Processus ponctuels déterminantaux pour les coresets

Nicolas Tremblay & Simon Barthelmé & Pierre-Olivier Amblard

CNRS, GIPSA-lab, Grenoble-INP, Univ. Grenoble-Alpes, Grenoble
E-mail: prenom.nom@gipsa-lab.fr

Résumé. Face à une grande masse de données dont on veut apprendre certains paramètres d'un modèle sous-jacent, une solution possible pour accélérer l'apprentissage est l'échantillonnage: garder uniquement une partie des données et estimer la solution du problème initial en apprenant uniquement sur ce petit sous-ensemble. Les *coresets* sont de tels sous-ensembles qui, étant donnée une certaine tâche d'apprentissage, garantissent que l'estimation obtenue sur ce sous-ensemble est une bonne approximation de ce qu'on aurait obtenu sur le grand jeu initial de données, à une erreur relative près. Une des directions de l'état de l'art pour générer de tels *coresets* est d'échantillonner aléatoirement des éléments du jeu initial de manière iid, via une densité de probabilité particulièrement bien choisie (proportionnelle à une métrique appelée "sensitivité"). Les sous-ensembles obtenus par échantillonnage indépendent souffrent néanmoins de redondance: nous explorons ici comment les processus ponctuels déterminantaux, grâce entre autres à leur capacité à conserver la diversité d'un jeu de données, peuvent apporter des améliorations aux théorèmes iid existants.

Mots-clés. Processus ponctuels déterminantaux, coresets, apprentissage

Abstract. When faced with a data set too large to be processed all at once, an obvious solution is to retain only part of it. In practice this takes a wide variety of different forms, and among them "coresets" are especially appealing. A coreset is a (small) weighted sample of the original data that comes with the following guarantee: a cost function can be evaluated on the smaller set instead of the larger one, with low relative error. For some classes of problems, and via a careful choice of sampling distribution (based on the so-called "sensitivity" metric), iid random sampling has turned to be one of the most successful methods for building coresets efficiently. However, independent samples are sometimes overly redundant, and one could hope that enforcing diversity would lead to better performance. The difficulty lies in proving coreset properties in non-iid samples. We show that the coreset property holds for samples formed with determinantal point processes (DPP). DPPs are interesting because they are a rare example of repulsive point processes with tractable theoretical properties, enabling us to prove general coreset theorems.

Keywords. Determinantal point processes, coresets, learning

1 Introduction

Considérons une tâche d'apprentissage à résoudre sur une grande masse de données. Une des solutions possibles pour accélérer le processus d'apprentissage quand les données sont trop volumineuses est l'échantillonnage: garder uniquement une partie de l'information, jeter le reste, et estimer la solution du problème initial en apprenant seulement à partir de l'information conservée. Pour fixer les idées, supposons que $\mathcal{X} = \{x_1, x_2, \ldots, x_n\}$ est un grand ensemble de n points dans $\mathbb{R}^d$: c'est le corpus initial à échantillonner. Supposons que nous cherchons à résoudre un problème d'apprentissage sur $\mathcal{X}$ qui consiste à trouver le paramètre (ou jeu de paramètres) θ qui minimise une fonction de coût définie sous la forme particulière suivante:

$$L(\mathcal{X}, \theta) = \sum_{x \in \mathcal{X}} f(x, \theta)$$

où f est une fonction à valeurs dans $\mathbb{R}^+$. De nombreux problèmes classiques d'apprentissage s'écrivent sous cette forme, notamment k-means, la régression linéaire ou logistique, les machines à vecteurs de support, l'approximation en rang bas de matrices, etc. À titre d'exemple, pour k-means, le jeu de paramètres θ est un ensemble de k centroïdes en dimension d: $\theta = \{c_1, \ldots, c_k\}$ et la fonction de coût s'écrit:

$$L(\mathcal{X}, \theta) = \sum_{x \in \mathcal{X}} \min_{c \in \theta} ||x - c||^2.$$

