Determinantal Point Processes for Coresets

Nicolas Tremblay, Simon Barthelme, Pierre-Olivier Amblard

Determinantal Point Processes for Coresets

Nicolas Tremblay, Simon Barthelmé, and Pierre-Olivier Amblard

Abstract. When one is faced with a dataset too large to be used all at once, an obvious solution is to retain only part of it. In practice this takes a wide variety of different forms, but among them "coresets" are especially appealing. A coreset is a (small) weighted sample of the original data that comes with a guarantee: that 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, iid random sampling has turned to be one of the most successful methods to build 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 construct general coreset theorems. We apply our results to the k-means problem, and give empirical evidence of the superior performance of DPP samples over state of the art methods.

1. Introduction

Given a learning task, if an algorithm is too slow on large datasets, one can either speed up the algorithm or reduce the amount of data. The theory of "coresets" gives theoretical guarantees on the latter option. A coreset is a weighted sub-sample of the original data, with the guarantee that for any learning parameter, the task's cost function estimated on the coreset is equal to the cost computed on the entire dataset up to a controlled relative error.

An elegant consequence of such a property is that one may run learning algorithms solely on the coreset, allowing for a significant decrease in the computational cost while guaranteeing a controlled error on the learning performance. There are many algorithms that produce coresets (see for instance [1–4]), with some tailored for a specific task (such as k-means, k-medians, logistic regression, etc.), and others more generic [5]. Also, note that there are coreset results both in the streaming setting and the offline setting: we choose here to focus on the offline setting. The state of the art for many problems consists in tailoring a sampling distribution for the dataset at hand, and then sampling iid from that distribution [4, 6, 7]. However, iid samples are generally inefficient, as an iid process is ignorant of the past, and thus liable to sample similar points repeatedly. An avenue for improvement is to produce samples that are less redundant than what iid sampling produces.

DPPs are known to produce diverse samples, and their theoretical properties are well-known [8]. We show here that DPPs can be used to produce diverse samples with the coreset property. Our theorems are quite generic, and assume mostly that the cost functions under study are Lipschitz. We have two main lines of argument: the first is that DPPs do indeed produce coresets, and the second is that DPPs should produce *better* coresets (than iid methods) if one uses the right marginal kernel to define the DPP. We apply our results to the k-means problem, for which the optimal marginal kernel is unfortunately out of reach. We nevertheless argue that a tractable approximation based on a Gaussian kernel and Random Fourier Features can work well in practice and show that it improves over the state of the art on various artificial and real-world datasets.

1.1. Related work

Various coreset construction techniques have been proposed in the past. We follow the recent review [9] to classify them in four categories:

1) Geometric decompositions [1, 2, 10, 11]. These methods propose to first discretize the ambient space of the data in a set of cells, snap each data point to its nearest cell in the discretization,

All three authors are with CNRS, Univ Grenoble-Alpes, Gipsa-lab, France.

and then use these weighted cells to approximate the target tasks. In all these constructions, the minimum number of samples required to guarantee the coreset property depends exponentially in the number of dimensions of the ambient space, making them less useful in high-dimensional problems.

2) Gradient descent [3, 12–14]. These methods have been originally designed for the smallest enclosing ball problem (*i.e.*, finding the ball of minimum radius enclosing all datapoints), and have been later generalized to other problems. One of the main drawback of these algorithms in the k-means setting for instance is their running time exponentially depending on the number of classes k [14]. Also, these algorithms provide only so-called weak coresets.
3) Random sampling [4, 6, 7, 15, 16]. The state of the art for many different tasks such as k-means or k-median is currently via iid random non-uniform sampling. The optimal probability distribution with which to sample datapoints should be set proportional to a quantity known as sensitivity and introduced by Langberg et al. [4], also known as statistical leverage scores in the related field of random linear algebra literature [17]. In practice, it is unpractical to compute all sensitivities: state of the art algorithms rely on bi-criteria approximations to find upper bounds of the sensitivity, and set the probability distribution proportional to this upper bound. More details on these results are provided in Section 2.4.
4) Sketching and projections [18–25]. Another direction of research regarding data reduction that provably keeps the relevant information for a given learning task is via sketches [19]: compressed mappings (obtained via projections) of the original dataset that are in general easy to update with new or changed data. Sketches are not strictly speaking coresets, and the difference resides in the fact that coresets are subsets of the data, whereas sketches are projections of the data. Note finally that the frontier between the two is permeable and some data summaries may combine both.

1.2. Contributions

Our main contribution falls into the random sampling category within which we propose to improve over iid sampling by considering negatively correlated point processes, *i.e.*, point processes for which sampling jointly two similar datapoints is less probable than sampling two very different datapoints. We decide to concentrate on a special type of negative correlation sampling: determinantal point processes, known to provide samples representing the "diversity" of the dataset [8]. To the best of our knowledge, we provide the first coreset guarantee using non-iid random sampling. DPPs are parametrized by a matrix called marginal kernel and denoted by K, whose diagonal elements encode for the inclusion probabilities of each sample, and non-diagonal elements encode the correlation between samples. We first show that whatever the choice of the non-diagonal elements of K, if the inclusion probabilities of the DPP are set proportional to the sensitivity, then the results are at least as good as the iid case. We further show with a variance argument that DPP sampling, due to the negative correlations encoded by the non-diagonal elements of K, necessarily provides better performances than its iid counterpart with same inclusion probabilities. Technical difficulties in controlling the concentration properties of correlated samples currently keeps us from exactly deriving the minimum coreset size one may hope for using DPPs.

We then apply our theorems to the k-means problem where the initial data consists in N points in $\mathbb{R}^d$. We discuss the ideal choice of marginal kernel K for DPP sampling in this case. This ideal kernel being prohibitive to compute in practice, we provide a heuristics based on random Fourier features of the Gaussian kernel. This heuristics outputs a coreset sample in $\mathcal{O}(Nm^2)$ time where m is the number of samples of the coreset, to compare to $\mathcal{O}(Nmd)$ the cost of the current state of the art iid sampling algorithm via bi-criteria approximation. m being necessarily larger than d to obtain the coreset guarantee, our proposition is computationally heavier, especially as m increases. We provide nonetheless empirical evidence showing that this additional cost comes with enhanced performances.

We also provide a side contribution that may be of independent interest: an explicit formula for the sensitivity in the 1-means case.

Finally, a Python toolbox will soon be available on the authors' website[1].

1.3. Organization of the paper

The paper is organized as follows. Section 2 recalls the background: the considered types of learning problems under consideration, the formal definition of coresets, sensitivities and DPPs. Section 3 presents our main theorems on the performance of DPPs to sample coresets. In Section 4, we show how these theorems are applicable to the k-means problem. We provide in Section 5 a discussion on the choice of marginal kernel adapted to the k-means problem, and detail our sampling algorithm. Finally, Section 6 presents experiments on artificial as well as real-world datasets comparing the performances of DPP sampling vs. iid sampling. Section 7 concludes the paper. Note that for readability's sake, we pushed many proofs and some implementation details in the Appendix.

2. Background

Let $\mathcal{X} = \{x_1, \dots, x_N\}$ be a set of N datapoints. Let Θ be a metric space of parameters, and θ an element of Θ. We consider cost functions of the form:

$$L(\mathcal{X}, \theta) = \sum_{x \in \mathcal{X}} f(x, \theta), \tag{1}$$

where f is a non-negative γ-Lipschitz function ($\gamma > 0$) with respect to θ, *i.e.*, $\forall x \in \mathcal{X}$:

$$\forall \theta \in \Theta \qquad f(x, \theta) \geqslant 0, \tag{2}$$

$$\forall (\theta, \theta') \in \Theta^2 \qquad |f(x, \theta) - f(x, \theta')| \leqslant \gamma \; d_\Theta(\theta, \theta'). \tag{3}$$

Many classical machine learning cost functions fall under this model: k-means (as will be shown in Section 4), k-median, logistic or linear regression, support-vector machines, etc.

