Algorithme de type “ Water-Filling ” maximisant le volume de matrices rectangulaires au-delà du rang

Claude Petit, Aline Roumy, Thomas Maugey

Algorithme de type «Water-Filling» maximisant le volume de matrices rectangulaires au-delà du rang

Claude PETIT Aline ROUMY Thomas MAUGEY

INRIA, Rennes, France

{claude.petit,aline.roumy,thomas.maugey}@inria.fr

Résumé – Cet article traite de maximisation du volume de sous-matrices extraites d'une matrice de données rectangulaire, lorsque le nombre de colonnes extraites est supérieur au rang. L'objectif est d'opérer une compression de la matrice de données en extrayant un échantillon représentatif des colonnes, tout en préservant la diversité de l'information contenue dans les données. Nous présentons un algorithme en un coup, non glouton, s'inspirant des techniques de «Water-Filling» traditionnellement dédiées aux stratégies d'égalisation dans les communications multicanaux. Le principe de l'algorithme repose sur une relaxation continue du programme d'optimisation initial, qui admet une solution simple conduisant à égaliser les valeurs propres de la matrice de covariance associée à la matrice de données. Les simulations montrent que l'algorithme proposé est plus performant que les méthodes d'échantillonnage basées sur des processus ponctuels déterminantaux (PPD) et atteint des performances similaires à celles des meilleurs algorithmes connus, pour une complexité moindre.

Abstract – In this paper, we propose an algorithm to extract a maximum volume rectangular submatrix of size greater than the dataset matrix rank. The objective is to compress the dataset by sampling the columns, while preserving the diversity of the underlying information. We use a continuous relaxation which admits a simple and closed form solution: the nonzero singular values of the extracted matrix must be equal. Our algorithm is inspired by a water-filling technique, traditionally dedicated to equalization strategies in communication channels. Simulations show that the proposed algorithm performs better than sampling methods based on determinantal point processes (PPDs) and achieves similar performance as the best known algorithm, with a lower complexity.

1 Introduction

Le volume d'une matrice de données est un paramètre important intervenant dans de nombreuses applications relatives aux sciences des données ou au traitement du signal en grande dimension : l'échantillonnage, la compression, ou certains algorithmes de fouille de données nécessitent la sélection d'un sous-ensemble représentatif de la base initiale, conservant le maximum d'information [1, 9]. Un critère fréquent de choix du sous-ensemble est la maximisation du volume de la sous-matrice extraite.

Les méthodes existantes d'extraction de sous matrices de volume maximal sont soit déterministes et optimisent le conditionnement des matrices ou construisent des approximations de faible rang de matrices carrées [6], soit aléatoires et sont alors basées sur l'échantillonnage de processus ponctuels déterminantaux (PPD «volume sampling») [7]. La plupart de ces méthodes sont construites de manière à extraire des sous-matrices de dimension plus faible que le rang de la matrice initiale et sont donc pénalisées pour extraire des sous matrices au-delà du rang, alors qu'il est fréquent de trouver des tableaux de données dont le nombre d'individus est largement supérieur au nombre de variables observées. À notre connaissance, il n'existe qu'un unique algorithme maximisant efficacement le volume d'une matrice rectangulaire de taille supérieure à son rang [10], mais cet algorithme est itératif.

Dans cet article, nous proposons une méthode de compression basée sur un échantillonnage des colonnes [9, 12, 1] dont le principe s'inspire d'une technique de «Water-Filling» [4]. Notre algorithme est en un coup et repose sur l'égalisation des valeurs singulières de la matrice extraite.

La suite de l'article s'articule comme suit : la formulation du problème et son analyse sont détaillées dans la section 2. La section 3 présente l'algorithme et sa complexité tandis que les performances et comparaisons avec les autres méthodes sont exposées dans la partie 4.

2 Formulation et analyse du problème

La base de données de cardinal n est modélisée par une matrice $A \in \mathbb{R}^{d\times n}$. Les vecteurs colonnes représentent les éléments de la base ou «items», les lignes représentent les caractéristiques décrivant les items via un espace latent isomorphe à $\mathbb{R}^d$. Pour permettre aux caractéristiques d'être comparables, nous supposons que chaque colonne de A est normalisée. Le problème est alors de sélectionner m colonnes ($d \leq m \leq n$) de la matrice de données (supposée être de rang d, inférieur à n et m), formant ainsi une sous-matrice rectangulaire $B \in \mathbb{R}^{d\times m}$, elle aussi de rang d. Le critère utilisé pour sélectionner la sous-matrice est la maximisation du volume de B.

