Université Paris-Saclay
UFR de Mathématiques

Projet d'apprentissage non-supervisé avancé

Pierre CAVALIER, Virgile Bertrand

Lien du repository git

Année universitaire 2023-2024

Résumé

Dans ce rapport nous faisons l'étude de l'article *The latent topic block model for the co-clustering of textual interaction data* publié en 2019 par Bergé et. al. [Ber+19]. Le modèle LTBM introduit dans l'article vise à réaliser un co-clustering d'interaction textuelle entre deux ensembles d'*individus*/*objets* disjoints. On pourra penser par exemple aux clients et produits d'un site de vente en ligne, les clients intéragissant avec les produits par des commentaires. Ou encore aux utilisateurs et vidéos d'une plateforme tel que Youtube.
Nous présentons ici ce modèle ainsi que l'algorithme proposé pour en estimer les paramètres. Celui ci repose sur une méthode variationnelle pour maximiser la vraissemblance du modèle. Nous en présentons également les performances sur quelques jeux de données synthétique.

Table des matières

1 Introduction

L'avènement du monde digital à conduit à une augmentation considérable du nombre de données disponibles. Les méthodes de clustering sont un outil afin de regrouper les données en groupes partageant des caractéristique commune et ainsi réduire la dimension des données. De nombreuses méthodes de co-clustering permettent de grouper au même temps les lignes et les colonnes d'une matrice d'intéraction entre deux groupes d'individus/objets, on citera notamment la Latent Block Model (LBM, [BLZ16]), adapté à de nombreux type de données (données réelles [Lom12], catégorielles [Ker+15]...). Ce modèle est par ailleurs très efficace pour les données en grande dimension [Chr17]. Cependant, comme la plus part des méthodes de co-clustering, le LBM ne permet pas d'utiliser des données texutelles.

L'article The latent topic block model for the co-clustering of textual interaction data de *Laurent R. Bergé, Charles Bouveyron, Marco Corneli et Pierre Latouche* [Ber+19] vise donc à étendre le LBM pour prendre en compte de tels données.

Plusieurs modèles existent afin de traiter des données textuelles, parmi lesquels le latent semantic indexing (LSI, [Dee+90]), le probabilistic latent semantic analysis (pLSI), et le latent Dirichlet allocation (LDA, [BNJ03b]). Le LDA, en raison de sa popularité croissante en analyse statistique de texte, possède de nombreuses extension, comme le correlated topic model (CTM), qui aborde les corrélations entre les thèmes. Le LTBM se base sur le LDA afin d'adapter le LBM au données textuels. Ce modèle est à penser en parallèle du stochastic topic block model (STBM, [BLZ16]) à la différence que celui ci ne considère pas un ensemble disjoints d'individus et d'objets mais un seul ensemble d'individu qui intéragissent entre eux, plus adapté à des clustering sur des réseaux sociaux tel que Twitter.

