Titre: Title:	Inférence du niveau d'expertise d'un auteur basée sur un corpus de textes avec une extension du Latent Dirichlet Allocation
Auteur: Author:	Mikaël Perreault
Date:	2021
Type:	Mémoire ou thèse / Dissertation or Thesis
Référence: Citation:	Perreault, M. (2021). Inférence du niveau d'expertise d'un auteur basée sur un corpus de textes avec une extension du Latent Dirichlet Allocation [Mémoire de maîtrise, Polytechnique Montréal]. PolyPublie. https://publications.polymtl.ca/6570/

URL de PolyPublie: PolyPublie URL:	https://publications.polymtl.ca/6570/
Directeurs de recherche: Advisors:	Michel C. Desmarais
Programme: Program:	Génie informatique

POLYTECHNIQUE MONTRÉAL
affiliée à l'Université de Montréal

Inférence du niveau d'expertise d'un auteur basée sur un corpus de textes avec une extension du Latent Dirichlet Allocation

MIKAËL PERREAULT
Département de génie informatique et génie logiciel

Mémoire présenté en vue de l'obtention du diplôme de *Maîtrise ès sciences appliquées*
Génie informatique

Avril 2021

POLYTECHNIQUE MONTRÉAL
affiliée à l'Université de Montréal

Ce mémoire intitulé :

Inférence du niveau d'expertise d'un auteur basée sur un corpus de textes avec une extension du Latent Dirichlet Allocation

présenté par **Mikaël PERREAULT**
en vue de l'obtention du diplôme de *Maîtrise ès sciences appliquées*
a été dûment accepté par le jury d'examen constitué de :

Guillaume-Alexandre BILODEAU, président
Michel DESMARAIS, membre et directeur de recherche
Foutse KHOMH, membre

DÉDICACE

Aux histoires qui ne seront jamais racontées

REMERCIEMENTS

Je tiens d'abord et avant tout à remercier mon directeur de recherche Michel Desmarais qui a su m'accompagner et m'encourager avec brio tout au long de cette maîtrise. Sans toutes nos véhémentes discussions sur des distributions de fréquences, j'aurais assurément moins été poussé à produire un travail de qualité et je lui en remercie. Je le remercie aussi pour sa remarquable patience et la générosité de son temps lorsque je ne savais plus de quel côté ramer. Je remercie aussi mon père que j'ai eu la chance de côtoyer pendant la première moitié de cette maîtrise avant de prendre mon envol, je ne pourrais avoir eu un meilleur colocataire. À ma mère, bien sûr, pour avoir été une oreille précieuse lorsque je tempêtais et pour m'avoir donné la chance de me rendre où j'en suis aujourd'hui. À Ilitea, qui m'a relu maintes fois et supporté vaillamment avec ses conseils judicieux. À la Realm, pour notre myriade d'expéditions dans un autre monde. À Mathieu pour avoir été une révélation pendant ma maîtrise. Au reste de ma famille et de mes amis qui m'ont épaulé. À mes collègues de laboratoire qui m'ont aidé et apporté une perspective différente sur mes travaux. Merci à vous.

RÉSUMÉ

L'inférence de l'expertise est une tâche fortement prisée dans le domaine de la modélisation textuelle. Plusieurs applications en découlent, que ce soit dans les sphères de l'éducation, de la revue d'articles scientifiques, de l'informatique, de la traduction ou de la gestion d'entreprise. Par exemple, si on possédait une connaissance accrue de l'expertise des auteurs d'articles scientifiques, il serait possible d'attribuer des réviseurs plus appropriés à des articles lors des conférences. Or, les méthodes actuelles pour inférer l'expertise des auteurs ne se reposent que sur des co-occurrences entre une requête spécifiée d'avance et les données textuelles de ces auteurs. Ces approches sont très limitantes, puisqu'elles permettent seulement de déterminer la puissance du lien entre un expert potentiel et une requête connue plutôt que de dégager une compréhension intelligente du niveau d'expertise général sur plusieurs sujets. De plus, des algorithmes de modélisation textuelle existent pour dégager les sujets présents dans un corpus donné. Ces sujets sont toutefois unidimensionnels, en ce sens qu'ils ne sont pas caractérisés par un niveau de profondeur.

Dans cet ordre d'idées, le présent mémoire propose une méthode d'inférence du niveau d'expertise des auteurs en se basant uniquement sur leurs données textuelles. Cette méthode d'inférence est une extension du Latent Dirichlet Allocation (LDA) et permet, en plus de segmenter un corpus en différents sujets latents, d'octroyer un niveau d'expertise pour chaque auteur à ces sujets. La technique se base aussi sur un ordonnancement du vocabulaire selon lequel les experts sont capables d'utiliser les mots les plus complexes contrairement aux novices.

La première question de recherche est la suivante : quelles sont les conditions opérationnelles du modèle LDA classique et dans quelle mesure l'hypothèse de génération des données de ce modèle est-elle conforme aux lois statistiques du langage ? Dans le but d'établir un *ground truth* solide propre aux modèles statistiques génératifs, la première contribution du mémoire consiste en la création de notre propre cadre de validation. On présente ce cadre dans lequel il est possible de générer un ensemble de données synthétiques à partir de certains paramètres et de déterminer la performance d'inférence de ceux-ci par le modèle à l'étude. Divers problèmes résultent de ce processus, tel que l'alignement des sujets latents, et des techniques pour les contourner sont détaillées. Ce cadre de validation a été utilisé pour analyser profondément le modèle LDA classique et deux principales contributions en découlent. D'une part, il a été déterminé que le *Collapsed Gibbs Sampling* est plus performant que l'inférence variationnelle en général. De plus, il a été statué que les α et β qui régissent la génération

de données synthétiques devaient être inférieurs à 1 pour assurer les bonnes performances de l'inférence. D'autre part, il a été trouvé que la combinaison d'hyperparamètres de Dirichlet $\alpha = 0.7$ et $\beta = 0.5$ est à prioriser pour la génération en vue d'obtenir un corpus qui fait le meilleur compromis entre une bonne performance de l'inférence de ces paramètres et une similitude marquée avec les lois statistiques du langage. Cette configuration constitue une base de référence pour les performances liées à un modèle sans inclusion du niveau d'expertise et on s'en sert à des fins de comparaison avec les performances de la nouvelle méthode qui considère ce niveau d'expertise.

La seconde question de recherche traitée est : comment peut-on faire interagir les lois statistiques du langage dans l'infrastructure LDA afin de déterminer l'expertise des auteurs propre à un sujet donné ? De ce fait, une autre contribution du mémoire réside dans la modélisation du niveau d'expertise à partir des lois statistiques du langage et de l'application de cette modélisation au sein d'une infrastructure LDA étendue qui infère non seulement les sujets d'un corpus textuel, mais aussi les expertises des auteurs liées à ces derniers. Il est conclu que plus le paramètre exponentiel d'une loi de Mandelbrot est faible et plus le niveau d'expertise est élevé. Ensuite, plusieurs hypothèses sont émises quant à la technique optimale pour inférer les fréquences de mots par auteur et par sujet dans l'infrastructure de LDA, ce qui constitue l'élément vital pour la détermination du niveau d'expertise. Par le biais de notre cadre de validation, il est prouvé qu'une méthode impliquant la pondération de la matrice de fréquences des mots par auteur avec les distributions de mots par sujet trouvées par LDA et qui sont communes à tous les auteurs est la plus appropriée pour inférer le paramètre d'expertise. La généralisation de la bonne performance de cette méthode a été explorée par le biais d'expériences concernant plusieurs situations d'expertise distinctes. Il a aussi été montré que la performance d'inférence (divergence KL entre les distributions générées et inférées) des distributions classiques de LDA obtenues avec cette nouvelle méthode est similaire à la performance de référence obtenue précédemment pour un modèle excluant l'expertise.

Or, bien que cette modélisation donne des résultats encourageants, il est important de réaliser qu'elle constitue les balbutiements du raffinement de LDA pour y inclure une notion d'expertise. Au terme de diverses analyses, il a été conclu que l'ordre de technicalité supposé des mots et que la dépendance au nombre de sujets latents spécifiés, à la taille du vocabulaire, au nombre de documents par auteur ainsi qu'au niveau d'uniformité de la matrice de sujets par auteur étaient tous des facteurs importants à considérer si on voulait amener cette méthode à un état pratique.

ABSTRACT

Expertise inference is a highly valued task in the field of textual modeling. It has many applications in the fields of education, scientific article review, computer science, translation and business management. For example, if we had a better knowledge of the expertise of the authors of scientific articles, it would be possible to assign more appropriate reviewers to articles at conferences. However, current methods for inferring the expertise of authors rely only on co-occurrences between a pre-specified query and the textual data of these authors. These approaches are very limiting, since they only allow to determine the strength of the link between a potential expert and a known query rather than to provide an intelligent understanding of the general level of expertise on several topics. Moreover, textual modeling algorithms exist to identify the topics present in a given corpus. However, these topics are one-dimensional, in the sense that they are not characterized by a level of depth.

In this vein, this paper proposes a method for inferring the level of expertise of authors based solely on their textual data. This inference method is an extension of the Latent Dirichlet Allocation (LDA) and allows, in addition to segmenting a corpus into different latent topics, to assign a level of expertise for each author to these topics. The technique is also based on a vocabulary ordering according to which experts are able to use the most complex words while novices are not.

The first research question is: What are the operational conditions of the classical LDA model and how well does the data generation assumption of this model conform to the statistical laws of language? In order to establish a robust ground truth for generative statistical models, the first contribution of the paper is the creation of our own validation framework. This framework is presented in which it is possible to generate a synthetic dataset from certain parameters and to determine the inference performance of these parameters by the model under study. There are various problems resulting from this process, such as the alignment of latent subjects, and techniques to circumvent them are detailed. This validation framework has been used to deeply analyze the classical LDA model and two main contributions result from it. On the one hand, it was determined that *Collapsed Gibbs Sampling* is more efficient than variational inference in general. Moreover, it was ruled that the α and β that govern the generation of synthetic data must be less than 1 to ensure good inference performance. On the other hand, it was found that the combination of Dirichlet hyperparameters $\alpha = 0.7$ and $\beta = 0.5$ is to be prioritized for generation in order to obtain a corpus that makes the best

compromise between good inference performance of these parameters and a strong similarity with the statistical laws of language. This configuration constitutes a reference base for the performances linked to a model without including the level of expertise and is used for comparison with the performances of the new method which considers this level of expertise.

The second research question addressed is: How can statistical language laws be interacted with the LDA infrastructure to determine subject-specific author expertise? Another contribution of the paper is the modeling of the level of expertise from statistical laws of language and the application of this modeling within an extended LDA infrastructure that infers not only the topics of a textual corpus, but also the expertise of the authors related to them. It is concluded that the lower the exponential parameter of a Mandelbrot law, the higher the level of expertise. Next, several hypotheses are put forward to decide the optimal technique for inferring word frequencies per author and per subject in the LDA infrastructure, which is the vital element for determining the level of expertise. Through our validation framework, it is shown that a method involving weighting the per-author word frequency matrix with the per-subject word distributions found by LDA that are common to all authors is most appropriate for inferring the expertise parameter. The generalization of the good performance of this method has been explored through experiments concerning several distinct expertise situations. It was also shown that the inference performance (KL divergence between the generated and inferred distributions) of the classical LDA distributions obtained with this new method is similar to the baseline performance obtained previously for a model excluding expertise.

Although this modeling is yielding encouraging results, it is important to realize that it is the first steps in the refinement of LDA to include a notion of expertise. After various analyses, it was concluded that the assumed order of technicality of the words and the dependence on the number of latent subjects specified, the size of the vocabulary, the number of documents per author and the level of uniformity of the matrix of subjects per author were all important factors to consider if this method was to be brought to a practical state.

TABLE DES MATIÈRES

LISTE DES TABLEAUX

LISTE DES FIGURES

LISTE DES ABRÉVIATIONS

APT	Author-Persona-Topic
AT	Author-Topic
CGS	Collapsed Gibbs Sampling
EMV	Estimation par Maximum de Vraisemblance
KL	Kullback-Leibler
LDA	Latent Dirichlet Allocation
LFP	Lexical Frequency Profile
MCMC	Markov Chain Monte Carlo
MCNL	Moindres Carrés Non-Linéaires
NIPS	Neural Information Processing Systems conference
RMSE	Root Mean Square Error
TREC	Text REtrieval Conference
UWL	University Word List

LISTE DES VARIABLES

a	Identifiant des auteurs
α	Hyperparamètre de Dirichlet contrôlant la distribution de sujets par document
β	Hyperparamètre de Dirichlet contrôlant la distribution de mots par sujet
B	Notation d'une distribution bêta
c	Paramètre de Mandelbrot (exposant)
C_{DK}	Matrice qui compte le nombre d'occurences qu'un document d est assigné au sujet k
C_{WK}	Matrice qui compte le nombre d'occurrences qu'un mot w est assigné à un sujet k
d	Identifiant des documents
$D\cos$	Métrique de distance cosinus
Dir	Notation d'une distribution de Dirichlet
D_{KL} (KL)	Métrique de divergence KL
D_r	Métrique de distance de corrélation
F	Matrice de fréquences de mots utilisés par auteur
$F^{(K)}$	Matrice des fréquences de mots par auteur et par sujet
γ	Matrice des paramètres d'expertise
k	Identifiant des sujets
Mand	Notation d'une distribution de Mandelbrot
N_a	Nombre d'auteurs
N_D	Nombre de documents
N_{Da}	Nombre de documents par auteur
N_E	Nombre d'exécutions du modèle
N_K	Nombre de sujets
N_V	Nombre de mots dans le vocabulaire
N_W	Nombre de mots uniques dans le corpus
N_{WD}	Nombre de mots par documents
ϕ	Distribution de mots par sujet
θ	Distribution de sujets par document
W	Matrice de la fréquence des mots dans chaque document
w	Identifiant des mots

X	Matrice représentant le corpus où chaque $X(d, w)$ représente l'identifiant du mot w dans le document d
Z	Matrice d'assignation d'un sujet à chaque mot du corpus

CHAPITRE 1 INTRODUCTION

La tâche d'inférer le niveau d'expertise est primordiale dans le monde de la modélisation textuelle. Que ce soit dans le domaine de l'écriture d'articles scientifiques, de l'éducation ou de la gestion d'entreprise, une grande valeur est accordée à l'identification du niveau d'expertise des individus ou du niveau de technicalité des données. Par exemple, si on veut choisir un individu pour une tâche selon son niveau d'expertise, on doit d'abord procéder à l'inférence de cette expertise à partir des données liées à celui-ci. Dans le cadre du présent mémoire, le modèle théorique d'inférence de l'expertise que nous développerons se servira de l'écriture d'articles scientifiques comme domaine d'application. Or, nos conclusions sont applicables à tous les domaines impliquant des données textuelles. Dans le but d'inférer le niveau d'expertise des auteurs d'articles scientifiques, on pourrait se baser sur la qualité des citations référentes et référées des articles écrits par l'auteur étudié ou par le niveau de technicalité de ses textes. C'est cette deuxième approche qui sera explorée. On devra donc développer un algorithme qui étudie les textes d'un auteur et qui infère le niveau d'expertise qui en découle.

1.1 Définitions et concepts de base

Afin de répondre à nos questions de recherche, des techniques liées au traitement de la langue naturelle seront utilisées.

Le traitement de la langue naturelle est un champ d'étude à la croisée de la linguistique et de l'informatique. Les chercheurs issus de ce domaine tirent profit de l'intelligence artificielle pour analyser un nombre important de données textuelles dans le but d'obtenir une compréhension plus profonde du langage : son utilisation, sa sémantique, sa traduction ainsi que ses nuances subtiles. Dans le cadre du mémoire, les lois statistiques du langage et des techniques provenant du traitement de la langue naturelle seront étudiées. En effet, la principale méthode qu'on utilisera pour inférer le niveau d'expertise des auteurs est le Latent Dirichlet Allocation (LDA).

Le Latent Dirichlet Allocation est un algorithme d'apprentissage automatique et plus spécifiquement d'apprentissage non supervisé. Il s'agit d'un modèle sujet (*topic model*) génératif. Définissons ces concepts.

L'apprentissage automatique est caractérisé par un système informatique qui peut performer la tâche d'apprendre les règles et la logique qui expliquent le comportement d'un jeu de données sans être explicitement programmé pour le faire. Les algorithmes qui en découlent se basent sur l'inférence statistique afin de paramétrer un modèle qui capture ce comportement. Habituellement, les données sont étiquetées, c'est-à-dire que l'on connaît directement la classe dont chaque exemple fait partie ou la valeur cible de chaque amalgame d'attributs. Lorsqu'on se trouve dans un cas comme celui-ci, cette sous-catégorie d'apprentissage automatique est nommée **l'apprentissage supervisé**. Or, pour ce qui est de notre cas, on ne possède pas les étiquettes des données, car le but de LDA est justement de déterminer celles-ci à partir des caractéristiques inhérentes aux exemples et non de leur étiquette. Quand on fait face à des exemples non étiquetés, la sous-catégorie d'apprentissage automatique connexe est **l'apprentissage non supervisé**. En apprentissage non supervisé, on se base davantage sur un regroupement de comportements similaires et une organisation intrinsèque pour identifier des facteurs latents qui composeront les étiquettes inférées.

Un modèle sujet est un modèle d'apprentissage automatique utilisé pour l'inférence des sujets que l'on peut observer dans un corpus donné. On caractérise souvent ces sujets de « latents » ou « abstraits », puisque ceux-ci ne sont pas déterminés d'avance et ne possèdent pas d'identification unique. Par exemple, les sujets trouvés par LDA ne sont qu'un regroupement de mots qui appartiennent à une classe spécifique.

Un modèle génératif est un modèle statistique qui évalue la probabilité conjointe d'une variable observée et une variable cible. Plus précisément, il cherche à estimer $P(X, Y)$. Ceux-ci sont souvent comparés aux modèles **discriminatifs** qui évaluent plutôt $P(Y|X)$. Étant donné que les modèles génératifs se reposent sur l'estimation d'une probabilité conjointe, ils essaient de trouver la combinaison de paramètres qui pourrait expliquer une génération de données basée sur $P(X, Y)$.

Le modèle LDA sera abordé rapidement au chapitre 2, étudié en profondeur au chapitre 3 et étendu au chapitre 4. On aura donc l'occasion de revenir sur ces concepts.

1.2 Éléments de la problématique

1.2.1 Motivation

Le domaine de la segmentation du texte en sujets a été profondément étudié au cours des 20 dernières années et le fruit de ces recherches est facilement observable. Par exemple, en

étant en mesure de regrouper les articles d'un journal numérique en fonction de leur thème, la navigation du lecteur s'en voit facilitée. On utilise aussi la modélisation par sujets pour recommander des items en analysant leur description. Par exemple, on peut appliquer un modèle de segmentation des sujets sur des résumés de films afin de recommander des films similaires à un utilisateur.

Pour ces exemples, on voit que le niveau d'expertise des items ou des utilisateurs n'est pas évalué. Cependant, un grand nombre d'applications existe où il est névralgique de définir une hiérarchie de complexité dans les recommandations. Prenons l'exemple d'un étudiant à l'école secondaire qui doit produire un travail de recherche sur les éclairs. Cet étudiant possède des connaissances de base en électricité, mais ignore tous les détails mathématiques qui caractérisent ce phénomène. Actuellement, Google n'offre pas de filtre de technicalité sur les pages qu'il recommanderait à l'étudiant. Celui-ci pourrait autant tomber sur une page destinée à expliquer la science pour enfants qu'un article universitaire très technique. Toutefois, si cet étudiant procède à une recherche Google classique, il trouvera probablement la page de Wikipédia d'abord, puisque c'est elle qui est la plus populaire. On peut alors se demander si cette recommandation est appropriée pour son niveau d'expertise. Aussi, si un étudiant en génie électrique effectue cette même recherche, comment ces recommandations devraient-elles être modifiées en fonction de son niveau d'expertise plus élevé ? On comprend donc que ce n'est pas seulement la segmentation des sujets qui est pertinente : une hiérarchie plus profonde et complexe peut être dégagée de ces données.

Prenons un autre cas de figure ; celui de la révision d'articles scientifiques en l'occurrence. Dans le contexte d'une conférence scientifique, les réviseurs que l'on sélectionne pour la révision d'un article scientifique sont habituellement choisis en votant pour les articles qui les intéressent. Or, il serait pertinent de dégager l'expertise des réviseurs en se basant sur leur corpus d'articles dans le but de les répartir de manière plus appropriée. Étant donné que le concept d'expertise est intuitif dans le domaine des auteurs d'articles scientifiques, celui-ci fera office de contexte d'application pour le présent travail. Par ailleurs, l'expertise des auteurs est pleine de sens seulement si elle s'inscrit dans un sujet donné. C'est pour cette raison que nous avons décidé d'ajouter une couche de complexité à un modèle de segmentation des sujets pour non seulement inférer les différents sujets d'un corpus, mais aussi pour inférer le niveau d'expertise des auteurs dans ces sujets.

Il est important de noter que, bien que le travail se concentre sur un cas où des auteurs écrivent des articles, les résultats obtenus peuvent être généralisés à tout autre domaine im-

pliquant une compréhension du niveau d'expertise basée sur des données textuelles. On peut penser aux domaines de l'éducation (où on propose des tuteurs ou un cheminement de cours), de l'informatique (où on peut attribuer des révisions de codes plus ou moins techniques en fonction du niveau d'expertise des programmeurs), de la traduction (où il est pertinent de déduire le niveau de technicalité d'un texte à traduire pour recommander des mots plus justes) ou au domaine de la gestion d'entreprise (que ce soit pour l'analyse des curriculum vitae ou la répartition de tâches).

1.2.2 Concepts avancés

Afin de comprendre les objectifs de recherche qui suivent, il est nécessaire d'introduire certains concepts.

Quand on fait référence à **l'expertise** d'un auteur, on fait référence à son niveau de maîtrise du **vocabulaire** propre à un **sujet**. Le vocabulaire est un ensemble de mots commun à tous les sujets étudiés au sein d'un corpus. Notons que l'on ordonnera son vocabulaire du mot le moins technique au plus technique et que cet ordre est différent pour chaque sujet. Un sujet est un regroupement sémantique donné en sortie de LDA qui segmente un corpus selon ses thèmes sous-jacents (par exemple, un corpus traitant de sciences naturelles peut être segmenté en physique, chimie et biologie en tant que sujets). Ensuite, notre hypothèse d'expertise se repose sur les **lois statistiques du langage**. Ces lois traitent de la fréquence théorique selon laquelle on devrait observer les occurrences de mots selon leur rang et cette fréquence est modulée selon le niveau d'expertise. C'est pour cette raison que la détermination de la **fréquence de mots en fonction de son rang par auteur et par sujet** sera névralgique dans le cadre du mémoire. De plus, lorsqu'on discute du modèle **LDA classique**, on fait référence au modèle LDA sans modifications ultérieures, c'est-à-dire, tel que présenté dans l'article original de Blei *et al.* (2003). Lorsqu'on discute de **LDA étendu**, on fait plutôt référence au modèle LDA qui comporte notre technique d'inférence de l'expertise.

Le modèle LDA classique est caractérisé par deux **hyperparamètres** importants qui seront traités prochainement mais que l'on introduit ici. Le premier hyperparamètre est α et il contrôle les distributions de sujets pour chaque article. Le second paramètre est β et il contrôle les distributions de mots par sujet. Finalement, on fera souvent référence à des **distributions générées et inférées**. Les distributions générées sont les distributions propres aux hyperparamètres α et β qui sont issues du processus de génération de données synthétiques. Les distributions inférées sont des approximations des distributions générées qui sont le résultat d'une technique d'inférence opérée par LDA.

1.3 Objectifs de recherche

L'objectif du mémoire consiste à développer une méthode qui permet d'inférer le niveau d'expertise des auteurs en se basant sur un corpus de textes écrits par ceux-ci. Cet objectif se décline en deux sous-objectifs. Ces deux sous-objectifs constituent les deux principales questions de recherche du mémoire et, par conséquent, ses deux principales contributions. D'une part, la première question de recherche est la suivante : quelles sont les conditions opérationnelles du modèle LDA classique et dans quelle mesure l'hypothèse de génération des données de ce modèle est-elle conforme aux lois statistiques du langage ? Afin de répondre à cette question, on devra procéder à l'élaboration d'un cadre de validation pour le modèle LDA classique afin de déterminer un *ground truth*. Ce cadre de validation implique une génération de données synthétiques et la prédiction de ces dernières par les méthodes d'inférence du modèle LDA classique. L'utilisation de données synthétiques permet avant tout de valider le modèle dans un cadre contrôlé où on connaît les paramètres latents (non observables). De ce fait, il sera possible d'établir les balises de validité du modèle LDA classique, c'est-à-dire les conditions dans lesquelles ce dernier est opérationnel. Une fois qu'on aura vérifié la bonne performance de notre cadre de validation sur un modèle connu, on en profitera pour explorer davantage le modèle LDA classique dans le but de, entre autres, établir une base de comparaison des performances auxquelles on devrait s'attendre lorsque viendra le temps de tester notre méthode d'inférence de l'expertise. Ici, on cherchera à accomplir les 3 sous-tâches suivantes :

- Déterminer la performance des méthodes d'inférence des distributions de LDA classique en fonction des hyperparamètres utilisés pour la génération des données synthétiques.
- Vérifier dans quelle mesure l'hypothèse de génération des données de LDA classique respecte les lois statistiques de la langue naturelle.
- Trouver quelle est la combinaison d'hyperparamètres pour la génération qui mène à la création du corpus qui fait le meilleur compromis entre réalisme et bonne performance de l'inférence.

D'autre part, la seconde question de recherche est la suivante : comment peut-on faire interagir les lois statistiques du langage dans l'infrastructure LDA afin de déterminer l'expertise des auteurs propre à un sujet donné ? On présentera donc notre méthode d'inférence du niveau d'expertise et on validera celle-ci à l'aide du même cadre de validation précédemment développé. Voici les sous-tâches associées à cet objectif :

- Élaborer l'analogie entre l'expertise et les lois statistiques du langage.
- Développer le nouveau modèle et le tester avec le cadre de validation.

— Identification des limitations quant à l'utilisation du nouveau modèle et application d'une analyse de sensibilité aux hyperparamètres.

1.4 Plan du mémoire

Ce mémoire commence par une revue de littérature qui traite des statistiques fondamentales d'un langage, des approches liées à la recherche d'experts ainsi que des approches liées à la modélisation du contenu.

Ensuite, le chapitre 3, portant sur LDA et le cadre de validation, présente d'abord une description sommaire du modèle LDA classique ainsi que du cadre de validation avec données synthétiques. On poursuit par la validation du modèle LDA classique avec ce cadre et on discute des spécificités liées à cette infrastructure de génération-inférence tel que l'alignement des sujets latents. On conclut ce chapitre par deux analyses du modèle LDA classique : l'une portant sur la performance des méthodes d'inférence en fonction des hyperparamètres de Dirichlet et l'autre concernant l'évaluation de la similitude des fréquences de mots générées par LDA avec les lois statistiques du langage. On conclut le chapitre 3 par la présentation du corpus faisant le meilleur compromis entre réalisme et bonne performance d'inférence que LDA peut générer ainsi qu'une présentation des statistiques complètes de l'inférence de ses paramètres.

Finalement, c'est au chapitre 4 que l'on élabore notre modèle d'inférence de l'expertise des auteurs. D'abord, on présente notre conception de l'expertise avec une analogie avec le paramètre de Mandelbrot en plus des méthodes employées pour inférer ce dernier. On poursuit avec les détails du modèle d'expertise et des méthodes explorées afin d'identifier les fréquences de mots par auteur et par sujet. Ces méthodes sont testées par notre cadre de validation et les principaux résultats sont rapportés. Après avoir présenté les limitations de la méthode ainsi qu'une analyse de sensibilité des hyperparamètres, on vient enfin conclure sur les résultats principaux du mémoire.

CHAPITRE 2 REVUE DE LITTÉRATURE

La revue de littérature sera séparée en trois parties. D'abord, les statistiques derrière l'utilisation des mots dans un texte ainsi que la théorie sur l'apprentissage d'un langage en lien avec le niveau de maîtrise de celui-ci seront traitées. Ensuite, les approches propres à la recherche d'experts seront abordées. Enfin, on survolera brièvement la technique la plus employée pour modéliser de façon latente les sujets abordés par un texte et on traitera des extensions de ce modèle permettant d'évaluer l'expertise des auteurs. L'objectif de cette revue de littérature est de montrer les limitations de l'approche de la recherche d'experts en lien avec la tâche d'inférer une expertise générale. On émet donc l'hypothèse qu'un modèle sujet raffiné par la fréquence d'utilisation des mots pourrait permettre une compréhension plus robuste de l'expertise.

2.1 Les statistiques fondamentales de l'utilisation d'un langage

L'objectif de cette section est de revoir les fondements mathématiques derrière l'utilisation des mots d'un langage. D'une part, nous verrons que la fréquence d'utilisation des mots peut être caractérisée mathématiquement de façon assez précise. D'autre part, nous discernerons les différences dans les fréquences de mots utilisés par les novices et les experts.

2.1.1 Lois statistiques sur la fréquence d'utilisation des mots

Le langage humain est un concept très complexe qui comporte énormément de subtilités. Que ce soit l'origine même des dialectes qui diffère ou les objectifs de communication qui sont variés, la portée du langage est vaste. De ce fait, il est étonnant de penser qu'une loi statistique simple peut caractériser la fréquence d'utilisation des mots dans une production écrite. C'est ce que Zipf (1949) tente d'identifier avec son principe intitulé *Principle of Least Effort*. Ce dernier stipule que la fréquence f d'un mot est inversement proportionnelle à son rang R :

$$f(R) \propto \frac{1}{R^{\alpha}}$$

où $\alpha \approx 1$ mais strictement supérieur à 1. Cette équation, que l'on appelle communément la Loi de Zipf, signifie que le mot le plus fréquent aura une fréquence d'apparition proportionnelle à 1, le second mot le plus fréquent aura une fréquence d'apparition proportionnelle à $\frac{1}{2^{\alpha}}$, $\frac{1}{3^{\alpha}}$ pour le troisième et ainsi de suite. On exprime cette loi sous la forme proportionnelle, puisque la fréquence réelle est dépendante du nombre total de mots dans le texte. Par ailleurs,

cette relation est universelle parmi les différentes langues, comme le démontre Calude et Pagel (2011). Ces chercheurs ont montré qu'il existe une corrélation $r = 0.73$ entre les lissages de Zipf de textes provenant de 17 langues issues de 6 familles de langage.

Bien que la loi de Zipf capture adéquatement le comportement général de la fréquence des mots, elle ne comporte pas beaucoup de flexibilité. Mandelbrot (1953) propose la modification suivante à la loi de Zipf :

$$f(R) \propto \frac{1}{(R+\beta)^{\alpha}}$$

où $\alpha \approx 1$ et $\beta \approx 2.7$ typiquement. Il est prouvé qu'on obtient un meilleur lissage en général avec cette équation. Les chercheurs de linguistique statistique ont nommé cette équation la loi de Quasi-Zipf (*near-Zipfian Law)*.

Enfin, Simon (1955) développe un modèle stochastique reprenant la base des travaux de Zipf et Mandelbrot. Ce modèle stochastique explique la logique derrière la loi de Zipf. Ce modèle se base sur deux hypothèses. D'une part, les mots fréquents qui ont été précédemment écrits auront davantage tendance à être réutilisés dans le futur. D'autre part, il existe une probabilité α qu'un nouveau mot soit employé pour la première fois dans le texte (ce α est différent du α utilisé dans l'équation de Mandelbrot). Ceci mène à la relation de fréquence suivante :

$$f(R) \propto \frac{\Gamma(R)\Gamma(\rho+1)}{\Gamma(R+\rho+1)}$$

où $\rho = \frac{1}{1-\alpha}$. Grâce à cette équation, Simon parvient à estimer correctement la fréquence des mots dans le texte *Ulysses* de James Joyce, une oeuvre classique reconnue pour sa richesse lexicale. Pour ce faire, il estime le paramètre α par le ratio du nombre de mots différents par le nombre de mots total dans le texte ($\alpha = 0.115$ pour *Ulysses*), ce qui semble une hypothèse raisonnable.

2.1.2 Différences d'utilisation du langage entre un novice et un expert

Maintenant que nous savons que des lois statistiques régissent l'utilisation d'un langage, il est important de déterminer si celles-ci sont modelées par l'expertise d'un individu. Laufer et Nation (1995) mènent une étude afin d'élucider cette question. Pour ce faire, ils instaurent une mesure de richesse lexicale : le *Lexical Frequency Profile* (LFP). Le LFP est un profil linguistique duquel on peut observer la proportion d'utilisation de mots appartenant à différents niveaux de langage. Ils définissent quatre niveaux de langage, soit les 1000 mots les plus communs de la langue anglaise (niveau 1), les seconds 1000 mots les plus communs (niveau

2), les mots provenant de la *University Word List* (UWL) de Xue et Nation (1984) (niveau 3) et les mots qui ne font pas partie des précédents groupes (niveau 4). La UWL est une liste de 836 mots qui ne sont pas présents dans les 2000 mots les plus fréquents et qui est constituée de termes souvent retrouvés dans des articles scientifiques ou textes académiques. Afin de mener à bien leur expérience, les chercheurs regroupent trois ensembles de personnes de niveau linguistique différent : novice, intermédiaire et avancé. Les individus composant ces trois groupes doivent créer deux productions écrites portant sur les deux mêmes sujets. De plus, ils doivent répondre à un test de vocabulaire. Le LFP est ensuite calculé pour chacun de ces groupes, un test ANOVA est performé pour déterminer si le LFP obtenu à chaque niveau de langage est différent pour tous les groupes, et la corrélation entre le LFP ainsi que le test de vocabulaire est obtenue.

Les résultats de cette expérience sont probants. D'abord, le test ANOVA montre que les LFP sont statistiquement différents parmi les groupes pour les niveaux de langage 1,3 et 4. Ensuite, les chercheurs constatent que le groupe novice possède, relativement parlant, le LFP le plus élevé pour le niveau 1 et les LFP les plus faibles pour les niveaux 3-4 et vice-versa pour le groupe avancé. Enfin, ils observent une corrélation positive entre le résultat du test de vocabulaire et le LFP des niveaux 3-4, une corrélation nulle avec le LFP du niveau 2 et une corrélation négative avec le LFP du niveau 1. Même si ces résultats semblent intuitifs, ils montrent que plus la connaissance d'un langage est acquise plus le locuteur a tendance à utiliser des mots rares et techniques.

En plus de la richesse du vocabulaire, on peut se demander si les expressions usuelles utilisées par les novices et les experts sont différentes. C'est aussi la question que se pose Cortes (2004) dans son papier. Elle procède à une analyse comparative d'un corpus regroupant des textes écrits par des étudiants de premier cycle universitaire (novices) ainsi que des articles provenant d'auteurs accomplis (experts). Les textes des novices et experts proviennent de deux domaines, soit l'histoire et la biologie. L'objectif de l'étude est de comparer les marqueurs de relation et les expressions usuelles pour les deux niveaux de connaissance et plusieurs différences sont observées.

Par exemple, les étudiants d'histoire utilisent beaucoup plus de marqueurs de relation temporels, tels que « *as the beginning of the* » et « *the end of the* », alors que les auteurs publiés se servent davantage des marqueurs de relation liés au contexte, comme « *at the turn of* » et « *in the course of* ». Aussi, les experts en biologie ont tendance à employer plus d'expressions statistiques et numérales que les étudiants, ceci témoignant d'un plus grand nombre

de références à la quantification et au traitement des résultats pour les textes d'experts. Finalement, pour les deux domaines, Cortes note une plus grande redondance dans les termes utilisés chez les étudiants, ce phénomène étant attendu selon elle, puisque les novices ont tendance à rester fixés sur des phrases qu'ils sont à l'aise d'utiliser.

Par ailleurs, la question de la variation du niveau d'expertise en fonction des particularités lexicales se pose aussi dans le contexte du langage artificiel. En effet, une analyse menée par Dakhel *et al.* (2021) a démontré une relation claire entre le niveau d'expertise des développeurs de code et les paramètres d'une loi de Zipf. Pour ce faire, les chercheurs ont décomposé un programme en un arbre syntaxique abstrait (*Abstract Syntactic Tree)* et les noeuds (majoritairement les noeuds terminaux soit les feuilles) ont été extraits. Les paramètres d'une distribution de Zipf ont été induits de la distribution de fréquence par le rang des noeuds. Les résultats démontrent qu'on peut ainsi distinguer les experts des novices du langage Python.

À la lumière de ces travaux, force est de constater que l'utilisation des mots peut être prédite par des lois statistiques simples. De plus, on remarque qu'il existe des différences d'utilisation du langage entre un novice et un expert. Que ce soit dans la richesse du vocabulaire ou par les divers thèmes qui ressortent des expressions usuelles employées, on peut émettre l'hypothèse qu'un expert soit identifiable en analysant ses particularités d'utilisation lexicale. Cette considération sera utile quand viendra le temps d'élaborer notre modèle d'inférence de l'expertise.

2.2 Les approches liées à la recherche d'experts

La recherche d'experts (*experts retrieval*) est un problème étudié dans la recherche d'informations (*information retrieval*) qui fait l'objet de nombreuses recherches depuis les années 1980. Certaines approches issues de ce sous-domaine ont été en mesure de générer des résultats intéressants pour ce qui est d'affecter un expert à une tâche. Or, dans le contexte du présent mémoire, soit celui d'inférer le niveau d'expertise d'un auteur basé sur un corpus de textes, ces méthodes comportent des limitations. Il est donc primordial d'étudier ces approches afin d'évaluer les avantages et les inconvénients de celles-ci pour notre travail.

Les données utilisées pour ces recherches proviennent du *Text REtrieval Conference* (TREC). Dans le cadre du *TREC Enterprise Track* de 2005, le W3C corpus avait été créé. Le W3C corpus contient des fichiers HTML de forums de courriels, de pages internet, de pages d'accueil des experts potentiels et autres. La taille du W3C corpus est de 330 037 documents. Lors

du *TREC Enterprise Track*, 50 requêtes avaient été générées pour 1092 experts potentiels et les experts appropriés pour chaque requête avaient été recueillis. Ces experts potentiels sont caractérisés par un identifiant unique, un nom ainsi qu'une adresse courriel. L'objectif des modèles suivants est de retrouver cette liste d'experts pour chaque requête.

2.2.1 Les modèles probabilistes génératifs directs

Les modèles génératifs directs estiment la force de l'association entre une requête et des experts potentiels par la co-occurrence de ces auteurs dans des documents qui utilisent des termes que l'on retrouve dans la requête à l'étude. On les distingue ici des modèles génératifs contenu (comme LDA), ces derniers permettant plutôt le regroupement du contenu selon différents sujets. Fang et Zhai (2007) élaborent le premier modèle génératif pour classifier les expertises des auteurs et les caractérisent sous deux variantes : le modèle génératif candidat et le modèle génératif requête.

Modèle génératif candidat

Pour ce qui est du modèle génératif candidat, on estime les associations entre une requête et des candidats basés sur la vraisemblance qu'un candidat ait été généré par une certaine requête. On dénote par q la requête pour laquelle on recherche des experts, par e les experts potentiels (ou candidats), par t les termes dans la requête et par d les documents associés aux auteurs. L'estimation de $P(e|q)$ nous informera sur la vraisemblance d'un expert selon une requête et il sera possible de classer ceux-ci selon cette probabilité. $P(e|q)$ peut être développée comme suit :

$$P(e|q) = \sum_{d} P(e|d,q)P(d|q)$$

où $P(d|q)$ mesure le degré de pertinence du document d par rapport à la requête q pour laquelle on recherche un expert. Si on applique le théorème de Bayes, on obtient :

$$P(d|q) = \frac{P(q|d)P(d)}{P(q)}$$

Ici, $P(d)$ est une distribution *a priori* sur les documents utilisés pour en favoriser certains (souvent supposée uniforme). La distribution $P(q)$ est la même pour tous les candidats et peut donc être omise lors du calcul des rangs des experts potentiels. Il est maintenant plus facile de calculer $P(q|d)$ par le biais d'un modèle de langage classique :

$$P(q|d) = \prod_{t \in q} P(t|d)^{n(t,q)}$$

Où $P(t|d)$ est la probabilité que le terme t apparaisse dans le document d et $n(t,q)$ représente le nombre de fois que le terme t survient dans la requête q.
Ainsi, en admettant l'indépendance conditionnelle de l'expert e et la requête q étant donné le document d, on a $P(e|d,q) = P(e|d)$. De cette manière, les formules ci-dessus expriment la probabilité d'un expert étant donné la requête par la sommation des probabilités d'un expert étant donné un document multiplié par la pertinence de ce document à la requête étudiée.

Modèle génératif requête

Suivant la même logique que pour le modèle candidat, le modèle génératif requête estime la force d'une association entre une requête et un expert potentiel basée sur la vraisemblance que cette requête soit générée par cet expert. On cherche ici aussi à estimer $P(e|q)$ et à classer les experts selon cette probabilité mais on procède différemment. En appliquant le théorème de Bayes sur $P(e|q)$, on obtient :

$$P(e|q) = \frac{P(q|e)P(e)}{P(q)}$$

Puisque la tâche est de classer la pertinence des experts selon la requête, on peut retirer $P(q)$ de l'équation ci-dessus. En effet, étant donné que la probabilité *a priori* de la requête est constante pour une requête donnée, ce facteur s'annule dans le calcul du classement. Par conséquent, l'objectif de ce modèle revient à estimer $P(q|e)P(e)$. D'abord, le fait d'inclure $P(e)$ dans le calcul est utile, puisqu'on peut venir ajouter des informations sur l'importance des experts potentiels par le biais d'une probabilité *a priori*. Pour ce qui est de la façon d'estimer $P(q|e)$, la réponse nous provient d'un article de Balog *et al.* (2006) dans lequel les auteurs développent deux modèles pour calculer $P(q|e)$. Seul le *Model 2* (tel qu'utilisé dans l'article) sera étudié, car c'est ce modèle qui est le plus intuitif et qui génère les résultats les plus robustes. On exprime donc $P(q|e)$ comme suit :

$$P(q|e) = \sum_d P(q|d,e)P(d|e) = \sum_d \left[\prod_{t \in q} P(t|d)^{n(t,q)} \right] P(d|e)$$

On suppose ici que les experts et les mots sont conditionnellement indépendants étant donné un document. Ainsi, on constate que le premier terme du membre droit décrit s'il est vrai que le document d est une preuve d'expertise sur q alors que le second terme représente la force de l'association entre le document d et l'expert e. Notons que $P(d|e)$ est estimée à partir d'un graphe de citations des documents.