Dans la suite, on note b_i les colonnes de B et B_k, $d \leq k \leq m$, pour une matrice extraite formée de k colonnes. $\Sigma = BB'$ est la matrice de covariance et $G = B'B$ la matrice de Gram de B. La notation $B \subset A$ signifie que les colonnes de B sont incluses dans l'ensemble des colonnes de A. Tr est l'opérateur trace et Sp indique le spectre. $||.||_p$ preprésente la norme p pour $p = 1, 2$.

Nous utilisons la définition de Ben-Israël pour le volume d'une matrice rectangulaire B de rang d [2, 12] :

$$\text{vol}(B) = \prod_{i=1}^{d} \sigma_i = \sqrt{\prod_{i=1}^{d} \lambda_i}, \tag{1}$$

où $\sigma_1 \geq \sigma_2 \geq ... \geq \sigma_d > 0$ sont les d valeurs singulières strictement positives de B et $\lambda_i = \sigma_i^2$ sont les valeurs propres

strictement positives de $\Sigma = BB'$ ou $G = B'B$. Cette définition généralise la définition du volume d'une matrice carrée par son déterminant. En particulier, comme $\Sigma = BB'$ est une matrice carrée de rang d, symétrique définie positive (SDP),

$$\text{vol}(B) = \sqrt{\det(BB')}. \tag{2}$$

Deux problèmes d'optimisation : de la maximisation du volume à l'égalisation du spectre. Notre objectif peut se formaliser par le problème d'optimisation suivant :

Problème 1 (Maximisation discrète du volume)**.** *Déterminer*

$$\max_{B\subset A, \forall i\ ||b_i||=1} \text{vol}(B)^2 = \max_{B\subset A, \forall i\ ||b_i||=1} \det(BB') \tag{3}$$

l'égalité provenant de (2).

La résolution du Pb. 1 est NP-difficile [12]. Pour contourner cette difficulté, nous modifions le problème en ne supposant plus que B est une sous-matrice de A.

Problème 2 (Relaxation continue du Pb. 1)**.** *Déterminer*

$$\max_{B:\forall i\ ||b_i||=1} \det(BB') = \max_{B:\text{Tr}(BB')=m} \det(BB') \tag{4}$$

Démonstration. (Résolution du Pb. 2) : nous résolvons d'abord le sous-problème correspondant au membre de droite de (4), puis nous présentons les relations entre les deux sous-problèmes avant d'expliciter les solutions du membre de gauche. $\det(BB') = \prod_{i=1}^{d} \lambda_i$ et $\text{Tr}(BB') = \sum_{i=1}^{d} \lambda_i$. Le maximum d'un produit de nombres positifs sous contrainte de somme fixée est exactement le cas d'égalité dans l'inégalité arithmético-géométrique [3]. L'égalité est atteinte si, et seulement si, tous les λ_i sont égaux et leur valeur commune est alors

$$\lambda = (\sum_{i=1}^{d} \lambda_i)/d = m/d. \tag{5}$$

Le volume maximum est $(m/d)^d$ et est atteint si, et seulement si, $BB' = \frac{m}{d} I_d$. Enfin, il existe au moins une matrice B, telle que $BB' = \frac{m}{d} I_d$, par exemple, $B = (\sqrt{\frac{m}{d}} I_d, 0)$. Le maximum est donc atteint.

Il s'agit ensuite de caractériser les matrices B solutions du deuxième sous-problème (à gauche) du Pb. 2 et de montrer l'égalité (4).

