de ces attributs. Nous pouvons ainsi avoir {Peugeot, 208, 4, 65, vert, 5, ... } ou {Renault, Clio, 6, 150, bleu, 8, ... }. La base de données $\mathcal{D}$ est un sous-ensemble de $\mathcal{X}$, ou peut être égale à $\mathcal{X}$ selon le cas. Avec l'exemple des voitures, ces deux ensembles ne sont pas égaux, car certains vecteurs ne sont pas possibles, comme le second exemple correspondant à une Clio de 150 CV ayant 8 places assises. Un exemple de filtrage des données, peut être de ne considérer que les voitures de marques françaises.

La visualisation des données est basée sur l'instanciation des variables visuelles en fonction des attributs. Les variables visuelles de l'ensemble $\mathcal{V}$ peuvent être de différents types, comme la position dans l'espace, la taille, la couleur et la transparence. D'après Bertin [Bertin 67], une variable visuelle peut être ordonnée (position, taille, transparence) ou pas (couleur). Un point p est un vecteur dont les composantes sont des variables visuelles. Nous appelons $\mathcal{P}$ l'espace de ces points.

A partir de ces considérations, nous définissons une visualisation.

Définition 5 *(map) Nous appelons map la fonction qui associe l'index d'une variable visuelle*

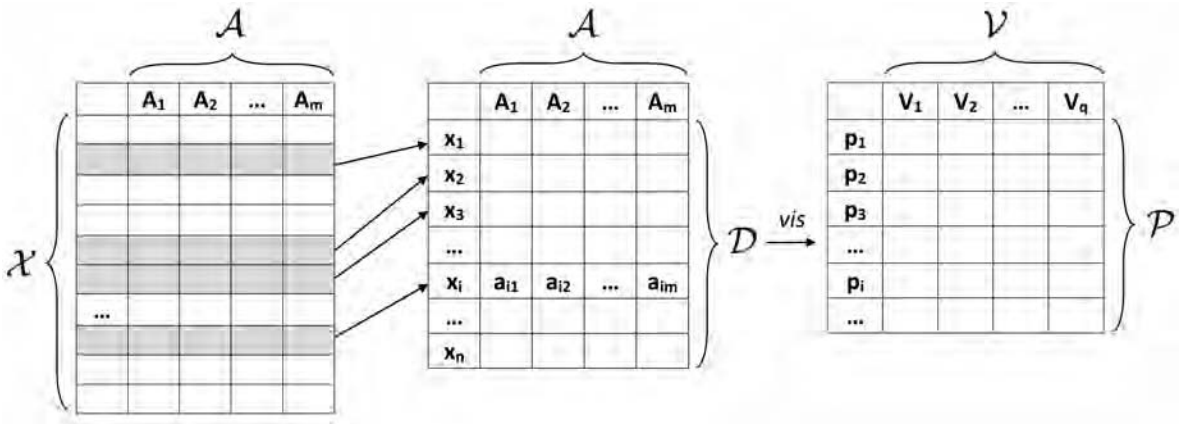


FIGURE 4.1 – Des attributs des données à la visualisation.
Gauche : ensemble $\mathcal{X}$ de tous les vecteurs possibles à partir des attributs de $\mathcal{A}$. Milieu : la base de données $\mathcal{D}$ est un sous-ensemble de $\mathcal{X}$. Droite : espace $\mathcal{P}$ des points dont les composantes sont des variables visuelles de l'espace $\mathcal{V}$.

à l'index de l'attribut correspondant.

Il n'est pas nécessaire que *map* soit une injection, parce qu'un attribut peut être associé à deux variables visuelles distinctes. Chaque variable visuelle V_i est alors associée à un attribut $A_{map(i)}$. Ainsi, nous obtenons un ensemble de q paires $(V_i, A_{map(i)})$.

Définition 6 *(vis) Une visualisation vis est une fonction de $\mathcal{X}$ vers $\mathcal{P}$ (i.e. $vis : \mathcal{X} \to \mathcal{P}$) élaborée de la manière suivante :*

- *map définit une fonction de $\mathcal{V}$ vers $\mathcal{A}$.*
- *Pour chaque paire $(V_i, A_{map(i)})$, nous avons une fonction $f_i : A_{map(i)} \to V_i$.*

Alors, étant donné $x \in \mathcal{X}$, nous avons :

$$vis(x) =< f_1(a_{map(1)}), \ldots, f_q(a_{map(q)}) > .$$

Avec l'exemple des voitures, supposons que l'on souhaite visualiser dans un espace bidimensionnel, les modèles en abscisse, et les puissances en ordonnée. Le modèle étant le deuxième attribut, ses valeurs possibles appartiennent à l'ensemble A_2. De même, les puissances appartiennent à A_4. Supposons que l'abscisse corresponde à la première variable visuelle, ses valeurs sont donc dans l'ensemble V_1. De la même manière les valeurs des ordonnées sont issues de V_2. Il existe donc une relation entre A_2 et V_1, et entre A_4 et V_2. Donc $map(1) = 2$, et $map(2) = 4$. Ainsi, la variable visuelle V_1, correspondant aux abscisses, est associée à l'attribut $A_{map(1)} = A_2$, c'est-à-dire aux modèles de voitures. De même, l'ordonnée V_2 est associée à $A_{map(2)} = A_4$, c'est-à-dire à la puissance. Pour les paires (V_1, A_2) et (V_2, A_4), nous avons donc deux fonctions respectivement f_1 et f_2 permettant d'obtenir, par exemple dans le cas de la première, pour toute valeur d'un attribut de l'ensemble A_2, c'est-à-dire pour tout modèle de voiture, une valeur de V_1, c'est-à-dire une abscisse.

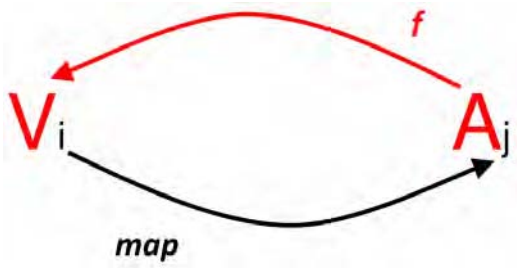


FIGURE 4.2 – Représentation schématique des fonctions *map* et f.
map associe l'index de la variable visuelle à celui de l'attribut. f associe l'attribut à la variable visuelle.

Ainsi, étant donnée une visualisation, un point affiché est caractérisé par ses variables visuelles, chacune correspondant à la valeur d'un attribut. Il n'est pas nécessaire d'utiliser toutes les variables visuelles possibles. L'utilisateur peut choisir, par exemple, de visualiser uniquement les points dans le plan. De plus, le nombre de variables visuelles doit être inférieur ou égal au nombre d'attributs. Notons que la fonction f_i dépend du type de variable visuelle et de l'attribut. Elle peut être par exemple une projection linéaire, en considérant les attributs numériques dans un espace de coordonnées cartésiennes, une fonction gradient, etc.

Etant donnée une visualisation, ce qui va être affiché dépend ensuite de la restriction qui est appliquée aux variables visuelles. Dans ce contexte, nous définissons la scène.

Définition 7 *(sc) Une scène sc est un sous-ensemble de $\mathcal{P}$ qui détermine quel point peut être effectivement affiché sur le périphérique. Nous avons :*

$$sc \subseteq \mathcal{P}.$$

Ainsi, la combinaison d'une visualisation *vis*, appliquée à un ensemble de données et à une scène *sc*, définit une image qui peut être affichée sur un périphérique, selon une projection du point considéré. En pratique, la scène dépend des différentes actions de l'utilisateur. Celles-ci peuvent être une interaction de type zoom/excentrement ou un positionnement de la caméra qui restreint la zone de l'espace à considérer. Enfin, la sélection de données (picking, brushing...) est également prise en compte par la scène.

Ainsi, le modèle de Card & Mackinlay est géré par la fonction *map* de la visualisation. Le choix de fonctions *map* et f pertinentes peut être piloté par les caractéristiques des variables visuelles décrites dans la sémiologie graphique. Un processus d'exploration visuelle de données consiste en une série de visualisations $vis_1, \ldots, vis_p$ en relation avec une série de scènes $sc_1, \ldots, sc_p$ qui correspondent à une série de vues et d'opérateurs appliqués à ces vues. Dans la suite du processus, nous nous intéressons uniquement à la caractérisation de la dernière visualisation et de la dernière scène.

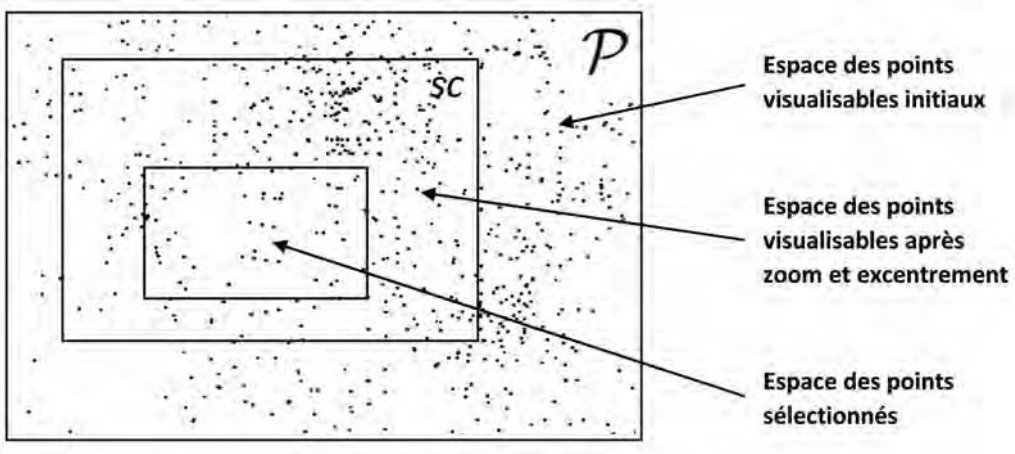


FIGURE 4.3 – Des données visualisables $\mathcal{P}$ à la sélection.
Dans un premier temps, $\mathcal{P}$ est réduit à *sc* par les opérateurs zoom et excentrement. Dans un second temps, les données sont sélectionnées pour être traitées par l'algorithme de Data Mining.

4.2 Sélection de données

Il existe plusieurs manières de sélectionner les données qui vont délimiter le périmètre de la scène *sc* :

- Par des filtrages appliqués aux attributs, afin de les restreindre. Il peut être en effet intéressant de réduire leur espace de valeurs ;
- Par un réglage de l'image à l'aide du zoom et de l'excentrement. Cela permet de ne garder que les données qui sont affichées, plus précisément celles qui sont dans les limites de la vue. En effet, nous ne prenons pas en compte l'occlusion qui va masquer des données de manière partielle ou complète. Bien qu'elles ne soient pas visibles, car affichées derrière d'autres données au premier plan, nous les gardons dans la sélection ;
- Par la sélection manuelle des données, selon des techniques d'interactions (Cf. Chapitre 3.1.3). Nous considérons l'opérateur de type brushing qui permet de sélectionner des données par leur survol avec un disque centré sur le curseur de la souris, ou par délimitation d'une zone rectangulaire à l'aide de celle-ci. Un autre type de sélection sera également mis en œuvre, basé sur des requêtes textuelles afin de pallier des difficultés de sélection manuelle. Ce type de sélection est appliqué aux attributs de type alphanumérique. Nous considérons, par ailleurs, que les opérateurs de sélection peuvent être combinés entre eux, et qu'une sélection peut être réduite par brushing.

En fonction de ses besoins, l'utilisateur assigne des attributs des données aux variables visuelles. Puis, il règle la représentation des données par zoom et excentrement, en ayant éventuellement filtré les données. La totalité des données ainsi visualisées peut constituer l'espace des données pris en compte dans la phase algorithmique suivante. Si ce n'est pas le cas, des opérateurs de sélection peuvent restreindre cet espace. Nous avons évoqué plus

haut une série de visualisations et de scènes élaborés par l'utilisateur durant l'exploration des données. Par un effet de linking (Cf. Chapitre 3.1.3), les scènes sont liées entre elles, car elles correspondent aux mêmes données représentées de manières différentes. Des sélections peuvent donc être réalisées dans ces scènes intermédiaires, ce qui réduit ainsi l'espace de visualisation. Elles peuvent notamment être appliquées à des attributs qui ne sont pas représentés dans la visualisation finale Ainsi, dans la figure 4.4, les données en rouge ont été sélectionnées dans la vue de gauche en prenant en compte les attributs A_1 et A_2. Dans la vue de droite, A_1 a été remplacé par A_3. La sélection reste active et les points rouges ont changé d'abscisse.

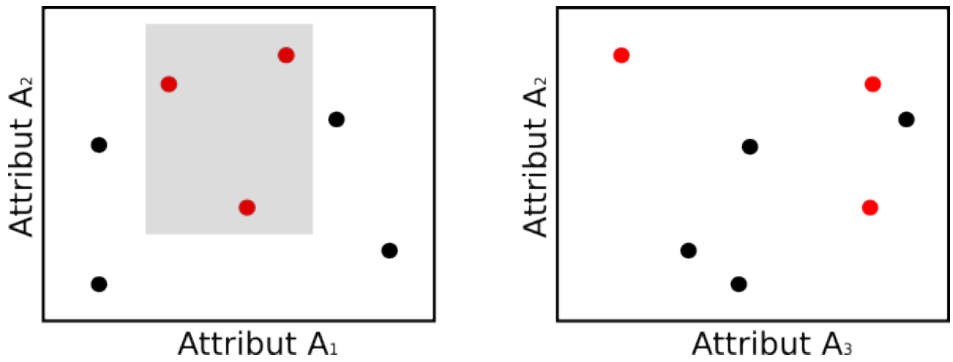


FIGURE 4.4 – Grâce au linking, la sélection des données est répercutée dans les différentes vues.

L'algorithme de Data Mining est appliqué au sous-ensemble des données visualisées $\mathcal{D}' \subseteq \mathcal{D}$, qui a été réduit par des opérateurs tels que le filtrage, le zoom, l'excentrement et la sélection.

Etant donnée une visualisation vis et une scène sc, $\mathcal{D}'$ est défini de la manière suivante :

$$\mathcal{D}' = \{x \in \mathcal{D} | vis(x) \in sc\}.$$

Cette sélection a des conséquences sur l'algorithme :

- Le zoom, l'excentrement et les opérateurs de sélection agissent en complément du filtrage. Comme le support et la confiance sont calculés à partir de cette sélection, l'algorithme ne pourra extraire des règles localement pertinentes que pour cette sélection, même si ce n'est pas le cas pour la base de données complète. La notion de règle d'association locale s'applique ainsi à une partie de la base de données ;
- Comme le nombre de données est diminué du fait de la sélection, la complexité de l'algorithme décroît en conséquence (Cf. Chapitre 1.5.1). Dans le cas d'Apriori, la décroissance est linéaire, d'un facteur lié à la proportion de données sélectionnées. Cela provient de ce que, puisque $\mathcal{D}' \subseteq \mathcal{D}$, nous avons alors $n' = |\mathcal{D}'| \leq n$;
- Si la sélection peut être simplement décrite, ajouter cette description dans la prémisse d'une règle d'association locale devient pertinente. Cependant, il n'est pas toujours facile de décrire simplement cette sélection, particulièrement quand elle est réalisée à l'aide de l'opérateur brushing utilisé comme un pinceau (Cf. Chapitre 3.1.3).

4.3 Discrétisation

Dans le but de trouver des règles d'association, les données numériques doivent être discrétisées, ainsi que les données nominales, si celles-ci sont en trop grand nombre. En effet, la complexité d'un algorithme croît fortement avec le nombre d'attribut. Dans le cas d'Apriori, cette croissance est exponentielle (Cf. Chapitre 1.5.1). Les attributs doivent donc être regroupés en catégories correspondant à des ensembles de valeurs. Comme le but de notre approche est de caractériser la visualisation, nous appliquons un algorithme de regroupement automatique des données, à partir des valeurs des variables visuelles, plutôt qu'à partir des valeurs des données initiales. Plus formellement, cela signifie que pour un attribut A_i, le groupe est constitué à partir des valeurs de $f_i(A_i)$. Comme f_i peut ne pas être linéaire, et dynamiquement modifié par l'utilisateur, il assure qu'il produira des règles d'association qui ont du sens dans la visualisation choisie par l'utilisateur.

L'algorithme que nous mettons en œuvre pour la discrétisation est celui des k-moyennes ou k-means [MacQueen 67]. Considérant un nuage de points, il consiste à choisir préalablement k positions, de manière aléatoire. Ces positions, appelées *centroïdes*, servent de barycentre des k groupes d'éléments, ou *clusters*, qui sont ensuite constitués autour d'eux. Pour cela, une distance a été préalablement définie. Pour chaque groupe, un nouveau centroïde est recalculé. Puis un nouveau calcul des groupes est réalisé, afin de redistribuer les éléments autour de ces centroïdes. L'algorithme s'arrête quand les groupes ne sont plus modifiés. Un autre algorithme similaire est k-médoïdes (*k-medoids*) [Hartigan 75], qui consiste à considérer comme centroïdes des points issus de la distribution initiale. Ces points sont appelés *médoïdes*.

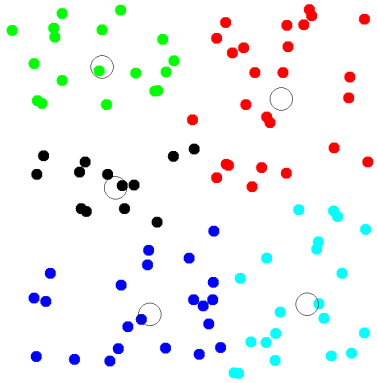


FIGURE 4.5 – Exemple de k-means, avec 5 centroïdes [MacQueen 67].
Les centroïdes sont représentés par des cercles.

Afin de déterminer le nombre de centroïdes, nous utilisons l'indice de Caliński-Harabasz [Caliński 74]. A partir du résultat d'un algorithme de partitionnement, comme k-means, l'indice est défini comme suit :

$$CH = \frac{n-k}{k-1} \frac{\sum_{c=1}^{k} n_c ||M_c - G||^2}{\sum_{c=1}^{k} \sum_{i \in I_c} ||P_{ic} - M_c||^2}$$

k est le nombre de clusters, n le nombre de points, M_c le centroïde du cluster I_c, P_{ic} le point i du cluster I_c et G le barycentre de l'ensemble des points.

La somme du numérateur indique la dispersion intra-clusters. Cette valeur doit être la plus élevée possible. La double somme du dénominateur donne la somme des dispersions intra-clusters. Cette valeur doit être la plus faible possible. Ainsi, plus les distances intra-clusters sont petites et les distances extra-clusters sont grandes, plus l'indice est élevé. Il est donc calculé pour plusieurs valeurs de k, et le regroupement de données retenu correspond à la valeur de k la plus importante.

Nous avons vu, dans le chapitre 1.4, que, dans le domaine du Visual Analytics, l'utilisateur doit pouvoir garder le pouvoir de décision afin d'orienter le processus. Pour cela, nous considérons qu'il lui est possible de remettre en cause le résultat des regroupements algorithmiques, notamment en redimensionnant les clusters calculés, ou en regroupant certains. En effet, il peut avoir un intérêt à le faire, étant donnée sa connaissance des données et du domaine d'application.

4.4 Calcul des itemsets fréquents

Si l'algorithme est appliqué à l'ensemble global des attributs, il peut en résulter une explosion combinatoire de nombre de règles d'association trouvées, ainsi qu'un temps de calcul très long. Comme nous l'avons vu précédemment, le nombre de variables visuelles contraint le nombre d'attributs et la taille des itemsets fréquents, ainsi que leur contenu. Du point de vue de l'utilisateur, il existe une réelle limitation car il est cognitivement difficile d'appréhender simultanément un grand nombre de dimensions [Miller 56]. Ainsi, l'algorithme ne prend en considération qu'un sous-ensemble d'attributs $\mathcal{A}' \subseteq \mathcal{A}$.

Étant donnée une visualisation vis, $\mathcal{A}'$ est défini de la manière suivante :

$$\mathcal{A}' = \{A_i \in \mathcal{A} | \exists V_j \in \mathcal{V}, map(j) = i\}.$$

La complexité d'Apriori est réduite d'un facteur $2^{m-|\mathcal{A}'|}$ et le nombre de règles est également naturellement réduit. Comme map n'est pas nécessairement une injection, et que le nombre de variables visuelles est limité à six dans notre étude, nous avons $|\mathcal{A}'| \leq 6$.

Cela nous permet de ne présenter que les règles d'association intéressantes pour l'utilisateur, dans un temps acceptable. Ce point est critique, parce qu'en considérant tous les attributs, le nombre de règles peut être très élevé et peut même dépasser le nombre de données initiales (Cf. Chapitre 3.2.2). Ainsi, le choix des variables visuelles par l'utilisateur détermine les données, plus précisément les attributs, qui sont soumis à l'algorithme, et cela permet de se focaliser sur les règles qui vont intéresser l'utilisateur. Ce choix correspond à la fonction map qui met en relation l'index des variables visuelles et celui des attributs correspondants.

4.5 Restriction des règles d'association

Nous avons vu que les règles d'association sont fortement dépendantes des données et des attributs sélectionnés. Cependant, la manière dont la correspondance est faite entre les variables visuelles et les attributs, et la manière dont les données sont affichées, que ce soit en vue orthographique ou en projection 3D, cela peut être utilisé pour contraindre la structure des règles d'association. Pour cela, nous nous appuyons sur l'organisation des variables visuelles, présentée dans la sémiologie graphique [Bertin 83] (Cf. Chapitre 2.2). Nous devons donc nous poser la question de la répartition des attributs dans les règles, c'est-à-dire ceux qui apparaissent dans la prémisse et ceux qui apparaissent dans la conclusion. Dans la suite de ce chapitre, nous assimilons la variable visuelle à l'attribut.

Les variables visuelles que nous utilisons sont la position, la couleur, l'alpha et la taille du point. Comme les dimensions dans le plan ont tous les niveaux d'organisation (associatif, sélectif, ordonné et quantitatif) (Cf. Figure 2.11), c'est en général le point de départ d'une visualisation. En effet, les points dans le plan sont d'abord exprimés. Pour utiliser d'autres dimensions, les autres variables visuelles sont alors exploitées, comme la couleur. Ce principe est applicable dans une visualisation en trois dimensions. En effet, les données sont d'abord exprimées dans l'espace, puis par d'autres variables visuelles en cas de nécessité.

Dans un espace bidimensionnel, il est naturel d'exprimer les ordonnées en fonction des abscisses. En trois dimensions, la représentation usuelle est la profondeur, ou la hauteur, en fonction du plan. En effet, la position est l'information qui est la mieux perçue [Card 99] dans une représentation visuelle. Nous considérons donc qu'il s'agit du point de départ dans l'élaboration de la règle d'association. En espace 2D, nous associons les attributs correspondant aux abscisses à la prémisse de la règle. L'ordonnée faisant partie de la position, elle bénéficie des mêmes caractéristiques que l'abscisse. Elle peut donc être en prémisse. Mais, comme elle est fonction de l'abscisse, elle peut être également dans la conclusion. Dans une représentation tridimensionnelle, la hauteur est fonction des coordonnées dans le plan. Donc elle peut se trouver aussi en prémisse en conclusion. Cependant, de même que nous avons considéré que l'abscisse est le point de départ de la vue bidimensionnelle, raison pour laquelle elle ne peut apparaître qu'en prémisse, dans une représentation tridimensionnelle, nous considérons l'abscisse et l'ordonnée comme le point de départ de la vue. Dans ce cas, l'ordonnée n'apparaît également que dans la prémisse. La dimension fait partie des variables de l'image, en ce sens qu'elle est utilisée pour l'élaborer. La taille du point peut donc se trouver, comme la position, dans la prémisse ou dans la conclusion de la règle.

La couleur ne dispose pas des mêmes niveaux d'organisation, car il s'agit d'une variable de séparation qui ne sert pas à élaborer l'image. Comparer les données en fonction de cette variable visuelle est moins facile sans les niveaux quantitatifs et ordonnés. C'est pourquoi, nous ne positionnons la couleur que dans la prémisse de la règle, en complément de l'abscisse, quand la visualisation est dans le plan, ou du plan, quand la visualisation est dans l'espace. De la même manière, la transparence est aussi difficile à discriminer, bien que ce soit une variable visuelle ordonnée. L'alpha va donc apparaître également uniquement dans la prémisse de la

règle.

Considérant les variables visuelles appartenant à l'ensemble $V_V = \{X, Y, Z, S, C, A\}$ (S est la taille, C la couleur et A l'alpha), la restriction sur les règles d'association, basées sur les attributs associés aux variables visuelles, grâce à la fonction *map*, peut maintenant être formalisée dans les équations (4.1) (présentation dans le plan) et (4.2) (présentation dans l'espace) en fonction la construction de la visualisation par l'utilisateur :

$$X \wedge \{Y, S, C, A\}^* \Rightarrow \{Y, S\}^+. \tag{4.1}$$

$$X \wedge Y \wedge \{Z, S, C, A\}^* \Rightarrow \{Z, S\}^+. \tag{4.2}$$

L'astérisque $*$ signifie qu'aucune, une ou plusieurs valeurs peuvent être utilisées. Le signe $+$ signifie qu'au moins une valeur doit être utilisée. Notons qu'une variable visuelle ne peut pas apparaître deux fois dans une règle.

4.6 Enrichissement de la visualisation par les résultats des algorithmes

Nous proposons, dans ce chapitre, d'exploiter les résultats des algorithmes en enrichissant la visualisation des données initiales, par assignation des variables visuelles. Une fois l'association réalisée, l'exploration des itemsets ou des règles est alors répercutée, par une mise à jour des mesures qui les caractérisent, vers l'espace des données.

4.6.1 De l'itemset fréquent à la visualisation

La mesure associée à l'itemset est son support (Cf. Chapitre 1.5.1), dont l'exploitation revient à valoriser les liens entre les attributs. Il peut ainsi être utilisé pour caractériser la visualisation par les itemsets. Pour cela, nous utilisons simultanément l'espace de visualisation des données, et un espace de visualisation des itemsets. La formalisation abordée jusqu'à présent portait sur l'exploration des données. Nous allons entreprendre ici l'exploration des itemsets fréquents, à l'aide d'un outil dédié. Cet outil, qui sera présenté plus loin, est utilisé pour explorer les résultats des algorithmes. Il dispose des fonctionnalités de l'explorateur des données, ainsi que d'autres adaptées à l'exploration des itemsets et des règles.

Définition 8 *(em) La contribution de la donnée dans les itemsets est définie par la variable em, de la manière suivante :*

- *Soit I un itemset, et x_i une donnée de l'ensemble des données $\mathcal{D}$;*
- *Si x_i satisfait l'itemset I, alors $em(x_i) = 1$;*
- *Sinon $em(x_i) = 0$.*

Dans cette définition, nous considérons que x_i satisfait l'itemset I, si les valeurs des attributs de x_i appartiennent aux classes d'attributs de I

Si plusieurs itemsets sont considérés, alors cette opération est réalisée pour chacun d'entre eux, et la variable *em* associée à chaque donnée est incrémentée en fonction de sa participation dans les itemsets. La contribution des données dans les itemsets est ensuite mise en valeur dans la visualisation initiale des données, par l'assignation de *em* à une variable visuelle. Si celle-ci est ordonnée, alors la quantité qu'elle représente indique la participation. Par exemple, en affectant *em* à la hauteur Z, et en représentant la vue en 3D, plus les données sont concernées par les itemsets, plus elles sont éloignées du plan XY, c'est-à-dire des autres données. Les zones de la visualisation, qui sont ainsi particularisées, correspondent aux données pour lesquelles de l'information peut être extraite par les algorithmes, car elles y sont impliquées.

4.6.2 De la règle à la visualisation

Nous reprenons le même concept que ci-dessus, mais, les règles étant caractérisées par plusieurs mesures, nous définissons plusieurs variables em_m qui quantifient l'implication de la donnée dans les règles sélectionnées. Ces variables portent sur la confiance et les mesures de qualité (Cf. Chapitre 1.5.3). Nous avons ainsi em_c pour la confiance, em_l pour le lift, em_{LO} pour Loevinger, etc. Si une seule règle est sélectionnée, alors em_m indique la mesure de cette règle. Cependant, dans le cas de la sélection de plusieurs règles, il peut être intéressant de considérer les valeurs minimales, maximales et moyennes du groupe de règles. Nous définissons ainsi, pour chaque variable em_m, trois variables em_{m_min}, em_{m_max} et em_{m_moy}. De plus, d'autres variables em_s caractérisent la structure de la règle, comme la présence de la donnée dans la prémisse avec em_{s_p}, dans la conclusion avec em_{s_c} ou dans les deux avec em_{s_pc}. Par exemple, si les attributs d'une donnée x_i satisfont la prémisse d'une règle, alors $em_{s\_p}(x_i) = 1$.

Les variables em_m, sont ensuite exploitées par leur assignation aux variables visuelles de la visualisation des données, comme cela a été le cas pour les itemsets, sachant qu'elles ne sont pas toutes visualisables simultanément, étant donnée leur quantité.

Chapitre 5

Génération de visualisations à partir d'un ensemble de règles d'association

Ce chapitre présente un processus, qui, à partir de règles d'association, permet de générer des visualisations, de deux manières différentes.

D'une part, il s'agit de générer automatiquement une visualisation à partir d'un ensemble de règles choisies par l'utilisateur. Ce choix est réalisable de différentes manières, que ce soit par exploration des règles dans un espace de visualisation, par requête dans une base de règles, ou par toute autre technique de sélection.

D'autre part, un ensemble de visualisations pertinentes sont générées automatiquement à partir de toutes les règles. A cette fin, ces dernières sont partitionnées dans plusieurs classes, en fonction d'un critère de distance que nous définissons, et chaque classe donne lieu ensuite à une visualisation de données.

Dans les deux cas, les 1-itemsets constituant les règles font l'objet d'un ordonnancement, qui va en induire un autre pour les attributs. A leur tour, ceux-ci vont ordonner les variables visuelles. Si les règles sont sélectionnées par exploration visuelle, alors ce processus peut être entrepris à la suite de celui qui a été décrit dans le chapitre précédent, permettant d'aller de la visualisation des données vers les règles. Cette séquence de processus peut alors constituer une boucle itérative.

Les règles d'association seront écrites de la forme $L_1, L_2 \cdots \Rightarrow R$, où les L_i sont les classes d'attributs apparaissant dans la prémisse, et R est une classe apparaissant dans la conclusion.

Les ensembles suivants seront introduits :

- L_l est l'ensemble des itemsets de la prémisse d'une règle ;
- $\mathcal{L}$ est l'ensemble des attributs de prémisses contenues dans toutes les règles ;
- $\mathcal{R}$ est l'espace des règles d'association.

5.1 Génération automatique d'une visualisation à partir d'un ensemble de règles choisies

Le chapitre 4 a présenté comment la visualisation et les opérateurs de sélection conditionnent la construction des règles d'association, en fonction de plusieurs paramètres. Afin de mettre en valeur les règles, nous allons, non plus enrichir la visualisation des données initiales à partir des mesures associées aux itemsets et aux règles d'association (Cf. Chapitre 4.6), mais caractériser les règles en configurant la visualisation des données, pour que ces dernières les représentent au mieux.

Pour cela, nous utilisons simultanément l'espace d'exploration des données et celui d'exploration des résultats algorithmiques. Nous considérons dans cette étude les règles de type *many-to-one*, à l'instar des études sur la représentation des règles d'association (Cf. Chapitre 3.2.2).

5.1.1 Calcul des 1-itemsets

Les 1-itemsets, constituant les règles, sont issus de l'algorithme k-means [MacQueen 67] (Cf. Chapitre 4.3). Il est possible d'obtenir les données en entrée de k-means de différentes manières. D'après le chapitre 4.2, la sélection des données dans l'espace de visualisation donne lieu à l'ensemble $\mathcal{D}'$. A partir de celui-ci, les valeurs des attributs associés aux variables visuelles ont été discrétisées, et ce sont les classes résultantes qui ont constitué les données en entrée de l'algorithme de fouille. Dans le cas présent, nous considérons à nouveau $\mathcal{D}'$, mais les classes résultantes portent sur tous les attributs de la base de transaction, ou un sous-ensemble choisi par l'utilisateur. Nous pouvons également considérer que toute la base de données ayant été sélectionnée, toutes les valeurs de tous les attributs choisis par l'utilisateur sont prises en compte. Si tous les attributs sont choisis, alors les classes prennent en compte la totalité de la base. Ce dernier cas montre que l'exploration visuelle des données peut ne pas être un préalable à ce qui suit dans cette partie. En effet, il peut être envisagé de générer des classes à partir d'une base de transactions, et de représenter sur une visualisation des données, le résultat d'un algorithme de fouille. Ainsi, quelle que soit la technique de choix des données et des attributs, nous considérons que nous sommes en présence des 1-itemsets L_i, associés, pour chacun d'eux, à un des attributs de l'ensemble $\mathcal{A} = \{A_1, A_2, ..., A_m\}$.

5.1.2 Sélection d'une règle

Nous considérons, dans un premier temps la sélection d'une règle. La généralisation à plusieurs règles sera ensuite exposée (Cf. Chapitre 5.1.4). Le point de départ de ce que nous appelons *processus retour*, de la règle vers la visualisation, est l'espace d'exploration des règles. Nous pourrions cependant utiliser une autre technique de sélection, éventuellement non graphique, comme un choix de règles dans une base de données à partir de critères définis par l'utilisateur. De même qu'il permet de sélectionner des itemsets fréquents (Cf. Chapitre

4.6.1), il permet de sélectionner également une ou plusieurs règles d'association. Les règles ainsi sélectionnées vont ensuite conditionner la visualisation des données initiales, afin de les caractériser en fonction de la sélection. Si celle-ci est modifiée, alors la visualisation des données est adaptée en conséquence. Afin de faciliter le pilotage de ce processus par l'utilisateur, nous considérons que l'attribut de la conclusion de la règle est contraint et choisi par celui-ci. Pour configurer la visualisation des données initiales, nous devons alors établir un ordonnancement des attributs associés aux 1-itemsets présents dans les prémisses des règles, qui sera ensuite répercuté sur les variables visuelles selon un ordonnancement s'appuyant sur la sémiologie de Bertin [Bertin 67] (Cf. Chapitre 2.2). La suite de ce chapitre aborde la manière dont nous ordonnons les attributs.

5.1.3 Ordonnancement des 1-itemsets d'une règle

Lorsque nous avons abordé la formalisation de la visualisation aux résultats des algorithmes, les règles d'association ont été restreintes en fonction des variables visuelles (Cf. Chapitre 4.5). Pour une visualisation en 2D, la variable Y a été associée préférentiellement à la conclusion. Il en de même pour Z dans une visualisation en 3D. Pour cela, dans notre étude, nous considérons que la conclusion est contrainte par l'utilisateur, en ce sens qu'il choisit l'attribut correspondant. Si ce choix est réalisé dans le cadre d'une exploration visuelle des règles, alors, lors de la sélection d'une ou plusieurs règles, l'attribut associé à Y ou Z, selon le type de visualisation choisie 2D ou 3D, constitue la conclusion. Dans la suite de ce chapitre, nous nous intéressons uniquement aux itemsets des prémisses.

L'objectif est d'établir l'ordre d'importance des 1-itemsets les uns par rapport aux autres. Grâce à l'algorithme k-means (Cf. Chapitre 4.3), les valeurs des attributs sont distribuées dans des groupes. Chacun constitue un 1-itemset qui est l'élément de base dans la construction des k-itemsets, à partir desquels sont calculées les règles (Cf. Chapitre 1.5.2). Pour ordonner les attributs, nous nous intéressons d'abord aux 1-itemsets, que nous appellerons simplement itemsets dans la suite de ce chapitre, et qu'il s'agit d'ordonner. Afin d'ordonner les itemsets, il est nécessaire de définir un critère qui permettra d'affirmer qu'un itemset est « meilleur » qu'un autre. Pour cela, nous étudions l'évolution du support et de la confiance quand les itemsets sont successivement extraits de la prémisse d'une règle. Par exemple, avec la règle $L_1, L_2, L_3 \Rightarrow R$, cela revient à considérer les règles $L_2, L_3 \Rightarrow R$, $L_1, L_3 \Rightarrow R$ et $L_1, L_2 \Rightarrow R$, qui sont obtenues à partir de sous-itemsets.

En ajoutant un itemset L_a à un k-itemset, le support diminue ou reste le même, étant donnée son antimonotonicité. Il en est de même si l'on ajoute L_b au lieu de L_a. Cependant, si l'ajout de L_a voit le support diminuer moins que l'ajout de L_b, cela signifie que le lien est plus fort entre L_a et le k-itemset, qu'entre L_b et celui-ci. Dans ce cas, nous considérons que L_a est meilleur que L_b vis-à-vis du k-itemset. En considérant que l'on retire les 1-itemsets d'un k-itemset au lieu de les ajouter, le meilleur est donc celui qui voit le support augmenter le moins possible.

La confiance, quant à elle, ne bénéficie pas de la propriété d'antimonotonicité, car l'ajout

d'un 1-itemset à un k-itemset peut provoquer une augmentation ou une diminution de celle-ci. Le cas le plus intéressant est celui de l'augmentation, car elle signifie que le 1-itemset améliore cette mesure. En considérant le processus inverse qui consiste à enlever un 1-itemset à un k-itemset, le cas intéressant est celui qui voit la confiance diminuer le plus. En effet, rajouter ce 1-itemset au $(k-1)$-itemset revient alors à augmenter la confiance le plus possible.

En combinant ces considérations sur le support et la confiance, un 1-itemset est donc meilleur si son retrait d'un k-itemset voit le support augmenter le moins possible et la confiance diminuer le plus possible.

Nous définissons ainsi la mesure de qualité d'un itemset dans une règle de la manière suivante :

Définition 9 *(QPL) La mesure de qualité $QPL(Rule_j, L_i)$ de l'itemset L_i dans la règle $Rule_j$ est obtenue de la manière suivante :*

- *$Rule_j - L_i$ définit la règle $Rule_j$ dont est extrait l'itemset L_i de la prémisse.*
- *La mesure QPL s'écrit :*

$$QPL(Rule_j, L_i) = -\frac{c(Rule_j - L_i) - c(Rule_j)}{s(Rule_j - L_i) - s(Rule_j) + 1}$$

où s et c sont le support et la confiance.

Elle quantifie la qualité de l'itemset L_i dans la règle $Rule_j$. Une bonne valeur doit être positive, sinon cela correspond à une augmentation de la confiance quand un itemset est enlevé de la prémisse. La valeur 1 ajoutée au dénominateur a pour but d'éviter que celui-ci soit nul. En effet, en enlevant L_i, le support peut ne pas augmenter, en restant identique. Grâce à son antimonotonicité, le dénominateur est alors supérieur ou égal à 1.

La mesure QPL permet ainsi d'ordonner les itemset de la prémisse. Afin d'obtenir une distance entre eux, nous normalisons la valeur de QPL, en la ramenant à un intervalle compris entre 0, pour l'itemset le moins bon, et 1 pour le meilleur.

Définition 10 *(QPLN) La valeur normalisée $QPLN$ est obtenue de la manière suivante :*

- *L_l est l'ensemble des itemsets de la prémisse de la règle $Rule_j$;*
- *Si $L_i \notin L_l$, alors :*

$$QPLN(Rule_j, L_i) = -1$$

- *Si $L_i \in L_l$ et $Card(L_l) = 1$, alors :*

$$QPLN(Rule_j, L_i) = 1$$

- *Si $L_i \in L_l$ et $Card(L_l) > 1$, alors :*

$$QPLN(Rule_j, L_i) = \frac{QPL(Rule_j, L_i) - Min(QPL(Rule_j, L_l))}{Max(QPL(Rule_j, L_l)) - Min(QPL(Rule_j, L_l))}$$

Ainsi, dans le cas où un 1-itemset ne fait pas partie de la prémisse, nous considérons que la valeur de $QPLN$ est égale à -1. Si la règle ne contient qu'un seul itemset dans la prémisse, alors elle est égale à 1. Si elle contient deux 1-itemsets, alors les valeurs de $QPLN$ sont égales à 0 et 1. Dans les autres cas, les valeurs des 1-itemsets intermédiaires appartiennent à l'intervalle $]0,1[$, en fonction de leur qualité.

Ainsi, $QPLN$ permet d'ordonner les 1-itemsets présents dans la prémisse d'une règle. Le paragraphe suivant en expose une généralisation, dans le cas de la sélection de plusieurs règles.

5.1.4 Ordonnancement des 1-itemsets d'un ensemble de règles

Lors de la sélection de plusieurs règles, la qualité d'un 1-itemset peut être bonne pour une règle, et moins bonne pour une autre. Pour obtenir une valeur de qualité globale d'un 1-itemset, nous pondérons les valeurs de $QPLN$ avec le support des règles, puis nous les additionnons.

Définition 11 *(QPLG) La qualité globale du 1-itemset L_i est obtenue de la manière suivante :*

- *k est le nombre de règles, dans lesquelles se trouve l'itemset L_i ;*
- *La qualité globale est :*

$$QPLG(L_i) = \sum_{j=1}^{k} s(Rule_j) QPLN(Rule_j, L_i)$$

Nous obtenons ainsi un ordonnancement des 1-itemsets.

5.1.5 Ordonnancement des attributs

D'après le chapitre 1.5.1, l'ensemble des m attributs est défini par $\mathcal{A} = \{A_1, A_2, ..., A_m\}$. Comme un attribut a été choisi par l'utilisateur et se trouve en conclusion des règles (Cf. Chapitre 5.1.2), celui-ci n'est donc pas pris en compte ici. Nous pouvons maintenant définir la qualité d'un attribut.

Définition 12 *(QPA) La qualité d'un attribut A_i est obtenue de la manière suivante :*

- *Nous notons n_{A_i} le nombre de clusters générés par k-means pour l'attribut A_i ;*
- *$\{L_{A_i^1}, L_{A_i^2}, \ldots, L_{A_i^{n_{A_i}}}\}$ est l'ensemble des clusters de cet attribut ;*
- *La qualité QPA de A_i est alors :*

$$QPA(A_i) = \sum_{j=1}^{j=n_{A_i}} QPLG(L_{A_i^j})$$

Ainsi, pour calculer la qualité d'un attribut, nous additionnons les $QPLG$ correspondant aux 1-itemsets associés à cet attribut. Cela donne, pour chaque attribut, une valeur qui permet de les classer, le meilleur étant celui qui a la plus forte valeur de QPA.

5.1.6 Génération de la visualisation

La visualisation vis a été définie à partir des fonctions map et f (Cf. Chapitre 4.1). Etant donné l'ordonnancement des attributs, il est possible de la déduire, ainsi que la scène sc.

Assignation des variables visuelles (map)

L'assignation des variables visuelles est ordonnée en fonction de leur niveau d'organisation (Cf. Figure 2.11), en reprenant l'ordonnancement des attributs de la prémisse. Selon la vue 2D ou 3D, l'attribut présent en conclusion de toutes les règles, qui a été choisi par l'utilisateur, est assigné respectivement à la variable visuelle Y ou Z. Les variables visuelles les plus importantes sont les variables de l'image, c'est-à-dire les dimensions du plan, la taille et la valeur. L'utilisateur peut être intéressé par une représentation du résultat en 2D ou en 3D. En fonction de son choix, les variables les plus importantes sont donc X (visualisation 2D), ou X et Y (visualisation 3D).

Pour la variable suivante, nous distinguons le cas d'un attribut numérique d'un attribut nominal. S'il est numérique, alors la taille des points sera préférée à un gradient de couleur, parce que ses variations sont mieux perçues et qu'il s'agit d'une variable quantitative. Les variables visuelles seront alors S puis C. S'il est nominal, alors la couleur discrète sera préférée à la taille des points, parce qu'elle permettra de mieux discriminer ceux-ci, étant donné qu'elle est associative. Les variables visuelles seront alors C puis S.

En dernier, nous utilisons l'alpha A, bien que ses variations soient souvent difficiles à percevoir (Cf. Chapitre 2.2).

En résumé, les variables visuelles sont ordonnées de la manière suivante, selon une importance décroissante :

- X, S, C puis A (visualisation 2D et deuxième attribut numérique) ;
- X, C, S puis A (visualisation 2D et deuxième attribut nominal) ;
- X, Y, S, C puis A (visualisation 3D et troisième attribut numérique) ;
- X, Y, C, S puis A (visualisation 3D et troisième attribut nominal).

Assignation de la scène (sc)

Les variables visuelles étant assignées, le processus retour détermine ensuite la scène sc, c'est-à-dire l'ensemble des points de l'espace $\mathcal{P}$ qui seront affichés (Cf. Chapitre 4.1). Pour cela, il est nécessaire de considérer les plages de valeurs contenues dans chaque classe d'attributs qui avait été calculée par l'algorithme k-means. La visualisation est alors adaptée pour ne présenter, dans la mesure du possible, que ces valeurs, notamment par un réglage automatique du zoom et de l'excentrement. Les points de l'espace des données $\mathcal{D}$, qui sont associées au groupe de règles, sont ainsi calculés, d'une part, à partir des itemsets fréquents qui ont servi à élaborer ces règles, et, d'autre part, à partir de la confiance minimale qui a été fixée par l'utilisateur, par exemple lors de l'exploration des données.

Nous définissons ainsi une fonction permettant d'obtenir une scène à partir d'une règle.

Définition 13 *(Rule2sc) La fonction Rule2sc permettant d'obtenir une scène à partir d'une règle $Rule_j$ est définie par :*

$$Rule2sc(Rule_j) = sc_{Rule_j}$$

où $Rule_j \in \mathcal{R}$ et $sc_{Rule_j} \in \mathcal{P}$.

En considérant toutes les règles contenues dans un groupe, alors la scène résultante est la réunion des scènes ainsi calculées.

Assignation des gradients (f)

Elle est réalisée par la fonction f, définie dans le chapitre 4.1 à la définition 2, qui fait le lien entre les valeurs d'attributs et les valeurs des variables visuelles.

5.2 Génération automatique de visualisations à partir de l'ensemble des règles

Dans le paragraphe précédent, nous avons montré comment est paramétrée la visualisation à partir d'un ensemble de règles choisies par l'utilisateur. Si cela est réalisé dans le cadre d'une exploration visuelle des règles, il répond à des critères propres à l'utilisateur. Ils peuvent s'appuyer, par exemple, sur la considération d'une ou plusieurs mesures associées aux règles, ou sur l'étude d'une zone de l'espace de visualisation des règles. Afin d'assister l'utilisateur dans la sélection des règles, nous les regroupons en fonction d'un critère de distance inter-règles que nous allons définir. A partir de celui-ci, les règles seront regroupées en groupes ou clusters, et ce sont ces deniers qui conditionneront la configuration de la visualisation des données initiales.

Nous considérons toutes les règles de la base, avec toujours la contrainte de la conclusion, qui fait partie des réglages de l'utilisateur. Par ailleurs, l'exploration visuelle des données n'est pas nécessaire pour obtenir les règles qui ont pu être obtenues en exécutant simplement un algorithme.

5.2.1 Définition de la distance entre deux règles

Nous appelons $\mathcal{L}$ l'ensemble des attributs de prémisses contenues dans toutes les règles :

$$\mathcal{L} = \{L_{A_1^1}, L_{A_1^2}, \ldots, L_{A_1^{n_{A_1}}}, L_{A_2^1}, L_{A_2^2}, \ldots, L_{A_2^{n_{A_2}}}, \ldots, L_{A_m^1}, L_{A_m^2}, \ldots, L_{A_m^{n_{A_m}}}\}$$

Chaque élément de cet ensemble définit une composante dans l'espace $\mathcal{L}^{Card(\mathcal{L})}$. Une règle peut être représentée dans cet espace, en fonction des $QPLN$ des 1-itemsets constituant sa prémisse. Pour cela, nous définissons le point Pr.

Définition 14 *(Pr) Un point de l'espace $\mathcal{L}^{Card(\mathcal{L})}$ est défini par :*

$$Pr(Rule_j) =< QPLN(Rule_j, L_{A_1^1}), QPLN(Rule_j, L_{A_1^2}), \ldots, QPLN(Rule_j, L_{A_m^{n_{A_m}}}) >$$

A partir des points Pr, il est alors possible d'en définir une distance $distR$ qui correspond à une distance inter-règles.

