Inférence d'interactions d'ordre supérieur et de complexes simpliciaux à partir de données de présence/absence

Mémoire

Xavier Roy-Pomerleau

Sous la direction de:

Louis J. Dubé, directeur de recherche
Patrick Desrosiers, codirecteur de recherche

Résumé

Malgré l'efficacité des réseaux pour représenter les systèmes complexes, de récents travaux ont montré que leur structure limite parfois le pouvoir explicatif des modèles théoriques, puisqu'elle n'encode que des relations par paire. Si une interaction plus complexe existe dans le système représenté, elle est automatiquement réduite à un groupe d'interactions par paire, c'est-à-dire d'ordre un. Il faut alors utiliser des structures qui prennent en compte les interactions d'ordre supérieur. Cependant, qu'elles soient ou non d'ordre supérieur, les interactions entre les éléments d'un système sont rarement explicites dans les jeux de données. C'est notamment le cas des données de présence/absence qui indiquent quelles espèces (animales, végétales ou autres) se retrouvent (ou non) sur un site d'observation sans indiquer les relations entre elles.

L'objectif de ce mémoire est alors de développer une technique d'inférence pour dénicher les interactions d'ordre supérieur au sein de données de présence/absence. Ici, deux cadres théoriques sont explorés. Le premier est basé sur la comparaison entre la topologie des données, obtenue grâce à une hypothèse souple, et celle d'un ensemble aléatoire. Le second utilise plutôt les modèles log-linéaire et les tests d'hypothèses pour inférer les interactions une à une jusqu'à l'ordre désiré. Ce cadre a permis d'élaborer plusieurs méthodes d'inférence qui génèrent des complexes simpliciaux (ou des hypergraphes) qui peut être analysés grâce aux outils standards de la science des réseaux en plus de l'homologie. Afin de valider ces méthodes, nous avons développé un modèle génératif de données de présence/absence dans lesquelles les véritables interactions sont connues. Des résultats concrets ont également été obtenus pour des jeux de données réelles. Notamment, à partir de données de présence/absence d'oiseaux nicheurs du Québec, nous avons réussi à inférer des cooccurrences d'ordre deux.

Abstract

Despite the effectiveness of networks to represent complex systems, recent work has shown that their structure sometimes limits the explanatory power of the theoretical models, since it only encodes dyadic interactions. If a more complex interaction exists in the system, it is automatically reduced to a group of pairwise interactions that are of the first order. We thus need to use structures that can take higher-order interactions into account. However, whether relationships are of higher order or not is rarely explicit in real data sets. This is the case of presence/absence data, that only indicate which species (of animals, plants or others) can be found (or not) on a site without showing the interactions between them.

The goal of this project is to develop an inference method to find higher-order interactions within presence/absence data. Here, two frameworks are examined. The first one is based on the comparison of the topology of the data, obtained with a non-restrictive hypothesis, and the topology of a random ensemble. The second one uses log-linear models and hypothesis testing to infer interactions one by one until the desired order. From this framework, we have developped several inference methods to generate simplicial complexes (or hypergraphs) that can be studied with regular tools of network science as well as homology. In order to validate these methods, we have developed a generative model of presence/absence data in which the true interactions are known. Results have also been obtained on real data sets. For instance, from presence/absence data of nesting birds in Québec, we were able to infer co-occurrences of order two.

Table des matières

Liste des tableaux

Liste des figures

Abréviations et symboles

Abréviations anglaises

OTU	**O**perational **T**axonomic **U**nit.
SCM	**S**implicial **C**onfiguration **M**odel.
MCMC	**M**onte-**C**arlo **M**arkov **C**hain.
MeSH	**Me**dical **S**ubject **H**eadings.

Chapitre 1 : Notions préliminaires

Symboles propres à la structure

$\mathcal{G}$	Réseau représentant la structure.
$\mathcal{V}$,$\mathcal{U}$	Ensembles de noeuds.
v_i	Noeud i de l'ensemble $\mathcal{V}$.
$\mathcal{E}$	Ensemble des liens.
N	Nombre de noeuds dans le réseau.
M	Nombre de liens dans le réseau.
∂i	Voisinage du noeud i.
k_i	Degré du noeud i.
$\mathbf{k}$	Séquence des degrés.
$\mathbf{A}$	Matrice d'adjacence.
a_{ij}	Élément (i, j) de la matrice d'adjacence.
Ω	Univers des possibilités pour une expérience aléatoire.
$\Omega_{CM}(\mathbf{k})$	Ensemble des réseaux ayant la séquence des degrés $\mathbf{k}$.
$\mathbf{B}$	Matrice de biadjacence.
b_{ij}	Élément (i, j) de la matrice de biadjacence.

Symboles propres aux probabilités et aux statistiques

$P(A)$	Probabilité de l'événement A.
$\emptyset$	Ensemble vide.
R_X	Support de la variable aléatoire X.
f	Fonction de densité.
F	Fonction de répartition.
$\bar{X}$	Moyenne arithmétique de la variable aléatoire X.
m_{ij}	Entrée (i, j) dans une table de contingence.
m_{+j}	j-ième entrée marginale dans la table de contingence.
p_{ij}	Probabilité d'observer de l'entrée (i, j) d'une table de contingence.
H_0	Hypothèse nulle.
H_1	Hypothèse alternative.
$\hat{m}_{ij}$	Entrée espérée dans la case (i, j) d'une table de contingence sous H_0.
ν	Nombre de degrés de liberté d'une loi χ^2.

Symboles propres à l'homologie

σ	Simplexe quelconque.
p_i	Point appartenant à un simplexe.
K	Complexe simplicial orienté quelconque.
$C_r(K)$	Groupe des chaînes de dimension r du complexe simplicial K.
f	Isomorphisme.
∂_r	Opérateur de frontière agissant sur les chaînes de $C_r(K)$.
$Z_r(K)$	Ensemble des cycles formés par des chaînes de $C_r(K)$.
ker	Noyau (*kernel*) d'une fonction.
$B_r(K)$	Ensemble des r-frontières formées par des chaînes de $C_r(k)$.
im	Image d'une fonction.
$H_r(K)$	r-ième groupe d'homologie du complexe simplicial K.
$\beta_r(K)$	r-ième nombre de Betti du complexe simplicial K.

Chapitre 2 : Inférence d'interaction d'ordre supérieurs : modèle des configurations simplicial et homologie significative

d_i	Degré généralisé du noeud i.
s_i	Nombre de noeuds appartenant à la facette i.
$\mathbf{d}$	Séquence des degrés généralisés d'un complexe simplicial.
$\mathbf{s}$	Séquence des tailles des facettes d'un complexe simplicial.
F	Ensemble des noeuds qui représentent une facette dans un réseau biparti.

S	Taille d'une facette quelconque.
N	Nombre de facettes dans un liste de facettes.

Chapitre 3 : Inférence d'interactions d'ordre supérieur : modèles log-linéaires

N	Nombre total d'observations dans une table de contingence.
u_i^A	Terme dans un modèle log-linéaire associé à la catégorie i de la variable A.
x_{ij}	Observation dans la case (i, j) d'une table de contingence.
ϕ	Mesure d'association entre deux variables aléatoires.
U_{AB}	Somme des termes en u dans la hiérarchie du terme u^{AB}.
$C_{\mathbf{y}}$	Configuration suffisante où $\mathbf{y}$ est un ensemble d'indices non sommés.
$\hat{x}_{ij}$	Entrée (i, j) d'une table modèle.
$L_1^{\max}$	Première valeur de L_1 qui permet d'annuler des entrées non nulles.
L_1^{seuil}	Valeur maximale de L_1 qui conserve un taux de succès de 100%.

Chapitre 4 : Inférence sur données synthétiques et réelles

$\mathbf{D}$	Ensemble de variables aléatoires.
Φ	Désigne un facteur dans un graphe de facteurs.
Z	Fonction de partition.
F	Facette d'un complexe simplicial.
$\boldsymbol{\sigma}$	Ensemble des simplexes qui composent une facette F.
ϕ_μ	Facteur élémentaire associé à un simplexe.
H_μ	Fonction d'énergie associée au facteur ϕ_μ.
X_i	Variable aléatoire associée à l'entité i dans la matrice de présence/absence.

Annexe A : Théorie des groupes et notions sur les applications

G	Groupe.
$*$	Opération d'un groupe.
e	Élément « unité » dans un groupe.
H	Sous-groupe quelconque du groupe G.
g_i	Élément i du groupe G.
f	Fonction.
$\text{im}f$	Image de la fonction f.

$\ker f$	Noyau de la fonction f.
$\oplus$	Opération associée à un groupe.
$\otimes$	Opération associée à un groupe, possiblement différente de $\oplus$.
$\sim$	Symbole d'une relation d'équivalence.

Liste des contributions

Articles

E. Laurence, C. Murphy, G. St-Onge, **X. Roy-Pomerleau**, V. Thibeault,
Detecting structural perturbations from time series using deep learning,
arXiv:2006.05232 (juin 2020).

Conférences

X. Roy-Pomerleau, L. J. Dubé, P. Desrosiers,
Inferring higher-order co-occurrence patterns and simplicial complexes from presence/absence data,
Workshop on Higher-order Interaction Networks (septembre 2019), Oxford, Royaume-Uni.

X. Roy-Pomerleau, J. Comte, W. F. Vincent, L. J. Dubé, P. Desrosiers,
Detecting higher-order co-occurrence patterns within aquatic bacterial communities across changing Nunavik permafrost landscape,
Sentinel North Annual Meeting (août 2019), Lévis, Canada.

V. Thibeault, G. St-Onge, **X. Roy-Pomerleau**, P. Desrosiers,
Predicting synchronization regimes with dimension reduction on modular graphs,
NetSci (mai 2019), Burlington, USA.

V. Thibeault, **X. Roy-Pomerleau**, G. St-Onge, J.-G. Young, L. J. Dubé, P. Desrosiers,
The impact of community structure on network dynamics : The case of synchronization,
Sentinel North Annual Meeting (août 2018), Québec, Canada.

V. Thibeault, **X. Roy-Pomerleau**, J.-G. Young, G. St-Onge, P. Desrosiers,
Synchronization dynamics on the stochastic bloc model,
NetSci (juin 2018), Paris, France.

Remerciements

Lors de cette aventure que représente la maîtrise, j'ai toujours bien été accompagné. Le travail présenté dans ce mémoire est d'ailleurs le fruit de collaborations et d'amitiés importantes à mes yeux. Même si mes mots ne peuvent que traduire l'ampleur de ma gratitude de manière malhabile, je souhaite souligner la contribution de plusieurs individus.

De nombreuses occasions ne se seraient jamais présentées à moi si je n'avais pas pu faire partie du groupe de recherche Dynamica. En ce sens, je suis redevable à mon directeur de recherche, Louis J. Dubé, qui, malgré une période de transition importante pour l'équipe, m'a accueilli tout en s'assurant de transmettre doucement le flambeau à mon estimé (co)directeur, Patrick Desrosiers, dont la rigueur, la curiosité et la motivation ont été les principaux moteurs pour m'aider à accomplir ce travail.

Je tiens aussi à remercier Antoine Allard, qui a su reprendre la moitié des rênes du groupe avec brio, et pour avoir accepté d'évaluer ce mémoire. Merci également à Jérôme Comte d'avoir inspiré une partie de ce projet et accepté de siéger sur le comité d'évaluation.

Je souhaite souligner le soutien de toute l'équipe Dynamica, Jean-Gabriel, Edward, Guillaume, Charles, Vincent, Béatrice et Francis. Que ce soit par des discussions enrichissantes, des parties de basket (parfois un peu trop intenses) ou des blagues savoureuses (parfois un peu trop intenses aussi), vous avez généré une foule de moments agréables qui ont su détendre l'atmosphère et me pousser à continuer. Faire partie de l'équipe avec vous a été un réel plaisir et je vous souhaite d'être, encore et toujours, aussi prolifiques dans vos projets.

À Ali, Ced, Charles, J, Nic, Pan et Vincent. Même si la dynamique de notre octuor a nettement évolué depuis la fin du baccalauréat en physique, je suis ravi que nous jouions toujours sur la même scène. Rien n'égale la mélodie de vos esprits et c'est, hors de tout doute, grâce à vous si progresser dans cette symphonie est si agréable. Depuis plus de cinq ans, nous avons complété bien des mouvements ensemble. Certains étaient plus joyeux que d'autres, mais même dans l'adversité, le fait d'être avec vous atténuait le crescendo de mes angoisses. Peu importe leur nature, chaque passage a laissé de doux échos qui réverbèrent toujours en moi. Je tiens d'ailleurs à vous remercier d'avoir écouté mes solos lors de certaines épreuves et répondu avec la délicatesse de vos archets bienveillants. Cette nouvelle étape étant terminée, j'espère de tout

cœur pouvoir continuer d'improviser avec vous, car il n'existe point de meilleur moyen pour faire résonner la table d'harmonie de mon bonheur.

Je veux également remercier chaleureusement mon camarade de longue date, Maxime, qui m'appuie beau temps mauvais temps. Lorsque je pense à notre amitié, je ne peux qu'être reconnaissant envers le hasard fortuit qui s'est produit dans notre chère ville natale lorsqu'il nous a mis sur le même chemin.

Un énorme merci à Naomie, qui m'a accompagné de sa présence lumineuse et pétillante dans mille et une expéditions flamboyantes.

Quelques remerciements spéciaux pour Anna, une amie que j'aurais aimé rencontrer plus tôt, Vincent V., pour son écoute et sa patience, et Minnie, with whom I'll hopefully drink cider again.

Finalement, je souhaite remercier ma famille, qui m'encourage depuis plus longtemps que je ne peux me rappeler. Plus particulièrement, merci à mon père, Daniel, et ma mère, Sylvie, pour leur rôle soutenu dans mon éducation (et bien d'autres sphères !), ainsi qu'à ma sœur, Madeleine, sur qui je peux toujours compter pour me dire d'aller au gym.

Introduction

La science des réseaux a réussi un tour de force en s'intégrant à de nombreux domaines scientifiques [4, 37, 47]. Sa réputation n'est d'ailleurs plus à faire, ayant permis d'étudier avec aisance les systèmes complexes, c'est-à-dire les systèmes composés de plusieurs éléments en interaction et dont les propriétés dites « émergentes » ne peuvent pas être expliquées par l'étude des éléments isolés. Le fait de considérer les interactions entre les éléments d'un système est donc l'une des clés de la réussite de cette science. En guise d'exemples, nous pouvons penser à l'étude d'écosystèmes dans lesquels les relations entre les organismes vivants sont multiples ou au cerveau composé de diverses cellules neuronales qui, ensemble, accomplissent des fonctions complexes comme la mémoire ou la coordination.

Malgré leur popularité, les réseaux sont limités par une hypothèse majeure, c'est-à-dire que les interactions considérées n'ont lieu qu'entre des paires d'éléments du système. Dans certains cas, cette hypothèse est valide et permet de représenter les interactions du système telles qu'elles sont. Dans d'autres cas, cette hypothèse n'est pas entièrement véridique, mais permet de réduire la complexité du modèle. Or, au même titre qu'un système complexe ne peut pas être compris parfaitement par l'étude de ses composantes simples, le fait de ne considérer que des interactions par paire peut faire ombrage à des propriétés importantes [7, 36]. Considérant cela, une nouvelle branche de la science des réseaux s'est développée dans les dernières années pour y inclure les **interactions d'ordre supérieur**. On leur trouve d'ailleurs déjà de nombreuses applications, que ce soit en neuroscience, en écologie, en épidémiologie et dans les réseaux de transport [6, 27, 32, 39].

Dans les faits, l'expression « interactions d'ordre supérieur » regroupe un large éventail de modèles. Par exemple, cela peut faire référence à des réseaux multicouches, dans lesquels la dynamique sur une couche influence celle sur l'autre et vice-versa. Plus concrètement, on pourrait penser à un réseau de transport en commun où une couche représente le métro et l'autre représente les autobus. Puisque des passagers s'échangent entre les autobus, entre les trains, mais aussi entre les deux couches, les deux dynamiques s'influencent mutuellement [39]. Parmi les différents modèles, nous pourrions également penser aux réseaux non markoviens, dans lesquels l'information sur le chemin emprunté par un marcheur est disponible sur d'une distance plus grande que le noeud précédent [7, 36]. Par exemple, dans un réseau de transport

aérien dit markovien, nous ne pourrions pas distinguer deux avions qui partent du même point A et atterrissent au même point B. Par contre, peut-être que ces deux avions, avant d'arriver au point A, partaient de deux endroits différents, rendant les chemins distinguables, mais nécessitant l'utilisation d'un réseau non markovien.

Dans ce mémoire, nous utilisons toutefois l'expression « interactions d'ordre supérieur » pour désigner des modèles combinatoires (*combinatorial higher-order models* [36]), c'est-à-dire des structures qui comportent des « liens » pouvant rattacher plus d'un noeud. Ces « liens » indiquent donc des interactions qui se produisent entre un groupe arbitraire d'éléments. Par analogie, pour préparer un mojito, nous devons faire en sorte que la lime, le sucre, la menthe, l'eau pétillante, le rhum et la glace interagissent tous ensemble. Le mélange ainsi produit forme un tout que nous ne pouvons pas simplement séparer en ses différentes paires d'ingrédients. Il en résulte d'ailleurs un délicieux cocktail d'interactions. Plus sérieusement, nous pouvons penser à des scientifiques qui collaborent pour écrire un article. Même s'il est possible qu'un groupe d'auteurs ait travaillé en paires, il y a sans doute eu des moments où tous se sont réunis pour en discuter. L'article en question est donc le fruit d'une coopération totale, et non pas la somme de chacune des paires. Autrement dit, les modèles combinatoires écartent l'hypothèse limitante des réseaux et permettent d'encoder des « superliens » pouvant regrouper plus de deux noeuds dans des structures mathématiques appelées hypergraphes et complexes simpliciaux.

C'est dans l'optique d'ajouter de telles interactions au sein d'un jeu de données que le projet de maîtrise contenu dans ce mémoire a pris forme. En effet, dans le cadre de la stratégie de recherche Sentinelle Nord de l'Université Laval [1], nous avons été approchés par deux biologistes, Jérôme Comte [2, 3] et Warwick F. Vincent[2], qui ont réalisé une analyse réseau sur des données de présence/absence de bactéries dans des thermokarsts au nord du Québec. Plus précisément, les données de présence/absence indiquent si une entité (ici une « espèce » bactérienne) a été retrouvée ou non sur un site échantillonné (ici les thermokarsts). Grâce à de telles données, il a été possible d'inférer des cooccurrences significatives entre les « espèces » de bactéries et d'identifier des relations de dépendance (représentées par des liens dans un réseau) et d'indépendance entre elles [16].

Dans ce contexte, nous nous sommes alors demandé s'il était possible de généraliser ce travail en identifiant des cooccurrences d'ordre supérieur, c'est-à-dire des relations de dépendance entre plus de deux espèces qui ne peuvent pas simplement être expliquées par les paires. Au

1. Il s'agit de projets transdisciplinaires financés par le Fonds d'excellence en recherche Apogée Canada visant à améliorer notre compréhension de l'environnement nordique et de son impact sur l'être humain et sa santé. L'un des principaux chantiers thématiques a pour objectif de décoder les interrelations entre systèmes complexes du Nord et mise sur la science des réseaux pour atteindre ce but. Notre groupe de recherche, Dynamica, fait partie de cette stratégie.

2. Centre d'études nordiques (CEN), Takuvik Joint International Laboratory & Département de biologie, Université Laval, Québec, QC G1V 0A6, Canada.

3. Institut de Biologie Intégrative et des Systèmes (IBIS), Université Laval, Québec, QC G1V 0A6, Canada.

départ, nous croyions qu'il était possible d'utiliser certains outils déjà développés, qui font d'ailleurs l'objet du deuxième chapitre de ce mémoire, mais ils se sont avérés inadéquats pour cette tâche.

Afin de répondre à notre question initiale, il était alors nécessaire d'emprunter une autre avenue. Or, peu d'outils semblent exister pour inférer ces interactions. Il fallait donc s'attaquer à cette lacune et c'est ainsi que nous avons élaboré ce projet de maîtrise dont l'objectif principal est de développer une technique d'inférence d'interactions d'ordre supérieur sur des données de présence/absence. La réalisation de notre objectif généralise la recherche de cooccurrences aux ordres supérieurs et bonifie l'analyse de ces jeux de données en dégageant une structure plus réaliste pour ceux-ci.

Implicitement, cet objectif en génère un second, qui est de montrer que la méthode d'inférence peut fonctionner sur de telles données, car, tel que nous l'avons stipulé, l'information sur les interactions n'est pas directement disponible. À cette fin, il est nécessaire d'élaborer un modèle génératif de données de présence/absence dans lesquelles les interactions sont connues, peu importe leur ordre. La méthode d'inférence doit alors être en mesure de retrouver les dépendances incorporées dans le modèle.

Typiquement, l'inférence est un problème statistique. D'ailleurs, pour déterminer les cooccurrences significatives au sein de données de présence/absence, il est coutume d'utiliser des modèles nuls et des tests d'hypothèses. Dans ce travail, nous utilisons également ces techniques en mettant l'accent sur des tests d'hypothèses basés sur les modèles log-linéaires, qui sont généralisables pour tous les ordres. De plus, afin de représenter les interactions, nous allons invoquer plusieurs structures reliées à la science des réseaux, comme les réseaux bipartis, les graphes de facteurs et les complexes simpliciaux.

Dans ce mémoire, nous présentons comment tous ces éléments ont été mis en relation pour développer différentes méthodes d'inférence. Au premier chapitre, nous expliquons des notions diversifiées, autant en lien avec les réseaux et les complexes simpliciaux qu'avec les probabilités et les statistiques, sur lesquelles l'ensemble du projet repose. Dans le deuxième chapitre, une première technique d'inférence est présentée. Cette dernière utilise des outils développés à l'extérieur de ce projet, comme le modèle simplicial des configurations, et constitue un premier jet pour inférer indirectement les interactions d'ordre supérieur au sein de données de présence/absence. Au troisième chapitre, nous présentons la théorie des modèles log-linéaires en inférence statistique et montrons comment ces derniers peuvent être utilisés pour construire des hypergraphes et des complexes simpliciaux. La robustesse de l'inférence, c'est-à-dire sa capacité à converger vers la même conclusion malgré une perturbation des données originales, est étudiée. Ensuite, au quatrième chapitre, nous développons un modèle permettant de générer des données de présence/absence qui, en moyenne, respectent les relations de dépendance que nous avons introduites dans l'algorithme. Grâce à ce dernier, nous présentons les performances

des méthodes d'inférence proposées sur des données bruitées pour retrouver les interactions véridiques. Ayant eu accès à divers jeux de données réels, nous présentons également les résultats des différentes méthodes d'inférence sur ces derniers. Finalement, nous concluons en présentant les succès et les limites des techniques étudiées en plus de fournir des pistes pour améliorer davantage nos méthodes d'inférence et l'analyse de leurs résultats.

Chapitre 1

Notions préliminaires

Dans ce chapitre, nous présentons les notions mathématiques de base qui sont utilisées tout au long de ce mémoire. Dans la section 1.1, des éléments de la science des réseaux complexes sont introduits. Par la suite, la section 1.2 contient un abrégé de notions de base en probabilités et statistiques. Finalement, la section 1.3 décrit une structure réseau particulière ainsi que ses caractéristiques topologiques.

1.1 Réseaux complexes

Les réseaux complexes sont des structures qui permettent d'illustrer les relations entre diverses entités au moyen de noeuds, représentant les entités, et de liens, représentant les relations. Un exemple d'un tel réseau se trouve à la figure 1.1. Grâce à cette idée, il est possible de représenter une panoplie de systèmes, comme un réseau social ou un réseau de composantes électroniques, et de caractériser leur structure. Les détails concernant ces différentes notions sont présentés ici et sont tirés des références [4, 37, 47].

1.1.1 Réseaux simples

La structure d'un réseau [1] à N noeuds, $\mathcal{G}$, est encodée dans deux ensembles, c'est-à-dire l'ensemble des noeuds $\mathcal{V} = \{v_1, v_2, \cdots, v_N\}$ et l'ensemble de liens $\mathcal{E}$ qui est formé de couples de noeuds (v_i, v_j) tirées de $\mathcal{V}$. Nous pouvons spécifier davantage la structure des liens en leur ajoutant une direction ou un poids. Or, dans ce mémoire, nous nous concentrons sur les réseaux simples, c'est-à-dire les réseaux où un noeud n'est jamais connecté avec lui-même et où les liens ne possèdent pas de poids. Il ne peut exister qu'un seul lien entre une paire de noeuds et ce lien est non dirigé. Un exemple de réseau simple est présenté à la figure 1.1.

Pour représenter un réseau, il est d'usage d'utiliser la **matrice d'adjacence** notée $\mathbf{A}$. Cette matrice est de taille $N \times N$, où N est le nombre de noeuds dans l'ensemble $\mathcal{V}$. L'élément a_{ij} de

1. Dans ce mémoire, nous utilisons les termes « réseau » et « graphe » de manière interchangeable.

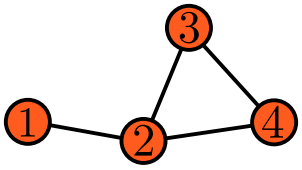

FIGURE 1.1 – Graphe simple à quatre noeuds.

la matrice est 1 s'il existe une connexion entre les noeuds v_i et v_j et 0 autrement. Puisque nous sommes en présence d'un réseau où les liens sont non dirigés, la matrice est symétrique, car si v_i est connecté à v_j, v_j est également connecté à v_i. De plus, les éléments sur la diagonale sont nuls, car les connexions du type (v_i, v_i) ne sont pas permises. La matrice d'adjacence du réseau 1.1 est

$$\mathbf{A} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix}. \tag{1.1}$$

À partir de la matrice d'adjacence, nous pouvons mesurer certaines quantités sur le réseau. L'une des plus communes est le **degré** k_i du noeud v_i, qui correspond au nombre de noeuds auquel il est rattaché. Autrement dit, il s'agit du nombre de liens qui relient v_i à d'autres noeuds. Pour obtenir cette quantité à partir de la matrice d'adjacence, nous pouvons utiliser l'équation

$$k_i = \sum_{j=1}^{N} a_{ij} = \sum_{j=1}^{N} a_{ji}. \tag{1.2}$$

L'égalité entre les deux sommes est valide lorsque nous sommes en présence d'un réseau simple non dirigé.

1.1.2 Modèle des configurations et modèle nul

Il existe de nombreuses méthodes pour générer des graphes aléatoirement. Dans ce mémoire, ce processus nous intéresse afin de comparer un réseau réel à un **modèle nul**.

En bref, un modèle nul est un outil de comparaison entre des données expérimentales et des données synthétiques générées aléatoirement [49]. Dans le contexte des réseaux, nous générons un échantillon de graphes qui possèdent tous la propriété #1 et laissons varier les autres propriétés aléatoirement. Nous comparons ensuite la valeur d'une autre propriété #2 mesurée sur le réseau réel, aux valeurs de cette même propriété mesurée pour chaque graphe de l'échantillon. À l'aide de tests d'hypothèses, couverts à la section 1.2.5, nous pouvons alors déterminer s'il est plausible de croire que la propriété #2 du réseau réel n'est pas aléatoire [49].

La première méthode de génération de graphes aléatoires que nous allons étudier est celle du modèle des configurations. Pour utiliser ce modèle, nous devons spécifier le degré désiré pour chacun des noeuds dans le réseau. Cette liste des degrés s'appelle une **séquence de degrés** notée $\mathbf{k} = [k_1, k_2, \cdots, k_N]$. Le modèle tente alors de générer des réseaux qui possèdent la même séquence de degrés que celle spécifiée. L'ensemble des graphes possibles ayant cette même séquence de degrés est noté $\Omega_{CM}(\mathbf{k})$. La probabilité de tirer un graphe particulier $\mathcal{G}$ est

$$P(\mathcal{G}) = \begin{cases} \frac{1}{|\Omega_{CM}(\mathbf{k})|} & \text{si } \mathcal{G} \text{ possède la séquence } \mathbf{k}, \\ 0 & \text{autrement.} \end{cases} \tag{1.3}$$

Puisque nous désirons générer des graphes simples, nous devons également interdire la génération de boucles, c'est-à-dire de liens (v_i, v_i), et la présence de plusieurs liens au sein d'une même paire. Il existe toutefois des algorithmes qui peuvent permettre ces types de liens s'ils sont d'intérêt dans le modèle [25].

Pour générer un graphe simple, il faut d'abord associer chacun des degrés de la séquence $\mathbf{k}$ à un noeud du réseau. Par la suite, pour chaque noeud v_i, nous générons k_i demi-liens. Ces demi-liens émanent des noeuds, mais leur terminaison n'est pas connectée à un autre demi-lien. Un algorithme choisit alors aléatoirement deux demi-liens avec probabilité uniforme et les connecte pour créer un lien. Il suffit alors de poursuivre ce processus avec tous les demi-liens. De cette manière, il peut arriver qu'une boucle ou qu'un multilien soit créé. Si c'est le cas, l'algorithme doit annuler la connexion et en chercher une autre ou recommencer du début. [2]

1.1.3 Réseau biparti et données de présence/absence

Tout au long de ce mémoire, nous aurons aussi besoin d'un second type de réseau, soit le **réseau biparti**. Un tel réseau est constitué de deux ensembles de noeuds, $\mathcal{U}$ et $\mathcal{V}$, et l'ensemble des liens $\mathcal{E}$ comprend des connexions du type (u_i, v_j). Il s'agit encore d'un réseau simple où les liens ne peuvent exister que s'ils relient des noeuds d'un ensemble à l'autre.

Pour les représenter de manière matricielle, nous pouvons utiliser la **matrice de biadjacence** notée $\mathbf{B}$. Cette dernière est de taille $N_{\mathcal{U}} \times N_{\mathcal{V}}$, où $N_{\mathcal{U}}$ est le nombre de noeuds dans l'ensemble $\mathcal{U}$ et $N_{\mathcal{V}}$ est le nombre de noeuds dans l'ensemble $\mathcal{V}$. L'élément b_{ij} est 1 s'il existe un lien (u_i, v_j) et 0 autrement. Un exemple de réseau biparti est donné au centre de la figure 1.2 et sa matrice de biadjacence est

$$\mathbf{B} = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix}. \tag{1.4}$$

2. Il existe des algorithmes plus performants pour générer des graphes avec le modèle des configurations [25].

Cette représentation est utile lorsque le système étudié comporte des noeuds qui sont de nature différente.

Dans ce mémoire, nous utilisons les graphes bipartis pour représenter des systèmes où il y a $N_{\mathcal{U}}$ « entités » dans l'ensemble $\mathcal{U}$ et $N_{\mathcal{V}}$ « sites » dans l'ensemble $\mathcal{V}$. Par exemple, les entités pourraient être des espèces animales, des mots ou des acteurs, tandis que les sites associés pourraient être différentes sections dans une forêt, des phrases ou des films. Des données se présentant sous cette forme correspondent à des **données de présence/absence**. Ces données sont généralement représentées sous forme de matrice de biadjacence, où l'élément b_{ij} est 1 si l'entité u_i s'est retrouvée sur le site v_j et 0 autrement. Dans ce mémoire, nous allons d'ailleurs traiter quatre jeux de données différents qui peuvent être considérés comme des données de présence/absence.

Le premier jeu de données provient de l'échantillonnage de thermokarsts au nord du Québec (Canada) pour y extraire les taxa bactériens. L'objectif derrière ces données était d'identifier les cooccurrences significatives entre ces taxa [16]. La définition d'« espèce » étant moins appropriée pour les bactéries, ce jeu de données utilise les unités taxonomiques opérationnelles (*Operational Taxonomic Units*) que nous allons abréger par OTUs dans le reste de ce mémoire. Dans le jeu de données original, on compte 38 sites et 2 611 OTUs. Dans la référence [16], ce jeu de données a été sous-échantillonné et présente plutôt 34 sites et 2 166 OTUs. Plus précisément, seules les OTUs qui se sont présentées sur au moins trois thermokarsts avec un minimum de 20 apparitions ont été conservées [16]. Ce jeu de données nous a été fourni par Jérôme Comte [3, 4] et Warwick F. Vincent[2]. Nous allons référer à ce jeu de données par l'expression **jeu de données d'OTUs**.

Le second jeu de données provient de l'échantillonnage de sites de la forêt boréale dans la région de la Côte-Nord au Québec pour identifier les espèces d'oiseaux par la méthode des points d'écoute [9]. À l'origine, ces données ont été utilisées dans des modèles pour déterminer si le caribou boréal était une espèce « parapluie » pour la faune de la forêt boréale. Ce jeu de données comprend 70 espèces d'oiseaux et 115 sites. Il nous a été fourni par Daniel Fortin [5]. Dans ce mémoire, nous allons référer à ce jeu de données par l'expression **jeu de données sur les oiseaux datant de 2016**.

Le troisième jeu de données est aussi en lien avec les oiseaux. Il s'agit du jeu de données *Québec Breeding Bird Atlas 2019* sur les oiseaux nicheurs du Québec, accessible sur le site web `http://www.naturecounts.ca/`. Celui-ci est également récolté par la méthode des points d'écoute et comprend 185 espèces et 1 382 sites. Il nous a été fourni par Alexandre Terrigeol[4]

3. Centre d'études nordiques (CEN), Takuvik Joint International Laboratory & Département de biologie, Université Laval, Québec, QC G1V 0A6, Canada.

4. Institut de Biologie Intégrative et des Systèmes (IBIS), Université Laval, Québec, QC G1V 0A6, Canada.

5. Département de biologie, NSERC-Université Laval Industrial Research Chair in Sylviculture and Wildlife, Université Laval, Québec QC GIV 0A6, Canada.

et Daniel Fortin[4]. Pour la suite, nous allons faire référence à ce jeu de données par l'expression **jeu de données sur les oiseaux datant de 2019**.[6]

Le quatrième jeu de données correspond à la base de données MEDLINE. Cette dernière comprend des publications scientifiques dans le domaine des sciences biomédicales. Chacune de ces publications est classée à l'aide d'étiquettes appelées *MeSH* (*Medical Subject Headings*). Un MeSH peut contenir plus d'un mot, mais chaque MeSH est associé à un identifiant unique. Les étiquettes sont alors les « entités » tandis que les publications scientifiques sont les « sites ». Cette base de données est accessible via le site web `https://www.nlm.nih.gov/`. Pour une année donnée, il est possible d'extraire les publications ainsi que les MeSH qui leur sont associés. Dans ce mémoire, nous allons référer à ce jeu de données par l'expression **jeu de données MEDLINE**.

Il est possible de transformer un réseau biparti pour obtenir un réseau simple ne possédant qu'un seul ensemble de noeuds. Nous appelons cette opération la **projection** du réseau biparti sur l'ensemble de noeuds $\mathcal{U}$ (ou $\mathcal{V}$ selon la projection désirée). Si la projection du réseau biparti est effectuée sur l'ensemble $\mathcal{U}$, nous connectons les noeuds u_i et u_j s'ils sont reliés à un ou plusieurs mêmes noeuds de l'ensemble $\mathcal{V}$.

Dans le contexte des données de présence/absence, cela signifie que nous relions deux entités qui sont apparues en même temps sur au moins un site. Si nous effectuons la projection sur le second ensemble de noeuds, nous relions plutôt les sites qui partagent des entités communes. Les projections du réseau biparti ayant la matrice d'adjacence (1.4) sont illustrées à la figure 1.2.

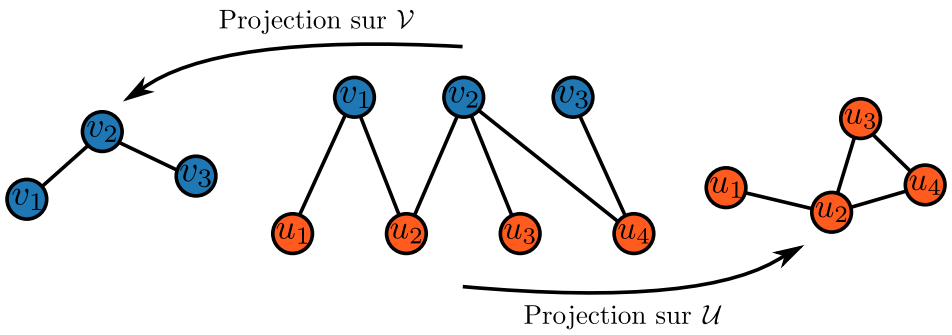

FIGURE 1.2 – (Centre) Réseau biparti contenant quatre noeuds dans l'ensemble $\mathcal{U}$ et trois noeuds dans l'ensemble $\mathcal{V}$. (Gauche) Projection du réseau biparti sur l'ensemble $\mathcal{V}$. (Droite) Projection du réseau biparti sur l'ensemble $\mathcal{U}$.

6. Nous remercions également l'Atlas des oiseaux nicheurs du Québec, le Regroupement QuébecOiseaux, Environnement et Changement climatique Canada, Oiseaux Canada ainsi que tous les participants qui ont récolté des données pour le projet.

1.1.4 Hypergraphes

Dans les deux types de réseaux présentés jusqu'à présent, les liens définissent les relations entre des paires de noeuds. Ainsi, lorsque nous utilisons ces formalismes pour représenter des systèmes réels, nous faisons l'hypothèse que les interactions entre les entités sont dyadiques. Or, au sens où nous l'entendons dans ce mémoire, une **interaction d'ordre supérieur** est une interaction qui ne peut être réduite à la somme des interactions par paire. Lorsque ces interactions sont considérées dans certains systèmes, il peut se produire des changements draconiens dans la dynamique ou dans notre interprétation de la structure [7, 27, 36].

Pour représenter ces interactions dans un graphe, nous avons donc besoin de « liens » qui relient plus d'un noeud à la fois, comme illustré à la figure 1.3. Un graphe qui contient de tels « liens » est appelé **hypergraphe** et les « liens » qui unissent plus de deux noeuds sont des **hyperliens**. Dans ce mémoire, nous nous intéressons à un type d'hypergraphe particulier appelé **complexe simplicial**. Plus de détails sur ces structures se retrouvent à la section 1.3, car nous devons définir davantage d'éléments pour les traiter.

Par contre, en guise d'introduction, spécifions qu'un complexe simplicial est un hypergraphe dans lequel les interactions d'ordre inférieur existent aussi au sein d'une interaction d'ordre supérieur, tel qu'illustré par le triangle coloré à la figure 1.3. Par exemple, imaginons un hypergraphe où les noeuds représentent des scientifiques et les liens représentent des collaborations. Un groupe qui se réunit forme alors un hyperlien. Or, il est possible qu'au cours du projet, les scientifiques aient eu à travailler en paires, représentées par les liens qui bornent le triangle coloré.

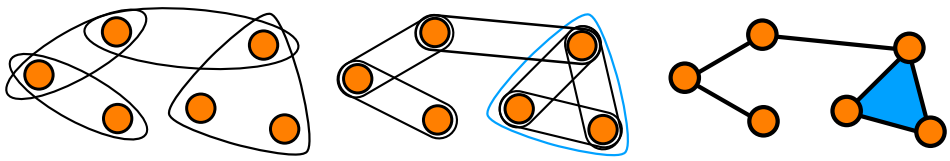

FIGURE 1.3 – À gauche, un hypergraphe où les liens englobent les noeuds qui forment l'interaction. Au centre, un complexe simplicial où les liens pour chaque paire existent aussi au sein de l'hyperlien à trois noeuds. À droite, la représentation standard du complexe simplicial au centre, où les liens prennent la même forme qu'un graphe simple, mais où la coloration d'une clique indique la présence de l'interaction d'ordre supérieur.

1.2 Probabilités et statistiques

Les méthodes d'inférences présentées dans ce mémoire nécessitent l'usage des probabilités et des statistiques. Voici donc une revue de différents concepts propres à ces disciplines. Les deux références utilisées pour définir ces concepts sont [1] et [20].

1.2.1 Probabilités

Afin de définir les notions principales de cette section, nous aurons recours à l'exemple du lancer d'un dé non truqué à six faces. Cet exemple constitue une **expérience aléatoire** où le résultat ne peut être prédit avec certitude. Il est toutefois possible de construire un cadre mathématique pour définir l'expérience et en dégager certaines conclusions. À cet effet, la première étape consiste à définir **l'univers des possibilités** Ω, c'est-à-dire l'ensemble de tous les résultats possibles de l'expérience. Dans le cas du dé, nous avons l'univers

$$\Omega = \{1, 2, 3, 4, 5, 6\}. \tag{1.5}$$

Cela signifie que lors de l'expérience, les seuls résultats possibles sont les chiffres de 1 à 6. Tout sous-ensemble formé à partir de l'ensemble Ω constitue un **événement**, c'est-à-dire que le résultat du lancer peut se situer dans ce sous-ensemble. Par exemple, il est possible de définir l'événement « obtenir un nombre pair » comme l'ensemble $A = \{2, 4, 6\} \subset \Omega$. L'obtention d'un chiffre précis ainsi que l'obtention d'un chiffre de 1 à 6 constituent également des événements.

Dans l'exemple du dé, une probabilité est associée à chaque événement, c'est-à-dire une fonction P sur les événements de Ω qui respecte les conditions suivantes :

1. $0 \leq P(A) \leq 1$ pour tout événement A ;
2. $P(\emptyset) = 0$ et $P(\Omega) = 1$;
3. Si $A_1, A_2, \ldots$ sont des événements disjoints, c'est-à-dire que $A_i \bigcap A_j = \emptyset$ pour $i \neq j$, alors
$$P\left(\bigcup_i A_i\right) = \sum_i P(A_i),$$
où $\bigcap$ désigne l'intersection entre des ensembles, $\emptyset$ désigne l'ensemble vide et $\bigcup$ désigne l'union d'ensembles.

Dans le cas d'un univers des possibilités dénombrable, il est possible de ne s'intéresser qu'aux événements qui ne contiennent qu'un élément. Dans ce cas, l'univers des possibilités s'écrit comme $\Omega = \{\omega_1, \omega_2, \ldots\}$, où les ω_i sont les résultats possibles. Nous définissons une probabilité sur Ω qui attribue un nombre réel à chaque événement $\{\omega_i\}$ de sorte que

1. $0 \leq P(\{\omega_i\}) \leq 1$;
2. $\sum_i P(\{\omega_i\}) = 1$.

Dans le cas du dé, la probabilité associée à chaque $\{\omega_i\}$ (excepté l'ensemble vide) est 1/6 (et 0 pour l'ensemble vide). Calculer la probabilité associée à un événement composé de plusieurs éléments, tel qu'obtenir un chiffre pair, revient à faire la somme des probabilités de chaque événement individuel. Par exemple, la probabilité d'obtenir un chiffre pair est

$$P(\{2,4,6\}) = P(\{2\},\{4\},\{6\}) = P(\{2\}) + P(\{4\}) + P(\{6\}) = \frac{1}{2}. \tag{1.6}$$

1.2.2 Variables aléatoires

Intuitivement, une **variable aléatoire** est une manière d'attribuer un nombre à un événement. Par exemple, dans le cas du lancer d'une pièce de monnaie, les résultats possibles sont « pile » ou « face ». Or, il serait aussi possible d'attribuer le chiffre 0 à l'événement « pile » et 1 à l'événement « face ».

Pour chacun des événements dans Ω, la variable aléatoire (notée par une lettre latine majuscule, typiquement X ou Y) prend donc une valeur numérique. Il s'agit donc d'attribuer une étiquette numérique à chaque événement. La variable aléatoire devient alors le nouvel objet d'étude. Au lieu d'une pièce de monnaie, nous avons une variable qui, lorsque l'expérience est réalisée, peut soit prendre la valeur 0, soit la valeur 1.

On l'appelle variable aléatoire, car sa valeur dépend du résultat de l'expérience aléatoire et, tant que l'expérience n'est pas réalisée, celle-ci n'a pas de valeur fixe. Plus précisément, une variable aléatoire est une fonction

$$\begin{aligned} X\colon \Omega &\to \mathbb{R}, \\ \omega_i &\mapsto x_i. \end{aligned}$$

Dans l'exemple du dé à six faces, $X(\omega_i) = \omega_i$, alors que dans le cas du lancer d'une pièce de monnaie, nous pourrions définir $X(\text{pile}) = 0$ et $X(\text{face}) = 1$, ou bien $X(\text{pile}) = 42$ et $X(\text{face}) = -1$. Dans les cas précédents, le rôle de la variable aléatoire est seulement d'étiqueter les événements pour plus de convenance lors des manipulations mathématiques.

Toutefois, cette fonction peut parfois prendre un sens plus précis. Par exemple, imaginons qu'on lance une pièce de monnaie dix fois de suite et que la séquence obtenue est enregistrée. L'univers des possibilités contient alors 2^{10} séquences de pile ou face distinctes. Si nous utilisons la notation 0 pour « pile » et « 1 » pour face, l'ensemble contient des séquences comme 0000000000, 0101010101 et 1111100000.

Dans ce cas-ci, nous pourrions définir une variable aléatoire X dont la règle est de compter le nombre de fois que « face » est observé dans une séquence. Ainsi, $X(0000000000) = 0$, alors que $X(0101010101) = X(1111100000) = 5$. Dans ce cas-ci, $X(\omega_i) \in \{0,1,2,3,4,5,6,7,8,9,10\}$ et ne correspond pas à une étiquette unique pour chacun des éléments dans Ω.

Lorsqu'une probabilité est définie sur Ω et que nous avons défini une (ou plusieurs) variable aléatoire X sur Ω, une seconde probabilité, cette fois-ci associée aux valeurs possibles de la variable aléatoire, est induite. Cette dernière est définie comme

$$P_X(B) = P(X \in B) = P(\{\omega_i \in \Omega | X(\omega_i) \in B\}), \tag{1.7}$$

où $B \subseteq \mathbb{R}$. Le fait d'avoir une variable aléatoire permet de poser la question : quelle est la probabilité d'obtenir l'ensemble des événements où la valeur de X appartient à l'ensemble B ?

Dans le cas du dé, l'événement « obtenir un nombre pair » correspond à poser $B = \{2, 4, 6\}$, de sorte que l'on demande quelle est la probabilité d'obtenir l'ensemble des événements dont les valeurs associées à la variable aléatoire sont $2, 4$ et 6. Dans le cas d'une séquence de dix lancers d'une pièce de monnaie, on pourrait choisir $B = \{5\}$, faisant en sorte que l'on demande quelle est la probabilité qu'une séquence contienne exactement 5 fois « face ».

Au fil de ce mémoire, nous aurons recours à deux types de variables aléatoires, soient les variables aléatoires **discrètes** et **continues**. Dans un premier temps, une variable aléatoire discrète signifie simplement qu'elle prend un nombre fini de valeurs, comme dans les exemples précédents. Parallèlement, une variable aléatoire continue possède une infinité non dénombrable de valeurs possibles.

L'ensemble des valeurs que peut prendre la variable aléatoire discrète X se nomme le **support** et est noté $R_X = \{x_1, x_2, \ldots\}$, où les x_i sont les valeurs possibles de X. À partir du concept de support, nous pouvons définir la **fonction de masse**, $P(X = x_i)$, où l'indice i est une étiquette associée aux des valeurs dans R_X. Cette fonction doit alors respecter les propriétés suivantes :

1. $P(X = x) \geq 0$ pour tout $x \in \mathbb{R}$;
2. $P(X = x_i) > 0$ pour tout $x_i \in R_X$;
3. $\sum_{x_i \in R_X} P(X = x_i) = 1$.

Dans le cas du dé, où la variable aléatoire X représente seulement le résultat du lancer, nous avons $P(X = x_i) = \frac{1}{6}$ pour tout $x_i \in \{1, 2, 3, 4, 5, 6\}$ (propriété 2), alors que $P(X = 2.3) = 0$. Cette fonction de masse est représentée à la figure 1.4.

Dans un deuxième temps, la variable aléatoire **continue** possède plutôt un support R_X formé d'un ou de plusieurs intervalles de nombres réels. Dans ce cas-ci la fonction de masse est remplacée par une **fonction de densité** $f(x)$. Plus précisément, cette fonction doit respecter :

1. $f(x) \geq 0$ pour tout $x \in \mathbb{R}$;
2. $f(x) > 0$ pour tout $x_i \in R_X$;
3. $\int_{-\infty}^{\infty} f(x)dx = 1$.

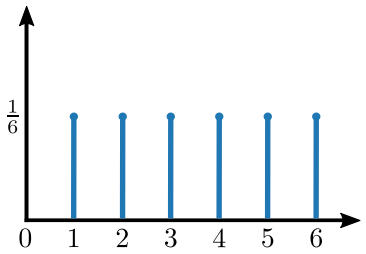

FIGURE 1.4 – Fonction de masse d'un dé à six faces.

Finalement, autant à partir d'une fonction de masse que d'une fonction de densité, nous pouvons construire la **fonction de répartition** $F(x)$ définie comme

$$F(x) = P(X \leq x), \tag{1.8}$$

pour $x \in \mathbb{R}$. Pour la fonction de répartition d'une variable aléatoire continue, il suffit de remplacer la fonction de masse par la fonction de densité. Dans le cas du dé, $F(x)$ correspond alors à la figure 1.5.

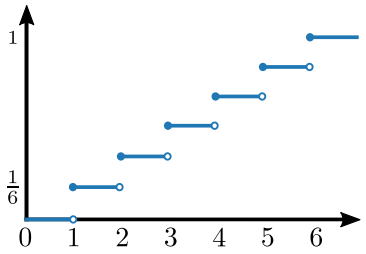

FIGURE 1.5 – Fonction de répartition d'un dé à six faces.

1.2.3 Lois de probabilités

Dans ce mémoire, nous aurons recours à des lois de probabilités (fonctions de masse et fonctions de densité) bien précises. La première loi présentée est à la base des deux autres, il s'agit de la **loi de Bernoulli**.

