Projet M2 IM
-
Introduction aux Topic Models

Ulysse Herbach

5 décembre 2014

Depuis l'avènement de l'informatique, la quantité de données générées chaque jour par les activités humaines a augmenté de manière considérable, à tel point qu'il est maintenant strictement impossible de les traiter manuellement. On cherche donc à effectuer cette tâche de manière *efficace* et *automatique* : naturellement, on se tourne encore vers l'informatique.

Parmi les différents types de données, on trouve notamment une immense quantité de textes provenant d'Internet : on voudrait être capable d'analyser ces textes pour en tirer, sous forme condensée, l'information la plus pertinente possible. C'est dans ce cadre qu'ont été introduits les *Topic Models* : il s'agit de comprendre la structure sous-jacente d'un corpus de textes en dégageant des thèmes, ou *topics*, supposés présents dans le corpus mais non connus à l'avance.

Au départ empiriques, puis basés sur une décomposition matricielle, les Topic Models ont connu une progression majeure lorsqu'on a pu leur donner une base entièrement probabiliste [5]. Ce sont ces modèles probabilistes qui nous intéressent ici : les thèmes sont alors considérés comme des variables aléatoires *latentes* (i.e. non observées) qui permettent de générer un corpus de textes. Leur force réside dans le fait qu'ils peuvent fournir des informations bien plus fines qu'un simple résumé du corpus, par exemple en prenant en compte la polysémie, c'est-à-dire la possibilité pour un mot d'appartenir à plusieurs thèmes.

Nous nous restreignons ici à l'approche historique en considérant des données textuelles, mais il est important de noter que les Topic Models peuvent potentiellement s'appliquer à n'importe quel type de données : ils sont à présent largement utilisés dans des domaines variés comme la génomique [8] ou l'exploration de bases de données constituées d'images [1].

1 Modèles et hypothèses associées

On décrit ici les quatre principaux Topic Models probabilistes, dont la sophistication croissante est en partie liée à leur évolution historique. Chaque modèle est associé à des hypothèses bien précises sur la façon dont les documents sont générés. On peut déjà dégager deux hypothèses communes à tous ces modèles :

- *bag-of-words* : on considère que l'ordre des mots n'a pas d'importance sur la façon de générer un document. En d'autres termes, un document est un "sac de mots" plutôt qu'une liste ordonnée.
- *bag-of-documents* : on fait une hypothèse similaire à l'échelle supérieure, en supposant que les documents n'ont pas d'ordre spécifique dans le corpus.

Ces hypothèses réduisent considérablement la difficulté d'analyse mathématique des modèles. Bien qu'assez fortes a priori, elles laissent une certaine marge de manoeuvre permettant de construire des modèles très sophistiqués [2].

Mathématiquement, ce type d'hypothèse se traduit par la notion d'*échangeabilité* d'un ensemble de variables aléatoires. Dans tout ce qui suit, les variables aléatoires considérées sont réelles ou discrètes, et les familles de variables aléatoires sont définies sur un même espace probabilisé $(\Omega, \mathcal{A}, \mathbb{P})$ et à valeurs dans un même espace mesurable $(E, \mathcal{E})$.

Définition 1. Un ensemble fini $\{X_1, \dots, X_n\}$ de variables aléatoires est dit *échangeable* si sa loi jointe est invariante par permutation, i.e. :

$$\forall \sigma \in S_n, \quad \mathcal{L}(X_{\sigma(1)}, \dots, X_{\sigma(n)}) = \mathcal{L}(X_1, \dots, X_n)$$

où S_n est le groupe des permutations de $\{1, \dots, n\}$. Un ensemble infini de variables aléatoires est dit *infiniment échangeable* si tous ses sous-ensembles finis sont échangeables.

Notons que si $\{X_1, \dots, X_n\}$ est échangeable, alors les variables aléatoires $X_1, \dots, X_n$ ont même loi : ceci est une conséquence immédiate de la définition (pour $i \neq j$, considérer la transposition $\sigma = (i\, j)$ et regarder les lois marginales). Par ailleurs, si les variables $X_1, \dots, X_n$ sont i.i.d, l'ensemble $\{X_1, \dots, X_n\}$ est échangeable puisqu'on a alors :

$$\mathcal{L}(X_{\sigma(1)}, \dots, X_{\sigma(n)}) = \mathcal{L}(X_1)^{\otimes n} = \mathcal{L}(X_1, \dots, X_n).$$

En revanche la réciproque est fausse [1]. En fait, l'infinie échangeabilité caractérise des variables aléatoires *conditionnellement* indépendantes et de même loi. Plus précisément, on a le théorème suivant.

Théorème 1 (De Finetti)**.** *Soit $\{X_i \,|\, i \in \mathbb{N}^*\}$ un ensemble infiniment échangeable. Alors il existe une variable aléatoire Y telle que, sachant Y, les X_i sont i.i.d., c'est-à-dire :*

$$\mathcal{L}(X_1, \dots, X_n|Y) = \mathcal{L}(X_1|Y) \otimes \cdots \otimes \mathcal{L}(X_n|Y)$$

pour tout $n \in \mathbb{N}^*$, avec $\mathcal{L}(X_i|Y) = \mathcal{L}(X_j|Y)$ pour tout $(i, j) \in \mathbb{N}^* \times \mathbb{N}^*$.

On renvoie à [3] pour une preuve de ce théorème. En notant $X = (X_1, \dots X_n)$ et P_Y la loi de Y, on a alors pour tout $A = (A_1, \dots, A_n) \in \mathcal{E}^n$:

$$\mathbb{P}(X \in A) = \int_E \mathbb{P}(X \in A|Y = y)\, dP_Y(y) = \int_E \left(\prod_{i=1}^{n} \mathbb{P}(X_i \in A_i|Y = y) \right) dP_Y(y). \qquad (1)$$

La loi de X s'interprète donc comme un mélange (potentiellement infini selon le support de Y) de lois produits. Ce résultat est parfois appelé *théorème de représentation de De Finetti* : on va voir qu'il justifie la structure sous-jacente de certains modèles.

1.1 Notations

L'idée générale des Topic Models est de décrire la loi d'un corpus aléatoire, i.e. la façon de générer les documents qui le constituent. On commence par fixer un dictionnaire dans lequel seront tirés tous les mots du corpus, et on note $M \in \mathbb{N}^*$ sa taille. Pour générer un document constitué de N mots, on considère des variables aléatoires $W_1, \dots, W_N$, à valeurs dans $\{1, \dots, M\}$, et on assimile le document à la variable aléatoire $D = (W_1, \dots, W_N)$. L'hypothèse *bag-of-words* se traduit alors mathématiquement par le fait que l'ensemble $\{W_1, \dots, W_N\}$ est échangeable.

Pour un N-uplet de mots $(w_1, \dots, w_N) \in \{1, \dots, M\}^N$, la probabilité que D soit constitué des mots $(w_1, \dots, w_N)$ est notée $P_D(w_1, \dots, w_N) = \mathbb{P}(D = (w_1, \dots, w_N))$.

1. Un exemple classique est l'urne de Pólya.

Quand il n'y aura pas de confusion possible, on notera, pour toute variable aléatoire discrète désignée par une lettre majuscule X et pour tout x dans l'image de X :

$$p(x) = P_X(x) = \mathbb{P}(X = x).$$

De même, si Y est une autre v.a. et si y est dans l'image de Y avec $p(y) = \mathbb{P}(Y = y) > 0$, on notera $p(x|y) = \mathbb{P}(X = x|Y = y)$.

1.2 Le modèle *unigram*

C'est le modèle probabiliste le plus simple possible. On suppose que :

- les documents sont générés de manière indépendante et de la même façon ;
- les mots de chaque document sont générés de manière indépendante, à partir d'une même mesure de probabilité discrète.