2.1. Problem considered

A standard learning task is to minimize the cost L over all $\theta \in \Theta$. We write:

$$\theta^{\text{opt}} = \operatorname*{argmin}_{\theta \in \Theta} L(\mathcal{X}, \theta), \qquad L^{\text{opt}} = L(\mathcal{X}, \theta^{\text{opt}}) \qquad \text{and} \qquad \langle f \rangle_{\text{opt}} = \frac{L^{\text{opt}}}{N}. \tag{4}$$

In some instances of this problem, e.g., if N is very large and/or if f is expensive to evaluate and should be computed as rarely as possible, one may rely on sampling strategies to efficiently perform this optimization task.

2.2. Coresets

Let $\mathcal{S} \subset \mathcal{X}$ be a subset of $\mathcal{X}$. To each element $s \in \mathcal{S}$, associate a weight $\omega(s) \in \mathbb{R}^+$. Define the estimated cost as:

$$\hat{L}(\mathcal{S}, \theta) = \sum_{s \in \mathcal{S}} \omega(s) f(s, \theta). \tag{5}$$

Definition 2.1 (Coreset)**.** *Let $\epsilon \in (0, 1)$. The weighted subset $\mathcal{S}$ is a ϵ-coreset for L if,* for any parameter θ*, the estimated cost is equal to the exact cost up to a relative error:*

$$\forall \theta \in \Theta \qquad \left| \frac{\hat{L}}{L} - 1 \right| \leqslant \epsilon. \tag{6}$$

This is the so-called "strong" coreset definition, as the ϵ-approximation is required for all $\theta \in \Theta$. A weaker version of this definition exists in the literature where the ϵ-approximation is only required for θ^{opt}. In the following, we focus on theorems providing the strong coreset property.

[1] Toolbox `DPP4Coreset` soon available at `http://www.gipsa-lab.fr/~nicolas.tremblay/index.php?page=downloads`

Let us write $\hat{\theta}^{\text{opt}}$ the optimal solution computed on the weighted subset $\mathcal{S}$. An important consequence of the coreset property is the following:

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

i.e.: running an optimization algorithm on the weighted sample $\mathcal{S}$ will result in a minimal learning cost that is a controlled ϵ-approximation of the learning cost one would have obtained by running the same algorithm on the entire dataset $\mathcal{X}$. Note that the guarantee is over costs only: the estimated optimal parameters $\hat{\theta}^{\text{opt}}$ and θ^{opt} may be different. Nevertheless, if the cost function is well suited to the problem: either there is one clear global minimum and the estimated parameters will almost coincide; or there are multiple solutions for which the learning cost is similar and selecting one over the other is not an issue.

In terms of computation cost, if the sampling scheme is efficient, N is very large and/or f is expensive to compute for each datapoint, coresets thus enable a significant gain in computing time.

2.3. Sensitivity

Langberg and Schulman [4] introduce the notion of sensitivity:

Definition 2.2 (Sensitivity)**.** *The sensitivity of a datapoint $x_i \in \mathcal{X}$ with respect to $f : \mathcal{X}, \Theta \to \mathbb{R}^+$ is:*

$$\sigma_i = \max_{\theta \in \Theta} \frac{f(x_i,\theta)}{L(\mathcal{X},\theta)} \qquad \in [0,1]. \tag{7}$$

Also, the total sensitivity is defined as :

$$\mathfrak{S} = \sum_{i=1}^{N} \sigma_i. \tag{8}$$

The sensitivity is sometimes called statistical leverage score [17]. It plays a crucial role in the iid random sampling theorems in the coreset literature as well as in the randomized numerical linear algebra literature [20]. In words, the sensitivity σ_i is the worse case contribution of datapoint x_i in the total cost. Intuitively, the larger it is, the larger its "outlierness" [26].

2.4. iid importance sampling and state of the art results

In the iid sampling paradigm, the importance sampling estimator of L is the following. Say the sample set $\mathcal{S}$ consists in m samples drawn iid with replacement from a probability distribution $\boldsymbol{p}$. Denote by ϵ_i the random variable counting the number of occurences of x_i in $\mathcal{S}$. One may define $\hat{L}$, the so-called importance sampling estimator of L, as :

$$\hat{L}(\mathcal{S},\theta) = \sum_i \frac{f(x_i,\theta)\epsilon_i}{mp_i}. \tag{9}$$

One can show that $\mathbb{E}(\epsilon_i) = mp_i$, such that $\hat{L}$ is an unbiased estimator of L:

$$\mathbb{E}(\hat{L}(\mathcal{S},\theta)) = L(\mathcal{X},\theta). \tag{10}$$

The concentration of $\hat{L}$ around its expected value is controlled by the following state of the art theorem:

Theorem 2.3 (Coresets with iid random sampling)**.** *Let $\boldsymbol{p} \in [0,1]^N$ be a probability distribution over all datapoints $\mathcal{X}$ with p_i the probability of sampling x_i and $\sum_i p_i = 1$. Draw m iid samples with replacement according to $\boldsymbol{p}$. Associate to each sample x_s a weight $\omega(s) = 1/mp_s$. The obtained weighted subset is a ϵ-coreset with probability $1-\delta$ provided that:*

$$m \geqslant m^* \tag{11}$$

with

$$m^* = \mathcal{O}\left(\frac{1}{\epsilon^2}\left(\max_i \frac{\sigma_i}{p_i}\right)^2 (d' + \log(1/\delta))\right), \tag{12}$$

where d' is the pseudo-dimension of Θ (a generalization of the Vapnik-Chervonenkis dimension). The optimal probability distribution minimizing m^* is $p_i = \sigma_i/\mathfrak{S}$. In this case, the obtained weighted subset is a ϵ-coreset with probability $1-\delta$ provided that:

$$m \geqslant \mathcal{O}\left(\frac{\mathfrak{S}^2}{\epsilon^2}(d' + \log(1/\delta))\right). \tag{13}$$

For instance, in the k-means setting[2]*, $d' = dk\log k$ and $\mathfrak{S} = \mathcal{O}(k)$ such that the coreset property is guaranteed with probability $1-\delta$ provided that:*

$$m \geqslant \mathcal{O}\left(\frac{k^2}{\epsilon^2}(dk\log k + \log(1/\delta))\right). \tag{14}$$

This theorem is taken from [16] but its original form goes back to [4]. Note that sensitivities cannot be computed rapidly, such that, as it is, this theorem is unpractical. Thankfully, bi-criteria approximation schemes (such as Alg. 2 of [16], or other propositions such as in [15, 28]) may be used to efficiently find an upper bound of the sensitivity for all i: $s_i \geqslant \sigma_i$. Noting $S = \sum s_i$, and setting $p_i = s_i/S$, one shows that the coreset property may be guaranteed in the iid framework provided that $m \geqslant \mathcal{O}\left(\frac{S^2}{\epsilon^2}(d' + \log(1/\delta))\right)$. This idea of using bi-criteria approximations to upper bound the sensitivity also goes back to [4] and has been used in many works on coresets [5, 7, 15, 28].

Note that if one authorizes coresets with negative weights (that is, authorizes negative weights in the estimated cost equation 5), then the stated result may be further improved [15]. Nevertheless, we prefer to restrict ourselves to positive weights as optimization algorithms such as Lloyd's k-means heuristics [29] are in practice more straightforward to implement on positively weighted sets rather than on sets with possibly negative weights.