Considérons d'abord une expérience aléatoire dont l'univers des possibilités est seulement constitué de deux éléments, comme le lancer d'une pièce de monnaie. L'appellation commune pour ces deux issues sont « succès » et « échec » et représentées par 1 et 0 respectivement. La variable aléatoire associée à cette expérience désigne alors seulement l'obtention d'un succès ou d'un échec ($X \in \{0, 1\}$). La loi de Bernoulli s'écrit

$$P(X = x) = p^x(1 - p)^{x-1}, \tag{1.9}$$

pour $x \in \{0, 1\}$ et où p est la probabilité d'observer un « succès ». Pour la pièce de monnaie, $p = 0.5$, de sorte que la probabilité d'observer un « échec » (que l'on pourrait associer à la valeur « pile ») est $1 - p = 0.5$.

Si une expérience aléatoire possédant une loi de Bernoulli est répétée n fois de manière indépendante et que la variable aléatoire X représente le nombre de « succès » dans la séquence, la variable aléatoire est distribuée selon une **loi binomiale**. La fonction de masse associée s'écrit comme

$$P(X = x) = \binom{n}{x} p^x (1 - p)^{(n-x)}, \tag{1.10}$$

où p est la probabilité d'obtenir un « succès », $x \in \{0, 1, 2, \ldots, n\}$ est le nombre de « succès » dans la séquence et $\binom{n}{x} = \frac{n!}{x!(n-x)!}$ est le coefficient binomial qui dénombre le nombre de séquences uniques qui contiennent x « succès ». Par exemple, si nous lançons une pièce de monnaie trois fois de suite, il n'y a que trois séquences qui contiennent un seul succès, c'est-à-dire les séquences 100, 010 et 001, ce qui est vérifié par $\binom{3}{1} = \frac{3!}{1!2!} = 3$.

La troisième loi discrète qui nous intéresse correspond à la **loi multinomiale**, qui correspond à une généralisation de la loi binomiale pour y inclure plus de deux catégories. Autrement dit, au lieu des deux catégories « succès » et « échec », nous pouvons considérer $k \geq 2$ catégories. Cette fois-ci, nous nous intéressons aux variables aléatoires X_i, où $i = 1, \ldots, k$. Chaque X_i correspond au nombre de fois où la catégorie i a été observée dans la séquence de n tirages. Chacune des catégories peut se présenter avec une probabilité respective p_i, de sorte que la probabilité d'obtenir une séquence spécifique est

$$p_1^{x_1} p_2^{x_2} \ldots p_k^{x_k}, \tag{1.11}$$

où x_i, avec $i = 1, 2, \ldots k$, est le nombre d'éléments de la catégorie i survenus lors de l'expérience.

Pour compter le nombre de séquences où l'on retrouve x_1, x_2,..., x_k éléments dans chaque catégorie, il faut utiliser le coefficient multinomial dont l'équation est

$$\binom{n}{x_1 x_2 \ldots x_k} = \frac{n!}{x_1! x_2! \ldots x_k!}. \tag{1.12}$$

Pour décortiquer l'équation (1.12), imaginons que nous effectuons n expériences. Dans la séquence, il y a donc x_1 éléments de la catégorie 1, x_2 éléments de la catégorie 2 et ainsi de suite jusqu'aux x_k éléments dans la catégorie k avec $x_1 + x_2 + \ldots + x_k = n$. Le nombre de séquences de n tirages où $X_1 = x_1$ est alors donné par $\binom{n}{x_1}$. Or, si $X_1 = x_1$, cela signifie qu'il reste $n - x_1$ tirages dont les résultats sont partagés entre les catégories restantes. Parmi cette séquence de $n - x_1$ tirages, on compte $\binom{n-x1}{x_2}$ arrangements possibles où l'on retrouve x_2 expériences. Pour la catégorie x_3 on en retrouve $\binom{n-x_1-x_2}{x_3}$, et ainsi de suite.

Pour connaître le nombre de séquences qui respecte la condition $x_1 + x_2 + \ldots + x_k = n$, il suffit alors de multiplier le nombre de séquences où l'on retrouve x_1 éléments de la catégorie

1, par le nombre de séquences où l'on retrouve x_2 éléments de la catégorie 2, et ainsi de suite, de sorte que

$$\frac{n!}{x_1!(n-x_1)!} \cdot \frac{(n-x_1)!}{x_2!(n-x_1-x_2)!} \cdot \ldots \cdot \frac{(n-x_1-x_2-\ldots-x_{k-1})!}{x_k!(n-x_1-x_2-\ldots-x_k)!} = \frac{n!}{x_1!x_2!\ldots x_k!}. \quad (1.13)$$

Il ne reste alors qu'à multiplier ce nombre de séquences possibles avec la probabilité de tirer une séquence où $X_1 = x_1$, $X_2 = x_2$,..., $X_k = x_k$, et $x_1 + x_2 + \ldots + x_k = n$, pour obtenir la loi multinomiale à k catégories, soit

$$P(X_1 = x_1, X_2 = x_2, \ldots, X_k = x_k) = \frac{n!}{x_1!x_2!\ldots x_k!} p_1^{x_1} p_2^{x_2} \ldots p_k^{x_k}. \quad (1.14)$$

Par exemple, nous pourrions avoir un sac de billes dans lequel il y a une bille rouge, une bille bleue, une bille verte et une bille jaune. Nous nous intéressons alors à la séquence des couleurs que nous produisons en pigeant une bille, en observant sa couleur et en la replaçant dans le sac. Pour connaître la probabilité d'obtenir une séquence qui contient x_1 fois la couleur bleue, x_2 fois la couleur rouge, x_3 fois la couleur verte et x_4 fois la couleur jaune, il suffirait d'utiliser l'équation (1.14) avec $k = 4$.

Finalement, les dernières lois que nous présentons sont issues d'une même famille, c'est-à-dire qu'elles partagent la même expression mathématique générale, mais dont le profil change en modifiant un paramètre libre. Il s'agit de la famille des lois χ^2 dont le support de la variable aléatoire est maintenant continu et correspond à $R_X = [0, \infty)$. L'expression qui régit cette famille est

$$f(X = x) = \frac{1}{2^{\nu/2}\Gamma(\nu/2)} x^{\nu/2-1} e^{-x/2}, \quad (1.15)$$

pour où Γ correspond à la fonction gamma définie comme

$$\Gamma(\alpha) = \int_0^\infty x^{\alpha-1} e^{-x} dx. \quad (1.16)$$

Le paramètre libre qui sélectionne une distribution précise dans la famille est « ν ». Ce dernier est le nombre de **degrés de liberté** de la loi χ^2 et prend des valeurs entières strictement positives. Plus d'information concernant cette loi et son usage se retrouvent dans la section suivante.

1.2.4 Statistique

La statistique est un domaine qui utilise le vocabulaire des probabilités afin de caractériser un phénomène à partir d'observations. Ces observations correspondent aux résultats de variables aléatoires. Ainsi, en présence d'une expérience aléatoire dont le processus stochastique sous-jacent est inconnu, mais où nous pouvons tout de même générer des observations, les outils du domaine de la statistique peuvent nous aider à caractériser le processus.

Formellement, une **statistique** est d'ailleurs définie comme une fonction de n observations d'une variable aléatoire X. Cela signifie que nous avons répété l'expérience n fois et que nous

avons observé le résultat de X pour chaque expérience. Dans les cas qui nous intéressent, chaque expérience est indépendante l'une de l'autre, c'est-à-dire que le résultat d'une expérience n'influence pas le résultat de la prochaine expérience. Ces dernières sont aussi identiquement distribuées, ce qui signifie que c'est le même processus aléatoire qui permet de générer une observation à chaque fois.

La suite de n observations d'une variable aléatoire X s'appelle un **échantillon aléatoire** d'une variable. Le résultat pour chaque observation est noté x_i avec $i = 1, 2, \ldots, n$. Avec cette séquence d'observations, nous pouvons calculer une statistique. Par exemple la moyenne arithmétique qui est donnée par l'expression

$$\bar{X} = \frac{1}{n} \sum_{i=1}^{n} x_i. \tag{1.17}$$

Notons qu'une fonction d'une variable aléatoire est aussi une variable aléatoire. Cela signifie qu'une statistique possède aussi une fonction de masse ou une fonction de densité.

Malgré la variabilité associée aux statistiques, il demeure possible de caractériser le processus aléatoire qui génère nos observations. Cela ne signifie pas que nous pouvons le faire avec certitude à chaque fois, mais en prenant certaines précautions, nous pouvons considérer que la caractérisation effectuée est plausible. Par exemple, si nous voulions estimer l'espérance d'un lancer de dé, il serait inefficace de ne se fier qu'à deux lancers. Par contre, si nous effectuons 1 000 lancers, le résultat risque d'être plus prêt de sa valeur théorique.

1.2.5 Tests d'hypothèses

Grâce à des observations et des statistiques, il est possible d'utiliser une technique d'inférence afin de juger de la plausibilité de deux hypothèses par rapport au processus qui génère les données. Dans les cas qui nous intéressent, nous aurons affaire à deux tests, soit le test d'indépendance et le test d'ajustement.

Le **test d'indépendance** permet de déterminer si deux variables aléatoires X et Y (ou plus pour la version généralisée du test) sont indépendantes ou non. Supposons que nous avons les deux **variables aléatoires catégoriques**, c'est-à-dire que les résultats de la variable aléatoire X peuvent être divisés en I catégories $(1, 2, \ldots, I)$ et que les résultats de la variable aléatoire Y peuvent être divisés en J catégories $(1, 2, \ldots, J)$. En effectuant l'expérience aléatoire plusieurs fois, nous allons construire une **table de contingence** de dimension $I \times J$ où nous allons compter le nombre de fois où les variables aléatoires sont tombées dans un couple de catégories précis. Un exemple d'une telle table est donné à la table 1.1.

Dans cette table, les m_{ij} représentent le nombre de fois que le couple des catégories i et j s'est

	$Y=1$	$Y=2$	$\dots$	$Y=J$	Total
$X=1$	m_{11}	m_{12}	$\dots$	m_{1J}	m_{1+}
$X=2$	m_{21}	m_{22}	$\dots$	m_{2J}	m_{2+}
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
$X=I$	m_{I1}	m_{I2}	$\dots$	m_{IJ}	m_{I+}
Total	m_{+1}	m_{+2}	$\dots$	m_{+J}	m_{++}

TABLE 1.1 – Table de contingence pour les variables aléatoires X et Y divisées en I et J catégories respectivement.

présenté dans les données, tandis que

$$m_{+j} = \sum_{i=1}^{I} m_{ij}, \tag{1.18}$$

$$m_{i+} = \sum_{j=1}^{J} m_{ij}, \tag{1.19}$$

$$m_{++} = \sum_{j=1}^{J} \sum_{i=1}^{I} m_{ij}. \tag{1.20}$$

Nous pouvons également estimer les probabilités de retrouver un couple (i, j) en divisant les m_{ij} par le nombre total d'observations

$$p_{ij} = P(X = x_i, Y = y_j) = \frac{m_{ij}}{m_{++}}. \tag{1.21}$$

Plus le nombre d'observations est grand, plus cette proportion se rapproche de la probabilité théorique du processus aléatoire sous-jacent. Grâce à la table, nous pouvons aussi évaluer **les probabilités marginales**, c'est-à-dire

$$p_{i+} = P(X = x_i) = \sum_{j=1}^{J} p_{ij} = \frac{m_{i+}}{m_{++}}, \tag{1.22}$$

$$p_{+j} = P(Y = y_j) = \sum_{i=1}^{I} p_{ij} = \frac{m_{+j}}{m_{++}}. \tag{1.23}$$

Test d'indépendance

Avec ces outils en main, nous pouvons maintenant tester deux hypothèses à partir des observations, c'est-à-dire

H_0 : X et Y sont indépendantes,

H_1 : X et Y ne sont pas indépendantes.

Mathématiquement, l'hypothèse H_0, aussi appelée **hypothèse nulle**, signifie qu'il est plausible d'écrire

$$P(X = i, Y = j) = P(X = i)P(Y = j), \tag{1.24}$$

pour tous les $i = 1, 2, \ldots, I$ et $j = 1, 2, \ldots, J$. Autrement dit, la probabilité conjointe peut s'écrire comme un produit des probabilités marginales, puisqu'elles sont indépendantes. La seconde hypothèse, **l'hypothèse alternative**, suggère plutôt que

$$P(X = i, Y = j) \neq P(X = i)P(Y = j) \tag{1.25}$$

pour au moins une paire (i, j).

Afin de juger quelle situation est la plus plausible, nous devons construire la table de contingence des valeurs espérées, notées $\hat{m}_{ij}$, sous l'hypothèse nulle. Il suffit alors d'évaluer le produit

$$\hat{m}_{ij} = m_{++}p_{i+}p_{+j}, \tag{1.26}$$

qui correspond à multiplier le nombre d'observations par la probabilité conjointe sous le modèle d'indépendance pour chaque paire (i, j). Nous comparons ensuite la table des valeurs observées à la table des valeurs attendues en utilisant la statistique

$$\chi_0^2 = \sum_{i=1}^{I}\sum_{j=1}^{J} \frac{(m_{ij} - \hat{m}_{ij})^2}{\hat{m}_{ij}}. \tag{1.27}$$

Le terme au numérateur correspond à une distance entre les valeurs observées m_{ij} et les valeurs attendues $\hat{m}_{ij}$ et pénalise les écarts de manière non linéaire. Pour un « grand » [7] nombre d'observations, la distribution de la statistique χ_0^2 tend vers une loi χ^2 à $(I - 1) \times (J - 1)$ degrés de liberté. Cela signifie que le test d'indépendance est un **test asymptotique**, où l'on fait l'approximation que la distribution de la statistique est une loi χ^2, même si le nombre d'observations n'est pas infini.

Cette approximation demeure toutefois raisonnable dans le cas où nous avons un « grand » nombre d'observations. L'avantage de cette approximation est qu'elle permet d'utiliser l'équation (1.15) pour évaluer la probabilité d'obtenir une statistique au moins aussi extrême que χ_0^2, c'est-à-dire

$$P(Z \geq \chi_0^2) = 1 - F(Z \geq \chi_0^2), \tag{1.28}$$

où Z est une variable aléatoire dont la loi est χ^2 avec $\nu = (I - 1) \times (J - 1)$ et où F est la fonction de répartition de cette même loi. La probabilité $P(Z \geq \chi_0^2)$ est appelée valeur-p (*p-value*). Cette probabilité pour une loi du χ^2 à trois degrés de liberté est représentée à la figure 1.6.

7. Il semble que la notion de grandeur est mal définie dans les ouvrages consultés. Nous ne pouvons donc pas adéquatement spécifier un nombre précis que nous supposons « grand ».

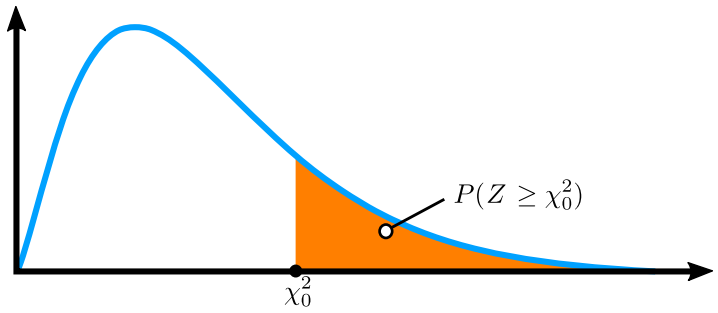

FIGURE 1.6 – Représentation graphique de la valeur-p de la statistique χ_0^2 pour une loi χ^2 à trois degrés de liberté.

Pour compléter le test d'hypothèse, la valeur-p est comparée à une valeur α, appelée seuil de critique du test, et correspond à la probabilité de rejeter l'hypothèse nulle alors qu'elle est vraie. Lorsque la valeur-p est inférieure à α, nous rejetons l'hypothèse nulle, alors que si elle est supérieure, nous ne pouvons pas rejeter l'hypothèse nulle.

Le non-rejet de l'hypothèse nulle ne signifie pas qu'elle est vraie, mais bien qu'elle semble plus plausible selon les données recueillies. Pareillement, si l'hypothèse nulle est rejetée, c'est que l'hypothèse alternative semble plus plausible, mais cela ne garantit pas qu'elle soit vraie. Il faut donc faire preuve de prudence en analysant les conclusions d'un test d'hypothèses et garder en tête qu'elles indiquent seulement ce qui semble le plus plausible à partir de nos données. Néanmoins, si la statistique est près de zéro (valeur-p près de 1), cela signifie que les observations sont près des valeurs attendues sous l'hypothèse nulle, d'où le fait que cette dernière hypothèse soit plausible. Dans le cas inverse, l'hypothèse nulle n'est pas plausible, signifiant que l'on doive se tourner vers l'hypothèse alternative.

Le fait que la statistique soit tirée d'une distribution χ^2 à $(I-1)\times(J-1)$ degrés de liberté s'explique comme ceci. Pour évaluer la table des valeurs attendues nous devons d'abord estimer les valeurs $\hat{m}_{i+}$ et $\hat{m}_{+j}$, qui sont respectivement égales à m_{i+} et m_{+j}. Le résultat de cette opération est que l'on vient fixer les entrées dans la marge de la table des valeurs espérées. Ensuite, supposons qu'il y a J catégories pour Y et que nous nous intéressons à la première ligne de la table de contingence. Puisque $m_{1+} = \sum_{j=1}^{J} m_{1j}$, cela signifie que si nous spécifions $J-1$ valeurs dans la première ligne de la table, nous pouvons évaluer la donnée manquante avec $m_{1k} = m_{1+} - \sum_{j=1;j\neq k}^{J} m_{1j}$, où m_{1k} est la valeur manquante. Le nombre de valeurs à spécifier correspond donc aux « degrés de liberté ». Lorsque nous avons une table de taille $I \times J$, il faut donc spécifier $(I-1)\times(J-1)$ valeurs pour en déduire le reste et la statistique associée tend vers une distribution χ^2 à $(I-1)\times(J-1)$ degrés de liberté.

Test d'ajustement

À la différence du test d'indépendance, le **test d'ajustement** permet plutôt d'évaluer s'il est plausible que des données d'un échantillon aléatoire soient distribuées selon une loi de probabilité précise, dont les paramètres sont spécifiés ou non. Autrement dit, on teste les hypothèses

$$\begin{aligned} &H_0 : X \text{ est distribuée selon } f(X;\theta), \\ &H_1 : X \text{ n'est pas distribuée selon } f(X;\theta), \end{aligned} \tag{1.29}$$

où $f(X;\theta)$ est la loi de probabilité et θ représente l'ensemble des paramètres de cette loi. Dans le cas qui nous intéresse, la loi $f(X;\theta)$ est une loi multinomiale dont les paramètres sont le nombre d'observations n et les probabilités de k catégories.

En effet, à partir d'une table de contingence qui présente $I \times J$ catégories, nous pouvons effectuer ce test pour déterminer s'il est plausible de croire que les observations soient distribuées selon une loi multinomiale précise. Pour effectuer ce test nous devons simplement comparer les observations m_{ij} aux observations espérées $\hat{m}_{ij}$ de la loi multinomiale en multipliant chacune des probabilités, p_{ij}, de la loi multinomiale par le nombre d'observations total m_{++}. Si les probabilités de la loi sont inconnues, il faut les estimer à l'aide de différentes techniques comme celle du maximum de vraisemblance. De la même manière que pour le test d'indépendance, nous comparons les effectifs observés avec les effectifs attendus grâce à la statistique (1.27).

Cette fois-ci, cette statistique est distribuée selon une distribution χ^2 à $k - r - 1$ degrés de liberté, où k correspond au nombre de catégories et r au nombre de paramètres estimés. Dans le cas où les paramètres sont spécifiés, $r = 0$ et nous retrouvons $k - 1$ degrés de liberté. Cela signifie que, sachant que nous avons n observations et que $\sum_{i,j} \hat{m}_{ij} = n$, nous devons spécifier $k - 1$ valeurs dans la table des valeurs espérées pour la caractériser au complet. L'évaluation de la valeur-p en relation au seuil critique α s'effectue alors de la même manière qu'au test précédent.

1.3 Topologie algébrique

L'une des manières de représenter les interactions d'ordre supérieur dans les réseaux est d'utiliser le formalisme des simplexes et des complexes simpliciaux. Pour bien comprendre ces structures, nous avons besoin de diverses notions mathématiques en lien avec la topologie. Les prochaines sections visent alors à établir ce cadre théorique pour décrire et caractériser les complexes simpliciaux. Les notions présentées sont principalement tirées des références [30, 45]. Les notions élémentaires nécessaires reliées à la théorie des groupes sont présentées dans l'Annexe A.

1.3.1 Complexes simpliciaux

Comme décrit précédemment, les complexes simpliciaux sont des hypergraphes. L'unité de base de ces structures s'appelle un **simplexe**. À l'origine, un simplexe est une construction géométrique qui permet de construire des polyèdres. En bref, un r-simplexe est la généralisation d'un triangle en r dimensions. Par exemple, un 0-simplexe est un point, un 1-simplexe est une ligne qui rejoint deux points, un 2-simplexe est un triangle plein et un 3-simplexe est un tétraèdre solide. Une représentation graphique de ces simplexes se trouve à la figure 1.7.

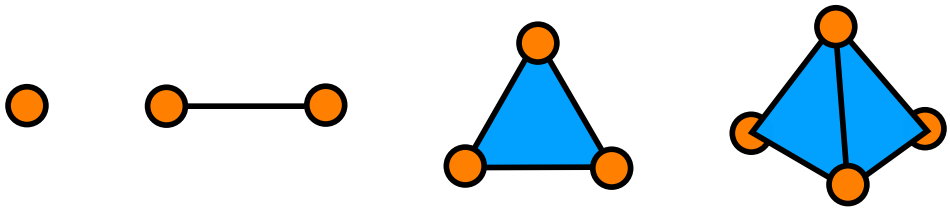

FIGURE 1.7 – De gauche à droite, un 0-simplexe (point), un 1-simplexe (ligne), un 2-simplexe (triangle plein) et un 3-simplexe (tétraèdre solide).

L'équation générale pour définir un r-simplexe se retrouve dans la référence [45], mais ne nous sera pas utile ici. Dans ce mémoire, on note un simplexe par la lettre σ et le représentons par un tuple de points. Par exemple, si σ est un r-simplexe, nous écrivons

$$\sigma = (p_0 p_1 p_2 \dots p_r), \tag{1.30}$$

où p_i pour $i = 0, 1, \dots, r$ sont les points qui le composent. Au sein d'un simplexe, chaque point ou groupe de points est aussi un simplexe. On parle alors des faces de σ. Dans l'exemple précédent, les faces (p_0) jusqu'à (p_r) sont des 0-simplexes, tandis que les faces $(p_0 p_1 p_2)$ et $(p_0 p_1 p_r)$ sont des 2-simplexes.

Nous pouvons également donner une orientation aux simplexes. Un simplexe orienté signifie que l'ordre des points spécifié est important et que pour se « déplacer » sur le simplexe d'un point à l'autre, nous devons suivre cet ordre. Par exemple, $(p_0 p_1 p_2)$ signifie que pour passer de p_0 à p_2, il faut d'abord passer par p_1. Une fois sur le point p_2, nous pouvons ensuite revenir sur le point p_0, car le tuple possède des frontières cycliques. D'ailleurs, les permutations cycliques des points sont équivalentes entre elles. En effet,

$$(p_0 p_1 p_2) = (p_2 p_0 p_1) = (p_1 p_2 p_0). \tag{1.31}$$

Nous pouvons aussi inverser l'orientation d'un simplexe avec un signe négatif

$$-(p_0 p_1 p_2) = (p_2 p_1 p_0). \tag{1.32}$$

Le déplacement sur le simplexe $(p_0 p_1 p_2)$ est illustré par les flèches dans la partie de gauche de la figure 1.8.

FIGURE 1.8 – À gauche, le 2-simplexe $(p_0p_1p_2)$ et, à droite, sa frontière donnée par $\partial_2(p_0p_1p_2) = (p_1p_2) + (p_0p_2) + (p_0p_1)$.

Un complexe simplicial K est alors un ensemble fini de simplexes qui répond aux deux conditions suivantes :

1. Les faces d'un simplexe $\sigma \in K$ appartiennent aussi à K,
2. Si σ et σ' sont deux simplexes appartenant à K, alors $\sigma \cap \sigma' = \emptyset$ ou $\sigma \cap \sigma' = \sigma_i$, où σ_i est une face commune à σ et σ'.

De plus, si les simplexes du complexe simplicial sont orientés, on dit que K est orienté.

La dimension d'un complexe simplicial est donnée par la dimension du plus grand simplexe qu'il contient. Ainsi, si nous construisons un complexe simplicial à partir de 0-simplexes et de 1-simplexes, la dimension du complexe simplicial est 1 et nous obtenons un réseau tel que celui décrit à la section 1.1. Un exemple de complexe simplicial de dimension 2 non orienté se trouve à la figure 1.3, tandis que la figure 1.9 représente un complexe simplicial orienté de dimension 1. [8]

1.3.2 Groupe des chaînes et frontière

Avec l'ensemble des r-simplexes d'un complexe simplicial, nous pouvons définir le **groupe des r-chaînes**, notées $C_r(K)$, d'un complexe simplicial orienté K. Plus précisément, $C_r(K)$ est un groupe abélien libre [9] généré par les r-simplexes orientés de K. Les éléments de $C_r(K)$ sont appelés des **r-chaînes**. En supposant qu'il y a N r-simplexes dans K, un élément c de $C_r(K)$ est défini comme

$$c = \sum_i^N c_i \sigma_i \tag{1.33}$$

où c_i est un nombre entier multipliant σ_i, qui dénote le i-ème r-simplexe dans K, avec $i = 1, 2, \ldots, N$. L'opération « + » sur deux éléments $c = \sum_i c_i\sigma_i$ et $c' = \sum_i c'_i\sigma_i$ de $C_r(K)$ est définie par

$$c + c' = \sum_i^N (c_i + c'_i)\sigma_i. \tag{1.34}$$

8. D'autres exemples apparaissent aussi aux figures 4.12 et 4.13.
9. Voir l'Annexe A pour une définition de l'expression « groupe abélien libre ».

L'élément neutre est alors

$$0 = \sum_i 0 \cdot \sigma_{r,i}, \tag{1.35}$$

tandis que l'élément inverse de c est

$$-c = \sum_i (-c_i)\sigma_{r,i}. \tag{1.36}$$

Si le complexe simplicial ne contient pas de r-simplexes, ce qui n'est possible que si $r > \dim K$, nous définissons $C_r(K)$ comme 0. Dans ce mémoire, lorsqu'un groupe de r-chaînes ne contient aucun simplexe, nous allons dire qu'il est **vide**.

Puisque l'opération « + » est commutative et que le groupe $C_r(K)$ est généré par N simplexes, $C_r(K)$ est un groupe abélien libre de rang N. Il est alors possible de définir un isomorphisme [10] f tel que

$$f : C_r(K) \cong \underbrace{\mathbb{Z} \oplus \ldots \oplus \mathbb{Z}}_{N}. \tag{1.37}$$

La règle précise de l'isomorphisme f est

$$c = \sum_i^N c_i \sigma_i \mapsto (c_1, c_2, \ldots, c_N). \tag{1.38}$$

Pour aller plus loin, il faut maintenant définir l'**opérateur de frontière** ∂_r agissant sur un r-simplexe σ_r pour obtenir la frontière dudit simplexe. Parallèlement, cela permettra d'introduire les concepts de frontière d'un r-simplexe et de cycle. Tout d'abord, pour $r = 0$, la frontière est définie comme étant nulle. L'application de ∂_0 sur les 0-simplexes donne systématiquement zéro. Pour $r > 0$, l'opérateur ∂_r est défini par

$$\partial_r \sigma_r \equiv \sum_{i=0}^{r} (-1)^i (p_0 p_1 \ldots \hat{p_i} \ldots p_r), \tag{1.39}$$

où $\hat{p_i}$ signifie que le point p_i est omis du simplexe résultant. Par exemple, le 2-simplexe $(p_0 p_1 p_2)$ est une chaîne dans le groupe $C_2(K)$. Si nous appliquons l'opérateur frontière sur ce dernier, nous obtenons l'équation

$$\partial_2(p_0 p_1 p_2) = (p_1 p_2) - (p_0 p_2) + (p_0 p_1) = (p_1 p_2) + (p_2 p_0) + (p_0 p_1). \tag{1.40}$$

Graphiquement, l'addition des 1-simplexes forme un triangle vide, ce qui correspond à l'enveloppe du 2-simplexe représentée à la figure 1.8.

Dans des cas comme celui-ci, l'opérateur ∂_r nous indique quels sont les $(r-1)$-simplexes qui délimitent le r-simplexe. Autrement dit, nous obtenons la **frontière** du r-simplexe. L'opérateur de frontière peut aussi agir sur un élément d'un groupe des chaînes quelconque. Par exemple,

10. Voir l'Annexe A pour une définition du terme « isomorphisme ».

si $c = \sum_i c_i \sigma_{r,i} \in C_r(k)$, où l'indice r indique la dimension du simplexe et l'indice i est une étiquette allant de 1 à N, alors

$$\partial_r c = \sum_{i=1}^{N} c_i \partial_r \sigma_{r,i}, \tag{1.41}$$

où le membre de droite est un élément du groupe des chaînes $C_{r-1}(K)$, puisque l'opérateur a pour effet de décomposer un r-simplexe en ses $\binom{r+1}{r}$ faces à $r-1$ dimensions. Plus formellement, ∂_r est alors une application [11]

$$\partial_r : C_r(K) \to C_{r-1}(K). \tag{1.42}$$

Cet opérateur est d'ailleurs un homomorphisme, ce qui se prouve rapidement puisque l'opération est la même dans les deux groupes et que ∂_r est un opérateur linéaire de sorte que pour $x, y \in C_r(K)$, nous avons

$$\partial_r(x + y) = \partial_r(x) + \partial_r(y). \tag{1.43}$$

L'intérêt des groupes des chaînes et des opérateurs de frontière est de définir le **complexe des chaînes**, c'est-à-dire une séquence de groupes abéliens libres et d'homomorphismes pour un complexe simplicial K. Plus précisément, pour un complexe simplicial K de dimension n, il est possible d'écrire le complexe des chaînes comme

$$0 \xrightarrow{\partial_{n+1}} C_n(K) \xrightarrow{\partial_n} C_{n-1}(K) \xrightarrow{\partial_{n-1}} \dots \xrightarrow{\partial_2} C_1(K) \xrightarrow{\partial_1} C_0(K) \xrightarrow{\partial_0} 0. \tag{1.44}$$

La raison derrière l'élaboration de cette structure est de pouvoir étudier les **r-cycles** ainsi que les **r-frontières** d'un complexe simplicial. Premièrement, les r-cycles sont les $c \in C_r(k)$ tels que

$$\partial_r c = 0. \tag{1.45}$$

L'ensemble des r-cycles est noté $Z_r(K)$ et il est un sous-ensemble de $C_r(K)$. $Z_r(K)$ est d'ailleurs appelé le groupe des r-cycles et correspond à $\ker \partial_r$. [12]

Deuxièmement, les r-frontières sont définies comme les éléments $c \in C_r(K)$ tels que

$$c = \partial_{r+1} d, \tag{1.46}$$

où $d \in C_{r+1}(K)$. L'ensemble des r-frontières, noté $B_r(K)$, est un sous-ensemble de $C_r(K)$ appelé le groupe des r-frontières. Par rapport à l'opérateur de frontière, $B_r(K)$ est donc $\operatorname{im} \partial_{r+1}$. [13]

Il existe une relation fondamentale entre ces groupes qui permet de définir intuitivement ce qu'est un « trou » dans le complexe simplicial et rigoureusement ce qu'est un groupe d'homologie. En effet, il se trouve que la composition de deux opérateurs de frontière consécutifs

11. Voir l'Annexe A pour une définition du terme « application ».
12. Le noyau (*kernel*), noté ker, est défini dans l'Annexe A.
13. L'image, notée im, est définie dans l'Annexe A.

dans le complexe des chaînes, appliquée sur les générateurs de $C_{r+1}(K)$, donne un résultat nul. Plus précisément,

$$\partial_r \circ \partial_{r+1}(\sigma_{r+1}) = 0, \tag{1.47}$$

pour tout $\sigma_{r+1} \in C_{r+1}(K)$, où $\sigma_{r+1} = (p_0 \dots p_{r+1})$. Ce résultat implique alors que $B_r(K) \subset Z_r(K)$, puisque l'application du premier opérateur sur les chaînes renvoie leur frontière et que la deuxième application envoie ces frontières à zéro. Autrement dit, les frontières des chaînes dans $C_{r+1}(K)$ sont des cycles dans $C_r(K)$.

1.3.3 Groupes d'homologie et nombres de Betti

Grâce à l'équation (1.47), nous pouvons définir intuitivement ce qu'est un « trou » dans un complexe simplicial. En effet, un trou est un cycle dans $C_r(K)$ qui n'est pas la frontière d'une chaîne dans $C_{r+1}(K)$. Par exemple, à la figure 1.9, les simplexes (p_0p_1), (p_1p_2) et (p_2p_0) forment un cycle, mais ce cycle n'est pas la frontière d'une chaîne de dimension plus élevée, car $C_r(K) = 0$ pour $r \geq 2$. Nous avons donc identifié un trou. Par contre, dans la situation présentée à la figure 1.8, ces mêmes 1-simplexes forment aussi un cycle, mais ce cycle est la frontière du 2-simplexe $(p_0p_1p_2)$. Ils ne forment donc pas un trou dans $C_1(K)$.

Mathématiquement, un trou est élément de $Z_r(K)/B_r(K)$. Ce groupe quotient [14] est d'ailleurs un invariant topologique appelé groupe d'homologie. En présence d'un complexe simplicial K de dimension n, le r-ième **groupe d'homologie** $H_r(K)$, avec $0 \leq r$, est défini comme

$$H_r(K) \equiv Z_r(K)/B_r(K), \tag{1.48}$$

où $H_r(K)$ est un groupe quotient.

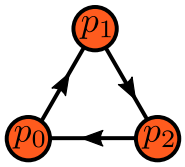

FIGURE 1.9 – Complexe simplicial orienté $K = \{p_0, p_1, p_2, (p_0p_1), (p_1p_2), (p_2p_0)\}$

Grâce à tous ces éléments, il est possible de définir le r-ième **nombre de Betti** pour un complexe simplicial K, noté $\beta_r(k)$, comme

$$\beta_r(K) = \dim\ H_r(K), \tag{1.49}$$

c'est-à-dire que $\beta_r(K)$ est le rang de $H_r(K)$. Intuitivement, nous pouvons aussi décrire le nombre de Betti $\beta_r(K)$ comme étant le nombre de cycles circonscrits par des r-simplexes qui

14. Voir l'Annexe A pour une définition de l'expression « groupe quotient ».

ne sont pas la frontière d'une $(r+1)$-chaîne. Cela n'est vrai que pour $r \geq 1$, car $\beta_0(K)$ indique plutôt le nombre de **composantes** dans le complexe simplicial. Plus précisément, une composante est un sous-ensemble de noeuds entre lesquels il existe au moins un chemin dans le complexe simplicial, c'est-à-dire une suite de liens. De plus, pour que l'ensemble corresponde à une composante, aucun noeud du complexe simplicial ne peut lui être ajouté arbitrairement sans briser cette propriété [47]. À la figure 1.10, le complexe simplicial possède deux composantes, de sorte que $\beta_0 = 2$.

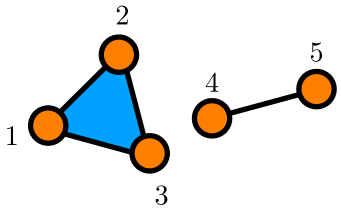

FIGURE 1.10 – Complexe simplicial à 5 noeuds ayant deux composantes.

Voici alors un exemple du calcul des nombres de Betti pour le complexe simplicial $K = \{p_0, p_1, p_2, (p_0p_1), (p_1p_2), (p_2p_0)\}$, représenté à la figure 1.9. Le complexe des chaînes associé à K est

$$C_2(K) \xrightarrow{\partial_2} C_1(K) \xrightarrow{\partial_1} C_0(K) \xrightarrow{\partial_0} 0, \tag{1.50}$$

où C_r, avec $r \geq 2$ sont vides, C_1 contient (p_0p_1), (p_1p_2) et (p_2p_0), et C_0 contient p_0, p_1 et p_2. Le premier groupe d'homologie est alors donné par $H_0 = Z_0/B_0 = \ker(\partial_0)/\text{im }(\partial_1)$. Puisque tout point dans C_0 appartient au noyau de ∂_0, $Z_0 = \ker(\partial_0) = C_0$. En ce qui concerne l'image de ∂_1, nous avons

$$\begin{aligned} \text{im }(\partial_1) &= \partial_1 \left[a(p_0p_1) + b(p_1p_2) + c(p_2p_0)\right] \\ &= a(p_1 - p_0) + b(p_2 - p_1) + c(p_0 - p_2) \\ &= (c-a)p_0 + (a-b)p_1 + (b-c)p_2 \\ &= \alpha p_0 + \beta p_1 + \gamma p_2, \end{aligned} \tag{1.51}$$

où nous avons posé $\alpha = (c-a)$, $\beta = (a-b)$ et $\gamma = (b-c)$. L'avantage d'avoir exprimé l'image de ∂_1 comme à la dernière ligne est que nous pouvons construire un système d'équations pour trouver la base de im (∂_1). En effet, en additionnant α et β, nous obtenons

$$\alpha + \beta = c - a + a - b = c - b = -\gamma, \tag{1.52}$$

ce qui nous indique que γ est un paramètre redondant. De ce fait, nous pouvons récrire

l'équation (1.51) comme

$$\begin{aligned}\alpha p_0 + \beta p_1 + \gamma p_2 &= \alpha p_0 + \beta p_1 - (\alpha + \beta) p_2 \\ &= \alpha(p_0 - p_2) + \beta(p_1 - p_2) \\ &= \alpha p' + \beta p'', \end{aligned} \tag{1.53}$$

où nous avons posé $p' = (p_0 - p_2)$ et $p'' = (p_1 - p_2)$. Cela signifie alors que nous pouvons générer toute l'image de ∂_1 grâce à l'équation (1.53). Autrement dit

$$\text{im } (\partial_1) = \left\{ \alpha p' + \beta p'' | \alpha, \beta \in \mathbb{Z} \right\}. \tag{1.54}$$

De ce fait, nous avons la relation

$$\text{im } (\partial_1) \cong \mathbb{Z} \oplus \mathbb{Z}, \tag{1.55}$$

avec l'isomorphisme

$$\begin{aligned} \phi \colon \text{im } (\partial_1) &\longrightarrow \mathbb{Z} \oplus \mathbb{Z} \\ \alpha p' + \beta p'' &\longmapsto (\alpha, \beta). \end{aligned} \tag{1.56}$$

De même, grâce à la relation (1.37), nous avons que $C_0 \cong \mathbb{Z} \oplus \mathbb{Z} \oplus \mathbb{Z}$. Ainsi,

$$H_0 = [\mathbb{Z} \oplus \mathbb{Z} \oplus \mathbb{Z}] \,/\, [\mathbb{Z} \oplus \mathbb{Z}] \cong \mathbb{Z}. \tag{1.57}$$

De la même manière, nous pouvons trouver $H_1 = Z_1/B_1 = \ker(\partial_1)/\text{im } (\partial_2)$. Dans ce cas-ci, im (∂_1) est le groupe trivial puisque C_2 est vide. Par contre, nous avons que

$$\begin{aligned}\ker(\partial_1) &= \partial_1 \left[a(p_0 p_1) + b(p_1 p_2) + c(p_2 p_0) \right] \\ &= a(p_1 - p_0) + b(p_2 - p_1) + c(p_0 - p_2) \\ &= (c - a) p_0 + (a - b) p_1 + (b - c) p_2 \\ &= 0. \end{aligned} \tag{1.58}$$

Cela implique alors que

$$(c - a) = 0 \Rightarrow c = a \tag{1.59}$$

$$(a - b) = 0 \Rightarrow a = b \tag{1.60}$$

$$(b - c) = 0 \Rightarrow b = c, \tag{1.61}$$

de sorte que $a = b = c$ et

$$\ker(\partial_1) = \left\{ a \left[(p_0 p_1) + (p_1 p_2) + (p_2 p_0) \right] | a \in \mathbb{Z} \right\} \cong \mathbb{Z}. \tag{1.62}$$

Ainsi, nous avons que

$$H_1 = \mathbb{Z}/\{0\} = \mathbb{Z}. \tag{1.63}$$

Les nombres de Betti associés à chacun des groupes sont alors $\beta_0 = 1$ et $\beta_1 = 1$, ce qui était attendu, puisque K ne comporte qu'une composante et qu'il n'existe qu'un cycle formé par des 1-simplexes qui n'est pas la frontière d'une 2-chaîne. Pour $r > 1$, $\beta_r = 0$, car tous les groupes des chaînes de dimension r sont vides. Il est à noter que nous pouvons utiliser des théorèmes pour obtenir plus rapidement Z_1 et B_1 [45]. Le tout a été présenté de cette manière, car cette approche permet de mieux comprendre ce qui se produit lors du calcul d'un groupe d'homologie.

C'est donc avec tous ces outils en main, autant en ce qui concerne les réseaux, les statistiques et l'homologie, que nous pouvons approcher la première méthode d'inférence d'interactions d'ordre supérieur qui est présentée dans le prochain chapitre.

Chapitre 2

Inférence d'interactions d'ordre supérieur : modèle simplicial des configurations et homologie significative

La première approche que nous avons testée pour l'inférence d'interactions d'ordre supérieur dans les données de présence/absence est indirecte. En effet, au lieu d'inférer les interactions, nous supposons leur présence par défaut et construisons un complexe simplicial à partir des simplexes obtenus. Un modèle génératif de complexes simpliciaux, similaire au modèle des configurations, est alors utilisé pour échantillonner aléatoirement des complexes simpliciaux. L'homologie des complexes simpliciaux de l'échantillon est alors comparée à l'homologie du complexe simplicial obtenu à partir des données de présence/absence. S'il s'avère que l'homologie de ce dernier est significativement différente de celle des complexes de l'échantillon, nous pouvons supposer que l'hypothèse de départ a induit une structure non aléatoire, laissant présager que les interactions d'ordre supérieur sont véridiques.

Dans ce chapitre, nous présentons d'abord le modèle génératif utilisé, c'est-à-dire le modèle simplicial des configurations. Par la suite, nous expliquons plus en profondeur comment nous pouvons nous en servir comme modèle nul. Ensuite, nous présentons le calcul numérique des nombres de Betti à l'aide du paquetage GUDHI ainsi qu'une technique pour déterminer les nombres de Betti non triviaux. Finalement, nous rassemblons tous ces éléments pour traiter les jeux de données.

2.1 Modèle simplicial des configurations

En science des réseaux, il existe de nombreux outils pour tester les modèles nuls. Par contre, au niveau des complexes simpliciaux, l'éventail de méthodes est un peu moins développé. En 2017, Young et al. [59] ont généralisé le modèle de configuration présenté à la section 1.1.2 afin de pouvoir générer des complexes simpliciaux aléatoirement en préservant certaines propriétés. Ce modèle simplicial des configurations (*Simplicial Configuration Model* abrégé SCM), sert de modèle nul pour analyser un complexe simplicial. Voici les rouages de cet outil que nous avons tirés de la référence [59].

2.1.1 Construction du SCM

D'abord, pour un complexe simplicial K, nous définissons les **facettes** comme étant des simplexes qui ne sont pas inclus dans des simplexes de taille supérieure. En contrepartie, les simplexes qui sont contenus au sein d'autres simplexes sont appelés les **faces**. Autrement dit, si K contient le simplexe $\sigma = (p_1, p_2)$ et $\tau = (p_1, p_2, p_3)$, nous disons que la facette τ possède la face σ. La structure d'un complexe simplicial est alors entièrement spécifiée par la liste de ses facettes [59].

Nous définissons ensuite le degré généralisé d_i d'un noeud v_i comme étant le nombre de facettes qui contiennent v_i [17]. Nous définissons aussi la taille s_i d'une facette σ_i comme étant le nombre de noeuds qu'elle contient, c'est-à-dire sa dimension plus un. Nous rassemblons chacune de ces informations dans les vecteurs $\mathbf{d} = (d_1, \ldots, d_n)$ et $\mathbf{s} = (s_1, \ldots, s_f)$, où n correspond au nombre de 0-simplexes dans K et f correspond au nombre de facettes dans K.

Le SCM est alors basé sur l'ensemble $\Omega(\mathbf{d}, \mathbf{s})$ des complexes simpliciaux possédant une séquence de degrés $\mathbf{d}$ et une séquence de tailles des facettes $\mathbf{s}$. Le modèle étant microcanonique, la probabilité de piger l'un de ces complexes simpliciaux est

$$P(K; \mathbf{d}, \mathbf{s}) = \begin{cases} \frac{1}{|\Omega(\mathbf{d},\mathbf{s})|} & \text{si } K \text{ possède les séquences } \mathbf{d} \text{ et } \mathbf{s} \text{ .} \\ 0 & \text{autrement,} \end{cases} \tag{2.1}$$

où $|\Omega(\mathbf{d}, \mathbf{s})|$ désigne le cardinal de l'ensemble.

Dès lors, si nous limitons la taille des facettes à 1 ou 2, le degré généralisé est équivalent à la définition « réseau » du degré et les complexes simpliciaux dans l'ensemble $\Omega(\mathbf{d}, \mathbf{s})$ correspondent à des réseaux. Nous retrouvons alors le modèle des configurations comme nous l'avons défini dans la section 1.1.2.

L'échantillonnage du SCM passe par le fait qu'il existe une représentation des complexes simpliciaux en réseaux bipartis en se basant sur la liste des facettes. En effet, nous pouvons considérer que chacune des facettes est représentée par un noeud contenu dans l'ensemble F, alors que les 0-simplexes à la base du complexe simplicial sont contenus dans l'ensemble V. Il

y a connexion entre ces deux types de noeuds si et seulement si la facette $\sigma_i \in F$ contient le noeud v_j dans le complexe simplicial [59]. Un exemple est représenté à la figure 2.1.

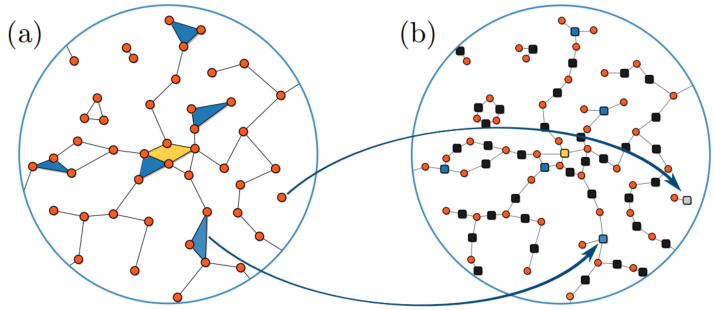

FIGURE 2.1 – (a) Complexe simplicial et (b) sa représentation en réseau biparti. Dans (b), les noeuds carrés représentent des facettes tandis que les noeuds oranges représentent les 0-simplexes du complexe simplicial. Image tirée de la référence [59].

Même s'il existe une représentation des complexes simpliciaux en réseaux bipartis, il n'est pas vrai que tous les réseaux bipartis représentent un complexe simplicial. En effet, les auteurs de la référence [59] pointent deux cas problématiques illustrés à la figure 2.2. Premièrement, pour deux noeuds dans F, il ne faut pas que l'ensemble des voisins de l'un des noeuds appartiennent aussi au second. Si c'est le cas, cela signifie que le noeud qui possède le moins de voisins n'est pas une facette, mais bien une face. Or, par définition, une facette ne peut contenir une facette et il y a incohérence.

Deuxièmement, nous ne pouvons pas permettre à une paire de noeuds d'être connectée par plus d'un lien. Si c'était le cas, le degré généralisé du 0-simplexe impliqué compterait plus qu'une fois la facette qui le contient. Toutefois, dans la représentation en complexe simplicial, cette même facette n'apparaît qu'une seule fois. De ce fait, le degré généralisé de ce 0-simplexe dans la représentation en réseau biparti est supérieur à son degré généralisé dans la représentation en complexe simplicial. Puisque les deux représentations ne préservent pas les degrés généralisés, elles ne sont pas équivalentes.

Malgré ces contraintes, un algorithme, basé sur la méthode de Monte-Carlo par chaîne de Markov (*Monte-Carlo Markov Chain*, abrégé MCMC), a été développé pour échantillonner l'espace des graphes bipartis qui préservent la séquence des degrés généralisés et la séquence des tailles des facettes du complexe simplicial original [59]. L'algorithme est d'ailleurs public et peut être consulté et utilisé en visitant la référence [58].

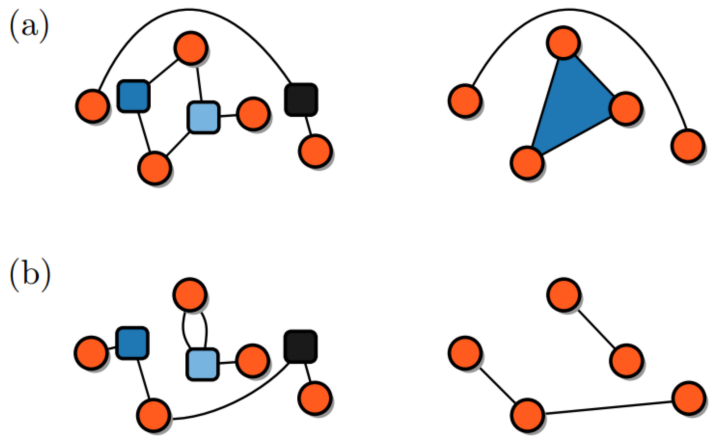

FIGURE 2.2 – Les deux graphes bipartis (colonne de gauche) possèdent les séquences $(\mathbf{d}, \mathbf{s}) = ([2, 2, 1, 1, 1]\,,\, [3, 2, 2])$, mais les complexes simpliciaux obtenus possèdent les séquences $(\mathbf{d}', \mathbf{s}') = ([1, 1, 1, 1, 1]\,,\, [3, 2])$ et $(\mathbf{d}', \mathbf{s}') = ([2, 1, 1, 1, 1]\,,\, [2, 2, 2])$, respectivement. En (a) l'une des facettes est en fait une face, tandis qu'en (b) il y a une paire de noeuds connectée par plus d'un lien. Image tirée de la référence [59].