2.2.2 Les modèles probabilistes discriminants

On introduit les modèles probabilistes discriminants, puisqu'ils comportent des avantages par rapport aux modèles génératifs, ces avantages venant au prix d'une complexité accrue du modèle. L'objectif du modèle génératif est d'évaluer la probabilité d'obtenir soit un expert (modèle génératif candidat) ou une requête (modèle génératif requête) étant donné les observations en supposant que ces observations soient générées soit par les requêtes, soit par les experts. Contrairement aux modèles génératifs, les modèles discriminants instaurent des paramètres pour calculer cette probabilité conditionnelle. Ces paramètres sont ensuite inférés grâce à des données d'entraînement. En d'autres mots, au lieu de spécifier l'élément qu'on veut prédire et les observations, on évalue plutôt directement si la probabilité conditionnelle d'une paire requête-expert est pertinente ou pas. Pour ce faire, Fang *et al.* (2010) implémentent un paramètre binaire de pertinence, dénoté r, afin d'élaborer leur modèle discriminant. Si r vaut 1, cela signifie que la paire requête-expert est pertinente et vice-versa. L'objectif de ce modèle est d'estimer la probabilité suivante :

$$P(r_{e,q} = 1|e, q) = \sum_{d} P(r_{q,d} = 1|q, d)P(r_{e,d} = 1|e, d)P(d)$$

Où le premier terme modélise la pertinence de l'association requête-document, le second terme modélise la pertinence de l'association expert-document et le troisième terme est une mesure d'importance des documents (généralement une simple distribution uniforme). Donc, si on détermine que l'association entre l'expert et le document est forte et que l'association entre ce même document et la requête étudiée l'est aussi, il est raisonnable de supposer que le lien entre l'expert et la requête étudiée est fort, d'où le caractère intuitif de cette approche, semblable au *Model 2* traité dans la section 2.2.1. Les deux premiers termes de l'équation ci-dessus peuvent être exprimés par des fonctions logistiques :

$$P(r_{q,d} = 1|q, d) = \sigma(\sum_{i=1}^{N_f} \alpha_i f_i(q, d))$$

$$P(r_{e,d} = 1|e, d) = \sigma(\sum_{j=1}^{N_g} \beta_g f_g(e, d))$$

où $\sigma(x) = \frac{1}{1+e^{-x}}$ est la fonction logistique classique, les α_i représentent les poids de chaque association requête-document $f_i(q, d)$ que l'on utilise comme *feature* dans l'entraînement, les β_j sont les poids de chaque *feature* expert-document $f_g(e, d)$. Les paramètres α et β sont trouvés en maximisant la log-vraisemblance des données d'entraînement. Comme il n'y a pas de solution analytique pour cette log-vraisemblance, la méthode numérique BGFS est utilisée

pour l'optimisation.

Le modèle discriminant comporte plusieurs avantages par rapport aux modèles génératifs. D'une part, on peut utiliser des *features* plus complexes, puisqu'on estime directement la probabilité d'une combinaison. D'autre part, la notion de pertinence est plus directe dans le modèle discriminant à cause de la paramétrisation mise en place. Aussi, l'erreur d'entraînement diminue à mesure que le nombre de données d'entraînement augmente. Ceci peut soit être causé par le caractère théorique de la paramétrisation ou par le fait que le modèle discriminant ne possède pas autant de restrictions que les modèles génératifs. Par exemple, les modèles discriminants ne supposent pas que la requête et l'expert soient conditionnellement indépendants pour un document donné. Toutefois, les modèles discriminants nécessitent beaucoup plus de données d'entraînement par rapport aux modèles génératifs, ce qui représente un important désavantage selon l'application.

2.2.3 Analyse de graphe

L'analyse de graphe évalue la force d'une association requête-expert par une étude des noeuds et relations provenant d'un graphe. Les noeuds incluent les experts potentiels, les documents qui leur sont associés, ainsi que différentes relations externes (dates, locations, évènements, réseaux sociaux) tandis que les relations sont des relations de pertinence. On retrouve dans l'article de Serdyukov *et al.* (2008) une description de l'intuition derrière les modèles graphiques pour la recherche d'experts. D'abord, on sélectionne un document ou un auteur au hasard. Ensuite, si un document a été lu à l'étape précédente, on peut soit consulter un auteur cité dans ce document ou consulter d'autres documents liés à celui-ci. Autrement, si un auteur avait été consulté à la première étape, on peut soit lire des documents de cet auteur ou choisir d'autres experts potentiels qui sont recommandés par l'auteur initial.

Dans cet ordre d'idée, une technique intitulée le *infinite random walk* est présentée dans le même article. Comme son nom l'indique, l'algorithme débute en choisissant un point de départ aléatoire dans le graphe d'experts. Ensuite, selon certaines probabilités, on peut se déplacer dans le graphique pour atteindre des noeuds de document ou d'expert. On peut répéter infiniment ce déplacement jusqu'à atteindre une convergence. Explicitement, on se

déplace dans le graphique selon les probabilités suivantes :

$$P_i(d) = \lambda P_J(d) + (1-\lambda) \sum_{e \to d} P(d|e) P_{i-1}(e)$$
$$P_i(e) = \lambda P_J(e) + (1-\lambda) \sum_{d \to e} P(e|d) P_{i-1}(d)$$
$$P_J(d) = P(d|q)$$
$$P_J(e) = \frac{cf(e, Top)}{|Top|}$$

Certains éléments sont à prendre en considération dans ce processus itératif. D'abord, on recueille les meilleurs documents (nombre déterminé arbitrairement) depuis un moteur de recherche. Cette banque de document est représentée par la variable Top. Aussi, on retrouve les probabilités $P_J(d)$ et $P_J(e)$. À chaque étape, celles-ci représentent la probabilité de sauter directement à un document ou un expert sans devoir passer par les liens usuels. On comprend donc que λ est la probabilité que, à chaque étape, l'utilisateur décide d'effectuer un saut plutôt que de continuer à suivre les liens usuels du graphique. Ceci a pour effet de concentrer la marche infinie sur les noeuds pertinents ; on observe donc que les experts qui sont situés à proximité des documents sont visités plus fréquemment au total. $P_J(D)$ est une mesure de pertinence du document alors que $P_J(e)$ est une mesure du niveau de pertinence de l'auteur donné par le moteur de recherche, puisque $cf(e, Top)$ est le nombre de meilleurs documents dans lequel l'expert e est mentionné et $|Top|$ est le nombre total de meilleurs documents recueillis. Ce processus itératif possède les caractéristiques d'une chaîne de Markov. Par conséquent, les distributions convergeront vers une valeur stationnaire et donc, on peut évaluer la probabilité que le candidat e soit un expert par $P_i(e)$ où i devient très grand, soit $P_\infty(e)$.

Il existe beaucoup d'autres instances de modèle graphique pour la recherche d'expertise, mais on présente ici le plus simple. Il est important de noter qu'il est possible de modéliser des liens plus complexes entre les auteurs, par exemple, avec l'inclusion du h-Index. Le h-index est une mesure de renommée des auteurs scientifiques. En pratique, il s'agit d'une quantification de l'importance du nombre de citations par rapport aux publications d'un auteur. Toutefois, en dépit du fait que cette métrique est utile pour estimer une distribution *a priori* sur P_e, il est difficile de l'utiliser si la tâche est de recueillir un expert alors que la requête est spécifiée explicitement.

2.2.4 Limitations des approches liées à la recherche d'experts

En résumé, les méthodes issues de la recherche d'experts sont performantes si la tâche est d'identifier un expert pertinent au premier rang alors que la requête est connue. Il devient difficile de trouver tous les experts pertinents et encore plus difficile de classer ceux-ci parmi tous les experts non pertinents. De plus, ces techniques ne résolvent pas directement le problème d'identification de l'expertise, mais elles répondent plutôt à la question suivante : quelle est la force du lien entre un expert potentiel et une requête connue ? Ce raisonnement comporte des défauts. Par exemple, si on prend un candidat qui comporte un grand nombre de co-occurrences avec la requête, mais que ces co-occurrences découlent d'une connaissance en surface de cette requête, alors ce candidat pourrait très bien ne pas être l'expert que l'on recherche.

Effectivement, il est important de noter que les modèles présentés jusqu'à maintenant n'analysent le langage des experts potentiels. Il s'agit plutôt de méthodes de recherche d'informations qui permettent d'identifier les co-occurrences entre une mention d'un auteur et des termes se retrouvant dans leurs documents qui concordent avec les termes recherchés dans une requête spécifiée d'avance. Il devient donc difficile de généraliser les résultats obtenus avec ces modèles puisqu'on ne possède pas la notion de profondeur dans ces co-occurrences. En effet, il serait plus intéressant de dégager les sujets pertinents d'un corpus et d'inférer un niveau d'expertise global des auteurs sur ces mêmes sujets. On émet l'hypothèse qu'avec l'introduction des modèles utilisant des variables latentes, présentés dans la section 2.3, des indications plus générales, robustes et automatiques sur l'expertise d'un auteur pourraient être obtenues.

2.3 Les approches liées à la modélisation du contenu

Contrairement aux modèles provenant de la théorie sur la recherche d'experts, les approches contenu utilisent des variables latentes pour la modélisation. Nous verrons d'abord les bases du *Latent Dirichlet Allocation* (LDA) dans un contexte plus large, pour ensuite étudier les modèles qui découlent de LDA et qui traitent de la modélisation de l'expertise.

2.3.1 Latent Dirichlet Allocation

Blei *et al.* (2003) ont développé le modèle LDA pour déterminer les sujets associés à un ensemble de documents. Cette section survolera la méthode afin de poser les bases pour la présentation des modèles subséquents. Une analyse complète de la théorie de LDA sera pré-

sentée au chapitre 3.

LDA est un modèle génératif probabiliste qui se base sur une architecture bayésienne. On introduit ici le concept de sujet latent. Chaque document est modélisé par une mixture de sujets latents et chaque sujet est modélisé par une mixture de mots. L'objectif de LDA est de déterminer le degrés d'appartenance aux différents sujets de tous les documents se trouvant dans le corpus étudié. Il est aussi possible de trouver les degrés d'appartenance de tous les mots d'un vocabulaire donné pour chacun des sujets latents. Par conséquent, les sujets ne sont pas spécifiés d'avance. En pratique, quand on parle de degré d'appartenance, on réfère à une distribution de probabilités. Donc, chaque document est modélisé par une distribution de probabilités sur les sujets et chaque sujet est modélisé par une distribution de probabilités sur les mots. Les distributions utilisées sont des distributions de Dirichlet (nous verrons pourquoi au chapitre 3). Les hyperparamètres de Dirichlet servant à générer ces distributions sont α le vecteur de distributions de sujets par document et β le vecteur de distributions de mots par sujet. Dans la nomenclature de l'article, la distribution des sujets pour chaque document est appelée θ, les sujets latents sont référés par $\mathbf{z}$ (dimension = nombre de sujets latents) et les mots sont référés par $\mathbf{w}$ (dimensions = nombre de mots dans le vocabulaire. La distribution postérieure étant donné un document est :

$$P(\theta, \mathbf{z}|\mathbf{w}, \alpha, \beta) = \frac{P(\theta, \mathbf{z}, \mathbf{w}|\alpha, \beta)}{P(\mathbf{w}|\alpha, \beta)}$$

Ici, le terme au dénominateur est insoluble, c'est-à-dire qu'il ne peut pas être calculé directement en marginalisant et en intégrant normalement.

Alors, afin d'estimer les paramètres du modèle, plusieurs techniques existent. Le *Collapsed Gibbs Sampling* (CGS) est une méthode qui sera présentée à la section 3.1.4. Dans l'article de Blei, on utilise plutôt l'inférence variationnelle. Dans le but d'approximer la vraie distribution postérieure, on définit une distribution variationnelle q comme suit :

$$q(\theta, \mathbf{z}|\gamma, \phi) = q(\theta|\gamma) \prod_{n=1}^{N} q(z_n|\phi_n)$$

où N est le nombre de mots dans un document. On constate ici que les hyperparamètres de Dirichlet et les mots sont remplacés par les paramètres variationnels γ et ϕ qui sont libres de toutes contraintes, ce qui rend le problème plus simple. Pour estimer ces paramètres, on doit minimiser la divergence de Kullback-Leibler (divergence KL, Kullback et Leibler (1951)),

entre la distribution variationnelle et la distribution postérieure :

$$(\gamma*, \phi*) = \underset{(\gamma,\phi)}{\operatorname{argmin}} \, \mathrm{D}(q(\theta, \mathbf{z}|\gamma, \phi) \;||\; P(\theta, \mathbf{z}|\mathbf{w}, \alpha, \beta)).$$

Étant donné qu'on ne peut pas calculer cette divergence KL directement, on reformule le problème à l'aide de la log-vraisemblance des données. Avec l'inégalité de Jensen, on a que :

$$\log P(\mathbf{w}|\alpha, \beta) \geq \mathrm{E}_q[\log P(\theta, \mathbf{z}, \mathbf{w}|\alpha, \beta)] - \mathrm{E}_q[\log q(\theta, \mathbf{z})]$$

Or, si on soustrait le membre droit au membre gauche de cette équation, on se retrouve avec la divergence KL précédemment énoncée. Ainsi, l'expression du membre droit fait office de borne inférieure à la log-vraisemblance pour une distribution variationnelle arbitraire. Si on dénote la borne inférieure, soit le membre droit de l'équation ci-dessus par $L(\gamma, \phi; \alpha, \beta)$, on a :

$$\log P(\mathbf{w}|\alpha, \beta) = L(\gamma, \phi; \alpha, \beta) + \mathrm{D}(q(\theta, \mathbf{z}|\gamma, \phi) \;||\; P(\theta, \mathbf{z}|\mathbf{w}, \alpha, \beta))$$

Par conséquent, puisque la log-vraisemblance reste constante, on comprend que maximiser la borne inférieure par rapport à γ et ϕ est analogue à miniser la divergence KL entre la distribution variationnelle et la vraie postérieure. Les détails de cette maximisation seront épargnés, puisqu'ils dépassent largement l'objectif de ce mémoire. Notons seulement qu'il est possible de retrouver ces paramètres variationnels et d'inférer les hyperparamètres de Dirichlet avec ceux-ci.

Enfin, l'article se termine par une démonstration de ce que LDA peut accomplir ; d'abord de façon qualitative, en montrant les termes les plus probables pour chaque sujet latent modélisé (il est souvent possible de déduire le concept du sujet latent avec ces termes). Ensuite, les chercheurs évaluent quantitativement leur modèle en calculant la perplexité sur un sous-corpus du *Text REtrieval Conference* (TREC). La perplexité est une métrique fréquemment utilisée en NLP, puisqu'elle représente l'inverse de la moyenne géométrique de la vraisemblance de chaque mot. Donc, plus la perplexité est basse et meilleur est le modèle. La perplexité peut être exprimée sous la forme suivante :

$$Perplexity(D_{test}) = \exp\left[- \frac{\sum_{d=1}^{M} \log P(\mathbf{w}_d)}{\sum_{d=1}^{M} N_d} \right]$$

où M représente le nombre de documents et N_d est le nombre total de mots dans un document.

Étant donné que le nombre de sujets latents est un hyperparamètre dans le modèle, la per-

plexité est calculée pour différents nombres de sujets spécifiés. Les chercheurs ont observé que plus le nombre de sujets augmente et plus la perplexité est basse. D'ailleurs, Chang *et al.* (2009) démontrent que la perplexité n'est pas la métrique à prioriser si on veut évaluer un modèle de sujets latents comme LDA, parce que la corrélation entre cette métrique et le jugement humain est souvent négative. Pour pallier ce problème, Mimno *et al.* (2011) proposent une nouvelle façon de mesurer la performance de LDA. Il s'agit d'un score de cohérence pour chacun des sujets trouvés par LDA. Le calcul de ce score se fait en mesurant la force de l'association sémantique entre les mots ayant les probabilités d'appartenance les plus élevées pour un sujet donné. De façon explicite, on calcule la cohérence comme suit :

$$C(z; \mathbf{w}^{(z)}) = \sum_{j=2}^{J} \sum_{i=1}^{j-1} \log \frac{D(w_j^{(z)}, w_i^{(z)}) + 1}{D(w_i^{(t)})}$$

Où :

- $\mathbf{w}^{(z)}$ est un vecteur comportant les J mots les plus probables pour le sujet z
- $D(w_j^{(z)}, w_i^{(z)})$ est le nombre de documents qui comporte au moins une fois les mots $w_j^{(z)}$ et $w_i^{(z)}$
- $D(w_i^{(t)})$ est le nombre de documents qui comporte le mot $w_i^{(t)}$

Il est prouvé que cette métrique est plus alignée avec le jugement humain pour l'évaluation d'un modèle comme LDA.

2.3.2 Modèle *Author-Topic*

Maintenant que les modèles sujet ont été introduits, il est important de comprendre comment on peut les utiliser pour inférer l'expertise des auteurs. Rosen-Zvi *et al.* (2004) présentent le modèle *Author-Topic* (AT), ce dernier étant très similaire à LDA classique. Au lieu de représenter les documents comme des entités distinctes, on modélise les auteurs comme étant responsables de la génération des sujets. Alors, on regroupe tous les documents écrits par l'auteur a sous un même macro-document, et on infère les paramètres LDA de la même façon. La distribution θ de l'article de Blei présenté en section 2.3.1 est ici la distribution de sujets par auteur. On retrouve aussi la même distribution de mots par sujet, dénotée par ϕ dans l'article de Rosen-Zvi. Le processus d'inférence est aussi analogue, puisque les chercheurs utilisent le CGS pour inférer les paramètres θ et ϕ. L'objectif du CGS est d'échantillonner de cette postérieure :

$$P(\mathbf{w}, \mathbf{z}, \phi, \theta | \mathbf{a}, \alpha, \beta) = \prod_d \prod_i P(w_{di} | z_{di}, \phi_{z_{di}}) P(z_{di} | \theta_{a_d}) \times \prod_t P(\phi_t | \beta) \prod P(\theta_a) | \alpha)$$

Où :

- **w** réfère aux mots
- **z** réfère aux sujets
- **a** réfère aux auteurs
- i est l'indice d'un mot
- d est l'indice d'un document
- t est l'indice d'un sujet
- a est l'indice d'un auteur

Donc par exemple, lorsqu'on utilise $\phi_{z_{di}}$, on réfère à la valeur de la distribution ϕ associée au mot i et au sujet z_{di}, c'est-à-dire le sujet assigné au mot i du document d. En appliquant cet algorithme, les paramètres peuvent être inférés par CGS :

$$\phi_{it} = \frac{C_{it}^{WT} + \beta}{\sum_{i'} C_{i't}^{WT} + V\beta}$$
$$\theta_{at} = \frac{C_{at}^{AT} + \alpha}{\sum_{t'} C_{at'}^{AT} + T\alpha}$$

Où :

- V est le nombre de mots total dans le vocabulaire
- T est le nombre total de sujets
- C_{it}^{WT} est le nombre de fois que le $i^{\text{ième}}$ mot est assigné au sujet t
- C_{at}^{AT} est le nombre de fois que l'auteur a est assigné au sujet t

Après avoir calculé ces deux distributions, les chercheurs procèdent à une recherche d'experts pour chaque sujet. Pour un sujet t, ils sélectionnent les 10 auteurs ayant la valeur de θ_t la plus élevée et ils les identifient comme expert du sujet. Or, cette méthode d'identification d'experts est problématique. En effet, étant donné que, pour un certain auteur, la répartition de son expertise est une distribution parmi les sujets trouvés, la somme des θ_t est de 1. Par conséquent, il s'agit d'une expertise relative parmi les sujets identifiés. Par exemple, prenons le cas où on exécute le modèle sur 50 auteurs qui traitent principalement d'apprentissage machine dans leurs articles. Considérons le cas où le modèle identifie 3 sujets : les réseaux de neurones, les machines à vecteurs de support et les modèles probabilistes génératifs. Supposons aussi qu'un auteur de méthodes quantitatives pour les populations de bactérie se glisse par erreur dans le corpus. Cet auteur n'utilise que très peu de termes reliés aux probabilités, mais aucun terme en lien avec les réseaux de neurones et les machines à vecteurs de support. Alors, le modèle AT estimera un θ_t très élevé à cet auteur pour le sujet des modèles probabilistes, puisque relativement parlant, l'importance de ce sujet est beaucoup plus grande. Conséquemment, cet auteur pourrait être identifié à tort comme expert pour ce sujet devant

un auteur accompli en apprentissage machine, car même si les connaissances de ce dernier en modèles probabilistes sont nettement supérieures, ces dernières sont beaucoup plus balancées entre les trois sujets identifiés par le modèle.

De ce fait, il est nécessaire de réfléchir à une méthode plus robuste afin de déterminer l'expertise. Toutefois, le modèle AT constitue un bon point de départ pour cette réflexion.

2.3.3 Modèle Author-Persona-Topic

Le modèle Author-Persona-Topic (APT) est présenté dans un article de Mimno et McCallum (2007). Ce modèle se base sur les mêmes concepts que le modèle AT, mais une couche supplémentaire est ajoutée : on introduit le concept de persona. Les chercheurs émettent l'hypothèse que chaque auteur est constitué d'une ou plusieurs personas. Le terme persona est issu du domaine de la psychologie et signifie ici un style d'écriture, une spécialisation des sujets d'expertise. Ainsi, chaque auteur possède une distribution de personas η_a (le nombre de personas pour un auteur est variable). Cette distribution de personas est aussi tirée d'une distribution de Dirichlet, paramétrée par γ_a. L'idée est de représenter chaque persona d'un auteur par une distribution de sujets, ces sujets étant encore déterminés par une distribution de mots.

On obtient donc la probabilité du corpus :

$$P(\mathbf{w}, \mathbf{z}, \mathbf{g}, \eta, \phi, \theta | \mathbf{a}, \alpha, \beta, \gamma) = \prod_d \left[P(g_d | n_{a_d}) \prod_i P(w_{di} | z_{di}, \phi_{z_{di}}) P(z_{di} | \theta_{g_d}) \right] \times \prod_t P(\phi_t | \beta) \prod_g P(\theta_g) | \alpha) \prod_a P(\eta_a | \gamma_a)$$

où g_d représente la persona sélectionnée pour un document d. On applique aussi le CGS pour estimer cette probabilité postérieure. En utilisant cet algorithme et en suivant la même logique que pour le modèle AT, on obtient les paramètres suivants pour l'association des sujets aux mots et aux personas :

$$\phi_{it} = \frac{C_{it}^{WT} + \beta}{\sum_{i'} C_{i't}^{WT} + V\beta}$$
$$\theta_{gt} = \frac{C_{gt}^{GT} + \alpha}{\sum_{t'} C_{gt'}^{GT} + T\alpha}$$

Enfin, on estime la distribution d'une persona sachant les assignations sujet-mot $\mathbf{z}$ de la façon suivante :

$$P(g_d | \mathbf{z}, a, \gamma_a) \propto \frac{\gamma_{a_g} + N_a^g}{\sum_{a_g} (\gamma_{a_g} + N_a^g)} \times \frac{\Gamma \sum_t (\alpha_t + N_{gd\setminus d}^t)}{\prod_t \Gamma(\alpha_t + N_{gd\setminus d}^t)} \frac{\prod_t \Gamma(\alpha_t + N_{gd}^t)}{\Gamma \sum_t (\alpha_t + N_{gd}^t)}$$

Où :

- N_a^g est le nombre de documents écrits par l'auteur a et assignés à la persona g
- $N_{gd\backslash d}^t$ représente le nombre de mots assignés au sujet t dans tous les documents autre que d qui sont assignés à la persona g_d

En ce qui concerne l'évaluation du modèle, les chercheurs simulent une tâche de recherche d'experts pour la revue d'articles scientifiques. Pour ce faire, ils exécutent leur modèle sur un corpus de la *Neural Information Processing Systems conference* (NIPS). Dans cette base de données, on peut y retrouver des archives d'auteurs ainsi que des articles déposés pour la revue. Comme il est difficile de trouver un *ground truth* sur la bonne association d'un réviseur avec un article, les chercheurs utilisent le jugement humain d'experts étant sur le panel de conférence de NIPS 2006. Les modèles AT et APT sont entraînés pour 75 (AT-75, APT-75) et 200 (AT-200, APT-200) sujets chacun et ceux-ci sont comparés avec un modèle génératif candidat (comme présenté en section 2.2.1). Différents nombres de réviseurs sont recueillis et les experts sur le panel jugent si, oui ou non, ces choix sont pertinents. La précision est ensuite calculée. Nous aurions voulu avoir accès à ces données dans le cadre du projet de maîtrise, mais celles-ci n'étaient pas disponibles. Voici le graphique des résultats obtenus :

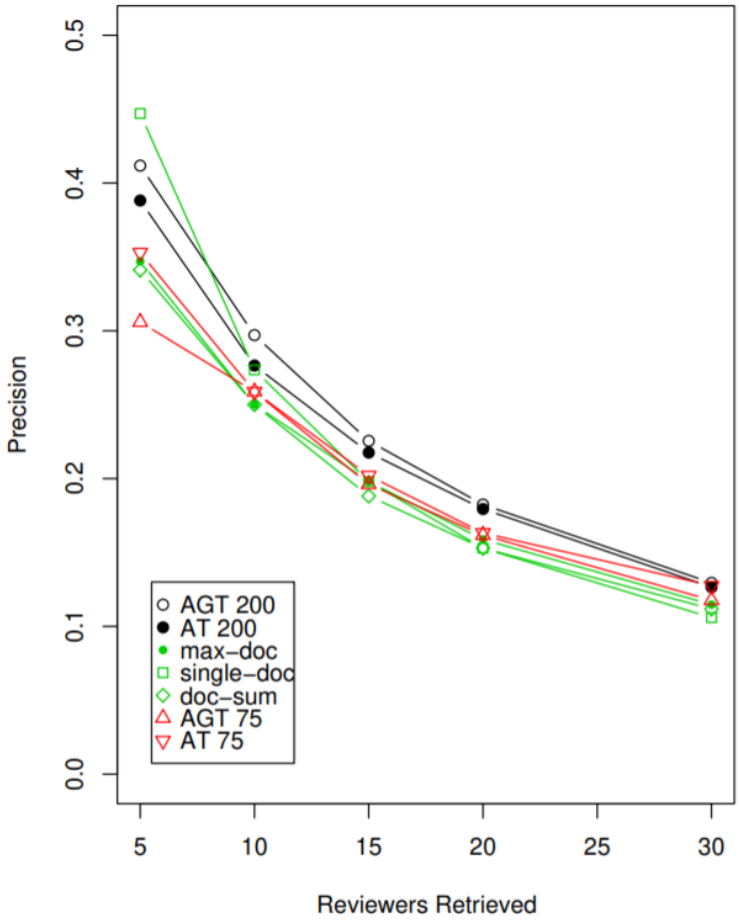


Figure 2.1 La précision de chaque modèle selon le nombre de documents recherchés (tiré de Mimno et McCallum (2007))

Mimno et McCallum concluent que, dans le cas où on considère seulement les cinq premiers réviseurs, le modèle avec la précision la plus élevée est le modèle génératif candidat, suivi par AT-200 et APT-200. Notons que la variante à 200 sujets surpasse dans tous les cas la variante à 75 sujets, et ce, pour tous les nombres d'experts recueillis. Comme il est attendu, plus le nombre de réviseurs recueillis augmente et moins la précision est bonne pour tous les modèles. Or, fait intéressant, la chute de précision est plus importante pour le modèle génératif candidat et le modèle AT-200 que pour le modèle APT-200. Effectivement, à 30 réviseurs recueillis, le meilleur modèle est APT-200, suivi par AT-200 et le modèle génératif candidat, soit l'ordre inverse de ce qui était observé pour 5 réviseurs. En fait, les performances du modèle génératif candidat deviennent nettement moins bonnes que les modèles AT-200 et APT-200 à partir de 10 réviseurs recueillis.

On voit donc que le modèle APT-200 possède un meilleur rappel que ce qui avait été obtenu auparavant, ce qui vient résoudre partiellement le problème d'expertise relative expliqué en section 2.3.2. Cependant, étant donné que le meilleur rappel est dû à des associations de sujets plus spécialisés et non à une compréhension plus profonde des connaissances de l'expert, on ne peut pas dire que ce problème est résolu dans son entièreté. En somme, les chercheurs concluent que le modèle APT-200 est le plus approprié pour la tâche de recherche d'experts dans le contexte de recherche de réviseurs pour articles scientifiques. Ceci s'explique par le fait qu'on a souvent besoin d'un bon rappel pour cette tâche, puisque de nombreuses contraintes externes au problème d'association existent, telles que les conflits d'intérêts ainsi que le quota d'articles à réviser par réviseurs. À cause de ces contraintes, certains réviseurs recueillis seront supprimés et c'est pourquoi il est souvent utile de générer un plus grand nombre de réviseurs pertinents que nécessaire.

2.3.4 *Toronto Paper Matching System*

Nous terminons cette section par la mention du *Toronto Paper Matching System* élaboré par Charlin et Zemel (2013). On croit pertinent de le mentionner, puisqu'il accomplit une tâche similaire au cadre d'application du mémoire, soit l'association entre des auteurs et des articles scientifiques à réviser. L'objectif de ce système est de fournir un service automatisé de répartition des articles aux réviseurs afin, d'une part, de réduire le temps ainsi que l'aspect laborieux de cette tâche et, d'autre part, d'obtenir des réviseurs plus adaptés à la révision d'un article en particulier. Une association réviseur-article est considérée comme adaptée si elle remplit les critères suivants :

- Le réviseur doit avoir l'expertise requise pour réviser l'article
- La charge de travail de chaque réviseur doit être inférieure à la charge de travail maximale définie
- Chaque article doit être révisé par un nombre minimal de réviseur
- Les réviseurs ne doivent pas avoir de conflit d'intérêt en lien avec les articles qu'ils révisent

Par conséquent, le système possède une architecture complexe d'optimisation où l'identification du niveau d'expertise n'est qu'une tâche parmi d'autres.

Dans le but de déterminer le niveau d'expertise, les chercheurs analysent la composition lexicale des articles à réviser ainsi que celle des archives des réviseurs potentiels. L'algorithme LDA est exécuté sur ces corpus et un score d'affinité est calculé. Une fois que ces scores d'affinité sont déterminés, une optimisation par régression linéaire est performée, celle-ci utilisant le score d'affinité comme attribut principal et l'expertise assignée aux réviseurs par

les membres de la chaire comme *ground truth*. Cette représentation de l'adéquation réviseur-article est ensuite utlisée au sein d'une infrastructure d'optimisation sous les contraintes précédemment énumérées pour générer l'association voulue. Alors, l'algorithme ne permet pas la compréhension du niveau d'expertise en tant que tel, car la similarité des sujets inférés par LDA n'a pas de profondeur technique. En effet, le score d'affinité calculé dans cette technique procure de l'information sur la similarité des sujets traités dans l'article cible et l'archive des réviseurs (ce qui est toutefois utile dans un contexte d'association) plutôt que d'un réel niveau d'expertise sous-jacent. Par ailleurs, le présent projet découle d'une initiative impliquant Laurent Charlin dont la motivation est d'améliorer la modélisation de l'expertise du *Toronto Paper Matching System*.

Ceci conclut la revue de littérature. Nous avons vu que l'utilisation d'un langage peut être prédite par des lois statistiques et qu'il existe des différences lexicales entre les productions écrites d'un novice et celles d'un expert. De plus, les avantages ainsi que les inconvénients des modèles liés à la recherche d'experts et des modèles à sujets latents ont été traités. À la lumière de ces constatations, on élaborera une nouvelle technique d'inférence de l'expertise qui se base sur la même infrastructure que LDA, mais qui incorpore des méthodes découlant des lois mathématiques régissant le langage et les particularités lexicales d'un expert.

CHAPITRE 3 LDA ET UN CADRE DE VALIDATION

Ce chapitre porte sur le cadre théorique de LDA. Puisque la solution proposée dans le mémoire est une extension de cet algorithme, il est nécessaire de poser les fondements de celui-ci. Dans un premier temps, les caractéristiques du modèle LDA classique seront expliquées. Ensuite, on présentera un cadre de validation du modèle grâce à des données synthétiques. Finalement, on utilisera ce cadre de validation pour effectuer trois analyses. D'abord, la performance des méthodes d'inférence des paramètres de LDA sera évaluée en fonction des hyperparamètres utilisés pour la génération de données et en fonction du nombre de sujets latents spécifiés. Ensuite, on établira la similitude entre les fréquences de mots observées dans les données générées et les fréquences théoriques de Zipf et Mandelbrot. Alors ce chapitre abordera la première question de recherche, soit : quelles sont les conditions opérationnelles du modèle LDA classique et dans quelle mesure l'hypothèse de génération de ce modèle est-elle conforme aux lois statistiques du langage ? Les liens qui seront découverts dans ces deux analyses n'ont pas été étudiés dans la littérature et ceux-ci constituent l'une des contributions du mémoire. Enfin, à la lumière de ces deux analyses, on choisira les hyperparamètres α et β pour la portion génération de données du cadre de validation qui fournissent le meilleur compromis entre performances de la méthode d'inférence (évaluée par la divergence KL entre les distributions ϕ générées et inférées) et la similitude avec les lois du langage. On conclura ce chapitre par la présentation des statistiques complètes d'un modèle LDA classique utilisant les hyperparamètres choisis pour la génération. On utilisera ces statistiques comme outil de comparaison avec les performances qui seront obtenues pour le nouveau modèle qui inclura l'expertise et qui sera présenté au chapitre 4.

3.1 Description du modèle LDA classique

Étant donné que la présentation de LDA est très variable selon les articles, nous traiterons des caractéristiques précises du modèle qui sera le sujet du mémoire. On discutera de l'objectif de cet algorithme, de la nomenclature que nous utiliserons, du processus de génération sur lequel LDA repose ainsi que d'une brève explication théorique de la distribution de Dirichlet. La présentation de la distribution de Dirichlet est importante, puisqu'elle permet d'expliquer pourquoi il est logique d'utiliser la technique d'inférence du *Collapsed Gibbs Sampling* (CGS) qui sera présentée à la fin de cette sous-section.

3.1.1 Objectif du modèle LDA classique

Comme expliqué dans la section 2.3.1, le modèle LDA classique (*Latent Dirichlet Allocation*) a été développé par Blei *et al.* (2003) et est ensuite devenu la méthode de référence pour la modélisation de sujets pour un corpus de documents.

LDA est une forme non supervisée d'apprentissage machine qui permet de segmenter différents groupes de mots appartenant à un corpus de documents. Ces groupes de mots segmentés forment les sujets latents qui caractérisent le corpus étudié. Ces sujets sont appelés latents, puisqu'ils ne sont pas observés directement et on les étiquette seulement grâce à la signification globale des mots qui les constituent. Par exemple, on pourrait poser l'étiquette « informatique » au sujet qui regroupe les mots « ordinateur », « algorithme » et « réseau ». Ces sujets identifiés sont les mêmes pour tous les documents du corpus. Or, LDA permet d'octroyer des probabilités d'appartenance à chacun des sujets, celles-ci étant fonction du document. Par conséquent, en dépit du fait qu'il s'agisse d'un modèle de segmentation non supervisée des données, on dit que LDA modélise les sujets traités par un corpus à cause de ces associations.

Les hyperparamètres en entrée du modèle sont les hyperparamètres de Dirichlet ainsi que le nombre voulu de sujets latents. Les paramètres de sortie sont les probabilités de chaque sujet par document ainsi que les probabilités de chaque mot par sujet.

3.1.2 Nomenclature du modèle LDA classique

LDA est un modèle probabiliste génératif, puisqu'il se base sur une hypothèse statistique de génération des données. C'est en essayant de retrouver les paramètres qui auraient donné lieu à cette génération qu'on peut inférer les sujets découlant d'un certain ensemble de données. Le processus génératif sur lequel est fondé LDA est le suivant :

Notons que θ et ϕ sont des matrices ;nous n'avons pas présenté les boucles pour les créer afin de ne pas alourdir sans raison l'algorithme. Ici, la distribution multinomiale représente une sélection à essai unique, celle-ci étant parfois appelée la distribution catégorique. L'objectif de cette génération est de créer un corpus de N_D documents, chacun de ces documents contenant N_{WD} mots. Ces mots sont tirés d'un vocabulaire V contenant N_V mots et, suite au processus de génération, N_W mots uniques auront été sélectionnés. Ce processus est caractérisé par deux distributions névralgiques. D'une part, la matrice ϕ est la distribution de mots pour chaque sujet. Elle est de dimension $[N_K \times N_V]$ où N_K représente le nombre de sujets latents

Algorithme 1:

Résultat : Génère un corpus de N_D documents de N_{WD} mots
$\phi \sim \text{Dir}(\beta)$
$\theta \sim \text{Dir}(\alpha)$
for *i=1 :*N_D **do**
  for *j=1 :*N_{WD} **do**
    $k_{ij} \sim \text{mult}(\theta_i)$
    $w_{ij} \sim \text{mult}(\phi_{k_{ij}})$
  end
end

Figure 3.1 Processus de génération des données de LDA

spécifié comme hyperparamètre. D'autre part, la matrice θ est la distribution de sujets pour chaque document et elle est de dimension $[N_D \times N_K]$. On introduit aussi la nomenclature k, d et w pour désigner respectivement un sujet, un document et un mot particulier. Notons ici que N_V et N_W n'auront pas la même valeur, puisque certains mots du vocabulaire ne seront pas choisis dans le processus de génération. En d'autres mots, dans le code de génération des données synthétiques, la dimension de ϕ est $[N_K \times N_V]$, mais on tronque les colonnes où le mot associé n'est pas choisi dans la génération. On ne veut pas garder les mots non choisis dans la matrice ϕ, car l'inférence de cette dernière ne considère que les données observées et, lorsque viendra le temps de calculer les métriques de comparaison entre les données générées et inférées, on doit supprimer les mots non choisis du vocabulaire initial afin de garder les matrices ϕ générées et inférées de la même taille. Par conséquent, la dimension de ϕ dans le code d'inférence est $[N_K \times N_W]$. En résumé, voici la nomenclature propre à la génération :

- d : Document particulier
- k : Sujet particulier
- w : Mot particulier
- V : Vocabulaire
- N_D : Nombre de documents dans le corpus
- N_K : Nombre de sujets latents spécifiés
- N_W : Nombre de mots uniques sélectionnés
- N_V : Nombre de mots dans le vocabulaire
- N_{WD} : Nombre de mots dans un document
- θ : Matrice de dimension $[N_D \times N_K]$ où chaque vecteur ligne représente la distribution de sujets pour un document
- ϕ : Matrice de dimension $[N_K \times N_W]$ où chaque vecteur ligne représente la distribution de mots pour un sujet.

— α : Vecteur d'hyperparamètres de Dirichlet pour θ
— β : Vecteur d'hyperparamètres de Dirichlet pour β

Pour illustrer concrètement le processus de génération, un exemple sera présenté. Prenons un cas où nous avons $N_D = 2$, $N_K = 3$, $N_V = 6$ et $N_{\text{WD}} = 2$. Voici le processus de génération d'un document associé à ces paramètres :

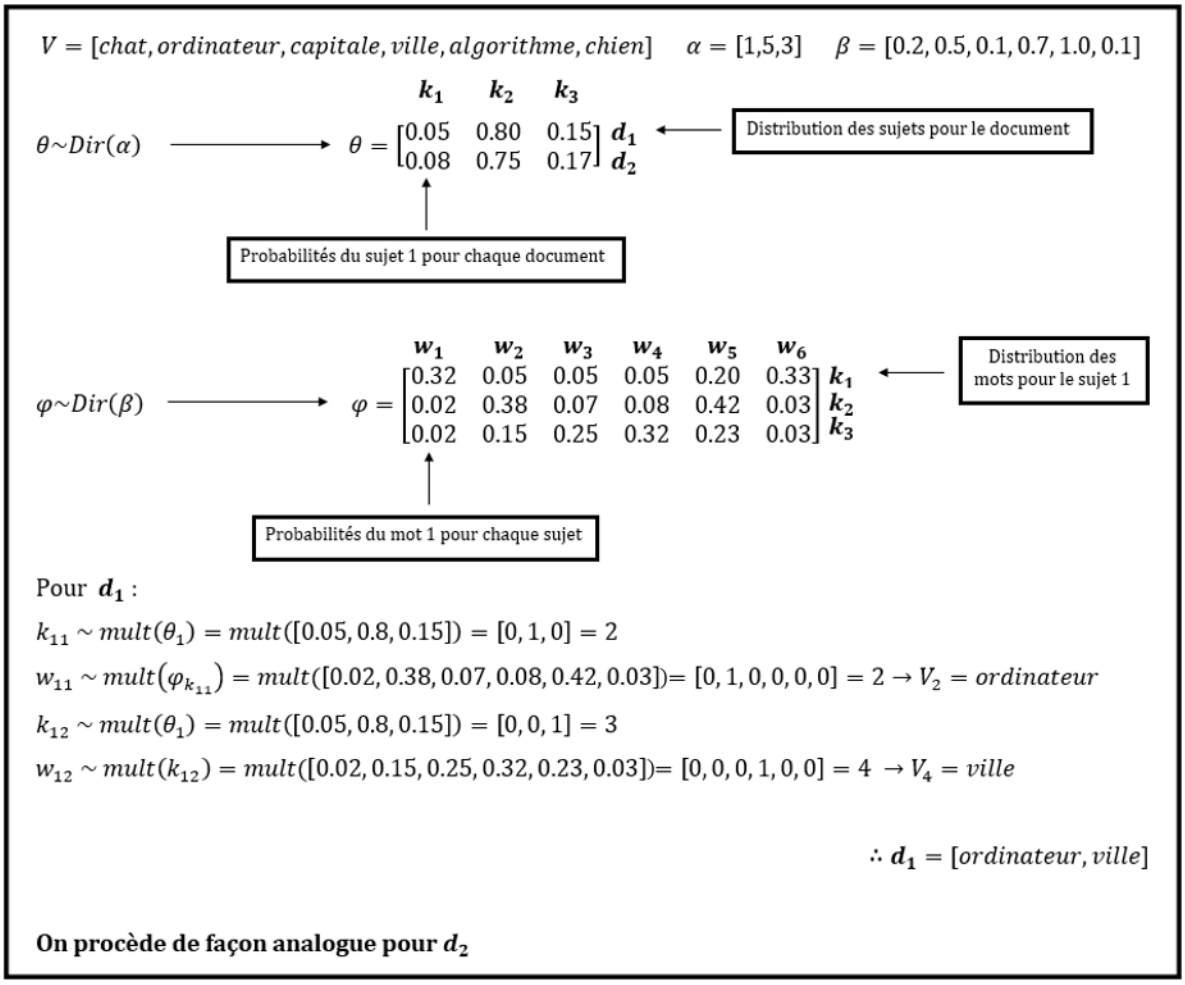


Figure 3.2 Exemple concret du processus de génération pour $N_D = 2$, $N_K = 3$, $N_V = 6$ et $N_{\text{WD}} = 2$

3.1.3 La distribution de Dirichlet

Dans le contexte de LDA, l'objectif de la distribution de Dirichlet est d'exprimer la variabilité d'une distribution multinomiale. Supposons que l'on fabrique des pièces de monnaie et on veut connaître la probabilité d'obtenir pile ou face après n lancers. On définit les probabilités des possibilités dans un vecteur $\theta = (\theta_1, \theta_2)$ avec θ_1 la probabilité d'obtenir pile et θ_2 la probabilité d'obtenir face. Puisque toutes les composantes de θ sont positives et doivent avoir une somme de 1, les résultats des lancers peuvent être représentés par une distribu-

tion multinomiale (aussi appelée distribution binomiale pour ce cas bivarié). Bien qu'il serait possible théoriquement de produire des pièces justes (probabilité égale de 0.5 pour θ_1 et θ_2), on connaît l'existence de défauts dans le processus de fabrication qui favorisent l'obtention du côté face, mais on ne sait pas dans quelle mesure. Afin de calculer la variabilité aléatoire de cette distribution multinomiale selon cette intuition, on a recours à la distribution de Dirichlet.