Finally, Braverman et al. (Thm 5.5 of [7]) improve the previous theorem by showing that under the same non-uniform iid framework, the coreset property is guaranteed provided that $m \geqslant \mathcal{O}\left(\frac{\mathfrak{S}}{\epsilon^2}(d'\log\mathfrak{S} + \log(1/\delta))\right)$, thus reducing the term in $\mathfrak{S}^2$ to $\mathfrak{S}\log\mathfrak{S}$. In this paper, we present results proportional to the squared total sensitivity $\mathfrak{S}^2$, and we thus prefer to focus on the results of Thm. 2.3 in order to ease comparison.

2.5. Correlated importance sampling

Eq. (9) is not suited to correlated sampling and, in the following, we will use a slightly different importance sampling estimator, more adapted to this case. Consider a point process defined on $\mathcal{X}$ that outputs a random sample $\mathcal{S} \subset \mathcal{X}$. For each data point x_i, denote by π_i its inclusion (or marginal) probability:

$$\pi_i = \mathbb{P}(x_i \in \mathcal{S}). \tag{15}$$

Moreover, denote by ϵ_i the random Boolean variable such that $\epsilon_i = 1$ if $x_i \in \mathcal{S}$, and 0 otherwise. In the following, we will focus on the following definition of the importance sampling cost estimator $\hat{L}$:

$$\hat{L}(\mathcal{S}, \theta) = \sum_i \frac{f(x_i, \theta)\epsilon_i}{\pi_i}. \tag{16}$$

By construction, $\mathbb{E}(\epsilon_i) = \pi_i$, such that $\hat{L}$ is an unbiased estimator of L:

$$\mathbb{E}(\hat{L}(\mathcal{S}, \theta)) = L(\mathcal{X}, \theta). \tag{17}$$

Studying the coreset property in this setting boils down to studying the concentration properties of $\hat{L}$ around its expected value.

[2] In the literature [15, 27], d' is often taken to be equal to dk. We nevertheless agree with [16] and their discussion in Section 2.6 regarding k-means' pseudo-dimension and thus write $d' = dk\log k$

2.6. Determinantal Point Processes

In order to induce negative correlations within the samples, we choose to focus on Determinantal Point Processes (DPP), point processes that have recently gained attention due to their ability to output "diverse" subsets within a tractable probabilistic framework (for instance with explicit formulas for marginal probabilities). In the following, $[N]$ denotes the set of N first integers $\{1, 2, \dots, N\}$.

Definition 2.4 (Determinantal Point Process [8])**.** *Consider a point process, i.e., a process that randomly draws an element $\mathcal{S} \in [N]$. It is determinantal if there exists a semi-definite positive matrix $\mathsf{K} \in \mathbb{R}^{N \times N}$ verifying $0 \preceq \mathsf{K} \preceq 1$ such that, for every $\mathcal{A} \subseteq \mathcal{S}$,*

$$\mathbb{P}(\mathcal{A} \subseteq \mathcal{S}) = \det(\mathsf{K}_{\mathcal{A}}),$$

where $\mathsf{K}_{\mathcal{A}}$ is the restriction of K to the rows and columns indexed by the elements of $\mathcal{A}$. K is called the marginal kernel of the DPP.

By definition, the probability of inclusion of i, denoted by π_i, is equal to K_{ii} and the expected number[3] of samples is $\mu = \sum_i \pi_i = \mathrm{Tr}(\mathsf{K})$. Moreover, to gain insight in the repulsive nature of DPPs, one may readily see that the joint marginal probability of sampling i and j reads: $\det(\mathsf{K}_{\{i,j\}}) = \pi_i\pi_j - \mathsf{K}_{ij}^2$ and is necessarily smaller than $\pi_i\pi_j$, the joint probability in the case of uncorrelated sampling. The stronger the "interaction" between i and j (encoded by the absolute value of element K_{ij}), the smaller the probability of sampling both jointly: this determinantal nature thus favors diverse sets of samples.

Our goal will be to design the best possible K such that sampling a DPP with marginal kernel K guarantees the coreset property with high probability.

3. Coreset theorems

We now detail our main contributions. In Sections 3.1 to 3.2, we present our main theorems providing sufficient conditions on the marginal probabilities (*i.e.*, the diagonal elements of K) to guarantee the coreset property. We will see that, similar to the iid case (theorem 2.3), the optimal marginal probability should be set proportional to the sensitivity. These theorems are valid for any choice of non-diagonal elements of the matrix K. We further discuss in Section 3.3 how one may take advantage of these additional degrees of freedom to improve the coreset performance over iid sampling.

3.1. Determinantal Point Processes for Coresets

Theorem 3.1 (DPP for coresets)**.** *Consider $\mathcal{S}$ a sample from a DPP with marginal kernel K. Let $\epsilon \in (0,1)$, $\delta \in (0,1)$. Denote by n the minimum number of balls of radius $\epsilon\langle f \rangle_{opt}/6\gamma$ necessary to cover Θ. With probability higher than $1-\delta$, $\mathcal{S}$ is a ϵ-coreset provided that*

$$\mu \geqslant \mu^* = \max(\mu_1^*, \mu_2^*) \tag{18}$$

with:

$$\mu_1^* = \frac{32}{\epsilon^2}\left(\epsilon \max_i \frac{\sigma_i}{\bar{\pi}_i} + 4\left(\max_i \frac{\sigma_i}{\bar{\pi}_i}\right)^2\right)\log\frac{10n}{\delta}, \tag{19}$$

$$\mu_2^* = \frac{32}{\epsilon^2}\left(\frac{\epsilon}{N\bar{\pi}_{min}} + \frac{4}{N^2\bar{\pi}_{min}^2}\right)\log\frac{10}{\delta}, \tag{20}$$

and $\forall i,\ \bar{\pi}_i = \pi_i/\mu$.

The proof is provided in Appendix A. Note that μ_1^* and μ_2^* are not independent of μ: they are in fact dependent via $\bar{\pi}_i = \pi_i/\mu$. While this formulation may be surprising at first, this is due to the fact that in non-iid settings, separating μ from π_i is not as straightforward as in the iid case (in Thm. 2.3, m and p_i are independent) . Also, we decide upon this particular formulation of the theorem to mimic classical concentration results obtained with iid sampling.

[3] in fact, the number of samples of a DPP is itself random: it is a sum of Bernoullis parametrized by K's eigenvalues (see [8] for details)

In order to simplify further analysis, we suppose from now on that $N\sigma_{\min} \geqslant 1$. As shown in the second lemma of Appendix B, this is in fact verified in the k-means case on which we will focus in Section 4. Nevertheless, the following results may be generalized to cases with unconstrained $\sigma_{\min}$ if needed.

Lemma 3.2. *If $N\sigma_{min} \geqslant 1$, then $\mu_1^* \geqslant \mu_2^*$ and the coreset property of Theorem 3.1 is verified if:*

$$\mu \geqslant \mu^* = \frac{32}{\epsilon^2}\left(\epsilon \max_i \frac{\sigma_i}{\bar{\pi}_i} + 4\left(\max_i \frac{\sigma_i}{\bar{\pi}_i}\right)^2\right)\log\frac{10n}{\delta} \tag{21}$$

with $\forall i,\ \bar{\pi}_i = \pi_i/\mu$.

Proof. Denote by j the index for which $\bar{\pi}_i$ is minimal and, provided that $N\sigma_{\min} \geqslant 1$, one has:

$$\max_i \frac{\sigma_i}{\bar{\pi}_i} N\bar{\pi}_{\min} \geqslant N\sigma_j \geqslant N\sigma_{\min} \geqslant 1.$$

This implies that:

$$(\max_i \frac{\sigma_i}{\bar{\pi}_i} N\bar{\pi}_{\min})^2 \geqslant \log\frac{10}{\delta} \ / \ (\log\frac{10}{\delta} + \log n), \tag{22}$$

as n is necessarily larger than 1. One can show that Eq. (22) is equivalent to $\mu_1^* \geqslant \mu_2^*$, such that $\mu^* = \max(\mu_1^*, \mu_2^*) = \mu_1^*$. □

One would like to have the coreset guarantee for a minimal number of samples, that is: to find the marginal probabilities π_i minimizing μ^*.