Définition 15 *(distR) La distance entre deux règles $Rule_j$ et $Rule_k$ est définie de la manière suivante :*

$$distR(Rule_j, Rule_k) = ||\overrightarrow{Pr(Rule_j)Pr(Rule_k)}||$$

Ainsi, à une règle $Rule_j$ est associé un point $Pr(Rule_j)$ dans l'espace $\mathcal{L}^{Card(\mathcal{L})}$. Dans cet espace, les composantes sont les valeurs de qualités $QPLN$. Ces valeurs sont comprises entre 0 et 1, et sont égales à -1 si la règle ne contient pas le 1-itemset considéré. En faisant l'hypothèse que le repère est orthonormé, la distance entre deux points est la norme du vecteur défini par ces points. Nous appelons alors cette norme la distance entre les deux règles associées à ces deux points.

Afin de montrer la pertinence de cette distance, nous allons considérer les exemples de règles suivants :

- $Rule_1 = (0.3, -1, 0.4, -1)$;
- $Rule_2 = (-1, -1, 0.2, -1)$;
- $Rule_3 = (-1, 0.1, -1, 0.7)$.

Les distances sont :

- $distR(Rule_1, Rule2) = 1.32$;
- $distR(Rule_1, Rule3) = 2.78$;
- $distR(Rule_2, Rule3) = 2.35$.

La première distance est sensiblement plus petite que les deux autres. En effet, $Rule_1$ et $Rule_2$ partagent le troisième 1-itemset et ne sont pas concernées par les deuxième et quatrième 1-itemset. $Rule_1$ et $Rule_3$ n'ont aucun 1-itemset en commun. La composante -1 signifiant que le 1-itemset correspondant ne fait pas partie de la règle, cela augmente la distance, ce qui est logique, car les règles n'ont pas de point commun pour cet itemset. Ainsi, $Rule_1$ et $Rule_3$ n'ayant pas de 1-itemset en commun, leur distance est élevée comparée à la précédente. $Rule_2$ et $Rule_3$ ne sont pas concernés par le premier 1-itemset, et ne partagent pas simultanément les autres itemsets. Leur distance est donc également élevée. Cet exemple montre que pour que la distance soit petite, il est nécessaire que les règles partagent les mêmes 1-itemsets. Cette distance sera ensuite d'autant plus petite que la différence entre les composantes communes est petite.

5.2.2 Regroupement des règles d'association en clusters

A partir de l'expression des règles dans l'espace des 1-itemsets, nous sommes donc en mesure de calculer les distances entre elles, ce qui va permettre de dégager des groupes

de règles en fonction de celles-ci. Pour cela, nous utilisons à nouveau l'algorithme k-means [MacQueen 67] (Cf. Chapitre 4.3). Nous considérons que l'utilisateur peut choisir un nombre fixe de clusters, ou un nombre maximum que le système optimisera en fonction du critère de Caliński-Harabasz [Caliński 74].

5.2.3 Paramétrage de la visualisation des données à partir des clusters de règles

Pour un cluster choisi par l'utilisateur, nous reprenons la méthode exposée dans le chapitre 5.1 pour ordonner les attributs des prémisses des règles qu'il contient, assigner les variables visuelles, la scène et les gradients. En choisissant un autre cluster de règles, la configuration de la visualisation est alors modifiée en conséquence.

Chapitre 6

Conclusion sur les liens entre la fouille de données automatique et la fouille visuelle de données

Cette partie a présenté les liens entre la fouille visuelle des données et la fouille automatique. Pour cela, nous avons abordé, dans un premier temps, le processus allant des données visualisées vers les résultats algorithmiques. Le processus inverse, des résultats vers les données, a ensuite fait l'objet du second chapitre.

Nous avons tout d'abord présenté une formalisation de la visualisation, qui permet de préciser la notion de filtre du modèle de Card & Mackinlay (Cf. Chapitre 2.3). Différents opérateurs sont mis en œuvre pour passer de l'espace des données initiales aux données sélectionnées. Cela passe par le filtrage, le zoom et l'excentrement, et enfin par la sélection. Ainsi, le lien entre les attributs de données et les variables visuelles a été formalisé par les fonctions map, f et vis. Puis, la scène sc permet de connaître les données visualisées. A partir de celle-ci, la sélection finalise la détermination des données qui sont effectivement exploitées par l'algorithme de Data Mining. Les opérateurs de sélection que nous utilisons prennent en compte l'hétérogénéité des données aéronautiques et leurs types, numériques, entiers ou flottants, et alphanumériques. La sélection est réalisée à la souris par des opérateurs de type *brushing* (Cf. Chapitre 3.1.3), et par des requêtes pour chaque attribut alphanumérique.

Les données traitées par l'algorithme de Data Mining reflètent la visualisation élaborée par l'utilisateur. Pour cela, les niveaux d'organisation des variables visuelles, issus de la sémiologie graphique de Bertin (Cf. Chapitre 2.2), conditionnent la structure des règles d'association extraites, en termes de prémisses et de conclusions. Les données doivent cependant faire l'objet de regroupements en étant traitées par l'algorithme de partitionnement k-means. L'exploration des résultats algorithmiques est ensuite mise à profit pour enrichir la visualisation des données initiales, par assignation de mesures aux variables visuelles, ce qui permet de les valoriser.

Dans un second temps, nous avons présenté une méthode pour générer automatiquement

des visualisations à partir des règles. Soit celles-ci sont choisies par l'utilisateur, soit elles sont considérées dans leur globalité, et, dans ce cas, elles sont regroupées en classes par l'algorithme k-means, en s'appuyant sur un critère de distance inter-règles que nous avons proposé. Chaque groupe de règles donne lieu à la génération automatique d'une visualisation, en ordonnant d'abord les 1-itemsets qui les constituent, puis les attributs des données. Cet ordre conditionne ensuite l'assignation des attributs aux variables visuelles, en s'appuyant sur la sémiologie graphique.

Les processus aller et retour, entre les données et les règles d'association, que nous décrivons dans cette formalisation, sont indépendants mais peuvent être chaînés. Dans ce cas, chacun peut être exécuté à la suite de l'autre, dans une boucle globale itérative. L'exploration des données donne lieu au lancement d'un algorithme, ce qui va alimenter l'exploration des itemsets et des règles d'association. Cela provoque un retour vers le visualisateur des données. A partir du changement de configuration de la vue initiale, l'utilisateur peut apporter des modifications, par des filtrages ou des choix d'assignations et de sélection, pour ensuite soumettre à nouveau les données à un algorithme qui peut être différent du premier (Cf. Chapitre 1.5). Les valeurs seuil de l'algorithme peuvent également être adaptées, par modification du support et de la confiance minimum. Il oriente et pilote ainsi le processus d'exploration et de fouille des données par itérations successives et raffinement de l'étude, en jouant un rôle central et décisionnel.

Ainsi, cette formalisation s'inscrit sous l'angle du Visual Analytics, selon lequel l'utilisateur reste le décideur dans le processus exploratoire, en adaptant la visualisation des données à ses besoins, ainsi que celle des résultats algorithmiques. De plus, il décide de la mise en œuvre et des conditions d'exécution des algorithmes de Data Mining. En effet, selon la question posée dans la recherche de connaissance, les algorithmes sont plus ou moins adaptés, et c'est en fonction de son expertise, qu'il optera pour un algorithme plutôt qu'un autre. Enfin, ce choix peut être modifié d'une itération à l'autre, si cela s'avère plus adapté aux besoins de l'utilisateur.

Dans la partie suivante du document, nous allons maintenant nous intéresser aux outils que nous avons élaborés, qui permettent d'explorer les résultats algorithmiques, et qui assurent le retour vers la visualisation des données initiales.

Cinquième partie

Visualisation des résultats de la fouille de données

Introduction

La partie précédente a permis d'établir un processus allant des données initiales jusqu'au résultat des algorithmes, en passant par leur exploration visuelle. De plus, il est itératif, ce qui permet à l'utilisateur d'explorer successivement les données et les résultats, puis de poursuivre, à nouveau à partir des données, pour affiner l'étude ou l'orienter différemment, ou alors recommencer à partir d'une nouvelle visualisation. Pour cela, il est nécessaire de pouvoir explorer les données, les itemsets et les règles d'association. L'exploration des données que nous avons entreprise reprend globalement le concept de l'outil FromDaDy [Hurter 09] (Cf. Chapitre 3.1.4), sachant que des fonctionnalités ont été ajoutées. Cet outil sera présenté dans le chapitre 12.3.

Nous présentons, dans cette partie, la visualisation et l'exploration du résultat des algorithmes. D'une part, la visualisation des itemsets fréquents (Cf. Chapitre 1.5.1) est réalisée sous une forme innovante basée sur un graphe circulaire concentrique, dans lequel les nœuds correspondent aux itemsets, et les arêtes entre ceux-ci correspondent aux liens entre les itemsets, plus précisément entre les supersets et les sous-sets (Cf. Chapitre 7). Ce graphe a fait l'objet d'une optimisation algorithmique du placement des nœuds, dont l'intérêt a été validé dans le cadre d'une expérimentation (Cf. Chapitre 8). Grâce à cette optimisation, un processus d'amélioration de son exploration, selon plusieurs techniques empruntées à la sémiologie graphique (Cf. Chapitre 2.2) et à l'InfoVis (Cf. Chapitre 3.1), permet de repérer l'implication des données dans la base, ainsi que la qualité des liens entre elles (Cf. Chapitre 9). D'autre part, la visualisation et l'exploration des règles est réalisée par l'association des mesures de qualités et des variables visuelles (Cf. Chapitre 10). D'après la partie 5, l'exploration de cet espace d'itemsets et de règles est ensuite exploitée par un retour vers la visualisation et l'exploration des données initiales.

La représentation circulaire des itemsets, l'amélioration de sa visualisation par l'assignation de résultats algorithmiques aux variables visuelles, et la mise en œuvre d'opérateurs de sélection, ont fait l'objet d'une publication à la conférence ISVC 2013 [Bothorel 13c].

Chapitre 7

Représentation circulaire des itemsets

7.1 Introduction

Un des problèmes récurrents des visualiseurs d'itemsets fréquents, sous forme de graphes, est la diminution de la lisibilité au fur et à mesure que le nombre de nœuds et d'arêtes augmente [Vijender Singh 11] (Cf. Chapitre 3.2.1). Dans une représentation linéaire, pour montrer un graphe dans son intégralité, il doit être étendu horizontalement. Une solution est de dupliquer les nœuds afin d'étirer l'espace de représentation, comme avec les coordonnées parallèles (Cf. Chapitre 3.1.1). Cependant cela pose rapidement un problème d'encombrement et de vision globale du graphe, car chaque nœud apparaît ainsi plusieurs fois. L'implication du nœud dans le graphe en fonction du nombre de connexions est immédiate quand il est unique, car elle ressort visuellement par le nombre de liens. Mais si celui-ci augmente, alors le graphe devient de plus en plus encombré. Pour permettre de garder l'unicité du nœud et pour avoir une élongation horizontale et verticale de l'espace de visualisation, nous proposons alors une représentation du graphe, sous la forme de cercles concentriques.

Ce chapitre présente cette structure circulaire, ainsi que son optimisation qui est double, tout d'abord en ajustant la taille des cercles, puis la position des nœuds.

7.2 Structure du graphe

La représentation des itemsets fréquents est basée sur des cercles concentriques. Les nœuds, représentant les itemsets, sont disposés sur des cercles qui contiennent chacun tous les itemsets d'un ordre donné. Les 1-itemsets sont sur le cercle de plus grand diamètre, les 2-itemsets sur le cercle inférieur suivant, et ainsi de suite jusqu'aux itemsets de plus grand ordre qui sont disposés sur le cercle de plus petit diamètre. Hormis les 1-itemsets qui correspondent aux classes des attributs des données, un itemset est une combinaison d'une paire de sous-itemsets,

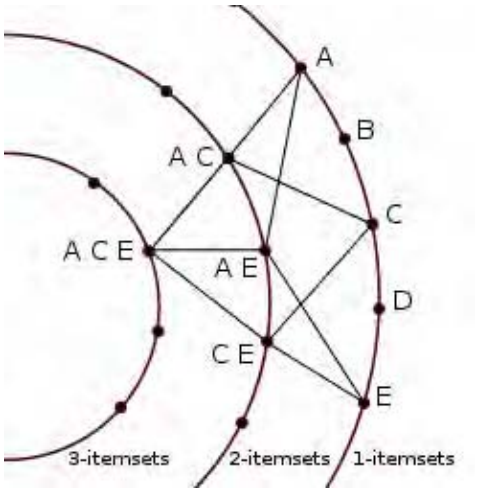


FIGURE 7.1 – Principe de construction du graphe circulaire des itemsets fréquents. $\{A,C,E\}$ est construit à partir de $\{A,C\}$ et de $\{A,E\}$, qui sont construits respectivement à partir de $\{A\}$ et $\{C\}$ et à partir de $\{A\}$ et $\{E\}$. $\{A,C,E\}$ est également construit à partir de $\{A,E\}$ et $\{C,E\}$.

sachant qu'il peut y avoir plusieurs paires. Ainsi, sur la figure 7.1, $\{A,C,E\}$ est construit à partir de $\{A,C\}$ et de $\{A,E\}$, mais également à partir de $\{A,E\}$ et $\{C,E\}$. Cette construction est représentée graphiquement par des segments reliant les itemsets. Ainsi, le segment reliant les itemsets $\{A,C\}$ et $\{A,C,E\}$ signifie que, comme le second itemset a un plus grand ordre que le premier, il est construit à partir de ce dernier. $\{A,C\}$ et $\{A,C,E\}$ sont donc respectivement sous-set et superset l'un de l'autre. La figure 7.1 montre, par les segments du graphe, la manière dont le 3-itemset est progressivement construit à partir des 1-itemsets. Comme un n-itemset est élaboré à partir de $(n-1)$-itemsets, les segments relient donc uniquement des cercles consécutifs. Ils relient le plus grand cercle des 1-itemsets au cercle des 2-itemsets, puis de ce cercle au cercle des 3-itemsets, et ainsi de suite progressivement jusqu'au plus petit cercle.

Les cercles concentriques sont, dans un premier temps, équidistants de façon à remplir la vue. De même, sur chaque cercle, les nœuds sont équidistants et sont répartis de manière homogène. La figure 7.2 montre un exemple de graphe circulaire contenant 15 1-itemsets, 41 2-itemsets, 39 3-itemsets et 11 4-itemsets.

Etant donnée la manière dont un itemset est construit à partir de ses sous-sets, l'effet combinatoire devrait voir le nombre de k-itemsets augmenter au fur et à mesure que la valeur de k augmente, et ainsi tendre vers un nombre très important d'itemsets sur le cercle le plus petit, qui deviendrait alors fortement surchargé. Cependant, la contrainte du support limite cet effet. En effet, seuls les itemsets fréquents apparaissent sur le graphe, et non la totalité, c'est-à-dire uniquement ceux dont le support est supérieur au support seuil. Cette contrainte fait que le nombre de k-itemsets fréquents commence par croître avec la valeur de k, puis il décroît rapidement quand la bordure [Bayardo 99] est atteinte (Cf. Chapitre 1.5.1). La figure 7.2 illustre bien ce point.

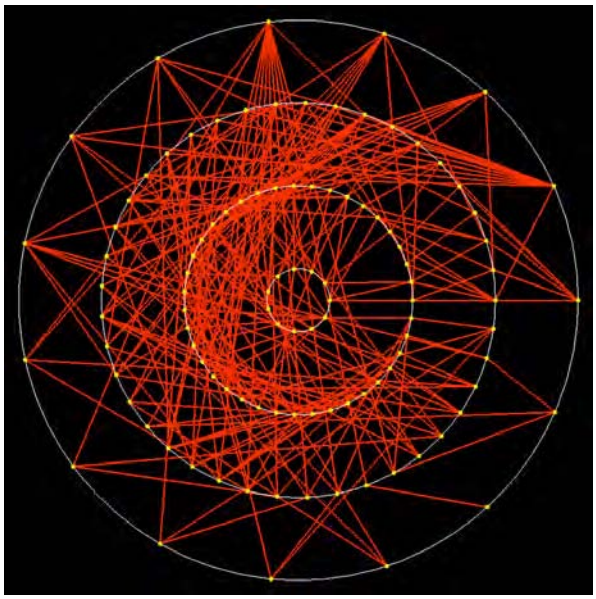


FIGURE 7.2 – Exemple de graphe circulaire représentant les itemsets fréquents et leurs liens.

D'après le chapitre 4.2, ce graphe n'est pas issu d'un algorithme de fouille de données appliqué à l'ensemble de la base de données, mais à un sous-ensemble obtenu par exploration visuelle. Cependant, en fonction de la sélection, du nombre d'attributs et de la valeur minimale du support, le nombre de nœuds et de connexions peut devenir rédhibitoire pour la lisibilité et l'exploration. Le problème classique d'encombrement d'un graphe est donc à nouveau posé. C'est pourquoi, dans la suite de ce chapitre, nous exposons une méthode d'optimisation pour réduire cet encombrement.

7.3 Optimisation du graphe

Le graphe circulaire est optimisé par la combinaison de deux approches :

- Optimisation du diamètre des cercles ;
- Optimisation de la position des itemsets.

En gardant une distance identique entre chaque paire de cercles consécutifs, les itemsets, répartis uniformément, peuvent se retrouver trop rapprochés, étant donnés leur nombre et le diamètre du cercle. Les segments associés sont alors également très proches. A cause de cela, nous fixons une distance minimale entre les itemsets d'un même cercle. A partir ce cette contrainte, le diamètre de chaque cercle est alors ajusté de façon à garantir cette distance minimale. Après avoir calculé le diamètre des cercles, des algorithmes sont mis en œuvre pour rapprocher les itemsets qui partagent les mêmes informations et les éloigner si ce n'est pas les cas.

Les données utilisées dans ce chapitre proviennent du *Mushroom Data Set*, issu du dépôt *Machine Learning Repository* de l'Université d'Irvine [1]. Cette base de données contient la description d'exemples hypothétiques de champignons correspondant à 23 espèces à lames. Cette base contient 8416 tuples de 23 attributs discrets concernant le caractère comestible, la forme du chapeau, l'odeur, l'anneau, etc.

7.4 Dimensionnement optimisé des cercles

Une distance minimale entre les itemsets d'un même segment est préalablement fixée, ainsi que le diamètre minimal du plus petit cercle. Ces contraintes sont établies par l'utilisateur, afin d'éviter des zones de congestion due à des nœuds, et donc à des segments, trop rapprochés. L'algorithme 1 calcule le diamètre des cercles en partant du plus petit, afin de respecter le diamètre minimal, et en allant progressivement jusqu'au plus grand. Pour chaque cercle, le diamètre est calculé en prenant la valeur maximale entre :

- le diamètre original, établi en gardant une distance constante entre deux cercles consécutifs ;
- le diamètre nécessaire pour pouvoir répartir les itemsets en respectant la distance minimale entre eux ;
- le diamètre nécessaire pour répartir les cercles sur la surface d'affichage de la manière la plus homogène possible.

Data: N=1 (number of the smallest circle)
Data: D_{min} and D_{max} (minimal and maximal diameters)
Data: d_{min} (minimum distance between 2 itemsets)
while $N < N_{max}$ **do**
  calculate $V1 = \frac{NbItems(N) \times d_{min}}{\pi}$;
  if $N = 1$ **then**
    calculate $D(N) = max(V1, d_{min})$;
  else
    calculate $V2 = D(N-1) + \frac{D_{max} - D(N-1)}{N_{max} - N + 1}$;
    calculate $D(N) = max(V1, V2, d_{min})$;
  end
  increase N ;
end
$D(N_{max}) = D_{max}$;

Algorithme 1: Calcul du diamètre des cercles.

Une fois le diamètre d'un cercle calculé, l'algorithme calcule le diamètre des cercles plus grands de façon à les répartir uniformément, puis il considère le cercle suivant. L'itération suivante consiste alors, soit à garder le diamètre du cercle qui vient d'être calculé, soit à

1. http ://archive.ics.uci.edu/ml

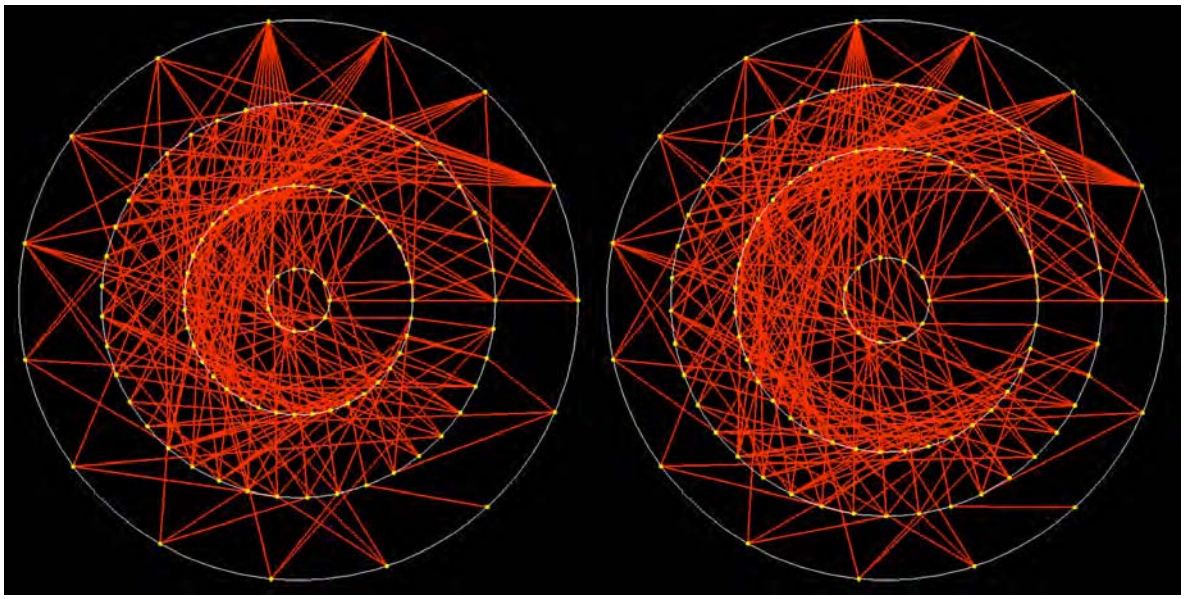


FIGURE 7.3 – Optimisation de la distance entre les cercles.
En augmentant la distance minimale entre les itemsets, le graphe de gauche est transformé pour obtenir le graphe de droite en augmentant le diamètre des cercles, afin de respecter cette distance minimale.

l'agrandir pour respecter la distance minimale entre les itemsets. Si la distance minimale entre les itemsets est augmentée, alors la distance entre eux doit être adaptée ainsi que le diamètre de chaque cercle. Cette optimisation est illustrée sur la figure 7.3.

7.5 Positionnement des itemsets

7.5.1 Définition de la distance

La vue globale de la structure du graphe est donnée par la position des nœuds et des segments qui les relient. Améliorer sa lisibilité revient à faire ressortir les nœuds qui sont reliés entre eux.

Un nœud étant un itemset, il représente une information issue d'un attribut ou d'une classe d'attributs. Rapprocher des nœuds signifie donc regrouper des informations. Pour cela, nous gardons la position discrète des nœuds et le maintien de la distance minimale entre deux nœuds. Si ces informations sont liées, alors le rapprochement des nœuds correspond à un partage d'information. Cette notion de lien entre les informations correspond à des liens entre les itemsets. Le lien est établi, par exemple, si un itemset est superset ou sous-set d'un autre, si deux itemsets contribuent à construire un superset commun ou sont supersets commun d'un même sous-set. Il est donc intéressant de regrouper les nœuds qui sont liés et de les éloigner s'ils ne le sont pas. Cela va ainsi créer des clusters d'itemsets, et donc d'information. Considérant que les clusters sont sur des cercles, il existe deux manières de calculer la distance

entre eux, dans le sens des aiguilles d'une montre ou dans le sens inverse. De plus, comme les itemsets sont liés entre deux cercles consécutifs, la distance entre les clusters de différents cercles doit être également considérée.

C'est pourquoi, l'optimisation globale du graphe, qui est obtenu en optimisant la position des itemsets, et donc des clusters, est le résultat de la combinaison de l'optimisation de la position des itemsets entre eux sur chaque cercle, et de l'optimisation de la position relative des clusters disposés sur les cercles. Nous considérons donc une distance appelée d que nous allons minimiser. Elle est la somme de la distance intra-cercle *icd* et de la distance inter-cercle *ecd*, que nous allons définir ci-dessous.

Distance intra-cercles

La distance intra-cercle *icd* est obtenue par l'algorithme LinLog [Noack 03], qui est un modèle d'énergie. Bien qu'il ait été conçu pour regrouper des nœuds de graphes généraux en clusters, nous l'appliquons avec la contrainte du positionnement des nœuds sur des cercles concentriques.

Définition 16 *($d(u,v)$) La distance entre deux nœuds d'un même cercle est définie de la manière suivante :*

- *Nous appelons V_i l'ensemble des nœuds sur le cercle C_i. V est l'union des ensembles de nœuds V_i :*
$$V = \bigcup_{i=1}^{N} V_i$$
- *E est l'ensemble des segments reliant les nœuds de cercles consécutifs, défini par :*
$$E = \bigcup_{i=1}^{N-1} E_i$$
Où $E_i = \{(u_i, w_{i+1}) \in V_i \times V_{i+1}, (u_i, w_{i+1}) \in E\}$.
- *La distance entre deux nœuds u et v d'un même cercle est définie par :*
$$d(u,v) = \|p_v - p_u\|$$

A partir de cette distance, il est possible de calculer l'énergie LinLog $U_{LinLog}(p)$ [Noack 03] d'un graphe p, définie par l'équation suivante :

$$U_{LinLog}(p) = \sum_{\{(u,w),(v,w)\} \in E_i^2} \|p_u - p_v\| - \sum_{\{u,v\} \in V_i^2} ln\|p_u - p_v\| \tag{7.1}$$

Le premier terme correspond à l'attraction entre les nœuds adjacents, et le second à la répulsion entre les nœuds.

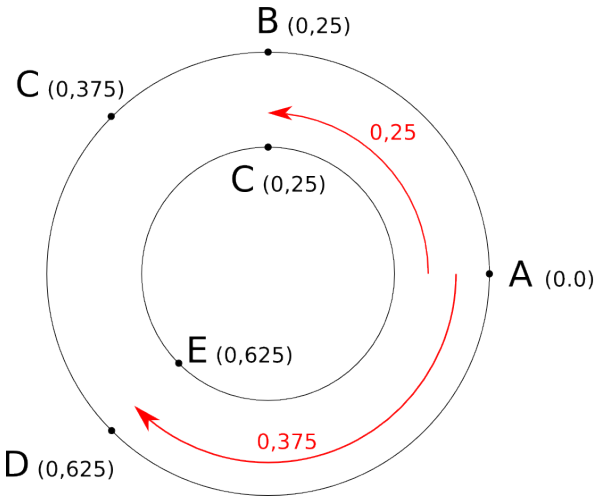


FIGURE 7.4 – Position angulaire et distance des itemsets.
La position angulaire est comprise en entre 0 et 1. La distance angulaire est la valeur absolue de la plus petite différence des positions angulaires entre deux itemsets. Sa valeur maximale est donc égale à 0,5. Ainsi, les distances (AB) et (AD) sont respectivement de 0,25 et 0,375. Les distances (BC) et (DE) sont égales à 0.

Cette énergie U_{LinLog} est ensuite calculée sur chaque cercle C_i, pour obtenir la distance résultante intra-cercles *icd*.

Définition 17 *(icd) La distance intra-cercles est définie par :*

$$icd = \sum_{i=1}^{N} U_{LinLog}(C_i)$$

Où C_i est le cercle de rang i.

Distance extra-cercles

ecd est la distance extra-cercles. Elle considère la différence angulaire entre les itemsets des différents cercles.

Définition 18 *($da(u,w)$) La distance angulaire entre deux nœuds d'un même cercle est définie de la manière suivante :*

- *Nous appelons ap la position angulaire d'un nœud. La position nulle est celle d'un nœud situé sur l'axe des abscisses positives d'un repère dont l'origine est le centre du cercle. Sa valeur, comprise entre 0 et 1, correspond à la fraction du cercle, quand le nœud se déplace sur celui-ci.*

- *La distance angulaire entre deux nœuds u et w est définie par :*

$$da(u,w) = min(|ap(u) - ap(w)|, 1 - |ap(u) - ap(w)|)$$

La position angulaire d'un point est illustrée par la ligure 7.4. Par exemple, si l'angle est $\frac{\pi}{2}$, alors elle est égale à 0.25 (Points B et C).

Comme les nœuds sont sur des cercles, la distance maximale est 0,5, selon que l'on considère le déplacement en tournant dans le sens des aiguilles d'une montre ou dans le sens inverse. Ainsi, la distance entre les points A et D est la valeur minimale entre $|0-0,625|$ et $1-(0-0,625)$, c'est-à-dire 0,375.

Définition 19 *(ecd) La distance extra-cercles est définie de la manière suivante :*

- *Considérant un cercle C_i, nous appelons ad_i la distance angulaire du cercle :*

$$ad_i = \sum_{\{(u,w)\} \in E_i} da(u,w)$$

- *La distance extra-cercle est alors définie par :*

$$ecd = \sum_{i=1}^{i=N-1} ad_i$$

Elle calcule la somme des différences de position angulaire entre chaque paire d'itemsets qui sont liés, entre chaque cercle C_i et le suivant.

Distance globale

A partir des distances intra-cercles et extra-cercles, la distance résultante peut être définie.

Définition 20 *(d) La distance globale du graphe est définie par :*

$$d = icd + ecd$$

Pour que cette distance d soit optimale, elle doit être minimale, parce qu'elle indique alors une distance minimale entre les nœuds d'un même cluster, et maximale entre les nœuds de clusters différents. C'est ce que nous allons aborder dans la suite de ce chapitre.

7.5.2 Optimisation du positionnement des itemsets par le recuit simulé

Pour trouver un minimum global à la distance d, nous utilisons l'algorithme du recuit simulé [Kirkpatrick 83, Cerny 85], inspiré du recuit utilisé dans l'industrie métallurgique. Il a pour objectif d'obtenir une structure cristalline ayant un niveau d'énergie minimal, en

contrôlant le réchauffement et le refroidissement du métal. Une analogie est faite avec cette technique pour obtenir un extremum global dans un système.

Nous considérons la fonction objectif f que nous devons minimiser. Par des recherches successives, les changements qui font décroître f sont acceptés, et quelques changements qui la font croître sont acceptés également, en fonction d'une probabilité p. Cette probabilité est définie par $p = exp(\frac{\delta f}{T})$, où δf est l'accroissement de f, et T est considéré comme la température qui agit en tant que paramètre de contrôle. Dans notre étude, la fonction f est la distance d qui doit être minimisée. Le paramétrage de l'algorithme est réalisé en fixant la valeur initiale de la température T et la manière dont elle décroît.

Cette méthode a été appliquée, à l'aide d'un processeur Xeon W5580, aux deux graphes de gauche de la figure 7.5, de supports respectifs 0,52 et 0,46. Le tableau 7.1 indique les résultats du calcul. Le recuit simulé est comparé avec *2-opt* [Croes 58] qui est un simple algorithme de type Hill Climbing pour trouver un minimum local. Nous pouvons constater que si nous calculons la somme de la longueur des segments, alors celle-ci diminue également. C'est la conséquence de l'optimisation du graphe, qui le simplifie en rapprochant les itemsets liés de manière pertinente, et donc en diminuant la longueur des segments.

	Somme de la longueur des segments	LinLog	Angles	Durée
Initial	137	-6587	84	
2-opt	90	-6902	39	15 s
Recuit simulé	82	-7000	33	156 s

(a)

	Somme de la longueur des segments	LinLog	Angles	Durée
Initial	378	-45430	196	
2-opt	232	-47295	101	75 s
Recuit simulé	212	-48025	86	772 s

(b)

TABLE 7.1 – Comparaisons de la combinaison de l'algorithme LinLog et de la distance angulaire, obtenus pour les graphes de la figure 7.5, avec l'algorithme 2-opt et le recuit simulé. (a) : support=0,52. (b) : support=0,46.

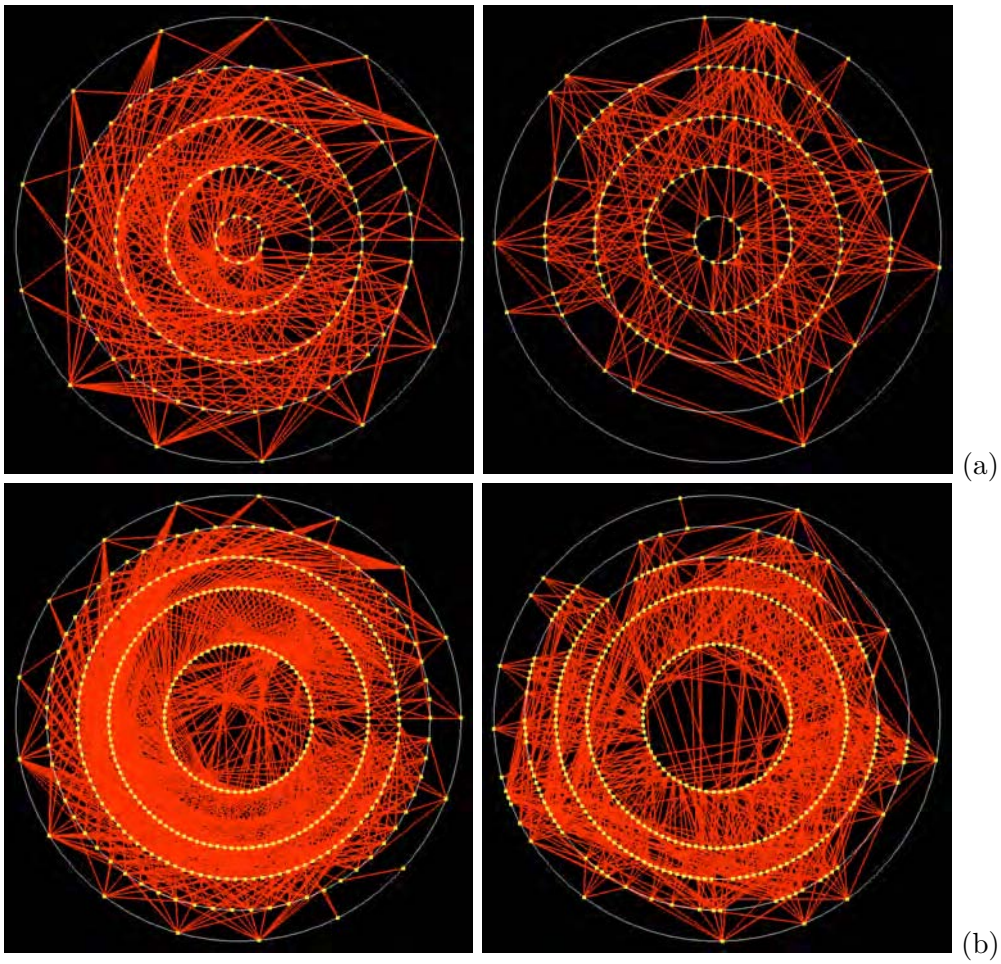


FIGURE 7.5 – Exemples d'optimisation du graphe circulaire.
(a) De gauche à droite : graphe original montrant des itemsets et les connexions entre eux (support=0,52 - distance minimale entre les itemsets = 0,05 - nombre de segments = 465 - somme de la longueur des segments = 137). Ensuite, le graphe est optimisé par recuit simulé (somme de la longueur des segments = 82). (b) La même séquence est réalisée avec un support égal à 0,46 (distance minimale entre deux itemsets = 0,03). Le graphe contient 1262 segments dont la somme des longueurs est 378 puis 212.

Chapitre 8

Validation expérimentale du graphe circulaire

8.1 Objectif et hypothèse

La visualisation des itemsets sous forme de graphes circulaires a fait l'objet d'une expérimentation, afin de valider l'intérêt de l'optimisation du placement des nœuds pour trouver de l'information avant toute interaction, c'est-à-dire lorsque la totalité du graphe est affichée. En effet, l'interaction avec un itemset peut avoir pour effet de masquer la partie du graphe qui ne concerne pas les supersets ni les sous-sets associés. Le but de cette expérimentation est d'étudier dans quelle mesure l'utilisateur peut explorer un graphe circulaire présenté intégralement.

Une taxonomie des tâches à réaliser pour valider un graphe a été présentée par Lee et al. en 2006 [Lee 06]. Elle repose notamment sur la tâche à réaliser. En présentant les k-itemsets sous forme de nœuds dans le graphe circulaire, une tâche consiste par exemple à identifier si deux k-itemsets se combinent en un superset commun, ce qui signifie qu'ils partagent de l'information. Cela peut être également la recherche de k-itemsets partageant le même superset. Étant donnée la tâche à réaliser et le type de graphe, les caractéristiques suivantes ont été mesurées :

- Adjacence : connexion directe entre deux nœuds consécutifs. Elle permet de considérer les supersets du niveau suivant d'un itemset, ou les sous-sets du niveau précédent ;
- Accessibilité : connexion directe ou indirecte entre deux nœuds. Si la connexion est directe, alors il s'agit du critère d'adjacence. Cela correspond à la recherche de supersets ou de sous-sets d'un itemset donné ;
- Connexion commune entre plusieurs nœuds. Cela concerne par exemple la recherche des supersets issus d'un itemset, ou ceux qui en sont à l'origine.

L'étude de ces critères a été ramenée à la considération d'une tâche élémentaire, qui

consiste à détecter si deux nœuds sont connectés ou pas, plus précisément si l'un des nœuds est un superset de l'autre. L'expérimentation a donc porté sur cette tâche. La performance qui a été mesurée s'appuie sur le temps de réponse et sur le taux de bonnes réponses. Nous appelons *qualité* de la réponse l'exactitude de celle-ci.

Dans cette expérimentation, nous faisons l'hypothèse que l'exploration du graphe optimisé apporte, par rapport au graphe original :

- (H1) un meilleur taux de bonnes réponses ;
- (H2) un meilleur temps de réponse.

Si ces hypothèses sont validées, cela montrera qu'il sera plus facile de faire le lien entre les itemsets pour détecter des supersets communs à des itemsets, et des sous-sets étant à l'origine d'un même itemset.

8.2 Tâche expérimentale

Aucune compétence particulière liée à un métier n'est requise de la part du sujet. Sa tâche consiste à dire si deux nœuds particularisés sont connectés. Des images lui sont présentées, chacune montrant un graphe optimisé ou non optimisé, dont deux nœuds sont particularisés par une couleur et une taille différentes des autres. Nous appelons la distance entre les deux nœuds le nombre minimum de segments qui permet de les relier. Elle est égale à 1 dans le cas de l'adjacence, et 2 ou 3 dans le cas de l'accessibilité. De plus, afin de tester des graphes plus ou moins chargés, différents nombres de nœuds par cercle sont considérés, ainsi que des graphes à quatre et cinq cercles.

Chaque paire de nœuds est proposée, dans deux essais différents, sur un graphe non optimisé ainsi que sur le graphe optimisé correspondant. De plus, pour un essai donné entre deux points connectés, un autre essai est proposé entre deux autres points non connectés impliquant les mêmes cercles. Pour tester différents graphes présentant des quantités d'informations plus ou moins volumineuses, deux graphes à quatre cercles et deux graphes à cinq cercles sont ainsi utilisés. Enfin, quatre essais avec des distances de 1 à 3 sont proposés. Cela donne donc le nombre d'images suivantes :

$$Nb_{images} = 2 \times 2 \times 4 \times 3 \times 4 = 192$$

Comme une situation est présentée dans les deux modes optimisé et non optimisé, il est nécessaire que l'ordre d'apparition ne soit pas toujours le même pour éviter tout biais lié à un éventuel effet d'apprentissage. Ainsi, l'ordre d'apparition des essais est aléatoire.

Ces exemples de graphes correspondent à des situations rencontrées en fouille de données, où le nombre d'attributs et les nombres de clusters associés peuvent être plus ou moins importants. Ils correspondent aux 1-itemsets, c'est-à-dire aux itemsets situés sur le cercle le plus grand du graphe.

8.3 Variables

Les variables indépendantes sont les suivantes :

- Optimisation du graphe : avec ou sans ;
- Distance entre les deux nœuds particularisés : 1, 2 ou 3. Elle joue le rôle d'indice de difficulté.

Le nombre de cercles et de nœuds constituent des facteurs secondaires car, même s'ils ont des conséquences sur les résultats mesurés, ils ne font l'objet d'aucune hypothèse. Dans un souci de contrebalancement, nous avons prêté attention à ce que ces facteurs soient uniformément répartis dans les différents essais.

Les variables dépendantes sont les suivantes :

- Temps de réponse. Il s'agit du temps entre l'apparition de l'image et la sélection de la réponse ;
- Qualité de la réponse. Celle-ci peut être correcte ou fausse.

8.4 Dispositif expérimental

Le dispositif expérimental est constitué d'un ordinateur, d'un écran 21 pouces et d'une souris. Le sujet est assis devant l'écran situé à une distance de 70 cm de sa tête, et ne doit pas avoir de problèmes visuels particuliers avec un port éventuel de verres correcteurs. Le logiciel de l'expérimentation présente deux zones (Cf. Figure 8.1) :

- Zone de l'image. Elle présente un graphe circulaire dont deux nœuds sont particularisés ;
- Zone de réponse. Il s'agit de deux boutons *Connectés* et *Non connectés* positionnés sur la droite.

Afin d'avoir une distance identique entre la position initiale du curseur et les boutons, et pour ne pas avoir à chercher le curseur à chaque essai, celui-ci est positionné au centre de chaque nouvelle image montrée. 192 images sont ainsi présentées au sujet. Chacune reste affichée tant que le sujet n'a pas cliqué sur l'un des deux boutons. Dès la validation, l'image disparaît pendant une durée de 0,3 seconde, puis l'image suivante est affichée. Cette attente a pour but d'éviter le double-clic involontaire. Un message indique la fin de l'expérimentation.

La consigne transmise au sujet est de sélectionner l'un des deux boutons pour dire si les deux points particularisés sont connectés ou non. Il lui est demandé de le faire le plus rapidement possible en essayant de bien répondre. Pour le familiariser avec le dispositif, une phase d'apprentissage préalable a été organisée avec cinq exemples de graphes illustrant les trois types distances, dans des situations plus ou moins chargées, optimisées et non optimisées. A l'issue de la passation, un entretien avec le sujet avait pour but d'identifier la stratégie mise en œuvre pour sélectionner une réponse.

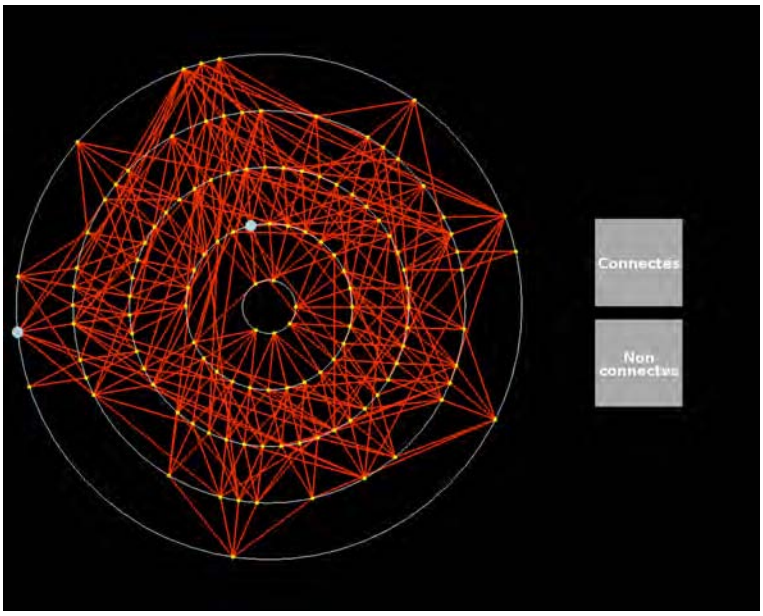
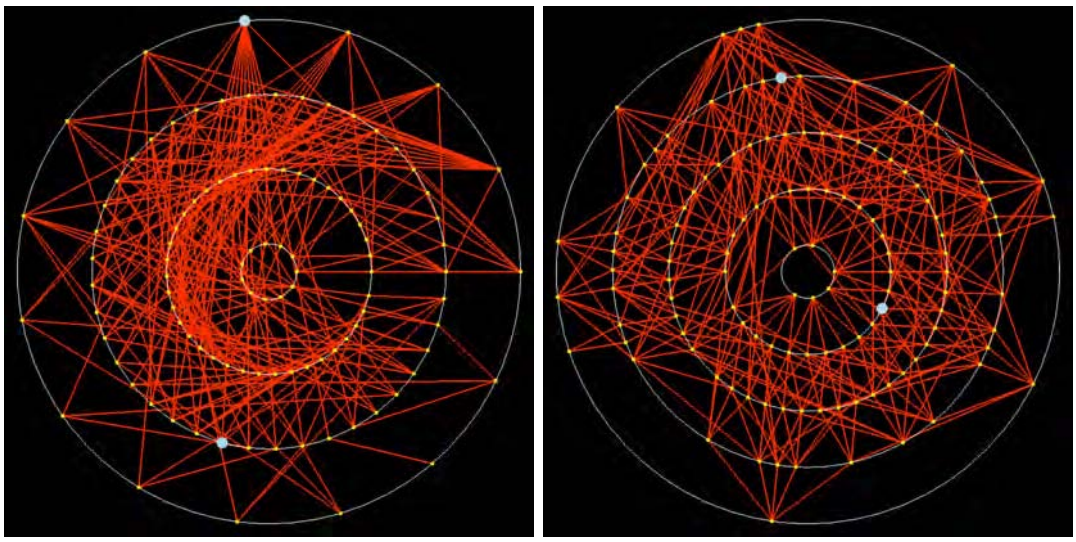
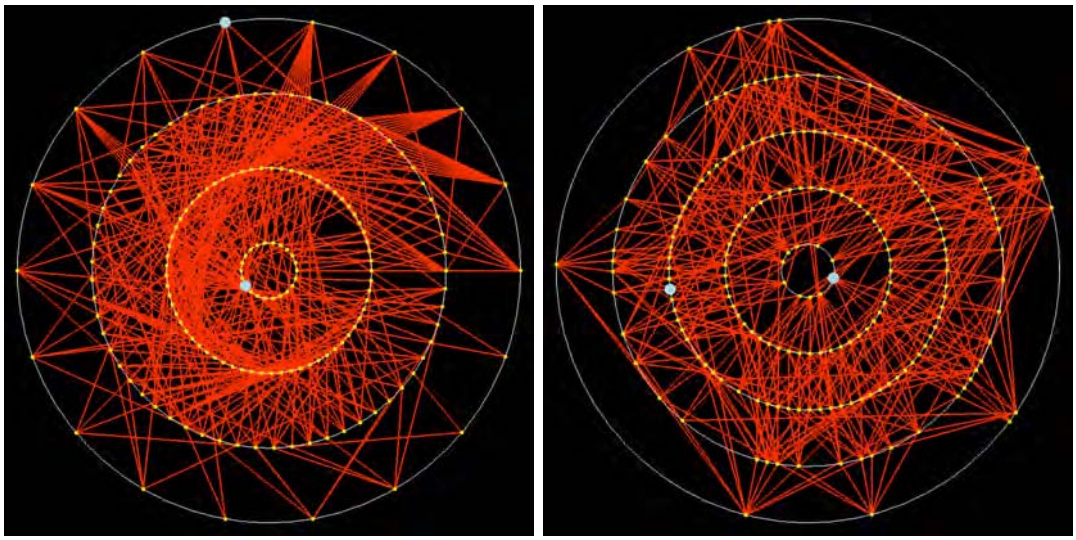


FIGURE 8.1 – Dispositif expérimental de validation de l'optimisation du graphe circulaire. Le sujet doit trouver si l'un des nœuds particularisés en bleu est superset de l'autre.



(a) (b)

FIGURE 8.2 – Exemples de graphes non chargés.
(a) : non optimisé, 4 cercles, distance = 1. (b) : optimisé, 5 cercles, distance = 2.