2 Théorie du LTBM

2.1 Le modèle

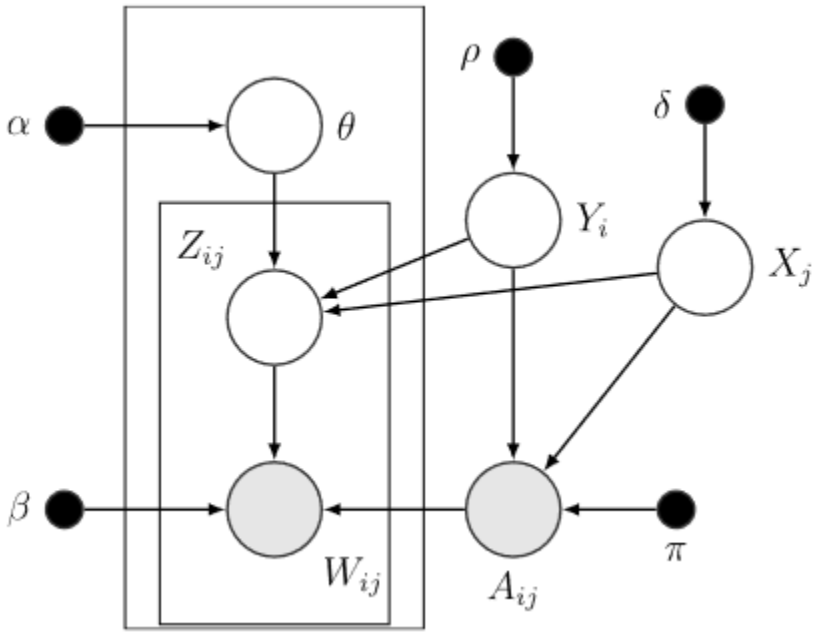

FIGURE 1 – Schéma du LTBM

Le LTBM cherche à modéliser l'intéraction entre deux ensembles distincts (*individus* et *objets*). On modélise cette intéarction par une matrice d'incidence $A \in \{0,1\}^{M \times P}$. Cependant contrairement au cas du LBM pour des données binaires, une interaction observé entre un individu i et un objet j (i.e. $A_{ij} = 1$) est enrichie par des données textuelles (on pourra par exemple penser à des commentaires de consommateurs sur des produits). Plus précisément si $A_{ij} = 1$, on dispose d'un ensemble de document $\{W_{ij}^d\,,\, d \in \{1,\dots,D_{ij}\}\}$, où $W_{ij}^d = (W_{ij}^{dn})_{n=1,\dots,N_{ij}^d}$ est un vecteur de mots pris dans un dictionnaire de longueur V.

2.1.1 Modélisation des interactions

Comme pour le LBM, on suppose que les individus (lignes de A) sont regroupé dans Q cluster. On introduit une matrice binaire Y de M lignes et Q colonnes tel que $Y_{iq} = 1$ ssi le i-ème individu appartient au q-ème cluster. On modèlise donc les lignes de Y par des vecteurs aléatoires indépendants tel que $\mathbb{P}(Y_{iq} = 1) = \rho_q$, où ρ est un vecteur du Q-simplexe (noté $\Delta(Q)$). De même on introduit une matrice X de taille $P \times L$ associé aux clusters colonnes, dont les lignes sont indépendantes et tel que $\mathbb{P}(X_{jl} = 1) = \delta_l$, où $\delta \in \Delta(L)$, L étant le nombre de cluster colonnes. On suppose de plus que les matrice alétoires Y et X sont indépendantes.

Comme pour le LBM, on suppose que la probabilité d'interactions entre un individu i et un objet j ne dépend que du cluster lignes de i (i.e. Y_i) et du cluster colone de j (i.e. X_j). Ainsi, conditionnellement à Y_i et X_j, A_{ij} suit une loi de Bernouilli : $\mathbb{P}(A_{ij} = 1 | Y_{iq} X_{jl} = 1) = \pi_{ql}$.

On notera π la matrice de probabilité d'interaction. On peut alors écrire la vraissemblance de la partie intéraction du modèle en conditionnant :

$$p(A, Y, X|\pi, \rho, \delta) = p(A|Y, X, \pi)\, p(Y|\rho)\, p(X|, \delta) \tag{1}$$

2.1.2 Modélisation des documents

Jusque ici notre modèle est en tout point similaire à un LBM binaire. Cependant nous n'avons pas encore pris en compte l'information apporté par les donées textuelles W.