Corollary 3.3. *If there exists $\alpha > 0$ and $\beta \geqslant 1$ such that:*

$$\forall i \qquad \alpha\sigma_i \leqslant \pi_i \leqslant \alpha\beta\sigma_i, \tag{23}$$

$$\textit{and} \qquad \frac{\alpha}{\beta} \geqslant \frac{32}{\epsilon^2}(\epsilon + 4\mathfrak{S})\log\frac{10n}{\delta}, \tag{24}$$

then $\mathcal{S}$ is a ϵ-coreset with probability at least $1-\delta$. In this case, the expected number of samples verifies:

$$\mu \geqslant \frac{32}{\epsilon^2}\beta\mathfrak{S}(\epsilon + 4\mathfrak{S})\log\frac{10n}{\delta}.$$

Proof. Let us suppose that there exists $\alpha > 0$ and $\beta \geqslant 1$ such that:

$$\forall i, \qquad \alpha\sigma_i \leqslant \pi_i \leqslant \alpha\beta\sigma_i.$$

Note that:

$$\epsilon \max_i \frac{\sigma_i}{\pi_i} + 4\left(\max_i \frac{\sigma_i}{\pi_i}\right)^2 \mu \leqslant \frac{\epsilon}{\alpha} + 4\frac{\mu}{\alpha^2} \leqslant \frac{\epsilon}{\alpha} + 4\frac{\beta\mathfrak{S}}{\alpha} = \frac{\beta}{\alpha}\left(\frac{\epsilon}{\beta} + 4\mathfrak{S}\right) \leqslant \frac{\beta}{\alpha}\left(\epsilon + 4\mathfrak{S}\right).$$

Thus, the inequality

$$\frac{\alpha}{\beta} \geqslant \frac{32}{\epsilon^2}\left(\epsilon + 4\mathfrak{S}\right)\log\frac{10n}{\delta}$$

implies:

$$1 \geqslant \frac{32}{\epsilon^2}\left(\epsilon \max_i \frac{\sigma_i}{\pi_i} + 4\left(\max_i \frac{\sigma_i}{\pi_i}\right)^2 \mu\right)\log\frac{10n}{\delta},$$

that we recognize as the coreset condition (21) by multiplying on both sides by μ: $\mathcal{S}$ is indeed a ϵ-coreset with probability larger than $1-\delta$. Moreover, in this case:

$$\mu = \sum_i \pi_i \geqslant \alpha \sum_i \sigma_i = \alpha\mathfrak{S} \geqslant \frac{32}{\epsilon^2}\beta\mathfrak{S}(\epsilon + 4\mathfrak{S})\log\frac{10n}{\delta}.$$

□

Corollary 3.3 is applicable to cases where $\sigma_{\max}$ is not too large. In fact, in order for $\alpha\sigma_i$ to be smaller than π_i, and thus smaller than 1 as π_i is a probability, α should always be set inferior to $\frac{1}{\sigma_{\max}}$. Now, if $\sigma_{\max}$ is so large that $\frac{1}{\sigma_{\max}} \leqslant \frac{32}{\epsilon^2}(\epsilon + 4\mathfrak{S}) \log \frac{10n}{\delta}$, then, even by setting β to its minimum value 1, there is no admissible α verifying both conditions (26) and (27). We refer to App. C for a simple workaround if this issue arises. We will further see in the experimental section (Section 6) that outliers are not an issue in practice.

3.2. m-Determinantal Point Processes for coresets

In some cases, we would like to specify deterministically the number of samples, instead of having a random number of them (with a given mean). This leads to m-DPPs: DPPs conditioned to output m samples.

Definition 3.4 (m-DPP [8])**.** *Consider a point process that randomly draws an element $\mathcal{S} \in [N]$. This process is an m-DPP with kernel K if:*

i) $\forall \mathcal{S}$ *s.t.* $|\mathcal{S}| \neq m$, $\quad \mathbb{P}(\mathcal{S}) = 0$
ii) $\forall \mathcal{S}$ *s.t.* $|\mathcal{S}| = m$, $\quad \mathbb{P}(\mathcal{S}) = \frac{1}{Z}\det(\mathsf{K}_{\mathcal{S}})$*, where Z is a normalization constant.*

Note that π_i, the marginal probability of inclusion, is not necessarily equal to K_{ii} anymore in the case of an m-DPP. In fact:

$$\pi_i = \frac{1}{Z} \sum_{\mathcal{S} \text{ s.t } |\mathcal{S}|=m \text{ and } i \in \mathcal{S}} \det(\mathsf{K}_{\mathcal{S}})$$

We deliver the following result assuming that $N\sigma_{\min} \geqslant 1$.

Theorem 3.5 (m-DPP for coresets)**.** *Let $\mathcal{S}$ be a sample from an m-DPP with kernel K, $\epsilon \in (0,1)$, and n the minimal number of balls of radius $\frac{\epsilon\langle f \rangle_{opt}}{6\gamma}$ necessary to cover Θ. We assume for simplicity that $N\sigma_{min} \geqslant 1$. $\mathcal{S}$ is a ϵ-coreset with probability larger than $1-\delta$ provided that:*

$$m \geqslant \frac{32}{\epsilon^2}\left(\max_i \frac{\sigma_i}{\bar{\pi}_i}\right)^2 \log\frac{4n}{\delta} \tag{25}$$

with $\bar{\pi}_i = \pi_i/m$. Also, if there exists $\alpha > 0$ and $\beta \geqslant 1$ such that:

$$\forall i \qquad \alpha\sigma_i \leqslant \pi_i \leqslant \alpha\beta\sigma_i, \tag{26}$$

$$\text{and} \qquad \frac{\alpha}{\beta} \geqslant \frac{32}{\epsilon^2}\mathfrak{S}\log\frac{4n}{\delta}, \tag{27}$$

then $\mathcal{S}$ is a ϵ-coreset with probability larger than $1-\delta$. In this case, the number of samples verifies:

$$m \geqslant \frac{32}{\epsilon^2}\beta\mathfrak{S}^2\log\frac{4n}{\delta}.$$

Proof. According to [30], replace Eq. (44) by:

$$m \geqslant \frac{8}{\epsilon^2}C^2\log\frac{2}{\delta}, \tag{28}$$

with $C = \max_i \frac{f(x_i,\theta)}{L\bar{\pi}_i}$, where $\bar{\pi}_i$ is a shorthand for π_i/m; and Eq. (46) by:

$$m \geqslant \frac{8}{\epsilon^2 N^2 \bar{\pi}^2_{\min}}\log\frac{2}{\delta}, \tag{29}$$

and change accordingly the rest of the proof. □

3.3. Links with the iid case and the variance argument

Let us first compare our results with Thm. 2.3 obtained in the iid setting. A few remarks are in order:

1) setting β to 1 and π_i to $\alpha\sigma_i$ in Thm 3.5, the minimum number of required samples is $\frac{32\mathfrak{S}^2}{\epsilon^2}(\log n + \log\frac{4}{\delta})$, to compare to $\mathcal{O}(\frac{\mathfrak{S}^2}{\epsilon^2}(d' + \log(1/\delta)))$ of Thm 2.3, where d' is the pseudo-dimension of Θ. n being the number of balls of radius $\frac{\epsilon\langle f\rangle_{\text{opt}}}{6\gamma}$ necessary to cover Θ, it will typically be $\frac{\epsilon\langle f\rangle_{\text{opt}}}{6\gamma}$ to the power of the ambient dimension of Θ (similar to d'). Both forms are very similar, up to the dependency in ϵ and in $\langle f\rangle_{\text{opt}}$ of the log term. This difference is due to the fact that coreset theorems in the iid case (see for instance [16]) take advantage of powerful results from the Vapnik-Chervonenkis (VC) theory such as the ones detailed in [31]. Unfortunately, these fundamental results are valid in the iid case, and are not easily generalized to the correlated case. Further work should enable to reduce this small gap.
2) Outliers are not naturally dealt with using our proof techniques, mainly due to our multiple use of the union bound that necessarily englobes the worse-case scenario. In fact, in the importance sampling estimator used in the iid case (Eq. 9), outliers are not problematic as they can be sampled several times. In our setting, outliers are constrained to be sampled only once, which in itself makes sense, but complicates the analysis. Empirically, we will see in Section 6 that outliers are not an issue.
3) Finally, our results take *only* into account the inclusion probability of the DPP, that is: the diagonal elements of K. These theorems are thus valid for any choice of non-diagonal elements (provided K stays semi-definite positive with eigenvalues between 0 and 1). As we will see with the following variance argument: a smart choice of K's non-diagonal elements will necessarily improve our results, thus outperforming the iid setting.

Theorem 3.6 (The variance argument)**.** *Given $\theta \in \Theta$, and writing Var_{iid} the variance of the importance sampling estimator of Eq. 16 in the iid case, we have:*

$$\text{Var}(\hat{L}) = \text{Var}_{iid} - \sum_{i\neq j} \frac{\mathsf{K}_{ij}^2}{\pi_i\pi_j} f(x_i,\theta)f(x_j,\theta). \tag{30}$$

As the function f is positive, the variance of $\hat{L}$ via DPP sampling is thus necessarily smaller than its iid counterpart with same probability of inclusion.

Proof. We have:

$$\text{Var}(\hat{L}) = \mathbb{E}(\hat{L}^2) - \mathbb{E}(\hat{L})^2 \tag{31}$$

$$= \sum_{i,j} \frac{\mathbb{E}(\epsilon_i\epsilon_j)}{\pi_i\pi_j} f(x_i,\theta)f(x_j,\theta) - L^2. \tag{32}$$

As $\mathcal{S}$ is sampled from a DPP, we have: $\mathbb{E}(\epsilon_i\epsilon_j) = \det(\mathsf{K}_{\{i,j\}}) = \pi_i\pi_j - \mathsf{K}_{ij}^2$, *i.e.*:

$$\text{Var}(\hat{L}) = \text{Var}_{iid} - \sum_{i\neq j} \frac{\mathsf{K}_{ij}^2}{\pi_i\pi_j} f(x_i,\theta)f(x_j,\theta), \tag{33}$$

where $\text{Var}_{iid} = \sum_{i,j} f(x_i,\theta)f(x_j,\theta) - L^2$ is the variance one would obtain with a Poisson-type uncorrelated sampling strategy with same marginal probability, *i.e.*, for processes such that $\mathbb{E}(\epsilon_i\epsilon_j) = \pi_i\pi_j$. □

As a consequence, *adding negative correlations (i.e., non-zero non-diagonal elements of K) necessarily decreases the estimation's variance.* To conclude this Section: we provided theorems that explain how the diagonal elements of K should be set in order to match the iid performance. And we show here that any choice of off-diagonal elements of K (provided K stays SDP with eigenvalues between 0 and 1) will *necessarily* improve the coreset performance of DPP sampling versus its iid couterpart. In addition, this variance equation will provide useful indications for our choice of marginal kernel in Section 5.

4. Application to k-means

The above results are valid for any learning problem of the form detailed in Section 2.1. We now specifically consider the k-means problem on a set $\mathcal{X}$ comprised of N datapoints in $\mathbb{R}^d$. This problem boils down to finding k centroids $\theta = (c_1, \dots, c_k)$, all in $\mathbb{R}^d$, such that the following cost is minimized:

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

Let ρ be the diameter of the minimum eclosing ball of $\mathcal{X}$ (the smallest ball enclosing all points in $\mathcal{X}$).

Theorem 3.1 is applicable to the k-means problem, such that:

Corollary 4.1 (DPP for k-means)**.** *Let $\mathcal{S}$ be a sample from a DPP with marginal kernel K. Let $\epsilon, \delta \in (0,1)^2$. With probability at least $1-\delta$, $\mathcal{S}$ is a ϵ-coreset provided that:*

$$\mu \geqslant \mu^* = \frac{32}{\epsilon^2} \left(\epsilon \max_i \frac{\sigma_i}{\bar{\pi}_i} + 4 \left(\max_i \frac{\sigma_i}{\bar{\pi}_i} \right)^2 \right) \left(kd \log \left(\frac{24\rho^2}{\epsilon \langle f \rangle_{opt}} + 1 \right) + \log \frac{10}{\delta} \right), \tag{34}$$

with $\forall i, \bar{\pi}_i = \pi_i / \mu$.

If there exists $\alpha > 0$ and $\beta \geqslant 1$ such that:

$$\forall i, \qquad \alpha \sigma_i \leqslant \pi_i \leqslant \alpha \beta \sigma_i, \tag{35}$$

$$\text{and} \qquad \frac{\alpha}{\beta} \geqslant \frac{32}{\epsilon^2} (\epsilon + 4\mathfrak{S}) \left(kd \log \left(\frac{24\rho^2}{\epsilon \langle f \rangle_{opt}} + 1 \right) + \log \frac{10}{\delta} \right), \tag{36}$$

then $\mathcal{S}$ is a ϵ-coreset with probability larger than $1-\delta$. In this case, the expected number of samples verifies:

$$\mu \geqslant \frac{32}{\epsilon^2} \beta \mathfrak{S} (\epsilon + 4\mathfrak{S}) \left(kd \log \left(\frac{24\rho^2}{\epsilon \langle f \rangle_{opt}} + 1 \right) + \log \frac{10}{\delta} \right).$$

Proof. Let us write $\mathcal{B}$ the minimum enclosing ball of $\mathcal{X}$, of diameter ρ. The potentially interesting centroids are necessarily included in $\mathcal{B}$ such that the space of parameters Θ in the k-means setting is the set of all possible k centroids in $\mathcal{B}$: $\Theta = \mathcal{B}^k$. The metric d_Θ we consider is the Hausdorff metric associated to the Euclidean distance:

$$\forall \theta, \theta', \qquad d_\Theta(\theta, \theta') = \max \left\{ \max_{c \in \theta} \min_{c' \in \theta'} \|c - c'\|_2 \,,\ \max_{c' \in \theta'} \min_{c \in \theta} \|c - c'\|_2 \right\}.$$

- An ϵ'-net of Θ. Consider $\Gamma_\mathcal{B}$ an ϵ'-net of $\mathcal{B}$: it consists in at least $(\frac{2\rho}{\epsilon'} + 1)^d$ small balls of radius ϵ' (see, e.g., Lemma 2.5 in [32]). Consider $\Gamma = \Gamma_\mathcal{B}^k$ of cardinality $|\Gamma| = (\frac{2\rho}{\epsilon'} + 1)^{kd}$. Let us show that Γ is an ϵ'-net of Θ, that is:

$$\forall \theta \in \mathcal{B}^k, \quad \exists \theta^* \in \Gamma \quad \text{s.t.} \quad d_\Theta(\theta, \theta^*) \leqslant \epsilon'.$$

In fact, consider $\theta = (c_1, \dots, c_k) \in \mathcal{B}^k$. By construction, as $\Gamma_\mathcal{B}$ is an ϵ'-net of $\mathcal{B}$, we have:

$$\forall i = 1, \dots, k \qquad \exists c_i^* \in \Gamma_\mathcal{B} \quad \text{s.t.} \quad \|c_i - c_i^*\| \leqslant \epsilon'.$$

Writing $\theta^* = (c_1^*, \dots, c_k^*) \in \Gamma$, one has:

$$d_\Theta(\theta, \theta^*) \leqslant \epsilon',$$

which proves that Γ is an ϵ'-net of Θ. The number of balls of radius $\epsilon' = \epsilon \langle f \rangle_{\text{opt}} / 6\gamma$ necessary to cover Θ is thus at least $n = (\frac{12\rho\gamma}{\epsilon \langle f \rangle_{\text{opt}}} + 1)^{kd}$.