Tout d'abord, $\max_{B:\forall i\ ||b_i||=1} \det(BB') \leq \max_{B:\text{Tr}(BB')=m} \det(BB')$ car

$$\forall i, ||b_i|| = 1 \Rightarrow \text{Tr}(BB') = \text{Tr}(B'B) = \sum_i ||b_i||^2 = m.$$

La réciproque est fausse. En effet, $\text{Tr}(BB') = m \not\Rightarrow \forall i,\ ||b_i|| = 1$. Aussi, pour montrer l'égalité dans (4), nous devons montrer qu'il existe au moins une matrice B vérifiant $BB' = \lambda I_d$ dont les colonnes sont normées. Ce résultat est une conséquence du théorème de Schur-Horn [8, Th.B.2. p. 302] qui donne une condition nécessaire et suffisante d'existence d'une matrice symétrique G de spectre $\lambda_1 \geq ... \geq \lambda_m$ et de coefficients diagonaux $c_1 \geq ... \geq c_m$. G existe si, et seulement si,

$$\sum_{i=1}^{s} c_i \leq \sum_{i=1}^{s} \lambda_i, \forall s < m \text{ et } \sum_{i=1}^{m} c_i = \sum_{i=1}^{m} \lambda_i. \tag{6}$$

Posons $\lambda_1 = ... = \lambda_d = m/d$, $\lambda_{d+1} = ... = \lambda_m = 0$ et $c_i = 1$ pour tout $i = 1, .., m$. D'après le théorème de Horn, il existe une matrice symétrique réelle $G \in \mathbb{R}^{m\times m}$ ayant pour spectre les $(\lambda_i)_i$ et pour termes diagonaux les $(c_i)_i$. G étant symétrique, elle se décompose en $G = B'B$ avec $B \in \mathbb{R}^{d\times m}$. B a nécessairement pour valeur singulière unique $\sqrt{\lambda} = \sqrt{m/d}$ de multiplicité d, et des colonnes vérifiant

$$||b_i||^2 = c_i = 1,\ \forall i = 1, .., d. \tag{7}$$

Par ailleurs, $BB' \in \mathbb{R}^{d\times d}$ a également pour unique valeur propre λ, de multiplicité d, ce qui implique que $BB' = \lambda I_d$ et B possède des colonnes normées. Le problème à gauche de (2) possède donc au moins une solution et son maximum $(m/d)^d$ est égal au problème de droite de (2). La Fig. 1 illustre les relations entre les différents ensembles matriciels en jeu. □

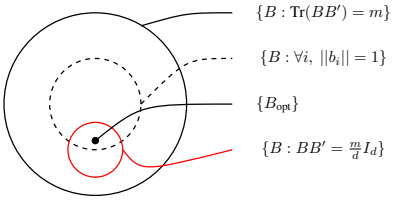

FIGURE 1 : L'espace de recherche des deux sous-problèmes du Pb. 2 ne coïncident pas : en noir pointillé pour le problème à gauche et en trait plein pour le problème de droite. Les solutions de ce dernier sont en rouge. Il existe une matrice notée B_{opt} de vecteurs colonne normés, candidate du problème de gauche. Les deux sous-problèmes atteignent donc la même valeur maximale.

Le Pb. 2, relaxation continue du Pb. 1, fournit une intuition pour résoudre ce problème discret : la matrice de covariance BB' doit être aussi «proche» que possible de la matrice scalaire λI_d. En d'autres termes, nous devons chercher à uniformiser l'ensemble des valeurs propres.

3 Algorithme proposé

Le problème d'égalisation des valeurs propres peut se formaliser de la façon suivante :

Problème 3 (Minimisation de la largeur du spectre)**.** *Déterminer*

$$\underset{B\subset A:\Sigma=BB'}{\text{argmin}} \sum_{i=1}^{d} \left(\lambda_i(\Sigma) - \overline{\lambda}\right)^2, \tag{8}$$

où $\overline{\lambda}$ valeur commune (inconnue) à atteindre.

Comme dans la section précédente, nous ne résolvons pas directement le Pb. 3, mais une relaxation continue dont la solution nous conduit à une méthode discrète d'égalisation.

Principe de l'algorithme : contraindre la largeur du spectre. L'étape d'initialisation sélectionne d colonnes de A et produit deux matrices B_d et $\Sigma_d = B_d B_d'$ de taille $d \times d$ en utilisant n'importe quel algorithme de maximisation du volume d'une matrice carrée, par exemple une recherche exhaustive si la dimension d est suffisamment petite, ou bien l'algorithme MAXVOL de [6].