La distribution associée est donc :

$$P_D(w_1, \ldots, w_N) = \prod_{n=1}^{N} p(w_n)$$

Notons que l'hypothèse d'échangeabilité des mots est clairement vérifiée. Ici, une façon de générer un corpus de m documents est de fixer N et M, puis considérer des variables i.i.d. $D_1, \ldots, D_m$, les documents contenant alors tous le même nombre de mots.

1.3 Le modèle *mixture of unigrams*

On introduit maintenant une variable latente "topic" notée Z, à valeurs dans $\{1, \ldots, K\}$ où K est le nombre de topics [7]. On fait les hypothèses suivantes :

- les documents sont générés de manière indépendante et de la même façon ;
- chaque document est associé à un unique topic ;
- tous les mots d'un document sont générés de manière indépendante, à partir d'une même mesure de probabilité discrète dépendant du topic associé au document.

La distribution associée est donc :

$$P_D(w_1, \ldots, w_N) = \sum_{z=1}^{K} p(z) \prod_{n=1}^{N} p(w_n|z)$$

Remarque 1. Ce modèle constitue une première illustration non triviale de l'hypothèse d'échangeabilité des mots. En effet, on peut le voir comme un cas particulier de la représentation (1) où $Y = Z$ est une v.a. discrète à valeurs dans $\{1, \ldots, K\}$.

1.4 Le modèle pLSI

Le modèle pLSI (*probabilistic Latent Semantic Indexing*) a été introduit en 1999 par Hoffmann [5] pour donner une base probabiliste rigoureuse au *Latent Semantic Indexing* (basé sur la décomposition de matrices en valeurs singulières). Cette approche offre plusieurs avantages par rapport au modèle déterministe, dont la possibilité d'utiliser l'arsenal statistique pour estimer les paramètres et la prise en compte explicite de la polysémie.

On adopte ici un point de vue différent des deux premiers modèles : on fixe dès le départ le nombre N de documents générés, puis on affecte arbitrairement un indice de

1 à N à chaque document [2]. Un document du corpus n'est alors plus décrit par la liste des mots qu'il contient, mais directement par son indice. Pour générer le corpus, on ne procède plus document par document mais mot par mot, pour chaque mot du corpus. Chaque mot du corpus correspond donc à la réalisation d'une variable aléatoire (D, W, Z), où :

- $D \in \{1, \dots, N\}$ est l'indice du document dans lequel se trouve le mot ;
- $W \in \{1, \dots, M\}$ est l'indice du mot dans le dictionnaire ;
- $Z \in \{1, \dots, K\}$ est le topic sous-jacent à *cette occurrence* du mot.

Soit n le nombre total de mots, non nécessairement distincts, du corpus. D'après ce qui précède, on peut décrire le modèle sous la forme d'une famille $((D_i, W_i, Z_i))_{1 \leq i \leq n}$ de n variables aléatoires. On fait alors les hypothèses suivantes :

- le processus de génération d'une occurence se répète n fois de manière indépendante, i.e. les variables (D_i, W_i, Z_i), $1 \leq i \leq n$ sont i.i.d ;
- conditionnellement à un topic, les documents et les mots sont indépendants, i.e. :

$$\forall i \in \{1, \dots, n\}, \quad \mathcal{L}(D_i, W_i \mid Z_i) = \mathcal{L}(D_i \mid Z_i) \otimes \mathcal{L}(W_i \mid Z_i).$$

Au final, la distribution d'une observation (d, w) est

$$p(d, w) = p(d) \sum_{z=1}^{K} p(w|z)p(z|d).$$

Remarque 2. Une différence fondamentale avec le modèle *mixture of unigrams* est que cette fois, les occurrences d'un même mot peuvent être associées à des topics différents. On a donc une souplesse double par rapport à ce qui précède : un document peut contenir plusieurs topics sous-jacents, et les occurrences d'un même mot dans plusieurs documents peuvent être associées à des topics différents.

En raison du changement d'approche par rapport aux modèles précédents, le modèle pLSI permet uniquement de décrire un corpus d'apprentissage fixé : comme le nombre de documents est fixé à l'avance, une fois les paramètres $p(z)$, $p(w|z)$ et $p(d|z)$ estimés il n'y a aucune façon naturelle d'attribuer une probabilité à un nouveau document. Lorsqu'on rajoute un nouveau document au corpus, il faut donc tout recalculer.

Enfin, le nombre de paramètres $p(z)$, $p(w|z)$, $p(d|z)$ d'un tel modèle est [3]

$$K + KM + KN,$$

ce qui donne une croissance linéaire par rapport au nombre de documents. Par exemple, dans le cas de la base de données que nous allons utiliser, on a $K = 20$, $M = 59809$ et $N = 19997$, ce qui fait... 1596140 paramètres. Outre les éventuels problèmes de vitesse de résolution, ce modèle présente donc un grand risque d'*overfitting* : il se trouve que c'est sa principale faiblesse [2]. Il semble donc important de chercher un modèle avec moins de paramètres : c'est le cas du modèle LDA que l'on aborde dans la section suivante.

2. Même si c'est plus implicite que dans les modèles précédents, le modèle pLSI fait donc également l'hypothèse "bag-of-documents".

3. Il y en a un peu moins si l'on prend en compte le fait que ce sont des probabilités (liées par des sommes qui valent 1), mais ce qui nous intéresse ici est l'ordre de grandeur.

1.5 Le modèle *Latent Dirichlet Allocation*

On souhaite construire un modèle de génération de documents dont le nombre de paramètres, contrairement au modèle précédent, ne dépend pas du nombre de documents. On souhaite aussi conserver son avantage principal, à savoir la souplesse au niveau des topics. Pour remplir ces deux objectifs en même temps, la clé est d'introduire un aléa supplémentaire : au lieu de considérer, à chaque génération d'un mot, un topic tiré selon une v.a. Z de loi fixée sur $\{1, \dots, K\}$, on prend Z de loi *aléatoire*, tirée pour chaque document.

Pour des raisons de simplicité on va plutôt considérer que Z suit une loi multinomiale de dimension K avec un lancer, ce qui revient au même. Une façon de générer aléatoirement cette loi est de considérer une loi de *Dirichlet* de paramètre $\alpha = (\alpha_1, \dots, \alpha_K)$ fixé, définie par sa densité sur $\mathbb{R}^K$:

$$f_\alpha(\theta) = f_\alpha(\theta_1, \dots, \theta_K) = \frac{\Gamma\left(\sum_{i=1}^K \alpha_i\right)}{\prod_{i=1}^K \Gamma(\alpha_i)} \theta_1^{\alpha_1 - 1} \dots \theta_K^{\alpha_K - 1}$$

si $\theta \in \Delta_K$ et $f_\alpha(\theta) = 0$ sinon, où

$$\Delta_K = \left\{ \theta \in \mathbb{R}^K \mid \forall i \in \{1, \dots, K\},\ \theta_i > 0 \quad \text{et} \quad \sum_{i=1}^K \theta_i = 1 \right\}.$$

Cette loi, notée $\mathrm{Dir}(\alpha)$, possède des propriétés très intéressantes (cf. remarque 3).

Le modèle *Latent Dirichlet Allocation* [2] (LDA) consiste donc, pour générer un document $D = (W_1, \dots, W_N)$ de N mots, à procéder de la manière suivante :

1. on tire θ selon une loi $\mathrm{Dir}(\alpha)$;
2. pour tout $n \in \{1, \dots, N\}$:
   (a) on tire un topic z_n selon une loi multinomiale $\mathcal{M}_K(1, \theta)$;
   (b) on tire un mot w_n dans $\{1, \dots, M\}$ avec une probabilité $p(w_n | z_n)$ fixée.