Un objectif usuel en apprentissage est de calculer $\theta^{\text{opt}} = \text{argmin}_\theta \, L(\mathcal{X}, \theta)$. Ce problème de minimisation (ou même les heuristiques habituellement utilisées pour le résoudre approximativement) peut s'avérer trop lourd à calculer quand la taille n du jeu de données grandit. Un coreset est un sous-ensemble $\mathcal{S}$ des points de $\mathcal{X}$ pour lequel il est garanti que la solution du problème de minimisation sur $\mathcal{S}$ est une bonne estimation de θ^{opt}, à une erreur relative près. La litérature existante traîte de différentes techniques pour obtenir de tels coresets: de manière aléatoire ou déterministe, en cherchant à minimiser la taille du coreset ou bien en mettant plus l'accent sur l'efficacité de l'échantillonnage, etc. (voir par exemple la revue récente des techniques coresets Munteanu (2018))

Parmi les différentes directions explorées par la communauté, l'échantillonnage iid avec remise a le double avantage de produire des coresets qui sont de petite taille et rapides à échantillonner (une fois que l'on connaît la bonne distribution de probabilité à utiliser, qui elle peut s'avérer difficile à calculer). Les sous-ensembles obtenus par échantillonnage indépendent souffrent néanmoins de redondance. En effet, une fois que l'on a échantillonné un élément x de $\mathcal{X}$, rien ne nous empêche d'échantillonner un voisin y arbitrairement proche de x, qui n'apporterait pas d'information nouvelle à $\mathcal{S}$.

Nous explorons ici comment les processus ponctuels déterminantaux, grâce entre autres à leur capacité à conserver la diversité d'un jeu de données, peuvent apporter des améliorations aux théorèmes iid existants; et conséquemment inspirer de meilleurs algorithmes pour produire des coresets plus performants.

2 Coresets: définitions et état de l'art

Tout d'abord, précisons certaines notations. Notons $\mathcal{S} = \{x_{s_1}, \ldots, x_{s_m}\}$ un sous-ensemble de $\mathcal{X}$ de taille m (possiblement avec des répétitions). A chaque élément x_s de $\mathcal{S}$ on associe un poids non-négatif $\omega(x_s) \in \mathbb{R}^+$. On associe à un tel sous-ensemble pondéré $\mathcal{S}$ une fonction de coût estimée $\hat{L}$:

$$\hat{L}(\mathcal{S}, \theta) = \sum_{x_s \in \mathcal{S}} \omega(x_s) f(x_s, \theta).$$

Définition *Soit $\epsilon \in]0, 1[$. Un sous-ensemble pondéré $\mathcal{S}$ est un* ϵ-coreset *pour la fonction de coût L si, pour tout paramètre θ, le coût estimé est égal au vrai coût à une erreur relative près:*

$$\forall \theta \quad \left| \frac{\hat{L}}{L} - 1 \right| \leqslant \epsilon.$$

Cette définition est très contraignante[1] dans la mesure où l'erreur doit être contrôlée *pour tout* θ. Son intérêt vient des propriétés suivantes. Notons $\hat{\theta}^{\text{opt}} = \text{argmin}_\theta \hat{L}(\mathcal{S}, \theta)$ le paramètre optimal de la fonction de coût estimée $\hat{L}$. Si $\mathcal{S}$ est un ϵ-coreset pour L, on a:

$$(1-\epsilon)L(\mathcal{X}, \theta^{\text{opt}}) \leqslant (1-\epsilon)L(\mathcal{X}, \hat{\theta}^{\text{opt}}) \leqslant \hat{L}(\mathcal{S}, \hat{\theta}^{\text{opt}}) \leqslant \hat{L}(\mathcal{S}, \theta^{\text{opt}}) \leqslant (1+\epsilon)L(\mathcal{X}, \theta^{\text{opt}})$$

i.e., le coût estimé sur le coreset en $\hat{\theta}^{\text{opt}}$ est une approximation contrôlée (à une erreur relative ϵ près) du coût qu'on aurait obtenu sur $\mathcal{X}$ en θ^{opt}. Si bien que si L a un minimum suffisamment marqué autour de θ^{opt}, alors $\hat{\theta}^{\text{opt}}$ s'avèrera être une très bonne approximation de θ^{opt}. En d'autres termes, lancer un algorithme d'optimisation sur le coreset $\mathcal{S}$ donnera un résultat qui est une approximation contrôlée du vrai résultat (qu'on aurait obtenu en lançant l'algorithme sur l'ensemble des données). Une fois un coreset identifié, et pour autant que le coreset soit de petite taille, les gains en temps de calcul (pour résoudre le problème d'optimisation) peuvent être immenses! Avant d'évoquer l'état de l'art puis notre contribution dans ce domaine, définissons le concept de sensitivité.