Dans la seconde étape, $m - d$ colonnes sont concaténées à B_d en une seule passe pour former les matrices B_m et $\Sigma_m = B_m B_m'$. Cette concaténation modifie les valeurs propres initiales $\lambda_1(\Sigma_d), ..., \lambda_d(\Sigma_d)$. L'objectif de l'algorithme est de sélectionner les $m - d$ colonnes pour égaliser au mieux les valeurs propres finales $\lambda_1(\Sigma_m), ..., \lambda_d(\Sigma_m)$.

L'effet sur B_d de la concaténation d'une ou plusieurs colonnes peut-être caractérisé par Σ_d et Σ_m, en utilisant la décomposition en opérateur de rang 1 :

$$\Sigma_m = B_m B'_m = \sum_{i=1}^{m} b_i b'_i = \Sigma_d + \sum_{i=d+1}^{m} b_i b'_i. \qquad (9)$$

Concaténer $m-d$ colonnes à B_d est ainsi équivalent à ajouter $m-d$ opérateurs de rang 1 à Σ_d. Notons Σ_k la matrice de covariance de B_k, $d \leq k < m$. D'après [5], si u_i est un vecteur propre normé de $\lambda_i(\Sigma_k)$, alors pour toute colonne b,

$$\lambda_i(\Sigma_k) \leq \lambda_i(\Sigma_k + bb') \leq \lambda_i(\Sigma_k) + ||b||^2 \qquad (10)$$

avec égalité dans la dernière inégalité si, et seulement si, b est un vecteur propre de $\lambda_i(\Sigma_k)$, auquel cas $(u'_i b)^2 = ||b||^2 = 1$. Par ailleurs, $\forall i = 1, ..., d$,

$$\lambda_i(\Sigma_k + bb') = \lambda_i(\Sigma_k) + \mu_{ik}, \text{ où } \mu_{ik} = \frac{(u'_i b)(v'_i b)}{u'_i v_i} \qquad (11)$$

est une fonction positive, continue de la variable b, qui tend vers 1 quand b tend vers u_i (v_i vecteur propre normé associé à $\lambda_i(\Sigma_k + bb')$, $u'_i v_i \neq 0$). Les valeurs propres de Σ_m sont obtenues à partir de celles de Σ_d en ajoutant les $m-d$ contributions μ_{ik} de chaque perturbation de rang 1 :

$$\lambda_i(\Sigma_m) = \lambda_i(\Sigma_d) + \epsilon_i, \ \epsilon_i = \textstyle\sum_k \mu_{ik}, \ \sum_i \epsilon_i = m - d. \qquad (12)$$

Comme le volume maximal est obtenu pour $\Sigma_m = \lambda I_d$, une stratégie est de sélectionner les colonnes pour réduire la largeur du spectre, c.-à-d. contraindre ϵ_i de telle sorte que pour tout i, $\lambda_i(\Sigma_d) + \epsilon_i$ soit aussi proche que possible d'une valeur commune $\overline{\lambda}$. La relaxation continue du Pb. 3 (B n'est plus considéré comme une sous-matrice de A) conduit à un problème convexe possédant une unique solution. Les conditions de Karush-Kuhn-Tucker donnent : $\forall i = 1, ..., d$,

$$\epsilon_i = 0 \iff \overline{\lambda} > \lambda_i(\Sigma_d), \qquad (13)$$

$$\epsilon_i > 0 \iff \epsilon_i = \overline{\lambda} - \lambda_i(\Sigma_d). \qquad (14)$$

Par analogie avec la technique de «water-filling», la quantité d'eau optimale est $m - d = \sum \epsilon_i$. La quantité d'eau versée W est une fonction croissante de la hauteur d'eau ν, continue et linéaire par morceaux, qui s'écrit

$$W(\nu) = \sum_{i=1}^{d} (\nu - \lambda_i(\Sigma_d))_+ . \qquad (15)$$

$\overline{\lambda}$ reste à déterminer : le niveau d'eau optimal est atteint pour $\nu = \overline{\lambda}$ tel que $W(\nu) = m-d$. On évalue alors numériquement $\overline{\lambda} = W^{-1}(m-d)$ (la figure 2 illustre la situation).