Au final, pour un document donné, les v.a. en jeu sont Θ, $Z_1, \dots, Z_N$ et $W_1, \dots, W_N$ avec les hypothèses suivantes :

- $\Theta \sim \mathrm{Dir}(\alpha)$
- $\forall n \in \{1, \dots, N\},\ \mathcal{L}(Z_n | \Theta) = \mathcal{M}_K(1, \Theta)$ et $\mathbb{P}(W_n = j | Z_n = e_i) = p_\beta(j, e_i) = \beta_{i,j}$ fixé,

où $(i, j) \in \{1, \dots, K\} \times \{1, \dots, M\}$, e_i est le i-ième vecteur de la base canonique de $\mathbb{R}^K$ et $\beta = (\beta_{i,j})$ est une matrice de taille $K \times M$. Le nombre total de paramètres du modèle est

$$K + KM$$

et on n'a donc pas de dépendance par rapport au nombre de documents du corpus.

La distribution associée au modèle est :

$$P_D(w_1, \dots, w_N) = \int_{\Delta_K} \left(\prod_{n=1}^N p(w_n | \theta) \right) f_\alpha(\theta)\, d\theta \tag{2}$$

avec

$$p(w_n | \theta) = \sum_{k=1}^K p_\beta(w_n | z_k) p(z_k | \theta).$$

La formulation (2) fait apparaître un mélange continu de lois produits : on est dans un cas particulier de la représentation de De Finetti (1), où Y suit une loi de Dirichlet. Ceci illustre une hypothèse d'échangeabilité spécifique à ce modèle : celle des topics au sein d'un document (ce qui est bien moins fort que l'hypothèse des Z_i i.i.d du modèle pLSI).

Remarque 3. On peut justifier l'utilisation de la loi de Dirichlet dans le modèle LDA de la façon suivante : d'un point de vue analytique, si l'on suppose que $\mathcal{L}(Z|\Theta = \theta) = \mathcal{M}_K(\theta, 1)$, on a alors *intérêt* à prendre $\mathcal{L}(\Theta) = \mathrm{Dir}_K(\alpha)$. En effet, on peut montrer que dans ce cas,

$$\mathcal{L}(\Theta|Z = z) = \mathrm{Dir}_K(\alpha + z) \quad \text{et} \quad \mathcal{L}(Z) = \mathcal{M}_K\left(\frac{\alpha}{\sum_{i=1}^K \alpha_i}, 1\right)$$

On dit que la loi de Dirichlet est *conjuguée* à la loi multinomiale. Des calculs explicites sur les algorithmes d'estimation (cf. suite) sont alors possibles, ce qui permet de les étudier analytiquement et parfois d'améliorer radicalement leur efficacité. On renvoie à [2] pour plus de précisions sur cet aspect.

2 Estimation dans le cas du modèle pLSI

Considérons le modèle pLSI défini plus haut : il s'agit maintenant de trouver une façon d'estimer les paramètres. On note dans cette section :

$$\alpha(z) = p(z), \quad \beta(d, z) = p(d|z) \quad \text{et} \quad \gamma(w, z) = p(w|z).$$

En fait, ce modèle peut être vu directement comme un modèle statistique classique, au sens où l'on considère des "observations" i.i.d (D_i, W_i, Z_i), $1 \leq i \leq n$. L'ennui est que l'on n'observe pas les variables Z_i : il va donc falloir faire sans. Au premier abord, on est tenté de procéder à une estimation par maximum de vraisemblance en considérant la log-vraisemblance "observée", que l'on peut calculer grâce à l'hypothèse d'indépendance conditionnelle :

$$\begin{aligned}
\ell(\bar{d}, \bar{w}; \alpha, \beta, \gamma) &= \sum_{i=1}^n \ln\left(\mathbb{P}(D_i = d_i, W_i = w_i)\right) \\
&= \sum_{i=1}^n \ln\left(\sum_{z=1}^K \alpha(z)\beta(d_i, z)\gamma(w_i, z)\right) \\
&= \sum_{d=1}^N \sum_{w=1}^M \bar{n}(d, w) \ln\left(\sum_{z=1}^K \alpha(z)\beta(d, z)\gamma(w, z)\right)
\end{aligned}$$

où $\bar{d} = (d_1, \ldots, d_n)$ et $\bar{w} = (w_1, \ldots, w_n)$ et $\bar{n}(d, w)$ est le nombre d'observations (d_i, w_i) égales à (d, w). Malheureusement, la forme de cette fonction (notamment la somme dans le log) de α, β, γ est telle qu'il semble illusoire de vouloir la maximiser...

On se tourne alors vers la log-vraisemblance "complétée" :

$$\ell_c(\bar{d}, \bar{w}, \bar{z}; \alpha, \beta, \gamma) = \sum_{i=1}^n \ln \alpha(z_i) + \ln \beta(d_i, z_i) + \ln \gamma(w_i, z_i).$$

Même si le calcul de celle-ci est impossible sans observer les Z_i on va voir qu'il est possible d'obtenir une estimation d'un maximum local de la vraisemblance observée ℓ via des calculs sur la vraisemblance complétée ℓ_c : c'est l'algorithme EM, pour *Expectation Maximization*.

2.1 Algorithme EM

On note ici $\theta = (\alpha, \beta, \gamma)$ pour simplifier. On souhaite construire un algorithme itératif qui, étant donné un paramètre initial $\theta^{(0)}$, converge vers un certain $\theta^{(\infty)}$ qui réalise un maximum local de ℓ. Le principe est de considérer l'espérance conditionnelle de ℓ_c sachant

les observations, puisqu'elle représente par définition l'information la plus précise que l'on puisse avoir sur ℓ_c : c'est l'étape *Expectation*. Plus précisément, on calcule la fonction

$$\theta \mapsto Q(\theta, \theta^{(t)}) = \mathbb{E}_{\theta^{(t)}} \left[\ell_c(\bar{D}, \bar{W}, \bar{Z}; \theta) | \bar{D}, \bar{W} \right]$$

où $\bar{D} = (D_1, \dots, D_n)$, $\bar{W} = (W_1, \dots, W_n)$, $\bar{Z} = (Z_1, \dots, Z_n)$ et $\mathbb{E}_{\theta^{(t)}}$ désigne l'espérance sous la loi $\mathcal{L}(\bar{D}, \bar{W}, \bar{Z})$ de paramètre $\theta^{(t)}$. La quantité $\ell_c(\bar{D}, \bar{W}, \bar{Z}; \theta)$ est donc vue dans l'espérance conditionnelle comme une variable aléatoire qui n'est fonction que de $\bar{Z}$.

Il s'agit ensuite de maximiser cette espérance : c'est l'étape *Maximization*. Cette fois-ci c'est faisable puisque la forme de la fonction à maximiser est bien plus simple (si si) :

$$\begin{aligned} Q(\theta, \theta^{(t)}) &= \sum_{i=1}^{n} \mathbb{E}_{\theta^{(t)}} [\ln \alpha(Z_i) + \ln \beta(D_i, Z_i) + \ln \gamma(W_i, Z_i) \mid D_i, W_i] \\ &= \sum_{d=1}^{N} \sum_{w=1}^{M} \bar{n}(d, w) \sum_{z=1}^{K} p^{(t)}(z|d, w)(\ln \alpha(z) + \ln \beta(d, z) + \ln \gamma(w, z)) \end{aligned}$$

où

$$p^{(t)}(z|d, w) = \frac{\alpha^{(t)}(z)\beta^{(t)}(d, z)\gamma^{(t)}(w, z)}{\sum_{z'} \alpha^{(t)}(z')\beta^{(t)}(d, z')\gamma^{(t)}(w, z')}.$$

Puisque les paramètres vérifient des conditions de somme égale à 1 (associées au fait que ce sont des probabilités), on peut appliquer le théorème des extrema liés. Par un calcul classique des multiplicateurs de Lagrange [6], on obtient le point $\theta^{(t+1)}$ suivant qui réalise le maximum de $Q(\cdot, \theta^{(t)})$:

$$\alpha^{(t+1)}(z) = \frac{1}{n} \sum_{d,w} \bar{n}(d, w) p^{(t)}(z|d, w)$$

$$\beta^{(t+1)}(d, z) = \frac{\sum_w \bar{n}(d, w) p^{(t)}(z|d, w)}{\sum_{d',w} \bar{n}(d', w) p^{(t)}(z|d', w)}$$

$$\gamma^{(t+1)}(w, z) = \frac{\sum_d \bar{n}(d, w) p^{(t)}(z|d, w)}{\sum_{d,w'} \bar{n}(d, w') p^{(t)}(z|d, w')}$$

avec $n = \sum_{d,w} \bar{n}(d, w)$, le nombre total d'observations.