2.1.2 Modèle nul

Avec ces outils en main, nous pouvons utiliser le SCM comme modèle nul. En effet, chaque observation générée par le SCM préserve les degrés généralisés des noeuds du complexe simplicial étudié ainsi que la taille de ses facettes. Nous pouvons donc comparer d'autres caractéristiques du complexe simplicial original à celles de l'ensemble de complexes simpliciaux échantillonné.

Dans le cas des données de présence/absence, nous n'avons pas directement accès à un complexe simplicial, mais bien à un réseau biparti. A priori, rien ne garantit que ce réseau respecte les conditions pour que sa représentation en complexe simplicial soit valide. Par exemple, il est possible qu'un ensemble d'entités présentes sur un site soit un sous-ensemble d'entités qui se retrouvent sur un second site. Nous sommes donc en présence d'un cas problématique, où une facette est en fait une face comme à la figure 2.2 (a). Heureusement, les auteurs de la référence [59] ont inclus des méthodes numériques pour transformer une matrice de biadjacence en liste de facettes où les facettes problématiques sont retirées [58].

De ce fait, nous pouvons transformer notre réseau biparti en complexe simplicial. Toutefois, s'il advient qu'une facette soit retirée de la liste, la réduction du nombre de facettes implique une réduction du nombre de sites. De plus, cette transformation se base sur une hypothèse quant à la présence d'interactions d'ordre supérieur dans les données. En effet, si n entités appartiennent à un site et que ces dernières ne sont pas sous-ensemble des entités sur un autre site, nous disons que les n entités forment un $(n-1)$-simplexe.

Une fois la liste de facettes obtenue, nous pouvons l'injecter dans l'algorithme de la référence

[58] pour générer un échantillon de complexes simpliciaux qui préservent la séquence des degrés généralisés et la taille des facettes. Considérant cet ensemble de complexes simpliciaux, le modèle nul peut être utilisé pour comparer l'homologie du complexe simplicial original à l'ensemble généré. L'idée derrière cette comparaison est qu'une différence significative est attendue entre les nombres de Betti d'un complexe simplicial « organisé » et ceux de l'ensemble aléatoire [59]. Le calcul numérique de ces quantités est présenté à la section 2.2.

Considérant cela, l'utilisation du SCM ne permet pas de dénicher directement des interactions d'ordre supérieur. Il faut d'abord faire l'hypothèse que les simplexes existent dans les données et, s'il s'avère que les nombres de Betti du complexe simplicial résultant se distinguent de l'ensemble aléatoire, nous pouvons seulement conclure que l'homologie du complexe simplicial ne semble pas aléatoire [59]. Par extension, cela suggère, mais ne confirme en rien, que d'avoir imposé les interactions d'ordre supérieur avec l'hypothèse ci-dessus est plausible. Au sens de l'inférence d'interactions d'ordre supérieur, cette démarche ne correspond donc qu'à une étape préliminaire qui pourrait suggérer une organisation non aléatoire des données.

2.2 Calcul numérique des nombres de Betti

Les complexes simpliciaux générés par le SCM ne sont généralement pas aussi simples que ceux des exemples de la section 1.3. Le calcul des nombres de Betti à la main est alors peu approprié. Heureusement, quelques outils numériques existent pour traiter ces cas fastidieux, comme *Perseus* [41], Dionysus 2 [42], PHAT [8] et GUDHI [12]. Parmi ces outils, nous avons décidé d'utiliser la bibliothèque logicielle GUDHI (version 2.3.0), car elle contient une structure de donnée, le *simplex Tree*, qui permet de générer un complexe simplicial à partir d'une liste de facettes [13]. De plus, une fois le complexe simplicial en mémoire, il est possible d'utiliser des fonctions pour mesurer les nombres de Betti. Toutefois, quelques problèmes ont été découverts lors de l'application de ces fonctions. Dans les prochaines sections, nous présentons les fonctions utilisées dans GUDHI, les problématiques rencontrées, ainsi que les solutions apportées.

2.2.1 GUDHI

Voici une brève description de *GUDHI* et de ses arbres simpliciaux (*Simplex Trees*). Nous prenons le temps d'expliquer ces éléments, car l'élaboration de solutions en lien avec les problèmes rencontrés passe par une compréhension des structures et des méthodes proposées dans le paquetage.

Pour commencer à construire un complexe simplicial, il suffit d'initialiser un arbre simplicial, qui correspond à un complexe simplicial vide. Par la suite, des simplexes sont insérés dans cette structure à l'aide de la méthode *insert*. En effectuant cette insertion, l'arbre simplicial ajoute automatiquement toutes les combinaisons de noeuds de taille égale ou inférieur à celle du simplexe. Plus précisément, si nous ajoutons la facette $\sigma = (123)$, alors les simplexes

(1), (2), (3), (12), (23) et (31) sont ajoutés automatiquement [13]. Autrement dit, l'arbre simplicial s'assure de générer et de garder mémoire toutes les faces d'un simplexe. De cette manière, l'objet résultant est bien un complexe simplicial.

À partir d'un arbre simplicial, nous pouvons également obtenir le j-squelette (j-*skeleton*) du complexe simplicial. Plus précisément, le j-squelette correspond à réduire tous les simplexes de dimension d plus grande que j en leurs $\binom{d+1}{j+1}$ faces de dimension j [13]. Par exemple, prendre le 1-squelette d'un 2-simplexe correspond à le réduire en trois 1-simplexes. Ce processus détruit alors l'information sur les facettes de dimension supérieure à j. En termes d'homologie, cela signifie que tous les groupes des chaînes d'indice plus grand que j sont maintenant vides. Toutefois, les groupes d'homologie d'indice inférieur à j ne sont pas modifiés. Un complexe simplicial de dimension 2 ainsi que son 1-squelette sont représentés à la figure 2.3

FIGURE 2.3 – (a) Complexe simplicial de dimension 2. (b) 1-squelette du complexe simplicial en (a).

À partir de ces structures, certaines mesures peuvent être prises, comme leur dimension, le nombre de simplexes ou le nombre de noeuds. Toutefois, une mesure utile pour caractériser la topologie du complexe simplicial est celle des nombres de Betti. Le comportement des méthodes utilisées pour obtenir ces nombres sera décrit dans la section suivante en raison de certains aspects problématiques.

2.2.2 Problématique : taille du complexe simplicial

Après avoir généré une liste de facettes avec les données de présence/absence, nous avons tout ce qu'il faut pour mesurer les nombres de Betti sur le complexe simplicial construit à l'aide de GUDHI. Or, en tentant de construire un complexe simplicial à partir des données d'OTUs, nous rencontrons rapidement un problème de mémoire.

En effet, comme mentionné précédemment, GUDHI doit garder en mémoire toutes les faces d'une facette. Ainsi, pour une facette de taille S insérée dans le complexe simplicial, il y a

$$\sum_{i=1}^{S} \binom{S}{i} = 2^S - 1, \tag{2.2}$$

simplexes en mémoire. En guise d'exemple, les facettes les plus grandes pour les données d'OTU complètes et les données filtrées sont respectivement 696 et 389. Cela représente plus de 10^{209} et 10^{117} objets en mémoire, seulement pour ces deux facettes. Dans nos observations, la quantité de mémoire vive nécessaire pour construire ces simplexes dépasse largement les quantités disponibles pour un ordinateur conventionnel. Il n'est donc pas possible de construire un complexe simplicial et de mesurer les nombres de Betti.

Pour de tels cas, il existe toutefois une solution susceptible de fonctionner. En effet, il est parfois possible de construire un complexe simplicial qui contient moins de simplexes, mais dont les groupes d'homologie sont identiques [50]. À cette fin, il faut d'abord transformer chaque facette du premier complexe simplicial en un noeud dans un second complexe simplicial. Un groupe de $k+1$ noeuds dans le second complexe simplicial forme un k-simplexe si les $k+1$ facettes correspondantes dans le premier complexe simplicial partagent au moins un noeud. Nous avons alors un second complexe simplicial dont l'homologie est équivalente au premier [50]. Il est toutefois à noter que cette transformation ne garantit pas que le second complexe simplicial possède des facettes de taille plus petite que le premier [50].

Par exemple, à la figure 2.4, le complexe simplicial de gauche possède trois 2-simplexes, neuf 1-simplexes et six 0-simplexes. En effectuant la transformation proposée, nous obtenons un complexe simplicial qui possède seulement trois 1-simplexes et trois 0-simplexes. Chacun des deux complexes simpliciaux possède aussi un cycle borné par trois 1-simplexes, impliquant qu'ils ont la même homologie.

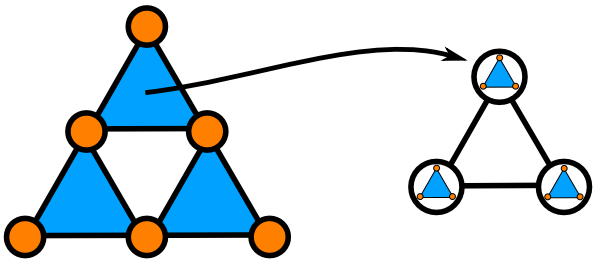

FIGURE 2.4 – À gauche, un complexe simplicial possédant trois 2-simplexes, neuf 1-simplexes et six 0-simplexes. À droite, le complexe simplicial transformé possédant trois 1-simplexes et trois 0-simplexes, mais préservant l'homologie du complexe simplicial de gauche.

Dans le cas d'un complexe simplicial dont la représentation en réseau biparti est valide, cette transformation revient à utiliser une projection plutôt qu'une autre. En effet, si nous avons projeté le réseau biparti sur les entités, chacune des facettes représente un site. Ainsi, si nous formons un nouveau complexe simplicial en connectant les facettes, c'est comme si nous connections les sites entre eux. Cela revient alors à prendre la projection du réseau biparti sur les sites.

En utilisant cette technique pour les données d'OTUs, nous obtenons une seule facette de taille 38 pour les données complètes et une seule facette de taille 34 pour les données filtrées. Dans les deux cas, l'homologie du complexe simplicial est triviale : tous les nombres de Betti sont nuls excepté $\beta_0 = 1$. L'utilisation de la transformation nous permet donc de traiter facilement les données d'OTUs. En ce qui concerne les données sur les oiseaux datant de 2016, la projection sur les espèces donne 143 facettes. Parmi celles-ci, la plus grande facette possède 19 noeuds. L'autre projection mène à 29 facettes et celle ayant la taille la plus élevée contient 148 noeuds. Pour le second jeu de données sur les oiseaux (2019), nous avons 876 facettes pour la projection sur les espèces et 76 facettes pour la projection sur les sites. Par contre, la taille des plus grandes facettes dans ces deux projections est 23 et 1 248. Pour les données sur les oiseaux, cette transformation n'est donc pas particulièrement avantageuse.

S'il advient que la transformation n'arrive pas à réduire la complexité du complexe simplicial, nous pouvons délibérément réduire sa complexité en utilisant le j-squelette. Toutefois, au lieu d'utiliser les fonctions de GUDHI, qui nécessitent d'abord d'avoir en mémoire le complexe simplicial original, nous pouvons construire le squelette en décomposant les facettes ayant une dimension supérieure à j en facettes ayant une dimension égale à j. Pour une facette de dimension $d > j$, il suffit de trouver toutes les $\binom{d+1}{j+1}$ combinaisons possibles.

Tel que vu précédemment, l'utilisation du j-squelette élimine de l'information sur l'homologie du complexe simplicial de base, car tous les groupes d'homologie d'indice supérieur à j sont maintenant vides. Afin de sélectionner le j-squelette le plus adéquat, c'est-à-dire celui qui n'élimine que les groupes d'homologie trivialement nuls, nous avons développé la méthode présentée à la section suivante.

2.2.3 Identification des nombres de Betti non triviaux

En utilisant GUDHI, on remarque que pour un complexe simplicial de dimensions d, les nombres de Betti sont calculés jusqu'à β_{d-1}. Par exemple, si le complexe simplicial est un triangle vide, comme dans l'exemple de la figure 1.9, nous aurons seulement $\beta_0 = 1$. Or, nous savons aussi que $\beta_1 = 1$, puisque les trois 1-simplexes forment un cycle qui n'est pas la frontière d'un 2-simplexe. De plus, tous les autres nombres de Betti, β_n avec $n > 1$, sont nuls puisque les groupes de chaînes d'indice $n > 1$ sont vides. Afin de forcer GUDHI à calculer β_d pour un complexe simplicial de dimension d, il suffit d'ajouter artificiellement une facette de dimensions $d+1$ qui est déconnectée du complexe simplicial original. GUDHI permet alors de calculer les nombres de Betti jusqu'à β_d. Toutefois, cette composante ajoute nécessairement une unité à β_0. Pour obtenir le β_0 du complexe simplicial original, il suffit alors de retirer 1 à β_0 pour annuler l'effet de la facette artificielle.

Malgré l'ajustement proposé ci-haut, le calcul des nombres de Betti jusqu'à β_d n'est pas toujours nécessaire. En effet, ayant nous-même accès à de l'information sur le complexe simplicial

via la liste des facettes, il est possible d'identifier quels sont les groupes d'homologie qui ne sont pas trivialement nuls. Cette étape est importante, puisque le calcul des nombres de Betti est une tâche computationnelle complexe. En effet, le temps de calcul de l'homologie augmente de manière exponentielle avec la dimension du complexe simplicial. Plus précisément, la complexité va comme $\mathcal{O}(m^3) > \mathcal{O}([2^\delta]^3)$, où m est le nombre de simplexes dans le complexe simplicial et δ est sa dimension [50].

Ainsi, nous pouvons donc accélérer la tâche si nous ne faisons que calculer ce qui est nécessaire. Par exemple, suite à la transformation proposée à la section 2.2.2, nous savons que les deux jeux de données d'OTUs sont réduits à une facette de taille 38 et une facette de taille 34. Avec notre ajustement, GUDHI calculera alors β_0 jusqu'à β_{37}, pour le premier jeu de données, et β_{33} pour le second. Or, seul β_0 n'est pas trivialement nul. En effet, les β_n avec $n \geq 1$ sont trivialement nuls, car les cycles présents dans la facette bornent toujours un simplexe d'ordre supérieur. Ainsi, nous pouvons sélectionner le 1-squelette de ces deux complexes simpliciaux et réduire le temps de calcul en ne mesurant que les nombres de Betti nécessaires. [1]

Afin d'identifier systématiquement quels sont les nombres de Betti qui ne sont pas trivialement nuls à partir d'une liste de facette, nous devons nous intéresser au nombre minimal de n-simplexes requis pour former un cycle dans Z_n. Pour obtenir ce nombre, nous pouvons utiliser la procédure suivante :

Algorithme 1 Méthode pour construire un cycle de taille minimale dans Z_n

Entrée Dimension n du groupe de cycles.
Sortie Liste des facettes de dimensions n qui forment le cycle le plus petit possible dans Z_n.

1: Créer un complexe simplicial formé d'un seul simplexe, ce dernier ayant une dimension $(n+1)$ avec $n \geq 1$.
2: Appliquer l'opérateur ∂_{n+1} sur le $(n+1)$-simplexe pour identifier les n-simplexes qui forment sa frontière, notée f.
3: Retirer le $(n+1)$-simplexe du complexe simplicial.
4: Ajouter les éléments de f dans le complexe simplicial.

Ainsi, comme le $(n+1)$-simplexe n'existe plus, f n'est plus une frontière, car C_{n+1} est maintenant vide. La chaîne f est alors un cycle. On peut imager cette méthode en disant que l'on retire l'interaction maximale d'un $(n+1)$-simplexe en ne retenant que sont enveloppe formée des n-simplexes. Par exemple, dans le cas d'un tétraèdre, il suffit de « vider » le coeur du solide et de ne garder que sa surface. Il est à noter que cette méthode ne fonctionne que pour $n \geq 1$, car si nous utilisons cette méthode pour $n = 0$, l'application de ∂_1 sur un 1-simplexe nous donne deux points. Or, pour ajouter une unité à β_0, nous n'avons besoin que d'un seul

1. Nous ne pourrions pas prendre le 0-squelette dans ce cas, car β_0 rapporterait le nombre de noeuds de la facette.

point.

Puisqu'un n-simplexe est la structure la plus simple que nous pouvons construire en n dimensions, les cycles obtenus en prenant la frontière de ces objets sont les plus courts. Nous pouvons d'ailleurs compter le nombre de n-simplexes requis pour former un tel cycle. Considérons d'abord un $(n+1)$-simplexe, noté σ, formé de $n+2$ points. Si nous voulons obtenir sa frontière, nous devons effectuer le calcul suivant :

$$\partial_{n+1}\sigma = \sum_{i=1}^{n+2}(-1)^i(p_1 \dots \hat{p_i} \dots p_{n+2}), \tag{2.3}$$

où p_i désigne les points qui constituent σ et $\hat{p_i}$ signifie d'omettre le point i.

Puisque chacun des termes de la somme comporte $n+1$ points, il s'agit d'une somme de n-simplexes. Le nombre de termes dans cette somme est alors donné par $\binom{n+2}{n+1}$, car parmi la séquence de $n+2$ points, nous voulons savoir combien de combinaisons il est possible de former avec $n+1$ points. La valeur de ce coefficient binomial est

$$\frac{(n+2)!}{(n+1)!1!} = n+2. \tag{2.4}$$

Cela signifie alors qu'un $(n+1)$-simplexe est enveloppé par $(n+2)$ simplexes de dimension n. Par exemple, pour que $\beta_2 = 1$ avec un minimum de simplexes, il faut vider un 3-simplexes (tétraèdre) de sorte qu'il ne reste que $\binom{4}{3} = 4$ simplexes de dimension 2. Autrement dit, nous pouvons écrire

$$\#\text{ minimal de simplexes de dimension } n \text{ pour former un cycle dans } Z_n = n+2. \tag{2.5}$$

Rappelons que cette formule ne fonctionne que pour $n \geq 1$. Si $n = 0$, nous n'avons besoin que d'un 0-simplexe pour former un cycle dans Z_0.

Sachant cela, nous pouvons maintenant inverser le problème. En effet, si la seule information que nous avons sur le complexe simplicial est le nombre de facettes N (en omettant l'information sur leur taille), nous pouvons borner l'indice des nombres de Betti en supposant que ces N facettes possèdent une dimension suffisante pour construire un trou de dimension maximale. Pour déterminer cette borne, nous devons d'abord répondre à la question suivante : « Si nous possédons N facettes, quelle dimension minimale doivent-elles avoir pour former un cycle dans Z_N ? »

Pour y répondre, il suffit d'isoler n dans l'équation (2.5) pour obtenir $n = N-2$. Cela signifie que si les N facettes sont de dimension $N-2$, alors le cycle de dimension maximale qu'il est possible de créer s'inscrit dans Z_{N-2}. Ainsi nous savons que nous ne pouvons pas créer de cycles dans des groupes ayant une dimension supérieure, faisant en sorte que $\beta_d = 0$ pour tous $d > N-2$. Il est à noter que nous spécifions que les facettes possèdent une dimension minimale de $N-2$, car même si les facettes étaient de dimension supérieure, il y aurait une

configuration de ces N facettes qui permet de créer un cycle dans Z_{N-2}. Par exemple, avec 3 facettes, le seul cycle que nous pouvons former prend la forme d'un triangle vide si nous avons seulement des 1-simplexes. Or, avec 3 facettes de dimension 2, nous serions aussi en mesure de créer ce cycle, tel qu'illustré à la figure 2.4, mais pas de cycle dans Z_2.

En réutilisant les données sur les OTUs et en appliquant cette méthode, nous obtenons que la dimension nécessaire de la facette doit être $n = -1$. Or, une telle dimension n'est pas possible pour un simplexe. Cette valeur nous indique alors qu'il n'est pas possible de former un cycle avec une seule facette. De même, si nous avions seulement deux facettes, nous aurions $n = 0$, ce qui est possible, mais inexact, car pour créer un « cycle » dans Z_0, nous avons seulement besoin d'un point. Ainsi, cette méthode pour borner l'indice des nombres de Betti n'est valide que pour des listes de plus de deux facettes. Autrement, le seul nombre de Betti non trivial est β_0. Cette méthode est donc très efficace lorsque le nombre de facettes est inférieur à la dimension du complexe simplicial. En effet, si nous avions 12 facettes de dimension 700, nous saurions qu'il est inutile de calculer β_d avec $d > 10$, puisque le cycle de dimension la plus grande que nous pouvons former avec 12 facettes de dimension adéquate se situe dans Z_{10}.

Lorsque nous considérons l'information sur la taille des facettes, nous pouvons raffiner la méthode précédente. En effet, on pourrait penser à un complexe simplicial qui comporte $N > 2$ facettes, mais où toutes les facettes sont des 0-simplexes. De ce fait, le seul groupe de chaîne possédant des éléments serait C_0 et le seul nombre de Betti non trivial serait β_0. Toutefois, la démarche précédente indique que le dernier nombre de Betti non trivial est β_{N-2}. Dans ce cas, la formule incite à calculer des nombres de Betti triviaux. La connaissance de la taille des facettes apparaît alors comme une information permettant de borner encore plus les nombres de Betti.

Plus précisément le nombre de facettes et l'histogramme du nombre de facettes de chaque taille permettent de borner de manière optimale les nombres de Betti. Rappelons qu'avec la formule (2.5) et un nombre N de facettes, nous pouvons obtenir la dimension minimale requise de ces facettes pour former le cycle de plus grande dimension possible, soit $N - 2$. Or, nous pouvons aussi obtenir la taille minimale S de ces facettes, puisque la taille S d'un simplexe de dimension n est reliée à la dimension par

$$S = n + 1, \tag{2.6}$$

de sorte qu la taille minimale des facettes pour former le cycle de plus grande dimension possible est

$$S = (N - 2) + 1 = N - 1. \tag{2.7}$$

L'équation (2.5) peut alors être transformée pour prendre en compte la taille des simplexes plutôt que leur dimension

$$\#\text{ minimal de simplexes de taille } S \text{ pour former un cycle dans } Z_{n=S-1} = S + 1. \tag{2.8}$$

De ce fait, si les N facettes sont de taille $S \geq N-1$, alors le dernier nombre de Betti non trivial est β_{N-2}.

Dans le cas où les facettes sont de taille $S < N-1$, il est possible de contraindre encore plus la borne, car nous savons que ces N facettes ne possèdent pas la taille minimale requise pour former un cycle dans Z_{N-2}. Or, il est possible que ces N facettes contribuent à la construction d'un cycle dans une dimension plus basse que $N-2$. Dès lors, si l'histogramme du nombre de facettes de chaque taille est connu, il est possible d'utiliser l'algorithme 2 pour connaître le dernier nombre de Betti non trivial.

Algorithme 2 Identification du groupe d'homologie non trivial d'indice maximal

Entrée Liste de facettes.
Sortie Indice du groupe d'homologie de dimension maximale pour lequel les nombres de Betti sont non triviaux.

1: Insérer les facettes de taille 2 ou plus dans une liste.
2: Compter le nombre de facettes dans cette liste.
3: Vérifier si ce nombre est plus grand ou égal à trois (dès qu'il existe au moins trois facettes de taille égale ou supérieure à 2, il est possible de construire des cycles dans Z_1). Si oui, poursuivre. Si non, le seul nombre de Betti non trivial est β_0.
4: Compter le nombre N_i de facettes de taille i, où $i = 2, \ldots, \max(S)$ et $\max(S)$ est la taille de la facette la plus grande.
5: À partir de $\max(S)$, itérer en ordre décroissant sur chaque i pour compter le nombre F de facettes de taille $S \geq i$ avec $F = \sum_{j=i}^{\max(S)} N_j$.
6: Si $F \geq i+1$ (conformément à (2.8)), cela signifie que la liste de facettes comporte assez de facettes de taille $S \geq i$ pour former un cycle dans Z_{i-1}. Ainsi le dernier nombre de Betti non trivial est $\beta_{i-1} = \beta_{F-2}$. Autrement, poursuivre l'itération en retirant une unité à i.

Une fois cet algorithme mis en place, il est possible de joindre toutes les méthodes présentées afin d'optimiser le temps de calcul et de réduire l'espace en mémoire pour mesurer les nombres de Betti non triviaux. La première étape est de vérifier si l'utilisation d'une projection plutôt qu'une autre permet de réduire la dimension du complexe simplicial. Par la suite, on utilise l'algorithme 2 pour borner les nombres de Betti. Une fois le seuil obtenu, la structure du complexe simplicial peut être réduite en prenant le j-squelette approprié qui permettra, au maximum, de calculer le dernier nombre de Betti non trivial.

La dimension appropriée du j-squelette est toujours une unité plus grande que la dimension du groupe d'homologie permettant de calculer le nombre de Betti désiré. En effet, si l'on sélectionne un squelette de la même dimension, des trous indésirables sont créés. Par exemple le nombre de Betti maximal d'un complexe simplicial composé de trois 2-simplexes est β_1. Toutefois, si l'on utilise le 1-squelette, tous les triangles « pleins » sont transformés en triangles vides, de sorte que $\beta_1 \geq 3$. Pourtant, les valeurs attendues pour β_1 pour le complexe simplicial d'origine sont plutôt 0 (si chaque pair de 2-simplexes ne partage pas un 0-simplexe) ou 1 (si

nous avons le motif de la figure 2.4). Une fois le squelette approprié construit, il suffit d'utiliser les routines de GUDHI pour mesurer les nombres de Betti.

En théorie, il s'agit des seules manipulations nécessaires. Toutefois, comme mentionné au début de cette sous-section, GUDHI mesure seulement les nombres de Betti jusqu'à β_{d-1}. Ainsi, pour un j-squelette, les nombres de Betti sont calculés jusqu'à β_{j-1}. Si notre algorithme indique toutefois que β_j est le dernier nombre de Betti non trivialement nul, il faut utiliser le $(j+1)$-squelette. Or, s'il advient qu'il n'existe pas de simplexe de dimension $(j+1)$ dans le complexe simplicial original, la dimension résultante du squelette sera égale à j. Les nombres de Betti sont alors mesurés jusqu'à β_{j-1}.

De ce fait, il est plus prudent d'ajouter systématiquement une facette de dimension $j+1$ dans le squelette résultant. Le rôle de cette facette est de permettre le calcul des nombres de Betti désirés dans le complexe simplicial original. Elle ne doit pas être connectée à une autre composante pour s'assurer qu'un nouveau trou ne soit pas créé accidentellement. Toutefois, puisqu'il s'agit d'une nouvelle composante dans le réseau, β_0 se voit nécessairement augmenté d'une unité. Il faut alors corriger le tir en retirant 1 à β_0 après l'avoir calculé.

Avec ces outils en main, nous pouvons donc calculer efficacement les nombres de Betti des complexes simpliciaux étudiés, ainsi que les complexes simpliciaux échantillonnés en utilisant le SCM.

2.3 Application aux jeux de données

Dans la présente section, nous présentons les résultats obtenus en utilisant le SCM comme modèle nul sur les données de présence/absence.

2.3.1 Jeux de données problématiques

En utilisant le SCM sur les données d'OTUs, il n'a pas été possible de générer un échantillon de complexes simpliciaux. En effet, il semble que la séquence des degrés généralisés ainsi que la séquence des tailles des facettes contraint l'algorithme MCMC dans un espace difficile à explorer, ou bien un espace où seul le complexe simplicial des données originales existe. Dans le premier cas, cela signifie que le temps nécessaire pour générer un échantillon est long. Dans le second cas, cela signifie que la distribution des nombres de Betti est simplement un delta de Dirac pour tout β_n. Nous comparons donc le complexe simplicial avec lui-même, ce qui ne constitue pas un modèle nul intéressant.

Considérant ces deux hypothèses, nous avons tenté de réduire la taille des facettes en retirant des OTUs en suivant certaines règles. Par exemple, nous avons filtré les données en retirant les OTUs qui, pour un site donné, se présentaient en proportion inférieure à un seuil donné.

De cette manière, certaines OTUs étaient éliminées de la matrice de présence/absence si elles ne se présentaient pas en proportion supérieure au seuil sur l'un des sites.

Cette opération diminuait la taille de certaines facettes et elle a permis la génération d'échantillons en utilisant le SCM. Toutefois, ces filtrations n'étant pas basées sur des suppositions biologiques, les résultats obtenus ne peuvent être analysés de manière rigoureuse. Une autre méthode pour réduire la taille des facettes aurait été de regrouper les OTUs en des classes plus grandes, mais, encore une fois, cela nécessite des connaissances en biologie.

Notons qu'il aurait pu être tentant de générer un échantillon de complexes simpliciaux à partir de la projection sur les sites, puisque cette projection possède la même homologie que la projection sur les espèces. Toutefois, l'ensemble des complexes simpliciaux qui préservent la séquence des degrés généralisés et la taille des facettes du complexe simplicial transformé n'est pas le même que celui de la projection sur les espèces. Ce faisant, l'interprétation des résultats aurait été différente.

2.3.2 Jeu de données traité

Le seul jeu de données rescapé des problèmes précédents correspond à celui sur les oiseaux datant de 2016. Sur ce dernier, nous avons généré un échantillon de 1000 observations et avons mesuré les nombres de Betti β_0 à β_{13}. La borne supérieure a été déterminée grâce à notre algorithme, sans quoi les nombres de Betti auraient été calculés jusqu'à β_{18}. Après avoir calculé ces nombres, nous avons toutefois constaté que les nombres de Betti β_n avec $n > 6$ sont nuls. Les distributions des nombres de Betti sont présentées à la figure 2.5.

Tel que nous pouvons l'observer sur ces figures, les nombres de Betti des données originales ne s'éloignent pas de manière significative de la distribution. En fait, ils se situent toujours dans la région qui comprend 95% des mesures sur l'ensemble aléatoire. De ce fait, la topologie du complexe simplicial construit à partir du jeu de données sur les oiseaux datant de 2016 ne semble pas posséder d'organisation particulière.

Le fait que nous ayons seulement été en mesure d'utiliser le SCM sur l'un des quatre jeux de données soulève la question suivante : « Est-il approprié d'utiliser le SCM dans le cas des données de présence/absence ? »

À la lumière des tests effectués dans le cadre de cette maîtrise, nous croyons que c'est l'hypothèse de base pour construire le complexe simplicial à partir des données qui s'avère problématique. En effet, elle stipule que k entités qui se présentent sur un même site forment un $(k-1)$-simplexe. Ainsi, par le simple fait que des entités se retrouvent dans le même espace, nous supposons qu'elles interagissent entre elles et que toutes les interactions sous-jacentes existent. Cela n'est pas impossible, mais nous n'avons pas de preuves plus sophistiquées de l'existence de ces interactions. Autrement dit, il s'agit de l'hypothèse la plus permissive que

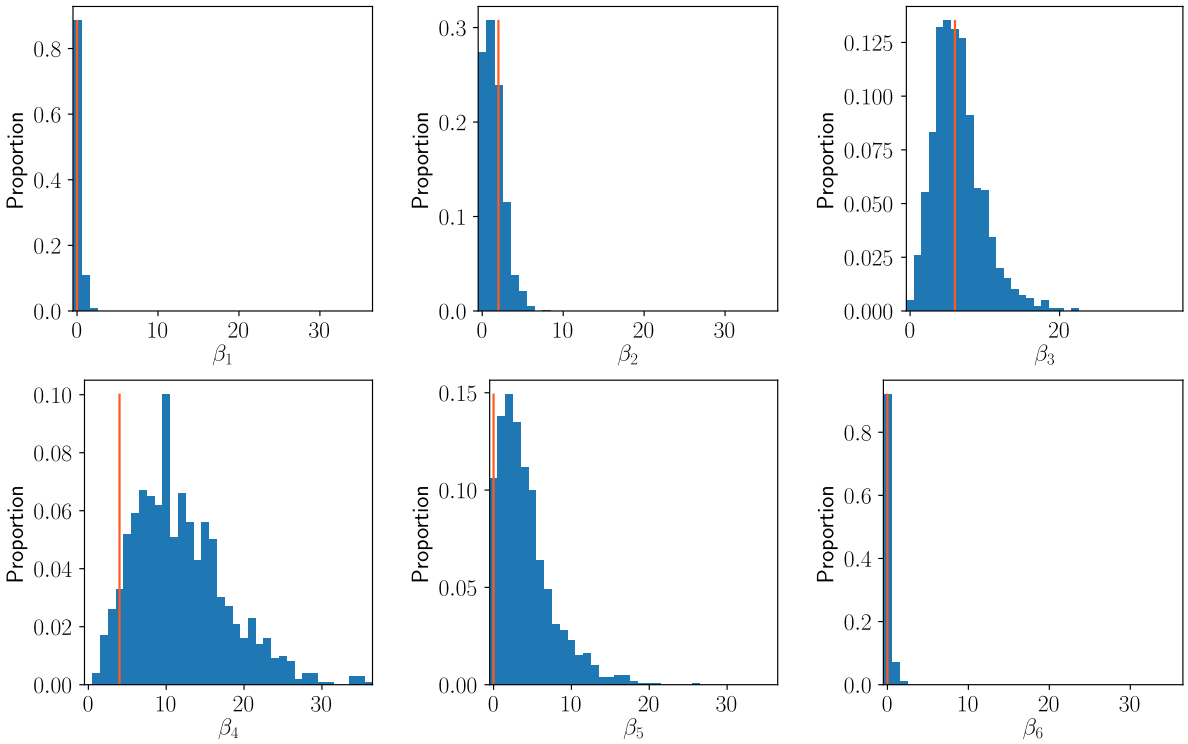

FIGURE 2.5 – Distribution des nombres de Betti, de β_1 à β_6. La ligne verticale orange représente la quantité de β_k dans le complexe simplicial créé à partir des données.

nous pouvons faire pour inférer des interactions d'ordre supérieur. La taille des facettes induite par cette hypothèse semble aussi être un facteur limitant, impliquant qu'il n'est pas possible de traiter tous les jeux de données avec cette approche.

Ainsi, nous croyons que le SCM devrait plutôt être utilisé lorsque l'hypothèse pour construire le complexe simplicial est plus plausible, par exemple dans le cas où des individus ont travaillé ensemble pour accomplir une tâche, comme les auteurs d'un article scientifique. Dans le chapitre suivant, nous avons donc élaboré un autre cadre théorique qui permet d'inférer les interactions une à une sur des bases statistiques. Ce cadre permet donc directement l'inférence d'interactions d'ordre supérieur et de complexes simpliciaux à partir de données de présence/absence. Cela résout donc le problème provenant de l'hypothèse trop souple posée dans le présent chapitre et ouvre la porte à l'utilisation du SCM sur des complexes simpliciaux construits avec circonspection.

Chapitre 3

Inférence d'interactions d'ordre supérieur : modèles log-linéaires

Les techniques d'inférence présentées dans ce chapitre s'attaquent au problème sous un angle complémentaire à celui du chapitre précédent. En effet, alors que nous inférions le complexe simplicial de la manière la plus permissive possible avec le SCM, ici nous cherchons à construire le complexe simplicial en vérifiant systématiquement s'il est plausible de croire qu'il existe une interaction entre les entités. Autrement dit, nous inférons d'abord les interactions par paire, pour ensuite inférer les interactions triples, puis quadruples, etc. La procédure est donc répétée jusqu'à ce que l'ordre désiré soit atteint. La structure résultante est alors un complexe simplicial dans lequel chaque simplexe représente une interaction significative.

Dans ce chapitre, les détails de ces techniques sont dévoilés dans l'ordre suivant. D'abord, à la section 3.1, les modèles log-linéaires sont présentés. Ces derniers sont essentiels pour généraliser l'inférence aux ordres supérieurs. Ensuite, à la section 3.2, nous expliquons comment les modèles log-linéaires interviennent dans l'élaboration des méthodes d'inférence proposées et expliquons les procédures pour construire un complexe simplicial (ou un hypergraphe) à partir de données de présence/absence. Finalement, à la section 3.3, nous caractérisons les performances des méthodes sur divers cas simples. Cette analyse nous permet d'ailleurs de faire des recommandations par rapport à la forme des données de présence/absence et d'évaluer les limites de ces méthodes.

3.1 Modèles log-linéaires

Les modèles log-linéaires sont utilisés pour analyser les interactions entre des variables catégoriques à partir d'une table de contingence [2, 35, 43, 56] telle la table 1.1 présentée à la section 1.2.5. Ils permettent de modéliser le logarithme des entrées dans la table par une série de facteurs. L'élaboration de méthodes utilisant ces modèles pour inférer les interactions entre les

variables remonte aux années 1960 [24]. Néanmoins, ces derniers, en plus d'être encore utilisés aujourd'hui, ont encore fait l'objet de diverses études dans la dernière décennie [51, 52, 53, 55].

De ces modèles est d'ailleurs née l'idée de *modèles graphiques*[1] dans lesquels les liens entre les variables catégoriques testées sont représentés dans un réseau [15, 34]. La technique d'inférence présentée ici s'inspire de ce concept pour la construction d'un complexe simplicial.

3.1.1 Tables en 2 dimensions

Comme décrit à la section 1.2.5, l'indépendance entre deux variables catégoriques signifie que nous pouvons écrire les probabilités conjointes comme le produit des probabilités marginales :

$$p_{ij} = p_{i+}p_{+j}, \tag{3.1}$$

pour tous les couples (i, j). Cela signifie également que nous pouvons écrire les entrées dans la table comme

$$m_{ij} = Np_{ij} = Np_{i+}p_{+j}, \tag{3.2}$$

où $N = m_{++}$ est le nombre total d'observations.

Plutôt que de formuler l'hypothèse nulle comme nous l'avons fait à la section 1.2.5, nous allons la formuler en termes de modèles log-linéaires, c'est-à-dire que nous allons récrire l'hypothèse d'indépendance en prenant le logarithme de l'équation (3.2) pour obtenir

$$\log(m_{ij}) = \log(N) + \log(p_{i+}) + \log(p_{+}j). \tag{3.3}$$

Pour alléger la notation, nous définissons

$$\log(N) = u, \tag{3.4}$$
$$\log(p_{i+}) = u_i^A, \tag{3.5}$$
$$\log(p_{+j}) = u_j^B, \tag{3.6}$$

où les indices supérieurs A et B font référence aux variables aléatoires associées à la probabilité marginale. Les indices i et j font référence à la i-ème et la j-ième entrée dans ces marges. Ainsi, l'expression réduite de l'équation (3.3) devient

$$\log(m_{ij}) = u + u_i^A + u_j^B. \tag{3.7}$$

Autrement dit, u_i^A représente l'effet de la catégorie i de la variable A sur le logarithme des entrées de la table et u_j^B représente l'effet de la catégorie j de la variable B. L'hypothèse nulle correspond alors à

$$H_0 : \log(m_{ij}) = u + u_i^A + u_j^B \text{ pour tous les couples } (i, j). \tag{3.8}$$

1. Il s'agit d'un mélange entre les modèles log-linéaires et les réseaux markoviens (*Markov fields*) [19, 34].

Supposons maintenant que les observations dans la table soient distribuées selon une loi multinomiale de la forme (1.12). À priori, nous ne connaissons pas les probabilités, p_{ij}, qui définissent cette loi. En effet, nous n'avons accès qu'à des observations, notées x_{ij}, dans chacune des cases d'une table de contingence qui correspond à une réalisation de cette loi multinomiale.

Nous ne pouvons alors qu'estimer les probabilités et les comptes espérés sous le modèle d'indépendance. À cette fin, nous pouvons utiliser la méthode du maximum de vraisemblance pour les paramètres d'une loi multinomiale. La vraisemblance d'une telle loi est égale à

$$\frac{N!}{\prod_{i,j} x_{ij}!} \prod_{i,j} \left(\frac{m_{ij}}{N}\right)^{x_{ij}}, \tag{3.9}$$

où x_{ij} est le nombre d'observations dans la case (i,j) et $m_{ij} = Np_{ij}$ est la valeur espérée dans la case (i,j) sachant que la probabilité d'obtenir un succès dans la case (i,j) est p_{ij}. Le logarithme de la vraisemblance s'écrit comme

$$\log\left(\frac{N!}{\prod_{i,j} x_{ij}!}\right) + \sum_{i,j} x_{ij} \log(m_{ij}) - N\log(N). \tag{3.10}$$

Pour maximiser l'expression, nous pouvons utiliser la méthode des multiplicateurs de Lagrange et l'ajout des contraintes de normalisation suivantes :

$$\sum_{i,j} m_{ij} = N, \tag{3.11}$$

$$\sum_{i,j} x_{ij} = N. \tag{3.12}$$

L'expression totale à maximiser est donc

$$\log\left(\frac{N!}{\prod_{i,j} x_{ij}!}\right) + \sum_{i,j} x_{ij} \log(m_{ij}) - N\log(N) + \lambda_1 \left(N - \sum_{ij} m_{ij}\right) + \lambda_2 \left(N - \sum_{ij} x_{ij}\right), \tag{3.13}$$

où λ_1 et λ_2 sont les multiplicateurs de Lagrange. Dans la deuxième somme de (3.13), nous observons le terme $\log(m_{ij})$ que nous pouvons remplacer par l'expression (3.7). L'optimisation du logarithme de la vraisemblance par rapport aux paramètres $u, u_i^A, u_j^B, \lambda_1$ et λ_2 permet alors d'obtenir les équations

$$\hat{m}_{++} = x_{++}, \tag{3.14}$$

$$\hat{m}_{i+} = x_{i+}, \tag{3.15}$$

$$\hat{m}_{+j} = x_{+j}, \tag{3.16}$$

où l'accent circonflexe indique qu'il s'agit de la valeur estimée pour les paramètres de la loi multinomiale. De plus, à partir du modèle log-linéaire (3.7), nous avons que

$$m_{++} = e^{u} \sum_{i} e^{u_i^A} \sum_{j} e^{u_j^B}, \tag{3.17}$$

$$m_{i+} = e^{u+u_i^A} \sum_{j} e^{u_j^B}, \tag{3.18}$$

$$m_{+j} = e^{u+u_j^B} \sum_{i} e^{u_i^A}, \tag{3.19}$$

de sorte que

$$m_{ij} = e^{u+u_i^A+u_j^B} = \frac{m_{i+}m_{+j}}{m_{++}}. \tag{3.20}$$

En combinant les équations (3.14) à (3.16) avec l'équation (3.20), nous obtenons l'expression suivante pour les estimateurs des valeurs espérées sous le modèle d'indépendance :

$$\hat{m}_{ij} = \frac{x_{i+}x_{+j}}{N}. \tag{3.21}$$

Avec l'équation (3.21), nous pouvons également construire la table des valeurs espérées sous le modèle d'indépendance et comparer cette table à la table des valeurs observées grâce à la statistique χ_0^2 de l'équation (1.27).[2] Avec cette statistique, nous pouvons effectuer le test d'indépendance de la section 1.2.5 en calculant sa valeur-p. Par contre, l'hypothèse nulle (3.8) étant associée à un modèle log-linéaire, nous disons donc plutôt qu'il s'agit d'un **test de modèles**. Si l'hypothèse nulle est acceptée, nous acceptons le modèle log-linéaire (3.8) pour décrire nos données.

Dans le cas d'un rejet, nous devons nous tourner vers l'hypothèse alternative qui considère une dépendance entre les deux variables. Nous pouvons d'ailleurs écrire un modèle log-linéaire qui inclut ce terme de dépendance dans la factorisation de la probabilité. Plus précisément, il s'agit du modèle

$$\log(m_{ij}) = u + u_i^A + u_j^B + u_{ij}^{AB}, \tag{3.22}$$

où u_{ij}^{AB} correspond à un terme de dépendance entre les deux variables pour la catégorie (i, j).

On parle ici de terme de dépendance, puisqu'il empêche d'écrire la probabilité conjointe sous la forme d'un produit de deux probabilités indépendantes. Ce modèle possède toutefois plus de paramètres libres qu'il n'y a de comptes à estimer. Par analogie, cela peut correspondre à un système d'équations sous-déterminé. Afin de fixer la solution, nous ajoutons les contraintes

2. Dans cette expression, les m_{ij} sont remplacés par les x_{ij}, tandis que les $\hat{m}_{ij}$ correspondent à ceux de l'équation (3.21).

suivantes sur le système [3] :

$$\sum_i u_i^A = \sum_j u_j^B = \sum_i u_{ij}^{AB} = \sum_j u_{ij}^{AB} = 0. \tag{3.23}$$

Ainsi, dans le modèle log-linéaire, le terme u correspond à un paramètre libre, alors que les contraintes des termes u_i^A et u_j^B impliquent qu'il y a $(I-1)$ et $(J-1)$ paramètres libres dans ces groupes respectifs et les deux dernières contraintes impliquent qu'il y a $(I-1)(J-1)$ termes u_{ij}^{AB} libres. Le nombre total de termes libres est donc $I \times J$, ce qui correspond au nombre total de cases élémentaires dans la table de contingence considérée.

Lorsqu'un modèle possède autant de paramètres libres que de cases élémentaires, celui-ci est qualifié de **modèle saturé**. Lorsque nous appliquons la méthode de maximisation de la vraisemblance en utilisant le modèle saturé (3.22) dans le logarithme de la vraisemblance (3.13), nous obtenons les estimateurs

$$\hat{m}_{ij} = x_{ij}. \tag{3.24}$$

Les estimateurs sont donc identiques aux valeurs observées.

De ce fait, la comparaison de la table observée à la table des valeurs espérées sous le modèle saturé donne toujours une statistique χ_0^2 nulle, faisant en sorte que le modèle saturé arrive toujours à expliquer les observations. Cela ne signifie toutefois pas que le modèle saturé est toujours le modèle le plus plausible. En effet, si le modèle d'indépendance arrive également à expliquer nos observations nous allons le favoriser par rapport au modèle saturé, puisqu'il s'agit d'un modèle plus simple [15].

L'un des intérêts des modèles log-linéaires réside dans cette comparaison entre un modèle non saturé et un modèle saturé. En effet, dans le cas d'une table en deux dimensions, si la conclusion du test statistique [4] est le rejet du modèle d'indépendance, cela signifie qu'il est plausible de croire que nous avons besoin d'un terme d'interaction entre les variables pour expliquer nos observations.

Dans le cas des données de présence/absence, les variables aléatoires considérées, disons A et B, représentent des entités s'étant présentées (ou non) sur les N sites échantillonnés. Chaque variable possède donc deux catégories, c'est-à-dire que leur support est $0, 1$, où 0 représente l'absence et 1 représente la présence. La table de contingence associée à cette situation est donc la suivante :

3. Il est à noter qu'il y a aussi des contraintes sur le modèle (3.7) qui correspondent aux deux premières sommes à l'équation (3.23). Nous n'allons pas plus loin sur ce sujet considérant que ces contraintes ne nous sont pas utiles pour obtenir les valeurs espérées [11], comme nous allons le voir dans la section 3.1.4. Notons également que le choix des contraintes est arbitraire et nous pouvons en utiliser d'autres. Cela modifie donc les valeurs des paramètres « u », mais, tel qu'indiqué, ces dernières n'interviennent pas dans le calcul des $\hat{m}_{ij}$ [11]. De plus, utiliser une contrainte plutôt qu'une autre ne change pas la valeur des comptes espérés [31, 46, 57].

4. Le nombre de degrés de liberté de la loi du χ^2 utilisée dans le test statistique correspond au nombre total de cases dans la table de contingence auquel on soustrait le nombre de paramètres libres du modèle log-linéaire [11, 15].

	$B=0$	$B=1$
$A=0$	x_{11}	x_{12}
$A=1$	x_{21}	x_{22}

.

Pour déterminer si les entités A et B se présentent de manière indépendante, la démarche précédente est alors tout indiquée, puisque si les sites ont été échantillonnés de manière indépendante et que chaque site est identiquement distribué, la table de contingence construite correspond à une réalisation d'une loi multinomiale. Ainsi, nous pouvons utiliser les données pour tester les hypothèses

$$\begin{aligned} H_0 &: \log(m_{ij}) = u + u_i^A + u_j^B \text{ pour tous les } i = 1, 2, ..., I \text{ et } j = 1, 2, ..., J; \\ H_1 &: \log(m_{ij}) = u + u_i^A + u_j^B + u_{ij}^{AB} \text{ pour au moins un couple } (i, j). \end{aligned} \tag{3.25}$$

Pour effectuer le test de modèles, nous devons calculer la valeur-p de la statistique χ_0^2 à partir d'une loi du χ^2 à 1 degré de liberté. En effet, comme les variables A et B possèdent deux catégories, le nombre de degrés de liberté est déterminé par $(2-1)\times(2-1) = 1$. Si l'hypothèse nulle est acceptée, cela signifie qu'il est plausible de croire que les entités se présentent sur les sites de manière indépendante, tandis que si elle est rejetée, il est plus plausible que les entités se présentent/s'absentent de manière dépendante.

Nous pouvons aussi représenter ces relations de dépendance (et d'indépendance) par un réseau. En effet, lorsque deux entités cooccurrent de manière dépendante sur les sites, cela signifie que nous pouvons tracer un lien entre elles. S'il n'existe pas de dépendance, ces dernières demeurent déconnectées dans le réseau, comme à la figure 3.1.

FIGURE 3.1 – Représentation réseau de H_0 et de H_1 pour deux entités. À gauche, l'hypothèse nulle est acceptée, tandis qu'à droite, cette dernière est rejetée, indiquant une dépendance plausible entre les entités.

3.1.2 Mesure d'association

À partir d'une table de contingence 2×2, nous pouvons aussi quantifier l'association entre les deux variables via le coefficient ϕ [18] dont l'équation est

$$\phi = \frac{x_{22}x_{11} - x_{21}x_{12}}{\sqrt{x_{1+}x_{2+}x_{+1}x_{+2}}}. \tag{3.26}$$

Cette mesure est bornée entre -1, qui signifie que les entrées non nulles se situent seulement sur l'antidiagonale de la table, et 1 qui signifie l'inverse. Dans le cas des données de présence/absence, une mesure de -1 signifie que si une espèce est présente sur un site, l'autre est

absente et vice-versa. Une mesure de 1 indique que les deux espèces sont présentes et absentes sur les mêmes sites.