(a) (b)

FIGURE 8.3 – Exemples de graphes chargés.
(a) : non optimisé, 4 cercles, distance = 3. (b) : optimisé, 5 cercles, distance = 2.

L'expérimentation a été réalisée sur un groupe de 20 sujets âgés de 27 à 58 ans. La moyenne d'âge est 43 ans et l'écart-type 7,4 ans. Tous sont des utilisateurs réguliers d'outils informatiques.

8.5 Résultats

Les données ont été traitées à l'aide de l'analyse de variance (ANOVA) appliquée à un groupe de sujets appariés, parce que tous les sujets ont été soumis aux mêmes essais, dans les mêmes conditions expérimentales. En effet, ils ont fait l'objet de mesures, dans les six situations différentes correspondant aux trois distances et aux deux types de graphe, optimisés et non optimisés. Le logiciel utilisé est Statistica, version 10.0.

La figure 8.4 montre le temps de réponse en fonction de la distance dans les deux conditions optimisée et non optimisée. L'optimisation a bien une incidence sur le temps de réponse ($F(1,19) = 18.533$, $p = .00038 < .05$), ainsi que la distance ($F(2,38) = 47.592$, $p = .00000 < .05$). Nous remarquons que l'hypothèse est vérifiée en augmentant la difficulté, c'est-à-dire que le temps de réponse est meilleur pour les graphes optimisés au fur et à mesure que la distance augmente. Par ailleurs, une interaction entre la distance et l'optimisation du graphe est constatée ($F(2,38) = 4{,}1316$, $p = .02379 < .05$). L'augmentation du temps de réponse avec la distance est plus importante sur le graphe non optimisé que sur le graphe optimisé. Le graphe optimisé est donc plus performant en termes de temps de réponse que le graphe non optimisé (H2).

La figure 8.5 montre la qualité de la réponse en fonction de la distance, toujours dans les

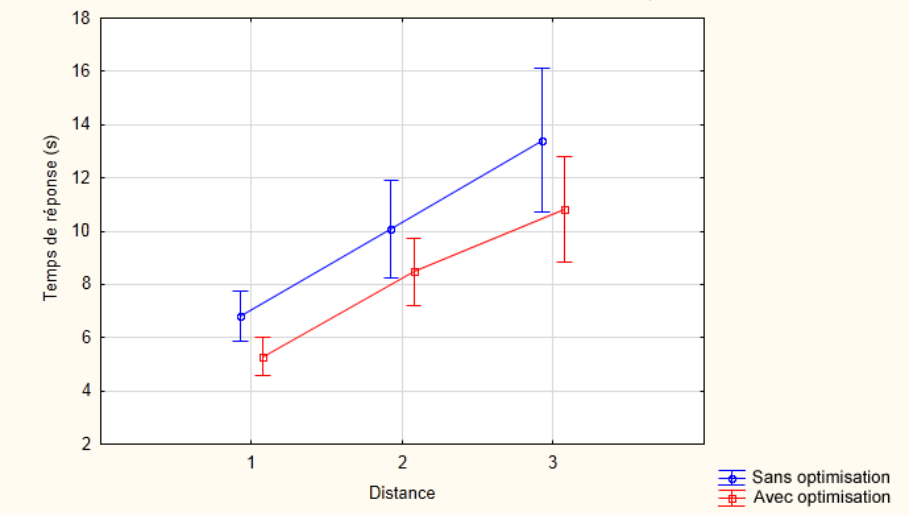


FIGURE 8.4 – Temps de réponse en fonction de la distance pour l'ensemble des essais.

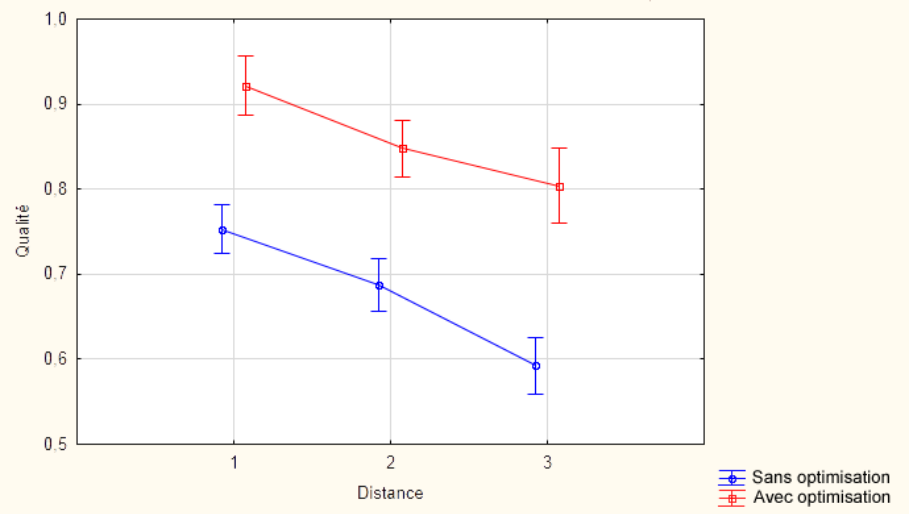


FIGURE 8.5 – Taux de bonnes réponses en fonction de la distance pour l'ensemble des essais.

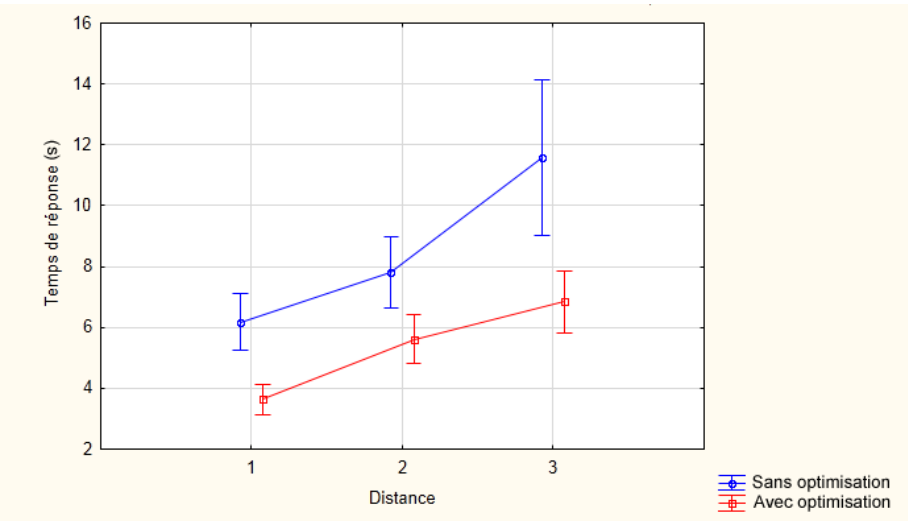


FIGURE 8.6 – Temps de réponse pour les essais de type « Connecté » ayant fait l'objet d'une bonne réponse.

deux conditions. L'influence de l'optimisation ($F(1,19) = 194.62$, $p = .00000 < .05$) et de la distance ($F(2,38) = 128.96$, $p = .00000 < .05$) sur la qualité de la réponse est bien vérifiée par l'analyse des données (H1). Elle indique que la qualité de la réponse est meilleure avec les graphes optimisés et décroît avec l'augmentation de la distance. Cependant l'interaction entre l'optimisation du graphe et la distance n'est pas avérée ($F(2,38) = 2.2465$, $p = .11964$). Cela signifie que la diminution de la qualité de la réponse en fonction de la distance n'est pas influencée par l'optimisation du graphe. Cependant, cela ne faisait partie des hypothèses de départ, et l'hypothèse selon laquelle le graphe optimisé apporte une meilleure qualité de réponse que le graphe non optimisé est vérifiée (H1).

L'entretien qui a eu lieu avec les sujets, sur leur stratégie mise en œuvre pour choisir une réponse, montre que 15% de ceux qui ne savaient pas répondre, quelle que soit la condition, sélectionnaient celle qui leur paraissait la plus probable, alors que 85% de ceux-ci sélectionnaient en général le bouton *Non connectés*. La qualité des réponses de type « Non connecté » n'est donc pas significative dans les cas d'essais difficiles. Cependant, cela ne remet pas en cause la tâche de l'utilisateur de ce graphe, qui est de repérer les connexions, et non les non-connexions. Nous nous sommes donc intéressés plus particulièrement aux essais de type « Connecté » pour lesquelles les sujets ont bien répondu, afin de valider notre hypothèse sur le temps de réponse, avec cette restriction. Ces essais de type « Connecté » ont donné lieu à 71% de bonnes réponses.

La figure 8.6 montre le temps de réponse en fonction de la distance dans les deux conditions. L'optimisation a bien un impact sur le temps de réponse ($F(1,19) = 44.421$, $p = .00000 < .05$), ainsi que la distance ($F(2,38) = 35.940$, $p = .00000 < .05$), et il n'y a pas d'interaction entre l'optimisation et la distance ($F(2,38) = 6.7859$, $p = .00302 < .05$). Sur cette figure, nous pouvons par ailleurs constater que le temps de réponse augmente sensiblement pour la distance 3 dans le graphe non optimisé, alors que cette augmentation est moindre dans le cas

du graphe optimisé. Cela montre que, quand le sujet détecte bien la connexion entre deux nœuds, il met sensiblement plus de temps au fur et à mesure que la difficulté augmente, ce qui est accru lorsque le graphe n'est pas optimisé.

8.6 Conclusion

Dans cette expérimentation, nous souhaitions montrer, à partir d'une tâche simple, que l'exploration du graphe optimisé était meilleure que celle du graphe non optimisé. La performance que nous avons mesurée est la qualité (H1) et le temps de la réponse (H2), en fonction de l'optimisation du graphe et de la difficulté, correspondant à la distance entre les nœuds, c'est-à-dire au nombre minimum de segments permettant de relier deux nœuds. Les hypothèses, selon lesquelles la qualité (H1) et le temps de réponse (H2) sont meilleurs sur les graphes optimisés que sur les non-optimisés, sont vérifiées, malgré la part d'aléatoire quand les sujets ne savent pas détecter la connexion entre deux nœuds. Cependant, dans tous les cas, ils mettent plus de temps à répondre si le graphe n'est pas optimisé et/ou si la distance est plus grande. La non-détection de la connexion montre les limites de cette représentation dès qu'elle n'est plus suffisamment lisible. Mais cette limite va être compensée par les opérateurs d'interaction, qui vont permettre de particulariser les liens ascendants et descendants à partir de ceux-ci. Ce point fera partie de l'optimisation du graphe, qui sera traitée dans le prochain chapitre.

Chapitre 9

Optimisation du graphe circulaire et illustration sur des benchmarks

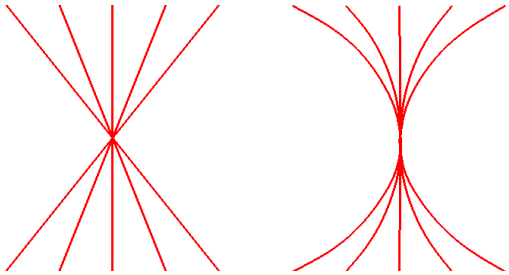
Plusieurs pistes d'amélioration du graphe circulaire ont été étudiées, afin d'accroître les liens entre les données, et pour détecter celles qui sont les plus impliquées dans la base. Dans un premier temps, les arêtes proches sont regroupées selon une technique de bundling. Grâce à l'optimisation (Cf. Chapitre 7.3), cette proximité a un sens, parce que les itemsets partageant des données communes sont rapprochés, puis cela est exploité dans le bundling. Le graphe est ensuite mis en valeur par l'assignation de mesures, issues de la fouille de données, aux variables visuelles, que sont la couleur, la transparence et la largeur de la ligne. Finalement, un opérateur de sélection permet de se focaliser sur des arêtes afin de particulariser les parties du graphe qui leur sont reliées. Ce processus d'optimisation est résumé dans la figure 9.1. Les données utilisées pour illustrer ce chapitre sont issues de la base Mushrooms de l'U.C.I. [1].

9.1 Principes du Graph Bundling

Alors que le nombre d'arcs peut être très élevé dans la représentation des données dans un graphe, les méthodes de bundling qui leur sont appliquées sont de plus en plus étudiées. Le bundling de graphes est issu des recherches sur les dessins confluents, en réduisant les graphes non planaires en graphes planaires, qui ont été présentées par Dickerson et al. [Dickerson 03]. Le but était de permettre aux arcs de se regrouper pour être fusionnés et tracés comme des

1. http ://archive.ics.uci.edu/ml/datasets/Mushroom

FIGURE 9.1 – Pipeline du processus d'amélioration de l'exploration du graphe.



(a) (b)

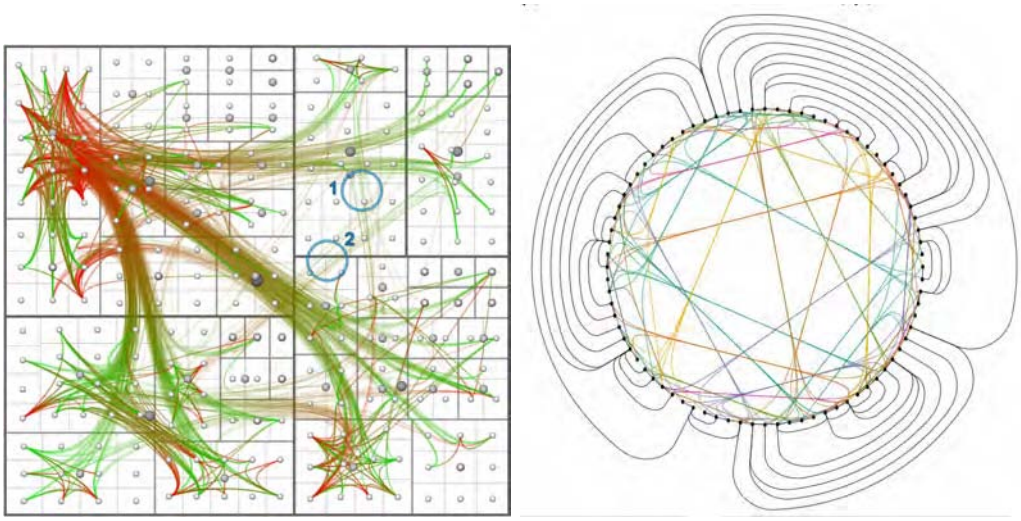
FIGURE 9.2 – Principe du regroupement des arcs, ou bundling.
(a) : les arcs sont des arêtes rectilignes. (b) : les arêtes ont été courbées pour obtenir des arcs qui fusionnent en un seul chemin.

chemins.

Le bundling diminue fortement l'encombrement du graphe dû à l'enchevêtrement des arêtes, en faisant passer par le même chemin les arcs qui sont liés (Cf. Figure 9.2). Les raisons de cet enchevêtrement, et les stratégies pour les réduire, ont été discutées dans une taxonomie présentée par Ellis & Dix [Ellis 07]. Le bundling peut être considéré comme un affinement de la densité spatiale des arcs, en la rendant élevée le long des grappes d'arcs, et basse ailleurs [Hurter 12]. En améliorant ainsi la lisibilité du graphe, il produit des images dans lesquelles sa structure est de haut niveau, et est plus facile à suivre, alors que les arcs individuels sont moins mis en valeur.

Plus tard, Holten [Holten 06] regroupe les arcs initiaux en arcs résultants dont le chemin suit des B-splines. Gansner & Koren [Gansner 07] regroupent les arcs dans une vue circulaire, en optimisant la position des nœuds, en faisant passer les arcs par l'extérieur du cercle, et en les regroupant à l'aide de splines. Dwyer at al. [Dwyer 07] utilisent un effet de forces pour incurver les arcs et minimiser leur croisement. Cela procure des vues ayant des arcs regroupés. Avec le bundling utilisant des forces (FDEB), les groupes sont créés par attraction de points de contrôle sur les arcs qui sont rapprochés [Holten 09]. Dans [Selassie 11], ce concept a été adapté en séparant les groupes en prenant en compte la direction des arcs, et ainsi en donnant des directions opposées aux groupes.

Plusieurs autres méthodes ont abordé le regroupement d'arcs de graphes encombrés. Pour accélérer le processus, en utilisant le regroupement par niveaux, la méthode MINGLE a été présentée par Gansner et al. [Gansner 11]. Les cartes de flux produisent un regroupement de nœuds binaires dans un graphe de flux pour acheminer les arcs incurvés [Phan 05] et un maillage de contrôle est utilisé pour acheminer des arcs incurvés, par exemple dans [Qu 07], où une triangulation de Delaunay réduit l'encombrement visuel du graphe. Cette triangulation est également mise en œuvre [Zhou 09] dans un regroupement d'arcs hiérarchique basé sur un modèle d'énergie. Une autre méthode, appelée Skeleton-Based Edge Bundling (SBEB) [Ersoy 11], utilise une technique de squelette pour construire itérativement les chemins du bundling. Les arcs sont progressivement attirés par des lignes centrales dépendant de leurs



(a) (b)

FIGURE 9.3 – Exemples de techniques de bundling.
(a) : Holten [Holten 06] regroupe les arcs en les acheminant le long d'une structure hiérarchique utilisant les B-splines. (b) : Gansner & Koren [Gansner 07] proposent d'améliorer une représentation circulaire par des métriques d'optimisation.

distances exprimées sous forme de champs.

Pour améliorer la lisibilité du graphe après utilisation d'un algorithme de bundling, des méthodes de rendu sont utilisées, par exemple en utilisant une interpolation de couleurs pour signifier la direction des arcs [Holten 06, Cui 08], et la transparence ou la teinte pour signifier la densité de l'arc ou sa longueur [Lambert 10b, Ersoy 11]. Les arcs d'un graphe après bundling peuvent être enfin dessinés comme des formes compactes et étirées dont la structure est renforcé par des techniques d'ombrage ou *shading* [Telea 10, Scheepens 11].

L'algorithme de bunding que nous mettons en œuvre dans notre étude est la méthode Kernel Density Estimation-based Edge Bundling (KDEEB) présentée par Hurter et al [Hurter 12]. L'algorithme, appliqué à des graphes généraux, est entièrement graphique. En effet, grâce à une construction d'images gérée par la carte graphique, le dessin d'un graphe est transformé en carte de densité utilisant une estimation de densité de kernel. Une technique d'amélioration de la netteté fusionne progressivement les maxima de hauteurs, en déplaçant les chemins du graphe le long du flux de gradient de hauteur. Par ailleurs, les arcs sont itérativement lissés pour faire disparaître les discontinuités dues au traitement du gradient.

9.2 L'optimisation du graphe circulaire par le Bundling

Les algorithmes de bundling ont pour objectif de simplifier un graphe en déformant puis fusionnant des arêtes proches. Nous avons choisi l'algorithme Kernel Density Estimation-based Edge Bundling [Hurter 12] pour sa simplicité de mise en œuvre et sa rapidité d'exécution. Il

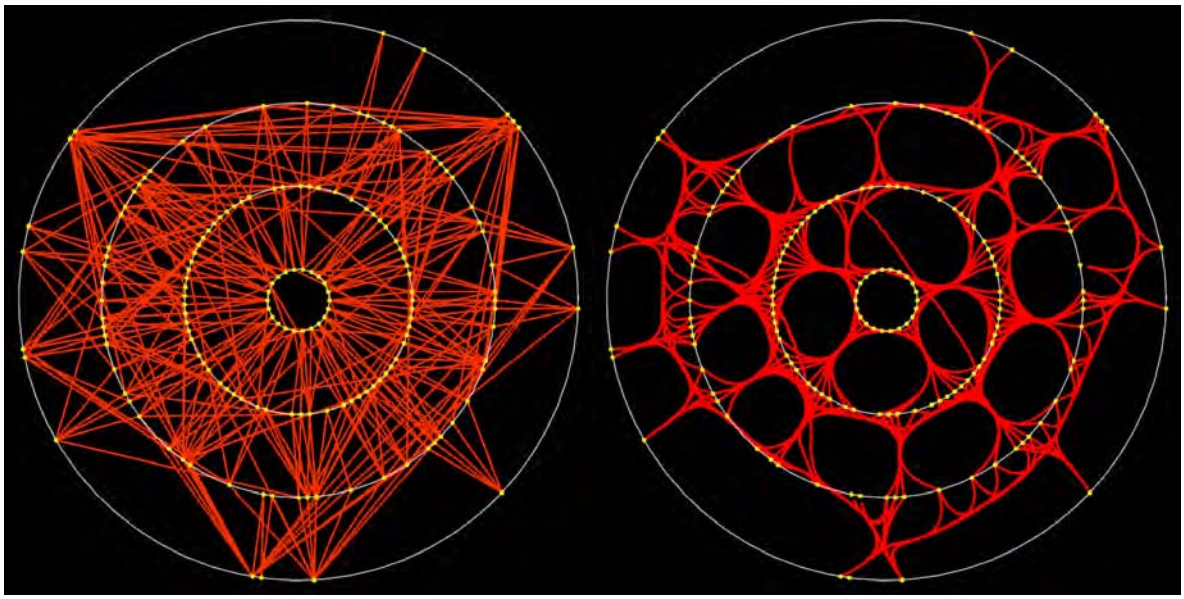


FIGURE 9.4 – L'algorithme de bundling courbe les arêtes pour obtenir des chemins reliant les nœuds.

est simple, parce qu'il suffit de la position de nœuds et des connexions associées pour utiliser l'algorithme. Il est rapide, parce que les calculs sont assurés par le processeur graphique, parallélisables et plusieurs fois plus rapides que les méthodes similaires. La figure 9.4 montre un exemple de graphe circulaire auquel a été appliqué l'algorithme KDEEB.

En considérant que chaque arête relie deux itemsets de deux cercles consécutifs, ses extrémités doivent rester sur leurs cercles respectifs, et rester à la même position, afin de garder le bénéfice de l'optimisation du placement des nœuds (Cf. Chapitre 7.3). C'est pourquoi, l'algorithme de bundling n'est pas appliqué à l'ensemble du graphe, mais successivement par groupes d'arêtes reliant des paires de cercles consécutifs. Ainsi, avec N cercles, il est mis en œuvre $N-1$ fois. Le graphe final est donc la combinaison de bundlings partiels, comme cela est illustré sur la figure 9.5. Comme les extrémités des arêtes ne sont pas déplacées, grâce à l'optimisation, les arêtes partageant la même information ont tendance à être fusionnées par l'algorithme de bundling.

9.3 Assignation de métriques aux variables visuelles

Dans la Sémiologie Graphique [Bertin 67], Jacques Bertin écrit que le graphique ne décrit que les liens établis entre les composants ou éléments (Cf. Chapitre 2.2). Le choix des variables visuelles est un facteur primordial qui contribue, non seulement à assurer une bonne lisibilité du graphique, mais également son intelligibilité. La couleur n'est pas par défaut une variable visuelle ordonnée. En effet, il n'est pas possible de dire que le bleu est plus grand ou plus petit que le jaune. Cependant, en utilisant un gradient d'une couleur à une autre, elle devient

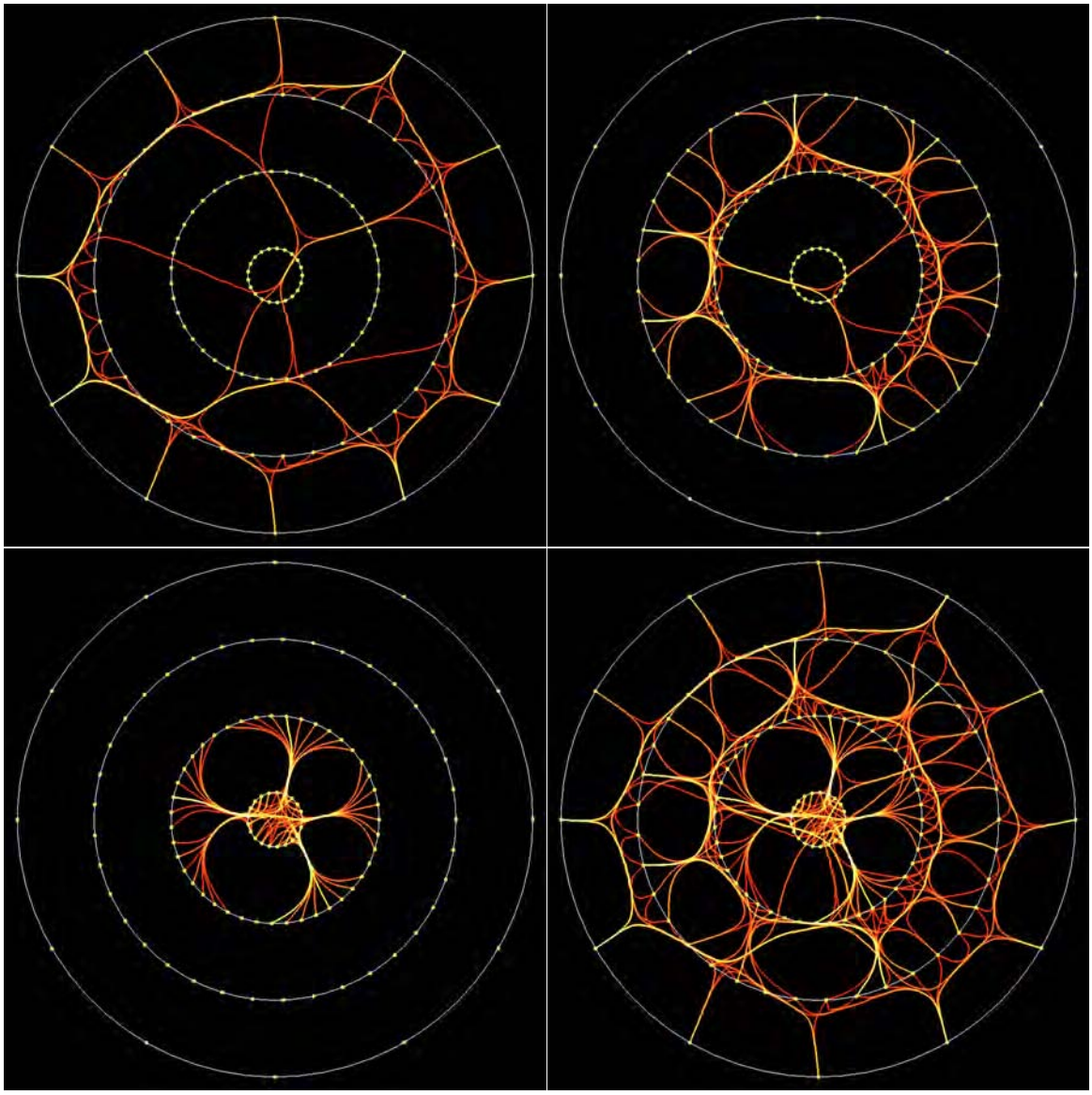


FIGURE 9.5 – La forme finale du graphe (en bas à droite) est la combinaison de bundlings appliqués aux arêtes reliant les paires de cercles successifs.

ordonnée. Par exemple, pour un gradient du rouge pour des valeurs inférieures au jaune pour des valeurs supérieures, il est possible de dire si telle couleur de ce gradient est avant telle autre. Dans l'optimisation du graphe, nous utiliserons de tels gradients de couleurs.

Le support et les autres mesures qui émanent des algorithmes de Data Mining [Guillet 07], comme la confiance ou le lift (Cf. Section 1.5.3), sont assignés à des variables visuelles du graphe, dans le but d'en faire ressortir graphiquement des propriétés et donc des données associées. En considérant la couleur et la largeur des arêtes, la visualisation propose une représentation de l'ensemble des itemsets. Elle est enrichie par les propriétés suivantes :

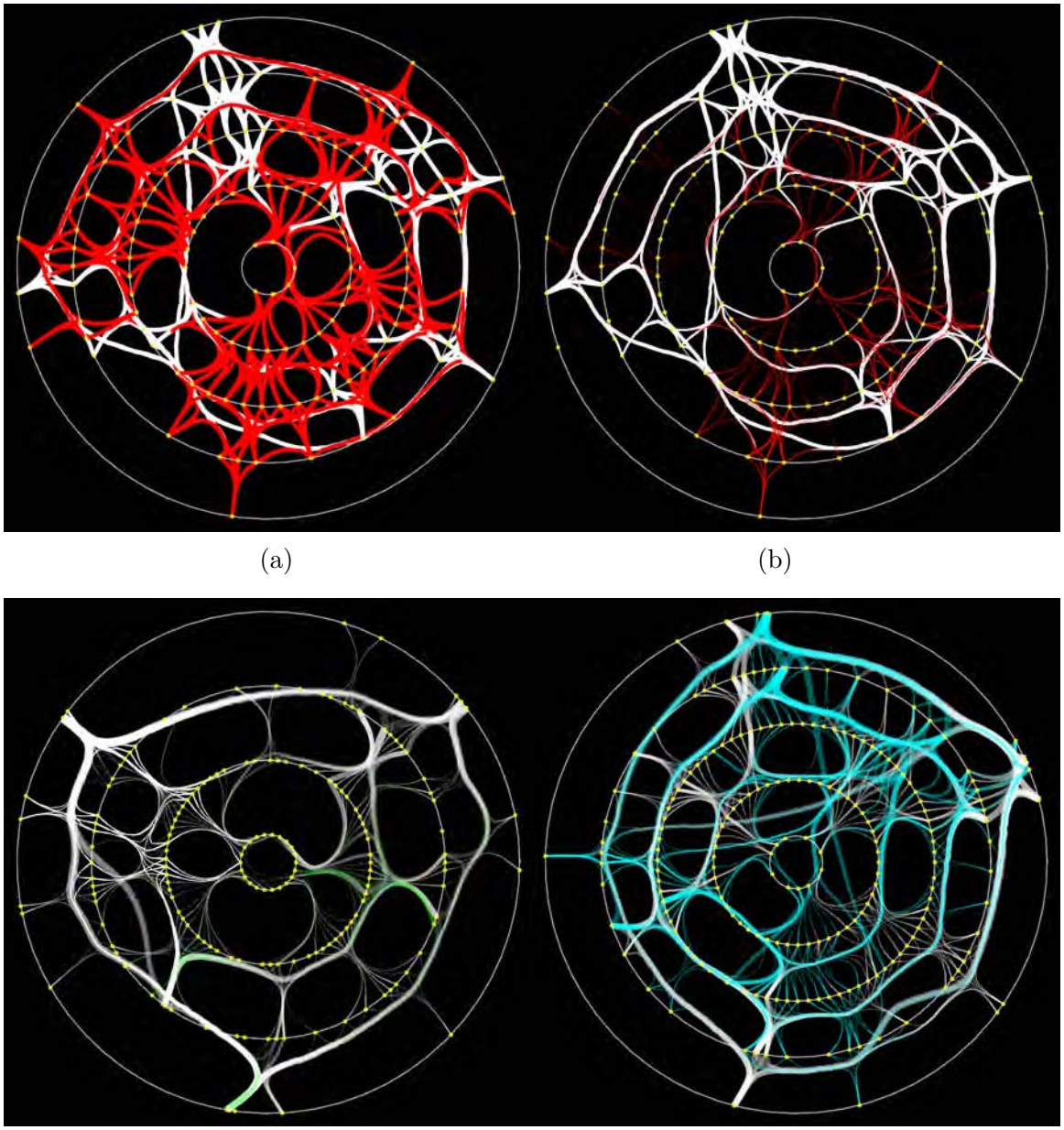
- Identification des itemsets pertinents ;
- Zones pertinentes du graphe circulaire.

9.3.1 Identification des itemsets pertinents

Grâce à l'association du gradient de couleur au support, ainsi qu'éventuellement la transparence et la largeur des arêtes, les itemsets ayant les valeurs de support les plus élevées sont accentués. De plus, l'évolution de ces variables visuelles du cercle le plus grand vers le plus petit, fournit de l'information au sujet de l'évolution du support au fur et à mesure que le nombre d'itemsets augmente. Sur le graphe (a) de la figure 9.6, le support est d'abord assigné à la couleur qui va du rouge, pour les plus faibles valeurs, au blanc, pour les plus élevées. Sur le (b), le support est assigné à la couleur, l'alpha et la largeur des lignes courbes. Cela permet d'atténuer les lignes qui correspondent aux supports les plus faibles, et ainsi de faire ressortir celles dont les itemsets ont les supports sont les plus élevés. Dans les graphes (c) et (d), nous montrons des exemples d'assignation de différentes mesures aux variables visuelles. Il s'agit de la confiance moyenne des règles, dans lesquelles sont impliqués les itemsets, du lift moyen et de la confiance maximale. Les mesures, abordées dans le chapitre 1.5.3, sont assignables, de la même manière, aux variables visuelles, en fonction des besoins de l'utilisateur.

9.3.2 Zones pertinentes du graphe circulaire

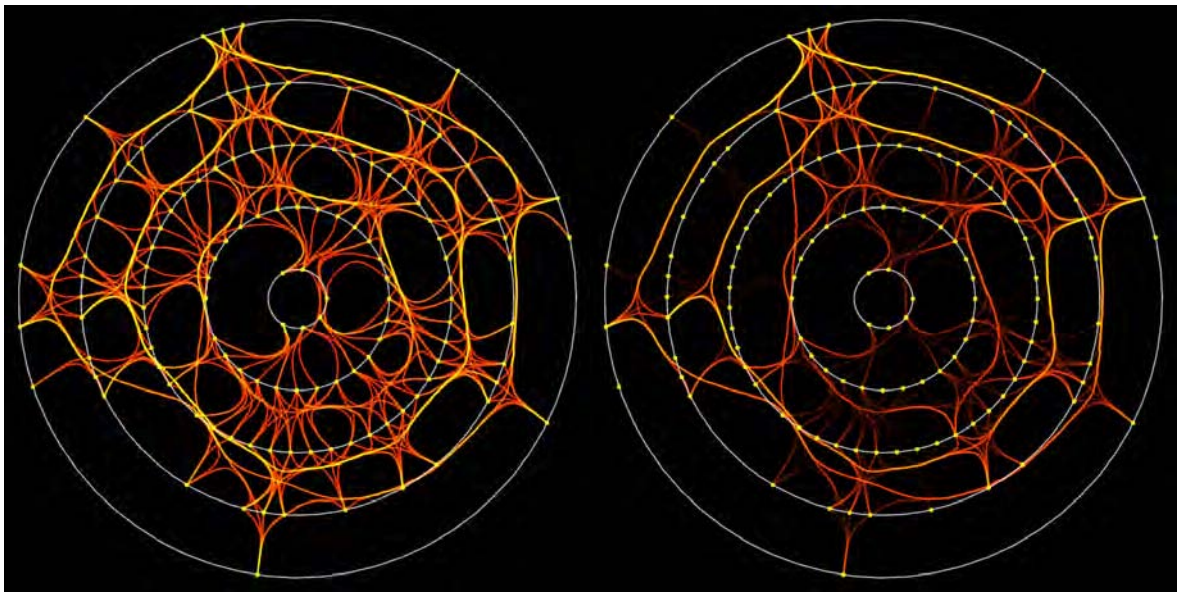
Une autre manière d'obtenir des informations supplémentaires sur les itemsets est l'utilisation d'accumulation de couleur et de transparence qui a pour effet de combiner les composantes (R, G, B, A) des lignes, pour obtenir une couleur et une transparence résultantes. L'utilisation de cette technique est une conséquence de l'algorithme de bundling, qui courbe les arêtes pour obtenir des lignes qui se chevauchent. Ainsi, plus les arêtes sont impliquées dans les lignes courbes, plus l'effet d'accumulation de couleur et/ou de transparence est fort. Ainsi, un itemset ayant un support élevé apparaît généralement avec une couleur élevée et une opacité élevée. De même, si les itemsets ont un support bas, alors ils sont moins visibles et leur couleur a une valeur basse, car l'accumulation est basse. Cette accumulation a pour effet de faire ressortir graphiquement les itemsets fréquents les plus impliqués dans la base de données, et d'atténuer les autres. La figure 9.7 montre d'abord une accumulation de couleurs, puis une accumulation combinée de la couleur et de l'alpha. Celles-ci, en lien avec l'optimisation du



(a) (b)

(c) (d)

FIGURE 9.6 – Exemples d'assignation de métriques aux variables visuelles. (a) : le support est assigné à la couleur (gradient de rouge à blanc). (b) : en plus de l'assignation du graphe précédent, le support est également assigné à l'alpha et la largeur des lignes. (c) : le support est assigné à la largeur des courbes, la confiance moyenne à la couleur (de vert à blanc) et le lift moyen à l'alpha. (d) : le support est toujours assigné à la largeur des courbes, le lift moyen à la couleur (de bleu à blanc) et la confiance maximale à l'alpha. L'assignation des mesures aux variables visuelles enrichit ainsi la quantité d'information visualisée.



(a) (b)

FIGURE 9.7 – Exemples d'accumulation de couleur (a) puis de l'alpha (b). L'accumulation accentue les effets du bundling. Plus le nombre de lignes impliquées est élevé dans les chemins résultants du bundling, plus l'accumulation est forte. Cela met en exergue les zones du graphe ayant le plus de lignes regroupées, et ainsi les zones où les itemsets partagent le plus d'information.

placement des itemsets sur le graphe circulaire, met ainsi en valeur les zones où les itemsets sont les plus impliqués, qui sont les zones de présence du maximum d'information pertinente. Elles correspondent aux itemsets ayant les supports les plus élevés et les plus fortement liés entre eux.

9.4 Sélection

Dans le but de focaliser l'exploration du graphe sur un ou plusieurs itemsets, un opérateur de sélection est utilisé. Grâce à la sélection, seules les connexions amont et aval des itemsets concernés sont visualisées, par un effet de propagation. Les autres connexions sont alors masquées. Ainsi, en sélectionnant l'itemset A de la figure 7.1, cela revient à particulariser ses connexions avec les itemsets AC et AE, et de AC et AE à ACE. La sélection de AC particularise ses connexions avec A, C et ACE. Enfin, la sélection de ACE particularise ses liens avec AC, AE et CE, et ensuite vers A, C et E. La figure 9.8 montre des exemples de sélection. Sur le graphe (a), la sélection du 1-itemset *Gill spacing=close* (support = 81,1%) montre qu'il a cinq supersets sur le second cercle. La propagation vers les cercles suivants donnent ensuite sept, quatre et finalement un itemset.

Il peut également intéressant de connaître quels sont les 1-itemsets qui partagent les

mêmes 2-itemsets qu'un 1-itemset sélectionné. Une fonctionnalité de rétro-propagation permet d'obtenir ce type d'information. Elle permet de particulariser différemment ces 1-itemsets, à l'aide d'une couleur définie par l'utilisateur. Dans notre illustration, il s'agit de la couleur verte. Ainsi, la sélection d'un 1-itemset provoque une propagation vers les supersets, à partir desquels se produit une rétro-propagation vers les 1-itemsets étant à l'origine de ces supersets. Grâce à cette fonctionnalité, quand un 1-itemset est sélectionné, il est facile de détecter les autres 1-itemsets avec lesquels il est lié. Les 1-itemsets correspondant aux attributs, cela permet de particulariser les autres attributs qui sont liés à un attribut particulier. Par ailleurs il est possible de réaliser cette propagation à partir de n'importe quel cercle du graphe. Le graphe (b) de la figure 9.8 illustre ce concept. Nous pouvons voir que le attributs qui partagent les mêmes supersets que *Gill size=broad* sont *Gill attachment=free*, *Veil color=white* et *Veil type=partial*.

Un autre type de sélection est appliqué aux arêtes par un opérateur de type brushing (Cf. Figure 9.8 (c)). En sélectionnant des arêtes, cela revient à sélectionner toutes les arêtes qui ont été regroupées par l'algorithme de bundling. Cela particularise ainsi les connexions amont et aval liées aux arêtes sélectionnées. C'est un moyen de détecter rapidement les itemsets liés à ces connexions.

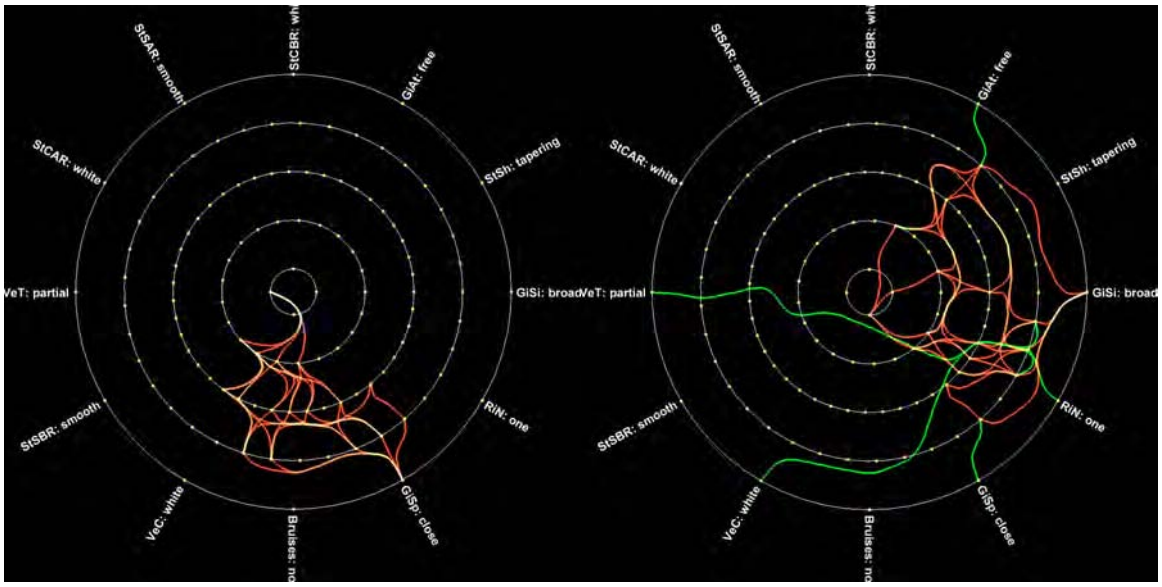
Ainsi, grâce à la sélection, il est possible de focaliser l'exploration sur un itemset ou un groupe d'itemsets, en faisant ressortir les autres itemsets qui leur sont liés, et en atténuant les autres parties du graphe.

9.5 Intérêt de la représentation optimisée des itemsets fréquents sur un graphe circulaire

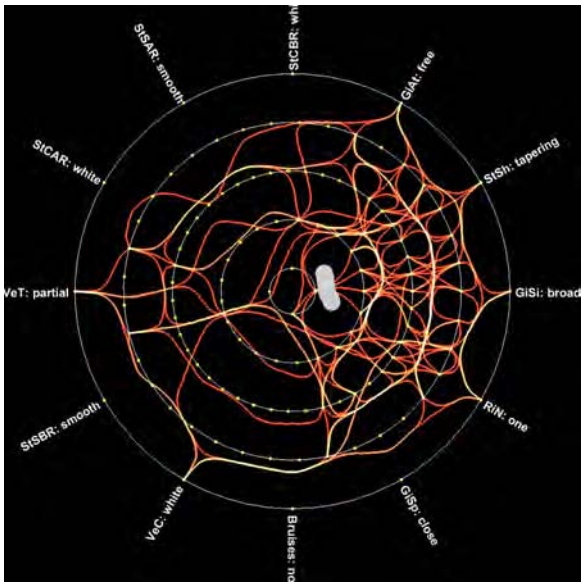
Dans la suite de ce chapitre, nous discutons de l'intérêt de notre approche en relation avec d'autres travaux. Pour cela, nous suivrons les étapes du processus d'amélioration du graphe circulaire (Cf. Figure 9.1).

9.5.1 Le graphe circulaire

Nous proposons une représentation innovante des itemsets fréquents sous forme de cercles concentriques. Généralement, les études proposent des visualisations horizontales [Yang 03, Leung 08a, Leung 08b, Leung 09], à l'exception, par exemple, de Keim et al. [Keim 05] avec le visualiseur hiérarchique radial FP-Viz, ou de la représentation de l'hypergraphe de Glatz et al. [Glatz 12]. Notre graphe n'est pas une vue compressée des données, comme c'est le cas pour FpViz [Leung 09] et WiFIsViz [Leung 08b], mais il présente la totalité des itemsets fréquents, représentés par des nœuds. Ceux-ci sont reliés entre eux par des arcs qui illustrent les relations entre eux. Ces connexions permettent d'avoir une connaissance de la manière dont les itemsets sont construits à partir de sous-sets, et de la manière dont ils interviennent dans la construction des supersets. Un avantage de la représentation circulaire est qu'elle est



(a) (b)



(c)

FIGURE 9.8 – Quelques exemples de sélections avec la base de données Mushroom de l'U.C.I. (a) : alors qu'un 1-itemset est sélectionné, la propagation montre quels itemsets proviennent de celui-ci. (b) : la sélection d'un 1-itemset est disséminée vers les supersets, puis des 2-itemsets vers leurs 1-itemsets associés. Cela montre ainsi quels attributs partagent les mêmes itemsets qu'un attribut donné. (c) : un opérateur de type brushing sélectionne des arêtes. Le graphe ne montre alors que les itemsets qui sont affectés par cette sélection.

obligatoirement étalée dans la vue, parce qu'elle s'étend à partir du cercle extérieur qui est proche des limites de celle-ci. De plus, la distance maximale entre deux itemsets de même ordre est le diamètre de leur cercle associé. Dans WiFIsViz et FpViz, les itemsets sont des lignes horizontales et parallèles contenant les nœuds correspondant aux attributs. Dans FIsViz [Leung 08a] et dans les travaux de Yang [Yang 03], les itemsets sont représentés par des lignes brisées. Comme pour FP-Viz [Keim 05], nous proposons une vue hiérarchique des itemsets basée sur des cercles. FP-Viz dispose les 1-itemsets sur le plus petit cercle et les plus grands itemsets sur le cercle extérieur. Dans notre approche, nous faisons l'inverse, en ce sens que les 1-itemsets sont sur le cercle extérieur et les plus grands itemsets sont sur le cercle intérieur. Ainsi, nous disposons de plus d'espace pour visualiser des données relatives aux 1-itemsets qui correspondent aux attributs des données. Ces informations sont ainsi plus lisibles que si les 1-itemsets avaient été disposés sur le cercle intérieur.

9.5.2 Optimisation du graphe

Pour améliorer la visualisation du graphe et donner un sens à la position des itemsets, qui sont par défaut disposés dans leur ordre d'apparition dans la base de données, nous les arrangeons à l'aide d'un processus d'optimisation. Celui-ci doit prendre en compte la distance entre les itemsets d'un même cercle et entre les itemsets de cercles consécutifs, en rapprochant ceux qui partagent de l'information commune et en les éloignant quand ce n'est pas le cas. Ce partage d'information a lieu quand deux itemsets sont sous-sets ou supersets d'un même itemset. Il s'agit donc de créer et d'arranger des groupes sur les cercles, relativement aux groupes des autres cercles. Rapprocher et éloigner des itemsets correspond à un phénomène d'attraction et de répulsion entre eux. Pour cela, nous faisons appel à un modèle d'énergie mettant en œuvre ce phénomène, pour positionner les itemsets sur un même cercle, en prenant en compte leurs connexions ascendantes et descendantes. De plus, ce modèle doit pouvoir gérer des petits groupes d'itemsets pouvant intervenir dans les cercles les plus petits. Pour cela, nous avons choisi le modèle LinLog [Noack 03]. Comme nous devons arranger également les groupes d'itemsets des différents cercles, les uns par rapport aux autres selon un principe d'attraction et de répulsion, nous considérons une distance angulaire entre les itemsets de cercles différents, nécessaire parce que les nœuds sont sur des cercles. Elle est nulle entre deux nœuds si leurs angles sont les mêmes. En effet, cela correspond à la plus petite distance qu'il peut y avoir entre deux nœuds, étant donnés deux cercles consécutifs. A partir de l'énergie du modèle LinLog que nous voulons minimiser et de la somme des distances angulaires que nous voulons également minimiser, nous calculons une distance résultante qui est la somme des distances intra-cercles et extra-cercles. Nous utilisons le même poids pour les deux distances parce qu'elles sont d'égale importance. En effet, pondérer de manière plus forte la distance intra-cercles risquerait de donner des groupes inter-cercles plus éloignés, et pondérer plus la distance extra-cercles étalerait les groupes au sein des cercles.