L'algorithme EM ainsi construit fournit une suite de paramètres $(\theta^{(t)})_{t \in \mathbb{N}}$. L'intuition que l'on avait en prenant l'espérance conditionnelle est justifiée par le résultat suivant :

Théorème 2. *La vraisemblance est croissante le long de l'algorithme EM. En d'autres termes, la suite $(\theta^{(t)})_{t \in \mathbb{N}}$ vérifie*

$$\forall t \in \mathbb{N}, \quad \ell(\bar{d}, \bar{w}; \theta^{(t+1)}) \geq \ell(\bar{d}, \bar{w}; \theta^{(t)}).$$

De plus, si $\theta^{(t)}$ n'est pas un point stationnaire, i.e. n'annule pas le gradient de ℓ, alors

$$\ell(\bar{d}, \bar{w}; \theta^{(t+1)}) > \ell(\bar{d}, \bar{w}; \theta^{(t)}).$$

Notons que malgré ce résultat intéressant, on est condamné à ne converger que vers un maximum local dépendant du point initial $\theta^{(0)}$: il faut donc être attentif à l'initialisation de l'algorithme. De plus, les résultats éventuels sur la vitesse de convergence, sont plus complexes et nécessitent des hypothèses supplémentaires : nous ne les abordons pas ici, et dans la pratique nous nous contenterons plutôt de constater empiriquement la convergence.

Dans la section suivante, on applique tout qui précède à une base de données textuelles "grandeur nature", puis on interprète les résultats.

3 Application à une vraie base de données

On travaille sur un corpus classique utilisé dans ce domaine : la base de données `20Newsgroups` [4], qui répertorie environ 20000 messages provenant en proportions égales de 20 groupes de discussion, soit 20 thèmes différents. L'avantage de ce type de données est que de vrais thèmes sont connus : on est donc capable de déterminer l'efficacité de différents modèles afin de pouvoir les comparer. On donne en annexe dans le tableau 1 l'intitulé de chaque catégorie, ainsi qu'un indice entre 1 et 20 qu'on lui affecte dans le cadre de notre étude.

3.1 Nettoyage des données

Avant de pouvoir s'attaquer à la modélisation, il nous faut d'abord déterminer un dictionnaire. Or, si l'on se penche sur les données, on se rend compte qu'un très grand nombre de mots ne va absolument pas nous aider à déterminer des topics sous-jacents : on a notamment des caractères alphanumériques, des conjonctions de coordination, des signes de ponctuations, etc. On procède donc à un nettoyage des données en utilisant un dictionnaire de *stopwords*, i.e. une liste de mots à enlever systématiquement des textes.

De plus, pour une meilleure efficacité de l'algorithme, on ne garde que les mots qui apparaissent au moins deux fois dans le corpus : on espère ainsi diminuer le biais dû aux fautes de frappes et autres néologismes personnels. Avec les notations précédentes, on fixe les constantes du modèle à $K = 20$, $M = 59809$ et $N = 19997$.

On décide ensuite d'implémenter le modèle pLSI. Les documents une fois nettoyés sont récupérés sous forme *sparse*, un fichier qui ne contient que les $\bar{n}(d, w)$ non nuls et qui sont en fait les seules entrées utiles. Notons que, si l'algorithme EM est très simple à programmer dans les cas d'école, il faut réfléchir un peu plus pour l'appliquer à une grande base de données, car le temps de calcul devient dans ce cadre une vraie question.

On implémente [5] donc un algorithme en C++ qui utilise au maximum cette structure sparse. Cela diminue drastiquement le nombre d'opérations à faire et donc la vitesse de calcul : on passe de $n = 2082550$ observations (couples (D_i, W_i)) à $u = 1501986$ entrées utiles. Il faut environ 5 secondes pour réaliser une itération de EM. Les figurent 1 et 2 en annexe illustrent les résultats de croissance et de convergence pour 100 itérations. En poursuivant jusqu'à 300 itérations, on constate qu'il n'y a pas de changement radical, à part pour les valeurs proches de 0.05 (ce qui a peu d'importance puisque les topics associés sont alors à peu près équiprobables). Les résultats sont donc globalement satisfaisants.

3.2 Clustering

Notre algorithme nous donne des estimations de $p(z), p(d|z)$ et $p(w|z)$ pour tous d, w, z. On cherche maintenant à faire un *clustering*, c'est-à-dire classer les différentes observations en assignant à chacune un topic $z \in \{1, \dots, K\}$. On n'est pas sûr de pouvoir assigner à une observation le véritable topic dont elle est issue, puisque par définition la variable Z n'est observée. En revanche, on peut trouver le topic qui maximise sa probabilité d'apparition : c'est la méthode dite du *Maximum A Posteriori* (MAP). L'idée est d'utiliser la formule de Bayes pour se ramener à maximiser des quantités connues.

4. `http://www.qwone.com/~jason/20Newsgroups`
5. `http://perso.eleves.bretagne.ens-cachan.fr/~uherb801/download/projet_topic_models`

Même si le modèle génératif permet à un document de contenir plusieurs topics, on considère ici qu'un document a toujours un topic "principal", i.e. majoritairement présent dans le document. À un document d donné, on associe le topic z si et seulement si

$$p(z|d) > p(z'|d), \quad \forall z' \neq z.$$

Cette maximisation est aisément réalisable : d'après la formule de Bayes

$$p(z|d) = \frac{p(z)}{p(d)} p(d|z),$$

le problème se ramène à maximiser $p(z)p(d|z)$, quantité que l'on sait estimer grâce aux estimations de $p(z)$ et $p(d|z)$.

3.3 Étude d'efficacité

Pour étudier l'efficacité du modèle, on a besoin d'identifier les clusters et les "vraies" catégories (on a justement pris $K = 20$ à cet effet). On procède de la manière suivante : à chaque cluster, on attribue la catégorie majoritairement présente dans ce cluster, i.e. celle qui contient le plus de documents associés au cluster. Avec cette technique, on est sûr que chaque cluster est associé à une seule catégorie. En revanche, il est possible que plusieurs clusters soient associés à la même catégorie.

On peut maintenant définir la mesure d'efficacité. Soit i un cluster et soit j la catégorie à laquelle il a été associé. On introduit alors les quantités suivantes :

- $C(i,j)$ = nombre de documents de la catégorie j affectés au cluster i
- $T_{clu}(i)$ = nombre de documents du cluster i
- $T_{cat}(j)$ = nombre de documents de la catégorie j
- $P_i = \dfrac{C(i,j)}{T_{clu}(i)}$ "mesure de précision du cluster i"
- $R_i = \dfrac{C(i,j)}{T_{cat}(j)}$ "mesure de rappel du cluster i"
- $F_i = 2\dfrac{P_i R_i}{P_i + R_i}$ "F-mesure (ou efficacité) du cluster i"

Remarque 4. On a au final

$$F_i = \frac{2C(i,j)}{T_{clu}(i) + T_{cat}(j)}.$$

Vérifions la cohérence de cette mesure. On a par définition

$$0 \leq C(i,j) \leq \min(T_{clu}(i), T_{cat}(j)) \leq \frac{1}{2}(T_{clu}(i) + T_{cat}(j))$$

d'où $0 \leq F_i \leq 1$, avec $F_i = 1$ si et seulement si $C(i,j) = T_{clu}(i) = T_{cat}(j)$, c'est-à-dire si et seulement si le cluster i et la catégorie j correspondent parfaitement.