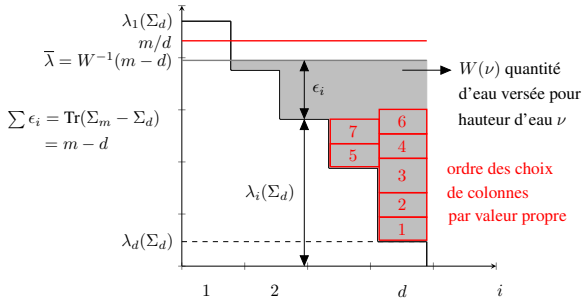

FIGURE 2 : Technique de «water-filling». La surface grise vaut $m-d$, la surface blanche vaut d, la somme des deux vaut m, égale à la surface du rectangle de hauteur m/d et largeur d. $W(\nu)$ est la quantité d'eau versée (en gris), fonction de la hauteur d'eau ν.

Une discrétisation du «water-filling» : revenons au Pb. 3, de nature discrète. L'algorithme de «water-filling» optimise l'allocation de la puissance de transmission dans les problèmes de communication multicanaux [4]. L'optimum est atteint lorsque la puissance totale est répartie équitablement entre chaque canal. Nous utilisons cette stratégie pour approcher la ou les solutions du Pb. 3. Trois opérations sont effectuées séquentiellement pour déterminer quelles colonnes sont affectées à quelle valeur propre et contribuerons à son égalisation :

1. Estimation de $\overline{\lambda}$ et de la valeur de chaque ϵ_i : la valeur de $\overline{\lambda} = W^{-1}(m-d)$ est déterminée numériquement, puis, pour tout i, on évalue $\epsilon_i = (\overline{\lambda} - \lambda_i(\Sigma_d))_+$.

2. Calcul et classement des produits scalaires $(u'_i b)^2$: l'équation (11) montre que la contribution ϵ_i à $\lambda_i(\Sigma_d)$ est corrélée à chacun des produits scalaires $(u'_i b)^2$ du vecteur propre u_i associé à $\lambda_i(\Sigma_d)$, avec les $n-d$ colonnes b non encore sélectionnées. Nous calculons $S = Q'A = (u'_i b_j)_{ij}$ et pour chaque u_i, nous classons les $n-d$ valeurs $(u'_i b_j)^2$ par ordre décroissant, afin de disposer des indices des colonnes les plus corrélées à u_i. La matrice R des rangs a pour ligne i les indices des $(u'_i b_j)^2$ classés du plus grand au plus petit.

3. Calcul du nombre de colonnes c_i affectées à $\lambda_i(\Sigma_d)$: dans (11), v_i est inconnu et l'équation (10) est une inégalité, de sorte que la contribution exacte des $(u'_i b_j)^2$, $j = d+1, ..., m$, à ϵ_i est inconnue. Plusieurs stratégies sont possibles. L'une d'elles a été développée dans un article en cours de soumission [11] : le nombre de colonnes est proportionnel à ϵ_i (ce qui revient à remplacer $(u'_i b_j)^2$ par 1). La méthode proposée ici part de la plus petite valeur propre $\lambda_d(\Sigma_d)$ et choisit les indices par ordre décroissant des corrélations $(u'_i b)^2$. Dès que la somme des contributions à $\lambda_i(\Sigma_d)$ dépasse l'écart $\lambda_{i-1}(\Sigma_d) - \lambda_i(\Sigma_d)$, les colonnes relatives à $\lambda_{i-1}(\Sigma_d)$ sont sélectionnées avec celles relatives à $\lambda_i(\Sigma_d), ..., \lambda_d(\Sigma_d)$, une par une, afin d'uniformiser au mieux les quantités $\lambda_i(\Sigma_d) + \epsilon_i$ (cf. Fig. 2).

Implementation et complexité :