L'article adopte le point de vu de la LDA [BNJ03a] qui considère que chaque mot dans un document suit un loi de mélange sur K topic latents avec K à déterminer (cf 2.4). En contraste avec la LDA, le LTBM fait le choix de considèrer que les topics latents n'est plus propre à chaque documents mais simplement aux cluster ligne de l'individu (i.e. Y_i) et du cluster colone de l'objet (i.e. X_j). On introduit donc Z_{ij}^{dn}, un vecteur aléatoire binaire de longueur K, tel que $Z_{ij}^{dnk} = 1$ ssi W_{ij}^{dn} est tiré selon le k-ème sujet. Ainsi, conditionnellement à Y_i et X_j, Z_{ij}^{dnk} suit une loi multinomiale :

$$Z_{ij}^{dnk}|Y_{iq}X_{jl}A_{ij} = 1 \sim \mathcal{M}(1, \theta_{ql} = (\theta_{ql1}, \ldots, \theta_{qlK})), \text{ pour un certain } \theta_{ql} \in \Delta(K)$$

De plus le LTBM suppose que conditionnellement à Y, X et θ les $Z_{ij}^{d1}, \ldots, Z_{ij}^{dN_{ij}^d}$ sont indépendants. Les proportions des sujets associès a chaque cluster θ_{ql} est également vue comme un vecteur aléatoire, suivant une distribution de Dirichlet de paramètres $\alpha = (\alpha_1, ..., \alpha_K)$. Enfin on suppose que conditionnelement à Z_{ij}^{dn}, W_{ij}^{dn} suit une loi multinomiale :

$$W_{ij}^{dn}|Z_{ij}^{dnk} = 1 \sim \mathcal{M}(1, \beta_k = (\beta_{k1}, \ldots, \beta_{kV})), \text{ pour un certain } \beta_k \in \Delta(V)$$

En conditionnant, on peut alors écrire la vraisemblance de la partie textuelle du modèle :

$$p(W, Z, \theta|A, Y, X, \beta, \alpha) = p(W|Z, A, \beta)\, p(Z|A, Y, X, \theta)\, p(\theta|\alpha) \tag{2}$$

2.2 Inférence

On décrit dans cette partie l'approche utilisé pour maximiser la vraisemblance du modèle. Afin de simplifier ce problème, l'article fait le choix de considérer $\alpha = (\alpha_1, \ldots, \alpha_K)$ fixé et on ne le considère donc pas comme un paramètre à optimiser. On cherche alors à maximiser la log-vraisemblance du modèle complet par rapport aux paramètres $(\pi, \rho, \delta, \beta)$ et (Y, X) :

$$\log p(W, A, Y, X|\pi, \rho, \delta, \beta) = \log p(W|A, Y, X, \beta) + \log p(A, Y, X|\pi, \rho, \delta) \tag{3}$$

Le terme $p(A, Y, X|\pi, \rho, \delta)$ est obtenu à partir de 2 et peut se calculé explicitement. Intéressons nous au terme $\log p(W|A, Y, X, \beta)$. Celui-ci, n'est pas calculable explicitement, on utilise donc une approche variationnelle pour l'estimer. En considérant une distribution q sur (Z, θ), on peut écrire la décomposition variationelle suivante :

$$\begin{aligned}\log p(W|A, Y, X, \beta) = &\int_\theta \sum_Z q(Z, \theta) \log \frac{p(W, Z, \theta|A, Y, X, \beta)}{q(Z, \theta)}\, d\theta \\ &- \int_\theta \sum_Z q(Z, \theta) \log \frac{p(Z, \theta|W, A, Y, X, \beta)}{q(Z, \theta)}\, d\theta\end{aligned}$$

On reconnaît alors que le deuxième terme est la divergence de Kullback-Leibler entre la distribution a-priori estimé $q(.)$ et la distribution réelle $p(\,.\,|W, A, Y, X, \beta)$ sur le couple (Z, θ). Celle ci est connue pour être positive ou nulle si et seulement si ces deux distributions sont égales. On a donc que :

$$\begin{aligned}\mathcal{L}(q(.)|A, Y, X, \beta) :&= \int_\theta \sum_Z q(Z, \theta) \log \frac{p(W, Z, \theta|A, Y, X, \beta)}{q(Z, \theta)}\, d\theta \quad &\leq \log p(W|A, Y, X, \beta)\\ &= \mathbb{E}_{q(Z,\theta)}\left[\log \frac{p(W, Z, \theta|A, Y, X, \beta)}{q(Z, \theta)}\right] &\leq \log p(W|A, Y, X, \beta)\end{aligned}$$