La distribution de Dirichlet prend en paramètre un vecteur α qui est de la même taille que le vecteur θ. Cet hyperparamètre a pour but de caractériser le poids des différentes possibilités dans la création d'une distribution multinomiale issue d'une distribution de Dirichlet. Par exemple, pour le cas de la pièce de monnaie, voici un graphique exprimant plusieurs fonctions de densité de Dirichlet selon différents vecteurs de paramètres α.

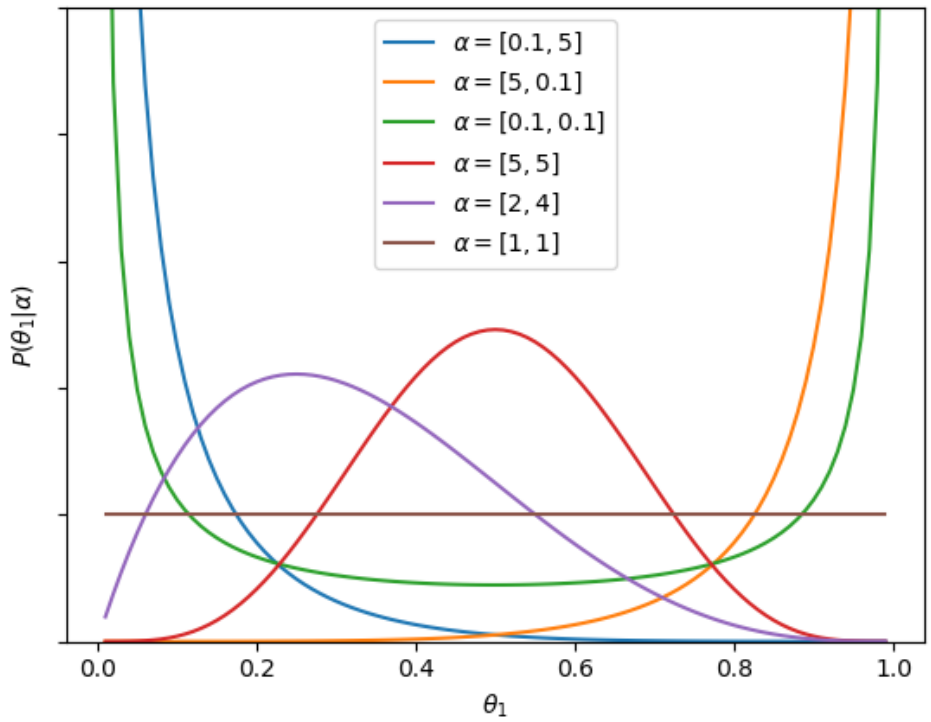


Figure 3.3 Distribution Beta dans le cas 2D pour différentes valeurs de α

Ici, on illustre le graphique de $P(\theta_1|\alpha)$ en fonction de θ_1, donc on veut déterminer la densité de probabilités associée avec le paramètre multinomial θ_1, et ce, pour différentes combinaisons de α_i. En d'autres mots, on veut répondre à la question suivante : quelle est la probabilité d'ob-

tenir une certaine probabilité de réussir un lancer pile, considérant notre croyance *a priori*, cette dernière étant modélisée par les α_i. Par exemple, la courbe mauve illustre notre cas où on sait que le côté face est favorisé par le processus de fabrication ($\alpha = [2, 4]$: on octroie une valeur de α plus grande à θ_2). On constate que $P(\theta_1|\alpha)$ est maximal lorsque $\theta_1 \approx 0.25$. Ceci signifie que, si on adoptait cette croyance *a priori* $\alpha = [2, 4]$, la distribution multinomiale qui décrirait de manière la plus probable le lancer de pièce serait $\theta = [0.25, 0.75]$, ce qui favorise grandement le côté face. En étudiant les autres combinaisons de α_i présentes dans ce graphique, on remarque certaines propriétés intéressantes de la distribution de Dirichlet. D'abord, les courbes bleues et oranges montrent les cas où l'une ou l'autre des possibilités est priorisée à l'extrême. Par exemple, si on croyait que le côté face était favorisé à l'extrême, la courbe bleue montre que la probabilité d'obtenir pile serait presque nulle. Pour une distribution où les α_i sont égaux, on dit que celle-ci est symétrique. De plus, si la distribution est symétrique et que les $\alpha_i > 1$, on produirait une pièce juste en moyenne. Par ailleurs, dans le cas théorique où les $\alpha_i \to \infty$, on produirait toujours des pièces justes. Par conséquent, plus le poids des α est élevé et moins les distributions multinomiales issues de cette distribution de Dirichlet auront de variabilité. Enfin, si tous les $\alpha_i = 1$, on obtient la distribution uniforme (courbe brune) et si la distribution est symétrique et que les $\alpha_i < 1$, on obtient une distribution en forme de U comportant des pics aux extrémités. Dans le cas limite d'une distribution symétrique où les $\alpha_i \to 0$ (représenté par la courbe verte), on aurait deux deltas de Dirac en $x = 0$ et $x = 1$ avec une probabilité associée de 0.5 à chaque extrémité et une probabilité de 0 partout ailleurs.

Le comportement d'une distribution de Dirichlet est facile à concevoir pour le cas bivarié, car on peut représenter les fonctions de densité sur un graphique. Or, en ce qui concerne LDA, on se retrouve dans un cas multivarié. Alors, essayons maintenant de visualiser un exemple de distribution $\theta \sim Dir(\alpha)$ pour LDA avec $K = 3$ sujets.

Étant donné que les paramètres multinomiaux doivent être positifs et avoir une somme 1, l'ensemble des valeurs pour le vecteur θ est restreint à un triangle : on appelle ce triangle un 2-simplex que voici :

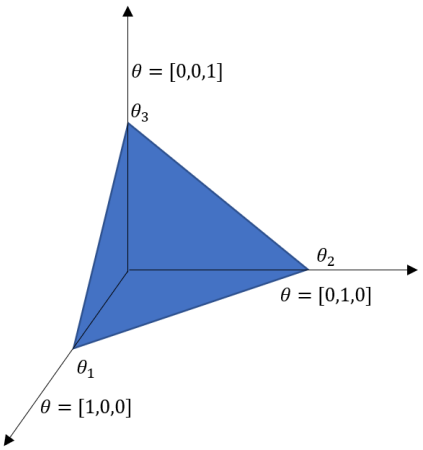


Figure 3.4 Représentation d'un 2-simplex

On représente maintenant sur cet espace les fonctions de densité d'une distribution de Dirichlet en 3 dimensions pour différents vecteurs α. Une couleur plus claire indique une probabilité plus élevée :

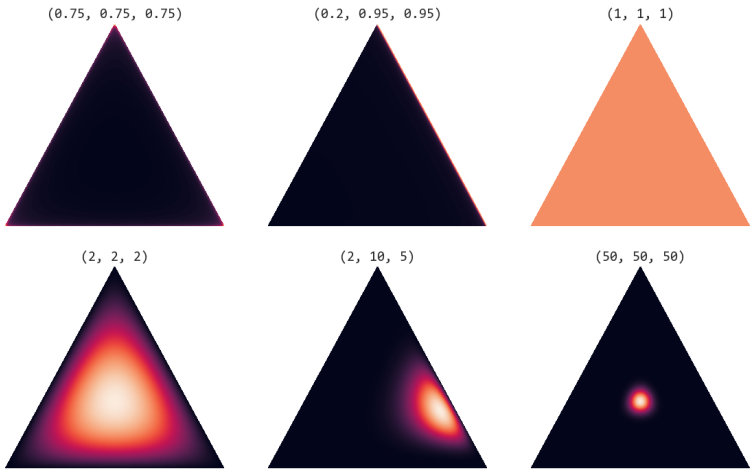


Figure 3.5 Fonction de densité d'une distribution de Dirichlet 3D pour différentes valeurs de α

Des constatations similaires au cas bivarié peuvent être observées. D'abord, lorsque la distribution est asymétrique, on voit que la fonction de densité se concentre vers les pointes où le α associé est plus élevé. Si les $\alpha_i > 1$, la fonction est déphasée vers les pointes depuis le centre du simplex, ce dernier représentant une distribution multinomiale juste $\theta = [\frac{1}{3}, \frac{1}{3}, \frac{1}{3}]$. On remarque aussi que plus le poids des α_i est important et plus la région probable se concentre en un point. Si les $\alpha_i < 1$, la fonction de densité est plutôt confinée aux bordures du simplex et le même type de déphasage vers les pointes a lieu. Enfin, on observe encore la distribution uniforme lorsque $\alpha_i = 1$ et les distributions lorsque les $\alpha_i \to 0$ ou $\alpha_i \to \infty$ sont analogues au cas bivarié.

Il est maintenant pertinent de se demander les raisons qui motivent le choix de la distribution de Dirichlet comme distribution *a priori* dans LDA. Lorsque l'on veut échantillionner sur une distribution postérieure et mettre à jour les paramètres d'une distribution *a priori* sans passer par une intégration numérique complexe, il est nécessaire que cette dernière soit une distribution dite conjuguée à la postérieure. Si ces distributions respectent cette condition, on peut avoir une forme fermée pour la postérieure, ce qui la rend plus facile à calculer. Une distribution *a priori* $P(\theta|\alpha)$ est définie conjuguée à une postérieure $P(\theta|\alpha, x)$ où x représente les données, si ces deux distributions font partie de la même famille. On dit donc que cette distribution *a priori* est la distribution *a priori* conjuguée à la fonction de vraisemblance.

Or, la distribution de Dirichlet est la distribution *a priori* conjuguée de la distribution multinomiale. En effet, voyons d'abord l'expression de la fonction de densité d'une distribution de Dirichlet à K dimensions (tirée de Sklar (2014)).

$$f(\theta) = \frac{\prod_{i=1}^{K} \Gamma(\alpha_i)}{\Gamma(\sum_{i=1}^{K} \alpha_i)} \prod_{i=1}^{K} \theta_i^{\alpha_i - 1} = \frac{1}{B(\alpha)} \prod_{i=1}^{K} \theta_i^{\alpha_i - 1}$$

Où $B(\alpha) = \frac{\Gamma(\sum_{i=1}^{K} \alpha_i)}{\prod_{i=1}^{K} \Gamma(\alpha_i)}$. Dans cette équation, le terme $\frac{1}{B(\alpha)}$ sert à normaliser l'expression alors que le terme $\prod_{i=1}^{K} \theta_i^{\alpha_i - 1}$ témoigne de la proportionnalité de $P(\theta|\alpha)$. Prenons un exemple où $\theta \sim \text{Dir}(\alpha)$ et où on observe avec les données un vecteur cumulant le nombre d'occurrences de chaque mot $w \sim \text{mult}(\theta)$. En utilisant la proportionnalité, on peut exprimer la postérieure

$P(\theta|\alpha, x)$ comme suit :

$$\begin{aligned} P(\theta|\alpha, w) &\propto P(w|\theta)P(\theta|\alpha) \\ &\propto \prod_{i=1}^{K} \theta^{w_i} \prod_{i=1}^{K} \theta^{\alpha^i - 1} \\ &\propto \prod_{i=1}^{K} \theta^{w_i + \alpha_i - 1} \end{aligned}$$

Alors, étant donné que la postérieure $P(\theta|\alpha, x) \propto \prod_{i=1}^{K} \theta^{w_i+\alpha_i-1}$ et que la distribution *a priori* $P(\theta|\alpha) \propto \prod_{i=1}^{K} \theta^{\alpha^i-1}$, on constate que ces deux distributions ont la même forme et, par conséquent, on dit qu'elles sont conjuguées. Par ailleurs, on remarque que, pour passer de la distribution *a priori* à la postérieure, on a qu'à ajouter les fréquences observées dans les données. Ainsi, grâce à cette propriété, on pourra utiliser des techniques d'inférence mathématiquement simples, telles que le CGS, afin de retrouver les paramètres θ.

3.1.4 Inférence avec le *Collapsed Gibbs Sampling* (CGS)

Il a été vu à la section 2.3.1 qu'il existe deux méthodes classiques pour inférer les paramètres θ et ϕ de LDA : l'inférence variationnelle et le CGS. Puisque l'inférence variationnelle a déjà été présentée dans la section 2.3.1, nous allons nous attarder sur le CGS.

Le CGS (George et McCulloch (1993)) est un algorithme de type MCMC (Markov Chain Monte Carlo). Cette méthode tente d'établir une chaîne de Markov qui possède la distribution postérieure qu'on veut évaluer en tant que distribution stationnaire. De cette manière, après un certain nombre d'itérations où on échantillonne sur la chaîne, ces échantillons devraient converger vers ceux qu'on devrait obtenir sur la vraie postérieure.

Dans le contexte de LDA, nous allons introduire 2 nouvelles variables :

- X : Une matrice de dimension $[N_D \times N_{\text{WD}}]$ représentant le corpus où chaque $X(d, w)$ représente l'identifiant du mot w dans le document d.
- Z : Une matrice de même dimension que X représentant l'assignation d'un sujet k à chaque mot du corpus. Par exemple, si $Z(d, w) = 1$, cela signifie que le mot w du document d est assigné au sujet 1.
- W : Une matrice de dimension $[N_D \times N_W]$ représentant la fréquence des mots dans chaque document.

Afin d'expliquer le CGS, commençons par exprimer la postérieure de LDA :

$$P(\theta, \phi, Z|W, \alpha, \beta) = \frac{P(\theta, \phi, Z, W|\alpha, \beta)}{P(W|\alpha, \beta)}$$

Étant donné que cette distribution est insoluble (bien que le numérateur possède une forme fermée, le terme de normalisation au dénominateur $P(W|\alpha, \beta)$ ne peut être calculé analytiquement), on a recours à une méthode d'échantillonnage. Les valeurs qu'on cherche à échantillonner sont les assignations Z d'un mot w_i à un sujet k_i. On cherche donc à évaluer la probabilité suivante :

$$P(z_i|Z^{(-i)}, \alpha, \beta, W)$$

Soit la probabilité d'obtenir une assignation particulière z_i sachant l'ensemble des autres assignations (exceptée i) $Z^{(-i)}$, les hyperparamètres de Dirichlet ainsi que les mots présents dans le corpus. On peut reformuler cette expression en utilisant la loi des probabilités conditionnelles :

$$P(z_i|Z^{(-i)}, \alpha, \beta, W) = \frac{P(z_i, Z^{(-i)}, W|\alpha, \beta)}{P(Z^{(-i)}, W|\alpha, \beta)} \propto P(Z, W|\alpha, \beta)$$

Il est maintenant possible d'exprimer $P(Z, W|\alpha, \beta)$ en intégrant sur θ et ϕ :

$$\begin{aligned} P(Z, W|\alpha, \beta) &= \int\int P(Z, W, \theta, \phi|\alpha, \beta)\,\mathrm{d}\theta\,\mathrm{d}\phi \\ &= \int P(Z|\theta)P(\theta|\alpha)\,\mathrm{d}\theta \int P(W|\phi_z)P(\phi|\beta)\,\mathrm{d}\phi \end{aligned}$$

Ces deux intégrales sont des distributions multinomiales où des distributions de Dirichlet ont été utilisées comme croyance *a priori*. Comme expliqué à la section 3.1.3, le fait que la distribution de Dirichlet soit la distribution *a priori* conjuguée à la distribution multinomiale explique pourquoi les mathématiques se simplifient grandement. En effet, de façon similaire à la démonstration du cas canonique présentée à la fin de la section 3.1.3, on peut montrer qu'une intégration numérique complexe n'est pas nécessaire, car ces intégrales possèdent une forme fermée. Cette forme est simple, puisqu'il s'agit de l'addition des hyperparamètres de Dirichlet avec les fréquences observées dans W et Z :

$$\begin{aligned} \int P(Z|\theta)P(\theta|\alpha)\,\mathrm{d}\theta &= \int \prod_{k=1}^{N_K} \theta_{d,z_k} \frac{1}{B(\alpha)} \prod_{k=1}^{N_K} \theta_{d,k}^{\alpha_k - 1}\,\mathrm{d}\theta \\ &= \prod_{d=1}^{N_D} \frac{1}{B(\alpha)} \int \prod_{k=1}^{N_K} \theta_{d,k}^{n_{d,k} + \alpha_k - 1}\,\mathrm{d}\theta_d \\ &= \prod_{d=1}^{N_D} \frac{B(n_{d,k} + \alpha - 1)}{B(\alpha)} \end{aligned}$$

où $n_{d,k}$ représente le nombre de fois que des mots du document d ont été assignés au sujet k. De façon analogue, on a :

$$\begin{aligned}\int P(W|\phi_z)P(\phi|\beta)\,\mathrm{d}\phi &= \int \prod_{d=1}^{N_D}\prod_{i=1}^{N_{\mathrm{WD}}} \phi_{z_{d,i},W_{d,X_{d,i}}} \frac{1}{B(\beta)} \prod_{w=1}^{N_W} \phi_{k,w}^{\beta_w-1}\,\mathrm{d}\phi \\ &= \prod_{k=1}^{N_K} \frac{1}{B(\beta)} \int \prod_{w=1}^{N_W} \phi_{k,w}^{n_{k,w}+\beta_w-1}\,\mathrm{d}\phi_k \\ &= \prod_{k=1}^{N_K} \frac{B(n_{k,w}+\beta-1)}{B(\beta)}\end{aligned}$$

où $n_{k,w}$ représente le nombre de fois que le mot w a été assigné au sujet k.

Puisque ces deux intégrales ont été exprimées en forme fermée, il est par conséquent aussi possible d'exprimer $P(z_i|Z^{(-i)},\alpha,\beta,W)$ en forme fermée :

$$P(z_i|Z^{(-i)},\alpha,\beta,W) \propto \prod_{d=1}^{N_D} \frac{B(n_{d,k}+\alpha-1)}{B(\alpha)} \prod_{k=1}^{N_K} \frac{B(n_{k,w}+\beta-1)}{B(\beta)} \propto \frac{n_{d,k}^{(-i)}+\alpha_k}{\sum_{k=1}^{N_K} n_{d,k}^{(-i)}+\alpha_k} \frac{n_{k,w}^{(-i)}+\beta_w}{\sum_{w=1}^{N_W} n_{k,w}^{(-i)}+\beta_w}$$

De cette manière, la probabilité d'une assignation peut être calculée. On donne le nom de *Collapsed Gibbs Sampling* (Porteous *et al.* (2008)) à cette forme simplifiée du *Gibbs Sampling*, cette dernière bénéficiant de la propriété de distribution conjuguée permettant d'outrepasser l'approximation numérique des intégrales et de se concentrer seulement sur la comptabilisation des observations.

Maintenant que la théorie générale sur le CGS a été présentée, nous allons expliquer le processus concret derrière l'algorithme. Pour ce faire on doit introduire les deux matrices de fréquence suivantes :

— C_{WK} : Une matrice de dimension $[N_W \times N_K]$ qui compte le nombre d'occurrences qu'un mot w est assigné à un sujet k.

— C_{DK} : Une matrice de dimension $[N_D \times N_K]$ qui compte le nombre d'occurrences qu'un document d est assigné à un sujet k

Ces deux matrices constituent la statistique suffisante au CGS. Le processus itératif de cet algorithme peut être écrit comme suit :

1. Initialisation aléatoire de la matrice d'assignation des sujets Z.
2. Incrémentation des matrices de fréquence C_{WK} et C_{DK} associée avec l'initialisation aléatoire
3. Initialiser le mot courant à $i = 1$ et le document courant à $j = 1$.

4. Décrémenter les colonnes de C_{WK} et C_{DK} correspondant au mot i et au document j.
5. Calculer la probabilité que le mot i appartienne à chaque sujet potentiel k_i avec l'équation 3.1 ci-dessous.
6. Normaliser les probabilités obtenues pour former une distribution.
7. Échantillionner avec une multinomiale la nouvelle assignation du mot i selon cette distribution.
8. Incrémenter les matrices de fréquence à la position associée à la nouvelle assignation.
9. Incrémenter i si on ne se trouve pas à la fin du document. Si on se trouve à la fin du document, incrémenter j et réinitialiser i à 1.
10. Répéter les étapes 4 à 9 pour chaque mot du corpus.
11. Répéter les étapes 3 à 10 pour chaque itération
12. Estimer les paramètres θ et ϕ avec les équations 3.2 et 3.3 ci-dessous.

De cette manière, il est possible d'estimer les paramètres θ et ϕ avec le CGS à partir des observations contenues dans la matrice de corpus X. Explicitement, la probabilité que le mot courant appartienne à chaque sujet potentiel k_i (étape 5) se calcule comme suit :

$$P(z_{k_i}|Z^{(-i)}, \alpha, \beta, W) = \frac{C_{\mathrm{DK}}(d, k_i) + \alpha(\mathbf{1})}{\sum_{k=1}^{N_K} C_{\mathrm{DK}}(d, k) + \alpha(\mathbf{2})} \times \frac{C_{\mathrm{WK}}(X(d, w), k_i) + \beta(\mathbf{3})}{\sum_{w=1}^{N_W} C_{\mathrm{WK}}(w, k) + \beta(\mathbf{4})} \tag{3.1}$$

Où :

1. Nombre d'occurrences que des mots du document d ont été assignés au sujet k_i.
2. Nombre de mots dans le document d.
3. Nombre d'occurrences que le mot w du document d a été assigné au sujet k_i.
4. Nombre de fois que le sujet k_i a été assigné au total.

En résumé, le premier terme témoigne de l'affinité relative du document d au sujet k_i tandis que le second terme témoigne de l'affinité relative du mot w du document d au sujet k_i. Notons que les hyperparamètres de Dirichlet sont utilisés à des fins de normalisation. On remarque donc que le CGS exprime la probabilité d'assignation de manière très intuitive.

En ce qui concerne l'étape 12 du processus itératif, on estime les paramètres θ et ϕ par :

$$\theta = \frac{C_{\mathrm{DK}}(d, k) + \alpha}{\sum_{k=1}^{N_K} C_{\mathrm{DK}}(d, k) + \alpha} \tag{3.2}$$

$$\phi = \frac{C_{\mathrm{WK}}(X(d, w), k) + \beta}{\sum_{w=1}^{N_W} C_{\mathrm{WK}}(w, k) + \beta} \tag{3.3}$$

Alors, on remarque que la probabilité de l'assignation d'un mot w provenant d'un document d à un sujet k se traduit comme la multiplication de $\theta(d, k)$ avec $\phi(k, w)$.

Le CGS comporte plusieurs avantages par rapport à l'inférence variationnelle pour le présent mémoire. D'abord, les mathématiques de son implémentation sont beaucoup plus simples, ce qui le rend aisément modifiable. De plus, le CGS étant une méthode MCMC qui n'admet aucun modèle prédéfini, la chaîne de Markov doit converger exactement sur la postérieure après une infinité d'itérations, ce qui fait en sorte que les estimations produites par le CGS comportent un faible biais et une variance élevée. Au contraire, l'inférence variationnelle est une méthode qui admet une routine d'optimisation plus efficace, mais au prix de certaines restrictions : un modèle, limité aux familles de distributions découlant de la paramétrisation, doit être admis. Ceci fait en sorte que les estimations produites par l'inférence variationnelle comportent une faible variance et un biais élevé. Alors, bien qu'il n'est pas possible d'obtenir une convergence exacte avec le CGS, on s'attend à avoir des résultats davantage précis avec cet algorithme qu'avec l'inférence variationnelle si on procède à un nombre suffisant d'itérations. Or, le fait de réduire l'erreur due à la méthode d'inférence peut s'avérer utile considérant notre objectif de développement d'une extension à LDA, cette dernière étant un futur algorithme dont les performances sont évidemment inconnues. Cependant, le temps d'exécution du CGS est plus dépendant du nombre de données que le temps d'exécution de l'inférence variationnelle, ce qui représente le principal avantage de cette dernière. Par conséquent, l'inférence variationnelle serait à prioriser dans un contexte réel où les nombres de sujets et de documents par corpus sont énormes. Toutefois, pour ce qui est du présent contexte académique, on émet l'hypothèse le CGS est un meilleur algorithme que l'inférence variationnelle.

3.2 Validation du modèle LDA classique avec des données synthétiques

La performance de LDA pour retrouver les sujets traités dans des corpus connus a déjà été exhaustivement étudiée dans plusieurs articles (Mukherjee et Blei (2009), Liu *et al.* (2011), Canini *et al.* (2009)). Par ailleurs, un bon nombre d'ensembles de données illustrant un regroupement de sujets existent sur le web. Or, peu de ces ensembles de données font interagir l'expertise des auteurs dans la segmentation des sujets. Cela pose problème, puisqu'on ne possède pas de *ground truth* pour tester notre futur algorithme d'inférence de l'expertise. De plus, il est nécessaire de tester le cadre de validation sur un modèle éprouvé, LDA classique en l'occurrence, avant de l'appliquer à notre modèle d'expertise.

3.2.1 Principes derrière le cadre de validation

Les bases de données faisant intervenir l'expertise des auteurs et la segmentation du corpus en sujets se font très rares sur le web. La base de données s'approchant le plus de ce que nous aurions besoin dans ce projet est celle du TREC que nous avons précédemment abordé. Cette base de données fait intervenir du texte contenu dans des courriels avec une forme d'expertise représentée par un graphe de citations. Cependant, bien que la base de données du TREC mette en relation les deux facteurs recherchés, à savoir les données textuelles et l'expertise des auteurs, celle-ci n'est pas favorable au but convoité. En effet, il serait difficile d'utiliser un modèle contenu, tel que LDA, pour inférer un nouveau paramètre lié à l'expertise, puisque l'expertise modélisée par les données du TREC n'est pas explicite. On devrait plutôt avoir recours à un module graphique pour déterminer cette expertise à partir des citations. Aussi, considérant que les sujets traités par les courriels de la base de données sont éclectiques et vagues, on conclut que la base de données du TREC n'est pas à prioriser pour le projet.

Afin de remédier à cette situation, l'option de créer notre propre base de données a été explorée. Par exemple, on aurait pu agréger des manuels scolaires portant sur différents sujets de niveaux primaires, secondaires et universitaires. Une autre option aurait pu être de regrouper des articles scientifiques de type grand public et de type plus spécialisés. On aurait aussi pu avoir recours aux citations pour déterminer les niveaux d'expertise des articles en émettant l'hypothèse qu'un article cité possède un niveau de technicalité moins grand que l'article qui le cite. Or, bien que cette solution pourrait être étudiée davantage dans un futur travail, nous avons abandonné l'idée de créer notre propre base de données pour deux principales raisons. D'une part, le fait d'agréger manuellement une quantité considérable d'articles scientifiques et de manuels scolaires portant sur des sujets divers et ayant des niveaux de technicalité différents, en plus du fait de traiter ces données textuelles afin de les rendre exploitables par l'algorithme d'inférence, constitue une tâche des plus laborieuses. D'autre part, le fait de définir une hiérarchie d'expertise parmi les documents réunis s'avère un exercice arbitraire considérant que nous ne possédons pas l'expertise requise pour déterminer si, par exemple, un article de médecine A est plus technique qu'un article de médecine B. Alors, il serait d'autant plus arbitraire d'affecter un score d'expertise gradué sur une échelle arbitraire et de comparer celui-ci à un *ground truth* tout autant arbitraire. De ce fait, il serait difficile de juger la performance d'un nouvel algorithme si nous n'avons pas de certitude sur les résultats qu'un algorithme performant devrait produire.

Dans la littérature, les méthodes pour déterminer les performances de LDA sont très similaires. D'abord on entraîne le modèle sur une base de données réelle. Ensuite, on calcule la

perplexité ou le score de cohérence sur les données (voir 2.3.1). Plus la perplexité est basse ou plus le score de cohérence est élevé et plus la performance du modèle LDA est bonne. Aussi, on peut procéder à un jugement humain qui teste si la segmentation des sujets faite par l'algorithme est pertinente.

Or, nous allons procéder différemment dans le cadre de ce mémoire. En effet, dans le but d'obtenir une métrique de comparaison fiable pour un nouvel algorithme, nous aurons plutôt recours à une génération synthétique des données. La principale motivation expliquant notre choix d'utiliser des données synthétiques est qu'il est préférable de tester une nouvelle méthode dans un environnement contrôlé où on possède le *ground truth* qu'on peut faire varier à notre guise en créant des cas de figure adaptés à diverses situations. Si l'algorithme développé ne fonctionne pas sur des données synthétiques, alors il est inutile de le tester sur des données réelles. Pour le modèle LDA classique, cette génération synthétique consiste à créer un corpus issu des distributions θ et ϕ connues. Ensuite, on entraîne LDA sur le corpus généré et on obtient les paramètres θ et ϕ inférés par le CGS ou l'inférence variationnelle. Enfin, afin d'évaluer la performance du modèle, on calcule certaines métriques de comparaison entre les θ et ϕ générés et ces mêmes paramètres obtenus par inférence. Si la comparaison est bonne, on dit que le modèle est performant. Toutefois, il est important de noter que des problèmes d'alignement surviendront entre les distributions générées et inférées. Étant donné que LDA est un modèle d'apprentissage non supervisé, les sujets ne sont pas déterminés *a priori*. Alors, l'ordre des distributions inférées ne concordera pas nécessairement avec celui des distributions générées et on devra trouver l'alignement correct en post-traitement (voir 3.2.2).

Dans un premier temps, on évaluera la performance du modèle LDA classique en utilisant ce cadre de validation. Bien que la performance de LDA classique soit éprouvée par la littérature, il nous sera utile de la déterminer avec ce cadre, car cette performance constituera un point de comparaison important lorsque viendra le temps d'évaluer, de par ce même cadre, le modèle LDA étendu faisant intervenir l'expertise. De plus, on profite dans ce chapitre de ce cadre de validation pour explorer les performances de LDA classique par rapport à la méthode d'inférence utilisée, aux hyperparamètres α et β employés lors de la génération, au nombre de sujets latents spécifiés et à la similitude des données générées avec les lois de Zipf et Mandelbrot.

3.2.2 Alignement des sujets

Le problème le plus important du cadre de validation est le choix de l'alignement des sujets inférés. En effet, prenons le cas d'une génération à 3 sujets et affectons les étiquettes

$\{A, B, C\}$ à ces sujets. Puisque l'ordre n'est pas spécifié, il n'y a rien dans le modèle LDA qui contre-indique une inférence de type $\{B, A, C\}$ ou $\{C, B, A\}$ par exemple. En somme, si aucune manipulation suite à l'inférence n'est appliquée, le bon alignement entre les distributions générées et inférées serait une coïncidence. Il est important de remédier à ce problème, car les métriques de comparaison présentées à la section 3.2.3 n'ont de sens que si les distributions sont alignées.

Certaines méthodes sophistiquées existent pour procéder à ce type d'alignement, telle que l'analyse de Procrustres présentée dans Gower (1975). Or, cet algorithme a été testé et aucun résultat probant n'a été obtenu. Après quelques autres tentatives infructueuses, nous avons dû avoir recours à une méthode moins étayée. Cette méthode consiste à calculer la corrélation, la distance cosinus ainsi que la divergence KL pour tous les alignements possibles, et ce, pour θ et pour ϕ. Ces alignements seront obtenus en permutant les lignes de ϕ et les colonnes de θ, puisqu'il s'agit de la dimension associée aux sujets. On fera autant de permutations nécessaires pour générer toutes les combinaisons possibles. Ensuite, pour chaque métrique de chaque paramètre, on sélectionne l'alignement dont la métrique associée témoigne de la meilleure performance. Finalement, on compare les 6 alignements (2 paramètres et 3 métriques) afin de valider que l'alignement optimal et cohérent pour toutes les métriques. Si on observe une incohérence, on retourne une erreur.

Malgré le fait que notre méthode ne soit pas efficace d'un point de vue de calcul informatique, elle donne de bons résultats. Ceux-ci seront montrés à la section 3.5 où on présente les statistiques du modèle LDA final.

3.2.3 Métriques de comparaison

Dans le but de comparer les distributions générées synthétiquement avec celles inférées par les modèles qu'on veut tester, on utilise trois métriques de comparaison : le coefficient de corrélation de Pearson, la distance cosinus et la divergence KL.

Coefficient de corrélation de Pearson

D'abord, le coefficient de corrélation de Pearson sur un échantillon sera calculé. Étant donné qu'il y a autant de distributions dans ϕ qu'il y a de sujets et qu'il y a autant de distributions dans θ qu'il y a de documents, le coefficient de corrélation présenté sera la moyenne des coefficients obtenus entre les rangées du même indice de chaque matrice. Nous allons dénoter les distributions générées par ϕ et θ et les distributions inférées par $\hat{\phi}$ et $\hat{\theta}$. Alors, l'expression

du coefficient de corrélation r, que l'on exprime sous la forme d'une distance Dr, pour une rangée est :

$$Dr_{\phi_k} = 1 - \frac{\sum_{w=1}^{N_W} \left(\phi_{k,w} - \overline{\phi_k}\right)\left(\hat{\phi}_{k,w} - \overline{\hat{\phi}_k}\right)}{\sqrt{\sum_{w=1}^{N_W} \left(\phi_{k,w} - \overline{\phi_k}\right)^2} \sqrt{\sum_{w=1}^{N_W} \left(\hat{\phi}_{k,w} - \overline{\hat{\phi}_k}\right)^2}}$$

$$Dr_{\theta_d} = 1 - \frac{\sum_{k=1}^{N_K} \left(\theta_{d,k} - \overline{\theta_d}\right)\left(\hat{\theta}_{d,k} - \overline{\hat{\theta}_d}\right)}{\sqrt{\sum_{k=1}^{N_K} \left(\theta_{d,k} - \overline{\theta_d}\right)^2} \sqrt{\sum_{k=1}^{N_K} \left(\hat{\theta}_{d,k} - \overline{\hat{\theta}_d}\right)^2}}$$

Et le coefficient moyen pour chaque distribution est obtenu par :

$$Dr_{\phi} = \frac{\sum_{k=1}^{N_K} r_{\phi_k}}{N_K}$$

$$Dr_{\theta} = \frac{\sum_{d=1}^{N_D} r_{\theta_d}}{N_D}$$

Donc, si on prend la distribution ϕ par exemple, on calcule le coefficient de corrélation de Pearson entre chaque rangée de la matrice générée et celle inférée. De cette manière, on obtiendra N_K coefficients, soit un pour chaque sujet. Enfin, on fait la moyenne de ces coefficients pour déterminer une représentation globale de la performance du modèle. Notons que plus Dr est petite et meilleure est la performance du modèle.

Distance cosinus

Pour la distance cosinus, on procède de manière analogue au calcul du coefficient de corrélation. On exprime d'abord la distance entre chaque rangée des matrices ϕ et θ :

$$D\cos_{\phi_k} = 1 - \frac{\phi_k \cdot \hat{\phi}_k}{\sqrt{\sum_{w=1}^{N_W} \phi_{k,w}^2} \sqrt{\sum_{w=1}^{N_W} \hat{\phi}_{k,w}^2}}$$

$$D\cos_{\theta_d} = 1 - \frac{\theta_d \cdot \hat{\theta}_d}{\sqrt{\sum_{k=1}^{N_K} \theta_{d,k}^2} \sqrt{\sum_{k=1}^{N_K} \hat{\theta}_{d,k}^2}}$$

Et la distance moyenne pour chaque distribution est obtenue par :

$$D\cos_{\phi} = \frac{\sum_{k=1}^{N_K} \cos_{\phi_k}}{N_K}$$

$$D\cos_{\theta} = \frac{\sum_{d=1}^{N_D} \cos_{\theta_d}}{N_D}$$

Notons que plus la distance cosinus est petite et meilleure est la performance du modèle.

Divergence KL

On termine cette section par la divergence KL, puisqu'il s'agit de la métrique qui sera le plus souvent utilisée dans le mémoire. En effet, la divergence KL est une métrique faite sur mesure pour les distributions de probabilités. En fait, la divergence KL représente l'espérance de la différence logarithmique entre deux distributions. Pour ce calcul, on doit définir une distribution caractérisant les observations et une distribution caractérisant la modélisation. Dans notre cas, les distributions caractérisant les observations sont ϕ et θ tandis que les distributions caractérisant la modélisation sont $\hat{\phi}$ et $\hat{\theta}$. Dans le contexte du cadre de validation, ϕ et θ seront les distributions générées alors que $\hat{\phi}$ et $\hat{\theta}$ seront les distributions inférées par LDA. En utilisant un processus similaire aux deux autres métriques, on exprime d'abord la divergence KL (pour des distributions discrètes) pour chaque rangée des matrices ϕ et θ :

$$D_{\mathrm{KL}}(\phi_k \parallel \hat{\phi}_k) = \sum_{w=1}^{N_W} \phi_{k,w} \log\left(\frac{\phi_{k,w}}{\hat{\phi}_{k,w}}\right)$$

$$D_{\mathrm{KL}}(\theta_d \parallel \hat{\theta}_d) = \sum_{k=1}^{N_K} \theta_{d,k} \log\left(\frac{\theta_{d,k}}{\hat{\theta}_{d,k}}\right)$$

Et la divergence KL pour chaque distribution est obtenue par :

$$D_{\mathrm{KL}}(\phi \parallel \hat{\phi}) = \frac{\sum_{k=1}^{N_K} D_{\mathrm{KL}}(\phi_k \parallel \hat{\phi}_k)}{N_K}$$

$$D_{\mathrm{KL}}(\theta \parallel \hat{\theta}) = \frac{\sum_{d=1}^{N_D} D_{\mathrm{KL}}(\theta_d \parallel \hat{\theta}_d)}{N_D}$$

Notons que plus la divergence KL est petite et meilleure est la performance du modèle.

3.2.4 Paramètres pour la génération et l'inférence

La génération d'un corpus se fait telle que décrite par l'algorithme présenté à la figure 3.1. Voici les paramètres pour la génération :

- $N_V = 1000$.
- $N_D = 100$.
- $N_{\mathrm{WD}} = 100$.
- Nous générerons des corpus à $N_K = 3$ et $N_K = 6$ pour évaluer la performance de LDA pour différentes segmentations des sujets.
- Les hyperparamètres α et β seront variables, puisqu'on veut tester la performance de LDA pour différentes combinaisons de ces hyperparamètres. Les valeurs que prendront

α et β seront [0.01, 0.1, 0.5, 0.7, 1.0, 1.5, 2, 3, 5]. Notons que les distributions de Dirichlet seront symétriques, ce qui signifie que tous les N_K α_i et les N_W β_i auront la même valeur pour une combinaison donnée.

En ce qui concerne l'inférence, on distingue deux variantes de LDA classique : le modèle Gensim et le modèle CGS. Le modèle Gensim, développé par Hoffman *et al.* (2010), est issu de la librairie Gensim de Python qui emploie une forme d'inférence variationnelle optimisée pour les corpus à nombre élevé de sujets et de documents. Voici les paramètres du modèle Gensim utilisé :

- random state = 100.
- update every = 1.
- chunksize = 100.
- passes = 10.
- alpha = automatic.
- minimum probability = 0.

Le modèle CGS est un algorithme LDA développé dans le cadre du mémoire en Python qui utilise le CGS comme méthode d'inférence. On utilise le code maison plutôt que le module CGS de la librairie Gensim afin d'obtenir une meilleure compréhension de l'algorithme, ce qui permettra de faciliter l'apport de modifications lorsque viendra le temps d'élaborer le modèle LDA étendu. Suite à une analyse de convergence, il a été conclu que 50 itérations pour le CGS étaient le meilleur compromis entre une inférence précise et un temps d'exécution bas.

3.3 Analyse des méthodes d'inférence en fonction des hyperparamètres de Dirichlet

Cette section et les deux suivantes présentent chacune une expérience afin d'évaluer le comportement du cadre de validation du modèle LDA classique mis en place. D'abord, on évalue la puissance de l'inférence sur un corpus généré par des hyperparamètres α et β variables.