Algorithme 1 : WaterMaxVol

Données : $A \in \mathbb{R}^{d \times n}, m \in [\![d, n]\!]$
Initialisation : $B_d \in \mathbb{R}^{d \times d}$ (par MaxVol)
$J =$ indices des colonnes de B ; $I = [\![d, n]\!] \setminus J$
Diagonaliser $\Sigma_d = B_d B'_d = Q'DQ$,
$Q = (u_1, .., u_d), D = \mathrm{diag}(\lambda_1, .., \lambda_d)$
$S = Q'A$, R matrice des rangs de $S^{\cdot 2} = \left((u'_i b_j)^2\right)_{ij}$
Évaluer $\overline{\lambda} = W^{-1}(m-d)$ puis $\epsilon = (\epsilon_1, ..., \epsilon_d)$
pour i *de* d *à* 1 **faire**
  $\psi = 0$ (accumule les produits scalaires $(u'_i b)^2$)
  pour j *de* d *à* i **faire**
    $\psi = \psi + S^{\cdot 2}(j, R(j, T_j))$
    T_j indice de la prochaine colonne disponible
    $J = J \cup T_j$; $I = I \setminus T_j$
    stop si $|J| = m - d$ ou $\psi > \lambda_i(\Sigma_d) - \lambda_{i+1}(\Sigma_d)$
  fin
fin
Résultat : $B = A_J \in \mathbb{R}^{d \times m}$

Complexité de l'algorithme, en nombre d'opérations : l'étape d'initialisation a la complexité de la méthode utilisée, par exemple $\mathcal{O}(nd^2)$ pour MAXVOL ; diagonalisation de B_d : $\mathcal{O}(d^3)$; évaluation de $S = Q'A$: $\mathcal{O}(nd^2)$; classement des d lignes de S : $\mathcal{O}(dn \ln n)$; évaluation de ϵ : $\mathcal{O}(n)$; chaque passage dans la double boucle sélectionne une colonne via un nombre constant d'opérations ; la complexité de la boucle est donc $\mathcal{O}(m-d)$.

La complexité globale de l'algorithme est $\mathcal{O}(nd^2)$ opérations, où d est supposé très inférieur à n. En utilisant des algorithmes efficaces de produit matriciel, il est possible d'abaisser chaque puissance cubique en $2+\delta$, avec $\delta \in]0,1[$.

La méthode traditionnelle d'échantillonnage de PPD est basée sur un algorithme en trois étapes, dont la première et la dernière sont les plus coûteuses : l'initialisation consiste à diagonaliser la matrice noyau $L = A'A$, pour une complexité en $\mathcal{O}(n^3)$ opérations. La troisième étape est une décomposition de Gram-Schmidt dont le coût est $\mathcal{O}(nm^3)$ pour les méthodes classiques et $\mathcal{O}(nm^2)$ pour les meilleurs algorithmes connus.

La dernière comparaison est effectuée avec l'algorithme RECT_MAXVOL dédié aux matrices rectangulaires [10]. L'initialisation a pour complexité $\mathcal{O}(nd^2)$, puis l'algorithme se déroule en $\mathcal{O}(nm^2)$ opérations. Pour chacune de ces deux étapes, comme d est supposé plus petit que n, notre méthode est moins complexe.

4 Résultats expérimentaux

La Figure 3 ci-après compare les volumes atteints par différentes méthodes. Tout d'abord, nous considérons une recherche exhaustive (en vert). Cette recherche, du fait de sa complexité, ne peut être mise en place en général, et est présentée ici de manière à comparer les méthodes existantes à ce maximum. La deuxième approche (en gris) est une sélection aléatoire selon une loi uniforme. Cette approche est extrêmement simple, mais extrait des matrices dont le volume est faible.

Nous considérons ensuite des sélections aléatoires basées sur les k-PPD. Ces approches permettent d'extraire un nombre de vecteurs plus petit que le rang d de la matrice A. Aussi, de manière à pourvoir extraire au delà du rang, nous considérons deux noyaux. Le premier est défini par la matrice de Gram modifiée $G = AA' + \delta I$, où $\delta > 0$ est un réel rendant G définie positive. Cette approche (en bleu clair) est notée DPPGram dans la figure. Le second noyau est défini à l'aide d'une fonction non linéaire $\phi : \mathbb{R}^d \longrightarrow \mathbb{R}^n$ plongeant les vecteurs colonnes de A dans un nouvel espace de description, de dimension $n > d$, de telle sorte que la famille $(\phi(b_i); i = 1, ..., n)$ soit linéairement indépendante. Une matrice noyau classique est alors déterminée par le produit scalaire euclidien $G_{ij} = \langle \phi(b_i), \phi(b_j) \rangle = \cos(b_i, b_j)$, pour chaque paire de colonnes b_i, b_j. Cette approche (en bleu foncé) est notée DPPKer dans la figure. Ces deux approches dépassent l'approche uniforme, mais le volume extrait demeure éloigné par rapport à la recherche exhaustive.