Pour minimiser la distance résultante, en cherchant un minimum global, nous utilisons l'algorithme du recuit simulé [Kirkpatrick 83, Cerny 85], qui est plus rapide que l'algorithme génétique [Ingber 92]. Ainsi, la proximité des itemsets devient pertinente, car elle est liée à un

partage d'informations entre eux. Cela signifie qu'ils sont associés pour créer des supersets, ou qu'ils ont des sous-sets en commun. Grâce au processus d'optimisation, les itemsets partageant de l'information sont donc rapprochés, et éloignés si ce n'est pas le cas. Cette pertinence dans leur positionnement a un impact direct sur les arêtes qui les relient, dont la proximité ou l'éloignement devient également lié à un partage d'information des itemsets qui leur sont associés.

L'expérimentation montre que la performance en termes de qualité et de temps de réponse est meilleure sur les graphes optimisés. En considérant la signification de la proximité des itemsets, que nous venons d'évoquer, l'optimisation du graphe facilite la détection d'un lien entre deux itemsets, et, dans le cas d'un éloignement, cela indique plus probablement une absence de lien. En d'autres termes, le positionnement relatif des itemsets et de leurs connexions permet d'avoir une connaissance des liens qui existent entre les attributs de la base de données, et c'est justement ce que nous cherchons à savoir dans une activité de fouille de données, lors de la recherche de motifs fréquents ou de règles d'association. Ainsi, l'optimisation permet, en regroupant et éloignant les itemsets et leurs groupes, d'explorer le graphe plus aisément. Cependant, malgré l'instauration d'une distance minimale entre deux itemsets et entre deux cercles, le graphe devient de plus en plus encombré quand le nombre d'itemsets et de connexions augmente, mais cet encombrement est retardé grâce à l'optimisation. Cependant, afin d'essayer de le limiter, nous faisons appel à une technique de bundling.

9.5.3 Le bundling

Nous considérons maintenant que, grâce à l'optimisation, la proximité des arêtes est liée à un partage d'informations entre les nœuds. Il devient alors pertinent de les transformer en arc, puis de les fusionner par un algorithme de bundling. Les chemins ainsi obtenus sont significatifs dans la structure générale du graphe et renforcent l'implication des itemsets. De plus, c'est une manière de diminuer son encombrement. Sans l'optimisation, le bundling agrégerait des arcs visuellement proches, mais dont la proximité n'aurait pas de sens. Dans notre étude, l'optimisation préalable est donc nécessaire avant de l'appliquer. Les études sur le bundling ont présenté des graphes circulaires ou des graphes radiaux pour illustrer des résultats d'algorithmes ou pour les améliorer [Holten 06, Gansner 07, Ersoy 11, Hurter 12, Hurter 13]. Mais ils ne comportent qu'un seul cercle ou anneau correspondant à une répartition de données. Dans notre étude, nous disposons de plusieurs cercles, et les données sont préalablement traitées pour les relier entre elles et les présenter autant sous forme initiale avec les attributs, mais également sous formes groupées avec les 2-itemsets, les 3-itemsets, etc.

9.5.4 L'apport de la sémiologie graphique

Nous renforçons, par un processus de rendu graphique, les caractéristiques du graphe ayant fait l'objet d'un algorithme de bundling, comme le poids des arcs, ainsi que les avantages issus de l'optimisation, comme sa simplification et une meilleure lecture de sa structure. Pour

cela, nous nous appuyons, de différentes manières, sur les apports de la sémiologie graphique [Bertin 67]. En assignant le support aux variables visuelles que sont la couleur et l'alpha, cela permet de faire ressortir son évolution le long des arcs. Nous utilisons un gradient de couleur, pour que cette variable visuelle devienne ordonnée au sens de la sémiologie graphique. Il est lié à une interpolation entre les valeurs des supports des itemsets qui sont aux extrémités de l'arc. En effet, seuls les itemsets, donc les nœuds, "supportent" la mesure issue de l'algorithme de Data Mining, mais, pour des raisons de fluidité de la visualisation, nous gardons une continuité dans l'affectation des variables visuelles. Cela permet de mieux voir leur évolution, donc l'évolution du support. Ainsi, comme le support correspond à l'implication des itemsets dans la base de données, le codage graphique met en valeur les zones du graphe où elles sont les plus impliquées. La largeur de la ligne étant une variable visuelle ordonnée, elle est également efficace pour montrer l'évolution du support. Il a été vu au chapitre 1.5.1 que le support est la mesure principale dans la recherche d'itemsets fréquents. En allant plus loin dans la fouille de données, les mesures associées aux règles d'association [Guillet 07] peuvent également être assignées aux variables visuelles. Ainsi, nous pouvons visualiser plusieurs informations associées aux itemsets grâce à ces affectations, dans le but d'améliorer l'exploration du graphe.

Dans [Holten 09, Selassie 11], dans le but de percevoir les poids des arcs issus de l'algorithme de bundling, plus le nombre d'arcs impliqués dans une ligne agrégée est élevé, plus le tracé de l'arc est large. [Holten 09] utilise également la couleur pour exprimer le nombre d'arcs agrégés, alors que [Selassie 11] utilise un code couleur pour indiquer la direction de l'arc. La direction de l'arc n'est pas intéressante pour nous, étant donnée la construction du graphe, qui présente les itemsets du plus grand cercle vers le plus petit. Mais représenter le poids des arcs apporte une information sur l'implication des itemsets dans ceux-ci. Dans notre étude, la largeur de la ligne et les autres variables visuelles sont utilisées, non pas pour exprimer le poids, mais pour exprimer les mesures interpolées des itemsets. Pour exprimer le poids des arcs, nous utilisons le blending.

Le blending d'un graphe a été étudié par Holten et al. [Holten 06, Holten 09]. Leur but est de mettre en valeur les petites arcs en atténuant les longs arcs auxquels ils peuvent être confondus. Cela s'avère utile pour détecter des arcs individuels ou des sous-arcs ayant un effet de bundling plus atténué. D'autres études, comme [Lambert 10a, Telea 10, Lambert 10b, Hurter 12], utilisent le blending ou le shading pour particulariser des informations du graphe. Le blending de couleur et d'alpha aide à mettre en valeur l'effet d'accumulation dû à l'algorithme de bundling. En effet, tous les arcs initiaux sont gardés par l'algorithme que nous avons utilisé, mais sont courbés pour obtenir des chemins qui les agrègent. Le blending aide alors à renforcer l'accumulation de ces arcs, et, par voie de conséquence, à réduire les arcs où se produit peu d'accumulation, et donc où se trouvent peu d'arcs agrégés. Ainsi, l'accumulation est liée au support, parce que plus celui-ci est élevé, plus la probabilité d'avoir un partage d'information est élevée. Ainsi, grâce à l'optimisation de placement des itemsets, le partage d'informations est plus probable. Cela aboutit donc à une nouvelle manière de mettre en valeur des zones du graphe où le support est le plus élevé.

9.5.5 Sélection

La dernière fonctionnalité du processus d'amélioration du graphe circulaire est la sélection d'itemsets et d'arcs. Elle permet de particulariser les arcs ascendants et descendants, à partir d'un ou plusieurs itemsets ou arcs, selon un mécanisme de propagation. A partir de la sélection, tous les supersets et les sous-sets associés à celle-ci sont particularisés par les arcs, tandis que les autres sont cachés. Le but est de pouvoir se focaliser sur les itemsets concernés par cette sélection, ainsi que sur ceux qui leur sont liés. La propagation aide à montrer quels sont les itemsets qui partagent la même information que ceux de la sélection. En la prolongeant jusqu'aux 1-itemsets, cela revient à montrer les attributs des données concernés par la sélection, et ainsi quels sont ceux qui sont mis en œuvre pour construire les itemsets communs à ceux de cette sélection. Ainsi, le partage d'information est particularisé. L'opérateur de sélection est un moyen de montrer l'efficacité du processus d'optimisation. En effet, il met en valeur la proximité des itemsets, en valorisant les parties du graphe qui sont dans la même zone. Cela permet alors de faire ressortir la structure du graphe et, par conséquent, la structure hiérarchique des itemsets.

Chapitre 10

Représentation des règles

Cette partie, qui a fait l'objet d'une publication à la conférence IHM 2011 [Bothorel 11], présente l'exploration de règles d'association, dans un contexte de Visual Analytics, selon lequel l'utilisateur joue un rôle central et décisionnel dans le processus. Pour cela, il doit être à même de configurer la visualisation d'un grand nombre de règles, en fonction de sa problématique, et de pouvoir les explorer. Le type de représentation que nous proposons, basé sur deux outils interconnectés, permet de les appréhender de manière globale, et de manière détaillée, sachant qu'elles peuvent être très nombreuses. Des outils de filtrage et de sélection invitent à une exploration par raffinement successif, en reprenant le principe du *Rules focusing* [Blanchard 07] (Cf. Chapitre 3.2.2). De plus, le nombre de mesures de qualité, proposées à l'utilisateur, étant éventuellement également conséquent, il peut choisir celles qui lui sont le plus adaptées.

10.1 Visualisation des règles

10.1.1 Scatter plot

La représentation globale des règles est réalisée à l'aide d'une technique de scatter plot, bi ou tridimensionnelle, en fonction du choix de l'utilisateur. Chacune est représentée par un point, dont les variables visuelles correspondent à une assignation de mesures de qualité (Cf. Chapitre 1.5.3), cela par caractérisation de la visualisation (Cf. Chapitre 2.3). Cette approche offre une vision simultanée d'un très grand nombre de règles, auxquelles sont associées plusieurs métriques, choisies par l'utilisateur. Son étude peut l'amener à se pencher plus spécifiquement sur les contre-exemples. Dans ce cas, les mesures de Loevinger [Loevinger 47] et de Conviction [Brin 97b] pourront être utilisées. La figure 10.1 (a) illustre ce principe d'assignation. La position des points, leur taille, leur couleur et leur transparence permettent de visualiser six mesures différentes. De plus, en intégrant les prémisses et les conclusions dans la liste des attributs, des sélections sur les itemsets sont envisageables, pour étudier les règles concernant un sous-ensemble des données.

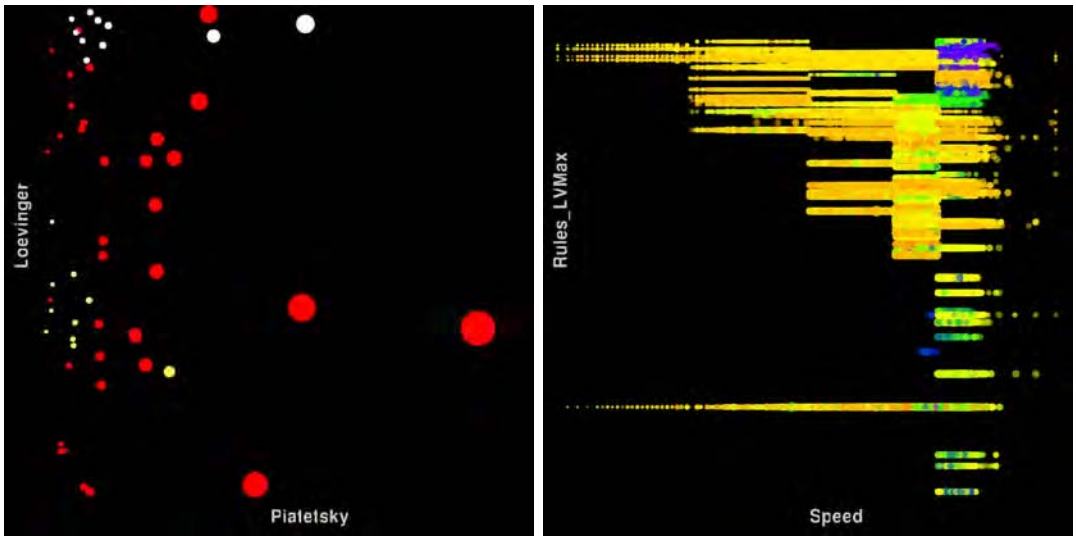
Donnée				Perception automatique							Perception contrôlée
Attribut	D	F	D'	X	Y	Z	T	R	-	[ ]	CP
X	Q	f	Q	P							
Y	Q	f	Q		P						
Mesure 1	Q	f	Q			P					
Confiance	Q	f	Q					C			
Support	Q	f	Q					S			

TABLE 10.1 – Assignation, aux variables visuelles, d'attributs et de mesures associées aux règles d'association.

Dans le chapitre 4.6 a été abordée la possibilité d'enrichir la visualisation des données à partir des résultats algorithmiques. En assignant des mesures de qualité aux attributs des données, il est alors possible d'étudier, en une vision commune et simultanée, les caractéristiques des règles et les données. Par exemple, dans le tableau 10.1, les attributs X et Y sont assignés aux variables de positions, tandis que trois mesures sont assignées à Z et aux variables rétiniennes, que sont la couleur et la taille des points. Les systèmes, que nous avons abordés au chapitre 3.2.2, mettent généralement en avant le support et la confiance, car le premier sert à déterminer les itemsets fréquents, tandis que le second sert à filtrer l'extraction des règles, en fixant une valeur seuil. Avec ce tableau, les points de plus fort support apparaissent plus gros, et ceux de plus forte confiance ont une valeur de couleur plus élevée, en considérant qu'un gradient est utilisé. Dans le cadre d'une représentation tridimensionnelle, Z permet de représenter une troisième mesure. La figure 10.1 (b) fournit un autre exemple de cette cohabitation, dans lequel un attribut est assigné à X, et une mesure à Y. Dans ce cas, il s'agit de la valeur Loevinger maximale. En effet, un point pouvant être concerné par plusieurs règles, il est nécessaire de les quantifier en une seule valeur pour apparaître sur cette figure. C'est pourquoi, pour chaque mesure de qualité, nous envisageons les valeurs minimale, moyenne et maximale.

10.1.2 Listes

De même qu'il est intéressant d'avoir une visualisation globale des règles, qui peut être affinée par des outils d'exploration, leur explicitation en détaille le contenu. Pour cela, les règles sont représentées également sous la forme d'une liste, comme illustré par la figure 10.2 (Cf. Chapitre 3.2.2). Chaque ligne indique l'intitulé exact d'une règle, par sa prémisse et sa conclusion. Cette liste est liée par linking à la visualisation scatter plot, ce qui assure un lien interactif entre les deux modes durant l'exploration. Sur la figure, les valeurs des mesures, associées à une règle, sont affichées. Dans cet exemple, il s'agit du support, de la confiance et du lift. Cela renforce l'exhaustivité de la connaissance de la règle.



(a) (b)

FIGURE 10.1 – Exemples d'affectation aux variables visuelles des mesures liées aux règles et des données.

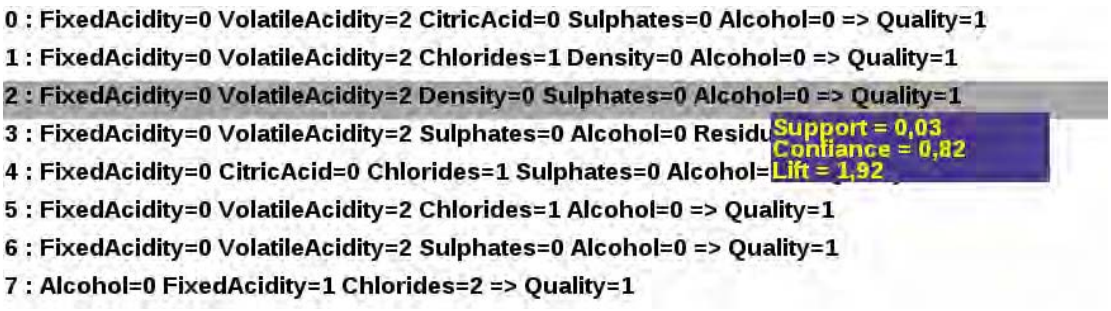


FIGURE 10.2 – Visualisation explicite des règles d'association sous forme de liste.

10.2 Exploration

L'exploration des règles est réalisée à l'aide de la visualisation scatter plot, selon les mêmes techniques que celles qui permettent d'explorer les données, et qui ont été formalisées dans les chapitres 4.1 et 4.2. Selon les assignations, la visualisation peut être en 2D ou en 3D. Des outils de zoom, d'excentrement et de filtrage sont utilisés, ce dernier étant appliqué aux mesures ou aux attributs affectés aux variables visuelles. Il est ainsi possible de n'afficher que les règles dont une ou plusieurs mesures sont inférieures ou supérieures à des seuils définis par l'utilisateur, voire comprises entre deux valeurs. Les règles sont sélectionnables par une technique de brushing, en déplaçant un disque centré sur le curseur de la souris, ou par délimitation d'une zone rectangulaire. De plus, en associant une variable visuelle avec les prémisses ou les conclusions, la sélection peut être affinée selon leur contenu. Par exemple, grâce à cette fonctionnalité, l'utilisateur peut ne garder, dans sa sélection, que les règles dont la prémisse contient l'attribut A_1 et l'attribut A_2. Les sélections permettent un affinement successif de l'exploration, en ne gardant que certaines règles, et en poursuivant leur exploration. Elles sont répercutées sur l'autre outil, pour assurer un lien entre les représentations, ce qui facilite l'exploration des règles, indifféremment sous les deux formes, tout en gardant une vision globale et détaillée.

10.3 Intérêt de cette approche

Notre approche consiste à présenter les règles d'association de deux manières. D'une part, la visualisation scatter plot offre une vision globale des règles, assistée par des opérateurs de paramétrage et de sélection. D'autre part, une représentation sous la forme d'une liste permet d'obtenir un intitulé exact des règles, et leurs caractéristiques [Hogan 04]. Cette double visualisation procure plusieurs avantages que nous allons évoquer ci-dessous.

L'assignation, aux variables visuelles, des données et des caractéristiques des règles d'association, permet de pallier l'inconvénient de la prédétermination de la visualisation, comme c'est généralement le cas des outils d'exploration de règles (Cf. Chapitre 3.2.2). En fonction de ses besoins, l'utilisateur peut configurer des visualisations de plusieurs types, notamment par le choix des mesures de qualités. La cohabitation, sur une même visualisation, de ces mesures et des attributs, permet de focaliser l'étude sur certains d'entre eux, en étudiant leur implication dans les résultats algorithmiques.

Cette méthode n'impose aucune limitation dans la structure des règles, car l'outil est compatible avec celles de types *one-to-one* [Wong 99b], *many-to-one* et toute autre forme. En effet, la visualisation scatter plot ne prend pas en compte la structure, et la largeur de la visualisation textuelle s'adapte à la taille des règles.

Les visualisations offrent une interactivité, grâce aux différents opérateurs facilitant l'exploration et le Rules Focusing [Blanchard 03, Blanchard 07]. D'une approche globale des règles, il est ainsi possible d'orienter l'étude, par affinement successif, d'une représentation à une autre.

La représentation scatter plot permet de représenter un très grand nombre de règles, et n'est pas limitée à quelques centaines [Blanchard 07, Blanchard 03] ou quelques milliers [Bruzzese 04]. Cela n'empêche pas, grâce à la représentation textuelle, d'avoir accès à leur intitulé exact plus explicite et abordable qu'une forme codée [Ertek 06]. Par la mise en œuvre de l'accumulation (Cf. Chapitre 3.1.4), il est possible d'étudier la densité de règles en fonction de critères choisis. Par exemple, en associant le support à la variable X et la confiance à Y, la visualisation reprend le concept de TwoKey Plot [Unwin 01].

Notre approche d'exploration de règles d'association s'inscrit dans une problématique de Visual Analytics, mettant en avant le rôle fondamental de l'utilisateur, ainsi que le niveau de paramétrage, grâce aux affectations et aux outils d'exploration et de sélection. Elle établit le lien entre les modèles et la visualisation (Cf. Figure 1.4), renforcé par l'intervention de l'opérateur humain, et montre que ce dernier garde un rôle décisionnel et central dans le processus de fouille de règles.

Chapitre 11

Conclusion sur la visualisation des résultats de la fouille de données

Cette partie a présenté des techniques de visualisation des résultats d'algorithmes de fouille de données. Les itemsets et les liens entre eux sont disposés sur un graphe constitué de plusieurs cercles concentriques, dont chacun d'eux contient des itemsets de même cardinal. La disposition des itemsets est optimisée de façon à ce que ceux qui partagent de l'information soient rapprochés, tandis que ceux qui n'en partagent pas sont éloignés. Grâce au LinLog et à une distance angulaire, ainsi qu'au recuit simulé, le graphe global est optimisé, ce qui en améliore l'interprétation et l'exploration. Cette représentation, qui a fait l'objet d'une expérimentation, est enrichie par la mise en œuvre de techniques, telles que le bundling, l'assignation de mesures associées aux itemsets et aux règles aux variables visuelles du graphe, ainsi que des outils de sélection.

La représentation des règles est réalisée sous forme de scatter plot et de listes, ce qui permet une exploration à la fois globale d'une grande quantité de règles, ainsi que locale d'un sous-ensemble de celles-ci. De plus, le contenu explicite ainsi que les mesures de qualités qui les caractérisent sont également accessibles. Une approche à plusieurs niveaux des règles est ainsi proposée.

Ces techniques d'exploration sont à mettre en regard avec les outils de visualisation des

itemsets et des règles d'association qui ont été évoqués dans l'état de l'art (Cf. partie 1). Le graphe circulaire permet de visualiser plus d'itemsets que dans des outils comme les coordonnées parallèles [Yang 03] qui nécessite de représenter le même 1-itemset plusieurs fois. La mise en évidence des itemsets reliés à une sélection a été étudiée par Ertek & Demiriz [Ertek 06]. La seule représentation circulaire que nous avons trouvée est FP-Viz [Keim 05] qui est une visualisation hiérarchique radiale, dans laquelle chaque secteur angulaire représente un itemset. Dans cette étude, Keim et al. assignent le support à la couleur, alors que nous proposons une plus grande variété dans l'utilisation des variables visuelles. Notre représentation des règles d'association n'est pas novatrice, car elle existe déjà dans des outils comme WebSphere Commerce Analyzer ou IRSetNav [Hogan 04]. Notre représentation visuelle des règles est multidimensionnelle, en proposant la visualisation de plusieurs mesures de qualité. Dans sa représentation matricielle tridimensionnelle [Wong 99b] Wong et al. visualisent uniquement le support, la confiance et le structure de la règle. Avec ARVIS [Blanchard 03, Blanchard 07], seules trois mesures sont visualisées. Notre approche est hybride, même si la représentation des règles est réalisée sur deux supports différents. Ce qui lui confère une unité est le lien que nous établissons entre ces deux outils, qui permet d'avoir des points de vue différents et configurables des règles. La visualisation scatter plot procure une vision d'ensemble de celles-ci, tandis que la liste en donne des détails. Le lien est réalisé par les outils d'interaction et le linking qui assure une cohérence entre les deux représentations.

Sixième partie

Application aux données aéronautiques

Introduction

Cette partie présente une mise en œuvre anthropocentrée des liens entre la fouille visuelle des données et l'approche algorithmique, Pour cela, une plate-forme a été développée pour permettre l'exploration des données et des résultats d'algorithmes. Elle repose sur une architecture modulaire permettant d'instancier les briques nécessaires à l'étude des données, qu'elles soient IHM ou algorithmiques. La plate-forme, répondant aux problématiques du Visual Analytics (Cf. Chapitre 1.4), s'appuie sur l'approche algorithmique (Cf. Chapitre 1.5), l'approche théorique de la visualisation (Cf. Chapitre 2), l'exploration visuelle des données (Cf. Chapitre 3.1), la visualisation des itemsets (Cf. Chapitre 3.2.1) et des règles d'association (Cf. Chapitre 3.2.2) et l'approche mixte (Cf. Chapitre 3.3).

Pour présenter notre démarche, nous aborderons les points suivants :

- le prétraitement des données ;
- l'exploration visuelle des données ;
- les choix de l'utilisateur pour piloter les algorithmes de fouille de données ;
- l'exploration visuelle des résultats des algorithmes ;
- l'interaction entre les explorations dans l'espace des données et l'espace des résultats ;
- l'analyse des résultats algorithmiques ;
- la réitération du processus global.

Le contexte d'application est l'exploitation de données aéronautiques issues des archives des systèmes français de gestion du trafic aérien. Dans un premier temps, afin de mieux les comprendre, nous présenterons la vie d'un vol dans les systèmes de navigation aérienne, en étendant ce concept au contexte international de modernisation des systèmes. Puis, nous étudierons les deux scénarios suivants :

- Scénario 1 : nous utilisons des données d'archivage des plans de vol. Afin d'extraire des informations générales sur les vols, nous pilotons l'algorithme de fouille de données à partir de la visualisation (Cf. Chapitre 4). Puis nous enrichissons la visualisation des données à l'aide des mesures caractérisant les règles d'association ;
- Scénario 2 : nous exploitons des données d'archivages des trajectoires des avions. En explorant les résultats des algorithmes, la visualisation initiale est automatiquement configurée à partir d'une sélection de règles, puis à partir de clusters de règles (Cf. Chapitre 5).

Le rôle de l'utilisateur sera mis en avant pour montrer son caractère central dans les choix d'exploitation de la plate-forme, tout en laissant une place noble à l'algorithme à plusieurs moments du processus d'extraction de connaissance.

Chapitre 12

La plate-forme VIDEAM (VIsual DrivEn dAta Miner)

12.1 Principe général et architecture de la plate-forme

La vision globale de la fouille de données, illustrée schématiquement par la figure 1 (Cf. Introduction), peut être décomposée en trois espaces :

- L'espace des données. Il est exploré par une fouille visuelle ;
- L'espace des itemsets et des règles d'association, qui, étant le résultat des algorithmes, sont une autre forme de données. Il est également exploré par une fouille visuelle ;
- Ces deux espaces sont reliés par des algorithmes. Ceux-ci permettent, à partir des données d'un espace, de produire de nouvelles données qui sont exploitées par l'autre espace. Ainsi, l'algorithme de fouille de données produit des itemsets et des règles, qui, à leur tour, produisent des attributs et une configuration de la représentation initiale des données.

Nous avons reproduit ce schéma global sous la forme de l'architecture d'une plate-forme de fouilles de données (Cf. Figure 12.1) que nous avons appelée Videam (VIsual DrivEn dAta Miner), pour signifier que les données et les règles sont le résultat d'algorithmes et que la fouille dans les deux espaces est un processus anthropocentré. La figure 12.2 présente cette plate-forme.

Au lieu d'avoir une application intégrée reproduisant cette architecture, nous avons développé plusieurs applications réalisant chaque brique. L'intérêt de cette approche est multiple :

- Les espaces des données et des règles peuvent être visualisés simultanément sur plusieurs écrans, ce qui permet d'en avoir une vision globale ;
- Afin d'avoir une représentation plein écran de chacun des espaces, leurs outils de paramétrage peuvent être affichés sur d'autres écrans ;

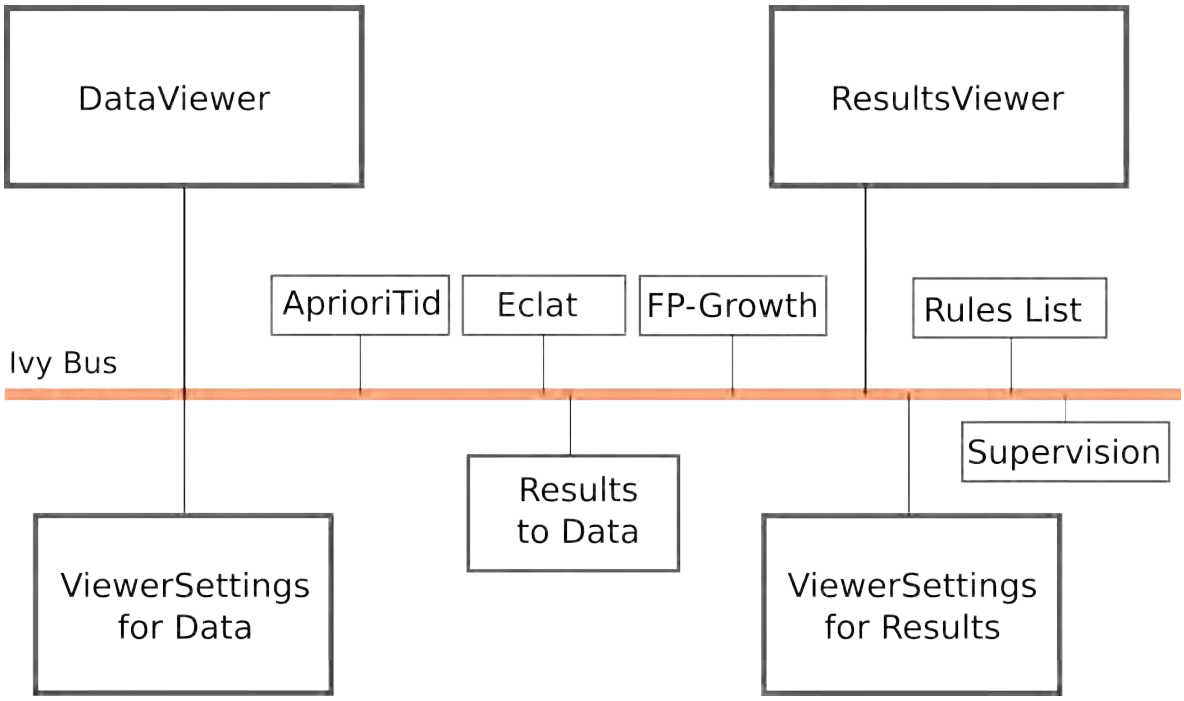


FIGURE 12.1 – Architecture de la plate-forme Videam.

FIGURE 12.2 – La plate-forme Videam.

- Les calculs peuvent être répartis sur plusieurs calculateurs, ce qui améliore les performances du système ;
- Le protocole de communication mis en œuvre, appelé bus Ivy, permet la cohabitation de plusieurs langages de programmation sur la plate-forme, en assurant les échanges entre les processus ;
- La simplicité d'utilisation du bus Ivy permet de remplacer une brique par une autre, sans incidence sur le comportement global de la plate-forme. Ainsi, si l'utilisateur souhaite mettre en œuvre un autre algorithme, en le programmant éventuellement dans un autre langage plus adapté, il suffit de ne pas utiliser ceux qui existent actuellement, mais d'exploiter cette autre brique, sans que cela ait une incidence sur les autres processus de Videam. Il est cependant nécessaire que le format des messages échangés soit le même ;
- Il est possible de n'exécuter qu'une partie de la plate-forme. Ainsi, si par exemple un utilisateur souhaite faire de la fouille de données non-algorithmique, il lui suffit de ne lancer que l'application de visualisation des données et l'application de réglage.

12.2 Le bus logiciel Ivy

Le bus logiciel Ivy [Buisson 02] [1] a été développé par le Centre d'Etude de la Navigation Aérienne [Poirot-Delpech 95] [2] dans les années 1990, et est sous licence LGPL [3]. Il permet, à des applications programmées dans différents langages, tels que C, C++, Java, Perl, C#, Python et OCaml, de dialoguer entre elles, et cela quelle que soit la plate-forme d'exécution des logiciels. En effet, Ivy est utilisé dans des environnements Linux, Mac OS et Windows. Il permet le développement et l'interconnexion d'agents sans serveur centralisé et sans annuaire de routage. Il est beaucoup utilisé dans le domaine du prototypage de systèmes interactifs, notamment dans le domaine de la recherche appliquée aux problématiques de contrôle aérien.

Ivy est un protocole de communication très simple à mettre en œuvre, basé sur un mécanisme de diffusion de messages sous forme textuelle. Pour utiliser ce protocole, un agent se déclare sur le bus, avec un nom. Il s'abonne à des messages selon un formalisme d'expressions régulières, ou *regexp*, qui permettent de décrire, sous la forme d'un motif, des chaînes de caractères. De plus, cet agent peut envoyer des messages textuels sur le bus. Si un message correspondant est envoyé sur le bus, tous les agents s'étant abonnés à ce message, auront une fonction qui sera appelée et qui pourra ainsi traiter le message reçu. Les agents n'ayant pas d'abonnement correspondant à ce message ne réagissent donc pas.

La mise en œuvre est réalisée en cinq étapes :

- Déclaration de l'agent sur le bus, par exemple par l'instanciation de la classe Ivy ;
- Abonnement à des messages, par une fonction *bind* ;
- Connexion au bus, par une fonction *start* ;

1. http ://www.eei.cena.fr/products/ivy
2. http ://fr.wikipedia.org/wiki/Centre_d'études_de_la_navigation_aérienne
3. http ://www.gnu.org/licenses/lgpl.html

- Réception de message et appel de fonction de type *callback* ;
- Fermeture de la connexion par une fonction *stop*.

Ainsi, l'utilisation du bus logiciel dans la plate-forme Videam permet aux processus de dialoguer, de ne pas être tous utilisés et d'être interchangeables.

12.3 Visualisation des données : DataViewer

L'application DataViewer est inspirée de FromDaDy [Hurter 09] et en reprend les principes généraux (Cf. Chapitre 3.1.4). Elle représente les données sous la forme d'un scatter plot en s'appuyant sur la sémiologie graphique [Bertin 67] (Cf. Chapitre 2.2) et sur le modèle de caractérisation des visualisations de Card & Mackinlay [Card 97] (Cf. Chapitre 2.3). Elle permet l'association des attributs des données avec les variables visuelles, telles que la position, la couleur, l'alpha et la taille du point. Il est également possible de relier les points par des lignes. Pour cela, la détermination d'un attribut permet de relier entre eux les points partageant la même valeur de cet attribut. La largeur des lignes varie alors en fonction de la taille des points extrêmes des segments.

Les données sont hétérogènes, car elles sont de différents types : numérique entier, numérique décimal et alphanumérique. Il s'agit d'une nécessité dans le domaine des données aéronautiques, car cette cohabitation des types est omniprésente dans les systèmes d'archivage.

12.3.1 Visualisation des données

DataViewer est mono-vue, en ce sens qu'il visualise les données dans une seule fenêtre, qui peut être retaillée ou affichée plein écran. La visualisation est par défaut tridimensionnelle, avec des projections orthographiques, ou en perspective (Cf. Figure 12.3). En projection orthographique, si deux des axes sont dans le plan de l'écran, alors cela revient à une visualisation bidimensionnelle. Ainsi, DataViewer permet de représenter les données en 2D ou en 3D. Le choix d'une projection sur un des plans défini par deux axes est réalisé au clavier, sachant que chaque changement de projection est réalisé par une animation, ce qui permet de garder une continuité dans les changements de visualisation, et ainsi des transitions fluides. La rotation de la vue est réalisée autour des axes, par une combinaison de touches clavier et de manipulation à la souris. Le zoom, centré sur la position du curseur, ainsi que l'excentrement de l'image, sont réalisés à la souris.

Les couleurs sont rendues à l'écran selon différentes modalités. Tout d'abord, la couleur correspond simplement à sa valeur issue d'une palette, codée selon les composantes RGB. Cependant, chaque composante R, G et B, peut aussi être affectée à un attribut. Cela permet ainsi de combiner trois attributs dans un seul rendu de couleur. La couleur peut également être affectée en fixant manuellement une valeur pour chacune des composantes. Une technique de Color Blending est alors utilisée. Si des points se recouvrent, chaque composante de la couleur est alors cumulée, et la couleur résultante est la combinaison de ces trois cumuls. Considérons

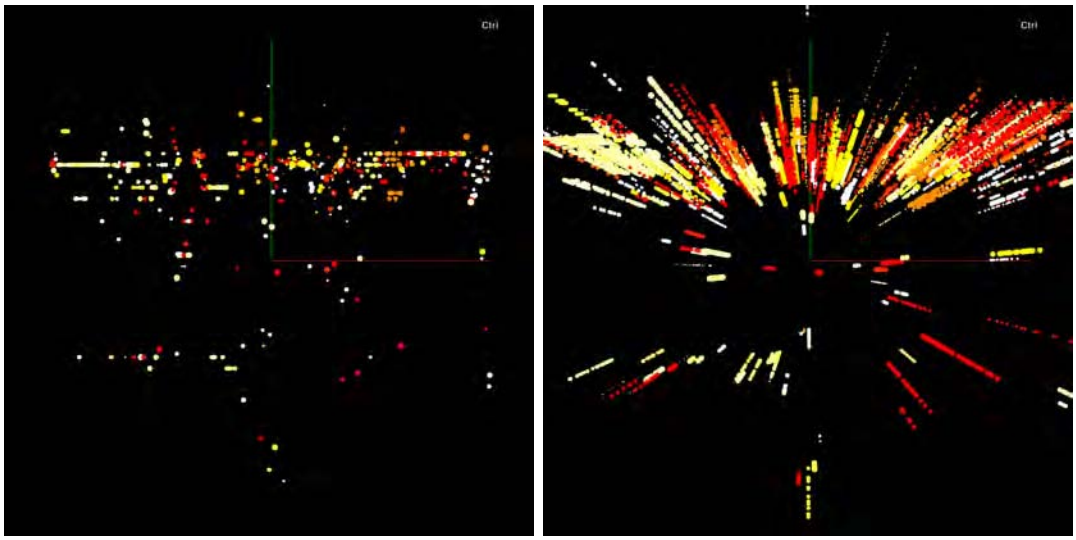


FIGURE 12.3 – Projection orthographique et perspective.

un point rouge dont la couleur RGB est égale à (0.8, 0.1, 0.0). En le superposant dix fois, la couleur résultante du point est (1.0, 1.0, 0.0), c'est-à-dire jaune. Ce concept est exploité pour représenter l'accumulation, qui sera exposée plus loin (Cf. Chapitre 12.3.3).

En plus de la couleur, l'alpha, qui correspond à l'opacité, peut être affecté à un attribut. Dans ce cas, une technique d'Alpha Blending est mise en œuvre. Elle permet, par simulation de la transparence, de composer un objet graphique avec le fond de l'image, ou un objet situé en arrière-plan [**?**].

12.3.2 Sélection des données

Elle est réalisée selon deux sortes d'opérateurs de type brushing (Cf. Chapitre 3.1.3). D'une part les données peuvent être sélectionnées par leur survol à l'aide d'un disque centré sur le curseur de la souris. D'autre part, la zone de sélection est délimitée par un rectangle, en déplaçant le curseur d'un coin de celui-ci vers le coin opposé. L'avantage de la première technique réside dans la grande souplesse de sélection, par un déplacement libre du curseur. Cependant, pour sélectionner des données contenues dans une plage de valeurs, cela peut s'avérer délicat. C'est pourquoi, nous utilisons également la seconde technique. Les données peuvent être également désélectionnées en utilisant les deux modes de brushing et une touche clavier. Enfin, il est possible de sélectionner ou désélectionner l'ensemble des données.

Trois modes d'affichage des sélections sont proposés et illustrés par la figure 12.4. Dans un premier temps, les données sélectionnées sont particularisées. La vue (a) les présente en jaune. Puis, sur la vue (b), seuls les points sélectionnés sont visualisés. Enfin, la vue (c) ne montre que les points non sélectionnés. Quel que soit le mode de représentation d'une vue faisant l'objet d'une sélection, il est toujours possible d'avoir accès à la vue initiale, afin de garder une vision globale des données. Par ailleurs, toute sélection ou désélection modifie

(a) (b) (c)

FIGURE 12.4 – Trois modes de visualisation avec sélection.

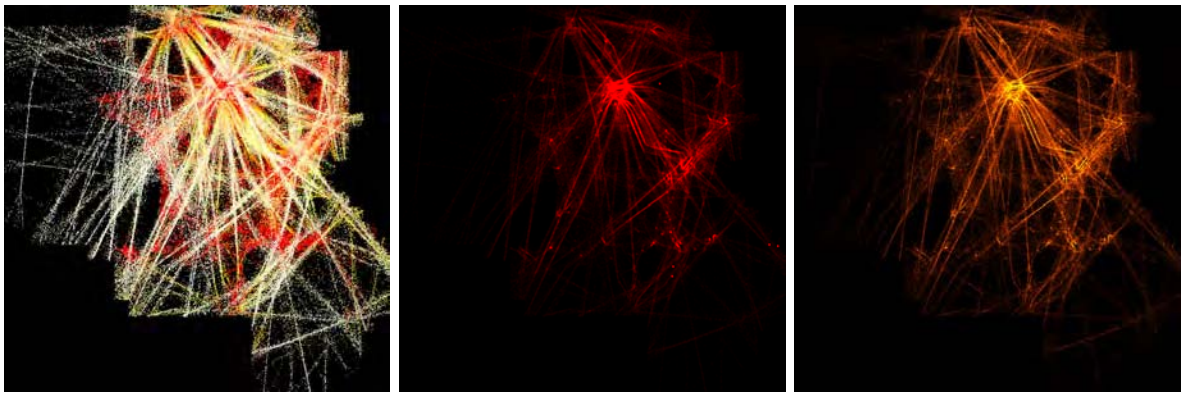
automatiquement la vue, en fonction de son mode de représentation. La couleur par défaut de la sélection est jaune, mais, en cas de risque de confusion, elle est modifiable à l'aide de la molette de la souris.

12.3.3 L'accumulation

DataViewer met en œuvre deux modes d'accumulation : à l'aide des cartes d'accumulation et par la technique de Color Blending évoquée plus haut dans le chapitre 12.3.1. Les cartes d'accumulation sont réalisées à l'aide de la technique *Kernel Density Estimation* [Silverman 86], qui a été exposée au chapitre 3.1.4. Dans les zones de forte densité de points, elle apporte de l'information visuelle sur la quantité d'accumulation. Les kernels sont dessinés dans des tons de rouge dans une texture, la valeur de la couleur décroissant du centre vers le bord du disque. A l'aide de la molette de la souris et du clavier, le diamètre de chaque kernel est réglable par l'utilisateur, ainsi que la valeur maximale du rouge. Le second mode est le Color Blending, qui, après un choix manuel des composantes R, G et B, accumule celles-ci en fonction de la superposition des points.

La figure 12.5 illustre l'accumulation appliquée à la représentation d'une journée de trafic contrôlé par les organismes français. Les positions successives des avions sont représentées sur la vue (a), sans accumulation, selon un code couleur correspondant à l'altitude. Les avions volant bas sont représentés en rouge, et ceux qui volent le plus haut sont en blanc. Sur la vue (b), la carte d'accumulation fait ressortir, par des points rouges une forte densité de trafic en région parisienne. Plus le trafic est dense, plus la largeur du tracé est élevée. Pour cette vue, la technique *Kernel Density Estimation* a été utilisée. Sur la vue (c), l'utilisateur a choisi une valeur de rouge sensiblement plus élevée que celle de la composante verte. La composante bleue est nulle. Le résultat montre que les zones de trafic les plus élevées tendent vers la couleur jaune, qui correspond à l'addition du rouge et du vert.

L'avantage des cartes d'accumulation est dans le kernel, par sa forme en bosse, qui



(a) (b) (c)

FIGURE 12.5 – Accumulation d'une journée de trafic au-dessus de la France. (a) : visualisation du trafic, sans accumulation. Grâce à l'accumulation, les zones à fortes densités sont particularisées, par des zones rouges au tracé plus large (*Kernel Density Estimation*) (b), ou par une couleur évoluant du rouge vers le jaune (c).

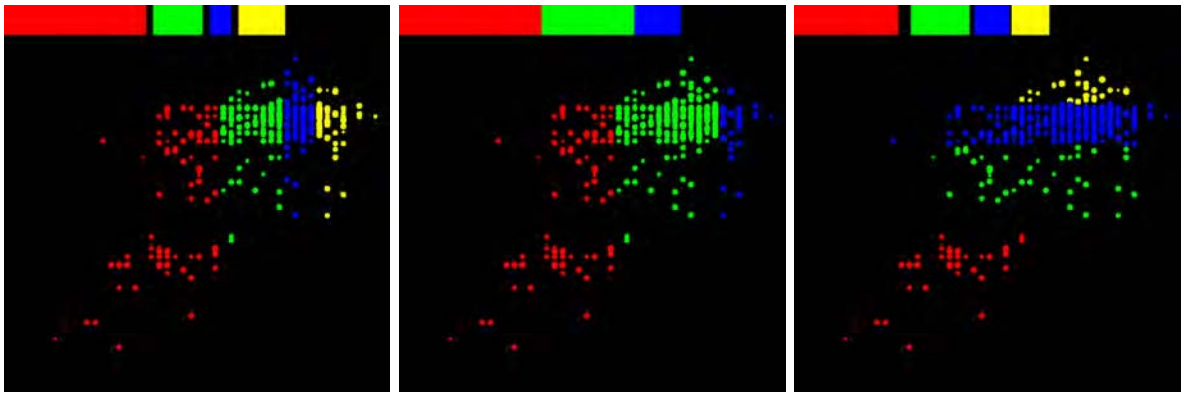
donne un effet de lissage dans l'image accumulée. Le Color Blending permet, quant à lui, de travailler directement sur l'image utilisée pour l'exploration visuelle des données. La largeur de l'accumulation est celle du point qui correspond à un kernel d'épaisseur nulle. Cela permet d'avoir un tracé plus fin, ce qui peut être, selon le cas, plus intéressant que le lissage dû au kernel.

12.3.4 Pilotage de l'algorithme de Data Mining

A partir de l'exploration visuelle à l'aide de DataViewer, l'utilisateur définit l'espace des données qui vont être prises en compte par l'algorithme de Data Mining. Pour cela, il dispose d'outils de filtrage et de sélection.

L'algorithme k-means [MacQueen 67] est alors exécuté pour la discrétisation (Cf. Chapitre 4.3). Il définit des clusters en fonction des assignations de variables visuelles. Le résultat est alors visualisé sous forme de groupes de couleurs différentes, selon un ordre prédéfini. La figure 12.6 (a) montre quatre clusters calculés selon l'axe des abscisses. Le calcul algorithmique des clusters ne prend pas en compte la connaissance de la base de données. Comme l'utilisateur dispose de celle-ci, il lui est possible de redimensionner les clusters, avec la possibilité d'en supprimer. Pour cela, il peut afficher une zone, dans le coin supérieur gauche, représentant, sous forme de rectangles colorés, la répartition des clusters. Les espaces entre les zones colorées sont dus à l'absence de valeurs pour l'attribut de l'axe considéré. A l'aide de la souris, il est alors possible de modifier le dimensionnement de ces rectangles, ce qui a une répercussion sur le contenu des clusters correspondant à la dimension considérée. Sur la vue (b), les clusters vert et bleu sont fusionnés et deviennent le cluster vert. Le cluster bleu devient alors le cluster jaune. La vue (c) illustre un résultat de k-means pour l'axe des ordonnées.

Cette configuration des clusters est ensuite soumise à l'algorithme de Data Mining choisi



(a) (b) (c)

FIGURE 12.6 – Résultat de k-means [MacQueen 67] appliqué aux variables visuelles X et Y. Les clusters peuvent être redimensionnés par l'utilisateur, comme par exemple les quatre clusters de l'axe des X (a) qui sont ramenés à trois (b). (c) : clusters de l'axe des ordonnées.

par l'utilisateur (Cf. Chapitres 4.4 et 4.5).

12.3.5 Communication avec Ivy

L'exploration visuelle des données dans DataViewer est réalisée à l'aide de la souris et du clavier. Cependant, afin de rendre la configuration et les réglages de l'IHM pilotables par des résultats algorithmiques, la plupart des commandes peuvent être réalisées également à distance, grâce au bus logiciel Ivy. Cela procure une souplesse et un haut niveau de paramétrage dans la mise en œuvre d'un tel outil, au sein d'une plate-forme comme Videam.

12.3.6 Aspects techniques sur DataViewer

DataViewer a été réalisé en C# sous environnement Linux. La boîte à outils graphique mise en œuvre pour l'interfaçage avec la carte graphique est *OpenTk*[4]. Elle permet d'exploiter des fonctionnalités telles que les VBO (Vertex Buffer Object), grâce auxquels les vertex et leurs attributs sont chargés dans la carte graphique, et les FBO (Frame Buffer Object) dans lesquels sont réalisés les rendus de textures, par exemple pour les cartes d'accumulation ou le brushing. De plus, les calculs de rendus sont réalisés au niveau des Vertex Shader, Geometry Shader et Fragment Shader, pour agir sur les composantes des objets graphiques. Le Vertex Shader permet de gérer la taille des points, le Geometry Shader intervient sur l'épaisseur variable des lignes el les transformant en quadrilatères, et le Pixel Shader agit sur la couleur. Pour cela, le langage GLSL (OpenGL Shading Language) a été utilisé. Il assure le contrôle des shaders et la maîtrise du rendu graphique de l'image, en descendant jusqu'au rendu du pixel. DataViewer permet ainsi d'afficher plus de 6 millions de points représentant chacun six

4. http ://www.opentk.com

attributs, soit 36 millions de données [5].