L'objectif de cette expérience est de déterminer la divergence KL entre les distributions ϕ et θ créées durant le processus de génération des données et celles inférées à partir de ces données, ce qui correspond à notre métrique de performance. On veut évaluer cette performance pour différentes combinaisons d'hyperparamètres α et β utilisés lors de la génération des données synthétiques. On veut aussi évaluer cette performance en fonction du modèle pour l'inférence : soit avec le modèle Gensim utilisant l'inférence variationnelle ou avec le modèle CGS utilisant une inférence par CGS. De plus, on veut évaluer la performance en fonction du nombre de sujets latents spécifiés : soit $N_K = 3$ ou $N_K = 6$. Notons que ce n'est

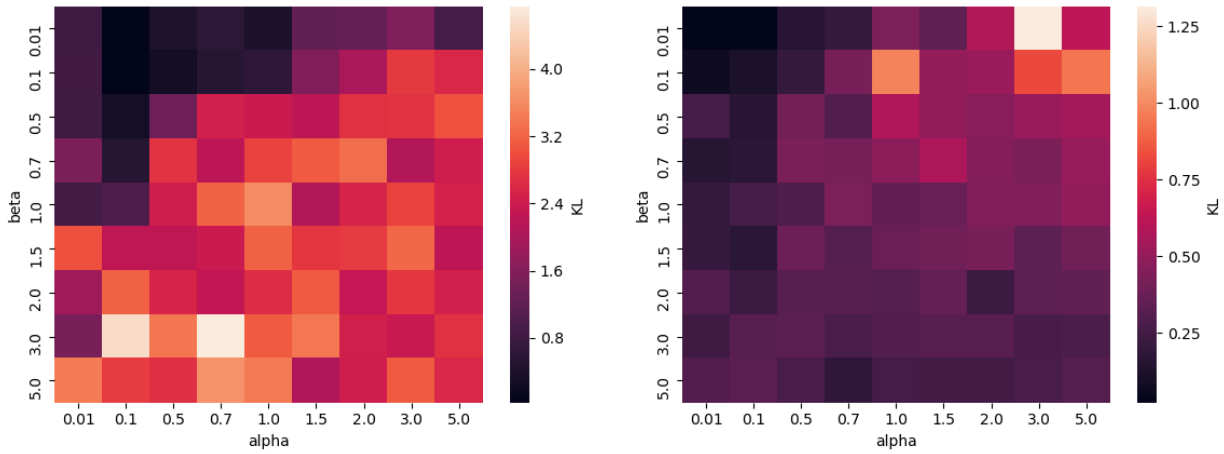
que la divergence KL qui sera calculée pour cette expérience et celles qui suivront ; on ne montre pas la corrélation ni la distance cosinus, puisque ces métriques ne nous apportent pas d'informations supplémentaires pour ce que l'on veut démontrer (le comportement des résultats est analogue à celui observé avec la divergence KL). On présentera la corrélation et la distance cosinus à la section 3.5 où on résume les performances du modèle LDA choisi.

Pour réaliser cette expérience, on utilise les paramètres pour la génération et l'inférence présentés en section 3.2.4. Les combinaisons de α et β employées pour la génération sont comprises dans l'intervalle [0.01, 0.1, 0.5, 0.7, 1.0, 1.5, 2, 3, 5] et on pose symétriques les distributions de Dirichlet, puisque nous ne possédons aucune information *a priori* qui favorise un mot ou un sujet en particulier. Par exemple, les hyperparamètres de la première combinaison testée auront comme valeurs $\alpha_i = 0.01$ et les $\beta_i = 0.01$. Ceux de la deuxième combinaison seront $\alpha_i = 0.1$ et $\beta_i = 0.01$ et ainsi de suite. Voici le processus suivi pour cette expérience :

1. Initialiser l'indicateur des combinaisons à $i = 1$ (la combinaison $\alpha = 0.01, \beta = 0.01$).
2. Générer le corpus pour la combinaison d'hyperparamètres courante et stocker les θ et ϕ.
3. Appliquer le modèle Gensim sur le corpus généré.
4. Appliquer le modèle CGS sur le corpus généré.
5. Stocker les $\hat{\theta}$ et les $\hat{\phi}$ pour les deux modèles.
6. Calculer les divergences KL entre les paramètres générés et ceux inférés.
7. Répéter 10 fois les étapes 2 à 6 pour réduire la variance des divergences KL obtenues.
8. Calculer les divergences KL moyennes pour les 10 corpus.
9. Incrémenter i.
10. Répéter les étapes 2 à 9 jusqu'à ce que toutes les 81 combinaisons soient testées.

Cette expérience a d'abord été réalisée avec une segmentation à 3 sujets et les résultats sont présentés sous la forme d'une *heatmaps* où chaque carré représente la divergence KL (obtenue à l'étape 8 du processus) pour une combinaison donnée. Notons que plus la couleur du carré est foncée, plus petite est la divergence KL et meilleure est la performance du modèle.

Voici les *heatmaps* pour les distributions θ et ϕ obtenues avec le modèle Gensim :

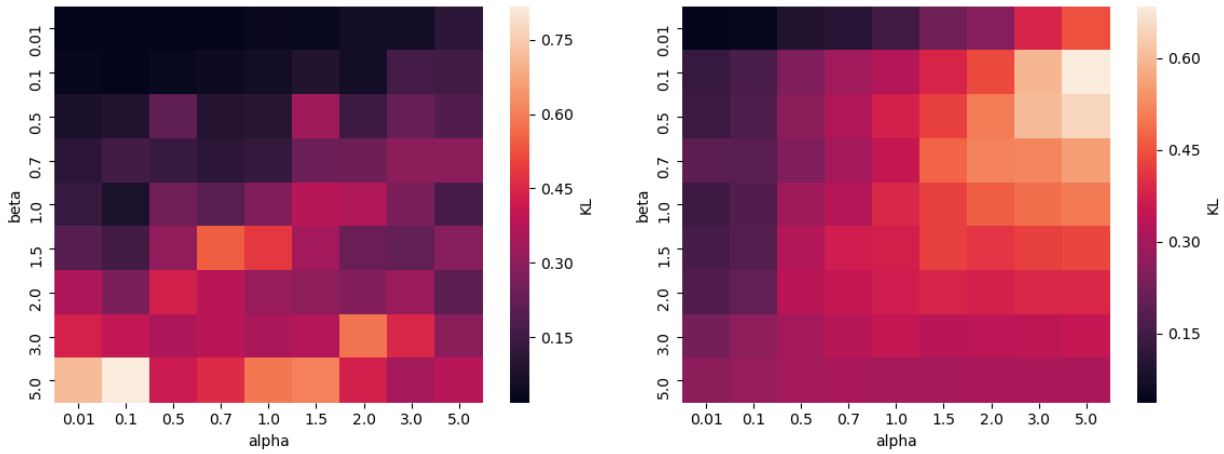


θ ϕ

Figure 3.7 Les divergences KL entre les hyperparamètres générés et inférés pour le modèle Gensim ($N_K = 3$ sujets)

Notons que l'échelle est différente pour les deux sous-figures (représentée à droite de sa sous-figure respective). On remarque que le modèle Gensim est meilleur pour prédire ϕ que θ. De plus, on constate que l'inférence de ϕ est beaucoup plus stable selon les hyperparamètres utilisés que celle de θ. Pour ϕ, on voit que la performance est meilleure lorsque α et β sont petits, c'est-à-dire lorsqu'un mot est fortement priorisé dans le vocabulaire et lorsqu'un sujet est aussi fortement priorisé dans les documents. On remarque aussi que l'inférence a davantage de difficulté à retrouver les paramètres lorsque β est petit et α est grand, c'est-à-dire lorsque la distribution de mots par sujet est fortement déphasée vers certains mots en particulier et que la distribution de sujets par document est presque uniforme. Pour θ, les performances sont meilleures lorsque les hyperparamètres sont petits, et on constate que la performance devient plus faible lorsque β augmente. En somme, étant donné que $\theta \sim \text{Dir}(\alpha)$ et que $\phi \sim \text{Dir}(\beta)$, on conclut que la variation de l'hyperparamètre de la croyance *a priori* de la multinomiale n'affecte presque pas la qualité de l'inférence de cette multinomiale. On conclut aussi que si θ et ϕ sont très élevés, il devient impossible de départager les sujets, puisque les distributions de mots et de sujets deviennent uniformes, donc indiscernables.

Voici maintenant les *heatmaps* pour le modèle CGS :



θ ϕ

Figure 3.9 Les divergences KL entre les hyperparamètres générés et inférés pour le modèle CGS ($N_K = 3$ sujets)

En ce qui concerne le modèle CGS, le comportement de l'inférence est similaire au modèle Gensim et la tendance observée par rapport au lien entre les hyperparamètres et les multinomiales associées est encore plus marquée pour le CGS. Or, contrairement à Gensim, la performance de l'inférence de θ s'apparente à celle de ϕ.

Comparons maintenant les performances du modèle Gensim et CGS pour θ sur la même échelle :

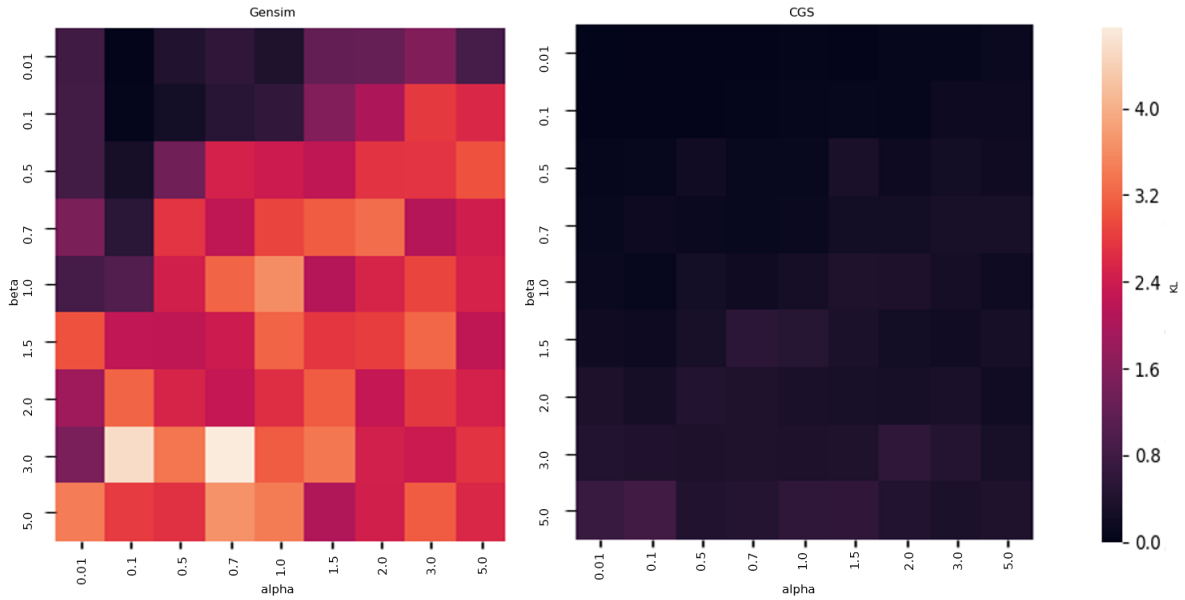


Figure 3.10 Divergence KL entre les méthodes d'inférence pour θ ($N_K = 3$ sujets)

Et pour ϕ :

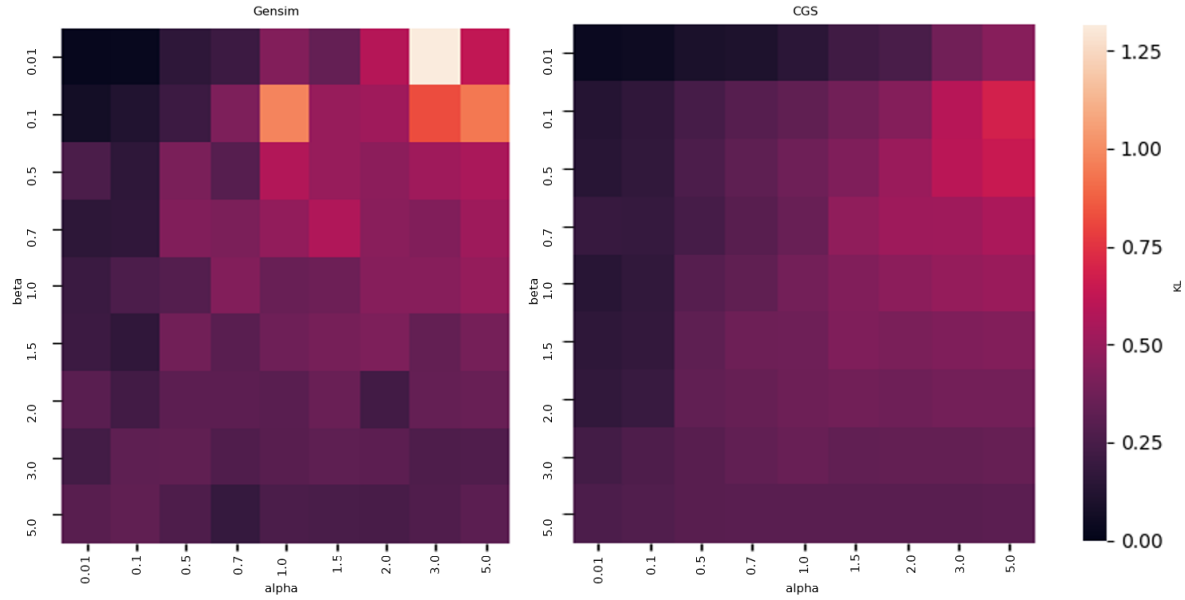


Figure 3.11 Divergence KL entre les méthodes d'inférence pour ϕ ($N_K = 3$ sujets)

On voit ici que la performance de CGS est nettement supérieure à celle de Gensim et c'est ce

qui était attendu par la théorie. Cette performance est davantage similaire pour l'inférence de ϕ que de θ où l'avantage du CGS est marqué.

On effectue la même expérience en spécifiant 6 sujets latents au lieu de 3. En premier lieu, on analysera si les différences entre les méthodes d'inférence sont les mêmes pour cette expérience. En second lieu, on analysera les différences pour chaque inférence entre l'expérience à 3 sujets et celle à 6 sujets. D'abord, voici la comparaison entre le modèle Gensim et CGS pour θ sur la même échelle :

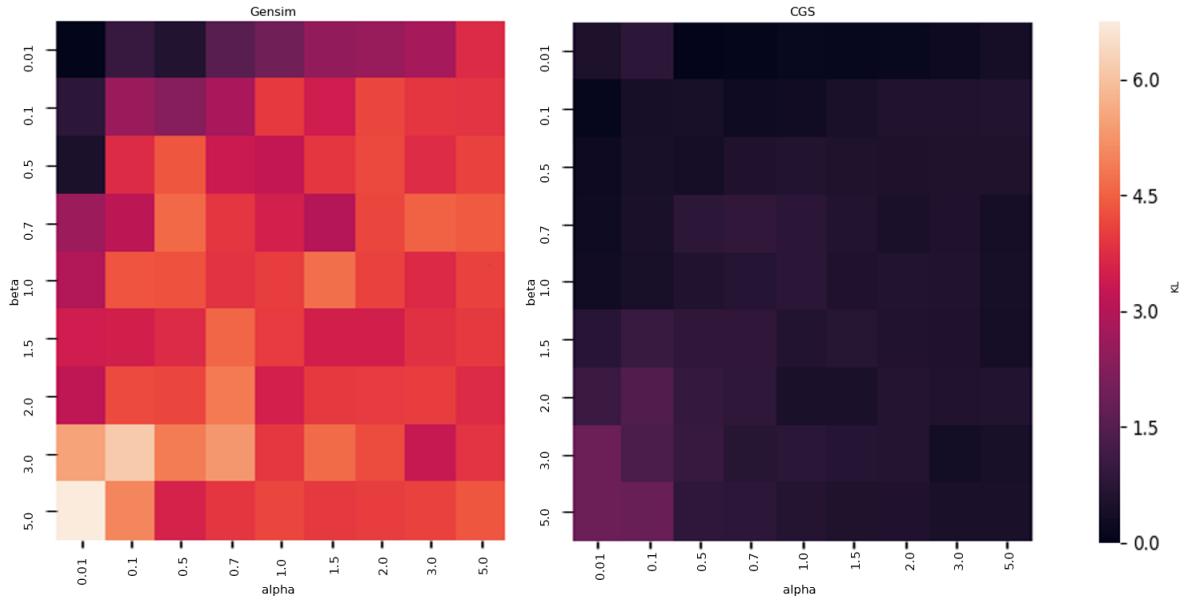


Figure 3.12 Divergence KL entre les méthodes d'inférence pour θ ($N_K = 6$ sujets)

Et pour ϕ :

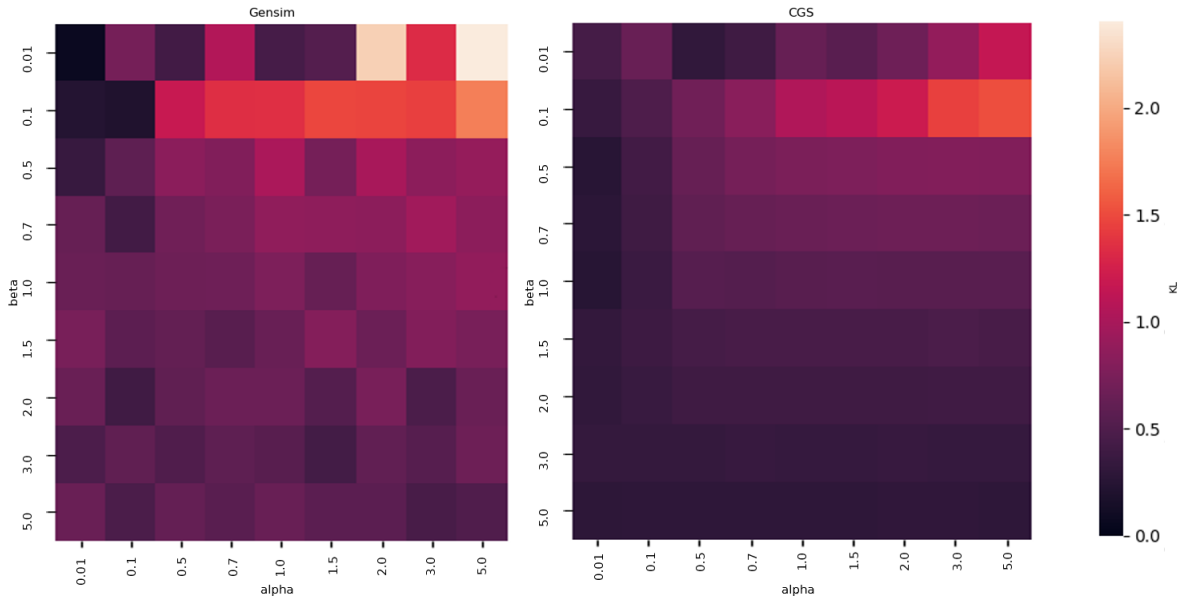
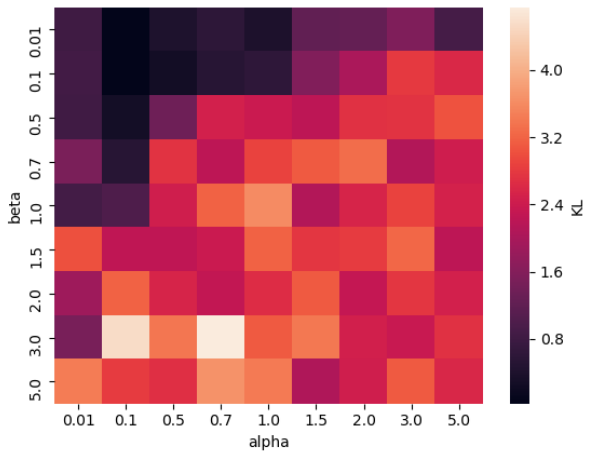


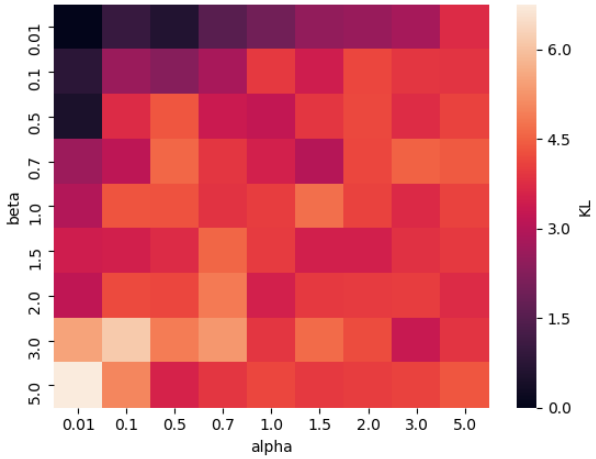
Figure 3.13 Divergence KL entre les méthodes d'inférence pour ϕ ($N_K = 6$ sujets)

On observe donc un comportement quasi identique par rapport à l'expérience à 3 sujets (figure 3.10 et 3.11). La seule différence peut être constatée en $\beta = 0.1$ pour l'inférence de ϕ (figure 3.13) où la performance relative semble être beaucoup moins bonne pour les modèles à 6 sujets.

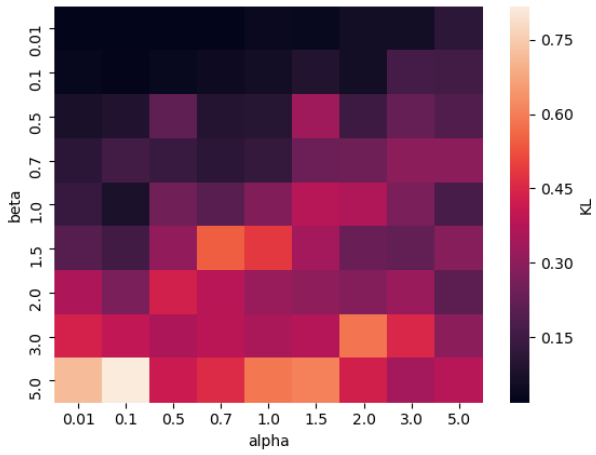
Ensuite, chaque inférence à 3 sujets est comparée avec son inférence à 6 sujets. D'abord pour θ :



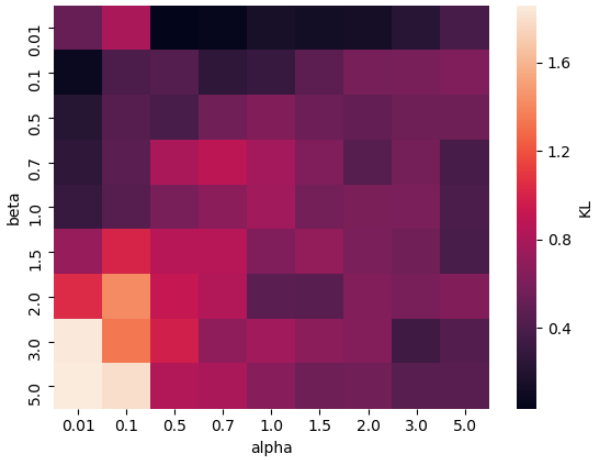
Gensim : $N_K = 3$



Gensim : $N_K = 6$



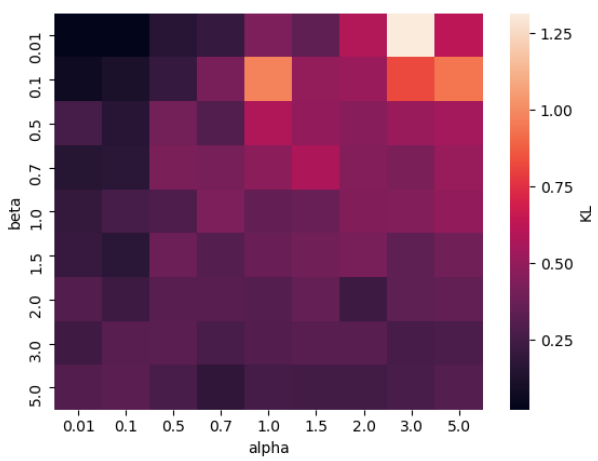
CGS : $N_K = 3$



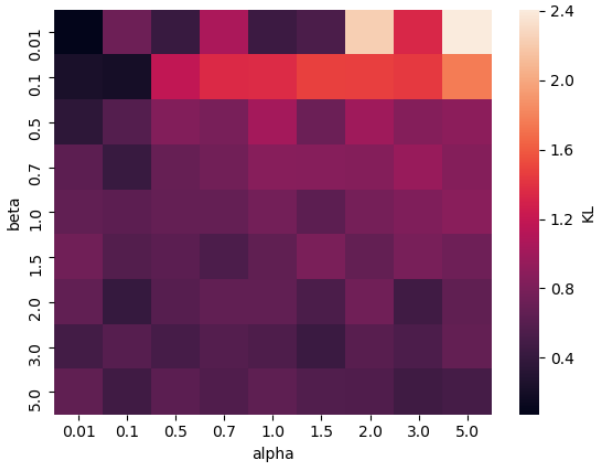
CGS : $N_K = 6$

Figure 3.15 Divergence KL entre les inférences à 3 sujets (gauche) et celles à 6 sujets (droite) pour θ

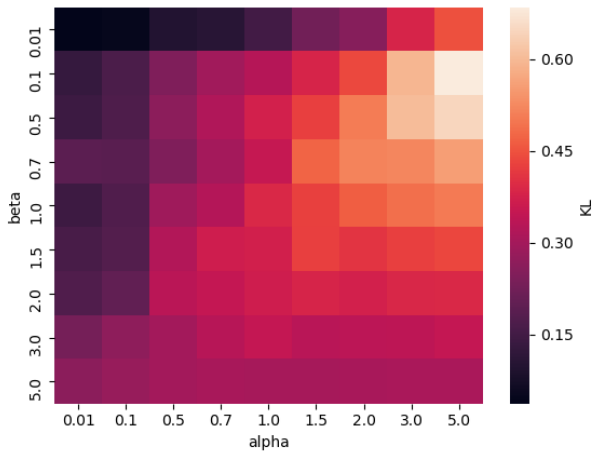
Ensuite pour ϕ :



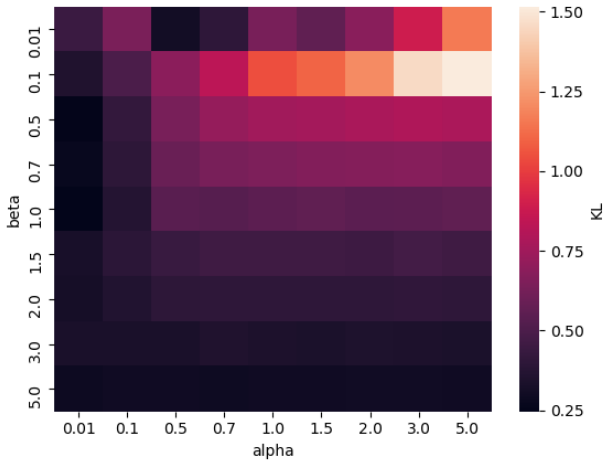
Gensim : $N_K = 3$



Gensim : $N_K = 6$



CGS : $N_K = 3$



CGS : $N_K = 6$

Figure 3.17 Divergence KL entre les inférences à 3 sujets (gauche) et celles à 6 sujets (droite) pour ϕ

Il est important de noter que les échelles sont différentes pour toutes les *heatmaps* et c'est surtout le comportement général qu'on analyse. Pour θ, ce comportement est très similaire pour l'expérience à 3 sujets et celle à 6 sujets. Pour ϕ, le comportement est similaire pour les deux expériences aussi, mais les mauvaises performances à $\beta = 0.1$ sont davantage marquées pour les modèles à 6 sujets.

Le tableau suivant montre les valeurs suivies de l'emplacement (α, β) des minimums et des maximums pour chaque expérience. De plus, on calcule aussi les moyennes $\overline{\theta}$ et $\overline{\phi}$ des paramètres obtenus pour chaque combinaison :

Tableau 3.1 Synthèse des divergences KL de l'analyse des hyperparamètres

Modèles	θ_{max}	θ_{min}	ϕ_{max}	ϕ_{min}	$\overline{\theta}$	$\overline{\phi}$
Gensim : $N_K = 3$	4.74 $(0.07, 3)$	0.030 $(0.1, 0.01)$	1.31 $(3, 0.01)$	0.023 $(0.01, 0.01)$	2.23	0.37
Gensim : $N_K = 6$	6.75 $(0.01, 5)$	0.0020 $(0.01, 0.01)$	2.41 $(5, 0.01)$	0.073 $(3, 5)$	3.67	0.76
CGS : $N_K = 3$	0.82 $(0.1, 5)$	0.018 $(0.01, 0.01)$	0.68 $(5, 0.1)$	0.037 $(0.01, 0.01)$	0.25	0.33
CGS : $N_K = 6$	1.86 $(0.01, 5)$	0.036 $(0.5, 0.01)$	1.51 $(5, 0.1)$	0.025 $(0.01, 1)$	0.61	0.53

À la lumière de ces résultats, on constate que les emplacements des minimums et maximums sont assez constants pour tous les modèles. De plus, le modèle CGS surpasse systématiquement le modèle Gensim, que ce soit pour l'expérience à 3 sujets ou celle à 6 sujets. Par ailleurs, on n'observe pas d'améliorations relatives claires du côté de l'inférence variationnelle par rapport au CGS à mesure que le nombre de sujets augmente : pour θ, la divergence KL est augmentée de 65% entre les modèles Gensim à 3 et 6 sujets contre 144% entre les modèles CGS tandis que pour ϕ, la divergence KL est augmentée de 105% entre les modèles Gensim contre 61% entre les modèles CGS. Or, le modèle Gensim possède un temps d'exécution plus court que le modèle CGS et cet avantage est plus marqué à mesure que le nombre de sujets augmente. D'une part, le temps d'exécution du modèle Gensim à 3 sujets est de 7 secondes alors que celui de Gensim à 6 sujets est de 8 secondes (augmentation de 14%). D'autre part, le temps d'exécution du modèle CGS à 3 sujets est de 20 secondes alors que celui de CGS à 6 sujets est de 38 secondes (augmentation de 90%).

En résumé, on constate que le cadre de validation concorde avec ce que la littérature prévoit et les résultats qui en découlent sont cohérents. Aussi, il a été possible d'explorer le comportement des méthodes d'inférences en fonction des hyperparamètres définis lors de la génération des données synthétiques et du nombre de sujets spécifiés. Finalement, les performances observées serviront de guide pour l'élaboration du nouveau modèle d'expertise. D'une part, compte tenu du contexte académique du mémoire, on préférera le CGS en raison de la plus grande précision de cette méthode par rapport à l'inférence variationnelle et du fait que

le temps d'exécution ne sera pas un facteur limitant. D'autre part, les *heatmaps* résumant la performance du CGS nous montrent que l'utilisation d'un α et d'un β inférieur à 1 serait bénéfique pour la génération dans le but d'optimiser la performance du nouvel algorithme.

3.4 Analyse de la similitude de la fréquence de mots générée par LDA avec les lois statistiques du langage

La seconde analyse consiste à valider la vraisemblance des corpus de mots générés par le cadre de validation. Nous avons vu dans la section 2.1.1 que des lois statistiques régissent la fréquence des mots qu'on observe dans un texte courant. Bien que cette fréquence soit modulée par le niveau d'expertise d'un auteur, ces lois statistiques constituent une référence de la vraisemblance d'un corpus. Le but de cette analyse sera donc de déterminer la similitude de la fréquence des mots générés par LDA avec la densité de fréquence des mots qu'on devrait théoriquement obtenir selon une loi de Zipf ou de Mandelbrot. En d'autres mots, on veut vérifier si la différence entre la distribution ϕ générée par une Dirichlet et la densité de fréquence de Zipf ou Mandelbrot est faible. On observera en premier lieu la différence entre une distribution ϕ générée par Dirichlet et la meilleure distribution de Zipf ou Mandelbrot qui lisse cette génération. On qualifie de « meilleure distribution » la distribution de Zipf ou Mandelbrot qui minimise la divergence KL avec la distribution générée. Cette différence sera calculée pour certaines combinaisons d'hyperparamètres (α, β) afin d'explorer l'impact de ces derniers sur la création d'un corpus vraisemblable. Les combinaisons étudiées seront les combinaisons de 2 éléments parmi les 9 éléments de l'ensemble suivants : $[0.1, 0.3, 0.5, 0.7, 1, 1.5, 2, 3, 5]$. Nous avons choisi d'omettre le paramètre 0.01, puisque certaines instabilités numériques au niveau de la similarité avec la loi de Zipf ont été observées (divergence KL infinie).

Le calcul de l'erreur entre les fréquences théoriques et celles observées se fera à l'aide d'une validation croisée à $N_F = 10$ replis. Lors d'une validation croisée, on segmente les données en deux groupes : les données d'entraînement et les données de test. Ici, les distributions ϕ générées constituent l'ensemble des données. Pour notre expérience, 70% des données seront considérées comme données d'entraînement tandis que le 30% restant constituera les données de test. L'objectif d'une validation croisée est d'utiliser les données d'entraînement pour estimer les paramètres d'un modèle quelconque. Ensuite, on applique ce modèle sur les données de test et on calcule l'erreur entre la valeur prédite et la valeur observée correspondante.

Pour la loi de Zipf, on cherche à estimer le paramètre c dans la fonction de densité :

$$f_{\text{Zipf}}(R) = \frac{1/R^c}{\sum_{n=1}^{N} 1/n^c}$$

Où f_{Zipf} est la densité de fréquence théorique de Zipf, R est le rang des mots, N est le nombre de mots dans le vocabulaire et c est le paramètre à estimer. La paramétrisation pour la loi de Mandelbrot est la suivante :

$$f_{\text{Mand}}(R) = \frac{1/(R+b)^c}{\sum_{n=1}^{N} 1/(n+b)^c}$$

Où f_{Mand} est la densité de fréquence théorique de Mandelbrot, où N est le nombre de mots dans le vocabulaire et où b et c sont les paramètres à estimer. Afin de faciliter l'optimisation, on fixe le paramètre b de Mandelbrot à 2.7, puisque c'est la valeur classique que l'on retrouve dans la littérature. Le seul paramètre à retrouver sera donc c pour la loi de Mandelbrot. Notons que les ϕ_{Loi} sont l'expression de $f_{\text{Zipf}}(R)$ et de $f_{\text{Mand}}(R)$ une fois le paramètre c optimisé et les ϕ_{Dir} sont les fréquences générées par Dirichlet.

La métrique utilisée pour calculer l'écart entre les distributions est la divergence KL. Étant donné que l'on cherche à quantifier la perte d'information engendrée par l'utilisation d'une distribution de Dirichlet pour approximer une loi du langage, on doit calculer pour chaque repli f :

$$D_{\text{KL-f}} = D_{\text{KL}}(\phi_{\text{Loi}} \parallel \phi_{\text{Dir}})$$

La divergence KL moyenne pour un corpus est obtenue par la moyenne de chaque repli :

$$D_{\text{KL}} = \frac{1}{N_F} \sum_{f=1}^{n_F} D_{\text{KL-f}}$$

Voici le processus pour la validation croisée à 10 replis :

1. Initialiser l'indicateur des combinaisons à 1 (par exemple, la première combinaison est $\alpha = 0.01, \beta = 0.01$).
2. Générer le corpus pour la combinaison d'hyperparamètres courante et stocker les ϕ_{Dir}.
3. Ordonner les ϕ_{Dir} de manière décroissante en fonction du rang (par exemple, le ϕ le plus élevé correspond au rang 1).
4. Sélectionner aléatoirement 70% des couples (R, ϕ_{Dir}) et les stocker dans une matrice de données d'entraînement. Stocker les 30% restant dans une matrice de données de test.

5. À partir des données d'entraînement, utiliser la fonction *curve fit* de *Scipy* et minimiser la divergence KL entre une loi de Zipf/Mandelbrot à paramètre variable et ϕ_{Dir} afin de trouver le paramètre c de f_{Zipf} et f_{Mand} qui optimise la similarité avec ϕ_{Dir}.
6. Calculer les ϕ_{Loi} correspondants aux rangs présents dans la matrice de données de test.
7. Calculer la divergence KL.
8. Répéter 10 fois les étapes 2 à 7 pour réduire la variance des divergences KL obtenues.
9. Calculer les divergences KL moyennes pour les 10 corpus.
10. Incrémenter l'indicateur des combinaisons de 1.
11. Répéter les étapes 2 à 10 jusqu'à ce que toutes les 81 combinaisons soient testées.

Suite à cette expérience, on peut créer une *heatmap* où chaque carré représente la divergence KL, pour une combinaison d'hyperparamètres donnée, entre la meilleure distribution de Zipf ou Mandelbrot et la distribution ϕ générée par Dirichlet. Plus la couleur des carrés est foncée et plus la divergence KL est faible, c'est-à-dire plus la similitude est puissante. D'abord, voici les résultats pour la loi de Zipf :

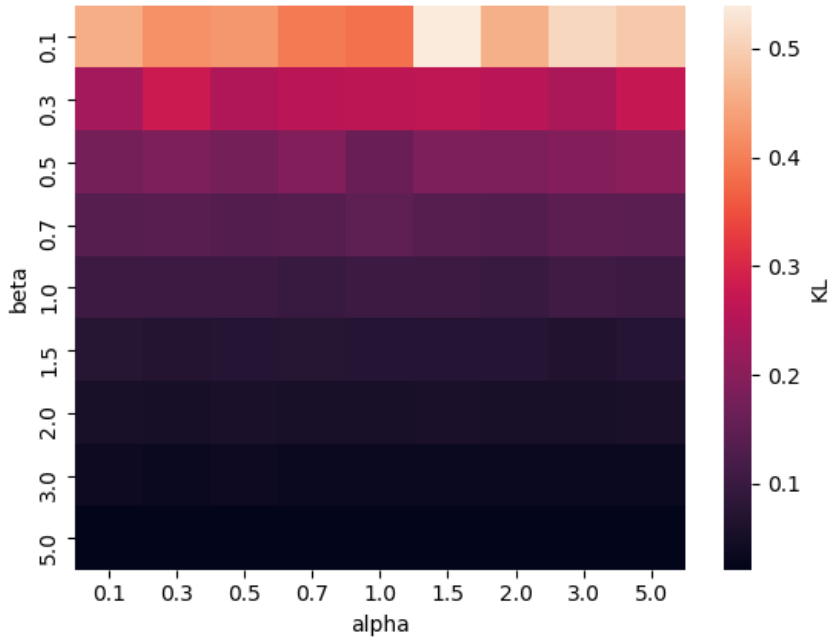


Figure 3.18 Divergences KL entre la meilleure distribution de Zipf et la distribution générée par Dirichlet

Et pour la loi de Mandelbrot :

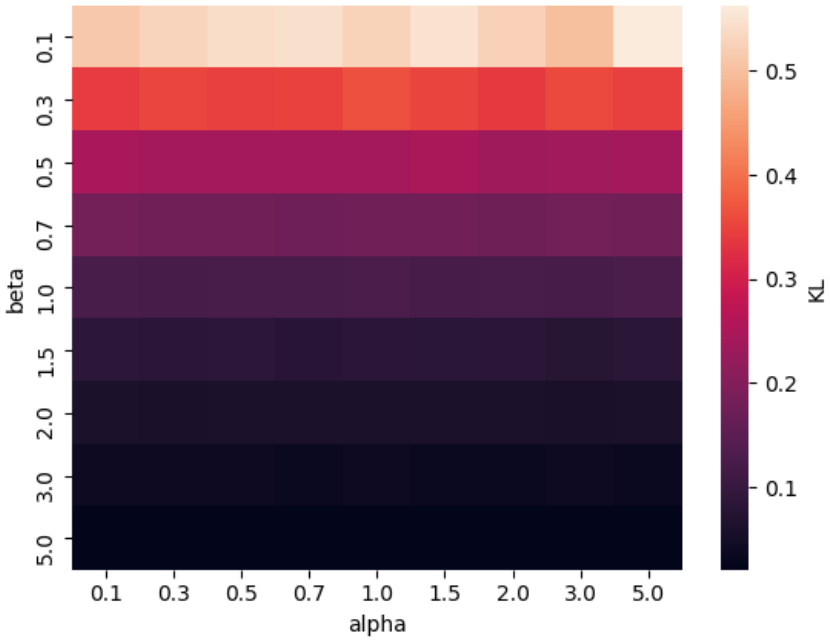


Figure 3.19 Divergences KL entre la meilleure distribution de Mandelbrot et la distribution générée par Dirichlet

Pour finir, on représente ces *heatmaps* sur la même échelle afin de les comparer :

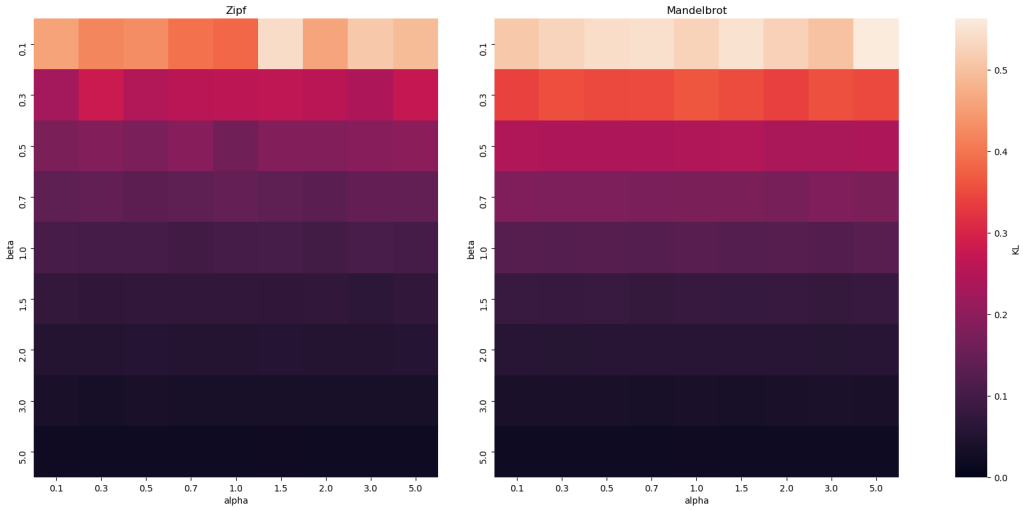


Figure 3.20 Comparaison de la similitude des distributions générées par Dirichlet avec la loi de Zipf (gauche) et celle de Mandelbrot (droite)

Grâce à ces trois figures, il est clair que la similitude entre les données générées et les lois du langage n'est dépendante que de l'hyperparamètre β, car on constate une invariance entre les colonnes des *heatmaps* et c'est ce qui était prévu. De plus, on remarque que l'erreur diminue plus β augmente. On remarque aussi que le niveau de similitude avec Mandelbrot est similaire à celui avec Zipf.