- $f(x, \theta)$ is γ-Lipschitz with $\gamma = 2\rho$. Consider any θ, θ' and $x \in \mathcal{X}$. We want to show that:

$$-\gamma \; d_\Theta(\theta, \theta') \leqslant f(x,\theta) - f(x, \theta') \leqslant \gamma \; d_\Theta(\theta, \theta').$$

Let us write $c = \text{argmin}_{t\in\theta} \|x-t\|^2$ the centroid in θ closest to x and $c' = \text{argmin}_{t'\in\theta'} \|x-t'\|^2$ the centroid in θ' closest to x. Moreover, let us write $\tilde{c}' = \text{argmin}_{t'\in\theta'} \|c-t'\|^2$ the centroid in θ' closest to c. Note that c' and $\tilde{c}'$ are not necessarily equal. By definition of c', one has:

$$\|x-c'\| \leqslant \|x-\tilde{c}'\|,$$

such that:

$$\|x-c'\|_2 - \|x-c\|_2 \leqslant \|x-\tilde{c}'\|_2 - \|x-c\|_2 \leqslant \|\tilde{c}'-c\|_2 \leqslant d_\Theta(\theta,\theta').$$

Thus:

$$\begin{aligned} f(x,\theta') - f(x,\theta) = \|x-c'(x)\|^2 - \|x-c\|^2 &= (\|x-c'\| - \|x-c\|)(\|x-c'\| + \|x-c\|) \\ &\leqslant (\|x-c'\| + \|x-c\|)\ d_\Theta(\theta,\theta') \leqslant 2\rho\ d_\Theta(\theta,\theta'). \end{aligned}$$

- Finally, $N\sigma_{\mathbf{min}} \geqslant 1$, as shown by the second Lemma of Appendix B.

Given all these elements, Thm. 3.1 is thus applicable to the k-means setting and one obtains the desired result. □

Similarly, in the case of fixed-size DPP, Thm. 3.5 is applicable to the k-means problem, such that:

Corollary 4.2 (m-DPP for k-means)**.** *Let $\mathcal{S}$ be a sample from an m-DPP with marginal kernel K. Let $\epsilon, \delta \in (0,1)^2$. With probability at least $1-\delta$, $\mathcal{S}$ is a ϵ-coreset provided that:*

$$m \geqslant m^* = \frac{32}{\epsilon^2}\left(\max_i \frac{\sigma_i}{\bar{\pi}_i}\right)^2 \left(kd\log\left(\frac{24\rho^2}{\epsilon\langle f\rangle_{opt}}+1\right) + \log\frac{4}{\delta}\right), \tag{37}$$

with $\forall i, \bar{\pi}_i = \pi_i/m$.

If there exists $\alpha > 0$ and $\beta \geqslant 1$ such that:

$$\forall i, \qquad \alpha\sigma_i \leqslant \pi_i \leqslant \alpha\beta\sigma_i, \tag{38}$$

$$\text{and} \qquad \frac{\alpha}{\beta} \geqslant \frac{32}{\epsilon^2}\mathfrak{S}\left(kd\log\left(\frac{24\rho^2}{\epsilon\langle f\rangle_{opt}}+1\right) + \log\frac{4}{\delta}\right), \tag{39}$$

then $\mathcal{S}$ is a ϵ-coreset with probability larger than $1-\delta$. In this case, the number of samples verifies:

$$m \geqslant \frac{32}{\epsilon^2}\beta\mathfrak{S}^2\left(kd\log\left(\frac{24\rho^2}{\epsilon\langle f\rangle_{opt}}+1\right) + \log\frac{4}{\delta}\right).$$

Remark 4.3. *Remember from Appendix C that σ_{max} should not be too large in order to be able to find admissible values of α verifying Eqs. (35) and (36): it should in fact verify $\frac{1}{\sigma_{max}} \geqslant \mathcal{O}(kd\mathfrak{S})$. Thankfully, this constraint is often very loose. For instance, in the case of 1-means (k-means with $k=1$) and supposing without loss of generality that the data is centered (i.e., $\sum_j x_j = 0$), we show in the first lemma of Appendix B that $\forall i,\ \sigma_i = \frac{1}{N}\left(1 + \frac{\|x_i\|^2}{v}\right)$, where $v = \frac{1}{N}\sum_x \|x\|^2$. Thus, $\mathfrak{S} = 2$, and the constraint boils down to:*

$$\max_i \frac{\|x_i\|^2}{v} \leqslant \mathcal{O}\left(\frac{N}{d}\right) \tag{40}$$

Suppose for instance that the underlying data distribution is a Gaussian centered in the origin with variance ν. v is an estimator of ν, such that $\max_i \frac{\|x_i\|^2}{v}$ is the normalized norm of the most extreme event of the drawing and is typically smaller than 10 with very high probability, implying a very loose constraint indeed in our context of large N. If for any reason there are true outliers for which $\frac{\|x_i\|^2}{v}$ is larger than $\mathcal{O}(N/d)$, then, following the strategy outlined in Appendix C, one samples them beforehands, associates to each of them a weight of 1, and then applies the sampling theorems to the rest of the data.

5. Implementation for k-means

5.1. The DPP's ideal marginal kernel

Following the theorems' results, the ideal strategy (although unrealistic) to build the ideal marginal kernel K would be as follows. 1/ Deal with outliers as explained in Appendix C until $\frac{1}{\sigma_{\max}} \geqslant \frac{32}{\epsilon^2}(\epsilon + 4\mathfrak{S})\left(kd\log\left(\frac{24\rho^2}{\epsilon\langle f\rangle_{\text{opt}}}+1\right)+\log\frac{10}{\delta}\right)$. 2/ Compute all σ_i. 3/ Set $\alpha = \frac{32}{\epsilon^2}(\epsilon+4\mathfrak{S})\left(kd\log\left(\frac{24\rho^2}{\epsilon\langle f\rangle_{\text{opt}}}+1\right)+\log\frac{10}{\delta}\right)$ and $\beta = 1$. 4/ Set all $\pi_i = \mathsf{K}_{ii}$ to $\alpha\sigma_i$. 4/ Find all non-diagonal elements of K in order to minimize for all θ the estimator's variance, as derived in Eq. (33):

$$\text{Var}(\hat{L}) = \text{Var}_{iid} - \sum_{i\neq j}\frac{\mathsf{K}_{ij}^2}{\pi_i\pi_j}f(x_i,\theta)f(x_j,\theta) \tag{41}$$

while constraining K to be a valid marginal kernel, *i.e.*: SDP with $0 \preceq \mathsf{K} \preceq 1$, 5/ sample a DPP with kernel K. On our way to derive a practical algorithm with a linear complexity in N, many obstacles stand before us: there is no known polynomial algorithm to compute all σ_i in the general setting, solving exactly the minimization problem of step 4 under eigenvalue constraint is involved, and sampling from this engineered ideal K costs $\mathcal{O}(N^3)$ number of operations. Designing a linear-time algorithm that provably verifies under a controlled error the conditions of our previous theorems is out-of-scope of this paper. In the following, we prefer to first recall the intuitions behind the construction of a good kernel, and then discuss the choice of kernel we advocate.