12.4 Configuration de la visualisation des données : ViewerSettings

L'application DataViewer est pilotée par ViewerSettings. Celle-ci est positionnée librement par-dessus DataViewer, ou sur un autre écran. Les communications entre les deux applications sont assurées via le bus logiciel Ivy. ViewerSettings est illustrée par la figure 12.7. Ses principales caractéristiques sont les suivantes, en fonction des numéros :

1. L'utilisateur associe les attributs à une des variables visuelles suivantes : X, Y, Z, alpha, couleur et taille de points, sachant qu'un attribut peut être affecté à plusieurs variables visuelles. L'affectation d'un attribut au champ *Lines* permet, en mode lignes, de relier tous les attributs de même valeur par un trait. Dans ce mode, s'il n'est pas possible de relier des points, alors ils sont affichés sous forme de points. La lettre H à droite de deux ou trois attributs X, Y et Z, indique qu'ils sont homogènes, c'est-à-dire que l'échelle de représentation est la même pour ces attributs.
2. Un gradient de couleur peut être associé à un attribut numérique. Si l'attribut est une chaîne de caractères alphanumériques, alors en dessous de sept valeurs différentes, l'affectation de la couleur donnera lieu à des valeurs RGB discrètes ;
3. Il est possible d'affecter séparément chaque composante de couleur à un attribut. La couleur résultante est alors la composition de ces trois composantes ;
4. A l'aide de la molette de la souris, l'utilisateur peut choisir librement une valeur pour chaque composante de couleur. Par Color Blending, la couleur résultante des points est alors le bilan de l'accumulation ;
5. Toutes les valeurs peuvent faire l'objet d'un filtrage en déplaçant à la souris les valeurs minimales et maximales. Cela s'applique également aux valeurs de type alphanumérique, car, dans un contexte aéronautique, des données telles que des indicatifs de vol, ou des noms d'aéroport, sont partiellement ordonnées. Ainsi, tous les vols commençant par AF sont de la compagnie Air France. De même, le nom des aéroports est codifié selon quatre lettres, appelées indicateur d'emplacement [OACI 12]. Pour la France, ils commencent tous par LF. Par exemple, le code de l'aéroport de Toulouse est $LFBO$;
6. Afin de compléter la fonction de filtrage, il est possible de sélectionner une partie des valeurs d'un attribut alphanumérique. Pour cela, le système traite des expressions régulières [**?**], qui sont des chaînes de caractères permettant de décrire des chaînes variables. Ainsi, si un attribut correspond aux indicatifs de vol, pour sélectionner tous les vols de la compagnie Brit'Air, l'expression régulière est $\hat{}BZ$;
7. La liste des attributs est représentée sous forme de boutons, afin de les sélectionner ou désélectionner, pour configurer l'algorithme de fouille de données. Par défaut, les

5. Ce dimensionnement a été réalisé avec un processeur graphique NVidia GTX580.

attributs sont sélectionnés, et seuls les attributs alphanumériques de moins de sept items sont sélectionnés ;

8. Il est nécessaire d'affecter au moins les variables visuelles X et Y. Une fois que c'est fait, l'interface DataViewer peut être lancée à l'aide du bouton *DataViewer*. Après cela, il est toujours possible de modifier les affectations dans ViewerSettings, ce qui aura pour effet de reconfigurer DataViewer ;
9. L'utilisateur choisit le nombre maximal de clusters par dimension, sachant qu'il lui sera possible de redimensionner ou de supprimer des clusters calculés par l'algorithme ;
10. L'algorithme de Data Mining est choisi à l'aide d'un menu. Celui-ci est fonction d'un fichier de configuration qui s'appuie sur la présence d'algorithmes qui peuvent être exécutés. Il peut donc varier ;
11. Les valeurs seuil du support et de la confiance sont fixées par l'utilisateur ;
12. La couleur du fond de l'image de DataViewer est réglable par l'utilisateur ;
13. Le bouton *Open* sert à charger un fichier de données. Il peut être de plusieurs types :
    - Le format texte (*.txt*) est le format source des données, défini de manière simple : une ligne correspond à un tuple ou transaction, dont les attributs sont séparés par un espace. Comme le traitement de ce format peut prendre du temps en présence de beaucoup de données, un fichier binaire *.bin* est généré. Il contient les structures de données décrites dans le fichier source. ViewerSettings détermine lui-même les types des attributs, en fonction des données ;
    - Le format ARFF (Attribute-Relation File Format) (*.arff*) [Witten 11] est un autre format de type texte. Il contient, dans une première partie, la description et le type des attributs. Dans la seconde, il décrit les transactions. Ce format a été défini par le Machine Learning Project de l'Université de Waikato [6] pour le logiciel WEKA [7] [Witten 11]. A partir de ce fichier, un fichier binaire *.bin* est généré ;
    - La format des fichiers de configurations (*.gwe*) qui référencent un fichier *.bin* et contiennent la configuration de DataViewer (affectation attributs/variables visuelles, gradient de couleur, etc.) Quand l'utilisateur modifie les affectations des variables visuelles, le fichier *.gwe* est mis à jour par appui de la touche *Save*.

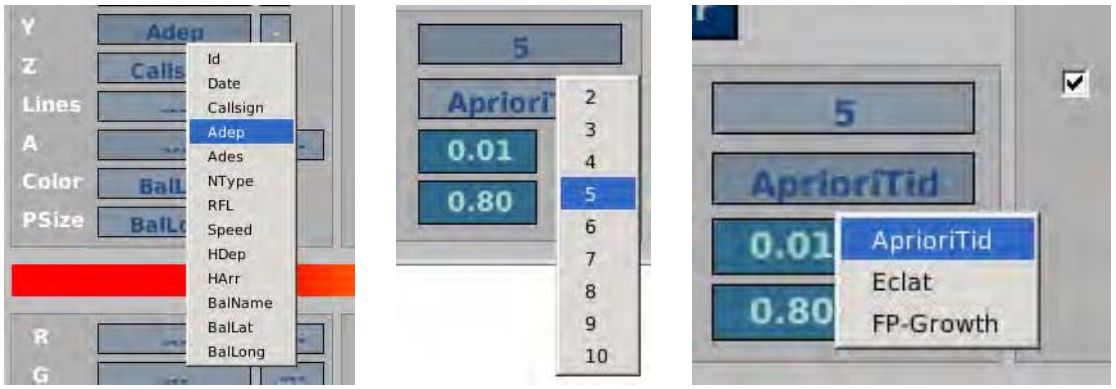
12.5 Visualisation des résultats d'algorithme : ResultsViewer

L'application ResultsViewer est une variante de DataViewer. L'utilisateur peut choisir l'un des trois types de visualisation (Cf. Partie V) :

- Visualisation des itemsets sous la forme de graphes circulaires présentés aux chapitres 7.1 et 9. Les arêtes reliant les itemsets sont des segments. Des ajouts ont été faits par rapport à DataViewer, pour représenter les itemsets et les sélectionner individuellement ;

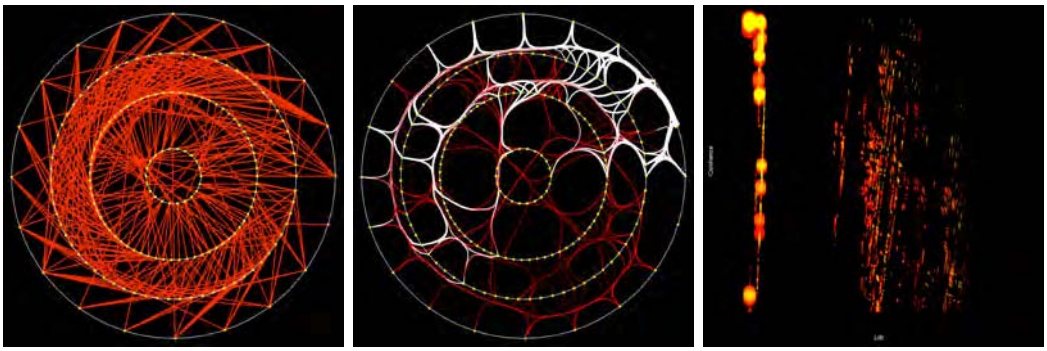
6. www.waikato.ac.nz
7. http ://www.cs.waikato.ac.nz/ml/weka

FIGURE 12.7 – L'application ViewerSettings pour paramétrer DataViewer.



(a) (b) (c)

FIGURE 12.8 – L'utilisateur paramètre le processus d'exploration à l'aide de ViewerSettings. Il affecte les attributs des données aux variables visuelles (a), choisit le nombre maximal de clusters générés par k-means (b), et sélectionne l'algorithme de Data Mining à exécuter (c).



(a) (b) (c)

FIGURE 12.9 – Trois modes d'affichage et d'exploration des résultats d'algorithmes. Graphe circulaire des itemsets, avec des segments (a), puis avec bundling (b) ; (c) : scatter plot des règles.

- Visualisation des itemsets sous la forme de graphes circulaires avec bundling. Les fonctionnalités ont été décrites au chapitre 9. Bien qu'il soit possible d'afficher les itemsets sous forme de scatter plot, le mode ligne est privilégié pour bien établir visuellement le lien entre eux. Les mesures associées aux itemsets et aux règles qui en découlent sont assignables aux variables visuelles à l'aide de ViewerSettings ;
- Visualisation des règles sous forme de scatter plot. Les attributs affectés aux variables visuelles sont les mesures associées aux règles.

12.6 Configuration de la visualisation des itemsets et des règles

L'application de pilotage de ResultsViewer est ViewerSettings présenté plus haut (Cf. Chapitre 12.4). Cependant, ses fonctionnalités sont adaptées aux itemsets et aux règles. La figure 12.10 illustre brièvement cette application en fonction des numéros :

1. L'utilisateur associe aux variables visuelles, non plus les attributs des données, mais les métriques associées aux résultats algorithmiques ;
2. Le nombre maximum de clusters de règles, calculé selon la méthode exposée dans le chapitre 5, est paramétrable ;
3. Il est possible de choisir si le nombre de clusters doit être exact, ou s'il doit être optimisé, en fonction du critère de Caliński-Harabasz [Caliński 74]. Dans ce second cas, il peut en résulter moins de clusters ;
4. Le chapitre 5 a exposé la configuration automatique de la visualisation des données en fonction des clusters de règles. Cette visualisation peut prendre en compte l'axe Z pour obtenir une vue tridimensionnelle. Les boutons 2D/3D permettent d'effectuer ce réglage.

Les autres modalités d'utilisation de cet outil ont été explicitées dans le chapitre 12.4.

FIGURE 12.10 – Une autre version de ViewerSettings pour paramétrer la visualisation des résultats d'algorithmes.

12.7 Intérêt de la plate-forme

Grâce à la mise en œuvre de Videam, nos travaux de recherche apportent des éléments de réponse aux recommandations de Shneiderman [Shneiderman 01], au sujet de la combinaison des approches visuelle et algorithmique (Cf. Chapitre 1.3). Elles concernent l'intégration du Data Mining et de la visualisation d'informations, le rôle central et décisionnaire de l'utilisateur, qui travaille dans un contexte collaboratif, et l'importance de la conception, notamment pour l'acceptabilité de l'outil de Data Mining.

Les fouilles visuelles et algorithmiques sont intégrées, par le lien étroit et collaboratif entre les visualisations et la mise en œuvre des algorithmes. L'exploration des données hétérogènes, leur sélection ainsi que le paramétrage de l'algorithme, servent à piloter ce dernier. En effet, l'utilisateur peut spécifier ce qu'il recherche, à l'aide des possibilités exploratoires de l'outil de visualisation des données. Pour cela, leurs attributs sont assignés aux variables visuelles, et des opérateurs de filtrage et de sélection sont utilisés. La visualisation devient alors un moyen de contrôler de processus et d'en faire partie, et non d'explorer simplement les données initiales ou les résultats des algorithmes. Cette coopération entre l'homme et le système s'inscrit dans le modèle *White Box* décrit par Bertini & Lalanne [Bertini 09] (Cf. Chapitre 3.1), dans le cadre de l'intégration de la visualisation et de la fouille algorithmique *(Integrated Visualization & Mining)*.

En retour, les caractéristiques des itemsets et des règles d'association sont exploitées pour enrichir et piloter la visualisation des données initiales, que ce soit par assignation des mesures aux variables visuelles, ou en paramétrant automatiquement le réglage de cette visualisation. La visualisation des données et celle des résultats algorithmiques sont ainsi également intégrées.

Le contexte social de l'utilisateur est pris en compte par l'aspect collaboratif que confère la plate-forme. En effet, l'exploration interactive des données et des résultats des algorithmes

est possible par la cohabitation des deux espaces de visualisation. De plus, le bus Ivy procure à ces espaces la possibilité d'être implantés dans des lieux distants. Il permet également de disposer d'un environnement hétérogène, que ce soit par les systèmes d'exploitation et par les langages de programmation utilisés.

Comme le système global est distribué, il est possible d'utiliser plusieurs algorithmes, à condition de respecter le formalisme d'échange entre les visualisations et les algorithmes de Videam. Ainsi, en fonction du besoin de l'utilisateur, il peut choisir celui qui est le plus adapté. Pour l'intégrer à la plate-forme, une brique logicielle est ajoutée, sans avoir à modifier le reste de Videam. Ce problème a été soulevé par Keim et al. [Keim 06] qui mentionnent la spécificité des outils, en fonction du domaine d'application. Ils préconisent la mise en œuvre d'interfaces pour permettre la communication entre différentes applications.

Il n'est pas nécessaire de lancer l'intégralité de la plate-forme, car elle peut être partiellement exploitée. En effet, il est possible, par exemple, de n'utiliser que l'outil de visualisation des données, sans exécuter les autres parties, ou de ne mettre en œuvre qu'un algorithme sans lancer les IHM de visualisation.

Dans les défis soulevés par le Visual Analytics [Keim 10a, Keim 06], dans le but de contrôler visuellement le Data Mining, l'accent est mis sur l'efficience des algorithmes et des traitements (Cf. Chapitre 1.4). Pour cela, la plate-forme effectue, selon le cas, les calculs avec le GPU, ou avec le CPU, en utilisant le multithreading, ce qui permet les calculs en parallèle. Selon le type de traitement, ils sont ainsi généralement effectués en quelques secondes. La seule exception est dans l'optimisation du placement des itemsets sur le graphe circulaire. Ainsi, l'exploration multi-spatiale des données, des itemsets et des règles est réalisée avec des temps de réponse rapides. L'effet immédiat d'une interaction, sur les différents outils, facilite alors l'étude des données et favorise le caractère itératif de celle-ci.

Enfin, Videam n'est pas conçue exclusivement pour l'étude des données aéronautiques. En effet, elle reste ouverte aux divers domaines d'applications et peut, le cas échéant, intégrer des briques spécifiques répondant aux besoins de ces domaines.

Chapitre 13

Les données aéronautiques

Afin de délimiter le périmètre de la définition d'une donnée aéronautique, nous considérons celles qui sont en lien avec le mouvement des avions. Celles-ci décrivent directement le vol, comme la trajectoire ou la vitesse, ont une influence sur celui-ci, comme les conditions météorologiques, ou sont la conséquence d'un vol, comme sa contribution à des calculs liés à l'utilisation d'une zone de l'espace aérien.

Pour étudier les données, nous allons, dans un premier temps, aborder la vie d'un vol, de la connaissance de celui-ci dans les systèmes de contrôle, jusqu'à son achèvement, en passant par la phase de vol proprement dite. Après avoir ensuite abordé leur archivage, nous évoquerons les futurs systèmes européens et leurs besoins en termes de données. Cela nous permettra de nous concentrer sur un élément clé, qui est la trajectoire, autour de laquelle se concentrent beaucoup d'enjeux.

13.1 La vie d'un vol dans les systèmes de navigation aérienne

Il existe deux types de règles pour voler dans l'espace aérien : les règles de vol à vue ou VFR (Visual Flight Rules) et les règles de vol aux instruments ou IFR (Instrument Flight Rules). Celles-ci sont fixées par l'Organisation de l'Aviation Civile Internationale [1] et donnent lieu à une réglementation, dont la version française est la Réglementation de la Circulation Aérienne, ou RCA [RCA 13]. Le principe de base du vol VFR repose sur l'expression « Voir et éviter ». Pour cela, des conditions doivent être respectées, notamment en lien avec la météorologie. Cela permet d'évoluer dans certaines zones de l'espace aérien, éventuellement sans contact radio avec les organismes de contrôle. Le vol VFR s'avère cependant être très contraignant dès qu'il s'agit de parcourir de longues distances, et est interdit dans certaines situations, comme la pratique du transport commercial de passagers. Les vols générant le plus de données sont réalisés aux instruments, et c'est cette règle de vol qui va maintenant être abordée dans cette étude.

1. http ://www.icao.int

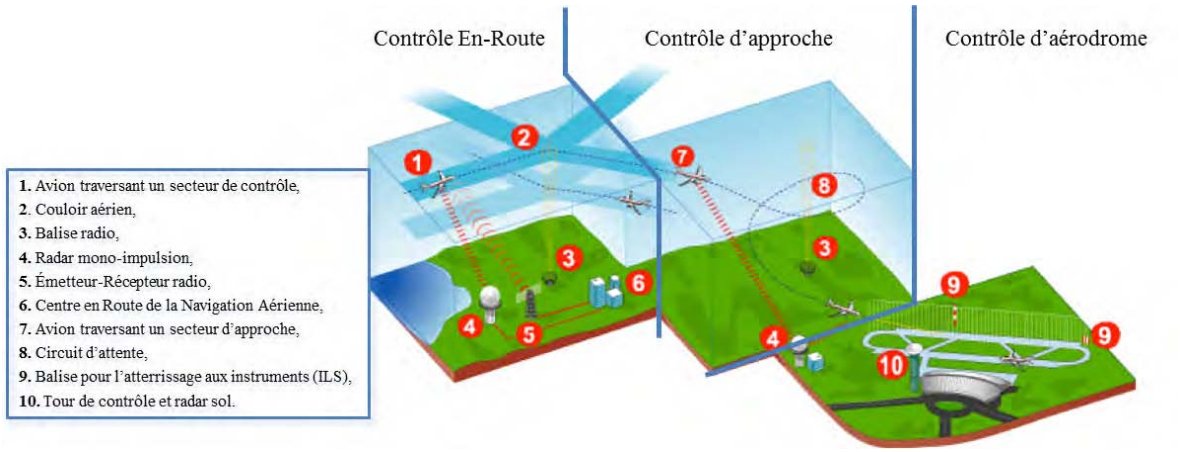


FIGURE 13.1 – Trois types de contrôle aérien.

Du décollage à l'atterrissage, l'avion volant en IFR parcourt différentes zones de l'espace aérien, sous la responsabilité de contrôleurs aériens, dont la spécialité dépend du type de zone. Il existe ainsi trois principaux services de contrôle schématisés par la figure 13.1. Ils peuvent être explicités en déroulant la vie du vol. Au départ, l'avion est en contact avec la tour, qui assure le contrôle d'aérodrome. Celui-ci gère les mouvements sur la plate-forme aéroportuaire, ainsi que les avions au décollage et à l'atterrissage. Après le décollage, l'avion est pris en charge par le contrôle d'approche qui s'occupe des arrivées et départ des aéroports. Il gère par exemple les flux à l'arrivée en assurant l'espacement des avions convergeant vers un même aéroport. Puis, l'avion passe sous la responsabilité du contrôle en route qui gère les phases de croisière. Plus tard, l'avion sera en contact avec l'approche de l'aéroport d'arrivée, puis avec la tour.

Selon le type de service, le contrôleur se trouve dans une tour de contrôle, dans une salle située dans cette tour, ou dans un des Centres en Route de la Navigation Aérienne (CRNA). En France, ceux-ci sont au nombre de cinq, et couvrent le territoire métropolitain jusqu'aux frontières, et jusqu'aux limites des eaux territoriales. Ils sont situés à Brest (LFRR), Bordeaux (LFBB), Aix-en-Provence (LFMM), Reims (LFEE) et Athis-Mons (LFPP) (Cf. Figure 13.2). Un avion volant aux instruments est toujours en contact radio avec un contrôleur, qui change au fur et à mesure de la progression du vol. Cela se fait par des modifications de fréquence qui permettent, par exemple, de passer de la fréquence de la tour de contrôle à la fréquence de l'approche. Pour ce qui concerne le contrôle en route, l'espace aérien est découpé en portions tridimensionnelles appelées *secteurs*. Selon la charge de travail, due, entre autres, à la quantité d'avions à gérer, les secteurs peuvent être regroupés entre eux ou dégroupés. Chaque secteur ou groupe de secteurs est géré par une paire de contrôleurs, dont un seul s'occupe des échanges vocaux avec le pilote.

Tout au long de la vie du vol IFR, celui-ci est connu des systèmes de contrôle, et donne lieu à de la production et de l'enregistrement de données. Les calculateurs sont situés dans les lieux d'implantation des services de contrôle, au Centre d'Exploitation des Systèmes de

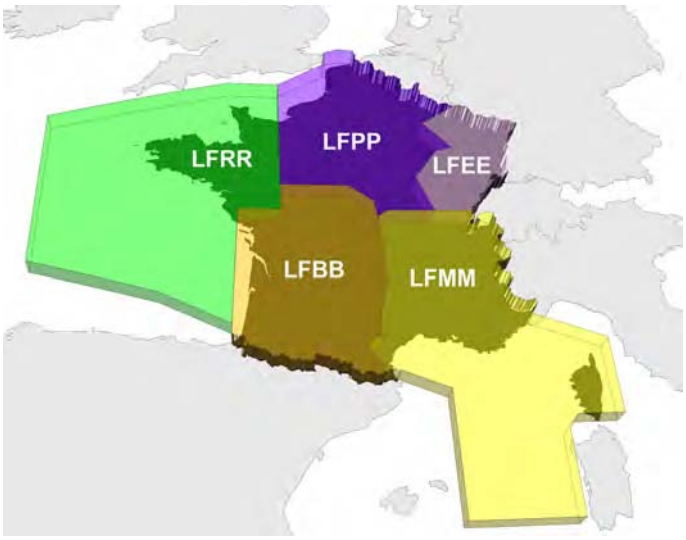


FIGURE 13.2 – Partie inférieure des zones de couverture des Centres en Route de la Navigation Aérienne français.

la Navigation Aérienne Centraux (CESNAC), qui est un service à l'échelle nationale, et au sein de services européens comme la CFMU (Central Flow Management Unit) dépendant de l'Organisation européenne pour la sécurité de la navigation aérienne (Eurocontrol [2]).

Le système français de gestion du trafic aérien est le CAUTRA [Poirot-Delpech 95], constitué de plusieurs sous-systèmes situés dans les CRNA ou au CESNAC. Ses constituants principaux sont le Système de Traitement Initial des Plans de vol (STIP), le Système de Traitement Radar (STR), et le Système de Traitement des Plans de Vol (STPV). Le STIP est situé au CESNAC, alors que les STPV et les STR sont implantés dans les CRNA. Le CAUTRA est relié à de multiples autres systèmes et organismes, comme les tours de contrôle, les approches, les systèmes météo, les systèmes militaires, la CFMU, etc. L'un de ces systèmes est SATIN (Système Automatisé de Traitement des Informations Nationales) qui fournit, de manière cyclique, des données, comme la description de l'espace aérien, des routes aériennes et des points de navigation. Il est également connecté à des systèmes d'archivage de données, utilisés essentiellement pour le calcul des redevances dues, par les exploitants aériens, pour la fourniture des services de la navigation aérienne, pour des besoins d'analyse en cas d'incident ou d'accident ou pour des études sur l'amélioration de l'espace aérien, comme le contour des secteurs. Selon le type de données, celles-ci ont une durée de vie légale.

Un vol fait d'abord l'objet d'un dépôt de plan de vol, qui est un contrat passé entre l'exploitant, ou le pilote, et le fournisseur de service de la navigation aérienne. Il contient de multiples informations relatives au vol souhaité, du décollage à l'atterrissage. Il s'agit, entre autres, des indications de temps (heures de décollage et d'atterrissage) et de lieux (aéroports de départ et d'arrivée, cheminement). Selon le cas, ce dépôt peut intervenir de plusieurs mois jusqu'à une heure avant le décollage. Le plan de vol est traité par le STIP et par la CFMU, qui

2. http ://www.eurocontrol.int

vont calculer une route et une heure de décollage la plus proche possible de l'heure demandée, mais prenant en compte les prévisions de trafic à l'échelle européenne. Cette heure peut être dans une fourchette de temps, appelée *créneau de décollage*. Le but est d'obtenir une gestion globale et harmonisée du trafic en limitant les coûts et la durée des éventuelles attentes. A l'issue du traitement initial, et en fonction de la route calculée, les STPV concernés par le vol sont informés de celui-ci. Quand l'avion décolle, grâce aux informations radar, le STR effectue une surveillance et une poursuite du vol.

Les données d'un vol ne concernent pas uniquement les événements commençant au décollage de l'avion jusqu'à son atterrissage. Leur production commence dès qu'il est connu des systèmes de traitements, et se poursuit jusqu'à l'archivage final. Cela passe par l'activation du vol dans les systèmes informatiques du CAUTRA, la mise en route des moteurs, le roulage, la phase de vol proprement dite, l'atterrissage, le roulage sur l'aéroport d'arrivée jusqu'au point de stationnement. Toutes ces étapes font intervenir un échange volumineux de données entre différents systèmes, ceux-ci étant à bord des avions, et au sol.

13.2 Traitement et archivage des données dans les systèmes de navigation aérienne

Plusieurs systèmes sont utilisés pour procéder à l'archivage des données aéronautiques. Ils sont gérés par les autorités délivrant les services de gestion de trafic aérien ou par les exploitants aéronautiques, dont les compagnies aériennes.

Du côté des autorités de contrôle, beaucoup de données sont issues du CAUTRA. Elles sont archivées et traitées par plusieurs systèmes, dont SYNOPSIS (SYstèmes Nationaux d'Optimisation de Prévision et de Statistique pour l'Information et le Suivi). Il met en œuvre plusieurs applications appelées services, dont :

- la réalisation des prévisions de trafic ;
- la fourniture d'une vision globale du trafic ;
- l'analyse du trafic ;
- l'archivage les données issues, notamment, du STIP, du STPV et du STR.

Parmi les logiciels assurant ces services, se trouvent COURAGE et PRESAGE, dont nous exploiterons les données dans le chapitre 14.1.

Le service COURAGE (Calcul Optimisé des Uceso et Régulations pour l'Amélioration de la Gestion de l'Espace) [DTI 10] permet de vérifier l'adéquation entre le trafic et la capacité de contrôle des CRNA. Pour cela, il compare les regroupements et dégroupements de secteurs avec le trafic. Il est utilisé dans deux phases de la gestion des flux de trafic :

- La phase pré-tactique : COURAGE prévoit le schéma d'ouverture, c'est-à-dire l'organisation des regroupements et dégroupements de secteurs en fonction du temps. Le but est d'assurer une bonne fluidité du trafic et de ne pas surcharger l'activité de contrôle. Cette phase est réalisée en coordination avec les autorités militaires qui gèrent l'activité

des zones de l'espace qui leur sont attribuées. En effet, quand certaines zones militaires ne sont pas actives, elles peuvent être empruntées par le trafic civil ;

- La phase d'analyse : elle permet d'étudier a posteriori une journée de trafic par le biais de différentes représentations, telles que des histogrammes de trafic, des courbes de charge, des décompositions de flux, des tableaux de vols...

Le service PRESAGE (PREvision de Situation Aérienne et de Gestion de l'Espace) [DTI 05] est une IHM permettant de présenter des situations aériennes passées, actuelles et prévues. Pour cela, il centralise les informations radar issues des cinq CRNA. En effet, chaque CRNA possède son propre STR dont la zone d'exploitation est illustrée sur la figure 13.2. De plus, il dispose des informations plan de vol de la journée en cours. Grâce à ce service, il est possible de prévoir et d'anticiper les problèmes de saturation de l'espace aérien, de faire un suivi global du trafic sur toute la France, et d'analyser le trafic passé de la journée.

13.3 Les données des compagnies aériennes

Elles sont échangées en temps réel avec les systèmes au sol des fournisseurs de services de la navigation aérienne et des compagnies aériennes. Elles sont également enregistrées à bord de l'avion, pour être analysées et archivées après le vol, ou pour fournir des informations après un incident ou un accident. Les données échangées avec le sol utilisent un système de communication codé appelé ACARS (Aircraft Communications Addressing and Reporting System) utilisant des fréquences VHF ou des communications par satellites. Il permet de communiquer automatiquement des informations sur l'état de l'avion, de communiquer manuellement des informations à caractère opérationnel et d'assurer des échanges vocaux. Les types de messages concernent, entre autres, les plans de vol, les autorisations des services de contrôle, appelées *clairances*, les informations météorologiques, des informations pour les équipages ou les services des compagnies aériennes...

Cependant, en cas d'accident, ces données se révèlent insuffisantes, et les boîtes noires apportent de nombreuses informations supplémentaires. Les événements récents que sont le crash du vol AF447, de Rio à Paris, le 1^{er} juin 2009 et la disparition en mer du vol MH370, de Kuala Lumpur à Pékin, le 8 mars 2014, montrent l'importance cruciale de ces données. Une partie de celles-ci est mentionnée dans le rapport [BEA 12] du Bureau Enquête Analyse [3] concernant l'accident du vol AF447 :

- Conversations entre les pilotes ;
- Echanges vocaux entre les pilotes et les organismes de contrôle ;
- Trajectoires dans les plans horizontal et vertical ;
- Variations de vitesse et d'accélération ;
- Vitesse et force du vent ;
- Alarmes à bord de l'avion ;
- Etats des équipements ;

3. http ://www.bea-gouv.fr

- Etc.

Nous avons évoqué, en introduction de ce mémoire, la recommandation de sécurité d'imposer, pour les avions effectuant du transport public de passagers, la transmission régulière de paramètres de base. A la suite de la disparition du vol MH370, cette recommandation a été renouvelée, puis entérinée par l'OACI. En effet, un accord a été signé le 14 mai 2014 [4], entre les états membres et les industriels pour mettre en œuvre un suivi des vols commerciaux, par la transmission vers les systèmes au sol de la position des avions long-courriers. De plus, en cas de situation d'urgence à bord de l'avion, d'autres types de données seront automatiquement transmises par celui-ci. Ces événements montrent l'évolution dans la transmission des données entre les avions et les systèmes au sol, qui tend vers une plus grande quantité d'échanges à l'échelle mondiale. Cette tendance est confirmée par le projet SESAR, que nous aborderons dans le chapitre 13.5.

13.4 Le rôle clé de la trajectoire

La trajectoire de vol fait partie des données aéronautiques les plus étudiées, car elle constitue un véritable enjeu. Elle correspond à l'ensemble des points de l'espace par lesquels l'avion passe effectivement. Comme à chaque point est associée une heure de passage, la trajectoire est donc quadridimensionnelle. Elle est dite 4D. Nous la différencions de la route qui est la ligne brisée reliant tous les points par lesquels l'avion est censé passer. Aujourd'hui, la trajectoire n'est pas déterministe, en ce sens qu'il n'est pas possible de connaître, avant le décollage, l'ensemble des futures positions de l'avion, même si la route est connue. Cela pose des problèmes dans les futurs systèmes de navigation aérienne. Pour appréhender cette notion, il est nécessaire de comprendre les étapes qui vont mener à la trajectoire réelle de l'avion.

Le pilote, ou la compagnie aérienne, fait part de son intention par le dépôt d'un plan de vol, qui est un formulaire dans lequel celui-ci est décrit. Entre autres informations, il contient le type de l'avion, sa vitesse, son niveau de vol de croisière ou RFL (Requested Flight Level), les aéroports de départ et de destination, la route prévue, l'heure de décollage et la durée du vol.

La route est une suite de points qui sont des balises associées à des moyens radio au sol, ou des coordonnées géographiques. Dans le premier cas, le point est identifié par un bigramme ou un trigramme, comme *TOU* pour Toulouse. Dans le second, il est identifié par un nom prononçable de cinq lettres. Il s'agit, par exemple, de *AFFRI* qui est un point situé près de la ville de Saint-Affrique dans l'Aveyron. La route demandée par le pilote est donc identifiée par une suite de points, ce qui constitue au final une ligne brisée qui reflète au mieux la trajectoire qu'il souhaite parcourir. Ces point sont dits *publiés* parce qu'ils apparaissent dans la documentation aéronautique officielle gérée en France par le Service d'Information Aéronautique [5]. Cette suite de points, ou *route déposée*, est traitée par le STIP qui, à partir

4. Air & Cosmos, mai 2014
5. http ://www.sia-gouv.fr

de différentes contraintes, entre autres liées à l'ouverture de zones militaires, va calculer une nouvelle route respectant au mieux la route demandée. Cette route calculée devient alors la route officielle de ce vol, appelée *route CAUTRA*. C'est à partir de celle-ci que le système va déduire les différents organismes de contrôle concernés par ce vol, notamment les CRNA, et ensuite les informer. Rappelons que le plan de vol est également traité par la CFMU, ce qui permet d'assurer une gestion globale du trafic à l'échelle européenne, de prévoir les évolutions de charge, et d'en informer les systèmes des états membres.

Tout au long de la vie du vol, l'avion est pris en charge par des organismes de contrôle que sont la tour, l'approche et l'en route. Ces différents organismes gèrent le vol matérialisé par un *strip*, qui est actuellement une bande de papier cartonné contenant des informations sur celui-ci. Un strip en route, comme celui de la figure 13.3, présente, sur la moitié de droite, une liste de balises, qui est un extrait de la route CAUTRA définie précédemment. Il s'agit des balises CHW, CLARA, GIRKO, SOKMU et MERUE. Elles ne concernent que l'espace aérien dont le contrôleur a la charge. Le strip du contrôleur du secteur suivant dispose à son tour d'une suite de balises correspondant aux points suivants de la route CAUTRA.

FIGURE 13.3 – Strip en route.

L'avion est censé survoler les balises les unes après les autres, ou passer *travers balises*, ce qui signifie passer suffisamment près, mais sans obligatoirement les survoler. Cependant, la trajectoire présente un tracé différent de la route CAUTRA, qui peut, dans certains cas, être sensiblement écartée. En effet, la route est une trajectoire théorique calculée à partir du souhait du pilote ou de la compagnie aérienne, mais plusieurs facteurs sont à l'origine d'une trajectoire différente. Ceux-ci sont, entre autres :

- Des consignes de contrôle, par exemple pour respecter les normes de séparation entre les avions ;
- Des événements météorologiques nécessitant une déviation (orage...), ou perturbant la trajectoire (vent en altitude) ;
- Des besoins d'exploitation, comme la réduction de la trajectoire pour gagner du temps et économiser le carburant.

Les figures 13.4 et 13.5 illustrent, respectivement dans le plan horizontal et dans le plan vertical, de tels écarts par rapport à la route prévue. La première montre, en vert et de gauche à droite, la route CAUTRA calculée pour un vol régulier effectuant la ligne Toulouse-Orly en septembre 2013. Le tracé a été tourné pour obtenir une image horizontale suffisamment lisible. Nous pouvons constater que du point A au point B, cette route coïncide avec la trajectoire de l'avion tracée en rouge. Cependant, à partir du point B, alors que la route change de direction, la trajectoire va tout droit pour rejoindre la route au point C. Cela correspond à une *directe*

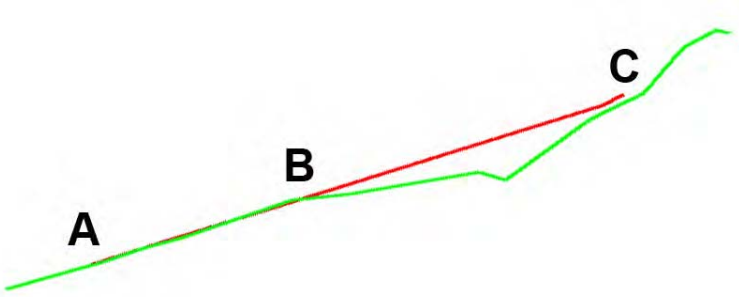


FIGURE 13.4 – Exemple d'écarts horizontaux entre la route et la trajectoire d'un avion. La route prévue est une ligne brisée constituée d'une suite de points (vert). La trajectoire réelle de l'avion est une courbe (rouge).

qui consiste à ne pas se diriger vers la balise suivante, mais vers une des prochaines balises de la route, pour raccourcir la trajectoire. Cette directe est le résultat d'une négociation entre le pilote et le contrôleur qui l'accorde si les conditions, notamment de trafic, le permettent.

La figure 13.5 illustre un écart, dans le plan vertical, par rapport à la trajectoire. Il s'agit d'un vol privé reliant Genève à Séville en décembre 2013. Le pilote a demandé un niveau de vol de croisière (RFL) 410 lors du dépôt de plan de vol, indiqué en vert. Un niveau étant exprimé en centaines de pied, il correspond à une altitude de 41000 pieds au-dessus de l'isobare 1013 hPa. La trajectoire en rouge montre que l'avion est monté plus haut, car il a effectivement atteint le niveau 450. Cet écart dans le plan vertical est également le résultat d'une négociation entre le pilote et le contrôleur.

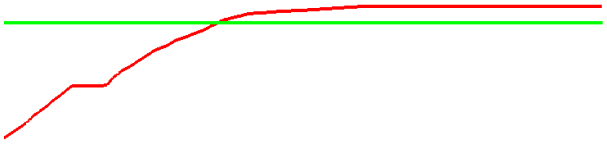


FIGURE 13.5 – Différence verticale entre le niveau prévu et les altitudes réelles de l'avion. Le niveau de croisière (RFL) demandé pour cet avion est indiqué en vert. Les niveaux réels, en rouge, sont issus des données de l'avion.

Ces deux exemples illustrent la différence qui peut exister entre une route 3D calculée et une trajectoire 3D effectivement parcourue. Différents facteurs, issus des systèmes de navigation aérienne ou des avions, expliquent ces écarts qui sont très fréquents. Il est donc difficile d'obtenir une bonne adéquation entre le prévu et le réalisé. Si cette adéquation était obtenue, il serait alors possible d'assurer une meilleure prédictibilité du trafic, et de l'optimiser en conséquence. Cela fait partie des défis à relever dans les futurs systèmes de gestion de trafic aérien, que nous allons aborder maintenant.

13.5 Les futurs systèmes

Le programme SESAR (Single European Sky ATM Research) [6] est le futur système européen de gestion du trafic aérien. Il s'appuie sur un partenariat entre la Commission Européenne, Eurocontrol et des industriels. De plus, les fournisseurs de service de la navigation aérienne et les grands aéroports européens sont parties prenantes du programme. Le but de SESAR est de renouveler le système de gestion du trafic européen à l'horizon 2020. Il lui a été fixé des objectifs, notamment de restructuration de l'espace aérien, pour absorber l'augmentation de trafic, améliorer la sécurité, et diminuer l'impact environnemental et les coûts.

L'un des piliers de ce programme est la gestion de la trajectoire de vol. Pour cela ont été définis les concepts de Shared Business Trajectory (SBT) et Reference Business Trajectory (RBT) [SJU 12]. Quand un plan de vol est déposé, sa trajectoire est initialement une intention (*business intention*) de la part de l'exploitant aéronautique. Une phase de négociation est alors entreprise avec les organismes de gestion de l'espace aérien, dont les aéroports. Une trajectoire SBT est alors créée. Pour les vols militaires, celle-ci s'appelle Shared Mission Trajectory (SMT). Quand l'avion est prêt pour le départ, la SBT devient la RBT, qui est la trajectoire de référence, pour laquelle, un engagement réciproque de la respecter est pris par tous les acteurs concernés. Ce concept permettra d'optimiser la gestion globale du trafic. L'exploitant aura une connaissance a priori de la trajectoire et pourra gérer au mieux ses vols. Le fournisseur de service de la navigation aérienne pourra d'optimiser la gestion de l'espace aérien, grâce à une connaissance plus précise, et, avec anticipation, de l'évolution de la situation aérienne. La gestion de la trajectoire nécessite un échange permanent de données entre les systèmes bord et les systèmes sols, ceux-ci étant très variés, afin d'avoir une connaissance commune du vol et la plus à jour possible. Cela concerne les services de navigation aérienne, les autorités militaires, les services météorologiques... Les échanges de données s'appuieront sur le projet SWIM (System Wide Information Management) qui garantira l'infrastructure et les moyens, afin de pouvoir les assurer, en toute sécurité, au sein du système de gestion du trafic aérien. Il préconise, entre autres, la mise à disposition des informations le plus tôt possible, pour permettre aux autres acteurs de s'adapter en conséquence avec suffisamment d'anticipation.

Les Etats-Unis proposent également un nouveau système de gestion du trafic aérien appelé NextGen [7] (Next Generation Air Transportation System) ayant des objectifs similaires à ceux de l'Europe. Il s'appuie également sur un échange permanent de données entre les différents acteurs. Pour cela, plusieurs systèmes, basés notamment sur l'utilisation de signaux GPS et l'amélioration des communications vocales, sont mis en œuvre.

6. http ://www.sesarju.eu
7. http ://www.faa.gov/nextgen

Chapitre 14

Exploitation des données aéronautiques

14.1 Scénario 1 : exploitation des données SATIN et COURAGE

14.1.1 Les données SATIN ET COURAGE

Le service SATIN, situé au CESNAC, fournit des données, dites nationales, à plusieurs services, dont le STIP, le STPV et le STR. Générées tous les 28 jours, elles constituent la *bande CA*, qui contient les informations suivantes :

- balises et points de navigation ;
- liste des secteurs ;
- routes aériennes ;
- itinéraires pour relier les aéroports aux routes aériennes ;
- types et performances des avions.

Ces données servent, par exemple, à élaborer la route CAUTRA à partir du plan de vol déposé par le pilote, et à déterminer les CRNA concernés par le vol, afin de provoquer des dessertes de strips.

Dans le chapitre 13.2, sur le traitement et archivage des données dans les systèmes de navigation aérienne, nous avons évoqué le service COURAGE [DTI 10]. Celui-ci utilise un format d'archivage de données plans de vol appelé *format COURAGE* [DTI 12]. Il est constitué de multiples champs appelés *cartes*, décrivant la vie du vol, comme :

- des informations issues du plan de vol : indicatif d'appel (callsign), aéroports de départ et de destination ;
- les routes 4D prévues et réalisées. Il s'agit de suites de balise au-dessus desquelles, ou au travers desquelles, l'avion est passé ;
- les heures d'entrées et de sorties des secteurs, ainsi que d'impressions de strips ;
- des informations sur les créneaux de décollage.

```
00 DEBUT DU FICHIER COURAGE
01 VERSION 4.2
02 13-11-2013
03 17-10-2013
04 6585 6456 6421 6456
06 NCR CRG CRE IFPL MOD IDLA ICHG FLS DES FSH RET DEP IDEP SLT SLC EVL FAB FPA FSA FAC AFP ABI
05
11
20 AFR1839 EDDV LFPG 4635 -1 E170 0
21 1170 360 431
22 0 N N 0
31 NOR KENUM GESLO IDOSA RAPOR VEDUS XERAM ENORI DEVIM LORNI PGNE
32 -243 -241 -234 -231 -227 -225 -221 -219 -218 -216 -207
33 360 360 360 360 310 310 310 260 178 150 40
41 LU LL 1P AP TE RB
42 -248 -248 -248 -248 -248 -243
43 -243 -242 -227 -223 -222 -208
44 -197
45 1 4 4 5 8 10
46 4 4 4 7 9 11
13
20 AFR1839 EDDV LFPG 4635 -1 E190 0
21 1379 360 431
22 =
31 =
32 -33 -31 -25 -22 -18 -16 -12 -9 -8 -7 2
33 360 360 360 360 310 310 310 260 177 150 40
```

FIGURE 14.1 – Début du fichier COURAGE du 13 novembre 2013.

A partir de fichiers COURAGE et SATIN, nous avons élaboré un jeu de données qui va être explicité dans le paragraphe suivant.

14.1.2 Exploitation des données SATIN ET COURAGE

Nous avons réalisé un fichier à partir de données COURAGE et SATIN, couvrant une période aéronautique chargée, du 1^{er} juillet au 31 août 2012, et une autre plus calme, du 15 octobre au 15 décembre de cette même année. Les données du CAUTRA ont été enrichies à l'aide de deux bases que nous avons élaborées, contenant 8000 aéroports et 750 exploitants aéronautiques.

Cela donne une base de 964000 transactions, chacune d'elle correspondant à un vol. Les attributs sont les suivants :

- *Ident* : identifiant de l'avion. Il s'agit du numéro du vol dans le CAUTRA, appelé *numéro CAUTRA*. Cette valeur entière est comprise entre 0 et 9999 ;
- *Month* : mois du vol, codé par une valeur entière ;
- *Callsign* : indicatif d'appel de l'avion, sous la forme d'une chaîne de huit caractères alphanumériques ;
- *TypeOper* : type d'exploitation de l'avion : vol privé, vol commercial régulier, vol d'affaire, vol charter à bas prix (*low cost*), vol d'avions d'état et cargo ;
- *Adep* : indicateur d'emplacement de l'aéroport de départ [OACI 12] ;
- *AdepLon* : longitude de l'aéroport de départ ;
- *AdepLat* : latitude de l'aéroport de départ ;
- *Ades* : indicateur d'emplacement de l'aéroport de destination ;
- *AdesLon* : longitude de l'aéroport de destination ;

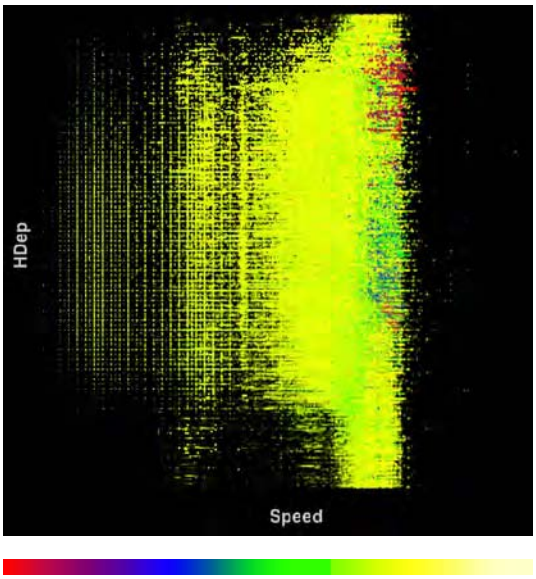


FIGURE 14.2 – Visualisation de cinq attributs correspondant à quatre mois de trafic en 2013. Le gradient de couleur est présenté en-dessous. Il correspond à la latitude de l'aéroport d'arrivée.

- *AdesLat* : latitude de l'aéroport de destination ;
- *Acft* : type de l'avion, sous la forme d'une chaîne de caractères ;
- *RFL* : niveau de croisière demandé dans le plan de vol ;
- *Speed* : vitesse de croisière demandée dans le plan de vol ;
- *HDep* : heure de départ ou d'entrée dans la zone de couverture nationale ;
- *HArr* : heure d'arrivée ou de sortie de la zone de couverture nationale.

Dans un premier temps, la base est intégralement soumise à l'algorithme Eclat, choisi par l'utilisateur (Cf. Chapitre 1.5.1), en considérant tous les attributs numériques, hormis l'identifiant, ainsi que le type d'opérateur qui comporte 6 valeurs. Chaque transaction contient donc 10 attributs. Les valeurs seuil de support et de confiance sont respectivement égales à 0,01 et 0,80. 12509 règles d'association sont ainsi extraites en 50 secondes.

Dans le but d'étudier les liens qu'il peut y avoir entre les attributs et les heures de départ, nous considérons ensuite la visualisation de la figure 14.2 dont la caractérisation de Card & Mackinlay est présentée dans le tableau 14.1. Nous avons affecté la vitesse à X, l'heure de départ, ou d'entrée au-dessus du territoire français, à Y, le RFL à la taille des points, la latitude de l'aéroport d'arrivée à la couleur, dont le gradient est présenté en bas de la figure, et la longitude de l'aéroport de destination à l'alpha. Cette figure apporte un certain nombre d'informations. Elle montre, par exemple, que, pour des vitesses faibles, l'heure de départ a lieu pendant la journée, alors qu'elle est susceptible d'avoir lieu à toute heure pour des avions ayant une vitesse élevée. Les vols dont l'aéroport de destination a une latitude basse sont représentés en rouge et en bleu, étant donné le gradient de couleur. Ces vols se retrouvent dans les fortes valeurs de vitesse. En effet, il est logique qu'un avion qui va loin vole rapidement.