Comme pour l'expérience précédente, on représente les valeurs ainsi que l'emplacement (α, β) des minimums et maximums de divergence KL. De plus, on calcule aussi la moyenne $\overline{KL}$ obtenue pour chaque combinaison d'hyperparamètres :

Tableau 3.2 Synthèse des divergences KL de l'analyse de la similitude de la fréquence de mots générés avec les lois statistiques du langage

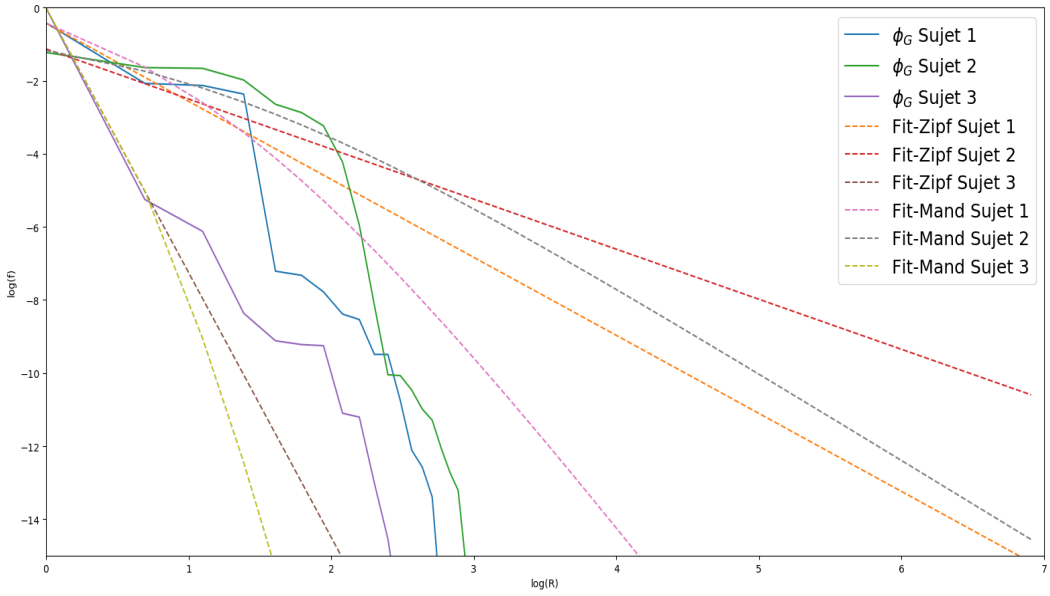
Lois	KL_{max}	KL_{min}	$\overline{KL}$
Zipf	0.54 $(1.5, 0.1)$	0.021 $(5, 5)$	0.15
Mandelbrot	0.56 $(5, 0.1)$	0.021 $(5, 5)$	0.18

Cette synthèse nous démontre que l'hyperparamètre α n'a pas d'impact dans la similitude. En effet, on constate que, pour les deux lois, les maximums et les minimums surviennent à des α similaires. Il est certain que α n'a pas d'impact dans la vraisemblance des données générées, ce qui est intuitif, puisque α régit la répartition des sujets dans les documents. Aussi, on voit que plus β augmente et plus l'erreur diminue. On peut se demander jusqu'à quel point cette erreur diminue en fonction de β. Nous avons fait une analyse de convergence et nous avons trouvé que la divergence KL pour Zipf et Mandelbrot est de l'ordre de 10^{-8} lorsque $\beta = 1000000$. Notons que cette divergence diminue systématiquement lorsque $\beta \to \infty$.

Alors, plus la distribution des mots par sujet est uniforme et plus la similitude avec les meilleures lois de Zipf et Mandelbrot est grande. Or, nous avons vu dans la section précédente que plus la distribution des mots par sujet est uniforme et plus il est difficile pour LDA de retrouver des sujets justes. Ceci pose un dilemme lorsque viendra le temps de décider quels hyperparamètres nous allons utiliser pour la génération des données synthétiques pour la validation du nouvel algorithme. Afin d'analyser plus profondément le comportement de l'erreur relative en fonction de β pour espérer répondre à ce dilemme, on représente les graphiques du logarithme de la fréquence en fonction du logarithme du rang pour les fréquences observées (exprimées par les distributions ϕ) et les fréquences théoriques (obtenues par *fit* sur les lois de Zipf et Mandelbrot), et ce, pour un modèle à 3 sujets. On effectue cette expérience,

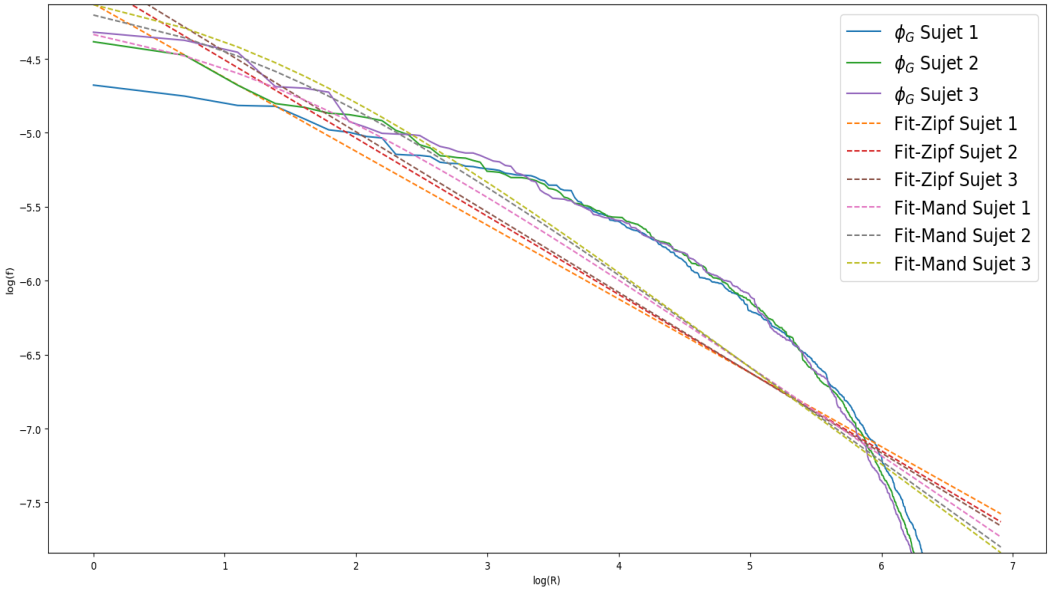
car on veut vérifier si les lois particulières de Zipf et Mandelbrot sont similaires avec celles observées dans la langue naturelle. En d'autres mots, on veut savoir si les paramètres optimaux des lois ressemblent aux paramètres habituellement trouvés dans la littérature. On se penchera surtout sur la loi de Mandelbrot, puisque c'est elle qui donne les meilleurs résultats. Rappelons que, selon la littérature, on devrait trouver un $b \approx 2.7$ et un $c \approx 1$ (voir section 2.1.1). On teste donc trois valeurs de β : 0.01, 0.5, 1000000. On fixe α à 0.7 puisqu'il n'a pas d'impact. Dans chaque sous-figure, on représente une vue d'ensemble des courbes (échelle variable) et on vient ensuite fixer l'échelle (échelle fixe) afin de mieux comparer les figures entre elles. Notons que les axes pour les sous-figures a) sont flexibles selon β alors que les axes pour les sous-figures b) sont fixés à $x = [0, 7]$ et $y = [-15, 0]$ pour tous les β afin de faciliter la comparaison.

Échelle variable

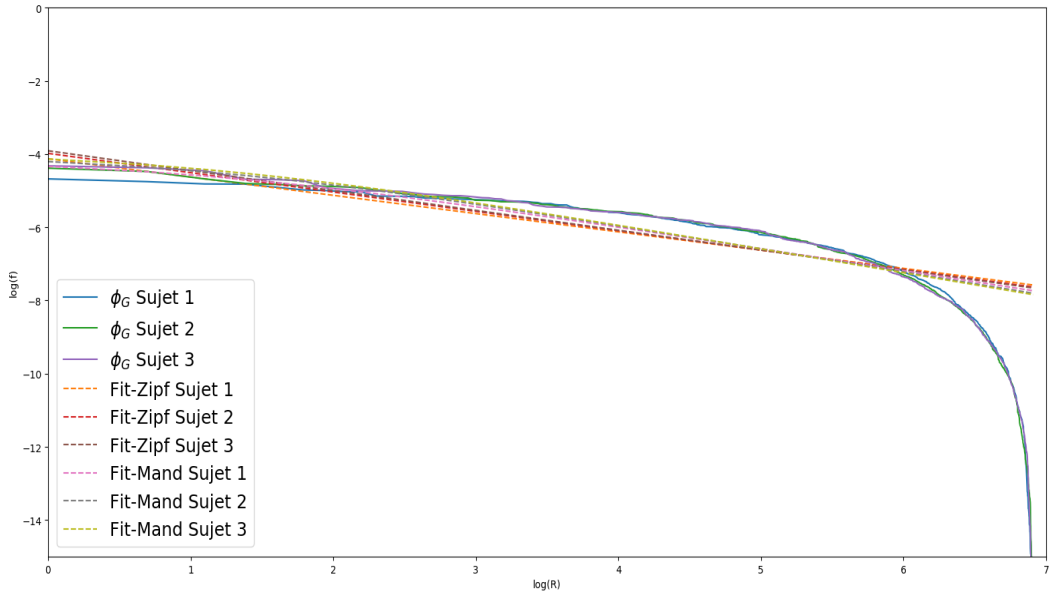


Échelle fixe

Figure 3.22 Fréquences générées et théoriques pour $\beta = 0.01$ avec une échelle variable (haut) et une échelle fixe (bas)

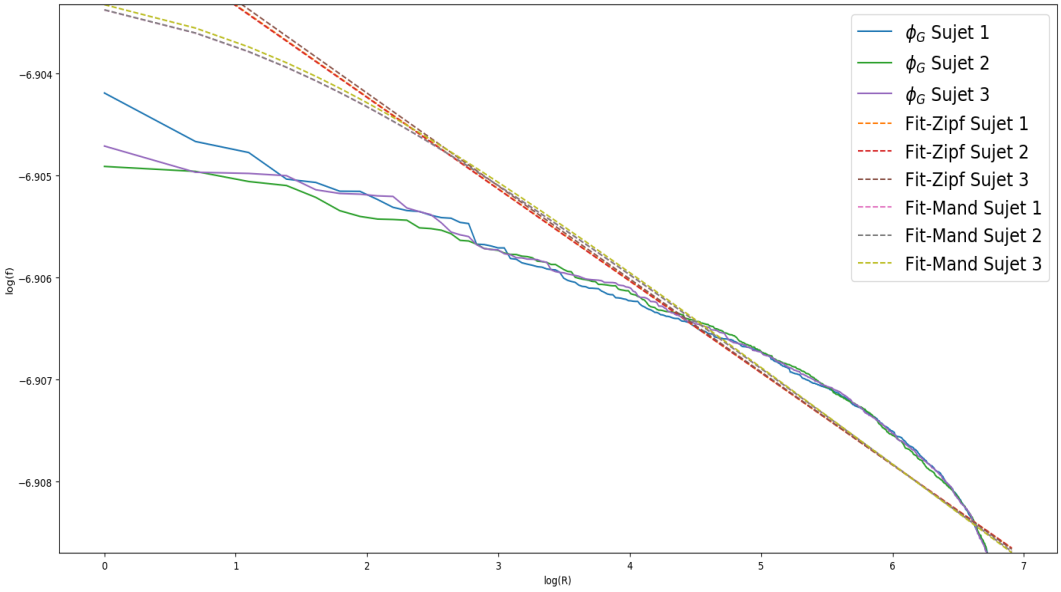


Échelle variable

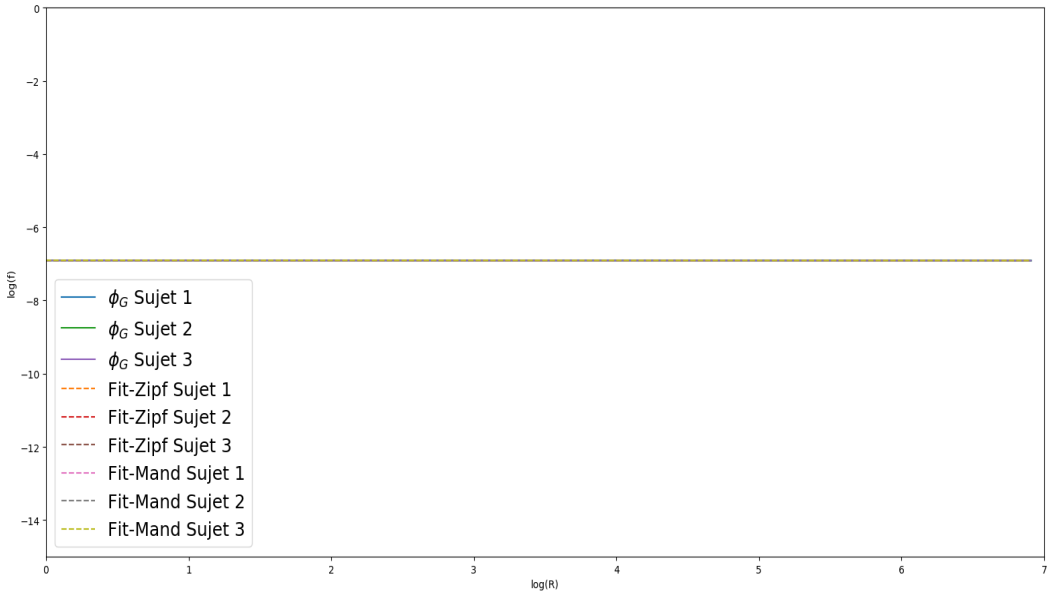


Échelle fixe

Figure 3.24 Fréquences générées et théoriques pour $\beta = 0.5$ avec une échelle variable (haut) et une échelle fixe (bas)



Échelle variable



Échelle fixe

Figure 3.26 Fréquences générées et théoriques pour $\beta = 1000000$ avec une échelle variable (haut) et une échelle fixe (bas)

En analysant la figure 3.22, on comprend immédiatement pourquoi la divergence KL entre les fréquences générées et théoriques est grande. En effet, puisque β est très petit, certains mots sont fortement priorisés et d'autres sont fortement délaissés. De ce fait, la chute de la courbe de données est soudaine et les courbes de *fit* ont de la difficulté à reproduire ce comportement tout en respectant les fréquences élevées des $e^3 \approx 20$ premiers rangs. Par ailleurs, le nombre de mots uniques choisis est petit : seulement $e^{4.7} \approx 110$ mots choisis parmi un vocabulaire de 1000. En d'autres mots, la divergence KL est grande, car les *fit* ne peuvent reproduire le comportement que de $\frac{e^3}{e^{4.7}} \approx 18\%$ des mots de son corpus.

Dans la figure 3.24, on voit que les fréquences générées sont distribuées de manière plus lisse pour $\beta = 0.5$ que pour $\beta = 0.01$. Aussi, davantage de mots uniques sont sélectionnés dans le vocabulaire initial, soit $e^{6.8} \approx 900$ mots. Il semble que les *fit* de Mandelbrot capture bien le comportement des données pour $\frac{e^{6.2}}{e^{6.8}} \approx 55\%$ des mots en plus de bien capturer le comportement des fréquences aux rangs élevés.

Enfin, la figure 3.26 illustre un cas où β tend virtuellement vers l'infini. On constate que presque tous les mots du vocabulaire sont utilisés au moins une fois ($e^{6.9} \approx 1000$). De plus, on remarque que les *fit* de Zipf et Mandelbrot capture bien le comportement de l'ensemble des données. Ceci s'explique par le fait que la distribution des fréquences observées est parfaitement uniforme et cette tendance peut être représentée par les lois de Zipf et Mandelbrot (si on prend une densité de fréquence avec paramètre c très petit). Cependant, l'allure de cette courbe est très différente de ceux qu'on peut retrouver dans la littérature. Alors, même si le *fit* est bon, les données générées ne sont pas nécessairement vraisemblables.

Dans le but de visualiser davantage les *fit* de Zipf et Mandelbrot sur la distribution générée par Dirichlet, on peut représenter les paramètres c optimaux obtenus suite à la validation croisée. La *heatmap* de la figure 3.27 regroupe le c optimal de la meilleure distribution de Zipf pour chaque combinaison d'hyperparamètre utilisée par la génération de Dirichlet :

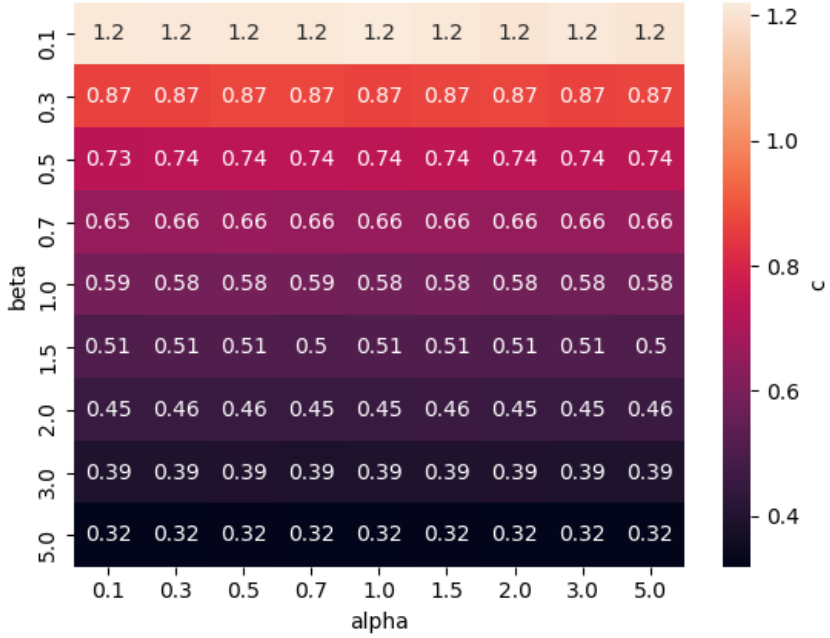


Figure 3.27 Paramètre c optimal de la meilleure distribution de Zipf pour chaque combinaison d'hyperparamètre de Dirichlet

Et pour Mandelbrot :

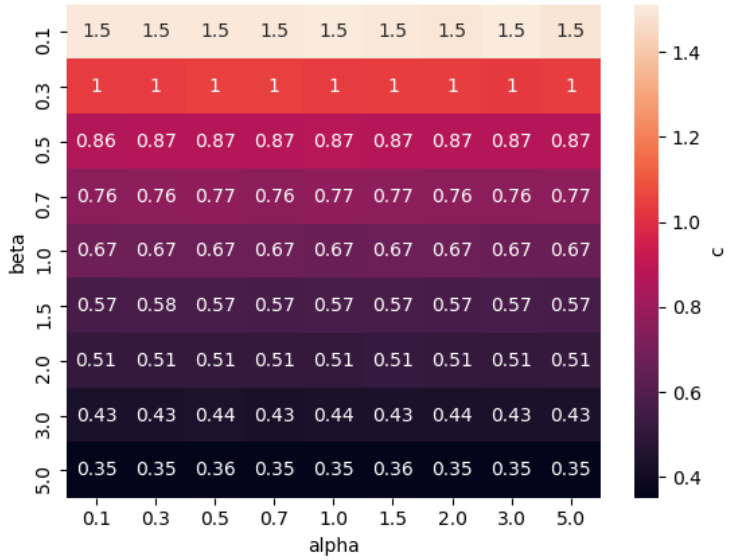


Figure 3.28 Paramètre c optimal de la meilleure distribution de Mandelbrot pour chaque combinaison d'hyperparamètre de Dirichlet

On constate que, malgré que la meilleure distribution de Zipf/Mandelbrot ait un meilleur *fit* avec Dirichlet pour des hauts β, les paramètres de la distribution ne sont pas du tout vraisemblables, le c de la littérature se situant près de 1. Par ailleurs, il s'agit d'un résultat intéressant, puisqu'on peut maintenant présumer les performances du modèle LDA classique inféré par Gensim ou CGS pour un corpus que l'on s'attend à retrouver dans des données réelles. En effet, on remarque que les hyperparamètres de Dirichlet $\beta = 0.3$ et $\beta = 0.5$ correspondent respectivement à $c = 1$ et $c = 0.87$ pour Mandelbrot, ce qui est très proche du paramètre c qui ressort typiquement de la langue naturelle. Alors, si on étudie un document moyen quelconque, il est raisonnable de s'attendre à des performances d'inférences qui correspondent aux divergences KL inscrites aux lignes $\beta = 0.3$ et $\beta = 0.5$ des *heatmaps* de la section 3.3. Nous avons de ce fait une nouvelle mesure témoignant de la justesse de l'algorithme LDA.

On peut même pousser l'analyse plus loin en calculant c pour des valeurs extrêmes de β tout en fixant $\alpha = 0.7$ puisqu'il n'a pas d'importance. On obtient pour Zipf :

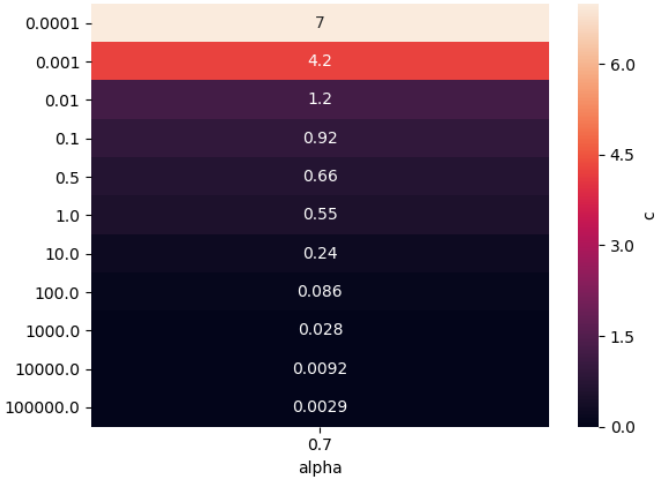


Figure 3.29 Paramètre c optimal de la meilleure distribution de Zipf pour des β extrêmes

Et pour Mandelbrot :

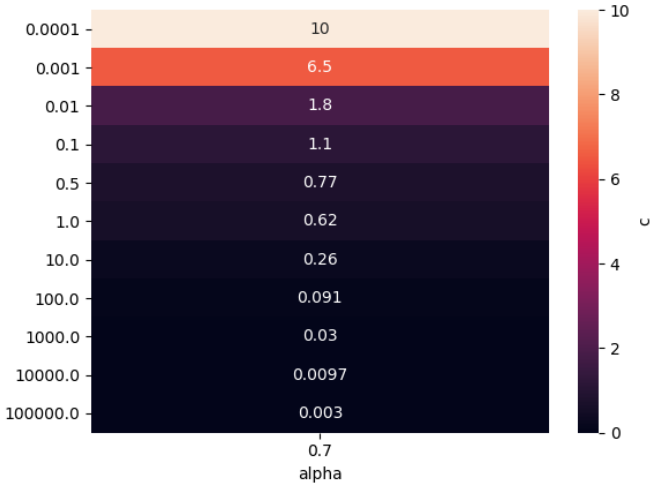


Figure 3.30 Paramètre c optimal de la meilleure distribution de Mandelbrot pour des β extrêmes

On voit donc que plus β augmente et moins le paramètre c optimal du *fit* concorde avec ce qui est observé dans la langue naturelle.

La dernière analyse nous apprend que la « meilleure distribution » de Zipf et Mandelbrot n'est pas suffisante pour témoigner de la vraisemblance des distributions générées, puisque ces lois du langage peuvent prendre des formes dégénérées trop extrêmes telles que la reproduction d'une loi uniforme. De ce fait, au lieu de calculer les meilleures distributions de Zipf/Mandelbrot, nous avons plutôt choisi de déterminer la divergence KL entre une loi de Zipf/Mandelbrot que l'on peut retrouver dans la langue naturelle et la distribution ϕ générée par Dirichlet. Par conséquent, aucune validation croisée n'est nécessaire pour cette analyse puisqu'aucun paramètre c n'est inféré. On fixe le c à une valeur classique de 1.01 pour Zipf et Mandelbrot. Voici ce que l'on obtient pour Zipf :

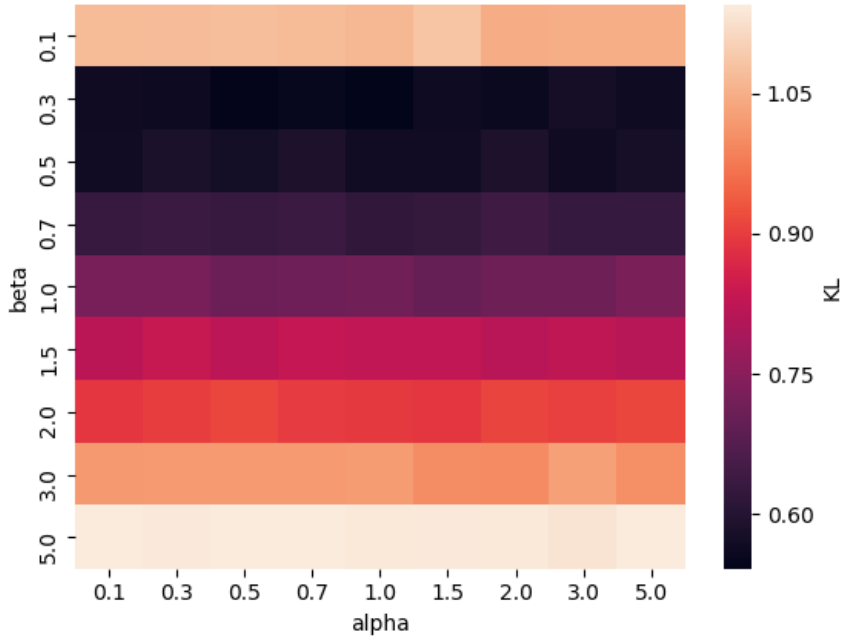


Figure 3.31 Divergences KL entre une distribution de Zipf avec $c = 1.01$ et la distribution générée par Dirichlet

Et pour Mandelbrot :

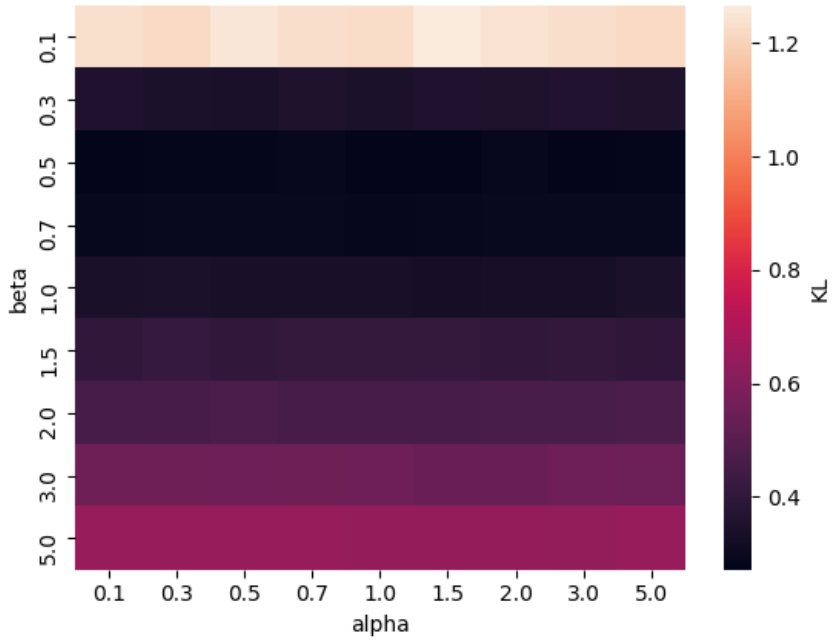


Figure 3.32 Divergences KL entre une distribution de Mandelbrot avec $c = 1.01$ et la distribution générée par Dirichlet

Les résultats obtenus sont intéressants. On constate que le β ne doit pas être trop grand ni trop petit pour générer une distribution vraisemblable. On remarque aussi que la divergence KL augmente systématiquement plus β augmente passé 0.3, preuve que la distribution purement uniforme n'est pas vraisemblable. On conclut qu'une valeur de β se situant entre 0.3 et 1 est acceptable au niveau de la vraisemblance des données générées.

À la lumière de ces résultats, on peut se demander quelle combinaison d'hyperparamètres serait préférable pour la portion génération du cadre de validation. La première expérience nous a montré que des α et β inférieurs à 1 étaient à priorisés afin d'optimiser la performance de l'inférence. Or, la seconde expérience nous a appris que plus β est grand et plus les données générées peuvent être lissées par les lois du langage. Cependant, nous avons vu que le fait que ces données peuvent être lissées par les lois du langage ne signifie pas directement qu'elles sont plus vraisemblables en raison du comportement dégénératif vers la loi uniforme de Zipf et Mandelbrot. De ce fait, l'analyse des figures 3.31 et 3.32 nous a indiqué qu'une

valeur de β se situant entre 0.3 et 1 contribue à une génération de données vraisemblables. On conclut donc que $\beta = 0.5$ est un bon compromis entre vraisemblance des données générées et performance de l'algorithme d'inférence. Aussi, puisque l'on veut un α inférieur à 1, mais que l'on veut permettre le plus de sujets possible par documents par souci de variété, on conclut qu'un $\alpha = 0.7$ est raisonnable.

En résumé, en ce qui concerne le cadre de validation, on recommande une génération où les paramètres θ et ϕ sont issus de distributions de Dirichlet symétriques et où les hyperparamètres utilisés sont $\alpha_i = 0.7$ et $\beta_i = 0.5$.

3.5 Statistiques complètes de l'inférence

Alors que les $\alpha_i = 0.7$ et les $\beta_i = 0.5$ ont été choisis, on peut présenter les statistiques complètes de l'inférence (voir section 3.2.3) pour ces paramètres. Ceci constitue la troisième et dernière expérience du chapitre. Ces informations constitueront la référence lorsque viendra le temps de valider le nouvel algorithme d'expertise.

D'abord, on prend encore un modèle à 3 sujets avec inférence par CGS à titre d'exemple afin de montrer un histogramme des distributions ϕ générées et inférées pour chaque mot du vocabulaire. On montre d'abord les distributions générées et inférées sur des graphiques différents afin de bien distinguer les sujets (chaque colonne de graphique représente un sujet) :

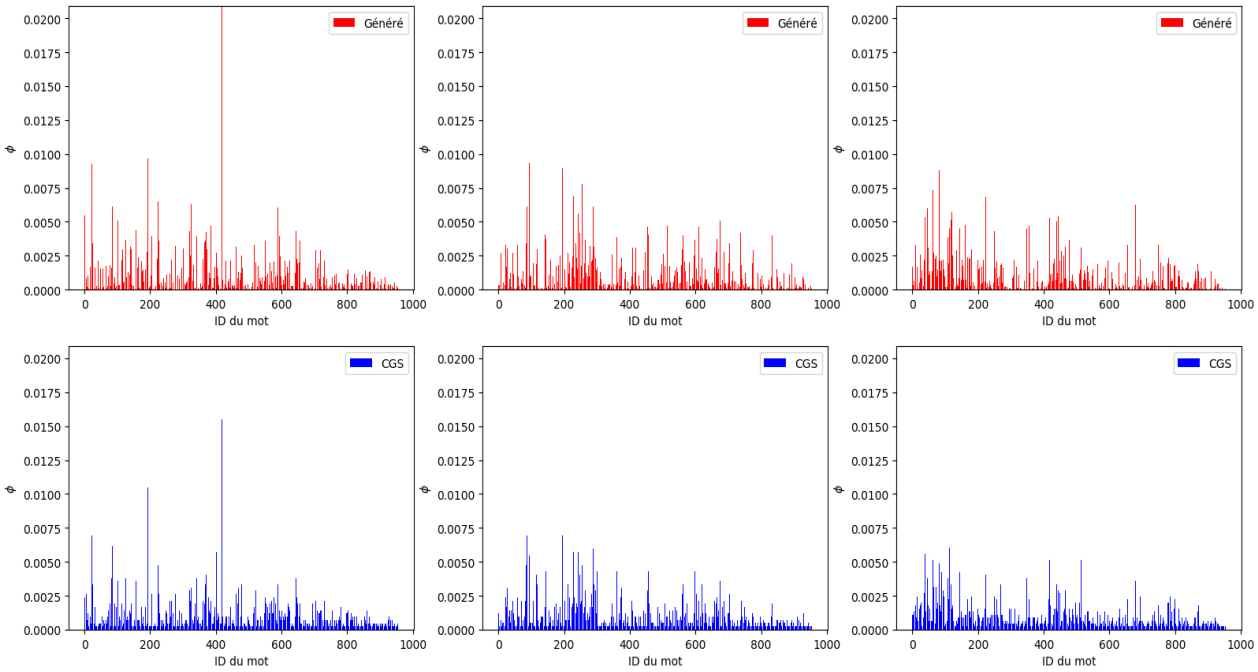


Figure 3.33 Les distributions ϕ générées et inférées pour les sujets 1 (gauche), 2 (milieu) et 3 (droite)

On remarque que les sujets sont suffisamment différents les uns des autres, les pics associés à certains mots étant caractéristiques.

Ensuite, on se sert de la figure 3.34 pour analyser la qualité de l'inférence. Dans un premier temps, on ordonne les identifiants des mots générés en fonction de leur rang (ordre décroissant de ϕ) et on trace la première courbe illustrant la distribution ϕ générée en fonction du rang des mots. Dans un deuxième temps, on représente sur la figure 3.34 l'histogramme de la distribution ϕ associée à ces mêmes identifiants de mots. Cette distribution ϕ est inférée par CGS. La figure comporte 3 graphiques pour chacun des sujets :

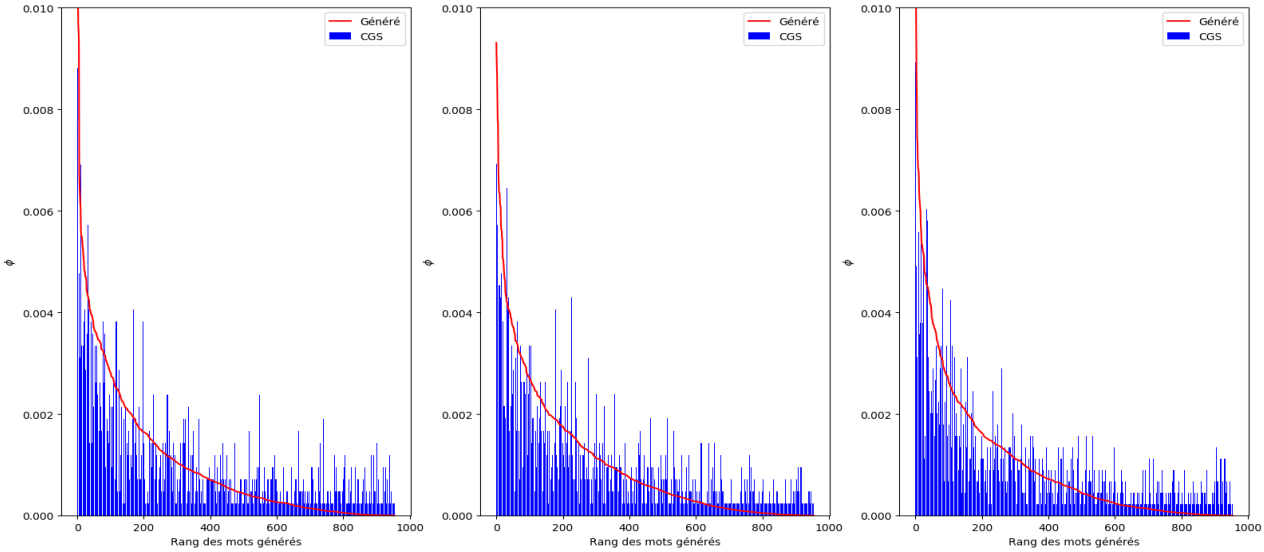


Figure 3.34 La distribution ϕ générée comparée à celle inférée ordonnées selon le rang des fréquences pour le sujet 1 (gauche), 2 (milieu) et 3 (droite)

On voit ici que les inférences sont adéquates, car la similitude entre la courbe et l'histogramme est marquée. On constate aussi que les inférences sont plus précises pour les fréquences élevées que pour les fréquences faibles, phénomène qui était attendu pour ces hyperparamètres.

Finalement, on présente un tableau synthétisant les performances de l'inférence en lien avec les trois métriques de comparaison discutées à la section 3.2.3. Pour chacune de ces métriques, on montre les moyennes μ ainsi que les écarts-types σ des valeurs de θ et ϕ de 100 corpus distincts :

Tableau 3.3 Synthèse des performances de l'inférence d'un modèle LDA classique avec inférence par CGS où $\alpha = 0.7$ et $\beta = 0.5$ sont utilisés pour la génération

Métriques	μ_θ	σ_θ	μ_ϕ	σ_ϕ
KL	0.10	0.03	0.31	0.07
Dr	0.14	0.05	0.21	0.07
$D\cos$	0.04	0.01	0.13	0.04

Ce tableau nous démontre une meilleure précision dans l'inférence de θ par rapport à ϕ, et ce, avec un écart-type plus faible. En somme, on juge bonne la qualité des inférences de θ et ϕ. On énumère aussi les paramètres des *fit* de Zipf et Mandelbrot sur les données générées :

— Le paramètre optimal du *fit* de Zipf sur les données générées est $c = 0.74$.

— Le paramètre optimal du *fit* de Mandelbrot sur les données générées est $c = 0.87$.

Enfin, on analyse l'efficacité de la méthode d'alignement présentée à la section 3.2.2. Le tableau suivant montre les 3 métriques de performance calculées pour les paramètres θ et ϕ, et ce, pour les 6 alignements possibles A_i d'un modèle à 3 sujets ($3! = 6$) :

Tableau 3.4 Métriques de performance pour les 6 alignements possibles d'un modèle à 3 sujets

Métriques	A_1	A_2	A_3	$\mathbf{A_4}$	A_5	A_6
KL_θ	1.49	0.95	1.01	**0.07**	1.42	1.03
KL_ϕ	1.04	0.77	0.78	**0.27**	1.04	0.81
Dr_θ	1.47	0.97	1.02	**0.07**	1.46	1.01
Dr_ϕ	0.97	0.70	0.70	**0.18**	0.96	0.72
$D\cos_\theta$	0.58	0.37	0.40	**0.02**	0.57	0.41
$D\cos_\phi$	0.57	0.41	0.40	**0.11**	0.56	0.43

On voit que le meilleur alignement ressort clairement : il s'agit de l'alignement 4 dont les résultats sont écrits en gras. Pour les 6 métriques, on remarque que c'est pour l'alignement 4 qu'on observe la meilleure performance. Par ailleurs, on constate qu'il existe 3 ordres de grandeur différents dans les résultats de chaque métrique. On remarque par exemple que, pour KL_θ, les valeurs 1.49 et 1.42 constituent le premier ordre de grandeur, les valeurs 0.95, 1.01 et 1.03 constituent le deuxième ordre de grandeur alors que la valeur 0.07 constitue le troisième ordre de grandeur. Or, cette segmentation d'ordre de grandeur est la même pour les 6 métriques : les alignements A_1 et A_5 sont les moins bons alignements, les alignements A_2, A_3 et A_6 sont les alignements centraux tandis que l'alignement A_4 est le meilleur. Or, sur les 6 combinaisons possibles qui caractérisent un alignement à 3 sujets de type $\{A, B, C\}$, il existe 2 combinaisons qui n'alignement correctement aucun sujet (ex :$\{B, C, A\}$) soit A_1 et A_5 dans ce cas, 3 combinaisons qui alignement correctement 1 sujet (ex :$\{A, C, B\}$) soit A_2,

A_3 et A_6 dans ce cas et 1 combinaison qui aligne correctement les 3 sujets ($\{A, B, C\}$) soit A_4 dans ce cas. Il est donc normal de constater cette hiérarchie dans les performances associées aux différents alignements. Par conséquent, on conclut que les résultats sont cohérents et que le meilleur alignement est facilement reconnaissable.

Nous avons procédé à la même expérience pour un modèle à 6 sujets et conclu que l'alignement optimal était aussi cohérent pour toutes les métriques. Cependant, étant donné que les combinaisons possibles pour un alignement à 6 sujets est au nombre de $6! = 720$, les résultats associés à cette expérience sont trop lourds pour être présentables dans ce mémoire. Attardons-nous tout de même au temps d'exécution des alignements. Pour un alignement à 3 sujets, le temps d'exécution est de quelques millisecondes. Pour un alignement à 6 sujets, le temps d'exécution est de 20 secondes. À titre indicatif, le temps d'exécution pour les 3628800 combinaisons d'un alignement à 10 sujets serait d'environ 14 heures. On comprend donc qu'on utilise cette méthode relevant de la force brute due au contexte académique du mémoire et au fait qu'il n'est pas nécessaire de spécifier un nombre élevé de sujets pour valider le fonctionnement de l'algorithme. Toutefois, si on voulait appliquer les travaux de ce mémoire dans un contexte réel, on devrait avoir recours à une méthode plus sophistiquée pour l'alignement afin d'en améliorer l'efficacité.