On constate que pour pLSI, 300 itérations de l'algorithme EM donnent un résultat assez satisfaisant, visible à la fois sur les clusters (tableau 2 en annexe) et sur les F-mesures (figure 3 en annexe).

4 Conclusion et perspectives

Dans l'ensemble, le modèle pLSI semble avoir très bien capté les thèmes sous-jacents de la base de données. Si l'on compare le contenu des clusters générés par l'algorithme avec les catégories qui leur ont été affectées, on observe une réelle correspondance. Les performances sont par ailleurs bien visibles sur les graphiques mis en annexe.

Le seul problème qui se pose parfois est qu'un topic a capté autre chose qu'une catégorie existante : on observe alors des incohérences avec l'intitulé de la catégorie, mais si on omet cette catégorie "imposée", on se rend compte que le topic décrit bien un thème, plus implicite qu'une catégorie mais néanmoins cohérent. Les temps de calculs restent acceptables, mais il faudrait sérieusement envisager un modèle comportant moins de paramètres si l'on avait affaire à une base de données plus grande à traiter.

Faute de temps, nous n'avons pas pu aborder en détail le modèle LDA : il serait intéressant de poursuivre notre étude en détaillant différents algorithmes permettant d'estimer ses paramètres, en particulier l'algorithme "EM variationnel" [2], ainsi que des méthodes MCMC comme l'échantillonneur de Gibbs [4].

Enfin, le théorème de De Finetti fait entrevoir toute une classe de modèles intéressants qui vont au delà du "simple" LDA : on sait en effet qu'il existe des variantes de ce théorème sous des hypothèses plus faibles. Celles-ci permettraient de concevoir de nouveaux modèles dans deux directions différentes :

- des modèles nécessitant moins d'hypothèses donc encore plus souples ;
- des modèles spécialisés associés à des hypothèses très spécifiques, par exemple pour traiter des données d'un type bien particulier (images, génomes...)

On constate que des résultats abstraits comme le théorème de représentation de De Finetti peuvent parfois guider l'intuition pour concevoir des modèles concrets efficaces.

Références

[1] E. Bart, M. Welling, and P. Perona, *Unsupervised organization of image collections : Taxonomies and beyond*, Trans. Pattern Recognit. Mach. Intell. **33** (2010).

[2] David M. Blei, Andrew Y. Ng, and Michael I. Jordan, *Latent Dirichlet Allocation*, Journal of Machine Learning Research (2003), no. 3, 993–1022.

[3] Y.S. Chow and H Teicher, *Probability theory : independence, interchangeability, martingales*, Springer Verlag, 1997.

[4] William M. Darling, *A Theoretical and Practical Implementation Tutorial on Topic Modeling and Gibbs Sampling*, Tech. report, School of Computer Science, University of Guelph, 2011.

[5] Thomas Hofmann, *Probabilistic Latent Semantic Indexing*, Proceedings of the Twenty-Second Annual International SIGIR Conference on Research and Development in Information Retrievial (1999).

[6] Liangjie Hong, *Probabilistic latent semantic analysis*, Tech. report, Department of Computer Science and Engineering, Lehigh University, 2012.

[7] K. Nigam, J. Lafferty, and A. McCallum, *Using maximum entropy for text classification*, IJCAI-99 Workshop on Machine Learning for Information Filtering (1999), 61–67.

[8] J. Pritchard, M. Stephens, and P. Donnelly, *Inference of population structure using multilocus genotype data*, Genetics **155** (2000).

Annexe

A Catégories de départ de la base *20Newsgroups*

Indice	Intitulé	Indice	Intitulé
1	« ALT »	11	« RecHockey »
2	« CompG »	12	« SciCrypt »
3	« CompOS »	13	« SciElectronics »
4	« CompIBM »	14	« SciMed »
5	« CompMAC »	15	« SciSpace »
6	« CompWindows »	16	« SocReligion »
7	« MISC »	17	« TalkGuns »
8	« RecAutos »	18	« TalkMideast »
9	« RecMotos »	19	« TalkMisc »
10	« RecBaseball »	20	« TalkReligion »

TABLEAU 1 – Catégories de la base de données *20Newsgroups*

B Performances de l'algorithme EM

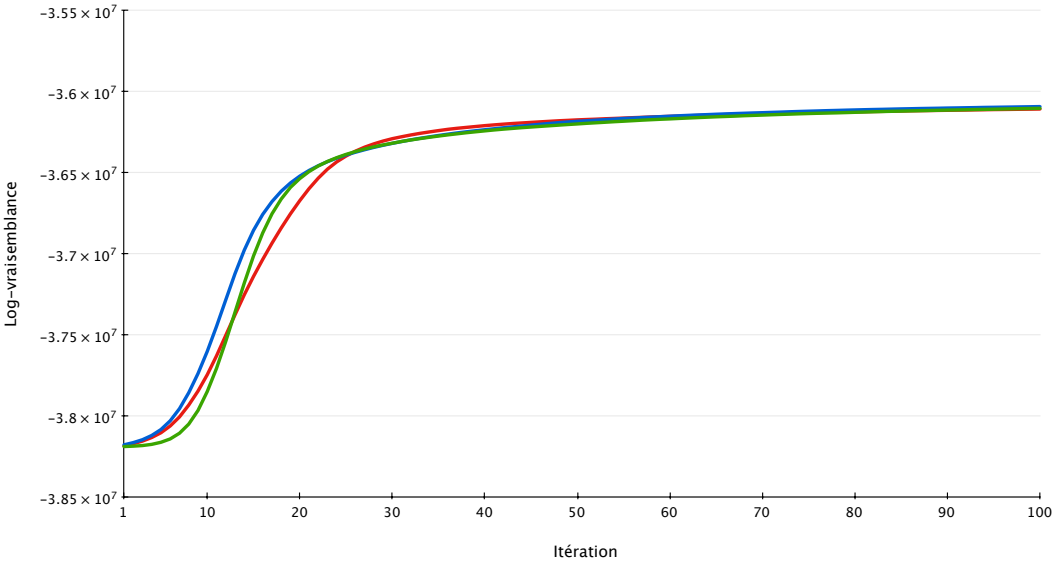

FIGURE 1 – Comparaison de la log-vraisemblance pour trois initialisations différentes. Notons que la vraisemblance initiale (non tracée ici) peut être assez différente selon le choix des paramètres, mais toutes les courbes deviennent très proches dès la première itération.