Data				Automatic perception							Controlled perception
Variable	D	F	D'	X	Y	Z	T	R	-	[ ]	CP
Speed	Q	f	Q	P							
HDep	Q	f	Q		P						
RFL	Q	f	Q				S				
AdesLat	Q	f	Q					C			
AdesLon	Q	f	Q					A			

TABLE 14.1 – Le modèle de Card & Mackinlay de la visualisation initiale des données du scénario 1

Nous choisissons d'extraire les règles d'associations à partir de la visualisation (Cf. Chapitre 4.5) en fixant le nombre maximum de clusters à cinq par dimension. Ceux-ci, calculés en 12 secondes par l'algorithme k-means, assisté du critère de Caliński-Harabasz sont présentés sur la figure 14.3 (Cf. Chapitre 4.3). Les clusters de la variable visuelle X sont redimensionnés pour avoir un meilleur étalement des basses vitesses. Cela est illustré par les vues (a) et (b).

En gardant les mêmes valeurs seuil de support de confiance, l'algorithme extrait 49 règles en 3 secondes. Celles-ci sont visualisées dans l'espace d'exploration des règles avec ResultsViewer, dans lequel nous avons affecté le support à X, la confiance à Y, le lift à Z, Piatetsky-Shapiro à la couleur, et Sebag-Schoenauer à la taille de point (Cf. Figure 14.4). La vue (a) montre que la plupart des règles ont un support relativement faible. La confiance s'étale de 0,80 à 1. La vue (b) fait apparaître un groupe de règles dont le lift, la confiance et la mesure Sebag-Schoenauer sont élevés. L'une d'elles est illustrée à la figure 14.5. Elle s'écrit

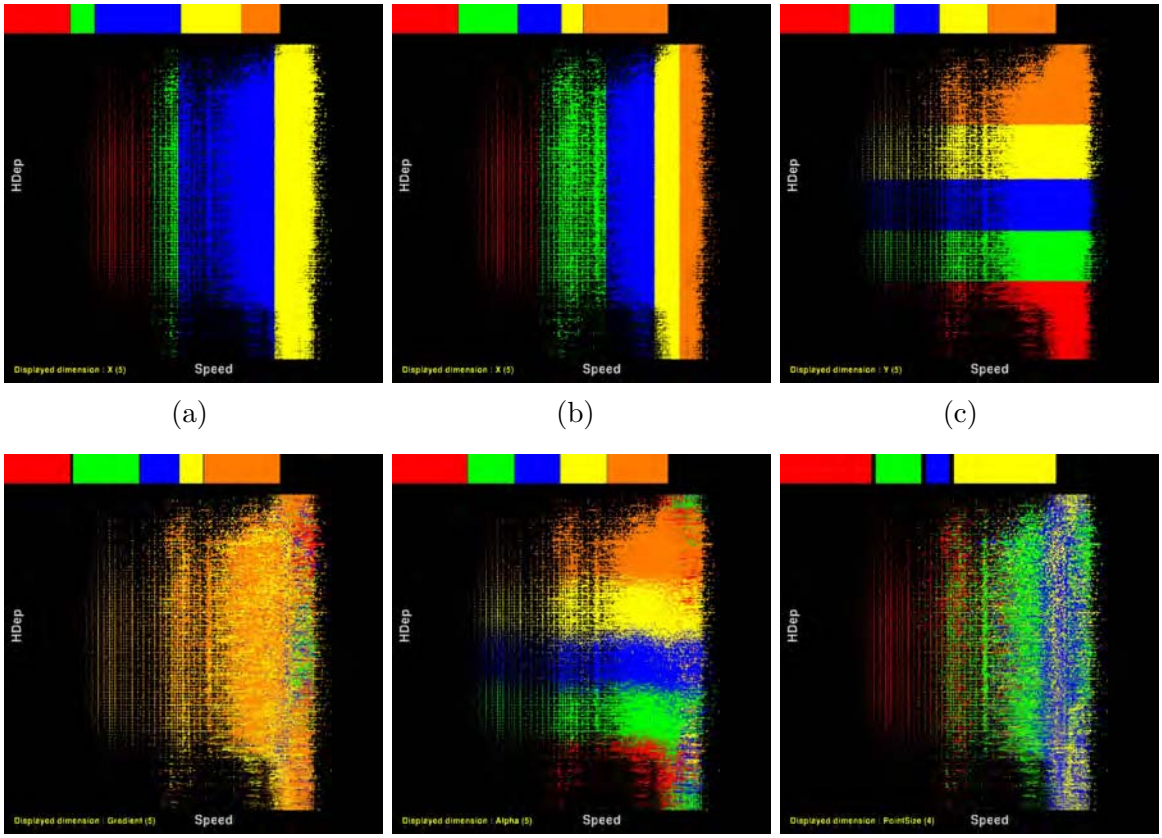
$$Vitesse \in [80, 212] \Rightarrow RFL \in [10, 200]$$

Cela signifie que si les avions volent lentement, alors leur niveau de vol de croisière est bas. En sélectionnant cette règle N°19 dans ResultsViewer, et en affectant le lift moyen à l'alpha dans DataViewer à partir de la visualisation de la figure 14.2, nous vérifions, sur la figure 14.6, que cela fait bien ressortir les valeurs basses de vitesse et de RFL.

Dans un second temps, nous souhaitons nous intéresser uniquement aux avions dont l'aéroport de départ est en Grande-Bretagne. Pour les sélectionner, nous assignons l'attribut ADEP à la variable Z, puis nous sélectionnons toutes les chaînes de caractères commençant par *EG*, comme cela est illustré sur la figure 14.7. Une fois cette sélection de 95800 transactions effectuée, nous dé-assignons la variable visuelle Z. Puis, après avoir calculé les clusters, Eclat, dont la valeur minimale de support est maintenant fixée à 0,05, extrait 26 règles en une seconde, qui sont représentées sur la figure 14.8 (a). Nous sélectionnons la règle dont le lift et la confiance sont les plus élevés. De plus, la taille du point la plus grande indique que la mesure de Sebag-Schoenauer est maximale pour cet ensemble de règles. Cette règle est :

$$Speed \in [426, 560]\ AdesLat \in [22.55, 41.71]\ HDep \in [586, 921] \Rightarrow RFL \in [330, 470]$$

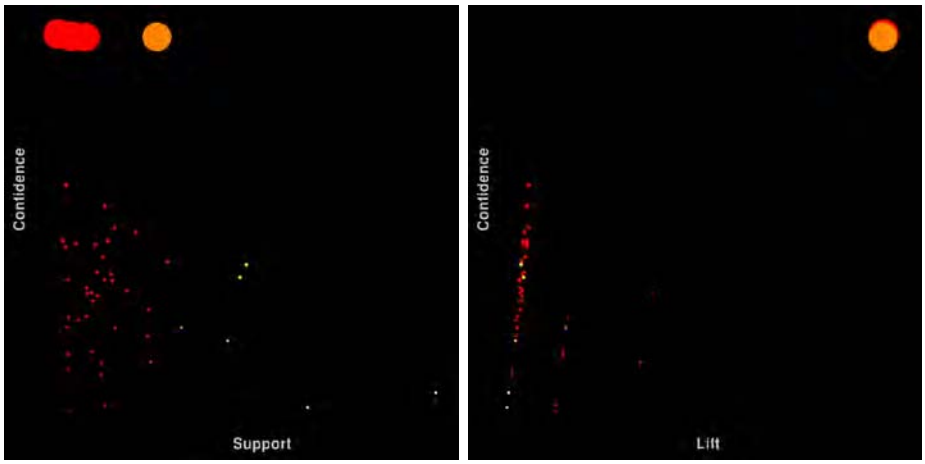
Les valeurs de support, de confiance, de lift et de Sebag-Schoenauer sont respectivement



(a) (b) (c)

(d) (e) (f)

FIGURE 14.3 – Représentation des clusters par dimension, calculés par k-means. (a) : X - (b) : X après modification par l'utilisateur - (c) : Y - (d) : gradient de couleur - (e) : alpha - (f) : taille des points.



(a) (b)

FIGURE 14.4 – Visualisation des règles d'association en fonction de plusieurs mesures.

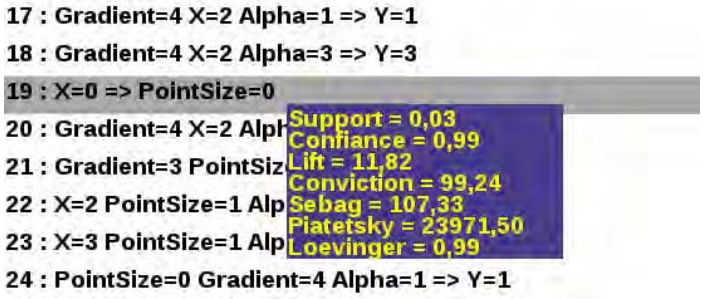


FIGURE 14.5 – Extrait de liste des règles d'association présentant plusieurs mesures.

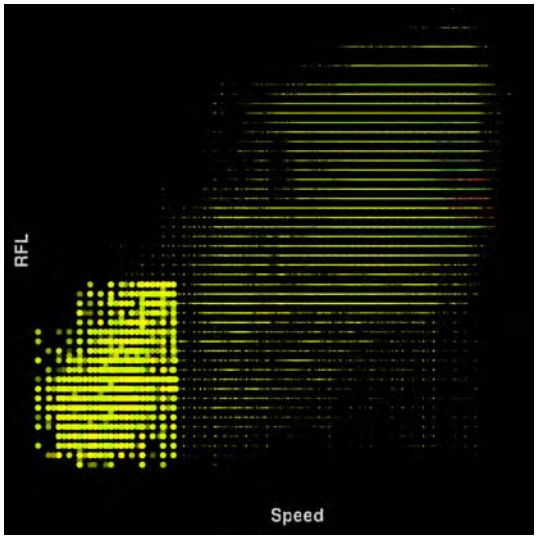


FIGURE 14.6 – La règle 19 de la figure 14.5 a été sélectionnée.
En affectant le lift moyen à l'alpha, la règle est vérifiée par la visualisation des données.

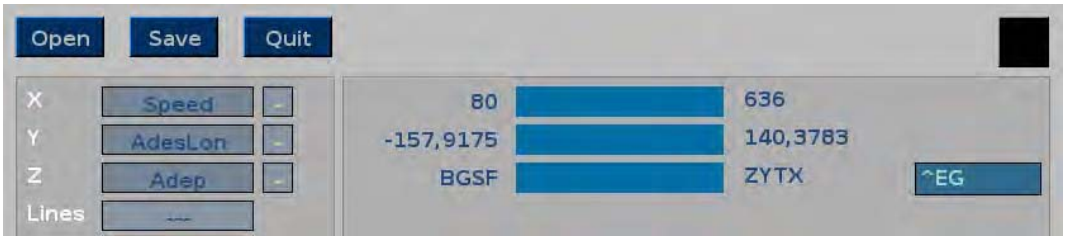
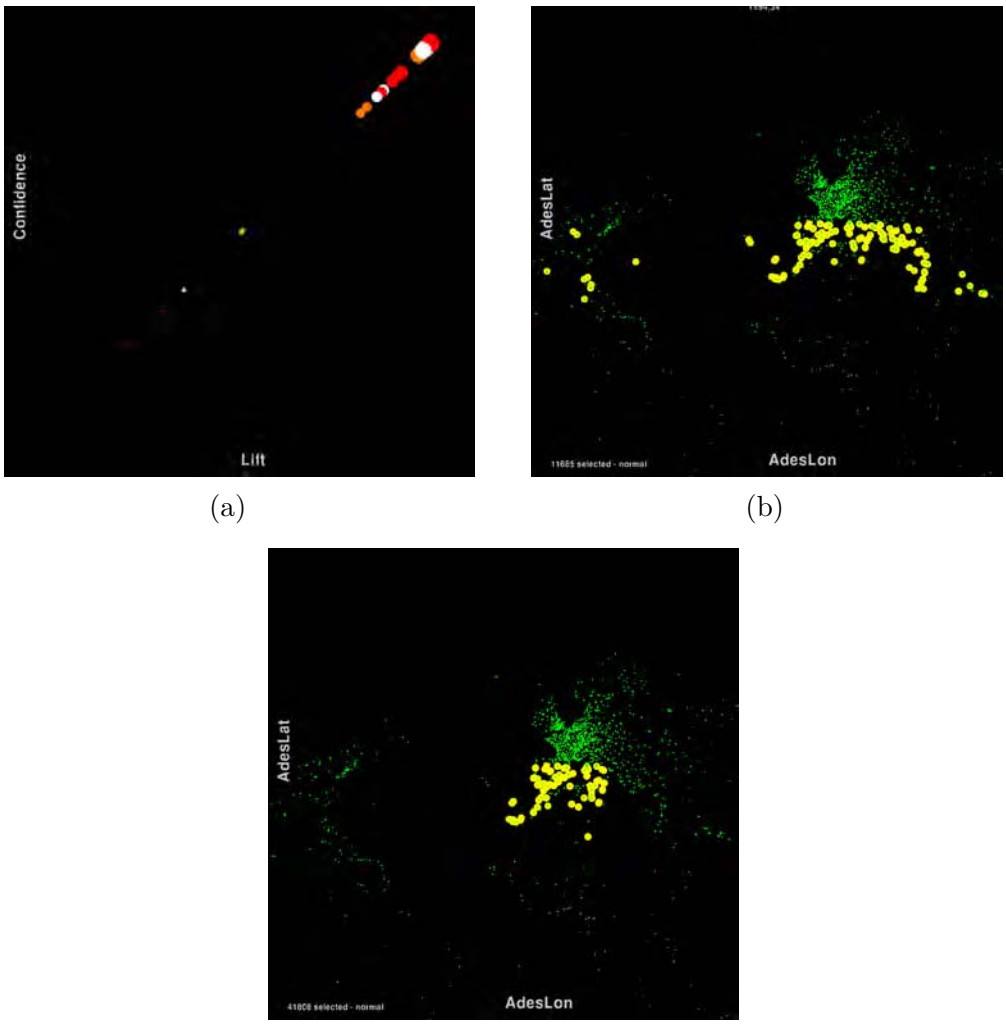


FIGURE 14.7 – Sélection des aéroports de départ situés en Grande-Bretagne.
Leur indicateur d'emplacement commence par *EG*.



(a) (b)

(c)

FIGURE 14.8 – Règles et données après sélection des départs britanniques.
(a) : les nouvelles règles générées. Destinations concernant la règle dont le lift (b) puis Sebag-Schoenauer (c) est le plus élevé.

égales à 0,12, 0,97, 1,31 et 39,34. Afin d'en comprendre la signification, nous avons modifié l'assignation des variables visuelles dans la visualisation des données, en affectant les coordonnées des aéroports de destination à X et Y et le lift moyen de la règle à la taille des points, que nous avons ensuite sélectionnés. La vue (b) de la figure 14.8 montre ainsi que les avions concernés par cette règle sont à destination de l'Europe du Sud, du Sud des Etats-Unis, et de l'Amérique Centrale. Ils volent à des vitesses élevées, et rentrent dans l'espace aérien français entre 9 heures et 15 heures. Elle s'énonce de la manière suivante : pour les vols au départ de Grande-Bretagne, volant dans ces conditions de vitesse, de latitude d'aéroport de destination et d'heure d'entrée au-dessus du territoire français, leur niveau de croisière est très élevé. Il s'agit de caractéristiques de vols moyen et long-courrier qui correspondent aux distances entre la Grande-Bretagne et les aéroports particularisés sur la vue (b).

Nous considérons maintenant la règle ayant le lift, la confiance et la mesure de Piatetsky-Shapiro les plus élevés. Elle est repérable par une couleur proche du blanc sur la figure 14.8 (a). Cette règle est la suivante :

$$Speed \in [426,560]\ AdesLat \in [22.55,41.71]\ AdesLon \in [-23.5,15.4] \Rightarrow RFL \in [330,470]$$

Les valeurs de support, de confiance, de lift et de Piatetsky-Shapiro sont respectivement égales à 0,43, 0,97, 1,31 et 9930,44. Cette règle est proche de la précédente, mais sa dépendance avec la longitude des aéroports de destination montre qu'elle ne concerne que des aéroports du Sud de l'Europe et du Nord de l'Afrique, comme en témoigne la vue (c).

Nous complétons l'étude de ces règles par celle des itemsets, à l'aide de la représentation circulaires des itemsets (Cf. Chapitre 7). Ils sont représentés par la figure 14.9 (a). Puis nous exécutons l'algorithme d'optimisation du positionnement des itemsets, basé sur les distances intra-cercles et inter-cercles, (Cf. Chapitre 7.5). Le nouveau graphe, généré en 2 minutes, est illustré par la vue (b). La nouvelle position des 1-itemsets montre que certains ont été regroupés sur le cercle extérieur, et d'autres éloignés. Cela illustre leur implication dans le partage d'information commune. Sur la vue (c), nous avons sélectionné l'itemset à partir duquel est extraite la règle illustrée par la Figure 14.8 (b). Par propagation, cela fait remonter jusqu'aux 1-temsets $Speed \in [426,560]$, $AdesLat \in [22.55,41.71]$, $HDep \in [586,921]$ et $RFL \in [330,470]$. De plus, ce 4-itemset sert à élaborer un seul 5-itemset. Ce dernier contient le 1-itemset supplémentaire $AdesLon \in [-23.49,15.44]$. L'observation de ce 1-itemset montre, dans la partie supérieure gauche du graphe, qu'il est à l'origine d'un nombre élevé d'arêtes. Pour obtenir la vue (d), nous avons mis en œuvre l'algorithme de bundling KDEEB qui est exécuté en 1 seconde (Cf. Chapitre 9.2). La couleur ayant pour composante RGB le triplet (0.9, 0.1, 0.01), le blending de couleur fait ressortir en jaune les zones du graphe donnant lieu à une forte accumulation. C'est le cas, par exemple, de $AdesLon \in [-23.49,15.44]$. Nous sélectionnons cet itemset sur la vue (e), en affectant la largeur au support, et la couleur à la confiance moyenne des règles concernées par les itemsets. La couleur est un gradient du rouge au blanc, en passant par le jaune. La largeur des arêtes montre l'évolution du support à partir de l'itemset sélectionné. Le nombre élevé d'arêtes, à partir de celui-ci, que nous avions constaté sur la vue (a), est confirmé par la grande largeur des courbes qui en émanent sur la

vue (e). L'évolution de la confiance est visualisée par celle de la couleur. Elle est élevée pour l'itemset sélectionné, et tend à diminuer, comme l'indique la variation de couleur vers le jaune puis le rouge. Cependant, un 5-itemset a une confiance élevée, car les arêtes qui l'atteignent sont blanches.

Nous sélectionnons cet itemset, pour obtenir la vue (f). Le graphe ne montre alors que ses sous-sets, jusqu'au premier niveau. La couleur blanche des arêtes indique bien une confiance élevée. Quant à la largeur, elle permet de constater le degré d'implication des itemsets dans la base, du fait de leur support. Ainsi, $AdesLon \in [-23.49, 15.44]$ est fortement impliqué, mais son voisin $Speed \in [426, 560]$ l'est également, ce qui s'explique par le nombre importants d'avions volant à vitesse élevée. Par ailleurs, nous pouvons constater que les itemsets des ordres 1 à 5 sont globalement regroupés dans le graphe circulaire, grâce à l'optimisation de placement des nœuds. Dans la vue (a), les deux 1-itemsets qui viennent d'être cités étaient diamétralement opposés.

Comme cette représentation des itemsets en graphe circulaire est reliée par linking à la représentation sous forme de règles, ainsi qu'à DataViewer, les interactions avec ce graphe permettent de mettre en valeur, et cela de manière instantanée, la visualisation des données initiales, en fonction des affectations des mesures liées aux itemsets.

14.1.3 Conclusion du scénario 1

Dans ce premier scénario, nous avons illustré le pilotage de la recherche d'itemsets et de règles d'associations à partir de la visualisation. Pour cela, le rôle de l'utilisateur a été mis en avant, par le choix des variables visuelles à partir desquelles sont construites les règles d'association, par la sélection des données, par le redimensionnement des clusters calculés par k-means et par le choix de l'algorithme de Data Mining.

A partir d'interactions avec les données et avec les résultats algorithmiques, nous avons étudié des règles en visualisant des mesures de qualités qui leur sont associées. Par sélection de données alphanumériques, nous avons ensuite extrait des itemsets et des règles établissant le lien entre les aéroports de départ et de destination, les vitesses et les heures. Cela a permis de montrer l'extraction de règles locales qui n'auraient pu être détectées qu'avec un support seuil plus faible, mais en ayant à faire face à un plus grand nombre de règles.

Nous poursuivons l'illustration du lien entre les espaces de données et de règles dans le scénario suivant, en mettant en œuvre l'enrichissement et le paramétrage de la visualisation des données à partir des règles d'association.

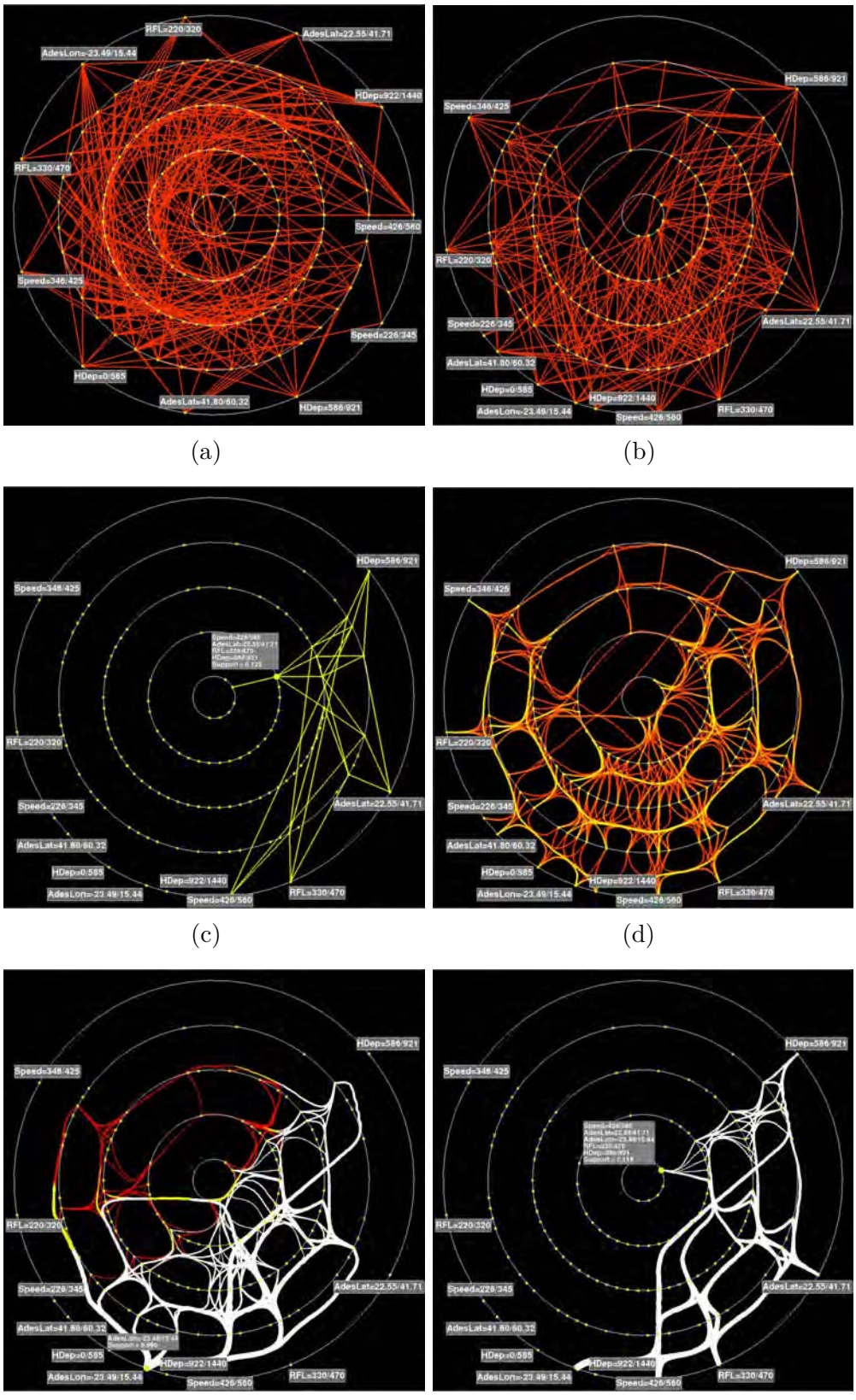


FIGURE 14.9 – Graphes circulaires des itemsets concernant les vols au départ de Grande-Bretagne.

```
00:00:35    157,656     50,031      389    433,521    338,346      -3305      7303 ACFT001  F900 FNLU LFPB
00:00:35    177,781    138,109      380     401,44    303,365          0      2755 ACFT002  B738 HESH EGCC
00:00:35   -260,812    -53,062      370    427,588    212,827          0      2213 ACFT003  A332 EGVN FHAW
00:00:35     78,156    184,781      334    419,019    272,791      -1266      7527 ACFT004  B763 HESH EGKK
00:00:39      69,75    186,672      311    426,489    310,199       -902      2767 ACFT005  A320 LTFE EGKK
00:00:39     87,547    172,719      340    398,145    311,649          0      2746 ACFT006  A320 LGIR EGKK
00:00:43     15,719    -15,188      380     441,65    341,038          0      5540 ACFT007  E135 LEGE EGSS
00:00:43    -44,375     63,625      341    464,282    346,201          0      5361 ACFT008  B734 LEAL EGKK
00:00:43       5,25    -26,422       41    114,478     98,273          0      7041
00:00:50    130,188    -21,656      350    432,642    166,399          0      1000 ACFT009  B733 LFPG LFMT
00:00:50    156,953     51,719      382    436,597    338,159      -3305      7303 ACFT001  F900 FNLU LFPB
00:00:50    213,547    -69,969        0        ,439    243,435          0      1000
00:00:50    -41,156      3,297      380    488,452     37,463          0      2373 ACFT010  B738 GCLP EHAM
00:00:50    -41,359     27,031      159    215,552     134,55        539      1737 ACFT011  ATP  LFRN LFML
```

FIGURE 14.10 – Début du fichier IMAGE du 2 août 2013.
Les indicatifs d'appel ont été anonymisés.

14.2 Scénario 2 : exploitation des données IMAGE

14.2.1 Les données IMAGE

Le chapitre 13.2 a abordé le service PRESAGE, permettant de présenter des situations aériennes passées, actuelles et prévues. Ce trafic est enregistré sous la forme de fichiers dans un format appelé IMAGE (Instrument Macroscopique d'Aide à la Gestion de l'Espace), du nom d'un ancien service qui a récemment fusionné avec PRESAGE. Ce format fédère les données issues des STR des cinq CRNA, afin d'avoir une vision globale de la situation aérienne au-dessus du territoire français. Les données auxquelles nous avons eu accès nous ont été fournies par l'ENAC, en format ASCII (Cf. Figure 14.10).

Elles contiennent les champs suivants :

- Heure ;
- Position de l'avion dans la grille CAUTRA en miles nautiques :
- Altitude envoyée par l'avion (Mode C) ;
- Vitesse en nœuds ;
- Cap en degrés ;
- Taux de montée ou de descente en pieds par minute ;
- Code transpondeur (Mode A) [1]
- L'indicatif d'appel (dans la figure14.10, ces indicatifs ont été anonymisés)
- Le type d'avion ;
- L'aéroport de départ ;
- L'aéroport de destination.

Dans un fichier IMAGE, la position de chaque avion est mise à jour en moyenne toutes les trois minutes, sachant que cela peut être très inférieur, comme en témoigne le vol *ACFT001* de la figure, qui est mis à jour au bout de 15 secondes.

1. Le transpondeur est un dispositif embarqué dans l'avion grâce auquel, par un traitement appelé *corrélation*, l'indicatif d'appel est affiché sur l'image radar du contrôleur.

14.2.2 Prétraitement des données

Les deux formats de données que sont IMAGE et COURAGE décrivent respectivement la trajectoire et la route de l'avion. Nous les avons traités afin de pouvoir utiliser des données issues des deux types de sources, en effectuant des calculs, notamment de distances. Cela correspond à la troisième phase de l'Extraction de Données à partir des Connaissances (Cf. Chapitre 1.1.2) qui consiste à nettoyer et prétraiter les données, et au processus de transformation du Visual Analytics (Cf. Chapitre 1.4).

Notre base de données initiale est constituée des enregistrements de trafic des journées du 7 au 9 août et du 7 au 9 novembre 2013. Nous avons effectué un prétraitement de cette base pour extraire des données exploitables par la plate-forme VIDEAM. Pour cela, il a fallu faire face à plusieurs contraintes.

Les calculs de distance que nous avons réalisés portent sur la distance horizontale entre la trajectoire et la route CAUTRA, ainsi que sur la distance verticale entre cette première et le RFL, qui est le niveau de vol de croisière demandé. Comme les vols apparaissent souvent de manière incomplète dans les fichiers auxquels nous avons eu accès, afin que leurs trajectoires soient suffisamment représentatives, nous avons choisi de ne garder que les vols qui apparaissent au moins cinq fois dans la base, ce qui correspond à environ 15 minutes de trajectoires minimum. A partir d'une trajectoire, nous calculons la distance entre chaque point et la route CAUTRA, qui est une ligne brisée. Cela permet d'avoir une connaissance des écarts latéraux entre la route prévue et la trajectoire réelle, et ainsi de mettre en exergue les directes entre points de la route. L'intérêt que nous portons à l'écart vertical entre la trajectoire et le RFL porte sur les vols volant au-dessus de ce dernier. En effet, un avion n'a a priori pas d'intérêt à voler plus bas que son niveau de vol demandé, parce que cela induit une plus grande consommation de carburant. De plus, il est possible qu'un avion atteigne son niveau de croisière dans la zone de couverture d'un CRNA étranger, et qu'il soit par exemple en descente dans celle d'un CRNA français ayant enregistré le trafic. Donc, nous avons souvent une information erronée sur cet écart. Pour ces raisons, nous avons écarté les vols dont le niveau maximum est inférieur au RFL. Enfin, comme nous nous intéressons également aux aéroports de départ et de destination, nous avons écarté les avions dont l'un de ces deux aéroports est inconnu. Dans ce cas, il est représenté dans le plan de vol par la chaîne de caractères *ZZZZ*.

A l'issue du prétraitement nous nous retrouvons en présence d'environ 1,2 million de transactions, correspondant à des vols dont les trajectoires couvrent l'ensemble du territoire français, du niveau 0 au niveau 500. Chaque transaction étant constituée de 23 attributs, la base contient 28,4 millions de données.

14.2.3 Exploitation des données IMAGE

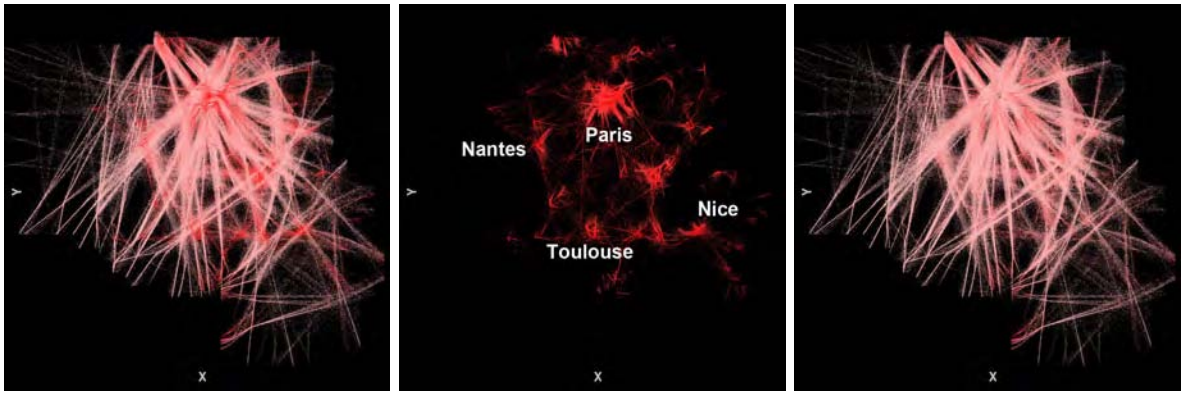
A partir des données issues du prétraitement, nous avons généré un fichier dont les attributs sont les suivants :

- *Ident* : identifiant de l'avion ;

- *Date* : mois du vol ;
- *Callsign* : indicatif d'appel de l'avion ;
- *TypeOper* : type d'exploitation de l'avion ;
- *Speed* : vitesse de croisière demandée dans le plan de vol ;
- *RFL* : niveau de croisière demandé dans le plan de vol ;
- *DistRFL* : écart vertical de l'avion par rapport du RFL, exprimé en centaines de pieds [2] ;
- *Adep* : indicateur d'emplacement de l'aéroport de départ [OACI 12] ;
- *AdepLon* : longitude de l'aéroport de départ ;
- *AdepLat* : latitude de l'aéroport de départ ;
- *Ades* : indicateur d'emplacement de l'aéroport de destination ;
- *AdesLon* : longitude de l'aéroport de destination ;
- *AdesLat* : latitude de l'aéroport de destination ;
- *Time* : heure du message piste, exprimée en minute depuis minuit ;
- X : abscisse de l'avion dans la grille CAUTRA. Il s'agit d'une grille centrée sur la position de coordonnées (47°N, 0°W) dans laquelle les coordonnées sont exprimées en 1/64 de miles nautique [3] par des nombres entiers ;
- Y : ordonnée de l'avion dans la grille CAUTRA ;
- *Distance* : distance horizontale entre la position de l'avion et la route, exprimée en miles nautiques ;
- VXY : vitesse de l'avion en nœuds [4] ;
- *Heading* : cap de l'avion en degrés par rapport au Nord ;
- *FL* : niveau de vol de l'avion en centaines de pieds ;
- *Rate* : taux de montée ou de descente de l'avion en pieds par minute.
- *DistRFLMax* : écart vertical maximal de l'avion par rapport du RFL, exprimé en centaines de pieds ;
- *DistRouteMax* : distance horizontale maximale entre la position de l'avion et la route, exprimée en miles nautiques ;

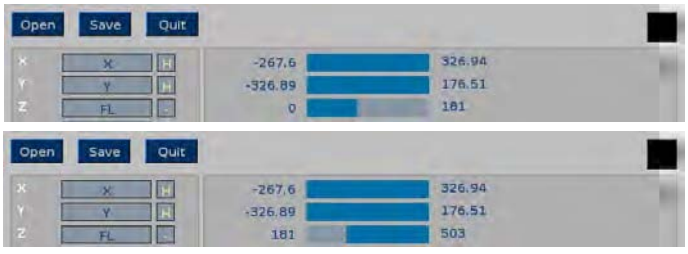
La figure 14.11 montre cette base de données en affectant les coordonnées des avions aux variables visuelles X et Y, et le niveau de vol à la couleur dont le gradient varie du rouge pour les valeurs faibles au blanc. Les points rouges correspondent aux vols volant le plus bas, et les plus clairs aux avions les plus hauts. Avec ViewerSettings, nous filtrons les vols volant à un niveau inférieur à 180 (Cf. Figure 14.12 (a)), puis au-dessus de ce niveau (Cf. Figure 14.12 (b)). Dans le premier cas, cela fait ressortir les mouvements d'avions au départ et à l'arrivée des grands aéroports, comme Paris, Toulouse, Nantes et Nice (Cf. Figure 14.11 (b)). Dans le second cas, il ressort essentiellement des avions en transit, facilement repérables par des longues trajectoires rectilignes (Cf. Figure 14.11 (c)).

2. 1 pied = 1ft = 30,48 cm
3. 1 mile nautique = 1 Nm = 1852 m
4. 1 nœud = 1 Kt = 1 Nm/h



(a) (b) (c)

FIGURE 14.11 – Représentation nationale de trafic géré par les cinq CRNA.
Il est présenté dans sa globalité (a), puis les avions en-dessous du niveau 180 (b), et ceux volant au-dessus de ce niveau (c).



(a)

(b)

FIGURE 14.12 – Filtrage des avions en fonction de leur niveau de vol pour paramétrer les vues de l'image 14.11.

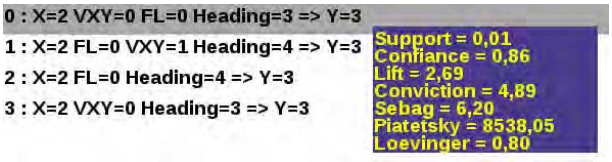


FIGURE 14.13 – Extraction de quatre règles.

Considération globale de la base, puis extraction et analyse de quatre règles d'association

Dans un premier temps, nous cherchons à obtenir une vision globale du mouvement des avions. Pour cela, dans DataViewer, nous sélectionnons les attributs X, Y, VXY, FL et $Heading$. Ils informent sur la position des avions, leur vitesse en valeur et en direction, et leur altitude. En fixant le nombre de clusters à cinq, k-means calcule ceux-ci en 23 secondes. Puis, les valeurs minimales de support et de confiance étant respectivement égales à 0,01 et 0,80, Eclat extrait quatre règles en trois secondes. La visualisation des données étant celle de la figure 14.11 (a), ces règles ont pour conclusion l'attribut Y, car il est affecté à la variable visuelle Y. Notons qu'à la différence du scénario 1, la visualisation ne pilote plus les règles, si ce n'est le choix de la conclusion des règles générées.

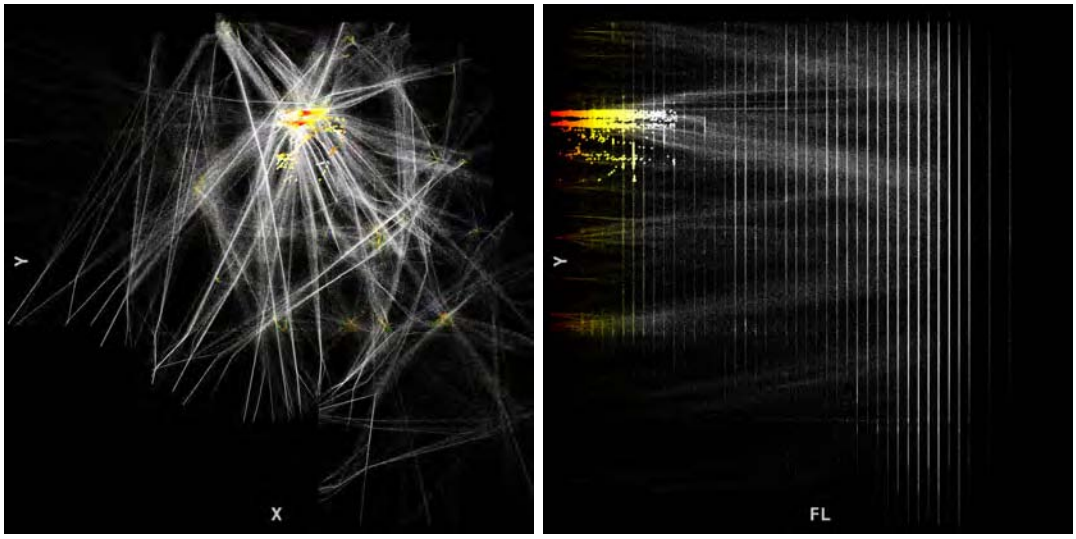
Les règles sont présentées de manière explicite, à l'aide de RulesList, par la figure 14.13. Toutes concernent le cluster numéro deux de l'attribut X et le numéro 3 de l'attribut Y, c'est-à-dire les points dont les abscisses sont comprises entre 3742 et 10425, et les ordonnées entre 1131 et 8073, dans la grille CAUTRA. Ainsi, les données situées dans ces intervalles d'attributs sont fortement liées.

En sélectionnant la première règle (Cf. Figure 14.13), nous assignons le lift moyen à la taille des points dans DataViewer, et la couleur au niveau de vol, avec un gradient allant du rouge au jaune, puis au blanc, cette dernière couleur étant obtenue à partir du niveau de vol 90. Par ailleurs, la variable visuelle Z est également affectée au niveau de vol. Le résultat est représenté par la figure 14.14, sous la forme de deux vues, la première montrant les avions selon leurs coordonnées X et Y, et la deuxième présentant l'ordonnée en fonction du niveau de vol. Le basculement d'une vue à l'autre dans DataViewer est réalisé à l'aide d'une rotation animée des points, pour que le plan de la vue passe de XY à ZY, et réciproquement. Nous constatons que les gros points concernent la région parisienne. Dans un souci de lisibilité, afin d'avoir une vision globale des données, nous avons quand même affecté une taille très petite aux points qui ne sont pas concernés par la règle sélectionnée.

En zoomant la vue 14.14 (a), et en ne présentant que les points impliqués dans la règle, nous obtenons maintenant la figure 14.15 (a), qui fait ressortir les trajectoires d'arrivée et de départ des aéroports parisiens. Les points rouges correspondent aux parties les plus basses des trajectoires, c'est-à-dire les pistes, le début de la montée et la fin de l'atterrissage avant le toucher des roues de l'avion. Tous ces points sont alignés et dans l'axe des pistes. C'est pourquoi leur orientation apparaît dans les segments dessinés par les points. La vue (b) est une carte d'aérodrome de Roissy Charles-de-Gaulle [SIA 14], publiée par le SIA [5]. L'aéroport du Bourget y apparaît également, au Sud-Ouest. Il est aisé de faire le rapprochement entre les deux vues de cette figure. Sur celle du haut, Roissy est représenté par deux traits correspondant aux doubles paires de pistes. En zoomant cette vue, et en diminuant la taille des points, ces quatre pistes apparaissent alors sur DataViewer.

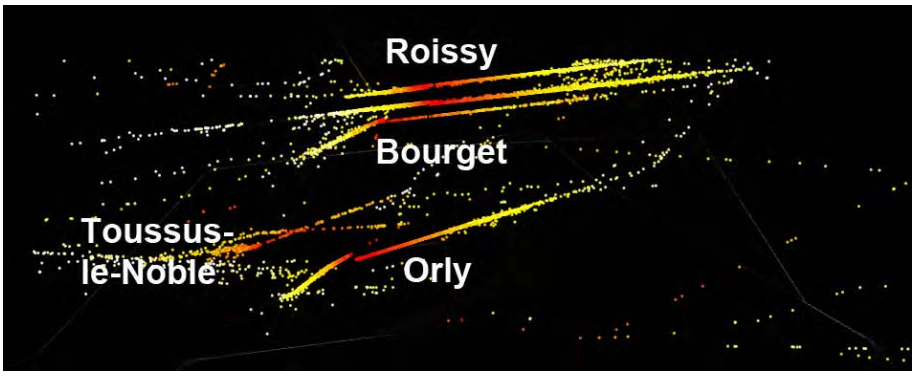
Les autres 1-itemsets de la règle sélectionnée de la figure 14.13 corroborent nos constatations. En effet, $VXY = 0$ correspond aux vitesses inférieures à 220 Kt, c'est-à-dire aux vitesses les plus

5. http ://www.sia-gouv.fr

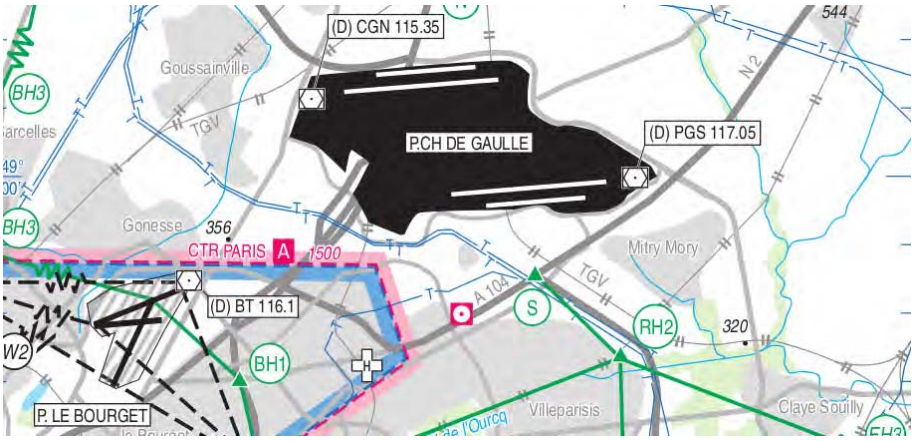


(a) (b)

FIGURE 14.14 – Affectation du lift de la règle sélectionnée à la taille du point. (a) : représentation géographique des avions. (b) : représentation de l'ordonnée en fonction du niveau de vol.



(a)



(b)

FIGURE 14.15 – (a) : Les points concernés par la règle sélectionnée correspondent aux mouvements d'avions à basse altitude de la région parisienne.

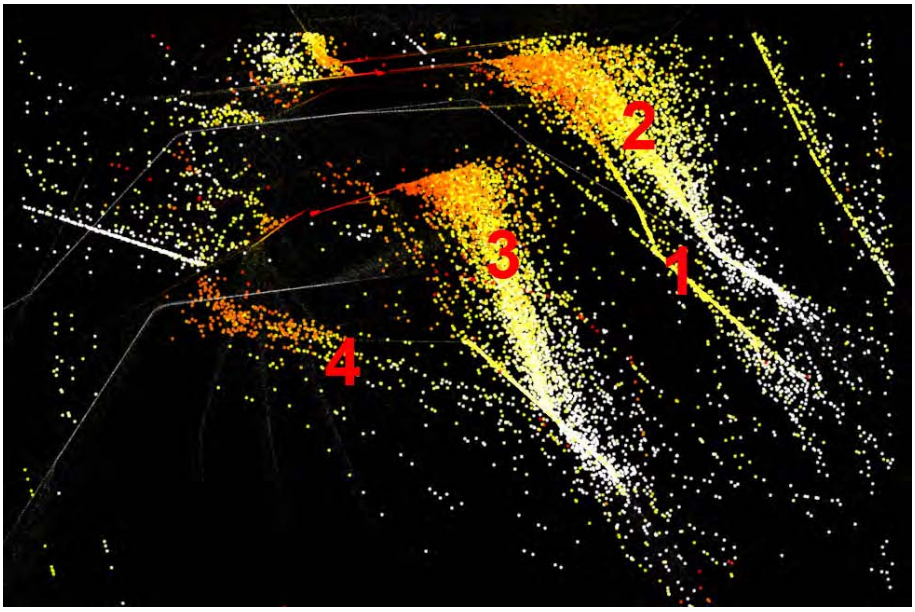
(b) La carte aéronautique[SIA 14] montre l'implantation de Roissy et du Bourget.

basses correspondant aux arrivées et départ. Quant à $FL = 0$, il correspond aux niveaux de vols inférieurs à 122, soit les plus bas, c'est-à-dire ceux qui sont utilisés lors de ces mêmes phases de vol. Enfin, $Heading = 3$ concerne les caps compris entre 200 et 280 degrés, c'est-à-dire entre le cap Sud et le cap Ouest. Ils coïncident avec les orientations de pistes apparaissant sur la figure 14.15 (a). Cette règle numéro 0 est à rapprocher de la règle numéro 3 qui s'appuie sur un sous-itemset, et dont les mesures de qualités sont très proches.