5.2. In practice: a marginal kernel based on the Gaussian kernel

In order for K to be a good candidate for coresets, it needs to verify the following two properties:

- As indicated by the theorems, the diagonal entries K_{ii} should increase as the associated σ_i increases.
- As indicated by the variance equation, off-diagonal elements should be as large as possible (in absolute value) given the eigenvalue constraints. In fact, we cannot set all non-diagonal entries of K to large values as the matrix's 2-norm would rapidly be larger than 1. We thus need to choose the best pairs (i,j) for which it is worth setting a large value of K_{ij}. A first glance at the variance equation indicates that the larger $f(x_i,\theta)f(x_j,\theta)$ is, the larger K_{ij} should be, in order to decrease the variance as much as possible. Recall nevertheless that in the coreset setting, all sampling parameters should be independent of θ. The off-diagonal elements should thus verify the following property: the larger is the correlation between x_i and x_j (the more similar are $f(x_i,\theta)$ and $f(x_j,\theta)$ for all θ), the larger K_{ij} should be.

We show in the following in what ways the choice of marginal kernel

$$\mathsf{K} = \mathsf{L}(\mathsf{I}+\mathsf{L})^{-1}$$

with L the Gaussian kernel matrix with parameter s:

$$\forall(i,j)\quad \mathsf{L}_{ij} = \exp^{-\frac{\|x_i-x_j\|^2}{s^2}},$$

is a good candidate to build coresets for k-means. Note that L is called the L-ensemble associated to K [8]. Let us write $\mathsf{U} = (\boldsymbol{u}_1|\ldots|\boldsymbol{u}_N)$ the orthonormal eigenvector basis of L and $\Lambda = \text{diag}(\lambda_1|\ldots|\lambda_N)$ its diagonal matrix of sorted eigenvalues, $0 \leqslant \lambda_1 \leqslant \ldots \leqslant \lambda_N$. U and Λ naturally depend on s. One shows for instance that, with respect to s, λ_N is a monotonically increasing function between 1 and N.

Concerning the off-diagonal elements of K, let us first note that if x_i and x_j are correlated (that is, in the k-means setting, if they are close to each other), then

$$\mathsf{K}_{ij} = \sum_k \frac{\lambda_k}{1+\lambda_k}u_k(i)u_k(j)$$

is large in absolute value. In fact, in the limit where $x_i = x_j$, then $\forall k, u_k(i) = u_k(j)$ and $\mathsf{K}_{ij} = \mathsf{K}_{ii} = \mathsf{K}_{jj}$. The determinant of the 2×2 submatrix of K indexed by i and j is therefore null: sampling both

will never occur. Thus, the closer are x_i and x_j, the lower is the chance of sampling both jointly. Moreover, if x_i and x_j are far from each other (for instance, in different clusters), then the entries i and j of $\mathsf{L}'s$ eigenvectors will be very different. Fo instance, say the dataset contains two well separated clusters of similar size. If the Gaussian parameter s is set to the size of these clusters, then the kernel matrix L will be quasi-block diagonal, with each block corresponding to the entries of each cluster. Also, each eigenvector $\boldsymbol{u}_k$ will have energy either in one cluster or the other such that K_{ij} is necessarily small if i and j belong to different clusters, and the event of sampling both jointly is probable.

Concerning the probability of inclusion of i, we have:

$$\mathsf{K}_{ii} = \sum_k \frac{\lambda_k}{1+\lambda_k} q_i(k)^2,$$

where $\boldsymbol{q}_i$ is the vector of size N verifying $\forall k, q_i(k) = u_k(i)$. For all i, $\|\boldsymbol{q}_i\|^2 = 1$. The probability of inclusion is thus directly linked to the values of k that contain the energy of $\boldsymbol{q}_i$: the more the energy of $\boldsymbol{q}_i$ is contained on high values of k, the larger is the probability of inclusion. Say we are again in a situation where the clusters and the choice of Gaussian parameter s are such that L is quasi block diagonal. Within each block, the eigenvector associated with the highest eigenvalue corresponds approximately to the constant vector. These eigenvectors being normalized, the associated entry of $q_i(k)$ is thus approximately equal to $1/\sqrt{\#C_i}$ where $\#C_i$ is the size of the cluster containing data x_i. Typically, if the cluster is small, that is, if $\#C_i$ tends to 1, the associated entry $q_i(k)$ tends to 1 as well, such that all the energy of q_i is drawn towards high values of k, thus increasing the probability of inclusion of i. In other words, the more isolated, the higher the chance of being sampled. This corresponds to the intuition one may obtain for the sensitivity σ_i. It has indeed been shown that the sensitivity may be interpreted as a measure of outlierness [26].

In the context of coresets for k-means, we thus advocate to sample DPPs via a Gaussian kernel L-ensemble. We now move on to detailing an efficient sampling implementation.

5.3. Efficient implementation

Sampling a DPP from the Gaussian L-ensemble verifying

$$\forall (i,j) \qquad \mathsf{L}_{ij} = \exp^{-\frac{\|x_i - x_j\|^2}{s^2}}$$

consists in the following steps:

1) Compute L.
2) Diagonalize L in its set of eigenvectors $\{\boldsymbol{u}_k\}$ and eigenvalues $\{\lambda_k\}$.
3) Sample a DPP given $\{\boldsymbol{u}_k\}$ and $\{\lambda_k\}$ via Alg. 1 of [8].

Step 1 costs $\mathcal{O}(N^2 d)$, step 2 costs $\mathcal{O}(N^3)$, step 3 costs $\mathcal{O}(N\mu^3)$. This naive approach is thus not practical. We detail in Appendix D how to reduce the overall complexity to $\mathcal{O}(N\mu^2)$, by 1/ taking advantage of Random Fourier Features (RFF) [33] to estimate a low dimensional representation $\Psi \in \mathbb{R}^{2r \times N}$ of the L-ensemble $\mathsf{L} \simeq \Psi^\intercal \Psi$, where r is the chosen number of features; and 2/ running a DPP sampling algorithm adapted to such a low rank representation.

In the experimental section, we will concentrate on m-DPPs as they are simpler to compare with state of the art methods that all have a fixed known-in-advance number of samples. The overall m-DPP sampling algorithm adapted to the k-means problem that we will consider is summarized in Alg. 1: given the data $\mathcal{X}$, the number of desired samples m, and the Gaussian parameter s, it outputs a weighted set of m samples $\mathcal{S}$ that is a good candidate to be a coreset if m is large enough. The runtime to build Ψ is $\mathcal{O}(Ndr)$; to compute C and diagonalize it is $\mathcal{O}(Nr^2)$; to sample a m-DPP given this dual eigendecomposition is $\mathcal{O}(Nm^2)$. Given that r is set to a few times m and that m is necessarily larger than d in order to obtain coresets for k-means, the overall runtime of Alg. 1 is $\mathcal{O}(Nm^2)$.

Given a number of samples m to draw, how should one set the Gaussian parameter s? The larger is s, the more repulsive is the m-DPP, and the smaller is the numerical rank of Ψ (the number of eigenvalues ν such that $N\nu$ is larger than the machine's precision). Now, numerical instabilities arise

Algorithm 1 The overall coreset sampling heuristics for k-means

Input: $\mathcal{X} = \{x_i\}$ a set of N points in $\mathbb{R}^d$, the Gaussian kernel parameter s, the number of samples m
- Draw $r \geqslant \mathcal{O}(m)$ random Fourier vectors associated to the Gaussian kernel with parameter s
- Compute the associated RFF matrix $\Psi \in \mathbb{R}^{2r \times N}$ as explained in Appendix D.1
- Compute $\mathsf{C} = \Psi\Psi^\intercal \in \mathbb{R}^{2r \times 2r}$ the dual representation
- Compute the eigendecomposition of C: obtain eigenvectors $\{\boldsymbol{v}_k\}$ and eigenvalues $\{\nu_k\}$
- Draw a sample $\mathcal{S}$ from a m-DPP with L-ensemble $\mathsf{L} = \Psi^\intercal\Psi$ as explained in Appendix D.3.
- Compute the marginal probabilities π_s for $s \in \mathcal{S}$ and set weights $\omega(s) = 1/\pi_s$, as in Appendix D.3.