3.6 Résumé des résultats

Ceci conclut ce chapitre portant sur la théorie de LDA et le cadre de validation. Nous avons d'abord présenté la théorie du modèle LDA classique où l'objectif du modèle, la nomenclature utilisée, une description de la distribution de Dirichlet ainsi qu'une présentation du CGS ont été abordés. Ensuite, nous avons expliqué pourquoi il était nécessaire de valider le modèle LDA classique avec des données synthétiques générées et les paramètres des modèles employés pour tester ce cadre ont été énumérés. De plus, nous avons présenté les métriques de comparaison entre les distributions générées et inférées. Finalement, deux analyses importantes ont été menées afin de répondre à la première question de recherche soit : quelles sont les conditions opérationnelles du modèle LDA classique et dans quelle mesure l'hypothèse de génération de ce modèle est-elle conforme aux lois statistiques du langage ? D'une part, nous avons établi l'impact de la performance du CGS et de l'inférence variationnelle en fonction des hyperparamètres utilisés pour la génération et en fonction du nombre de sujets latents spécifiés. Nous avons déterminé qu'en général, les α et β doivent être inférieurs à 1 pour assurer les bonnes performances de l'inférence. D'autre part, nous avons étudié la similitude des données générées avec les lois statistiques du langage. Nous avons trouvé que plus β était

grand et plus les lois de Zipf et Mandelbrot étaient en mesure de capturer les données. Or, nous avons aussi déterminé que, lorsque β devient trop grand, les paramètres optimaux de Mandelbrot ne sont pas réalistes. En bref, il a été conclu que la combinaison $\alpha = 0.7$ et $\beta = 0.5$ constituait un bon compromis entre la performance de l'inférence et la similitude avec les lois du langage. Pour clore ce chapitre, en plus d'avoir validé la méthode d'alignement du cadre de validation, nous avons calculé les statistiques complètes d'un modèle LDA classique avec ces hyperparamètres, ces dernières qui feront office de référence lorsque viendra le temps de valider de façon analogue le nouveau modèle d'expertise qui sera élaboré au chapitre suivant.

CHAPITRE 4 MODÈLE D'EXPERTISE DES AUTEURS ET SON INFÉRENCE

Après avoir abordé les balises théoriques du modèle LDA classique, on présente notre extension au modèle pour inférer le niveau d'expertise des auteurs. On répondra donc à la seconde question de recherche, soit : comment peut-on faire interagir les lois statistiques du langage dans l'infrastructure LDA afin de déterminer l'expertise des auteurs propre à un sujet donné ? D'abord, on présentera de façon générale le modèle d'expertise et on étayera notre hypothèse sur la mesure de l'expertise en étudiant une analogie avec le paramètre c de Mandelbrot. Ensuite, on utilisera le précédent cadre de validation pour démontrer qu'il est possible d'obtenir de bonnes performances d'inférence pour θ et ϕ en utilisant une distribution de Mandelbrot pour la génération de ϕ plutôt qu'une distribution de Dirichlet. Ensuite, on pourra présenter la nomenclature en lien avec le nouveau modèle ainsi que les manipulations mathématiques nécessaires pour inférer l'expertise propre à un sujet et à un auteur. On présentera par la suite une série d'expériences de comparaison afin de démontrer la validité des méthodes et de comparer les résultats obtenus avec ce qu'on obtiendrait sans l'inclusion de l'inférence d'expertise. Finalement, on évaluera une limite de la méthode ainsi que la sensibilité à ses hyperparamètres.

4.1 Description générale du modèle d'expertise

Avant de décortiquer chaque élément du modèle, il est nécessaire de donner une vision générale de celui-ci. Tandis que l'objectif du modèle LDA classique était d'inférer les distributions θ et ϕ, on introduira un nouveau paramètre afin de quantifier le niveau d'expertise de chaque auteur pour chaque sujet. L'objectif du modèle d'expertise sera donc d'inférer, à partir d'un corpus de textes associés à certains auteurs, les mêmes paramètres θ, ϕ en plus du paramètre γ, soit le paramètre d'expertise. Chaque auteur possédera une valeur de gamma associée à chaque sujet. Par exemple, comme nous utiliserons 3 sujets dans notre modèle, chaque auteur sera caractérisé par 3 paramètres γ distincts, ces derniers témoignant de leur niveau d'expertise dans un sujet donné. Notons que plus le paramètre gamma est petit et plus le niveau d'expertise de l'auteur est élevé.

On peut visualiser cette expertise par sujet avec la figure 4.1 :

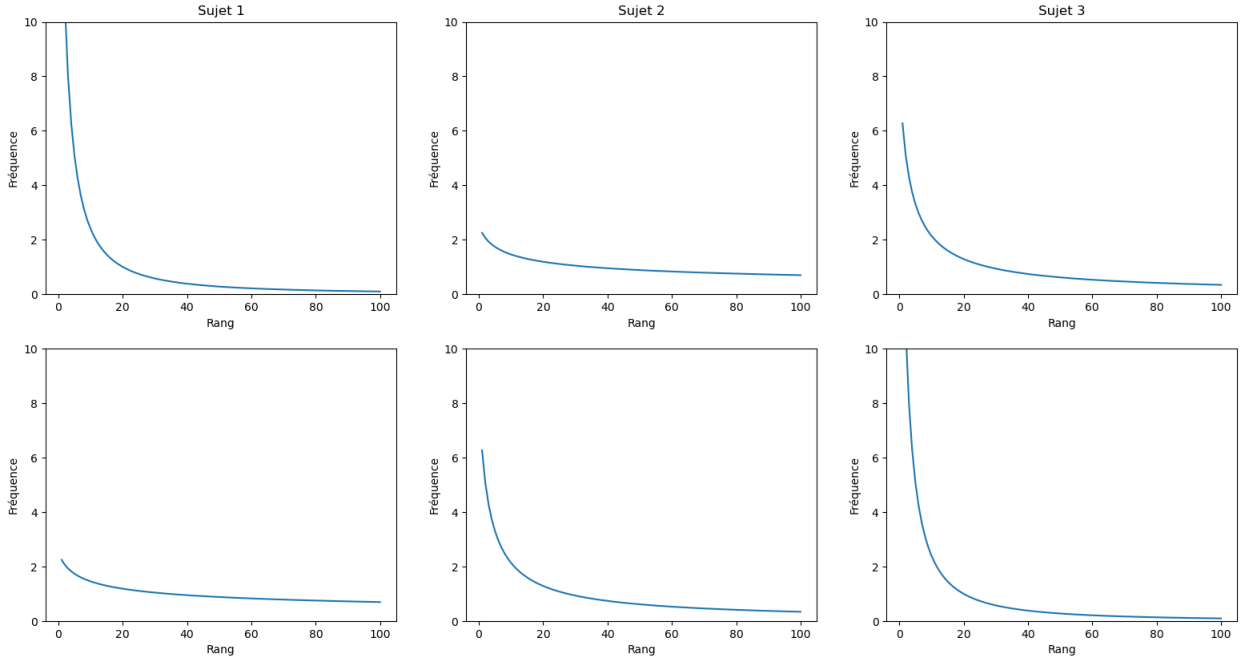


Figure 4.1 Représentation de l'expertise pour 2 auteurs et 3 sujets

Ici, chaque auteur est représenté par une ligne et chaque sujet est représenté par une colonne. Les graphiques illustrent la fréquence selon laquelle un vocabulaire de 100 mots est réparti en fonction du rang des mots (les rangs les plus faibles correspondent aux mots les plus communs) et en fonction des sujets. De plus 3 niveaux d'expertise sont représentés : le novice ($\gamma = 1.5$), le moyen ($\gamma = 0.87$) et l'expert ($\gamma = 0.35$). En ce qui concerne l'auteur 1, on remarque qu'il est novice dans le sujet 1, expert dans le sujet 2 et moyen dans le sujet 3. En effet, on constate qu'il n'utilisera que les mots les plus communs dans le sujet 1 alors qu'il peut sans problème employer les mots plus complexes dans le sujet 2. Pour ce qui est de l'auteur 2, il est expert dans le sujet 1, moyen dans le sujet 2 et novice dans le sujet 3. On note alors que les expertises sont différentes pour chaque sujet et c'est selon cette hypothèse que nous allons baser notre modèle.

Dans le but d'inférer ce paramètre γ, il sera nécessaire de faire un post-traitement des distributions inférées par LDA pour déterminer les fréquences de mots par sujet des auteurs. Cette identification des fréquences de mots par sujet constitue l'extension au modèle LDA classique. Une fois qu'on a ces fréquences de mots par sujet, il sera possible de calculer le paramètre γ avec un *fit* de Mandelbrot.

Afin de valider les bonnes performances d'inférence de notre modèle, on aura recours à un cadre de validation similaire à celui qui a été détaillé au chapitre 3. La génération des distributions θ et ϕ se fera comme suit :

$$\theta \sim \text{Dir}(\alpha)$$
$$\phi \sim \text{Mand}(\gamma)$$

On se base ainsi sur une hypothèse que la distribution de mots par sujet est dépendante d'une loi de Mandelbrot paramétrée par γ soit le paramètre d'expertise. De plus, alors qu'on avait autant de distributions ϕ qu'on avait de sujets dans LDA classique, on a désormais autant de distributions ϕ que le produit entre le nombre d'auteurs et le nombre de sujets.

4.2 Lien entre l'expertise, le paramètre de Mandelbrot et le paramètre γ

Dans cette sous-section, on expliquera pourquoi on pose que le paramètre d'expertise γ de notre modèle équivaut au paramètre c d'une loi de Mandelbrot, sachant que l'ordre de technicalité des mots est connu.

Nous avons vu à la section 3.4 qu'il était possible d'obtenir les meilleurs *fit* de Mandelbrot sur les données générées par Dirichlet. Quand on parle de « meilleur *fit* », on fait référence à la distribution de Mandelbrot obtenue avec le paramètre c qui minimise la divergence KL avec une distribution de Dirichlet donnée. En effet, on a conclu que, sur l'étendue extrême des β [0.0001, 100 000], l'étendue des paramètres de Mandelbrot était sur l'intervalle [0.003, 10]. On peut représenter l'allure des courbes de densité de fréquences obtenues avec ces paramètres. Notons que les paramètres considérés sont ceux obtenus pour les puissances de 10 de β utilisées à la figure 3.29. On montre les courbes sur une échelle log-log afin de faciliter la compréhension.

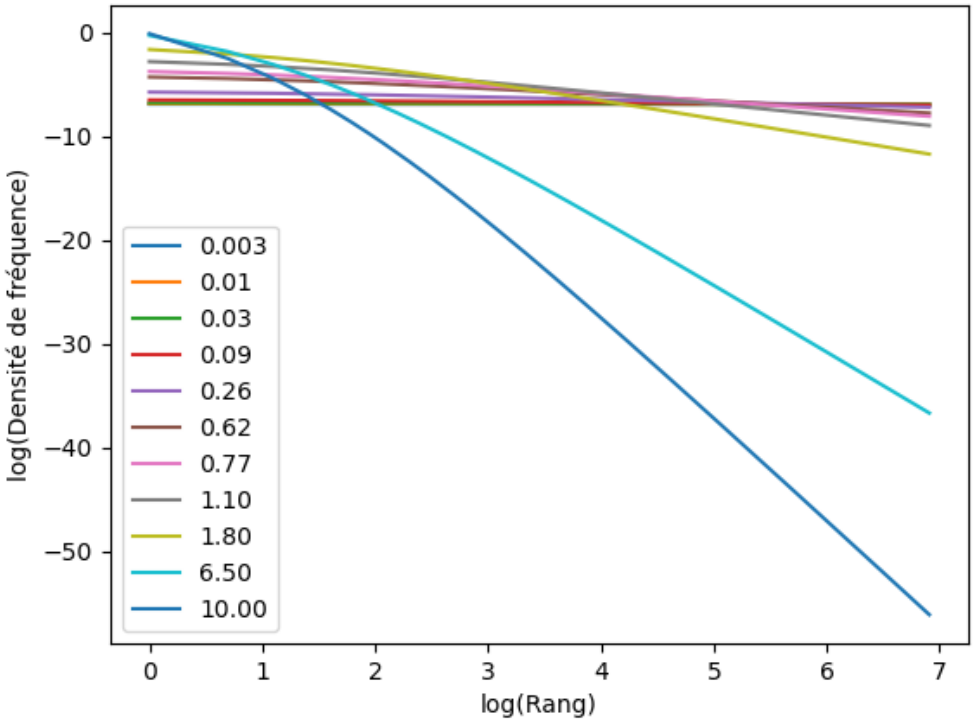


Figure 4.2 Densité de fréquences pour différents paramètres de Mandelbrot

On remarque sur cette figure que plus le paramètre de Mandelbrot diminue et plus la courbe s'aplatit. Pour un $c = 0.003$, on obtient une distribution quasi uniforme. Notons que le rang ici est synonyme de rareté du mot. Par exemple, les mots à rang faible sont les mots communs que les novices et les experts utilisent. Or, plus on augmente le rang et plus on rencontre des mots rares et techniques. Alors, on émet l'hypothèse que le vocabulaire d'un novice est restreint aux mots les plus communs alors que celui d'un expert est davantage flexible. L'expert *parfait* peut, en théorie, utiliser un mot très simple ou très complexe, et ce, de manière équiprobable. De ce fait, si on considère un vocabulaire fixe dans le corpus, il est raisonnable d'émettre l'hypothèse que plus le paramètre de Mandelbrot d'une distribution de mots est faible et plus l'expertise de l'auteur lié à cette distribution est élevée. En effet, si on prend en exemple la courbe associée à un paramètre de Mandelbrot faible, soit $c = 0.0003$, on voit bien qu'un expert *parfait* peut utiliser tous les mots de manière équiprobable. À l'inverse, la courbe associée à un paramètre de Mandelbrot élevé, soit $c = 10$, nous montre que le vocabulaire du *parfait* novice ne sera restreint qu'à quelques mots seulement. Évidemment, le concept de novice et d'expert *parfaits* n'est que théorique : dans les expériences, on observera plutôt des expertises se situant entre ces extrêmes. Par ailleurs, cette hypothèse de

l'expertise est vraie dans la mesure où l'on ordonne les mots selon leur niveau de technicalité pour chaque sujet, car sinon, on pourrait confondre l'expertise avec le concept de richesse du vocabulaire qui est distinct même si lié à l'expertise.

En somme, l'objectif du nouveau modèle d'expertise sera de déterminer le c inféré des distributions de mots par sujet et par auteur. Ce c inféré sera le paramètre γ d'expertise pour un auteur et un sujet donné. Afin d'évaluer la justesse de cette inférence, on aura recours au même cadre de validation que celui présenté au chapitre 3. Ce cadre de validation nous servira à déterminer la corrélation entre les $c = \gamma$ caractérisant les distributions générées et les $c = \gamma$ des distributions inférées. Or, on doit d'abord vérifier que ce cadre de validation est toujours fonctionnel si on remplace la distribution de Dirichlet pour la génération par une distribution de Mandelbrot.

4.3 Performances de l'inférence avec une génération par loi de Mandelbrot

Lors de l'élaboration du cadre de validation, il était logique de sélectionner une distribution de Dirichlet pour la génération, puisque l'objectif du cadre était de retrouver les paramètres α et β inférés. Or, en pratique, on pourrait retrouver n'importe quelle distribution de mots dans les textes qu'on étudie et, en principe, l'algorithme du CGS peut modéliser les distributions génératives sous la forme d'une distribution de Dirichlet. En d'autres mots, si on suppose que $\phi \sim \text{Mand}(c)$, on devrait obtenir de bonnes corrélations entre les distributions générées et inférées tout en gardant l'infrastructure d'inférence qui présuppose une estimation d'une postérieure de Dirichlet. Nous allons tester cette hypothèse avec l'expérience suivante.

Cette expérience est analogue à celle présentée à la section 3.5, mais nous utiliserons des distributions ϕ générées avec Mandelbrot plutôt qu'avec Dirichlet. Ainsi, on aura $\phi \sim \text{Mand}(c)$ au lieu de $\phi \sim \text{Dir}(\beta)$. Nous garderons la distribution générée des sujets par Dirichlet, soit $\theta \sim \text{Dir}(\alpha)$. Nous utiliserons un α de 0.7 et un c de 0.87. On prend $c = 0.87$, puisque c'est le c du meilleur *fit* de Mandelbrot pour une génération à $\alpha = 0.7$ et $\beta = 0.5$, soit les hyperparamètres choisis pour la présentation des statistiques complètes retenues à des fins de comparaison à la section 3.5. Si notre hypothèse est juste, on devrait s'attendre à obtenir des performances d'inférence similaire à celles synthétisées dans le tableau 3.3. Par ailleurs, on doit supposer un ordre dans le vocabulaire pour générer les données avec Mandelbrot. En effet, pour chaque sujet, on doit disposer d'un rang *a priori* des mots du vocabulaire pour évaluer leur distribution de fréquence selon Mandelbrot. De plus, cet ordre doit être différent pour chaque sujet, car ce n'est pas parce qu'un mot est rare ou technique dans un sujet qu'il

l'est aussi dans un autre. Malgré le fait qu'il existe une corrélation dans le rang des mots inter sujets, nous avons opté pour l'hypothèse simplifiée d'un ordre aléatoire qui change de sujet en sujet. Les résultats des moyennes et écarts-types des performances de l'inférence pour 100 corpus sont synthétisés dans le tableau suivant :

Tableau 4.1 Synthèse des performances de l'inférence d'un modèle LDA avec génération par Mandelbrot où $\alpha = 0.7$ et $c = 0.87$

Métriques	μ_θ	σ_θ	μ_ϕ	σ_ϕ
KL	0.09	0.05	0.22	0.06
Dr	0.13	0.08	0.09	0.06
$D\cos$	0.03	0.02	0.07	0.04

Considérons maintenant seulement la divergence KL et comparons avec les résultats obtenus à l'expérience de la section 3.3 (génération avec Dirichlet $\alpha = 0.7$, $\beta = 0.5$).

Tableau 4.2 Comparaison des divergences KL entre une génération avec Dirichlet et une génération avec Mandelbrot pour ϕ

Générations ϕ	μ_θ	σ_θ	μ_ϕ	σ_ϕ
Dirichlet	0.10	0.03	0.31	0.07
Mandelbrot	0.09	0.05	0.22	0.06

La qualité de l'inférence est inchangée pour θ, ce qui est normal puisque la distribution utilisée pour la génération est la même. On observe aussi que la qualité de l'inférence pour ϕ est bonne lorsqu'on utilise une distribution de Mandelbrot pour la génération. Par ailleurs, les performances semblent meilleures pour la génération avec Mandelbrot. Toutefois, il est important de noter que cette amélioration n'est probablement pas due à la nature de la distribution générative : comme on compare une distribution de Dirichlet de $\beta = 0.5$ avec la meilleure distribution de Mandelbrot en termes de *fit*, il ne s'agit pas d'une analogie parfaite. Il est donc probable que la distribution issue d'une Mandelbrot $c = 0.87$ est mieux adaptée à LDA, comme l'étaient les distributions de Dirichlet en fonction de leur β. Cependant, le fait important à retenir est qu'il est possible d'utiliser une distribution de Mandelbrot dans la génération et, par le fait même, il sera possible de se servir du cadre de validation pour démontrer l'inférence du paramètre c.

4.4 Méthode d'inférence de l'expertise des auteurs par sujet

Dans cette section, on détaillera le processus afin d'inférer l'expertise des auteurs propre à un sujet donné. Dans un premier temps, on expliquera quelles seront les méthodes d'inférence considérées pour le paramètre de Mandelbrot. Ensuite, après avoir abordé la nomenclature caractérisant le nouveau modèle d'expertise, on verra comment l'inférence de ce paramètre interagit dans le modèle LDA. Cette extension à LDA, permettant l'inférence d'un paramètre d'expertise par sujet par auteur, constitue l'une des contributions du mémoire.

4.4.1 Estimation du paramètre de Mandelbrot

Étant donné une distribution de densités de fréquence, on veut trouver le paramètre c de Mandelbrot qui donne le meilleur *fit* avec les données. Pour ce faire, on a recours à deux méthodes d'estimation : les moindres carrés non linéaires ainsi que l'estimation par maximum de vraisemblance. Détaillons les deux approches.

Moindres carrés non linéaires

La méthode des moindres carrés non linéaires suppose qu'une gamme de fonctions non linéaires $\mathbf{f}(\mathbf{x})$ ont généré une série d'observations $\mathbf{b}$. De façon générale, pour m observations d'une fonction recherchée de n paramètres, elle suppose que :

$$\begin{aligned} f_1(x_1, x_2, ...x_n) &= b_1 \\ f_2(x_1, x_2, ...x_n) &= b_2 \\ &\vdots \\ f_m(x_1, x_2, ...x_n) &= b_m \end{aligned}$$

De façon générale, on a que $f_1 = f_2 = f_m$. Une fois le problème posé, le système à résoudre est :

$$\min_{\mathbf{x}} \frac{1}{2} ||\mathbf{f}(\mathbf{x}) - \mathbf{b}||_2^2$$

La résolution numérique peut se faire de plusieurs manières. Dans le cadre du présent projet, on utilise la librairie *curve fit* de *Scipy* qui se sert de l'algorithme de Levenberg-Marquardt pour résoudre le système non linéaire (Moré (1978)). Dans le contexte d'une distribution de Mandelbrot, le vecteur de paramètres $\mathbf{x}$ n'est seulement que le paramètre c, la distribution de Mandelbrot générée constitue les observations du vecteur $\mathbf{b}$, et la fonction non linéaire recherchée est l'équation de la densité de fréquence de Mandelbrot.

Estimation par maximum de vraisemblance

L'estimation par maximum de vraisemblance est une méthode d'estimation adaptée pour les distributions de probabilités. Cette dernière consiste à minimiser la log-vraisemblance des données. Plus spécifiquement, on doit minimiser :

$$\min_{c} - \sum_{R=1}^{N_W} \log f_{\text{Mand}(R;c)}$$

Afin de procéder à cette minimisation, on a recours à la librairie *minimize* de *Scipy* qui utilise la méthode BFGS (Fletcher (2013)) pour la résolution numérique.

Comparaison des deux méthodes d'inférence du paramètre de Mandelbrot

Pour comparer la méthode des moindres carrés (MCNL) à l'estimation par maximum de vraisemblance (EMV), on génère une distribution de Mandelbrot de 1000 mots avec un c connu. Une fois cette distribution générée, on essaie d'estimer le paramètre c qui donne le meilleur *fit* avec les données et on vient comparer les estimations obtenues. Nous avons généré 100 distributions de Mandelbrot avec des c choisis sur l'intervalle linéaire [0.01, 7]. Une composante de bruit $\epsilon \sim \mathcal{N}(\mu = 0, \sigma^2 = 10^{-4})$ a aussi été ajoutée aux distributions de Mandelbrot. On regroupe les estimations par les deux méthodes dans le graphique suivant :

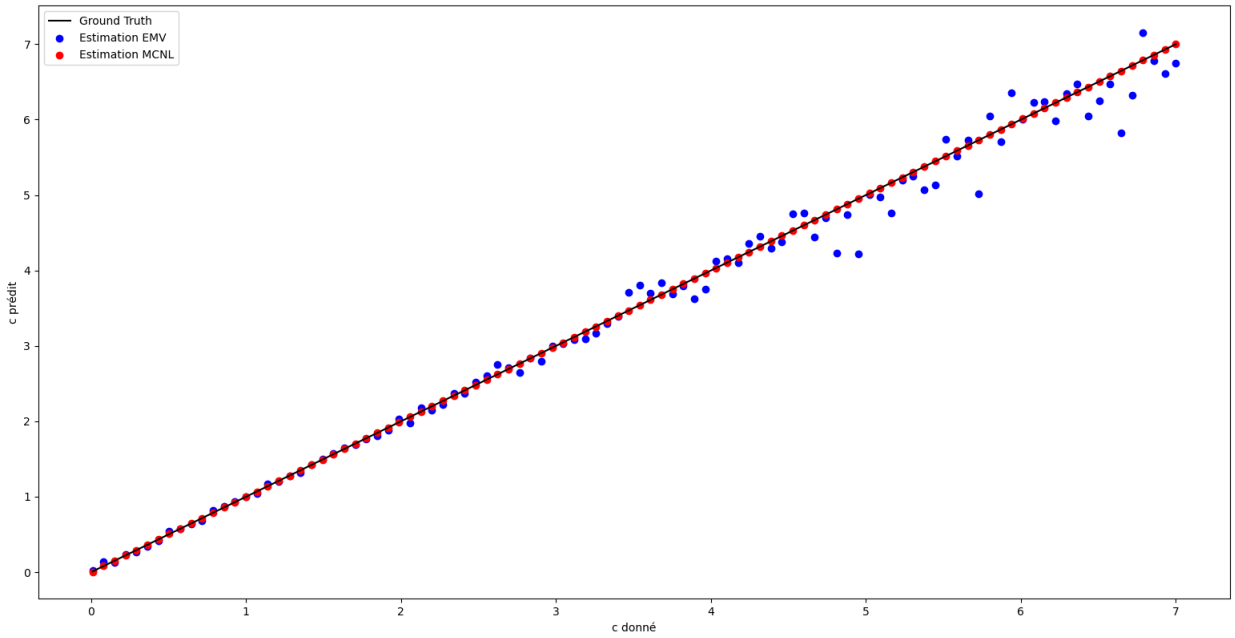


Figure 4.3 Comparaison des méthodes d'inférence du paramètre de Mandelbrot

Les moindres carrés donnent de meilleurs résultats que l'estimation par maximum de vraisemblance. En effet, les prédictions données par les MCNL sont quasi parfaites, soit $R^2 = 0.9999998$ et RMSE $= 0.0009$, alors que celles données par l'EMV sont en deçà, mais tout de même bonnes, soit $R^2 = 0.993$ et RMSE $= 0.17$. Les bonnes performances des MCNL sont expliquées par le fait que la présente optimisation est convexe et que l'espace des paramètres recherchés est de 1 dimension. Concernant l'EMV, même si les performances sont moins bonnes (quoiqu'acceptables), nous allons la conserver pour les expériences subséquentes, car il est possible que cette méthode devienne plus performante si on l'applique sur des distributions inférées par LDA. En effet, ces méthodes d'estimations ne seront pas utilisées sur des lois de Mandelbrot directement générées, ce qui rendra le problème d'optimisation moins convexe.

4.4.2 Nomenclature du modèle d'expertise

L'idée derrière ce nouveau modèle sera de représenter chaque auteur comme une entité qui écrit des documents, qui possède une distribution de mots propre à elle et qui est caractérisée par un paramètre d'expertise. Dans les expériences futures, il sera parfois nécessaire de trouver des paramètres liés à un auteur en particulier. Ainsi, on distingue les paramètres liés à un auteur en particulier par l'indice a et ceux communs à tous les auteurs par l'indice c. Par

exemple, le paramètre ϕ que l'on retrouvait dans LDA classique était commun pour tous les auteurs et donc, on le désignera par ϕ_c. Or, lorsqu'on traitera des distributions de mots propres à un auteur (qu'elles soient le résultat d'une manipulation faite après l'application de LDA sur tout le corpus ou le résultat d'une application de LDA sur le sous-corpus de l'auteur), on les désignera par ϕ_a. Voici d'autres notes concernant la nomenclature :

- N_a désigne le nombre d'auteurs uniques dans un corpus.
- On applique l'indice a sur des matrices déjà désignées pour signifier que la matrice est par rapport à un auteur en particulier. Par exemple, on a la répartition de sujets par document θ qui était une matrice de dimension $[N_D \times N_K]$. On pourrait maintenant avoir la matrice θ_a qui représente la répartition moyenne des sujets pour un auteur et qui serait de dimension $[N_a \times N_K]$.
- La matrice F est une matrice de fréquence de mots utilisés par auteur. Elle est de dimension $[N_a \times N_W]$. Par exemple, l'indice $(1, 2)$ correspond aux nombres d'occurrences que le mot d'indice 2 a été utilisé dans le sous-corpus du premier auteur. Elle correspond à des observations directes.
- La matrice $F^{(K)}$ est une matrice désignant les fréquences de mots par auteur et par sujet. Elle est de dimension $[(N_a \times N_K) \times N_W]$. Comme les sujets sont inférés, cette matrice est obtenue suite à des manipulations mathématiques et ne découle pas directement des observations.
- La matrice γ est la matrice regroupant les paramètres d'expertise des auteurs. Elle est de dimension $[N_a \times N_K]$, puisque chaque auteur dispose d'un paramètre d'expertise par sujet.

4.4.3 Procédure d'inférence du paramètre d'expertise γ

Si on voulait inférer γ de manière optimale, on devrait utiliser une technique dite *end-to-end* intégrée dans l'algorithme du CGS. Toutefois, nous avons essayé d'implémenter une telle technique sans succès. Par conséquent, on a recours à une méthode alternative. Dans notre cas, pour procéder à l'inférence du paramètre d'expertise, on doit avoir une distribution de fréquences des mots par sujet et par auteur. Or, suite à l'application de LDA classique, ces distributions de mots ϕ sont communes à tous les auteurs et on ne peut donc pas *a priori* faire la distinction entre ceux-ci dans le but de leur attribuer un paramètre d'expertise propre. La principale différence du nouveau modèle par rapport au modèle LDA classique réside dans l'attribution de certaines portions du corpus à des auteurs. Il sera désormais possible d'attribuer des fréquences de mots et de sujets à des auteurs en particulier et, de cette manière, on pourra obtenir des distributions de mots/sujets propres à ceux-ci. Ensuite, comme expliqué dans la section 4.2, si on connaît les distributions de fréquence de mots

par sujet de chaque auteur, on pourra appliquer un *fit* de Mandelbrot sur celles-ci afin de déterminer la matrice γ. Détaillons les principales étapes qui nous permettront de procéder à cette inférence.

LDA classique sur tout le corpus

Une fois le corpus synthétique généré, on veut obtenir les distributions θ et ϕ qui sont communes à tout ce corpus (sans faire la distinction entre les auteurs). Pour ce faire, on applique LDA classique sur tout le corpus comme au chapitre 3 (avec la même méthode d'alignement des sujets de la section 3.2.2) et on obtient les matrices θ_c et ϕ_c.

Segmentation en fonction des auteurs

À cette étape, on ordonne la matrice θ_c de manière à ce qu'elle regroupe les documents de même auteur. Par exemple, si on a un corpus où 5 auteurs ont écrit chacun 100 documents, les 100 premières lignes de θ_c seront les documents de l'auteur 1 et ainsi de suite. On doit ensuite calculer la matrice θ_a qui regroupe les θ moyens de chaque auteur. Le paramètre $\theta_{a,k}$ qui indique l'hyperparamètre de Dirichlet pour la distribution du sujet k pour les documents de l'auteur a est obtenu par :

$$\theta_{a,k} = \frac{\sum_{d_a=1}^{N_{D,a}} \theta_{d,k}}{N_{Da}}$$

où N_{Da} représente le nombre de documents par auteur. Suite à l'évaluation de ces paramètres, on peut calculer la matrice θ_a qui est de dimension $[N_a \times N_K]$. Cette matrice illustre la proportion d'utilisation des sujets latents inférés à partir du corpus commun pour chaque auteur.

Ensuite, on calcule la matrice F des fréquences de mots par auteur. Cette matrice n'est que le résultat de la concaténation de la matrice X du chapitre 3 et de la réassignation des fréquences aux bons indices de mots. Après cette manipulation, on obtient la matrice F de dimension $[N_a \times N_w]$ qui illustre l'utilisation des différents mots par chaque auteur. Or, on doit transformer cette matrice afin de la rendre dépendante des sujets latents.

Matrice de fréquences par sujet par auteur

Pour obtenir $F_a^{(K)}$, on calcule la valeur espérée de F_a en se basant sur une probabilité obtenue de 3 méthodes différentes :

(A) Pour la valeur espérée de F_a par θ_a, on a :

$$F_{a,k} = \theta_{a,k} \times F_a$$

Ainsi, on pondère tout simplement la matrice des fréquences par la probabilité qu'un auteur a d'écrire sur un sujet donné. Après la concaténation, on obtient $F_a^{(K)}$ de dimension $[N_K \times N_W]$. Enfin, il suffit de concaténer sur les lignes les a matrices $F_a^{(K)}$ pour obtenir F^K de dimension $[(N_a \times N_K) \times N_W]$.

(B) Pour la méthode de la valeur espérée de F_a par ϕ_c, on n'utilise pas les informations contenues dans θ_a, car on juge que la valeur abstraite des sujets latents communs inférés dans ϕ_c est suffisante et même bénéfique pour l'inférence des matrices de fréquence propres aux auteurs. On calcule d'abord ϕ_{c-norm} en normalisant chaque colonne de ϕ_c pour obtenir des sommes de colonne unitaires. On possède désormais un résultat intéressant : la colonne w de ϕ_{c-norm} représente la probabilité générale d'attribution d'un mot w à un des sujets latents de ϕ_c, et ce, pour tous les auteurs sans distinction. Finalement, on peut calculer $F_a^{(K)}$ de la façon suivante :

$$F_a^{(K)} = \phi_{c-norm} \times \text{diag}(F_a)$$

On procède de façon analogue pour obtenir $F^{(K)}$ à partir de $F_a^{(K)}$.

(C) Pour la méthode de la valeur espérée de F_a par θ_a et ϕ_c, on doit d'abord transformer la matrice ϕ_c en ϕ_a. Étant donné que ϕ_c représente la distribution moyenne des mots pour chaque sujet en termes d'auteurs, on peut pondérer par l'utilisation moyenne d'un sujet par un auteur afin d'obtenir la distribution des mots par sujet pour un auteur en particulier. De façon explicite, on obtient ϕ_a comme suit :

$$\phi_a = \text{diag}(\theta_a) \times \phi_c$$

Ensuite, on calcule $\phi_{a-\text{norm}}$ en normalisant chaque colonne de ϕ_a pour obtenir des sommes de colonne unitaires. On a donc que la colonne w de $\phi_{a-\text{norm}}$ représente la probabilité d'attribution à un sujet du mot w pour un auteur a. On peut donc calculer $F_a^{(K)}$:

$$F_a^{(K)} = \phi_{a-norm} \times \text{diag}(F_a)$$

Évaluation de γ

La dernière étape de ce processus consiste à estimer la matrice γ. Pour trouver $\gamma_{a,k}$, c'est-à-dire le paramètre d'expertise d'un auteur a pour un sujet k, on calcule le paramètre c de Mandelbrot qui donne le meilleur *fit* avec la ligne normalisée et ordonnée k de F_a. Mathématiquement, on a :

$$\gamma_{a,k} = \arg\min_{c} |F_{a,k} - f_{\text{Mand}}(c)|$$

Les $\gamma_{a,k}$ pourront être calculés à l'aide des méthodes des MCNL ou de l'EMV. L'interprétation du paramètre d'expertise est donc analogue à celle liée au paramètre de Mandelbrot. La plage attendue des valeurs s'étend sur l'intervalle $[0, \infty]$ où 0 représente une expertise parfaite et ∞ représente une expertise nulle. Notons que, puisqu'il s'agit d'une prédiction, la matrice γ inférée par ce processus sera notée $\hat{\gamma}$.

Exemple du processus

Terminons cette section en illustrant le processus d'inférence de l'expertise par un exemple. Puisque la méthode de la pondération de F par θ_a et ϕ_c est la plus complexe, nous allons expliciter celle-ci. Prenons le cas où 2 auteurs écrivent 2 documents chacun où 10 mots par document sont tirés d'un vocabulaire de 4 mots et où on spécifie 3 sujets latents.

Après avoir appliqué LDA sur tout le corpus, on obtient les matrices θ_c, ϕ_c et F suivantes :

$$\theta_c = \begin{bmatrix} 0.3 & 0.4 & 0.3 \\ 0.6 & 0.2 & 0.2 \\ 0.1 & 0.2 & 0.7 \\ 0.1 & 0.3 & 0.6 \end{bmatrix}, \phi_c = \begin{bmatrix} 0.3 & 0.2 & 0.1 & 0.4 \\ 0.3 & 0.3 & 0.3 & 0.1 \\ 0.1 & 0.7 & 0.1 & 0.1 \end{bmatrix}, F = \begin{bmatrix} 6 & 4 & 2 & 8 \\ 1 & 4 & 11 & 4 \end{bmatrix}$$

Notons que les données dans les matrices ont été choisies aléatoirement. Les deux premières lignes de θ_c représentent la répartition de sujets des documents de l'auteur 1. De plus, F_1 représente la fréquence d'utilisation de chaque mot du vocabulaire pour l'auteur 1. Calculons θ_a :

$$\theta_a = \begin{bmatrix} 0.45 & 0.3 & 0.25 \\ 0.1 & 0.25 & 0.65 \end{bmatrix}$$

Chaque ligne de θ_a forme aussi une distribution. Montrons maintenant ϕ_{1-norm}, soit les distributions de mots par sujet de l'auteur 1 :

$$\phi_{1-norm} = \begin{bmatrix} 0.5400 & 0.2535 & 0.2813 & 0.7660 \\ 0.3600 & 0.2535 & 0.5625 & 0.1277 \\ 0.1000 & 0.4930 & 0.1563 & 0.1064 \end{bmatrix}$$

Cette matrice regroupe les probabilités d'appartenance à un certain sujet de chaque mot pour l'auteur 1. Par exemple, le premier mot peut appartenir au sujet 1 avec une probabilité de 0.54, au sujet 2 avec une probabilité de 0.36 et au sujet 1 avec une probabilité de 0.1. Notons que les colonnes de cette matrice forment une distribution. Si on compare la première colonne de ϕ_{1-norm} avec la première colonne de ϕ_c, la probabilité d'appartenance du mot 1 au sujet 1 par rapport au sujet 2 est bonifiée lorsque l'on sait que le document a été écrit par l'auteur 1. Ceci est logique, puisque l'on sait également que l'auteur 1 est plus probable d'écrire sur le sujet 1 par rapport au sujet 2 comme le témoigne sa distribution θ_{a1}. Calculons maintenant $F_{a1}^{(K)}$:

$$F_{a1}^{(K)} = \begin{bmatrix} 3.2400 & 1.0141 & 0.5625 & 6.1277 \\ 2.1600 & 1.0141 & 1.1250 & 1.0213 \\ 0.6000 & 1.9718 & 0.3125 & 0.8511 \end{bmatrix}$$

Ainsi, la première ligne de F est répartie selon les sujets. Par exemple, l'auteur 1 a utilisé le mot 1 à 6 reprises. Or, selon notre calcul, on a qu'une proportion de $\frac{3.24}{6}$ de cette fréquence est attribuable au sujet 1, $\frac{2.16}{6}$ au sujet 2 et $\frac{0.6}{6}$ au sujet 3. Une fois FK calculée, on possède toutes les informations nécessaires pour déterminer les γ, puisqu'on dispose des fréquences par sujet et par auteur.

Pour l'auteur 1, les γ qu'on obtiendrait après l'application des MCNL serait :

$$\gamma_{a1} = [3.53, 1.56, 3.07]$$

Pour cet exemple, l'auteur 1 posséderait le plus d'expertise dans le sujet 2 et le moins d'expertise dans le sujet 1, et ce, même si ses écritures portent davantage sur ce sujet. Bien sûr, il s'agit ici d'un cas inventé de toutes pièces (aucune corrélation entre les valeurs choisies pour θ_c et ϕ_c) afin d'illustrer la logique des calculs ; il serait étonnant, quoique possible, d'observer pour un auteur une expertise plus faible dans un sujet qu'il traite davantage.

4.5 Expérience et présentation des résultats

Cette section porte sur les expériences faites dans le but de valider les méthodes proposées pour l'inférence de l'expertise des auteurs. Celle-ci sera divisée comme suit. D'abord, on établira les hyperparamètres utilisés pour les expériences. Ensuite, la méthodologie des expériences ainsi que les métriques recueillies seront expliquées. On poursuivra par la présentation des résultats. On conclura par une discussion sur ceux-ci et une analyse de sensibilité des hyperparamètres.

4.5.1 Hyperparamètres utilisés

De la même façon qu'au chapitre 3, on montre les hyperparamètres qui caractériseront notre modèle ainsi que notre cadre de validation :

- $N_V = 1000$
- $N_a = 5$ (nombre d'auteurs uniques)
- $N_{\text{WD}} = 100$
- $N_K = 3$
- On garde un hyperparamètre $\alpha = 0.7$ pour la génération de θ. Toutefois, ϕ sera dorénavant généré par une distribution de Mandelbrot dont les paramètres seront définis plus loin.
- $N_{Da} = 100$ (nombre de documents par auteur). On décide de fixer ce paramètre à 100 pour rester comparable aux expériences du chapitre 3 pour les divergences KL calculées pour la performance.
- $N_E = 10$ (nombre d'exécutions). Comme au chapitre 3, on exécutera le cadre de validation à 10 reprises pour réduire la variance sur les métriques de performance obtenues.
- L'inférence se fera à l'aide de l'algorithme de *Gibbs Sampling* codé maison. Hormis

quelques modifications concernant la forme segmentée par auteur, le processus d'inférence général de l'algorithme reste le même.

4.5.2 Méthodologie

L'objectif de cette sous-section est d'établir notre méthodologie de test du modèle d'expertise. À la manière du chapitre 3, nous allons faire l'utilisation du cadre de validation pour vérifier la performance du modèle. Pour ce faire, 4 expériences seront menées. Pour chaque expérience, nous allons spécifier une matrice de paramètres d'expertise γ (propre à cette expérience et à un contexte d'expertise donné) et nous allons procéder à la génération de données synthétiques en nous basant sur cette matrice. Ensuite, nous allons utiliser 12 différentes méthodes pour tenter de retrouver la matrice γ et nous calculerons certaines métriques de performance comparant la similarité entre les distributions générées et inférées. Abordons d'abord ce qui caractérise les expériences pour ensuite traiter des méthodes de recherche de γ.

Expériences

Les expériences sont caractérisées par un choix spécifique de la matrice γ de génération. Pour chaque expérience, on créera γ selon un contexte d'expertise donné. On teste plusieurs contextes d'expertise afin d'évaluer la performance du modèle en contrôlant un maximum de variables. De cette manière, on pourra valider si la méthode performe mieux pour les niveaux d'expertise bas ou élevés et on pourra repérer plus facilement les anomalies. Notons que, pour les expériences 1 à 3, on échantillonnera l'expertise sur une distribution uniforme avec des bornes de ± 0.1 autour du γ visé. Voici ces contextes :

- **Expérience 1** : On suppose que tous les auteurs sont experts dans chaque sujet. En se référant à la figure 3.28, on constate que le plus haut β étudié, soit $\beta = 5$, correspond à un paramètre c de Mandelbrot de 0.35. La matrice d'expertise sera donc $\gamma \sim U(0.34, 0.36)$.
- **Expérience 2** : On suppose que tous les auteurs sont novices dans chaque sujet. En se référant à la figure 3.28, on constate que le plus faible β étudié soit $\beta = 0.1$ correspond à un paramètre c de Mandelbrot de 1.5. La matrice d'expertise sera donc $\gamma \sim U(1.4, 1.6)$.
- **Expérience 3** : On suppose que tous les auteurs possèdent une expertise moyenne dans chaque sujet. En se référant à la figure 3.28 et à la conclusion du chapitre 3 stipulant qu'une génération réaliste était caractérisée par des hyperparamètres de Dirichlet

$\alpha = 0.7$ et $\beta = 0.5$, on constate que l'intersection de ces hyperparamètres correspond à un paramètre c de 0.87. La matrice d'expertise sera donc $\gamma \sim U(0.77, 0.97)$.