Nous pourrions reprendre le même raisonnement avec les règles 1 et 2, la seconde étant constituée d'un sous-itemset de la première. Les 1-itemsets des attributs X et Y sont les mêmes. Mais les valeurs de VXY sont comprises entre 221 et 322 Kts, tandis que le cap des avions est compris entre 280 et 360 degrés. Il s'agit donc de positions d'avions situées plus haut et volant entre le cap Ouest et le cap Nord. En reprenant les mêmes réglages qui ont permis d'obtenir la figure 14.15 (a), nous obtenons la figure 14.16, après avoir sélectionné les règles 1 et 2. Pour cela, nous avons mis une taille de points très petite pour les données non concernées par cette règle, afin de montrer le tracé des pistes. Il ressort de cette figure, que les données impliquées dans les règles sélectionnées correspondent essentiellement aux flux d'arrivée vers Roissy, le Bourget et Orly. Lorsque qu'un avion quitte la phase de vol en route, pour se diriger vers un aéroport, il peut être amené à suivre une procédure d'attente, qui consiste à décrire un hippodrome, jusqu'à ce qu'il puisse continuer son vol. A l'issue de cette attente, il suit une *procédure d'approche standard*, qui consiste à suivre un cheminement publié dans la documentation aéronautique. Cela est illustré par le point 1 de la figure. Si le trafic le permet, il fait l'objet d'un guidage radar du contrôleur, ce qui optimise la trajectoire de l'avion, sans avoir à suivre de cheminement publié. Dans ce cas, les trajectoires d'arrivée vers les aéroports sont plus dispersées, mais sont convergentes. Les points 2, 3 et 4 montrent cette situation. Le point 2 montre le flux d'arrivée vers Roissy et Orly. Les points 3 et 4 montrent les flux d'arrivée vers le Bourget, le premier en atterrissant vers l'Ouest, et le second en atterrissant vers l'Est. Comme le niveau de vol est assigné à la couleur, le gradient passant du blanc au jaune, puis au rouge, montre que les avions sont bien en descente.

Extraction de règles à partir d'un sous-ensemble de la base

Dans un second temps nous nous intéressons à l'écart vertical entre le niveau de vol demandé par l'exploitant de l'avion, et le niveau auquel il a réellement volé. Le niveau demandé est le RFL qui est noté dans la case *NIVEAU* du plan de vol de l'avion (Cf. Figure 14.17, case A) [RCA 13]. A droite de celle-ci se trouve le champ *ROUTE* (case B), dans lequel est indiquée la suite de points de navigation. Il est possible d'y préciser des changements de niveau, ce qui permet d'avoir plusieurs RFL pour un même plan de vol. Ceux-ci n'apparaissent malheureusement pas dans les archives COURAGE, et nous n'avons eu donc accès qu'au RFL initial. C'est donc par rapport à celui-ci que nous avons considéré l'écart maximum entre le niveau de vol demandé et le niveau réel le plus haut effectivement atteint par les avions. Si le pilote demande plusieurs niveaux, cet écart pourra donc être important. Cependant, il est intéressant de s'y pencher, car il peut apporter des informations intéressantes.



(a)

FIGURE 14.16 – (a) : Les points concernés par les règles sélectionnées particularisent les arrivées vers les aéroports parisiens.

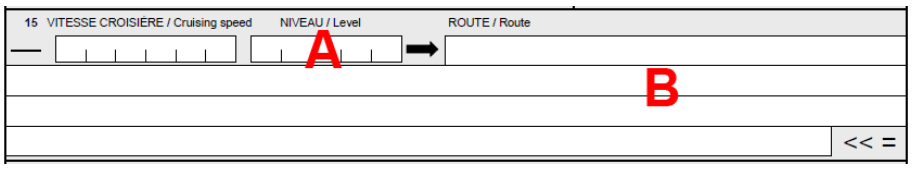


FIGURE 14.17 – Extrait du formulaire plan de vol.

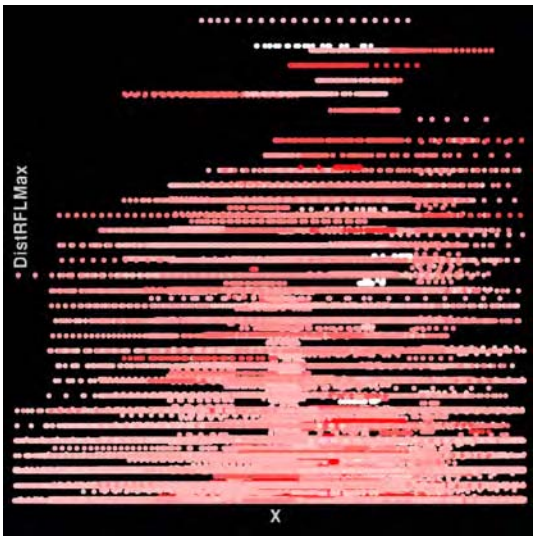


FIGURE 14.18 – Visualisation de trois attributs correspondant à six jours de trafic en 2013. La vitesse de croisière a été assignée au gradient de couleur.

Ainsi, en repartant de la visualisation de la figure 14.11 (a), DistRFLMax est assigné à la variable visuelle Z, puis nous visualisons les données dans le plan XZ (Cf. Figure 14.18). La plupart des points sont situés dans la partie basse de la figure. En effet, les vols dont la valeur *DistRFLMax* est inférieure ou égale à 20 représentent 65% du total. Ce ne sont pas ces vols qui vont nous intéresser dans notre étude, parce que cet écart de 2000 Ft par rapport au FL initial est faible. Nous sélectionnons donc les autres vols à l'aide du brushing dans DataViewer, ainsi que les attributs cochés dans la figure 14.19. Nous ne prenons pas en compte pas les attributs de position X et Y, afin de ne pas retrouver les règles que nous avons extraites précédemment au sujet des aéroports parisiens. Cependant, les critères de position se retrouvent dans les coordonnées des aéroports. De plus, nous cherchons des règles éventuellement en lien avec le type d'exploitant et un effet saisonnier. C'est pourquoi, nous avons sélectionné *TypeOper* et *Date*. Nous rappelons que l'échantillon de trafic provient de données des mois d'août, qui est fortement touristique, et de novembre. Nous avons ainsi choisi 12 attributs.

A partir de la visualisation de la figure 14.18 et des attributs de la figure 14.19, k-means calcule les clusters en 3 secondes. Leur nombre maximum ayant été fixé à cinq, le nombre de clusters des attributs numériques ainsi calculés est compris entre trois et cinq. L'algorithme Eclat, que nous avons choisi, extrait alors 12127 règles d'association en 13 secondes, pour des valeurs seuil de support et de confiance respectivement égales à 0,01 et 0,80. Celles-ci sont représentées sur la figure 14.20. La vue (b) montre deux zones de règles, celles de droite ayant un lift élevé, supérieur à 11, alors que celles de gauche ont un lift maximum égal à 1,23.

Nous sélectionnons la règle A de la vue (b) ayant les valeurs de confiance et de lift les plus élevées. Celle-ci est :

$$TypeOper = LowCost\ AdepLon = 3\ AdepLat = 4\ AdesLat = 3\ RFL = 1 \Rightarrow DistRFLMax = 2$$

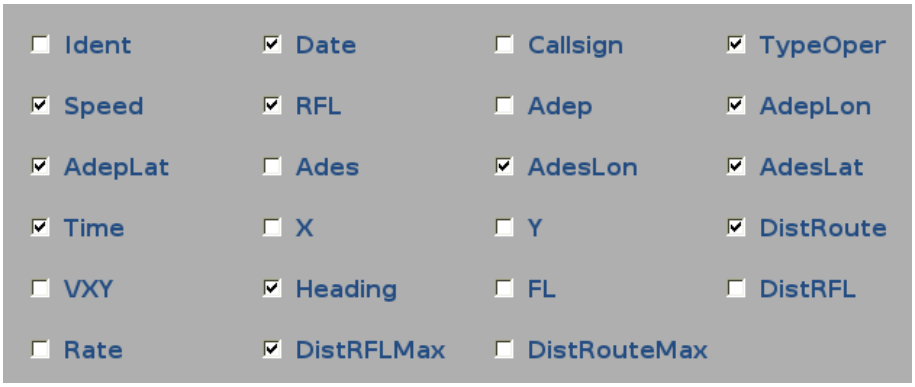
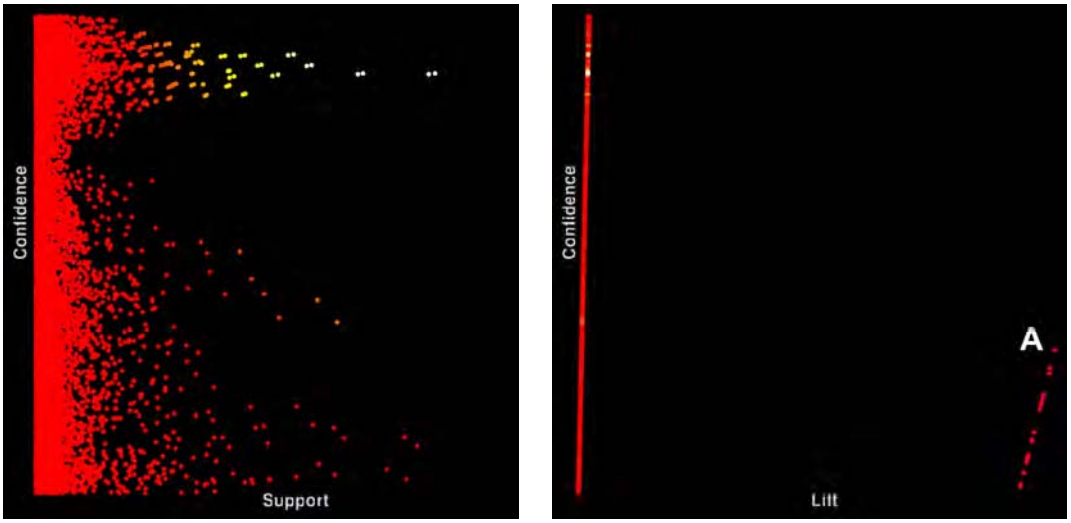
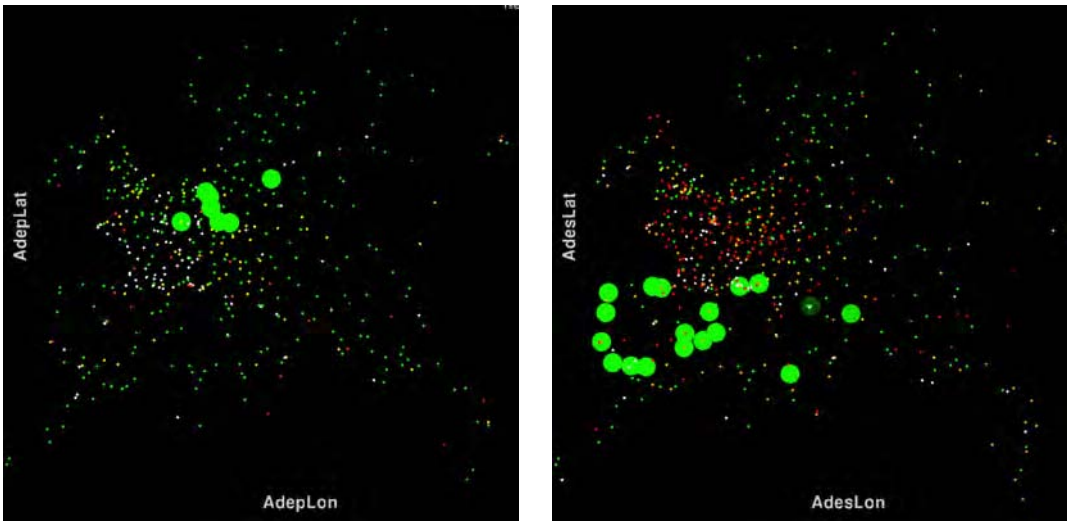


FIGURE 14.19 – Sélection des attributs pris en compte par l'algorithme de Data Mining pour le scénario N°2.



(a) (b)

FIGURE 14.20 – Visualisation des règles d'association en fonction de plusieurs mesures, pour le scénario N°2.



(a) (b)

FIGURE 14.21 – Conséquence, sur la représentation des aéroports, de la sélection des règles de la figure 14.20 (b).

Les valeurs de support, de confiance et de lift de cette règle sont respectivement égales à 0,015, 0,86 et 12.37. Elle signifie que si les avions respectent des conditions relatives aux coordonnées des aéroports de départ et de destination, si leur RFL demandé est compris entre 150 et 230, et si ce sont des compagnies LowCost, alors ils volent entre 6500 et 12800 Ft plus haut. Comme des conditions sont relatives aux positions des aéroports de départ et de destination, nous affectons leurs coordonnées aux variables visuelles X et Y et la couleur aux types d'opérateur, le vert correspondant ici aux compagnies LowCost. Après avoir assigné le lift moyen à la taille des points, la figure 14.21 fait ainsi ressortir les aéroports de départ (a) et de destination (b) des vols concernés par la règle sélectionnée. Les départs ont lieu en France et en Allemagne. Quant aux destinations, elles sont sur les côtes ibériques et méditerranéennes. Une autre manière d'étudier cette règle est par la visualisation des positions des avions concernés par celle-ci. Pour cela, nous les assignons aux variables visuelles X et Y. Nous obtenons ainsi la figure 14.22 sur laquelle les trajectoires sont bien orientées Nord-Est Sud-Ouest, conformément aux positions des aéroports de départ et de destination.

Une fois la règle isolée, il est nécessaire de pouvoir l'analyser pour la comprendre. Les vols pour lesquels *DistRFLMax* est inférieur à 20 correspondent à 65% des données. En d'autres termes, 65% des avions atteignent un niveau de vol maximum 2000 Ft au-dessus de celui qui a été demandé, dans le champ niveau, lors du dépôt de plan de vol. Dans les 35% autres cas, l'écart entre le RFL est le niveau maximum atteint est plus conséquent. Cela provient de pratiques d'exploitation, qui peuvent être dues aux régulations. En effet, lorsque le système a connaissance d'un plan de vol, la CFMU peut imposer une régulation pour éviter un encombrement de l'espace aérien (Cf. Chapitre 13.1). Elle prend la forme d'un créneau de décollage constitué d'une heure de début et de fin, entre lesquelles l'avion doit décoller. Si ce créneau n'est pas respecté, alors le départ est retardé, éventuellement de manière conséquente,

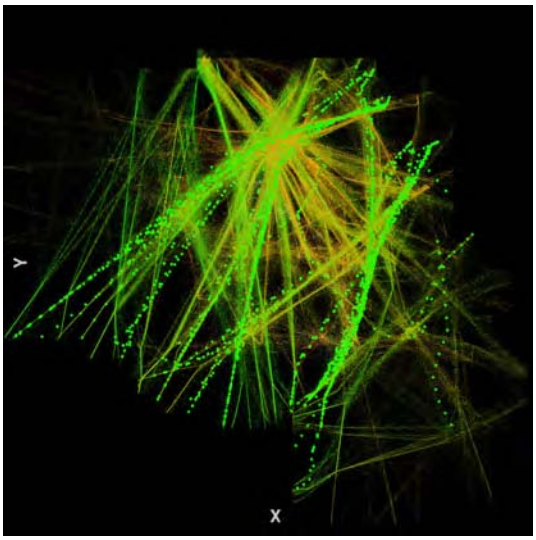


FIGURE 14.22 – Conséquence, sur la représentation des trajectoires d'avions, de la sélection de la règle A de la figure 14.20 (b).

avec une réaction en chaîne à laquelle la compagnie aérienne doit faire face. S'il est prévu que ce même avion soit utilisé pour assurer une autre liaison à l'issue de ce vol, alors elle peut être retardée, ce qui peut nécessiter d'impliquer un autre avion de la compagnie, voire d'une autre. Pour éviter des régulations ou en limiter les effets, il est ainsi possible de demander plusieurs RFL successifs dans le champ *ROUTE* du plan de vol (Cf. Figure 14.17 (B)). Cela permet de passer au-dessous ou au-dessus des zones à l'origine des régulations.

Paramétrage automatique de la visualisation à partir d'un ensemble de règles choisies

Afin de poursuivre l'exploration de l'espace des règles de la figure 14.20, nous sélectionnons les 34 règles situées à droite de la vue (b), afin de paramétrer automatiquement, à partir de celles-ci, la visualisation des données initiales (Cf. Chapitre 5.1). Comme nous cherchions les attributs dont dépend $DistRFLMax$, celui-ci est affecté à la variable visuelle Y. Les attributs les plus proches sont RFL, puis $TypeOper$, puis $Heading$ et enfin $AdepLat$ (Cf. Chapitre 5.1.4). Ils sont assignés aux variables visuelles, pour obtenir la caractérisation de la visualisation de la figure 14.23 (a) dans le tableau 14.2. ViewerSettings est configuré en conséquence (Cf. Figure 14.23 (b)). Notons que, comme $TypeOper$ est de type nominal, il est assigné à des couleurs discrètes plutôt qu'à la taille des points. Le tableau 14.3 montre cette correspondance. La visualisation des données a par ailleurs fait l'objet d'un zoom et d'un excentrement automatiques, pour être adaptée aux plages de valeurs des attributs concernés par les 1-itemsets des règles sélectionnées. Ainsi, les valeurs limites de RFL sont 52 et 328 ; les valeurs limites de $DistRFLMax$ sont 130 et 340. Ce sont donc ces données visualisées qui illustrent les règles sélectionnées. L'attribut le plus lié à l'écart maximum de RFL est le RFL, et c'est d'autant plus fort quand l'écart est important. Ensuite la dépendance provient de l'exploitant aéronautique, comme nous l'avons constaté en sélectionnant une règle. Puis elle

(a) (b)

FIGURE 14.23 – Configuration automatique de la visualisation des données à partir d'une sélection de règles de la figure 14.20.

Data				Automatic perception							Controlled perception
Variable	D	F	D'	X	Y	Z	T	R	-	[ ]	CP
RFL	Q	f	Q	P							
DistRFLMax	Q	f	Q		P						
Heading	N	f	N				S				
TypeOper	Q	f	Q					C			
AdepLat	Q	f	Q					A			

TABLE 14.2 – Tableau de Card & Mackinlay de la visualisation générée automatiquement par la sélection de règles de la figure 14.20.

provient du cap, et enfin de la latitude de l'aéroport de départ. Comme nous l'avions évoqué dans le chapitre 2.2, l'affectation d'une variable visuelle à l'alpha peut ne pas avoir d'effet sensible. C'est le cas dans cet exemple.

La partie haute de cette vue peut paraître surprenante, car les valeurs de *DistRFLMax* sont alors importantes. Le point le plus élevé correspond à un vol d'essai d'Airbus. Le RFL est le 70, alors que cet avion est monté au niveau 410, soit un écart de 340. En effet, il est logique et courant que les vols d'essai aient des profils de trajectoires sensiblement différents de celui des vols communs. La couleur blanche du vol indique qu'il s'agit d'un vol privé. Ce vol a bien un indicatif d'appel Airbus, et non celui d'une compagnie aérienne.

Nous illustrons maintenant, par la figure 14.24, l'inclusion, dans le groupe des 34 règles, de la règle sélectionnée dans la figure 14.20 (b). La configuration automatique de la visualisation lors de la sélection des 34 règles apparaît dans la vue (a), sachant que nous avons fait ressortir

Affaire	
Cargo	
Etat	
Low Cost	
Normal	
Privé	

TABLE 14.3 – Assignation automatique des couleurs aux types d'exploitants aéronautiques.

les données concernées par ces règles en affectant la taille des points au lift moyen. Dans la vue (b), une seule règle est sélectionnée. L'ensemble des points particularisés est bien un sous-ensemble des points de la vue (a). Une vision d'ensemble des données dans la vue (c) montre que le réglage automatique délimite bien les données concernées par la règle unique.

Paramétrages automatiques de la visualisation à partir de l'ensemble des règles

La dernière partie de ce scénario concerne le paramétrage automatique de la visualisation des données initiales à partir de toutes les règles (Cf. Chapitre 5.2). A partir de ViewerSettings, nous choisissons le nombre de clusters de règles égal à 10, ainsi que le mode de visualisation 2D. La figure 14.25 montre quatre visualisations automatiques qui s'avèrent être intéressantes. L'assignation des attributs aux variables visuelles est incrustée en haut à droite de chacune d'elles. La couleur utilisée est un gradient du rouge au blanc, en passant par l'orange et le jaune. Il est indiqué en haut de la figure.

La visualisation (a) montre *DistRFLMax* en fonction de *RFL*, comme nous l'avons déjà vu en sélectionnant 34 règles. Elle concerne les valeurs de *RFL* supérieures à 20 et toutes les valeurs de *DistRFLMax*. En d'autres termes, l'ensemble des données sélectionnées est présentée. La taille des points correspond à la distance entre la position de l'avion et la route calculée par le CAUTRA. L'homogénéité globale de cette variable visuelle indique qu'elle varie assez peu. Cependant, quelques gros écarts apparaissent très distinctement. A l'aide de ResultsViewer, nous cherchons alors les règles contenant des 1-itemsets relatifs à l'attribut *DistRoute*. Nous constatons que, parmi les cinq clusters de *DistRoute*, seuls les 1-itemsets $DistRoute = 0$ et $DistRoute = 1$ apparaissent dans les règles. Les gros points de la vue (a), correspondant à des valeurs élevées de *DistRoute*, ne concernent donc aucune règle. En poursuivant l'étude, par la considération des 1-itemsets $DistRoute = 1$, nous mettons en évidence la règle $RFL = 3\ DistRoute = 1 \Rightarrow DistRFLMax = 0$. Elle signifie : si le RFL est compris entre 310 et 430, et si l'écart horizontal de trajectoire est compris entre 6 et 20 Nm, alors l'écart maximal entre le niveau de vol maximal et le RFL est inférieur à 6000 Ft. Cette visualisation (a) permet ainsi de découvrir l'existence de règles d'associations mettant en relations les écarts de niveaux et de trajectoire.

La visualisation (b) montre *DistRFLMax* en fonction de la latitude de l'aéroport de départ. Le lien entre ces attributs a déjà été abordé (Cf. Figure 14.21). La diminution de la taille des

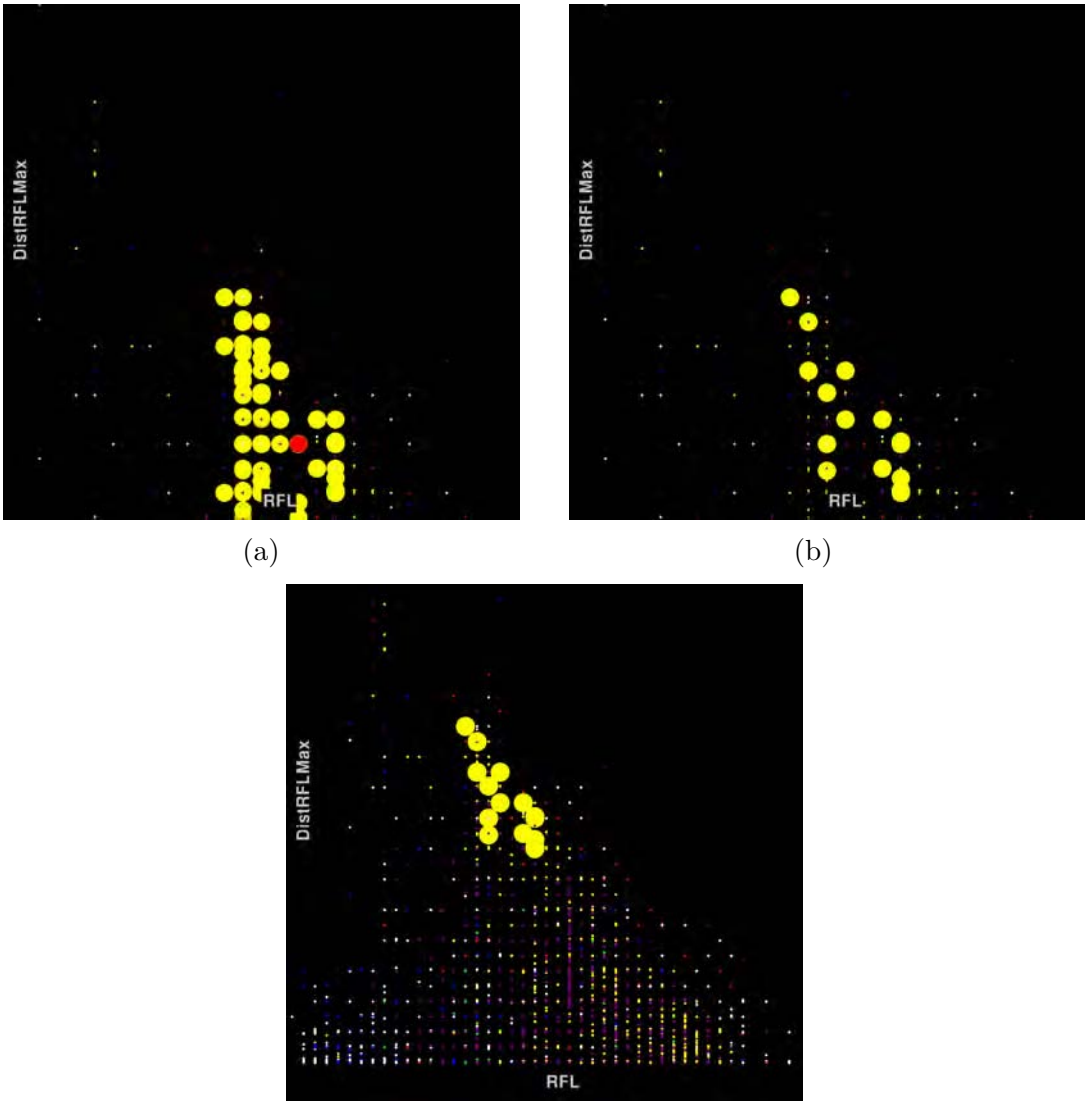
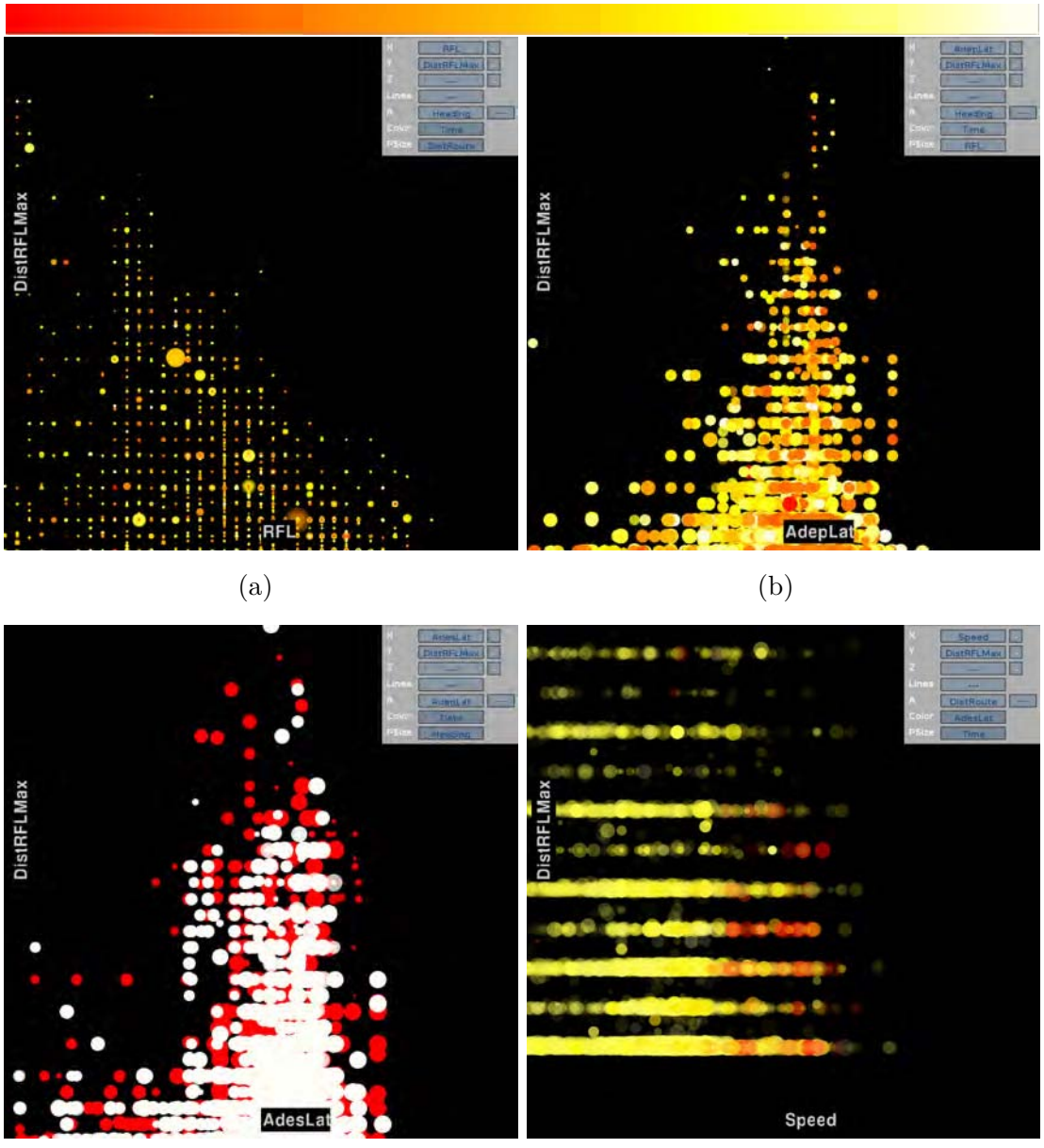


FIGURE 14.24 – Comparaison des données concernées par la sélection de 34 règles (a) et par la sélection d'une de ces règles (b) et (c).



(a) (b)

(c) (d)

FIGURE 14.25 – Paramétrage automatique de la visualisation à partir de clusters de règles.

points au fur et à mesure que *DistRFLMax* augmente, est une autre manière d'illustrer que plus le RFL est petit, plus *DistRFLMax* peut être grand (Cf. Figure 14.23). Comme dans la vue (a), l'heure et le cap sont respectivement assignés à la couleur et à l'alpha. Les variations de ce dernier sont très peu visibles.

La visualisation (c) rappelle la précédente, parce que les assignations de la variable visuelle X concernent des latitudes d'aéroports. Un intérêt de celle-ci est de pouvoir détecter d'éventuels effets saisonniers, parce que la date est assignée à la couleur. Ainsi, les données relatives au mois d'août sont rouges et celles de novembre sont blanches. Les enveloppes globales des données sont similaires, mais nous pouvons constater qu'il semble y avoir plus d'aéroports de destinations ayant des latitudes basses en août qu'en novembre. De plus, les valeurs élevées de *DistRFLMax* sont plus nombreuses en août également. Cela s'explique par un plus fort trafic vers des régions chaudes, en période estivale, notamment de vols charter et low cost.

Dans la visualisation (d), *DistRFLMax* est surtout dépendant de la vitesse, mais il ne concerne que les valeurs de vitesse les plus élevées. L'importance de l'heure et de la latitude de l'aéroport de départ apparaît également ici. Un autre intérêt de cette visualisation est dans l'affectation de la transparence. Souvent, l'affectation d'un attribut à cette variable visuelle peut s'avérer délicate (Cf. Chapitre 2.2). Cependant, malgré l'effet cumulatif de la transparence, certains points apparaissent avec une opacité maximale, ce qui indique des écarts élevés entre la trajectoire et la route prévue.

14.2.4 Conclusion du scénario 2

Dans ce second scénario, nous avons exploité des données IMAGE qui fédèrent des informations de trafic issues des cinq CRNA. A l'issue du prétraitement, nous avons, dans un premier temps, et à partir de l'ensemble de la base, extrait des règles d'association reliant les informations de position, de vitesse et de cap des avions. Leur analyse a permis de faire ressortir, d'une part les mouvements sur les aéroports parisiens et les phases de décollage et d'atterrissage, et, d'autre part, les trajectoires d'arrivées et de départ durant les phases d'approche. Dans un second temps, nous avons extrait des règles à partir d'un sous-ensemble de la base, en considérant l'écart maximal entre le RFL et le niveau maximum des avions. Cela a servi de support à l'illustration du paramétrage automatique de la visualisation, à partir de règles sélectionnées, puis à partir de clusters de règles calculés par le système.

Le lien entre l'espace des données et l'espace des règles a été montré, en agissant sur l'un ou l'autre, en fonction des besoins de l'utilisateur, et en exploitant les possibilités des différents outils de la plate-forme Videam. Ainsi, l'affectation des mesures de qualité aux variables visuelles a été plusieurs fois mise en œuvre pour faire ressortir des données impliquées dans des règles sélectionnées, tout en atténuant les autres. Cela a permis d'extraire des informations intéressantes et pertinentes d'un point de l'exploitation des données aéronautiques. Pour cela, chaque mise en exergue de données a donné lieu à une interprétation, mettant en évidence des zones dans lesquelles elles sont fortement liées au regard de considérations de trafic et de pratiques des exploitants aériens.

Chapitre 15

Conclusion sur l'application aux données aéronautiques

Nous avons illustré, dans cette partie, les liens entre l'approche algorithmique et l'exploration visuelle des données, des itemsets et des règles (Cf. Partie IV), ainsi que les visualisations des résultats algorithmiques (Cf. Partie V). Dans le cadre de deux scénarios, nous nous sommes appuyés sur des données d'archivage récentes, issues des systèmes français de navigation aérienne, portant sur la description des espaces aériens et du trafic qui y circule, qu'il soit au départ ou à destination de la France, mais également en transit.

La plate-forme Videam a ainsi été présentée. A l'issue d'une phase initiale de prétraitement, les données multidimensionnelles sont explorées dans DataViewer. L'utilisateur effectue des choix d'assignation d'attributs aux variables visuelles, de configuration de l'interface et de paramétrage de la fouille automatique de données. A partir de cela, l'algorithme extrait des itemsets et des règles d'association, dont la structure peut être contrainte par la visualisation des données. Les résultats algorithmiques sont ensuite explorés à l'aide de ResultsViewer et RulesList, qui offrent une représentation des itemsets et des règles sous forme graphiques et textuelles. A partir de ces applications, l'utilisateur peut enrichir la visualisation des données initiales, grâce aux mesures associées aux itemsets et aux règles. Il peut également configurer automatiquement cette visualisation en fonction d'une sélection de règles, ou de l'ensemble des règles qui sont regroupées de manière algorithmique.

Cette exploration multi-spatiale des données repose sur le rôle central de l'utilisateur qui, à plusieurs moments du processus, entreprend des choix de sélection, de configuration et d'interaction entre les deux espaces. De plus, il lui est possible de réitérer le processus, à partir d'une autre partie de l'espace des données, afin d'affiner la recherche d'informations. La souplesse d'utilisation de la plate-forme est due à son caractère modulaire qui repose sur un bus logiciel, grâce auquel, les briques, constituant Videam, peuvent être activées ou pas, changées et éventuellement enrichies de nouvelles fonctionnalités.

Tout au long de l'illustration, nous avons précisé les durées d'exécution des algorithmes,

pour faire ressortir le caractère temps réel de l'exploration. Les algorithmes Eclat et k-means demandent peu de temps, notamment grâce aux calculs en parallèle du multithreading. Les durées les plus longues que nous avons constatées concernent l'optimisation du graphe circulaire des itemsets. Une évolution de cette plate-forme pourrait porter sur l'optimisation de ce traitement. Par ailleurs, les données en entrée de Videam sont sous forme de fichiers, et il pourrait être judicieux d'adapter une connexion entre la plate-forme et une base de données pour les obtenir par le biais de requêtes.

Septième partie

Conclusion

Le fort accroissement de la quantité de données générées quotidiennement, auquel nous assistons depuis plusieurs années, est un phénomène qui semble n'être qu'à ses débuts. En effet, à l'ère de l'objet connecté, qu'il soit PC, smartphone, montre et demain vêtement, la production de données constitue un flot continu. Aujourd'hui, ce n'est plus la capacité de stockage qui en fixe les limites, mais la capacité de traitement. De nombreux domaines se penchent sur cette mine d'informations, et cela à divers titres, comme le marketing, la sécurité, les problématiques scientifiques, etc.

Parmi ces domaines, la gestion du trafic aérien n'échappe pas à cette tendance. Les événements récents, comme le crash du vol AF447 et la disparition du vol MH370, ont rappelé l'importance croissante des données, et les difficultés de leur recueil. Une conséquence sera l'augmentation du flux échangé entre les appareils embarqués et le sol. Quant aux futurs systèmes européen et nord-américain de gestion du trafic aérien, ils ont pour objectifs une meilleure connaissance des trajectoires des avions, avec une prédictibilité la plus haute possible, afin d'optimiser les flux et diminuer ainsi les délais et les coûts. Pour cela, ils nécessiteront de plus grands échanges d'informations entre les systèmes. Ces deux exemples montrent que la gestion des données est devenue un véritable enjeu dans ce domaine également. Cela est confirmé par la création d'un service centralisé européen, qui, d'ici 2019, proposera une base de données aéronautiques européenne pour l'ensemble des pays membres d'Eurocontrol. Son rôle sera d'adapter la fourniture d'informations à la gestion du trafic aérien et de fournir des services.

Une approche, permettant d'appréhender cet afflux de données et leur traitement, est le Visual Analytics, qui a vu le jour, dans le contexte des attentats de septembre 2001, suite à la création du DHS [1] pour prévenir toute menace envers les intérêts américains. Dans cette mouvance, le programme VisMaster a été lancé en 2008, par la Commission Européenne, pour mettre en commun les efforts de la recherche et de l'industrie, afin d'étudier cette science du raisonnement analytique facilité par les interfaces visuelles interactives. Il s'agit ainsi d'un domaine pluridisciplinaire visant à combiner les capacités de l'homme et du système, en exploitant le meilleur de chacun. Du côté de l'homme, se trouvent la connaissance et la créativité. Quant au système, il est performant pour ses capacités de stockage et de calcul.

Dans ce contexte, ce travail de thèse a eu pour objectif d'étudier des pistes de recherche pour aborder, selon une approche anthropocentrée, les défis soulevés par le Visual Analytics, appliqués aux données aéronautiques. Notre problématique était d'explorer des voies pour améliorer la collaboration entre l'homme et le système, en combinant leurs apports respectifs, tout en laissant à l'humain la maîtrise globale du processus d'extraction de données. Dans les approches décrites par Bertini & Lalanne [Bertini 09], nous nous situons dans la troisième catégorie, intégrant la visualisation et la fouille (*Integrated Visualization & Mining*), selon le modèle *White Box*, dans lequel l'homme et la machine coopèrent durant toutes les étapes du processus.

Nous l'avons ramenée à quatre questions portant sur le pilotage de l'algorithme par l'exploration visuelle, la représentation des résultats algorithmiques, leur exploration interactive

1. US Department of Homeland Security

avec celle des données, et la mise en œuvre d'un système ouvert et évolutif pour accueillir notre étude. Ces questions sont reprises dans les contributions ci-dessous.

15.1 Contributions

15.1.1 Pilotage du Data Mining par la visualisation

Les itemsets et les règles d'association sont les conséquences des choix de l'utilisateur. Il les réalise en assignant les attributs aux variables visuelles, en filtrant et en sélectionnant les données, et en configurant l'algorithme. La structure des règles est ainsi conditionnée par la visualisation et l'exploration des données initiales, en respectant les niveaux d'organisation des variables visuelles.

Une formalisation a été proposée, étendant le modèle de Card & Mackinlay de caractérisation de la visualisation. Elle prend en compte la restriction de l'espace des points à la partie visualisée, puis à la partie sélectionnée.

15.1.2 Enrichissement et configuration automatique de la visualisation des données par l'exploration des itemsets et des règles

L'exploration de l'espace des règles d'association enrichit l'espace des données de plusieurs manières :

- Enrichissement des données par l'assignation de mesures, associées aux itemsets et aux règles, aux variables visuelles ;
- Configuration automatique de la visualisation à partir d'une sélection de règles d'association par l'utilisateur ;
- Configuration automatique de la visualisation à partir de clusters de règles calculés par le système, selon un algorithme que nous proposons.

Ce retour, des résultats algorithmiques vers la visualisation des données, est ensuite source d'un affinement de l'exploration des données, qui donne lieu, de manière itérative, à des nouveaux cycles d'analyse, des données vers les résultats, puis des résultats vers les données.

15.1.3 Graphe circulaire et optimisé de présentation et d'exploration des itemsets

Nous proposons une nouvelle manière de présenter les itemsets, dans un graphe constitué de cercles concentriques, dont la structure montre leur construction progressive, à partir des attributs présentés de manière unique. Des optimisations de la structure de ce graphe, par la taille des cercles et la disposition des nœuds, le rendent plus lisible en regroupant les itemsets partageant de l'information commune, et en les éloignant dans le cas contraire. Des techniques

issues du domaine de l'InfoVis en améliorent ensuite l'exploration, notamment par le bundling et l'assignation de mesures aux variables visuelles.

15.1.4 Exploration multidimensionnelle des règles d'association

Les règles d'association sont explorées en exploitant la caractérisation des visualisations et la sémiologie graphique. En assignant les mesures de qualité aux variables visuelles, l'utilisateur peut explorer l'espace des règles en fonction de critères qui lui incombent. Par une approche exploratoire et progressive, le Rules Focusing permet un affinement de l'étude des règles.

L'exploration est réalisée de manières globale et locale par deux visualisations, scatter plot et textuelle, reliées entre elles pour assurer le linking. Le spectre exploratoire couvre ainsi de la totalité d'une grande quantité de règles jusqu'aux détails de l'une d'entre elle, par la connaissance de son intitulé et des mesures associées.

15.1.5 Plate-forme multi-spatiale et modulaire d'exploration des données et des résultats d'algorithme

Videam présente, dans une plate-forme que nous avons souhaité cohérente, un double espace d'exploration de données hétérogènes et de résultats d'algorithmes. Ces espaces sont configurables, notamment par l'assignation des attributs des données et des mesures des règles aux variables visuelles. L'exploitation du bus logiciel Ivy rend Videam hautement configurable, dans un environnement hétérogène et éventuellement distant. Les différentes briques qui la constituent ne sont pas nécessaires pour toutes les tâches, ce qui permet d'obtenir plusieurs configurations en fonction des besoins. De plus, d'autres fonctionnalités, algorithmiques ou sous forme d'IHM, peuvent être intégrées à l'ensemble sans avoir à modifier l'existant.

La plate-forme montre le rôle central de l'utilisateur dans le processus global de recherche d'informations, par toutes les possibilités d'intervention qu'il peut avoir dans celui-ci. Il agit, de manière itérative, sur l'exploration des données, sur le paramétrage et le choix de l'algorithme de Data Mining, sur la structure des règles extraites et sur le mécanisme de retour des règles vers les données.

15.2 Perspectives

Nos contributions ont porté essentiellement sur le lien entre la visualisation des données et la fouille de données algorithmiques. Dans la suite de nos travaux, nous envisageons d'approfondir ces deux aspects.

Des choix ont dû être faits pour les algorithmes, comme la configuration automatique de la visualisation des données. Celle-ci repose sur un calcul de l'importance des 1-itemsets dans les règles, en fonction de l'évolution du support et de la confiance. Il serait intéressant d'améliorer la quantification de cette importance en se basant sur d'autres mesures caractérisant les règles,

comme celles qui prennent en compte les contre-exemples. De plus, nous avons contraint cette configuration par la variable visuelle Y, en n'appliquant l'algorithme qu'aux prémisses. Une suite de ce travail serait de permettre des conclusions non contraintes, ainsi qu'éventuellement des règles de type *many-to-two*. De même, pour les autres algorithmes ou leur paramétrage, comme celui-ci bundling, d'autres solutions ou des variantes permettraient peut-être d'obtenir de meilleurs résultats.

Nous envisageons par ailleurs d'étudier une amélioration de la représentation des résultats algorithmiques. Afin d'accroître la lisibilité du graphe circulaire, nous pourrions procéder à des regroupements d'itemsets, en utilisant par exemple des méthodes similaires à celle qui a été utilisée pour le regroupement des règles d'association (Cf. Chapitre 5.2). L'exploration des règles selon deux modes de représentations peut être fusionnée par l'utilisation de zooms proposant un enrichissement de l'information selon le rapprochement (Cf. Chapitre 3.1.3).

Les perspectives portent également sur des protocoles de validation, par exemple pour évaluer la configuration automatique des visualisations à partir des règles d'association. D'une manière plus générale, la question de la validation de la plate-forme Videam se pose. Dès 2009, Puolamäki & Bertone [Puolamäki 10] se demandaient comment juger une contribution au Visual Analytics, parce qu'il est difficile de mesurer ce type d'approche. Le métier de *Data Miner* n'existe pas encore, au sein des organismes de gestion du trafic aérien. Il nécessite des connaissances multiples, notamment techniques, statistiques et du fonctionnement des systèmes et des données qu'ils produisent. C'est pourquoi, l'évaluation globale de Videam s'avère délicate et peut commencer par une validation des différents aspects, comme nous l'avons fait pour l'optimisation de la vue circulaire des itemsets.

Notre travail a pour domaine d'application les données aéronautiques, dont l'évolution, par la richesse de leur contenu, par leur quantité, et par les besoins des futurs systèmes de gestion de la navigation aérienne, montre leur importance croissante. Des perspectives sont envisageables au regard de ce domaine.

Les jeux de données auxquels nous avons eu accès portent sur des durées qui peuvent s'avérer insuffisantes en fonction des besoins de l'étude. Un passage à l'échelle, prenant en compte plusieurs mois, voire plusieurs années, peut être ainsi nécessaire. Pour cela, de nouvelles briques seraient à envisager dans la plate-forme, si ce n'est la modification de l'existant. Dans tous les cas, une amélioration des performances serait à étudier, notamment dans l'optimisation du graphe circulaire, afin de rendre plus fluide l'interaction entre les deux espaces.

L'accès aux données et leur prétraitement seraient à améliorer. En effet, elles se présentent actuellement sous forme de fichiers. Les obtenir par requêtes à une base de données permettrait de prendre en compte une partie du prétraitement et de se connecter à des bases en ligne. La question se poserait alors de l'ajout de nouvelles données dans des classes existantes, ce qui nécessiterait un apprentissage supervisé, à la différence de l'utilisation de k-means. Une plate-forme modulaire comme Videam permet l'ajout de ce type de fonctionnalité sans avoir à procéder à une refonte du système global. De plus, une connexion à une base de données permettrait d'y stocker les classes ainsi calculées, pour y accéder éventuellement à distance.

Notre approche consiste à analyser des données a posteriori dans un double espace d'exploration. Une ouverture vers d'autres systèmes de gestion ou d'analyse du trafic aérien pourrait être profitable. Par exemple, dans le cadre d'une analyse d'incident ou d'accident, en rejouant une situation sur une image radar, une interaction bidirectionnelle avec Videam permettrait d'enrichir l'information provenant des enregistrements, en s'appuyant sur un historique lié aux paramètres composant cette situation.

Nous espérons que ce travail de thèse aura permis de montrer le potentiel que constitue la manne de données aéronautiques produites en permanence, ainsi que l'intérêt du Visual Analytics pour l'appréhender.

Références bibliographiques

[Agrawal 93] R. Agrawal, T. Imielinski et A. Swami. *Mining Association Rules Between Sets of Items in Large Databases*. Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data, pages 207–216, New York, NY, USA, 1993. ACM.

[Agrawal 94a] R. Agrawal et R. Srikant. *Fast Algorithms for Mining Association Rules*. J. B. Bocca, M. Jarke et C. Zaniolo, éditeurs, VLDB '94, Proceedings of 20th International Conference on Very Large Data Bases, September 12-15, 1994, Santiago de Chile, Chile, pages 487–499. Morgan Kaufmann, 1994.

[Agrawal 94b] R. Agrawal et R. Srikant. *Fast Algorithms for Mining Association Rules in Large Database*. Research Report RJ 9839. IBM Almaden Research Center, San Jose, California, 1994.

[Azé 02] J. Azé et Y. Kodratoff. *Evaluation de la résistance au bruit de quelques mesures d'extraction de règles d'association.* D. Hérin et D. A. Zighed, éditeurs, EGC, volume 1 de *Extraction des Connaissances et Apprentissage*, pages 143–154. Hermes Science Publications, 2002.

[Bayardo 98] R. J. Bayardo Jr. *Efficiently Mining Long Patterns from Databases*. Proceedings of the 1998 ACM SIGMOD International Conference on Management of Data, SIGMOD '98, pages 85–93, New York, NY, USA, 1998. ACM.

[Bayardo 99] R. J. Bayardo Jr. et R. Agrawal. *Mining the Most Interesting Rules*. Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '99, pages 145–154, New York, NY, USA, 1999. ACM.

[BEA 12] BEA. *Rapport final - Accident survenu le 1er juin 2009 à l'Airbus A330-203 immatriculé F-GZCP exploité par Air France vol AF 447 Rio de Janeiro - Paris*. Jul 2012.

[Becker 87] R. A. Becker et W. S. Cleveland. *Brushing scatterplots*. Technometrics, vol. 29, pages 127–142, May 1987.

