THESE

Pour l'obtention du grade de Docteur délivré par

L'UNIVERSITE DE POITIERS

Ecole Doctorale Mathématiques, Informatique, Matériaux, Mécanique, Energétique

Spécialité : Mathématiques et leurs interactions

présentée par

Alandra Zakkour

Classification à partir d'une classe de q-distributions de probabilités

Directeur de Thèse : **Yousri Slaoui**

Codirecteur de Thèse : **Cyril Perret**

Présentée et soutenue le 10 décembre 2024

COMPOSITION DU JURY

Rapporteurs :	**Mustapha Rachdi**	PR, Université Grenoble Alpes
	Anne Françoise Yao	PR, Université Clermont Auvergne
Examinateurs :	**Abdallah El Hamidi**	MCF-HDR, Université de La Rochelle
	Ali Gannoun	PR, Université de Montpellier
	Solym Manou-Abi	MCF-HDR, Université de Poitiers
	Elias Ould Said	PR, Université Littoral Côte d'Opale
	Cyril Perret	MCF, Université de Poitiers
	Yousri Slaoui	MCF-HDR, Université de Poitiers

Dédicace

Aux meilleurs parents du monde, Jacques Zakkour et Nawal Azar,

C'est avec fierté et gratitude que je vous dédie ce travail. Votre soutien inconditionnel et vos grands sacrifices ont été la principale motivation pour parvenir à cet objectif. Je vous remercie tous les deux du fond du cœur pour chaque moment d'amour, de soins et de confiance que vous m'avez accordé. Sans vous, ce rêve n'aurait pas été possible.

"When a father gives to his son, both laugh ; when a son gives to his father, both cry." - William Shakespeare

Je vous aime et vous apprécie pour toujours.

Votre fille,

Alandra

Remerciements

Je tiens à exprimer ma profonde gratitude à mes directeurs de thèse, Dr. Yousri Slaoui et Dr. Cyril Perret, pour la confiance qu'ils m'ont accordée, ainsi que pour leur aide et leur soutien tout au long de ces trois années de recherche. Leur encadrement attentif et leur disponibilité constante ont été essentiels à la réussite de ce travail. Un grand merci pour avoir toujours pris soin de m'assurer d'être bien installé. Je suis honorée d'avoir eu l'opportunité de collaborer avec vous deux !

Je souhaite exprimer ma profonde gratitude à la Professeure Anne Françoise Yao et au Professeur Mustapha Ruchdi pour l'honneur qu'ils m'ont fait en acceptant d'être les rapporteurs de ma thèse. Je remercie également le Professeur Elias Ould Said, le Professeur Ali Gannoun, ainsi que le Docteur Abdallah El Hamidi et le Docteur Solym Manou-Abi pour avoir accepté de faire partie de mon jury.

Je tiens à exprimer ma profonde gratitude à M. Thierry Olive et à Mme Caroline Bordes, du laboratoire CeRCA à Poitiers, pour leur précieuse collaboration dans le cadre du modèle de mélange présenté au chapitre 5. Je souhaite également remercier M. Jean Dumoncel, avec qui j'ai eu le plaisir de collaborer pour la publication du package "fitexgauss" en langage R. Par ailleurs, je remercie ma collègue Imen Badrani, ainsi que son directeur de thèse de l'Université de Sfax, en Tunisie, pour notre collaboration fructueuse sur le développement du q-calcul, présenté dans le chapitre 6. Enfin, je remercie Mme Christine Ros et M. David Chesnet, également du CeRCA, pour leur collaboration dans la réalisation de la partie d'application présentée dans ce même chapitre.

Merci à tous les membres du Laboratoire de Mathématiques et Applications (LMA) et au Centre de Recherche sur la Congnition et l'Apprentissage (CeRCA). Un grand merci au directeur du LMA, M. Boris Pasquier et à la directrice du CeRCA, Mme Christel Bidet-Ildei pour m'avoir acceuillie.
A Mme Nathalie Mongin, Mme Jocelyne Attab, Mme Aurore Lecoustre, Mme Valerie Meynet et M. Philippe Augustin, merci pour votre aide et vos services tout au long de ces années.

Je remercie chaleureusement tous mes collègues pour leur gentillesse, et tout particulièrement Rim Mheich et Zeina Rammal pour leur soutien moral et leurs sourires, qui ont été précieux, notamment durant les moments stressants.

Un grand merci à tous les Poitevins. Merci, Poitiers, pour les beaux souvenirs et d'avoir été mon lieu de résidence pendant ces années.
Merci à toutes les personnes que j'ai rencontrées à Poitiers, vous êtes toutes adorables. Merci d'avoir été ma famille à l'étranger. Merci pour votre soutien constant, pour les moments agréables, les sorties et les sourires.

Avec fierté, je voudrais m'adresser à ma famille :

A mes chères soeurs Amanda et Santa, et à mon cher frère Johnny : Merci pour vos encouragement, votre support et votre amours. Merci pour les rires de fraternité et les blagues spéciales avec vous.
A ma merveilleuse maman Nounni, mille mercis ne suffisent pas. Merci pour tous tes sacrifices et ton soutien constant, même par appel vidéo. Merci pour les moments où nous avons pleuré ensemble au téléphone, chacune de notre côté, lors de mes périodes de faiblesse, de détresse et les moments où tu m'as manqué. Aujourd'hui, c'est ton rêve qui se réalise, maman, c'est ton rêve, Nounni !
A mon cher papa Jicko : merci de toujours nous offrir ce que nous souhaitons. Je te promets de toujours te rendre fier de moi.
Ma promesse pour vous, Jicko et Nounni : je mettrai toujours un sourire sur vos visages.
A nos deux princesses de la famille, mes nièces Mary-Jane et Liana, vous êtes la lumière de nos vies. Vous posséderez toujours mon cœur, et vous resterez à jamais mes petites.

A tous mes amis, en France et au Liban, un grand merci.

A mon pays d'accueil, la France : MERCI !
A mon pays d'origine, le Liban : MERCI !

Je désire terminer ces remerciements avec une pensée inspirante d'Albert Einstein : "La vie, c'est comme une bicyclette : il faut avancer pour ne pas perdre l'équilibre."

Je vous souhaite à tous une bonne lecture !

Productions Scientifiques

Pendant ces trois années de thèse, les travaux ont conduit à :

Des articles scientifiques :

- **Un article publié :**
  Zakkour, A., Perret, C. and Slaoui, Y. Stochastic Expectation Maximization Algorithm for Linear Mixed-Effects Model with Interactions in the Presence of Incomplete Data, Entropy, 25 (2023). DOI: 10.3390/e25030473.
- **Une mise à jour du package "ExGaussEstim" dans R :**
  ExGaussEstim : Quantile Maximization Likelihood Estimation and Bayesian Ex-Gaussian Estimation. Yousri Slaoui, Abir El Haj, Alandra Zakkour, Caroline Bordes, Cyril Perret, Jean Dumoncel.
  DOI: 10.32614/CRAN.package.ExGaussEstim.
- **Un article sous révision :**
  Alandra Zakkour, Cyril Perret and Yousri Slaoui, Handling Missing Data and Parameters Initialization Method for the Ex-Gaussian Distribution, Journal of Applied Statistics.
- **Un article soumis :**
  Alandra Zakkour, Cyril Perret and Yousri Slaoui, Handling Missing Data for the Ex-Gaussian Distribution Using Data Augmentation algorithm, Sankhya, The Indian Journal of Statistics.
- **Un package soumis dans R, nommé "fitexgauss" :**
  Alandra Zakkour, Yousri Slaoui, Cyril Perret and Jean Dumoncel.
- **Deux articles en cours de préparation :**
  — Alandra Zakkour, Yousri Slaoui, Caroline Bordes, Thierry Olive, Cyril Perret, Using finite mixture models to model the inter-keystroke interval distribution : explorations of several alternatives.
  — Imen Badrani, Mondher Damak, Cyril Perret, Yousri Slaoui and Alandra Zakkour, The Ex-Gaussian distribution using q-calculus.

Des conférences scientifiques :

- 55e Journées de Statistique de la SFdS à l'Université de Bordeaux, France, 2024.
- 54e Journées de Statistique de la SFdS à l'Université libre de Bruxelles, Belgique, 2023.

Des rencontres scientifiques :

- MARGAUx PhD Days, au Laboratoire de Mathématiques et de leurs Applications de Pau (LMAP), Université de Pau, France, 2024.
- Séminaire des doctorants au Laboratoire de Mathématique et Applications (LMA), Université de Poitiers, France, 2024.
- Apprentissage Statistiques et intelligence artificielle ASIA 2023 à l'Université de Poitiers, France, 2023.

Résumé

La recherche sur les processus cognitifs, notamment ceux liés à la production écrite, exige une analyse rigoureuse des temps de réaction. L'un des défis majeurs réside dans la gestion des données manquantes. Nous avons abordé ce problème en utilisant un Modèle Linéaire à Effets Mixtes (`LMEM`), couramment employé dans les études cognitives, auquel nous avons intégré un algorithme stochastique basé sur la méthode de l'Espérance-Maximisation (`EM`), plus précisément l'algorithme d'Espérance-Maximisation Stochastique (`SEM`). Bien que la distribution Ex-Gaussienne (Ex-Gauss) soit avantageuse pour modéliser les temps de réaction, l'obtention d'estimations fiables reste complexe en raison des biais liés aux valeurs aberrantes. Pour y remédier, nous avons eu recours à l'inférence bayésienne et proposé la méthode Bootstrap pour améliorer l'initialisation des paramètres. De plus, nous avons mis en place une méthode d'augmentation des données afin de pallier le problème des données manquantes, particulièrement en l'absence d'une grande base de données. Nous avons exploré les modèles de mélange pour analyser les pauses dans la production dactylographiée, en comparant divers scénarios pour déterminer le nombre de mélanges, leur nature (additif ou multiplicatif), la distribution la plus appropriée (Ex-Gauss, Ex-Wald ou Log-Normale), ainsi que l'impact des variables externes sur l'estimation des modèles. Enfin, nous avons ouvert la voie à l'utilisation du calcul quantique (q-calcul) en remplacement du calcul classique pour explorer la densité Ex-Gauss, nommée q-Ex-Gauss. Nous avons obtenu la forme générale de cette densité, qui servira ensuite de fonction de base pour analyser les temps de réaction, afin de déterminer si le q-calcul peut fournir des estimations plus efficaces que le calcul classique.

Mots clés. Temps de réaction ; Pauses ; Modèle Linéaire à Effets Mixtes ; Données manquantes ; Algorithme d'Espérance-Maximisation Stochastique ; Estimations ; Distribution Ex-Gauss ; Inférence Bayésienne ; Initialisation ; Bootstrap ; Augmentation des données ; Modèle de mélange ; q-calcul ; q-Ex-Gauss.

Abstract

Research on cognitive processes, particularly those related to written production, requires a thorough analysis of reaction times. One of the main challenges is managing missing data. We addressed this issue using a Linear Mixed-Effects Model (`LMEM`), commonly applied in cognitive studies, to which we incorporated a stochastic algorithm based on the Expectation-Maximization (`EM`) method, specifically the Stochastic Expectation-Maximization (`SEM`) algorithm. Although the Ex-Gaussian distribution (Ex-Gauss) is advantageous for modeling reaction times, obtaining reliable parameter estimates remains complex due to biases caused by outliers. To address this, we employed Bayesian inference and proposed the Bootstrap method to improve parameter initialization. Additionally, we implemented a data augmentation method to mitigate the issue of missing data, especially when a large dataset

is not available. We also explored mixture models to analyze pauses in written production, comparing different scenarios to determine the number of mixtures, their nature (additive or multiplicative), the most suitable distribution (Ex-Gauss, Ex-Wald, or Log-Normal), and the impact of external variables on model estimation. Finally, we paved the way for the use of quantum computing (q-calculus) as an alternative to classical computing in exploring the Ex-Gauss density, referred to as q-Ex-Gauss. We derived the general form of this density, which will serve as a base function to analyze reaction times, in order to assess whether q-calculus can lead to more efficient estimates than classical computing.

Keywords. Reaction times ; Pauses ; Linear Mixed-Effects Model ; Missing data ; Stochastic Expectation-Maximization Algorithm ; Estimations ; Ex-Gauss distribution ; Bayesian Inference ; Initialization ; Bootstrap ; Data Augmentation ; Mixture Model ; q-calculus ; q-Ex-Gauss.

Table des matières

1 Introduction Générale **7**

2 Algorithme d'espérance-maximisation stochastique pour modèle linéaire à effets mixtes avec interactions en présence de données incomplètes **11**

2.1 Introduction 11
2.2 Le modèle proposé 14
2.2.1 L'approche de Henderson 18
2.2.2 Des cas particuliers 18
2.3 Données Incomplètes 20
2.3.1 Données censurées 20
2.3.2 Données manquantes 21
2.4 `SEM` et ses concurrents 27
2.4.1 Application dans un `LMEM` 29
2.5 Convergence des paramètres 32
2.6 Application numérique 33
2.6.1 Données simulées 33
2.6.2 Données réelles 39
2.7 Conclusion et Perspectives 39
Code source en R 41

3 Traitement des données manquantes et méthode d'initialisation des paramètres pour la distribution Ex-Gaussienne **45**

3.1 Introduction 45
3.2 Initialisation des paramètres 47
3.2.1 Distribution Ex-Gauss 48
3.2.2 Initialisation par Bootstrap en Estimation Bayésienne 49
3.3 Application Numérique 54
3.3.1 Données simulées 54
3.3.2 Données réelles 65
3.4 Conclusion et Perspective 67
Code source en R 69

4 Gestion des données manquantes pour la distribution Ex-Gaussienne en utilisant la méthode d'augmentation de données 71

4.1 Introduction . . . 71

4.2 Méthodologie . . . 72

4.2.1 Méthode de `MCMC` . . . 73

4.2.2 Algorithme `DA` proposé . . . 74

4.3 Application numérique . . . 79

4.3.1 Conclusion et Perspective . . . 84

Code source en R . . . 87

5 Modèles de mélange additifs et multiplicatifs sous différentes distributions 89

5.1 Introduction . . . 89

5.2 Modèle de mélange . . . 90

5.2.1 Modèle de mélange additif . . . 93

5.2.2 Modèle de mélange multiplicatif . . . 93

5.2.3 Estimation du modèle . . . 94

5.3 Distributions . . . 96

5.3.1 Distribution de Wald . . . 96

5.3.2 Distribution Ex-Wald . . . 97

5.3.3 Distribution Log-Normale . . . 98

5.4 Sélection du modèle . . . 99

5.5 Application numérique . . . 100

5.6 Conclusion . . . 103

Code source en R . . . 105

6 Distribution q-Ex-Gaussienne 109

6.1 Introduction . . . 109

6.2 Notions de q-calcul . . . 110

6.2.1 Opération de q-addition . . . 110

6.2.2 Opération de q-multiplication . . . 111

6.2.3 Fonction q-exponentielle . . . 112

6.3 Les principaux résultats . . . 113

6.3.1 Distribution q-Exponentielle . . . 113

6.3.2 Distribution q-Gaussienne . . . 116

6.3.3 Distribution q-Ex-Gauss 119
6.4 Application sur des données réelles 122
6.5 Conclusion 124

7 Conclusion Générale **125**

Bibliographie **127**

Chapitre 1

Introduction Générale

L'écriture est un processus cognitif complexe qui mobilise une coordination fine entre le cerveau et le corps, dépassant largement la simple transcription d'idées sur un support physique. Chaque étape, de la planification à la transcription, implique une gestion subtile des ressources cognitives, favorisant une production fluide du texte. Comme l'ont souligné [98], les avancées théoriques et méthodologiques dans l'étude de l'écriture montrent que l'orthographe et la rédaction de mots nécessitent une interaction constante entre des processus linguistiques et moteurs. Les pauses, souvent perçues comme des interruptions, jouent un rôle crucial dans l'organisation des pensées et la gestion cognitive. En réalité, ces moments permettent aux écrivains d'anticiper la suite du texte et de préparer la transcription des idées. Par exemple, [1] ont démontré que les pauses préparent les étapes suivantes de la rédaction, tandis que [32] ont montré que les pauses intra- et inter-mots reflètent des phases distinctes du traitement linguistique.

La gestion des pauses dans l'écriture a été analysée par [3] et [48], qui soulignent que ces moments sont des phases de réflexion où l'esprit traite la complexité de la tâche. L'analyse des frappes au clavier ([6]) permet d'observer en temps réel les processus cognitifs sous-jacents à l'écriture, éclairant la manière dont la charge cognitive est répartie pendant la rédaction.

L'étude des temps de réaction, initiée par [40], a également contribué à la compréhension des processus cognitifs rapides associés à l'écriture. [9] ont approfondi cette approche en examinant l'influence de la fréquence des mots et de la lexicalité sur la reconnaissance, révélant des dynamiques cachées dans le traitement lexical. Pour modéliser ces variations, des modèles statistiques avancés comme les distributions Ex-Gaussiennes ont été utilisés ([110]), ainsi que des modèles bayésiens pour une estimation plus robuste des processus cognitifs complexes ([46]).

L'algorithme Espérance-Maximisation (`EM`), introduit par [37], est l'une des méthodes clés pour estimer les paramètres des modèles complexes, notamment dans le traitement des données manquantes, un problème fréquent dans les études cognitives. Des approches complémentaires, comme les modèles de mélange gaussien, ont permis de mieux comprendre la diversité des comportements cognitifs et moteurs dans l'écriture ([21], [48]).

L'analyse des pauses cognitives offre également des informations précieuses sur les processus de planification et de révision en cours. [51] ont démontré que la position des pauses dans un texte prédit leur durée, reflétant ainsi la complexité cognitive associée à chaque étape de la rédaction. L'utilisation de modèles bayésiens et d'approches robustes, comme l'estimation par maximum de vraisemblance ([92]), améliore la précision des modèles tout en réduisant la sensibilité aux anomalies.

Loin d'être une simple tâche mécanique, l'écriture engage une multitude de processus cognitifs et moteurs. Grâce à l'intégration de modèles statistiques, tels que les distributions Ex-Gaussiennes et les modèles de mélange, il devient possible de mieux comprendre les interactions entre cognition et motricité, ainsi que l'impact des pauses et des temps de réaction sur la production écrite. Ces recherches fournissent une base solide pour approfondir l'analyse des processus cognitifs sous-jacents à l'écriture et pour élaborer des modèles plus adaptés à la complexité de cette tâche.

Un des défis majeurs dans l'analyse des temps de réaction lors de la production écrite est la gestion des données manquantes. L'algorithme Expectation-Maximization (`EM`) est largement utilisé pour estimer les paramètres en ajustant les données manquantes de manière itérative ([37]). [100] ont proposé des techniques pour traiter les valeurs aberrantes et les données manquantes dans les temps de réaction, suggérant des méthodes de filtrage pour améliorer la précision des analyses. L'imputation multiple, introduite par [107], est également couramment utilisée pour gérer les données manquantes, réduisant ainsi les biais d'estimation.

Dans le chapitre 2, nous avons développé un modèle linéaire à effets mixtes en tenant compte des interactions fixes-fixes et fixes-aléatoires, en proposant l'algorithme Espérance-Maximisation Stochastique (`SEM`) pour traiter le problème des données manquantes.

En psychologie cognitive, les temps de réaction sont fréquemment modélisés par une distribution Ex-Gaussienne, qui combine une distribution normale pour les réponses rapides et une distribution exponentielle pour les réponses lentes. Cette approche, courante dans le domaine ([83]), permet de décomposer les temps de réaction en trois paramètres : la moyenne (μ) et l'écart-type (σ) de la composante gaussienne, et le paramètre tau (τ) capturant la queue exponentielle. Dans le chapitre 3, nous avons estimé les temps de réaction suivant une distribution Ex-Gauss en utilisant l'inférence bayésienne, tout en introduisant la méthode Bootstrap comme point de départ pour la prédiction des réponses.

Dans le chapitre 4, nous avons étendu cette méthode avec une approche d'augmentation des données pour améliorer l'estimation des temps de réaction sous distribution Ex-Gauss et réduire le biais lié aux données manquantes.

Le chapitre 5 explore l'hypothèse que les pauses représentent différentes activités cognitives au sein d'un modèle de mélange. Nous y examinons le nombre d'activités, le type de mélange, et les facteurs externes influençant ces pauses.

Dans le chapitre 6, nous introduisons le calcul quantique dans nos recherches pour déterminer la forme générale de la densité de la distribution q-Ex-Gauss. Le calcul quantique, également appelé q-calcul, est une généralisation des calculs classiques, où le paramètre q introduit un facteur de déformation dans les opérations algébriques standards ([119]). Ce type de calcul permet d'ajuster la manière dont certaines propriétés, comme la symétrie ou la linéarité, sont gérées dans les distributions statistiques, en fournissant un cadre plus flexible pour modéliser des phénomènes complexes. En appliquant cette approche au modèle Ex-Gauss, nous visons à tester

si cette distribution offre une estimation plus efficace des pauses cognitives comparée aux méthodes classiques.

Chapitre 2

Algorithme d'espérance-maximisation stochastique pour modèle linéaire à effets mixtes avec interactions en présence de données incomplètes

Dans ce chapitre, nous avons présenté une méthode stochastique pour résoudre le problème des données non observées. Nous avons utilisé un modèle psychologique de type linéaire à effets mixtes, en tenant compte des interactions entre les variables fixes ainsi que des interactions entre les variables fixes et aléatoires. Les résultats de ce chapitre ont donné lieu à la publication de l'article suivant, réalisé en collaboration avec mes directeurs de thèse, M. Yousri Slaoui et M. Cyril Perret : Zakkour, A. ; Perret, C. ; Slaoui, Y. *Stochastic Expectation Maximization Algorithm for Linear Mixed-Effects Model with Interactions in the Presence of Incomplete Data*, Entropy, 25 (2023), pp. 473.

2.1 Introduction

Comme toutes les disciplines scientifiques, la psycholinguistique s'appuie sur des tests d'hypothèses pour étayer ses propositions théoriques. Depuis le début des années 2000, les chercheurs ont adopté les modèles linéaires à effets mixtes (Linear Mixed-Effects Models, `LMEM`). Tel que décrit par [7], un `LMEM` permet de prendre en compte correctement l'une des caractéristiques des expériences psycholinguistiques : la présence de deux variables à effets aléatoires. En effet, la structure expérimentale des expériences menées dans ce domaine implique qu'un groupe de participants traite un ensemble de stimuli (voir ci-après). Ainsi, l'analyse statistique doit permettre l'inclusion de ces deux variables à effets aléatoires, à savoir les participants et les items, dans la structure du modèle. L'introduction du `LMEM` a donc constitué une avancée méthodologique importante pour la psycholinguistique.

Rappelons tout d'abord que le `LMEM` ([38]) est une extension du modèle linéaire simple qui permet de représenter les effets fixes et aléatoires. En 1861, Airy a présenté pour la première fois un modèle à effets aléatoires à un facteur unique (c'est-à-dire un modèle avec une variable aléatoire et sans variable fixe), qui est devenu plus tard connu sous le nom de modèle à effets aléatoires, et ce concept a été approfondi par Scheffé en 1956 ([128]). À partir de 1990, [128] a souligné que les `LMEM`s sont devenus populaires dans de nombreuses applications de recherche, notamment l'économie,

la sociologie, l'assurance, l'agronomie, l'épidémiologie, la génétique, etc. Ils sont utilisés pour l'analyse des données longitudinales, la modélisation multiniveaux et les petites estimations. L'analyse de ces modèles est présentée dans [70] ; pour plus de documentation sur la méthodologie, les résultats théoriques et les logiciels, voir les livres [47, 69, 87, 113]. `LMEM`s sont ajustés et analysés dans `R` en utilisant le package `lme4` ou `lmertest` ([14]).

Les interactions jouent un rôle clé dans l'ajustement des `LMEM`s pour évaluer leur influence conjointe sur la réponse. Dans [23], l'auteur a présenté les interactions fixes entre deux facteurs dans un `LMEM`, en utilisant le test F de l'interaction du produit maximal. [129] s'est intéressé à la taille de l'échantillon d'un `LMEM`. Il a proposé une formule pour estimer la taille de l'échantillon en testant un modèle mixte contenant des interactions fixes. En 2019, [104] a proposé une méthode d'estimation pour récupérer les effets principaux et les interactions, car la méthode existante ne permettait pas l'intégration de ces effets dans un cadre de données mixtes. Ils ont approuvé le fait que la méthode proposée donnait des résultats optimaux aux conditions de leurs applications. Dans le même temps, [55] a présenté l'estimation des interactions fixes. Ces interactions sont normalement introduites par le produit des variables, mais les transformations algébriques ont révélé que cette technique ne produisait pas d'estimateur intra-unité. Une méthode de Monte Carlo a confirmé que l'estimateur `FE` (effets fixes) de (x, z) était biaisé si l'une de ces variables était corrélée avec une autre. Pour présenter l'interaction entre x et z, il est possible d'utiliser la syntaxe actuelle $x * z$. Cette considération est appelée "double-demeaned" ; elle est moins efficace que la FE standard et ne fonctionne qu'avec $T < 2$. Pour l'application dans `R`, la méthode chaîne de Markov Monte Carlo (Markov Chain Monte Carlo, `MCMC`) est appliquée en utilisant le package `mcmc`. [86] a présenté le package `MCMCpack` qui contient des fonctions pour effectuer l'inférence bayésienne en utilisant une simulation a posteriori pour un certain nombre de modèles statistiques (`https://CRAN.R-project.org/package=MCMCpack`), tandis que [57] presente le package `MCMCglmm` (`https://cran.r-project.org/web/packages/MCMCglmm/index.html`) pour la méthode `MCMC` d'ajustement des modèles mixtes linéaires généralisés pour les vecteurs à réponses multiples. Cette méthode a également été utilisée par [76] pour identifier des paramètres inconnus dans le domaine biologique, comme la détection de la concentration de molécules cibles, en raison de l'importance de cette méthode pour l'extraction des informations. La méthode MCMC a été appliquée dans les travaux de [94] pour résoudre des problèmes mécaniques à l'aide d'une méthode bayésienne.

Il y a cependant un point qui n'a pas reçu beaucoup d'attention de la part des psycholinguistes. L'expérimentation avec des êtres humains est souvent sujette à de nombreux aléas. Imaginons des chercheurs dont le but est de comprendre comment un adulte récupère l'orthographe d'un mot dans sa mémoire ([99]). Ils demandent à un groupe de trente participants d'écrire le nom de 150 images présentées sur un écran d'ordinateur. Un dispositif leur permet de mesurer le temps entre l'apparition de l'image sur l'écran et le premier geste d'écriture du participant. Ces chercheurs peuvent potentiellement collecter 4500 valeurs de `RT`. Toutefois, entre les essais per-

dus pour des raisons techniques, les erreurs d'orthographe, et certaines habitudes lors de l'analyse des données (par ex, censure des données supérieures à deux écarts types de la moyenne, censure des données à droite), un nombre non négligeable de données sont supprimées. Par exemple, [99] a indiqué avoir supprimé un peu plus de 20% des données.

La question des données manquantes et censurées a reçu relativement peu d'attention en psycholinguistique (voir cependant [100, 122]). C'est particulièrement vrai depuis l'introduction des `LMEM`. Les psycholinguistes supposent que ces modèles peuvent être exécutés sur un échantillon de données avec des "trous".
La stratégie d'analyse est alors du type "garder la case vide", en ignorant le biais que cela introduit dans l'estimation des paramètres du modèle et donc des décisions prises. L'objectif de ce travail est donc de développer des algorithmes pour gérer la présence de données manquantes et censurées dans ces expériences psycholinguistiques. Deux points sont à prendre en compte. D'une part, les plans d'expériences en psychologie scientifique peuvent impliquer des hypothèses sur les interactions entre deux ou plusieurs variables à effet fixe. D'autre part, les chercheurs suggèrent, pour des raisons théoriques, que toutes les interactions entre les variables à effet fixe et les variables à effet aléatoire rendues possibles par le modèle devraient être incluses ([13]). La présence potentielle de ces deux types d'interactions (variables fixes–fixes ; variables aléatoires-fixes) sont des contraintes sur le développement de la procédure de données manquantes procédure.

L'une des solutions qui gère les données manquantes peut être l'algorithme Espérance-Maximisation (Expectation-Maximization algorithm, `EM`). Il s'agit d'un algorithme très utile pour l'estimation de la fonction de maximum de vraisemblance. Cette méthode peut être une solution lorsque les seules données disponibles ne permettent pas d'estimer les paramètres ([37]), et/ou que l'expression de la vraisemblance est analytiquement impossible à maximiser.

En d'autres termes, il vise à fournir un estimateur lorsque le problème vient de la présence de données manquantes. Lorsque les données sont incomplètes, la connaissance de ces valeurs permettrait d'estimer les paramètres. L'algorithme `EM` tire son nom du fait qu'à chaque itération, il opère deux étapes distinctes :

Etape E : est appelée étape d'espérance (**E**), où l'on cherche à trouver la valeur attendue des variables non observées ou inconnues compte tenu des données observées et de la valeur des paramètres.

Etape M : Cette étape est appelée étape de maximisation (**M**) ; dans cette étape, nous maximisons la log-vraisemblance attendue en utilisant l'estimation des données inconnues réalisée à l'étape précédente. Ces estimations des paramètres sont ensuite utilisées pour déterminer la distribution des variables inconnues dans l'itération suivante.

A certains moments, les étapes d'espérance ou de maximisation sont impossibles à appliquer directement ([36]) ; à partir de là, l'utilisation d'une forme d'extension de `EM` est utile, comme `MCEM`, où l'étape (E) est remplacée par une simulation de Monte

Carlo, ou SAEM, où l'étape (E) est remplacée par une Approximation Stochastique. SAEM était une solution de modèles non linéaires à effets mixtes (NLMEM). [77] a proposé une nouvelle méthodologie pour l'estimation du maximum de vraisemblance dans les mélanges de modèles non linéaires à effets mixtes. L'algorithme MSAEM (mélange SAEM) qui en résulte est maintenant mis en œuvre dans l'outil logiciel Monolix. Le but de ce chapitre était d'exécuter l'algorithme SEM dans le cadre d'un LMEM, en incluant les données incomplètes et en prenant en considération, pour la première fois, les interactions. L'algorithme SEM, tel que proposé par [115], a été appliqué à des modèles hiérarchiques à intercepts aléatoires avec données censurées, où ils ont démontré son efficacité en surpassant les méthodes traditionnelles. Dans notre travail, nous proposons une extension de cet algorithme en intégrant les interactions entre les variables fixes (interaction fixe-fixe) ainsi que celles entre les variables fixes et aléatoires (interaction fixe-aléatoire), afin de mieux capter la complexité des données dans un cadre plus général.

Ce chapitre est structuré comme suit : dans la section 2.2, nous présentons notre modèle psycho-statistique. La section 2.3 détaille les types de données incomplètes, à savoir les données manquantes et les données censurées. La section 2.4 décrit trois algorithmes basés sur la méthode de maximisation des espérances (EM) et leur application dans ce chapitre : le premier est l'algorithme SAEM, le deuxième est notre proposition, l'algorithme SEM, et le troisième repose sur la méthode MCMC. La section 2.5 présente une méthode pour atteindre la convergence. Dans la section 2.6, nous comparons les résultats obtenus à partir de données psychologiques simulées et de données réelles. Enfin, la section 2.7 conclut l'étude en discutant des perspectives et des orientations futures.

2.2 Le modèle proposé

[58] a comparé cinq types de modèle linéaire à effets mixtes afin de prouver l'importance de prendre les effets maximaux. Ils ont présenté les deux variables aléatoires sujet (i.e., participants), conçue par S, et éléments (i.e., items), conçue par I.
En considérant i comme l'indice des items, s comme l'indice des sujets et c comme l'observation, le premier modèle était le modèle maximal qui contenait les variables fixes et aléatoires avec les interactions, y compris également ρ_S et ρ_I :

$$Y_{c,i,s} = \beta_0 + S_{0,s} + I_{0,i} + (\beta_1 + S_{1,s} + I_{1,i})X_c + \epsilon_{c,i,s},$$

$$\begin{pmatrix} S_{0,s} \\ S_{1,s} \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} \tau_{00}^2 & \rho_S \tau_{00} \tau_{11} \\ \rho_S \tau_{00} \tau_{11} & \tau_{11}^2 \end{pmatrix} \right),$$

$$\begin{pmatrix} I_{0,i} \\ I_{1,i} \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} w_{00}^2 & \rho_I w_{00} w_{11} \\ \rho_I w_{00} w_{11} & w_{11}^2 \end{pmatrix} \right),$$

$$\epsilon_{c,i,s} \sim \mathcal{N}(0, \sigma^2).$$

Le second modèle diffère par ses paramètres de corrélation, où $S_{0,s}$ et $S_{1,s}$ (respectivement $I_{0,i}$ et $I_{1,i}$) sont indépendants. La valeur de ρ_S (respectivement ρ_I) a été fixée à zéro :

$$Y_{c,i,s} = \beta_0 + S_{0,s} + I_{0,i} + (\beta_1 + S_{1,s} + I_{1,i})X_c + \epsilon_{c,i,s}, \tag{2.1}$$

$$\begin{pmatrix} S_{0,s} \\ S_{1,s} \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} \tau_{00}^2 & 0 \\ 0 & \tau_{11}^2 \end{pmatrix}\right),$$

$$\begin{pmatrix} I_{0,i} \\ I_{1,i} \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} w_{00}^2 & 0 \\ 0 & w_{11}^2 \end{pmatrix}\right),$$

$$\epsilon_{c,i,s} \sim \mathcal{N}(0, \sigma^2).$$

Dans le troisième modèle, ils ont réduit l'équation (2.1) en ignorant les pentes aléatoires des items (l'interaction entre les effets fixes et aléatoires des items), où w_{11} a été fixé à 0 :

$$Y_{c,i,s} = \beta_0 + S_{0,s} + I_{0,i} + (\beta_1 + S_{1,s})X_c + \epsilon_{c,i,s},$$

$$\begin{pmatrix} S_{0,s} \\ S_{1,s} \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} \tau_{00}^2 & 0 \\ 0 & \tau_{11}^2 \end{pmatrix}\right),$$

$$I_{0,i} \sim \mathcal{N}(0, w_{00}^2),$$

$$\epsilon_{c,i,s} \sim \mathcal{N}(0, \sigma^2).$$

Le quatrième modèle conserve la pente aléatoire des items et exclut celle du sujet, où $\tau_{11} = 0$:

$$Y_{c,i,s} = \beta_0 + S_{0,s} + I_{0,i} + (\beta_1 + I_{1,i})X_c + \epsilon_{c,i,s},$$

$$S_{0,s} \sim \mathcal{N}(0, \tau_{00}^2),$$

$$\begin{pmatrix} I_{0,i} \\ I_{1,i} \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} w_{00}^2 & 0 \\ 0 & w_{11}^2 \end{pmatrix}\right),$$

$$\epsilon_{c,i,s} \sim \mathcal{N}(0, \sigma^2).$$

Le cinquième modèle excluait les deux pentes aléatoires (pour les items et le sujet), où $\tau_{11} = w_{00} = 0$:

$$Y_{c,i,s} = \beta_0 + S_{0,s} + I_{0,i} + \beta_1 X_c + \epsilon_{c,i,s},$$

$$S_{0,s} \sim \mathcal{N}(0, \tau_{00}^2),$$

$$I_{0,i} \sim \mathcal{N}(0, w_{00}^2),$$

$$\epsilon_{c,i,s} \sim \mathcal{N}(0, \sigma^2).$$

Motivés par des données psychologiques, dans lesquelles nous avons généralement deux variables aléatoires, les participants désignés par S et les éléments (items) désignés par I, nous considérons un modèle qui contient p covariables $\{x_1; x_2; \ldots; x_p\}$ et ces deux variables aléatoires. La variable prédite, également appelée la variable d'intérêt, est désignée par Y. Puis, nous considérons le modèle linéaire mixte suivant :

$$Y = \beta_{0,s,i} + \beta_{1,s,i}x_1 + \beta_{2,s,i}x_2 + \ldots + \beta_{p,s,i}x_p + r_{j,i,s}, \tag{2.2}$$

avec r le résidu du modèle qui suit une distribution normale. Nous donnons :

$$\begin{cases} \beta_{0,s,i} = \delta_0 + S_{0,s} + I_{0,i} \\ \beta_{1,s,i} = \delta_1 + S_{1,s} + I_{1,i} \\ \beta_{2,s,i} = \delta_2 + S_{2,s} + I_{2,i} \\ \vdots \\ \beta_{p,s,i} = \delta_p + S_{p,s} + I_{p,i}. \end{cases}$$

En remplaçant ces considérations dans l'équation (4.6), notre modèle peut être réécrit comme suit :

$$\begin{aligned} Y = {} & \delta_0 + \delta_1 x_1 + \delta_2 x_2 + \ldots + \delta_p x_p \\ & + S_{0,s} + S_{1,s} x_1 + S_{2,s} x_2 + \ldots + S_{p,s} x_p \\ & + I_{0,i} + I_{1,i} x_1 + I_{2,i} x_2 + \ldots + I_{p,i} x_p \\ & + r. \end{aligned}$$

En prenant

$$X = \begin{bmatrix} 1 & x_1 & x_2 & \ldots & x_{p+k} \end{bmatrix} = \begin{bmatrix} 1 & x_j \end{bmatrix}, \delta = \begin{pmatrix} \delta_0 & \delta_j \end{pmatrix}^T$$

et en désignant par T la matrice transposée,

$$Z = \begin{bmatrix} z_j \end{bmatrix} = \begin{bmatrix} x_j \end{bmatrix}, u = \begin{bmatrix} u_j \end{bmatrix} = \begin{bmatrix} S_{j,l,s} + I_{j,l,i} \end{bmatrix}^T ;$$

$$\text{et } e_{jl} = \begin{bmatrix} e_j \end{bmatrix}.$$

Cela nous amène à réécrire notre modèle comme suit :

$$Y = X\delta + \sum_{k=1}^{K} z_k u_k + e = X\delta + Zu + e,$$

où $\mathbb{E}(Y) = X\delta$ et $\mathrm{Var}(Y) = \mathbb{V} = ZGZ^T + R$, avec $u \sim \mathcal{N}(0, G)$, $e \sim \mathcal{N}(0, R)$ et $\mathrm{Cov}(u, e^T) = 0$.

Maintenant, en incluant toutes les interactions, le modèle général peut être écrit comme suit :

$$Y = \mathrm{F} + \mathrm{I_{FF}} + \mathrm{R} + \mathrm{I_{FR}} + e,$$

où nous prenons :

$$\begin{aligned}
\mathrm{F} &= \underbrace{\delta_0 + \delta_1 x_1 + \ldots + \delta_p x_p}_{\text{variables fixes}}, \\
\mathrm{I_{FF}} &= \underbrace{\delta_{p+1} x_{p+1} + \ldots + \delta_{p+k} x_{p+k}}_{\text{l'interaction entre les variables fixes}}, \\
\mathrm{R} &= \underbrace{(S_{1,s} x_1 + I_{1,i} x_1) + \ldots + (S_{p,s} x_p + I_{p,i} x_p)}_{\text{variables aléatoires}}, \\
\mathrm{I_{FR}} &= \underbrace{(S_{p+1,s} x_{p+1} + I_{p+1,i} x_{p+1}) + \ldots + (S_{p+k,s} x_{p+k} + I_{p+k,i} x_{p+k})}_{\text{l'interaction entre les variables fixes et aléatoires}}, \\
\text{and } e &= \underbrace{(S_{0,s} + I_{0,i}) + r}_{\text{résidus}}.
\end{aligned}$$

Nous fixons k, le nombre d'interactions, à $k = 2^p - p - 1$ et laissons $j, l \in [1, p+k]$. Notre variable aléatoire participants est la suivante :

$$S_{j,l,s} \sim \mathcal{N}(0, \Sigma_S),$$

et la matrice de variance-covariance des participants s'écrit comme suit

$$\Sigma_S = \begin{cases} \sigma^2_{j,S} & ; \text{si } j = l \\ \rho_S \sigma_{j,S} \sigma_{l,S} & ; \text{sinon.} \end{cases}$$

Nous considérons que la variable aléatoire items suit également une distribution normale :

$$I_{j,l,I} \sim \mathcal{N}(0, \Sigma_I);$$

alors, la matrice variance-covariance des items est donnée par :

$$\Sigma_I = \begin{cases} \sigma^2_{j,I} & ; \text{si } j = l \\ \rho_I \sigma_{j,I} \sigma_{l,I} & ; \text{sinon.} \end{cases}$$

Pour $j, l \in [1, p+k]$, la matrice de variance-covariance G est égale à :

$$G = \Sigma_u = \begin{cases} \sigma^2_{j,S} + \sigma^2_{j,I} & ; \text{si } j = l \\ \rho_S \sigma_{j,S} \sigma_{l,S} + \rho_I \sigma_{j,I} \sigma_{l,I} & ; \text{sinon.} \end{cases}$$

La matrice R est de la forme :

$$R = \begin{cases} I & ; \text{si } j = l \leq n - p - k \\ \sigma^2_{e_j} & ; \text{si } j = l > n - p - k \\ 0 & ; \text{sinon,} \end{cases}$$

où n représente le nombre total d'observations. Par conséquent, avec $Z = (z_1, \ldots, z_{p+k})$, la variance de Y est égale à :

$$\begin{aligned}\mathbb{V} &= \Bigg[(\sum_{j=1}^{p+k} z_j\Sigma_{u_1,u_j})z_1 + (\sum_{j=1}^{p+k} z_j\Sigma_{u_2,u_j})z_2 + \ldots + (\sum_{j=1}^{p+k} z_j\Sigma_{u_p,u_j})z_p + \ldots \\ &\quad +(\sum_{j=1}^{p+k} z_j)\Sigma_{u_{p+k},u_j} z_{p+k}\Bigg] + R.\end{aligned}$$

L'objectif est de prédire notre modèle en déterminant les paramètres δ et u ; nous proposons d'utiliser le système linéaire de Henderson ([50]) basé sur la fonction de maximum de vraisemblance.

2.2.1 L'approche de Henderson

Pour estimer les paramètres $\hat{\delta}$ et $\hat{u}$, nous devons maximiser la fonction logarithmique $(\hat{\delta}, \hat{u}) = \arg\max_{\delta,u} \ln f(Y,u)$. Nous définissons $f(Y,u) = f(Y|u)f(u)$ avec $Y|u \sim \mathcal{N}(X\delta + Zu, R)$ et $u \sim \mathcal{N}(0,G)$, donc nous pouvons écrire :

$$\begin{aligned}-2\ln f(Y|u) &= N\ln(2\pi) + ln \mid R \mid +(Y - X\delta - Zu)^{'}R^{-1}(Y - X\delta - Zu),\\ -2\ln f(u) &= q\ln(2\pi) + \ln \mid G \mid +u^{'}G^{-1}u.\end{aligned}$$

Désignons par $l(\delta, u; Y) = \ln f(Y,u)$; pour maximiser cette fonction, nous devons annuler les dérivées premières de $\ln f(Y,u)$ par rapport à δ et u :

$$\begin{aligned}\frac{\partial[-2l(\delta,u;y)]}{\partial\delta} &= -2X^{'}R^{-1}(y - X\delta - Zu) = 0,\\ \frac{\partial[-2l(\delta,u;y)]}{\partial u} &= -2Z^{'}R^{-1}(y - X\delta - Zu) + 2G^{-1}u = 0.\end{aligned} \tag{2.3}$$

A partir des deux équations dans (5.2.3), Henderson a présenté le modèle linéaire mixte sous le système :

$$\begin{bmatrix}X^{'}R^{-1}X & X^{'}R^{-1}Z\\ Z^{'}R^{-1}X & Z^{'}R^{-1}Z + G^{-1}\end{bmatrix}\begin{bmatrix}\hat{\delta}\\ \hat{u}\end{bmatrix} = \begin{bmatrix}X^{'}R^{-1}y\\ Z^{'}R^{-1}y\end{bmatrix} \tag{2.4}$$

Dans la section suivante, nous montrons comment le modèle peut être simplifié si nous prenons les deux cas suivants : premièrement, seule la partie de l'interaction fixe-fixe est présentée et deuxièmement, il n'y a pas d'interaction.