Définition *La* sensitivité *d'un élément x_i de $\mathcal{X}$ pour la fonction de coût L est:*

$$\sigma_i = \max_\theta \frac{f(x_i, \theta)}{L(\mathcal{X}, \theta)} \quad \in [0, 1].$$

La somme de toutes les sensitivités est notée $\mathfrak{S} = \sum_i \sigma_i$.

L'état de l'art pour trouver des coresets est (entre autres) l'échantillonnage iid selon une densité de probabilité bien particulière. En effet, on a le théorème suivant:

[1] Il existe une version plus faible de cette définition, que nous ne considérerons pas, où il suffit que l'erreur soit contrôlée autour de $\theta = \theta^{\text{opt}}$

Théorème [Coresets avec échantillonnage iid, voir Langberg (2010) ou Bachem (2017)] *Soit $\boldsymbol{p} \in [0,1]^n$ une distribution de probabilité définie sur l'ensemble des points de $\mathcal{X}$ avec p_i la probabilité d'échantillonner $\boldsymbol{x}_i$ et $\sum_i p_i = 1$. Tirer aléatoirement m échantillons avec remise suivant $\boldsymbol{p}$. Associer à chaque échantillon $\boldsymbol{x}_s$ un poids*[2] *$\omega(\boldsymbol{x}_s) = 1/mp_s$. Le sous-ensemble pondéré obtenu est un ϵ-coreset avec probabilité supérieure à $1-\delta$ si $m \geqslant m^*$ avec:*

$$m^* = \mathcal{O}\left(\frac{1}{\epsilon^2}\left(\max_i \frac{\sigma_i}{p_i}\right)^2 (d' + \log(1/\delta))\right),$$

où d' est la pseudo-dimension (une généralisation de la dimension de Vapnik-Chervonenkis) de Θ, l'espace des paramètres. La distribution de probabilité optimale qui minimise m^* est $p_i = \sigma_i/\mathfrak{S}$. Dans ce cas, la propriété coreset est vérifiée pour $m \geqslant \mathcal{O}\left(\frac{\mathfrak{S}^2}{\epsilon^2}(d' + \log(1/\delta))\right)$.

À titre d'exemple, dans le contexte de k-means, $d' = dk\log k$ et $\mathfrak{S} = O(k)$, si bien qu'il suffit de $m = \mathcal{O}\left(\frac{dk^3 \log k}{\epsilon^2}\right)$ échantillons pour avoir un coreset avec grande probabilité: la taille du coreset minimal est indépendante de n! Cependant, ce remarquable résultat est rendu possible en relègant toute la difficulté du problème dans le calcul de la sensitivité, qui s'avère difficile[3]. En pratique, la communauté a développé des techniques d'encadrement des σ_i pour passer outre leur calcul exact, au prix de theorèmes –et donc d'algorithmes– moins puissants (voir par exemple Bachem (2017)).

3 Processus ponctuels déterminantaux: définitions et application aux coresets

L'idée générale de nos travaux est de passer d'une stratégie d'échantillonnage aléatoire iid à une stratégie avec dépendance pour éviter la redondance typique des sous-ensembles obtenus par tirage indépendant, et donc espérer générer des coresets de plus petite taille. Intuitivement, pour le problème coreset, nous avons envie d'un échantillonnage répulsif: si un élément x est échantillonné nous n'avons pas envie d'échantillonner un autre élément de $\mathcal{X}$ qui soit trop proche de x. Les stratégies d'échantillonnage répulsif sont très nombreuses. En revanche, il en existe très peu qui soient répulsives et tractables analytiquement (par exemple de constante de normalisation connue, de probabilités marginales à tous les ordres connues, etc.): les processus ponctuels déterminantaux (on utilisera le sigle anglais DPP) combinent ces deux propriétés, et c'est pourquoi nous les considérons.

[2] le lecteur reconnaîtra les poids usuels de l'échantillonnage par importance, qui assurent $\mathbb{E}(\hat{L}) = L$.