Nous pouvons d'ailleurs relier ce coefficient à la statistique χ_0^2 en effectuant les manipulations suivantes. D'abord, nous avons

$$\begin{aligned}\chi_0^2 &= \sum_{i,j} \frac{(x_{ij} - \hat{m}_{ij})^2}{\hat{m_{ij}}} \\ &= \left(\sum_{i,j} \frac{x_{ij}^2}{\hat{m}_{ij}} \right) - N \\ &= N \left[\left(\sum_{i,j} \frac{p_{ij}^2}{p_{i+}p_{+j}} \right) - 1 \right]. \end{aligned} \tag{3.27}$$

Dans l'équation (3.27), nous avons utilisé le fait que $p_{ij} = x_{ij}/N$ et $\hat{m}_{ij} = p_{i+}p_{+j}$. De plus, nous avons que $p_{ij} \leq p_{i+}$ et $p_{ij} \leq p_{+j}$, où l'égalité est respectée lorsque les cases voisines à p_{ij} sont nulles. Si cette dernière condition est respectée et que nous sommes en présence d'une table 2×2, la somme à l'équation (3.27) donne 2.

En vertu de l'inégalité sur les p_{ij}, 2 est la valeur maximale de la somme de l'équation (3.27). Lorsque les probabilités sont toutes identiques dans la table, nous obtenons la valeur minimale de 1. Autrement dit, pour une table de contingence 2×2, nous avons les relations

$$0 \leq \chi_0^2 \leq N \quad \Rightarrow \quad 0 \leq \frac{\chi_0^2}{N} \leq 1. \tag{3.28}$$

En développant l'expression (3.27) pour la table 2×2, nous obtenons alors

$$\chi_0^2 = N \left[\frac{(p_{11}p_{22} - p_{12}p_{21})^2}{p_{1+}p_{+1}p_{2+}p_{+2}} \right]. \tag{3.29}$$

En réutilisant la relation $p_{ij} = x_{ij}/N$ dans l'équation (3.29) et en divisant le tout par N, nous obtenons alors ϕ^2. Selon l'équation (3.28), cette mesure est alors bornée entre 0 et 1. En prenant la racine de cette expression, nous retrouvons l'équation (3.26) qui est bornée entre -1 et 1. Le signe positif devant la racine est obtenu si le produit de la diagonale est supérieur au produit de l'antidiagonale, et l'inverse pour le signe négatif.

Ainsi, lorsque le coefficient ϕ s'approche d'une de ses valeurs extrêmes, cela signifie que la statistique χ_0^2 s'approche de sa valeur maximale, ce qui suggère également une valeur-p qui s'approche de zéro et un rejet de l'hypothèse nulle. Au contraire, lorsque le coefficient s'approche de zéro, la statistique s'approche également de zéro et la valeur-p s'approche de l'unité, faisant en sorte que l'on accepte l'hypothèse nulle. En mesurant le coefficient ϕ pour chacune des paires, nous pouvons facilement quantifier la relation de cooccurrence entre les espèces.

3.1.3 Tables en 3 dimensions et plus

Les modèles log-linéaires peuvent être généralisés à des tables de contingence à trois dimensions et plus. En effet, nous pourrions considérer une table formée à partir de trois variables A, B et C chacune séparée en I, J et K catégories. La table résultante correspond alors à un cube contenant $I \times J \times K$ cases distinctes. Les observations ainsi que les valeurs espérées possèdent alors trois indices. Nous allons respectivement les nommer x_{ijk} et $\hat{m}_{ijk}$, avec $i = 1, 2, \ldots, I$, $j = 1, 2, \ldots, J$ et $k = 1, 2, \ldots, K$. En présence d'une telle situation, le modèle log-linéaire saturé correspond à

$$\log(m_{ijk}) = u + u_i^A + u_j^B + u_k^C + u_{ij}^{AB} + u_{ik}^{AC} + u_{jk}^{BC} + u_{ijk}^{ABC}, \tag{3.30}$$

où les termes en u sont contraints par les équations

$$\sum_i u_i^A = \sum_j u_j^B = \sum_k u_k^C = 0, \tag{3.31}$$

$$\sum_i u_{ij}^{AB} = \sum_j u_{ij}^{AB} = 0, \tag{3.32}$$

$$\sum_i u_{ik}^{AC} = \sum_k u_{ik}^{AC} = 0, \tag{3.33}$$

$$\sum_j u_{jk}^{BC} = \sum_k u_{jk}^{BC} = 0, \tag{3.34}$$

$$\sum_i u_{ijk}^{ABC} = \sum_j u_{ijk}^{ABC} = \sum_k u_{ijk}^{ABC} = 0. \tag{3.35}$$

De cette manière, nous avons

$$1 + (I-1) + (J-1) + (K-1) + (I-1)(J-1) + (I-1)(K-1) + (J-1)(K-1) + (I-1)(J-1)(K-1) = IJK \tag{3.36}$$

paramètres libres, soit le nombre de cases dans la table de contingence. Le terme u_{ijk}^{ABC} est parfois appelé terme d'interaction du second ordre (*second-order interaction*) comme dans les références [10, 11].

À partir du modèle (3.30), il est possible de générer d'autres modèles en retirant certains termes en u. Nous allons toutefois nous limiter aux modèles dits « hiérarchiques ». Plus précisément, il s'agit des modèles où les termes associés à un groupe de variables ne peuvent exister que si tous les termes d'ordre inférieur formés à partir des combinaisons des variables de ce groupe s'y trouvent également. Autrement dit, pour que le terme u_{ij}^{AB} existe dans le modèle, il faut aussi retrouver u, u_i^A et u_j^B. De même, pour retrouver le terme u_{ijk}^{ABC}, il est nécessaire que $u_{ij}^{AB}, u_{ik}^{AC}, u_{jk}^{CB}, u_i^A, u_j^B, u_k^C$ et u soient présents. Le modèle $u + u_i^A + u_j^B + u_{ik}^{AC}$ n'est donc pas hiérarchique.

Ces modèles présentent trois avantages. Premièrement, leur interprétation est plus aisée que les modèles non hiérarchiques [11]. Par exemple,

$$\log(m_{ijk}) = u + u_i^A + u_j^B + u_k^C, \tag{3.37}$$

suggère que les trois variables sont complètement indépendantes [5], alors que

$$\log(m_{ijk}) = u + u_i^A + u_j^B + u_k^C + u_{ij}^{AB} + u_{ik}^{AC} \tag{3.38}$$

signifie l'indépendance conditionnelle entre les variables B et C sachant A, et le modèle

$$\log(m_{ijk}) = u + u_i^A + u_j^B + u_k^C + u_{ij}^{AB} + u_{ik}^{AC} + u_{jk}^{BC} \tag{3.39}$$

signifie que chaque paire de variables n'est pas affectée par la catégorie de la troisième variable [11]. Autrement dit, pour le modèle (3.39), il n'existe pas d'interaction d'ordre supérieur entre les trois variables. Pour la suite de ce mémoire, nous appelons **modèle sous-saturé** tout modèle hiérarchique dans lequel seul le terme d'interaction combinant toutes les variables est absent, comme le modèle (3.39).

Deuxièmement, pour ce type de modèle, nous pouvons utiliser un algorithme itératif, présenté à la section 3.1.4, pour obtenir les $\hat{m}_{ijk}$ sous un certain modèle log-linéaire.

Finalement, le principe de hiérarchie rappelle la construction des simplexes. En effet, un simplexe de dimensions $d \geq 1$ contient aussi tous les simplexes ayant une dimension inférieure à d. Cet élément est au coeur des méthodes présentées, comme nous allons le voir à la section 3.2.

3.1.4 Statistiques et configurations suffisantes

En présence d'une table à plus de deux dimensions, il n'est pas toujours possible d'utiliser la méthode de la maximisation de la vraisemblance pour obtenir une expression pour les $\hat{m}_{ijk}$ sous un modèle log-linéaire hiérarchique [10, 11, 15]. Par exemple, pour le modèle sous-saturé en trois dimensions, il est impossible d'obtenir une expression pour m_{ijk} en manipulant les sommes de type (3.14) à (3.16).

Il existe toutefois un algorithme itératif qui permet de converger vers ces valeurs, mais ce dernier requiert l'identification des statistiques et des configurations suffisantes [11]. Pour les obtenir, nous devons écrire le logarithme de la vraisemblance du processus qui génère les tables de contingence. En trois dimensions, nous avons la vraisemblance

$$\log\left(\frac{N!}{\prod_{i,j,k} x_{ijk}!}\right) + \sum_{i,j,k} x_{ijk} \log(m_{ijk}) - N\log(N). \tag{3.40}$$

5. Ce modèle est analogue au modèle (3.7) en deux dimensions.

Le seul terme qui contient m_{ijk} dans l'expression est $\sum_{i,j,k} x_{ijk} \log(m_{ijk})$, que nous identifions comme le noyau (*kernel* [11]) du logarithme de la vraisemblance. Si nous insérons la définition du modèle saturé (3.30) dans le noyau nous déduisons

$$\begin{aligned}\sum_{i,j,k} x_{ijk} \log(m_{ijk}) = Nu &+ \sum_i x_{i++} u_i^A + \sum_j x_{+j+} u_j^B + \sum_k x_{++k} u_k^C \\ &+ \sum_{ij} x_{ij+} u_{ij}^{AB} + \sum_{ik} x_{i+k} u_{ik}^{AC} \\ &+ \sum_{jk} x_{+jk} u_{jk}^{BC} + \sum_{ijk} x_{ijk} u_{ijk}^{ABC}. \end{aligned} \tag{3.41}$$

De cette manière, nous pouvons déjà identifier les statistiques suffisantes comme étant les termes en x dans chacune des sommes [10, 11].

Pour le modèle saturé, les $\hat{m}_{ijk}$ sont exactement équivalents aux observations. L'identification des statistiques suffisantes n'est donc pas nécessaire. Par contre, pour trouver les statistiques suffisantes d'un autre modèle hiérarchique, il faut d'abord retirer les sommes qui contiennent les termes en u qui ne font pas partie du modèle considéré.

Par exemple, si nous posons $u_{ijk}^{ABC} = 0$, nous obtenons le modèle sous-saturé et le noyau devient

$$\begin{aligned}\sum_{i,j,k} x_{ijk} \log(m_{ijk}) = Nu &+ \sum_i x_{i++} u_i^A + \sum_j x_{+j+} u_j^B + \sum_k x_{++k} u_k^C \\ &+ \sum_{ij} x_{ij+} u_{ij}^{AB} + \sum_{ik} x_{i+k} u_{ik}^{AC} + \sum_{jk} x_{+jk} u_{jk}^{BC}. \end{aligned} \tag{3.42}$$

Pour savoir quelles statistiques suffisantes nous devons utiliser dans l'algorithme, il faut transformer davantage le noyau. À cette fin, nous allons retravailler l'expression du modèle log-linéaire.

D'abord, il est nécessaire d'identifier les termes dont la position est la plus élevée dans la hiérarchie. Dans ce cas-ci, nous avons u_{ij}^{AB}, u_{ik}^{AC} u_{jk}^{BC}. Par la suite, nous regroupons les termes d'ordre inférieur avec les termes d'ordre supérieur, si les exposants du terme d'ordre inférieur sont tous en commun avec le terme d'ordre supérieur. Par exemple, nous pourrions former les groupes $u_i^A + u_{ij}^{AB}$, $u_j^C + u_{ik}^{AC}$ et $u_i^B + u_{jk}^{BC}$. Nous complétons ensuite chacun des groupes en ajoutant les termes manquants dans la hiérarchie. Pour que cette expression du modèle log-linéaire soit équivalente à la première, nous devons aussi soustraire les termes que nous ajoutons. Pour le premier groupe $(u_i^A + u_{ij}^{AB})$, nous avons alors $u_i^A + u_j^B - u_j^B + u_{ij}^{AB}$. En effectuant cette opération sur tous les groupes et en regroupant les termes négatifs, nous obtenons

$$\log(m_{ijk}) = u + (u_i^A + u_j^B + u_{ij}^{AB}) + (u_i^A + u_k^C + u_{ik}^{AC}) + (u_j^B + u_k^C + u_{jk}^{BC}) - (u_i^A + u_j^B + u_k^C), \tag{3.43}$$

où nous posons ensuite les variables

$$U_{AB} = (u_i^A + u_j^B + u_{ij}^{AB}), \tag{3.44}$$

$$U_{AC} = (u_i^A + u_k^C + u_{ik}^{AC}), \tag{3.45}$$

$$U_{BC} = (u_j^B + u_k^C + u_{jk}^{BC}). \tag{3.46}$$

Le noyau devient alors

$$\sum_{i,j,k} x_{ijk} \log(m_{ijk}) = Nu + \left[\sum_{ij} x_{ij+} U_{AB} + \sum_{ik} x_{i+k} U_{AC} + \sum_{jk} x_{+jk} U_{BC} \right] - \left[\sum_{i} x_{i++} u_i^A + \sum_{j} x_{+j+} u_j^B + \sum_{k} x_{++k} u_k^C \right]. \tag{3.47}$$

Le premier groupe de sommes est celui qui permet d'identifier l'ensemble des statistiques suffisantes, puisque le second est redondant en raison de la définition des termes en « U ». Nous rappelons que les statistiques suffisantes sont les termes en x (incluant $x_{+++} = N$). Or, avec la forme (3.47), nous avons regroupé les statistiques suffisantes en groupes, que nous appelons les **configurations suffisantes** [10, 11, 29].

Une configuration correspond à une somme des x_{ijk} sur l'un des indices. Il s'agit donc d'un regroupement d'entrées dans les marges de la table. Ces dernières sont notées $C_{\mathbf{y}}$, où $\mathbf{y}$ est un ensemble d'indices sur lesquels la somme n'a pas été faite. Par exemple $C_{12} = x_{ij+} = \sum_k x_{ijk}$. Dans l'expression (3.47), les configurations suffisantes sont C_{12}, C_{13} et C_{23}.

Une méthode systématique pour obtenir les configurations suffisantes ainsi qu'une méthode pour déterminer s'il est possible d'estimer directement les comptes espérés sont présentées au chapitre 3 de la référence [11]. Une fois ces configurations obtenues, nous pouvons utiliser la méthode itérative proportionnelle [6] (*iterative proportional fitting*) pour obtenir les $\hat{m}_{ijk}$ [11, 15, 21].

Pour le modèle sous-saturé en trois dimensions, l'algorithme itératif est le suivant. Nous posons d'abord les valeurs initiales $\hat{m}_{ijk}^{(0)} = 1$, où l'exposant (0) désigne l'état des valeurs estimées à l'itération 0. Par la suite il suffit de calculer les comptes espérés en utilisant chacune des configurations, disons dans l'ordre $C_{12} = x_{ij+}$, $C_{13} = x_{i+k}$ et $C_{23} = x_{+jk}$ dans les expressions suivantes :

$$\hat{m}_{ijk}^{(1)} = \hat{m}_{ijk}^{(0)} \frac{x_{ij+}}{\hat{m}_{ij+}^{(0)}}, \tag{3.48}$$

puis

$$\hat{m}_{ijk}^{(2)} = \hat{m}_{ijk}^{(1)} \frac{x_{i+k}}{\hat{m}_{i+k}^{(1)}}, \tag{3.49}$$

6. La méthode de Newton-Raphson pourrait aussi être utilisée [29].

et finalement

$$\hat{m}_{ijk}^{(3)} = \hat{m}_{ijk}^{(2)} \frac{x_{+jk}}{\hat{m}_{+jk}^{(2)}}. \tag{3.50}$$

Cela correspond donc à un cycle complet.

L'identification des configurations suffisantes permet d'écrire la fraction mise en évidence dans les expressions (3.48) à (3.50). En effet, si nous désirions calculer les comptes espérés sous un autre modèle, il suffirait de modifier ces fractions en mettant les configurations suffisantes au numérateur et les comptes espérés sommés sur les mêmes indices au dénominateur.

Après ce premier cycle, les calculs (3.48) à (3.50) sont alors répétés en prenant les dernières valeurs estimées comme point de départ. Le critère de convergence est choisi à notre guise et prend la forme $|\hat{m}_{ijk}^{(3)} - \hat{m}_{ijk}^{(3r-3)}| < \delta$, où r est un entier strictement positif et $\delta \ll 1$. Des preuves que l'algorithme converge vers les valeurs espérées sont données dans [11, 21] et la méthode générale pour construire l'algorithme se retrouve dans [11].

Malgré l'efficacité de cette méthode, il faut interpréter la table des valeurs estimées avec prudence. En effet, la convergence de l'algorithme n'est pas gage de validité du test [29, 54, 55]. Il arrive que l'algorithme converge vers une solution alors que celle-ci ne cadre pas tout à fait avec le modèle log-linéaire que nous testons. Heureusement, ces cas problématiques sont très faciles à identifier : la table des valeurs espérées comporte des entrées nulles [14, 29]. Cela peut se produire dans deux situations que nous pouvons distinguer. Le premier cas survient lorsqu'il existe au moins une entrée nulle dans au moins une configuration suffisante. Dans ce cas, certaines valeurs estimées dans les équations (3.48) à (3.50) deviennent nulles dès le premier cycle. Si, pour les prochains cycles, nous considérons que le terme $\frac{0}{0} = 0$, ces entrées nulles persisteront dans la table, autrement, l'algorithme renvoie des valeurs invalides. Si ce cas se présente, certains ouvrages suggèrent d'utiliser les modèles quasi log-linéaires, qui tiennent compte des entrées nulles dans la table [11, 22]. Cela ne signifie pas que le modèle que nous testons est invalide, mais qu'il existe un modèle plus approprié pour tenir compte de la structure des données.

Le second cas survient lorsqu'il existe des entrées nulles dans la table qui sont disposées de manière spécifique sans créer d'entrée nulle dans les configurations suffisantes. Dans ce cas, la table espérée converge vers la table des valeurs observées, ce qui mène à une statistique χ_0^2 nulle. Dans la littérature, ce cas problématique est associé à des estimateurs par maximum de vraisemblance « inexistants » [29]. [7] Il existe différentes techniques pour identifier ce cas problématique et le traiter si les données le permettent. Dans certains cas, on parle d'extraire les estimés par maximum de vraisemblance étendus [8] (*extended maximum likelihood estimates*) [24, 26, 28, 29, 54, 55].

7. Cette idée d'inexistence des valeurs estimées a aussi été critiquée par quelques auteurs [3].
8. Pour une initiation à ces concepts, voir [38] et [5].

Comme nous allons le voir à la prochaine section, si l'un de ces problèmes se présente durant l'inférence, nous disons que la conclusion pour le groupe d'entités étudié est indéterminée. Nous préférons cette approche, car mélanger plusieurs techniques d'inférence pourrait générer des ambiguïtés dans l'interprétation de la structure du complexe simplicial inféré. En ne conservant qu'une seule méthode, les résultats obtenus peuvent tous être interprétés de la même manière.

3.2 Construction du complexe simplicial

Comme mentionné précédemment, nous pouvons généraliser les modèles log-linéaires pour un nombre arbitraire de variables. Dans le contexte des données de présence/absence cela signifie que nous pouvons tenter de déterminer quelles sont les interactions significatives autant pour les paires que les triplets, les quadruplets, etc. À cette fin, il suffit d'appliquer l'une des méthodes suivantes.

3.2.1 Méthode asymptotique

Pour construire le complexe simplicial, nous aurons recours à des tests de modèles. Dans la méthode qui suit, nous considérons que nous avons assez d'observations dans la matrice de présence/absence pour que la distribution de la statistique χ_0^2 tende asymptotiquement vers la loi du χ^2 à un degré de liberté. C'est pourquoi nous disons qu'il s'agit de la **méthode asymptotique**. Les étapes pour appliquer cette méthode vont comme suit.

Dans un premier temps, à partir de la matrice de présence/absence, nous pouvons former une table de contingence pour toutes les paires d'éléments. Chacune des variables aléatoires peut donc prendre deux états, 0 dénotant l'absence et 1 dénotant la présence. Pour chacune des tables, nous pouvons effectuer le test d'hypothèses pour deux variables que nous rappelons ici :

$$\begin{aligned} H_0 &: \log(m_{ij}) = u + u_i^A + u_j^B \quad \text{pour tous les} \quad i = 1, 2 \text{ et } j = 1, 2 \\ H_1 &: \log(m_{ij}) = u + u_i^A + u_j^B + u_{ij}^{AB} \text{pour au moins un couple } (i, j). \end{aligned} \tag{3.51}$$

Pour tester l'hypothèse nulle, nous devons d'abord obtenir les comptes espérés en utilisant la méthode itérative présentée à la section 3.1.4 (adaptée pour deux variables) ou en utilisant l'expression analytique (3.21).

Ensuite, nous calculons la statistique χ_0^2 entre la table des valeurs espérées sous l'hypothèse nulle et la table observée. Dans le cas où chaque variable aléatoire dans la table de contingence possède deux catégories et que nous testons le modèle sous-saturé, le nombre de degrés de liberté de la loi est toujours 1. En effet, si nous avons d variables aléatoires, le nombre de degrés de liberté est

$$2^d - \sum_{i=0}^{d-1} \binom{d}{i} (2-1)^i = 2^d - (2^d - 1) = 1, \tag{3.52}$$

où 2^d représente le nombre de cases dans la table de contingence et la somme représente le nombre de paramètres dans le modèle log-linéaire sous-saturé. Sachant que la statistique est distribuée selon la loi du χ^2 à un degré de liberté, nous pouvons calculer la valeur-p associée à χ_0^2. Si le test statistique indique que l'hypothèse nulle est plausible, nous considérons alors que les états de présence de chacune des entités étudiées sont indépendants. À l'inverse, le rejet de l'hypothèse nulle implique une dépendance statistique que nous pouvons représenter par un lien dans un réseau, comme à la figure 3.1.

Une fois chaque paire testée, nous obtenons un réseau où les cooccurrences significatives par paire sont identifiées par les liens. Cependant, certaines des cooccurrences seront négatives, c'est-à-dire que leur coefficient ϕ est négatif, alors que d'autres seront positives. Il peut alors être intéressant de ne garder que les interactions positives (ou négatives) lors de la recherche d'interaction d'ordre supérieur afin que l'interprétation de la relation entre les espèces soit plus aisée.

En contrepartie, les éléments qui se retrouvent sans lien sont considérés comme indépendants de tous les autres éléments.[9] Cela ne correspond pas à une vérité absolue, mais nous indique que nous n'avons pas assez de preuves pour ne pas croire l'hypothèse nulle.

Par la suite, pour dénicher les interactions d'ordre supérieur, deux possibilités s'offrent à nous. La première, qui cadre le plus avec la construction d'un complexe simplicial, est d'identifier n-simplexes « vides »[10] et de regarder si nous pouvons les élever au rang de n-simplexes. Pour le reste de ce mémoire, nous allons référer à cette méthode comme étant la **méthode d'inférence par étape**.

Par exemple, après la première étape où nous avons inféré les 1-simplexes, nous devons identifier les triangles vides du réseau et tester ces triplets pour une dépendance d'ordre supérieur. Autrement dit, nous testons les triangles pour savoir s'ils restent vides (un cycle composé de trois 1-simplexes) ou si nous pouvons les élever au rang de 2-simplexes. Ce processus est représenté graphiquement à la figure 3.2. Nous disons que cette méthode cadre mieux dans la construction d'un complexe simplicial, car si un groupe de trois noeuds est élevé au rang de 2-simplexe, c'est d'abord parce que tous les 1-simplexes qui font partie du 2-simplexe sont présents dans la structure que nous avons inférée à l'étape précédente.

La seconde méthode consiste à tester systématiquement tous les triplets qu'il est possible de former avec les entités de la matrice de présence/absence. De cette manière, peu importe la structure qui unit ou non les noeuds d'un triplet à l'étape précédente, nous testons ce dernier pour l'élever au rang d'hyperlien. Par exemple, il peut arriver qu'un triplet qui ne contient qu'un lien (ou deux, ou pas du tout) entre toutes les paires de noeuds ne puisse être

9. Cela n'est vrai que lorsque le calcul de la table espérée n'est pas un cas problématique comme ceux présentés à la section 3.1.4. Autrement, le lien entre les deux espèces est ambigu et nous rejetons la paire du complexe simplicial.

10. C'est-à-dire que nous identifions les cycles qui pourraient être la frontière d'un n-simplexe.

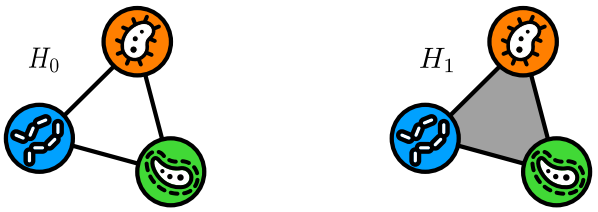

FIGURE 3.2 – Représentation réseau de H_0 et de H_1 pour trois entités. À gauche, le modèle sous-saturé est accepté. À droite, le modèle est rejeté et le triangle vide est élevé au rang de 2-simplexe.

expliqué que par le modèle saturé. Par contre, en itérant sur tous les triplets, nous obtenons un hypergraphe. Nous ne pouvons donc pas nécessairement utiliser les outils pour caractériser l'homologie de ces structures. Du moins, si nous le faisons, il faut interpréter les résultats sachant que l'hypergraphe ne respecte pas les règles de construction d'un complexe simplicial. Pour le reste de ce mémoire, nous allons référer à cette méthode en utilisant l'expression **méthode d'inférence systématique**.

Peu importe l'approche sélectionnée, le test des interactions d'ordre supérieur s'effectue de manière analogue à celui des interactions par paire. Plus précisément, après avoir construit des tables de contingence à partir de trois éléments (ou plus si nous testons des quadruplets, des quintuplets, etc.) de la matrice de présence/absence, nous devons poser une hypothèse nulle. À chaque ordre que nous testons, l'hypothèse nulle correspond au modèle sous-saturé tandis que l'hypothèse alternative correspond au modèle saturé.

Encore une fois, nous utilisons la méthode itérative pour obtenir les valeurs espérées et le test statistique est le même. Si l'hypothèse nulle est acceptée, il faut être prudent avec son interprétation. Il pourrait être tentant de conclure que le groupe d'éléments forme un cycle avec des simplexes dont la dimension est une unité inférieure à la dimension de la table, mais cela n'est pas toujours exact. En toute rigueur, pour savoir quelle forme prendra le motif, nous devons effectuer d'autres tests avec des modèles log-linéaires dans lesquels des termes en « u » sont retirés. Il s'agit alors d'un processus de sélection de modèle [2, 11, 15].

Idéalement, tous les modèles hiérarchiques seraient alors testés et le modèle le plus simple, c'est-à-dire celui qui comporte le moins de paramètres, qui arrive à expliquer nos observations est retenu. Dans les faits, le modèle sous-saturé est celui qui produira toujours la statistique χ_0^2 la plus basse, car c'est celui qui comporte le plus de paramètres avant le modèle saturé. Il modélise alors mieux les observations que les autres modèles non saturés. Cependant, un modèle plus simple peut quand même être plus plausible que le modèle sous-saturé. En effet, dans le cas des tables à 2^d dimensions, où d est le nombre de variables, le retrait d'un terme en u aura pour effet d'augmenter le nombre de degrés de liberté du test d'une unité. En augmentant le nombre de degrés de liberté, les lois du χ^2 deviennent plus conservatrices,

c'est-à-dire que pour une statistique donnée, la valeur-p croît avec le nombre de degrés de liberté. Ce comportement est représenté à la figure 3.3. Ainsi, même si des modèles plus simples peuvent produire des statistiques χ^2_0 plus grandes que celles du modèle sous-saturé, il est possible que ces modèles soient acceptés.

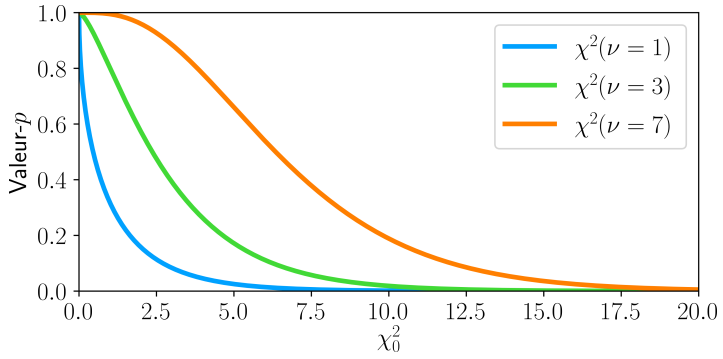

FIGURE 3.3 – Évolution de la valeur-p en fonction de la statistique χ^2_0 pour des lois du χ^2 à différents degrés de liberté (ν). Pour une statistique donnée, la valeur-p croît avec le nombre de degrés de liberté. La loi $\chi^2(\nu = 1)$ est donc la plus permissive, c'est-à-dire qu'elle permettra de rejeter l'hypothèse nulle pour la statistique χ^2_0 la plus basse.

Dans tous les cas, si le modèle sous-saturé est accepté, cela signifie que nous n'avons pas besoin d'un terme d'interaction d'ordre supérieur pour expliquer nos observations. Devant ce constat, nous pouvons soit tenter de trouver quel est le meilleur modèle pour le groupe d'entités étudié ou simplement continuer avec le prochain groupe sans se soucier du processus de sélection de modèle. Nous recommandons d'ailleurs cette seconde approche lorsque nous construisons le complexe simplicial en utilisant la méthode par étape. De cette manière, nous ne remettons donc pas en doute les simplexes inférés à l'ordre précédent. Même si cette seconde approche est approximative, elle demeure raisonnable dans l'optique où nous cherchons principalement les interactions d'ordre supérieur.

De plus, la recherche du meilleur modèle pour un groupe peut s'avérer laborieuse. En effet, en 3 dimensions, il y a 8 modèles hiérarchiques possibles, tandis qu'en 4 dimensions, il y en a 113 [15]. Cette recherche représente alors un goulot d'étranglement important pour la méthode, même s'il existe des algorithmes pour effectuer cette tâche [2, 11, 15].

Dans le cas où le modèle sous-saturé est rejeté, il faut également faire preuve de prudence. En réalité, il faudrait effectuer la même procédure que lorsque l'hypothèse nulle est acceptée pour sélectionner le meilleur modèle. En effet, même si le modèle sous-saturé est rejeté un modèle plus simple pourrait être accepté en raison du nombre de degrés de liberté du test. Par exemple, la table $2 \times 2 \times 2$ suivante :

	$B=0$	$B=1$
$A=0$	59	32
$A=1$	45	57

$C=0$

	$B=0$	$B=1$
$A=0$	54	64
$A=1$	49	40

$C=1$

produit la statistique 10.703 avec une valeur-p de 0.01344 pour le modèle $\log(m_{ijk}) = u + u_i^A + u_j^B + u_k^C + u_{jk}^{AC}$ (à 3 degrés de liberté) tandis qu'elle produit la statistique 8.974 avec une valeur-p de 0.00274 pour le modèle sous-saturé. Avec $\alpha = 0.01$, le modèle $\log(m_{ijk}) = u + u_i^A + u_j^B + u_k^C + u_{jk}^{AC}$ n'est donc pas rejeté.

Si tous les modèles sont rejetés lors du processus de sélection, nous pouvons alors conclure qu'une interaction d'ordre supérieure a été trouvée. Pour accélérer le processus, nous pouvons encore une fois éviter l'étape de la sélection du modèle et conclure qu'il existe une interaction d'ordre supérieur si le modèle sous-saturé est rejeté. Il s'agit donc d'une conclusion approximative en vertu du contre-exemple précédent, mais raisonnable considérant que le modèle sous-saturé, même s'il a été rejeté et qu'un modèle plus simple pourrait être accepté, est celui qui possède le plus grand pouvoir explicatif puisque la statistique χ_0^2 est la plus basse. Son rejet nous indique alors que, malgré son pouvoir explicatif, nous n'avons pas pu expliquer les valeurs observées. Si un modèle plus simple y arrive, c'est simplement parce que la loi du χ^2 associée au test est plus conservatrice, comme à la figure 3.3.

Les approches décrites pour construire un complexe simplicial ou un hypergraphe sont résumées dans les algorithmes 3 et 4 respectivement. Notons que pour le reste de ce mémoire, les étapes de sélection de modèle (qualifiées d'« optionnelles ») sont écartées des analyses. Toutefois, pour dégager le complexe simplicial (ou l'hypergraphe) le plus fidèle à l'analyse statistique, ces étapes devraient être réalisées.

Algorithme 3 Construction d'un complexe simplicial à partir de données de présence/absence

Entrées Matrice de présence/absence de taille $M \times N$;
Seuil α.

Sortie Liste des 1-simplexes jusqu'au d-simplexes qu'il est possible d'inférer.

1: Construire la table de contingence pour toutes les $\binom{M}{2}$ paires qu'il est possible de former dans la matrice.
2: Pour chaque table, effectuer le test du modèle d'indépendance et retenir les tables dont la valeur-p est inférieur à α.
3: Construire le complexe simplicial résultant.
4: Identifier les cycles composés de $k+1$ simplexes de taille k, où k est la dimension des tables de contingence de l'ordre analysé précédemment.
5: Pour chaque cycle trouvé, construire la table de contingence à partir des observations sur les entités appartenant au cycle.
6: Tester le modèle sous-saturé à $k+1$ variables et retenir les tables dont la valeur-p est inférieur à α.
7: (Optionnel) Effectuer une sélection de modèle pour chacune des tables avec le seuil α.
8: Construire le complexe simplicial résultant avec les nouveaux simplexes et répéter les étapes 4 à 8 avec k faisant référence à la taille des simplexes de l'étape 8.

Algorithme 4 Construction d'un hypergraphe à partir de données de présence/absence

Entrées Matrice de présence/absence de taille $M \times N$;
Seuil α.

Sortie Liste des hyperliens de taille 2 jusqu'à k qu'il est possible d'inférer.

1: En commençant par $k = 2$, où k est le nombre de variables considéré, construire les $\binom{M}{k}$ tables de contingence de dimension k qu'il est possible de former dans la matrice.
2: Tester le modèle sous-saturé à k variables et retenir les tables dont la valeur-p est inférieur à α.
3: (Optionnel) Effectuer une sélection de modèle pour chacune des tables avec le seuil α.
4: Répéter la procédure en augmentant k d'une unité.

Peu importe l'approche sélectionnée, elle est répétée jusqu'à ce que l'ordre désiré soit atteint. Par contre, il peut arriver que les données étudiées limitent d'elles-mêmes l'ordre que nous pouvons atteindre. En effet, à mesure que nous augmentons l'ordre étudié, les tables de contingence possèdent de plus en plus de cases à remplir. Si nous n'avons pas assez d'observations, certaines entrées dans la table peuvent être nulles et produire les problèmes décrits dans la section 3.1.4. Par contre, même si le nombre d'observations n'est pas adéquat pour faire un test asymptotique ou produire des tables qui ne sont pas problématiques, il reste parfois possible d'inférer les interactions convoitées avec la méthode présentée à la section suivante.

3.2.2 Méthodes exactes

Les méthodes présentées à la section précédente se basent sur un test d'hypothèses avec la statistique χ_0^2. Comme décrit à la section 1.2.5, il s'agit alors de tests asymptotiques. Cela signifie que nous avons besoin de plusieurs données pour que la statistique calculée, χ_0^2, soit issue d'une loi du χ^2 avec le nombre approprié de degrés de liberté. Certains ouvrages mentionnent qu'il est nécessaire d'avoir un « grand » nombre d'observations, sans toutefois spécifier une borne inférieure [1, 2, 11]. La seule recommandation trouvée provient de la référence [23] et propose qu'un « grand » nombre d'observations est atteint si la taille de l'échantillon est, au minimum, dix fois plus grands que le nombre de cases dans la table de contingence. Il est toutefois spécifié qu'il ne s'agit pas d'une règle absolue et qu'il arrive que le test asymptotique soit également utilisé pour des échantillons de taille plus petite. Néanmoins, cela signifie que si le nombre d'observations n'est pas suffisant, le calcul de la valeur-p est faussé et les conclusions du test d'hypothèses sont possiblement erronées.

De plus, lorsqu'un échantillon est de petite taille, certaines entrées dans les tables observées peuvent être nulles, menant parfois aux deux situations problématiques présentées dans la section 3.1.4. Dans ces situations, l'algorithme itératif converge vers une table identique à la table observée, faisant en sorte que $\chi_0^2 = 0$. Toutefois, il ne faut pas interpréter cette statistique comme étant une preuve que l'hypothèse nulle est plausible, mais bien qu'il n'est pas approprié de tester le modèle sur le groupe d'éléments étudié. Autrement dit, il ne faut pas considérer qu'il s'agit d'un vrai négatif dans le complexe simplicial. Cette subtilité est alors importante lorsque nous interprétons la structure résultante.

Malgré tout, si nous craignons que nous n'ayons pas suffisamment de données pour effectuer le test asymptotique, nous pouvons utiliser un test exact qui permettra d'extraire l'information convoitée dans une certaine mesure.

Méthode exacte à 1 degré de liberté

Ce test implique l'ajout de quelques étapes dans l'inférence afin de générer la distribution exacte de la statistique χ_0^2. En utilisant la distribution exacte, le calcul de la valeur-p devient, lui aussi, exact. Par contre, les étapes supplémentaires ralentissent le processus de construction du complexe simplicial. Il existe deux approches différentes pour générer la distribution exacte pour une table donnée. Le processus de la première méthode se trouve dans l'algorithme 5. De cette manière, nous générons la loi exacte du χ^2 à 1 degré de liberté. En effet, comme nous l'avons vu avec l'équation (3.52), l'estimation des valeurs espérées pour le modèle sous-saturé donne une statistique à un degré de liberté.

Examinons maintenant chacune des étapes afin de comprendre la logique derrière cette procédure. D'abord, les deux premières étapes sont identiques au test asymptotique utilisé à la section 3.2. Elles sont essentielles pour déterminer χ_0^2 entre la table de l'étape 1 et celle de

Algorithme 5 Génération de la distribution exacte à 1 degré de liberté

Entrées Matrice de présence/absence ;
Nombre de statistiques, n, désiré pour former la distribution ;
Seuil α.

Sortie Distribution exacte à 1 degré de liberté comportant n statistiques.

1: Construire la table de contingence pour un groupe d'entités.
2: Estimer sa table des valeurs espérées pour le modèle sous-saturé.
3: Générer un échantillon de n tables à partir d'une loi multinomiale ayant les mêmes valeurs espérées que celles de la table de l'étape 2.
4: Pour chaque observation dans l'échantillon, trouver la table des valeurs espérées pour le modèle sous-saturé.
5: Calculer la statistique χ_0^2 pour chaque table de l'échantillon.
6: Construire l'histogramme des statistiques obtenues précédemment.

l'étape 2. Ensuite, pour générer la distribution exacte de la statistique, nous avons besoin d'échantillonner une loi multinomiale ayant un ensemble de paramètres $\mathbf{\Theta}$. En théorie, avec cette procédure, nous obtiendrions une série de statistiques à un degré de liberté peu importe les paramètres choisis. Or, la spécification des paramètres ajoute une hypothèse supplémentaire dans la procédure et ce ne sont pas toutes les hypothèses qui sont sensées.

En effet, en choisissant des paramètres, nous émettons l'hypothèse que le processus qui a généré nos données à l'étape 1 correspond à la loi multinomiale spécifiée. Ainsi, en utilisant les valeurs espérées de la table de l'étape 2 pour spécifier les paramètres, nous faisons l'hypothèse que le modèle log-linéaire sous-saturé est à l'origine de nos observations. Nous restons donc cohérents avec l'hypothèse nulle que nous désirons tester.

À l'étape 3, l'échantillon est aussi généré par la multinomiale qui correspond au modèle log-linéaire sous-saturé. Toutefois, puisqu'il s'agit d'un processus aléatoire, certaines tables s'éloigneront des valeurs espérées de cette distribution, tout comme la table de l'étape 1 s'éloigne de celle à l'étape 2. Par la suite, à l'étape 4, nous devons trouver les tables espérées de chacune des tables de l'échantillon en utilisant la même forme de modèle log-linéaire. C'est en utilisant cette logique que les statistiques calculées à l'étape 5 possèdent seulement 1 degré de liberté. Avec ce calcul, nous vérifions à quel point des tables générées par un processus en accord avec le modèle log-linéaire sous-saturé s'éloignent de leur table de valeurs espérées, qui ont été obtenues en utilisant un modèle sous-saturé.

Finalement, avec les étapes 5 et 6, nous obtenons la distribution de la statistique. En raison de la variabilité imposée par le processus aléatoire, la série de statistiques sera distribuée d'une certaine manière ; c'est la distribution exacte à 1 degré de liberté. La valeur-p associée à la statistique χ_0^2 calculée entre la table de l'étape 1 et de l'étape 2 peut alors être obtenue. Si nous désirons également faire la sélection du modèle le plus adéquat, il suffit d'utiliser les autres modèles log-linéaire aux étapes 2 et quatre.

Méthode exacte à $2^d - 1$ degrés de liberté

La seconde méthode est similaire, mais utilise une hypothèse différente pour calculer la statistique. La méthode complète se trouve dans l'algorithme 6.

Algorithme 6 Génération de la distribution exacte à $2^d - 1$ degrés de liberté

Entrées Matrice de présence/absence ;
Nombre de statistiques, n, désiré pour former la distribution ;
Seuil α.

Sortie Distribution exacte à $2^d - 1$ degrés de liberté comportant n statistiques.

1: Construire la table de contingence pour un groupe d'entités.
2: Estimer sa table des valeurs espérées pour le modèle sous-saturé.
3: Générer un échantillon de n tables à partir d'une loi multinomiale ayant les mêmes valeurs espérées que celles de la table de l'étape 2.
4: Calculer χ_0^2 entre les tables de l'échantillon et la table des valeurs espérées de l'étape 2.
5: Construire l'histogramme des statistiques obtenues précédemment.

Le résultat de cette méthode est une loi exacte du χ^2 à $2^d - 1$ degrés de liberté, où d est la dimension de la table de contingence de l'étape 1.

Les trois premières étapes de cette méthode exacte sont identiques à celles de la première méthode exacte. Par contre, à l'étape 4, nous calculons une statistique qui n'a pas la même signification que celle de l'étape 5 de la méthode précédente. En effet, ici nous calculons la distance entre les tables générées et la table espérée de la loi multinomiale (celle de l'étape 2). Ce test correspond alors au test d'ajustement présenté à la section 1.2.5, d'où un nombre de degrés de liberté égal à $2^d - 1$.

En principe, lorsque nous estimons les paramètres de la loi lors d'un test d'ajustement, nous perdons des degrés de liberté [1]. Il y a donc une certaine ambiguïté dans le nombre de degrés de liberté de cette méthode, car nous avons estimé les paramètres de la loi multinomiale en utilisant la méthode itérative et un modèle log-linéaire. Toutefois, nous considérons que nous avons spécifié, et non pas estimé les paramètres directement, car il ne s'agit pas des paramètres les plus probables pour cette loi. En effet, si nous avions simplement utilisé la méthode du maximum de vraisemblance (sans faire intervenir les modèles log-linéaires) pour trouver les paramètres de la loi, nous aurions obtenu $p_{ij} = \frac{x_{ij}}{N}$. Ainsi, même si nous avons utilisé nos données pour estimer des paramètres, nous considérons que le fait d'avoir spécifié une loi multinomiale qui diffère de la loi la plus probable conserve le nombre de degrés de liberté du test à $2^d - 1$.

Les deux méthodes exactes présentées explorent donc des espaces différents et le rejet ou la conservation de l'hypothèse nulle n'a pas la même signification dans les deux cas. Avec la première méthode, le test statistique est le même que lorsque nous utilisions la distribution

asymptotique dans la section 3.2. Son interprétation est donc la même qu'auparavant. Plus précisément, la statistique χ_0^2 est ici une mesure de distance entre les observations et les valeurs espérées sous un modèle log-linéaire donné. Ainsi, si le processus qui a généré les observations respecte le modèle log-linéaire, la distribution exacte nous permet de calculer à quel point il est fréquent d'obtenir une certaine distance entre des tables observées et leurs valeurs espérées sous un certain modèle log-linéaire. Le test d'hypothèses est alors un test de modèles.

Dans le second cas, la statistique correspond à une mesure de distance entre les observations et les valeurs espérées d'une loi multinomiale. Ainsi, au lieu de générer des tables espérées pour chaque table de l'échantillon, nous mesurons la distance entre ces tables et la table espérée de la loi multinomiale. L'histogramme nous permet ensuite de calculer à quel point il est fréquent d'obtenir une table qui se situe à une certaine distance des valeurs espérées de la loi multinomiale. Nous avons donc les hypothèses

$$H_0 : \text{Les observations sont générées par } mult(\boldsymbol{\Theta}),$$
$$H_1 : \text{Les observations sont générées par une autre loi multinomiale..}$$

Si nous acceptons l'hypothèse nulle, cela signifie qu'il est plausible que nos observations soient générées par la loi multinomiale spécifiée et, par extension, que le modèle log-linéaire utilisé pour trouver les paramètres de la loi multinomiale est plausible. Si nous rejetons l'hypothèse nulle, nous concluons qu'il existe une interaction d'ordre supérieur entre les entités testées. De plus, comme le modèle sous-saturé produit la plus petite statistique en comparaison avec les autres modèles hiérarchiques non saturés et que le nombre de degrés de liberté du test est le même peu importe le modèle log-linéaire testé, nous n'avons pas besoin de tester les autres modèles si H_0 est rejetée. [11]

Toutefois, nous devons apporter la nuance suivante. Le rejet de l'hypothèse nulle signifie seulement qu'il existe des paramètres plus adéquats pour expliquer nos observations avec la loi spécifiée. Or, entre le modèle sous-saturé et le modèle saturé, il existe une infinité de paramètres que nous pouvons tester. Ainsi, lorsque nous concluons qu'il existe une interaction d'ordre supérieur entre les entités testées, il s'agit d'une approximation.

Visiblement, ces deux méthodes présentent quelques avantages et inconvénients. L'avantage de la première méthode est que le test statistique utilisé correspond à un test de modèles. Il s'agit simplement de la version « exacte » du test asymptotique effectué dans la section 3.2. Elle est donc à favoriser en ce sens.

Toutefois, dans un contexte où peu de données sont disponibles, certaines complications peuvent survenir. En effet, aux étapes 2 et 4, il se peut que l'algorithme converge vers une solution problématique comme celles présentées à la section 3.1.4. Dans les deux cas, il faut arrêter le test et on ne peut rien conclure sur le groupe d'entités examiné. Lorsque le problème

11. Cela n'est vrai que si nous acceptons notre argument par rapport aux degrés de liberté de ce test.

survient à l'étape 2, il faut arrêter la procédure pour les mêmes raisons que celles présentées à la section 3.1.4. À l'étape 4, le fait qu'une table espérée converge vers une solution problématique rend la démarche invalide, car cela signifie que nous devons rejeter cette table de l'échantillon. Or, ce rejet influence la distribution exacte que nous tentons de générer.

En comparaison, la seconde méthode présente alors trois avantages. Premièrement, elle est plus rapide que la première méthode, puisqu'elle comporte une étape en moins. Deuxièmement, lorsque la table des valeurs espérées existe à l'étape 2, les étapes subséquentes ne sont pas problématiques, car nous ne calculons pas les tables espérées pour les tables de l'échantillon. Troisièmement, le second test est, en principe, plus conservateur que le premier, car la distribution exacte possède plus de degrés de liberté que celle de la première méthode. Cela augmente donc le nombre de faux négatifs, mais réduit aussi le nombre de faux positifs. [12] Nous pouvons alors être plus confiants lorsque des interactions d'ordre supérieur sont inférées. Le désavantage de cette approche est donc que nous ne pouvons pas directement l'interpréter comme un test de modèles.

Peu importe l'approche exacte sélectionnée, il est important de garder en tête que le test exact ne peut pas non plus tout récupérer et que ses conclusions sont à interpréter avec précaution. Par exemple, si nous avons seulement 4 observations dans une table, il y a lieu de penser que la table n'est pas représentative du processus aléatoire sous-jacent et que notre conclusion n'est pas robuste. La robustesse d'une conclusion et les performances de l'inférence en utilisant ces différentes méthodes sont d'ailleurs étudiées dans la section qui suit.

12. Nous obtenons un **faux positif** si la méthode a inféré une interaction alors qu'il n'y en a pas (c'est-à-dire que nous avons accepté l'hypothèse alternative alors qu'elle est fausse ou, inversement, que nous avons rejeté l'hypothèse nulle alors qu'elle est vraie). De même, un **faux négatif** survient si nous n'avons pas inféré une interaction alors qu'il y en a une en réalité (nous avons accepté l'hypothèse nulle alors qu'elle est fausse).

3.3 Mesure de performance de l'inférence statistique

Puisque l'inférence est de nature statistique, cette dernière est sensible aux erreurs dans la prise de données. En effet, la conclusion d'un test statistique peut différer de la réalité si l'échantillon traité n'est pas assez représentatif de cette dernière. Afin de caractériser les performances de l'inférence sur de telles données, nous allons procéder de la manière suivante.

D'abord, nous allons concevoir une table de contingence de dimension d avec N observations dans laquelle nous choisissons les entrées. Nous appelons cette table la **table modèle**. Ensuite, nous effectuons le test du modèle sous-saturé en d dimensions sur cette dernière et la conclusion du test est alors considérée comme absolue. Par la suite, nous allons perturber la table modèle en ajoutant et en retirant des observations dans les cases tout en conservant le nombre total d'observations à N. Nous mesurons alors la distance entre la table modèle et la table perturbée en utilisant la distance L_1 dont l'expression est

$$L_1 = \sum_{i,j} |x_{ij} - \hat{x}_{ij}|, \tag{3.53}$$

où x_{ij} représente l'entrée (i, j) de la table perturbée et $\hat{x}_{ij}$ représente l'entrée (i, j) de la table modèle. Puisque le nombre d'observations est contraint à sommer à un entier N, l'ajout d'une unité dans l'une des cases oblige le retrait d'une unité dans une autre (et vice-versa). Cela a pour effet d'augmenter la distance L_1 de 2 pour chaque observation « mal placée » par rapport à la table modèle.