2.2.2 Des cas particuliers

Cas 1 : Interaction Fixe-Fixe

Dans ce cas, on considère les interactions entre les variables fixes. Ainsi, dans la partie aléatoire, le vecteur u est égal à 0 :

$$\begin{aligned}u &= \begin{bmatrix}u_1 & u_2 \ldots u_p & u_{p+1} \ldots u_{p+k}\end{bmatrix}^T\\ &= \begin{bmatrix}0 & 0 & \ldots & 0 & 0 & \ldots & 0\end{bmatrix}^T;\end{aligned}$$

Par conséquent, la matrice $\mathbb{V}$ est égale à :

$$\mathbb{V} = R = \begin{cases} I & ; \text{si } j = l \leq n-p-k \\ \sigma^2_{e_j} & ; \text{si } j = l > n-p-k \\ 0 & ; \text{sinon.} \end{cases}$$

Les matrices du système linéaire de Henderson sont simplifiées comme suit, où r détermine les lignes, c détermine les colonnes, et $l, h \in [1, p+k]$:

$$(X^{'}R^{-1}X)_{M\times N} = \begin{cases} (n-p-k) + \sigma^{-2}_{e_1} + \ldots + \sigma^{-2}_{e_p} + \sigma^{-2}_{e_{p+1}} + \ldots + \sigma^{-2}_{e_{p+k}} & ; \text{si } r = c = 1 \\ \sum_{j=1}^{n-p-k} x_{l,j} + \sum_{o=1}^{p+k} x_{l,n-p-k+o}\sigma^{-2}_{e_o} & ; \text{si } r = 1; c \neq 1 \\ \sum_{j=1}^{n-p-k} x_{l,j}x_{h,j} + \sum_{o=1}^{p+k} x_{l,n-p-k+o}x_{h,n-p-k+o}\sigma^{-2}_{e_o} & ; \text{sinon.} \end{cases}$$

$$(X^{'}R^{-1}y)_{M\times 1} = \begin{cases} \sum_{j=1}^{n-p-k} y_j + \sum_{o=1}^{p+k} y_{n-p-k+o}\sigma^{-2}_{e_o} & ; \text{si } r = 1 \\ \sum_{j=1}^{n-p-k} x_{l,j}y_j + \sum_{o=1}^{p+k} x_{l,n-p-k+o}y_{n-p-k+o}\sigma^{-2}_{e_o} & ; \text{sinon.} \end{cases}$$

$$(X^{'}R^{-1}Z)_{M\times N} = \begin{cases} \sum_{j=1}^{n-p-k} z_{l,j} + \sum_{o=1}^{p+k} z_{l,n-p-k+o}\sigma^{-2}_{e_o} & ; \text{si } r = 1 \\ \sum_{j=1}^{n-p-k} x_{l,j}z_{h,j} + \sum_{o=1}^{p+k} x_{l,n-p-k+o}z_{h,n-p-k+o}\sigma^{-2}_{e_o} & ; \text{sinon.} \end{cases}$$

$$(Z^{'}R^{-1}Z)_{M\times N} = \begin{cases} \sum_{j=1}^{n-p-k} z^2_{l,j} + \sum_{o=1}^{p+k} z^2_{l,n-p-k+o}\sigma^{-2}_{e_o} & ; \text{si } r = c \\ \sum_{j=1}^{n-p-k} z_{l,j}z_{h,j} + \sum_{o=1}^{p+k} z_{l,n-p-k+o}z_{h,n-p-k+o}\sigma^{-2}_{e_o} & ; \text{sinon.} \end{cases}$$

$$(Z^{'}R^{-1}y)_{M\times 1} = \left[\sum_{j=1}^{n-p-k} z_{l,j}y_j + \sum_{o=1}^{p+k} z_{l,n-p-k+o}y_{n-p-k+o}\sigma^{-2}_{e_o}\right].$$

Cas 2 : Pas d'interactions

Dans ce cas, il n'y a pas d'interactions entre les variables ; notre modèle est simplifié, la partie aléatoire avec l'interaction fixe-fixe étant ignorée :

$$X = \begin{bmatrix} 1 & x_j \end{bmatrix} ; \delta = \begin{pmatrix} \delta_0 & \delta_j \end{pmatrix}^T ; Z = \begin{bmatrix} z_j \end{bmatrix} ; u = \begin{bmatrix} u_j \end{bmatrix} = \begin{bmatrix} 0 & 0 & \ldots & 0 \end{bmatrix}^T ; e_{jl} = \begin{bmatrix} e_j \end{bmatrix}.$$

La matrice $\mathbb{V}$ est réduite à :

$$\mathbb{V} = R = \begin{cases} I & ; \text{si } j = l \leq n-p \\ \sigma^2_{e_j} & ; \text{si } j = l > n-p \\ 0 & ; \text{sinon.} \end{cases}$$

Alors, toutes les matrices du système linéaire sont simplifiées pour atteindre p au lieu de $p+k$. Pour tout $l, h \in [1, p]$:

$$(X^{'}R^{-1}X)_{M\times N} = \begin{cases} (n-p) + \sigma^{-2}_{e_1} + \ldots + \sigma^{-2}_{e_p} & ; \text{si } r = c = 1 \\ \sum_{j=1}^{n-p} x_{l,j} + \sum_{o=1}^{p} x_{l,n-p+o}\sigma^{-2}_{e_o} & ; \text{si } r = 1, c \neq 1 \\ \sum_{j=1}^{n-p} x_{l,j}x_{h,l} + \sum_{o=1}^{p} x_{l,n-p+o}x_{h,n-p+o}\sigma^{-2}_{e_o} & ; \text{sinon.} \end{cases}$$

$$(X^{'}R^{-1}y)_{M\times 1} = \begin{cases} \sum_{j=1}^{n-p} y_j + \sum_{o=1}^{p} y_{n-p+o}\sigma_{e_o}^{-2} & ; \text{si } r = 1 \\ \sum_{j=1}^{n-p} x_{l,j}y_j + \sum_{o=1}^{p} x_{l,n-p+o}y_{n-p+o}\sigma_{e_o}^{-2} & ; \text{sinon.} \end{cases}$$

$$(X^{'}R^{-1}Z)_{M\times N} = \begin{cases} \sum_{j=1}^{n-p} z_{l,j} + \sum_{o=1}^{p} z_{l,n-p+o}\sigma_{e_o}^{-2} & ; \text{si } r = 1 \\ \sum_{j=1}^{n-p} x_{l,j}z_{h,j} + \sum_{o=1}^{p} x_{l,n-p+o}z_{h,n-p+o}\sigma_{e_o}^{-2} & ; \text{sinon.} \end{cases}$$

$$(Z^{'}R^{-1}Z)_{M\times N} = \begin{cases} \sum_{j=1}^{n-p} z_{l,j}^2 + \sum_{o=1}^{p} z_{l,n-p+o}^2\sigma_{e_o}^{-2} & ; \text{si } r = c \\ \sum_{j=1}^{n-p} z_{l,j}z_{h,j} + \sum_{o=1}^{p} z_{l,n-p+o}z_{h,n-p+o}\sigma_{e_o}^{-2} & ; \text{sinon.} \end{cases}$$

$$(Z^{'}R^{-1}y)_{M\times 1} = \left[\sum_{j=1}^{n-p} z_{l,j}y_j + \sum_{o=1}^{p} z_{l,n-p+o}y_{n-p+o}\sigma_{e_o}^{-2}\right].$$

Le but de ce chapitre est de résoudre le problème des données incomplètes dans un modèle linéaire à effets mixtes (`LMEM`) en présence d'interactions fixe-fixe et fixe-aléatoire. Ainsi, dans la section suivante, nous présentons ces données afin de résoudre le problème de leur présence.

2.3 Données Incomplètes

On peut parler de données incomplètes lorsque les valeurs de notre vecteur de réponse ne sont pas toutes observées pour de nombreuses raisons ; on peut également dire qu'il s'agit d'une question sans réponse.

Ces types de données constituent un problème courant reconnu par les statisticiens. [17] a présenté de nombreuses solutions pour traiter ce problème, basées sur des imputations simples et multiples. De plus, [115] a proposé de traiter les données incomplètes dans un modèle hiérarchique en utilisant l'algorithme `SEM`.

Dans ce chapitre, nous nous intéressons aux données qui contiennent deux types de valeurs incomplètes : les données censurées désignées par Y_c et les données manquantes Y_{miss}.

Alors, nous examinons ces deux types de données dans les deux sections suivantes.

2.3.1 Données censurées

Les données censurées sont des données pour lesquelles nous ne connaissons pas l'heure exacte de l'événement.
Il existe deux types de données censurées : les données censurées à droite et les données censurées à gauche.

Données censurées à droite : il s'agit de données pour lesquelles l'événement n'a pas encore eu lieu lorsque l'étude est terminée.
Données censurées à gauche : il s'agit de données où l'événement est survenu avant le début de l'étude.

Nous considérons ici les deux types de censure des données à droite et à gauche. Soit t le niveau de censure et T l'ensemble des indices censurés qui peut être écrit sous la forme :

$$T = \{(i_1, \ldots, i_{p+k}) \in I, Y_{i_1,\ldots,i_{p+k},j} < t \text{ ou } Y_{i_1,\ldots,i_{p+k},j} > t\},$$

et laissons $Y = (Y_o, Y_c)$, avec $Y_o = \{Y_{i_1,\ldots,i_{p+k},j}, (i_1, \ldots, i_{p+k}) \notin T\}$ le vecteur des variables de réponse observées et $Y_c = \{Y_{i_1,\ldots,i_{p+k},j}, (i_1, \ldots, i_{p+k}) \in T\}$ définissant le vecteur des variables de réponse censurées.

Dans les études psycholinguistiques, les chercheurs éliminent souvent les valeurs extrêmes des temps de réponse en conservant uniquement celles qui se situent dans un intervalle défini par la moyenne plus ou moins un certain nombre d'écarts-types, comme 2,5 écarts-types ($\mu \pm 2.5 \times \sigma$), afin d'exclure les réponses aberrantes ou inappropriées.

Le deuxième type des données incomplètes est celui des données manquantes, défini dans la sous-section suivante.

2.3.2 Données manquantes

En général, les valeurs manquantes sont produites lorsqu'une valeur dans les données n'est pas représentée pour une variable donnée, pour de nombreuses raisons qui peuvent être liées à l'objectif de l'étude (par exemple, les participants ne répondent pas aux questions). Cette situation se présente dans de nombreuses études de recherche, en particulier lors de la collecte de données et lorsque les participants sont étudiés sur une certaine période de temps.

Dans un premier temps, les études ont été élaborées en supposant qu'il n'y avait pas de valeurs manquantes. A la fin des années 1980, avec les progrès technologiques, ce problème a attiré l'attention de nombreux chercheurs qui souhaitaient étudier plusieurs techniques pour le traiter. Selon les raisons de leur absence, ces valeurs peuvent être divisées en trois types : valeurs manquantes complètement au hasard (Missing Completely At Random, `MCAR`), valeurs manquantes au hasard (Missing At Random, `MAR`) et valeurs manquantes non au hasard (Missing Not At Random, `MNAR`).

Dans la partie suivante, nous présentons les différents types de données manquantes et la manière de les générer à l'aide de diverses méthodes.

Types de données manquantes et Imputations

Soit M l'indicateur de données manquantes qui définit ce qui est connu et ce qui est manquant. La variable de réponse est désignée par Y, où $Y = \{Y_o, Y_{no}\}$ avec Y_o

la partie observée et Y_{no} la partie non observée. [109] désigne la partie manquante par Y_{miss}. Considérons un ensemble de données de p variables et n sujets, avec $Y = \{Y_1, \ldots, Y_p\}$. Y devrait ressembler à une matrice de n lignes et p colonnes. Dans la littérature sur les données manquantes, voir aussi [43, 80].

Nous montrons dans cette section les trois types de données manquantes et comment les générer dans une étude de simulation basée sur une règle dans chaque cas ([132]) :

— `MCAR` : Le type de valeur manquante au hasard est un cas particulier du type `MAR` présenté après dans cette section. Dans ce type, la valeur est manquée par hasard et l'absence n'est pas liée au sujet, de sorte que la distribution de M ne dépend pas de Y_o ou de Y_{miss} et est identique pour toutes les observations :

$$\mathbb{P}(M|Y) = \mathbb{P}(M),$$

L'avantage statistique des données `MCAR` est que l'analyse reste impartiale, aucune des variables n'étant affectée plus qu'une autre.

Supposons que chaque sujet ait une probabilité π d'être absent d'une variable, la règle des données manquantes est donnée par $\mathbb{P}(M = 1) = \pi$. À partir de cette règle, nous pouvons déterminer diverses propriétés associées aux données `MCAR`, telles que le pourcentage attendu de valeurs manquantes et le nombre attendu de modèles de données manquantes dans les données `MCAR`.

Si nous travaillons avec un modèle univarié, représentons n le nombre de sujets dans les données et K la variable aléatoire indiquant le nombre de sujets avec des valeurs manquantes dans les données qui suit une distribution binomiale donnée par $K \sim B(n, \pi)$, où $0 \leq \pi \leq 1$. Puisque $\mathbb{E}(K) = n\pi$ et $\mathbb{V}(K) = n\pi(1-\pi)$, le pourcentage attendu de valeurs manquantes est :

$$\begin{aligned}\mathbb{E}(\Pi) &= \frac{1}{n}\mathbb{E}(K) \\ &= \pi,\end{aligned}$$

où $\Pi = \frac{K}{n}$ est la variable aléatoire qui définit le pourcentage estimé de valeurs manquantes dans un échantillon. La variance de ce pourcentage estimé est donnée par :

$$\begin{aligned}\mathbb{V}(\Pi) &= \frac{1}{n^2}\mathbb{V}(K) \\ &= \frac{\pi(1-\pi)}{n}.\end{aligned}$$

Dans ce cas, nous avons deux motifs de données manquantes, le motif 1 inclut les sujets avec des données complètes ; le motif 2 inclut les sujets avec des données manquantes. Soit I_j la variable indicatrice de l'événement, où le motif j est présent chez au moins un sujet de l'échantillon avec $j \in \{1, 2\}$. La probabilité que le motif 1 soit présent chez au moins un sujet est $\mathbb{P}(I_1 =$

$1) = \mathbb{E}(I_1) = 1 - \pi^n$. La probabilité que le motif 2 soit présent chez au moins un sujet est $\mathbb{P}(I_2 = 1) = \mathbb{E}(I_2) = 1 - (1 - \pi)^n$. Soit D le nombre de motifs de données manquantes distincts, $D = \sum_{j=1}^{2} I_j$; nous pouvons déterminer la valeur attendue :

$$\begin{aligned} \mathbb{E}(D) &= \sum_{j=1}^{2} \mathbb{E}(I_j) \\ &= (1 - \pi^n) + (1 - (1 - \pi)^n) \\ &= 2 - \pi^n - (1 - \pi)^n. \end{aligned} \tag{2.5}$$

Par conséquent, dans les données `MCAR` qui contiennent plus de deux motifs de données manquantes, le nombre attendu de motifs distincts dans un échantillon est donné par :

$$\mathbb{E}(D) = m - \sum_{j=1}^{m} (1 - \eta_j)^n,$$

avec $\eta_1, \ldots, \eta_m$ étant les probabilités correspondantes pour les motifs $1, \ldots, m$, et dépendant uniquement de la probabilité de valeurs manquantes dans chaque variable i (c'est-à-dire $\pi_1, \ldots, \pi_l$). $m = 2^l$ est le nombre de motifs et l est le nombre total de variables. Généralement, la méthode la plus utilisée pour implémenter le type `MCAR` consiste à supprimer aléatoirement le pourcentage souhaité de valeurs manquantes.

— `MAR` : Les données manquantes au hazard sont le cas le plus classique ; [106] ont défini les données manquantes comme étant `MAR` si la distribution de ces données ne dépendait pas de Y_{miss} :

$$\mathbb{P}(M|Y) = \mathbb{P}(M|Y_o).$$

Ce type de données se produisent lorsque l'absence peut être expliquée par les variables pour lesquelles des informations complètes existent. Plus généralement, nous sommes dans ce type si la cause des valeurs manquantes n'est pas liée à l'objectif de l'étude. Si Y_1 a des valeurs manquantes, alors elle est régressée sur les autres variables : Y_2 à Y_l. Les valeurs manquantes de Y_1 sont alors remplacées par les valeurs prédictives obtenues. De même, si Y_2 a des valeurs manquantes, alors les variables Y_1, Y_3 à Y_p sont utilisées dans le modèle de prédiction comme variables indépendantes. Ainsi, les valeurs manquantes sont remplacées par les valeurs prédites. Les règles de données manquantes pour le type `MAR` peuvent être organisées en plusieurs catégories : la méthode du seuil unique, la méthode des seuils multiples, la méthode des percentiles et la méthode de régression logistique.

Ici, nous définissons ces quatres règles.

Méthode du seuil unique : Considérons Y_1 comme la variable avec des valeurs manquantes et Y_2 comme le prédicteur avec un point de coupure a. Si

un sujet a $Y_2 \geq a$, alors sa probabilité d'avoir une valeur manquante dans Y_1 est π_1, et si $Y_2 < a$, alors sa probabilité d'avoir une valeur manquante dans Y_1 est π_2. Soit U un indicateur qui prend la valeur 1 ($U = 1$) lorsque $Y_2 \geq a$ et $U = 0$ lorsque $Y < a$. La règle des données manquantes est donnée par :

$$\mathbb{P}(M = 1|U = 1) = \pi_1 \, ; \, \mathbb{P}(M = 1|U = 0) = \pi_2,$$

où les paramètres associés à cette règle sont π_1 et π_2, et π_0 est la probabilité que Y_2 soit égal ou supérieur au seuil : $\mathbb{P}(Y_2 \geq a) = \mathbb{P}(U = 1) = \pi_0$. Pour trouver le pourcentage attendu de valeurs manquantes, nous calculons d'abord la probabilité inconditionnelle qu'un sujet ait des valeurs manquantes dans Y_1 :

$$\begin{aligned}\pi_{miss} &= \mathbb{P}(M = 1) \\ &= \mathbb{P}(M = 1|U = 1)\mathbb{P}(U = 1) + \mathbb{P}(M = 1|U = 0)\mathbb{P}(U = 0) \\ &= \pi_1\pi_0 + \pi_2(1 - \pi_0).\end{aligned}$$

Le pourcentage attendu de valeurs manquantes dans un échantillon est donné par :

$$\begin{aligned}\mathbb{E}(\Pi) &= \pi_{miss} \\ &= \pi_1\pi_0 + \pi_2(1 - \pi_0),\end{aligned}$$

et la variance de ce pourcentage estimé de valeurs manquantes est écrite comme :

$$\mathbb{V}(\Pi) = \frac{\pi_{miss}(1 - \pi_{miss})}{n}.$$

Méthode des seuils multiples : Un avantage de cette méthode est qu'elle peut être utilisée pour créer une relation non linéaire entre l'indicateur de données manquantes et leur prédicteur. Pour créer cette relation non linéaire, nous devons fixer un seuil supérieur et un autre inférieur. Supposons que nous fixons les deux points de coupure a et $-a$, $U = 1$ lorsque $Y_2 \geq a$ ou $Y_2 \leq -a$, et $U = 0$ lorsque $-a < Y_2 < a$. La règle des données manquantes peut être écrite comme :

$$\mathbb{P}(M = 1|U = 1) = \pi_1 \, ; \, \mathbb{P}(M = 1|U = 0) = \pi_2, \tag{2.6}$$

et nous pouvons voir que cette règle de données manquantes est la même que celle de la méthode du seuil unique. Dans le cas d'une relation linéaire, nous devons spécifier deux ou plusieurs points de coupure dans la prédiction. Supposons que nous prenons les points quartiles (Q_1, Q_2 et Q_3) comme points de coupure, soit V une variable aléatoire discrète uniforme créée en fonction des valeurs de Y_2 et prenant les valeurs suivantes :

$$V = \begin{cases} 1 & \text{if } Y_2 < Q_1 \\ 2 & \text{if } Q_1 \leq Y_2 < Q_2 \\ 3 & \text{if } Q_2 \leq Y_2 < Q_3 \\ 4 & \text{if } Y_2 \geq Q_3. \end{cases} \tag{2.7}$$

Ensuite, pour le cas de relation linéaire, la règle des données manquantes est donnée par :

$$\mathbb{P}(M = 1|V = 1) = \pi_1\,;\, \mathbb{P}(M = 1|V = 2) = \pi_2;$$
$$\mathbb{P}(M = 1|V = 3) = \pi_3\,;\, \mathbb{P}(M = 1|V = 4) = \pi_4.$$

Soit n le nombre total de sujets et n_j le nombre de sujets dans chaque groupe de quartile. La variance de l'estimation de π_0 est :

$$\mathbb{V}(\Pi_0) = \frac{\pi_0(1-\pi_0)}{n},$$

et la variance de l'estimation de π_j où $j \in 1, 2, 3, 4$ est définie comme suit :

$$\mathbb{V}(\Pi_j) = \frac{\pi_j(1-\pi_j)}{n_j}.$$

Nous pouvons calculer la probabilité que chaque sujet soit manquant en calculant la probabilité marginale de $M = 1$:

$$\begin{aligned}\pi_{miss} &= \mathbb{P}(M = 1)\\ &= \pi_1\pi_0 + \pi_2\pi_0 + \pi_3\pi_0 + \pi_4\pi_0,\end{aligned}$$

et le pourcentage attendu de valeurs manquantes peut être écrit comme :

$$\begin{aligned}\mathbb{E}(\Pi) &= \pi_{miss}\\ &= \pi_1\pi_0 + \pi_2\pi_0 + \pi_3\pi_0 + \pi_4\pi_0;\end{aligned}$$

Ainsi, la variance de ce pourcentage estimé est donnée par :

$$\mathbb{V}(\Pi) = \frac{\pi_{miss}(1-\pi_{miss})}{n}.$$

Pour implémenter la règle de données manquantes dans (4.5), les chercheurs suppriment généralement π_1 sujets avec $Y_2 < Q_1$, π_2 sujets avec $Q_1 \leq Y_2 < Q_2$, et ainsi de suite.

Méthode des percentiles : Ce cas est une extension de la méthode des seuils multiples, où chaque sujet a une probabilité d'être manquant qui dépend de son rang de percentile dans le prédicteur de données manquantes.

S'il existe une relation directe entre l'indicateur de données manquantes et le prédicteur, alors la règle de ces valeurs est définie par :

$$\mathbb{P}(M = 1|Y_2 = q_k) = \frac{k}{100},$$

où q_k est la valeur de Y_2 correspondant à son $k$ème percentile. S'il y a une relation indirecte, alors la règle des données manquantes est définie comme :

$$\mathbb{P}(M = 1|Y_2 = q_k) = 1 - \frac{k}{100}.$$

Méthode de régression logistique : Lors de la génération de données `MAR` en utilisant la méthode de régression logistique, le modèle de régression logistique est considéré comme la règle des données manquantes et les coefficients de régression populationnels du modèle sont les paramètres associés à la règle de ces données. Ensuite, le modèle de régression logistique pour le sujet i est écrit comme suit :

$$\log\left(\frac{\mathbb{P}(M_i = 1|y_{2,i})}{1 - \mathbb{P}(M_i = 1|y_{2,i})}\right) = \beta_0 + \beta_1 y_{2,i},$$

où $y_{2,i}$ est la valeur du sujet i sur Y_2. La probabilité d'être manquant pour chaque sujet est donnée par :

$$\mathbb{P}(M_i = 1|y_{2,i}) = \frac{1}{1 + e^{-\beta_0 - \beta_1 y_{2,i}}}.$$

Étant donné que la fonction ci-dessus est continue, cela signifie que la probabilité d'être manquant pour Y_1 augmente ou diminue progressivement à mesure que la valeur de Y_2 augmente. Avec la régression logistique, il n'y a pas de formule simple pour calculer le pourcentage attendu de données manquantes, donc nous estimons ce pourcentage en calculant la moyenne des probabilités dans un échantillon de grande taille :

$$\pi_{miss} = \frac{1}{n}\sum_{i=1}^{n} \frac{1}{1 + e^{-\beta_0 - \beta_1 y_{2,i}}}.$$

— `MNAR` : Dans les données manquantes d'une manière non aléatoires, la cause est liée à la variable d'intérêt. Dans ce type, la probabilité qu'une observation soit manquante dépend de la valeur observée Y_o et des données manquantes Y_{miss}. Si Y_1 n'est ni `MCAR` ni `MAR`, alors c'est `MNAR`. Générer des données `MNAR` est similaire à générer des données `MAR` ; la seule différence est que dans les données `MNAR`, la probabilité de valeurs manquantes dépend de la valeur de la variable elle-même et non des valeurs observées des autres variables. Par conséquent, nous pouvons changer le prédicteur pour la variable avec des valeurs manquantes, puis utiliser l'une des méthodes présentées ci-dessus pour générer des données `MAR`. Par exemple, pour transformer la règle (2.6) qui génère des données `MAR` en une règle qui génère des données `MNAR`, il suffit de remplacer la variable Y_2 par Y_1 ; ainsi, la règle correspondante pour générer des données `MNAR` est que lorsque la valeur de Y_1 est au-dessus d'un seuil a, Y_1 a une probabilité π_1 d'être manquant, sinon, Y_1 a une probabilité π_2 d'être manquant.

Dans la section suivante, nous présentons l'algorithme proposé, appelé `SEM`, ainsi que quelques méthodes concurrentes pour résoudre le problème des vecteurs de réponse incomplets dans un modèle linéaire à effets mixtes avec interactions.

2.4 SEM et ses concurrents

Comme nous l'avons mentionné précédemment, l'algorithme EM est largement utilisé pour la gestion des données incomplètes ; dans cette situation, la fonction de maximum de vraisemblance est difficile ou impossible à utiliser pour estimer le vecteur de paramètres θ du modèle considéré. Nous formalisons directement une itération qui permet de comprendre clairement le fonctionnement de cet algorithme :

- Soit $y = (y_1, \ldots, y_n)$ les observations indépendantes et identiquement distribuées (i.i.d.) de la vraisemblance $p(y|\theta)$.
- La maximisation de $\log p(y|\theta)$ est impossible.
- On considère des données latentes $z = (z_1, \ldots, z_n)$ qui rendent possible la maximisation de la vraisemblance des données complètes lorsqu'elles sont connues.
- Comme nous ne connaissons pas ces données z, nous estimons la vraisemblance des données complètes en tenant compte de toutes les informations connues, de sorte que l'estimateur est donné comme suit (étape E) :

$$Q(\theta|\theta_{k-1}) = \mathbb{E}[\log p(y, z; \theta|y; \theta_{k-1})], \tag{2.8}$$

  où θ_{k-1} est le vecteur des paramètres à l'itération $k-1$.
- Enfin, nous maximisons cette vraisemblance estimée pour déterminer la nouvelle valeur du paramètre (étape M). Ainsi, le passage de l'itération $k-1$ à l'itération k dans l'algorithme consiste à déterminer le vecteur des paramètres à l'itération k, θ_k :

$$\theta_k = \arg\max_{\theta} Q(\theta|\theta_{k-1}).$$

  où θ_0 est choisi arbitrairement. Lorsque l'une de ces deux étapes est impossible à réaliser, nous pouvons envisager une forme d'extension de l'algorithme EM, comme les algorithmes SAEM, SEM ou MCEM. Dans la sous-section suivante, nous déduisons l'algorithme SAEM de la formule de l'algorithme EM.

Algorithme SAEM

L'algorithme d'Espérance-Maximisation par Approximation Stochastique (Stochastic Approximation Expectation Maximization algorithm, SAEM) a été proposé par [36], dans lequel l'étape E a été remplacée par une approximation stochastique. Cet algorithme a été introduit pour la première fois par [103] et également utilisé par [20], où l'algorithme génère des itérations de la forme :

$$\theta_k = \theta_{k-1} - \gamma_k[h(\theta_{k-1}) + \epsilon_k],$$

où γ_k est une séquence de tailles de pas positives, h est une fonction de θ, and ϵ_k is a constant such that $\epsilon_k = \mathbb{E}[h(\theta_k)]$. A partir de cette forme, on obtient l'algorithme SAEM où l'étape E de l'algorithme EM est divisée en deux étapes ; considérons l'itération k :

Tout d'abord, nous échantillonnons une réalisation z_k de la variable latente à partir de la distribution conditionnelle $(p_{\theta_{k-1}}(z|y))$ de z étant donné y, en utilisant la valeur du paramètre θ_{k-1} à l'itération $k-1$.

Deuxièmement, en utilisant la réalisation z_k de la première étape, nous mettons à jour la valeur de $Q_k(\theta|\theta_{k-1})$ (see, (2.8)) à travers une procédure d'approaximation stochastique.

L'algorithme se poursuit ensuite comme suit :

Etape d'initialisation : Initialiser θ_0 dans un ensemble compact fixe.

Ensuite, pour $k \geq 1$, la $k^{\text{ème}}$ itération consiste en trois étapes :

Etape de simulation : simuler z_k à partir de la distribution conditionnelle $p_{\theta_{k-1}}(.|y)$.

Etape d'approximation stochastique : calculer la quantité

$$Q_k(\theta) = Q_{k-1}(\theta) + \gamma_k \left(\frac{1}{m(k)} \sum_{j=1}^{m(k)} \log f(z_k(j); \theta) - Q_{k-1}(\theta) \right),$$

ou m_k est le nombre de simulations à chaque itération.

Etape de maximisation : mettre à jour la valeur du paramètre en fonction de $\theta_k = \arg\max_\theta Q_k(\theta)$.

Nous examinons ensuite l'algorithme `SEM` pour montrer en quoi il diffère d'un `SAEM`.

Algorithme `SEM`

L'approche `SEM` est un cas spécifique de l'algorithme `SAEM`. Dans l'algorithme `SEM`, la taille du pas est fixée à zéro $\gamma_k = 0$ et le nombre de simulations par itération est constant.

Cette méthode est une extension de l'algorithme `EM` et constitue une solution lorsqu'il est impossible de compléter le maximum de vraisemblance.
Cet algorithme a été introduit par [28], où entre les étapes E et M, ils ont utilisé une étape stochastique qui consistait à générer un échantillon complet contenant des variables latentes à partir de la densité conditionnelle, sur la base des données observées. A partir d'une position initiale θ_0, une itération k de l'algorithme `SEM`, où, à chaque θ_k, est associé un θ_{k+1}, est définie comme suit :

Etape E : Calculer la densité conditionnelle $f(y|x;\theta_k)$;

Etape S : Tirez z_k de la distribution conditionnelle, puis obtenez l'échantillon complet $y_k = (x, z_k)$;

Etape M : Mettre à jour les paramètres θ_{k+1} en maximisant la fonction de vraisemblance basée sur le vecteur complet y_k.

Puis, nous examinons l'algorithme `MCEM` pour montrer comment une simulation de Monte Carlo est utilisée dans l'algorithme `EM`.

Algorithme `MCEM`

En 1990, [127] a proposé de remplacer l'étape d'espérance pour calculer $Q(\theta|\theta_{k-1})$ par une intégration de Monte Carlo, d'où le nom d'algorithme `MCEM` (Monte Carlo `EM`). A l'itération k, l'étape E est remplacée par la procédure suivante :

Etape de simulation (Etape-S) : générer $m(k)$ réalisations $z_k(j)$ (avec $j = 1, \ldots, m(k)$)) du vecteur de données manquantes sous la fonction de distribution $p(z; \theta_k)$.

Integration de Monte Carlo : calculer l'approximation de $Q(\theta|\theta_{k-1})$ selon :

$$Q_k(\theta) = \frac{1}{m(k)} \sum_{j=1}^{m(k)} \log f(z_k(j); \theta).$$

L'**étape de Maximisation** ne change pas.

L'algorithme `SAEM` est plus efficace que l'algorithme `MCEM`, dans lequel, à chaque itération, la simulation des valeurs manquantes est répétée et celles obtenues précédemment ne sont pas utilisées. Dans `SAEM`, toutes les valeurs manquantes simulées contribuent à l'étape d'estimation et c'est l'avantage de cet algorithme où l'étape de maximisation est moins coûteuse que l'étape de simulation.

Comme indiqué précédemment, ces algorithmes peuvent être utilisés de manière à traiter les valeurs non observées des données qui sont définies dans la secton suivante.

2.4.1 Application dans un `LMEM`

Dans ce chapitre, nous proposons de traiter les problèmes de données manquantes et des données censurées en présence d'interactions en utilisant l'algorithme `SEM`, proposé par [115]. Cette méthode est utilisée pour estimer les paramètres lorsqu'il est compliqué d'utiliser l'algorithme `EM` ; il s'agit d'un cas particulier d'imputations multiples (`MI`).

Pour comprendre `MI` ([56]), nous devons connaître l'idée d'une imputation simple (`SI`) dans laquelle les données non observées $\{Y_{no}\}$ (manquantes ou censurées) sont remplacées par une valeur, puis les paramètres sont estimés à l'aide de méthodes connues afin de maximiser la fonction de vraisemblance. Pour obtenir une estimation robuste, l'imputation simple peut être répétée plusieurs fois avec différentes valeurs de $\{Y_{no}\}$ et les résultats peuvent être combinés. Cette méthode est appelée imputation multiple.
Nous avons appliqué l'algorithme après avoir croisé les valeurs `MAR` avec les valeurs censurées (voir Algorithm 1). Dans l'algorithme `SEM` ([28]), les valeurs de $\{Y_{no}\}$ sont tirées de la distribution conditionnelle des données non observées compte tenu des données observées en utilisant les valeurs actuelles des paramètres. Nous avons généré des échantillons de $P_\Theta(Y_{no}|Y_o)$ où cette distribution a été calculée à l'aide de

l'échantillonnage de Gibbs ; la procédure est soulignée dans l'Algorithme 1 (la méthode `SEM` utilisant l'échantillonnage de Gibbs a été développée dans [115]). Pour plus d'informations sur l'algorithme `EM`, voir [108].

Algorithm 1 `SEM` algorithm : N is the number of iterations of the `SEM` algorithm, M is the burn-in level, $Y_1, \ldots, Y_n$ is the response vector, G is the number of iterations of Gibbs sampling, $Y[u_i]$ is the response vector with respect to the ith random variable, and fixed is the summation of the fixed effects with the fixed interaction part.

Input: N, M, $Y_1, \ldots, Y_n$, and G.

1: Random initialization of $\mathbf{\Theta}$
2: **for** $j = 1, \ldots, N$ **do**
3: **for** $g = 1, \ldots, G$ **do**
4: draw $e^{(g)}$ from $\mathcal{N}(0, \sigma_e^2$, lower $= \min(e) - \text{fixed} - u^{(g-1)}$, upper $= \max(e) - \text{fixed} - u^{(g-1)})$
5: draw $u_1^{(g)}$ from $\mathcal{N}(0, \Sigma_{u_1}$, lower $= \max(\min(Y[u_1]) - u_2^{(g-1)} - e^{(g)})$, upper $= \min(\max(Y[u_1]) - u_2^{(g-1)} - e^{(g)}))$
6: draw $u_2^{(g)}$ from $\mathcal{N}(0, \Sigma_{u_2}$, lower $= \max(\min(Y[u_2]) - u_1^{(g)} - e^{(g)})$, upper $= \min(\max(Y[u_2]) - u_1^{(g)} - e^{(g)}))$
7: **end for**

output: $\mathbf{Y}_{no}$ obtained from sampled $(u^{(G)}, e^{(G)})$.

8: $\mathbf{\Theta}_{j+1} = \text{argmax}_{\mathbf{\Theta}}\, L\,(\mathbf{\Theta}|\mathbf{Y}_o, \mathbf{Y}_{no})$
9: **end for**
10: $\hat{\mathbf{\Theta}} = (N - M)^{-1} \sum_{j=M+1}^{N} \Theta_j$

output: $\widehat{\mathbf{\Theta}}$.

Dans l'algorithme `SEM`, Θ_{j+1} (le paramètre à l'état $j+1$) ne dépend que de Θ_j et Y_j. En prenant T l'opérateur de l'algorithme EM et M, l'opérateur qui associe Θ_{j+1} à Y_j via l'étape M, le paramètre actualisé Θ_{j+1} peut être écrit comme suit (voir [28]) :

$$\Theta_{j+1} = T(\Theta_j) + V(\Theta_j, Y_j),$$

où $V(\Theta_j, Y_j) = M(Y_j) - T(\Theta_j)$.

Ensuite, la forme étendue de `SEM` nous conduit à l'algorithme `SAEM` (voir aussi Algorithm 2). A partir d'une position initiale Θ_0, l'itération de l'algorithme `SAEM`, à laquelle pour chaque Θ_j, un Θ_{j+1} est associé, est définie par l'équation :

$$\Theta_{j+1} = T(\Theta_j) + \gamma_j V(\Theta_j, Y_j),$$

où γ_j est la taille du pas, qui est une séquence décroissante de nombres réels positifs commençant par $\gamma_0 = 1$, $\gamma_j \mapsto 0$ lorsque $j \mapsto \infty$. Les performances théoriques et pratiques de l'algorithme `SAEM` dépendent fortement de la vitesse de convergence vers 0 et de la décroissance régulière de la séquence γ_j, ce qui explique l'importance du choix de la taille du pas.

[27] a effectué de nombreux tests pour voir quelle séquence était la plus efficace, et ils ont obtenu deux types de séquences qui étaient significatives, l'une d'un mode lent :

$$\gamma_j = \cos(\frac{j \times \pi}{2N}),$$

l'autre type était de type linéaire :

$$\gamma_j = -\frac{(j+1)}{N}.$$

[29] ont choisi la séquence décroissante de leurs expériences précédentes [30] ; ils ont pris cette séquence comme étant :

$$\gamma_j = \cos(N \times \alpha)\mathbb{1}_{\{0 \leq j \leq 20\}} + \frac{c}{\sqrt{j}}\mathbb{1}_{\{j>20\}}, \text{ où } \cos(20 \times \alpha) = \frac{c}{\sqrt{20}} = 0.3.$$

Cette séquence a également été choisie par [96, 101] dans un modèle mixte non linéaire comme 1 à la première itération et $(j-N)^{-1}$ au cours des dernières itérations.

Algorithm 2 `SAEM` algorithm : N is the number of iterations of the `SEM` algorithm, M is the burn-in level, $Y_1, \ldots, Y_n$ is the response vector, G is the number of iterations of Gibbs's sampling, γ_j is the decreasing sequence, $Y[u_i]$ is the response vector with respect to the ith random variable, and fixed is the summation of the fixed effects with the fixed interaction part.

Input: N, M, $Y_1, \ldots, Y_n$, and G.

1: Random initialization of $\mathbf{\Theta}$
2: **for** $j = 1, \ldots, N$ **do**
3: **for** $g = 1, \ldots, G$ **do**
4: draw $e^{(g)}$ from $\mathcal{N}(0, \sigma_e^2$, lower $= \min(e) - \text{fixed} - u^{(g-1)}$, upper $= \max(e) - \text{fixed} - u^{(g-1)})$
5: draw $u_1^{(g)}$ from $\mathcal{N}(0, \Sigma_{u_1}$, lower $= \max(\min(Y[u_1]) - u_2^{(g-1)} - e^{(g)})$, upper $= \min(\max(Y[u_1]) - u_2^{(g-1)} - e^{(g)}))$
6: draw $u_2^{(g)}$ from $\mathcal{N}(0, \Sigma_{u_2}$, lower $= \max(\min(Y[u_2]) - u_1^{(g)} - e^{(g)})$, upper $= \min(\max(Y[u_2]) - u_1^{(g)} - e^{(g)}))$
7: **end for**

output: $\mathbf{Y}_{no}$ obtained from sampled $(u^{(G)}, e^{(G)})$.

8: $\mathbf{\Theta}_{j+1} = \text{argmax}_{\mathbf{\Theta}}\, \gamma_j \times L\,(\mathbf{\Theta}|\mathbf{Y}_o, \mathbf{Y}_{no})$
9: **end for**
10: $\hat{\mathbf{\Theta}} = (N-M)^{-1} \sum_{j=M+1}^{N} \Theta_j$

output: $\widehat{\mathbf{\Theta}}$.

Lors de l'application de ces deux algorithmes, nous avons été confrontés à un problème de convergence causé par la présence d'une interaction entre les variables fixes et aléatoires. Par conséquent, dans la section suivante, nous présentons comment traiter ce problème en utilisant l'algorithme de Monte Carlo Hamiltonien.

2.5 Convergence des paramètres

En introduisant l'effet de l'interaction fixe-aléatoire dans la section d'application, nous avons été confrontés à un problème de convergence dans les algorithmes `SEM` et `SAEM` lors de l'utilisation de la forme standard de Monte Carlo (`MC`). Afin de résoudre ce problème, nous proposons d'utiliser l'algorithme hybride `MC` ([19]), qui utilise l'algorithme symétrique de Metropolis-Hastings pour accepter ou rejeter une proposition basée sur la fonction hamiltonienne, d'où le nom de l'algorithme, l'algorithme de Monte Carlo hamiltonien (`HMC`). Ensuite, nous présentons la manière de mettre en œuvre l'algorithme `HMC`.

Implementation

La fonction hamiltonienne $H(p, q)$ s'écrit en termes de densité conjointe $\pi(p, q)$:

$$H(p, q) = -\log \pi(p, q),$$

qui se décompose en deux termes :

$$\begin{aligned} H(p, q) &= -\log \pi(p|q) - \log \pi(q) \\ &= K(p, q) + V(p, q). \end{aligned}$$

Le premier terme correspond à la densité de la distribution cible, le second est déterminé par la distribution cible lorsque $K(p, q)$ n'est pas contraint et doit être spécifié par l'implémentation.

La probabilité d'acceptation du passage de l'état (p_0, q_0) à (p_i, q_i) est déterminée à l'aide de l'algorithme de Metropolis-Hastings ([60, 91]), $Q(p_i, q_i|p_0, q_0)$ étant la densité de probabilité définissant chaque proposition :

$$a(p_i, q_i|p_0, q_0) = \min\left(1, \frac{Q(p_0, q_0|p_i, q_i)\pi(p_i, q_i)}{Q(p_i, q_i|p_0, q_0)\pi(p_0, q_0)}\right).$$

En se référant à la méthode symétrique de Metropolis ([102]), la probabilité d'acceptation est simplifiée sous la forme :

$$\begin{aligned} a(p_i, q_i|p_0, q_0) &= \min\left(1, \frac{\pi(p_i, q_i)}{\pi(p_0, q_0)}\right) \\ &= \min\left(1, \frac{\exp(-H(p_i, q_i))}{\exp(-H((p_0, q_0))}\right) \\ &= \min(1, \exp(-H(p_i, q_i) + H(p_0, q_0))). \end{aligned}$$

Dans la section suivante, nous présentons les principaux résultats numériques basés sur une simulation psychologique et des données réelles.

2.6 Application numérique

Cette section vise à comparer l'algorithme SEM proposé dans ce chapitre avec d'autres méthodes en présence de données incomplètes dans un LMEM complet. Dans ce travail, nous nous intéressons à l'imputation des données manquantes de type MAR avec quatre pourcentages de présence (0%, 5%, 10% et 20%), ces mêmes pourcentages étant également utilisés pour les données censurées, ce qui donne ainsi 16 cas de données incomplètes. La méthode proposée a d'abord été appliquée à des données simulées, puis à des données réelles.

2.6.1 Données simulées

Les données simulées ont été créées à partir d'une base de données obtenue expérimentalement. Afin d'explorer le processus cognitif de récupération en mémoire de l'orthographe de mots français, 30 participants devaient écrire à la main le nom de 150 dessins d'objets (items), constituant une base de données de 4500 temps de réaction (RT) ([99]). Après avoir supprimé les RTs correspondant à des erreurs techniques (données MCAR, par exemple lorsque la tablette graphique ne détecte pas le stylo et que le temps de réponse n'est pas enregistré), des erreurs d'orthographe (données MAR, où le mot écrit par le participant est incorrect et le temps de réponse est exclu), et des données censurées supérieures à 2.5 d'écart-type par rapport à la moyenne (données MNAR, où seules les valeurs dans cet intervalle sont conservées, selon la tradition dans ce type d'analyse), il restait 3434 valeurs.

[99] a effectué une analyse de régression linéaire impliquant neuf variables à effet fixe. Dans le présent travail, nous en avons retenu deux. La première était une estimation de l'âge auquel le nom de l'image a été acquis dans l'enfance (âge d'acquisition, AoA ci-après). Ce facteur est l'un des prédicteurs les plus importants des RTs de dénomination d'image ([97]). La deuxième variable à effets fixes correspondait au nombre de lettres dans le nom d'une image (Lett_cat). Nous avons effectué une division médiane pour obtenir une variable catégorielle. Ainsi, les étiquettes d'images avec un nombre de lettres inférieur à la médiane de l'échantillon ont été considérées comme des mots courts et celles avec un nombre de lettres supérieur ont été considérées comme des mots longs. L'interaction entre ces deux variables à effets fixes a été incluse dans le modèle. Enfin, il y avait deux variables à effets aléatoires : les items, c'est-à-dire, les étiquettes de 150 images produites par les 30 participants, et les participants eux-mêmes. L'interaction fixe-aléatoire entre la variable AoA et les participants a également été incluse dans le LMEMs.