[3] avant nos travaux, la sensitivité n'avait même pas de forme analytique connue dans aucune des applications classiques mentionnées en introduction. Nous n'en parlerons pas ici mais les lemmes 23 et 25 de notre article Tremblay (2019) donnent la forme analytique de la sensitivité pour le problème 1-means ainsi que pour la régression linéaire.

L'objet central des DPPs est appelé L-ensemble, et n'est rien d'autre qu'une matrice semi-définie positive $\mathsf{L} \in \mathbb{R}^{n \times n}$ (à ne pas confondre avec la fonction coût L), que l'on supposera symmétrique ici. On écrit ses valeurs propres $0 \leqslant \lambda_1 \leqslant \lambda_2 \leqslant \ldots \leqslant \lambda_n$. Dans la suite, nous notons $2^{[n]}$ l'ensemble de tous les sous-ensembles des n premiers entiers.

Définition *Considérons un processus ponctuel, i.e., un processus qui tire aléatoirement un élément $\mathcal{S} \in 2^{[n]}$. Il est* déterminantal *avec L-ensemble L si*

$$\mathbb{P}(\mathcal{S}) = \frac{\det(\mathsf{L}_\mathcal{S})}{\det(\mathsf{I} + \mathsf{L})},$$

où $\mathsf{L}_\mathcal{S}$ est la restriction de L aux lignes et aux colonnes indexées par $\mathcal{S}$.

Les quelques propriétés suivantes sont bien connues, voir Kulesza (2012) pour des détails. Tout d'abord, la normalisation est bien correcte: $\sum_\mathcal{S} \det(\mathsf{L}_\mathcal{S}) = \det(\mathsf{I} + \mathsf{L})$. Aussi, toutes les probabilités d'inclusion, à tous les ordres, sont explicites:

$$\forall \mathcal{A} \in 2^{[n]} \qquad \mathbb{P}(\mathcal{A} \subseteq \mathcal{S}) = \det(\mathsf{K}_\mathcal{A})$$

où $\mathsf{K} = \mathsf{L}(\mathsf{I} + \mathsf{L})^{-1} \in \mathbb{R}^{n \times n}$ est appelé *noyau marginal*. En particulier, la probabilité d'échantillonner i, que l'on note génériquement π_i, vaut simplement K_{ii}. En outre, le caractère répulsif des DPPs est visible par exemple en considérant la probabilité jointe de tirer deux éléments i et j: $\mathbb{P}(\{i,j\} \subseteq \mathcal{S}) = \det(\mathsf{K}_{\{i,j\}}) = \pi_i \pi_j - \mathsf{K}_{ij}^2 \leqslant \pi_i \pi_j$ la probabilité jointe du processus de Poisson (indépendant) associé.

On peut également mentionner que le nombre d'éléments d'un DPP est lui-même aléatoire et distribué selon une somme indépendante de n lois de Bernoulli de paramètres $\{\frac{\lambda_i}{1+\lambda_i}\}$. Dans de nombreux cas pratiques, l'utilisateur préfère spécifier de manière déterministe le nombre d'échantillons, ce qui a ammené à la définition des m-DPPs: des DPPs conditionnés à échantillonner m éléments. Les m-DPPs sont plus utiles en pratique. En revanche, ils sont moins tractables. En particulier, il n'existe plus en général de noyau marginal pour calculer les probabilités d'inclusion. Dans d'autres travaux, voir Barthelmé (2019), nous avons développé des techniques approchées pour les calculer efficacement.

Nous terminons cette suite de définitions avec les DPPs de projection:

Définition *Un* DPP de projection *est un m-DPP dont le L-ensemble est une projection de rang m: $\mathsf{L} = \mathsf{U}\mathsf{U}^\top$, où $\mathsf{U} \in \mathbb{R}^{n \times m}$ vérifie $\mathsf{U}^\top \mathsf{U} = \mathsf{I}_m$.*

Lemme *Un DPP de projection de L-ensemble L est aussi un DPP, de noyau marginal L.*

Ce lemme, issu de Barthelmé (2019), explique pourquoi les DPPs de projection sont des objets très utiles: ils ont à la fois le côté pratique des m-DPPs (un nombre d'échantillons fixe) et la simplicité analytique des DPPs (par exemple π_i est simplement L_{ii}, *i.e.*, la somme des carrés de la i-ème ligne de U).