A nouveau la distribution $q(Z, \theta)$ n'a pas d'expression explicite, cependant en utlisant une approxiamtion par champ moyen (qui revient à supposer qu'il y a indépendance de la loi a-priori) on peut réécrire :

$$q(Z, \theta) = q(\theta)q(Z) = q(\theta) \prod_{i=1}^{M} \prod_{j=1}^{P} \prod_{d=1}^{D_{ij}} \prod_{n=1}^{N_{ij}^d} q(Z_{ij}^{dn}) \quad \text{(par indépendance des } Z_{ij}^{dn})$$

Dès lors, on va chercher à maximiser la borne inférieur suivant de la log-vraisemblance :

$$\log p(W, A, Y, X|\pi, \rho, \delta, \beta) \geq \mathcal{L}(q(.)|A, Y, X, \beta) + \log p(A, Y, X|\pi, \rho, \delta) \tag{4}$$

où le second terme est indépendant de l'approximation variationnelle utilisé. On obtient alors la procèdure d'estimation suivante :

(i) Y et X étant fixé, on optimise la borne inférieur en appliquant un algorithme VEM [Hat86] :
   - On optimise $\mathcal{L}$ par rapport à la distribution a-priori $q(Z, \theta)$ en fixant les paramètres $(\pi, \rho, \delta, \gamma)$ (E-step)
   - La distribution a priori étant fixé on optimise $\mathcal{L}$ par rapport aux paramètres $(\pi, \rho, \delta, \gamma)$ (M-step)

(ii) Les paramètres du modèle $(\pi, \rho, \delta, \gamma)$ étant fixé, on applique un algorithme de recherche gloutonne pour optimiser Y et X

La proposition 1 de l'article caractérise la loi a-piori $q(Z_{ij}^{dn})$ comme une loi mutlinomiale et en donne les paramètres optimaux toutes choses étant fixé par ailleurs. La proposition 2 fait de même pour la loi $q(\theta)$ (cette fois caractérisé par une loi de Dirichlet).

Armé de ces deux résultats on peu alors réaliser la E-step de l'algorithme VEM. Enfin la proposition 3 de l'article donne les paramètres $(\pi, \rho, \delta, \beta)$ optimaux et permet donc d'implémenter l'algorithme proposé.

2.3 Initialisation

Les algorithmes de maximisation de la vraisemblance sont connus pour être sensible à l'initialisation et ne fournissent pas de garantie de convergence vers un maximum global. Dans notre cas, nous avons besoin d'initialiser Y et X afin de pouvoir appliquer l'algorithme VEM. Plusieurs solution sont proposé

— Initialisation aléatoire de Y et X en faisant tourner plusieurs fois l'algorithme et conserver le meilleur résultat.

— application d'autre méthode de clustering (k-means..)

L'article présente également une méthode spécifique au problème considéré. Celle ci crée une matrice de similarité sur les lignes (resp. sur les colonnes) grâce au résultat d'une LDA sur l'ensemble de tout les documents, puis applique un clustering spectrale sur celle ci afin d'associer à chaque ligne (resp. colonne) un cluster initial.

2.4 Sélection de modèle

Jusqu'ici nous avons considéré les nombre de cluster Q et L, et K le nombre de sujet connu. Pour les optimiser il est nécessaire de disposer d'un critère de sélection de modèle. De plus le choix d'un critère de sélection doit être fait précautionneusement, l'erreur d'un critère asymptotique tel que le BIC étant doublé (pour M et P). De plus il faut renoncer à l'utilisation de critère non asymptotique en raison de la difficulté combinatoire ajouté par les variables latentes. L'article propose un critère de sélection de modèle reposant sur le critère ICL (*integrated classifiction lokelihood* [BCG00], et est donnée par la proposition 4 de l'article. Celui ci est tester par l'intermédiaire de données simulés ou Q, L et K sont connus.