[Ben Said Guefrech 12] Z. Ben Said Guefrech. *A virtual reality-based approach for interactive and visual mining of association rules.* PhD thesis, Université UNAM, 2012.

[Benzécri 77] J.-P. Benzécri. *Histoire et préhistoire de l'analyse des données. Partie V : l'analyse des correspondances.* vol. 2, pages 9–40, 1977.

[Bertin 67] J. Bertin. Semiologie graphique. les diagrammes, les réseaux, les cartes. Gauthier-Villars, 1967.

[Bertin 83] J. Bertin. Semiology of graphics. University of Wisconsin Press, 1983.

[Bertini 09] E. Bertini et D. Lalanne. *Surveying the Complementary Role of Automatic Data Analysis and Visualization in Knowledge Discovery.* Proceedings of the ACM SIGKDD Workshop on Visual Analytics and Knowledge Discovery : Integrating Automated Analysis with Interactive Exploration, VAKD '09, pages 12–20, New York, NY, USA, 2009. ACM.

[Blanchard 03] J. Blanchard, F. Guillet et H. Briand. *A User-driven and Quality-oriented Visualization for Mining Association Rules.* Proceedings of the Third IEEE International Conference on Data Mining, ICDM '03, pages 493–, Washington, DC, USA, 2003. IEEE Computer Society.

[Blanchard 07] J. Blanchard, F. Guillet et H. Briand. *Interactive visual exploration of association rules with rule-focusing methodology.* Knowl. Inf. Syst., vol. 13, no. 1, pages 43–75, 2007.

[Bothorel 11] G. Bothorel, M. Serrurier et C. Hurter. *Utilisation d'outils de visual data mining pour l'exploration d'un ensemble de règles d'association.* 23rd French Speaking Conference on Human-Computer Interaction, IHM '11, pages 12 :1–12 :4, New York, NY, USA, 2011. ACM.

[Bothorel 13a] G. Bothorel, M. Serrurier et C. Hurter. *From Visualization to Association Rules : an automatic approach.* R. Durikovic et H. Rushmeier, éditeurs, 29th Spring Conference on Computer Graphics Proceedings. Comenius University, Bratislava, 2013.

[Bothorel 13b] G. Bothorel, M. Serrurier et C. Hurter. *Mining aeronautical data by using visualized driven rules extraction approach.* ISIATM 2013, 2nd International Conference on Interdisciplinary Science for Innovative Air Traffic Management, Toulouse : France. HAL, 2013.

[Bothorel 13c] G. Bothorel, M. Serrurier et C. Hurter. *Visualization of Frequent Itemsets with Nested Circular Layout and Bundling Algorithm.* G. Bebis, R. Boyle, B. Parvin, D. Koracin, B. Li, F. Porikli, V. B. Zordan, J. T. Klosowski, S. Coquillart, X. Luo, M. Chen et D. Gotz, éditeurs, ISVC (2), volume 8034 de *Lecture Notes in Computer Science*, pages 396–405. Springer, 2013.

[Braga 02] D. Braga, A. Campi, M. Klemettinen et P. Lanzi. *Mining Association Rules from XML Data*. 2454, pages 21–30. Springer Berlin Heidelberg, 2002.

[Brin 97a] S. Brin, R. Motwani et C. Silverstein. *Beyond Market Baskets : Generalizing Association Rules to Correlations*. Proceedings of the 1997 ACM SIGMOD international conference on Management of data, SIGMOD '97, pages 265–276, New York, NY, USA, 1997. ACM.

[Brin 97b] S. Brin, R. Motwani, J. D. Ullman et S. Tsur. *Dynamic Itemset Counting and Implication Rules for Market Basket Data*. Proceedings of the 1997 ACM SIGMOD International Conference on Management of Data, SIGMOD '97, pages 255–264, New York, NY, USA, 1997. ACM.

[Bruzzese 04] D. Bruzzese et P. Buono. *Combining Visual Techniques for Association Rules Exploration*. Proceedings of the Working Conference on Advanced Visual Interfaces, AVI '04, pages 381–384, New York, NY, USA, 2004. ACM.

[Bruzzese 08] D. Bruzzese et C. Davino. *Visual Data Mining*. chapitre Visual Mining of Association Rules, pages 103–122. Springer-Verlag, Berlin, Heidelberg, 2008.

[Buisson 02] M. Buisson, A. Bustico, S. Chatty, F.-R. Colin, Y. Jestin, S. Maury, C. Mertz et P. Truillet. *Ivy : Un Bus Logiciel Au Service Du Développement De Prototypes De Systèmes Interactifs*. Proceedings of the 14th French-speaking Conference on Human-computer Interaction (Conférence Francophone Sur L'Interaction Homme-Machine), IHM '02, pages 223–226, New York, NY, USA, 2002. ACM.

[Caliński 74] T. Caliński et J. Harabasz. *A dendrite method for cluster analysis*. Communications in Statistics-Simulation and Computation, vol. 3, no. 1, pages 1–27, 1974.

[Card 97] S. K. Card et J. Mackinlay. *The structure of the information visualization design space*. Proceedings of the 1997 IEEE Symposium on Information Visualization (InfoVis '97), pages 92–, Washington, DC, USA, 1997. IEEE Computer Society.

[Card 99] S. K. Card, J. D. Mackinlay et B. Shneiderman, éditeurs. Readings in information visualization : using vision to think. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1999.

[Carr 86] D. B. Carr, R. J. Littlefield et W. L. Nichloson. *Scatterplot Matrix Techniques for Large N*. Proceedings of the Seventeenth Symposium on the Interface of Computer Sciences and Statistics on Computer Science and Statistics, pages 297–306, New York, NY, USA, 1986. Elsevier North-Holland, Inc.

[Cerny 85] V. Cerny. *Thermodynamical Approach to the Traveling Salesman Problem : An Efficient Simulation Algorithm.* Journal of Optimization Theory and Applications, vol. 45, pages 41–51, 1985.

[Chen 04] C. Chen. Information visualization : Beyond the horizon. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2004.

[Chernoff 73] H. Chernoff. *The Use of Faces to Represent Points in K-Dimensional Space Graphically.* Journal of the American Statistical Association, vol. 68, no. 342, pages 361–368, 1973.

[Chevrin 07] V. Chevrin, O. Couturier, E. M. Nguifo et J. Rouillard. *Recherche anthropocentrée de règles d'association pour l'aide à la décision.* 2007.

[Cleveland 85] W. S. Cleveland. The elements of graphing data. Wadsworth Publ. Co., Belmont, CA, USA, 1985.

[Cleveland 88] W. C. Cleveland et M. E. McGill. Dynamic graphics for statistics. CRC Press, Inc., Boca Raton, FL, USA, 1st edition, 1988.

[Cleveland 93] W. S. Cleveland. Visualizing data. Hobart Press, 1993.

[Clos 07] K. J. Clos, W. Pedryez, R. W. Swiniarski et L. A. Kurgan. Data mining : A knowledge discovery approach. Springer, 2007.

[Cornuéjols 10] A. Cornuéjols et L. Miclet. Apprentissage artificiel : Concepts et algorithmes. Eyrolles, Jun 2010.

[Couturier 04] O. Couturier, E. M. Nguifo et B. Noiret. *Recherche de règles d'association hiérarchiques par une approche anthropocentrée.* G. Hébrail, L. Lebart et J.-M. Petit, éditeurs, EGC, volume RNTI-E-2 de *Revue des Nouvelles Technologies de l'Information*, pages 567–572. Cépaduès Editions, 2004.

[Couturier 05] O. Couturier. *Contribution à la fouille de données : règles d'association et interactivité au sein d'un processus d'extraction de connaissances dans les données.* PhD thesis, Université d'Artois, Lens, 2005.

[Couturier 06] O. Couturier, J. Rouillard et V. Chevrin. *Une approche hybride pour une meilleure visualisation de grands ensembles de règles d'association.* 10ème Conférence Francophone ERGO-IA (ERGOIA '06), Biarritz, France, Oct 2006.

[Couturier 07a] O. Couturier, T. Hamrouni, S. B. Yahia et E. M. Nguifo. *A scalable association rule visualization towards displaying large amounts of knowledge.* Proceedings of the 11th International Conference Information Visualization, pages 657–663, Washington, DC, USA, 2007. IEEE Computer Society.

[Couturier 07b] O. Couturier, J. Rouillard et V. Chevrin. *An Interactive Approach to Display Large Sets of Association Rules.* HCI (8), pages 258–267, 2007.

[Cox 97] M. Cox et D. Ellsworth. *Application-controlled Demand Paging for Out-of-core Visualization.* Proceedings of the 8th Conference on Visualization, VIS '97, pages 235–, Los Alamitos, CA, USA, 1997. IEEE Computer Society Press.

[Croes 58] G. Croes. *A Method for Solving Traveling Salesman Problems.* Operation Research, vol. 6, pages 791–812, 1958.

[Cui 08] W. Cui, H. Zhou, H. Qu, P. C. Wong et X. Li. *Geometry-Based Edge Clustering for Graph Visualization.* IEEE Transactions on Visualization and Computer Graphics, vol. 14, no. 6, pages 1277–1284, Nov 2008.

[Dickerson 03] M. Dickerson, D. Eppstein, M. T. Goodrich et J. Y. Meng. *Confluent Drawings : Visualizing Non-planar Diagrams in a Planar Way.* G. Liotta, éditeur, Graph Drawing, volume 2912 de *Lecture Notes in Computer Science*, pages 1–12. Springer, 2003.

[d'Ocagne 85] M. d'Ocagne. Coordonnées parallèles et axiales. 1885.

[Don 07] A. Don, E. Zheleva, M. Gregory, S. Tarkan, L. Auvil, T. Clement, B. Shneiderman et C. Plaisant. *Discovering Interesting Usage Patterns in Text Collections : Integrating Text Mining with Visualization.* Proceedings of the Sixteenth ACM Conference on Conference on Information and Knowledge Management, CIKM '07, pages 213–222, New York, NY, USA, 2007. ACM.

[DTI 05] DTI. *SYNOPSIS PRESAGE - Manuel d'utilisation.* 2005.

[DTI 10] DTI. *COURAGE - Manuel d'utilisation.* 2010.

[DTI 12] DTI. *Dossier de Définition d'Interface PREVI - Fichiers COURAGE.* 2012.

[Dwyer 07] T. Dwyer, K. Marriott et M. Wybrow. *Integrating edge routing into force-directed layout.* Proceedings of the 14th international conference on Graph drawing, GD '06, pages 8–19, Berlin, Heidelberg, 2007. Springer-Verlag.

[Ellis 07] G. Ellis et A. Dix. *A Taxonomy of Clutter Reduction for Information Visualisation.* IEEE Transactions on Visualization and Computer Graphics, vol. 13, no. 6, pages 1216–1223, Nov 2007.

[Ersoy 11] O. Ersoy, C. Hurter, F. Paulovich, G. Cantareiro et A. Telea. *Skeleton-Based Edge Bundling for Graph Visualization.* IEEE Transactions on Visualization and Computer Graphics, vol. 17, no. 12, pages 2364–2373, Dec 2011.

[Ertek 06] G. Ertek et A. Demiriz. *A Framework for Visualizing Association Mining Results.* pages 593–602. ISCIS 2006, Springer-Verlag, 2006.

[Escovar 05] E. L. G. Escovar, M. Biajiz et M. T. P. Vieira. *SSDM : A Semantically Similar Data Mining Algorithm.* C. A. Heuser, éditeur, SBBD, pages 265–279. UFU, 2005.

[Eurocontrol 13] Eurocontrol. *Eurocontrol Seven-Year Forecast*. Feb 2013.

[Fayyad 96a] U. Fayyad, G. Piatetsky-shapiro et P. Smyth. *From Data Mining to Knowledge Discovery in Databases*. AI Magazine, vol. 17, pages 37–54, 1996.

[Fayyad 96b] U. M. Fayyad, G. Piatetsky-Shapiro et P. Smyth. *Advances in Knowledge Discovery and Data Mining*. chapitre From Data Mining to Knowledge Discovery : An Overview, pages 1–34. American Association for Artificial Intelligence, 1996.

[Fekete 04] J.-D. Fekete. *The InfoVis Toolkit*. Proceedings of the IEEE Symposium on Information Visualization, INFOVIS '04, pages 167–174, Washington, DC, USA, 2004. IEEE Computer Society.

[Frawley 92] W. J. Frawley, G. Piatetsky-shapiro et C. J. Matheus. *Knowledge Discovery in Databases : an Overview*, 1992.

[Furnas 86] G. W. Furnas. *Generalized Fisheye Views*. Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '86, pages 16–23, New York, NY, USA, 1986. ACM.

[Gansner 07] E. R. Gansner et Y. Koren. *Improved circular layouts*. Proceedings of the 14th international conference on Graph drawing, GD '06, pages 386–398, Berlin, Heidelberg, 2007. Springer-Verlag.

[Gansner 11] E. R. Gansner, Y. Hu, S. North et C. Scheidegger. *Multilevel agglomerative edge bundling for visualizing large graphs*. Proceedings of the 2011 IEEE Pacific Visualization Symposium, Pacific Vis '11, pages 187–194, Washington, DC, USA, 2011. IEEE Computer Society.

[Geng 06] L. Geng et H. J. Hamilton. *Interestingness Measures for Data Mining : A Survey*. ACM Comput. Surv., vol. 38, no. 3, Sep 2006.

[Gens 13] F. Gens. *IDC Worldwide ICT Industry Predictions 2014*, 2013.

[Glatz 12] E. Glatz, S. Mavromatidis, B. Ager et X. Dimitropoulos. *Visualizing big network traffic data using frequent pattern mining and hypergraphs*. First IMC Workshop on Internet Visualization (WIV 2012), Boston, Massachusetts, USA, Nov 2012. Springer.

[Goodman 88] R. M. F. Goodman et P. Smyth. *Information-Theoretic Rule Induction*. Proc. of the 8th ECAI, pages 357–362, Munich, Germany, 1988.

[Gras 79] R. Gras. *Contribution à l'étude expérimentale et à l'analyse de certaines acquisitions cognitives et de certains objectifs didactiques en mathématiques*. Thèse d'etat, Université de Rennes I, 1979.

[Gras 10] R. Gras et R. Couturier. *Spécificité de l'A.S.I. par rapport à d'autres mesures de qualité de règles d'association*. 5ème Colloque International sur Analyse Statistique Implicative, pages 175–198, Palerme, Italy, Nov 2010.

[Guillet 07] F. Guillet et H. J. Hamilton. Quality measures in data mining, volume 43 de *Studies in Computational Intelligence*. Springer, 2007.

[Hahsler 11] M. Hahsler et S. Chelluboina. *Visualizing Association Rules in Hierarchical Groups*. 42nd Symposium on the Interface : Statistical, Machine Learning, and Visualization Algorithms (Interface 2011), 2011.

[Han 00] J. Han, J. Pei et Y. Yin. *Mining Frequent Patterns Without Candidate Generation*. Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data, SIGMOD '00, pages 1–12, New York, NY, USA, 2000. ACM.

[Han 04] J. Han, J. Pei, Y. Yin et R. Mao. *Mining Frequent Patterns without Candidate Generation : A Frequent-Pattern Tree Approach*. Data Mining and Knowledge Discovery, vol. 8, no. 1, pages 53–87, Jan 2004.

[Hao 01] M. C. Hao, U. Dayal, M. Hsu, T. Sprenger et M. H. Gross. *Visualization of Directed Associations in e-Commerce Transaction Data*. Proceedings of the 3rd Joint Eurographics - IEEE TCVG Conference on Visualization, EGVISSYM'01, pages 185–192, Aire-la-Ville, Switzerland, Switzerland, 2001. Eurographics Association.

[Hartigan 75] J. A. Hartigan. Clustering algorithms. John Wiley & Sons, Inc., New York, NY, USA, 99th edition, 1975.

[Havre 00] S. Havre, B. Hetzler et L. Nowell. *ThemeRiver : Visualizing Theme Changes over Time*. Proceedings of the IEEE Symposium on Information Vizualization 2000, INFOVIS '00, pages 115–, Washington, DC, USA, 2000. IEEE Computer Society.

[Healey 90] C. G. Healey. *Visualization of Multivariate Data using Preattentive Processing*. PhD thesis, University of Colorado, 1990.

[Healey 08] C. Healey, S. Kocherlakota, V. Rao, R. Mehta et R. St. Amant. *Visual Perception and Mixed-Initiative Interaction for Assisted Visualization Design*. IEEE Transactions on Visualization and Computer Graphics, vol. 14, pages 396–411, Mar 2008.

[Healey 12] C. Healey et J. Enns. *Attention and Visual Memory in Visualization and Computer Graphics*. IEEE Transactions on Visualization and Computer Graphics, vol. 18, no. 7, pages 1170–1188, Jul 2012.

[Herbert 12] G. Herbert. *2012 Energy Summit*, 2012.

[Hofmann 00a] H. Hofmann. Exploring categorical data : interactive mosaic plots. 2000.

[Hofmann 00b] H. Hofmann, A. P. J. M. Siebes et A. F. X. Wilhelm. *Visualizing Association Rules with Interactive Mosaic Plots*. Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge

Discovery and Data Mining, KDD '00, pages 227–235, New York, NY, USA, 2000. ACM.

[Hogan 04] J. Hogan, P. Montague, M. Purvis et C. Steketee, éditeurs. Experiences in building a tool for navigating association rule result sets. CRPIT'04 : Proceedings of the second Australasian workshop on information security, data mining, web intelligence, and software internationalisation, Australian Computer Society, 2004.

[Holten 06] D. Holten. *Hierarchical Edge Bundles : Visualization of Adjacency Relations in Hierarchical Data.* IEEE Transactions on Visualization and Computer Graphics, vol. 12, no. 5, pages 741–748, Sep 2006.

[Holten 09] D. Holten et J. J. van Wijk. *Force-Directed Edge Bundling for Graph Visualization.* Comput. Graph. Forum, vol. 28, no. 3, pages 983–990, 2009.

[Horn 98] W. Horn, C. Popow et L. Unterasinger. *Metaphor graphics to visualize ICU data over time*, 1998.

[Hurter 09] C. Hurter, B. Tissoires et S. Conversy. *FromDaDy : Spreading Aircraft Trajectories Across Views to Support Iterative Queries.* IEEE Transactions on Visualization and Computer Graphics, vol. 15, pages 1017–1024, 2009.

[Hurter 10] C. Hurter, B. Tissoires et S. Conversy. Accumulation as a tool for efficient visualization of geographical and temporal data (geova(t) - geospatial visual analytics : Focus on time interacting with temporal data., guimaraes portugal. 2010.

[Hurter 11] C. Hurter, A. Telea et O. Ersoy. *MoleView : An Attribute and Structure-Based Semantic Lens for Large Element-Based Plots.* IEEE Trans. Vis. Comput. Graph., vol. 17, no. 12, pages 2600–2609, 2011.

[Hurter 12] C. Hurter, O. Ersoy et A. Telea. *Graph Bundling by Kernel Density Estimation.* Comp. Graph. Forum, vol. 31, no. 3pt1, pages 865–874, Jun 2012.

[Hurter 13] C. Hurter, O. Ersoy et A. Telea. *Smooth Bundling of Large Streaming and Sequence Graphs.* Proc. IEEE PacificVis, 2013.

[Hurter 14] C. Hurter, A. R. Taylor, S. Carpendale et A. Telea. *Color Tunneling : Interactive Exploration and Selection in Volumetric Datasets.* PacificVis 2014, IEEE Pacific Visualization Symposium, pages pp 225–232, Yokohama, Japan, Mar 2014. IEEE.

[Ingber 92] L. Ingber et B. Rosen. *Genetic Algorithms and Very Fast Simulated Reannealing : A Comparison.* Math. Comput. Model., vol. 16, no. 11, pages 87–100, Nov 1992.

[Inselberg 81] A. Inselberg. *N-dimensional graphics, Part I - lines and hyperplanes.* Rapport technique, 1981.

[Johanson 76] D. C. Johanson et M. Taieb. *Plio-pleistocene hominid discoveries in Hadar, Ethiopia*. Nature, vol. 260, page 293, 1976.

[Johansson 09] S. Johansson. *Visual Exploration of Categorical and Mixed Data Sets*. Proceedings of the ACM SIGKDD Workshop on Visual Analytics and Knowledge Discovery : Integrating Automated Analysis with Interactive Exploration, VAKD '09, pages 21–29, New York, NY, USA, 2009. ACM.

[Johnson 91] B. Johnson et B. Shneiderman. *Tree-Maps : A Space-filling Approach to the Visualization of Hierarchical Information Structures*. Proceedings of the 2Nd Conference on Visualization, VIS '91, pages 284–291. IEEE Computer Society Press, 1991.

[Keim 95] D. A. Keim, M. Ankerst et H.-P. Kriegel. *Recursive Pattern : A Technique for Visualizing Very Large Amounts of Data*. Proceedings of the 6th Conference on Visualization '95, VIS '95, pages 279–, Washington, DC, USA, 1995. IEEE Computer Society.

[Keim 02] D. A. Keim, M. C. Hao, U. Dayal et M. Hsu. *Pixel Bar Charts : A Visualization Technique for Very Large Multi-attribute Data Sets*. Information Visualization, vol. 1, no. 1, pages 20–34, Mar 2002.

[Keim 03] D. Keim et M. Ward. *Visual Data Mining Techniques*. Intelligent Data Analysis : An Introduction, pages 403–427, 2003.

[Keim 04] D. A. Keim, M. Sips et M. Ankerst. *Visual Data-Mining Techniques*. Visualization Handbook, pages 831–843, 2004.

[Keim 05] D. A. Keim, J. Schneidewind et M. Sips. *FP-Viz : Visual Frequent Pattern Mining*. Proceedings of IEEE Symposium on Information Visualization (InfoVis '05), Poster Paper, 2005.

[Keim 06] D. A. Keim, F. Mansmann, J. Schneidewind et H. Ziegler. *Challenges in visual data analysis*. In Proceedings of the Tenth International Conference on Information Visualization, pages 9–16, 2006.

[Keim 08a] D. Keim, G. Andrienko, J.-D. Fekete, C. Görg, J. Kohlhammer et G. Melançon. *Information Visualization*. chapitre Visual Analytics : Definition, Process and Challenges, pages 154–175. Springer-Verlag, Berlin, Heidelberg, 2008.

[Keim 08b] D. A. Keim, F. Mansmann, J. Schneidewind, J. Thomas et H. Ziegler. *Visual Data Mining*. chapitre Visual Analytics : Scope and Challenges, pages 76–90. Springer-Verlag, Berlin, Heidelberg, 2008.

[Keim 10a] D. Keim, J. Kohlhammer, G. Ellis et F. Mansmann, éditeurs. Mastering the Information Age : Solving Problems with Visual Analytics. VisMaster, 2010.

[Keim 10b] D. A. Keim, F. Mansmann et J. Thomas. *Visual Analytics : How Much Visualization and How Much Analytics ?* SIGKDD Explor. Newsl., vol. 11, no. 2, pages 5–8, May 2010.

[Kian Huat 02] O. Kian Huat, O. Kok Leong, N. Wee Keong et L. Ee-Peng. *CrystalClear : Active Visualization of Association Rules.* In ICDM'02 International Workshop on Active Mining AM2002. Press, 2002.

[Kirkpatrick 83] S. Kirkpatrick, C. D. Gelatt et M. P. Vecchi. *Optimization by simulated annealing.* Science, vol. 220, pages 671–680, 1983.

[Koffka 35] K. Koffka. Principles of gestalt psychology. Routledge & Kegan Paul Ltd, 1935.

[Kuntz 06] P. Kuntz, R. Lehn, F. Guillet et B. Pinaud. *Découverte interactive de règles d'association via une interface visuelle.* P. Kuntz et F. Poulet, éditeurs, Visualisation en Extraction des Connaissances, volume RNTI-E-7 de *Revue des Nouvelles Technologies de l'Information (RNTI)*, pages 113–125. Cépaduès, 2006.

[Lallich 04] S. Lallich et O. Teytaud. *Evaluation et validation de l'intérêt des règles d'association.* Revue des Nouvelles Technologies de l'Information, vol. 1, no. 2, pages 193–218, 2004.

[Lambert 10a] A. Lambert, R. Bourqui et D. Auber. *Winding roads : routing edges into bundles.* Proceedings of the 12th Eurographics / IEEE - VGTC conference on Visualization, EuroVis '10, pages 853–862, Aire-la-Ville, Switzerland, Switzerland, 2010. Eurographics Association.

[Lambert 10b] A. Lambert, R. Bourqui et D. Auber. *3D Edge Bundling for Geographical Data Visualization.* Proceedings of the 14th International Conference on Information Visualization (IV '10), pages 329–335, United Kingdom, Jul 2010.

[Lamping 95] J. Lamping, R. Rao et P. Pirolli. *A Focus+Context Technique Based on Hyperbolic Geometry for Visualizing Large Hierarchies.* Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '95, pages 401–408, New York, NY, USA, 1995. ACM Press/Addison-Wesley Publishing Co.

[Lan 01] *3D Data Management : Controlling Data Volume, Velocity, and Variety.* Application Delivery Strategies, no. 949, Feb 2001.

[Lee 06] B. Lee, C. Plaisant, C. S. Parr, J.-D. Fekete et N. Henry. *Task taxonomy for graph visualization.* Proceedings of the 2006 AVI workshop on BEyond time and errors : novel evaluation methods for information visualization, BELIV '06, pages 1–5, New York, NY, USA, 2006. ACM.

[Lerman 81] I. C. Lerman, R. Gras et H. Rostam. *Élaboration et évaluation d'un indice d'implication pour des données binaires. 2.* Mathématiques et Sciences Humaines, vol. 75, pages 5–47, 1981.

[Leung 08a] C. K.-S. Leung, P. P. Irani et C. L. Carmichael. *FIsViz : a frequent itemset visualizer.* Proceedings of the 12th Pacific-Asia conference

on Advances in knowledge discovery and data mining, PAKDD '08, pages 644–652, Berlin, Heidelberg, 2008. Springer-Verlag.

[Leung 08b] C. K.-S. Leung, P. P. Irani et C. L. Carmichael. *WiFIsViz : Effective Visualization of Frequent Itemsets*. Proceedings of the 2008 Eighth IEEE International Conference on Data Mining, ICDM '08, pages 875–880, Washington, DC, USA, 2008. IEEE Computer Society.

[Leung 09] C. K.-S. Leung et C. L. Carmichael. *FpViz : a visualizer for frequent pattern mining*. Proceedings of the ACM SIGKDD Workshop on Visual Analytics and Knowledge Discovery : Integrating Automated Analysis with Interactive Exploration, VAKD '09, pages 30–39, New York, NY, USA, 2009. ACM.

[Loevinger 47] J. Loevinger. *A Systematic Approach to the Construction and Evaluation of Tests of Ability*. volume 4. American Psychological Association, 1947.

[Mackinlay 86] J. Mackinlay. *Automating the design of graphical presentations of relational information*. ACM Trans. Graph., vol. 5, pages 110–141, April 1986.

[Mackinlay 91] J. D. Mackinlay, G. G. Robertson et S. K. Card. *The Perspective Wall : Detail and Context Smoothly Integrated*. Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '91, pages 173–176, New York, NY, USA, 1991. ACM.

[MacQueen 67] J. B. MacQueen. *Some Methods for Classification and Analysis of MultiVariate Observations*. L. M. L. Cam et J. Neyman, éditeurs, Proc. of the fifth Berkeley Symposium on Mathematical Statistics and Probability, volume 1, pages 281–297. University of California Press, 1967.

[Marcus 03] A. Marcus, L. Feng et J. I. Maletic. *3D Representations for Software Visualization*. Proceedings of the 2003 ACM Symposium on Software Visualization, SoftVis '03, pages 27–, New York, NY, USA, 2003. ACM.

[Martin 95] A. R. Martin et M. O. Ward. *High Dimensional Brushing for Interactive Exploration of Multivariate Data*. Proceedings of the 6th Conference on Visualization, VIS '95, pages 271–, Washington, DC, USA, 1995. IEEE Computer Society.

[Masquelier 12] G. Masquelier et C. Masquelier. Le grand livre de la gestalt. 2012.

[Mayorga 13] A. Mayorga et M. Gleicher. *Splatterplots : Overcoming Overdraw in Scatter Plots*. IEEE Transactions on Visualization and Computer Graphics, vol. 19, no. 9, pages 1526–1538, Sep 2013.

[Meo 98] R. Meo, G. Psaila et S. Ceri. *An Extension to SQL for Mining Association Rules*. Data Mining and Knowledge Discovery, pages 195–224, 1998.

[Miller 56] G. A. Miller. *The Magical Number Seven, Plus or Minus Two : Some Limits on Our Capacity for Processing Information.* The Psychological Review, vol. 63, no. 2, pages 81–97, Mar 1956.

[Mitchell 97] T. M. Mitchell. Machine learning. McGraw-Hill, Inc., New York, NY, USA, 1st edition, 1997.

[Mittelstadt 12] S. Mittelstadt, M. Behrisch, S. Weber, T. Schreck, A. Stoffel, R. Pompl, D. Keim, H. Last et L. Zhang. *Visual Analytics for the Big Data Era ; A Comparative Review of State-of-the-art Commercial Systems.* Proceedings of the 2012 IEEE Conference on Visual Analytics Science and Technology (VAST), VAST '12, pages 173–182, Washington, DC, USA, 2012. IEEE Computer Society.

[Munzner 97] T. Munzner. *H3 : Laying out Large Directed Graphs in 3D Hyperbolic Space.* Proceedings of the 1997 IEEE Symposium on Information Visualization (InfoVis '97), INFOVIS '97, pages 2–, Washington, DC, USA, 1997. IEEE Computer Society.

[Noack 03] A. Noack. *An Energy Model for Visual Graph Clustering.* Proceedings of the 11th International Symposium on Graph Drawing (GD 2003), LNCS 2912, pages 425–436. Springer-Verlag, 2003.

[OACI 12] OACI. Location indicators. International Civil Aviation Organization, 145 edition, 2012.

[Oleg Sindiy 13] K. Oleg Sindiy, S. Litomisky et F. D. Davidoff. *Introduction to Information Visualization (InfoVis) Techniques for Model-Based Systems Engineering.* 2013 Conference on Systems Engineering Research, Procedia Computer Science, pages 49–58. Elsevier, 2013.

[Pasquier 99a] N. Pasquier, Y. Bastide, R. Taouil et L. Lakhal. *Discovering Frequent Closed Itemsets for Association Rules.* Proceedings of the 7th International Conference on Database Theory, ICDT '99, pages 398–416, London, UK, UK, 1999. Springer-Verlag.

[Pasquier 99b] N. Pasquier, Y. Bastide, R. Taouil et L. Lakhal. *Efficient Mining Of Association Rules Using Closed Itemset Lattices.* Information Systems, vol. 24, pages 25–46, 1999.

[Pearl 88] J. Pearl. Probabilistic reasoning in intelligent systems : Networks of plausible inference. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1988.

[Pearson 96] K. Pearson. *Mathematical contributions to the theory of evolution. III. Regression, heredity, and panmixia.* Philosophical Transactions of the Royal Society of London. Series A, vol. 187, pages 253–318, Jan 1896.

[Phan 05] D. Phan, L. Xiao, R. Yeh, P. Hanrahan et T. Winograd. *Flow Map Layout.* Proceedings of the Proceedings of the 2005 IEEE Symposium

on Information Visualization, INFOVIS '05, pages 29–, Washington, DC, USA, 2005. IEEE Computer Society.

[Piatetsky-Shapiro 91a] G. Piatetsky-Shapiro. *Discovery, analysis and presentation of strong rules.* G. Piatetsky-Shapiro et W. J. Frawley, éditeurs, Knowledge Discovery in Databases, pages 229–248. AAAI Press, 1991.

[Piatetsky-Shapiro 91b] G. Piatetsky-Shapiro. *Knowledge Discovery in Real Databases : A Report on the IJCAI-89 Workshop.* AI Mag., vol. 11, no. 5, pages 68–70, Jan 1991.

[Playfair 86] W. Playfair. The commercial and political atlas : Representing, by means of stained copper-plate charts, the progress of the comerce, revenues, expenditure and debts of england during the whole of the eighteenth century. James Corry, London, 1786.

[Playfair 01] W. Playfair. The statistical breviary ; shewing the resources of every state and kingdom in europe. T. Bensley, Bolt Court, Fleet Street, 1801.

[Playfair 02] W. Playfair. Eléments de statistique où l'on démontre d'après un principe entièrement neuf, les ressources de chaque royaume, etat et république de l'europe ; suivi d'un état sommaire des principales puissanceset colonie de l'indostan. Batilliot and Genets, Paris, 1802.

[Poirot-Delpech 95] S. Poirot-Delpech. *Biographie du CAUTRA : naissance et développement d'un système d'informations pour la circulation aérienne.* Thèse de sociologie, Université Paris I, 1995.

[Potter 13] M. C. Potter, B. Wyble, C. E. Hagmann et E. S. McCourt. *Detecting meaning in RSVP at 13 ms per picture.* Atten Percept Psychophys, 2013.

[Pumain 02] D. Pumain et M.-C. Robic. *Le rôle des mathématiques dans une « révolution » théorique et quantitative : la géographie française depuis les années 1970.* Revue d'Histoire des Sciences Humaines, vol. 6, pages 123–144, 2002.

[Puolamäki 10] K. Puolamäki et A. Bertone. *Introduction to the Special Issue on Visual Analytics and Knowledge Discovery.* SIGKDD Explor. Newsl., vol. 11, no. 2, pages 3–4, May 2010.

[Qu 07] H. Qu, H. Zhou et Y. Wu. *Controllable and progressive edge clustering for large networks.* Proceedings of the 14th international conference on Graph drawing, GD '06, pages 399–404, Berlin, Heidelberg, 2007. Springer-Verlag.

[Rao 94] R. Rao et S. K. Card. *The Table Lens : Merging Graphical and Symbolic Representations in an Interactive Focus + Context Visualization for Tabular Information.* Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '94, pages 318–322, New York, NY, USA, 1994. ACM.

[RCA 13] Réglementation de la circulation aérienne. Service de l'information aéronautique, 2013.

[Rekimoto 97] J. Rekimoto. *Pick-and-drop : A Direct Manipulation Technique for Multiple Computer Environments*. Proceedings of the 10th Annual ACM Symposium on User Interface Software and Technology, UIST '97, pages 31–39, New York, NY, USA, 1997. ACM.

[Rider 44] F. Rider. The scholar and the future of the research library. a problem and its solution. Hadham Press, 1944.

[Roberts 07] J. C. Roberts. *State of the Art : Coordinated & Multiple Views in Exploratory Visualization*. Proceedings of the Fifth International Conference on Coordinated and Multiple Views in Exploratory Visualization, CMV '07, pages 61–71, Washington, DC, USA, 2007. IEEE Computer Society.

[Robertson 91] G. G. Robertson, J. D. Mackinlay et S. K. Card. *Cone Trees : Animated 3D Visualizations of Hierarchical Information*. Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '91, pages 189–194, New York, NY, USA, 1991. ACM.

[Savasere 95] A. Savasere, E. Omiecinski et S. B. Navathe. *An Efficient Algorithm for Mining Association Rules in Large Databases*. Proceedings of the 21th International Conference on Very Large Data Bases, VLDB '95, pages 432–444, San Francisco, CA, USA, 1995. Morgan Kaufmann Publishers Inc.

[Scheepens 11] R. Scheepens, N. Willems, H. van de Wetering, G. Andrienko, N. Andrienko et J. J. van Wijk. *Composite Density Maps for Multivariate Trajectories*. IEEE Transactions on Visualization and Computer Graphics, vol. 17, no. 12, pages 2518–2527, Dec 2011.

[Sebag 88] M. Sebag et M. Schoenauer. *Generation of Rules with Certainty and Confidence Factors from Incomplete and Incoherent Learning Bases*. J. Boose, B. Gaines, et M. Linster, editors, Proc. of the European Knowledge Acquisition Workshop (EKAW '88), 1988.

[Selassie 11] D. Selassie, B. Heller et J. Heer. *Divided Edge Bundling for Directional Network Data*. IEEE Transactions on Visualization and Computer Graphics, vol. 17, no. 12, pages 2354–2363, Dec 2011.

[SES 99] *La création du ciel unique européen*. Communication de la Commission au Conseil et au Parlement Européen, no. COM/99/0614, Dec 1999.

[Shi 12] C. Shi, W. Cui, S. Liu, P. Xu, W. Chen et H. Qu. *RankExplorer : Visualization of Ranking Changes in Large Time Series Data*. IEEE Transactions on Visualization and Computer Graphics, vol. 18, no. 12, pages 2669–2678, 2012.

[Shneiderman 96] B. Shneiderman. *The Eyes Have It : A Task by Data Type Taxonomy for Information Visualizations*. Proceedings of the 1996 IEEE

Symposium on Visual Languages, pages 336–, Washington, DC, USA, 1996. IEEE Computer Society.

[Shneiderman 01] B. Shneiderman. *Inventing Discovery Tools : Combining Information Visualization with Data Mining*. Proceedings of the 4th International Conference on Discovery Science, DS '01, pages 17–28, London, UK, 2001. Springer-Verlag.

[SIA 14] SIA. *Atlas VAC*. 2014.

[Silverman 86] B. W. Silverman. Density estimation for statistics and data analysis. Chapman & Hall, London, 1986.

[Simoff 08] S. J. Simoff, M. H. Böhlen et A. Mazeika, éditeurs. Visual data mining : Theory, techniques and tools for visual analytics. Springer-Verlag, Berlin, Heidelberg, 2008.

[SJU 12] SJU. *ATM Master Plan*. Oct 2012.

[Slocum 83] T. S. Slocum. *Predicting visual Clusters on Graduated Circle Maps*. The American Cartographer, vol. 10, pages 59–72, 1983.

[Sun 13] G.-D. Sun, Y.-C. Wu, R.-H. Liang et S.-X. Liu. *A Survey of Visual Analytics Techniques and Applications : State-of-the-Art Research and Future Challenges*. J. Comput. Sci. Technol., vol. 28, no. 5, pages 852–867, 2013.

[Swayne 91] D. F. Swayne, D. Cook et A. Buja. *Xgobi : Interactive Dynamic Graphics In The X Window System With A Link To S*, 1991.

[Symanzik 98] J. Symanzik, R. Pajarola et P. Widmayer. *XGobi And XploRe Meet Virgis*. In : 1998 Proceedings of the Section on Statistical Graphics. American Statistical Association, pages 50–55, 1998.

[Szabo 95] K. Szabo, P. Stucki, P. Aschwanden, T. Ohler, R. Pajarola et P. Widmayer. *A Virtual Reality based System Environment for Intuitive Walk-Throughs and Exploration of Large-Scale Tourist Information*, 1995.

[Tahir Guettala 12] A. E. Tahir Guettala, F. Bouali, C. Guinot et G. Venturini. *Un assistant utilisateur pour le choix et le paramétrage des méthodes de fouille visuelle de données.* Y. Lechevallier, G. Melançon et B. Pinaud, éditeurs, EGC, volume RNTI-E-23 de *Revue des Nouvelles Technologies de l'Information*, pages 399–404. Hermann-Éditions, 2012.

[Techapichetvanich 05] K. Techapichetvanich et A. Datta. *VisAR : A New Technique for Visualizing Mined Association Rules*. Proceedings of the First International Conference on Advanced Data Mining and Applications, ADMA'05, pages 88–95, Berlin, Heidelberg, 2005. Springer-Verlag.

[Telea 10] A. Telea et O. Ersoy. *Image-based edge bundles : simplified visualization of large graphs*. Proceedings of the 12th Eurographics / IEEE -

VGTC conference on Visualization, EuroVis '10, pages 843–852, Aire-la-Ville, Switzerland, Switzerland, 2010. Eurographics Association.

[Thomas 05] J. J. Thomas et K. A. Cook. Illuminating the path : The research and development agenda for visual analytics. National Visualization and Analytics Ctr, 2005.

[Toivonen 96] H. Toivonen. *Sampling Large Databases for Association Rules.* Proceedings of the 22th International Conference on Very Large Data Bases, VLDB '96, pages 134–145, San Francisco, CA, USA, 1996. Morgan Kaufmann Publishers Inc.

[Tominski 12] C. Tominski, H. Schumann, G. L. Andrienko et N. V. Andrienko. *Stacking-Based Visualization of Trajectory Attribute Data.* IEEE Trans. Vis. Comput. Graph., vol. 18, no. 12, pages 2565–2574, 2012.

[Treinish 94] L. Treinish. *Ozone Animation.* IBM, 1994.

[Treisman 85] A. Treisman. *Preattentive Processing in Vision.* Comput. Vision Graph. Image Process., vol. 31, no. 2, pages 156–177, Aug 1985.

[Tufte 83] E. R. Tufte. The visual display of quantitative information. Graphics Press, Cheshire, CT, USA, 1983.

[Tufte 90] E. R. Tufte. Envisioning information. Graphics Press, Cheshire, CT, USA, 1990.

[Tukey 65] J. W. Tukey. *The Technical Tools of Statistics.* pages 23–28, 1965.

[Tukey 77] J. W. Tukey. Exploratory data analysis. Addison-Wesley, 1977.

[Unwin 01] A. Unwin, H. Hofmann et K. Bernt. *The TwoKey Plot for Multiple Association Rules Control.* L. D. Raedt et A. Siebes, éditeurs, PKDD, volume 2168 de *Lecture Notes in Computer Science*, pages 472–483. Springer, 2001.

[Vaillant 05] B. Vaillant, P. Meyer, E. Prudhomme, S. Lallich, P. Lenca et S. Bigaret. *Mesurer l'intérêt des règles d'association.* Atelier Qualité des Données et des Connaissances (DQK 05), EGC 05, Paris, pages 69–78, Jan 2005.

[Vijender Singh 11] D. G. Vijender Singh. *Survey of Finding Frequent Patterns in Graph Mining : Algorithms and Techniques.* International Journal of Soft Computing and Engineering, vol. 1, pages 19–23, Jun 2011. ISSN 2231-2307.

[Ware 00] C. Ware. Information visualization : Perception for design. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2000.

[Westphal 98] C. Westphal et T. Blaxton. Data mining solutions : Methods and tools for solving real-world problems. John Wiley & Sons, Inc., New York, NY, USA, 1998.

[Wettel 11] R. Wettel, M. Lanza et R. Robbes. *Software Systems As Cities : A Controlled Experiment.* Proceedings of the 33rd International Conference on Software Engineering, ICSE '11, pages 551–560, New York, NY, USA, 2011. ACM.

[Wilkinson 99] L. Wilkinson. The grammar of graphics. Springer-Verlag New York, Inc., 1999.

[Williams 96] G. J. Williams et Z. Huang. *Modelling the KDD Process – A Four Stage Process...* 1996.

[Witten 11] I. H. Witten, E. Frank et M. A. Hall. Data mining : Practical machine learning tools and techniques. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 3rd edition, 2011.

[Wong 99a] P. C. Wong. *Guest Editor's Introduction : Visual Data Mining.* IEEE Computer Graphics and Applications, vol. 19, no. 5, pages 20–21, 1999.

[Wong 99b] P. C. Wong, P. Whitney et J. Thomas. *Visualizing Association Rules for Text Mining.* Proceedings of the 1999 IEEE Symposium on Information Visualization, pages 120–, Washington, DC, USA, 1999. IEEE Computer Society.

[Wong 04] P. C. Wong et J. Thomas. *Guest Editors' Introduction–Visual Analytics.* Sep 2004.

[Yang 03] L. Yang. *Visualizing frequent itemsets, association rules, and sequential patterns in parallel coordinates.* Proceedings of the 2003 international conference on Computational science and its applications : PartI, ICCSA'03, pages 21–30, Berlin, Heidelberg, 2003. Springer-Verlag.

[Yang 05] L. Yang. *Pruning and Visualizing Generalized Association Rules in Parallel Coordinates.* IEEE Trans. Knowl. Data Eng., vol. 17, no. 1, pages 60–70, 2005.

[Zaki 97] M. J. Zaki, S. Parthasarathy, M. Ogihara et W. Li. *New Algorithms for Fast Discovery of Association Rules.* Rapport technique, Rochester, NY, USA, 1997.

[Zhang 00] T. Zhang. *Association Rules.* T. Terano, H. Liu et A. L. P. Chen, éditeurs, PAKDD, volume 1805 de *Lecture Notes in Computer Science*, pages 245–256. Springer, 2000.

[Zhao 04] K. Zhao, B. Liu, T. M. Tirpak et A. Schaller. *V-Miner : Using Enhanced Parallel Coordinates to Mine Product Design and Test Data.* Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '04, pages 494–502, New York, NY, USA, 2004. ACM.

[Zhou 09] H. Zhou, X. Yuan, W. Cui, H. Qu et B. Chen. *Energy-Based Hierarchical Edge Clustering of Graphs.* In Proc. of the 2008 IEEE Pacific Visualization Symposium, pages 55–62. Blackwell Publishing Ltd, 2009.

Résumé

Depuis quelques années, nous assistons à une véritable explosion de la production de données dans de nombreux domaines, comme les réseaux sociaux ou le commerce en ligne. Ce phénomène récent est renforcé par la généralisation des périphériques connectés, dont l'utilisation est devenue aujourd'hui quasi-permanente. Le domaine aéronautique n'échappe pas à cette tendance. En effet, le besoin croissant de données, dicté par l'évolution des systèmes de gestion du trafic aérien et par les événements, donne lieu à une prise de conscience sur leur importance et sur une nouvelle manière de les appréhender, qu'il s'agisse de stockage, de mise à disposition et de valorisation.

Les capacités d'hébergement ont été adaptées, et ne constituent pas une difficulté majeure. Celle-ci réside plutôt dans le traitement de l'information et dans l'extraction de connaissances. Dans le cadre du Visual Analytics, discipline émergente née des conséquences des attentats de 2001, cette extraction combine des approches algorithmiques et visuelles, afin de bénéficier simultanément de la flexibilité, de la créativité et de la connaissance humaine, et des capacités de calculs des systèmes informatiques.

Ce travail de thèse a porté sur la réalisation de cette combinaison, en laissant à l'homme une position centrale et décisionnelle. D'une part, l'exploration visuelle des données, par l'utilisateur, pilote la génération des règles d'association, qui établissent des relations entre elles. D'autre part, ces règles sont exploitées en configurant automatiquement la visualisation des données concernées par celles-ci, afin de les mettre en valeur. Pour cela, ce processus bidirectionnel entre les données et les règles a été formalisé, puis illustré, à l'aide d'enregistrements de trafic aérien récent, sur la plate-forme Videam que nous avons développée. Celle-ci intègre, dans un environnement modulaire et évolutif, plusieurs briques IHM et algorithmiques, permettant l'exploration interactive des données et des règles d'association, tout en laissant à l'utilisateur la maîtrise globale du processus, notamment en paramétrant et en pilotant les algorithmes.

Mots-clés *: Visual Analytics, Fouille de données, Règles d'association, Mesures de qualité, Sémiologie graphique.*

Abstract

In the past few years, we have seen a large scale data production in many areas, such as social networks and e-business. This recent phenomenon is enhanced by the widespread use of devices, which are permanently connected. The aeronautical field is also involved in this trend. Indeed, its growing need for data, which is driven by air trafic management systems evolution and by events, leads to a widescale focus on its key role and on new ways to manage it. It deals with storage, availability and exploitation.

Data hosting capacity, that has been adapted, is not a major challenge. The issue is now in data processing and knowledge extraction from it. Visual Analytics is an emerging field, stemming from the September 2001 events. It combines automatic and visual approaches, in order to benefit simultaneously from human flexibility, creativity and knowledge, and also from processing capacities of computers.

This PhD thesis has focused on this combination, by giving to the operator a centered and decision-making role. On the one hand, the visual data exploration drives association rules extraction. They correspond to links between the data. On the other hand, these rules are exploited by automatically configurating the visualization of the concerned data, in order to highlight it. To achieve this, a bidirectional process has been formalized, between data and rules. It has been illustrated by air trafic recordings, thanks to the Videam platform, that we have developed. By integrating several HMI and algorithmic applications in a modular and upgradeable environment, it allows interactive exploration of both data and association rules. This is done by giving to human the mastering of the global process, especially by setting and driving algorithms.

Keywords *: Visual Analytics, Data Mining, Association Rules, Quality Measures, Graphic Semiology.*