Dans Tremblay (2019), nous détaillons un ensemble de résultats sur l'utilisation des DPPs au problème des coresets, dont nous reportons ci-dessous un seul morceau

choisi. Nous notons $\hat{L}_{iid}$ l'estimateur de L associé au sous-ensemble $\mathcal{S}$, obtenu par échantillonnage iid de m éléments selon une densité de probabilité $\boldsymbol{p}$ et pondéré par les poids d'échantillonnage d'importance, comme expliqué dans le théorème de la section précédente. Nous comparons cet estimateur avec l'estimateur $\hat{L}$ associé au sous-ensemble $\mathcal{S}$ obtenu par DPP de projection de L-ensemble $\mathsf{L} = \mathsf{U}\mathsf{U}^\top$ où $\mathsf{U} \in \mathbb{R}^{n\times m}$ est tel que i/ $\mathsf{U}^\top\mathsf{U} = \mathsf{I}_m$ (pour que le DPP associé soit bien un DPP de projection), ii/ la probabilité marginale d'échantillonner i, $\pi_i = \mathsf{L}_{ii} = \sum_j \mathsf{U}_{ij}^2$, est fixée[4] à mp_i (cela permet de se comparer équitablement au cas iid). On montre dans Tremblay (2019) qu'un tel U existe nécessairement, et qu'il en existe en général de nombreux. On a:

Théorème *La variance de l'estimateur $\hat{L}$ est nécessairement inférieure à celle de l'estimateur iid équivalent. En particulier:*

$$\forall\theta\in\Theta \qquad \mathrm{Var}(\hat{L}) = \mathrm{Var}(\hat{L}_{\mathrm{iid}}) - \frac{m-1}{m}\left\|\sum_i \frac{f(\boldsymbol{x}_i,\theta)}{mp_i}\tilde{\boldsymbol{v}}_i\right\|^2,$$

où $\tilde{\boldsymbol{v}}_i \in \mathbb{R}^{m^2-m}$ est le vecteur diagramme (cf. Copenhaver (2014)) de la i-ème ligne de U.

En d'autres termes, il est toujours préférable d'échantillonner des coresets via un DPP de projection que de manière iid. Ce résultat se décline de différentes manières, par exemple via des théorèmes coresets généraux de la forme de celui présenté plus haut. En contrepartie d'une meilleure performance des DPPs (vérifiée expérimentalement également), échantillonner un DPP est plus coûteux algorithmiquement qu'échantillonner de manière iid. En effet, dans le cas d'un DPP de projection de rang m, le coût d'échantillonnage coûte $\mathcal{O}(nm^2)$. Nous pointons le lecteur intéressé à notre article Tremblay (2019) pour de plus amples détails.

Bibliographie

O. Bachem, M. Lucic, A. Krause. Practical Coreset Constructions for Machine Learning. *arXiv:1703.06476 [stat]*, 2017.

S. Barthelmé, P.-O. Amblard, N. Tremblay. Asymptotic equivalence of fixed-size and varying-size determinantal point processes. *Bernoulli*, 25(4B):3555–3589, 2019.

M. Copenhaver, Y. Kim, C. Logan, K. Mayfield, S. Narayan, M. Petro, J. Sheperd. Diagram vectors and tight frame scaling in finite dimensions. *Operators and Matrices*, 8(1):73–88, 2014.

A. Kulesza, B. Taskar. Determinantal Point Processes for Machine Learning. *Foundations and Trends in Machine Learning*, 5(23):123–286, 2012.

M. Langberg, L. Schulman. Universal ϵ-approximators for integrals. In *Proceedings of the twenty-first annual ACM-SIAM symposium on Discrete Algorithms*, pp. 598–607, 2010.

A. Munteanu, C. Schwiegelshohn. Coresets-Methods and History: A Theoreticians Design Pattern for Approximation and Streaming Algorithms. *Künstliche Intelligenz*, 32, pp.37–53, 2018.

N. Tremblay, S. Barthelmé, P.-O. Amblard. Determinantal point processes for coresets. *Journal of Machine Learning Research*, 20(168):1–70, 2019.

[4]pour que ce soit soit bien défini, on suppose que toute entrée de $\boldsymbol{p}$ est inférieure à $1/m$