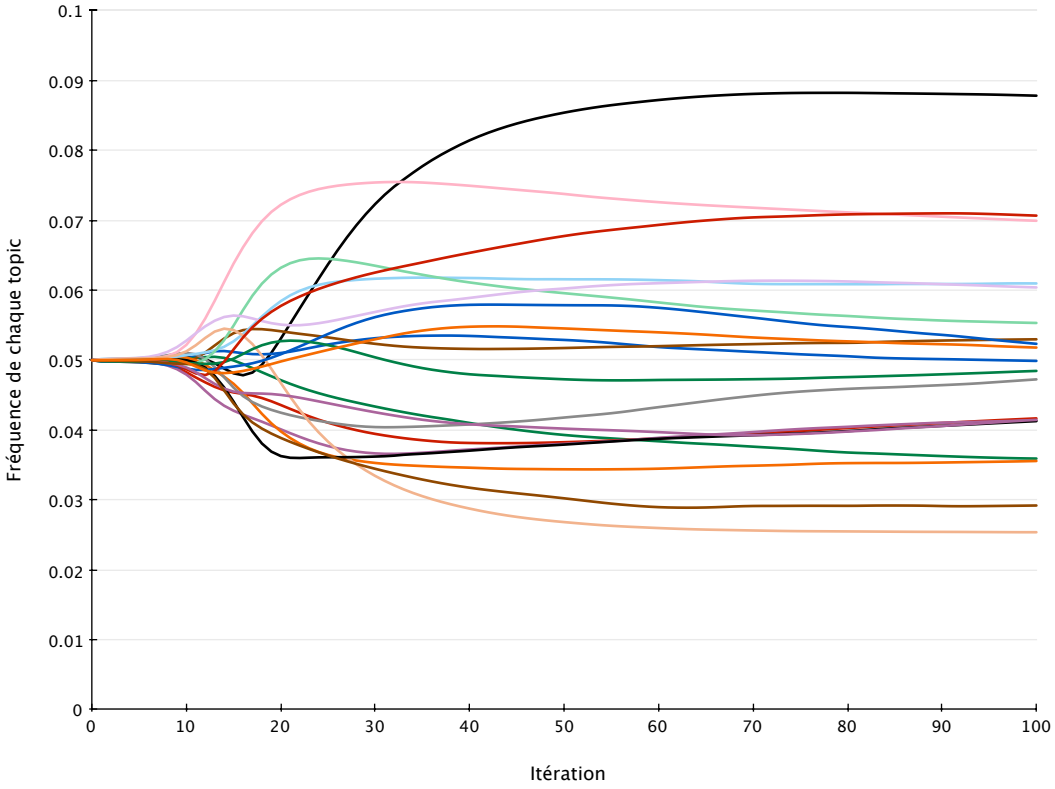

FIGURE 2 – Évolution des probabilités respectives des 20 topics le long de l'algorithme EM. Chaque courbe représente la valeur du coefficient $\alpha(z)$ associé au topic z en fonction du nombre d'itérations. On peut remarquer que, même si des bifurcations apparaissent, les $\alpha(z)$ restent proches de la valeur 0.05, qu'ils auraient si les topics étaient équiprobables.

« Médecine »	« Baseball »	« Espace »	« Informatique »
medicine	braves	shuttle	file
risk	sox	payload	format
candida	pitcher	radar	pub
diseases	hitter	comet	information
women	cubs	titan	send
physician	batting	jet	number
hicnet	alomar	orbiter	address
infections	clemens	aerospace	contact
physicians	phillies	centaur	ftp
lyme	innings	baalke	site

TABLEAU 2 – Résultats du clustering : quelques thèmes générés par le modèle pLSI

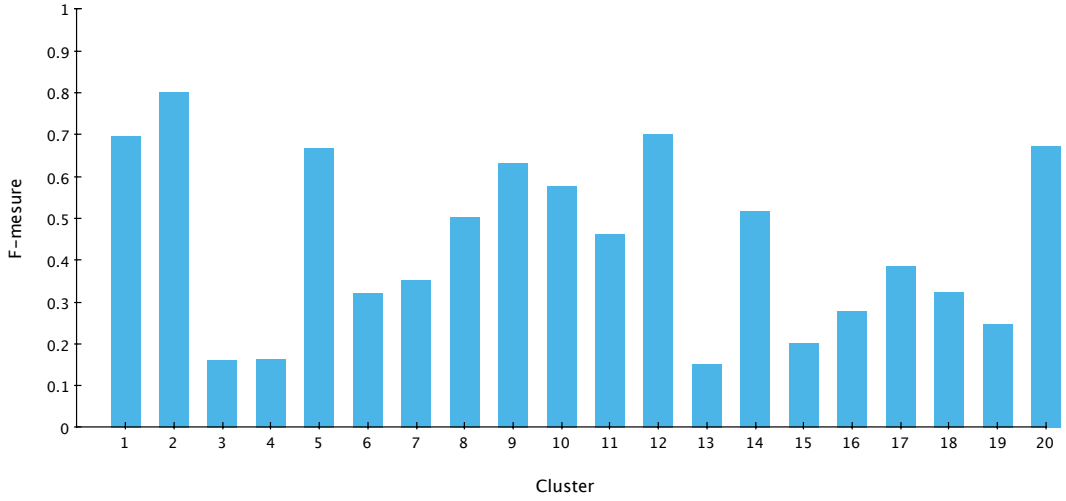

FIGURE 3 – F-mesure de chacun des 20 clusters (topics) en sortie de l'algorithme EM

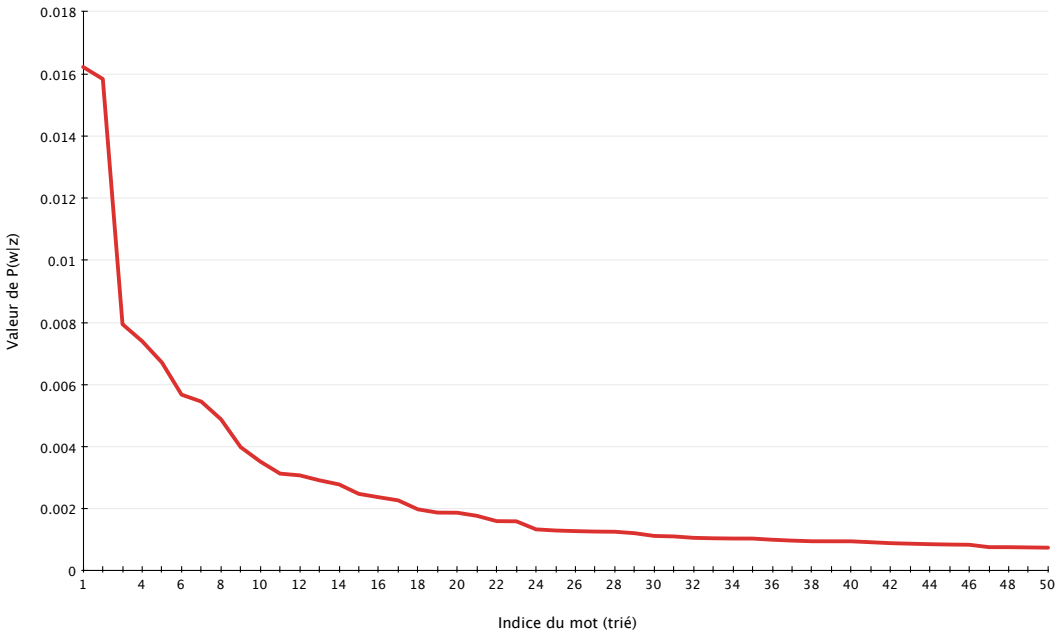

FIGURE 4 – Un autre critère d'efficacité possible : l'étude de $p(w|z)$ en fonction de w pour un topic z fixé (ici le topic $z = 8$) permet de voir si les topics "sélectionnent" bien des mots particuliers. Une forte concentration vers les abscisses petites correspond à un topic très sélectif, tandis qu'une courbe constante correspond à un topic non sélectif donc inutile.