Par conséquent, le modèle peut être écrit comme suit :

$$\begin{aligned}\texttt{RTs} = {} & \mathrm{b}_0 + \mathrm{b}_1 \times \texttt{AoA} + \mathrm{b}_2 \times \texttt{Lett_cat} + \mathrm{b}_3 \times \texttt{AoA} \times \texttt{Lett_cat} \\ & + \text{participants} + \text{items} + \texttt{AoA} \times \text{participants} + e.\end{aligned}$$

Dans cette section, nous illustrons l'algorithme proposé en comparant les résultats aux algorithmes SAEM et MCMC en utilisant différentes proportions de données incomplètes. La comparaison a été effectuée en calculant l'erreur absolue moyenne (Mean

Absolute Error, MAE), qui mesure les erreurs entre les vecteurs de référence et les vecteurs prédits (avec Θ comme valeur de référence et $\hat{\Theta}$ comme valeur prédite) ; l'équation standard de la MAE est donnée par :

$$\texttt{MAE} = \frac{1}{n}\sum_{i=1}^{n} |\hat{\Theta}_i - \Theta_i|.$$

Nous avons également utilisé la corrélation linéaire de Pearson (LCor) et la corrélation de rang de Spearman (RCor) pour comparer les performances des ces algorithmes en analysant les relations entre leurs variables. Ces deux indicateurs apportent des perspectives complémentaires : la corrélation linéaire permet d'évaluer la force des relations directes entre les variables, tandis que la corrélation de rang capture les relations monotones, même non linéaires, et est moins sensible aux valeurs aberrantes. Afin d'identifier l'algorithme le plus performant, nous avons comparé les corrélations obtenues pour chaque algorithme à un vecteur de référence, dont les valeurs idéales sont (LCor=1, Rcor=1). Le cas dont les valeurs de corrélation sont les plus proches de ce vecteur de référence est considéré comme le plus efficace, car il présente les relations les plus fortes et les plus cohérentes entre ses variables. Les formules des corrélations utilisées dans nos analyses sont présentées ci-dessous :

$$\texttt{LCor} = \frac{\mathrm{Cov}(\hat{\Theta}_i, \Theta_i)}{\sigma(\hat{\Theta}_i)\sigma(\Theta_i)} ;$$

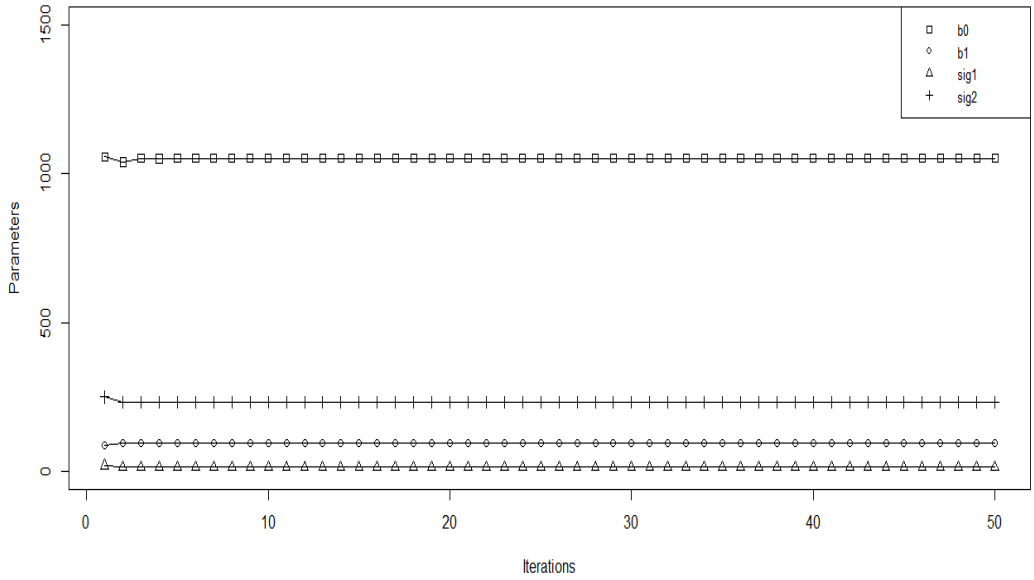
$$\texttt{RCor} = 1 - \frac{6\sum d^2}{n(n^2-1)} ;$$

où d^2 est le carré de la différence entre les rangs des deux coordonnées.

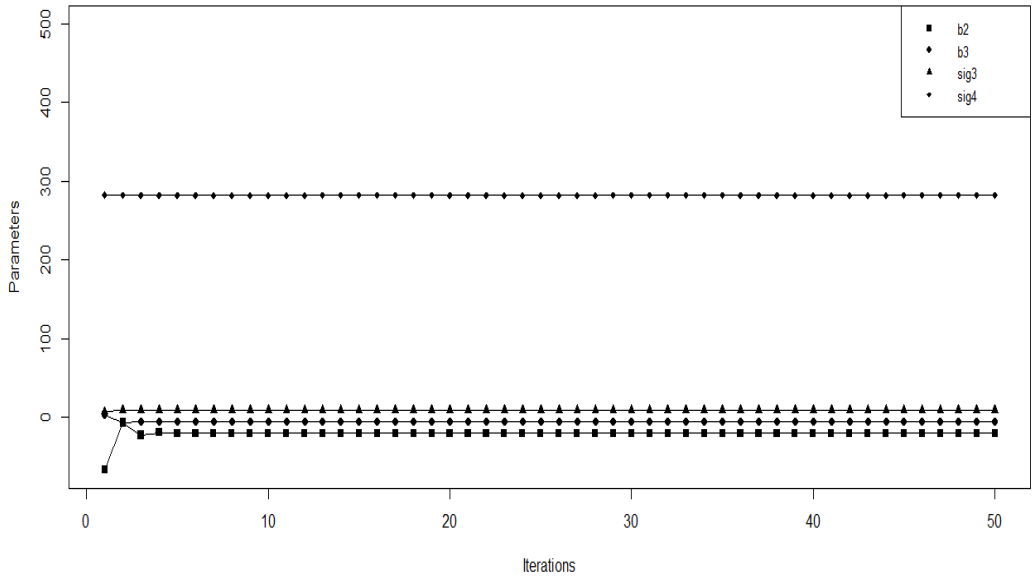
Dans cette section numérique, nous fournissons les 16 cas pour voir la performance des méthodes en absence de l'un de ces deux types considérés de valeurs non observées (type MAR/données censurées) ou en présence d'un pourcentage faible ou élevé. En outre, nous avons utilisé les données simulées de 4500 observations après avoir testé les méthodes proposées sur un plus grand nombre d'observations. Nous avons obtenu les mêmes résultats que pour 4500 valeurs.

Après la mise en œuvre de notre algorithme, nous avons vérifié la convergence en illustrant les valeurs des paramètres estimés obtenues à chaque itération ; ceci est présenté dans la Figure 2.1.

Dans la Figure 2.2, nous avons tracé les résidus normalisés de chaque cas. Nous avons observé une dispersion similaire entre les données simulées standard et l'algorithme SEM, tandis que nous avons remarqué une dispersion faible mais visible en mettant en oeuvre l'algorithme SAEM, et enfin, nous avons observé que lorsque nous avons utilisé la méthode MCMC, les résidus ont été dispersés loin de la ligne normale.

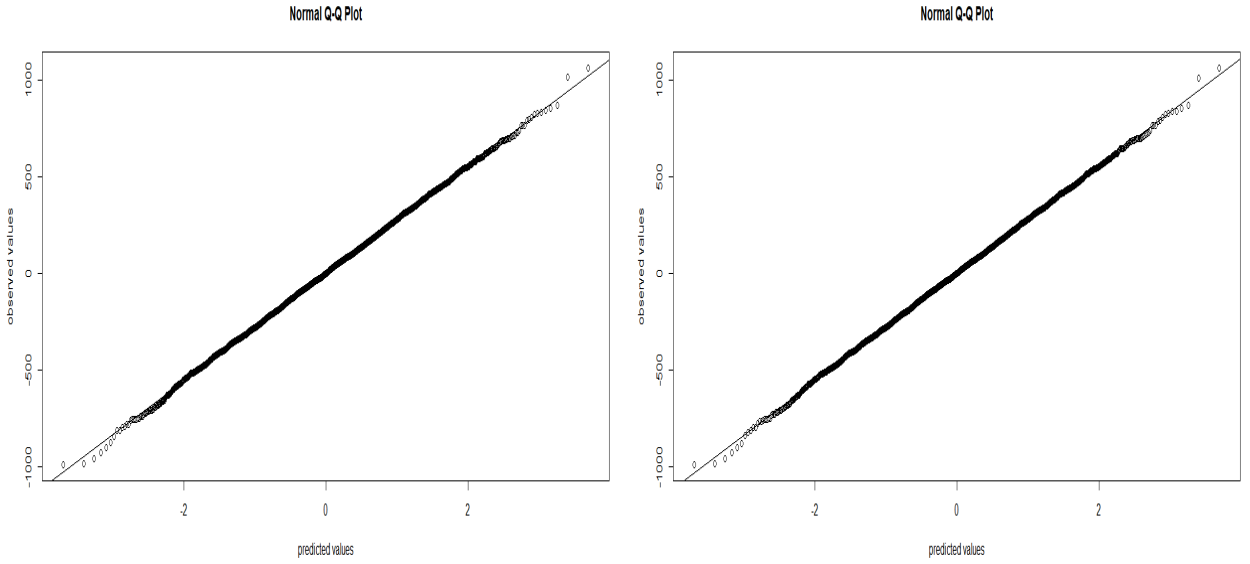


(**a**)

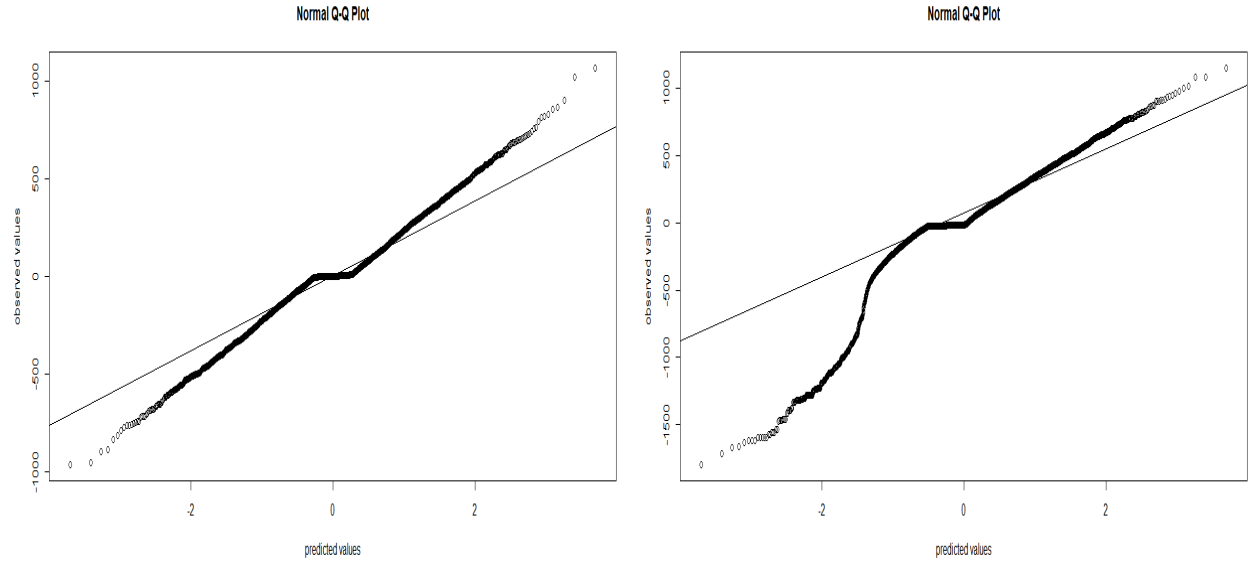


(**b**)

FIGURE 2.1 – Estimations des paramètres à chaque itération dans `SEM` avec une période de rodage de M=10 ; (**a**)-paramètres b_0, b_1, sig_1 et sig_2, (**b**)-paramètres b_2, b_3, sig_3 et sig_4.



(**a**) (**b**)



(**c**) (**d**)

FIGURE 2.2 – Résidus normalisés présentés ; (**a**)-données simulées standard, (**b**)-algorithme `SEM`, (**c**)-algorithme `SAEM`, (**d**)-algorithme `MCMC`.

	(0%, 0%)	(0%, 5%)			(0%, 10%)			(0%, 20%)		
$\hat{\Theta}_i$	**Reference**	SEM	SAEM	MCMC	SEM	SAEM	MCMC	SEM	SAEM	MCMC
b_0	**1045**	1044	**1045**	1068	**1044**	1043	1010	**1047**	**1047**	822.8
b_1	**93.16**	**93.54**	88.51	48.82	**93.91**	89.37	41.25	**93.51**	88.84	77.96
b_2	**−52.28**	**−51.30**	−42.33	−36.75	**−52.50**	−47.30	51.49	−30.27	**−51.66**	18.26
b_3	**2.80**	**3.17**	−0.18	6.92	**4.43**	5.52	−14.44	**−1.79**	5.02	−18.20
sig_1	**14.57**	**14.65**	14.43	5.29	**14.65**	12.54	12.56	**14.70**	12.17	11.30
sig_2	**230.8**	**231.7**	225.9	350.5	**231.1**	223.0	430.5	**233.2**	219.0	538.7
sig_3	**9.79**	**9.81**	9.84	2.80	**9.80**	8.13	8.61	**9.81**	9.12	8.94
sig_4	**282.0**	**282.0**	275.5	283.1	**282.0**	269.4	277.8	**282.0**	253.5	253.9
MAE	**0**	**0.421**	3.637	28.07	**0.505**	4.628	51.84	**3.935**	6.575	83.64
LCor	**1**	**0.999**	**0.999**	0.992	**0.999**	**0.999**	0.971	**0.999**	**0.999**	0.915
RCor	**1**	**1**	**1**	0.904	**1**	**1**	0.738	**1**	**1**	0.833

	(5%, 0%)			(5%, 5%)			(5%, 10%)			(5%, 20%)		
$\hat{\Theta}_i$	SEM	SAEM	MCMC	SEM	SAEM	MCMC	SEM	SAEM	MCMC	SEM	SAEM	MCMC
b_0	**1044**	1038	1004	1069	**1037**	1019	1069	**1036**	963	1078	**1038**	780
b_1	**93.30**	94.33	93.79	87.57	**90.04**	52.82	88.07	**90.90**	45.29	86.52	**90.75**	80.88
b_2	−46.69	**−53.57**	−49.05	**−49.96**	−42.94	−28.23	**−52.68**	−47.94	55.58	−32.18	**−50.21**	26.67
b_3	**1.04**	4.58	6.55	8.27	**1.27**	8.63	9.70	**6.75**	−11.90	**3.82**	5.61	−15.79
sig_1	**15.19**	13.65	16.59	**15.25**	17.02	8.34	15.36	**13.93**	16.19	**15.79**	12.80	4.10
sig_2	**231.0**	223.7	213.6	**229.8**	218.7	332.4	**228.9**	216.0	412.7	**232.4**	211.9	518.8
sig_3	**9.94**	7.93	11.07	**9.97**	8.63	5.05	**9.95**	7.52	10.61	**10.11**	8.42	6.56
sig_4	**282.4**	281.4	347.6	**282.6**	275.0	349.4	**282.6**	268.7	342.2	**282.6**	253.2	317.6
MAE	**1.146**	2.646	16.78	**4.970**	5.516	34.43	**5.089**	6.287	62.03	8.118	**8.108**	88.97
LCor	**0.999**	**0.999**	0.996	**0.999**	**0.999**	0.991	**0.999**	**0.999**	0.971	**0.999**	**0.999**	0.911
RCor	**1**	**1**	**1**	**1**	**1**	0.928	**1**	**1**	0.738	**1**	**1**	0.809

TABLE 2.1 – Comparaison des algorithmes avec différents pourcentages de données manquantes, pour un nombre d'itérations de 50. Les écarts-types sig_1, sig_2, sig_3 et sig_4 correspondent respectivement aux participants, aux items, à la partie I_{FR} et à l'erreur. Les pourcentages à gauche montrent les données MAR, et ceux à droite les données censurées.

TABLE 2.1 – *Cont.*

	(10%, 0%)			**(10%, 5%)**			**(10%, 10%)**			**(10%, 20%)**		
$\hat{\Theta}_i$	SEM	SAEM	MCMC	SEM	SAEM	MCMC	SEM	SAEM	MCMC	SEM	SAEM	MCMC
b_0	1070	**1037**	981	1069	**1035**	999	1070	**1034**	938	1080	**1037**	759
b_1	86.59	**94.05**	90.52	87.58	**90.27**	50.902	88.10	**90.80**	46.43	86.29	**90.33**	81.31
b_2	−45.96	**−50.70**	−62.01	**−45.00**	−39.50	−44.14	**−47.79**	−44.57	45.21	−28.34	**−48.01**	23.38
b_3	6.79	**5.26**	12.25	6.69	**1.46**	14.45	8.13	**6.78**	−8.83	**2.53**	5.81	−16.26
sig_1	**14.80**	24.43	4.36	**14.97**	27.35	25.00	**15.15**	22.09	30.51	**15.93**	21.39	18.64
sig_2	**228.7**	220.1	205.1	**230.0**	215.3	323.1	**229.1**	212.5	401.1	**232.5**	208.7	506.5
sig_3	**9.79**	11.46	5.94	**9.89**	12.10	6.336	**9.88**	10.16	13.87	**10.11**	10.50	8.73
sig_4	**282.4**	279.6	398.0	**282.5**	273.2	389.6	**282.5**	267.2	379.8	**282.5**	251.8	354.3
MAE	**5.608**	4.699	30.12	**5.387**	8.279	40.15	**5.424**	8.191	68.85	8.747	9.636	93.20
LCor	**0.999**	**0.999**	0.990	**0.999**	**0.999**	0.987	**0.999**	**0.999**	0.969	**0.999**	**0.999**	0.908
RCor	**1**	**1**	0.904	**1**	**1**	0.976	**1**	**1**	0.833	**1**	**1**	0.833
	(20%, 0%)			**(20%, 5%)**			**(20%, 10%)**			**(20%, 20%)**		
$\hat{\Theta}_i$	SEM	SAEM	MCMC	SEM	SAEM	MCMC	SEM	SAEM	MCMC	SEM	SAEM	MCMC
b_0	**1045**	1035	896	**1045**	1044	908	**1045**	1036	857	**1051**	**1039**	687
b_1	**93.34**	94.05	100.8	**93.70**	80.78	64.43	**94.05**	90.14	57.35	**92.78**	89.41	91.59
b_2	−35.09	**−38.36**	2.54	**−34.54**	−128.87	21.66	−36.11	**−34.79**	97.19	−19.91	**−39.24**	85.55
b_3	−1.82	**1.90**	−16.60	**−1.55**	39.61	−13.36	−0.36	**4.00**	−31.09	−4.73	**3.67**	−42.98
sig_1	**12.12**	26.88	29.49	**12.44**	11.70	39.89	**12.62**	25.58	33.25	**13.62**	26.53	22.49
sig_2	**230.9**	211.4	204.7	**231.8**	118.2	315.9	**231.3**	204.4	387.9	**233.1**	200.9	485.3
sig_3	8.84	11.33	**8.95**	**8.98**	5.15	11.48	**9.03**	10.91	10.93	**9.41**	11.76	6.227
sig_4	**282.3**	276.2	465.9	**282.3**	265.3	453.6	**282.3**	264.0	442.2	**282.4**	249.2	411.6
MAE	**3.261**	8.002	57.07	**3.374**	32.83	67.38	**2.994**	10.905	93.01	**6.343**	12.47	117.2
LCor	**0.999**	**0.999**	0.969	**0.999**	0.991	0.969	**0.999**	**0.999**	0.944	**0.999**	0.998	0.873
RCor	**1**	**1**	0.976	**1**	0.9285	0.928	**1**	**1**	0.761	**1**	**1**	0.833

Nos résultats sont visualisés quantitativement dans le Tableau 2.1, nous avons marqué en gras pour chaque cas la valeur la plus proche de la référence parmi les trois méthodes considérées (le vecteur de référence était ici le vecteur simulé qui ne contenait pas de valeur manquante). Sur la base de la valeur MAE, nous pouvons observer que SEM a donné les valeurs les plus basses dans tous les cas, sauf dans le cas de MAR à 5% et de censure à 20%, où SAEM (8.108) a battu SEM (8.118) d'une marge très étroite. Nous pouvons constater que l'algorithme SEM, nous permet d'avoir une meilleurs estimation des paramètres aussi bien dans le cas (0%, 10%) que dans le cas (20%, 5%). Nous pouvons constater également, que dans le cas où le pourcentage de MAR est de 20%, la méthode SAEM nous permet d'avoir une meilleurs estimation de b_2 et b_3, cependant, l'algorithme SEM, permet de fournir une meilleurs estimation pour les autres paramètres du modèle. Nous pouvons observer, que la méthode MCMC, ne permet pas de fournir une bonne estimation des paramètres en comparaison avec la méthode SAEM et la méthode SEM. De plus, nous pouvons voir que pour LCor (respectivement, Rcor), les algorithmes SEM et SAEM ont donné approximativement les mêmes résultats 0.999 (respectivement, 1) dans tous les cas. Ceci a montré que ces deux méthodes avaient une faible différence, sauf dans le cas de (20%, 5%) où la RCor de l'algorithme SAEM a diminué de 1 à 0.9285. Le MCMC avait un RCor com-

pris entre une valeur maximale de 1 et une valeur minimale de 0.738 ; cette valeur maximale a été obtenue avec 5% de MAR et l'absence de données censurées.

2.6.2 Données réelles

Dans cette section, nous avons appliqué les trois algorithmes à la base de données utilisée pour créer les données simulées. Des 4500 RT enregistrés au cours de l'expérience (30 participants produisant les noms de 150 items), 115 ont été supprimés par la censure à droite (2.56% de MNAR). [99] a également supprimé 951 valeurs, créant 21.13% de données MCAR/MAR manquantes. En appliquant les trois algorithmes proposés avec certaines données manquantes pour résoudre le problème en présence d'interactions fixes et fixes-aléatoires, nous avons obtenu les résultats présentés dans le Tableau 2.2. Les résultats ont été comparés par rapport au vecteur initial dans lequel les valeurs manquantes ont été traitées en gardant les cas vides (i.e., Keeping Empty, KE). Il est intéressant de noter que c'est ainsi que les données manquantes et censurées sont traitées dans les études psycholinguistiques.

$\hat{\Theta}$	KE	SEM	SAEM	MCMC
b_0	852.5	875.7	892.334	1298
b_1	146.6	134.2	120.4	−161.4
b_2	158.6	126.1	79.63	−335.3
b_3	−58.12	−42.08	−26.72	146.6
sig_1	184.3	196.0	203.5	216.9
sig_2	133.0	127.8	94.87	221.05
sig_3	37.05	28.67	32.84	66.13
sig_4	279.5	302.5	262.9	516.6

TABLE 2.2 – Comparaison des paramètres initiaux de la méthode KE avec les paramètres estimés après l'application des algorithmes SEM, SAEM et MCMC. Les paramètres initiaux présentés ici sont ceux obtenus à partir du modèle qui contenait 1066 Na ; nous avions 115 données censurées et 951 données manquantes.

2.7 Conclusion et Perspectives

Dans cette étude, nous avons proposé un algorithme basé sur la maximisation stochastique de l'espérance (SEM) pour estimer les paramètres d'un modèle linéaire à effets mixtes (LMEM) qui contenait des interactions fixe-fixe et fixe-aléatoire en présence de différents pourcentages de données incomplètes.

Les données simulées et réelles ont montré que l'algorithme SEM proposé donnait les meilleurs résultats par rapport aux autres concurrents, le biais des paramètres étant plus faible que celui des algorithmes SAEM et MCMC.

Nous envisageons d'étendre cette approche en considérant des niveaux d'interaction plus complexes, tels que les effets aléatoires et les interactions entre plus de deux variables. Une autre direction pourrait être d'explorer une extension du présent travail en intégrant un modèle généralisé dans le cadre de la régression logistique, en attribuant la valeur 1 aux `Na` et zéro aux valeurs observées.

Dans le chapitre suivant, nous analysons la distribution des temps de réaction, décrite dans la littérature comme suivant une loi ex-gaussienne et modélisée par [46] à l'aide d'une approche bayésienne. Nous proposons une méthode d'estimation bayésienne améliorée par rapport à celle de [46], en adoptant une approche précise pour l'initialisation des paramètres, tout en continuant à aborder le problème des données manquantes de manière efficace.

Code source en R

Nous présentons ici le code R de l'algorithme `SEM` que nous avons développé dans ce chapitre. Dans ce code, `RT` représente le temps de réaction (variable d'intérêt), `AoA` désigne l'âge d'acquisition, `Lett_cat` correspond au nombre de lettres dans le nom de l'image (ces deux dernières étant des variables fixes), tandis que `participants` et `items` représentent respectivement les variables aléatoires des participants et des items.

```
library(lme4); # for lmer
library(msm); # for rtnorm
# initial model
model_initial=lmer(RT~AoA*Lett_cat+(1+AoA|participants)+(1|items), data,
control = lmerControl(optimizer="bobyqa", calc.derivs = FALSE));

# random initialization of parameters
b0=fixef(model_initial)[1];
b1=fixef(model_initial)[2];
b2=fixef(model_initial)[3];
b3=fixef(model_initial)[4];

sig_items=attr(VarCorr(model_initial)$items,"stddev");
sig_parts=attr(VarCorr(model_initial)$participants,"stddev")[1];
sig_parts_conver<-rep(0.0,itermax);
sig_parts.AoA=attr(VarCorr(model_initial)$participants,"stddev")[2];
sig_Err=attr(VarCorr(model_initial),"sc");

Err=rnorm(n, mean=0, sd=sig_Err);
Rand_parts=rep(0,n);
Rand_items=rep(0,n);
Rand_parts.AoA=rep(0,n);
Rand_parts=rep(rnorm(np,mean=0, sd=sig_parts),each=ni);
Rand_items=rep(rnorm(ni,mean=0, sd=sig_items),each=np);
Rand_parts.AoA=rep(rnorm(np,mean=0, sd=sig_parts.AoA),each=ni);

Lett_cat_1=vector(length=n)
for(i in 1:n){
 Lett_cat_1[i]<-ifelse(test=(Lett_cat[i]==Lett_cat[2]),1,0)
}

u=runif(1,min=0,max=1);
# acceptance prob based on the Hamiltonian MC
p<-function(x,newx){
 a=min(1,exp(x-newx));
 if(u<a){
```

```
  x=newx;
 }
 else{
  x=x;
 }
}

# SEM (Algorithm 1)
itermax=50; # max number of iteration for SEM
burnin=10; # burnin length
GS=5; # number of iterations for Gibbs~sampling

# main loop
for(iter in 1:itermax){
 # fixed part
 fixed=b0+b1*AoA+b2*Lett_cat_1+b3*AoA*Lett_cat_1;
 # Gibbs sampling
 for(j in 1:GS) {
  # Err with contraints
  newErr=rtnorm(length(Err),sd=sig_Err,
  lower=min(Err)-fixed-Rand_parts-Rand_items-Rand_parts.AoA,
  upper=max(Err)-fixed-Rand_parts-Rand_items-Rand_parts.AoA);
  p(Err,newErr);
  # Rand_parts with contraints
  for (a in 1:length(factor(participants))){
   Subset1=(participants == participants[a]) & censored;
   newRand_parts=Rand_parts;
   if (sum(Subset1)>0) {
    low=max(min(!is.na(data[participants == participants[a],5]))-Rand_items[Subset1]
    -Rand_parts.AoA[Subset1]-Err[Subset1]-fixed[Subset1]);
    upp=min(max(!is.na(data[participants == participants[a],5]))-Rand_items[Subset1]
    -Rand_parts.AoA[Subset1]-Err[Subset1]-fixed[Subset1]);
   upp=ifelse(upp-low>0,upp,upp+abs(upp-low)+100);
   newRand_parts[Subset1]=rtnorm(1,sd=sig_parts, lower=low,upper=upp);
   }
  }
  p(Rand_parts,newRand_parts);
  # Rand_items with contraints
  for (a in 1:length(factor(items))){
   Subset2=(items == items[a]) & censored;
   newRand_items=Rand_items;
   if (sum(Subset2)>0){
    low=max(min(!is.na(data[items == items[a],5]))-Rand_parts[Subset2]
    -Rand_parts.AoA[Subset2]-Err[Subset2]-fixed[Subset2]);
    upp=min(max(!is.na(data[items == items[a],5]))-Rand_parts[Subset2]
```

```
    -Rand_parts.AoA[Subset2]-Err[Subset2]-fixed[Subset2]);
    upp=ifelse(upp-low>0,upp,upp+abs(upp-low)+100);
    newRand_items[Subset2]=rtnorm(1,sd=sig_items, lower=low,upper=upp);
   }
  }
  p(Rand_items,newRand_items);
  # Rand_parts.AoA with contraints
  for (a in 1:length(factor(participants))){
   Subset1=(participants == participants[a]) & censored;
   newRand_parts.AoA=Rand_parts.AoA;
   if (sum(Subset1)>0) {
    low=max(min(!is.na(data[participants == participants[a],5]))-Rand_items[Subset1]
    -Rand_parts[Subset1]-Err[Subset1]-fixed[Subset1]);
    upp=min(max(!is.na(data[participants == participants[a],5]))-Rand_items[Subset1]
    -Rand_parts[Subset1]-Err[Subset1]-fixed[Subset1]);
    upp=ifelse(upp-low>0,upp,upp+abs(upp-low)+100);
    newRand_parts.AoA[Subset1]=rtnorm(1,sd=sig_parts.AoA, lower=low,upper=upp);
   }
  }
  p(Rand_parts.AoA,newRand_parts.AoA);
  # end of Gibbs sampling algorithm
 }
 # compute the new response vector
 RT[censored]=round(fixed[censored]+Rand_parts[censored]+Rand_items[censored]
 +Rand_parts.AoA[censored]+Err[censored],0);
 # train the model with the complete data
 model_final<-lmer(RT~AoA*Lett_cat+(1+AoA|participants)+(1|items), data,
 control = lmerControl(optimizer="bobyqa",calc.derivs = FALSE));
 # end of SEM algorithm
}
```

Chapitre 3

Traitement des données manquantes et méthode d'initialisation des paramètres pour la distribution Ex-Gaussienne

Dans ce chapitre, nous avons présenté une méthode permettant d'améliorer l'estimation des paramètres de la distribution Ex-Gaussienne pour un vecteur de réponse de petite taille. Cette méthode repose sur l'utilisation d'une initialisation bien définie du vecteur des paramètres, plutôt que d'une initialisation arbitraire. De plus, elle a également été appliquée à des données incomplètes, pour lesquelles nous avons proposé plusieurs approches de gestion. Ce chapitre, réalisé en collaboration avec mes directeurs de thèse, M. Yousri Slaoui et M. Cyril Perret, correspond à un article actuellement en révision dans le *Journal of Applied Statistics*. L'article est intitulé : *Handling Missing Data and Parameter Initialization Method for the Ex-Gaussian Distribution.*

3.1 Introduction

La modélisation des temps de réaction (`RT`) pose des défis en raison de leur asymétrie positive, ce qui signifie qu'il est plus probable d'observer des `RT` longs que ce que prédirait une distribution normale ([123]). Pour prendre en compte cette spécificité, plusieurs distributions de probabilité ont été proposées, telles que la distribution de Wald, également connue sous le nom de distribution Inverse Gaussienne ([110]), la distribution de Weibull ([82]) et la distribution Log-Normale ([121]). Toutefois, la distribution qui s'ajuste le mieux aux données observées est la distribution Ex-Gaussienne (Ex-Gauss), issue de la convolution d'une distribution normale et d'une distribution exponentielle (par exemple, [26, 46, 64, 83]).

[46] a proposé une méthode bayésienne pour estimer les trois paramètres Ex-Gaussienne (Ex-Gauss), μ, σ et τ. L'objectif de cette méthode était d'obtenir des estimations non biaisées dans des conditions restrictives d'un petit échantillon ($n < 100$), fréquemment observées dans les études de psychologie scientifique. Cette méthodologie bayésienne emploie une initialisation aléatoire des paramètres et est implémentée dans le package R, [116]. Cependant, l'utilisation d'une initialisation aléatoire conduit à un sur-ajustement des paramètres Ex-Gauss, en particulier le paramètre de distribution exponentielle. L'objectif principal de notre étude est d'améliorer la méthode d'estimation en utilisant une initialisation appropriée dans l'approche bayésienne basée

sur le rééchantillonnage, en particulier avec la méthode du Bootstrap, afin d'obtenir des estimations plus robustes. Cette approche vise à réduire le surapprentissage observé avec l'initialisation aléatoire et à améliorer la précision des paramètres de la distribution Ex-Gauss.

Récemment, les méthodes de rééchantillonnage ont suscité un intérêt croissant dans les domaines de l'éducation et de la psychologie. Cependant, de nombreux chercheurs continuent de privilégier des méthodes statistiques plus traditionnelles, souvent par manque de familiarité avec ces nouvelles techniques ou en raison de certaines barrières perçues. Parmi celles-ci, on peut citer le fossé éducatif, c'est-à-dire une formation insuffisante dans ces méthodes, les limitations des logiciels disponibles pour les mettre en œuvre, ainsi que la perception d'un risque lié à l'expérimentation avec des techniques moins éprouvées. Pour aborder ces questions, [130] présente dans son article les concepts des différents types de rééchantillonnage avec des arguments et des applications logicielles. L'une des méthodes puissantes de rééchantillonnage avec remplacement des données observées est la méthode bootstrap utilisée pour estimer la distribution d'échantillonnage d'une statistique. Depuis la fin des années 1970, cette approche a gagné en popularité en raison de sa simplicité et de son efficacité à résoudre des problèmes statistiques où elle crée des rééchantillons directement à partir des données observées elles-mêmes plutôt que de dépendre de distributions théoriques. [44] présente également la méthode Jackknife, une approche de rééchantillonnage similaire au bootstrap. La réalisation de simulations implique un processus simple lors de l'utilisation des techniques de rééchantillonnage. Le travail de [31] vise à examiner la méthodologie bootstrap existante dans le contexte de la simulation, où il explore les versions paramétriques et non paramétriques de cette méthode afin de montrer qu'elles peuvent être une solution à de nombreux problèmes.

Le bootstrap est considéré comme un outil inestimable dans l'analyse statistique moderne. Sa flexibilité le rend adapté à différentes techniques statistiques, de l'estimation simple aux modèles de régression complexes. C'est une approche qui peut fournir des estimations précises, particulièrement lorsque les approximations habituelles sont invalides. Cette technique consiste à créer plusieurs échantillons bootstrap en sélectionnant aléatoirement des observations à partir de l'ensemble de données d'origine avec remplacement. Chaque échantillon a la même taille que l'échantillon initial, permettant ainsi aux observations d'être incluses plusieurs fois ou pas du tout. L'article de [45] présente la méthode bootstrap pour l'estimation de l'erreur standard et de l'intervalle de confiance du bootstrap. Pour plus de détails sur cette méthode et ses applications pratiques, voir le tutoriel de [35, 126]. [126] a illustré des applications dans le domaine de la chimio métrie.
Dans ce chapitre, nous avons comparé notre méthode de bootstrap proposée à l'initialisation aléatoire et à la méthode de l'estimation du maximum de vraisemblance (`MLE`).

Le second objectif de ce travail est de traiter le problème des données manquantes de type `MAR`, il s'agit du cas le plus classique défini par [106] où la valeur manquante

est prédite uniquement sur la base des données observées (voir, [111]). Pour passer en revue les types de ces données non observées et explorer les stratégies analytiques pour traiter ce problème (voir, [61, 85]).
La méthode MI, déjà présentée dans le chapitre précédent (Chapitre 2), reste une approche clé pour traiter les données manquantes. Pour rappel, elle consiste à remplacer les valeurs manquantes par plusieurs ensembles de valeurs plausibles, permettant des inférences statistiques plus robustes.
[56] a utilisé la méthode MI pour les données manquantes dans des modèles multiniveaux où la distribution conjointe est décomposée en plusieurs parties qui correspondent à la réponse et aux variables explicatives. [79] a présenté un article bref pour aborder de nombreuses questions sur la méthode d'imputation multiple, telles que les domaines d'application de cette méthode, ses limitations et comment interpréter les résultats. Cette méthode est également présentée par [53] sur la base d'un modèle de régression hiérarchique pour résoudre le problème de l'imputation pour plusieurs enquêtes au sein d'une seule. Certains des problèmes liés aux données manquantes sont discutés dans l'article de [125], où il présente également le processus d'imputation multiple avec un exemple illustrant chaque étape pour offrir une compréhension plus claire du processus.
Dans notre étude, nous avons proposé de comparer la méthode MI avec trois autres méthodologies documentées dans la littérature. Pour de plus amples informations sur l'approche proposée, veuillez vous référer à [5, 107].

Afin de valider nos propositions, nous avons illustré leur application en utilisant à la fois des données simulées et des données réelles. La comparaison est effectuée en utilisant l'augmentation du risque absolu (Absolute Risk Increase, ARI) pour les vecteurs de paramètres, obtenus par notre méthode proposée et l'approche précédemment utilisée.

Ce chapitre est organisé comme suit : La section 3.2 présente notre méthodologie, qui est divisée en deux sous-sections 3.2.1 et 3.2.2. Dans la sous-section 3.2.1, nous rappelons la distribution Ex-Gauss avec une présentation de la fonction de vraisemblance maximale. Ensuite, l'algorithme de bootstrap que nous avons proposé, utilisé pour initialiser le vecteur de paramètres de départ dans l'estimation bayésienne, est présenté dans la sous-section 3.2.2. Dans la section 3.3, nous comparons les résultats obtenus à partir des données simulées et réelles. Enfin, dans la section 3.4, nous concluons l'étude proposée pour obtenir des estimations non biaisées des trois paramètres de la distribution Ex-Gaussienne avec des échantillons de petite taille avec ou sans Na.

3.2 Initialisation des paramètres

Comme mentionné dans l'introduction, la distribution Ex-Gauss est largement utilisée pour modéliser les RTs en raison de sa capacité à ajuster les données asymétriques. L'estimation des paramètres de cette distribution, représentés par le vecteur

$\theta = (\mu, \zeta, \lambda)$, nécessite une méthode spécifique, et l'initialisation de ce vecteur est cruciale pour garantir des estimations fiables.

[46] a utilisé des valeurs aléatoires comme point de départ pour estimer les paramètres via une approche bayésienne, tandis que notre méthodologie repose également sur une inférence bayésienne, mais avec des valeurs initiales dérivées de la méthode de rééchantillonnage Bootstrap. Par ailleurs, nous avons abordé un problème fréquent dans le domaine psychologique : l'occurrence de données manquantes dans le vecteur de réponse. Pour résoudre ce problème, nous avons proposé une comparaison de la méthode d'Imputation Multiple (`MI`) avec trois autres alternatives.

Avant de présenter notre méthodologie, rappelons brièvement les propriétés de la distribution Ex-Gauss utilisée dans cette étude, dont l'objectif est d'obtenir des estimations non biaisées pour des petits échantillons.

3.2.1 Distribution Ex-Gauss

Une variable aléatoire Z qui suit une distribution Ex-Gauss ([46]) peut être décrite comme la somme de deux variables aléatoires indépendantes X et Y. Où X suit une distribution gaussienne avec une moyenne μ et un écart type σ, et Y est exponentielle de taux λ.
Avec $x = (x_1, \ldots, x_n)$ étant le vecteur de réponse de n observations et $\tau = \frac{1}{\lambda}$, cette distribution possède une fonction de densité de probabilité écrite comme suit :

$$f(x|\mu, \sigma, \tau) = \frac{1}{\tau} \exp\left(\frac{\mu - x}{\tau} + \frac{\sigma^2}{2\tau^2}\right) \phi\left(\frac{x-\mu}{\sigma} - \frac{\sigma}{\tau}\right), \tag{3.1}$$

où $\phi(z)$ de l'équation (3.1) est la fonction de répartition de la distribution gaussienne standard évaluée de $-\infty$ à z et est donnée par :

$$\phi\left(\frac{x-\mu}{\sigma} - \frac{\sigma}{\tau}\right) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\frac{x-\mu}{\sigma} - \frac{\sigma}{\tau}} \exp\left(-\frac{y^2}{2}\right) dy.$$

Pour simplifier le calcul, nous prenons $\zeta = \sigma^2$ et $\lambda = \frac{1}{\tau}$. Soit $\theta = (\mu, \zeta, \lambda)$ l'ensemble des paramètres à estimer.

[92] a fourni une explication didactique de la méthode d'Estimation du Maximum de Vraisemblance (`MLE`) pour la distribution Ex-Gauss avec des exemples illustratifs. La fonction de vraisemblance de cette distribution, étant donné les paramètres, peut être écrite comme suit :

$$L(\mu, \zeta, \lambda) = \lambda^n \exp\left(\sum_{i=1}^{n} (\mu - x_i)\lambda + \frac{n\lambda^2\zeta}{2}\right) \Pi_{i=1}^{n} \phi\left(\frac{x_i - \mu}{\sqrt{\zeta}} - \lambda\sqrt{\zeta}\right), \tag{3.2}$$

À partir de l'équation (3.2), la fonction de log-vraisemblance est écrite comme suit :

$$\begin{aligned} l &= \log L(\mu, \zeta, \lambda) \\ &= n \log \lambda + \left(\sum_{i=1}^{n} (\mu - x_i)\lambda + \frac{n\lambda^2 \zeta}{2} \right) \\ &\quad + \sum_{i=1}^{n} \log \phi \left(\frac{x_i - \mu}{\sqrt{\zeta}} - \lambda \sqrt{\zeta} \right). \end{aligned} \tag{3.3}$$

Pour estimer notre vecteur de paramètres $\theta = (\mu, \zeta, \lambda)$, nous devons maximiser la fonction de log-vraisemblance l présentée dans l'équation (3.3). Cette maximisation est obtenue en mettant les dérivées partielles de l par rapport à θ égales à zéro :

$$\begin{cases} \dfrac{\partial l}{\partial \mu} &= n\lambda + \sum_{i=1}^{n} \dfrac{\Phi_1}{\phi\left(\dfrac{x_i - \mu}{\sqrt{\zeta}} - \lambda\sqrt{\zeta} \right)} \\ \dfrac{\partial l}{\partial \zeta} &= \dfrac{n\lambda^2}{2} + \sum_{i=1}^{n} \dfrac{\Phi_2}{\phi\left(\dfrac{x_i - \mu}{\sqrt{\zeta}} - \lambda\sqrt{\zeta} \right)} \\ \dfrac{\partial l}{\partial \lambda} &= \dfrac{n}{\lambda} + \sum_{i=1}^{n} (\mu - x) + n\zeta\lambda + \sum_{i=1}^{n} \dfrac{\Phi_3}{\phi\left(\dfrac{x_i - \mu}{\sqrt{\zeta}} - \lambda\sqrt{\zeta} \right)} \end{cases}$$

Où les fonctions Φ_1, Φ_2 et Φ_3 sont les dérivées de la fonction ϕ par rapport à chaque paramètre et sont définies comme suit :

$$\begin{cases} \Phi_1 &= \dfrac{\partial \phi\left(\dfrac{x - \mu}{\sqrt{\zeta}} - \lambda\sqrt{\zeta} \right)}{\partial \mu} = -\dfrac{1}{\zeta} \times \phi' \left(\dfrac{x - \mu}{\sqrt{\zeta}} - \lambda\sqrt{\zeta} \right) \\ \Phi_2 &= \dfrac{\partial \phi\left(\dfrac{x - \mu}{\sqrt{\zeta}} - \lambda\sqrt{\zeta} \right)}{\partial \zeta} = \left(-\dfrac{x - \mu}{2\zeta^{\frac{3}{2}}} - \dfrac{\lambda}{2\sqrt{\zeta}} \right) \times \phi' \left(\dfrac{x - \mu}{\sqrt{\zeta}} - \lambda\sqrt{\zeta} \right) \\ \Phi_3 &= \dfrac{\partial \phi\left(\dfrac{x - \mu}{\sqrt{\zeta}} - \lambda\sqrt{\zeta} \right)}{\partial \lambda} = -\sqrt{\zeta} \times \phi' \left(\dfrac{x - \mu}{\sqrt{\zeta}} - \lambda\sqrt{\zeta} \right) \end{cases}$$

Dans la sous-section suivante, nous présentons notre proposition utilisant l'algorithme de Bootstrap pour initialiser le vecteur de paramètres, ce qui nous permet d'obtenir une estimation plus précise.