La dernière méthode concurrente est l'approche RectMaxVol (en noir). Le volume extrait est très proche de celui atteint par une recherche exhaustive. L'approche demeure complexe, comme expliqué au paragraphe précédent, notamment du fait de l'approche glouton et de la résolution d'un problème d'optimisation à chaque itération.

Enfin, la méthode proposée ici (WM2, en jaune) et la variante proposée dans [11] (WM1, en rouge) obtiennent une matrice extraite dont le volume est similaire et dont les performances sont proches de celles de RectMaxVol et de la recherche exhaustive. L'écart de performances reste très significatif par rapport aux méthodes basées sur les PPD pour la première moitié des valeurs de m. Lorsque m est proche de n, toutes les méthodes obtiennent approximativement les mêmes résultats, car les choix possibles de colonnes sont restreints.

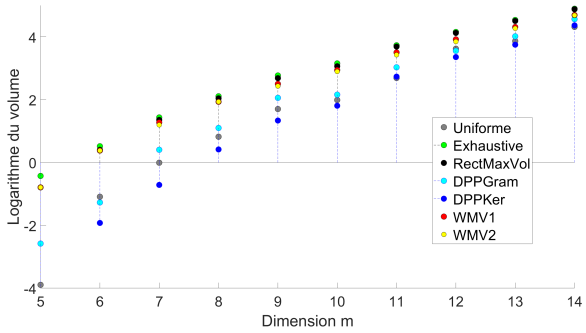

FIGURE 3 : Comparaison de 7 algorithmes de maximisation du volume, en fonction de m variant de 5 à 14. $d = 5$, $n = 20$. Chaque point représente la moyenne, pour 10 échantillons indépendants, du logarithme du volume. $A \in \mathbb{R}^{d \times n}$ à coefs gaussiens i.i.d. $\mathcal{N}(0,1)$.

5 Conclusion

Dans cet article, nous avons proposé une méthode de maximisation du volume d'une matrice rectangulaire dont le nombre de colonnes est supérieur à son rang. L'algorithme s'inspire des techniques de «water-filling» et repose sur la relaxation continue du problème d'optimisation initial. Celle-ci admet une solution simple conduisant à égaliser les valeurs propres de la matrice de covariance associée. Les simulations montrent que cet algorithme a de meilleures performances que les PPD et des performances proches du meilleur algorithme connu pour une complexité inférieure.

Références

[1] Ayoub BELHADJI, Rémi BARDENET et Pierre CHAINAIS : A determinantal point process for column subset selection. *Journal of Machine Learning Research*, 21(197):1–62, 2020.

[2] Adi BEN-ISRAEL : A volume associated with m x n matrices. *Linear Algebra and its Applications*, 167:87–111, 04 1992.

[3] D.P. BERTSEKAS : *Nonlinear Programming*. Athena, 1999.

[4] Thomas M. COVER et Joy A. THOMAS : *Elements of Information Theory 2nd Edition*. Wiley-Interscience, 2006.

[5] Gene H. GOLUB et Charles F. VAN LOAN : *Matrix Computations*. The Johns Hopkins University Press, 1996.

[6] S. A. GOREINOV et COLL. : How to find a good submatrix. *In Matrix Methods*. 2010.

[7] Alex KULESZA et Ben TASKAR : *Determinantal Point Processes for Machine Learning*. 2012.

[8] Albert W. MARSHALL, Ingram OLKIN et Barry C. ARNOLD : *Inequalities*, volume 143. Springer, second édition, 2011.

[9] Thomas MAUGEY et Laura TONI : Large database compression based on perceived information. *IEEE Signal Processing Letters*, 27:1735–1739, 2020.

[10] Aleksandr MIKHALEV et I.V. OSELEDETS : Rectangular maximum-volume submatrices. *Linear Algebra Appl.*, 2017.

[11] Claude PETIT, Aline ROUMY et Thomas MAUGEY : A water-filling algorithm maximizing the volume of submatrices above the rank. *submitted to EUSIPCO*.

[12] Ali ÇIVRIL et Malik MAGDON-ISMAIL : On selecting a maximum volume sub-matrix of a matrix and related problems. *Theor. Comput. Sci.*, (47), 2009.