Afin de mesurer les performances de l'inférence statistique face au bruit, nous générons alors toutes les tables perturbées pour une distance L_1 donnée. Le test du modèle sous-saturé est effectué sur toutes ces tables perturbées et le taux de succès, c'est-à-dire le rapport entre nombre de fois où la conclusion est identique à celle de la table modèle et le nombre total de tables perturbées, est mesuré.

À partir d'une même table modèle, nous pouvons finalement mesurer le taux de succès pour diverses valeurs de L_1. Dans les sections qui suivent, à moins d'avis contraire, le test d'hypothèse a été effectué à l'aide de la méthode asymptotique pour des valeurs de $N \geq 100$ et $\alpha = 0.01$. Ces choix permettent d'explorer des valeurs de L_1 dans un temps raisonnable.

3.3.1 Algorithme de génération des tables perturbées

La génération de tables perturbées est un problème combinatoire. Par exemple, pour la table de contingence où tous les états sont équiprobables et où nous avons 100 observations, l'ensemble des tables de norme $L_1 = 2$ qui conservent $N = 100$ correspond à toutes les $\frac{4!}{1!1!2!} = 12$ permutations distinctes de la table suivante :

Dans ce cas-ci, il existe deux classes de tables, soit celles où 24 et 26 se retrouvent sur la diagonale (ou l'antidiagonale) et celles où 24 et 26 se retrouvent dans la même rangée (ou

	$B=0$	$B=1$
$A=0$	24	26
$A=1$	25	25

.

colonne). La valeur-p associée aux quatre tables de la première classe est 0.997, tandis que celle associée aux huit de la seconde classe est 0.841. De ce fait, pour toutes les tables perturbées de distance $L_1 = 2$ par rapport à la table modèle, le taux de succès est de 100%, puisque le test d'indépendance sur les 12 tables perturbées donne la même conclusion que la table modèle. En trouvant toutes les tables possibles pour une valeur de L_1 donnée et en calculant le taux de succès, nous pouvons tracer des figures telle la figure 3.5 qui indique à partir de quelle distance par rapport à la table modèle il n'est plus possible d'obtenir un taux de succès de 100%.

Afin de générer toutes les tables perturbées qui possèdent une distance $L_1 = 2k$ par rapport à la table modèle, où k est un entier positif, nous procédons comme suit. D'abord, nous trouvons toutes les partitions de l'entier k de longueur 1 jusqu'à 3 inclusivement. Les partitions d'un entier correspondent aux manières d'écrire cet entier par une somme d'autres entiers positifs, sans considérer les permutations possibles. [13] Par exemple, le chiffre 4 possède cinq partitions qui sont

$$\begin{aligned} 4 &= 4, \\ &= 3 + 1, \\ &= 2 + 2, \\ &= 2 + 1 + 1, \\ &= 1 + 1 + 1 + 1. \end{aligned}$$

Ici, il n'y a que quatre partitions dont la longueur est comprise entre 1 et 3 inclusivement. Une fois ces dernières identifiées, nous les transformons en séquences de nombres. Par exemple $3+1$ deviendrait la séquence $[3, 1]$. Nous générons également une copie de toutes ces séquences et multiplions chacune des entrées par -1. Ainsi, en concaténant une séquence positive avec une séquence négative, nous savons que la somme des éléments de la séquence résultante sera nulle.

Pour trouver toutes les perturbations, il faut alors former tous les couples qui comportent une séquence positive ainsi qu'une séquence négative. Cependant, nous retenons seulement les séquences concaténées qui contiennent entre 2 et 4 éléments inclusivement. Si la séquence comporte moins de 4 éléments, nous lui ajoutons des entrées nulles jusqu'à ce qu'il y ait 4 éléments au total. L'idée derrière cela est de pouvoir transformer ces séquences en matrices

13. Si nous considérions aussi les permutations, nous aurions plutôt les différentes **compositions** de l'entier k.

2×2 que nous allons ajouter à la table modèle. [14]

Puisque la somme sur les éléments de la séquence est nulle, nous savons que la perturbation, une fois ajoutée à la table modèle, conservera le nombre d'observations de celle-ci. Finalement, pour trouver toutes les tables perturbées de distance $L_1 = 2k$, il suffit de générer toutes les permutations distinctes de ces séquences, de les transformer en matrice 2×2 et de les ajouter à des copies de la table modèle. Un résumé de cette procédure se trouve dans le bloc 7.

Le taux de succès pour une distance $L_1 = 2k$ peut alors être mesuré comme décrit précédemment. Il est toutefois à noter que même si la somme des séquences concaténées est nulle, cela ne garantit pas que la table perturbée sera valide. Par exemple, si la table modèle est

	$B = 0$	$B = 1$
$A = 0$	20	40
$A = 1$	40	20

,

et que nous désirons trouver les tables de norme $L_1 = 50$, la séquence $[-25, 25, 0, 0]$ créerait, la table

	$B = 0$	$B = 1$
$A = 0$	-5	65
$A = 1$	40	20

.

Or, une table de contingence ne doit posséder que des entrées égales ou supérieures à 0. [15] Alors même si la séquence $[-25, 25, 0, 0]$ somme à zéro, la table résultante doit être rejetée et ne comptera pas pour le calcul du taux de succès. Cependant, les permutations distinctes de $[-25, 25, 0, 0]$, telles $[0, 25, -25, 0]$ et $[0, -25, 0, 25]$, produisent des tables valides sur lesquelles nous pouvons tester le modèle d'indépendance.

La valeur de L_1 qui permet de créer des perturbations capables d'annuler la plus petite entrée non nulle dans la table de contingence est notée $L_1^{\max}$. Idéalement, l'analyse des perturbations devrait être limitée entre $L_1 = 0$ et $L_1^{\max}$, puisque les entrées nulles dans une table de contingence peuvent causer des problèmes lors du test d'hypothèses.

Aussi, d'un point de vue expérimental, si le bruit possède une amplitude égale à nos mesures, nous ne pouvons pas faire la distinction entre nos observations et du bruit. Il n'est donc pas rigoureux d'analyser de telles données. De plus, si nous explorons des perturbations dont la distance L_1 est supérieure à $L_1^{\max}$, la diminution draconienne du nombre de tables valides fait augmenter le taux de succès par rapport à la distance L_1 précédente, tel qu'illustré à la figure

14. C'est également pour cette raison qu'à la première étape, nous ne conservons que les partitions de taille égale ou inférieure à 3.

15. Idéalement, elles seraient supérieures à zéro pour éviter certains problèmes tels que ceux présentés à la section 3.1.4.

3.4. Pourtant, l'inférence ne devient pas réellement plus performante et ce comportement est un bon indicateur que l'analyse des perturbations doit être arrêtée.

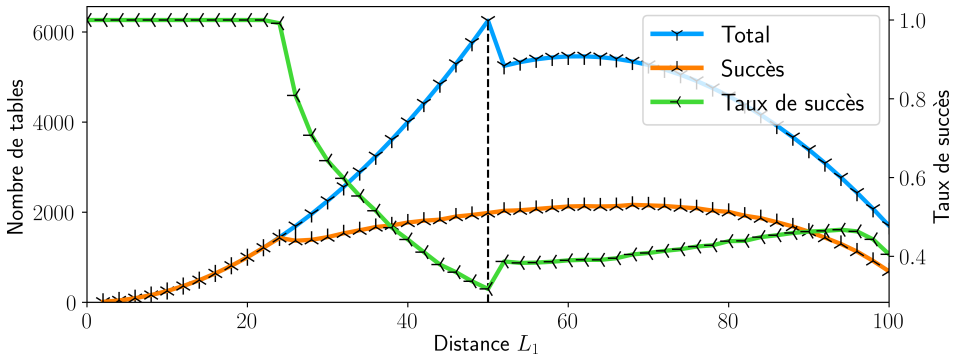

FIGURE 3.4 – Nombre total de tables (bleu), nombre total de succès (orange) et taux de succès (vert) en fonction de la distance L_1 par rapport à la table modèle 2×2 qui possède des entrées égales à 25. L'échelle de la courbe verte est représentée à droite du graphique. Un succès correspond à accepter H_0. La barre pointillée verticale se situe à $L_1 = L_{1_{max}} = 50$.

Algorithme 7 Génération de l'ensemble des tables valides se situant à une distance L_1 donnée d'une table modèle

Entrées Table modèle de dimension 2×2 ;
Un entier k strictement positif tel que $2k$ est la valeur de L_1 désirée.
Sortie L'ensemble des tables valides se situant à une distance L_1 de la table modèle.

1: Générer les partitions de l'entier k ayant une longueur de 1 à 3 termes.
2: Pour chaque partition, créer une liste qui contient les termes. Nous obtenons ainsi l'ensemble des listes positives.
3: Créer une copie de chaque liste et multiplier les termes dans ces copies par -1. Nous obtenons ainsi l'ensemble des listes négatives.
4: Créer toutes les combinaisons composées d'une liste positive et d'une liste négative.
5: Concaténer chaque paire de listes et ne conserver que les listes résultantes de taille 2 à 4 inclusivement.
6: Pour chaque liste résultante de taille inférieure à 4, ajouter des entrées nulles dans la liste de sorte qu'elle contienne 4 éléments.
7: Pour chaque liste résultante, générer toutes les permutations distinctes des éléments qu'elle contient.
8: Transformer ces permutations en matrices 2×2.
9: Ajouter ces matrices à des copies de la table modèle en ne conservant que les matrices résultantes dont aucune entrée est négative.

3.3.2 Performance sur les tables 2×2

À partir du coefficient ϕ de la section 3.1.2, nous pouvons trouver la forme des tables 2×2 qui respectent l'indépendance pure ($\phi = 0$) et la dépendance pure ($\phi = \pm 1$). Un exemple de table qui respecte l'indépendance pure est

	$B=0$	$B=1$
$A=0$	x	x
$A=1$	x	x

,

où $x > 0$. Une telle table produit une statistique $\chi^2_0 = 0$ lorsque comparées à la table des valeurs espérées sous le modèle d'indépendance. Sur la figure 3.5, le taux de succès pour retrouver l'indépendance sur l'ensemble des tables perturbées de distance L_1 par rapport à la table modèle est tracé en fonction des distances L_1 pour différentes valeurs de N. Les entrées dans ces tables sont égales à $N/4$.

Pour N croissant, les valeurs maximales de L_1 qui permettent de conserver un taux de succès de 100% (notées L_1^{seuil}) sont 22, 34 et 42. Cela indique que les tables qui représentent le modèle d'indépendance pure sont de plus en plus résistantes aux perturbations à mesure que le nombre d'observations augmente.

Cela suggère de nouveau qu'il est préférable, lors de la prise de mesures, de recueillir l'échantillon le plus grand possible. Il est à noter que le taux de succès minimal observé sur la figure 3.5 pour la courbe $N = 100$ est atteint pour une valeur de $L_{1_{max}} = 50$. Pour cette valeur, il existe des perturbations de la forme $[-25, 25, 0, 0]$, qui vont permettre d'annuler l'une des entrées dans la table modèle.

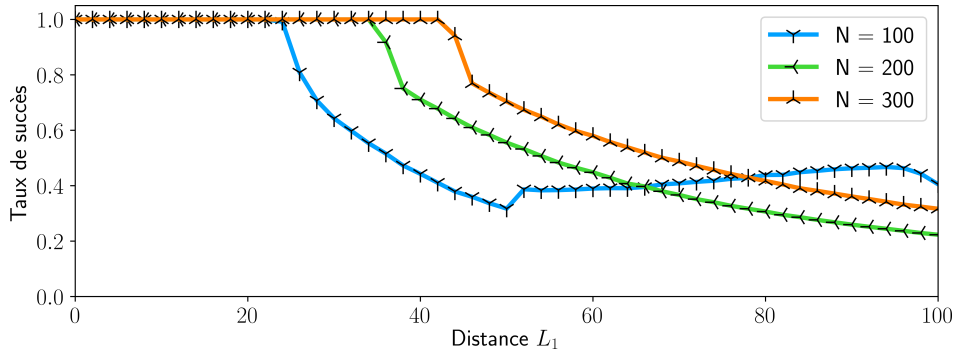

FIGURE 3.5 – Taux de succès pour retrouver l'indépendance pure en fonction des perturbations de la table modèle pour différentes valeurs de N. Pour chaque valeur de N, la table modèle possède des entrées égales à $N/4$.

Le second cas de figure correspond à celui des tables 2×2 qui représentent le modèle de dépendance pure, c'est-à-dire les tables de la forme

	$B=0$	$B=1$
$A=0$	x	0
$A=1$	0	y

et

	$B=0$	$B=1$
$A=0$	0	y
$A=1$	x	0

,

avec $x > 0$ et $y > 0$. Ces dernières produisent une statistique $\chi_0^2 = N$ lorsque comparées à la table des valeurs espérées sous le modèle d'indépendance. Ainsi, pour rejeter le modèle d'indépendance, il faut au minimum $N = 7$, car à $N = \chi_0^2 = 6$ la valeur-p est 0.0143 alors qu'elle est de 0.00815 pour $N = \chi_0^2 = 7$. Les courbes du taux de succès pour des tables dont les entrées positives sont égales à $N/2$ se retrouvent sur la figure 3.6.

Pour N croissant, les valeurs de L_1^{seuil} sont 66, 80 et 96. Il s'agit donc de la même tendance que pour le cas de l'indépendance pure, mais elles sont plus robustes que dans le premier cas.

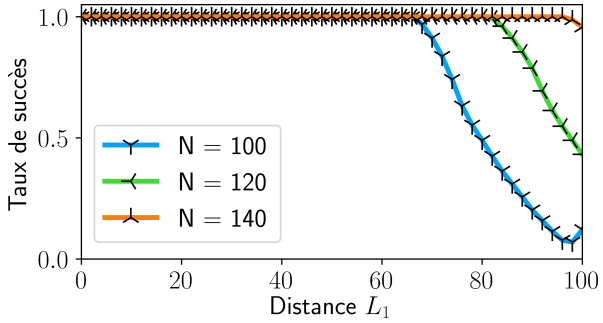

FIGURE 3.6 – Taux de succès pour retrouver la dépendance pure en fonction des perturbations de la table modèle pour différentes valeurs de N. Pour chaque valeur de N, la table modèle possède des entrées égales à $N/2$ sur la diagonale et 0 sur l'antidiagonale.

Il est à noter que comparer entre elles les tables qui respectent la dépendance pure comporte une ambiguïté qui n'est pas présente dans le cas des tables qui respectent l'indépendance pure. En effet, une table qui respecte l'indépendance pure produit une statistique χ_0^2 nulle puisque $x_{ij} = \hat{m}_{ij}$ pour tous les couples (i, j). Les résultats pour une table à N_1 observations peuvent alors être comparés à ceux d'une table à N_2 observations (avec $N_1 \neq N_2$) puisque les deux tables modèles possèdent la même statistique.

Or, pour deux tables qui ne respectent pas l'indépendance pure, la statistique χ_0^2 est plus grande que 0. La somme dans l'équation (3.27) n'est donc pas nulle et la statistique dépend également de N. Contrairement au cas de l'indépendance pure, modifier les entrées de la première table par un facteur N_2/N_1 ne préserve pas la statistique de la table à N_1 observations.

Dans la figure 3.6, les statistiques pour chacune des courbes sont alors 100, 120 et 140. Or, pour de telles valeurs, les valeurs-p sont pratiquement nulles, ce qui justifie la comparaison. Pour des tables n'ayant pas une statistique aussi élevée, il faut toutefois trouver un moyen de créer des tables qui possèdent un nombre d'observations différent, mais la même statistique.

À cette fin, nous devons donc résoudre le système d'équations sous-déterminé

$$\sum_{i,j} \frac{(x_{ij} - \hat{m}_{ij})^2}{\hat{m}_{ij}} = \chi_0^2, \tag{3.54}$$

$$\sum_{i,j} x_{ij} = N. \tag{3.55}$$

Dans ce système les x_{ij} sont les inconnus et $\hat{m}_{ij}$ est une fonction des inconnus, c'est-à-dire

$$\hat{m}_{ij} = \frac{x_{i+}x_{+j}}{N}, \tag{3.56}$$

tandis que les paramètres sont N et χ_0^2.

Il existe potentiellement une infinité de solutions à ce système. Afin de le contraindre davantage, nous pouvons fixer deux termes x_{ij} et résoudre le système par des méthodes numériques comme la méthode de la sécante ou la méthode de Newton-Raphson. Ainsi, nous pouvons obtenir une table avec $N_2 > N_1$ qui préserve la même statistique. Cela dit, cette procédure présente quelques problèmes.

D'abord, la solution trouvée peut comporter des valeurs non entières ou négatives. Pour le bien de la comparaison, les valeurs non entières sont acceptées, puisqu'elles permettent tout de même d'appliquer le test statistique. Du point de vue des tables de probabilités, ces entrées non entières, divisées par le nombre total d'observations, donnent des probabilités valides. C'est plutôt au sens des tables de contingence qu'il est inexact de considérer ces tables. D'ailleurs, si des entrées négatives se présentent, la table résultante est rejetée.

Le second problème de cette procédure est qu'elle ne conserve pas les p_{ij} de la table modèle, ce qui préserve une certaine ambiguïté dans la comparaison entre les deux tables. En effet, même si nous avons trouvé une table valide ayant la même statistique que la première, il existe un ensemble de tables ayant la même statistique. Nous pouvons explorer cet ensemble en fixant différents x_{ij} à différentes valeurs (pourvu de respecter la contrainte (3.55)).

Afin de palier ce problème, nous avons fait le choix de toujours fixer les mêmes entrées d'une table à l'autre et de faire en sorte que les p_{ij} des entrées fixées soient les mêmes d'une table à l'autre. Toutefois, cela ne garantit pas que les p_{ij} qui n'ont pas été fixés seront égaux d'une table à l'autre.

Dans le cas de la dépendance pure, nous pouvons toutefois tester des tables qui possèdent la même statistique, mais qui présentent des entrées différentes. Par exemple, les tables

	$B=0$	$B=1$
$A=0$	20	0
$A=1$	0	80

,

	$B=0$	$B=1$
$A=0$	30	0
$A=1$	0	70

,

	$B=0$	$B=1$
$A=0$	40	0
$A=1$	0	60

,

	$B=0$	$B=1$
$A=0$	50	0
$A=1$	0	50

,

produisent chacune une statistique $\chi^2_0 = 100$, mais leur tolérance aux perturbations est différente, tel qu'illustré à la figure 3.7. Pour des valeurs de plus en plus symétriques dans la table modèle, les valeurs de L_1^{seuil} sont 44, 56 et 66. Les valeurs de $L_1^{\max}$ suivent également cette croissance selon N. Ainsi, plus des entrées dans la table sont près de 0 pour une table respectant le modèle de dépendance pure, plus L_1^{seuil} est bas.

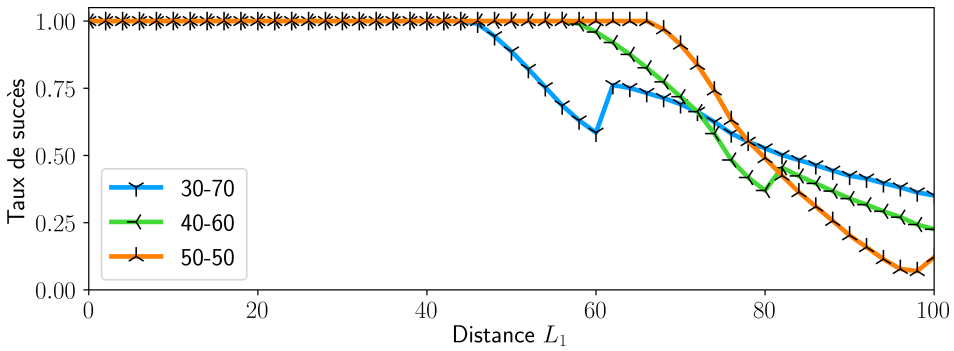

FIGURE 3.7 – Taux de succès pour retrouver la dépendance pure en fonction des perturbations de la table modèle. Les courbes représentent des tables qui respectent la dépendance pure avec $N = 100$, mais dont les entrées x_{11} sont 30, 40 et 50, alors que les entrées x_{22} sont $N - x_{11}$.

En utilisant la méthode proposée pour générer des tables qui possèdent la même statistique, mais un nombre total d'observations différent, nous avons généré trois graphiques du taux de succès en fonction de la distance L_1. Les trois statistiques sélectionnées sont 10 (valeur-$p = 0.00157$), 2.5 (valeur-$p = 0.1138$) et 0.5 (valeur-$p = 0.4795$). Cela a permis de produire les figures 3.8, 3.9 et 3.10. Les tables associées à chacune des figures se retrouvent à l'Annexe B. Il est à noter que les entrées n'ont pas été transformées en entier, car même si les valeurs non entières ne respectent pas l'essence d'une table de contingence, elles permettent le calcul d'une table de valeurs espérées et de conserver la statistique désirée.

Sur ces trois figures, nous remarquons que les courbes suivent les mêmes tendances que celles exposées dans le cas de l'indépendance et de la dépendance pure. En effet, lorsqu'inférieurs à 1.0, les taux de succès pour une distance L_1 donné sont toujours ordonnés en ordre croissant selon N, c'est-à-dire que le taux de succès le plus bas est associé à $N = 100$, suivi de celui pour $N = 200$ jusqu'au plus élevé à $N = 400$.[16] En général, les valeurs de L_1^{seuil} suivent également une croissance selon N. Les seules exceptions se produisent aux figures 3.8 et 3.9 où les valeurs

16. Cela n'est vrai que pour $0 \leq L_1 \leq min(L_{1_{max}})$, où $min(L_{1_{max}})$ est la plus petite valeur de $L_1^{\max}$ parmi les courbes comparées

de L_1^{seuil} sont 2, 2, 2 et 4 pour la première figure et 6, 8, 10 et 10 pour la deuxième. Il arrive donc que le taux de succès puisse passer sous les 100% aux mêmes valeurs de L_1, même si deux courbes possèdent un nombre d'observations différent.

Malgré l'utilisation des valeurs de N similaires à celles utilisées dans les cas d'indépendance et de dépendance pure, les valeurs de L_1^{seuil} pour ces trois figures sont inférieures à la valeur de L_1^{seuil} la plus basse atteinte dans le cas de l'indépendance pure, c'est-à-dire $L_1^{\text{seuil}} = 22$. Ainsi, il semble que les tables modèles n'appartenant pas à une forme pure d'indépendance ou de dépendance sont généralement plus sensibles aux perturbations. Cela est alors un rappel que la prise de mesures doit être la plus rigoureuse possible, c'est-à-dire en récoltant un échantillon de grande taille et en utilisant des instruments et des protocoles précis.

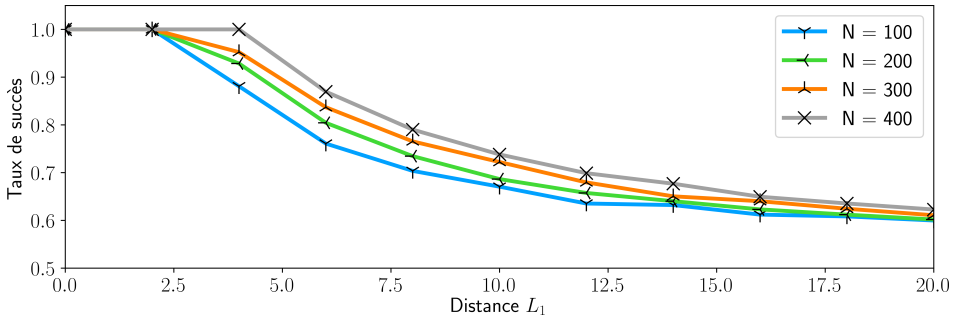

FIGURE 3.8 – Taux de succès pour retrouver la dépendance en fonction des perturbations de tables modèles 2×2 produisant une statistique $\chi_0^2 = 10$ (modèle de dépendance) pour différentes valeurs de N. Les tables utilisées se retrouvent à l'Annexe B.

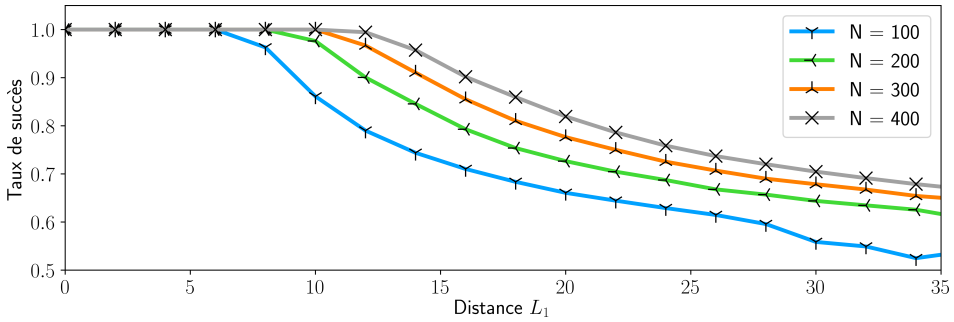

FIGURE 3.9 – Taux de succès pour retrouver l'indépendance en fonction des perturbations de tables modèles 2×2 produisant une statistique $\chi_0^2 = 2.5$ (modèle d'indépendance) pour différentes valeurs de N. Les tables utilisées se retrouvent à l'Annexe B.

L'analyse des perturbations est une bonne opportunité pour comparer les résultats d'un test exact et d'un test asymptotique. À la figure 3.11, nous avons utilisé la méthode exacte qui génère une distribution à $2^d - 1$ degrés de liberté et un test asymptotique sur une table modèle de la forme

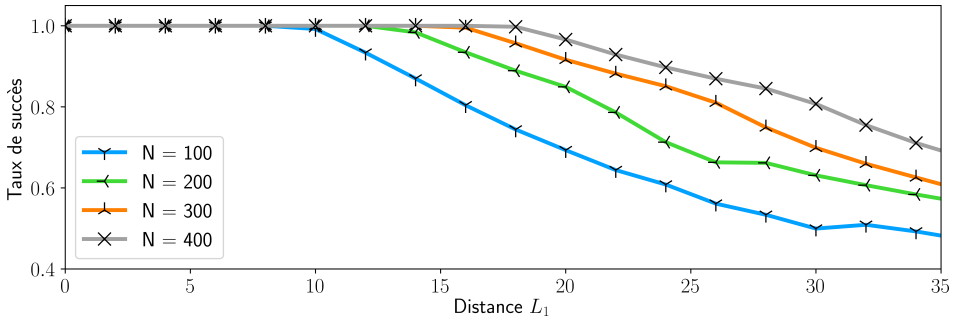

FIGURE 3.10 – Taux de succès pour retrouver l'indépendance en fonction des perturbations de tables modèles 2×2 produisant une statistique $\chi^2_0 = 0.5$ (modèle d'indépendance) pour différentes valeurs de N. Les tables utilisées se retrouvent à l'Annexe B.

	$B = 0$	$B = 1$
$A = 0$	10	0
$A = 1$	0	10

.

Cette table est donc un modèle de dépendance avec une statistique $\chi^2_0 = N = 20$. Nous pouvons alors observer que les résultats du test exact sont plus conservateurs que pour le test asymptotique, puisque les valeurs de L_1^{seuil} sont 4 et 8. Ainsi, le test exacte rejette moins longtemps H_0 que le test asymptotique. Pour de petites valeurs de N, le test exact est donc plus conservateur, ce qui est souhaitable lorsque nous avons peu d'observations.

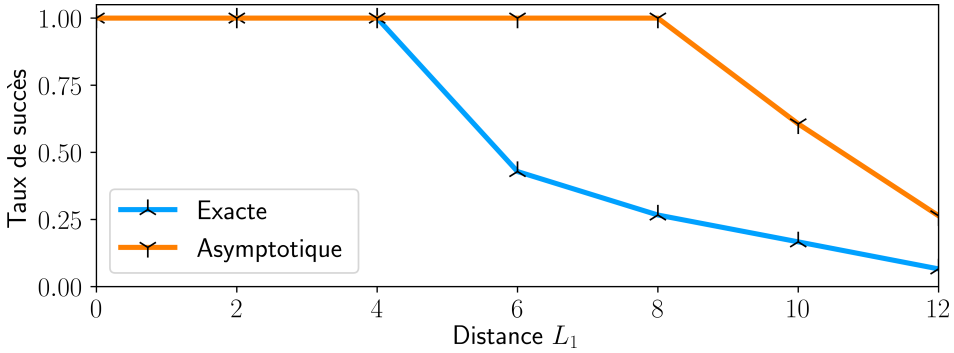

FIGURE 3.11 – Taux de succès pour retrouver la dépendance pure en fonction des perturbations de la table modèle 2×2 possédant des entrées égales à 10 sur la diagonale et 0 sur l'antidiagonale. La méthode exacte produisant une loi du χ^2 à $2^d - 1$ degrés de liberté (courbe bleue) est comparée à un test asymptotique (courbe orangée).

3.3.3 Performance sur les tables $2 \times 2 \times 2$

En augmentant le nombre de dimensions dans la table de contingence, nous augmentons également le nombre de perturbations possibles. En effet, en trois dimensions, l'algorithme

présenté à la section 3.3.1 doit maintenant retenir les perturbations dont la longueur est située entre 2 et 8 inclusivement en plus de générer toutes les permutations distinctes possibles. Il n'est donc pas toujours possible d'obtenir le taux de succès pour une valeur de L_1 dans un temps raisonnable. Or, l'étape limitante n'est pas nécessairement l'identification de toutes les perturbations, mais bien le calcul de la statistique sur chacune des tables perturbées en passant par l'algorithme itératif.

Afin d'accélérer le processus et d'évaluer le taux de succès à une distance L_1 donnée, nous pouvons générer l'entièreté des perturbations et sélectionner un échantillon aléatoire pour estimer le taux de succès à cette même valeur. Nous utiliserons donc cette technique pour évaluer les performances sur des tables $2 \times 2 \times 2$.

La première étape est alors d'identifier quelles tables modèles représentent les cas limites, comme nous l'avons fait pour les tables 2×2. Dans ce cas-ci, le test d'hypothèse à effectuer correspond au test du modèle sous-saturé. De manière analogue aux tables 2×2, l'une des tables qui nous permettra d'accepter systématiquement l'hypothèse nulle est celle où les entrées sont supérieures à zéro et égales entre elles. Autrement dit, il s'agit des tables ayant la forme

	$B=0$	$B=1$
$A=0$	x	x
$A=1$	x	x

$C=0$ $\qquad$

	$B=0$	$B=1$
$A=0$	x	x
$A=1$	x	x

$C=1$,

avec $x > 0$. Dans un tel cas, l'algorithme itératif produit une table de valeurs espérées identique à la table modèle, ce qui produit une statistique $\chi_0^2 = 0$ et implique une valeur-p de 1. Ce résultat est le même, peu importe le modèle log-linéaire hiérarchique testé. Cela implique alors que le modèle le plus plausible est le modèle d'indépendance complète de l'équation (3.37), puisqu'il s'agit du modèle le plus simple qui permet d'accepter l'hypothèse nulle.

En effectuant une somme sur l'un des axes de la table $2 \times 2 \times 2$, nous pouvons retrouver les trois tables 2×2 formées par les couples de variables aléatoires. Dans le cas présent, celles-ci prennent la forme

	$B=0$	$B=1$
$A=0$	$2x$	$2x$
$A=1$	$2x$	$2x$

,

peu importe le couple de variables considéré. De ce fait, les paires de variables aléatoires sont aussi considérées comme étant indépendantes, ce qui est attendu si le modèle le plus approprié sur la table $2 \times 2 \times 2$ est celui de l'indépendance totale.

Pour générer les courbes dans cette section, nous avons généré 10 échantillons de plus de 1000 tables pour chaque valeur de L_1.[17] Les courbes représentent alors le taux de succès moyen sur ces 10 échantillons et les régions ombragées correspondent à l'écart-type autour de ces moyennes.

La courbe du taux de succès pour les tables $2 \times 2 \times 2$ où les variables sont purement indépendantes se retrouve à la figure 3.12. Les tables analysées possèdent des entrées égales à $N/8$. Ici, le succès représente l'acceptation du modèle sous-saturé, même si nous savons que le meilleur modèle pour ces tables est celui de l'indépendance complète. Pour N croissant, les valeurs approximatives pour L_1^{seuil} sont 20, 30 et 38. La tendance est donc exactement la même que dans le cas des tables 2×2.

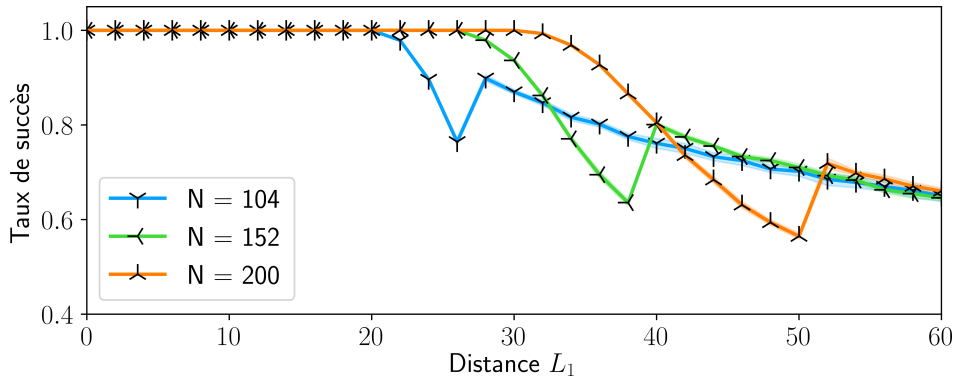

FIGURE 3.12 – Taux de succès pour retrouver l'indépendance pure en fonction des perturbations de la table modèle pour différentes valeurs de N. Pour chaque valeur de N, la table $2 \times 2 \times 2$ modèle possède des entrées égales à $N/8$. Les écarts-types sont cachés par l'épaisseur des traits.

À l'opposé, en s'inspirant de l'équation (3.27), nous pouvons inférer que les tables $2 \times 2 \times 2$ qui possèdent une antidiagonale nulle dans l'une des couches et une diagonale nulle dans l'autre couche, c'est-à-dire les tables de la forme

	$B=0$	$B=1$
$A=0$	x	0
$A=1$	0	x

$C=0$

	$B=0$	$B=1$
$A=0$	0	x
$A=1$	x	0

$C=1$,

où $x > 0$, produiront une statistique $\chi^2_0 = N$ si nous testons le modèle d'indépendance complète. Cela est également vrai lorsque nous testons le modèle sous-saturé avec la méthode itérative (3.48), puisqu'au premier tour, les valeurs de $\hat{m}_{ijk}$ seront $x/2$ et, au deuxième tour,

17. Pour $L_1 < 4$, le nombre total de perturbations est inférieur à 1000. Les taux de succès sont alors exacts. De plus, nous utilisons l'expression « plus de 1000 tables », car pour chaque valeur de L_1, la taille de l'échantillon n'était pas constante, mais toujours supérieure à 1000.

elles resteront identiques. Cela produit donc une statistique $\chi_0^2 = 4x = N$. Les courbes du taux de succès ont été tracées à la figure 3.13 où les entrées non nulles dans les tables sont égales à $N/4$. Pour N croissant, les valeurs estimées de L_1^{seuil} sont 28, 38 et 48. Encore une fois, il s'agit de la même tendance que pour les tables 2×2.

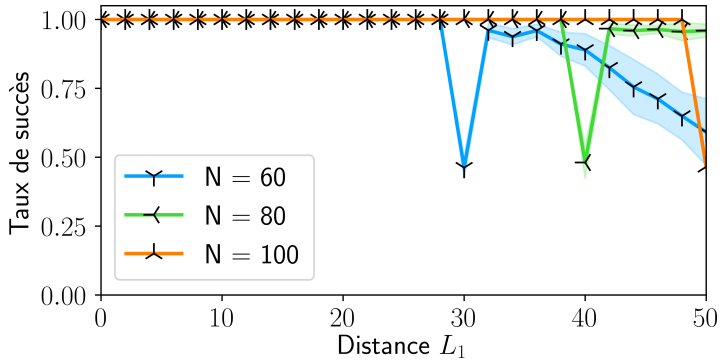

FIGURE 3.13 – Taux de succès pour retrouver la dépendance pure en fonction des perturbations de la table modèle pour différentes valeurs de N. Pour chaque valeur de N, la table $2 \times 2 \times 2$ modèle possède des entrées égales à $N/4$ sur la diagonale de la première couche et sur l'antidiagonale de la seconde. Les autres entrées étaient nulles.

En sommant sur l'un des axes de la table modèle, nous obtenons les tables 2×2 dont l'allure est la suivante :

	$B = 0$	$B = 1$
$A = 0$	x	x
$A = 1$	x	x

.

Ces dernières sont identiques aux tables où le modèle d'indépendance triple est le plus plausible, à un facteur près.

Cela signifie alors que l'inférence peut indiquer que trois variables sont indépendantes en paires et indépendantes en triplet ou indépendantes en paires mais dépendantes en triplet. Dans ce cas, le rejet du modèle sous-saturé produit un hyperlien entre les trois variables. Si cette situation s'était présentée dans nos données expérimentales et que nous avions utilisé la méthode d'inférence par étape pour inférer les 2-simplexes, cette interaction n'aurait pas été détectée. C'est seulement en testant tous les triplets, même ceux où les entités sont restées complètement ou partiellement indépendantes au premier tour, que nous aurions pu inférer cette interaction d'ordre supérieur.

De manière analogue aux tables 2×2, nous pouvons aussi nous éloigner du cas particulier où toutes les entrées dans la table $2 \times 2 \times 2$ sont égales et observer les tables

	$B=0$	$B=1$
$A=0$	x	0
$A=1$	0	y

$C=0$

	$B=0$	$B=1$
$A=0$	0	w
$A=1$	z	0

$C=1$

,

où $x > 0$, $y > 0$, $z > 0$, $w > 0$ et où la diagonale dans une couche doit être nulle et l'antidiagonale dans l'autre couche doit l'être également. Dans ce cas-ci, le comportement de la statistique selon le modèle testé est un peu différent, mais, empiriquement nous avons observé que peu importe le modèle, la statistique est systématiquement plus grande ou égale à N. Comme les conclusions sont les mêmes que pour le cas 2×2, les figures réalisées se retrouvent plutôt à l'Annexe C.

À la différence des tables 2×2, il n'est pas possible d'utiliser le système d'équations composé des équations (3.54) et (3.55) pour obtenir des tables qui possèdent la même statistique sous le modèle sous-saturé. En effet, avec ce modèle, nous ne possédons pas d'expression analytique pour les $\hat{m}_{ijk}$ [18], d'où la nécessité d'utiliser l'algorithme itératif.

Cela fait en sorte que nous ne pouvons pas résoudre le système d'équations par la méthode de la sécante ou par la méthode de Newton-Raphson. Or, puisque les tendances dans les cas limites sont similaires à celles des tables 2×2, nous pouvons supposer que le comportement serait le même pour des tables $2 \times 2 \times 2$ ayant la même statistique. Afin de s'en convaincre, nous avons tout de même tracé la courbe de robustesse pour quelques cas particuliers présentés à la figure 3.14.

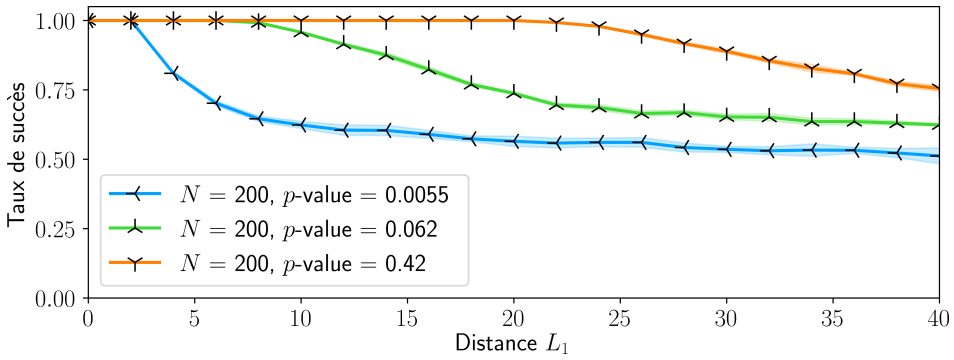

FIGURE 3.14 – Taux de succès pour diverses tables $2 \times 2 \times 2$. À chaque courbe est associée une table modèle ayant produit les valeurs-p 0.0055, 0.062 et 0.42 lors du test du modèle sous-saturé. Chacune des tables possède $N = 200$ observations.

En plus des cas présentés ci-haut, l'analyse des perturbations peut aussi être effectuée sur des données réelles. En effet, même si les tables obtenues sont influencées par les instruments et

18. Des expressions analytiques existent toutefois pour les autres modèles hiérarchiques à trois variables.

la méthode d'échantillonnage, l'analyse des perturbations permet d'ajuster notre niveau de confiance en nos conclusions et notre méthode expérimentale.

Par exemple, outre le seuil de confiance α, nous pouvons décider de ne conserver que les conclusions qui persistent sur une certaine plage de valeurs de la distance L_1. Cette approche est alors plus conservatrice et permet de réduire le nombre de faux positifs. Aussi, si la majorité des tables tombent à un taux de succès de 50% pour de petites valeurs de bruit (disons $L_1 = 4$), alors il y a lieu de penser que la méthode d'échantillonnage pourrait être améliorée. L'observateur pourrait alors décider d'augmenter son nombre d'observations ou bien utiliser des instruments plus précis.

Quoiqu'il en soit, l'analyse des perturbations révèle donc plusieurs tendances générales. Premièrement, plus le nombre d'observations est élevé, plus la valeur de L_1^{seuil} le sera aussi. Deuxièmement, entre L_1^{seuil} et $L_1^{\max}$, les taux de succès seront plus élevés pour la table qui possède plus d'observations qu'une autre, même si les deux tables possèdent la même statistique. Troisièmement, la valeur de L_1^{seuil} dépend également de la statistique associée à la table modèle. Il semble que plus la statistique produit une valeur-p près des valeurs extrêmes, c'est-à-dire 1 et 0, plus la valeur de L_1^{seuil} augmente. Il semble également que ces constats soient les mêmes pour toutes les tables dans toutes les dimensions. Finalement, l'analyse des perturbations peut aussi être un outil pour ajuster notre niveau de confiance par rapport aux conclusions obtenues sur des données expérimentales. La prochaine section s'attarde d'ailleurs à l'application des méthodes sur des données synthétiques et réelles.

Chapitre 4

Inférence d'interactions d'ordre supérieur sur données synthétiques et réelles

Les méthodes d'inférence étant complètes, il ne reste plus qu'à les appliquer sur des données. Afin de les tester dans un environnement contrôlé, nous avons élaboré un algorithme de génération de données synthétiques. Ce dernier, basé sur les graphes de facteurs et la méthode du rejet pour l'échantillonnage, est d'ailleurs présenté à la section 4.1. Les quelques scénarios que nous avons générés sont présentés par la suite pour faire l'analyse des performances des méthodes d'inférence. À la section 4.2, nous appliquons également les méthodes d'inférence sur des données réelles de présence/absence provenant de l'échantillonnage d'OTUs, d'espèces d'oiseaux et de MeSHes dans la base de données MEDLINE.

4.1 Données synthétiques

Le but principal du cadre théorique développé au chapitre 3 est l'inférence d'interactions d'ordre supérieur au sein de données de présence/absence. Bien que quelques jeux de données nous aient été offerts, il était judicieux d'élaborer un modèle génératif pour produire des données synthétiques. En effet, de telles données permettent de nous assurer que nos observations sur les « sites » sont indépendantes et identiquement distribuées.

De plus, avec ce modèle génératif, nous connaissons la nature de toutes les interactions entre les groupes d'entités du complexe simplicial. Il est donc possible de faire une comparaison directe entre les résultats d'une méthode d'inférence et la vérité. Avant de pouvoir se lancer dans l'analyse de différents scénarios, nous expliquons d'abord comment fonctionne le modèle génératif. Nous entamons le pas en présentant le concept au coeur de cette méthode, c'est-à-dire les graphes de facteurs. Les explications qui suivent sont basées sur les références [34] et

[44].

4.1.1 Graphes de facteurs

Soit **D**, un ensemble de variables aléatoires. Un **facteur**, dénoté par Φ, est une fonction des valeurs des variables dans **D** vers $\mathbb{R}$. Pour les cas qui nous intéressent, l'image des facteurs sera limitée à l'intervalle $[0, \infty)$.

Par exemple, pour deux variables aléatoires, disons X et Y, nous écrivons le facteur qui relie ces variables comme $\Phi(X, Y)$. Si le support de ces deux variables aléatoires est $\{0, 1\}$, alors $\Phi(X, Y)$ doit pouvoir prendre quatre valeurs, notées $\Phi(0, 0)$, $\Phi(0, 1)$, $\Phi(1, 0)$, $\Phi(1, 1)$.

La valeur d'un facteur pour un état donné indique, en quelque sorte, à quel point un état est probable. Ainsi, en comparant deux valeurs d'un facteur, nous pouvons dire quelle situation est la plus probable. Par exemple, si $\Phi(0, 0) < \Phi(1, 1)$, nous pouvons dire qu'il est plus probable d'obtenir le résultat $(X = 1, Y = 1)$ que le résultat $(X = 0, Y = 0)$ lors d'une expérience aléatoire.

À partir des facteurs, nous pouvons d'ailleurs définir les probabilités conjointes des états des variables aléatoires dans l'ensemble **D**. En effet, en normalisant chaque valeur la somme de tous les facteurs, nous pouvons écrire

$$P(X = x, Y = y) = \frac{1}{Z}\Phi(x, y), \tag{4.1}$$

où

$$Z = \sum_{R_x, R_y} \Phi(X, Y) \tag{4.2}$$

est la fonction de partition. Si nous avions plusieurs variables aléatoires dans le système considéré et que nous avions défini M facteurs, la probabilité conjointe s'écrirait comme

$$P(\mathbf{D}) = \frac{1}{Z}\prod_{i=1}^{M}\Phi_i, \tag{4.3}$$

où

$$Z = \sum_{\forall \text{ états de } \mathbf{D}} \left(\prod_{i=1}^{M}\Phi_i\right). \tag{4.4}$$

Grâce à ces facteurs, nous pouvons définir un **graphe de facteurs** qui nous sera utile pour générer des données synthétiques. Un graphe de facteurs est un graphe biparti non dirigé. Les noeuds du premier ensemble correspondent aux variables aléatoires du système, tandis que les noeuds du second ensemble représentent les facteurs. Une variable aléatoire est reliée à un facteur si ce facteur est fonction de cette variable. Le graphe des facteurs est alors une représentation graphique de l'expression de la probabilité conjointe des états du système donnée à l'équation (4.3). Un exemple de graphe de facteurs pour les variables aléatoires A, B, C et D est représenté à la figure 4.1 (a).

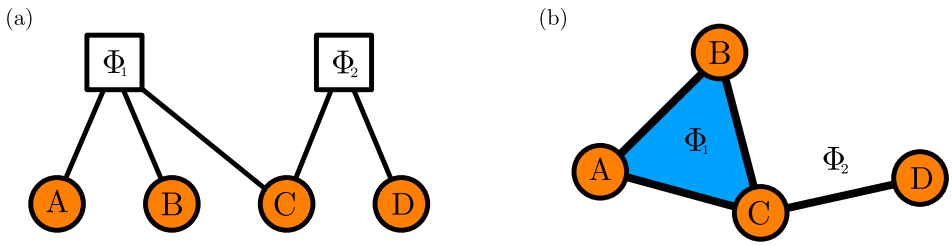

FIGURE 4.1 – a) Graphe de facteurs pour les variables aléatoires A, B, C et D. b) Projection sur les variables aléatoires pour obtenir un complexe simplicial.

4.1.2 Modèle génératif

À l'image du modèle d'Ising, si nous connaissons la distribution des probabilités conjointes d'un graphe de facteurs, ce qui est le cas si nous concevons nous-mêmes les facteurs, nous pouvons générer des états du système. Dans ce cas, au lieu d'être une chaîne où les variables prennent les valeurs -1 ou 1, nous obtenons un réseau biparti où les noeuds associés aux variables aléatoires prennent les états 1 (présence) ou 0 (absence). Or, comme nous l'avons vu à la section 2.1, il est possible d'obtenir un complexe simplicial à partir d'un réseau biparti en respectant certaines conditions. De ce fait, si nous construisons adéquatement le graphe de facteurs, nous pouvons générer des données de présence/absence où les relations entre les variables peuvent être représentées par un complexe simplicial, comme à la figure 4.1 (b). Si une telle représentation est possible, les méthodes d'inférence proposées au chapitre 3 doivent être en mesure de retrouver ce complexe simplicial à partir de données de présence/absence. Pour les tester, nous devons donc construire un **modèle génératif** de données de présence/absence qui possède plusieurs caractéristiques que nous présentons ici.

La première étape dans l'élaboration du modèle est la fabrication d'un complexe simplicial pouvant représenter un graphe de facteurs. Dans un tel complexe simplicial, chaque 0-simplexe correspond à une variable aléatoire dont les états sont $\{0, 1\}$. En contrepartie, chaque facette est associée à un facteur comme à la figure 4.1 (b). Par contre, à cette étape, nous ne nous soucions pas des valeurs des facteurs et chaque facette ne fait qu'indiquer quels 0-simplexes lui appartiennent. Comme nous l'avons vu au chapitre 2, un complexe simplicial est entièrement défini par sa liste de facette. Ainsi, nous pouvons construire le complexe simplicial de différentes manières : soit déclarer une liste de facettes spécifique, utiliser le SCM pour extraire une liste de facettes à partir d'un réseau biparti ou générer une liste de facettes aléatoirement. [1]

La deuxième étape est de définir une expression pour chacun des facteurs. En effet, à chacune des facettes, F, est associé un facteur Φ qui est fonction des variables aléatoires représentées par les 0-simplexes qui la composent. Pour une facette donnée, nous écrivons l'expression générale

1. Si nous générons une liste aléatoirement, il faut s'assurer qu'elle ne possède pas de facettes qui sont en fait des faces.

du facteur comme $\Phi(\boldsymbol{\sigma})$, où $\boldsymbol{\sigma} \equiv \{\sigma_i \subseteq F\}$ est l'ensemble des simplexes qui composent la facette. À priori, ces facteurs ne sont limités que par notre imagination, mais pour faciliter la suite, nous allons contraindre leur forme.