3.2.2 Initialisation par Bootstrap en Estimation Bayésienne

L'Algorithme 3 représente la procédure générale de Bootstrap. Nous commençons avec un ensemble de données initial $D = \{x_1, x_2, \ldots, x_n\}$ contenant n observations. Un échantillon bootstrap, noté $D = \{x_{i_1}^*, x_{i_2}^*, \ldots, x_{i_n}^*\}$, est généré en sélectionnant

aléatoirement n observations à partir de D avec remplacement. Cela signifie que D^* est un nouvel ensemble de données obtenu à partir de D, où la même observation peut être sélectionnée plusieurs fois en raison du processus de remplacement. Pour chaque échantillon bootstrap D^*, nous calculons la statistique d'intérêt (telle que la moyenne, la variance, ...). Cette procédure est répétée B fois.

Algorithm 3 The General Procedure of Bootstrap : D is the initial data set, B is the Number of bootstrap sample, and F is the function of statistic interest.

Input: D, B and F.

1: **for** $b = 1, \ldots, B$ **do**

2: $D_b^* =$ sample with replacement n elements of D ;
$F_b^* =$ statistic(D_b^*).

3: **end for**

output: $F^* = \frac{1}{B}\sum_{b=1}^{B} F_b^*$

Algorithm 4 The Proposed Bootstrap application : D is the initial data set ; B is the Number of bootstrap sample ; F_μ, F_ζ and F_λ represent the statistic interest of μ, ζ and λ.

Input: D, B, F_μ, F_ζ and F_λ.

1: **for** $b = 1, \ldots, B$ **do**

2: $D_b^* =$ sample with replacement n elements of D ;
$F_{\mu,b}^* = \mathbb{E}(D_b^*) - \left(0.5 * \sqrt{\mathbb{V}(D_b^*)}\right)$;
$F_{\zeta,b}^* = \mathbb{V}(D_b^*) - \left(0.5 * \sqrt{\mathbb{V}(D_b^*)}\right)^2$;
$F_{\lambda,b}^* = \frac{1}{0.5 * \sqrt{\mathbb{V}(D_b^*)}}$.

3: **end for**

output: $F_\mu^* = \frac{1}{B}\sum_{b=1}^{B} F_{\mu,b}^*$, $F_\zeta^* = \frac{1}{B}\sum_{b=1}^{B} F_{\zeta,b}^*$ and $F_\lambda^* = \frac{1}{B}\sum_{b=1}^{B} F_{\lambda,b}^*$.

Dans l'approche bayésienne, nous commençons avec des distributions a priori sur les paramètres qui doivent être estimés. En notant θ les paramètres à prédire et x le vecteur d'échantillons observés, la densité a posteriori de θ en utilisant le théorème de Bayes est donnée par :

$$\mathbb{P}(\theta|x) = \frac{\mathbb{P}(x|\theta)\mathbb{P}(\theta)}{\mathbb{P}(x)}.$$

Nous étudions la distribution Ex-Gaussienne où le vecteur θ contient les trois paramètres à estimer, $\theta = (\mu, \zeta, \lambda)$, avec la densité de probabilité $f(x|\mu, \zeta, \lambda)$ présentée dans l'équation (3.1). En appliquant Bayes' avec nos paramètres, nous obtenons la distribution a posteriori suivante :

$$f(\mu, \zeta, \lambda|x) = \frac{f(x|\mu, \zeta, \lambda)f(\mu)f(\zeta)f(\lambda)}{f(x)},$$

où $f(\mu)$, $f(\zeta)$ et $f(\lambda)$ sont les densités a priori pour le vecteur de paramètres θ et $f(x)$ est la probabilité de x.

La première étape consiste à choisir la distribution de probabilité a priori pour les paramètres. Au lieu de prendre une initialisation arbitraire (μ_0, ζ_0 et λ_0), nous proposons d'appliquer la méthode de bootstrap à chaque paramètre. Cette méthode est résumée dans l'Algorithme 4, où l'initialisation de [46] est utilisée pour le calcul de la statistique d'intérêt pour chaque échantillon bootstrap. En prenant B comme le nombre d'échantillons bootstrap, nous définissons les valeurs initiales comme suit :

$$\begin{cases} \mu &= \frac{1}{B}\sum_{b=1}^{B}\mu_b \\ \zeta &= \frac{1}{B}\sum_{b=1}^{B}\zeta_b \\ \lambda &= \frac{1}{B}\sum_{b=1}^{B}\lambda_b \end{cases} \tag{3.4}$$

Avec μ_b, ζ_b et λ_b étant les paramètres μ, ζ et λ respectivement obtenus dans chaque échantillon bootstrap. Ces valeurs sont utilisées dans les distributions a priori pour obtenir la distribution a posteriori. Comme $f(x)$ est indépendant des paramètres, nous pouvons dire que la probabilité a posteriori est proportionnelle au numérateur et s'écrit comme suit :

$$f(\mu,\zeta,\lambda|x) \propto f(x|\mu,\zeta,\lambda)f(\mu)f(\zeta)f(\lambda).$$

En prenant les valeurs de μ, ζ et λ présentées dans l'équation (3.4), $f(x|\mu,\zeta,\lambda)$ s'exprime comme suit :

$$f(x|\mu,\zeta,\lambda) \propto \frac{1}{B^n}\left(\sum_{b=1}^{B}\lambda_b\right)^n \exp\left(\frac{1}{B}\sum_{i=1}^{n}\left(\frac{1}{B}\sum_{b=1}^{B}\mu_b - x_i\right)\sum_{b=1}^{B}\lambda_b + \frac{n\left(\sum_{b=1}^{B}\lambda_b\right)^2\sum_{b=1}^{B}\zeta_b}{2B^3}\right)$$
$$\times \prod_{i=1}^{n}\phi\left(\frac{\left(x_i - \frac{1}{B}\sum_{b=1}^{B}\mu_b\right)}{\sqrt{\frac{1}{B}\sum_{b=1}^{B}\zeta_b}} - \frac{\sum_{b=1}^{B}\lambda_b\sqrt{\sum_{b=1}^{B}\zeta_b}}{B\sqrt{B}}\right). \tag{3.5}$$

Nous avons travaillé avec les mêmes distributions de paramètres que dans [46], et nous avons donc implémenté les distributions suivantes :

$$\begin{cases} \mu \sim \mathcal{N}(\alpha_1,\alpha_2) \\ \zeta \sim \mathcal{IG}(\beta_1,\beta_2) \\ \lambda \sim \mathcal{G}(\gamma_1,\gamma_2) \end{cases} \tag{3.6}$$

avec $\mathcal{N}$, $\mathcal{IG}$ et $\mathcal{G}$ représentant respectivement les distributions normale, inverse-Gamma et Gamma.

Par conséquent, les densités de ces paramètres (équation (3.6)) s'écrivent comme suit :

$$f(\mu) = \frac{1}{\alpha_2\sqrt{2\pi}}\exp\big(-\frac{1}{2}\big(\frac{\mu-\alpha_1}{\alpha_2}\big)^2\big); \tag{3.7}$$

$$f(\zeta) = \frac{(\beta_2)^{\beta_1}(\frac{1}{\zeta})^{\beta_1+1}}{\Gamma(\beta_1)}\exp\big(\frac{-\beta_2}{\zeta}\big); \tag{3.8}$$

$$f(\lambda) = \frac{\lambda^{(\gamma_1-1)}\gamma_2^{\gamma_1}}{\Gamma(\gamma_1)} \exp\big(-\gamma_2\lambda\big). \tag{3.9}$$

Les valeurs des paramètres μ, ζ et λ présentées dans l'équation (3.4) sont substituées dans les équations (3.7), (3.8) et (3.9) pour obtenir :

$$f(\mu) = \frac{1}{\alpha_2\sqrt{2\pi}} \exp\left(-\frac{1}{2}\left(\frac{\frac{1}{B}\sum_{b=1}^{B}\mu_b - \alpha_1}{\alpha_2}\right)^2\right); \tag{3.10}$$

$$f(\zeta) = \frac{(\beta_2)^{\beta_1}}{\Gamma(\beta_1)} \times \left(\frac{B}{\sum_{b=1}^{B}\zeta_b}\right)^{\beta_1+1} \exp\left(\frac{-B\beta_2}{\sum_{b=1}^{B}\zeta_b}\right); \tag{3.11}$$

$$f(\lambda) = \frac{(B^{-1}\sum_{b=1}^{B}\lambda_b)^{\gamma_1-1}\gamma_2^{\gamma_1}}{\Gamma(\gamma_1)} \exp\left(-\frac{\gamma_2}{B}\sum_{b=1}^{B}\lambda_b\right). \tag{3.12}$$

De plus, nous notons que la distribution conditionnelle de μ est donnée par :

$$\begin{aligned} f(\mu|x,\zeta,\lambda) &= \frac{f(x,\mu,\zeta,\lambda)}{f(x,\zeta,\lambda)} \\ &= \frac{f(\mu,\zeta,\lambda|x)}{f(\zeta,\lambda|x)}; \end{aligned} \tag{3.13}$$

Puisque $f(\mu,\zeta,\lambda|x)$ est proportionnel à $f(x,\mu,\zeta,\lambda)$ et que $f(\zeta,\lambda)$ n'implique pas le paramètre μ, nous pouvons réécrire l'équation (3.13) comme suit :

$$f(\mu|x,\zeta,\lambda) \propto f(x,\mu,\zeta,\lambda). \tag{3.14}$$

La densité conjointe $f(x,\mu,\zeta,\lambda)$ est obtenue en multipliant $f(x|\mu,\zeta,\lambda)$ dans l'équation (3.5) par les densités a priori $f(\mu)$, $f(\zeta)$ et $f(\lambda)$ dans les équations (3.10), (3.11) et (3.12). Ainsi, l'expression finale est présentée par :

$$\begin{aligned} f(x,\mu,\zeta,\lambda) \propto{}& \left(\sum_{b=1}^{B}\lambda_b\right)^n \exp\left(\frac{1}{B}\sum_{i=1}^{n}\left(\frac{1}{B}\sum_{b=1}^{B}\mu_b - x_i\right)\sum_{b=1}^{B}\lambda_b + \frac{n\left(\sum_{b=1}^{B}\lambda_b\right)^2\sum_{b=1}^{B}\zeta_b}{2B^3}\right) \\ &\times \prod_{i=1}^{n}\phi\left(\frac{\left(x_i - \frac{1}{B}\sum_{b=1}^{B}\mu_b\right)}{\sqrt{\frac{1}{B}\sum_{b=1}^{B}\zeta_b}} - \frac{\sum_{b=1}^{B}\lambda_b\sqrt{\sum_{b=1}^{B}\zeta_b}}{B\sqrt{B}}\right) \\ &\times \frac{1}{\alpha_2}\exp\left(-\frac{1}{2}\left(\frac{\frac{1}{B}\sum_{b=1}^{B}\mu_b - \alpha_1}{\alpha_2}\right)^2\right) \\ &\times \frac{\beta_2^{\beta_1}}{\Gamma(\beta_1)}\left(\frac{B}{\sum_{b=1}^{B}\zeta_b}\right)^{\beta_1+1}\exp\left(\frac{-B\beta_2}{\sum_{b=1}^{B}\zeta_b}\right) \\ &\times \frac{\left(B^{-1}\sum_{b=1}^{B}\lambda_b\right)^{\gamma_1-1}\gamma_2^{\gamma_1}}{\Gamma(\gamma_1)}\exp\left(-\frac{\gamma_2}{B}\sum_{b=1}^{B}\lambda_b\right). \end{aligned} \tag{3.15}$$

En se basant sur l'équation (3.14), nous pouvons dériver la distribution conditionnelle pour chaque paramètre en considérant un à la fois et en ignorant les deux autres dans l'équation (3.15). Par conséquent, les distributions a posteriori conditionnelles de μ, ζ et λ peuvent être exprimées comme suit :

$$\begin{aligned} f(\mu|x,\zeta,\lambda) \propto \exp & \left(\frac{1}{B} \sum_{i=1}^{n} \left(\frac{1}{B} \sum_{b=1}^{B} \mu_b - x_i \right) \sum_{b=1}^{B} \lambda_b \right) \\ & \times \prod_{i=1}^{n} \phi \left(\frac{\left(x_i - \frac{1}{B} \sum_{b=1}^{B} \mu_b \right)}{\sqrt{\frac{1}{B} \sum_{b=1}^{B} \zeta_b}} - \frac{\sum_{b=1}^{B} \lambda_b \sqrt{\sum_{b=1}^{B} \zeta_b}}{B\sqrt{B}} \right) \\ & \times \exp \left(- \frac{\left(\frac{1}{B} \sum_{b=1}^{B} \mu_b - \alpha_1 \right)^2}{2\alpha_2^2} \right). \end{aligned} \tag{3.16}$$

$$\begin{aligned} f(\zeta|x,\mu,\lambda) \propto \exp & \left(\frac{n \sum_{b=1}^{B} \zeta_b \left(\sum_{b=1}^{B} \lambda_b \right)^2}{2B^3} \right) \\ & \times \phi \left(\frac{\left(x_i - \frac{1}{B} \sum_{b=1}^{B} \mu_b \right)}{\sqrt{\frac{1}{B} \sum_{b=1}^{B} \zeta_b}} - \frac{\sum_{b=1}^{B} \lambda_b \sqrt{\sum_{b=1}^{B} \zeta_b}}{B\sqrt{B}} \right) \\ & \times \left(\frac{B}{\sum_{b=1}^{B} \zeta_b} \right)^{\beta_1+1} \exp \left(\frac{-B\beta_2}{\sum_{b=1}^{B} \zeta_b} \right). \end{aligned} \tag{3.17}$$

$$\begin{aligned} f(\lambda|x,\mu,\zeta) \propto & \left(\sum_{b=1}^{B} \lambda_b \right)^n \exp \left(\frac{1}{B} \sum_{i=1}^{n} \left(\frac{1}{B} \sum_{b=1}^{B} \mu_b - x_i \right) \sum_{b=1}^{B} \lambda_b + \frac{n \left(\sum_{b=1}^{B} \lambda_b \right)^2 \sum_{b=1}^{B} \zeta_b}{2B^3} \right) \\ & \times \prod_{i=1}^{n} \phi \left(\frac{\left(x_i - \frac{1}{B} \sum_{b=1}^{B} \mu_b \right)}{\sqrt{\frac{1}{B} \sum_{b=1}^{B} \zeta_b}} - \frac{\sum_{b=1}^{B} \lambda_b \sqrt{\sum_{b=1}^{B} \zeta_b}}{B\sqrt{B}} \right) \\ & \times \left(B^{-1} \sum_{b=1}^{B} \lambda_b \right)^{\gamma_1-1} \exp \left(-\frac{\gamma_2}{B} \sum_{b=1}^{B} \lambda_b \right). \end{aligned} \tag{3.18}$$

Par conséquent, les équations (3.16), (3.17) et (3.18) sont les fonctions d'estimation conditionnelle des paramètres dans lesquelles la méthode de bootstrap proposée, résumée dans l'Algorithm 4, est appliquée pour initialiser le vecteur de paramètres avant que l'approche bayésienne ne soit calculée.

Dans la dernière partie de notre investigation, nous visons à aborder le problème des données manquantes (`NA`) de type `MAR`.

3.3 Application Numérique

L'objectif de cette section est de comparer notre méthode de bootstrap proposée, présentée dans Algorithme 4, à deux autres méthodes pour l'initialisation des paramètres, puis nous nous intéressons à trouver une méthode pour traiter le problème des `Na`. Nous avons exploré notre approche en utilisant des données simulées et réelles.

Pour la méthode d'initialisation, la première approche comparée était l'initialisation arbitraire utilisée par [46] (appelée par la suite : méthode `Arb.`), où les paramètres initiaux sont choisis comme suit :

$$\mu^{(0)} = \mathbb{E}(x) - 0.8 \times \sqrt{\mathbb{V}(x)};\ \zeta^{(0)} = 0.36 \times \mathbb{V}(x);\ \lambda^{(0)} = \frac{1}{0.8 \times \sqrt{\mathbb{V}(x)}}.$$

Dans la deuxième méthode, les paramètres initiaux sont obtenus en utilisant la méthode `MLE` présentée dans la sous-section 3.2.1. Ces deux méthodes (`Arb.` et `MLE`) sont comparées à la méthode Bootstrap proposée, où le vecteur initial est pris comme suit :

$$\mu^{(0)} = \frac{1}{B}\sum_{b=1}^{B}\mu_b^*,\ \zeta^{(0)} = \frac{1}{B}\sum_{b=1}^{B}\zeta_b^*,\ \lambda^{(0)} = \frac{1}{B}\sum_{b=1}^{B}\lambda_b^*.$$

En prenant les paramètres de chaque échantillon Bootstrap comme suit :

$$\mu_b^* = \mathbb{E}(x_b^*) - 0.5 \times \sqrt{\mathbb{V}(x_b^*)};\ \zeta_b^* = 0.75 \times \mathbb{V}(x_b^*);\ \lambda_b^* = \frac{1}{0.5 \times \sqrt{\mathbb{V}(x_b^*)}}$$

avec x_b^* étant l'échantillon Bootstrap (échantillon de x avec remplacement).

La taille de l'échantillon simulé est $n = 50$. Le choix de cette valeur est basé sur le fait qu'elle correspond à une taille d'échantillon expérimentale couramment utilisée en psychologie scientifique.

3.3.1 Données simulées

Nous simulons trois ensembles de données $x = (x_1, \ldots, x_n)$ qui suivent la distribution Ex-Gauss, avec $n = 50$ observations. Les paramètres utilisés dans les trois simulations sont les suivants :

- **Data1** : $\{\mu = 4, \sigma = 2, \tau = 1\}$
- **Data2** : $\{\mu = 40, \sigma = 20, \tau = 10\}$
- **Data3** : $\{\mu = 400, \sigma = 200, \tau = 100\}$

Les deux premiers ensembles de paramètres (Data1 et Data2) sont conçus pour tester si la taille des paramètres a un impact sur la qualité des estimations. Le troisième ensemble de paramètres a été choisi pour correspondre aux valeurs classiquement obtenues dans les études en psychologie utilisant les temps de réaction. Depuis les

travaux de [40], le temps nécessaire à un participant pour produire une action motrice suite à la présentation d'un stimulus (par exemple, un éclat de lumière, un son, ...) est considéré comme impliquant trois phases : le traitement perceptuel d'environ 150 ms, un traitement cognitif très court (moins de 50 ms) et le traitement moteur d'environ 150 ms. Étant donné qu'il existe une variabilité considérable entre les différentes réponses d'un participant, la série de paramètres $\mu = 400$, $\sigma = 200$ et $\tau = 100$ semble assez proche de ce qui serait obtenu pour un participant typique dans une expérience classique.

Nous estimons les paramètres de ces ensembles de données en utilisant la méthode bayésienne proposée par [46], avec un niveau de rodage fixé à 2500 itérations sur 5000 échantillons de Gibbs.

Pour visualiser la comparaison de nos résultats, nous nous basons sur la Réduction du Risque Relatif (`RRR`) et l'Augmentation du Risque Relatif (Absolute Risk Increase, `RRI`) (voir [12]), qui ont une forme générale de :

$$\begin{aligned} RRR = \frac{\theta - \hat{\theta}}{\theta} &= 1 - RR \\ &= -RRI, \end{aligned}$$

où θ est le vecteur de paramètres réels, $\hat{\theta}$ est le vecteur de paramètres prédit et RR ([118]) est le Risque Relatif (`RR`) défini comme suit :

$$RR = \frac{\hat{\theta}}{\theta}.$$

Dans cette étude, notre vecteur de paramètres θ contient les trois paramètres μ, ζ et λ. Ainsi, pour comparer les méthodes, nous avons utilisé la formule suivante de `ARI` :

$$\begin{aligned} \texttt{ARI} &= \sum_{i=1}^{p} |RR_i - 1| \\ &= \sum_{i=1}^{p} |RRI_i|. \end{aligned} \tag{3.19}$$

Avec p étant la longueur du vecteur de paramètres θ (le nombre de paramètres à estimer). Ce critère `ARI` dans l'équation (3.19) peut être interprété comme :

$$\texttt{ARI} \approx \begin{cases} 0 & \text{si la prédiction est précise} \\ 1 & \text{sinon} \end{cases}$$

Nous avons également calculé la Corrélation de Rang de Spearman (`Rcor`), connue en analyse statistique sous la forme :

$$\texttt{Rcor} = 1 - \frac{6 \times \sum d^2}{n(n^2 - 1)},$$

où d est la différence entre deux rangs et n est la longueur du vecteur de paramètres.

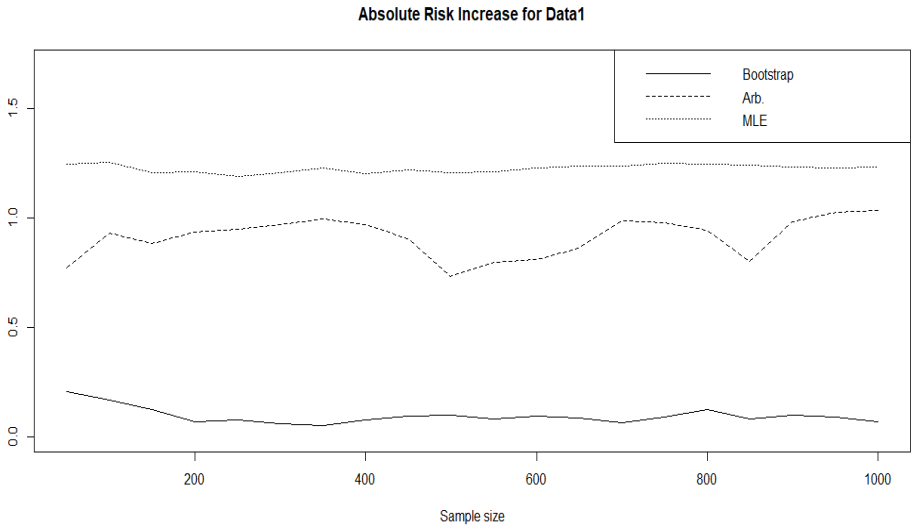
L'analyse est réalisé sur des données simulées à travers les trois exemples (Data1, Data2 et Data3). Le Tableau 3.1 illustre une comparaison entre les méthodes `Arb.` et `MLE` avec l'approche Bootstrap proposée pour l'initialisation des paramètres. Nous avons marqué en gras pour chaque cas la valeur la plus proche de la référence parmi les trois méthodes considérées (les vecteurs de référence ici étaient les données simulées).

	Data1			
	Reference	`Arb.`	`MLE`	Bootstrap
μ	**4**	3.43	4.66	**3.67**
σ	**2**	1.57	1.62	**1.64**
τ	**1**	0.67	0.16	**0.96**
`ARI`	**0**	0.67	1.02	**0.28**
`Rcor`	**1**	**1**	**1**	**1**
	Data2			
	Reference	`Arb.`	`MLE`	Bootstrap
μ	**40**	25.90	51.04	**41.17**
σ	**20**	16.07	**19.53**	18.79
τ	**10**	20.54	1.18	**6.08**
`ARI`	**0**	1.60	1.18	**0.48**
`Rcor`	**1**	-0.5	**1**	**1**
	Data3			
	Reference	`Arb.`	`MLE`	Bootstrap
μ	**400**	72.25	497.63	**397.13**
σ	**200**	42.77	**203.78**	181.96
τ	**100**	420.41	12.17	**80.19**
`ARI`	**0**	4.80	1.14	**0.29**
`Rcor`	**1**	-0.5	**1**	**1**

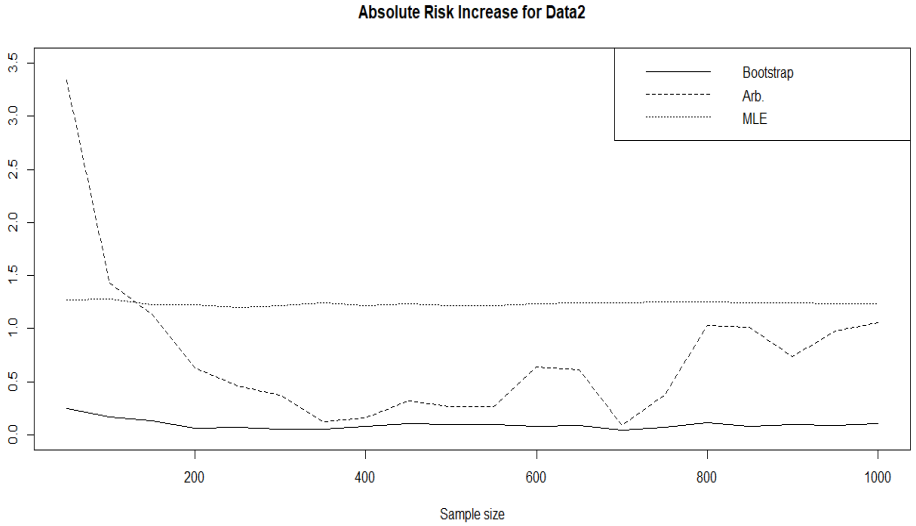
TABLE 3.1 – Estimation des paramètres et valeurs de la mésure `ARI` pour la distribution Ex-Gaussienne dans les bases de données simulées complètes Data1, Data2 et Data3.

Sur la base des valeurs `Rcor` présentées dans le Tableau 3.1, il est évident que l'initialisation arbitraire est la moins efficace. De plus, selon les critères `ARI`, la méthode Bootstrap s'avère la plus efficace, car elle donne systématiquement les valeurs les plus basses à travers les trois ensembles de données simulées. Par exemple, les valeurs `ARI` pour la méthode Arb. dans Data3 sont de 4.8, pour la méthode `MLE` sont de 1.14, et pour la méthode Bootstrap sont de 0.29. Ces résultats confirment la validité de notre proposition.

La Figure 3.1 fournit une validation supplémentaire pour la méthode de bootstrap, en représentant les valeurs comparées des critères `ARI` présentées dans le Tableau 3.1. Il est évident d'après la Figure 3.1 que la méthode de bootstrap proposée donne les plus petites valeurs de `ARI`, qui se rapprochent de zéro à mesure que la taille de l'échantillon augmente. De plus, la figure montre que l'initialisation arbitraire présente des valeurs instables des critères `ARI`, ce qui indique sa variabilité et son manque de fiabilité par rapport à la méthode de Bootstrap.

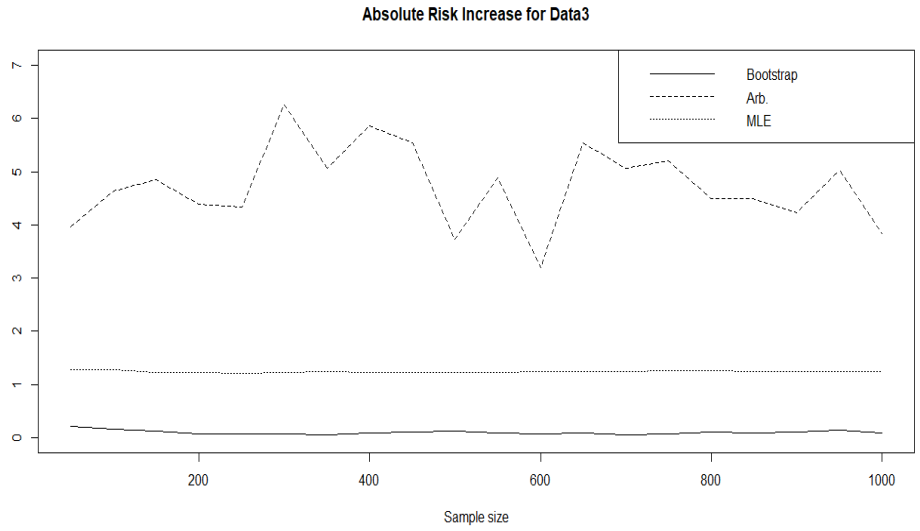


(**a**)



(**b**)

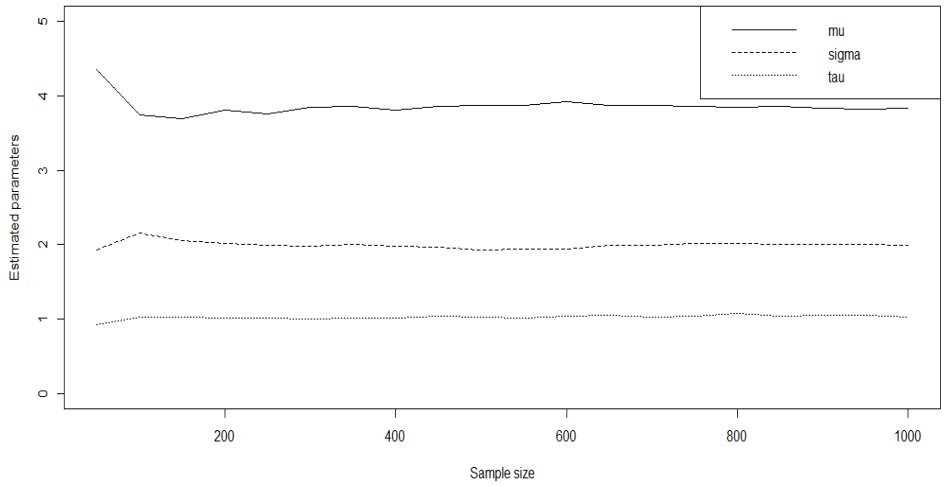
FIGURE 3.1 – Variation des critères ARI en fonction de la taille de l'échantillon à travers les trois méthodes ; (**a**)-Data1, (**b**)-Data2, (**c**)-Data3.



(**c**)

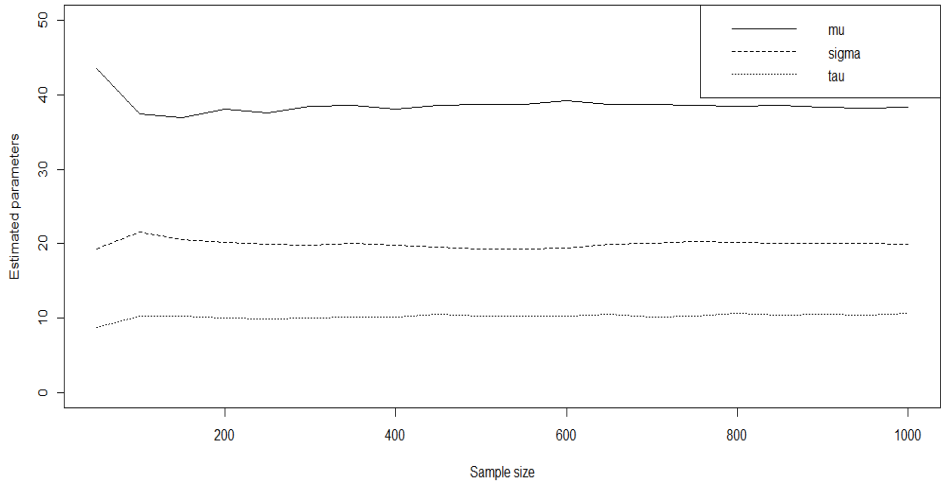
FIGURE 3.1 – *Cont.*

La Figure 3.2 montre la convergence de ces paramètres dans la méthode de bootstrap vers les valeurs réelles pour Data1 dans la Figure 3.2(a) ($\mu = 4$, $\sigma = 2$, $\tau = 1$), Data2 dans la Figure 3.2(b) ($\mu = 40$, $\sigma = 20$, $\tau = 10$) et Data3 dans la Figure 3.2(c) ($\mu = 400$, $\sigma = 200$, $\tau = 100$).

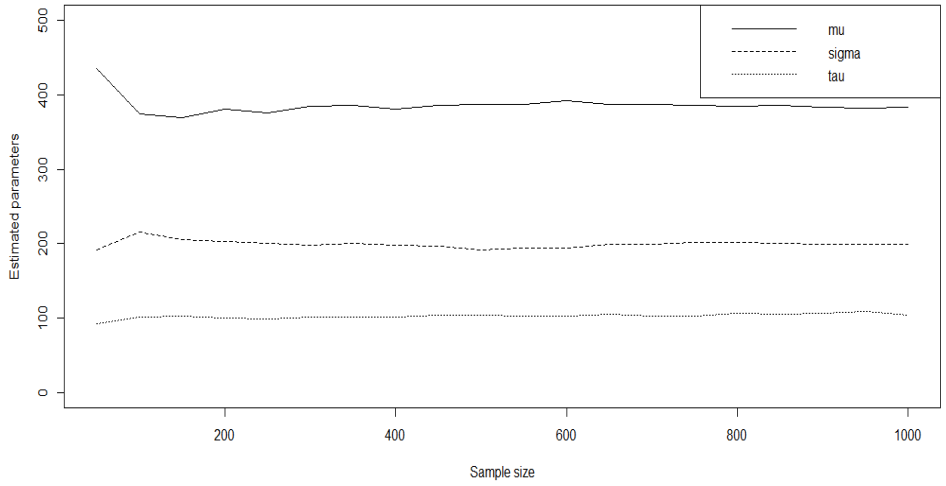


(**a**)

FIGURE 3.2 – Valeurs des paramètres estimés obtenues pour diverses tailles d'échantillon en utilisant la méthode de bootstrap ; (**a**)-Data1, (**b**)-Data2, (**c**)-Data3.



(**b**)



(**c**)

FIGURE 3.2 – *Cont.*

Solutions pour les Données Incomplètes

Maintenant, en utilisant la méthode de bootstrap proposée pour l'initialisation, nous étudions les pourcentages d'imputation des `Na` de type `MAR` avec 5%, 10%, 15% à travers quatre méthodes :

— Méthode standard : C'est une méthode très connue où les valeurs `Na` sont remplacées par la moyenne des observations, appelée plus tard `"Std"`.

— Méthode des K plus proches voisins : Dans cette méthode, les valeurs `Na` sont

prédites en fonction des valeurs des voisins observés les plus proches.
Appelée méthode "K-NN", elle a été proposée par [49] et développée par la suite par [34]. Pour plus de détails sur l'approche K-NN, des informations supplémentaires peuvent être trouvées dans [59] et [117] dans la littérature.

— Méthode bayésienne : Les valeurs manquantes sont prédites en utilisant l'estimation bayésienne avec échantillonnage de Gibbs. Cette méthode est appelée plus tard "B".

— Méthode de l'imputation multiple : Les valeurs Na sont remplacées par des valeurs numériques, cette technique est répétée plusieurs fois et les résultats sont combinés. Elle est appelée méthode "MI".

Le Tableau 3.2 présente les méthodes d'imputation des valeurs Na à travers les trois données simulées (Data1, Data2 et Data3) en présence de 5%, 10%, 15% ou 20% de valeurs MAR. La comparaison est également effectuée sur les valeurs de Rcor et ARI pour les méthodes comparées : Std, K-NN, B et MI.
Comme dans le Tableau 3.1, nous avons mis en gras la valeur la plus proche de la référence pour chaque cas à travers toutes les méthodes considérées. Dans ce contexte, les données simulées font référence au vecteur sans aucune valeur manquante.

D'après le Tableau 3.2, il est évident que les résultats manquent de stabilité. Lors de la prise en compte de la corrélation, toutes les méthodes reçoivent systématiquement le même classement dans tous les cas.
Même selon les critères ARI, les résultats ne permettent pas de déterminer la meilleure méthode pour résoudre le problème des données manquantes.
Pour un taux de valeurs manquantes de 5%, les critères ARI sont les plus bas pour la méthode MI dans Data2 et Data3, tandis que la valeur la plus petite dans Data1 est pour la méthode K-NN (0.31).
Avec 10% de Na, la méthode B est la meilleure pour Data1 et Data2. Pour Data3, le critère ARI pour la méthode MI a une valeur de 0.28, identifiée comme la plus basse parmi les autres.
Dans les premières données simulées (Data1) avec un taux MAR de 15%, les méthodes B et d'Imputation Multiple affichent les plus petites valeurs. Dans Data2, la méthode MI a la valeur la plus basse de ARI, à 0.29. Cependant, dans Data3, la méthode B a la plus petite valeur de ARI, soit 0.24.
Pour un taux de Na de 20%, chaque méthode s'avère plus efficace pour un ensemble de données différent. Dans Data1, la méthode MI affiche une petite valeur de ARI, égale à 0.31. En revanche, dans Data2, la méthode B produit la meilleure valeur de ARI (0.32), tandis que dans Data3, la méthode Std détient la plus petite valeur de ARI, soit 0.29.

Data1									
		5%				10%			
	Reference	Std	K-NN	B	MI	Std	K-NN	B	MI
μ	**4**	3.77	3.77	3.77	**3.78**	**3.71**	3.63	3.62	**3.71**
σ	**2**	1.48	1.53	**1.54**	1.53	1.48	**1.66**	1.46	1.54
τ	**1**	0.95	**0.97**	0.94	0.96	0.94	**1.00**	0.95	0.94
ARI	**0**	0.35	**0.31**	0.35	0.32	0.38	0.36	**0.32**	0.34
Rcor	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**
Data2									
		5%				10%			
	Reference	Std	K-NN	B	MI	Std	K-NN	B	MI
μ	**40**	41.21	42.70	41.21	**40.93**	40.85	41.50	41.02	**40.64**
σ	**20**	18.62	16.95	18.66	**19.35**	18.45	18.48	**19.05**	18.68
τ	**10**	7.12	6.38	7.16	**7.82**	6.67	5.91	**7.13**	6.98
ARI	**0**	0.38	0.58	0.38	**0.27**	0.43	0.52	**0.35**	0.38
Rcor	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**
Data3									
		5%				10%			
	Reference	Std	K-NN	B	MI	Std	K-NN	B	MI
μ	**400**	396.73	**396.74**	396.73	396.72	**399.38**	388.22	396.75	398.98
σ	**200**	183.24	182.88	182.95	**186.07**	179.45	**182.93**	182.75	182.53
τ	**100**	79.14	**79.77**	79.45	78.62	**81.33**	80.68	78.82	80.93
ARI	**0**	0.300	0.295	0.295	**0.291**	0.29	0.307	0.305	**0.28**
Rcor	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**

a -

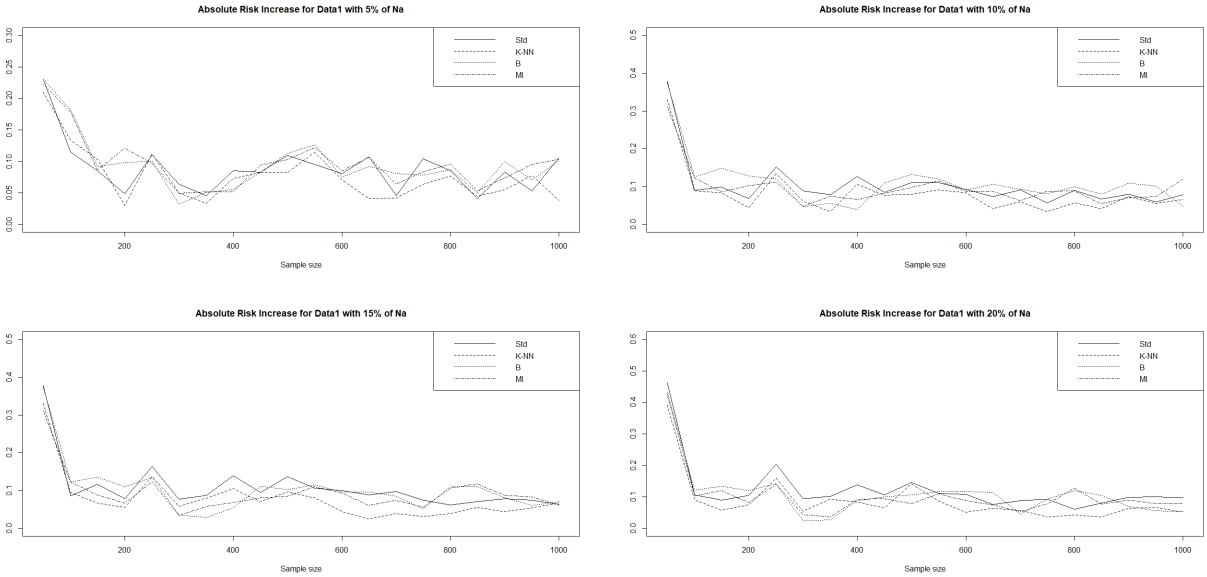
TABLE 3.2 – Estimation des paramètres et valeurs de ARI pour la distribution Ex-Gaussienne à différents niveaux de données Na. **a** - montre les résultats pour 5% et 10% de Na, tandis que **b** - présente les résultats pour 15% et 20% de Na.

Data1									
		15%				20%			
	Reference	Std	K-NN	B	MI	Std	K-NN	B	MI
μ	**4**	3.69	3.61	3.53	**3.73**	3.73	3.67	3.58	**3.81**
σ	**2**	1.44	1.43	**1.62**	1.54	1.44	1.46	**1.62**	1.58
τ	**1**	0.97	**0.99**	0.96	0.95	**0.97**	0.95	0.95	0.93
ARI	**0**	0.37	0.38	**0.33**	**0.33**	0.37	0.38	0.33	**0.31**
Rcor	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**
Data2									
		15%				20%			
	Reference	Std	K-NN	B	MI	Std	K-NN	B	MI
μ	**40**	**40.08**	41.12	41.57	40.09	41.05	40.89	42.23	**40.77**
σ	**20**	17.67	18.48	18.75	**19.36**	18.42	19.02	**20.26**	18.81
τ	**10**	6.13	6.07	7.33	**7.42**	6.29	6.95	**7.46**	7.24
ARI	**0**	0.50	0.49	0.36	**0.29**	0.47	0.37	**0.32**	0.35
Rcor	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**
Data3									
		15%				20%			
	Reference	Std	K-NN	B	MI	Std	K-NN	B	MI
μ	**400**	384.76	394.04	396.72	**401.31**	391.82	**406.3**	407.65	411.04
σ	**200**	165.40	177.86	**188.91**	181.73	178.13	175.26	**183.42**	178.4
τ	**100**	79.64	**81.75**	81.43	81.33	**83.23**	74.12	69.20	72.92
ARI	**0**	0.41	0.30	**0.24**	0.28	**0.29**	0.39	0.40	0.40
Rcor	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**

b -

TABLE 3.2 – *Cont.*

(**a**) Data1



(**b**) Data2

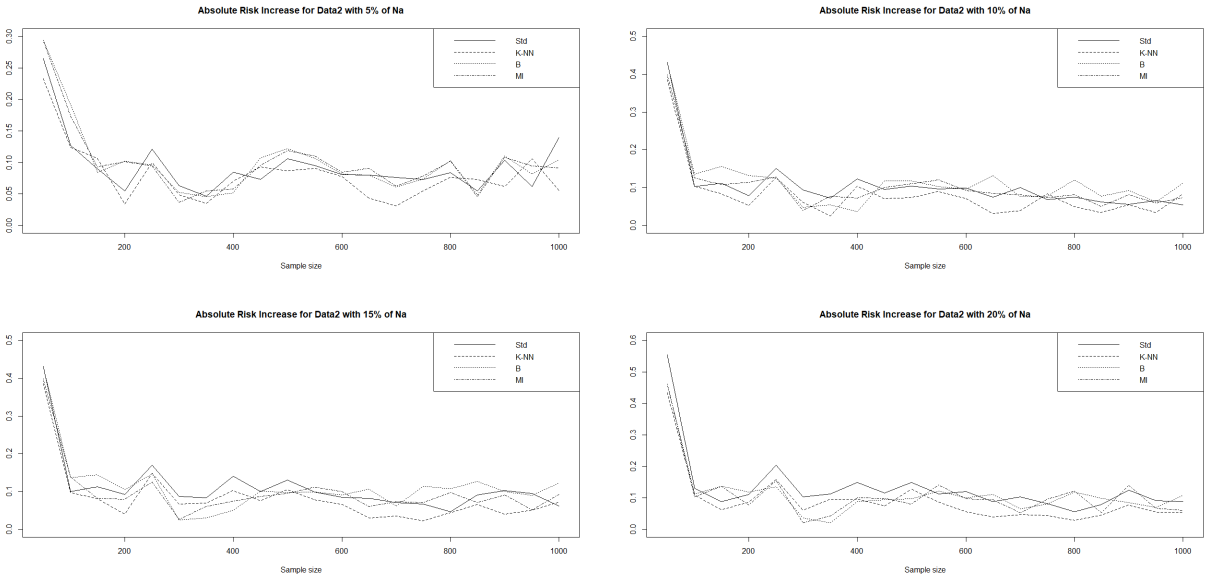


FIGURE 3.3 – Variation des critères `ARI` en fonction de la taille de l'échantillon à travers différentes méthodes d'imputation de `Na` pour (**a**)-Data1, (**b**)-Data2 et (**c**)-Data3. Chaque sous-figure représente quatre pourcentages différents de `Na`.