3 Performance du modèle

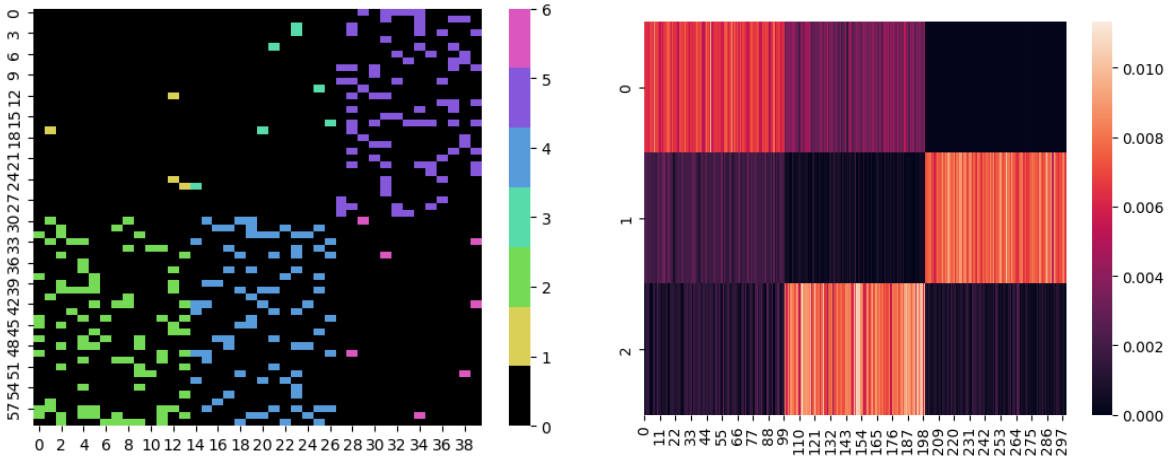

FIGURE 2 – Résultat de notre algorithme

Afin d'évaluer le modèle, nous avons souhaité le tester sur des jeux de données simulés ainsi que sur des jeux de données réelles. Nous avons alors chercher une implémentation sans parvenir à en trouver une. Nous avons alors tenté de l'implémenter en python. Malheureusement notre implémentation souffre du fait que le nombre de documents D_{ij} est propre à chaque interaction, de même que le nombre de mots N_{ij}^d est propre à chaque documents. En effet ces tailles hétérogènes nous ont conduit à utiliser des listes pour stocker certains variables du modèles (en partiucler les entrées textuelles W et ϕ, la variable stockant les paramètres

des lois $q(Z_{ij}^{dn})$), nous empêchant de vectoriser nombres d'opérations de l'algorithme. Bien que fonctionnel notre algorithme demande un temps de calcul trop long pour pouvoir mener une étude sur les performance du modèle.

Le co-clustering est généré sur la figure de droite, pour chaque objet, on définit la répartition entre les topic (le β). Pour chaque interaction on tire une centaine de mots du topic en ajoutant un bruit (40% des mots ne proviennent pas du topic principal). On obtient finalement le co-clustering de la figure de droit on remarque bien la présence de 3 clusters avec la couleur indiquant le topic dont ont été tiré les textes.

On notera tout de même que l'article présente des résultats encourageant pour ce modèle, celui ci ayant de meilleurs performances qu'un LMB ou qu'une LDA. Il semble également s'adapter sans difficultés aux cas des matrices d'incidences *sparse*, les auteurs le testant sur un jeu de données ou plus de 98% de la matrice d'incidence A est nulle avec des résultats concluants.

4 Conclusion

L'article présente une nouvelle approche de co-clustering qui intègre à la fois des informations d'incidence et textuelles, se distinguant ainsi des méthodes telles que LBM qui se basent uniquement sur la structure de la matrice d'incidence. Le modèle génératif nouvellement proposé, LTBM, est détaillé, accompagné d'une procédure d'estimation visant à ajuster le modèle aux données. Une comparaison avec le LBM démontre la pertinence du LTBM. De plus, l'évolutivité de l'algorithme d'estimation est soulignée, le rendant adapté à des ensembles de données volumineux. Enfin, l'approche prend en considération des matrices d'incidence sparse.