La première contrainte correspond à imposer que chaque facteur puisse être factorisé d'une manière particulière. Plus précisément, nous allons imposer que $\Phi(\boldsymbol{\sigma})$ puisse être factorisé comme un produit de facteurs élémentaires ϕ_μ. Le nombre de facteurs élémentaires est déterminé par le nombre de simplexes dans $\boldsymbol{\sigma}$. Pour une facette de dimension d, c'est-à-dire un d-simplexe, nous avons

$$\sum_{\mu=1}^{d+1} \binom{d+1}{\mu} = 2^{d+1} - 1 \tag{4.5}$$

simplexes dans $\boldsymbol{\sigma}$, de sorte que

$$\Phi(\boldsymbol{\sigma}) = \phi_1(\sigma_1)\phi_2(\sigma_2)\ldots\phi_{2^{d+1}-1}(\sigma_{2^{d+1}-1}), \tag{4.6}$$

où l'indice μ des ϕ_μ ne représente qu'une étiquette arbitraire pour représenter un simplexe dans l'ensemble $\boldsymbol{\sigma}$.

Cette factorisation est nécessaire, puisque nous désirons faire le pont entre les facteurs et les modèles log-linéaires. En effet, si nous pouvons factoriser ainsi les facteurs associés à chaque facette, nous pouvons faire de même avec la probabilité conjointe associée à chaque facette (à une constante de normalisation près). Dès lors, en prenant le logarithme de l'expression (4.6), nous obtenons la forme d'un modèle log-linéaire où le logarithme de chacun des facteurs élémentaires représente un terme en u dans le modèle. Plus précisément,

$$\log\left[\Phi(\boldsymbol{\sigma})\right] = \sum_{\mu=1}^{2^{d+1}-1} \log\left[\phi_\mu(\sigma_\mu)\right] = \sum_{\mu=1}^{2^{d+1}-1} u_\mu, \tag{4.7}$$

où $u_\mu \equiv \log\left[\phi_\mu(\sigma_\mu)\right]$.

La deuxième contrainte va de pair avec la première, puisqu'elle permet d'écrire une factorisation comme à l'équation (4.6). En effet, nous suggérons que chacun des facteurs élémentaires ϕ_μ soit une fonction exponentielle telle que

$$\phi_\mu(\sigma_\mu) = e^{-H_\mu(\sigma_\mu)}, \tag{4.8}$$

où H_μ est une **fonction d'énergie**.[2] Ainsi, pour une facette donnée, nous avons

$$\Phi(\boldsymbol{\sigma}) = \prod_{\mu=1}^{2^{d+1}-1} e^{-H_\mu(\sigma_\mu)} = e^{-\sum_\mu H_\mu(\sigma_\mu)} = e^{-H(\boldsymbol{\sigma})}, \tag{4.9}$$

2. Nous l'appelons ainsi pour reconnaître la forme canonique et l'héritage de la physique statistique. Ce même vocabulaire est employé dans la référence [34]. En principe, nous trouverions également un paramètre de température, mais ce dernier est fixé à 1 dans le reste de ce mémoire.

où $H(\boldsymbol{\sigma}) = \sum_\mu H_\mu(\sigma_\mu)$ est la fonction d'énergie totale. Cette forme implique alors que les variables aléatoires appartiennent aux familles exponentielles de distributions. Toute famille exponentielle possède une fonction génératrice des moments, ce qui constitue un net avantage lorsque le calcul de ces quantités est désiré. Un autre avantage de cette fonction est qu'elle rend l'élaboration d'une méthode d'échantillonnage plus aisée, car elle permet d'utiliser des algorithmes connus comme la méthode du rejet.

Par la suite, il ne reste qu'à fabriquer une expression pour $H(\boldsymbol{\sigma})$ qui respectera la première contrainte. À cette fin, nous suggérons de prendre la somme de termes qui représentent les cases d'une table de contingence. Par exemple, si nous sommes en présence de la facette $[1, 2]$, cela signifie que $\boldsymbol{\sigma} = \{1, 2, (12)\}$. À chacun des 0-simplexes, nous associons une variable aléatoire, dans ce cas-ci X_1 et X_2, dont le support est $\{0, 1\}$. La fonction d'énergie est alors

$$H(\boldsymbol{\sigma}) = -\left[aX_1X_2 + b(1 - X_2)X_1 + c(1 - X_1)X_2 + d(1 - X_2)(1 - X_1)\right], \tag{4.10}$$

où les paramètres a, b, c et d sont réels. Il est à noter que l'utilisation d'un signe négatif dans l'équation (4.10) fait en sorte que le paramètre le plus grand est associé à l'énergie la plus basse, ce qui correspond à l'état le plus probable. Les probabilités associées à chacun des états, données par l'expression (4.9) en conjonction avec l'équation (4.3), sont

$$P(X_1 = 1, X_2 = 1) = \frac{e^a}{Z}, \tag{4.11}$$

$$P(X_1 = 1, X_2 = 0) = \frac{e^b}{Z}, \tag{4.12}$$

$$P(X_1 = 0, X_2 = 1) = \frac{e^c}{Z}, \tag{4.13}$$

$$P(X_1 = 0, X_2 = 0) = \frac{e^d}{Z}, \tag{4.14}$$

où $Z = e^a + e^b + e^c + e^d$. L'avantage de choisir la forme (4.10) est alors que la probabilité associée à chacun des états est égale à l'exponentielle du coefficient du terme non nul dans la fonction d'énergie, divisé par la fonction de partition. C'est pour cette raison que nous disons que les termes dans l'équation (4.10) représentent les cases de la table de contingence. L'expression développée du facteur associé à $\boldsymbol{\sigma}$ est donc

$$\begin{aligned}\Phi(\sigma) &= e^{-H(\boldsymbol{\sigma})} \\ &= e^{aX_1X_2 + b(1-X_2)X_1 + c(1-X_1)X_2 + d(1-X_1)(1-X_2)} \\ &= e^{(a-b-c+d)X_1X_2} e^{(b-d)X_1} e^{(c-d)X_2} e^d.\end{aligned} \tag{4.15}$$

Ce dernier possède la forme de l'équation (4.6), tel que désiré. Nous pouvons alors écrire le modèle log-linéaire comme

$$\log\left[P(X_1 = x, X_2 = y)\right] = \log\left(\frac{e^d}{Z}\right) + \log\left(e^{(b-d)x}\right) + \log\left(e^{(c-d)y}\right) + \log\left(e^{(a-b-c+d)xy}\right). \tag{4.16}$$

Autrement dit, nous pouvons écrire l'équation

$$\log\left[P(X_1 = i, X_2 = j)\right] = u + u_i^{X_1} + u_j^{X_2} + u_{ij}^{X_1 X_2}, \tag{4.17}$$

où nous reconnaissons la forme du modèle log-linéaire saturé pour deux variables. Or, cette expression ne garantit pas la dépendance entre les variables. En effet, ce sont les coefficients a à d qui dictent la relation entre les variables. Par exemple, si $a = b = c = d$, le seul terme non nul dans l'équation (4.16) est $\log\left(\frac{e^d}{Z}\right)$ et la probabilité pour chacun des états devient 1/4. En appliquant le test d'indépendance sur cette table de probabilités, nous obtenons alors une statistique de 0 et une valeur-p de 1.

Ainsi, après avoir sélectionné différentes valeurs de a, b, c et d pour la facette $[1, 2]$, il est important d'effectuer le test d'indépendance sur sa table de probabilités multipliée par le nombre d'observations que nous voulons générer. Cela permet de connaître la dépendance statistique réelle entre les variables X_1 et X_2.[3] Si la valeur-p obtenue est inférieure au seuil α que nous avons fixé, alors les deux variables aléatoires sont considérées comme dépendantes.

Pour un $(M-1)$-simplexe, la forme générale de l'équation (4.10) est

$$\sum_{\eta_1 \in \{0,1\}} \cdots \sum_{\eta_N \in \{0,1\}} c_{\eta_1,\ldots,\eta_N} \prod_i X_i^{\eta_i} (1 - X_i)^{1-\eta_i} \tag{4.18}$$

où η_i est un paramètre qui détermine si c'est la variable X_i ou $(1 - X_i)$ qui fait partie du terme et où les coefficients $c_{\eta_1,\ldots,\eta_N}$ sont réels et définis préalablement. Le développement de l'équation (4.18) est alors un polynôme où toutes les combinaisons des variables aléatoires, de longueur 1 à N existent. La forme du facteur de l'équation (4.6) est alors respectée. Du même coup, nous pouvons associer un terme en u à chacun de ces termes dans le modèle log-linéaire résultant, comme nous l'avons fait aux équations (4.16) et (4.17). En utilisant l'équation (4.18) pour un 2-simplexe, nous avons la fonction d'énergie suivante :

$$\begin{aligned} H(\sigma) =& c_{111} X_1 X_2 X_3 + c_{110}(1 - X_3) X_2 X_1 + c_{101}(1 - X_2) X_3 X_1 + c_{011}(1 - X_1) X_3 X_2 \\ &+ c_{100}(1 - X_3)(1 - X_2) X_1 + c_{010}(1 - X_3)(1 - X_1) X_2 + c_{001}(1 - X_2)(1 - X_1) X_3 \\ &+ c_{000}(1 - X_3)(1 - X_2)(1 - X_1), \end{aligned} \tag{4.19}$$

où les coefficients de la forme $c_{\eta_1 \eta_2 \eta_3}$ sont réels.

Afin de complètement définir un facteur, il ne reste plus qu'à spécifier judicieusement les coefficients dans la fonction d'énergie. À cette fin, nous suggérons une simplification supplémentaire en choisissant des coefficients qui correspondent au logarithme d'un nombre réel strictement

3. Nous pouvons également mesurer le coefficient ϕ décrit à la section 3.1.2. Empiriquement, il semble que pour $|\phi| > 0.1$, la valeur-p associée au test d'indépendance sera inférieure à 0.01.

positif. Par exemple, en définissant les coefficients [4]

$$a = \ln(a'), \tag{4.20}$$

$$b = \ln(b'), \tag{4.21}$$

$$c = \ln(c'), \tag{4.22}$$

$$d = \ln(d'), \tag{4.23}$$

où a', b', c' et d' sont des réels strictement positifs, les probabilités données aux équations (4.11) à (4.14) deviennent

$$P(X_1 = 1, X_2 = 1) = \frac{a'}{a' + b' + c' + d'}, \tag{4.24}$$

$$P(X_1 = 1, X_2 = 0) = \frac{b'}{a' + b' + c' + d'}, \tag{4.25}$$

$$P(X_1 = 0, X_2 = 1) = \frac{c'}{a' + b' + c' + d'}, \tag{4.26}$$

$$P(X_1 = 0, X_2 = 0) = \frac{d'}{a' + b' + c' + d'}. \tag{4.27}$$

Si la fonction de partition $Z = a' + b' + c' + d'$ est normalisée, les probabilités sont alors directement égales aux nombres a' à d' pour leur état respectif. Autrement dit, en choisissant des logarithmes de nombres réels strictement positifs pour les coefficients dans la fonction d'énergie et en s'assurant que la fonction de partition soit normalisée, nous définissons directement la table de probabilités pour un groupe de variables. Par exemple, avec les équations (4.24) à (4.27), nous avons la table de probabilités suivante :

	$X_2 = 0$	$X_2 = 1$
$X_1 = 0$	d'	c'
$X_1 = 1$	b'	a'

,

que nous pouvons aisément transformer en table modèle en la multipliant par un nombre d'observations N donné. Le test du modèle sous-saturé peut alors être effectué sur la table modèle pour déterminer si l'interaction que nous avons définie est statistiquement significative ou non. Évidemment, pour définir manuellement des interactions significatives dans le graphe, il est plutôt conseillé de faire l'inverse, c'est-à-dire de trouver une table modèle qui permet de rejeter le modèle sous-saturé pour ensuite définir les coefficients du facteur.

En définissant des facteurs pour chacune des facettes, nous définissons également la distribution de probabilités conjointes totale du graphe. En effet, pour obtenir cette distribution, il suffit de multiplier les Φ_i entre eux et de diviser le tout par la fonction de partition, comme nous l'avons fait à l'équation (4.3). Grâce à la distribution de probabilités conjointes totale, nous pouvons utiliser la méthode du rejet pour l'échantillonnage.

4. Pour cet exemple, nous utilisons les coefficients a, b, c et d, qui seraient plutôt notés c_{11}, c_{10}, c_{01} et c_{00} si nous avions directement utilisé l'équation (4.19).

En bref, la méthode du rejet consiste à échantillonner des états en respectant la distribution des probabilités du système, c'est-à-dire l'équation (4.3). La première étape est de générer aléatoirement un vecteur d'états à partir d'une loi uniforme. Par la suite, il faut décider si nous acceptons d'utiliser cet état ou non. À cette fin, nous calculons d'abord quelle est la probabilité associée à l'état en question à partir de l'équation (4.3). Ensuite, nous pigeons un nombre aléatoire entre 0 et 1 et, s'il est inférieur à la probabilité de l'état, nous acceptons l'état. Autrement, il est rejeté et la démarche reprend avec un nouveau vecteur d'états. C'est ce processus de sélection qui assure que les états pigés respectent la distribution des probabilités conjointes du système. [5]

En utilisant cette technique, nous générons alors des vecteurs $\boldsymbol{s}_i$ où chacune des entrées, s_{ij}, représente l'état de présence de l'entité j dans le vecteur $\boldsymbol{s}_i$. Nous identifions alors ces vecteurs comme des « sites » et si nous concaténons ces vecteurs pour former une matrice, nous obtenons une matrice de présence/absence. Cette matrice peut alors être injectée dans l'une des méthodes d'inférence pour inférer un complexe simplicial. Ce dernier peut alors être comparé au complexe simplicial associé à la projection du graphe de facteurs sur les variables aléatoires. Par contre, le processus de génération de données synthétiques étant stochastique, de la variabilité est introduite dans les données. Il est donc attendu que le complexe simplicial inféré diffère du complexe simplicial original dans une certaine mesure. Un résumé du processus global de création d'un graphe de facteurs ainsi que de données de présence/absence se trouve dans le bloc 8. [6]

5. Dans notre cas, nous n'avons pas implémenté directement cette méthode, car nous avons utilisé la règle de décision de Metropolis-Hastings, c'est-à-dire que la probabilité d'accepter l'état est 1 si son énergie est inférieure à l'état précédent ou égale à $e^{-(E_n - E_{n-1})}$ si l'énergie de l'état présent, E_n, est supérieure à l'énergie de l'état précédant E_{n-1} [48]. Cette variante respecte également la distribution conjointe des états du système.

6. Il existe quelques problèmes avec cette méthode qui peuvent altérer les résultats escomptés. Toutefois, des solutions sont proposées dans les sections 4.1.5 et 4.1.5. Ces solutions permettent aussi d'automatiser la recherche de facteurs des étapes 2 à 7 de l'algorithme 8.

Algorithme 8 Création d'un graphe de facteurs et génération de données de présence/absence

Sorties Graphe de facteurs ;
Distribution de probabilités du graphe de facteurs ;
Des données de présence/absence.

1: Générer une liste de facettes, soit déclarer une liste de facettes spécifique, utiliser le SCM pour extraire une liste de facettes à partir d'un réseau biparti ou générer une liste de facettes aléatoirement. La liste de facettes doit respecter les règles de la section 2.1.1.
2: Pour chaque facette, déterminer la forme de la fonction d'énergie en utilisant l'équation (4.18).
3: Pour chaque facette, construire une table de contingence de taille 2^d, où d est le nombre de variables dans la facette.
4: Pour chaque table, déterminer des entrées qui permettent de rejeter le modèle sous-saturé à d variables.
5: Transformer chacune des entrées en prenant le logarithme naturel.
6: Associer ces valeurs transformées au coefficient correspondant dans la fonction d'énergie.
7: Calculer le facteur résultant avec l'équation (4.9).
8: Une fois tous les facteurs obtenus, calculer la distribution des probabilités conjointes avec l'équation (4.3).
9: Utiliser la méthode du rejet pour générer des états de présence/absence à partir de cette distribution de probabilités.

4.1.3 Exemples à deux noeuds

Afin de tester l'algorithme de génération de données synthétiques et le comportement des méthodes d'inférence sur ces dernières, nous avons créé des graphes de facteurs à deux et trois variables aléatoires. Cela nous a permis de générer des données de présence/absence pour des paires et des triplets où les relations entre les variables étaient connues. À moins d'avis contraire, pour chaque cas étudié, nous avons généré dix échantillons de 1 000 matrices de présence/absence qui comportent $N = 100$ sites. Pour l'inférence, nous avons utilisé des tests asymptotiques avec $\alpha = 0.01$ sur les tables de contingence résultantes et H_0 correspond au modèle sous-saturé, peu importe la dimension.

Pour les paires, il existe deux familles, soit celle qui respecte l'indépendance et celle qui respecte la dépendance. Pour définir les facteurs, nous avons simplement utilisé la fonction d'énergie (4.10). Les coefficients ont été sélectionnés en inventant des tables modèles qui respectaient la relation désirée comme nous l'avons fait aux sections 3.3.2 et 3.3.3.

Indépendance de deux variables

Pour respecter l'indépendance entre deux variables, nous pouvons utiliser des coefficients a à d égaux entre eux dans l'équation (4.10). Ainsi, les probabilités associées à chacun des états de présence/absence sont 1/4. Les valeurs dans la table modèle sont alors $N/4$, ce qui respecte le modèle d'indépendance. Pour cette situation, nous avons tracé le taux d'erreur en fonction de α à la figure 4.2. Le taux d'erreur représente le nombre de tables dans l'échantillon où H_0

a été refusée alors qu'elle est vraie. On parle alors d'erreur de type 1 [1]. De plus, comme nous connaissons la table modèle utilisée pour définir le facteur, nous pouvons également mesurer la distance L_1 entre les tables de contingence obtenues à partir de l'échantillon et la table modèle. L'histogramme du nombre de tables de contingence à une certaine distance L_1 de la table modèle est représenté à la figure 4.3.

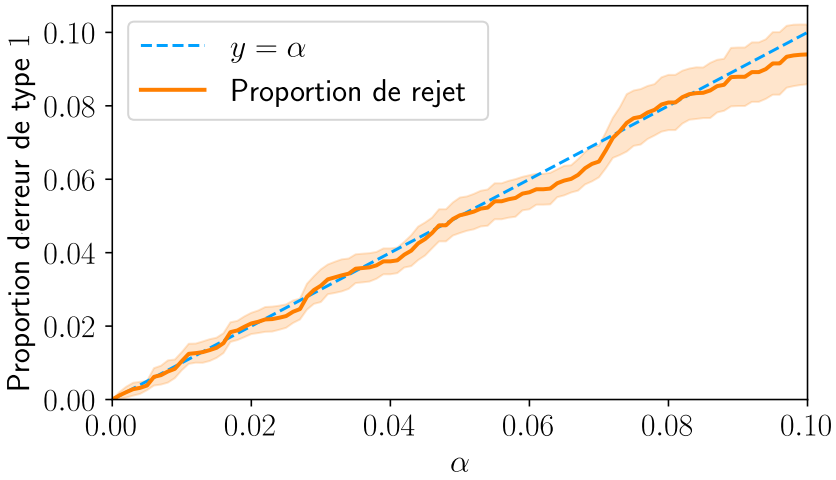

FIGURE 4.2 – Taux d'erreur de type 1 en fonction du paramètre α pour des simulations sur deux variables indépendantes. La courbe orange représente la moyenne sur les dix échantillons de 1 000 matrices de présence/absence et l'écart-type est représenté par la zone orangée.

Dans la figure 4.2, nous pouvons constater qualitativement que la proportion de faux positifs[7] suit le seuil α. Cette tendance était attendue, car α représente la probabilité de rejeter l'hypothèse nulle alors qu'elle est vraie. Nous remarquons également que l'écart-type grandit avec une augmentation de α. Ainsi, nous conseillons une valeur de α sous 5%, même s'il s'agit d'une valeur typique [1]. De cette manière, les interactions inférées sont plus certaines et le nombre de faux positifs diminue.

Dans la figure 4.3, il est possible d'observer qu'il est plus probable d'obtenir des tables qui s'éloignent d'une distance $L_1 = 12$ de la table originale. La valeur de L_1 la plus élevée dans l'échantillon est 42. De plus, comme nous l'avons indiqué dans l'analyse de la figure 3.5, $L_1^{\text{seuil}} = 22$ pour la table modèle. Ainsi, pour des valeurs de L_1 entre 24 et 42 inclusivement, le taux de succès n'est pas assurément 100%. Le processus a donc possiblement généré des données qui ne permettront pas d'obtenir la bonne conclusion, même si l'hypothèse nulle est

7. Rappelons qu'un **faux positif** signifie que la méthode a inféré une interaction alors qu'il n'y en avait pas en réalité. Il s'agit alors d'une erreur de type 1. Dans la même veine, un **faux négatif** signifie que la méthode n'a pas détecté d'interaction alors qu'il y en avait une. Il s'agit alors d'une erreur de type 2. Finalement, nous obtenons un **vrai positif** si la méthode a inféré une interaction présente dans le modèle et un **vrai négatif** si aucune interaction n'a été trouvée entre des variables alors que ces mêmes variables ne présentent pas l'interaction testée dans le modèle.

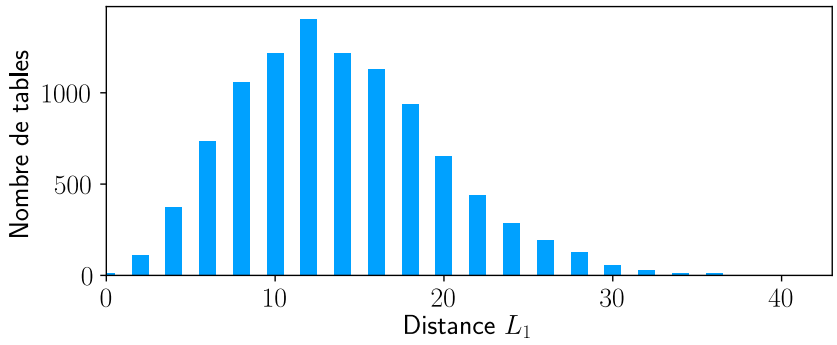

FIGURE 4.3 – Histogramme du nombre de tables obtenues ayant une distance L_1 par rapport à la table modèle 2×2 dont les entrées sont 25. L'histogramme est un assemblage des dix échantillons de 1 000 matrices de présences/absence transformées en tables de contingence.

vraie. C'est d'ailleurs ce que nous observons sur la figure 4.2, où le taux d'erreur de type 1 n'est pas nul pour la plage de α analysée.

Dépendance entre deux variables

Pour le modèle de dépendance entre deux variables, nous avons utilisé des probabilités de 48% sur la diagonale ($a = d = \ln(48)$ dans l'équation (4.10)) et 2% sur l'antidiagonale ($b = c = \ln(2)$ dans l'équation (4.10)). Nous avons fait ce choix, car si nous avions mis des probabilités nulles sur l'antidiagonale, le modèle génératif n'aurait jamais sélectionné les états $(0, 1)$ et $(1, 0)$. Nous aurions donc obtenu des tables parfaitement diagonales et la méthode d'inférence ne se serait jamais trompée. Or, malgré les probabilités spécifiées, la méthode d'inférence est arrivée à la bonne conclusion pour chaque matrice de présence/absence analysée. Pour comprendre ce résultat, nous avons donc construit l'histogramme des distances L_1 par rapport à la table modèle à la figure 4.4 et nous avons tracé la figure 4.5, qui représente le taux de succès de l'inférence en fonction de la distance L_1 comme nous le faisions dans la section 3.3.

Sur la figure 4.4, nous pouvons constater que la plage des valeurs de L_1 pour les tables obtenues dans l'échantillon s'étend de 0 à 42 inclusivement. En regardant sur la figure 4.5, nous pouvons voir que le taux de succès est de 100% sur toute cette plage. En fait, la valeur de L_1^{seuil} pour la table modèle est 58. Ainsi, dans ces dix échantillons, il était impossible que la méthode d'inférence se trompe, malgré la variabilité.

Évidemment, nous aurions pu sélectionner une table modèle qui respecte également le modèle de dépendance, mais dont la tolérance aux perturbations est inférieure. Cela aurait permis de générer des données où un taux d'erreur aurait été observé. D'ailleurs, pour cette situation, il n'est pas nécessaire de tracer une figure du taux d'erreur de type 1 en fonction de α, car nous savons que l'hypothèse nulle est fausse. Il serait plus informatif de tracer le taux d'erreur de type 2. Une telle erreur signifie que nous avons accepté l'hypothèse nulle alors qu'elle est

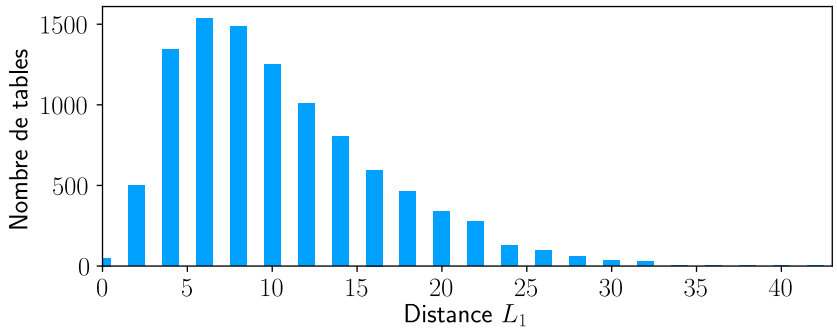

FIGURE 4.4 – Histogramme du nombre de tables obtenues ayant une distance L_1 par rapport à la table modèle 2×2 dont les entrées sur la diagonale sont 48 et celles sur l'antidiagonale sont 2. L'histogramme est un assemblage des dix échantillons de 1 000 matrices de présences/absence transformées en tables de contingence.

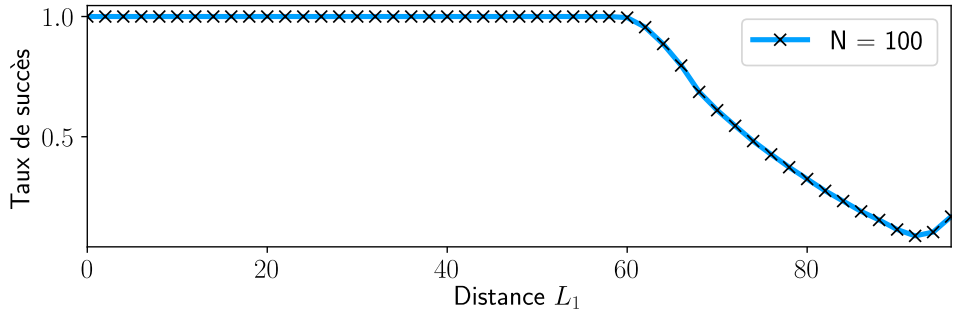

FIGURE 4.5 – Taux de succès pour retrouver la dépendance en fonction des perturbations de la table modèle 2×2 dont les entrées sur la diagonale sont 48 et celles sur l'antidiagonale sont 2.

fausse [1]. Cela correspond à un **faux négatif**, car la méthode n'a pas inféré d'interaction alors qu'il y en avait une. Dans notre cas, puisque le taux de succès est 100% pour tous les échantillons, la proportion d'erreurs de type 2 est nulle. [8]

4.1.4 Exemples à trois noeuds

Avec les exemples à trois variables, nous pouvons commencer à utiliser l'inférence pour détecter des interactions d'ordre supérieur. En effet, nous pouvons d'abord inférer la relation entre chaque paire pour ensuite vérifier s'il existe une interaction d'ordre supérieur entre les trois variables.

Indépendance complète de trois variables

Le premier cas traité est celui de l'indépendance complète entre trois variables. Nous avons choisi $N = 104$ et toutes les entrées de la table modèle sont égales à 13. Pour déterminer s'il

8. À la différence du taux d'erreur de type 1, déterminé par α que nous avons posé nous-mêmes, le taux d'erreur de type 2 est associé à une probabilité notée β [1].

existe une interaction d'ordre supérieur ou non, nous allons utiliser la méthode d'inférence par étape. Par contre, comme nous l'avons vu à la section 4.1.3, le taux de faux positifs équivaut à α lorsque nous testons le modèle d'indépendance pour deux variables. Ainsi, même dans cet exemple où les trois variables sont indépendantes, des liens seront inférés entre les paires. Dans les faits, en effectuant les tests pour les trois paires au sein des matrices de présence/absence, nous avons inféré l'existence de liens dans 284 cas. Cette valeur se rapproche de la valeur espérée pour $\alpha = 0.01$ sur les 30 000 paires testées (3 paires par matrice pour 10 000 matrices).

Malgré ces faux positifs, aucun réseau inféré après cette première étape d'inférence ne comportait un triangle vide. En effet, on retrouvait seulement 282 cas où un seul lien était présent entre les trois entités et 2 cas où deux paires étaient connectées parmi les trois entités. De ce fait, avec la méthode d'inférence par étape, aucune interaction à trois variables n'a été détectée. En contrepartie, la méthode d'inférence systématique a trouvé des interactions d'ordre supérieur alors qu'il n'y en a pas. Nous avons tracé la proportion d'erreurs de type 1 en fonction de α sur la figure 4.6) et l'histogramme des distances L_1 entre les tables obtenues et la table modèle sur la figure 4.7.

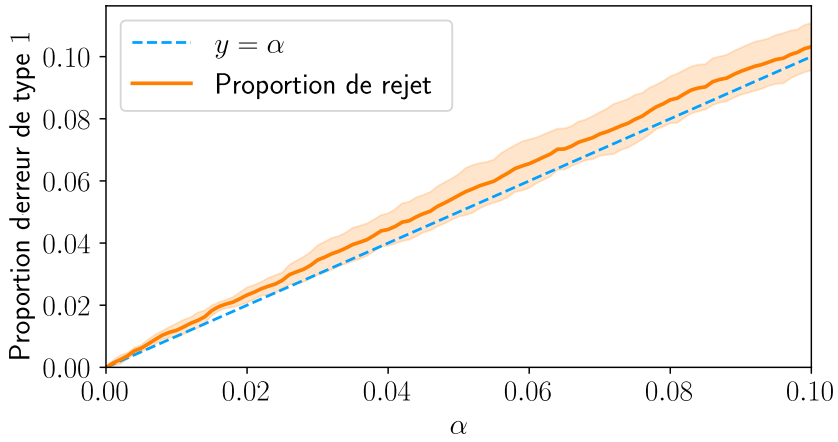

FIGURE 4.6 – Taux d'erreur de type 1 en fonction du paramètre α pour des simulations sur trois variables indépendantes en utilisant la méthode d'inférence systématique. Dans les tests effectués, H_0 correspond au modèle sous-saturé à trois variables. La courbe orange représente la moyenne sur les dix échantillons de 1 000 matrices de présence/absence et l'écart-type est représenté par la zone orangée.

À la différence de la figure 4.2, l'hypothèse nulle utilisée à la figure 4.6 correspond au modèle sous-saturé à trois variables. Cette figure permet de constater de nouveau que la proportion d'erreurs de type 1 suit qualitativement la valeur de α et que l'écart-type augmente également avec ce paramètre. Cela signifie également qu'à la différence de la méthode d'inférence par étape, la méthode systématique se trompera plus facilement pour l'inférence d'interactions d'ordre supérieur.

En ce qui concerne l'histogramme de la figure 4.7, nous pouvons constater que la valeur

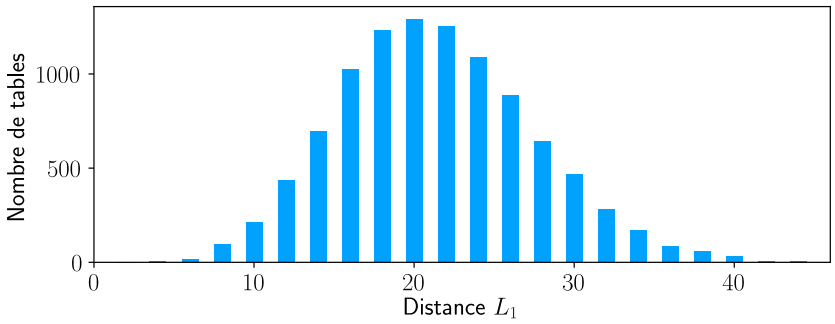

FIGURE 4.7 – Histogramme du nombre de tables obtenues ayant une distance L_1 par rapport à la table modèle $2 \times 2 \times 2$ dont les entrées sont 13. L'histogramme est un assemblage des dix échantillons de 1 000 matrices de présences/absence transformées en tables de contingence.

maximale de L_1 atteinte est 46. De plus, la valeur de L_1^{seuil} pour cette même table modèle, obtenue grâce à la figure 3.12, est 20 alors que $L_1^{\max} = 26$. Il est donc attendu que l'hypothèse nulle soit rejetée en effectuant le test sur certaines données. Notons que le fait que le processus génératif crée des données dont la table de contingence dépasse $L_1^{\max}$ n'est pas un problème, car ce dernier ne peut générer que des tables valides. Toutefois, rappelons que pour des valeurs plus grandes ou égales à $L_1^{\max}$, il est possible de générer des tables où l'amplitude du bruit est égale aux entrées dans la table modèle.

Triangle vide

Le deuxième cas correspond à celui où chaque paire est dépendante, mais où l'interaction d'ordre supérieur n'est pas présente. Autrement dit, nous avons affaire à un triangle vide. Nous avons d'ailleurs séparé ce cas en deux situations distinctes.

Pour la première situation, nous avons utilisé les mêmes probabilités que pour l'exemple de dépendance dans la section 4.1.3, c'est-à-dire 0.48 sur la diagonale et 0.02 sur l'antidiagonale. Par contre, nous avons généré un graphe de facteurs avec les facettes $[0, 1]$, $[0, 2]$ et $[1, 2]$, plutôt que de spécifier une fonction d'énergie comme celle de l'équation (4.19). L'inférence a donc déniché des liens entre chacune des paires pour les dix échantillons de 1000 matrices. Autrement dit, la première étape permettait d'identifier des triangles vides à chaque fois.

Toutefois, à l'étape du test du modèle sous-saturé à trois variables, 9 924 exemples créaient des cas problématiques pour l'algorithme itératif. En effet, avec un modèle de dépendance par paire aussi fort, la table de contingence avec les trois variables prend généralement la forme

	$B=0$	$B=1$
$A=0$	x	0
$A=1$	0	0

$C = 0$

	$B=0$	$B=1$
$A=0$	0	0
$A=1$	0	y

$C = 1$,

où $x > 0$ et $y > 0$. De nombreuses entrées dans les configurations suffisantes sont alors nulles et l'algorithme itératif retourne une table espérée identique à la table observée. Ainsi, les triangles vides restaient vides par défaut. Malgré tout, nous savons qu'il s'agit de la bonne conclusion, puisque nous pouvons expliquer exactement les interactions entre ces trois entités par des interactions dyadiques.

Puisque la situation précédente était trop extrême et qu'elle ne permettait pas de compléter le test du modèle sous-saturé dans 99.24% des cas, nous avons construit un exemple où la table modèle $2 \times 2 \times 2$ est

	$B=0$	$B=1$
$A=0$	155	15
$A=1$	15	15

$C = 0$

	$B=0$	$B=1$
$A=0$	15	15
$A=1$	15	155

$C = 1$

.

Pour cet exemple, la valeur-p associée au modèle sous-saturé est approximativement égale à 1.0, alors que les valeurs-p des autres modèles hiérarchiques non saturés sont toutes inférieures à 2×10^{-15}. Le seul modèle qui arrive à expliquer nos observations est donc bien celui où chaque paire est indépendante du niveau de la troisième variable (triangle vide). Grâce à cette table, aucun problème de convergence de l'algorithme itératif n'a été observé.

Ainsi, pour construire le graphe de facteurs, nous avons utilisé la facette $[1, 2, 3]$ et spécifié les coefficients dans l'équation (4.19). Ces derniers sont $a = h = \ln(155)$ et $\ln(15)$ pour tous les autres. Dans les dix échantillons de 1 000 matrices, la méthode d'inférence par étape arrive à inférer un lien entre chacune des paires. Cela signifie que nous avons inféré un triangle vide pour toutes les matrices de présence/absence générées. Ainsi, nous pouvons vérifier si ces triangles restent vides ou non en passant à la deuxième étape d'inférence. Or, ce cas-ci revient à tester tous les triplets dans chacune des matrices de présence/absence. De ce fait, il est attendu que la méthode d'inférence se trompera en proportion α. Le taux d'erreur de type 1 a d'ailleurs été tracé à la figure 4.8, où nous pouvons observer que le taux d'erreur de type 1 suit α comme il se doit.

Dépendances conditionnelles à trois variables

Parmi les exemples où le modèle sous-saturé est rejeté, nous avons aussi le cas où les dépendances sont conditionnelles. En effet, avec la table modèle

	$B=0$	$B=1$
$A=0$	41	35
$A=1$	41	59

$C = 0$

	$B=0$	$B=1$
$A=0$	76	59
$A=1$	39	50

$C = 1$

,

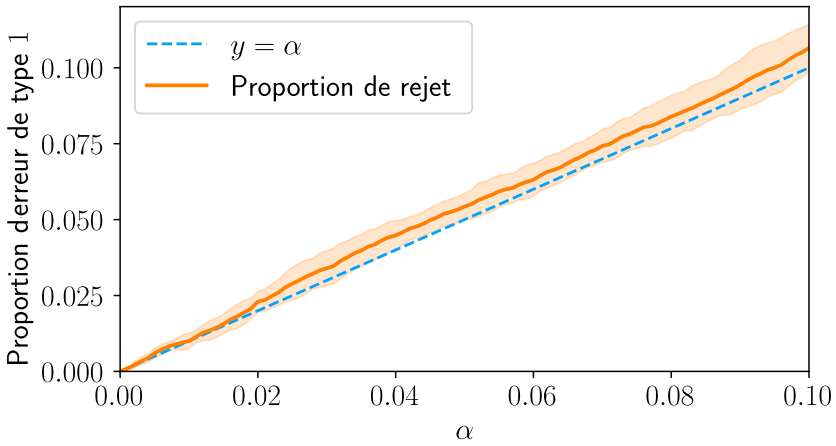

FIGURE 4.8 – Taux d'erreur de type 1 en fonction du paramètre α pour des simulations sur trois variables dépendantes en paires (triangle vide) en utilisant la méthode d'inférence systématique. Dans les tests effectués, H_0 correspond au modèle sous-saturé à trois variables. La courbe orange représente la moyenne sur les dix échantillons de 1 000 matrices de présence/absence et l'écart-type est représenté par la zone orangée.

la valeur-p du modèle hiérarchique où u_{ik}^{AC} et u_j^B sont présents est 0.0680, celle du modèle u_{ik}^{AC} et u_{jk}^{BC} donne 0.0439, celle du modèle u_{ik}^{AB} et u_{ik}^{AC} donne 0.8772 et celle du modèle sous-saturé est 0.9603.

Sur le nombre total de matrices générées, l'inférence ne détecte aucun lien entre les trois paires dans 950 cas, tandis qu'elle en détecte un dans 4 556 cas, deux dans 4 195 cas et trois dans 299 cas. Considérant les valeurs-p des modèles de dépendance conditionnelle, il semble donc raisonnable qu'une majorité des cas ne présente qu'un seul ou deux liens. De ce fait, avec la méthode d'inférence par étape, seuls 299 cas peuvent être testés pour la présence d'une interaction d'ordre supérieur à trois variables. Avant même d'effectuer ce test, nous avons donc inféré l'absence d'une telle interaction dans 9 701 cas. En ce qui concerne les 299 triangles vides qui doivent rester vides, la proportion de faux positifs a été tracée à la figure 4.9.

Si nous utilisons plutôt la méthode d'inférence systématique pour dénicher les interactions d'ordre supérieur, nous obtenons encore une proportion de faux positifs qui suit la valeur de α. Cette courbe a été tracée à la figure 4.10.

Même si les proportions sont semblables aux figures 4.9 et 4.10, les proportions dans la première sont calculées sur les 299 triangles vides tandis qu'elles sont calculées sur dix échantillons de 1 000 triplets dans la deuxième. En prenant pour acquis que la proportion de faux positifs est égale à α, cela signifie le nombre de faux positifs avec $\alpha = 0.01$ est 100 lorsque nous utilisons la méthode d'inférence systématique. En contrepartie, avec la méthode d'inférence par étape, nous obtenons 3 faux positifs au total pour $\alpha = 0.01$. Cela augmente donc le nombre de vrais négatifs à 9 997 pour les interactions d'ordre supérieur avec cette méthode. De ce fait, utiliser

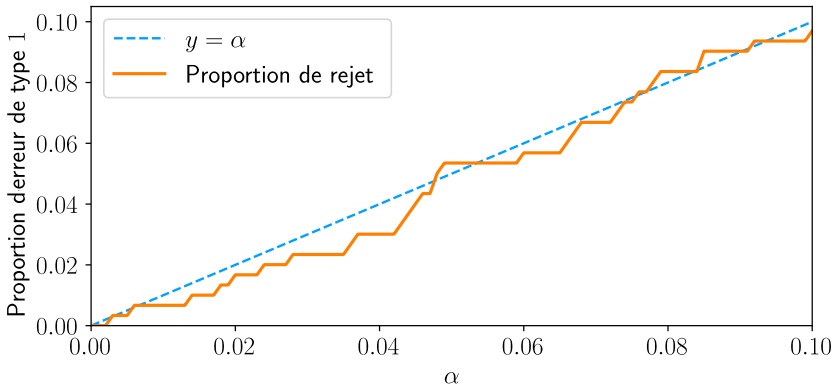

FIGURE 4.9 – Taux d'erreur de type 1 en fonction du paramètre α pour les en évaluant l'interaction d'ordre supérieur sur les 299 triangles vides trouvés dans l'exemple où les trois variables présentent des dépendances conditionnelles.

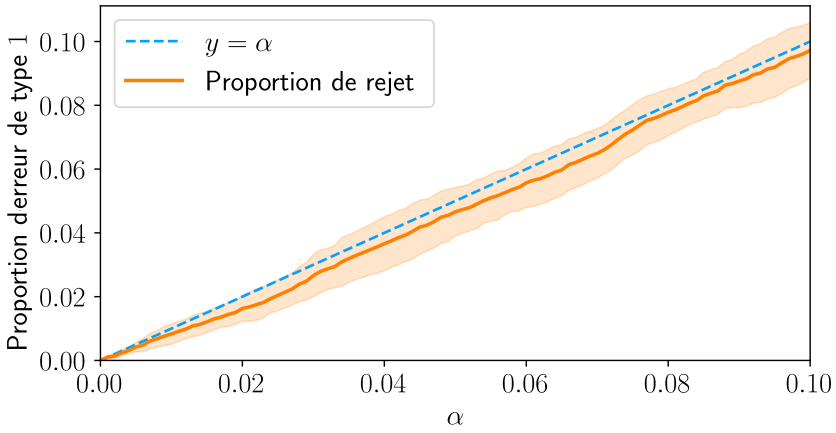

FIGURE 4.10 – Taux d'erreur de type 1 en fonction du paramètre α pour des simulations sur trois variables conditionnellement dépendantes en utilisant la méthode d'inférence systématique. Dans les tests effectués, H_0 correspond au modèle sous-saturé à trois variables. La courbe orange représente la moyenne sur les dix échantillons de 1 000 matrices de présence/absence et l'écart-type est représenté par la zone orangée.

la méthode d'inférence par étape est avantageux pour réduire le nombre de faux positifs.

Dépendance d'ordre supérieur à trois variables

Le dernier cas étudié correspond à celui où les trois variables présentent une dépendance d'ordre supérieur (2-simplexe). Pour ce dernier, nous allons tester les deux méthodes d'inférence, soit celle par étape et celle systématique. La première table modèle utilisée est

	$B=0$	$B=1$
$A=0$	51	25
$A=1$	19	64

$C=0$

	$B=0$	$B=1$
$A=0$	63	85
$A=1$	54	39

$C=1$

.

Cette dernière produit une valeur-p de 5.247×10^{-9} pour le modèle sous-saturé. En testant les autres modèles, la valeur-p la plus élevée qui a été obtenue est 3.695×10^{-8}. Tous les modèles sont donc rejetés, ce qui nous permet de conclure qu'il existe une interaction d'ordre supérieur entre les trois variables. Sur les 10 000 matrices générées, l'inférence ne détecte aucun lien dans 3 551 cas, tandis qu'elle en détecte un dans 4 781 cas, deux dans 1 557 cas et trois dans 111 cas.

Cette fois-ci, nous avons tracé la proportion d'erreurs de type 2, c'est-à-dire la probabilité d'accepter l'hypothèse nulle alors qu'elle est fausse, à la figure 4.11. Pour cette figure, c'est la méthode d'inférence systématique qui est utilisée. Nous pouvons alors constater qu'à partir de $\alpha = 0.015$, la méthode statistique ne se trompe jamais sur la conclusion. Si nous utilisons la méthode d'inférence par étape, nous ne pouvons tester que les 111 cas où un triangle vide est d'abord inféré à la première étape. Parmi ceux-ci, la valeur-p la plus élevée pour le test du modèle sous-saturé est 1.063×10^{-4}. Ainsi, pour chaque valeur de α supérieure à 1.063×10^{-4}, le taux d'erreur de type 2 est nul.

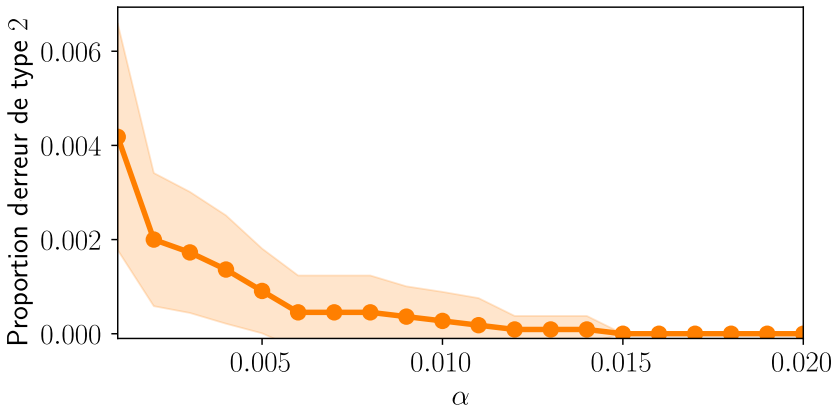

FIGURE 4.11 – Taux d'erreur de type 2 en fonction du paramètre α pour des simulations sur trois variables formant une interaction d'ordre supérieur, en utilisant la méthode d'inférence systématique. Dans les tests effectués, H_0 correspond au modèle sous-saturé à trois variables. La courbe orange représente la moyenne sur les dix échantillons de 1000 matrices de présence/absence et l'écart-type est représenté par la zone orangée.

À l'inverse des cas présentés aux figures 4.9 et 4.10, le fait d'utiliser la méthode d'inférence par étape nous fait manquer des interactions d'ordre supérieur dans l'analyse. Il s'agit d'une conséquence directe du fait qu'un triangle doit d'abord être inféré à la première étape. Pour-

tant, avec cet exemple, nous voyons qu'il ne s'agit pas d'une condition nécessaire pour qu'une interaction d'ordre supérieur soit présente.

Puisque dans l'exemple précédent il n'y avait que 111 cas où trois liens étaient inférés à la première étape, nous avons tenté de trouver des probabilités qui permettaient de générer une majorité de cas à trois liens. Nous nous sommes arrêtés sur la table

	$B=0$	$B=1$
$A=0$	77	12
$A=1$	15	87

$C=0$

	$B=0$	$B=1$
$A=0$	68	67
$A=1$	65	9

$C=1$

,

dont la valeur-p associée au test du modèle sous-saturé est 2.3165×10^{-27}. Il s'agit d'ailleurs de la valeur-p la plus élevée parmi toutes celles obtenues en testant tous les modèles hiérarchiques à trois variables. Autrement dit, tous les modèles non saturés sont fortement rejetés.

Sur les 10 000 matrices générées, on compte 137 cas où la première étape ne détecte aucun lien entre les trois paires, 930 cas où un seul lien est présent, 3 170 cas où deux liens sont présents et 5 763 cas avec trois liens. En utilisant la méthode d'inférence systématique, la valeur-p la plus élevée obtenue lors du test du modèle sous-saturé à trois variables est 9.5143×10^{-16}. Ainsi, l'hypothèse nulle est rejetée dans 100% des cas pour $\alpha > 9.5143 \times 10^{-16}$.

Encore une fois, par contre, si nous ne testons que le triangle vide, cela signifie que nous ne pouvons tester que 5 763 triplets. La méthode d'inférence par étape manque donc les 4 237 autres interactions d'ordre supérieur, puisqu'elle ne peut pas les tester. Par contre, sur tous les triplets testés, elle arrive à inférer l'interaction d'ordre supérieur.

Ces deux derniers exemples montrent alors qu'il est possible d'inférer une interaction d'ordre supérieur en testant un triangle vide. Cet élément est important, car la technique d'inférence par étape est basée sur l'hypothèse qu'un cycle composé de k simplexes ayant une dimension $k-2$ peut être signe d'une interaction d'ordre supérieur. Or les deux exemples montrent aussi qu'il n'est pas nécessaire que cette condition soit respectée pour trouver des interactions d'ordre supérieur.

Malgré tout, il serait intéressant de trouver la forme des tables qui permettent systématiquement d'inférer la présence de trois liens à la première étape de la méthode et une interaction d'ordre supérieur à la seconde. Cela pourrait permettre de différencier les cas où l'approche systématique des triplets doit être utilisée et celle où la construction du complexe simplicial étape par étape est adéquate.