(**c**) Data3

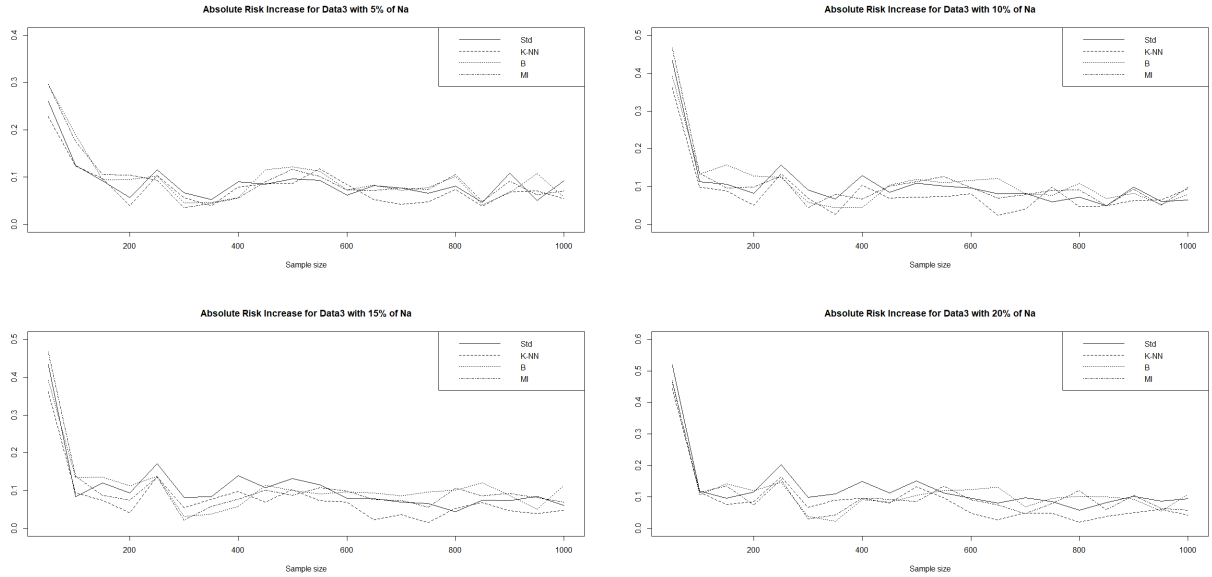


FIGURE 3.3 – *Cont.*

La Figure 3.3 illustre la variation du `ARI` parmi tous les cas, mettant en évidence l'instabilité des méthodes proposées. Cette variation dépend de la taille des paramètres des données et du pourcentage de données manquantes, comme observé dans le Tableau 3.2.

3.3.2 Données réelles

Dans cette section, nous avons utilisé des données psychologiques réelles pour démontrer notre méthodologie. Les participants ont vu des images de différents objets sur un écran. Ils devaient écrire le nom de chaque objet. Les temps de réaction ont été enregistrés depuis le début de la présentation des images jusqu'aux réponses écrites des participants pour analyse. Ces données ont été collectées pour un prétest dans le cadre d'une expérience sur l'influence de la difficulté d'orthographe sur l'accès aux codes orthographiques.

La base de données globale comprend 36 participants, chacun fournissant des données de `RT` pour quatre items : "Clown", "Hamac", "Licorne" et "Mouche." Par conséquent, notre analyse est menée sur quatre ensembles de données spécifiques :

— Clown : 36 participants répondent à l'item "clown", contenant 11% de `Na`.
— Hamac : 36 participants répondent à l'item "hamac" ; contenant 19% de `Na`.
— Licorne : 36 participants répondent à l'item "licorne" ; contenant 6% de `Na`.
— Mouche : 36 participants répondent à l'item "mouche" ; contenant 14% de `Na`.

En appliquant les méthodes proposées aux quatre ensembles de sujets comportant des données manquantes pour résoudre leurs problèmes tout en prédisant la distribution Ex-Gauss et en utilisant la méthode Bootstrap pour l'initialisation, les estimations des paramètres obtenues sont présentées dans le Tableau 3.3.

Les résultats obtenus sur les ensembles de données réelles en utilisant les méthodes proposées pour gérer les problèmes de données manquantes ont été comparés au vecteur de paramètres obtenu en laissant vides les cas des `Na` correspondants (cette méthode est appelée ("Keeping Empty", `KE`). Il est remarquable de noter que c'est ainsi que les problèmes de données manquantes sont généralement résolus dans les études psychologiques.

En se basant sur les estimations des paramètres présentées dans le Tableau 3.3, nous pouvons observer que les valeurs obtenues dans les quatre ensembles de données en appliquant la méthode `KE` sont significativement différentes des autres, surtout pour le paramètre exponentiel τ, où la différence est considérable. Par exemple, dans l'ensemble de données "Clown", la valeur de τ est d'environ 9, alors que pour les autres méthodes proposées, elle varie entre 223 et 235. Du point de vue de la psychologie expérimentale, ce résultat est très important. Des travaux récents suggèrent une association entre les trois paramètres de la distribution Ex-Gauss et des processus cognitifs spécifiques (par exemple, [48]). L'étude de la cognition nécessite donc des estimations des paramètres de la distribution Ex-Gaussienne aussi non biaisées que possible. Par exemple, parmi les quatre items rapportés dans cet exemple, le mot "Clown" est le seul à présenter une difficulté orthographique en français. Plusieurs éléments théoriques suggèrent que cette caractéristique devrait avoir un impact sur le paramètre τ. Les valeurs de τ estimées en utilisant la méthode `KE` sont très similaires pour les quatre items, alors que la valeur de ce paramètre est deux fois plus élevée pour le mot "Clown" que pour les trois autres items en utilisant les autres méthodes. Sans tenir compte de manière adéquate des données manquantes, ce résultat n'apparaît pas, bien qu'il soit attendu d'un point de vue théorique.

			Clown		
	KE	Std	K-NN	B	MI
μ	798.335	810.849	834.554	811.743	815.829
σ	318.353	223.994	262.357	233.948	243.246
τ	9.968	223.848	236.753	228.647	234.450
			Hamac		
	KE	Std	K-NN	B	MI
μ	978.375	995.393	985.607	1013.319	940.786
σ	270.948	269.670	289.902	288.508	332.284
τ	8.023	98.331	99.444	90.315	100.628
			Licorne		
	KE	Std	K-NN	B	MI
μ	882.372	886.278	871.774	887.351	859.716
σ	218.480	267.843	273.047	266.711	272.895
τ	6.565	93.630	105.321	100.729	103.369
			Mouche		
	KE	Std	K-NN	B	MI
μ	948.610	960.045	960.358	953.239	951.055
σ	222.886	264.051	285.814	276.189	270.754
τ	6.9935	90.544	87.514	94.313	93.011

TABLE 3.3 – Estimation de la distribution Ex-Gauss en utilisant diverses méthodes pour gérer les données manquantes à travers quatre ensembles de sujets, chacun distingué par des pourcentages de Na.

3.4 Conclusion et Perspective

Notre chapitre vise à atteindre deux objectifs principaux. Dans la phase initiale de cette étude, nous avons proposé une méthodologie plus efficace pour obtenir des estimations moins biaisées dans la distribution Ex-Gaussienne pour un petit échantillon ($n < 100$) grâce à l'inférence bayésienne. Notre approche reposait sur une méthode d'initialisation des paramètres, où nous avons appliqué la technique de rééchantillonnage basée sur le Bootstrap pour dériver nos paramètres initiaux au lieu d'utiliser une initialisation aléatoire. Les résultats simulés ont montré que notre méthode de Bootstrapping proposait les meilleurs résultats pour estimer les trois paramètres μ, σ et τ.

Le deuxième objectif de ce travail était de résoudre le problème des données manquantes après avoir fixé la méthode d'initialisation. En appliquant le rééchantillon-

nage par Bootstrap pour commencer avec le vecteur de paramètres initiaux, nous avons proposé de résoudre le problème des données manquantes de type `MAR` dans différents scénarios, chacun représentant un pourcentage différent de `Na`. Nous avons proposé quatre méthodes : l'approche standard `Std`, où les valeurs manquantes sont remplacées par la moyenne des données observées ; la méthode `K-NN`, qui remplace les valeurs `Na` par celles des voisins les plus proches ; la méthode `B`, où les données manquantes sont prédites en utilisant les principes bayésiens ; et enfin, la méthode `MI`, utilisant l'Imputation Multiple pour prédire les données manquantes. Les résultats obtenus sur les ensembles de données simulées et réelles appliqués à différents pourcentages de `Na` (5%, 10%, 15%, 20%) ont montré qu'il y a une compétition entre les méthodes, en particulier entre les méthodes bayésienne (`B`) et d'imputation multiple (`MI`). Nos applications numériques ont été réalisées en utilisant le logiciel statistique R.

Compte tenu de ces résultats, nous proposons dans le chapitre qui suit d'étendre notre méthodologie en considérant la procédure d'augmentation de données ([8, 63]) pour obtenir des estimations plus robustes dans le cas de vecteurs de données incomplets.

Code source en R

Nous présentons ici le code R de la méthode Bootstrap, développée dans ce chapitre, pour l'initialisation des paramètres dans le cadre de l'inférence bayésienne.

```
library(gamlss.dist)
library(dlm)

n=50
x <- rexGAUS(n, 400, 200, 100)

mu <- rep(NA, nSamples)
zeta <- rep(NA, nSamples)
lambda <- rep(NA, nSamples)

# Function to estimate mu[1] from bootstrapped samples
mu_estimator <- function(x) {
 return(mean(x) - (sd(x) * 0.5))
}

# Function to estimate sigma[1] from bootstrapped samples
variance_estimator <- function(x) {
 return(var(x) - (sd(x) * 0.5)^2)
}

# Function to estimate lambda[1] from bootstrapped samples
lambda_estimator <- function(x) {
 return(1/(sd(x) * 0.5))
}

# Number of bootstrap samples
num_bootss <- 10000

# Perform bootstrapping to estimate mu[1], sigma[1], and lambda[1]
mu_boots <- replicate(num_boots, mu_estimator(sample(x, replace = TRUE)))
variance_boots <- replicate(num_boots, variance_estimator(sample(x, replace = TRUE)))
lambda_boots <- replicate(num_boots, lambda_estimator(sample(x, replace = TRUE)))

mu[1] <- mean(mu_boots)
zeta[1] <- mean(variance_boots)
lambda[1] <- mean(lambda_boots)
```

Chapitre 4

Gestion des données manquantes pour la distribution Ex-Gaussienne en utilisant la méthode d'augmentation de données

Ce chapitre présente une continuation du chapitre précédent, portant sur la gestion des données manquantes pour un vecteur de réponse de distribution Ex-Gaussienne. Nous y introduisons la méthode d'augmentation des données pour résoudre ce problème. Ce chapitre, réalisé en collaboration avec mes directeurs de thèse, M. Yousri Slaoui et M. Cyril Perret, correspond à un article soumis au *Sankhya, The Indian Journal of Statistics*, intitulé : *Handling Missing Data for the Ex-Gaussian Distribution Using the Data Augmentation Method.*

4.1 Introduction

En nous appuyant sur nos résultats précédents, qui ont mis en évidence la compétition entre diverses méthodes de gestion des données manquantes, notamment entre les approches bayésiennes et `MI`, nous introduisons l'algorithme d'augmentation des données (Data Augmentation, `DA`) ([8]) comme une solution alternative pour traiter les données manquantes (`Na`). Cette approche, largement répandue dans l'apprentissage automatique et la science des données, améliore les ensembles de données en générant des échantillons supplémentaires par des transformations variées.

L'augmentation des données permet d'enrichir artificiellement les ensembles d'entraînement, en introduisant de la variabilité et du réalisme, ce qui aide les modèles d'apprentissage automatique à apprendre des schémas plus robustes et généralisés. Elle aide à prévenir le surapprentissage, réduit le biais, et améliore les performances sur des données nouvelles ou réelles.

Cette technique est particulièrement efficace dans des domaines comme la vision par ordinateur, où des transformations telles que la rotation et la translation simulent des variations du monde réel. Dans le traitement du langage naturel, des méthodes comme le remplacement de synonymes et la paraphrase permettent d'améliorer la robustesse des modèles ([112]). De même, dans l'analyse des séries temporelles, l'augmentation des données contribue à améliorer la capture des dépendances temporelles et la précision des modèles ([4, 65]).

Dans ce chapitre, nous avons utilisé l'algorithme DA pour gérer les données manquantes de type MAR. Nous avons exploré trois variantes de cet algorithme ([62, 75]) : l'algorithme standard d'augmentation des données (DA), l'algorithme d'augmentation des données avec paramètre étendu (PX-DA) et l'algorithme d'augmentation des données avec paramètre étendu de Haar (Haar PX-DA). Ces variantes sont comparées aux autres méthodes que nous avons étudiées précédemment.

Enfin, pour valider nos hypothèses, nous avons appliqué ces méthodes à des données psychologiques simulées et réelles. L'efficacité de chaque approche a été évaluée en comparant les vecteurs de paramètres obtenus avec ceux dérivés des méthodes précédentes, en utilisant la mesure ARI.

Ce chapitre est organisé comme suit : Dans la Section 4.2, nous donnons un aperçu de la méthode Markov Chain Monte Carlo (MCMC), qui sert de base au sein du cadre DA. Nous approfondissons ensuite les concepts fondamentaux et les bases théoriques de nos algorithmes proposés, en présentant les trois versions et leur application pour prédire les données manquantes afin d'obtenir des estimations non biaisées. Les résultats numériques sont présentés dans la Section 4.3, où nous comparons les résultats obtenus à partir des ensembles de données simulées (sous-section 4.3) et des données psychologiques réelles (sous-section 4.3). La Section 4.3.1 propose une conclusion de cette étude et ouvre une perspective pour nos futurs projets.

4.2 Méthodologie

Soit $Y = \{Y_1, \ldots, Y_n\}$ notre vecteur de réponses complet contenant des valeurs manquantes de type MAR. Y peut être écrit comme : $Y = \{Y_o, Y_m\}$, avec Y_o et Y_m représentant respectivement les données observées et manquantes.
La fonction de densité du vecteur complet Y peut être représentée comme le mélange des composantes observées et manquantes :

$$f(y) = \pi \times f_o(y) + (1 - \pi) \times f_m(y), \tag{4.1}$$

avec π représentant la proportion des données observées, $f_o(y)$ et $f_m(y)$ étant respectivement les fonctions de densité des composantes observées et manquantes.
Étant donné que la fonction de densité est divisée en 2 composantes dans l'équation (4.1), le vecteur complet des paramètres θ est également divisé en θ_o et θ_m, représentant respectivement les paramètres associés aux données observées et manquantes.

Pour traiter le problème de la présence de données Na, nous avons proposé l'algorithme DA, défini comme une technique statistique visant à augmenter artificiellement l'ensemble de formation en créant des copies modifiées d'un jeu de données en utilisant les données existantes. Cela inclut des modifications mineures du jeu de données ou l'utilisation de l'apprentissage profond pour générer de nouveaux points de données. Il s'agit d'une méthode populaire de Markov Chain Monte Carlo (MCMC) comme présenté dans [63]. Par conséquent, dans la section suivante, nous présentons un aperçu simple de la méthode MCMC connues.

4.2.1 Méthode de MCMC

Le méthode Monte Carlo (MC) est une technique de calcul conçue pour approximer les solutions à des problèmes mathématiques et statistiques complexes grâce à l'échantillonnage aléatoire. Son nom est inspiré par le célèbre Casino de Monte-Carlo à Monaco, qui symbolise l'élément de chance central à cette technique.
Fondamentalement, la méthode MC consiste à générer des échantillons aléatoires à partir d'une distribution de probabilité donnée pour estimer des résultats numériques. Ces échantillons sont ensuite utilisés pour calculer les quantités souhaitées, telles que la valeur d'une intégrale, la solution à une équation ou le comportement d'un système ([73]).
Cette méthode est particulièrement avantageuse pour traiter des problèmes qui sont analytiquement ingérables ou computationnellement prohibitifs à résoudre par des approches déterministes. En échantillonnant à partir de la distribution de probabilité du problème et en moyennant les résultats, les simulations MC peuvent fournir des approximations précises avec un niveau de confiance spécifié.

Supposons que $f_X : \mathbb{R}^p \to [0, \infty)$ est une fonction de densité de probabilité, et que $g : \mathbb{R}^p \to \mathbb{R}$ est une fonction d'intérêt. Estimer directement cette fonction en utilisant l'équation suivante est impossible car l'intégrale ne peut pas être calculée analytiquement :

$$\hat{g}(x) = \mathbb{E}[g(x)] = \int g(x)f(x)dx, \tag{4.2}$$

Basé sur la méthode MC, nous pouvons estimer l'espérance dans l'équation (4.2) en simulant des variables aléatoires indépendantes et identiquement distribuées (iid) $X_1, X_2, \ldots, X_n \sim f(.)$.
L'estimation est alors donnée par ([84]) :

$$\hat{g}(x) = \frac{1}{n}\sum_{i=1}^{n} g(X_i).$$

Une chaîne de Markov, notée par $X_n : n \in \mathbb{N}$ ([22]), où X_n représente l'état au temps n, est un modèle stochastique nommé d'après le mathématicien russe Andrey Markov. Elle satisfait la propriété de Markov, qui stipule que pour tout temps n et états $i_0, i_1, \ldots, i_n$, la probabilité de transition vers l'état i_{n+1} dépend uniquement de l'état actuel i_n et est indépendante des états précédents.
Dans une chaîne de Markov, chaque paire d'états i et j a une probabilité de transition de l'état i à l'état j en un seul pas. Ces probabilités sont organisées dans une matrice de transition P, où P_{ij} représente l'élément à la $i^{\text{ème}}$ ligne et à la $j^{\text{ème}}$ colonne. Si la chaîne de Markov est homogène, les probabilités de transition restent constantes dans le temps, ce qui signifie que la matrice de transition P ne change pas. De plus, une distribution de probabilité initiale sur les états, notée π_0, est nécessaire dans une chaîne de Markov, représentant les probabilités de commencer dans chaque état.

Ainsi, la solution de la méthode `MCMC` est de construire une chaîne de Markov X_n sur X qui possède une distribution stationnaire $\pi(.)$; c'est-à-dire que $\pi \times P = \pi$. Cette méthode combine des éléments à la fois de la méthode de Monte Carlo et des chaînes de Markov.
Ces méthodes englobent une variété de techniques puissantes pour l'échantillonnage à partir de distributions de probabilité complexes, offrant des solutions polyvalentes à un large éventail de défis computationnels. De telles approches, mises en avant par des travaux fondamentaux en statistiques computationnelles et en physique, ont révolutionné le domaine en permettant aux chercheurs d'explorer efficacement des espaces de paramètres de haute dimension et de faire des inférences probabilistes dans divers domaines.
Des exemples notables de ces techniques incluent l'algorithme de Metropolis-Hastings ([91]), l'échantillonnage de Gibbs ([54]), l'échantillonnage Hamiltonien ([42, 93]), et d'autres, chacun offrant des avantages et des applications uniques.

Dans la section suivante, nous présentons notre algorithme proposé pour traiter le problème des données manquantes de type "Missing at Random".

4.2.2 Algorithme `DA` proposé

L'algorithme `DA` ([8]) opère sur une fonction de distribution conjointe donnée par $f : \mathbb{R}^p \times \mathbb{R}^q \to [0, \infty)$. Dans les cas où la simulation directe à partir de la densité marginale f_Y est infaisable, rendant les méthodes `MC` inapplicables, la méthode `DA` est employée. Cela implique de simuler à partir des densités conditionnelles $f_{Y|Z}(y|z)$ et $f_{Z|Y}(z|y)$, comme décrit dans la section suivante.

Algorithme `DA` Standard

Tout d'abord, nous commençons le processus en construisant la chaîne de Markov, en supposant que toutes les chaînes de Markov discutées ici sont Harris ergodiques. Une chaîne de Markov X est dite Harris ergodique si elle satisfait les trois conditions suivantes :

- **Irréductibilité** : Pour chaque paire d'états i et j, il existe un entier n tel que la probabilité de transition de l'état i à l'état j après n étapes, notée $P^n(i, j)$, est positive.
- **Apériodicité** : La chaîne est apériodique si, pour tous les états i, le plus grand commun diviseur (PGCD) de tous les nombres possibles d'étapes n nécessaires pour revenir à l'état i à partir de cet état est égal à 1. Formellement, cela signifie que $d(i) = \text{PGCD}\{n \geq 1 : P_{ii}^n > 0\} = 1$.
- **Récurrence de Harris** : Une chaîne de Markov est dite de Harris récurrente si elle satisfait à la fois les conditions suivantes :
  - **Récurrence positive** : Le temps moyen de retour à l'état i en partant de l'état i est fini, c'est-à-dire que $\mathbb{E}_i[X_i] < \infty$.

— **Récurrence nulle** : La chaîne revient à l'état i avec une probabilité de 1, ce qui signifie que $\lim_{n\to\infty} P^n(i,i) = 1$.

Soit $Y = \{y \in \mathbb{R}^p; f_Y(y) > 0\}$ et $Z = \{z \in \mathbb{R}^q; f_Z(z) > 0\}$ deux chaînes construites, nous supposons :
— $f(y,z) = 0$ chaque fois que $(y,z) \notin Y \times Z$,
— $f_Z(z) = \int_{\mathbb{R}^p} f(y,z)dy$.

Maintenant, nous définissons une fonction $k : Y \times Y \to [0,\infty)$ comme suit :

$$k(y^{'}|y) = \int_Z f_{Y|Z}(y^{'}|z) f_{Z|Y}(z|y)dz. \tag{4.3}$$

Puisque l'intégrale dans l'équation (4.3) est un produit de densités conditionnelles, nous pouvons dire que k est toujours positive.
De plus,

$$\begin{aligned}
\int_Y k(y^{'}|y)dy^{'} &= \int_Y \left[\int_Z f_{Y|Z}(y^{'}|z) f_{Z|Y}(z|y)dz \right] dy^{'} \\
&= \int_Z f_{Z|Y}(z|y) \left[\int_Y f_{Y|Z}(y^{'}|z)dy^{'}dz \right] \\
&= \int_Z f_{Z|Y}(z|y)dz \\
&= 1.
\end{aligned}$$

Ainsi, pour chaque $y \in Y$, $k(y^{'}|y)$ st non négatif et son intégrale est égale à 1. Par conséquent, la fonction k iest une densité de transition de Markov (`Mtd`) qui définit une chaîne de Markov.

Nous pouvons prouver que la chaîne de Markov Y est réversible par rapport à f_Y à partir de ces qualités :

$$\begin{aligned}
k(y^{'}|y) f_Y(y) = f_Y(y) \int_Z f_{Y|Z}(y^{'}|z) f_{Z|Y}(z|y)dz = \int_Z \frac{f(y^{'},z) f(y,z)}{f_Z(z)}dz \\
k(y|y^{'}) f_Y(y^{'}) = f_Y(y^{'}) \int_Z f_{Y|Z}(y|z) f_{Z|Y}(z|y^{'})dz = \int_Z \frac{f(y,z) f(y^{'},z)}{f_Z(z)}dz.
\end{aligned} \tag{4.4}$$

Ainsi, nous pouvons dire à partir de l'équation (4.4) que : $k(y^{'}|y) f_Y(y) = k(y|y^{'}) f_Y(y^{'})$ $\implies$ Y est réversible par rapport à f_Y.

Dans notre étude, nous notons que le vecteur de réponse Y est divisé en deux sous-ensembles : la partie observée Y_o et la partie manquante Y_m. L'objectif est de prédire le composant manquant Y_m en utilisant la technique d'augmentation de données. Lors de l'application de cette approche pour imputer les valeurs manquantes de type `MAR`, nous pouvons écrire une itération de notre méthodologie proposée, comme présenté dans l'Algorithme 5.

Cet algorithme DA peut être étendu à une autre version où, plutôt que de simuler directement à partir de la densité conditionnelle $f_{Y|Z}(.|z)$, nous simulons à partir de $f_{Y|Z}(.|z^{'})$, où $z^{'}$ est un vecteur dérivé d'une fonction de transformation. Cette nouvelle version est présentée dans la section suivante.

Algorithm 5 DA Algorithm : Y is the initial incomplete data set ; Y_o is the observed vector ; Y_m is the missed vector ; n is the sample size of Y_m.

Input: Y.

1: $Z \sim f_{Z|Y}(.|y_o)$ # call the observed vector z
2: **for** $i = 1, \ldots, n$ **do**
3: $\quad Y_m[i] \sim f_{Y|Z}(.|z)$
4: **end for**

output: Y.

Algorithme PX-DA

Maintenant, nous considérons un groupe $G \subset \mathbb{R}^d$ et nous supposons que nous avons une classe de fonctions $t_g : Z \to Z$ indexée par $g \in G$. Soit $J_g(j)$ le Jacobien de la transformation $j = t_g^{-1}(z)$, donc, par exemple, dans le cas univarié, $J_g(j) = \frac{\partial}{\partial j} t_g(j)$.

Notons que :

$$\int_Z f(y, t_g(z))|J_g(z)|dz = \int_Z f(y, j)dj = f_Y(y).$$

Nous supposons que $w : G \to [0, \infty)$ est une fonction de distribution de probabilité, et nous pouvons définir $f^{(w)} : Y \times Z \times G \to [0, \infty)$ comme suit :

$$f^{(w)}(y, z, g) = f(y, t_g(z))|J_g(z)|w(g),$$

$$f^{(w)}(y, z) = \int_G f^{(w)}(y, z, g)dg.$$

L'idée d'introduire le paramètre supplémentaire g pour former un nouvel algorithme DA a été développée indépendamment par [90], qui l'ont appelé "Marginal Augmentation", et par [81], qui l'ont nommé "Parameter Expanded-Data Augmentation (PX-DA)". Cette nouvelle version de l'algorithme DA est basée sur $f^{(w)}(y, z)$.

En exploitant la relation entre $f^{(w)}(y, z)$ et $f^{(w)}(y, z, g)$, il est possible de développer des méthodes indirectes pour échantillonner à partir de $f^{(w)}_{Y|Z}$ et $f^{(w)}_{Z|Y}$ en utilisant uniquement des échantillons de $f_{Y|Z}$, $f_{Z|Y}$ et $w(g)$, sans avoir besoin d'une autre densité.

Nous proposons alors la procédure suivante :

— Tout d'abord, nous commençons par $f^{(w)}_{Z|Y}(z|y)$:

$$
\begin{aligned}
f^{(w)}_{Z|Y}(z|y) &= \frac{\int_G f^{(w)}(y,z,g)\,dg}{f_Y(y)} \\
&= \int_G \frac{f(y,t_g(z))}{f_Y(y)}|J_g(z)|w(g)\,dg \\
&= \int_G f_{Z|Y}(t_g(z)|y)|J_g(z)|w(g)\,dg. \qquad (4.5)
\end{aligned}
$$

— Ensuite, nous échantillonnons $G \sim f^{(w)}_{G|Z}(.|z)$:

$$
\begin{aligned}
f^{(w)}_{G|Z}(g|z) &= \frac{\int_Y f^{(w)}(y,z,g)\,dy}{\int_G \int_Y f^{(w)}(y,z,g)\,dy\,dg} \\
&\propto \int_Y f^{(w)}(y,z,g)\,dy \\
&= f_Z(t_g(z))|J_g(z)|w(g). \qquad (4.6)
\end{aligned}
$$

— Enfin, nous échantillonnons $Y \sim f^{(w)}_{Y|Z,G}(.|z,g)$:

$$
\begin{aligned}
f^{(w)}_{Y|Z,G}(y|z,g) &= \frac{f^{(w)}(y,z,g)}{\int_Y f^{(w)}(y,z,g)\,dy} \\
&= \frac{f(y,t_g(z))|J_g(z)|w(g)}{f_Z(t_g(z))|J_g(z)|w(g)} \\
&= f_{Y|Z}(y|t_g(z)). \qquad (4.7)
\end{aligned}
$$

Algorithm 6 PX-DA Algorithm : Y is the initial incomplete data set ; Y_o is the observed vector ; Y_m is the missed vector ; t is a transformation function with $t_a(b) = a + b$; w is the working prior ; n is the sample size of Y_m.

Input: Y ; w ; $t_a(b)$.

1: $Z \sim f_{Z|Y}(.|y_o)$ # call the result z
2: $G \sim w(.)$ # call the result g,
   $G' \sim f^{(w)}_{G|Z}(.|t_g^{-1}(z))$ # call the result g',
   $z' = t_{g'}(t_g^{-1}(z))$
3: **for** $i = 1, \ldots, n$ **do**
4: $\quad Y_m[i] \sim f_{Y|Z}(.|z')$
5: **end for**

output: Y.

Par conséquent, les équations (4.5), (4.6) et (5.2.3) conduisent à une itération de la version de l'algorithme `DA` appelée `PX-DA`, appliquée dans notre étude comme le montre l'Algorithme 6.

Il existe en fait des arguments théoriques ainsi que des preuves empiriques suggérant que l'algorithme `PX-DA` performe mieux lorsque la fonction de distribution de probabilité $w(.)$ devient plus "diffuse". D'autre part, il est clair que ce développement échoue si w est inappropriée. Cependant, [81] ont montré que, s'il existe une certaine structure de groupe dans le problème, il est possible de construire un algorithme `PX-DA` valide en utilisant une densité de Haar inappropriée à la place de w. Cette version est appelée algorithme "Haar Parameter Expanded Data Augmentation (`Haar PX-DA`)" et est présentée dans la section suivante.

Algorithme `Haar PX-DA`

Soit G un groupe topologique, c'est-à-dire un groupe pour lequel les fonctions $(g_1, g_2) \mapsto g_1 g_2$ et $g \mapsto g^{-1}$ sont continues. Soit e l'élément neutre du groupe et supposons que $t_e(z) = z \quad \forall z \in Z$ et que $t_{g_1 g_2}(z) = t_{g_1}(t_{g_2}(z)) \quad \forall g_1, g_2 \in G$ et $\forall z \in Z$.

Une fonction $\mathcal{X} : G \to \mathbb{R}_+$ est appelée multiplicateur si elle satisfait les deux conditions suivantes :

— $\mathcal{X}$ est continue,

— $\mathcal{X}(g_1 g_2) = \mathcal{X}(g_1)\mathcal{X}(g_2) \quad \forall g_1, g_2 \in G$.

Pour tout $g \in G$ et toute fonction intégrable $h : Z \to \mathbb{R}$, nous avons :

$$\mathcal{X}(g) \int_Z h(t_g(z))dz = \int_Z h(z)dz.$$

Supposons que le groupe G possède une mesure de Haar à gauche [1] de la forme $\mathbf{v}_l(g)dg$, où dg désigne la mesure de Lebesgue [2] sur G. La mesure de Haar à gauche satisfait l'équation suivante :

$$\int_G h(\tilde{g}z)\mathbf{v}_l(g)\, dg = \int_G h(g)\mathbf{v}_l(g)\, dg,$$

pour tout $\tilde{g} \in G$ et pour toutes les fonctions intégrables $h : G \to \mathbb{R}$; dans la plupart des applications, cette mesure sera inappropriée, c'est-à-dire que $\int_G \mathbf{v}_l(g)\, dg = \infty$.

Enfin, supposons que :

$$q(z) := \int_G f_Z(t_g(z))\mathcal{X}(g)\mathbf{v}_l(g)\, dg$$

1. Une mesure de Haar à gauche sur un groupe est un type particulier de mesure qui capture le "volume" ou la "taille" des sous-ensembles du groupe de manière compatible avec la structure du groupe. Plus précisément, c'est une mesure qui est invariante par les translations à gauche par des éléments du groupe. Cela signifie que si vous traduisez un ensemble en multipliant chaque élément de cet ensemble par un élément fixe du groupe, la mesure de l'ensemble ne change pas.

2. La mesure de Lebesgue, notée dg, est une mesure définie sur certains espaces, typiquement des espaces euclidiens comme $\mathbb{R}^n$. C'est un moyen d'assigner une notion de "taille" ou de "volume" aux sous-ensembles de ces espaces qui généralise le concept de longueur, d'aire ou de volume. Lorsqu'elle est étendue à certains groupes comme $\mathbb{R}^n$ équipés de l'addition, la mesure de Lebesgue capture la notion intuitive de "taille" de manière compatible avec les translations.

soit strictement positif pour tout $z \in Z$ et fini pour presque tout $z \in Z$. Nous énonçons maintenant une itération de l'algorithme `Haar PX-DA` ([81]) appliqué dans notre cas pour prédire les valeurs des données manquantes (Algorithme 7).

Algorithm 7 Haar PX-DA Algorithm : Y is the initial incomplete data set ; Y_o is the observed vector ; Y_m is the missed vector ; t is a transformation function with $t_a(b) = a + b$; $\mathcal{X}(.)$ is a multiplier ; $\mathbf{v}_l(.)$ is a left Haar measure ; n is the sample size of Y_m.

Input: Y ; $t_a(b)$; $\mathcal{X}(.)$; $\mathbf{v}_l(.)$.

1: $Z \sim f_{Z|Y}(.|y_o)$ # call the result z
2: $G \sim f_Z(t_g(z))\mathcal{X}(g)\mathbf{v}_l(g)$ # call the result g,
   $z^{'} = t_g(z)$
3: **for** $i = 1, \ldots, n$ **do**
4: $\quad Y_m[i] \sim f_{Y|Z}(.|z)$
5: **end for**

output: Y.

Dans la section suivante, nous présentons certaines applications numériques utilisant des données psychologiques simulées et réelles.

4.3 Application numérique

Dans cette section, nous nous intéressons à la validation de la méthode proposée pour imputer les données manquantes. Nous explorons notre approche en utilisant à la fois des données simulées et réelles.

Données simulées

Nous avons testé les trois versions de notre algorithme proposé sur les mêmes ensembles de données simulées (Voir Chapitre 3), à savoir Data1, Data2 et Data3, et nous avons comparé les résultats obtenus avec ceux des méthodes précédemment proposées.

Le Tableau 4.1 présente les méthodes d'imputation `Na` appliquées aux ensembles de données simulés en présence de 5%, 10%, 15% ou 20% de valeurs `MAR`. La comparaison est basée aussi bien sur les valeurs `ARI`, les valeurs les plus faibles étant mises en évidence en gras pour indiquer la meilleure méthode d'imputation.

				Data1				
	θ	Std	K-NN	B	MI	DA	PX	Haar PX-DA
μ	**4**	3.77	3.77	3.77	3.78	3.75	3.76	3.86
σ	**2**	1.48	1.53	1.54	1.53	2.31	2.30	2.30
τ	**1**	0.95	0.97	0.94	0.96	0.98	0.99	0.97
ARI	**0**	0.35	0.31	0.33	0.32	0.23	**0.21**	**0.21**
				Data2				
	θ	Std	K-NN	B	MI	DA	PX	Haar PX-DA
μ	**40**	41.21	42.70	41.21	40.93	36.97	36.94	36.84
σ	**20**	18.62	16.95	18.66	19.35	18.28	18.30	18.18
τ	**10**	7.12	6.38	7.16	7.82	9.70	10.33	10.06
ARI	**0**	0.38	0.58	0.38	0.27	0.19	0.19	**0.17**
				Data3				
	θ	Std	K-NN	B	MI	DA	PX	Haar PX-DA
μ	**400**	396.73	396.74	396.73	396.72	380.71	377.31	379.32
σ	**200**	183.24	182.88	182.95	186.07	184.75	184.03	184.07
τ	**100**	79.14	79.77	79.45	78.62	79.01	80.60	80.00
ARI	**0**	0.30	**0.29**	**0.29**	**0.29**	0.33	0.33	0.33

a -

TABLE 4.1 – Estimations des paramètres et valeurs ARI pour la distribution Ex-Gauss à différents niveaux de Na. **a** - montre les résultats pour 5%, **b** - pour 10%, **c** - pour 15% et **d** - pour 20%.

Data1								
	θ	Std	K-NN	B	MI	DA	PX	Haar PX-DA
μ	**4**	3.71	3.63	3.62	3.74	3.84	3.82	3.99
σ	**2**	1.48	1.66	1.46	1.54	2.35	2.30	2.32
τ	**1**	0.94	1.00	0.95	0.94	0.98	0.92	0.95
ARI	**0**	0.38	0.36	0.32	0.34	0.23	0.26	**0.21**

Data2								
	θ	Std	K-NN	B	MI	DA	PX	Haar PX-DA
μ	**40**	40.85	41.50	41.02	40.64	36.40	36.94	36.06
σ	**20**	18.45	18.48	19.05	18.68	17.75	17.78	18.04
τ	**10**	6.67	5.91	7.13	6.98	9.38	9.46	10.11
ARI	**0**	0.43	0.52	0.35	0.38	0.26	0.24	**0.20**

Data3								
	θ	Std	K-NN	B	MI	DA	PX	Haar PX-DA
μ	**400**	399.38	388.2	396.75	398.98	381.55	372.67	379.35
σ	**200**	179.45	182.93	182.75	182.53	185.85	187.83	184.14
τ	**100**	81.33	80.68	78.82	80.93	86.68	77.78	79.87
ARI	**0**	0.29	0.30	0.30	0.28	**0.25**	0.35	0.33

b -

Data1								
	θ	Std	K-NN	B	MI	DA	PX	Haar PX-DA
μ	**4**	3.69	3.61	3.53	3.73	3.70	3.50	3.76
σ	**2**	1.44	1.43	1.62	1.54	2.32	2.46	2.28
τ	**1**	0.97	0.99	0.96	0.95	1.04	0.96	0.99
ARI	**0**	0.37	0.38	0.33	0.33	0.27	0.39	**0.20**

Data2								
	θ	Std	K-NN	B	MI	DA	PX	Haar PX-DA
μ	**40**	40.08	41.12	41.57	40.09	35.76	37.76	38.12
σ	**20**	17.67	18.48	18.75	19.36	17.88	17.71	18.36
τ	**10**	6.13	6.07	7.33	7.42	9.53	9.25	9.80
ARI	**0**	0.50	0.49	0.36	0.29	0.25	0.24	**0.14**

Data3								
	θ	Std	K-NN	B	MI	DA	PX	Haar PX-DA
μ	**400**	384.76	394.04	396.72	401.31	373.81	390.06	364.54
σ	**200**	165.40	177.86	188.91	181.73	181.10	187.17	177.57
τ	**100**	79.64	81.75	81.43	81.33	79.51	81.65	72.49
ARI	**0**	0.41	0.30	**0.24**	0.28	0.36	0.27	0.47

c -

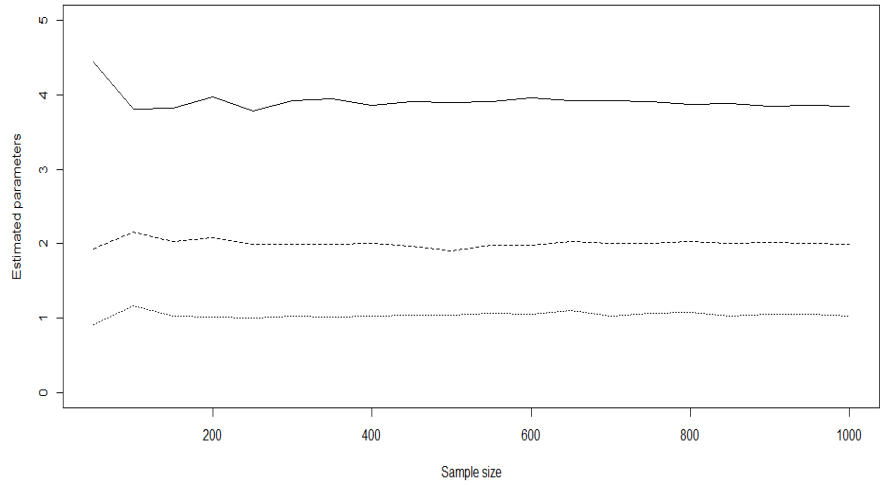
TABLE 4.1 – *Cont.*

				Data1				
	θ	Std	K-NN	B	MI	DA	PX	Haar PX-DA
μ	**4**	3.73	3.67	3.58	3.81	3.68	3.76	3.91
σ	**2**	1.44	1.46	1.62	1.58	2.32	2.33	2.38
τ	**1**	0.97	0.95	0.95	0.93	1.04	1.02	0.96
ARI	**0**	0.37	0.38	0.33	0.31	0.29	0.25	**0.24**
				Data2				
	θ	Std	K-NN	B	MI	DA	PX	Haar PX-DA
μ	**40**	41.05	40.89	42.23	40.77	38.60	34.84	34.79
σ	**20**	18.42	19.02	20.26	18.81	17.68	18.26	18.15
τ	**10**	6.29	6.95	7.469	7.24	1.04	1.02	0.96
ARI	**0**	0.47	0.37	0.32	0.35	0.25	**0.22**	0.28
				Data3				
	θ	Std	K-NN	B	MI	DA	PX	Haar PX-DA
μ	**400**	391.82	406.3	407.65	411.04	398.44	379.80	381.46
σ	**200**	178.13	175.26	183.42	178.4	186.33	187.92	180.43
τ	**100**	83.23	74.12	69.20	72.92	74.43	84.82	86.99
ARI	**0**	0.29	0.39	0.40	0.40	0.32	**0.26**	0.27

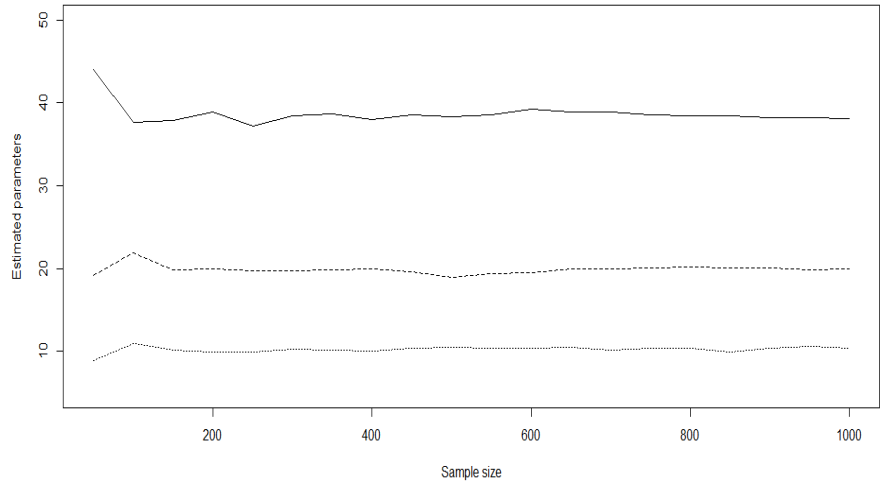
d -

TABLE 4.1 – *Cont.*

A partir du Tableau 4.1, nous pouvons voir qu'Algorithme 7 donne de meilleurs résultats pour résoudre le problème des données manquantes dans presque tous les cas expérimentés, à l'exception du troisième ensemble de données simulées (Data3) ou pour le pourcentage le plus élevé (20%). Dans ces cas, par exemple, avec 5% (ou 10%) de `Na` dans Data3, les meilleures méthodes étaient `K-NN`, `B` ou `MI`, avec la valeur `ARI` la plus basse étant de 0,29. Dans le cas de 10% de `Na`, l'algorithme `DA` proposé avait la valeur `ARI` la plus basse de 0,25. Pour 15% de `Na` dans Data3, la méthode `B` a la meilleure valeur de `ARI`, égale à 0,24. En revanche, pour 20% de valeurs manquantes dans Data2 et Data3, la meilleure valeur de `ARI` est obtenue avec l'algorithme `PX` proposé.
De plus, la Figure 4.1 montre la convergence de ces paramètres dans la méthode `Haar PX-DA` vers les valeurs vraies pour Data1 dans la Figure 4.1 (a) ($\mu = 4$, $\sigma = 2$, $\tau = 1$), Data2 dans la Figure 4.1 (b) ($\mu = 40$, $\sigma = 20$, $\tau = 10$) et Data3 dans la Figure 4.1 (c) ($\mu = 400$, $\sigma = 200$, $\tau = 100$).