C Résultats détaillés - Modèle pLSI

Thème 1	Thème 2	Thème 3	Thème 4	Thème 5
key	window	god	car	disease
encryption	server	jesus	cars	patients
clipper	motif	people	insurance	cancer
keys	widget	christian	pay	medicine
security	lib	bible	engine	doctors
technology	xterm	church	buy	diet
algorithm	font	christians	abortion	vitamin
escrow	usr	christ	ford	infection
communications	directory	faith	coverage	patient
encrypted	clients	gods	driving	symptoms

Thème 6	Thème 7	Thème 8	Thème 9	Thème 10
team	space	image	israel	windows
games	moon	file	israeli	drive
players	shuttle	images	arab	card
baseball	satellite	graphics	arabs	mac
season	cost	format	palestinians	disk
won	station	pub	palestinian	software
player	pat	information	lebanese	memory
morris	mars	send	policy	computer
pitching	vehicle	color	lebanon	apple
braves	satellites	number	israelis	monitor

Thème 11	Thème 12	Thème 13	Thème 14	Thème 15
government	wire	article	turkish	gun
fire	circuit	science	armenian	guns
koresh	wiring	objective	armenians	police
rights	cable	moral	armenia	crime
law	radio	morality	turks	firearms
compound	neutral	values	genocide	criminals
state	signal	theory	history	weapon
militia	supply	things	russian	carry
constitution	voltage	agree	muslim	cops
amendment	input	frank	soviet	clayton

Thème 16	Thème 17	Thème 18	Thème 19	Thème 20
water	home	group	committee	kent
light	apartment	posting	national	gant
nuclear	food	groups	convention	books
black	mother	american	business	cheers
energy	woman	san	united	sandviknewtonapplecom
sky	wife	jobs	parties	game
air	kids	last	institute	hirschbeck
hole	women	clinton	drug	umpire
henry	young	money	karl	pitch
universe	hours	francisco	libertarian	dreams

Tableau 3 – Résultats du clustering : les 10 mots les plus probables de chaque thème

D Preuve des résultats sur l'algorithme EM

D.1 Formulation explicite du maximum

Afin d'obtenir une formulation explicite de l'étape *Maximization* de l'algorithme EM, on va utiliser la méthode des multiplicateurs de Lagrange, i.e. le théorème suivant :

Théorème 3 (Extrema liés)**.** *Soient $p, r \in \mathbb{N}^*$, U un ouvert de $\mathbb{R}^p$, et $f, g_1, \dots, g_r \in \mathcal{C}^1(U, \mathbb{R})$. On considère l'ensemble suivant :*

$$\Gamma = \{x \in U \mid \forall i \in \{1, \dots, r\},\ g_i(x) = 0\}.$$

On suppose que $f_{|\Gamma}$ admet un extremum local en $a \in \Gamma$ et que la famille $(\mathrm{d}g_i(a))_{1 \leq i \leq r}$ est libre dans $(\mathbb{R}^p)'$. Alors les formes linéaires $\mathrm{d}f(a), \mathrm{d}g_1(a), \dots, \mathrm{d}g_r(a)$ sont liées. En d'autres termes, il existe $\lambda_1, \dots, \lambda_r \in \mathbb{R}$ appelés multiplicateurs de Lagrange, tels que

$$\mathrm{d}f(a) = \sum_{i=1}^{r} \lambda_i \, \mathrm{d}g_i(a).$$

Pour simplifier, on note $\theta = (\alpha, \beta, \gamma) \in \mathbb{R}^K \times \mathbb{R}^{KN} \times \mathbb{R}^{KM} = \mathbb{R}^{K+KN+KM}$ les coefficients $(\alpha(z))_{1 \leqslant z \leqslant K}$, $(\beta(d,z))_{1 \leqslant d \leqslant N, 1 \leqslant z \leqslant K}$ et $(\gamma(w,z))_{1 \leqslant w \leqslant M, 1 \leqslant z \leqslant K}$. On cherche donc à appliquer le théorème précédent avec $p = K + KN + KM$ la fonction $f : \theta \mapsto Q(\theta, \theta^{(t)})$ définie sur $U = (\mathbb{R}_+^*)^p$ par :

$$\begin{aligned} Q(\theta, \theta^{(t)}) &= \sum_{i=1}^{n} \mathbb{E}_{\theta^{(t)}}[\ln \alpha(Z_i) + \ln \beta(D_i, Z_i) + \ln \gamma(W_i, Z_i) \mid D_i, W_i] \\ &= \sum_{d=1}^{N} \sum_{w=1}^{M} \bar{n}(d,w) \sum_{z=1}^{K} p^{(t)}(z|d,w)(\ln \alpha(z) + \ln \beta(d,z) + \ln \gamma(w,z)) \end{aligned}$$

où

$$p^{(t)}(z|d,w) = \frac{\alpha^{(t)}(z)\beta^{(t)}(d,z)\gamma^{(t)}(w,z)}{\sum_{z'} \alpha^{(t)}(z')\beta^{(t)}(d,z')\gamma^{(t)}(w,z')}.$$

Les $r = 1 + 2K$ contraintes de maximisation sont :

$$\sum_{z=1}^{K} \alpha(z) = 1 \quad \text{et} \quad \forall z \in \{1 \dots, K\}, \quad \sum_{d=1}^{N} \beta(d,z) = \sum_{w=1}^{M} \gamma(w,z) = 1$$

On peut les réécrire sous la forme

$$g_\alpha(\theta) = 0 \quad \text{et} \quad \forall z \in \{1 \dots, K\}, \quad g_{\beta,z}(\theta) = g_{\gamma,z}(\theta) = 0$$

où $g_\alpha(\theta) = \sum_{z=1}^{K} \alpha(z) - 1$, $g_{\beta,z}(\theta) = \sum_{d=1}^{N} \beta(d,z) - 1$ et $g_{\gamma,z}(\theta) = \sum_{w=1}^{M} \gamma(w,z) - 1$.

Tout d'abord, on remarque que U est bien un ouvert (convexe) de $\mathbb{R}^p$ et que f et g_α, $g_{\beta,z}$ et $g_{\gamma,z}$ sont de classe $\mathcal{C}^1$ sur U. De plus, f est strictement concave sur U en tant que somme finie de fonctions strictement concaves (logarithmes). L'ensemble Γ défini comme ci-dessus est également convexe et borné. Enfin, on a pour tout $i \in \{1, \dots, p\}$, $f(\theta) \to -\infty$ quand $\theta_i \to 0^+$. On en déduit que $f_{|\Gamma}$ admet un maximum global. De plus, comme $f_{|\Gamma}$ est strictement concave, ce maximum global est atteint en un unique point $\theta^{(t+1)} \in \Gamma$.

D'après le théorème, on a alors l'existence de $2K+1$ réels λ_α, $(\lambda_{\beta,z})_{1 \leqslant z \leqslant K}$ et $(\lambda_{\beta,z})_{1 \leqslant z \leqslant K}$ tels que

$$\mathrm{d}f(\theta^{(t+1)}) = \lambda_\alpha \, \mathrm{d}g_\alpha(\theta^{(t+1)}) + \sum_{z=1}^{K} \lambda_{\beta,z} \, \mathrm{d}g_{\beta,z}(\theta^{(t+1)}) + \sum_{z=1}^{K} \lambda_{\gamma,z} \, \mathrm{d}g_{\gamma,z}(\theta^{(t+1)}),$$

ce qui se traduit par :

$$\forall z \in \{1\dots,K\}, \quad \frac{1}{\alpha^{(t+1)}(z)} \sum_{d=1}^{N}\sum_{w=1}^{M} \bar{n}(d,w)p^{(t)}(z|d,w) = \lambda_\alpha$$

$$\forall d \in \{1\dots,N\}, \quad \forall z \in \{1\dots,K\}, \quad \frac{1}{\beta^{(t+1)}(d,z)} \sum_{w=1}^{M} \bar{n}(d,w)p^{(t)}(z|d,w) = \lambda_{\beta,z}$$

$$\forall w \in \{1\dots,M\}, \quad \forall z \in \{1\dots,K\}, \quad \frac{1}{\gamma^{(t+1)}(w,z)} \sum_{d=1}^{N} \bar{n}(d,w)p^{(t)}(z|d,w) = \lambda_{\gamma,z}$$

En utilisant le fait que $\theta^{(t+1)} \in \Gamma$, on obtient finalement :

$$\alpha^{(t+1)}(z) = \frac{1}{n}\sum_{d,w} \bar{n}(d,w)p^{(t)}(z|d,w)$$

$$\beta^{(t+1)}(d,z) = \frac{\sum_w \bar{n}(d,w)p^{(t)}(z|d,w)}{\sum_{d',w} \bar{n}(d',w)p^{(t)}(z|d',w)}$$

$$\gamma^{(t+1)}(w,z) = \frac{\sum_d \bar{n}(d,w)p^{(t)}(z|d,w)}{\sum_{d,w'} \bar{n}(d,w')p^{(t)}(z|d,w')}$$

avec $n = \sum_{d,w} \bar{n}(d,w)$, le nombre total d'observations.