— **Expérience 4** : On suppose que les auteurs possèdent une expertise répartie sur un large intervalle d'expertise $\gamma \sim U(0.25, 1.6)$ pour chaque sujet.

Évidemment, c'est l'expérience 4 qui représente le cas le plus intéressant, car c'est celle-ci qui reproduit un comportement que l'on pourrait observer dans la réalité. Or, il sera pertinent d'étudier le comportement pour des expertises constantes afin d'identifier de potentielles anomalies de nos méthodes d'inférence.

Méthodes d'inférence

Pour chacune des expériences, on utilisera 12 méthodes pour calculer une prédiction de la matrice γ, notée $\hat{\gamma}$. Les méthodes 1 à 4 représentent des seuils d'excellence ou de médiocrité des performances, les méthodes 5 et 6 explorent l'effet d'utiliser LDA à multiples reprises sur les sous-corpus des auteurs alors que les méthodes 7 à 12 sont des déclinaisons des méthodes **(A), (B), (C)** décrites précédemment. Ce sont les méthodes 7 à 12 qui tentent de répondre à la présente question de recherche et les autres méthodes font office de comparaison. Voici un tableau récapitulatif qui résume ces méthodes et des détails sur les méthodes de comparaison suivront (pour consulter les détails de **(A), (B), (C)**, voir section 4.4.3) :

Tableau 4.3 Présentation des méthodes d'inférence de γ du modèle d'expertise

Numérotation	Type	Méthode d'inférence de γ
M_1	Seuil de médiocrité	MCNL
M_2	Seuil de médiocrité	EMV
M_3	Seuil d'excellence	MCNL
M_4	Seuil d'excellence	EMV
M_5	Sous-corpus	MCNL
M_6	Sous-corpus	EMV
M_7	Méthode **(A)**	MCNL
M_8	Méthode **(A)**	EMV
M_9	Méthode **(B)**	MCNL
M_{10}	Méthode **(B)**	EMV
M_{11}	Méthode **(C)**	MCNL
M_{12}	Méthode **(C)**	EMV

Voici maintenant les détails de chaque méthode de comparaison :

— **Seuil de médiocrité** : On génère des distributions de fréquence par auteur et par sujet aléatoires (initialement tirées de $U(0, 1)$ puis normalisées) et on applique le *fit* de

Mandelbrot afin de trouver $\hat{\gamma}$. On s'attend à ce que cette méthode donne de mauvais résultats mais on l'inclut à des fins de comparaison avec les meilleures méthodes. On référera à cette méthode par « Aléatoire ».

— **Seuil d'excellence** : On applique l'inférence de γ sur la matrice $F^{(K)}$ obtenue à partir des distributions générées, c'est-à-dire sur les distributions ϕ obtenues suite au processus de génération. C'est cette méthode qui devrait nous donner les meilleures performances et on pourra connaître quel est le résultat optimal attendu. Or, il est évident que l'intérêt de celle-ci n'est aussi présent qu'à des fins de comparaisons avec les méthodes subséquentes, puisqu'aucune inférence n'a lieu. On référera à cette méthode par « Généré ».

— **Sous-corpus** : On segmente les sous-corpus des auteurs et on applique LDA sur chaque sous-corpus indépendamment l'un de l'autre. Les sujets latents trouvés seront propres à un sous-corpus en particulier et ne pourront donc pas être généralisés pour l'ensemble des auteurs. Cependant, on pourra vérifier l'impact de cette segmentation sur l'inférence de γ. Une fois les distributions de fréquence par auteur et par sujet trouvées, on applique les MCNL pour trouver $\hat{\gamma}$. On référera à cette méthode par « Sous-corpus ».

Alors, rappelons que les méthodes 1 et 2 constituent un seuil de médiocrité tandis que les méthodes 3 et 4 constituent un seuil d'excellence. Les méthodes 7 à 12 constituent la cible du mémoire alors que les méthodes 5 et 6 servent à déterminer si on aurait intérêt à appliquer LDA plusieurs fois au sein d'un corpus. On référera aux méthodes 7 à 12 par leur lettre appropriée la section (4.4.3).

4.5.3 Métriques de performance

En ce qui concerne l'estimation de γ, on aura recours à la distance de corrélation (section 3.2.3), ainsi qu'à la RMSE (*Root Mean Square Error*) dont voici l'équation :

$$\text{RMSE} = \sqrt{\frac{\sum_{i=i}^{N}(\gamma_i - \hat{\gamma}_i)^2}{N}}$$

On utilise la RMSE et la corrélation pour évaluer la performance sur l'inférence de γ, car il est important de savoir si l'erreur absolue est grande (capturé par la RMSE) et si le comportement relatif des expertises d'une même expérience est similaire (capturé par la distance de corrélation).

De plus, on calculera une divergence KL (équation en section 3.2.3) entre les distributions

plus classiques θ et ϕ obtenues de différentes façons. On les présente dans un tableau où les colonnes identifient le nom de la métrique ainsi que les deux distributions comparées :

Tableau 4.4 Présentation des divergences KL calculées pour la validation du modèle d'expertise

Métrique	Distribution 1	Distribution 2
KL_{θ_c}	θ généré	θ_c inféré
KL_{ϕ_c}	$\phi \sim \mathrm{Mand}(0.87)$ généré	ϕ_c inféré
$\mathrm{KL}_{\theta_{\mathrm{sc}}}$	θ_a généré	θ inféré du sous-corpus
$\mathrm{KL}_{\phi_{\mathrm{sc}}}$	ϕ_a généré	ϕ inféré du sous-corpus
KL_{θ_a}	θ_a généré	θ_a inféré
$\mathrm{KL}_{\phi_{a7-8}}$	ϕ_a généré	$F^{(K)}$ des méthodes 7 et 8
$\mathrm{KL}_{\phi_{a9-10}}$	ϕ_a généré	$F^{(K)}$ des méthodes 9 et 10
$\mathrm{KL}_{\phi_{a11-12}}$	ϕ_a généré	$F^{(K)}$ des méthodes 11 et 12

Notons que la matrice $F^{(K)}$ est transformée en distributions avant le calcul de la divergence KL.

4.5.4 Présentation des résultats

C'est dans cette section que l'on rapportera les principaux résultats et observations des expériences définies dans la section 4.5.3. Ceux-ci seront par la suite interprétés dans la section 4.5.5. Rappelons que l'on performe le modèle sur un corpus avec 5 auteurs uniques et où on spécifie 3 sujets latents.

Expérience 1

Rappelons que le but de cette expérience est de supposer que tous les auteurs sont uniformément experts dans chaque sujet en posant $\gamma \sim U(0.34, 0.36)$ pour la génération. On recueille les résultats dans les deux tableaux suivants. D'abord, on présente une synthèse des performances de l'inférence de γ, soit les métriques obtenues suite à l'application des 12 méthodes développées précédemment :

Tableau 4.5 Synthèse des performances de l'inférence de γ pour l'expérience 1

Méthodes	Dr	RMSE
M_1 : (Aléatoire-MCNL)	0.90	0.059
M_2 : (Aléatoire-EMV)	0.94	0.066
M_3 : (Généré-MCNL)	0.022	0.043
M_4 : (Généré-EMV)	0.12	0.069
M_5 : (Sous-corpus-MCNL)	0.48	0.076
M_6 : (Sous-corpus-EMV)	0.66	0.15
M_7 : (**A**-MCNL)	0.51	0.061
M_8 : (**A**-EMV)	0.71	0.051
M_9 : (**B**-MCNL)	0.30	0.096
M_{10} : (**B**-EMV)	0.49	0.19
M_{11} : (**C**-MCNL)	0.43	0.093
M_{12} : (**C**-EMV)	0.83	0.18

Les distances de corrélation sont toutes mauvaises sauf celles trouvées par la méthode 3 et 4 sur les distributions générées. Toutefois, les RMSE sont basses pour toutes les méthodes MCNL, comparativement à ce que l'on observera pour les expériences subséquentes. De ce fait, on conclut que lorsque les distributions de mots sont quasi uniformes, l'algorithme comprend qu'une personne a un niveau d'expertise élevé du sujet, mais il a de la difficulté à départager les experts les uns des autres. En somme, à partir d'un certain seuil d'expertise, il est possible d'inférer ce seuil d'expertise, mais il est difficile d'évaluer le niveau relatif une fois ce seuil passé. De plus, la RMSE de la distribution aléatoire est faible : ceci est attendu puisque la distribution aléatoire est synonyme de répartition presque uniforme des mots, ce qui s'apparente bien à une distribution avec un paramètre de Mandelbrot faible. En effet, notre analogie de l'expertise stipule qu'un expert peut sélectionner n'importe quel mot du vocabulaire sans limites.

On illustre la qualité de l'inférence avec un graphique de la comparaison entre les γ inférés et le *ground truth*, soit les γ générés en l'occurrence. Les points de couleur noire représentent le *ground truth*, les points de couleur bleue représente les γ inférés par les MCNL et les points de couleur rouge représente les γ inférés par l'EMV. On représente l'inférence pour les méthodes 9 et 10.

Figure 4.4 Comparaison entre les γ inférés avec les méthodes 9-10 et le *ground truth* pour l'expérience 1

On voit bien dans ce graphique le comportement observé dans les résultats, soit une distance de corrélation élevée et une RMSE faible. Les deux estimations surévaluent le *ground truth*, et cette tendance est plus marquée par l'EMV. Le biais est légèrement capturé (une erreur importante subsiste) alors que la tendance relative n'est pas capturée. Ce phénomène est probablement dû au fait que, étant donné que les distributions de mots de chaque auteur sont très semblables, il est difficile de les départager et ce phénomène est plus marqué plus le niveau d'expertise est haut (ce qui cause la plus grande erreur de biais à mesure que le paramètre d'expertise descend).

On présente maintenant les divergences KL calculées sur les distributions classiques :

Tableau 4.6 Synthèse des divergences KL pour l'expérience 1

Méthodes	θ	ϕ
Commun (c)	0.45	0.71
Sous-corpus (sc)	0.53	0.28
Par auteur (a_{7-8})	0.0048	0.11
Par auteur (a_{9-10})	0.0048	0.34
Par auteur (a_{11-12})	0.0048	0.34

Rappelons que les divergences KL qui avaient été trouvées au chapitre 3 pour une génération-inférence avec LDA classique où $\alpha = 0.7$ et $\beta = 5$ (le point de comparaison, puisque le meilleur *fit* de Mandelbrot avec cette distribution était à $c = 0.35$) était $\mathrm{KL}_\theta = 0.48$ et $\mathrm{KL}_\phi = 0.28$. On remarque d'abord que KL_{θ_a}, soit la divergence KL calculée sur la moyenne des θ par auteur est très faible. Ceci s'explique par le fait que, parce que les auteurs possèdent un grand nombre de documents chacun et que la répartition des sujets de ces derniers est dictée par une loi de Dirichlet symétrique, les moyennes des θ par auteur seront d'environ $\theta_a = [0.33, 0.33, 0.33]$. Cette valeur est donc moins intéressante pour la présente comparaison, mais elle sert de vérification sommaire. Cependant, les autres valeurs de ce tableau sont toutes pertinentes. D'abord, les divergences KL liées à θ sont similaires à ce que l'on a obtenu avec LDA classique, ce qui est normal, puisque l'on n'a pas changé la façon de générer et inférer la répartition de sujets par document. Une divergence KL de 0.45 étant relativement haute, on comprend que le modèle a encore de la difficulté à associer les bons sujets aux documents lorsque les distributions de mots par sujet sont très uniformes. Ensuite, pour ce qui est des divergences KL liées à ϕ, les ϕ_c sont peu similaires, ce qui est attendu (car la ϕ_c générée hérite d'une expertise moyenne et ne tient donc pas compte de l'expertise propre à chaque auteur). Aussi, la divergence KL par sous-corpus est comparable à la valeur de LDA classique, ce qui est logique. Or, les valeurs de KL_{ϕ_a} sont intéressantes à analyser. D'abord, ce n'est pas parce que le KL_{ϕ_a} est faible que la distance de corrélation entre les γ générés et inférés sera faible aussi. En effet, on constate que KL_{ϕ_a} est faible pour les méthodes 7-8, mais que la distance de corrélation est la plus élevée des méthodes 7 à 12 (probablement à cause du faible biais des prédictions).

On représente les distributions ϕ du tableau (sauf ϕ_{sc}) dans un graphique pour en comparer l'allure avec la distribution ϕ générée. La distribution générée est représentée en rouge alors que les différentes distributions inférées sont illustrées en noir. Notons que l'on représente les distributions pour un auteur unique seulement, car puisque l'on fixe le niveau d'expertise pour tous les auteurs, la différence entre les auteurs serait minime.

Figure 4.5 Les distributions ϕ générées et inférées de l'expérience 1 pour les sujets 1 (gauche), 2 (milieu) et 3 (droite)

Comme les γ propres aux différents sujets sont tous bas, les distributions sont quasi uniformes. La distinction entre les sujets est donc moins évidente. On voit aussi que, étant donné que les auteurs sont tous experts dans tous les sujets, ils pourront utiliser tous les mots sans exception, et ce, de manière relativement équiprobable.

On finit par montrer la différence entre la distribution ϕ_{a9-10} et la distribution ϕ générée, une fois celles-ci ordonnées selon le rang des fréquences :

Figure 4.6 La distribution ϕ générée comparée à celle inférée par la méthode 9-10 de l'expérience 1 ordonnées selon le rang des fréquences pour le sujet 1 (gauche), 2 (milieu) et 3 (droite)

L'inférence a donc plus de difficultés à capturer le comportement relativement uniforme de la distribution générée, mais c'est ce qui était aussi observé pour LDA classique pour un cas où $\beta = 5$.

Expérience 2

Le but de cette expérience est de supposer que tous les auteurs sont uniformément novices dans chaque sujet en posant $\gamma \sim U(1.4, 1.6)$ pour la génération. On recueille les résultats dans les deux tableaux suivants.

D'abord, on présente une synthèse des performances de l'inférence de γ, soit les métriques obtenues suite à l'application des 12 méthodes développées précédemment :

Tableau 4.7 Synthèse des performances de l'inférence de γ pour l'expérience 2

Méthodes	Dr	RMSE
M_1 : (Aléatoire-MCNL)	0.91	1.20
M_2 : (Aléatoire-EMV)	0.89	1.11
M_3 : (Généré-MCNL)	0.047	0.015
M_4 : (Généré-EMV)	0.080	0.021
M_5 : (Sous-corpus-MCNL)	0.56	0.24
M_6 : (Sous-corpus-EMV)	0.63	0.37
M_7 : (**A**-MCNL)	0.54	0.41
M_8 : (**A**-EMV)	0.63	0.31
M_9 : (**B**-MCNL)	0.13	0.026
M_{10} : (**B**-EMV)	0.28	0.041
M_{11} : (**C**-MCNL)	0.10	0.024
M_{12} : (**C**-EMV)	0.26	0.047

Les performances sont mauvaises pour les méthodes 1, 2, 5, 6, 7 et 8. En ce qui concerne les méthodes 1 et 2, il est logique que, dès que l'on observe une priorisation de certains mots dans le vocabulaire, ses performances soient médiocres. Pour les méthodes 5 et 6, il est intéressant de constater que le fait d'avoir des sujets latents qui ne correspondent pas à ceux qui ont été générés a un impact sur la qualité de l'inférence du paramètre d'expertise. Pour les méthodes 7 et 8, on conclut qu'il est névralgique d'utiliser la granularité présente dans la matrice ϕ_c pour un cas où un déphasage important dans l'utilisation des mots est observé. Il est aussi pertinent de voir que la distance de corrélation ainsi que la RMSE sont bonnes pour les méthodes 7, 8, 9 et 10. Ceci nous prouve que l'algorithme développé fonctionne.

On illustre la qualité de l'inférence avec un graphique de la comparaison entre les γ inférés et le *ground truth*. Le code de couleur est le même que celui utilisé précédemment.

Figure 4.7 Comparaison entre les γ inférés avec les méthodes 9-10 et le *ground truth* pour l'expérience 2

Ce graphique nous démontre la bonne inférence de γ pour cette expérience. Bien qu'on observe une petite sous-évaluation du *ground truth* pour les deux méthodes, les résultats sont très bons.

On présente maintenant les divergences KL calculées sur les distributions classiques :

Tableau 4.8 Synthèse des divergences KL pour l'expérience 2

Méthodes	θ	ϕ
Commun (c)	0.020	0.45
Sous-corpus (sc)	0.087	0.30
Par auteur (a_{7-8})	0.0029	0.90
Par auteur (a_{9-10})	0.0029	0.12
Par auteur (a_{11-12})	0.0029	0.11

Rappelons que les divergences KL qui avaient été trouvées au chapitre 3 pour une génération-inférence avec LDA classique où $\alpha = 0.7$ et $\beta = 0.1$ était $\mathrm{KL}_\theta = 0.10$ et $\mathrm{KL}_\phi = 0.30$. Les métriques KL_{θ_c} et $\mathrm{KL}_{\theta_{sc}}$ sont plus faibles que ce qu'on avait obtenu avec LDA classique et ceci est potentiellement dû au fait que l'on a 500 documents au total au lieu de 100 documents. Ce n'est donc pas alarmant. Aussi, les similarités KL_{ϕ_a} des méthodes 9, 10, 11 et 12 sont meilleures que $\mathrm{KL}_{\phi_{sc}}$ pour cette plage de γ générée. Puisqu'on arrive facilement à bien segmenter les sujets et les distributions de mots qui leur sont associés, on comprend que notre méthode développée en 4.4.3 pour inférer les fréquences de mots par auteur et par sujet a d'autant plus de valeur dans ce cas. On voit aussi l'impact d'impliquer ϕ_c dans la pondération de F démontré par la piètre performance de $\mathrm{KL}_{\phi_{a7-8}}$. Par ailleurs, il est assez étonnant de constater que $\mathrm{KL}_{\phi_{a9-10}}$ et $\mathrm{KL}_{\phi_{a_{11-12}}}$ soient meilleures que la divergence KL qui avait été obtenue au chapitre 3. Ceci peut être expliqué par la comparaison imparfaite entre une inférence faite sur des ϕ générées par Dirichlet versus des ϕ générées par Mandelbrot, par la sensibilité de la métrique de performance à des hyperparamètres comme le nombre de documents par personne ou simplement par le fait qu'il serait préférable de déterminer l'expertise des auteurs en tout temps afin d'obtenir une représentation des mots par sujet

plus juste.

On représente les distributions ϕ du tableau (sauf ϕ_{sc}) dans un graphique pour en comparer l'allure avec la distribution ϕ générée :

Figure 4.8 Les distributions ϕ générées et inférées de l'expérience 2 pour les sujets 1 (gauche), 2 (milieu) et 3 (droite)

Ici, les sujets sont bien délimités et bien différents des autres. Les valeurs de γ étant tous élevées, les novices dans un sujet donné n'utiliseront qu'une poignée de mots simples.

On finit par montrer la différence entre la distribution ϕ_{a9-10} et la distribution ϕ générée, une fois celles-ci ordonnées selon le rang des fréquences :

Figure 4.9 La distribution ϕ générée comparée à celle inférée par la méthode 9-10 de l'expérience 2 ordonnées selon le rang des fréquences pour le sujet 1 (gauche), 2 (milieu) et 3 (droite)

Malgré quelques anomalies, l'inférence est donc adéquate.

Expérience 3

Rappelons que le but de cette expérience est de supposer que tous les auteurs sont d'expertise moyenne dans chaque sujet en posant $\gamma \sim U(0.77, 0.97)$ pour la génération. On recueille les résultats dans les deux tableaux suivants.

D'abord, on présente une synthèse des performances de l'inférence de γ, soit les métriques obtenues suite à l'application des 12 méthodes développées précédemment :

Tableau 4.9 Synthèse des performances de l'inférence de γ pour l'expérience 3

Méthodes	Dr	RMSE
M_1 : (Aléatoire-MCNL)	0.91	0.57
M_2 : (Aléatoire-EMV)	0.93	0.48
M_3 : (Généré-MCNL)	0.023	0.011
M_4 : (Généré-EMV)	0.069	0.025
M_5 : (Sous-corpus-MCNL)	0.15	0.12
M_6 : (Sous-corpus-EMV)	0.22	0.12
M_7 : (**A**-MCNL)	0.53	0.21
M_8 : (**A**-EMV)	0.55	0.17
M_9 : (**B**-MCNL)	0.10	0.019
M_{10} : (**B**-EMV)	0.20	0.033
M_{11} : (**C**-MCNL)	0.13	0.022
M_{12} : (**C**-EMV)	0.32	0.038

Les résultats d'inférence sont similaires à ceux obtenus à l'expérience 2. Ce résultat est attendu, car l'expérience 3 est un cas de figure moins extrême, mais partageant les mêmes caractéristiques que celui étudié à l'expérience 2 (la segmentation des sujets est bien délimitée). On observe cependant une meilleure performance des méthodes par sous-corpus même si l'algorithme développé des méthodes 9, 10, 11 et 12 donne une performance significativement meilleure (surtout en termes de RMSE).

On illustre la qualité de l'inférence avec un graphique de la comparaison entre les γ inférés et le *ground truth*. Le code de couleur est le même que celui utilisé précédemment.

Figure 4.10 Comparaison entre les γ inférés avec les méthodes 9-10 et le *ground truth* pour l'expérience 3

On remarque aussi une répartition des estimations similaire à celle qu'on a trouvée pour l'expérience 2 quoiqu'on pourrait noter ici un biais inférieur.

On représente maintenant les divergences KL calculées sur les distributions classiques.

Tableau 4.10 Synthèse des divergences KL pour l'expérience 3

Méthodes	θ	ϕ
Commun (c)	0.058	0.10
Sous-corpus (sc)	0.086	0.21
Par auteur (a_{7-8})	0.0022	0.43
Par auteur (a_{9-10})	0.0022	0.16
Par auteur (a_{11-12})	0.0022	0.16

Rappelons que les divergences KL qui avaient été trouvées au chapitre 3 pour une génération-inférence avec LDA classique où $\alpha = 0.7$ et $\beta = 0.5$ était $\text{KL}_\theta = 0.10$ et $\text{KL}_\phi = 0.30$. Les conclusions que l'on tire pour ce tableau sont analogues à celles de l'expérience 2, tout en constatant un avantage moins marqué de KL_a par rapport à KL_{sc}.

On représente les distributions ϕ du tableau (sauf ϕ_{sc}) dans un graphique pour en comparer l'allure avec la distribution ϕ générée :

Figure 4.11 Les distributions ϕ générées et inférées de l'expérience 3 pour les sujets 1 (gauche), 2 (milieu) et 3 (droite)

Le comportement ici est analogue à celui observé pour l'expérience 2.

On finit par montrer la différence entre la distribution ϕ_{a9-10} et la distribution ϕ générée, une fois celles-ci ordonnées selon le rang des fréquences :

Figure 4.12 La distribution ϕ générée comparée à celle inférée par la méthode 9-10 de l'expérience 3 ordonnées selon le rang des fréquences pour le sujet 1 (gauche), 2 (milieu) et 3 (droite)

L'inférence est donc adéquate.

Expérience 4

Le but de cette expérience est de supposer que les auteurs possèdent une expertise différente en fonction des sujets en posant $\gamma \sim U(0.25, 1.6)$ pour chaque auteur et chaque sujet. On recueille les résultats dans les deux tableaux suivants.

D'abord, on présente une synthèse des performances de l'inférence de γ, soit les métriques obtenues suite à l'application des 12 méthodes développées précédemment :

Tableau 4.11 Synthèse des performances de l'inférence de γ pour l'expérience 4

Méthodes	Dr	RMSE
M_1 : (Aléatoire-MCNL)	1.05	0.64
M_2 : (Aléatoire-EMV)	0.99	0.57
M_3 : (Généré-MCNL)	0.0010	0.017
M_4 : (Généré-EMV)	0.0027	0.034
M_5 : (Sous-corpus-MCNL)	0.012	0.16
M_6 : (Sous-corpus-EMV)	0.023	0.21
M_7 : (**A**-MCNL)	0.54	0.32
M_8 : (**A**-EMV)	0.48	0.30
M_9 : (**B**-MCNL)	0.10	0.12
M_{10} : (**B**-EMV)	0.13	0.16
M_{11} : (**C**-MCNL)	0.22	0.18
M_{12} : (**C**-EMV)	0.30	0.22

La meilleure méthode d'inférence pour le cas réel simulé dans cette expérience est la méthode 9. Par ailleurs, son avantage sur les autres méthodes est plus marqué que pour les expériences précédentes. On constate aussi une meilleure performance des sous-corpus, ce qui est normal puisque les auteurs ont un niveau d'expertise différent les uns des autres. En somme, tous les résultats sont sensés.

On illustre la qualité de l'inférence avec un graphique de la comparaison entre les γ inférés et le *ground truth*. Le code de couleur est le même que celui utilisé précédemment.

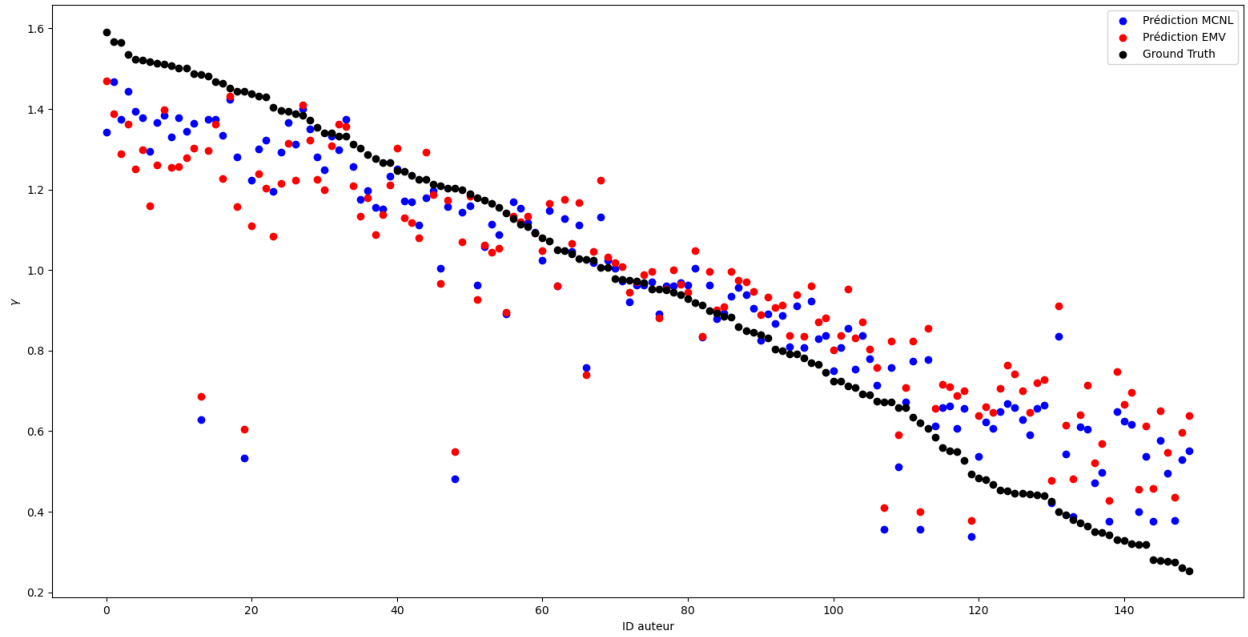


Figure 4.13 Comparaison entre les γ inférés avec les méthodes 9-10 et le *ground truth* pour l'expérience 4

À la lumière de ces représentations, on comprend que l'inférence est en général meilleure (sauf quelques anomalies) pour les hauts γ, soit les novices, tandis que cette inférence a plus de difficulté à capturer les faibles γ, soit les experts purs. C'est aussi cette tendance qui avait été observée dans les précédentes expériences. En bref, on conclut que l'inférence est acceptable, car même si les faibles γ sont plus ardus à prédire, leur estimation est passable.

Tableau 4.12 Synthèse des divergences KL pour l'expérience 4

Méthodes	θ	ϕ
Commun (c)	0.21	0.22
Sous-corpus (sc)	0.16	0.24
Par auteur (a_{7-8})	0.09	0.55
Par auteur (a_{9-10})	0.09	0.25
Par auteur (a_{11-12})	0.09	0.30

Puisque nos distributions ϕ générées sont teintées par l'expertise variable des auteurs, on ne

possède pas de comparaison directe avec des divergences KL obtenues pour LDA classique au chapitre 3. Alors, le seul élément intéressant qu'on peut constater dans ce tableau est que la KL_{a9-10} est comparable à la KL_{sc}.

Étant donné qu'on dispose maintenant de niveaux d'expertise différents en fonction de l'auteur, on peut montrer les distributions ϕ du tableau (sauf ϕ_{sc}) pour un auteur en particulier. Montrons d'abord les distributions inférées pour l'auteur 1 sélectionné au hasard dans le corpus. Notons que les paramètres γ qui caractérisent le niveau d'expertise de cet auteur sont $\gamma_1 = [0.44, 1.04, 0.76]$.

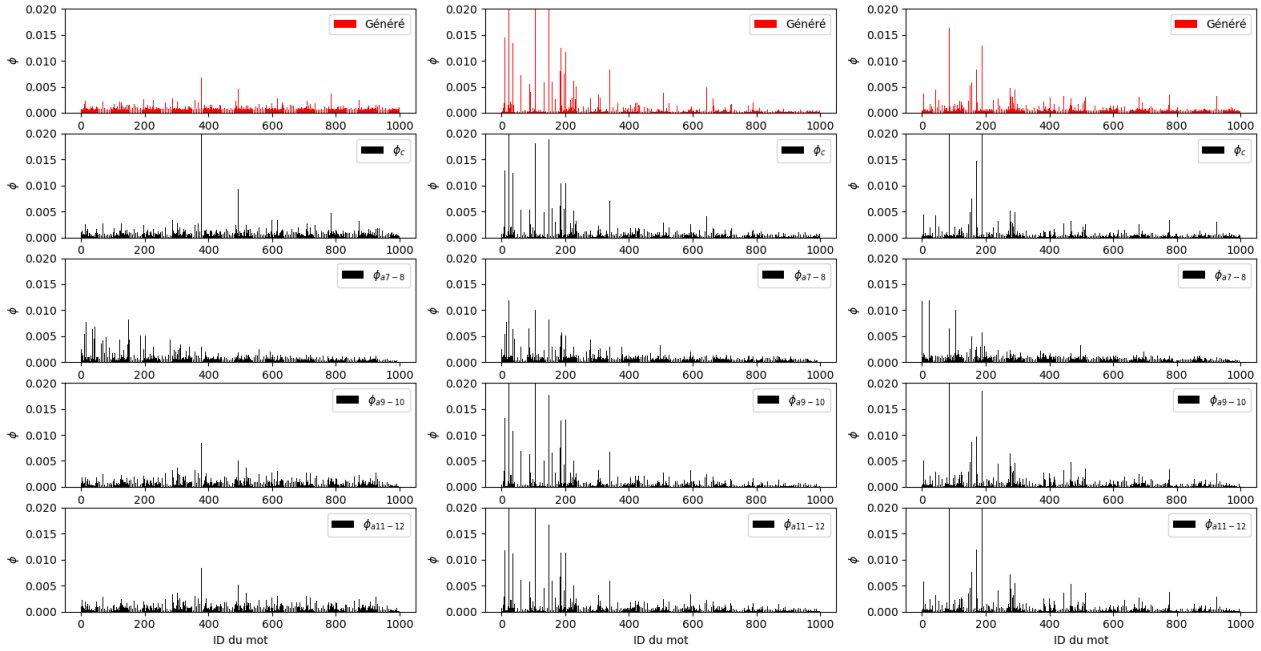


Figure 4.14 Les distributions ϕ générées et inférées de l'expérience 4 pour l'auteur 1 pour les sujets 1 (gauche), 2 (milieu) et 3 (droite)

En premier lieu, il est intéressant de constater l'association entre les niveaux d'expertise et le ϕ généré associé. En ordre croissant du niveau d'expertise, nous avons que la deuxième colonne de graphiques illustre un niveau d'expertise moins élevé que la troisième colonne, cette dernière qui illustre un niveau d'expertise moins élevé que la première colonne. Par conséquent, on observe aisément que les graphiques de la deuxième colonne ne priorisent fortement que certains mots alors que la première colonne démontre une utilisation plus uniforme de ces mots. En second lieu, pour ce qui est des distributions inférées, la quatrième

et la cinquième ligne de graphiques donnent une meilleure approximation des distributions générées que les graphiques de la deuxième et troisième ligne. Par exemple, pour le premier sujet, notre algorithme a un impact significatif sur l'uniformisation des probabilités de ϕ_c et il en résulte une meilleure estimation.

On représente ensuite ces mêmes distributions, mais pour l'auteur 2 sélectionné aussi au hasard dans le corpus. Notons que les paramètres γ qui caractérisent le niveau d'expertise de cet auteur sont $\gamma_2 = [1.26, 0.31, 1.48]$.

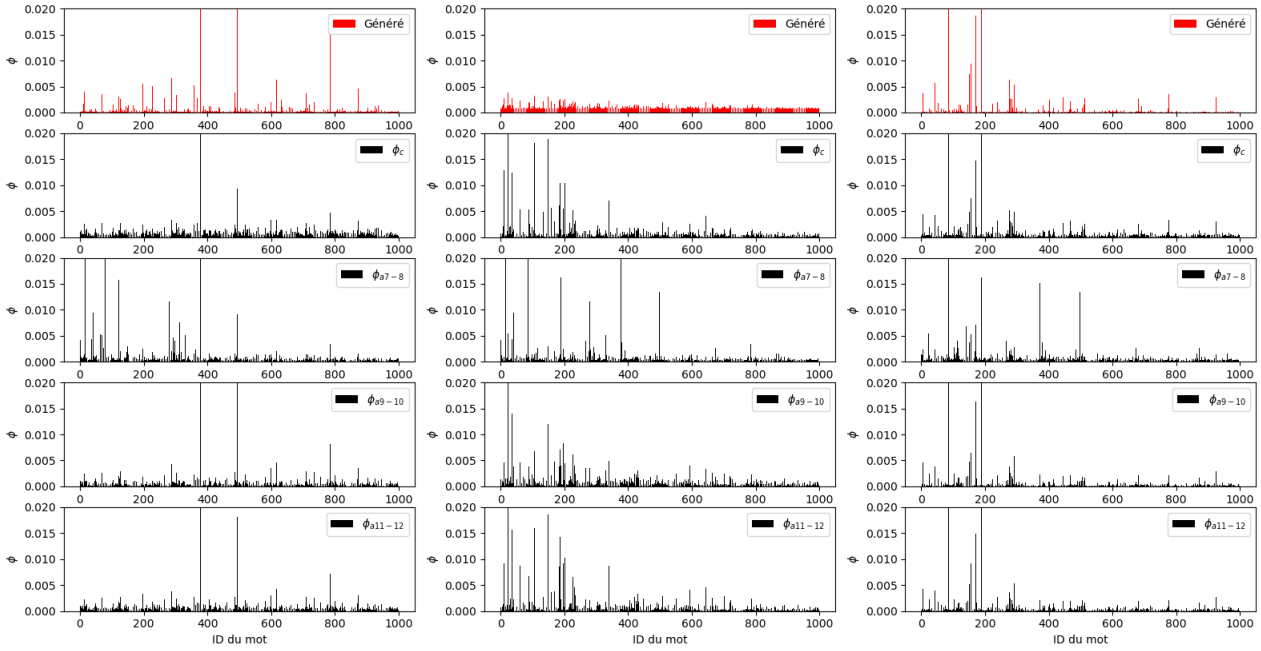


Figure 4.15 Les distributions ϕ générées et inférées de l'expérience 4 pour l'auteur 2 pour les sujets 1 (gauche), 2 (milieu) et 3 (droite)

Avec cette figure, on peut comparer les deux auteurs ainsi que leur niveau d'expertise respectif. Par exemple, si on analyse le graphique du coin supérieur droit, on remarque que les pics de l'auteur 2 sont plus prononcés que ceux de l'auteur 1. Ceci est attendu, puisque l'auteur 2 est moins expert que l'auteur 1 dans le sujet 3, ce qui fait en sorte que le vocabulaire de l'auteur 2 est encore plus restreint aux mots les moins techniques du sujet 3. Si on s'attarde à la deuxième colonne de graphiques, aucune des distributions inférées ne parvient à capturer convenablement le sujet 2. Or, on constate que c'est ϕ_{9-10} qui capture le mieux le

comportement de la distribution générée. Ceci étant dit, on confirme notre hypothèse qu'il est beaucoup plus difficile d'inférer le niveau d'expertise lorsque celui-ci est élevé. Enfin, si on compare les distributions ϕ générées du sujet 2 pour les deux auteurs (l'auteur 1 y est relativement novice alors que l'auteur 2 est expert), ce sont les mêmes mots qui sont les plus populaires pour les deux auteurs. Toutefois, l'impact de ces mots populaires est beaucoup moins important pour l'auteur 2, car la distribution est plus uniforme. Par conséquent, il est vrai qu'il utilisera les mêmes mots moins techniques que l'auteur 1, mais son vocabulaire ne sera pas restreint comme celui de ce dernier et il sera en mesure d'employer les mots plus techniques.

On finit par montrer la différence entre la distribution ϕ_{a9-10} et la distribution ϕ générée de l'auteur 2, une fois celles-ci ordonnées selon le rang des fréquences :

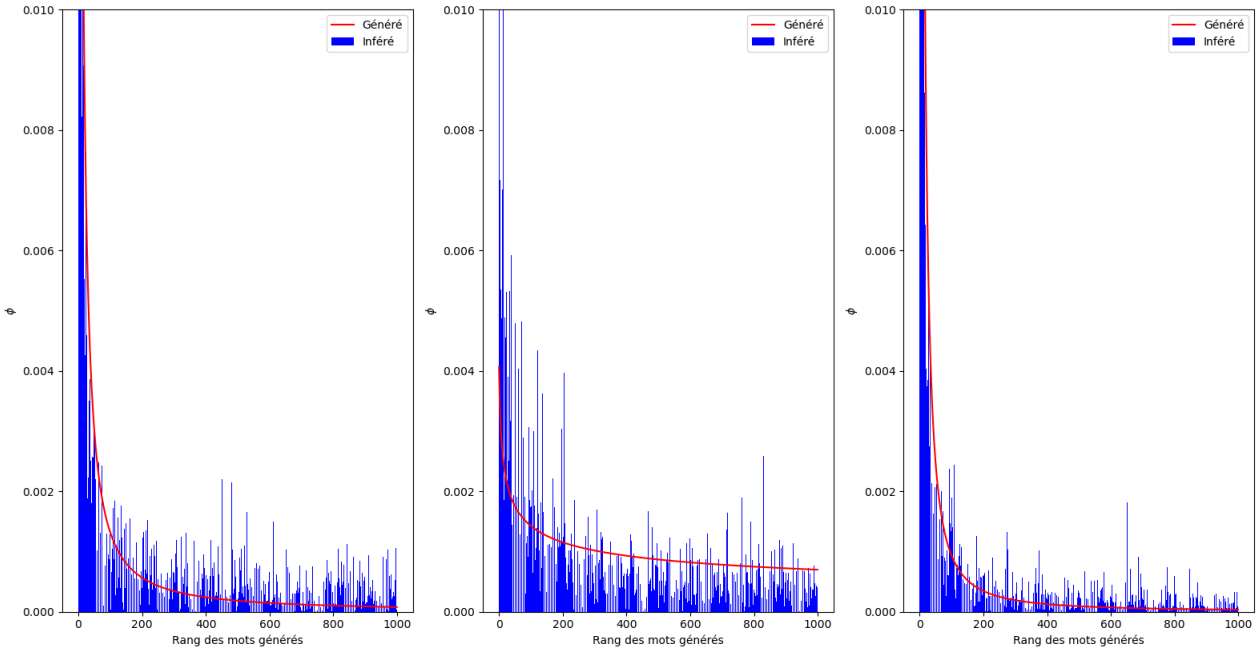


Figure 4.16 La distribution ϕ générée comparée à celle inférée par la méthode 9-10 de l'expérience 4 pour l'auteur 2 ordonnées selon le rang des fréquences pour le sujet 1 (gauche), 2 (milieu) et 3 (droite)

On conclut donc que plus le niveau d'expertise est faible et meilleure est la qualité de l'inférence. Pour le sujet 2, les deux distributions concordent moins, même si l'approximation est passable.

4.5.5 Discussion : expérience avec θ_a forcé et analyse de sensibilité

Tendances générales

À la lumière des résultats obtenus, on peut tirer certaines conclusions. D'abord, pour toutes les expériences, les niveaux d'expertise ont pu être inférés convenablement par les méthodes 9, 10, 11 et 12. Bien que l'expérience 1 nous ait démontré que les performances sont basses lorsqu'on tente d'inférer un niveau d'expertise élevé, on conclut que ce sont ces méthodes qui sont fonctionnelles. De plus, la méthode 9 est la technique qui donne les meilleures performances dans la majorité des cas. Si on compare cette méthode avec la méthode 11, on comprend que l'avantage d'impliquer la pondération par θ_a sur F n'est pas clair et potentiellement que cette implication perturbe les sujets latents communs initialement trouvés par LDA. Par ailleurs, les méthodes 5-6 liées à l'analyse des sous-corpus ne donnent pas de bonnes performances. On conclut donc qu'il est important de considérer le corpus dans son ensemble pour inférer un niveau d'expertise relatif à des sujets latents qui sont communs à tous les auteurs.

En ce qui concerne la différence entre les MCNL et l'EMV, comme il avait été démontré dans les tests préliminaires, les MCNL semblent avoir un avantage systématique pour ce type de problème. Cependant, l'EMV est meilleure lorsque la performance est mauvaise tel qu'observé dans certains cas de la méthode 2 ou la méthode 8.