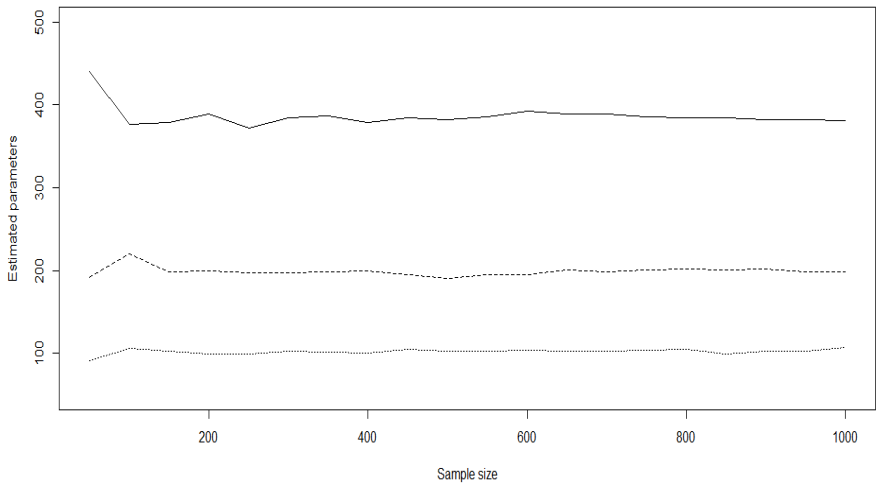


(**a**)



(**b**)

FIGURE 4.1 – Valeurs des paramètres estimés obtenues pour différentes tailles d'échantillon dans les trois simulations utilisant la méthode `Haar PX-DA` (Algorithme 7) ; (**a**)-Data1, (**b**)-Data2, (**c**)-Data3.



(**c**)

FIGURE 4.1 – *Cont.*

Données réelles

Dans cette section, nous reprenons l'application sur les mêmes données réelles, comprenant les 4 sujets (Clown, Hamac, Licorne et Mouche) et les 36 participants, utilisées dans le chapitre précédent (Chapitre 3). Chaque vecteur de réponse est constitué des noms des sujets écrits par ces participants ; nous obtenons finalement 4 vecteurs de réponse contenant différents pourcentages de `Na`.

En appliquant les trois versions proposées de l'algorithme `DA` et la méthode bayésienne sur les quatre ensembles de données avec des données manquantes pour résoudre leurs problèmes tout en prédisant la distribution Ex-Gauss, les estimations des paramètres obtenues sont présentées dans le Tableau 4.2.

Les résultats sur les ensembles de données réelles en utilisant les méthodes proposées ont été comparés aussi avec la méthode `KE`.

D'après le Tableau 4.2, nous pouvons constater la différence dans les valeurs prédites des paramètres entre la méthode `KE` et les autres, en particulier pour le paramètre τ, où cette différence est substantielle dans tous les ensembles de données. Notamment, dans "Clown", la différence de prédiction pour τ varie entre 219 et 225.

4.3.1 Conclusion et Perspective

L'objectif de ce chapitre était de traiter le problème des données `Na` classées comme `MAR` sous divers scénarios, chacun représentant un pourcentage différent de don-

Licorne					
	KE	B	DA	PX	Haar PX-DA
μ	882.37	887.35	895.18	871.91	892.22
σ	218.48	266.71	271.89	279.04	271.15
τ	6.56	100.72	95.95	92.31	97.66
Clown					
	KE	B	DA	PX	Haar PX-DA
μ	798.33	811.74	789.21	806.24	773.56
σ	318.35	233.94	238.01	237.93	240.63
τ	9.96	228.64	234.90	229.27	232.50
Mouche					
	KE	B	DA	PX	Haar PX
μ	948.61	953.23	969.37	947.97	939.98
σ	222.88	276.18	273.10	283.67	270.37
τ	6.99	94.31	82.03	121.23	101.42
Hamac					
	KE	B	DA	PX	Haar PX-DA
μ	978.375	1013.319	1008.51037	915.1171	980.1016
σ	270.948	288.508	300.08854	315.5851	298.3052
τ	8.023	90.315	92.41165	111.5001	100.9760

TABLE 4.2 – Estimation de la distribution Ex-Gauss en utilisant diverses méthodes pour gérer les données manquantes dans quatre ensembles de données, chacun étant caractérisé par des pourcentages variables de Na.

nées non observées, afin d'obtenir une estimation non biaisée pour de petites tailles d'échantillons dans une distribution Ex-Gauss. Initialement, nous avons exploré plusieurs méthodes dans des études antérieures, y compris les méthodes std, K-NN, B et MI.

Les résultats obtenus en utilisant ces méthodes ont montré une grande variabilité en fonction de la taille des paramètres (μ, ζ et τ) et du pourcentage de Na. Ces variations indiquent une instabilité, rendant ces méthodes inadaptées pour une application générale.

Pour surmonter ces limitations, nous avons introduit une approche innovante basée sur l'algorithme DA. Cette méthode vise à augmenter artificiellement la taille de l'ensemble de données en générant des échantillons supplémentaires à partir des données existantes.
Nous avons appliqué l'algorithme DA en utilisant trois versions et les avons com-

parées avec les méthodes utilisées dans notre étude précédente. Les résultats des ensembles de données simulés et réels, avec des pourcentages variables de données non observées (5%, 10%, 15%, 20%), ont démontré l'efficacité de la troisième version de l'algorithme `DA`, nommée `Haar PX-DA` (Algorithme 7), dans tous les cas sauf pour les grandes valeurs des paramètres (Data3) ou un niveau élevé de `Na` (20%).

Nous pouvons donc conclure que le choix de la méthode pour traiter le problème des `Na` dans une distribution Ex-Gauss pour les données `RT` de petite taille d'échantillon ($n = 50$) dépend de deux critères : la taille des paramètres et le pourcentage de valeurs manquantes.

Une piste de recherche future serait d'élargir notre étude pour inclure tous les cas pertinents et résoudre complètement ce problème.

Code source en R

Nous présentons le code R des trois versions de la méthode d'augmentation de données (DA) proposée dans ce chapitre pour traiter les données manquantes (Na), en tenant compte de nos données réelles.

```
# Replacement of NA with DA algorithm
  x_DA <- x
  for (iter in 1:5) {
   y <- sample(x_DA[!is.na(x_DA)], size = length(x_DA[!is.na(x_DA)]))
   for(i in 1:length(x_DA)){
    if(is.na(x_DA[i])){
     x_DA[i] <- sample(y, size = 1)
    }
   }
  }


# Replacement of NA with PX-DA algorithm
  x_PXDA <- x
  # Define transformation functions between two arguments
  t_g <- function(y, g) {
   return(g + y)
   }

 # Define the inverse of the transformation function
   t_g_inverse <- function(y, g) {
    return(y - g)
   }

   for (iter in 1:5) {
    y <- sample(x_PXDA[!is.na(x_PXDA)], size = length(x_PXDA[!is.na(x_PXDA)]))

    U <- rgamma(length(y), (mean(y)^2)/(sd(y)^2), (sd(y)^2)/mean(y))
    G <- sqrt(U)

    # Define G'
    G_p <- y - G

    # Define V
    V <- rgamma(length(G_p), (mean(G_p)^2)/(sd(G_p)^2), (sd(G_p)^2)/mean(G_p))

    # Define Y'
    y_p <- V + G_p
```

```
    for(i in 1:length(x_PXDA)){
     if(is.na(x_PXDA[i])){
      x_PXDA[i] <- sample(y_p, size = 1)
     }
    }
   }


# Replacement of NA with Haar PX-DA algorithm
    x_HPXDA <- x

   for (iter in 1:5) {
    y <- sample(x_HPXDA[!is.na(x_HPXDA)], size = length(x_HPXDA[!is.na(x_HPXDA)]))

    U <- rgamma(length(y), (mean(y)^2)/(sd(y)^2), (sd(y)^2)/mean(y))
    G <- sqrt(U)

    # Define Y'
    y_p <- t_g(y, G)
    for(i in 1:length(x_HPXDA)){
     if(is.na(x_HPXDA[i])){
      x_HPXDA[i] <- sample(y_p, size = 1)
      }
     }
    }
```

Chapitre 5

Modèles de mélange additifs et multiplicatifs sous différentes distributions

Dans ce chapitre, nous nous intéressons à l'analyse des durées de pauses dans les productions dactylographiée afin de déterminer le modèle d'estimation approprié. La question centrale concerne le type de modèle à adopter : s'agit-il d'un mélange de plusieurs activités cognitives, et si tel est le cas, ce mélange est-il de nature additive ou multiplicative ? De plus, nous examinons si la présence ou l'absence de variables externes peut influencer ces durées. Ce travail, est une collaboration avec M. Thierry Olive, Mme Caroline Bordes du laboratoire CeRCA et mes directeurs de thèse M. Yousri Slaoui et M. Cyril Perret.

5.1 Introduction

L'écriture de texte est un processus complexe qui engage plusieurs niveaux de traitement cognitif, allant de la conceptualisation à la rédaction, en passant par la planification et la révision du texte. Pour comprendre ces processus, il est crucial d'utiliser des méthodes d'analyse en temps réel capables de capter les activités cognitives au moment où elles se produisent. Parmi les variables temporelles collectées, l'analyse des pauses se révèle particulièrement utile, s'appuyant sur deux principes de la chronométrie mentale : la durée des processus mentaux est mesurable, et la durée des pauses reflète la complexité des processus en cours.

Plusieurs facteurs influencent les pauses en écriture, tels que les interruptions environnementales, le genre textuel, la familiarité sémantique, la structure syntaxique, et la longueur des mots. Les deux caractéristiques principales étudiées sont la durée des pauses et leur emplacement dans le texte. Ces analyses mettent en lumière la complexité du système de production du langage, où la durée et la fréquence des pauses sont déterminées par une multitude de processus cognitifs. Par exemple, des pauses plus longues sont généralement observées aux frontières syntaxiques complexes, notamment au début des phrases et des propositions, où se déroulent les activités de planification linguistique et conceptuelle. De plus, la durée des pauses est souvent influencée par les activités de révision, soulignant la nécessité d'interpréter les pauses dans le contexte des modèles théoriques de la production du langage.

L'interprétation des pauses est toutefois complexe, en raison des nombreux facteurs pouvant influencer leur durée et leur fréquence. Les pauses cognitives, qui reflètent les processus cognitifs en cours, présentent souvent une distribution asymétrique avec une queue à droite plus ou moins marquée.

Dans des études antérieures utilisant des modèles de mélange fini, les données étaient souvent normalisées par une transformation logarithmique. Nous proposons ici d'utiliser des distributions de probabilité qui tiennent compte directement de cette asymétrie, évitant ainsi les difficultés liées à la transformation des données.

En conclusion, l'analyse des pauses dans l'écriture à l'aide de modèles de mélange fini permet une compréhension plus fine des processus cognitifs en jeu, tout en offrant des perspectives prometteuses pour la modélisation et la prédiction des pauses en production de texte.

L'enregistrement des frappes au clavier constitue un outil essentiel pour analyser les processus cognitifs impliqués dans l'écriture. En particulier, les intervalles entre les frappes (Inter-Keystroke Intervals, `IKI`) sont souvent étudiés pour différencier les pauses cognitives de celles liées à la sélection des mots ou aux mouvements moteurs. Traditionnellement, cette différenciation repose sur un seuil arbitraire. Toutefois, considérant que la durée des pauses est influencée par plusieurs facteurs, des modèles de mélange fini ont été proposés pour mieux comprendre cette distribution.

[6] ont initié cette approche en suggérant que la distribution des durées de pause pourrait être décrite par un mélange de distributions Log-Normales. Dans ce chapitre, nous approfondissons cette méthodologie en explorant divers aspects liés à l'utilisation des modèles de mélange fini pour analyser les pauses. Nous nous sommes concentrés sur trois points clés : d'abord, l'étude des distributions de probabilité des sous-distributions (Log-Normale, Ex-Wald, et Ex-Gauss) ; ensuite, l'examen de la nature de la relation entre ces sous-distributions (additive ou multiplicative) ; et enfin, l'intégration de covariables comme contraintes dans ces modèles.

Ce chapitre est organisé comme suit : la section 5.2 fournit une définition des modèles de mélange et examine les deux principaux types : additif et multiplicatif (voir sous-sections 5.2.1 et 5.2.2). La sous-section 5.2.3 décrit la méthode d'estimation des modèles de mélange à l'aide de l'algorithme `EM`. Les distributions comparées dans ce chapitre, ainsi que leurs méthodes d'estimation par maximum de vraisemblance, sont abordées dans les sous-sections 5.3.1, 5.3.2 et 5.3.3. La section 5.4 présente également les deux critères utilisés pour sélectionner le modèle optimal. Les résultats numériques sont détaillés dans la section 5.5. Enfin, la section 5.6 conclut notre étude.

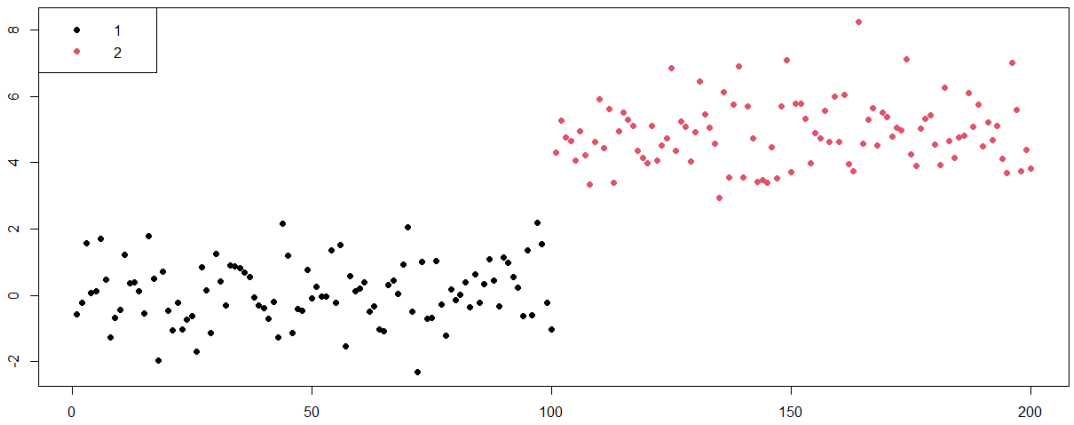
5.2 Modèle de mélange

Un modèle de mélange ([88, 89]) décrit une population comme un mélange de deux ou plusieurs sous-populations, chacune ayant sa propre distribution de probabilité. Le modèle de mélange est utilisé pour estimer les paramètres et les proportions de mélange de chaque composant qui s'ajustent le mieux aux données. Cette estimation peut être réalisée en utilisant plusieurs méthodes, telles que l'estimation du maximum de vraisemblance, l'estimation bayésienne et l'algorithme d'espérance-maximisation (`EM`). Le choix de la méthode dépend des propriétés des données et de

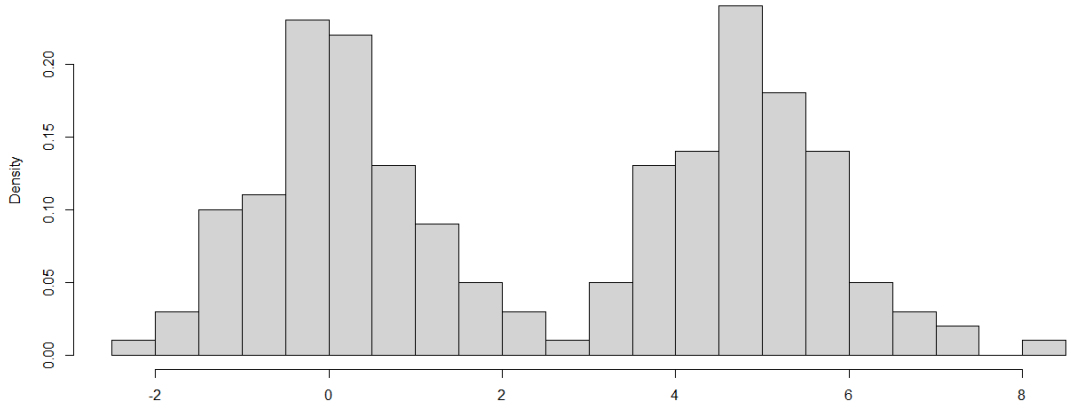
l'application spécifique.
Soit X une variable aléatoire générée à partir d'un mélange de K distributions de composantes, et soit $f_k(x)$ la fonction de densité de probabilité de la k^{me} distribution de composante. Nous notons par $p_1, p_2, \ldots, p_k$ les proportions de mélange. Chaque p_k est un nombre non négatif qui satisfait la condition suivante : $p_1+p_2+\ldots+p_k = 1$.

La Figure 5.1 illustre un modèle de mélange de deux distributions gaussiennes. Dans la sous-Figure (a), les observations sont classifiées en deux groupes distincts, représentant les deux composantes du mélange. La répartition des points montre clairement la séparation entre ces deux classes. La sous-Figure (b) présente un histogramme des observations. Les deux pics distincts dans l'histogramme indiquent visuellement la présence de deux sous-distributions, confirmant que le modèle est bien un mélange de deux lois gaussiennes.

La Figure 5.2 présente un modèle de mélange de trois distributions gaussiennes. L'histogramme des observations est superposé aux courbes de densité correspondant à chacune des trois composantes. Les trois pics distincts, visibles à travers l'histogramme et les courbes de densité, permettent de visualiser la séparation du modèle en trois sous-distributions, confirmant ainsi la présence des trois composantes gaussiennes.



(**a**) Classification des observations, où 1 représente la première classe et 2 la deuxième classe.



(**b**) Histogramme présentant les densités.

FIGURE 5.1 – Représentations graphiques d'un modèle de mélange de deux gaussiens.

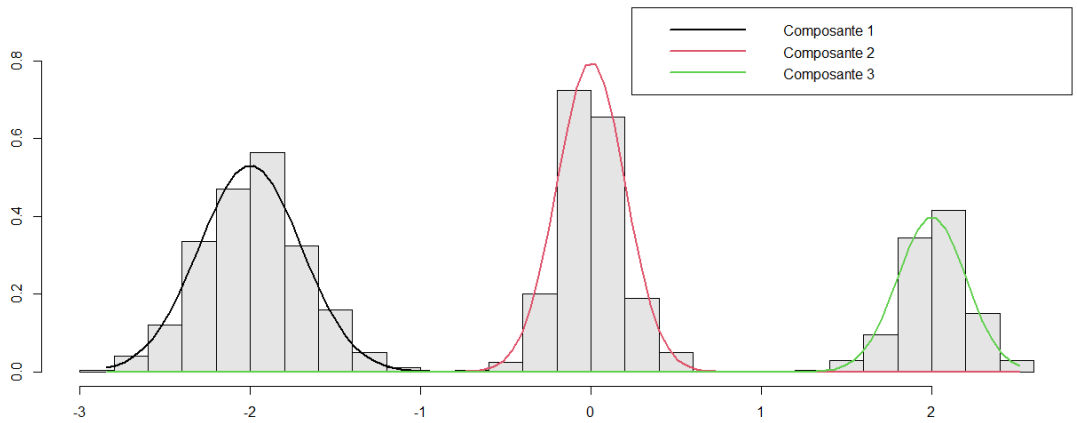


FIGURE 5.2 – Mélange de trois gaussiennes.

La méthode des modèles de mélange se divise en deux types : le mélange additif et le mélange multiplicatif, qui sont définis ci-après.

5.2.1 Modèle de mélange additif

Un modèle de mélange additif est un type de modèle de mélange où la fonction de densité de probabilité du mélange est donnée par la somme des fonctions de densité de probabilité de ses distributions de composantes.
La fonction de densité du modèle de mélange additif peut être exprimée comme suit :

$$\begin{aligned} f(x) &= p_1 \times f_1(x) + p_2 \times f_2(x) + \ldots + p_K \times f_K(x) \\ &= \sum_{k=1}^{K} (p_k \times f_k(x)). \end{aligned}$$

Ce type de modèle de mélange est plus flexible et plus facile à manipuler que le modèle multiplicatif, qui est défini dans la sous-section suivante.

5.2.2 Modèle de mélange multiplicatif

Un modèle de mélange multiplicatif est un type de modèle de mélange où la fonction de densité de probabilité du mélange est donnée par le produit des fonctions de densité de probabilité de ses distributions de composantes. Ainsi, la fonction de densité de probabilité de ce type de modèle de mélange est écrite comme suit :

$$\begin{aligned} f(x) &= p_1 \times f_1(x) \times p_2 \times f_2(x) \times \ldots \times p_K \times f_K(x) \\ &= \Pi_{k=1}^{K} (p_k \times f_k(x)). \end{aligned}$$

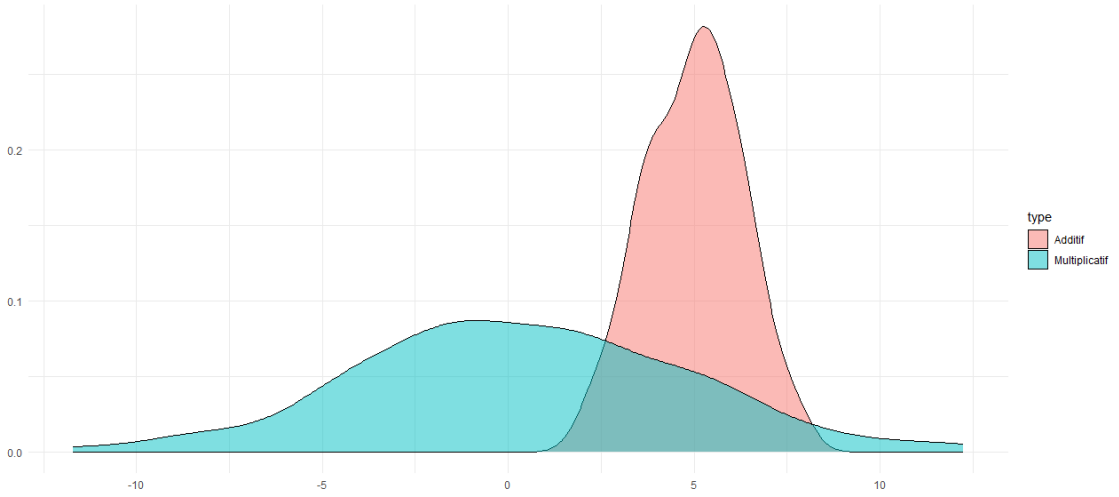


FIGURE 5.3 – Comparaison des densités d'un modèle de mélange additif et multiplicatif de deux gaussiennes.

La Figure 5.3 illustre la différence entre les deux types de mélanges (additif vs multiplicatif). La densité du mélange additif présente un pic marqué, indiquant une forte concentration de valeurs, tandis que le mélange multiplicatif montre une densité plus plate et uniformément répartie. Cela suggère que le mélange additif met en évidence une tendance centrale, alors que le mélange multiplicatif offre une distribution plus équilibrée des données.

Pour estimer les paramètres dans un modèle de mélange, nous appliquons l'algorithme `EM`, défini dans la sous-section suivante, qui est utilisé lorsque la fonction de vraisemblance maximale est impossible à compléter ([37, 124]).

5.2.3 Estimation du modèle

Comme défini précédemment, la fonction de densité de probabilité d'un modèle de mélange fini est donnée comme suit :

$$f(x_i;\psi) = \sum_{k=1}^{K} p_k f_k(x_i;\theta_k),$$

où $X = (x_1, \ldots, x_n)$ est un vecteur de l'ensemble de données observées générées par le modèle de mélange, θ_k sont les paramètres de la fonction de densité de probabilité f_k dans la $k^{\text{ème}}$ composante, et $\psi = \{(p_k, \theta_k) \mid k = 1, \ldots, K\}$ est le vecteur total des paramètres estimés du modèle.
En général, pour estimer ψ, nous maximisons la fonction de log-vraisemblance don-

née par :

$$\log L(\psi, X) = \log\left(\prod_{i=1}^{n} f(x_i;\psi)\right)$$
$$= \sum_{i=1}^{n} \log\left(\sum_{k=1}^{K} p_k f_k(x_i;\theta_k)\right). \quad (5.1)$$

Dans ce cas, l'expression de la fonction de vraisemblance est analytiquement impossible à maximiser en raison des données incomplètes. Pour résoudre ce problème, [37] ont présenté l'algorithme `EM`, où l'idée générale est de générer une variable cachée qui indique quel composant du modèle permet de générer chaque observation. Les données incomplètes sont étiquetées par z_i, pour $i = 1, \ldots, n$. Pour l'échantillon i, le vecteur d'étiquette z_i est de dimension K, où $z_{ik} = 1$ ou 0 selon que x_i a été généré par le $k^{\text{ème}}$ composant du mélange ou non.
Ainsi, le vecteur complet est noté par $Y = (y_1, \ldots, y_n) = (X, Z)$. L'algorithme `EM` est appliqué pour estimer les valeurs des données incomplètes z_{ik} en utilisant cette variable Z. L'équation 5.1 du vecteur complet est réécrite comme suit :

$$\log L_c(\psi, Y) = \sum_{i=1}^{n}\sum_{k=1}^{K} z_{ik} \log(p_k f_k(x_i;\theta_k)).$$

En utilisant la formule de Bayes, la fonction de vraisemblance est décomposée comme suit :

$$L(\psi; X) = L_c(\psi; Y) - L(Z|Y;\psi),$$

En prenant l'espérance conditionnelle des données observées, nous obtenons :

$$\log L(\psi; X) = \mathbf{E}[\log L_c(\psi; Y)|X] - \mathbf{E}[\log L(Z|Y;\psi)|X]$$
$$= Q(\psi;\psi^{(t)}) + H(\psi;\psi^{(t)}).$$

Chaque itération de l'algorithme `EM` est constituée de 2 étapes : `E` (Espérance) et `M` (Maximisation). La première étape (`E`) consiste à calculer l'espérance conditionnelle de la fonction de log-vraisemblance complète basée sur les données observées X et en utilisant l'estimation actuelle de ψ à l'itération t ($\psi^{(t)}$). À l'itération $t+1$, l'étape `E` requise est donnée par :

$$Q(\psi;\psi^{(t)}) = \mathbf{E}\left[\sum_{i=1}^{n}\sum_{k=1}^{K} z_{ik}\log(p_k^{(t)} f_k(x_i;\theta_k^{(t)}))|X,\psi^{(t)}\right]$$
$$= \sum_{i=1}^{n}\sum_{k=1}^{K} \mathbf{E}[z_{ik}|X,\psi^{(t)}]\log(p_k^{(t)} f_k(x_i;\theta_k^{(t)}))$$
$$= \sum_{i=1}^{n}\sum_{k=1}^{K} t_k(x_i;\psi^{(t)})\log(p_k^{(t)} f_k(x_i;\theta_k^{(t)})),$$

avec $t_k(x_i;\psi) \in [0,1]$ indiquant si l'échantillon i de l'observation x_i a été généré par le $k^{\text{ème}}$ composant du modèle, cette quantité est définie par :

$$t_k^{(t+1)}(x_i;\psi) = \frac{p_k^{(t)} f_k(x_i;\theta_k^{(t)})}{\sum_{h=1}^{K} p_h^{(t)} f_h(x_i;\theta_h^{(t)})}.$$

La deuxième étape (M) à l'itération $t+1$ consiste à maximiser $Q(\psi;\psi^{(t)})$ par rapport à ψ, ce qui nous permet d'obtenir l'estimation mise à jour $\psi^{(t+1)}$:

$$\psi^{(t+1)} = \underset{\psi}{\operatorname{argmax}} \sum_{i=1}^{n}\sum_{k=1}^{K} t_k^{(t+1)}(x_i;\psi^{(t)}) \log(p_k^{(t)} f_k(x_i;\theta_k^{(t)})). \tag{5.2}$$

L'estimation des proportions dans l'équation (5.2) est obtenue en maximisant sous la contrainte $\sum_{k=1}^{K} p_k = 1$:

$$\hat{p} = \underset{p}{\operatorname{argmax}} \sum_{1,K} t_k \log p_k,$$

En utilisant la méthode des multiplicateurs de Lagrange, nous pouvons estimer p comme suit :

$$p_k^{(t+1)} = \frac{\sum_{k=1}^{K} t_k^{(t)}}{n}.$$

Pour un mélange gaussien, les paramètres (μ_k, σ_k) sont estimés par :

$$\mu_k^{(t+1)} = \frac{\sum_{i=1}^{n} t_k(y_i;\psi^{(t)}) y_i}{\sum_{i=1}^{n} t_k(y_i;\psi^{(t)})}$$
$$\sigma_k^{(t+1)} = \frac{\sum_{i=1}^{n} t_k(y_i;\psi^{(t)})(y_i - \mu_k^{(t+1)})^2}{\sum_{i=1}^{n} t_k(y_i;\psi^{(t)})}$$

Dans ce chapitre, nous nous intéressons à l'application de cet algorithme à différentes distributions, toutes présentées et définies dans les sections suivantes. Parmi celles-ci, la distribution Ex-Gaussienne est détaillée dans le Chapitre 3.

5.3 Distributions

5.3.1 Distribution de Wald

Une variable aléatoire X suit la distribution de Wald, également connue sous le nom de distribution Inverse Gaussienne, et dépend de deux paramètres : la moyenne μ et le paramètre de forme λ. Nous pouvons écrire la fonction de densité sous la forme suivante :

$$f(x,\mu,\lambda) = \sqrt{\frac{\lambda}{2\pi x^3}} \exp\left(- \frac{\lambda(x-\mu)^2}{2\mu^2 x} \right) \tag{5.3}$$

À partir de l'équation 5.3, la fonction de vraisemblance $L(\mu, \lambda)$ peut être définie comme suit :

$$L(\mu, \lambda) = \Pi_{i=1}^{n} \left[\sqrt{\frac{\lambda}{2\pi x_i^3}} \exp\left(- \frac{\lambda(x_i - \mu)^2}{2\mu^2 x_i} \right) \right], \tag{5.4}$$

En prenant le logarithme dans l'équation (5.4), nous obtenons la fonction de log-vraisemblance :

$$l = \log(L(\mu, \lambda)) = \sum_{i=1}^{n} \left[\frac{1}{2} \log(\lambda) - \log(\pi x_i^3) - \frac{\lambda(x_i - \mu)^2}{2\mu^2 x_i} \right].$$

L'objectif est de trouver les valeurs de μ et λ en maximisant la fonction de log-vraisemblance. Cela peut être réalisé en égalant à zéro les dérivées partielles de l par rapport à μ et λ :

$$\frac{\partial \hat{l}}{\partial \hat{\mu}} = \frac{\lambda(x - \mu)}{\mu^3 x},$$
$$\frac{\partial \hat{l}}{\partial \hat{\lambda}} = \frac{1}{2\lambda} - \frac{(x - \mu)^2}{2\mu^2 x}.$$

Dans la section suivante, nous présentons la distribution Ex-Wald.

5.3.2 Distribution Ex-Wald

Une variable aléatoire T qui suit la distribution Ex-Wald ([110]) repose sur le même principe que la distribution Ex-Gaussienne. Elle est définie comme la somme de deux variables aléatoires indépendantes D et M. Où D suit la distribution Wald, qui dépend de trois paramètres : la moyenne μ, le paramètre de variance σ^2 et le paramètre de forme λ. La fonction de densité de probabilité d'un modèle Wald est donnée dans l'équation (5.3).

La fonction de densité g de la variable aléatoire exponentielle M, avec le paramètre de taux $\gamma > 0$, est donnée par :

$$g(x; \gamma) = \gamma \exp(-\gamma x).$$

Ainsi, la densité de T est définie comme suit :

$$h(t; \mu, \sigma, \lambda, \gamma) = \gamma \exp\left[-\gamma t + \frac{\lambda(\mu - k)}{\sigma^2} \right] \times F(t; k, \sigma, \lambda), \tag{5.5}$$

où k et F sont définis comme suit :

$$k \equiv \sqrt{\mu^2 - 2\gamma\sigma^2} \geq 0,$$

$$F(x; \mu, \sigma, \lambda) = \phi\left(\frac{\mu x - \lambda}{\sigma\sqrt{x}} \right) + \exp\left(\frac{2\lambda\mu}{\sigma^2} \right) \times \phi\left(- \frac{\mu x + \lambda}{\sigma\sqrt{x}} \right),$$

où $\phi()$ est la fonction de répartition de la loi normale standard.

Nous pouvons définir la fonction de vraisemblance de la densité (équation (5.5)) comme suit :

$$\begin{aligned} L(\mu,\sigma,\lambda,\gamma;t) &= \Pi_{i=1}^{n} h(t_i,\mu,\sigma,\lambda,\gamma) \\ &= \gamma^n \exp\left[\sum_{i=1}^{n}\left(-\gamma t_i + \frac{\lambda(\mu-k)}{\sigma^2}\right)\right] \times \Pi_{i=1}^{n} F(t_i;k,\sigma,\lambda), \end{aligned}$$

En appliquant le logarithme de L, nous obtenons :

$$\begin{aligned} l &= \log L(\mu,\sigma,\lambda,\gamma) \\ &= n\log\gamma + \sum_{i=1}^{n}\left(-\gamma t_i + \frac{\lambda(\mu-k)}{\sigma^2}\right) + \sum_{i=1}^{n}\log F(t_i;k,\sigma,\lambda). \end{aligned} \tag{5.6}$$

Ensuite, nous maximisons la fonction de log-vraisemblance l (équation (5.6)). Ainsi, nous obtenons les équations suivantes :

$$\begin{aligned} \frac{\partial\hat{l}}{\partial\hat{\mu}} &= n\frac{\lambda}{\sigma^2} + \sum_{i=1}^{n}\frac{f_1}{F(t_i;k,\sigma,\lambda)}, \\ \frac{\partial\hat{l}}{\partial\hat{\sigma}} &= \frac{-2n\lambda(\mu-k)}{\sigma^3} + \sum_{i=1}^{n}\frac{f_2}{F(t_i;k,\sigma,\lambda)}, \\ \frac{\partial\hat{l}}{\partial\hat{\lambda}} &= n\frac{(\mu-k)}{\sigma^2} + \sum_{i=1}^{n}\frac{f_3}{F(t_i;k,\sigma,\lambda)}, \\ \frac{\partial\hat{l}}{\partial\hat{\gamma}} &= \frac{n}{\gamma} - \sum_{i=1}^{n} t_i + \sum_{i=1}^{n}\frac{f_4}{F(t_i;k,\sigma,\lambda)}, \end{aligned} \tag{5.7}$$

où les fonctions f_1, f_2, f_3 et f_4 dans l'équation (5.7) sont les dérivées de la fonction F par rapport à chaque paramètre et sont définies comme suit :

$$\begin{aligned} f_1 &= \frac{\partial F(t;k,\sigma,\lambda)}{\partial\mu}, \\ f_2 &= \frac{\partial F(t;k,\sigma,\lambda)}{\partial\sigma}, \\ f_3 &= \frac{\partial F(t;k,\sigma,\lambda)}{\partial\lambda}, \\ f_4 &= \frac{\partial F(t;k,\sigma,\lambda)}{\partial\gamma}. \end{aligned}$$

Dans la section suivante, nous présentons la distribution Log-Normale.

5.3.3 Distribution Log-Normale

Soit X une variable aléatoire qui suit la distribution Log-Normale. Nous pouvons alors dire que $Y = \ln X$ suit la distribution normale. La fonction de densité de X

est définie comme suit :

$$f(x, \mu, \sigma) = \frac{1}{x\sigma\sqrt{2\pi}} \exp\left(- \frac{(\log(x) - \mu)^2}{2\sigma^2} \right) \tag{5.8}$$

À partir de la fonction de densité (équation (5.8)), la vraisemblance est définie comme suit :

$$L(\mu, \sigma) = \Pi_{i=1}^{n} \left[\frac{1}{x_i\sigma\sqrt{2\pi}} \exp\left(- \frac{(\log(x_i) - \mu)^2}{2\sigma^2} \right) \right],$$

Ainsi, la fonction de log-vraisemblance est donnée par :

$$l = \log(L(\mu, \sigma)) = \sum_{i=1}^{n} \left[- \log(x_i\sigma\sqrt{2\pi}) - \frac{(\log(x_i) - \mu)^2}{2\sigma^2} \right].$$

Ensuite, la maximisation de la fonction de vraisemblance est obtenue à partir des équations suivantes :

$$\frac{\partial \hat{l}}{\partial \hat{\mu}} = \sum_{i=1}^{n} \frac{\log(x_i) - \mu}{\sigma^2},$$
$$\frac{\partial \hat{l}}{\partial \hat{\sigma}} = \sum_{i=1}^{n} \left[- \frac{1}{\sigma} + \frac{(\log(x_i) - \mu)^2}{\sigma^3} \right].$$

L'objectif de ce chapitre est de comparer les différents modèles présentés afin de sélectionner le meilleur. C'est pourquoi, dans la section suivante, nous présentons deux critères sur lesquels nous nous basons pour choisir le modèle le plus adapté.

5.4 Sélection du modèle

Il existe différentes méthodes pour sélectionner le meilleur modèle. Dans notre étude, nous visons à comparer divers scénarios en fonction de la distribution, du type de mélange et du nombre de modèles inclus dans le mélange. Pour ce faire, nous avons recours à deux indicateurs afin de choisir le meilleur modèle.

L'un des indices les plus connus est le Critère d'Information Bayésien (Bayesian Information Criteria, `BIC`), défini comme suit :

$$\texttt{BIC} = -2\log(\hat{L}) + k\log(n) \tag{5.9}$$

où :
$\hat{L}$ est la fonction de vraisemblance du modèle ;
k est le nombre de paramètres estimés dans le modèle ;
n est la taille de l'échantillon.

Le BIC permet de comparer les modèles en prenant en compte à la fois leur qualité d'ajustement et leur complexité.

[21] ont démontré que le Critère de Vraisemblance Intégrée des Données Complètes (Integrated Complete-data Likelihood, ICL) est moins sensible que le BIC lors de l'ajustement d'un modèle de mélange et de la sélection de modèles produisant une classification significative des données. Le critère ICL est donné par la formule :

$$\texttt{ICL} = -\log(\hat{L}) + \frac{k}{2}\log(n) \tag{5.10}$$

5.5 Application numérique

Les données de dactylographie proviennent du projet de recherche Pro-Text (n°ANR-18-CE23-0024-01). Un participant (étudiant de premier cycle) a rédigé un texte argumentatif sur la légalisation du cannabis pendant 30 minutes. L'outil Inputlog ([78]) a été utilisé pour enregistrer les frappes et les actions du rédacteur. Les analyses ont été effectuées sur l'ensemble des 5028 intervalles entre les frappes (Inter-Keystrokes Interval, IKI). Cela comprenait toutes les activités de dactylographie, c'est-à-dire la production de texte, la ponctuation, les frappes associées (par exemple, TAB, ESPACE, FLÈCHES), ...

Deux facteurs expérimentaux ont été inclus afin d'explorer leur influence sur la classification des pauses :

- Le fait que les IKI soient intra-mot ou inter-mots, codé 0 pour les IKI intra-mot, et 1 pour les inter-mots.
- Le fait que le participant change de main pour frapper deux touches successives, codé 0 pour deux touches successives frappées avec la même main, et 1 pour deux touches successives frappées en changeant de main.

Les deux mesures de qualité d'ajustement (BIC et ICL) présentés dans les équations (5.9) et (5.10) pour tous les modèles de mélange fini testés sont rapportées dans Tableau 5.1. Pour chacun, nous fournissons les valeurs pour les trois distributions : Ex-Gaussienne, Ex-Wald et Log-Normale. Le nombre k correspond au nombre de sous-distributions. La valeur de qualité d'ajustement du meilleur modèle est indiquée en gras.

a - Modèle Additif sans covariables

	Ex-Gaussienne		**Ex-Wald**		**Log-Normale**	
k	BIC	ICL	BIC	ICL	BIC	ICL
2	11275.08	5637.54	**28955.59**	**14477.80**	13460.97	**6730.49**
3	10607.20	5303.60	29394.10	14697.05	15061.32	7530.66
4	10717.95	5358.97	30080.83	15040.42	**13314.07**	6657.03
5	11309.48	5654.74	29790.51	14895.25	13492.84	6746.42
6	11176.03	5588.01	29638.98	14819.49	15862.32	7931.16
7	**10501.73**	**5250.87**	32301.31	16150.65	17478.98	8739.49
8	11286.81	5643.40	31216.87	15608.44	18601.10	9300.55

b - Modèle Additif avec covariables

	Ex-Gaussienne		**Ex-Wald**		**Log-Normale**	
k	BIC	ICL	BIC	ICL	BIC	ICL
2	10333.79	5166.90	33195.93	16597.96	9279.85	4639.93
3	**10048.81**	**5024.41**	31422.82	15711.41	9100.95	4550.47
4	10248.03	5124.02	31885.98	15942.99	9071.19	4535.60
5	11637.56	5818.78	31961.60	15980.80	10045.63	5022.82
6	12168.69	6084.34	29689.77	**14844.89**	8630.92	4315.46
7	10892.39	5446.19	31496.78	15748.39	**8217.21**	**4108.61**
8	10556.69	5278.35	**30412.69**	15206.35	8943.28	4471.64

c - Modèle Multiplicatif sans covariables

	Ex-Gaussienne		**Ex-Wald**		**Log-Normale**	
k	BIC	ICL	BIC	ICL	BIC	ICL
2	**18252.28**	**9126.14**	33590.53	16795.27	**14183.04**	**7091.52**
3	27697.87	13848.94	38314.43	19157.22	30048.70	15024.35
4	35977.49	17988.75	**40308.71**	**20154.36**	40258.99	20129.49
5	34314.02	17157.01	43479.00	21739.50	25729.75	12864.87
6	37127.90	18563.95	47348.75	23674.38	15432.40	7716.20
7	45490.26	22745.13	50386.16	25193.08	29015.40	14507.70
8	50297.87	25148.93	49182.75	24591.38	17059.89	8529.94

d - Modèle Multiplicatif avec covariables

	Ex-Gaussienne		**Ex-Wald**		**Log-Normale**	
k	BIC	ICL	BIC	ICL	BIC	ICL
2	**15260.40**	**7630.20**	**34986.87**	**17493.43**	-209834.10	-104917.00
3	20531.09	10265.54	41149.08	20574.54	14156.20	7078.10
4	28717.87	14358.93	40651.26	20325.63	**-33271.71**	-16635.85
5	25660.39	12830.19	40111.57	20055.79	-24753.30	-12376.65
6	25939.33	12969.66	36162.39	18081.19	-5696.53	-2848.27
7	35689.63	17844.81	43673.11	21836.56	-100161.10	**-50080.56**
8	40568.26	20284.13	49320.21	24660.11	-14420.97	-7210.49

TABLE 5.1 – Comparaison des modèles de mélange additif et multiplicatif avec et sans covariables.

Ces analyses permettent de dégager plusieurs observations. Tout d'abord, il apparaît que la version additive des modèles offre, dans la plupart des cas, un ajustement plus précis que la version multiplicative. Cela suggère que, pour les données analysées, l'association additive des sous-distributions reflète plus fidèlement la répartition des durées de pause. Ensuite, l'analyse du choix de distribution révèle que celui-ci oscille principalement entre la distribution Log-Normale et la distribution Ex-Gaussienne, en fonction des situations spécifiques. En revanche, la distribution Ex-Wald semble moins adaptée dans le contexte de ces données. Un autre point important est l'impact positif de l'intégration de covariables, qui améliore systématiquement l'ajustement et modifie également le nombre de sous-distributions nécessaires. Par exemple, pour une distribution Ex-Gaussienne avec un modèle additif, le nombre de sous-distributions, k, diminue de 7 à 3. Enfin, les deux indices de qualité d'ajustement utilisés ont donné des résultats cohérents dans la majorité des cas.

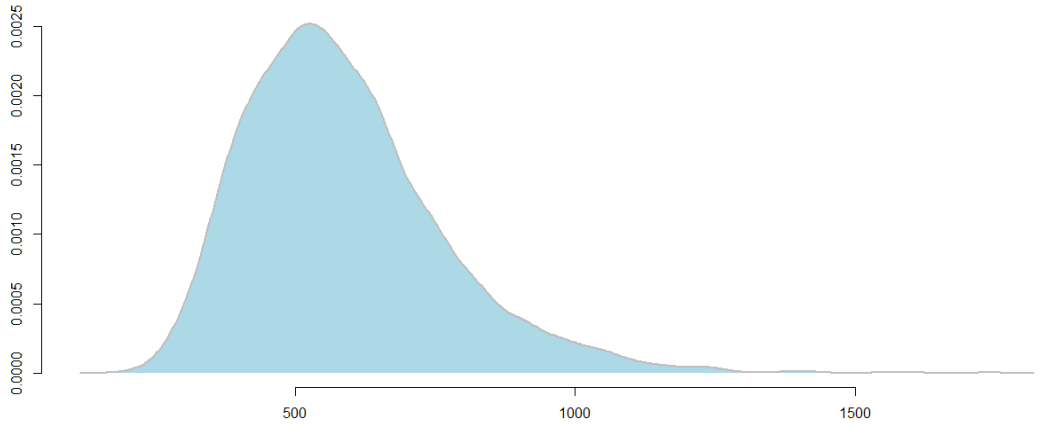


FIGURE 5.4 – Visualisation de la densité du modèle de mélange le plus performant.