4.1.5 Exemples à plus de trois noeuds

Pour générer un complexe simplicial possédant plusieurs facettes, il serait laborieux de spécifier les probabilités comme nous l'avons fait dans les exemples précédents. Nous avons donc automatisé la recherche de probabilités qui permettent le rejet de l'hypothèse nulle sur la facette spécifiée. L'algorithme responsable de cette recherche est présenté dans le bloc 9 ci-bas. C'est en appliquant cet algorithme sur chacune des facettes que nous obtenons un graphe de facteurs pour la liste de facettes spécifiée.

Algorithme 9 Méthode de sélection de probabilités de chaque facteur

Entrées	Facette de k noeuds ; Nombre d'observations N ; Seuil α.
Sortie	Liste de coefficients pour l'équation (4.18) qui permet d'inférer un $(k-1)$-simplexe.

1: Générer une liste de nombres aléatoires réels dans l'intervalle $(0, 1]$ de longueur 2^k, où k est le nombre de noeuds.
2: Normaliser la liste de nombres aléatoires pour obtenir le vecteur de probabilités $\mathbf{p}$.
3: Générer une réalisation d'une loi multinomiale à N observations et probabilités $\mathbf{p}$.
4: Calculer la valeur-p du modèle sous-saturé sur la réalisation de l'étape précédente.
5: Si la valeur-p est inférieure à α, les logarithmes des entrées dans la réalisation correspondent aux coefficients recherchés pour la facette en entrée.
6: (Optionnel) Pour une facette à 3 noeuds, appliquer la procédure 10 (voir plus bas) pour s'assurer que les coefficients impliquent la présence des interactions entre chaque paire.
7: Sinon, recommencer du début.

Même si nous savons que la table modèle trouvée pour le triplet $[1, 2, 3]$ permet de rejeter le modèle sous-saturé, cela ne garantit pas qu'il s'agit d'un 2-simplexe. En effet, comme nous l'avons constaté dans les deux derniers exemples de la dernière section, il est possible d'identifier des interactions d'ordre supérieur véridiques sur des triplets qui ne forment pas un triangle vide en utilisant la méthode d'inférence systématique. Pour savoir si une interaction d'ordre supérieur est réellement un 2-simplexe dans notre graphe de facteur, il faut d'abord trouver si chaque 1-simplexe au sein du 2-simplexe existe dans notre modèle. Pour obtenir cette information, nous pouvons utiliser la distribution des probabilités conjointes (4.3) qui est entièrement déterminée lorsque les facteurs sur chaque facette sont définis. En effet, en sommant sur les états de certaines variables dans le graphe de facteurs, nous obtenons la distribution des probabilités conjointes pour le reste des variables. Par exemple, si $\mathbf{D}$ contient plusieurs variables, dont X_1 et X_2, nous avons que

$$P(X_1, X_2) = \sum_{\mathbf{D}\setminus\{X_1,X_2\}} P(\mathbf{D}), \tag{4.28}$$

où la somme est sur l'ensemble des états des variables dans $\mathbf{D}$ sauf X_1 et X_2. Ainsi, nous pouvons obtenir les tables de probabilités pour un couple de variables et les transformer en table

modèle par une multiplication par N. Nous pouvons alors tester le modèle d'indépendance entre toutes les paires de variables et déterminer si, en trouvant le facteur pour la facette $[1, 2, 3]$, nous avons également spécifié des relations de dépendance entre les paires.

Afin de tester l'efficacité de l'algorithme 9, nous avons fait la recherche de 1 000 graphes de facteurs pour lesquels la seule facette spécifiée est $[1, 2, 3]$ avec $N = 500$ et $\alpha = 0.01$. Dans l'échantillon, nous avons compté combien de graphes ne comportaient pas de 1-simplexe et combien en comportaient un, deux ou trois. Ces valeurs sont respectivement 56, 269, 455 et 220.[9] Nous pouvons alors constater que la majorité des graphes de facteurs générés ne permettent pas l'obtention de liens entre chaque paire et que la facette $[1, 2, 3]$ est un 2-simplexe que dans 22% des cas. Considérant cela, nous avons ajouté une option à l'algorithme 9 afin de pouvoir générer des tables modèles de dimension $2 \times 2 \times 2$ où les interactions par paires existent également.

La modification est simple : lorsqu'une table modèle $2 \times 2 \times 2$ est trouvée à l'étape 5 de l'algorithme 9, nous pouvons sommer sur chacun de ses axes pour trouver les tables 2×2 de chaque paire. Nous testons alors le modèle d'indépendance sur chacune de ces tables et, si ce modèle est rejeté pour chaque paire, nous acceptons la table $2 \times 2 \times 2$. Sinon, nous répétons la procédure 9 du début. Ces étapes sont résumées dans l'algorithme 10.

En utilisant cette modification pour générer un échantillon de 1 000 graphes de facteurs avec la facette $[1, 2, 3]$, $N = 500$ et $\alpha = 0.01$, nous obtenons 1 000 cas où il est possible d'inférer trois 1-simplexes.[10]

Algorithme 10 Méthode pour assurer qu'une facette à trois noeuds est un 2-simplexe

Entrées	Table $2 \times 2 \times 2$ respectant l'étape 5 de l'algorithme 9 ; Seuil α.
Sortie	Liste de coefficients de l'équation (4.19) pour la facette d'entrée qui assurent qu'il s'agit d'un 2-simplexe.

1: Sommer sur chacun des axes de la table $2 \times 2 \times 2$ pour obtenir les trois tables 2×2 associées aux paires de variables.
2: Calculer la valeur-p de la statistique du test d'indépendance pour chacune des tables.
3: Si la valeur-p est inférieure à α pour toutes les tables 2×2, accepter les coefficients trouvés par l'algorithme 9.
4: Autrement, recommencer du début la procédure de l'algorithme 9.

Pour un graphe de facteurs comportant plus d'une facette, le fait que nous ayons accès à la

9. La valeur-p moyenne pour l'ensemble des tables modèles 2×2 est 0.118 tandis que la valeur-p moyenne pour les tables $2 \times 2 \times 2$ est 5.90×10^{-4}. La valeur-p maximale obtenue parmi les 2-simplexes est 0.00994, ce qui signifie que toutes les tables modèles trouvées pour la facette $[1, 2, 3]$ permettent l'inférence d'une interaction d'ordre supérieur.

10. La valeur-p moyenne pour les 1-simplexes est 6.93×10^{-4} tandis que celle pour le 2-simplexe est 7.04×10^{-4}. La valeur-p maximale obtenue parmi les 2-simplexes est 0.00995.

distribution de probabilités conjointes nous permet d'identifier quelles sont les interactions significatives et celles qui ne le sont pas même si elles n'ont pas été directement spécifiées dans la liste de facettes. Par exemple, si la liste de facettes est $[[1,2,3],[4,5]]$ et que nous utilisons la méthode de recherche permettant d'obtenir des 2-simplexes, nous nous attendons à retrouver les 1-simplexes $[1,2]$, $[1,3]$, $[2,3]$ et $[4,5]$.

Qu'en est-il alors des liens non spécifiés comme $[1,5]$ ou $[3,4]$? En principe, comme ils ne sont pas spécifiés, nous nous attendons à ce que ces paires de variables soient indépendantes. Par contre, comme la recherche des facteurs pour les facettes spécifiées est aléatoire, il peut arriver qu'il y ait des **interactions induites** entre des variables aléatoires. Autrement dit, cela signifie qu'en calculant la table des probabilités conjointes pour une paire de variables, la table modèle (pour un N et un α donné) permet de considérer que les deux variables sont dépendantes même si nous n'avons pas spécifié cette facette dans la liste. La liste de facettes qui comprend également les interactions induites est appelée **liste de facettes effective**.

À partir de la distribution de probabilités, nous pouvons alors effectuer le test du modèle sous-saturé pour toutes les paires et tous les triplets de variables aléatoires du graphe de facteurs. Les paires et les triplets qui rejettent l'hypothèse nulle sont alors considérés comme les interactions que l'inférence doit retrouver, c'est-à-dire les vrais positifs, même s'ils n'ont pas été directement spécifiés. Or, si l'inférence indique qu'il y a une interaction pour une paire ou un triplet où le modèle sous-saturé n'a pas été rejeté, nous considérons qu'elle a trouvé un faux positif.

Exemples à deux facettes

Afin de tester les performances des méthodes d'inférence, nous avons généré deux graphes de facteurs en utilisant la recherche de facteurs qui assure qu'une interaction d'ordre supérieur est un 2-simplexe lorsque spécifié dans la liste de facettes. Les paramètres spécifiés ainsi que les listes de facettes sont $N = 100$, $\alpha = 0.01$, $[[1,2,3],[4,5]]$ et $[[1,2,3],[3,4]]$. Les valeurs-p associées à chaque paire et chaque triplet pour les deux graphes de facteurs sont indiquées dans les tableaux 4.1 et 4.2. [11] Les complexes simpliciaux issus des listes de facettes sont représentés aux figures 4.12 (a) et (b).

Nous avons également calculé les valeurs-p de ces interactions pour différentes valeurs de N. À cette fin, nous utilisions le même graphe de facteurs, mais nous multiplions les tables de probabilités de chacune des interactions par N avant de tester le modèle sous-saturé.

Sur les deux tables nous pouvons remarquer que la valeur-p des interactions spécifiées dans la liste de facettes décroît avec une augmentation de N. En un sens, ces interactions deviennent donc plus certaines à mesure que nous avons plus d'observations, ce qui est attendu. Au tableau

11. Pour le tableau 4.1, les interactions non significatives ne sont pas affichées, car leur valeur-p demeurait approximativement 1.0 peu importe N.

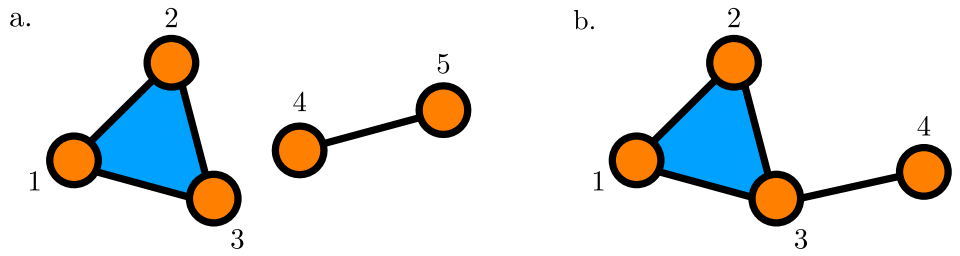

FIGURE 4.12 – a) Complexe simplicial comportant les facettes $[1, 2, 3]$ et $[4, 5]$. b) Complexe simplicial comportant les facettes $[1, 2, 3]$ et $[3, 4]$.

4.2, nous pouvons également observer l'apparition d'interactions induites pour les paires $[1, 4]$ et $[2, 4]$. En effet, ces dernières n'ont pas été spécifiées dans la liste de facettes et, à $N = 100$ observations, leur valeur-p ne permettait pas de rejeter l'hypothèse nulle.

Par contre, en augmentant le nombre d'observations, nous avons de plus en plus de preuves que ces paires sont dépendantes, ce qui permet éventuellement un rejet du modèle sous-saturé. Cette dépendance n'est toutefois par surprenante, car le noeud 4 fait partie de la facette $[3, 4]$, qui a été spécifiée au départ. Il est donc possible que le noeud 4 agisse comme le noeud 3 (ou comme son inverse) faisant en sorte qu'aux yeux des noeuds 1 et 2, le noeud 4 est indiscernable du noeud 3.

Cela transparaît également dans la diminution de la valeur-p pour l'interaction $[1, 2, 4]$, qui décroît également avec une augmentation du nombre d'observations. Pour $N = 1000$ la valeur-p de ce triplet se situe à une distance de 7.80×10^{-3} du seuil α. En générant des matrices de présence/absence dans cette situation, il y a donc fort à parier que le 2-simplexe $[1, 2, 4]$ sera inféré, même s'il ne s'agit pas d'un vrai positif à proprement parler.

Interaction	$N = 100$ Valeur-p	$N = 500$ Valeur-p	$N = 1000$ Valeur-p
$[1, 2]$	1.77×10^{-3}	2.77×10^{-12}	4.90×10^{-23}
$[1, 3]$	1.87×10^{-5}	1.07×10^{-21}	9.95×10^{-42}
$[2, 3]$	8.35×10^{-13}	1.27×10^{-57}	2.36×10^{-113}
$[4, 5]$	1.47×10^{-4}	2.11×10^{-17}	3.41×10^{-33}
$[1, 2, 3]$	2.03×10^{-5}	1.59×10^{-21}	2.17×10^{-41}

TABLE 4.1 – Valeurs-p pour la table modèle de chacune des interactions définies dans le graphe de facteurs avec $N = 100$, $\alpha = 0.01$ et la liste de facettes $[1, 2, 3]$, $[4, 5]$ représentée à la figure 4.12 (a). Les valeurs-p sont également calculées en conservant les probabilités du graphe de facteurs, mais en augmentant N. Notez que les valeurs-p pour tous les autres 1-simplexes et 2-simplexes non spécifiés dans la table étaient approximativement 1.0 pour toutes les valeurs de N.

Pour chaque valeur de N et pour chaque graphe de facteurs, nous avons d'ailleurs généré 1 000

Interaction	$N = 100$ Valeur-p	$N = 500$ Valeur-p	$N = 1000$ Valeur-p
$[1, 2]$	2.48×10^{-3}	1.34×10^{-11}	1.11×10^{-21}
$[1, 3]$	5.24×10^{-3}	4.32×10^{-10}	1.07×10^{-18}
$[2, 3]$	1.96×10^{-4}	8.30×10^{-17}	5.19×10^{-32}
$[3, 4]$	3.30×10^{-3}	5.03×10^{-11}	1.52×10^{-20}
$[1, 4]$	4.12×10^{-1}	6.66×10^{-2}	9.50×10^{-3}
$[2, 4]$	2.74×10^{-1}	1.14×10^{-2}	5.04×10^{-4}
$[1, 2, 3]$	2.81×10^{-3}	2.37×10^{-11}	3.42×10^{-21}
$[1, 2, 4]$	4.54×10^{-1}	9.39×10^{-2}	1.78×10^{-2}
$[1, 3, 4]$	9.99×10^{-1}	9.99×10^{-1}	9.99×10^{-1}
$[2, 3, 4]$	9.99×10^{-1}	9.99×10^{-1}	9.99×10^{-1}

TABLE 4.2 – Valeurs-p pour la table modèle de chacune des interactions possibles dans le graphe de facteurs avec $N = 100$, $\alpha = 0.01$ et la liste de facettes $[1, 2, 3]$, $[3, 4]$ représentée à la figure 4.12 (b). Les valeurs-p sont également calculées en conservant les probabilités du graphe de facteurs, mais en augmentant N.

matrices de présence/absence. Le nombre de vrais positifs et de faux positifs inférés dans ces échantillons sont rapportés dans les tableaux 4.3 et 4.4. Nous avons également spécifié les valeurs-p maximales obtenues pour les vrais et les faux positifs. Les trois rangées du milieu font référence à une recherche de 2-simplexes avec la méthode d'inférence par étape (donc seuls les triangles vides ont été testés), tandis que les trois rangées du bas font référence à la méthode d'inférence systématique (hyperliens).

En considérant le tableau 4.1, nous devrions retrouver 4 000 vrais positifs pour les 1-simplexes et 1000 vrais positifs pour les 2-simplexes dans le tableau 4.3. Nous pouvons d'ailleurs constater qu'avec une augmentation du nombre d'observations, les méthodes d'inférence convergent vers ces valeurs. Les valeurs-p de ces vrais positifs se rapprochent de plus en plus de zéro, ce qui était attendu puisqu'il s'agit de la même tendance pour leur table modèle. Les valeurs-p des faux positifs se situent entre α et 10^{-4} pour les 1-simplexes et entre α et 10^{-5} pour les 2-simplexes.

Avec ces données, la valeur-p minimale pour les faux positifs ne semble pas présenter de décroissance avec N. Ainsi, notre « pouvoir discriminatoire » entre les vrais positifs et les faux positifs augmente avec la valeur de N. En effet, la valeur-p maximale pour les vrais positifs s'éloigne de plus en plus de la valeur-p minimale des faux négatifs avec une augmentation de N. Nous pourrions donc sélectionner une valeur de α entre la valeur-p maximale des vrais positifs et la valeur-p minimale des faux négatifs pour retrouver tous les vrais positifs sans obtenir de faux négatifs.

Le nombre de faux positifs est d'ailleurs relié à α. En effet, dans la liste de facettes spécifiée, il est possible de tester $\binom{5}{2} = 10$ paires. Parmi celles-ci, on en compte quatre qui sont des vrais positifs. Le nombre attendu de faux positifs est alors $(10 - 4) \times \alpha \times 1000 = 60$. En ce qui

	N	# Vrais positifs	# Faux positifs	Valeur-p maximale (VP)	Valeur-p minimale (VP)	Valeur-p maximal (FP)	Valeur-p minimale (FP)
1-simplexe	100	3617	76	8.99×10^{-1}	2.50×10^{-19}	9.93×10^{-3}	2.07×10^{-4}
1-simplexe	500	4000	68	3.39×10^{-6}	7.58×10^{-73}	9.94×10^{-3}	1.54×10^{-4}
1-simplexe	1000	4000	52	2.58×10^{-11}	2.76×10^{-133}	9.58×10^{-3}	1.29×10^{-4}
2-simplexe	100	617	0	1.00×10^{0}	6.52×10^{-45}	-	-
2-simplexe	500	1000	0	1.44×10^{-9}	2.60×10^{-50}	-	-
2-simplexe	1000	1000	0	8.01×10^{-22}	4.81×10^{-72}	-	-
Hyperlien	100	882	84	1.00×10^{0}	6.52×10^{-45}	9.76×10^{-3}	1.88×10^{-4}
Hyperlien	500	1000	93	1.44×10^{-9}	2.59×10^{-50}	9.99×10^{-3}	2.84×10^{-5}
Hyperlien	1000	1000	87	8.01×10^{-22}	4.81×10^{-72}	9.98×10^{-3}	2.47×10^{-4}

TABLE 4.3 – Nombre de vrais positifs et de faux positifs inférés dans un échantillon de 1000 matrices de présence/absence générées par un graphe de facteurs dont les interactions sont identifiées dans le tableau 4.1. Les valeurs-p maximales obtenues pour les vrais et les faux positifs sont également indiquées. Pour les rangées 4 à 6, la méthode d'inférence par étape a été utilisée, tandis que pour les rangées 7 à 9, la méthode d'inférence systématique a été utilisée.

concerne les 2-simplexes, le nombre de faux positifs attendu est $\left[\binom{5}{3} - 1\right] \times \alpha \times 1000 = 90$. Notons que pour les rangées 4 à 6, le nombre de faux positifs est nul, car seuls les triangles vides peuvent être testés.

Pour $N = 500$ et $N = 1\,000$ le triangle formé de $[1, 2]$, $[2, 3]$ et $[1, 3]$ est testé dans tous les cas de l'échantillon, car ces 1-simplexes sont retrouvées dans 100% des cas à l'étape précédente. Rappelons toutefois que nous avons utilisé la méthode de recherche de facteurs où ce phénomène doit être respecté. Si nous avions utilisé l'algorithme 9 sans la modification 10, nous aurions pu tomber sur un graphe de facteurs où l'interaction $[1, 2, 3]$ n'est pas un 2-simplexe et la méthode d'inférence par étape aurait eu de la difficulté à l'identifier.

En ce qui concerne le tableau 4.4, nous devrions retrouver 4 000 1-simplexes pour $N = 100$ et $N = 500$ tandis que nous devrions en retrouver 6 000 pour $N = 1\,000$. En ce qui concerne les 2-simplexes, nous devrions en retrouver 1000 peu importe la valeur de N. Au tableau, nous observons en effet une augmentation du nombre de vrais positifs selon N. Or, le nombre de faux positifs ne semble pas suivre la tendance dictée par le paramètre α. En effet, pour les 1-simplexes, nous devrions retrouver environ 20 faux positifs, tandis que nous devrions en retrouver 30 pour les 2-simplexes. Toutefois, le nombre réel de faux positifs dépasse toujours ces estimations (excepté pour la rangée 5).

Ce comportement était toutefois attendu en raison des interactions induites. En effet, contrairement à l'exemple du tableau 4.1, certaines interactions non spécifiées possèdent un valeur-p qui se situent entre 1.0 et 0.01. Ces dernières se rapprochent d'ailleurs de α à mesure que N augmente. Puisque l'échantillonnage d'un graphe de facteurs est un processus stochastique, il faut alors s'attendre à de la variabilité qui peut faire pencher une interaction d'un côté ou de l'autre du seuil critique α.

	N	# Vrais positifs	# Faux positifs	Valeur-p maximale (VP)	Valeur-p minimale (VP)	Valeur-p maximal (FP)	Valeur-p minimale (FP)
1-simplexe	100	2786	105	9.93×10^{-1}	1.71×10^{-12}	9.46×10^{-3}	3.04×10^{-5}
1-simplexe	500	4000	648	7.97×10^{-4}	1.80×10^{-29}	9.90×10^{-3}	8.21×10^{-9}
1-simplexe	1000	5327	0	9.96×10^{-0}	7.59×10^{-48}	-	-
2-simplexe	100	251	0	9.16×10^{-1}	1.05×10^{-8}	-	-
2-simplexe	500	1000	24	2.76×10^{-4}	1.33×10^{-23}	9.94×10^{-3}	9.25×10^{-5}
2-simplexe	1000	1000	173	2.67×10^{-10}	7.27×10^{-35}	9.99×10^{-3}	3.77×10^{-8}
Hyperlien	100	653	70	9.94×10^{-1}	9.61×10^{-10}	9.94×10^{-3}	2.90×10^{-4}
Hyperlien	500	1000	197	2.76×10^{-4}	1.33×10^{-23}	9.97×10^{-3}	1.44×10^{-06}
Hyperlien	1000	1000	376	2.67×10^{-10}	7.27×10^{-35}	9.98×10^{-3}	3.77×10^{-8}

TABLE 4.4 – Nombre de vrais positifs et de faux positifs inférés dans un échantillon de 1000 matrices de présence/absence générées par un graphe de facteurs dont les interactions sont identifiées dans le tableau 4.2. Les valeurs-p maximales obtenues pour les vrais et les faux positifs sont également indiquées. Pour les rangées 4 à 6, la méthode d'inférence par étape a été utilisée, tandis que pour les rangées 7 à 9, la méthode d'inférence systématique a été utilisée.

Ce phénomène est encore plus apparent pour $N = 500$, où les valeurs-p des interactions non spécifiées se situent plus près de α que pour $N = 100$. Pour $N = 1\,000$, l'identification de faux positifs pour les 1-simplexes est impossible, car toutes les paires sont considérées comme étant significatives.

Pour les 2-simplexes, la diminution de la valeur-p pour l'interaction $[1, 2, 4]$ en fonction de N semble aussi être la cause de l'augmentation du nombre de faux positifs. Par exemple, si nous considérons l'interaction $[1, 2, 4]$ comme étant un vrai positif pour $N = 1\,000$ dans le cas où nous testons tous les triplets, le nombre total de vrais positifs grimpe à 1 369 tandis que le nombre de faux positifs tombe à 7. C'est donc dire que 369 des faux positifs captés sont reliés à l'interaction dont la valeur-p se situe près de α.

Cette diminution des valeurs-p pour les interactions non spécifiées explique également pourquoi le pouvoir discriminatoire entre les vrais et les faux positifs augmente moins rapidement pour cet exemple que pour le dernier.

Au terme de ces deux exemples, il est encore apparent qu'une augmentation du nombre d'observations est une condition nécessaire pour une inférence de qualité. Cela augmente non seulement notre certitude en les interactions inférées, mais permet aussi de capturer des interactions plus difficiles à détecter lorsque le nombre d'interactions est plus bas.

Exemple à 10 noeuds

Le dernier exemple traité est un complexe simplicial qui comporte 10 noeuds, illustré à la figure 4.13. Ce dernier nous a permis de constater que la construction d'un graphe de facteurs par la méthode proposée comportait une lacune.

En effet, lorsque nous spécifions la liste de facettes, nous nous attendons à ce que la liste des

vrais positifs soit identique à la liste de facettes, à moins d'être en présence d'interactions induites, ce qui ajoute des facettes à la liste de facettes effective. Or, il se trouve qu'en tentant de générer l'exemple de la figure 4.13, nous avons également observé le phénomène inverse, c'est-à-dire la destruction d'interactions spécifiées.

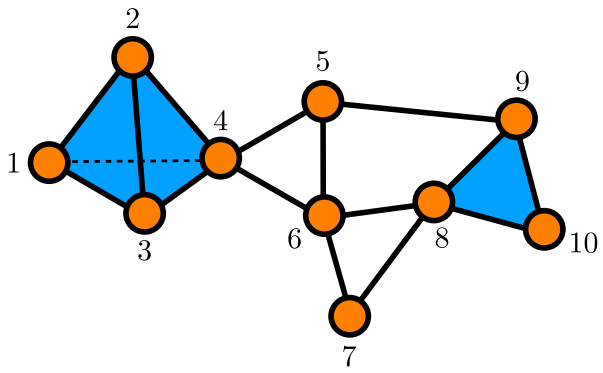

FIGURE 4.13 – Complexe simplicial à 10 noeuds utilisé pour générer les données du dernier exemple.

Le problème provient du fait que tous les facteurs sont définis un après l'autre. Or, si des noeuds participent à plusieurs interactions, cela signifie que plusieurs facteurs sont attribués à leur état de présence et d'absence. Le résultat effectif de cette combinaison apparaît alors dans la distribution des probabilités conjointes.

Toutefois, rien ne garantit que la combinaison conserve les interactions telles que définies dans la liste de facettes. Au même titre que nous pouvons générer des interactions induites grâce aux facteurs trouvés, il est donc possible de détruire les interactions spécifiées. Nous appelons la liste de facettes qui résulte de l'étude de la distribution des probabilités conjointe la **liste effective**.

Il faut donc être prudent lorsque nous appliquons l'algorithme 9 sur une liste de facettes, car la liste spécifiée et la liste effective peuvent différer. Afin de trouver un graphe de facteurs où ces deux listes sont identiques, il faut alors itérer sur plusieurs graphes de facteurs. La procédure est présentée dans l'algorithme 11.

Algorithme 11 Méthode itérative pour assurer que le graphe de facteurs possède une liste de facettes effective identique à celle spécifiée

Entrées Liste de facettes ;
Nombre d'observations N ;
Seuil α.

Sortie Graphe de facteurs ayant une liste de facettes effective identique à la liste spécifiée.

1: Appliquer l'algorithme 9 pour chaque facette dans la liste.
2: Une fois le graphe de facteurs obtenu, obtenir les tables de probabilités en utilisant une formule du type (4.28) pour tous les groupes de variables de taille 2 à k, où k est la taille de la plus grande facette dans la liste spécifiée.
3: Multiplier par N chacune des entrées dans les tables de probabilités associées aux groupes de l'étape précédente pour obtenir leur table modèle.
4: Calculer la valeur-p de la statistique du test du modèle sous-saturé pour chacune des tables de l'étape précédente.
5: Ajouter les groupes de variables dont la table a produit une valeur-p inférieure à α dans la liste des facettes effective.
6: Vérifier si les facettes effectives sont les mêmes que les facettes spécifiées. Si c'est le cas, retenir le graphe de facteurs.
7: Si ce n'est pas le cas, recommencer du début.

Avec cette dernière, nous avons tenté de générer un graphe de facteurs pour différentes valeurs de N. Nous avons alors remarqué que l'obtention d'une liste de facettes effective identique à la liste de facettes spécifiée est plus rapide lorsque le nombre d'observations N est élevée. Pour $N = 1\,000$, le nombre d'itérations moyen pour deux listes de facettes égales est 51.14 (avec un maximum de 193 itérations et un minimum de 2 sur un échantillon de 100 recherches), tandis que pour $N = 100$, nous n'avons pas pu trouver un nombre moyen en raison de la lenteur du processus. Toutefois, l'algorithme a réussi à converger vers la bonne liste en 611 itérations et vers la bonne liste avec trois interactions induites supplémentaires en 454 itérations.

Pour ne pas tomber dans une boucle sans fin en utilisant l'algorithme 11, il peut alors être utile de changer le critère d'arrêt de l'étape 6. Par exemple, nous pourrions arrêter la procédure si la liste effective contient au moins n facettes communes avec la liste spécifiée ou si l'algorithme n'a pas convergé après un nombre arbitraire d'itérations.

L'ordre dans lequel nous spécifions les interactions semble aussi avoir une influence sur la convergence vers la liste de facettes spécifiée. En spécifiant d'abord les 1-simplexes dans la liste, l'algorithme semble converger plus rapidement qu'en spécifiant les facettes dans un ordre aléatoire ou en commençant par les 2-simplexes. Une étude de convergence plus approfondie est alors de mise.

Néanmoins, c'est avec l'un des graphes de facteurs trouvés pour $N = 1\,000$ et $\alpha = 0.01$ que nous avons généré 100 matrices de présence/absence. Le nombre de vrais positifs et de faux

positifs pour les 100 matrices se retrouvent dans les histogrammes 4.14 à 4.18. Les données sur le nombre total de vrais positifs et de faux positifs ainsi que les valeurs-p maximales et minimales pour ces deux catégories sont rapportées dans le tableau 4.6. La liste de facettes effective se retrouve à l'Annexe D.

À la figure 4.14, nous pouvons observer que l'inférence a pu inférer au moins 9 des vrais 1-simplexes sur un total de 16 pour toutes les réalisations. En moyenne, nous retrouvons 13.19 1-simplexes par matrice de présence/absence. Au total, l'inférence a donc permis de retrouver 82.4% des liens. Notons également que quelques interactions spécifiées (voir l'Annexe D) possèdent des valeurs-p de l'ordre de α, c'est-à-dire 10^{-3}, les rendant possiblement difficiles à détecter en présence de perturbations, comme aux figures 3.8 et 3.14.

En ce qui concerne les faux 1-simplexes inférés, la majorité des réalisations comporte entre 0 et 2 faux positifs. En moyenne, on retrouve 0.84 faux positif par réalisation pour un total de 84 faux positifs. Or, ce nombre surpasse le nombre de faux positifs espéré, qui se situe plutôt à 29.

Dans le tableau 4.5, nous avons rapporté les faux 1-simplexes ayant apparu plus de 3 fois dans l'ensemble des réalisations. La valeur-p associée à la table modèle obtenue avec la distribution de probabilité conjointe ainsi que le nombre d'occurrences sont aussi indiqués. Nous pouvons constater que tous ces faux positifs appartiennent à un motif semblable à celui de la figure 4.12 (b), c'est-à-dire que le 1-simplexe indiqué est rattaché à un 2-simplexe. Considérant les résultats pour le cas de la figure 4.12 (b), il est alors attendu que ces interactions augmentent le nombre de faux positifs, même s'il ne s'agit pas d'interactions induites à proprement parler.

1-simplexe	Valeur-p	Nombre d'occurrences
$[1, 5]$	0.3005	6
$[1, 6]$	0.3149	11
$[2, 5]$	0.3079	11
$[2, 6]$	0.3222	11
$[5, 8]$	0.3632	5
$[5, 10]$	0.4154	6
$[7, 9]$	0.5509	5

TABLE 4.5 – Identification des 1-simplexes faussement inférés qui sont apparus plus de 3 fois dans les 100 réalisations. La valeur-p de ces interactions a été calculée avec les tables modèles obtenues à partir la distribution de probabilités conjointes du graphe de facteurs.

Puisque nous avons utilisé $N = 1\,000$, nous aurions pu nous attendre à un nombre inférieur de faux positifs considérant l'exemple 4.12 (b). Toutefois, à la différence de cet exemple, le graphe de facteurs a été trouvé en spécifiant directement $N = 1000$ et $\alpha = 0.01$. Dans les deux exemples à deux facettes, les graphes de facteurs ont été spécifiés avec $N = 100$ et $\alpha = 0.01$, faisant en sorte que l'augmentation de N réduisait les valeurs-p des interactions qui n'étaient

pas purement indépendantes.

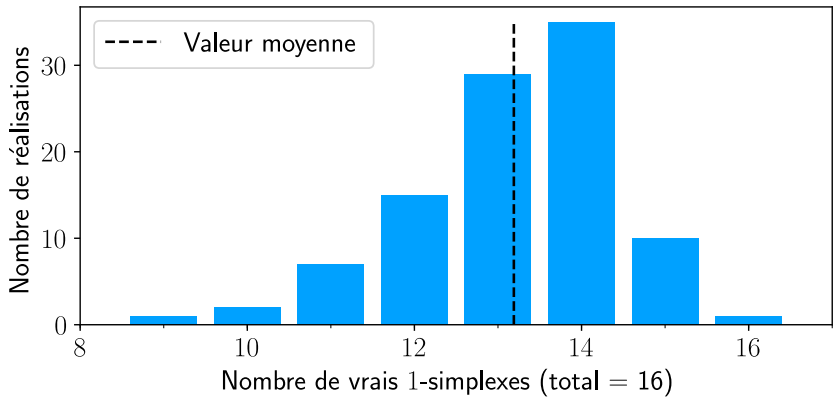

FIGURE 4.14 – Histogramme du nombre de réalisations dans lesquelles k vrais positifs (1-simplexes) ont été détectés.

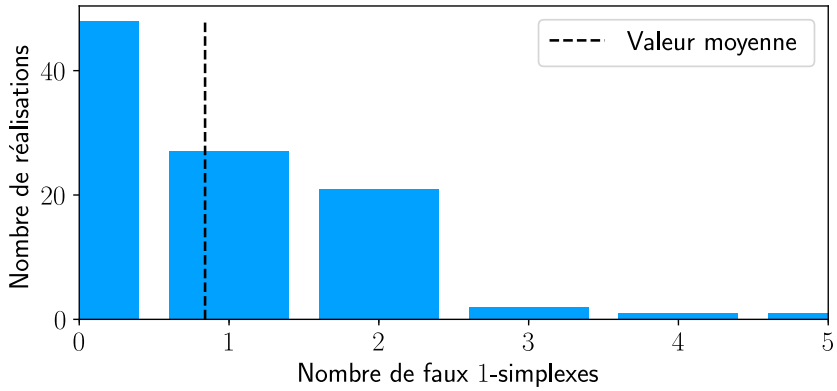

FIGURE 4.15 – Histogramme du nombre de réalisations dans lesquelles k faux positifs (1-simplexes) ont été détectés.

Pour les vrais 2-simplexes, nous avons utilisé les deux méthodes d'inférence. Dans la figure 4.16, c'est la méthode d'inférence par étape qui a été utilisée. On y observe que la majorité des cas comporte 4 ou 5 vrais positifs avec une moyenne de 4.47 par réalisation. La méthode a donc inféré 89.4% du nombre total de 2-simplexes. En contrepartie, la méthode systématique présente une moyenne de 4.74 2-simplexes par réalisation, ce qui donne un total de 94.8% sur l'ensemble. En ce sens, elle est donc plus performante que la méthode d'inférence par étape.

Toutefois, c'est lorsque nous considérons la présence de faux positifs que la méthode par étape surpasse la méthode systématique. Pour la première méthode, aucun histogramme des faux positifs n'a été produit, car seules trois réalisations comportaient chacune un faux positif alors les 97 autres n'en possédaient aucune. Or, en testant systématiquement les triplets, le nombre de faux positifs espéré est 115.

Comme nous pouvons le voir dans le tableau 4.5 et dans l'histogramme 4.18, le nombre moyen

de faux 2-simplexes par réalisation est 1.34, ce qui porte le nombre total de faux 2-simplexes à 134. Ainsi, la méthode d'inférence par étape sacrifie l'identification de vrais positifs au profit d'un nombre de faux positifs nettement inférieur à la méthode systématique, comme nous l'avons déjà observé maintes fois.

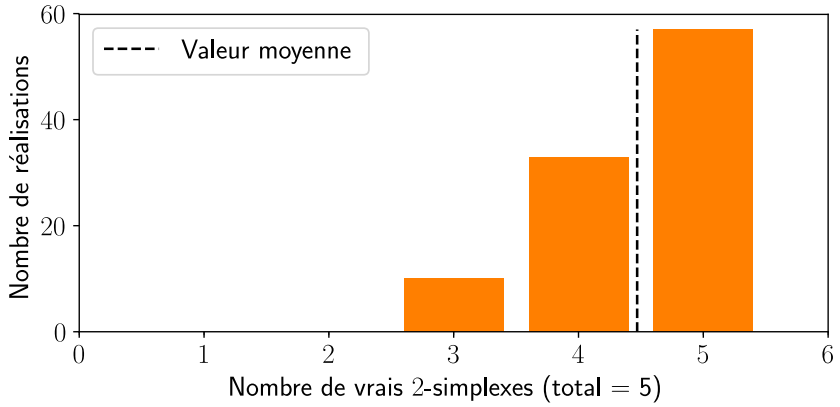

FIGURE 4.16 – Histogramme du nombre de réalisations dans lesquelles k vrais positifs (2-simplexes) ont été détectés avec la méthode d'inférence par étape.

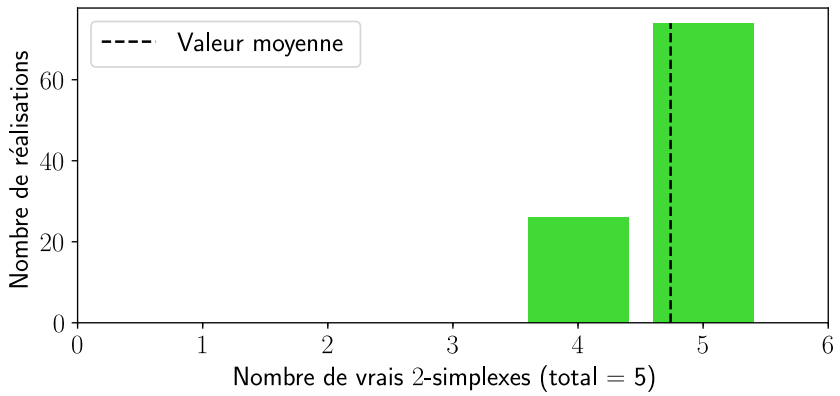

FIGURE 4.17 – Histogramme du nombre de réalisations dans lesquelles k vrais positifs (2-simplexes) ont été détectés avec la méthode d'inférence systématique.

À la lumière de tous ces exemples, nous avons pu observer comment générer des données synthétiques avec l'algorithme proposé. Nous avons également mis à l'épreuve la méthode d'inférence, qui réussit à retrouver des interactions que nous avons nous-mêmes définies entre des variables aléatoires, et ce, même dans un scénario complexe. Considérant ces performances, il semble alors à propos d'utiliser les méthodes d'inférence sur des données réelles. C'est d'ailleurs ce que nous effectuons dans la prochaine section.

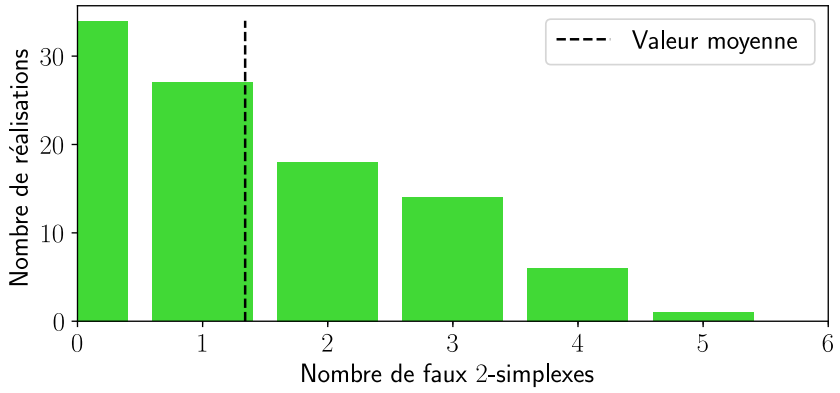

FIGURE 4.18 – Histogramme du nombre de réalisations dans lesquelles k faux positifs (2-simplexes) ont été détectés avec la méthode d'inférence systématique.

	# Vrais positifs	# Faux positifs	Valeur-p maximale (VP)	Valeur-p minimale (VP)	Valeur-p maximal (FP)	Valeur-p minimale (FP)
1-simplexe	1319	84	9.49×10^{-1}	3.10×10^{-59}	9.91×10^{-3}	3.61×10^{-5}
2-simplexe	447	3	8.12×10^{-1}	1.75×10^{-41}	7.19×10^{-3}	5.23×10^{-3}
Hyperlien	474	134	8.12×10^{-1}	1.75×10^{-41}	9.93×10^{-3}	1.66×10^{-5}

TABLE 4.6 – Nombre de vrais positifs et de faux positifs inférés dans un échantillon de 100 matrices de présence/absence générées par un graphe de facteurs obtenu avec $N = 1000$ et $\alpha = 0.01$ et dont la liste de facettes spécifiée correspond à celle du complexe simplicial de la figure 4.13. Les valeurs-p maximales obtenues pour les vrais et les faux positifs sont également indiquées. Pour la rangée 2, la méthode d'inférence par étape a été utilisée, tandis que pour la rangée 3, la méthode d'inférence systématique a été utilisée.

4.2 Données réelles

Les jeux de données que nous allons traiter dans la présente section correspondent à ceux qui ont été présentés à la section 1.1.3 du chapitre 1. Pour tous les tests d'hypothèse effectués, nous avons utilisé $\alpha = 0.01$.

4.2.1 Jeu de données d'OTUs

Le jeu de données des OTUs est celui qui comporte le moins d'observations. En effet, nous ne possédons que 34 observations pour les 2166 OTUs. L'échantillon ne respecte donc pas le critère de la référence [23] qui indique que nous devrions avoir un nombre d'observations au moins dix fois plus grand que le nombre de cases dans la table de contingence.

De plus, en vertu des mesures de performances de la section 3.3, nous savons qu'un « petit » nombre d'observations se traduit par des conclusions peu robustes. Rappelons également que les observations, même si elles sont indépendantes les unes des autres, ne sont probablement pas identiquement distribuées, car les thermokarsts possèdent des propriétés physico-chimiques différentes [16].

La robustesse de nos conclusions n'est donc pas garantie, et celles-ci doivent donc être interprétées avec un certain scepticisme. Cependant, nous pouvons tout de même analyser les données en faisant abstraction de ces lacunes. De plus, il s'agit d'une bonne opportunité pour comparer la méthode asymptotique par étape et les deux approches exactes proposées au chapitre 3.

Méthode asymptotique

En premier lieu, en appliquant la méthode d'inférence asymptotique, nous inférons 148 669 1-simplexes sur une possibilité de $\binom{2166}{2} = 2\,344\,695$. Même si nous ne savons pas quelles sont les véritables interactions, nous pouvons estimer le nombre de faux positifs dans le cas extrême où l'indépendance serait vraie pour chaque paire. En effet, il suffit de multiplier le nombre de tests effectués, c'est-à-dire 2 344 695, par α, qui correspond à la probabilité de rejeter l'hypothèse nulle alors qu'elle est vraie. On estime alors le nombre de faux positifs à 23 447 dans un tel scénario. Ce nombre étant inférieur au nombre de liens inférés, nous pouvons supposer que la plupart des interactions identifiées sont de vrais positifs. Toutefois, il faut se rappeler que le peu d'observations ne permet pas d'atteindre des conclusions robustes. Cela peut également influencer à la hausse (ou à la baisse) le nombre de faux positifs.

En ce qui concerne les interactions d'ordre deux, nous en obtenons 366 118 si nous testons systématiquement tous les triplets sur une possibilité de $\binom{2166}{3} = 1\,691\,306\,660$. Dans ce cas-ci, on ne peut pas appliquer directement la formule précédente pour estimer le nombre de faux positifs. En effet, sur les 1 691 306 660 triplets possibles, seuls 19 585 553 d'entre eux ont pu être testés, car les autres tables ne possédaient pas des formes valides pour l'algorithme itératif. Ainsi, il faut plutôt estimer le nombre de faux positifs avec le nombre de triplets dont

la table est valide, ce qui donne 195 856 faux positifs. Encore une fois, même si le nombre d'interactions inférées est supérieur au nombre de faux positifs espéré, il faut se rappeler que le peu d'observations ne nous permet pas de conclure en toute rigueur si les interactions sont des vrais positifs ou non.

Au terme de la première étape d'inférence, 6 064 859 triangles vides ont été générés en connectant toutes les paires significatives. Or, seuls 192 831 triangles ont pu être testés et aucun ne permet de rejeter le modèle sous-saturé à trois variables. Cela suggère qu'un nombre d'observations plus important est nécessaire pour inférer de véritables 2-simplexes.

Méthodes exactes

Comme l'inférence via les méthodes exactes est plus lente que l'inférence via la méthode asymptotique, nous allons seulement nous concentrer sur les paires. Or, avec autant de paires possibles, l'obtention des distributions exactes est un long processus. Heureusement, il se trouve que le nombre de tables de contingence uniques de taille 2×2 est inférieur au nombre de paires possibles.

Ainsi, nous pouvons extraire la liste des tables uniques et générer la distribution exacte de la statistique χ_0^2 pour chacune. [12] Nous pouvons ensuite associer une valeur-p à la table de contingence et trouver toutes les paires d'OTUs qui génèrent cette même table. Dans les analyses qui suivent, les distributions exactes ont été formées en générant un million de statistiques χ_0^2 à partir des algorithmes 5 et 6.

Pour chacune des méthodes d'inférence exacte, nous avons rapporté le nombre de tables uniques valides, le nombre de tables uniques valides ayant une valeur-p inférieure à $\alpha = 0.01$, ainsi que le nombre de 1-simplexes inférés sur les 2 344 695 possibilités dans le tableau 4.7. Il est important de noter que parmi les 4 395 tables 2×2 uniques valides, 37 tables présentent des valeurs nulles dans la table espérée sous le modèle d'indépendance. Cela signifie alors que les méthodes exactes ne pourront pas traiter ces 37 cas (à moins de conclure directement l'indépendance directement).

Dans le tableau 4.7, nous pouvons constater que les deux méthodes exactes présentent moins de tables uniques valides pour l'analyse. Cela était attendu en raison de l'impossibilité d'obtenir la distribution exacte si les tables espérées comportent des entrées nulles aux étapes 2 ou 4 de l'algorithme 5 pour la méthode à 1 degré de liberté et à l'étape 2 de l'algorithme 6 pour la méthode à 3 degrés de liberté.

Pour l'ensemble des 923 tables uniques identifiées avec la méthode exacte à 1 degré de liberté, nous avons calculé la valeur-p moyenne. Nous avons également effectué ce calcul sur ces mêmes 923 tables, mais avec les données des deux autres méthodes. La méthode asymptotique obtient

12. Nous avons également essayé pour les triplets, mais l'identification de tables uniques valides a été jugée trop lente et a été abandonnée.

Méthode	Nombre de tables uniques valides	Tables ayant une valeur-$p < \alpha$	Nombre de 1-simplexes
Asymptotique	4395	797	148 669
Exacte 1 degré	923	212	471
Exacte 3 degrés	4358	266	7 318

TABLE 4.7 – Nombre de tables uniques valides, nombre de tables uniques valides ayant une valeur-p inférieure à $\alpha = 0.01$ et nombre de 1-simplexes inférés à l'aide de la méthode d'inférence utilisant une distribution asymptotique et les deux méthodes utilisant une distribution exacte à 1 degré de liberté et à 3 degrés de liberté sur les données d'OTUs.

une valeur-p moyenne de 0.258 avec un écart-type de 0.298, tandis qu'elle est de 0.265 avec un écart-type de 0.301 pour la méthode à un degré de liberté et de 0.499 avec un écart-type de 0.382 pour la méthode à trois degrés de liberté.

Qualitativement, la méthode à 1 degré de liberté semble donc produire des résultats semblables à la méthode asymptotique sur l'ensemble. Cela demeure vrai pour les 212 tables ayant produit des valeurs-p sous α, car les valeurs-p moyennes pour ces deux méthodes sont 2.88×10^{-3} et 2.91×10^{-3} avec des écarts-types respectifs de 3.03×10^{-3} et 2.90×10^{-3}. L'utilisation de la méthode exacte à 1 degré de liberté n'est donc peut-être pas essentielle dans ce cas. Par contre, on notera qu'elle demeure plus conservatrice, car elle ne permet pas de tester autant d'interactions que la méthode asymptotique, d'où la diminution de tables uniques valides ayant une valeur-p inférieure à α.

En comparaison, la méthode exacte à 3 degrés de liberté semble être plus conservatrice que la méthode asymptotique, car elle produit des valeurs-p qui sont, en moyenne, plus élevées. C'est donc pour cette raison que son nombre de tables uniques valides ayant une valeur-p inférieure à α est plus bas que celui de la méthode asymptotique.

Les méthodes d'inférence, qu'elles soient exactes ou asymptotiques, ont donc permis d'identifier des interactions par paires. Pour les interactions d'ordre supérieur, nous avons seulement testé la méthode asymptotique par étape et la méthode asymptotique systématique. Cette dernière est la seule ayant permis d'inférer des interactions d'ordre supérieur, mais le peu d'observations nous oblige d'interpréter les conclusions avec prudence.

En ce qui concerne les méthodes exactes, il semble que 34 observations est un nombre d'observations assez grand pour que la différence entre les valeurs-p de la méthode asymptotique et exacte à 1 degré de liberté ne soit pas significative. En ce sens, il serait intéressant de caractériser le nombre d'observations nécessaire pour voir une différence entre la méthode exacte à 1 degré de liberté et la méthode asymptotique. De même, il pourrait être intéressant de déterminer la taille de l'échantillon nécessaire pour construire la distribution exacte adéquate-

ment. Des tests pour le comportement des méthodes exactes sur des tables $2 \times 2 \times 2$ seraient également à propos.

4.2.2 Jeu de données sur les oiseaux datant de 2016

Ce jeu de données comporte 70 espèces et 185 sites. Il respecte donc le critère arbitraire où le nombre d'observations est supérieur à 10 fois le nombre de cases dans les tables 2×2 et $2 \times 2 \times 2$. Sachant cela, nous pouvons appliquer la méthode asymptotique avec confiance. Par contre, nous allons également tester les deux méthodes exactes comme nous l'avons fait à la section précédente.