Pour les futures recherches, des extensions possibles sont suggérées. La sélection du modèle nécessite actuellement le calcul de l'ICL pour toutes les valeurs de Q, L et K dans une plage donnée, ce qui peut conduire à un grand nombre de modèles à tester. Des alternatives reposant sur des schémas de recherche gourmands pourraient être explorées, par exemple en étendant l'algorithme de recherche avant au LTBM. L'idée serait d'effectuer des déplacements sur la grille en Q, L, K où les déplacements acceptés induisent la plus forte augmentation du critère ICL. Une recherche non exhaustive basée sur des algorithmes génétiques est également mentionnée. Enfin, il serait intéressant de dériver une mesure pour évaluer l'importance des documents et de la matrice binaire dans la partition des lignes/colonnes proposée par l'algorithme.

5 Répartition du travail de Rédaction

Bien que notre travail ai été effectué en parallèle, les principaux contributeurs des différentes parties sont les suivants :

— Pierre Cavalier : Introduction, Conclusion, Partie 3.

— Virgile Bertrand : Abstract, Partie 2.

Références

[Hat86] Richard J. Hathaway. « Another interpretation of the EM algorithm for mixture distributions ». In : *Statistics Probability Letters* 4.2 (1986), p. 53-56. ISSN : 0167-7152. DOI : https://doi.org/10.1016/0167-7152(86)90016-7. URL : https://www.sciencedirect.com/science/article/pii/0167715286900167.

[Dee+90] Scott Deerwester et al. « Indexing by Latent Semantic Analysis ». In : *Journal of the American Society for Information Science* 41.6 (1990), p. 391.

[BCG00] C. Biernacki, G. Celeux et G. Govaert. « Assessing a mixture model for clustering with the integrated completed likelihood ». In : *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22.7 (2000), p. 719-725. DOI : 10.1109/34.865189.

[BNJ03a] David M. Blei, Andrew Y. Ng et Michael I. Jordan. « Latent Dirichlet allocation ». In : *Journal of Machine Learning Research* 3.4-5 (2003). Cited by : 28086, p. 993-1022. URL : https://www.scopus.com/inward/record.uri?eid=2-s2.0-0141607824&partnerID=40&md5=505ce8839ae28d1cb56a7ff91bd0ad2d.

[BNJ03b] David M. Blei, Andrew Y. Ng et Michael I. Jordan. « Latent dirichlet allocation ». In : *J. Mach. Learn. Res.* 3 (2003), p. 993-1022. ISSN : 1532-4435. DOI : http://dx.doi.org/10.1162/jmlr.2003.3.4-5.993. URL : http://portal.acm.org/citation.cfm?id=944937.

[Lom12] Aurore Lomet. « Sélection de modèle pour la classification croisée de données continues ». Thèse de doct. Compiègne, 2012.

[Ker+15] Christine Keribin et al. « Estimation and selection for the latent block model on categorical data ». In : *Statistics and Computing* 25.6 (nov. 2015), p. 1201-1216. ISSN : 1573-1375. DOI : 10.1007/s11222-014-9472-2. URL : https://doi.org/10.1007/s11222-014-9472-2.

[BLZ16] Charles Bouveyron, P Latouche et Rawya Zreik. « The stochastic topic block model for the clustering of vertices in networks with textual edges ». In : *Statistics and Computing* (2016). DOI : 10.1007/s11222-016-9713-7. URL : https://hal.science/hal-01299161.

[Ber+19] L. R. Bergé et al. « The latent topic block model for the co-clustering of textual interaction data ». In : *Computational Statistics & Data Analysis* (2019).

[Chr17] Valérie Robert Christine Keribin Gilles Celeux. « he Latent Block Model : a useful model for high dimensional data ». In : *ISI 2017 - 61st world statistics congress* (Jul 2017).