La Figure 5.4 illustre la densité du modèle de mélange de trois distributions Ex-Gaussiennes de manière additive, en tenant compte des variables extérieures. Selon le Tableau 5.1, ce modèle a montré la meilleure performance.

	k = 1	k = 2	k = 3
$\texttt{IKI}_1$	27.67	39.20	33.14
$\texttt{IKI}_2$	28.23	38.60	33.17
$\texttt{IKI}_3$	24.75	42.37	32.89
$\texttt{IKI}_4$	25.92	41.07	33.01
$\texttt{IKI}_5$	25.72	41.28	32.99
$\texttt{IKI}_6$	28.14	38.70	33.16

TABLE 5.2 – Pourcentages d'appartenance des six premières durées `IKI` pour un modèle de mélange additif de distribution Ex-Gaussienne avec covariables.

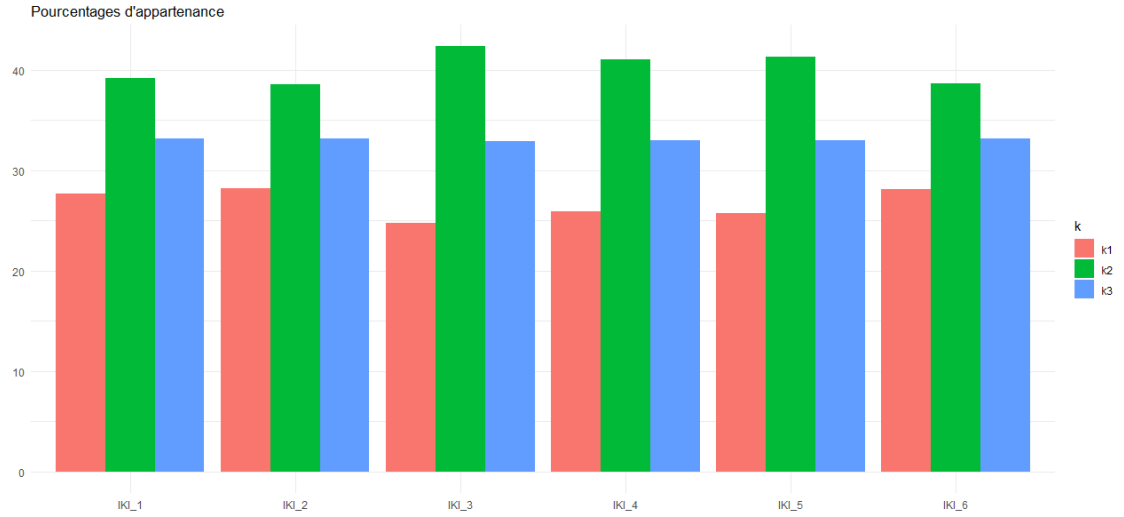


FIGURE 5.5 – Histogramme illustrant la répartition des six premières observations parmi les trois classes dans un modèle de mélange additif de trois distributions Ex-Gaussiennes, en tenant compte des covariables.

Tableau 5.2 présente les pourcentages d'appartenance des six premières valeurs de `IKI` pour un mélange additif de trois distributions Ex-Gaussiennes, en tenant compte des deux covariables. Ce tableau montre que les six premières valeurs sont associées à la deuxième classe. De plus, on peut déduire de ces observations qu'environ 28 % de la deuxième valeur sont influencés par la première distribution, 39 % par la deuxième, et 33 % par la troisième. Ces résultats sont également illustrés dans la Figure 5.5.

5.6 Conclusion

Nous avons comparé des modèles de mélange fini appliqués soit aux données brutes, soit aux résidus d'une analyse de régression linéaire avec deux variables indépendantes. Une meilleure qualité d'ajustement a été systématiquement observée pour les modèles contraints par des covariables liées à si les pauses étaient entre mots ou au sein d'un mot, et lorsque la pause résultait d'un déplacement de doigt de la même main ou des deux mains. Il est intéressant de noter que le nombre k de sous-distributions a diminué avec l'inclusion des covariables. La réduction du nombre de sous-distributions indique que certaines d'entre elles se réfèrent spécifiquement à des aspects du traitement indexés par les facteurs. Cette approche permet de conserver plus de pauses et, surtout, d'éliminer l'influence des facteurs non pertinents.

Les modèles favorisant une association additive des différentes sous-distributions se sont révélés plus efficaces pour expliquer les données que ceux basés sur une association multiplicative.

En somme, les analyses statistiques que nous avons menées offrent une représentation

mathématique des durées de pause. Ces durées résultent de l'influence combinée de plusieurs sous-distributions, chacune contribuant à un certain pourcentage à la durée totale d'une pause, avec des variations significatives d'une pause à l'autre. Il apparaît que cette influence est de nature additive. Les sous-distributions peuvent être liées à différents niveaux de traitement, ce qui est confirmé par la réduction du nombre de sous-distributions lorsque des covariables sont intégrées comme contraintes dans les modèles de mélange. De plus, ces sous-distributions sont bien décrites par des distributions de probabilité capables de capturer l'asymétrie vers la droite, ce qui justifie notre proposition d'utiliser la distribution Ex-Gaussienne. Sur le plan cognitif, la littérature souligne qu'il est difficile de relier une pause à un seul niveau de traitement ([95]). Il est plus plausible qu'une pause soit associée à un ensemble de processus cognitifs, dont l'efficacité détermine partiellement sa durée. Les résultats de cette étude apportent des preuves en faveur d'une association additive. Les modèles de mélange fini Ex-Gaussiens additifs pourraient ainsi constituer une base solide pour explorer, modéliser et même prédire les pauses dans la production de texte. Toutefois, cela nécessite un développement théorique plus approfondi et des preuves empiriques supplémentaires sur la manière dont les pauses reflètent différents processus cognitifs.

Dans le chapitre suivant, nous proposons d'intégrer le calcul quantique afin d'obtenir la distribution q-Ex-Gauss et de la comparer à la distribution Ex-Gauss. Cette approche offre une méthode approfondie pour analyser les données psycholinguistiques et traiter efficacement les données manquantes (`Na`).

Code source en R

Nous présentons le code R utilisé dans ce chapitre pour le mélange de deux modèles suivant une distribution Ex-Gaussienne, avec ou sans covariables.

```
library(MASS)
library(pracma)

# Additive mixture of 2 Ex-Gaussian models without covariates

q=2; # number of classes (number of models)

# Definition of the probability density function of the standard Ex-Gaussian model

pdf_ex_gaussian <- function(x, mu, s, t) {

t*exp(((mu - x)*t)+((t^2)*(s^2))/2)*erfc(((x-mu)/s)-(s*t))

}

# Definition of the density function of the mixture model

pdf_mixture <- function(x, theta) {

  theta[7] * pdf_ex_gaussian(x, theta[1], theta[3], theta[5])
  + theta[8] * pdf_ex_gaussian(x, theta[2], theta[4], theta[6])
 }

# Parameters estimation

itermax=50;

mu1_conver<-rep(0.0,itermax);
s1_conver<-rep(0.0,itermax);
t1_conver<-rep(0.0,itermax);
w1_conver<-rep(0.0,itermax);
mu2_conver<-rep(0.0,itermax);
s2_conver<-rep(0.0,itermax);
w2_conver<-rep(0.0,itermax);
t2_conver<-rep(0.0,itermax);

# Parameters Initialization

for(i in 1:5){
```

```
w=runif(q);   # the mixture proportions;
w<-w/sum(w);
mu=c(w*mean);
s=c(w*sigma);
t=c(w*lambda);
theta <- c(mu,s,t,w)  # the parameter vector;

best_likelihood <- -Inf
best_parameters <- theta

prob <- matrix(0, nrow = n, ncol = q)

for(i in 1:itermax){

 mu1_conver[i]=mu[1];
 s1_conver[i]=s[1];
 w1_conver[i]=w[1];
 t1_conver[i]=t[1];
 mu2_conver[i]=mu[2];
 s2_conver[i]=s[2];
 w2_conver[i]=w[2];
 t2_conver[i]=t[2];

 # Definition of the membership probabilities

 for(i in 1:n){
   for(j in 1:q){
    prob[i,j]<- w[j] * pdf_ex_gaussian(y[i], mu[j], s[j],t[j])
   }
   prob[i, ] <- prob[i, ] / sum(prob[i, ])
    }
   prob[is.na(prob)] <- mean(prob,na.rm=TRUE)


  # Parameters update

  for (j in 1:q) {
    N_j <- sum(prob[, j])

    mu[j] <- sum(prob[, j] * y) / N_j

    s[j] <- sqrt(sum(prob[, j] * (y - mu[j])^2) / N_j)

    t[j] <- 1/mu[j]
```

```
    w[j] <- N_j / n
 }
 theta=c(mu,s,t,w);

 # Likelihood of the model
 likelihood <- sum(pdf_mixture(y, theta), na.rm = TRUE)

  # Best likelihood
  if (!is.na(likelihood) && (is.na(best_likelihood) || likelihood > best_likelihood))
      best_likelihood <- likelihood
      best_parameters <- theta  # Sauvegarder les meilleurs paramètres du modèle
    }

 }
}

theta<-best_parameters

# Density function for each class

f1=pdf_ex_gaussian(y, theta[1], theta[3], theta[5])
f2=pdf_ex_gaussian(y, theta[2], theta[4], theta[6])
f=pdf_mixture(y, theta)

# New vector of y (mixture of y1 and y2)
y_mix=theta[7]*(rnorm(n,theta[1],theta[3])+rexp(n,theta[5]))
+theta[8]*(rnorm(n,theta[2],theta[4])+rexp(n,theta[6]));

# for the mixture of a model with covariates,
# we repeat the same script by taking the vector y_weighted
that contain the variables :

model=lm(y~1+pause_location+hand_switch,data=data)
residuals<-residuals(model)
residual_variances <- residuals^2
weights <- 1 / residual_variances
weighted_model <- lm(y~1+pause_location+hand_switch,data=data, weights = weights)
y_weighted<-predict(weighted_model,data=data)
```

Chapitre 6

Distribution q-Ex-Gaussienne

Dans ce chapitre nous passons du calcul classique au calcul quantique pour analyser la distribution Ex-Gaussienne, ce qui nous permet d'obtenir une fonction de densité de distribution nommé q-Ex-Gaussienne. Ce chapitre constitue seulement une introduction à cette distribution afin d'ouvrir la porte vers la question d'analyse des pauses et si cette densité peut présenter des estimations plus efficace. Ce travail est une collaboration avec Imen Badrani, une collègue de l'Université de Sfax, Tunisie ; sa directeur de thèse, M. Mondher Damak et mes directeurs de thèse M. Yousri Slaoui et M. Cyril Perret.

6.1 Introduction

Le q-calcul, souvent appelé calcul quantique, a été développé au début du XXe siècle par Jackson Frederick ([66, 67]) dans ses recherches sur les fonctions q et les opérateurs de q-dérivation. Depuis, ce domaine a connu un développement considérable, s'appliquant à la physique statistique et aux probabilités. [33] ont proposé une introduction aux concepts fondamentaux du q-calcul, tandis que [114] a mis en avant les opérateurs de q-addition et de q-multiplication, essentiels pour comprendre les propriétés des fonctions q-logarithmiques et q-exponentielles.

Le q-calcul permet de généraliser les distributions statistiques classiques, ce qui a conduit à des applications diverses. Par exemple, [39] ainsi que [52] ont exploré la distribution q-Gaussienne, qui constitue une extension de la distribution gaussienne traditionnelle, utilisée dans des domaines variés comme la mécanique statistique et l'économie.

Ces généralisations ont des applications pratiques, notamment à travers la distribution q-exponentielle étudiée par [18] et [41]. Celle-ci offre une flexibilité accrue par rapport aux distributions exponentielles classiques, permettant une modélisation plus précise de divers phénomènes physiques et statistiques. D'autres chercheurs, tels que [72] et [74], ont utilisé le q-calcul dans des modèles tels que la distribution binomiale et la distribution Weibull, tandis que [25] ont examiné les paramètres d'estimation pour la distribution binomiale q.

Les travaux de Tsallis ([119, 120]) ont marqué un tournant dans le domaine, en généralisant l'entropie de Boltzmann-Gibbs, ce qui a conduit à des distributions de Tsallis, comme la q-Gaussienne, largement employées dans la modélisation de systèmes complexes en mécanique statistique et au-delà.

En résumé, le q-calcul constitue un cadre puissant pour généraliser des concepts mathématiques et statistiques, permettant une modélisation plus adéquate de systèmes non extensifs et de processus complexes. Ces avancées jettent les bases pour de nouvelles recherches dans des domaines aussi variés que la physique théorique, les mathématiques appliquées et l'ingénierie.

L'objectif principal de ce chapitre est de proposer une généralisation de la distribution Ex-Gauss en utilisant le calcul quantique. Pour cela, nous adoptons la méthode de Tsallis, qui repose sur l'intégrale classique sur la ligne réelle ainsi que sur la fonction q-Exponentielle. Une variable aléatoire suivant une distribution q-Ex-Gauss est définie comme la somme de deux variables indépendantes : l'une suit une distribution q-Gaussienne, et l'autre une distribution q-exponentielle.

Ce chapitre est structuré comme suit : la Section 6.2 fournit un aperçu de plusieurs concepts issus de la q-théorie, incluant l'opérateur de q-addition, l'opérateur de q-produit et la fonction q-exponentielle. La Section 6.3 rappelle les distributions q-exponentielle et q-gaussienne. Par la suite, le chapitre présente l'objectif principal, qui consiste à construire la distribution q-Ex-Gauss. Une démonstration de la fonction de densité de probabilité de cette distribution est fournie, accompagnée de simulations illustrant les résultats obtenus. La Section 6.4 présente une application sur des données psychologiques réelles, en comparant la distribution Ex-Gauss et la distribution q-Ex-Gauss. Enfin, la Section 6.5 conclut le chapitre.

6.2 Notions de q-calcul

Nous introduisons à présent les définitions essentielles qui seront employées tout au long de ce chapitre.

6.2.1 Opération de q-addition

La q-addition de deux nombres x et y est définie de la manière suivante :

$$x \oplus_q y = x + y + (1-q)xy,$$

avec $q = 1$, on tombe sur l'addition classique. Il est évident que la q-addition est commutative et associative :

$$x \oplus_q y = y \oplus_q x,$$
$$x \oplus_q (y \oplus_q z) = (x \oplus_q y) \oplus_q z,$$

mais elle n'est pas distributive par rapport à la multiplication habituelle :

$$a(x \oplus_q y) \neq ax \oplus_q ay.$$

En appliquant l'associativité de l'opération $\oplus_q$, nous obtenons la formule suivante :

$$\underbrace{x \oplus_q x \oplus_q \cdots \oplus_q x}_{n \text{ fois}} = \frac{1}{1-q}\{[1+(1-q)x]^n - 1\}.$$

L'élément neutre de la q-addition est zéro :

$$x \oplus_q 0 = x.$$

Par inversion, nous pouvons définir la q-soustraction comme :

$$x \ominus_q y = \frac{x - y}{1 + (1 - q)y}.$$

6.2.2 Opération de q-multiplication

Le q-produit de x et y est défini par :

$$x \otimes_q y = \left[x^{1-q} + y^{1-q} - 1\right]_+^{\frac{1}{1-q}},$$

où :

$$[x]_+ = \begin{cases} x, & \text{si } x \geq 0 \\ 0, & \text{sinon.} \end{cases}$$

On retrouve le produit classique lorsque q tend vers 1 :

$$\lim_{q \to 1}(x \otimes_q y) = xy.$$

L'opération q-multiplication satisfait aussi la commutativité et l'associativité :

$$x \otimes_q y = y \otimes_q x,$$

$$x \otimes_q (y \otimes_q z) = (x \otimes_q y) \otimes_q z.$$

De plus, elle n'est pas distributive :

$$a(x \otimes_q y) \neq ax \otimes_q y, \ (a, x, y \in \mathbb{R}).$$

Sur la base de l'associativité de $\otimes_q$, nous avons la formule suivante :

$$x^{\otimes_q n} = \underbrace{x \otimes_q x \otimes_q \cdots \otimes_q x}_{n \text{ fois}} = \left[nx^{1-q} - (n-1)\right]_+^{\frac{1}{1-q}}.$$

L'élément neutre du q-produit est le 1 :

$$x \otimes_q 1 = x.$$

Encore une fois, par inversion, une q-division peut être définie comme suit :

$$x \oslash_q y = \left[x^{1-q} - y^{1-q} + 1\right]_+^{\frac{1}{1-q}}.$$

Notez que, pour $q \neq 1$, la division par zéro est autorisée.

6.2.3 Fonction q-exponentielle

Dans la littérature, on recense plusieurs formulations de la fonction q-exponentielle, chacune correspondant à un type particulier de q-calcul. Ces différentes variantes trouvent des applications variées dans des domaines tels que les mathématiques combinatoires et la physique statistique ([24, 71]). Dans le cadre de notre travail, la fonction q-exponentielle est définie comme suit :

$$e_q^x = \begin{cases} [1+(1-q)x]^{\frac{1}{1-q}}, & \text{si } q \in \mathbb{R} \setminus \{1\},\ 1+(1-q)x > 0, \\ 0, & \text{sinon.} \end{cases}$$

La Figure 6.1 représente graphiquement e_q^x en fonction de x pour différentes valeurs de $q \neq 1$. La fonction présente des comportements variés en fonction des valeurs de q. Pour des valeurs négatives de q (par exemple, $q = -\frac{5}{2}, -\frac{3}{2}, -\frac{1}{2}$), elle croît lentement pour $x > 0$ et affiche une transition douce autour de $x = 0$. Lorsque $q = 0$, la fonction devient linéaire pour $x > 0$, avec $e_q^x = 1 + x$. Pour de faibles valeurs positives de q (telles que $q = \frac{1}{2}, \frac{3}{2}$), la croissance de la fonction pour $x > 0$ s'accélère, mais elle reste proche de zéro pour $x < 0$. À mesure que q augmente, la croissance devient plus rapide qu'une exponentielle, notamment pour les grandes valeurs de x. Enfin, pour des valeurs élevées de q (comme $q = \frac{5}{2}, 3, 4, 5$), la fonction connaît une croissance très marquée pour $x > 0$, atteignant un pic autour de $x = 0$ avant de chuter rapidement vers zéro. Pour $x < 0$, la fonction reste proche de zéro quelle que soit la valeur de q.

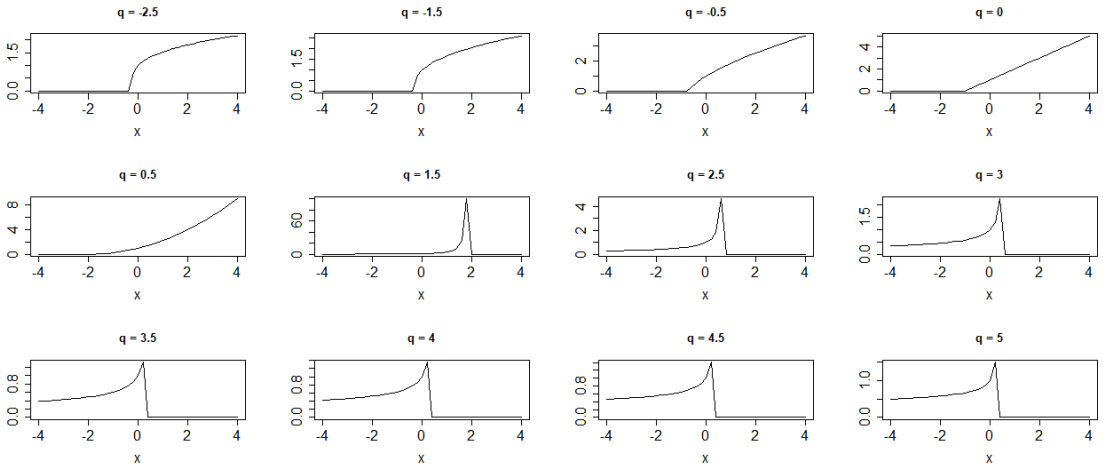


FIGURE 6.1 – e_q^x en fonction de x, avec $q \neq 1$.

À partir de la Figure 6.2, on peut observer que la fonction q-exponentielle converge vers la fonction exponentielle classique lorsque q tend vers 1.

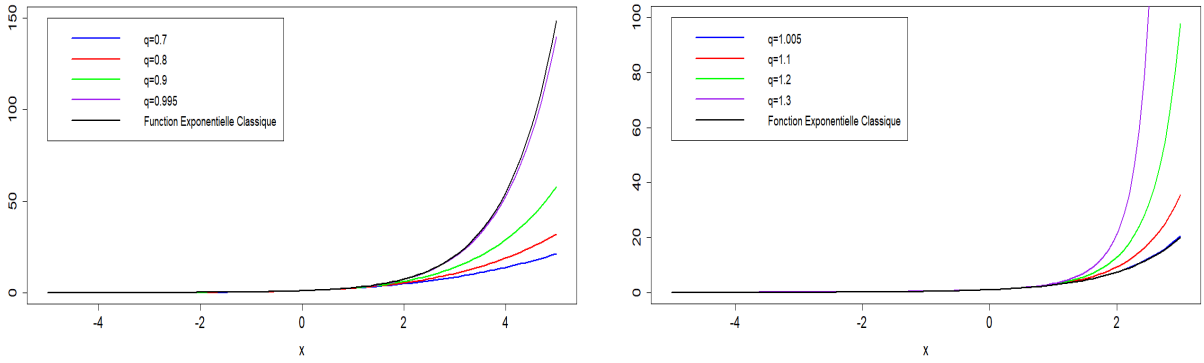


FIGURE 6.2 – Convergence de la fonction e_q^x, avec $q \to 1$.

Notez que nous avons les propriétés suivantes :

1. $e_q^0 = 1$
2. $e_q^{qx} \neq (e_q^x)^q$
3. $\frac{d}{dx} e_q^x = (e_q^x)^q$
4. $e_q^{x \oplus_q y} = e_q^x e_q^y$
5. $e_q^{x+y} = e_q^x \otimes_q e_q^y$
6. $e_q^{x \ominus_q y} = \frac{e_q^x}{e_q^y}$
7. $e_q^{x-y} = e_q^x \oslash_q e_q^y$

6.3 Les principaux résultats

Cette section a pour objectif de définir la fonction de densité de probabilité de la distribution q-Ex-Gauss en s'appuyant sur le calcul quantique (q-calcul).

Soit X une variable aléatoire suivant une loi q-Ex-Gauss. Cette variable peut être décomposée comme la somme de deux variables indépendantes A et B, soit $X = A + B$, où A suit une distribution q-Exponentielle de paramètre $\lambda > 0$, et B suit une distribution q-Gaussienne avec pour paramètres μ et σ^2.

Dans les sections suivantes, nous introduirons d'abord les distributions q-Exponentielle et q-Gaussienne, avant d'en déduire la distribution q-Ex-Gauss.

6.3.1 Distribution q-Exponentielle

La fonction de densité de probabilité de la distribution q-Exponentielle, notée q-Exp(q, λ), est donnée par ([18]) :

$$\psi(x) = \frac{1}{Z_q} e_q^{-\lambda x}, \quad x \in [0, A_q[, \quad q < 2, \tag{6.1}$$

avec λ comme paramètre d'échelle positive, où la fonction Z_q est définie par :

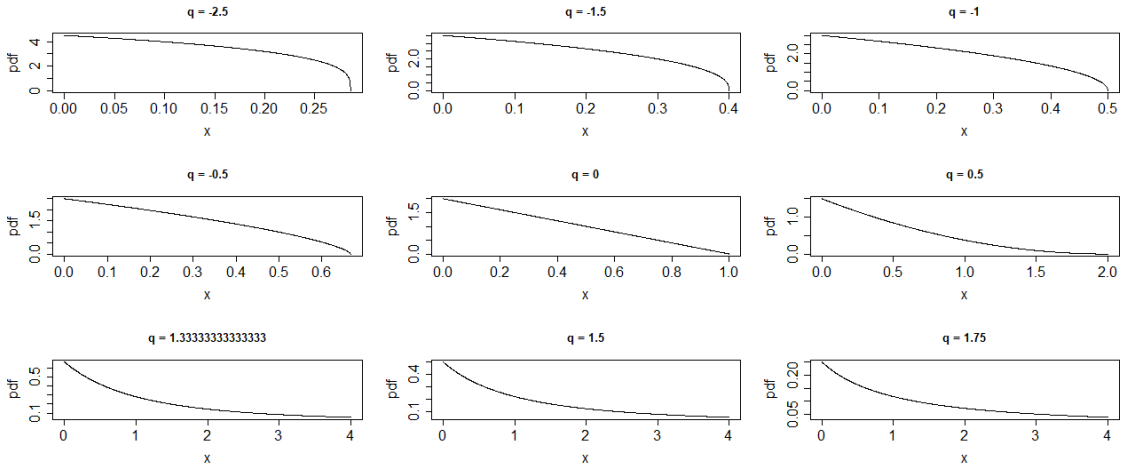
$$Z_q(\lambda) = \frac{1}{\lambda(2-q)}, \quad q < 2,$$

et A_q désigne le point d'extrémité droit :

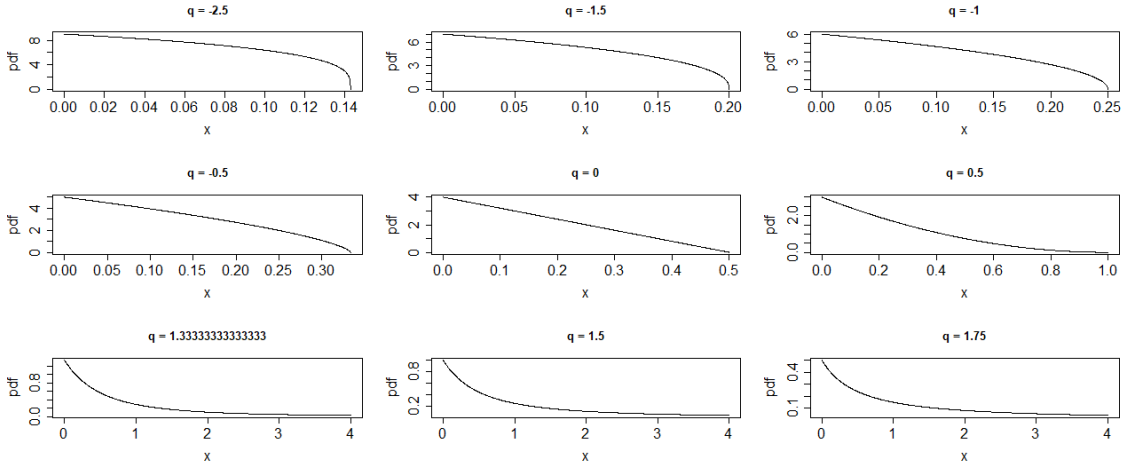
$$A_q = \begin{cases} +\infty & \text{si } 1 \leq q < 2, \\ \dfrac{1}{\lambda(1-q)} & \text{si } q < 1. \end{cases}$$

La variable aléatoire X suivant une distribution q-Exponentielle avec le paramètre $\lambda > 0$ est notée $X \sim q - \mathrm{Exp}(q, \lambda)$.

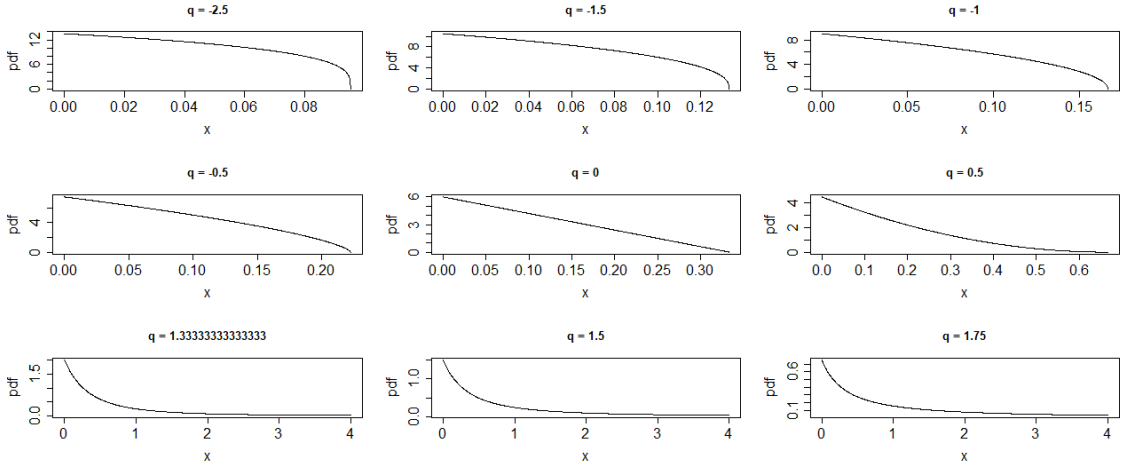
La Figure 6.3 suivante illustre les représentations graphiques de la fonction de densité de probabilité $\psi(x)$ de la distribution q-Exponentielle pour $q < 2$ avec $\lambda = 1, 2$ et 3. Le comportement de cette distribution varie fortement en fonction de q. Pour des valeurs négatives de q, la distribution décroît rapidement et est limitée à un intervalle fini. Lorsque q se rapproche de zéro, la décroissance devient plus linéaire. Pour des valeurs positives de q, en particulier lorsque q dépasse 1, la distribution prend la forme d'une décroissance exponentielle avec des queues beaucoup plus lourdes.



(**a**) $\lambda = 1$



(**b**) $\lambda = 2$



(**c**) $\lambda = 3$

FIGURE 6.3 – La fonction de densité $\psi(x)$ pour différentes valeurs de λ.

Comme indiqué dans la Figure 6.4, lorsque q tend vers 1 dans $\psi(x)$, la distribution Exponentielle classique est obtenue.

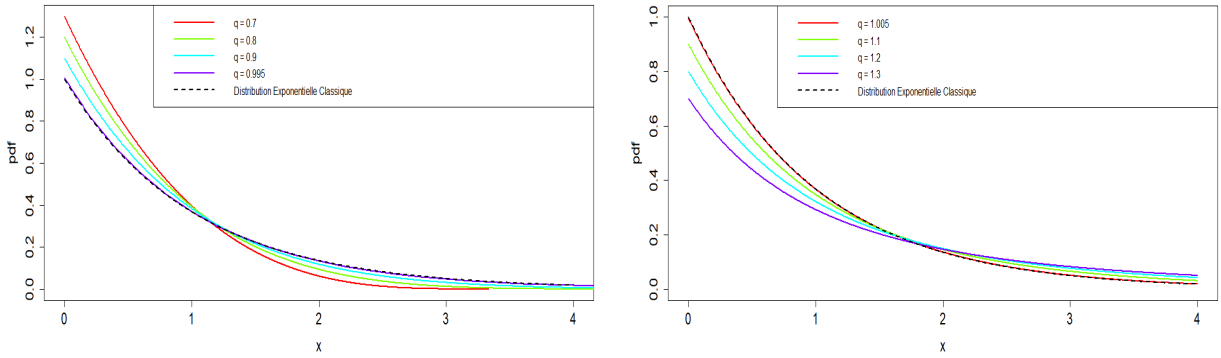


FIGURE 6.4 – Convergence de la distribution q-Exponentielle pour $\lambda = 1$ et $q \to 1$.

6.3.2 Distribution q-Gaussienne

Afin de mieux comprendre la distribution q-Gaussienne, nous nous appuyons sur les travaux de [52]. Soit β un réel positif défini par :

$$\beta = \frac{1}{3-q}.$$

La distribution q-Gaussienne, notée $N_q(\mu, \sigma^2)$, est caractérisée par les paramètres μ et $\sigma^2 > 0$, et sa fonction de densité est donnée par :

$$\phi(x) = \frac{\sqrt{\beta}}{\sigma C_q} e_q^{-\frac{\beta(x-\mu)^2}{\sigma^2}}, \quad x \in D_q, \quad q < 3, \tag{6.2}$$

où D_q est défini comme suit :

$$D_q = \begin{cases} \mathbb{R} & \text{si } 1 \leq q < 3, \\ \left]-\dfrac{\sigma}{\sqrt{\beta(1-q)}} + \mu, \dfrac{\sigma}{\sqrt{\beta(1-q)}} + \mu\right[& \text{si } q < 1, \end{cases}$$

C_q représente la constante de normalisation, donnée par :

$$C_q = \begin{cases} \left(\dfrac{1}{q-1}\right)^{\frac{1}{2}} B\left(\dfrac{3-q}{2(q-1)}, \dfrac{1}{2}\right) & \text{si } 1 < q < 3, \\ \sqrt{\pi} & \text{si } q = 1, \\ \left(\dfrac{1}{1-q}\right)^{\frac{1}{2}} B\left(\dfrac{2-q}{1-q}, \dfrac{1}{2}\right) & \text{si } -\infty < q < 1, \end{cases}$$

où $B(a, b)$ désigne la fonction bêta.

La représentation graphique de C_q est donnée dans la Figure 6.5, où l'on observe que la valeur de C_q atteint $\sqrt{\pi}$ lorsque $q = 1$. Deux courbes sont présentées dans cette figure : la courbe noire représente C_q, qui démarre près de 0 pour des valeurs négatives de q, reste faible jusqu'à $q = 1$, puis croît rapidement lorsque q dépasse 2. La ligne horizontale rouge indique la constante $\sqrt{\pi}$ (approximativement 1,772) comme référence. C_q reste en dessous de $\sqrt{\pi}$ pour $q < 1$, l'intersecte à $q = 1$, et la dépasse nettement pour $q > 1$.

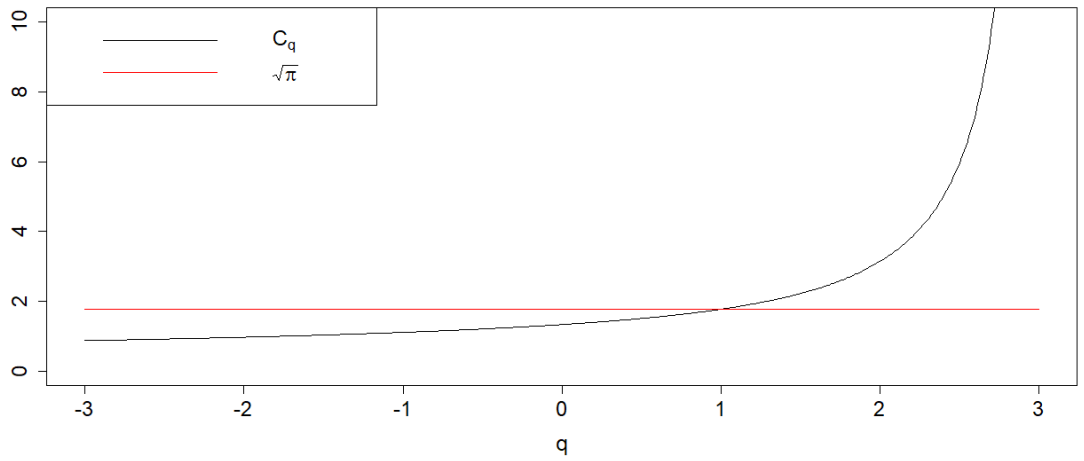
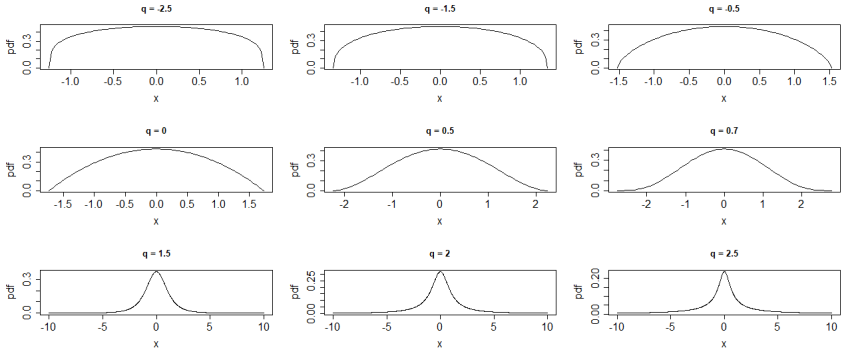


FIGURE 6.5 – La fonction de normalisation C_q en fonction de q, avec $q < 3$

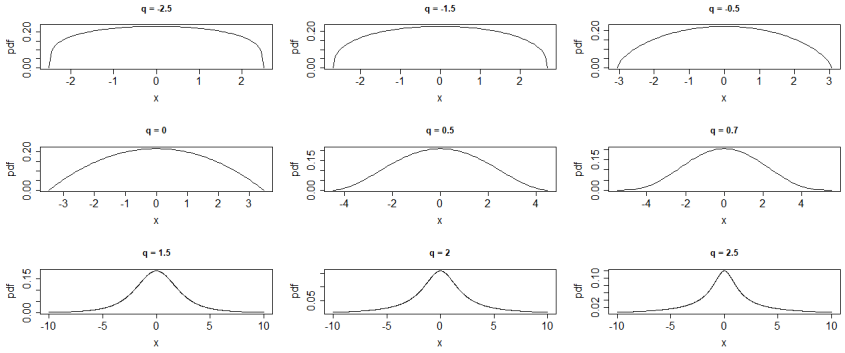
Une variable aléatoire q-Gaussienne générale X avec pour paramètres μ et σ^2 est notée $X \sim N_q(\mu, \sigma^2)$. Le cas particulier où $\mu = 0$ et $\sigma^2 = 1$ est désigné comme une variable aléatoire q-Gaussienne standard, notée $Y \sim N_q(0, 1)$. La densité de probabilité de la distribution q-Gaussienne standard s'exprime ainsi :

$$\phi(y) = \frac{\sqrt{\beta}}{C_q} e_q^{-\beta y^2}.$$

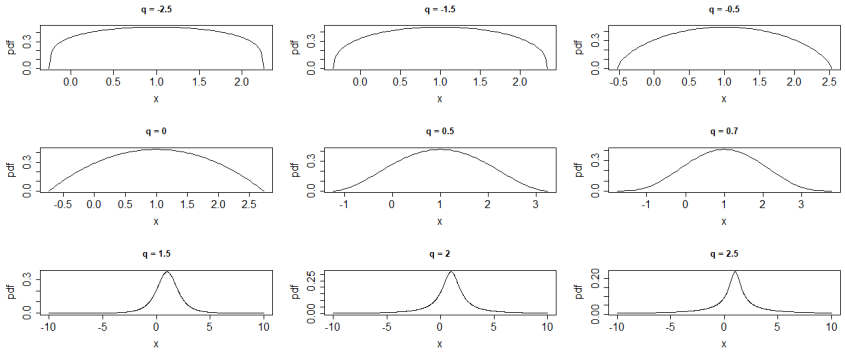
La Figure 6.6 illustre les représentations graphiques de la fonction de densité de probabilité $\phi(x)$ de la distribution q-Gaussienne pour $q < 3$. La forme de la distribution q-Gaussienne varie considérablement en fonction de q. Pour des valeurs négatives de q, la distribution est aplatie, rappelant une distribution uniforme. Lorsque q se rapproche de zéro, elle devient plus lisse et en forme de cloche, avec une dispersion plus large qu'une gaussienne standard. Pour des valeurs positives de q supérieures à 1, la distribution se caractérise par un pic au centre et des queues plus lourdes, ce qui lui permet de représenter des valeurs extrêmes.



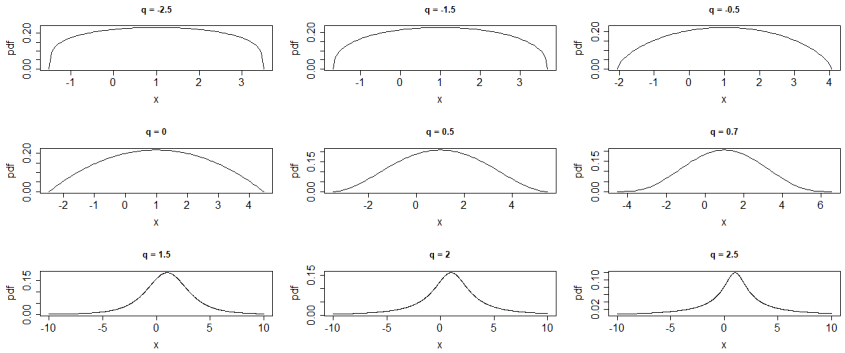
(**a**) $\mu = 0, \sigma = 1$



(**b**) $\mu = 0, \sigma = 2$



(**c**) $\mu = 1, \sigma = 1$



(**d**) $\mu = 1, \sigma = 2$

FIGURE 6.6 – La fonction de densité $\phi(x)$ pour différentes valeurs de μ et σ.

Comme indiqué dans la Figure 6.7, lorsque q tend vers 1 dans $\phi(x)$, la distribution Gaussienne classique est obtenue.

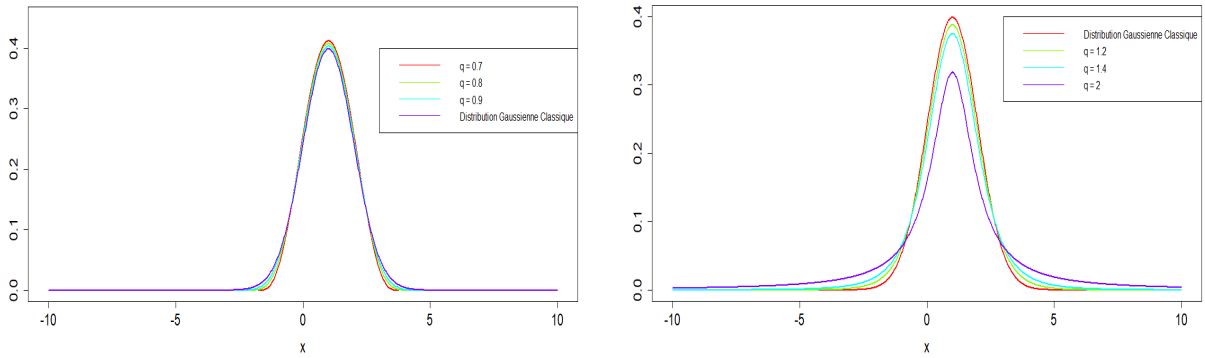


FIGURE 6.7 – Convergence de la distribution q-Gaussienne pour $\mu = 1$, $\sigma = 1$ et $q \to 1$.