Méthode asymptotique

Pour ces données, nous inférons 149 1-simplexes sur un total de 2 415 possibilités et 124 hyperliens sur un total de 54 740 si nous testons systématiquement tous triplets. En estimant le nombre total de faux positifs de la même manière que dans l'exemple précédent, nous devrions alors retrouver 24 faux 1-simplexes.

En ce qui concerne les hyperliens, seuls 7 581 triplets ont pu être testés. Le nombre de faux positifs espéré est donc 76. Sinon, parmi les 96 triangles vides inférés à la première étape, seuls 80 peuvent être testés et deux d'entre eux sont devenus des 2-simplexes. Peu importe l'approche utilisée, il y a donc lieu de penser que les 2-simplexes inférés et certains hyperliens sont bien des cooccurrences d'ordre supérieur.

Méthodes exactes

Pour ce jeu de données, nous avons fait une analyse semblable à celle de la section précédente et avons présenté les résultats dans le tableau 4.8 pour les 1-simplexes et le tableau 4.9 pour les hyperliens.

Dans les deux cas, les données semblent suivre les mêmes tendances qu'à la section précédente. En effet, en calculant les valeurs-p moyennes des trois méthodes sur les 399 tables valides identifiées de la méthode exacte à 1 degré de liberté, nous obtenons 0.284, 0.286 et 0.533. Dans l'ordre ces dernières sont associées à la méthode asymptotique, exacte à 1 degré de liberté et exacte à 3 degrés de liberté respectivement.

En ce qui concerne les hyperliens, nous remarquons encore l'aspect conservateur des méthodes exactes. Pourtant nous utilisons un jeu de données où le nombre d'observations est supérieur à dix fois le nombre de cases dans une table $2 \times 2 \times 2$. Malgré cela, la méthode exacte à 1 degré de liberté étant sensible aux cases comportant peu d'observations, elle ne peut être utilisée dans tous les cas.

Considérant que les deux méthodes exactes produisent deux hyperliens, nous avons vérifié s'il s'agissait des deux 2-simplexes trouvés avec la méthode d'inférence par étape entre les

Méthode	Nombre de tables uniques	Tables ayant une valeur-$p < \alpha$	Nombre de 1-simplexes
Asymptotique	1090	128	149
Exacte 1 degré	399	79	79
Exacte 3 degrés	1089	45	46

TABLE 4.8 – Nombre de tables uniques valides, du nombre de tables uniques valides ayant une valeur-p inférieure à $\alpha = 0.01$ et du nombre de 1-simplexes inférés à l'aide de la méthode d'inférence utilisant une distribution asymptotique et les deux méthodes utilisant une distribution exacte à 1 degré de liberté et à 3 degrés de liberté sur le jeu de données des oiseaux datant de 2016.

triplets $[24, 32, 52]$ et $[38, 53, 67]$. Nous avons donc identifié les triplets et leur valeur-p pour les trois méthodes dans le tableau 4.10. Comme nous pouvons le constater, il s'agit de six triplets différents, donc aucune méthode exacte n'a pu capter les deux 2-simplexes inférés avec la méthode asymptotique par étape. Nous pouvons aussi remarquer que les valeurs-p pour la méthode exacte à 7 degrés de liberté sont plus conservatrices, comme attendu. [13] Nous remarquons aussi que les valeurs-p de la méthode exacte à 1 degré de liberté sont du même ordre que celles de la méthode asymptotique. L'utilisation de distributions exactes n'est donc possiblement pas essentielle.

Méthode	Nombre de tables uniques	Tables ayant une valeur-$p < \alpha$	Nombre d'hyperliens
Asymptotique	7 372	128	124
Exacte 1 degré	113	2	2
Exacte 7 degrés	7372	2	2

TABLE 4.9 – Nombre de tables uniques valides, du nombre de tables uniques valides ayant une valeur-p inférieure à $\alpha = 0.01$ et du nombre de 2-simplexes inférés à l'aide de la méthode d'inférence utilisant une distribution asymptotique et les deux méthodes utilisant une distribution exacte à 1 degré de liberté et à 7 degrés de liberté sur le jeu de données des oiseaux datant de 2016.

Pour ce jeu de données, nous avons donc pu inférer des interactions d'ordre supérieur. Le caractère conservateur de la méthode d'inférence par étape nous permet d'ailleurs de croire que les deux 2-simplexes trouvés sont bien réels. Par contre, une analyse plus approfondie est nécessaire pour déterminer s'il s'agit d'un phénomène purement mathématique ou si ces 2-simplexes possèdent bien une signification dans le monde biologique.

13. Rappelons que le nombre degrés de liberté de cette méthode va comme $2^d - 1$, où d est le nombre de variables.

	Valeur-p		
Triplet	Asymptotique	Exacte 1 degré	Exacte 7 degrés
$[24, 32, 52]$	1.60×10^{-3}	-	1.88×10^{-1}
$[38, 53, 67]$	2.74×10^{-3}	-	2.16×10^{-1}
$[21, 32, 58]$	4.04×10^{-3}	3.96×10^{-3}	3.07×10^{-1}
$[21, 40, 43]$	7.48×10^{-3}	7.45×10^{-3}	4.12×10^{-1}
$[21, 32, 63]$	8.00×10^{-6}	-	7.27×10^{-3}
$[23, 52, 63]$	4.25×10^{-8}	-	4.14×10^{-4}

TABLE 4.10 – Valeurs-p obtenues avec la méthode asymptotique et les deux méthodes exactes pour différents triplets. Les triplets $[24, 32, 52]$ et $[38, 53, 67]$ correspondent aux 2-simplexes trouvés par la méthode d'inférence asymptotique par étape, tandis que les deux triplets $[21, 32, 58]$ et $[21, 40, 43]$ ont été découverts par la méthode exacte à un degré de liberté sur le jeu de données des oiseaux datant de 2016. Les deux derniers triplets ont été découverts par la méthode exacte à 7 degrés de liberté.

4.2.3 Jeu de données sur les oiseaux datant de 2019

Ce jeu de données comporte 115 espèces et 1382 sites. Nous allons donc seulement examiner la méthode asymptotique, car comme nous l'avons vu dans l'exemple précédent, il semble que la méthode exacte à 1 degré de liberté produise des valeurs-p semblables à la méthode asymptotique pour au moins 185 observations.

Dans ce jeu de données, nous comptons 590 1-simplexes sur un possibilité de 6 555 et 892 hyperliens sur une possibilité de 246 905. Le nombre de faux positifs espéré pour les 1-simplexes est donc 66. En ce qui concerne les hyperliens, même si nous avons plus de 1 000 observations, il n'y avait que 40 554 triplets qui ont pu être analysés. Le nombre de faux positifs espéré est alors 406. En utilisant plutôt la méthode d'inférence par étape, nous pouvons identifier 953 triangles vides, mais seuls 847 d'entre eux peuvent être testés et 36 sont des 2-simplexes.

Si nous construisons alors le complexe simplicial avec les 590 1-simplexes et les 36 2-simplexes, nous obtenons une structure dans laquelle le nombre de Betti β_2 n'est pas trivial (comparativement à la section 4.2.2 où il y avait seulement deux 2-simplexes, ne permettant pas de former un cycle). En calculant les trois premiers nombres de Betti, nous obtenons alors $\beta_0 = 5$, $\beta_1 = 444$ et $\beta_2 = 0$.

Parmi les 5 composantes trouvées avec β_0, 4 d'entre elles sont en fait des espèces d'oiseaux complètement indépendantes des autres. La cinquième composante connecte donc le reste des 111 espèces. En ce qui concerne β_1, nous n'avons pas trouvé de signification aux cycles. Par contre, β_2 peut nous renseigner sur le nombre maximal de candidats possibles pour une interaction d'ordre trois. Dans ce cas-ci, ce dernier est nul, faisant en sorte que nous ne pourrons pas trouver de 3-simplexes dans ces données.

Au terme des trois derniers exemples, nous pouvons constater que la présence d'éléments nuls dans les tables de contingence gêne l'inférence d'interactions d'ordre supérieur. En effet, même si le nombre d'observations est considéré comme étant acceptable dans deux exemples, la méthode itérative n'arrive pas à tester pour tous les candidats possibles, autant pour les hyperliens que les 2-simplexes.

4.2.4 Jeu de données MEDLINE

Le dernier jeu de données est celui qui se prête le mieux à l'analyse, car il comporte un nombre important d'observations qui, en principe, répondent aux critères d'indépendance et de distribution identique. Ici, les entités correspondent aux *MeSH* (*Medical Subject Headings*).

Dans ce cas, une interaction, qu'elle soit d'ordre supérieur ou non, peut signifier que nous avons une unité linguistique significative. Le sens donné à la combinaison des mots est donc différent des mots seuls. Par exemple, nous pourrions avoir les étiquettes *souris*, *maladie* et *poumons*. Chaque paire de mots est sensée et peut nous renseigner sur le contenu de l'article : (maladie, souris), (maladie, poumons), (souris, poumons). Après la lecture de chaque pair, il semble alors naturel de former le triplet : (maladie, poumons, souris). Ce triplet rend alors le contenu de l'article encore plus explicite. En utilisant la méthode d'inférence, nous déterminons donc quelles sont les cooccurrences triples significatives et si elles peuvent s'expliquer seulement par les paires ou non. Sur ce jeu de données, des analyses de cooccurrences par paire utilisant le test d'indépendance et la statistique χ_0^2 ont déjà été effectuées [33]. Notre technique généralise alors cette procédure pour les ordres supérieurs.

Pour tester la méthode d'inférence sur ces données, nous avons choisi l'année 1980, qui comporte 14 274 MeSH et 280 646 publications scientifiques. Parmi celles-ci, nous avons sélectionné aléatoirement 50 000 publications, puis filtré la matrice résultante pour retirer les MeSH qui étaient absents dans ce sous-ensemble de publications. Dans les MeSH restants, nous en avons sélectionné 3 500 aléatoirement. La matrice de présence/absence résultante est alors une matrice de dimension $3\,500 \times 50\,000$ qui comporte des MeSH qui se présentent une fois ou plus dans les 50 000 publications.

Nous avons fait ce choix pour accélérer la recherche de cooccurrences d'ordre supérieur. En effet, réduire le nombre de publications considéré augmente la vitesse de l'algorithme (le nombre d'itérations par seconde), tandis que réduire le nombre de MeSH réduit le nombre total d'itérations. De plus, pour ce jeu de données, nous avons seulement utilisé la méthode d'inférence par étape.

Ainsi, nous avons inféré 38 843 1-simplexes sur une possibilité de 6 123 250 et 5 190 2-simplexes sur une possibilité de 76 833 triangles vides. Il est à noter que pour obtenir ce nombre de 2-simplexes, tous les 1-simplexes dont la valeur-p se situe sous α sont conservés, peu importe le signe de leur coefficient ϕ. Ainsi, pour construire les 2-simplexes, il est possible que des paires

présentant une association négative soient rattachées avec des paires dont le coefficient est positif. L'interprétation du 2-simplexe n'est alors pas aussi aisée que dans notre exemple avec les termes *souris*, *maladie* et *poumons*.

Grâce à ces données, nous avons pu construire un complexe simplicial de dimension 2 et nous avons pu déterminer ses nombres de Betti qui sont $\beta_0 = 65$, $\beta_1 = 30\,809$ et $\beta_2 = 591$. Parmi les 65 composantes déterminées par le calcul de β_0, nous en comptons 61 qui sont des MeSH isolés, c'est-à-dire qu'ils ne forment aucune connexion avec les autres noeuds. Autrement dit, ils se présentent de manière indépendamment des autres. [14] Les quatre autres composantes méritent alors d'être étudiées plus en profondeur, car elles pourraient révéler des communautés centrées autour d'un thème.

Un sens pourrait également être dégagé des nombres β_1 et β_2. Par exemple, β_1 est le nombre de cycles composés par des simplexes de dimension 1. Il y a donc 30 809 chaînes de concepts où une étiquette ne forme un sens qu'avec ses deux voisins immédiats dans le complexe simplicial, mais où il ne semble pas y avoir de sens à connecter ces deux voisins entre eux. En ce qui concerne les cycles formés par les 2-simplexes, nous pourrions trouver une signification similaire, mais il ne s'agit pas d'un objectif ici. Par contre, au sens de l'inférence d'interactions d'ordre supérieur, ces 591 cycles ouvrent la porte à la recherche de 3-simplexes.

En effet, dans la méthode d'inférence par étape, l'ordre 3 ne peut exister que si quatre 2-simplexes sont connectés de manière à former l'enveloppe d'un tétraèdre. Tel que vu précédemment, un tel motif permet d'ajouter une unité au compte β_2. Évidemment, rien ne garantit que les 591 cycles identifiés soient seulement des tétraèdres vides. Il existe d'autres agencements de 2-simplexes qui permettent d'ajouter une unité à β_2, mais qui nécessitent plus que quatre 2-simplexes.

Afin d'identifier les candidats pour l'inférence d'interaction d'ordre 3, nous avons élaboré un algorithme pour trouver les cliques de taille 4 où chacun des $\binom{4}{3}$ 2-simplexes possibles dans cette clique existe dans le complexe simplicial. Cela permet de trouver 430 cliques, ce qui signifie que les 161 autres cycles comptés dans β_2 comportent plus de quatre 2-simplexes. En testant le modèle sous-saturé à quatre variables sur ces 430 cliques à l'aide de l'algorithme itératif, nous obtenons 21 3-simplexes. En ajoutant ces structures dans le complexe simplicial, les nouveaux nombres de Betti sont alors $\beta_0 = 65$, $\beta_1 = 30\,809$, $\beta_2 = 570$ et $\beta_3 = 0$. Ces résultats laissent alors entrevoir que nous pourrions trouver davantage d'interactions sur le jeu de données complet et que nous pourrions chercher des simplexes de dimension supérieure à 3.

Au terme de ce chapitre, nous pouvons constater que les méthodes d'inférence réussissent à trouver des relations véridiques dans un contexte où les interactions sont connues. La méthode

14. Notons toutefois que si nous avions considéré le jeu de données total, il est plausible que ces MeSH aient formé des connexions avec les MeSH que nous n'avons pas considérés ici.

d'inférence par étape semble aussi être plus performante que la méthode systématique pour ne pas inférer de faux positifs, au détriment d'un nombre de vrais positifs inférieur. Malgré tout, les performances des deux méthodes d'inférence ouvrent la porte à la recherche de cooccurrences d'ordre supérieur dans les jeux de données réels. Les quatre exemples présentés ont permis de constater de nouveau que le nombre d'observations est un paramètre limitant pour l'inférence d'interactions d'ordre supérieur, surtout dans un contexte où nous utilisons l'inférence par étape pour produire un complexe simplicial. Néanmoins, des interactions d'ordre supérieur ont bien été détectées dans trois jeux de données, allant même jusqu'à l'ordre 3 pour le dernier. Nous avons aussi montré que la construction du complexe simplicial et le calcul de ses nombres de Betti peuvent nous renseigner sur diverses caractéristiques, comme le nombre d'entités complètement indépendantes ou le nombre de candidats potentiels pour l'inférence d'interactions au prochain ordre. Les complexes simpliciaux inférés possédant des facettes de taille moindre que ceux construits au chapitre 2, nous pourrions également utiliser de nouveau le SCM pour déterminer si l'homologie des structures inférées est significative par rapport à un ensemble aléatoire.

Conclusion et perspectives

Le raffinement d'un modèle est parfois nécessaire pour représenter plus adéquatement la nature. Les modèles réseaux, venant avec leur lot d'hypothèses et de limites, n'échappent pas à cette réalité. Bien qu'ils permettent indéniablement d'étudier de nombreux systèmes plus en profondeur, ils ne peuvent que représenter des interactions d'ordre un, c'est-à-dire des interactions par paire. Évidemment, ce ne sont pas toutes les interactions qui peuvent être réduite à l'ordre un sans perte d'information importante. Pour raffiner ces modèles et notre compréhension, il est donc nécessaire d'ajouter les interactions d'ordre supérieur dans l'équation. En ce sens, il est possible d'utiliser d'autres modèles, comme les hypergraphes et les complexes simpliciaux. Par contre, pour qu'un modèle bénéficie des outils développés pour intégrer les interactions d'ordre supérieur, il faut être en mesure de déterminer la nature des interactions dans le système.

À cet effet, l'objectif de ce mémoire était de développer une technique d'inférence pour extraire les interactions, peu importe leur ordre, à partir de données de présence/absence. Le but derrière cette méthode est ultimement de construire un complexe simplicial (ou un hypergraphe) qui représente plus fidèlement la structure du système étudié. L'inférence utilisée étant de nature statistique, elle est sensible aux écarts entre les observations expérimentales et la réalité. Ainsi, nous avions pour second objectif de montrer que la méthode d'inférence arrive à retrouver les interactions véridiques dans un contexte où les données sont bruitées.

Dans la première approche explorée au chapitre 2, l'inférence d'interactions d'ordre supérieur était indirecte. En effet, nous présupposions l'existence de ces interactions en faisant l'hypothèse que des entités se présentant sur un même site formaient un simplexe. Nous comparions ensuite la topologie du complexe simplicial à celle de complexes simpliciaux générés par le modèle simplicial des configurations. Si la topologie de notre complexe simplicial s'avérait significativement différente de celle des complexes de l'ensemble, nous aurions alors conclu que les simplexes obtenus à partir de notre hypothèse de construction initiale étaient plausibles. Cette conclusion manque toutefois de fondements, car une topologie particulière pour un complexe simplicial ne confirme ou n'infirme pas la présence d'interactions d'ordre supérieur. Elle stipule seulement qu'une caractéristique structurelle n'est pas aléatoire. Néanmoins, il s'agissait d'une première approche intéressante considérant l'existence des outils nécessaires pour

générer le modèle nul et mesurer numériquement les nombres de Betti.

De toute manière, pour la plupart des données, nous n'avons pas pu appliquer cette procédure, car la dimension des présumés complexes simpliciaux ainsi que leur séquence des degrés généralisés et leur séquence des tailles des facettes étaient trop contraignantes pour générer un ensemble de complexes simpliciaux avec le SCM. Il est toutefois important de comprendre que l'échec de cette méthode n'exclut pas l'utilisation du SCM dans le contexte des données de présence/absence. Par exemple, nous pourrions vérifier si cette procédure est applicable sur les complexes simpliciaux inférés à partir des différentes méthodes proposées au chapitre 3. Ces derniers sont probablement plus souples que ceux construits par l'hypothèse initiale du chapitre 2. Mettre en conjonction ces deux techniques est d'ailleurs une perspective emballante, car d'un côté nous inférons les interactions d'ordre supérieur et de l'autre nous pouvons déterminer si la topologie du système étudiée n'est pas aléatoire.

L'exploration du SCM a aussi eu des retombées intéressantes, car elle nous a permis de développer une procédure pour dénicher quels groupes d'homologie sont non triviaux à partir de l'information contenue dans une liste de facettes. Cette technique peut être jumelée à d'autres techniques pour réduire la complexité numérique du complexe simplicial étudié. Résultat, la rapidité du calcul des nombres de Betti est augmentée lors de l'utilisation du paquetage GUDHI et le stress sur la mémoire vive de l'ordinateur est réduit. Cette exploration nous a aussi inspirés pour la construction du modèle génératif du chapitre 4, car elle nous a renseignés sur le lien important entre les réseaux bipartis et les complexes simpliciaux.

Au chapitre 3, nous avons développé un cadre théorique pour l'inférence de complexes simpliciaux (et d'hypergraphes) à partir des modèles log-linéaires. Ce dernier répond donc précisément à l'objectif de notre travail. En effet, les approches développées à partir de ce cadre permettent de dénicher les dépendances significatives ordre par ordre. Cependant, elles sont parfois limitées par le nombre d'observations dans la matrice de présence/absence et ne peuvent être utilisées que dans les cas où les données sont indépendantes et identiquement distribuées. Dans le cas où peu d'observations sont disponibles, nous avons montré comment nous pouvions utiliser deux méthodes exactes pour tenter d'inférer des interactions. Davantage de tests seraient pertinents pour déterminer l'intervalle du nombre d'observations dans lequel il est pertinent (ou non) d'utiliser une méthode exacte plutôt qu'une des méthodes asymptotiques, et ce, pour des tables de contingence de différentes dimensions [15]. Les outils d'analyse de performance, également développés au chapitre 3, pourraient être utilisés à cet effet. D'un point de vue pratique, l'analyse du taux de succès en fonction de la distance L_1 pourrait aussi servir à caractériser le niveau de certitude des conclusions sur des données bruitées réelles.

Pour améliorer le cadre théorique sur lequel se basent les différentes méthodes, plusieurs pistes

15. Rappelons que le seul critère arbitraire rencontré dans la littérature stipule que le nombre minimal d'observations pour l'utilisation d'un test asymptotique correspond à dix fois le nombre de cases dans la table de contingence [23]. L'identification d'une borne claire demeure alors un problème ouvert.

sont disponibles. Par exemple, les modèles quasi log-linéaire ainsi que la théorie des maximums de vraisemblance étendus (*extended maximum likelihood*) pourraient être explorés. En effet, ces deux éléments représentent des avenues potentielles pour gérer les cas problématiques qui se présentent lorsque peu d'observations sont disponibles ou lorsque les tables de contingence présentent une disposition particulière des éléments nuls et non nuls. De même, nous pourrions tenter d'étendre le cadre théorique à des données qui ne sont pas indépendantes ou identiquement distribuées, ce qui peut survenir si les sites sont reliés ou si plusieurs observations sont faites au sein d'un même site dans différentes conditions.

En ce qui concerne les relations entre les entités, il serait à propos de vérifier l'influence du processus de sélection de modèles complet sur le complexe simplicial. De plus, l'utilisation de statistiques différentes, comme la statistique G^2 ou la statistique χ_0^2 normalisée [2, 11], pourrait aussi être envisagée. Pour quantifier les associations, l'utilisation de mesures propres à la théorie de l'information est également à propos.

Au chapitre 4, nous avons conçu un modèle génératif pour synthétiser des données de présence/absence dans lesquelles les dépendances sont connues en utilisant des principes évoqués dans les trois autres chapitres. Ce modèle a permis de montrer que les méthodes d'inférence sont en mesure de retrouver les dépendances réelles entre les variables aléatoires. Par contre, étant des méthodes statistiques, il arrive que ces dernières se trompent dans leurs conclusions. Lorsque la méthode d'inférence systématique est utilisée sur des données synthétiques, le nombre de faux positifs est approximativement égal au seuil critique multiplié par le nombre de tests complétés. En contrepartie, la méthode d'inférence par étape, laquelle permet de construire un complexe simplicial, induit moins de faux positifs au détriment d'un nombre de faux négatifs plus élevé. Pour comparer davantage les deux méthodes, il pourrait être intéressant de caractériser le risque de deuxième espèce de chacune des méthodes.

Finalement, grâce aux résultats encourageants sur les données synthétiques, nous avons également traité des données de présence/absence réelles pour dénicher les interactions d'ordre supérieur. Dans tous les jeux de données, des interactions d'ordre supérieur ont été trouvées en utilisant la méthode d'inférence systématique. Par contre, seuls trois jeux de données ont permis d'inférer un complexe simplicial de dimension supérieure à 1 en utilisant la méthode d'inférence par étape. En effet, dans le jeu de données d'OTUs, l'inférence d'un complexe simplicial n'a pas été possible, probablement en raison du faible nombre d'observations. Dans le jeu de données sur les oiseaux datant de 2019 et dans le jeu de données MEDLINE, nous avons aussi caractérisé la topologie des complexes simpliciaux inférés en calculant leurs nombres de Betti. Nous avons également montré qu'un nombre de Betti positif, pour la dernière dimension calculée, indique que nous pouvons appliquer la méthode d'inférence par étape à l'ordre suivant. C'est ainsi que nous avons pu dénicher des cooccurrences d'ordre trois dans le jeu de données MEDLINE.

Ces résultats ouvrent alors la porte à de nouvelles questions. Par exemple, considérant le peu d'observations dans le jeu de données d'OTUs, il est naturel de se demander si l'utilisation des méthodes proposées était adéquate et si les hyperliens trouvés ne sont pas que des artefacts mathématiques. Une augmentation du nombre d'observations serait également de mise, autant pour être plus confiant des interactions trouvées, que pour permettre l'utilisation de la méthode par étape et l'identification de 2-simplexes. En ce qui concerne les autres jeux de données, plusieurs avenues s'offrent à nous. Tout d'abord, il serait intéressant de déterminer la signification des cooccurrences, car l'analyse ne révèle pas pourquoi les interactions par paire ne peuvent expliquer l'ordre supérieur. Grâce à des hypothèses sur la signification de ces interactions, d'autres travaux pourraient être effectués pour tenter de les observer directement en nature. Cela permettrait de prouver, de manière expérimentale, que les interactions d'ordre supérieur inférées sont réelles. De même, la topologie des complexes simpliciaux devrait être étudiée plus en profondeur pour établir pourquoi il existe des « trous » dans la structure et quelle est leur signification. Quoiqu'il en soit, tous ces résultats montrent qu'il est possible d'inférer des interactions d'ordre supérieur à partir de données de présence/absence. Les objectifs de ce projet sont donc atteints et il ne reste plus qu'à explorer cette panoplie de perspectives emballantes pour bonifier davantage l'analyse de cooccurrences d'ordre supérieur.

Annexe A

Théorie des groupes et notions sur les applications

La notion de groupe occupe une place prépondérante dans le formalisme de l'homologie. En fait, des quantités importantes, comme les nombres de Betti, sont obtenues suite à la construction de ce que l'on appelle les groupes d'homologie. Ces derniers sont dotés de caractéristiques spécifiques qui seront présentées ici. Pour débuter, rappelons certaines définitions en lien avec les groupes qui sont principalement tirées des références [40] et [45].

Un **groupe**[1] $(G, *)$ d'ordre n est un ensemble G de n éléments $\{g_1, ..., g_n\}$, muni d'une opération $*$ qui associe à deux éléments de G un troisième, c'est-à-dire $g_i * g_j = g_k$ et qui satisfait les quatre conditions suivantes

1. Fermeture : $\forall\ g_i, g_j \in G, g_i * g_j \in G$;
2. Associativité : $g_i * (g_j * g_k) = (g_i * g_j) * g_k$;
3. Existence de l'unité, notée e et telle que : $g_i * e = e * g_i = g_i \quad \forall \quad g_i \in G$;
4. Existence de l'inverse : Pour tout $g_i \in G$, il existe un élément $-g_i$ tel que $g_i * -g_i = -g_i * g_i = e$.

Si l'opération du groupe est commutative, c'est-à-dire $g_i * g_j = g_j * g_i$, alors le groupe est dit **abélien**.

Soit g_i, un élément d'un groupe G. Pour $n \in \mathbb{Z}$, ng_i signifie

$$\underbrace{g_i * \cdots * g_i}_{n} \quad (\text{si } n > 0), \tag{A.1}$$

et

$$\underbrace{(-g_i) * \cdots * (-g_i)}_{n} \quad (\text{si } n < 0). \tag{A.2}$$

1. Cette définition est recopiée de [40].

Si $n = 0$, nous avons $0g_i = e$.

Prenons alors r éléments $g_1, ..., g_r \in G$. Les éléments de G de la forme

$$n_1 g_1 * \cdots * n_r g_r \quad (n_i \in \mathbb{Z}, 1 \leq i \leq r), \tag{A.3}$$

forment un sous-groupe de G noté par H. Les éléments $g_1, ..., g_r$ sont alors les **générateurs** du sous-groupe H. De plus, si $n_1 g_1 * \cdots * n_r g_r = e$ seulement lorsque $n_1 = \cdots = n_r = 0$, alors les générateurs sont **linéairement indépendants**. Par définition, un groupe abélien généré par r éléments linéairement indépendants est appelé **groupe abélien libre de rang r**.

Par exemple, l'ensemble des entiers $\mathbb{Z}$, avec l'addition usuelle $+$, est un groupe abélien libre de rang 1 généré par l'élément 1. En effet, la seule manière d'obtenir l'élément unité est si $n_1 = 0$ dans l'expression $n_1 1$. De plus, comme il n'y a qu'un seul générateur cela répond directement à la condition d'indépendance linéaire. De même, $\mathbb{Z} \times \mathbb{Z} = \{(x, y) | x, y \in \mathbb{Z}\}$, où $\times$ dénote donc le produit cartésien, est un groupe abélien libre de rang 2 dont les générateurs sont $(1, 0)$ et $(0, 1)$. L'opération $+$ dans ce groupe désigne alors l'addition des nombres qui sont aux mêmes positions dans chaque tuple $(a, b) + (c, d) = (a + c, b + d)$. De ce fait, pour obtenir $n_1(1, 0) + n_2(0, 1) = (0, 0)$, il faut que $n_1 = n_2 = 0$.

Un autre type de groupe qui nous intéresse correspond aux **groupes cycliques**. Un groupe G est cyclique s'il est généré par un élément g tel que $G = \{e, \pm g, \pm 2g, ...\}$. Si $ng \neq e$ pour tout $n \in \mathbb{Z} \setminus \{0\}$, il s'agit d'un groupe cyclique infini, tandis que si $ng = e$ avec $n \in \mathbb{Z} \setminus \{0\}$, il s'agit d'un groupe cyclique fini.

Applications

Un autre élément important à revoir est celui d'une **application**, aussi connue sous le nom de fonction. Une application est un processus f par lequel chaque élément d'un ensemble, disons X, est assigné à un élément dans l'ensemble Y. Ce processus est noté par

$$f : X \to Y. \tag{A.4}$$

Il est aussi possible d'écrire l'application f en faisant intervenir les éléments des ensembles et l'opération effectuée par f comme ceci

$$f : x \mapsto f(x), \tag{A.5}$$

où $x \in X$, $f(x) \in Y$. Ici $f()$ pourrait être la fonction exp() ou sin() par exemple.

Pour une application f donnée, il est possible de s'intéresser à l'image de f qui est définie comme

$$\text{im } f = \{y \in Y | y = f(x) \ \forall \ x \in X\} \subseteq Y. \tag{A.6}$$

Dans la même veine, nous définissons aussi le noyau (*kernel*) comme

$$\ker f = \{x \in X | f(x) = 0\}, \tag{A.7}$$

où 0 est l'élément neutre.

Ces applications comportent parfois des propriétés précises qui permettent de les caractériser davantage. En effet, une application f est **injective** si, pour deux éléments x et x' dans l'ensemble X, $x \neq x'$ implique que $f(x) \neq f(x')$. L'injectivité signifie alors qu'à chaque x est assigné un y qui lui est propre. Une autre caractéristique commune est la **surjectivité**, c'est-à-dire que pour chaque $y \in Y$, il existe au moins un élément $x \in X$ tel que $f(x) = y$.

Si ces deux caractéristiques sont attribuées à une application, cette dernière est qualifiée de **bijective**. Plus précisément, cela signifie que pour tout y dans Y correspond un x dans X tel que $y = f(x)$ et que $f(x)$, pour deux x différents, ne partagent pas le même y.

Ces derniers concepts permettent alors de définir des applications spéciales qui apparaissent fréquemment en topologie algébrique, soient l'**homomorphisme** et l'**isomorphisme**. Pour obtenir ces structures, il faut que les deux ensembles X et Y qui sont mis en relation soient chacun dotés d'une opération particulière, disons $\oplus$ pour X et $\otimes$ pour Y, faisant en sorte que $x \oplus x' \in X$ et $y \otimes y' \in Y$. L'application $f : X \to Y$ est alors qualifiée d'homomorphisme si

$$f(x \oplus x') = f(x) \otimes f(x'), \tag{A.8}$$

pour tous les x et x' appartenant à X. Si l'homomorphisme f est aussi bijectif, f est qualifié d'isomorphisme. S'il existe un isomorphisme f mettant en relation deux ensembles X et Y dotés d'une opération, on dit que X est isomorphe à Y et le tout est noté par $X \cong Y$.

Une autre relation importante qui apparaît dans les calculs des nombres de Betti est la **relation d'équivalence**, notée par le symbole $\sim$. Soient a, b et c, trois éléments d'un ensemble. On dit qu'il existe une relation d'équivalence si les critères suivants sont respectés :

1. Réflexion : $a \sim a$;
2. Symétrie : si $a \sim b$, alors $b \sim a$;
3. Transitivité : si $a \sim b$ et $b \sim c$, alors $a \sim c$.

Pour un ensemble donné, il reste alors à définir la règle qui permet d'écrire la relation d'équivalence.

En guise d'exemple, prenons l'ensemble des entiers $\mathbb{Z}$ et disons qu'il existe une relation d'équivalence entre les entiers qui, lorsqu'ils sont divisés par 2, donnent le même reste. Les nombres pairs sont alors tous équivalents entre eux, puisque leur reste est nécessairement 0. De la même manière, les nombres impairs sont aussi équivalents entre eux, mais cette fois-ci parce que leur reste est toujours 1.

Le dernier exemple permet d'attaquer deux autres concepts importants, c'est-à-dire les **classes d'équivalence** et les **quotients**. En effet, en prenant un ensemble X ainsi qu'une relation d'équivalence $\sim$ entre ces éléments, nous obtenons une partition de X en des sous-ensembles

disjoints appelés des classes d'équivalence. Plus précisément, une classe d'équivalence, notée $[a]$ est définie comme

$$[a] = \{x \in X | x \sim a\}. \tag{A.9}$$

Ici, a est alors appelé le représentant de la classe d'équivalence. Dans l'exemple précédent, le représentant de la classe d'équivalence des nombres pairs pourrait être 0 (ou n'importe quel autre nombre pair) et celui des nombres impairs pourrait être 1 (ou n'importe quel autre nombre impair). Nous avons donc la classe d'équivalence [0] qui regroupe tous les entiers pairs et la classe d'équivalence [1] qui regroupe tous les entiers impairs. L'union des deux classes d'équivalence [0] et [1] permet de réobtenir $\mathbb{Z}$.

L'ensemble des classes d'équivalence obtenues à partir d'une relation d'équivalence $\sim$ sur un ensemble X est appelé l'**espace quotient**, qui est noté par $X/\sim$. Ainsi, dans l'exemple où des entiers sont équivalents s'ils possèdent le même reste lorsque divisés par 2, nous avons l'espace quotient $\mathbb{Z}/\sim = \{[0], [1]\}$. De plus, il est possible de définir un isomorphisme entre $\mathbb{Z}/\sim$ et $\mathbb{Z}_2$, c'est-à-dire le groupe cyclique d'ordre 2 dont l'opération + est l'addition modulo 2, telle que $0+0=0$, $0+1=1+0=1$ et $1+1=0$. En effet, si nous définissons $\oplus$ dans $\mathbb{Z}/\sim$ comme étant l'addition modulo 2 des classes d'équivalence selon $[0]\oplus[0]=[0]$, $[0]\oplus[1]=[1]\oplus[0]=[1]$ et $[1]\oplus[1]=[0]$, alors nous sommes en mesure de définir l'application

$$f : \mathbb{Z}/\sim \to \ \mathbb{Z}_2, \tag{A.10}$$

qui envoie [0] vers 0 et [1] vers 1. L'application est donc bijective.

Nous pouvons alors démontrer que f est homomorphe comme ceci. Soient a et $b \in \mathbb{Z}/\sim$. Il y a deux résultats possibles à l'opération $\oplus$ selon les valeurs de a et b, c'est-à-dire $a \oplus b = [0]$, si $a = b$, et $a \oplus b = [1]$, si $a \neq b$. Alors, avec $a = b$, nous avons

$$\begin{aligned} f(a \oplus b) &= f([0]) \\ &= 0 \\ &= f(a) + f(a) \\ &= f(a) + f(b). \end{aligned} \tag{A.11}$$

De plus, si $a \neq b$, nous avons

$$\begin{aligned} f(a \oplus b) &= f([1]) \\ &= 1 \\ &= f(a) + f(b). \end{aligned} \tag{A.12}$$

Ainsi,

$$f(a \oplus b) = f(a) + f(b), \tag{A.13}$$

pour tous les a et b dans $\mathbb{Z}/\sim$, prouvant ainsi que f est un homomorphisme. Cette application étant également bijective, f est aussi un isomorphisme et $\mathbb{Z}/\sim \cong \mathbb{Z}_2$.

Dans le contexte de l'homologie, nous nous intéresserons à des groupes construits à partir de quotients précis. Ces groupes sont appelés groupes d'homologie [2], mais l'appellation générale d'un groupe obtenu à partir d'un quotient est **groupe quotient**. Pour obtenir un groupe quotient, nous avons besoin d'un groupe G et d'un **sous-groupe normal** de G, noté H. Un sous-groupe est normal si $g * h * (-g) \in H$ pour tout $g \in G$ et $h \in H$. Si cette condition est respectée, nous pouvons construire un groupe quotient en munissant G d'une relation d'équivalence $\sim$ telle que, pour $g_i, g_j \in G$, nous avons $g_i \sim g_j$ s'il existe un $h \in H$ tel que $g_j = g_i * h$. De ce fait, nous écrivons plutôt l'espace quotient $G/\sim$ comme G/H. L'espace quotient G/H devient un groupe quotient s'il est muni d'une opération $\circ$ telle que $[g_i] \circ [g_j] = [g_i * g_j]$.

2. Voir la section 1.3.

Annexe B

Tables utilisées pour la figure 3.8

	$B=0$	$B=1$
$A=0$	63.69416035	15
$A=1$	10	11.30583965

	$B=0$	$B=1$
$A=0$	135.06281265	30
$A=1$	20	14.93718735

	$B=0$	$B=1$
$A=0$	206.49477294	45
$A=1$	30	18.50522706

	$B=0$	$B=1$
$A=0$	278.09251464	60
$A=1$	40	21.90748536

Tables utilisées pour la figure 3.9.

	$B=0$	$B=1$
$A=0$	42.62613217	15
$A=1$	25	17.37386783

	$B=0$	$B=1$
$A=0$	93.4870287	30
$A=1$	50	26.5129713

	$B=0$	$B=1$
$A=0$	144.48792507	45
$A=1$	75	35.51207493

	$B=0$	$B=1$
$A=0$	195.71877794	60
$A=1$	100	44.28122206

Tables utilisées pour la figure 3.10

	$B=0$	$B=1$
$A=0$	7.51280963	20
$A=1$	15	57.48719037

	$B=0$	$B=1$
$A=0$	13.42317634	40
$A=1$	30	116.57682366

	$B=0$	$B=1$
$A=0$	19.12598277	60
$A=1$	45	175.87401723

	$B=0$	$B=1$
$A=0$	24.71979651	80
$A=1$	60	235.28020349

Annexe C

Taux de succès de tables $2 \times 2 \times 2$

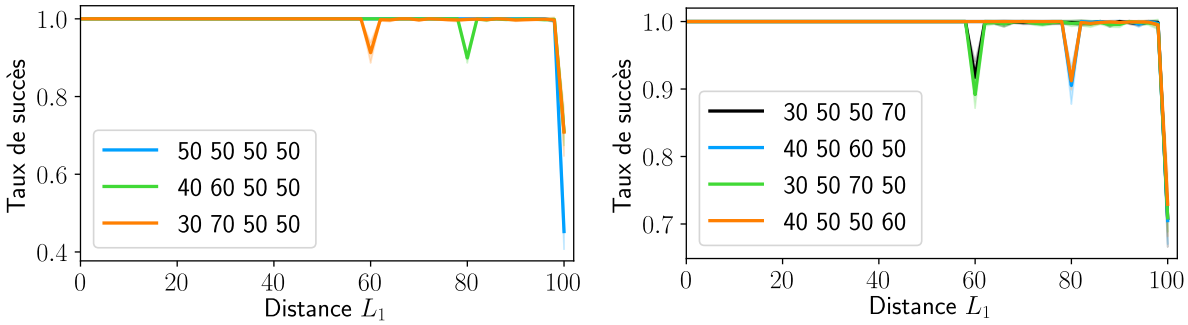

FIGURE C.1 – Taux de succès pour retrouver la dépendance en fonction des perturbations de tables modèles $2 \times 2 \times 2$ asymétriques. Les valeurs des entrées dans la légende correspondent aux variables x, y, z et w de la table 3.3.3 et représentent la table modèle.

En testant les tables asymétriques, les conclusions demeurent les mêmes que pour les tables 2×2, c'est-à-dire que la plus petite valeur dans la table de contingence limite la tolérance aux perturbations. Plus cette valeur minimale est basse, plus la valeur de L_1^{seuil} et de L_1^{max} sera basse comparativement à la table 3.3.3 où les entrées sont égales à 50.

Annexe D

Liste de facettes du complexe simplicial de la figure 4.6

Interaction	Valeur-p
$[0,1]$	8.02×10^{-48}
$[0,2]$	1.36×10^{-20}
$[0,3]$	1.521×10^{-28}
$[1,2]$	8.64×10^{-21}
$[1,3]$	9.83×10^{-28}
$[2,3]$	2.89×10^{-5}
$[3,4]$	3.13×10^{-3}
$[3,5]$	4.13×10^{-3}
$[4,5]$	3.30×10^{-3}
$[4,8]$	2.06×10^{-3}
$[5,6]$	3.62×10^{-3}
$[5,7]$	4.09×10^{-3}
$[6,7]$	8.01×10^{-3}
$[7,8]$	7.45×10^{-12}
$[7,9]$	1.94×10^{-4}
$[8,9]$	2.47×10^{-16}
$[0,1,2]$	1.13×10^{-6}
$[0,1,3]$	2.97×10^{-9}
$[0,2,3]$	9.37×10^{-11}
$[1,2,3]$	5.81×10^{-23}
$[7,8,9]$	9.11×10^{-16}

TABLE D.1 – Liste de facettes effective du complexe simplicial à 10 noeuds de la figure 4.13 et la valeur-p de leur table modèle.

Bibliographie

[1] Adjengue L., *Méthodes statistiques : Concepts, applications et exercices*, Presses internationales Polytechnique, 2014.

[2] Agresti A., *Categorical Data Analysis*, John Wiley & Sons, 2013.

[3] Baker R., Clarke M. et Lane P., *Zero entries in contingency tables*, Computational Statistics & Data Analysis, 3 (1985), p. 33.

[4] Barabási A.-L., *Network Science*, Cambridge University Press, 2016.

[5] Barlow R., *Extended maximum likelihood*, Nuclear Instruments and Methods in Physics Research, 297 (1990), p. 496.

[6] Bassett D. S. et Sporns O., *Network neuroscience*, Nature Neuroscience, 20 (2017), p. 353.

[7] Battiston F. et coll. , *Networks beyond pairwise interactions : Structure and dynamics*, Physics Reports, En impression (2020), https://doi.org/10.1016/j.physrep.2020.05.004.

[8] Bauer U., Kerber M., Reininghaus J. et Wagner H., *Phat – persistent homology algorithms toolbox*, Journal of Symbolic Computation, 78 (2017), p. 76.

[9] Bichet O., Dupuch A., Hébert C., Le Borgne H. et Fortin D., *Maintaining animal assemblages through single-species management : the case of threatened caribou in boreal forest*, Ecological Applications, 26 (2016), p. 612.

[10] Birch M. W., *Maximum likelihood in three-way contingency tables*, Journal of the Royal Statistical Society. Series B, 25 (1963), p. 220.

[11] Bishop Y. M. M., Fienberg S. E. et Holland P. W., *Discrete Multivariate Analysis : Theory and Applications*, Springer, 2007.

[12] Boissonat J.-D. et coll. , *GUDHI*, http://gudhi.gforge.inria.fr/, (2019).

[13] Boissonnat J.-D. et Maria C., *The simplex tree : An efficient data structure for general simplicial complexes*, Algorithmica, 70 (2014), p. 406.

[14] Brown M. et C. F., *On maximum likelihood estimation in sparse contingency tables*, Computational Statistics & Data Analysis, 1 (1983), p. 3.

[15] Christensen R., *Log-Linear Models and Logistic Regression*, Springer, 1997.

[16] Comte J., Lovejoy C., Crevecoeur S. et Vincent W. F., *Co-occurrence patterns in aquatic bacterial communities across changing permafrost landscapes*, Biogeosciences, 13 (2016), p. 175.

[17] Courtney O. T. et Bianconi G., *Generalized network structures : The configuration model and the canonical ensemble of simplicial complexes*, Phys. Rev. E, 93 (2016), p. 062311.

[18] Cramér H., *Mathematical Methods of Statistics*, Princeton University Press, 1946.

[19] Darroch J. N., Lauritzen S. L. et Speed T. P., *Markov fields and log-linear interaction models for contingency tables*, The Annals of Mathematical Statistics, 8 (1980), p. 522.

[20] DeGroot M. H. et Schervish M. J., *Probability and Statistics*, Pearson Education, 2012.

[21] Fienberg S. E., *An iterative procedure for estimation in contingency tables*, The Annals of Mathematical Statistics, 41 (1970), p. 907.

[22] ——, *The analysis of incomplete multi-way contingency tables*, Biometrics, 28 (1972), p. 177.

[23] ——, *The Analysis of Cross-Classified Categorical Data*, Springer, 2007.

[24] Fienberg S. E. et Rinaldo A., *Three centuries of categorical data analysis : Log-linear models and maximum likelihood estimation*, Journal of Statistical Planning and Inference, 137 (2007), p. 3430.

[25] Fosdick B. K., Larremore D. B., Nishimura J. et Ugander J., *Configuring random graph models with fixed degree sequences*, SIAM Review, 60 (2018), p. 315.

[26] Friedlander M., *Fitting log-linear models in sparse contingency tables using the emleloglin r package*, arXiv :1611.07505v2, (2016).

[27] Grilli J., Barabás G., Michalska-Smith M. J. et Allesina S., *Higher-order interactions stabilize dynamics in competitive network models*, Nature, 548 (2017), p. 210.

[28] Haberman S. J., *Log-linear models for frequency data : Sufficient statistics and likelihood equations*, The Annals of Statistics, 1 (1973), p. 617.

[29] ——, *The Analysis of Frequency Data*, University of Chicago Press, 1974.

[30] Hatcher A., *Algebraic Topology*, Cambridge University Press, 2002.

[31] Holt D., *Log-linear models for contingency table analysis : On the interpretation of parameters*, Sociological Methods & Research, 7 (1979), p. 330.

[32] Iacopo I., Petri G., Barrat A. et Latora V., *Simplicial models of social contagion*, Nature Communications, 10 (2019), p. 2485.

[33] Kastrin A., Rindflesch T. C. et Hristovski D., *Large-scale structure of a network of co-occurring mesh terms : Statistical analysis of macroscopic properties*, PLOS ONE, 9 (2014), p. e102188.

[34] Koller D. et Friedman N., *Probabilistic Graphical Models : Principles and Techniques*, MIT Press, 2009.

[35] Kroonenberg P. M., *Applied Multiway Data Analysis*, John Wiley & Sons, 2008.

[36] Lambiotte R., Rosvall M. et Scholtes I., *From networks to optimal higher-order models of complex systems*, Nat. Phys., 15 (2019), p. 313.

[37] Latora V., Nicosia V. et Russo G., *Complex Networks*, Cambridge University Press, 2017.

[38] Lyons L., Allison W. W. M. et Pañella Comellas J., *Maximum likelihood or extended maximum likelihood ? an example from high energy physics*, Nuclear Instruments and Methods in Physics Research, 297 (1986), p. 530.

[39] Manfredi S., Di Tucci E. et Latora V., *Mobility and congestion in dynamical multilayer networks with finite storage capacity*, Phys. Rev. Lett., 120 (2018), p. 068301.

[40] Mathieu P., *Méthodes mathématiques en physique : Notes de cours*, Université Laval, 2017.

[41] Mischaikow K. et Nanda V., *Morse theory for filtrations and efficient computation of persistent homology*, Discrete & Computational Geometry, 50 (2013), p. 330.

[42] Morozov D., *Dionysus 2*, `https://github.com/mrzv/dionysus`, (2019).

[43] Mukhopadhyay P., *Complex Surveys : Analysis of Categorical Data*, Springer, 2016.

[44] Mézard M. et Montanari A., *Information, Physics and Computation*, Oxford University Press, 2009.

[45] Nakahara M., *Geometry, Topology and Physics*, Institute of Physics Publishing, 2003.

[46] Nelder J. A., *A reformulation of linear models*, Journal of the Royal Statistical Society : Series A (General), 140 (1977), p. 48.

[47] Newman M., *Networks : An Introduction*, Oxford University Press, 2010.

[48] ——, *Computational Physics*, Createspace Independent Pub, 2013.

[49] Orsini C. et coll. , *Quantifying randomness in real networks*, Nature Communications, 6 (2015), p. 8627.

[50] Patania A., Petri G. et Vaccarino F., *The shape of collaborations*, EPJ Data Science, 6 (2017), p. e102188.

[51] Petitjean F., Allison L. et Webb G., *A statistically efficient and scalable method for log-linear analysis of high-dimensional data*, 2014 IEEE International Conference on Data Mining, (2014), p. 480.

[52] Petitjean F. et Webb G., *Scaling log-linear analysis to datasets with thousands of variables*, SIAM International Conference on Data Mining, (2015), p. 469.

[53] Petitjean F., Webb G. et Nicholson A. E., *Scaling log-linear analysis to high-dimensional data*, 2013 IEEE 13th International Conference on Data Mining, (2013), p. 597.

[54] Rinaldo A., *Computing Maximum Likelihood Estimates in Log-Linear Models*, Rapport technique, https://kilthub.cmu.edu/articles/Computing_Maximum_Likelihood_Estimates_in_Log-Linear_Models/6586505/1, Carnegie Mellon University, 2005.

[55] Rinaldo A. et Fienberg S., *Maximum likelihood estimation in log-linear models*, The Annals of Statistics, 40 (2012), p. 996.

[56] Rudas T., *Lectures on Categorical Data Analysis*, Springer, 2018.

[57] Searle S. R. et Gruber M. H. J., *Linear Models*, John Wiley & Sons, 2017.

[58] Young J.-G., *C++ MCMC sampler for the simplicial configuration model*, https://github.com/jg-you/scm, (2017).

[59] Young J.-G., Petri G., Vaccarino F. et Patania A., *Construction of and efficient sampling from the simplicial configuration model*, Phys. Rev. E, 96 (2017), p. 032312.