Expérience avec θ_a forcé

Les résultats semblent donc encourageants pour l'algorithme développé. Toutefois, nous avons remarqué que la métrique KL_{θ_a} était toujours très faible par rapport à KL_{θ_c} et il a été noté que ceci s'expliquait par le fait que chaque auteur était associé à un grand nombre de documents afin de pouvoir établir la comparaison avec les travaux menés au chapitre 3 (la longueur de chaque sous-corpus du chapitre 4 équivaut à la longueur du corpus global du chapitre 3). Alors, étant donné que la distribution de Dirichlet employée pour la génération de θ est symétrique, on se retrouve avec des θ_a qui se rapproche de distributions uniformes. Rappelons que θ_a est la distribution de sujets inférée par auteur alors que θ_c est la distribution de sujets par document. Alors, c'est ce qui explique pourquoi les performances des méthodes 9-10 sont très proches de celles des méthodes 11-12 (si $\theta_a = [0, 33, 0.33, 0.33]$, ces deux méthodes sont identiques, car la pondération par θ_a ne change pas ϕ_c après sa normalisation unitaire). Il est

important de noter que ce n'est pas réellement un problème, puisque dans la littérature de LDA, il est souvent dit, au sein de corpus réalistes, qu'il est rare d'observer un grand écart entre θ et la distribution uniforme lorsqu'on considère un important nombre de documents. Or, puisqu'on introduit la notion d'auteurs dans ce mémoire et que les méthodes 11-12 ne se distinguent des méthodes 9-10 que par la pondération supplémentaire de F par θ_a, il serait pertinent de considérer une expérience où on force les θ_a de manière à augmenter leur variance et c'est cette analyse que l'on performe dans cette sous-section. L'objectif de cette analyse est de s'assurer que les performances d'une inférence avec θ_a forcé ne se détériorent pas de façon importante. On génère les θ_a par une distribution de Dirichlet avec $\alpha = 0.7$. Pour la plage de γ sélectionnée, on considère un cas analogue à l'expérience 4, soit un γ généré aléatoire sur la plage $[0.25, 1.6]$

Afin d'évaluer l'impact d'un θ_a à variance haute et forcée et de comparer la performance d'une inférence ainsi perturbée à la performance obtenue à l'expérience 4, on commence par montrer les mêmes métriques d'inférence que l'on a calculées pour cette expérience.

Tableau 4.13 Synthèse des performances de l'inférence de γ pour l'expérience θ_a forcé

Méthodes	Dr	RMSE
M_1 : (Aléatoire-MCNL)	0.86	0.64
M_2 : (Aléatoire-EMV)	0.87	0.56
M_3 : (Généré-MCNL)	0.0028	0.024
M_4 : (Généré-EMV)	0.0046	0.039
M_5 : (Sous-corpus-MCNL)	0.53	0.34
M_6 : (Sous-corpus-EMV)	0.55	0.33
M_7 : (**A**-MCNL)	0.66	0.31
M_8 : (**A**-EMV)	0.63	0.30
M_9 : (**B**-MCNL)	0.47	0.27
M_{10} : (**B**-EMV)	0.48	0.27
M_{11} : (**C**-MCNL)	0.57	0.28
M_{12} : (**C**-EMV)	0.59	0.29

D'une part, les performances sont significativement plus faibles que pour l'expérience 4. Parce que les θ_a sont forcés, certains auteurs ne disposent pas d'un nombre de données suffisant pour inférer leur niveau d'expertise sur les sujets dont le θ_a est faible (certains auteurs ont un θ_a inférieur à 0.05 pour un sujet donné). Or, la meilleure méthode d'inférence est encore la méthode 9. Malgré que la performance de cette méthode soit inférieure à celle qu'on obtenait à l'expérience 4, on la juge passable, car elle démontre une tendance cohérente en termes de distance de corrélation et une RMSE assez faible.

On illustre la qualité de l'inférence avec un graphique de la comparaison entre les γ inférés et le *ground truth*. Le code de couleur est le même que celui utilisé précédemment.

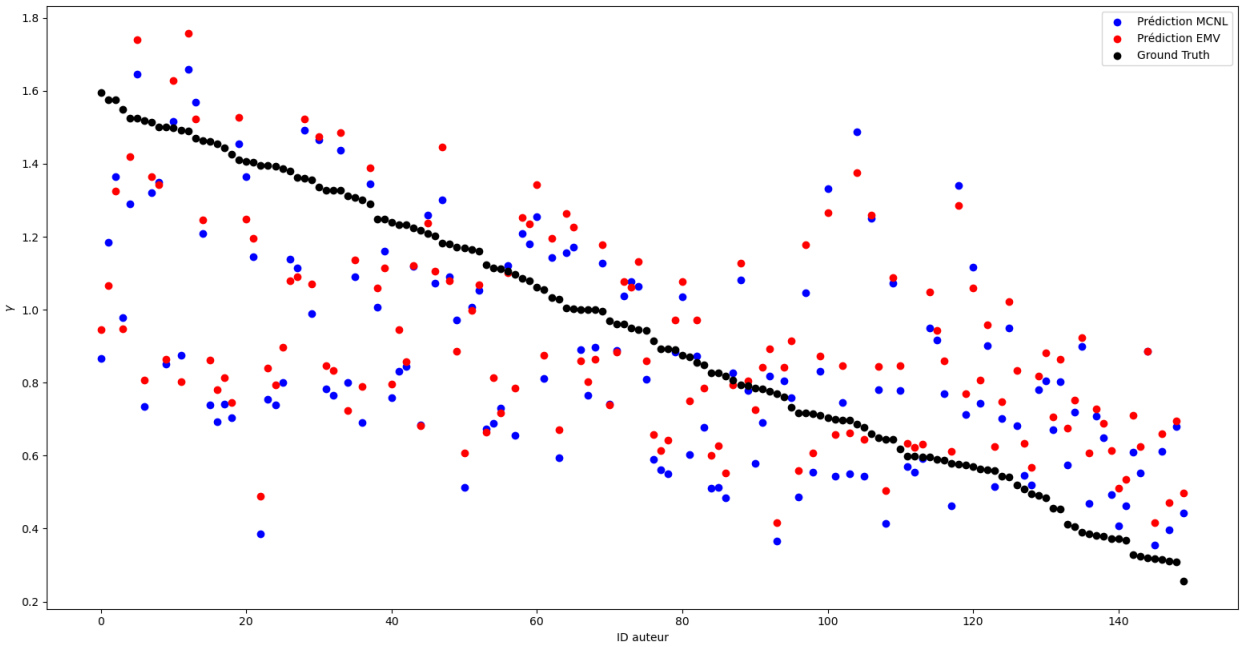


Figure 4.17 Comparaison entre les γ inférés avec les méthodes 9-10 et le *ground truth* pour l'expérience θ_a forcé

Sur ce graphique, on voit tout de suite que l'inférence des γ est moins bonnes que pour l'expérience 4. Cependant, les performances sont cette fois plus mauvaises pour les hauts γ, ce qui est étonnant. On conclut somme toute que l'inférence est acceptable même si un travail supplémentaire pourrait être fait pour ajuster l'algorithme à un cas où la variance des θ_a en fonction des auteurs est augmentée.

On poursuit par la présentation des divergences KL des distributions classiques :

Tableau 4.14 Synthèse des divergences KL pour l'expérience θ_a forcé

Méthodes	θ	ϕ
Commun (c)	0.34	0.38
Sous-corpus (sc)	0.54	0.77
Par auteur (a_{7-8})	0.25	0.72
Par auteur (a_{9-10})	0.25	0.50
Par auteur (a_{11-12})	0.25	0.55

On constate d'abord que les KL_{θ_a} sont plus élevées et que celles-ci se rapprochent de KL_{θ_c} et c'est ce que l'on cherche avec cette expérience. Si on compare ces distributions avec celles obtenues pour l'expérience 4, la moins bonne similarité est dans la même proportion pour toutes les métriques. En considérant qu'on obtenait un KL_{ϕ} aux environs de 0.30 pour une génération LDA classique avec $\alpha = 0.7$, on conclut qu'il est normal d'avoir une similarité plus faible si on fragmente ces distributions ϕ en fonction du niveau d'expertise. Aussi, une divergence KL de 0.5 pour les distributions inférées par les méthodes 9-10 est plus qu'acceptable.

On peut montrer les mêmes graphiques des distributions ϕ du tableau (sauf ϕ_{sc}) pour un auteur en particulier. Montrons d'abord les distributions inférées pour l'auteur 1 sélectionné au hasard dans le corpus. Notons que les paramètres γ qui caractérisent le niveau d'expertise de cet auteur sont $\gamma_1 = [0.71, 0.95, 0.96]$.

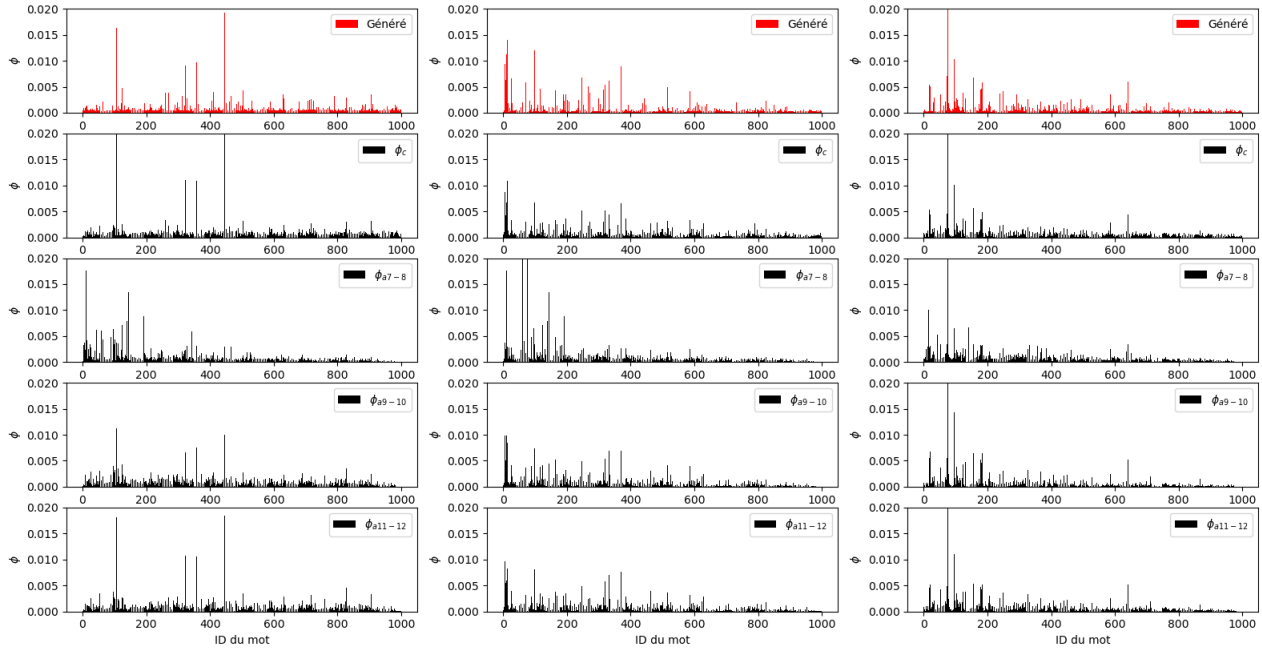


Figure 4.18 Les distributions ϕ générées et inférées de l'expérience θ_a forcé de l'auteur 1 pour les sujets 1 (gauche), 2 (milieu) et 3 (droite)

Pour cet auteur, qui possède des niveaux d'expertise moyens (pas expert ni novice), l'inférence des distributions générées est bonne surtout pour les sujets 2 et 3. En effet, on peut aussi déduire ce comportement de la figure 4.17 où une bonne inférence semble survenir pour des valeurs de γ aux environs de 1.

On représente ensuite ces mêmes distributions, mais pour l'auteur 2 sélectionné aussi au hasard dans le corpus. Notons que les paramètres γ qui caractérisent le niveau d'expertise de cet auteur sont $\gamma_2 = [1.00, 1.16, 0.31]$.

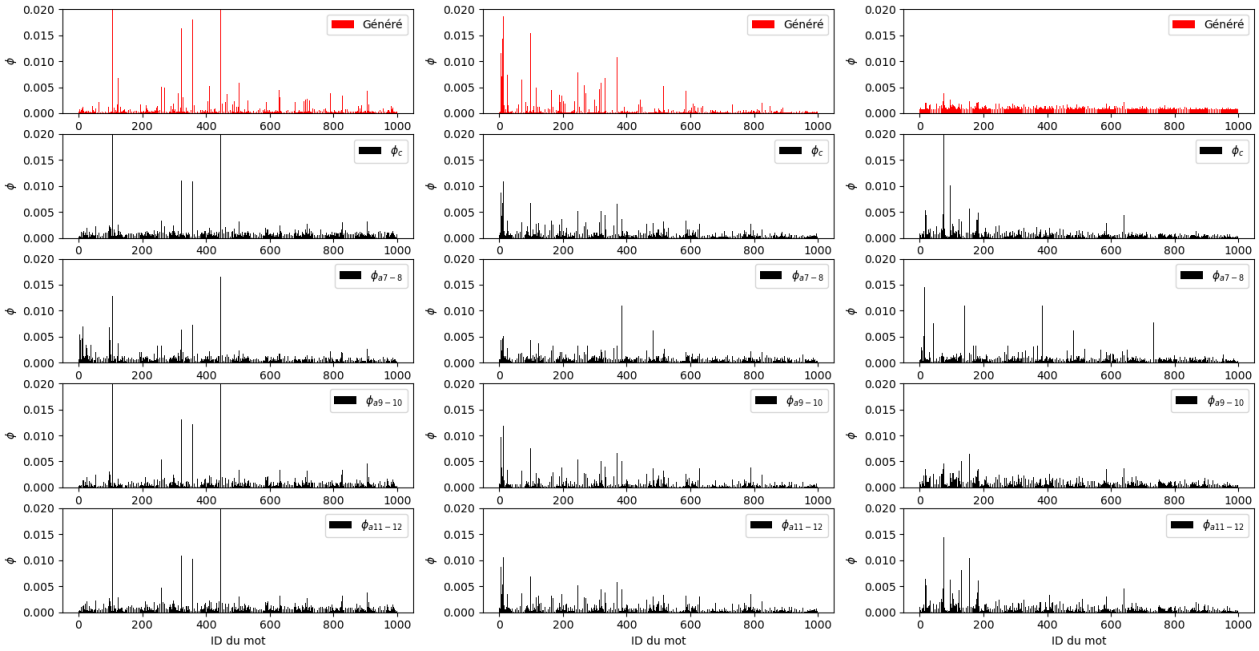


Figure 4.19 Les distributions ϕ générées et inférées de l'expérience θ_a forcé pour l'auteur 2 pour les sujets 1 (gauche), 2 (milieu) et 3 (droite)

Avec cette figure, l'inférence du premier sujet semble adéquate. Cependant, des problèmes commencent à survenir pour les sujets 2 et 3. En ce qui concerne le sujet 2, il n'y a pas assez d'emphase sur les mots populaires ; les distributions inférées uniformisent trop leur prédiction. Pour ce qui est du sujet 3, c'est tout le contraire, car les inférences ont de la difficulté à se distancer de la distribution ϕ_c et mettent par le fait même trop d'emphase sur les mots populaires. Or, l'inférence des méthodes 9-10 illustrée par la distribution ϕ_{a9-10} est la seule qui parvient à uniformiser adéquatement la prédiction. Notons ici que ce sont des exemples pris au hasard et qu'il arrive des cas où une distribution générée par un γ faible n'est pas bien inférée par les méthodes 9-10. On essaie tout de même de vous montrer des exemples représentatifs des résultats généraux observés.

On finit par montrer la différence entre la distribution ϕ_{a9-10} et la distribution ϕ générée de l'auteur 2, une fois celles-ci ordonnées selon le rang des fréquences :

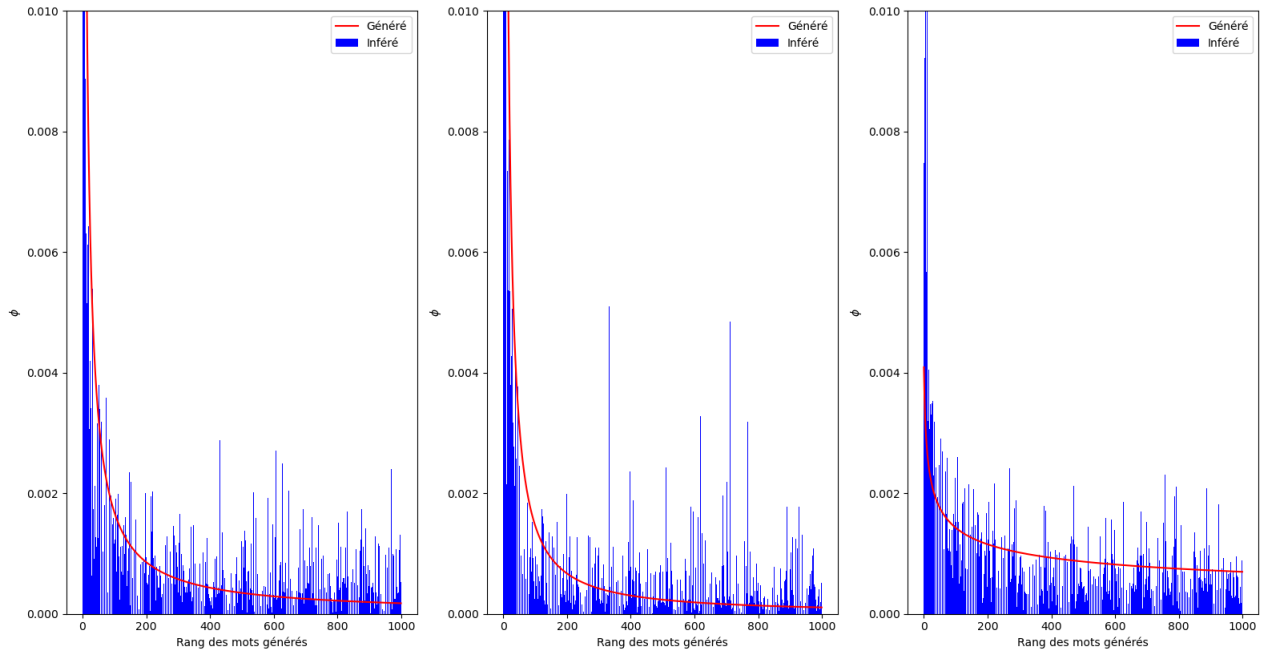


Figure 4.20 La distribution ϕ générée comparée à celle inférée par la méthode 9-10 de l'expérience θ_a forcé pour l'auteur 2 ordonnées selon le rang des fréquences pour le sujet 1 (gauche), 2 (milieu) et 3 (droite)

L'inférence la moins bonne est clairement celle du sujet 2 et ceci corrobore avec les résultats de la figure 4.17. Toutefois, on considère que les inférences sont en général correctes quoique moins bonnes que celles de l'expérience 4.

Analyse de sensibilité

Pour terminer ce chapitre, nous avons décidé de mener une analyse de sensibilité dans le but de déterminer l'effet des différents hyperparamètres sur la performance de l'inférence du paramètre d'expertise. L'analyse de sensibilité consiste à faire varier séquentiellement un hyperparamètre à la baisse et à la hausse, effectuer le procédé génération-inférence avec la méthode 9 et de noter les deltas des performances de distance de corrélation et de RMSE entre les γ générés et inférés, soit les mêmes performances que l'on a recueillies à la sous-section précédente.

L'expérience de référence sera l'expérience 3, puisqu'elle avait de bonnes performances stables. De cette manière, on pourra évaluer plus directement l'impact des hyperparamètres. Les performances de référence seront donc une $D_r = 0.10$ et une RMSE $= 0.019$. Voici les hyperparamètres qui seront sujets à l'étude :

- N_{D_a} : Nombre de documents par auteur, on fera varier ce paramètre de $\pm 50\%$.
- N_a : Nombre d'auteurs, on fera varier ce paramètre de $\pm 60\%$.
- N_{WD} : Nombre de mots par document, on fera varier ce paramètre de $\pm 50\%$.
- N_V : Nombre de mots dans le vocabulaire, on fera varier ce paramètre de $\pm 50\%$.
- N_K : Nombre de sujets latents spécifiés, on fera varier ce paramètre de -33%, car on ne peut avoir moins de 2 sujets, et de $+100\%$ puisqu'on veut reproduire le cas à 6 sujets du chapitre 3.
- α : L'hyperparamètre de Dirichlet utilisé pour la génération de θ, on fera varier ce paramètre de $\pm 60\%$.

On présente les résultats de l'analyse de sensibilité dans le tableau suivant. Lorsqu'on observe un delta de performance négatif, ceci signifie que la performance obtenue pour une analyse donnée est inférieure à la performance de référence, c'est-à-dire une D_r ou une RMSE plus élevée.

Tableau 4.15 Résultats de l'analyse de sensibilité

Paramètres	$\delta Dr(\%)$	δRMSE(%)
Référence	+0	+0
$N_{D_a}(-50\%)$	-130	-53
$N_{D_a}(+50\%)$	-10	-5
$N_a(-60\%)$	-10	-15
$N_a(+60\%)$	-10	-15
$N_{\text{WD}}(-50\%)$	-40	-16
$N_{\text{WD}}(+50\%)$	-50	-32
$N_V(-50\%)$	-140	-68
$N_V(+50\%)$	+1	+0
$N_K(-33\%)$	+50	+26
$N_K(+100\%)$	-280	-158
$\alpha(-60\%)$	-60	-42
$\alpha(+60\%)$	+0	-16

À la lumière de l'analyse de sensibilité, on constate que notre instinct avait choisi de bons hyperparamètres pour les expériences de la sous-section précédente. En effet, on obtient plus fréquemment une dégradation des performances si on fait varier les hyperparamètres que ce

soit à la baisse ou à la hausse. Aussi, il est intéressant de remarquer que les combinaisons d'hyperparamètres sont souvent plus importantes que l'effet d'un hyperparamètre unique. Par exemple, un nombre de mots par document trop faible est nuisible aux performances alors qu'un nombre de mots par document trop haut l'est tout autant. Par conséquent, une question de proportion entre les différents hyperparamètres existe afin d'optimiser les performances et il est pertinent de constater que les proportions que l'on avait à titre de référence sont bonnes.

On peut toutefois remarquer des tendances générales uniques pour certains hyperparamètres :

- Il est nuisible d'avoir un nombre de documents par auteur ou une taille de vocabulaire trop bas.
- Plus on a de sujets latents spécifiés et moins bonnes sont les performances, ainsi que l'avait prédit les analyses du chapitre 3.

Somme toute, on juge que l'analyse de sensibilité donne des résultats sensés. Effectivement, même lorsque $\delta D_r = -130\%$ et $\delta\text{RMSE} = -53\%$, comme pour l'analyse de $N_{D_a}(-50\%)$, on se rappelle que dans les faits, on a $D_r = 0.23$ et $\text{RMSE} = 0.029$, ce qui est tout de même très bas. Toutefois, on devra être sur nos gardes si on veut appliquer la méthode sur une échelle de proportion entre les hyperparamètres très différente de ce qui a été proposé pour la référence.

4.5.6 Conclusion du chapitre

Ceci marque la fin du chapitre 4 où certaines hypothèses ont été posées et un grand nombre de travaux ont été accomplis et présentés afin de répondre à la question de recherche suivante : comment peut-on faire interagir les lois statistiques du langage dans l'infrastructure LDA afin de déterminer l'expertise des auteurs propre à un sujet donné ? Rappelons que, bien que ce modèle est utilisé pour évaluer l'expertise d'auteurs d'articles scientifiques, il s'agit d'une première ébauche d'une technique permettant de segmenter un corpus textuel selon ses sujets en plus d'associer une profondeur technique à ces derniers.

D'abord, une description générale du modèle d'expertise a été abordée. Ensuite, l'analogie entre l'expertise et la distribution de Mandelbrot a été élaborée : plus le paramètre de Mandelbrot est faible et plus le niveau d'expertise est élevé et on a situé cette hypothèse dans un contexte où l'ordre de technicalité des mots est défini.

Nous avons ensuite présenté des performances d'inférence des distributions classiques θ et ϕ de LDA si on génère les ϕ avec une loi de Mandelbrot plutôt que Dirichlet. Il a été conclu

qu'il est aussi possible d'utiliser notre cadre de validation pour inférer le paramètre de Mandelbrot, car la divergence KL pour ϕ était meilleure que celle obtenue au chapitre 3 pour une génération Dirichlet avec des hyperparamètres comparables. De plus, deux méthodes d'inférence du paramètre c de Mandelbrot ont été explorées, soit les MCNL et l'EMV. On a vu que les MCNL semblaient plus appropriés pour prédire c et ce comportement a été confirmé dans les expériences subséquentes.

Après avoir présenté la nomenclature propre au nouveau modèle d'expertise, la logique de ce modèle a été traitée. Nous avons détaillé 4 expériences illustrant des situations d'expertise différentes, 12 méthodes pour trouver les fréquences de mots par auteur et par sujet ainsi que 8 métriques ayant un rapport avec les divergences KL des distributions classiques de LDA. La méthode 9 a été identifiée comme étant la plus appropriée pour notre problème. En effet, pour l'expérience 4, soit l'expérience qui s'apparente le plus à un cas réel, on a trouvé que la D_r et la RMSE pour l'inférence de γ était respectivement de 0.10 et 0.12 pour cette méthode, ce qui est très bon. Nous avons aussi conclu que la divergence KL calculée entre les ϕ générées et inférées ($\mathrm{KL}_{\phi 9-10} = 0.25$) était comparable à celle obtenue au chapitre 3, ce qui est encourageant. Il a aussi été conclu qu'il n'était pas nécessaire ni même souhaitable d'appliquer LDA sur les sous-corpus propres à un auteur donné dans le but d'inférer son niveau d'expertise de façon plus précise.

Toutefois, en raison du volume de documents générés, nous avons constaté que les θ_a propres à chaque auteur se rapprochaient d'une distribution uniforme. Même si un cas où les θ_a diffèrent grandement n'est pas très réaliste, il s'agit d'une limitation de notre approche et nous avons décidé de proposer une expérience où on fixe les θ_a pour chaque auteur. Les performances obtenues ont été moins bonnes que celles de l'expérience 4 mais tout de même acceptables : une $D_r = 0.47$, une $\mathrm{RMSE} = 0.27$ et une $\mathrm{KL}_{\phi 9-10} = 0.5$.

Le chapitre 4 a été conclu par une analyse de sensibilité où on a mesuré l'impact de la variation de certains hyperparamètres. Il a été trouvé que cet impact n'est que très rarement linéaire et que la combinaison entre les hyperparamètres est plus importante. Or, il a été trouvé qu'un nombre trop bas de documents par auteur et de mots dans le vocabulaire était nuisible aux performances. On a aussi déterminé que le nombre de sujets latents spécifiés avait un impact significatif sur les résultats.

CHAPITRE 5 CONCLUSION

Ce mémoire présente une méthode pour inférer l'expertise des auteurs en se basant sur un corpus de textes. Ces travaux s'inscrivent dans un contexte plus large de modélisation textuelle où il est nécessaire d'inférer un niveau d'expertise dans des sujets inférés par LDA. La méthode repose sur une extension de LDA qui donne de bons résultats même si certaines anomalies dans les performances et quelques limitations de la méthode existent. Or, il s'agit des balbutiements en matière d'inférence de l'expertise et ces travaux devront être raffinés dans le futur pour être utilisables dans un contexte pratique.

5.1 Synthèse des travaux

Dans un premier temps, c'est au chapitre 3 que nous avons mis au point un cadre de validation afin de vérifier les performances d'un modèle statistique génératif et plus spécifiquement de LDA. Nous avons utilisé ce cadre de validation pour mener deux analyses en lien avec le modèle LDA classique. D'une part, l'impact de la performance du CGS et de l'inférence variationnelle en fonction des hyperparamètres utilisés pour la génération et en fonction du nombre de sujets latents spéficiés a été étudié. Il a été conclu que le CGS donne généralement de meilleures performances que l'inférence variationnelle et que les α et β doivent être inférieurs à 1 pour assurer les bonnes performances de l'inférence. D'autre part, la similitude des données générées avec les lois statistiques du langage a été étudiée. Il a été trouvé que β devait se situer entre 0.3 et 1 pour produire un corpus généré réaliste, c'est-à-dire en accord avec les fréquences statistiques de Zipf-Mandelbrot. On a terminé ce chapitre en montrant les performances d'inférence d'un modèle dont les hyperparamètres de génération étaient $\alpha = 0.7$ et $\beta = 0.5$, puisque c'est cette configuration qui constituait le meilleur compromis entre une inférence adéquate et une bonne similitude avec les lois du langage. Une méthode d'alignement des sujets latents a aussi été présentée et éprouvée. Nous avons donc répondu à la première question de recherche, soit : quelles sont les conditions opérationnelles du modèle LDA classique et dans quelle mesure l'hypothèse de génération des données de ce modèle est-elle conforme aux lois statistiques du langage ?

Dans un second temps, nous avons élaboré notre modèle d'inférence de l'expertise des auteurs au chapitre 4. Nous avons commencé par l'exploration de la notion de l'expertise et de quelle façon on pouvait la quantifier : il a été statué que plus le paramètre de Mandelbrot était faible et plus le niveau d'expertise était élevé dans un contexte où l'ordre de technicalité des

mots est connu. Il a aussi été démontré qu'il est possible d'utiliser notre cadre de validation en se servant d'une loi de Mandelbrot pour la génération de ϕ au lieu d'une distribution de Dirichlet. Par la suite, le nouveau modèle d'expertise a été abordé en détaillant différentes méthodes pour inférer les fréquences de mots par auteur et par sujet. Ces méthodes ont été testées avec plusieurs expériences caractérisées par des situations d'expertise différentes. En ce qui concerne l'expérience où les données sont générées à partir d'une expertise aléatoire des auteurs (expérience 4), il a été déterminé que la méthode de la pondération de la matrice de fréquence des auteurs par les distributions ϕ communes à tous les auteurs était la meilleure pour inférer le paramètre d'expertise. D'ailleurs, on a montré que la divergence KL entre les ϕ générées et inférées de cette méthode pour cette expérience est comparable à la même divergence KL qu'on avait calculée au chapitre 3 pour un cas où l'expertise n'était pas prise en compte. Il a aussi été conclu qu'il n'était pas souhaitable d'appliquer LDA sur les sous-corpus propres à un auteur donné en vue d'inférer son expertise sur les sujets latents communs à tout le corpus. Pour finir, les limitations de la méthode ont été explorées. D'une part, en proposant une expérience où on force une variance élevée aux distributions de sujets par auteur, on a pu constater des performances moins bonnes quoiqu'acceptables. De plus, nous avons procédé à une analyse de sensibilité dans laquelle on a découvert, d'une part, que l'impact des hyperparamètres sur les performances était non linéaire. D'autre part, il a aussi été trouvé que le nombre de documents par auteur, le nombre de mots dans le vocabulaire ainsi que le nombre de sujets latents spécifié pouvaient avoir un impact significatif direct sur les résultats. Rappelons que, bien que le modèle d'expertise développé se concentre sur l'évaluation de l'expertise des auteurs d'articles scientifiques, il s'agit d'une première ébauche d'une technique permettant de segmenter un corpus textuel selon ses sujets en plus d'associer une profondeur technique à ces derniers. Nous avons donc répondu à la seconde question de recherche, soit : comment peut-on faire interagir les lois statistiques du langage dans l'infrastructure LDA afin de déterminer l'expertise des auteurs propre à un sujet donné.

5.2 Limitations de la solution proposée

Bien que la solution développée donne des résultats cohérents, elle constitue les premiers pas dans ce champ d'étude et comporte donc certaines limitations. Identifions les trois principales limitations de l'approche.

D'abord, à la lumière de l'expérience avec θ_a forcé et l'analyse de sensibilité, il est clair que la méthode n'est pas applicable dans tous les contextes. En effet, si on veut bénéficier d'une performance optimale de la méthode, la combinaison d'hyperparamètres doit être proche de

celle qui a été utilisée comme référence. Toutefois, il est important de noter que la détérioration en termes de performance a été calculée dans l'analyse de sensibilité sous forme relative et qu'en forme absolue, nous avons toujours une bonne idée du paramètre d'expertise, et ce, pour une large majorité des combinaisons d'hyperparamètres. Également, la distribution θ moyenne propre à chaque auteur ne doit pas se distancer trop de la distribution uniforme sous peine d'avoir une réduction significative de la qualité de l'inférence même si cette dernière reste acceptable.

Ensuite, le fait que la méthode repose sur la connaissance d'un ordre de technicalité dans les mots lors de la génération constitue l'une de ses faiblesses. Dans notre cas, puisque la génération avait été produite selon cet ordre connu, on pouvait simplement ordonner les distributions inférées et on avait systématiquement que les mots les plus fréquents étaient des mots de technicalité faible. Par exemple, il était impossible d'avoir un auteur caractérisé par une distribution fortement déphasée de mots qui priorise seulement les mots hautement techniques : si l'auteur était novice, la conception de la génération faisait en sorte que seulement les mots novices étaient employés tandis que si l'auteur était expert, tous les mots étaient utilisés (techniques comme novices). C'est cette raison qui explique pourquoi notre méthode est liée à l'expertise et non à la richesse du vocabulaire. Or, dans un cas réel où certains auteurs experts pourraient n'utiliser que des mots techniques sans employer de mots simples, il faudrait que cet ordre de technicalité soit connu et que les ϕ inférés ne soient pas ordonnés avant l'application des MCNL pour arriver à un résultat similaire au nôtre.

Finalement, le cadre de validation est autant synonyme de force que de faiblesse de la méthode proposée. Il s'agit principalement d'une force, puisque ce processus intéressant nous permet de comprendre en profondeur comment LDA se comporte et de nous servir des connaissances acquises lorsque viendra le temps d'avoir un regard critique sur les informations qui sortent de ce modèle. Par contre, il s'agit également d'une faiblesse dans notre cas, car beaucoup d'hypothèses émises n'ont pas pu être vérifiées sur des données réelles. Si on présume que toutes les hypothèses que l'on a utilisées pour la génération sont correctes, on vit dans l'utopie. La vérité ne devrait pas se situer loin de ce qui a été simulé, mais le fait de ne pas avoir testé notre méthode sur des données réelles constitue assurément l'une de ses faiblesses.

À la lumière des limitations proposées, on peut réfléchir sur la portée du présent travail et aux conditions nécessaires à son utilisation. En premier lieu, on a démontré qu'il était possible d'estimer la paramétrisation d'une loi de Mandelbrot sur les fréquences de mots propres aux sujets inférés par LDA. Bien qu'il n'ait pas été prouvé directement sur des données réelles

que cette mesure témoigne indubitablement du niveau d'expertise, on se repose sur des études qui ont démontré que le niveau d'expertise était dépendant de cette paramétrisation (Laufer et Nation (1995), Dakhel *et al.* (2021)). De plus, comme on se repose également sur le fait que l'ordre de technicalité des mots est connu (condition non nécessaire, puisque la corrélation entre la facilité d'un mot et sa fréquence est positive), ce lien est d'autant plus accepté. Par ailleurs, nous sommes d'avis que le cadre de validation élaboré fournit une mesure numérique qui est plus objective que l'appréciation subjective qui aurait été obtenue par le sondage de différents experts. Avec notre cadre, on distingue ainsi précisément les cas où l'algorithme réussit bien sa tâche de ceux où ce dernier échoue. Nous n'aurions pas pu procéder à cette analyse rigoureuse avec des méthodes de validation plus classiques. Toutefois, la prochaine étape logique serait de valider la technique avec des vraies données.

5.3 Améliorations futures

L'identification des limitations de la méthode mène à un questionnement quant aux améliorations futures qui pourraient être apportées à celle-ci.

D'abord, dans le but de résoudre le problème de la dépendance à un ordre de technicalité des mots connu d'avance, on pourrait ajuster la méthode d'inférence des paramètres. En effet, au lieu de procéder à l'inférence de l'expertise après avoir appliqué LDA, on croit qu'il serait possible de développer une méthode par CGS qui inclut la technicalité des mots ainsi que l'expertise dans l'expression de la probabilité postérieure du problème. Une fois cette probabilité postérieure élaborée, on procéderait de la même manière, soit en échantillonnant sur le corpus et en évaluant les probabilités qu'un mot appartienne non seulement à un sujet, mais aussi à un niveau d'expertise. Cette possibilité complexifie grandement les mathématiques du problème, mais on pense qu'elle nous permettrait d'avoir un processus d'inférence *end-to-end* de l'expertise plus robuste et représentif. Notons que nous avons tenté de développer cette approche dans le mémoire, mais que les résultats obtenus étaient médiocres à cause des difficultés techniques rencontrées.

Ensuite, dans le même ordre d'idées de la limitation concernant les données réelles, une amélioration évidente consisterait à tester le cadre de validation en plus de notre méthode d'inférence de l'expertise sur des données réelles et de constater s'il est vrai que les performances sont plus ou moins bonnes dépendamment des contextes traités dans le mémoire. On pourrait monter une base de données de manuels scolaires ou d'articles scientifiques d'un niveau d'expertise incrémental afin de posséder une étiquette de cette dernière sur chaque

document.

Pour terminer, une voie intéressante à explorer serait l'inclusion de données supplémentaires qui caractérisent les auteurs ajoutés à leurs documents écrits. Ces données pourraient agir comme les métadonnées de produits dans les recommandations d'un commerce électronique. On pourrait, par exemple, ajouter un graphe de citations entre les auteurs ou une pondération par leur h-index pour déduire une approximation de l'ordre de technicalité des mots qu'ils utilisent. Une fois cet ordre estimé, on pourrait avoir recours à la méthode développée dans le mémoire pour inférer le niveau d'expertise de ces auteurs.

Bibliographie

Balog, K., Azzopardi, L. et De Rijke, M. (2006). Formal models for expert finding in enterprise corpora. *In Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 43–50.

Blei, D. M., Ng, A. Y. et Jordan, M. I. (2003). Latent dirichlet allocation. *Journal of Machine Learning research*, 3(null):993–1022.

Calude, A. S. et Pagel, M. (2011). How do we use language ? shared patterns in the frequency of word use across 17 world languages. *Philosophical Transactions of the Royal Society B : Biological Sciences*, 366(1567):1101–1107.

Canini, K., Shi, L. et Griffiths, T. (2009). Online inference of topics with latent dirichlet allocation. *In Artificial Intelligence and Statistics*, pages 65–72.

Chang, J., Boyd-Graber, J., Wang, C., Gerrish, S. et Blei, D. M. (2009). Reading tea leaves : How humans interpret topic models. *In Neural Information Processing Systems*.

Charlin, L. et Zemel, R. (2013). The toronto paper matching system : an automated paper-reviewer assignment system.

Cortes, V. (2004). Lexical bundles in published and student disciplinary writing : Examples from history and biology. *English for specific purposes*, 23(4):397–423.

Dakhel, A. M., Desmarais, M. C. et Khomh, F. (2021). Assessing developer expertise from the statistical distribution of programming syntax patterns.

Fang, H. et Zhai, C. (2007). Probabilistic models for expert finding. *In* Amati, G., Carpineto, C. et Romano, G., éditeurs : *Advances in Information Retrieval*, pages 418–430, Berlin, Heidelberg. Springer Berlin Heidelberg.

Fang, Y., Si, L. et Mathur, A. P. (2010). Discriminative models of integrating document evidence and document-candidate associations for expert search. *In Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '10, page 683–690, New York, NY, USA. Association for Computing Machinery.

Fletcher, R. (2013). *Practical methods of optimization*. John Wiley & Sons.

George, E. I. et McCulloch, R. E. (1993). Variable selection via gibbs sampling. *Journal of the American Statistical Association*, 88(423):881–889.

Gower, J. C. (1975). Generalized procrustes analysis. *Psychometrika*, 40(1):33–51.

Hoffman, M., Bach, F. R. et Blei, D. M. (2010). Online learning for latent dirichlet allocation. *In advances in neural information processing systems*, pages 856–864.

Kullback, S. et Leibler, R. A. (1951). On information and sufficiency. *Annals of Mathematical Statistics*, 22(1):79–86.

Laufer, B. et Nation, P. (1995). Vocabulary size and use : Lexical richness in l2 written production. *Applied linguistics*, 16(3):307–322.

Liu, Z., Li, M., Liu, Y. et Ponraj, M. (2011). Performance evaluation of latent dirichlet allocation in text mining. *In 2011 Eighth International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)*, volume 4, pages 2695–2698. IEEE.

Mandelbrot, B. (1953). An informational theory of the statistical structure of language. *Communication theory*, 84:486–502.

Mimno, D. et McCallum, A. (2007). Expertise modeling for matching papers with reviewers. *In Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '07, page 500–509, New York, NY, USA. Association for Computing Machinery.

Mimno, D., Wallach, H., Talley, E., Leenders, M. et McCallum, A. (2011). Optimizing semantic coherence in topic models. *In Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 262–272.

Moré, J. J. (1978). The levenberg-marquardt algorithm : implementation and theory. *In Numerical analysis*, pages 105–116. Springer.

Mukherjee, I. et Blei, D. M. (2009). Relative performance guarantees for approximate inference in latent dirichlet allocation. *In Advances in Neural Information Processing Systems*, pages 1129–1136.

Porteous, I., Newman, D., Ihler, A., Asuncion, A., Smyth, P. et Welling, M. (2008). Fast collapsed gibbs sampling for latent dirichlet allocation. *In Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 569–577.

Rosen-Zvi, M., Griffiths, T., Steyvers, M. et Smyth, P. (2004). The author-topic model for authors and documents. *In Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence*, UAI '04, page 487–494, Arlington, Virginia, USA. AUAI Press.

Serdyukov, P., Rode, H. et Hiemstra, D. (2008). Modeling multi-step relevance propagation for expert finding. *In Proceedings of the 17th ACM Conference on Information and Knowledge Management*, CIKM '08, page 1133–1142, New York, NY, USA. Association for Computing Machinery.

Simon, H. A. (1955). On a class of skew distribution functions. *Biometrika*, 42(3/4):425–440.

Sklar, M. (2014). Fast mle computation for the dirichlet multinomial. *arXiv preprint arXiv :1405.0099*.

Xue, G. et Nation, I. (1984). A university word list. *Language learning and communication*, 3(2):215–229.

Zipf, G. K. (1949). *Human behavior and the principle of least effort*. Addison-Wesley.