6.3.3 Distribution q-Ex-Gauss

Soit X une variable aléatoire suivant une distribution q-Ex-Gaussienne :

$$X \sim \text{q-Ex-Gauss}(q, \mu, \sigma, \lambda)$$

Dans ce cas, X peut être exprimé comme la somme de deux variables indépendantes, A et B. La variable A suit une distribution q-Exponentielle avec un taux λ (voir équation (6.1)), tandis que B suit une distribution q-Gaussienne avec les paramètres μ et σ (voir équation (6.2)). Par conséquent, la densité de probabilité de $X = A + B$ est obtenue en effectuant la convolution des fonctions de densité ψ et ϕ :

- $1 < q < 2$:

$$g(x) = \int_{-\infty}^{\infty} \phi(t)\psi(x-t)dt = \int_{-\infty}^{x} \phi(t)\psi(x-t)dt + \int_{x}^{\infty} \phi(t)\psi(x-t)dt$$

Comme $\psi(x-t) = 0$ pour $t > x$, cela devient :

$$g(x) = \int_{-\infty}^{x} \phi(t)\psi(x-t)dt \tag{6.3}$$

Ensuite, en substituant les équations (6.1) et (6.2) dans l'équation (6.3), on obtient :

$$g(x) = \frac{\sqrt{\beta}}{\sigma C_q Z_q} \int_{-\infty}^{x} e_q^{-\beta \frac{(t-\mu)^2}{\sigma^2}} e_q^{-\lambda(x-t)} dt$$

En utilisant les notations de q-multiplication et de q-addition :

$$g(x) = \frac{\sqrt{\beta}\, e_q^{-\lambda x}}{\sigma C_q Z_q} \otimes_q \int_{-\infty}^{x} e_q^{-\beta \frac{(t-\mu)^2}{\sigma^2} \oplus_q \lambda t} dt$$
$$= \frac{\sqrt{\beta} e_q^{-\lambda x}}{\sigma C_q Z_q} \otimes_q \int_{-\infty}^{x} e_q^{-\beta \frac{(t-\mu)^2}{\sigma^2} + \lambda t} \otimes_q e_q^{-\beta \lambda t (1-q) \frac{(t-\mu)^2}{\sigma^2}} dt$$

- $q < 1$:

$$g(x) = \int_{-\frac{\sigma}{\sqrt{\beta(1-q)}}+\mu}^{\frac{\sigma}{\sqrt{\beta(1-q)}}+\mu} \phi(t)\psi(x-t)dt = \int_{-\frac{\sigma}{\sqrt{\beta(1-q)}}+\mu}^{x} \phi(t)\psi(x-t)dt$$

Ainsi, on obtient :

$$g(x) = \frac{\sqrt{\beta} e_q^{-\lambda x}}{\sigma C_q Z_q} \otimes_q \int_{-\frac{\sigma}{\sqrt{\beta(1-q)}}+\mu}^{x} e_q^{-\beta \frac{(t-\mu)^2}{\sigma^2} + \lambda t} \otimes_q e_q^{-\beta \lambda t (1-q) \frac{(t-\mu)^2}{\sigma^2}} dt$$

Nous effectuons un changement de variable qui peut être appliqué dans les deux cas :

$$y = h(t) = \frac{(t-\mu)}{\sigma} - \lambda\sigma$$

Ainsi, nous avons :

$$t = \sigma y + \lambda\sigma^2 + \mu, dy = \frac{1}{\sigma} dt \implies dt = \sigma dy$$

si :

$$t = x \implies y = h(x),$$
$$t \to -\infty \implies y \to -\infty$$

En réécrivant la densité de probabilité $g(x)$, on obtient :

- $1 < q < 2$:

$$g(x) = \frac{\sqrt{\beta}}{C_q Z_q} e_q^{-\lambda x} \otimes_q \int_{-\infty}^{h(x)} e_q^{-\beta(y+\lambda\sigma)^2 + \lambda(\sigma y + \lambda\sigma^2 + \mu)} \otimes_q e_q^{-\beta\lambda(1-q)(y+\lambda\sigma)^2(\sigma y + \lambda\sigma^2 + \mu)} dy$$

- $q < 1$:

$$g(x) = \frac{\sqrt{\beta}}{C_q Z_q} e_q^{-\lambda x} \otimes_q \int_{-\frac{1}{\sqrt{\beta(1-q)}}-\lambda\sigma}^{h(x)} e_q^{-\beta(y+\lambda\sigma)^2 + \lambda(\sigma y + \lambda\sigma^2 + \mu)} \otimes_q e_q^{-\beta\lambda(1-q)(y+\lambda\sigma)^2(\sigma y + \lambda\sigma^2 + \mu)} dy$$

Maintenant, nous posons :

$$u(y) = e_q^{-\beta(y+\lambda\sigma)^2 + \lambda(\sigma y + \lambda\sigma^2 + \mu)} \text{ et } v(y) = e_q^{-\beta\lambda(1-q)(y+\lambda\sigma)^2(\sigma y + \lambda\sigma^2 + \mu)}$$

$u(y)$ et $v(y)$ peuvent être écrits en fonction de la q-multiplication :

$$u(y) = e_q^{-\beta y^2} \otimes_q e_q^{\lambda\sigma(1-2\beta)y} \otimes_q e_q^{\lambda^2\sigma^2(1-\beta)+\lambda\mu}$$

$$v(y) = e_q^{-\beta\lambda\sigma(1-q)y^3} \otimes_q e_q^{-\beta\lambda(1-q)(3\lambda\sigma^2+\mu)y^2} \otimes_q e_q^{-\beta\lambda^2\sigma(1-q)(3\lambda\sigma^2+2\mu)y} \otimes_q e_q^{-\beta\lambda^3\sigma^2(1-q)(\lambda\sigma^2+\mu)}$$

Donc, l'expression $u(y) \times v(y)$ est donnée par :

$$\begin{aligned} u(y) \times v(y) = e_q^{-\beta\lambda(1-q)y^3} &\otimes_q e_q^{-\beta(1+\lambda(1-q)(3\lambda\sigma^2+\mu))y^2} \\ &\otimes_q e_q^{\lambda\sigma((1-2\beta)-\beta\lambda(3\lambda\sigma^2+2\mu))y} \\ &\otimes_q e_q^{\lambda^2\sigma^2((1-\beta)-\beta\lambda(1-q)(\lambda\sigma^2+\mu))+\lambda\mu} \end{aligned}$$

Ainsi, la fonction $g(x)$, qui est la densité de probabilité q-Ex-Gauss, peut être représentée comme suit :

$$\begin{aligned} g(x) = \frac{\sqrt{\beta}\, e_q^{\lambda^2\sigma^2\left((1-\beta)-\beta\lambda(1-q)(\lambda\sigma^2+\mu)\right)+\lambda(\mu-x)}}{C_q Z_q} \\ \otimes_q \int_{S_q}^{\frac{x-\mu}{\sigma}-\lambda\sigma} e_q^{-\beta\lambda\sigma(1-q)y^3} \otimes_q e_q^{-\beta\left(1+\lambda(1-q)(3\lambda\sigma^2+\mu)\right)y^2} \\ \otimes_q e_q^{\lambda\sigma\left((1-2\beta)-\lambda\beta(1-q)(3\lambda\sigma^2+2\mu)\right)y} dy, \quad x \in M_q \end{aligned} \tag{6.4}$$

où S_q est défini comme suit :

$$S_q = \begin{cases} -\infty & \text{si } 1 \leq q < 2, \\ -\dfrac{1}{\sqrt{\beta(1-q)}} & \text{si } q < 1, \end{cases}$$

et M_q est défini comme suit :

$$M_q = \begin{cases} \mathbb{R} & \text{si } 1 \leq q < 3, \\ \left]-\dfrac{\sigma}{\sqrt{\beta(1-q)}} + \mu, \dfrac{\sigma}{\sqrt{\beta(1-q)}} + \mu + \dfrac{1}{\lambda(1-q)}\right[& \text{si } q < 1, \end{cases}$$

La Figure 6.8 suivante illustre les représentations graphiques de la fonction de densité de probabilité $g(x)$ de la distribution q-Ex-Gauss pour différentes valeurs de q (avec $q < 2$), avec $\mu = 1$, $\sigma^2 = 1$ et $\lambda = 1$. Pour des valeurs négatives de q, la distribution présente des décroissances abruptes et des intervalles plus resserrés. À mesure que q approche de zéro, la distribution prend une forme proche d'une gaussienne avec des queues modérées. Lorsque q dépasse 1, les queues s'alourdissent et le pic s'élargit.

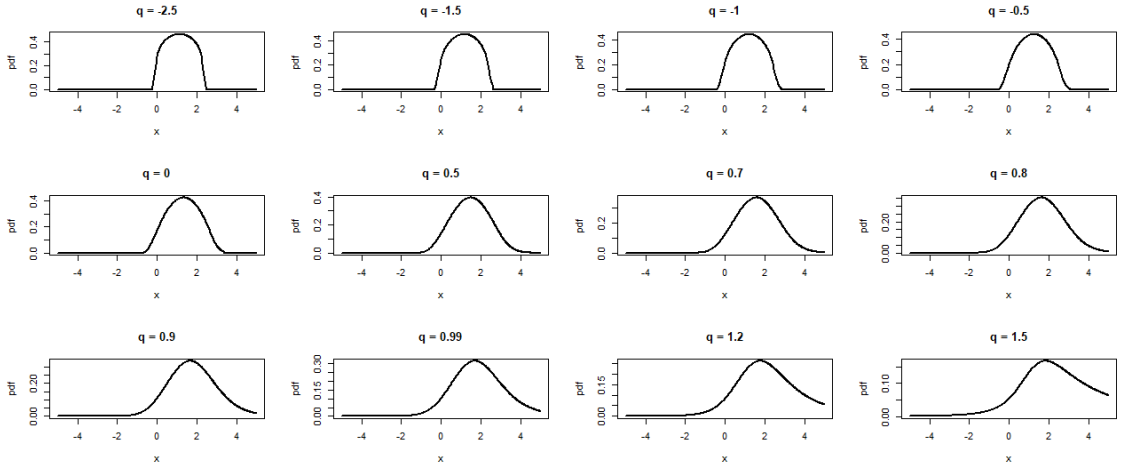


FIGURE 6.8 – La fonction de densité $g(x)$ pour différentes valeurs de q avec q<2, $\mu = 1$, $\sigma = 1$ et $\lambda = 1$.

Comme indiqué dans la Figure 6.9, lorsque q tend vers 1 dans $g(x)$, la distribution Ex-Gauss classique est obtenue.

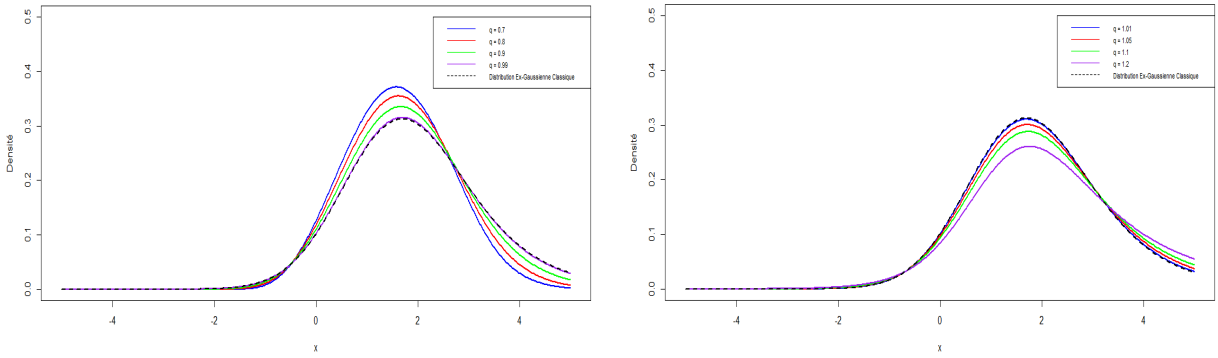


FIGURE 6.9 – Convergence de la distribution q-Ex-Gauss pour $\mu = 1$, $\sigma = 1$, $\lambda = 1$ et $q \to 1$.

Dans la section suivante, nous présentons une application de la densité obtenue grâce au calcul quantique sur des données psychologiques réelles, en la comparant à la distribution Ex-Gauss, qui repose sur des méthodes classiques.

6.4 Application sur des données réelles

Les jeux de données utilisés dans cette étude ont été collectés lors d'une expérience psychologique menée auprès d'étudiants en deuxième année de psychologie à l'Université de Poitiers. L'expérience consistait à présenter un stimulus visuel, un simple carré blanc, sur un écran. Chaque participant devait répondre le plus rapidement

possible dès l'apparition du stimulus, affiché 150 fois au cours de l'expérience. Les temps de réaction ont été enregistrés pour chaque réponse, constituant un vecteur de 150 observations pour chaque participant, représentant ainsi les temps de réaction individuels. [1]
En utilisant ce vecteur de temps de réaction, nous avons comparé les deux distributions, Ex-Gauss et q-Ex-Gauss.

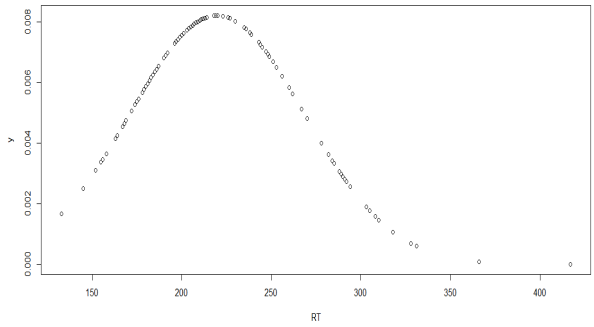
Dans le Tableau 6.1, on observe que les deux paramètres μ et σ présentent des valeurs similaires dans les deux modèles, tandis que le paramètre $\tau = \frac{1}{\lambda}$ montre une différence notable entre les deux modèles.

	Ex-Gauss	q-Ex-Gauss
μ	219.7845	219.7882
σ	48.4025	48.5722
λ	0.0063	0.0045

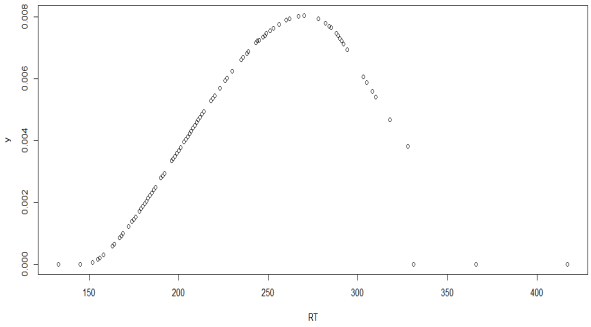
TABLE 6.1 – Comparaison des paramètres (μ, σ, λ) entre les distributions Ex-Gauss et q-Ex-Gauss.

Dans la Figure 6.10, nous illustrons graphiquement les densités des distributions Ex-Gauss et q-Ex-Gauss, mettant en évidence l'évolution de la forme de la densité en fonction des différentes valeurs du paramètre q.
La Figure 6.10 (a) représente la densité de la distribution Ex-Gauss, qui exhibe une forme asymétrique caractéristique avec une queue à droite. Ce type de distribution est bien connu dans les recherches psychologiques sur les temps de réaction, où la plupart des réponses sont concentrées autour d'une valeur centrale, suivie d'une décroissance graduelle, représentée par cette queue.
Dans les graphiques de la Figure 6.10 (b), (c), et (d), qui illustrent la densité de la distribution q-Ex-Gauss pour différentes valeurs de q, on observe que la forme classique est altérée au fur et à mesure que q s'écarte de 1. Avec q= -0.8, la densité montre une asymétrie plus marquée, tandis qu'à q= 0.8, la forme devient plus aplatie. Enfin, pour q= 1.2, la densité présente une augmentation notable vers les queues, ce qui reflète une croissance de la variance. On remarque donc que la valeur de q modifie la forme et l'étalement de la distribution : une augmentation de q tend à augmenter la densité des valeurs extrêmes, ce qui se traduit par des valeurs de densité plus élevées dans les queues, tandis qu'une diminution de q concentre la densité autour de la moyenne.

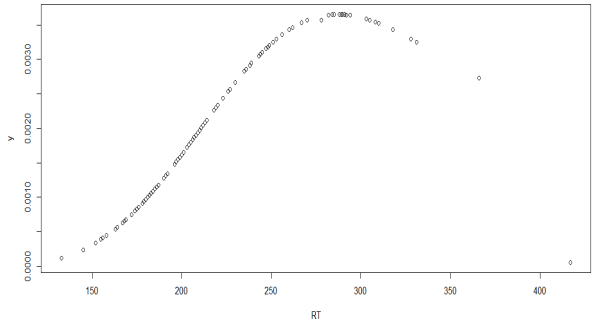
1. L'expérience et la collecte des données ont été réalisées par Christine Ros et David Chesnet du CeRCA (Centre de Recherche sur la Cognition et l'Apprentissage, Université de Poitiers), que je tiens à remercier chaleureusement.



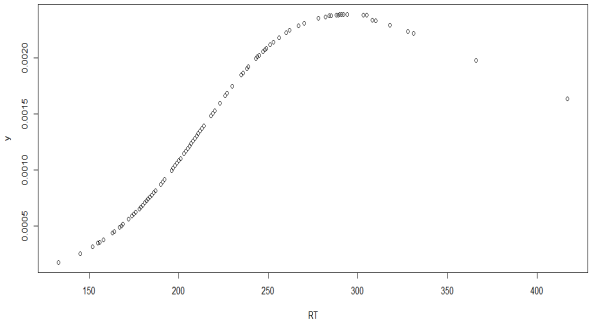
(**a**) Ex-Gauss



(**b**) q-Ex-Gauss, q= -0.8



(**c**) q-Ex-Gauss, q= 0.8



(**d**) q-Ex-Gauss, q= 1.2

FIGURE 6.10 – Représentation graphique de la densité des distributions Ex-Gauss et q-Ex-Gauss.

6.5 Conclusion

Dans ce chapitre, notre objectif principal était de généraliser la forme de la distribution q-Ex-Gauss$(q, \mu, \sigma, \lambda)$. Ce passage d'un calcul classique à un calcul quantique ouvre la voie à l'exploration de l'impact potentiel du q-calcul sur l'analyse des données psychologiques liées à la production écrite. Nous avons testé la densité obtenue sur des données réelles, et les résultats provisoires montrent que cette densité pourrait, à l'avenir, fournir des estimations efficaces. Dans les études menées jusqu'à présent, la valeur de q a été fixée ; dans nos travaux futurs, nous nous intéresserons à l'estimation de q ainsi que des trois paramètres μ, σ et τ tout en proposant des méthodes adaptées pour l'estimation de ces paramètres.

Chapitre 7

Conclusion Générale

Cette thèse a permis d'explorer en profondeur les processus cognitifs liés aux données psycholingustiques, en mettant l'accent sur les temps de réaction et les pauses dactylographiées. Grâce à une combinaison d'approches théoriques et empiriques, nous avons proposé plusieurs méthodologies innovantes pour traiter les défis associés à l'estimation des paramètres et à la gestion des données manquantes dans des contextes de petits échantillons. L'intégration de l'inférence bayésienne, des algorithmes de maximisation stochastique de l'espérance, ainsi que des méthodes d'augmentation de données, a constitué un apport majeur pour améliorer la robustesse des modèles statistiques utilisés.

Dans les premiers chapitres, nous avons démontré l'efficacité d'un modèle linéaire à effets mixtes (`LMEM`) pour estimer les paramètres en présence de données manquantes, avec un biais des paramètres plus faible par rapport aux approches concurrentes. L'introduction de l'algorithme `SEM` a permis d'affiner cette estimation en tenant compte des interactions entre variables fixes et aléatoires.

Le travail sur la distribution Ex-Gauss a apporté des améliorations significatives dans l'estimation des paramètres pour des petits échantillons, particulièrement grâce à l'utilisation de l'algorithme de Bootstrap pour une initialisation plus fiable. L'analyse des données incomplètes avec des méthodes telles que l'imputation multiple (`MI`) et les approches bayésiennes (`B`) a montré des résultats prometteurs, malgré une compétition entre ces deux méthodes selon les scénarios étudiés. En réponse aux limites des approches standards pour la gestion des données manquantes, nous avons introduit une méthode d'augmentation de données basée sur l'algorithme `DA`, qui s'est avérée particulièrement efficace pour les petits échantillons. Cette méthode a permis de compenser la variabilité observée dans les résultats, et d'offrir des estimations plus stables dans des conditions de données manquantes importantes. Ces avancées ont abouti à la soumission d'un package en langage R, nommé "fitexgauss", qui permet d'estimer la distribution Ex-Gauss en intégrant l'ensemble de nos résultats, notamment pour les petits échantillons et les données incomplètes.

L'analyse des pauses cognitives, un élément central de notre étude, a révélé que ces moments de réflexion sont mieux modélisés par des modèles de mélange additifs, plutôt que multiplicatifs. Cela démontre que les pauses résultent de la combinaison de plusieurs processus cognitifs simultanés, plutôt que d'interactions entre ces processus. Cette approche fournit une nouvelle perspective pour comprendre la complexité cognitive sous-jacente aux moments de pause dans la production écrite.

Enfin, nous avons introduit l'utilisation du calcul quantique à travers la distribution q-Ex-Gauss. Cette généralisation du modèle classique Ex-Gauss, fondée sur le q-

calcul, constitue un cadre prometteur pour améliorer l'analyse des pauses et des temps de réaction.

Dans le cadre de travaux futurs, nous envisageons de comparer les performances de cette nouvelle distribution à celles du modèle classique, en utilisant diverses méthodes d'estimation des paramètres, telles que la méthode bayésienne déjà appliquée à l'Ex-Gauss, la méthode des q-moments proposée dans [15, 16], ou encore d'autres approches. Par ailleurs, nous prévoyons d'étendre l'application du q-calcul au modèle de mélange présenté dans cette thèse. Ces travaux convergent vers la mise en place d'un modèle robuste et d'un package dédié, mis à la disposition des psychologues.

Bibliographie

[1] Alamargot, D., Dansac, C., Chesnet, D., and Fayol, M. *Parallel processing before and after pauses : A combined analysis of graphomotor and eye movements during procedural text production*, Writing and Cognition, (2007), pp. 11-29.

[2] Aldridge, M., and Fontaine, L. *Using keystroke logging to capture the impact of cognitive complexity and typing fluency on written language production*, Observing writing : Insights from keystroke logging and handwriting, (2019), pp. 285-305.

[3] Alves, R.A., Castro, S.L., and Olive, T. *Execution and pauses in writing narratives : Processing time, cognitive effort and typing skill*, International Journal of Psychology, 43 (2008), pp. 969-979.

[4] Annaki, I. and Rahmoune, M. *Overview of Data Augmentation Techniques in Time Series Analysis*, International Journal of Advanced Computer Science and Applications, 15 (2024), pp. 1201-1211.

[5] Austin, P., White, I., Lee, D. and Buuren, S. *Review Missing Data in Clinical Research : A Tutorial on Multiple Imputation*, Canadian Journal of Cardiology, 37 (2021), pp. 1322-1331.

[6] Baaijen, V. M., Galbraith, D., and de Glopper, K. *Keystroke Analysis : reflections on procedures and measures*, Written Communication, 29 (2012), pp. 246-277.

[7] Baayen, R.H., Davidson, D.J. and Bates, D.M. *Mixed-effects modeling with crossed random effects for subjects and items*, Journal of Memory and Language, 29 (2008), pp. 390-412. [CrossRef]

[8] Backlund, G. *An Overview of the Data Augmentation Algorithm*, Department of Statistics, University of Florida, (2017). [CrossRef]

[9] Balota, D. A., and Spieler, D. H. *Word frequency, repetition, and lexicality effects in word recognition tasks : Beyond measures of central tendency*, Journal of Experimental Psychology : General, 128 (1999), pp. 32–55.

[10] Balota, D. A., and Yap, M. J. *Moving beyond the mean in studies of mental chronometry : the power of response time distribution analyses*, Current Directions in Psychological Science, 20 (2011), pp. 160-166.

[11] Barhoumi, M. *Traitement des Données Manquantes dans les Données de Panel : Cas des Variables Dépendantes Dichotomiques*, PhD diss., Université Laval, (2006). [CrossRef]

[12] Barratt, A., Wyer, C.P., Hatala, R., McGinn, T., Dans, L.A., Keitz, S., Moyer, V. and For, GG. *Evidence-Based Medicine Teaching Tips Working Group. Tips for learners of evidence-based medicine : 1. Relative risk reduction, absolute risk reduction and number needed to treat*, Journal of Applied Mathematics and Computation, 171 (2004).

[13] Barr, D.J., Levy, R., Scheepers, C. and Tily, H.J. *Random effects structure for confirmatory hypothesis testing : Keep it maximal*, Journal of Memory and Language, 68 (2013), pp. 255-278. [CrossRef]

[14] Bates, D., Mächler, M., Bolker, M.B. and Walker, C.S. *Fitting Linear Mixed-Effects Models Using lme4*, Journal of Statistical Software, 67 (2015), pp. 1-48. [CrossRef]

[15] Ben Mrad, O., Masmoudi, A. and Slaoui, Y. *Some properties of q-Gaussian distributions*, Communication in Statistics-Theory and Methods, 53 (2024), pp. 6315-6337.

[16] Ben Mrad, O., Masmoudi, A. and Slaoui, Y. *The discrete q-Gaussian distribution $N_q(\mu, \sigma^2)$: Properties and parameters estimation*, Physics Letters A, 493 (2024).

[17] Bennett, D.A. *How can I deal with missing data in my study*, Australian and New Zealand Journal of Public Health, 25 (2001), pp. 464–469.

[18] Bercher, J. F. and Vignat, C. *A new look at q-exponential distributions via excess statistics*, Physica A : Statistical Mechanics and its Applications, 387 (2008), pp. 5422–5432.

[19] Betancourt, M. *A conceptual introduction to Hamiltonian Monte Carlo*, arXiv :1701.02434, (2017).

[20] Bhatnagar, S., Prasad, H. and Prashanth, L. *Stochastic Recursive Algorithms for Optimization*, Springer : Heidelberg, Germay, 434 (2013), pp. 17-28.

[21] Biernacki, C., Celeux, G. and Govaert, G. *Assessing a Mixture Model for Clustering with the Integrated Completed Likelihood*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 22 (2000), pp. 719-725.

[22] Billingsley, P. *Statistical methods in Markov chains*, The annals of mathematical statistics, (1961), pp.12-40.

[23] Boik, J.R. *The Analysis of Two-Factor Interactions in Fixed Effects Linear Models*, Journal of Educational Statistics, 18 (1993), pp. 1-40. [CrossRef]

[24] Borges, P.E. *A possible deformed algebra and calculus inspired in nonextensive thermostatistics*, Physica A : Statistical Mechanics and its Applications, 340 (2004), pp. 95-101.

[25] Bouzida, I., Masmoudi, A. and Zitouni, M. *Estimation parameters for the Binomial q-distribution*, Communications in Statistics-Theory and Methods, 50 (2020), pp. 1-13.

[26] Burbeck, S. and Luce, R. *Evidence from auditory simple reaction times for both change and level detectors*, Perception & psychophysics, 32 (1982), pp. 117–133.

[27] Celeux, G. and Diebolt, J. *Une Version de Type Recuit Simule de L'algorithme EM*, Ph.D. Thesis, INRIA, Le Chesnay, France, (1989).

[28] Celeux, G. and Diebolt, J. *The SEM algorithm : A probabilistic teacher algorithm derived from the EM algorithm for the mixture problem*, Computational Statistics Quarterly, 2 (1985), pp. 73-82.

[29] Celeux, G., Chauveau, D. and Diebolt, J. *On Stochastic Versions of the EM Algorithm*, Ph.D. Thesis, INRIA, Le Chesnay, France, (1995).

[30] Celeux, G. and Diebolt, J. *A stochastic approximation type EM algorithm for the mixture problem*, International Journal of Probability and Stochastic Processes, 41 (1992), pp. 119-134. [CrossRef]

[31] Cheng, R.C. *Resampling methods*, Handbooks in OR & MS, 13 (2006), pp. 415-453.

[32] Chenu, F., Pellegrino, F., Jisa, H., and Fayol, M. *Interword and intraword pause threshold in writing*, Frontiers in Psychology, (2014), pp. 5-183.

[33] Cheung, P. and Kac, V.G. *Quantum calculus*, Heidelberg : Springer, (2001).

[34] Cover, T. and Hart, P. *Nearest neighbor pattern classification*, IEEE transactions on information theory, 13 (1967), pp. 21-27.

[35] Davison, A. and Hinkley, D. *Bootstrap Methods and Their Applications*, Cambridge Univ. Press, Cambridge, (1997).

[36] Delyon, B., Lavielle, M. and Moulines, E. *Convergence of a stochastic approximation version of the EM algorithm*, Annals of Statistics, 1 (1999), pp. 94-128. [CrossRef]

[37] Dempster, A., Laird, N. and Rubin, D. *Maximum-likelihood from incomplete data via the EM algorithm*, The Journal of the Royal Statistical Society, Series B, 39 (1977), pp. 1-38.

[38] Dempster, A.P., Rubin, D.B. and Tsutakawa, R.K. *Estimation in Covariance Components Models*, Journal of the American Statistical Association, 76 (1981), pp. 341-353. [CrossRef]

[39] Diaz, R. and Pariguan, E. *On the Gaussian q-distribution*, Journal of mathematical analysis and applications, 358 (2009), pp. 1-9.

[40] Donders, F.C. *On the speed of mental processes*, Acta Psychologica, 30 (1869/1969), pp. 412-431.

[41] Droguett, E. L., Lins, I. D., Moura, M. J. d. C. and Negreiros, A. C. S. V. d. *Reliability data analysis of systems in the wear-out phase using a (corrected) q-Exponential likelihood*, Reliability Engineering & System Safety, 197 (2020), pp. 106-787.

[42] Duane, S., Kennedy, A.D., Pendleton, B.J. and Roweth, D. *Hybrid Monte Carlo*, Physics Letters B, 195 (1987), pp. 216-222.

[43] Enders, C.K. *Applied Missing Data Analysis* ; Guilford Press : New York, NY, USA, (2010).

[44] Efron, B. *Bootstrap methods : another look at the jackknife*, Annals of Statistics, 7 (1979), pp. 1-26.

[45] Efron, B. and Tibshirani, R. *An Introduction to the Bootstrap*, Chapman & Hall/CRC, New York, (1993).

[46] El Haj, A., Slaoui, Y., Solier, C. and Perret, C *Bayesian Estimation of The Ex-Gaussian Distribution*, Statistics, Optimization And Information Computing, 9 (2021), pp. 809-819.

[47] Faraway, J.J. *Extending the Linear Model with R*, 2nd ed. ; CRC Press : Boca Raton, FL, USA, (2016).

[48] Fitousi, D. *Linking the ex-Gaussian parameters to cognitive stages : Insights from the Linear Ballistic Accumulator (LBA) model*, The Quantitative Methods for Psychology, 162 (2020), pp. 91-106.

[49] Fix, E. and Hodges, J. *Discriminatory Analysis. Nonparametric Discrimination : Consistency Properties*, International Statistical Review, 57 (1951), pp. 238-247.

[50] Foulley, J.L. *Mixed Model Methodology. Part I : Linear Mixed Models*, In Technical Report, Université de Montpellier : Montpellier, France, (2015).

[51] Foulin, J. N. *To what extent does pause location predict pause duration in adults' and children's writing ?*, Cahiers de Psychologie Cognitive/Current Psychology of Cognition, 17 (1998), pp. 601–620.

[52] Furuichi, S. (2009). *On the maximum entropy principle and the minimization of the Fisher information in Tsallis statistics*, Journal of Mathematical Physics, 50 (2009), pp. 1-13.

[53] Gelman, A., Gary, K. and Chuanhai, L. *Not asked and not answered : multiple imputation for multiple surveys*, Journal of the American Statistical Association, 93 (1999), pp. 846-857.

[54] Geman, S. and Geman, D. *Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 6 (1984), pp. 721-741.

[55] Giesselmann, M. and Schmidt-Catran, A.W. *Interactions in fixed effects regression models*, Sociological Methods & Research, 51 (2022), pp. 1100-1127. [CrossRef]

[56] Grund, S., Lüdtke, O. and Robitzsch, A. *Multiple imputation of missing data in multilevel models with the R package mdmb : A flexible sequential modeling approach*, Behavior Research Methods, 53 (2021), pp. 2631–2649.

[57] Hadfield, J.D. *MCMC methods for multi-response generalized linear mixed models : The MCMCglmm R package*, Journal of Statistical Software, 33 (2010), pp. 1-22. [CrossRef]

[58] Matuschek, H., Kliegl, R., Vasishth, S., Baayen, H. and Bates, D. *Balancing Type I error and power in linear mixed models*, Journal of Memory and Language, 94 (2017), pp. 305-315. [CrossRef]

[59] Hastie, T., Tibshirani, R. and Friedman, J. *Prototype methods and nearest-neighbors. The Elements of Statistical Learning : Data Mining, Inference, and Prediction*, Springer Series in Statistics, (2001), pp. 411-435.

[60] Hastings, W.K. *Monte Carlo sampling methods using Markov Chains and their applications*, Biometrika, 57 (1970), pp. 97-109. [CrossRef]

[61] Heitjan, D. and Basu, S. *Distinguishing "Missing at Random" and "Missing Completely at Random"*, The American Statistician, 50 (1996), pp. 207-213.

[62] Hobert, J. and Marchev, D. *A theoretical comparison of the data augmentation, marginal augmentation and PX-DA algorithms*, The Annals of Statistics, 36 (2008), pp. 532-554.

[63] Hobert, J.P. *The Data Augmentation Algorithm : Theory and Methodology*, Handbook of Markov Chain Monte Carlo, (2011), pp. 253-291.

[64] Hohle, R.H. *Inferred components of reaction times as function of foreperiod duration*, Journal of Experimental Psychology. 69 (1965), pp. 382-386.

[65] Iglesias, G., Talavera, E., González-Prieto, A., Mozo, A. and Gómez-Canaval, S. *Data Augmentation techniques in time series domain : a survey and taxonomy*, Neural Computation & Application, 35 (2023), pp. 10123–10145.

[66] Jackson, F. H. *On q-functions and a certain difference operator*, Earth and Environmental Science Transactions of the Royal Society of Edinburgh, 46 (1909), pp. 253-281.

[67] Jackson, F. H. *On q-definite integrals*, The Quarterly Journal of Pure and Applied Mathematics, 41 (1910), pp. 193–203.

[68] Jain, A., Bansal, R., Kumar, A. and Singh, K.D. *A comparative study of visual and auditory reaction times on the basis of gender and physical activity levels of medical first year students*, International Journal of Applied and Basic Medical Research, (2015).

[69] Jiang, J. *Asymptotic Analysis of Mixed Effects Models* ; CRC Press : Boca Raton, FL, USA, (2017).

[70] Jiang, J., Wand, M.P. and Bhaskaran, A. *Usable and precise asymptotics for generalized linear mixed model analysis and design*, The Journal of the Royal Statistical Society, Series B, 84 (2022), pp. 55-82. [CrossRef]

[71] Joo, S.H. *q -Calculus Revisited*, arXiv preprint arXiv :2106.03855, (2021).

[72] Jose, K. K. and Naik, S. R. *On the q-Weibull Distribution and Its Applications*, Communications in Statistic—Theory and Methods, 38 (2009), pp. 912–926.

[73] Kalos, M.H. and Whitlock, P.A. *What is Monte Carlo ?*, John Wiley & Sons, Ltd, (2008), pp. 1-5.

[74] Kemp, A. W. *The q-beta-geometric distribution as a model for fecundability*, Communications in Statistics-Theory and Methods, 30 (2001), pp. 2373–2384.

[75] Khare, K. and Hobert, J. *A spectral analytic comparison of trace-class data augmentation algorithms and their sandwich variants*, The Annals of Statistics, 39 (2011), pp. 2585-2606.

[76] Khodadadian, A., Parvizi, M., Teshnehlab, M. and Heitzinger, C. *Rational Design of Field-Effect Sensors Using Partial Differential Equations, Bayesian Inversion, and Artificial Neural Networks*, Sensors, 22 (2022), pp. 4785. [CrossRef]

[77] Lavielle, M. and Mbogning, C. *An improved SAEM algorithm for maximum likelihood estimation in mixtures of non linear mixed effects models*, Statistics and Computing, 24 (2014), pp. 693-707. [CrossRef]

[78] Leijten, M., and Van Waes, L. *Keystroke Logging in Writing Research : Using Inputlog to Analyze Writing Processes*, Written Communication, 30 (2013), pp. 358-392. [CrossRef]

[79] Li, P., Stuart, E.A. and Allison, D.B. *Multiple Imputation : A Flexible Tool for Handling Missing Data*, JAMA, 314 (2015).

[80] Little, R.J. and Rubin, D.B. *Statistical Analysis with Missing Data* ; John Wiley & Sons : Hoboken, NJ, USA, (2019).

[81] Liu, J. and Wu, Y. *Parameter expansion for data augmentation*, Journal of the American Statistical Association, 94 (1999), pp. 1264-1274.

[82] Logan, G. *Toward an Instance Theory of Automatization*, Psychological Review, 95 (1988), pp. 492–527.

[83] Luce, R.D. *Response times : Their role in inferring elementary mental organization*, Oxford : Oxford University Press, (1986).

[84] Mackay, D.J.C. *Introduction to Monte Carlo methods*, Springer, (1998), pp. 175-204.

[85] Mack, C., Su, Z., and Westreich, D. *Managing missing data in patient registries : addendum to registries for evaluating patient outcomes : a user's guide*, Agency for Healthcare Research and Quality (US), (2018).

[86] Martin, D.A., Quinn, M.K. and Park, H.J. *MCMCpack : Markov chain Monte Carlo in R*, Journal of Statistical Software, 42 (2011). [CrossRef]

[87] McCulloch, C.E. ; Searle, S.R. ; Neuhaus, J.M. *Generalized, Linear, and Mixed Models*, 2nd ed. ; John Wiley & Sons : New York, NY, USA, (2008).

[88] McLachlan, G. and Peel, D. *Finite Mixture Models*, Wiley Series in Probability and Statistics, John Wiley & Sons, (2000).

[89] McLachlan, G. J., Lee, S. X. and Rathnayake, S. I. *Finite mixture models*, Annual Review of Statistics, 6 (2019), pp. 355-378.

[90] Meng, X. and Van Dyk, D. *Seeking efficient data augmentation schemes via conditional and marginal augmentation*, Biometrika, 86 (1999), pp. 301-320.

[91] Metropolis, N., Rosenbluth, A.W., Rosenbluth, M.N., Teller, A.H. and Teller, E. *Equation of State Calculations by Fast Computing Machines*, The Journal of Chemical Physics, 21 (1953), pp. 1087-1092. [CrossRef]

[92] Myung, I.J. *Tutorial on maximum likelihood estimation*, Journal of Mathematical Psychology, 47 (2003), pp. 90-100.

[93] Neal, R.M. *MCMC using Hamiltonian dynamics*, Chapman & Hall / CRC Press, (2011).

[94] Noii, N., Khodadadian, A., Ulloa, J., Aldakheel, F., Wick, T., Francois, S. and Wriggers, P. *Bayesian Inversion with Open-Source Codes for Various One-Dimensional Model Problems in Computational Mechanics*, Archives of Computational Methods in Engineering, 29 (2022), pp. 4285-4318. [CrossRef]

[95] Olive, T., Alves, R. A., and Castro, S. L. *Cognitive processes in writing during pause and execution periods*, European Journal of Cognitive Psychology, 21 (2009), pp. 758-785.

[96] Panhart, X. and Samson, A. *Extension of the SAEM algorithm for nonlinear mixed models with 2 levels of random effects*, Biostatistics, 10 (2008), pp. 121-135. [CrossRef]

[97] Perret, C.and Bonin, P. *Which variables should be controlled for to investigate picture naming in adults ? A Bayesian meta-analysis*, Behavior Research Methods, 51 (2019), pp. 2533-2545. [CrossRef]

[98] Perret, C. and Olive, T. *Spelling and writing words : Theoretical and methodological advances*, Brill, 39(2019).

[99] Perret, C. and Solier, C. *Application of a Bayesian approach for exploring the impact of syllable frequency in handwritten picture naming*, Journal of Cognitive Psychology, 34 (2022), pp. 622-635. [CrossRef]

[100] Ratcliff, R. *Methods for dealing with reaction time outliers*, Psychological Bulletin, 114 (1993), pp. 510-532. [CrossRef]

[101] Richard, F., Samson, A. and Cuénod, A.C. *A SAEM algorithm for the estimation of template and deformation parameters in medical image sequences*, Statistics and Computing, 19 (2009), pp. 465-478. [CrossRef]

[102] *Roberts, O. G. Rosenthal, S. J. General state space Markov chains and MCMC algorithms*, Probability Surveys, 1 (2004), pp. 20-71.

[103] Robbins, H. and Monro, S. *A Stochastic Approximation Method*, Annals of Mathematical Statistics, 22 (1951), pp. 400-407. [CrossRef]

[104] Robin, G., Josse, J., Moulines, E. and Tibshirani, R. *Main effects and interactions in mixed and incomplete data frames*, Journal of the American Statistical Association, 115 (2019), pp. 1292-1303. [CrossRef]

[105] Roelofs, A. *One hundred fifty years after Donders : Insignts form unpublished data, a replication, and modeling of his reaction times*, Acta Psychologica, 191 (2018), pp. 228–233.

[106] Rubin, D.B. *Inference and Missing Data*, Biometrika, 63 (1976), pp. 581-590.

[107] Rubin, D.B. *Multiple Imputation for Nonresponse in Surveys*, Wiley Series in Probability and Statistics, 10 (1987).

[108] Santos, F. *L'algorithme EM : Une Courte Présentation*, CNRS, UMR, 5199 : Talence, France, (2015).

[109] Schafer, L.J. and Graham, W.J. *Missing Data : Our View of the State of the Art*, Psychological Methods, 7 (2002), pp. 147–177.

[110] Schwarz, W. *The ex-Wald distribution as a descriptive model of response times*, Behavior Research Methods, Instruments, & Computers, 33 (2001), pp. 457-469.

[111] Seaman, S., Galati, K., Jackson, D. and Carlin, J. *What is meant by "missing at random" ?*, Statistical Science, (2013), pp. 257-268.

[112] Shorten, C., Khoshgoftaar, T.M. and Furht, B. *Text Data Augmentation for Deep Learning*, Journal of Big Data, 101 (2021).

[113] Stroup, W. *Generalized Linear Mixed Models*, CRC Press : Boca Raton, FL, USA, (2013).

[114] Suyari, H. *Mathematical structures derived from the q-multinomial coefficient in Tsallis statistics*, Physica A : Statistical Mechanics and its Applications, 368 (2006), pp. 63–82.

[115] Slaoui, Y. and Nuel, G. *Parameter Estimation in a Hierarchical Random Intercept Model with Censored Response : An Approach using a SEM Algorithm and Gibbs Sampling*, Sankhya : The Indian Journal of Statistics, 76 (2014), pp. 210–233.

[116] Slaoui, Y., El Haj, A., Zakkour, A., Bordes, C., Perret, C. and Dumoncel, J. *ExGaussEstim : Quantile Maximization Likelihood Estimation and Bayesian Ex-Gaussian Estimation*, R package version 0.1.2. 2023, [CrossRef.

[117] Sun, J., Du, W. and Shi, N. *A Survey of kNN Algorithm*, Information Engineering and Applied Computing, 1 (2018).

[118] Tenny, S. and Hoffman, M.R. *Relative Risk*, StatPearls Publishing, Treasure Island (FL), (2023).

[119] Tsallis, C. *Possible generalization of Boltzmann-Gibbs statistics*, Journal of statistical physics, 52 (1988), pp. 479-487.

[120] Tsallis, C. *Nonadditive entropy and nonextensive statistical mechanics-an overview after 20 years*, Brazilian Journal of Physics, 39 (2009), pp. 337-356.

[121] Van der Linden, W. *A Lognormal Model for Response Times on Test Items*, Journal of Educational and Behavioral Statistics, 31 (2006), pp. 181–204.

[122] Van Selst, M. and Jolicoeur, P. *A solution to the effect of the size on outlier elimination*, The Quarterly Journal of Experimental Psychology, 47 (1994), pp. 631-650. [CrossRef]

[123] Van Zandt, T. *How to fit a response time distribution ?* Psychonomic, Bulletin & Review, 7 (2000), pp. 424-465.

[124] Vannoorenberghe, P. *Estimation of finite mixture models by a credible EM algorithm*, Traitement du signal, 24 (2007), pp. 103-114.

[125] Wayman, J. *Multiple imputation for missing data : What is it and how can I use it*, Annual Meeting of the American Educational Research Association, 2 (2003).

[126] Wehrens, R., Putter, H. and Buydens, L. *The bootstrap : a tutorial*, Chemometrics and Intelligent Laboratory Systems, 54 (2000), pp. 35-52.

[127] Wei, G. and Tanner, M. *A Monte-Carlo implementation of the EM algorithm and the Poor's Man's data augmentation algorithm*, Journal of the American Statistical Association, 85 (1990), pp. 699-704. [CrossRef]

[128] West, T.B., Welch, B.K. and Galecki, T.A. *Linear Mixed Models, A Practical Guide Using Statistical Software* ; Chapman & Hall : New York, NY, USA, (2006).

[129] Yi, Q. and Panzarella, T. *Estimating Sample Size for Tests on Trends across Repeated Measurements with Missing Data Based on the Interaction Term in a Mixed Model*, Elsevier : Amsterdam, The Netherlands, 23 (2002), pp. 481-96.

[130] Yu, C.H. *Resampling methods : concepts, applications, and justification*, Practical Assessment, Research, and Evaluation, 8 (2019).

[131] Zakkour, A., Perret, C. and Slaoui, Y. *Stochastic Expectation Maximization Algorithm for Linear Mixed-Effects Model with Interactions in the Presence of Incomplete Data*, Entropy, 25 (2023).

[132] Zhang, X. *Tutorial : How to Generate Missing Data For Simulation Studies*, The Quantitative Methods for Psychology, 19 (2023), pp. 100-122.