D.2 Croissance de la vraisemblance

L'algorithme EM ainsi construit fournit une suite de paramètres $(\theta^{(t)})_{t\in\mathbb{N}}$. L'intuition que l'on avait en prenant l'espérance conditionnelle est justifiée par le résultat suivant :

Théorème 4. *La vraisemblance est croissante le long de l'algorithme EM. En d'autres termes, la suite $(\theta^{(t)})_{t\in\mathbb{N}}$ vérifie*

$$\forall t \in \mathbb{N}, \quad \ell(\bar{d},\bar{w};\theta^{(t+1)}) \geq \ell(\bar{d},\bar{w};\theta^{(t)}).$$

De plus, si $\nabla_\theta \ell(\bar{d},\bar{w};\theta^{(t)}) \neq 0$, alors

$$\ell(\bar{d},\bar{w};\theta^{(t+1)}) > \ell(\bar{d},\bar{w};\theta^{(t)}).$$

Preuve. Dans ce qui suit, on note pour simplifier $\bar{X} = (\bar{D},\bar{W})$, $\bar{x} = (\bar{d},\bar{w})$, $\mathbb{P}_\theta(\bar{X}) = \mathbb{P}_\theta(\bar{X} = \bar{x})$ et $\mathbb{P}_\theta(\bar{Z}|\bar{X}) = \mathbb{P}_\theta(\bar{Z} = \bar{z}, \bar{X} = \bar{x})$. L'idée est de conditionner par la variable latente $\bar{Z}$ en utilisant la relation :

$$\mathbb{P}_\theta(\bar{Z}|\bar{X}) = \frac{\mathbb{P}_\theta(\bar{Z},\bar{X})}{\mathbb{P}_\theta(\bar{X})}.$$

On a alors, pour tout $\theta \in U$:

$$\ell(\bar{X};\theta) = \ln\left(\mathbb{P}_\theta(\bar{X})\right) = \ln\left(\mathbb{P}_\theta(\bar{Z},\bar{X})\frac{\mathbb{P}_\theta(\bar{X})}{\mathbb{P}_\theta(\bar{Z},\bar{X})}\right) = \ln\left(\mathbb{P}_\theta(\bar{Z},\bar{X})\right) - \ln\left(\mathbb{P}_\theta(\bar{Z}|\bar{X})\right),$$

ce qui donne, en notant $\ell(\bar{Z}|\bar{X};\theta) = \ln\left(\mathbb{P}_\theta(\bar{Z}|\bar{X})\right)$:

$$\ell(\bar{X};\theta) = \ell_c(\bar{X},\bar{Z};\theta) - \ell(\bar{Z}|\bar{X};\theta).$$

Soit $t \in \mathbb{N}$, on considère que $\theta^{(t)}$ et $\theta^{(t+1)}$ sont fixés comme ci-dessus. On a alors, pour tout $\theta \in U$, en prenant l'espérance par rapport à la loi $\mathbb{P}_{\theta^{(t)}}(\,\cdot\,|\bar{X})$:

$$\begin{aligned}\ell(\bar{X};\theta) = \mathbb{E}_{\theta^{(t)}}[\ell(\bar{X};\theta)|\bar{X}] &= \mathbb{E}_{\theta^{(t)}}[\ell_c(\bar{X},\bar{Z};\theta)|\bar{X}] - \mathbb{E}_{\theta^{(t)}}[\ell(\bar{Z}|\bar{X};\theta)|\bar{X}] \\ &= Q(\theta,\theta^{(t)}) - Q_1(\theta,\theta^{(t)})\end{aligned}$$

où $Q_1(\theta,\theta^{(t)}) = \mathbb{E}_{\theta^{(t)}}[\ell(\bar{Z}|\bar{X};\theta)|\bar{X}]$. On a finalement :

$$\ell(\bar{X};\theta^{(t+1)}) - \ell(\bar{X};\theta^{(t)}) = \underbrace{Q(\theta^{(t+1)},\theta^{(t)})}_{①} - Q(\theta^{(t)},\theta^{(t)}) - (\underbrace{Q_1(\theta^{(t+1)},\theta^{(t)})}_{②} - Q_1(\theta^{(t)},\theta^{(t)}))$$

Tout d'abord, il est clair que ① $\geqslant 0$ par définition de $\theta^{(t+1)}$. Il s'agit ensuite de montrer que ② $\leqslant 0$: on va pour cela utiliser l'inégalité de Jensen avec la concavité du logarithme. On a en effet, pour tout $\theta \in U$:

$$\begin{aligned}Q_1(\theta,\theta^{(t)}) - Q_1(\theta^{(t)},\theta^{(t)}) &= \mathbb{E}_{\theta^{(t)}}[\ell(\bar{Z}|\bar{X};\theta) - \ell(\bar{Z}|\bar{X};\theta^{(t)})|\bar{X}] \\ &= \mathbb{E}_{\theta^{(t)}}\left[\ln\left(\frac{\mathbb{P}_\theta(\bar{Z}|\bar{X})}{\mathbb{P}_{\theta^{(t)}}(\bar{Z}|\bar{X})}\right)|\bar{X}\right] \\ &\leqslant \ln\left(\mathbb{E}_{\theta^{(t)}}\left[\frac{\mathbb{P}_\theta(\bar{Z}|\bar{X})}{\mathbb{P}_{\theta^{(t)}}(\bar{Z}|\bar{X})}|\bar{X}\right]\right) \\ &\leqslant \ln\left(\sum_{\bar{z}} \frac{\mathbb{P}_\theta(\bar{Z}=\bar{z}|\bar{X})}{\mathbb{P}_{\theta^{(t)}}(\bar{Z}=\bar{z}|\bar{X})}\mathbb{P}_{\theta^{(t)}}(\bar{Z}=\bar{z}|\bar{X})\right) \\ &\leqslant \ln\left(\sum_{\bar{z}} \mathbb{P}_\theta(\bar{Z}=\bar{z}|\bar{X})\right) \\ &\leqslant \ln(1) \\ &\leqslant 0\end{aligned}$$

et une fois ce résultat appliqué en $\theta = \theta^{(t+1)}$, on obtient l'inégalité escomptée.

Supposons maintenant [6] que $\nabla_\theta \ell(\bar{d},\bar{w};\theta^{(t)}) \neq 0$. L'inégalité précédente nous montre que la fonction $\theta \mapsto Q_1(\theta,\theta^{(t)})$ est maximale en $\theta = \theta^{(t)}$. On en déduit que $\nabla_\theta Q_1(\theta,\theta^{(t)}) = 0$ en $\theta = \theta^{(t)}$, puis que

$$\nabla_\theta Q(\theta,\theta^{(t)}) = \nabla_\theta \ell(\bar{d},\bar{w};\theta) \neq 0$$

en $\theta = \theta^{(t)}$, d'où $Q(\theta^{(t+1)},\theta^{(t)}) > Q(\theta^{(t)},\theta^{(t)})$ et finalement $\ell(\bar{X};\theta^{(t+1)}) > \ell(\bar{X};\theta^{(t)})$.

□

6. On suppose implicitement que $\theta \mapsto \ell(\bar{d},\bar{w};\theta)$ et toutes les autres fonctions considérées sont régulières, ce qui est notamment le cas pour le modèle pLSI.