Output: $\{\mathcal{S}, \omega\}$ a weighted sample of size m.

while sampling an m-DPP if the numerical rank of Ψ decreases below m: s should not be set too large. Also, the smaller is s, the closer is L to the identity matrix, such that the closer is the m-DPP to uniform sampling without replacement: s should not be set too small. We will see in the following experimental section how the choice of s affects results.

6. Experiments

6.1. Different strategies to compare...

We will empirically compare results obtained with the four following approaches:

1) `m-DPP` : The strategy summarized in Alg. 1.
2) `matched iid` : An iid sampling strategy with replacement, matched to `m-DPP`. More precisely, m samples are drawn iid with replacement, the probability of selecting x_i at each draw being set to $p_i = \pi_i/m$, where π_i is the marginal probability of drawing x_i in `m-DPP`.
3) `uniform iid` : Uniform iid sampling with replacement.
4) `sensitivity iid` : The current state of the art iid sampling based on a bi-criteria approximation to upper bound the sensitivity (Alg. 2 of [16]), or, if available (for instance in the case of 1-means as in Section 6.2.1), an analytical formula of the sensitivity.

For the three iid methods (methods 2, 3 and 4), we will use the importance sampling estimator adapted to iid sampling of Eq. (9). For method 1, we will use the importance sampling estimator adapted to correlated sampling of Eq. (16).

Empirically, when the ambient dimension d is small, performance of all methods is enhanced if the weights in $\hat{\mathsf{L}}$ are set via Voronoi cells rather than set to inverse probabilities: given the sample $\mathcal{S}$ of size m, compute its Voronoi tessellation in m cells, and associate to each sample s a weight $\omega(s)$ equal to the number of datapoints in its associated Voronoi cell. We will call the associated cost estimators $\hat{L}$ the Voronoi estimators.

For completeness, we compare all these methods with another negatively correlated sampling method called D^2-sampling (commonly used for k-means++ seeding [34]):

5) `D`^{2} : sample the first element of $\mathcal{S}$ uniformly at random. Each subsequent element of $\mathcal{S}$ is drawn according to a probability proportional to the squared distance to the closest of the already sampled elements. The marginal probabilities are not known in this algorithm, so we will only be able to build the associated Voronoi cost estimator.

To measure the performance of each method, we will empirically estimate the probability that, given the method's sampled weighted subset, it verifies the coreset property of Eq. 6 for a given randomly chosen θ (setting ϵ to 0.1). On the artificial data models we investigate, we estimate this probability via 50 randomly chosen θ on 1000 realizations of the data. On the real-world datasets, we estimate this probability via 5000 randomly chosen θ. We will in general plot this probability versus the number of samples: the closer it is to 1, the better the sampling method for coresets.

In Sections 6.2.2 and 6.2.3, we will not only compare the coreset property of the samples obtained by each method, we will also compare the result of Lloyd's classical k-means heuristics [29] performed on the entire data versus the result obtained on the weighted samples of each method. To be precise,

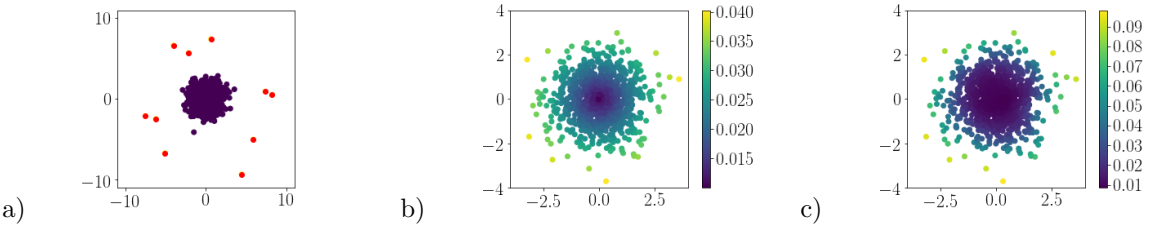

Figure 1. *a) A realization of an artificial dataset of $N = 1000$ data points; blue points are drawn from an isotropic Gaussian, a proportion $q = 0.01$ of the points are drawn as outliers (displayed in red). b) In a case without outliers, and for $m = 20$, we represent the inverse importance sampling weights of* `sensitivity iid`*, i.e.: $m\sigma_i/\mathfrak{S}$. c) On the same data realization, and also setting $m = 20$, we represent the inverse importance sampling weights of* `m-DPP`*: the inclusion probability π_i.*

once the k-means heuristics on the weighted subset outputs k centroids, we classify all nodes (sampled or not) according to their closest distance to the centroids: this gives us a partition that we then compare using the Adjusted Rand (AR) similarity index [35] to the ground truth associated to the dataset. The AR index is a number between -1 and 1: the closer it is to 1, the closer are the partitions, the better the sampling method.

6.2. ...on different datasets

6.2.1. To start with: a well controlled case

We start with a perfectly controlled case: the 1-means case, for which we show in the first lemma of Appendix B that, supposing without loss of generality that the data is centered ($\sum_j x_j = 0$), the sensitivity verifies the following analytic form:

$$\sigma_i = \frac{1}{N}\left(1 + \frac{\|x_i\|^2}{v}\right), \tag{42}$$

where $v = \frac{1}{N}\sum_{x\in\mathcal{X}} \|x\|^2$. We are thus able to compare our method versus the ideal iid sampling scheme for which we set p_i, the probability of drawing x_i, exactly to its ideal value given in Thm 2.3: $p_i = \sigma_i/\mathfrak{S} = \sigma_i/2$.

We will work on a simple isotropic Gaussian dataset of $N = 1000$ points in dimension $d = 2, 20$ or 100. A percentage q of the N points are drawn as outliers (uniformly in the ambient space and far from the Gaussian mean). An instance of such a dataset in $d = 2$ dimensions, and with $q = 0.01$ is shown in Fig. 1a.

We start by showing in Fig. 2 the results of `m-DPP` versus the number of dimensions and the choice of parameter s for the Gaussian kernel. All shown results are with $q = 0$ (no outlier) and with a number of random Fourier features $r = 200$. Several comments are in order. Firstly, compared to the importance sampling estimator, the Voronoi estimator produces good results in low dimensions, and fails as the dimension increases. Secondly, the performance of all methods increase and uniformize as the dimension increases. This is due to the fact that in large dimensions, interpoint distances tend to uniformize such that any pair of points tend to be representative of all interpoint distances, thus simplifying the problem of finding good coresets. This may also explain why the choice of s is less crucial in higher dimension. In low dimensions, however, the choice of s has a strong impact on performance. The best choice for s depends in fact on the number of samples m one requires: as m increases, s should be set smaller. This is in fact natural: if one desires a very short summary of the dataset (small m), the repulsion of the DPP has to be strong in order to sample a diverse subset. Whereas if the length of the summary is less constrained, s should be decreased to allow for a less coarse-grained description. This observation leads to the natural question of the optimal s given the data and m. We currently lack of a satisfying answer to this question, both theoretically and empirically. A usual heuristics in kernel methods is to set s to the average (or median) interdistance of the points in the

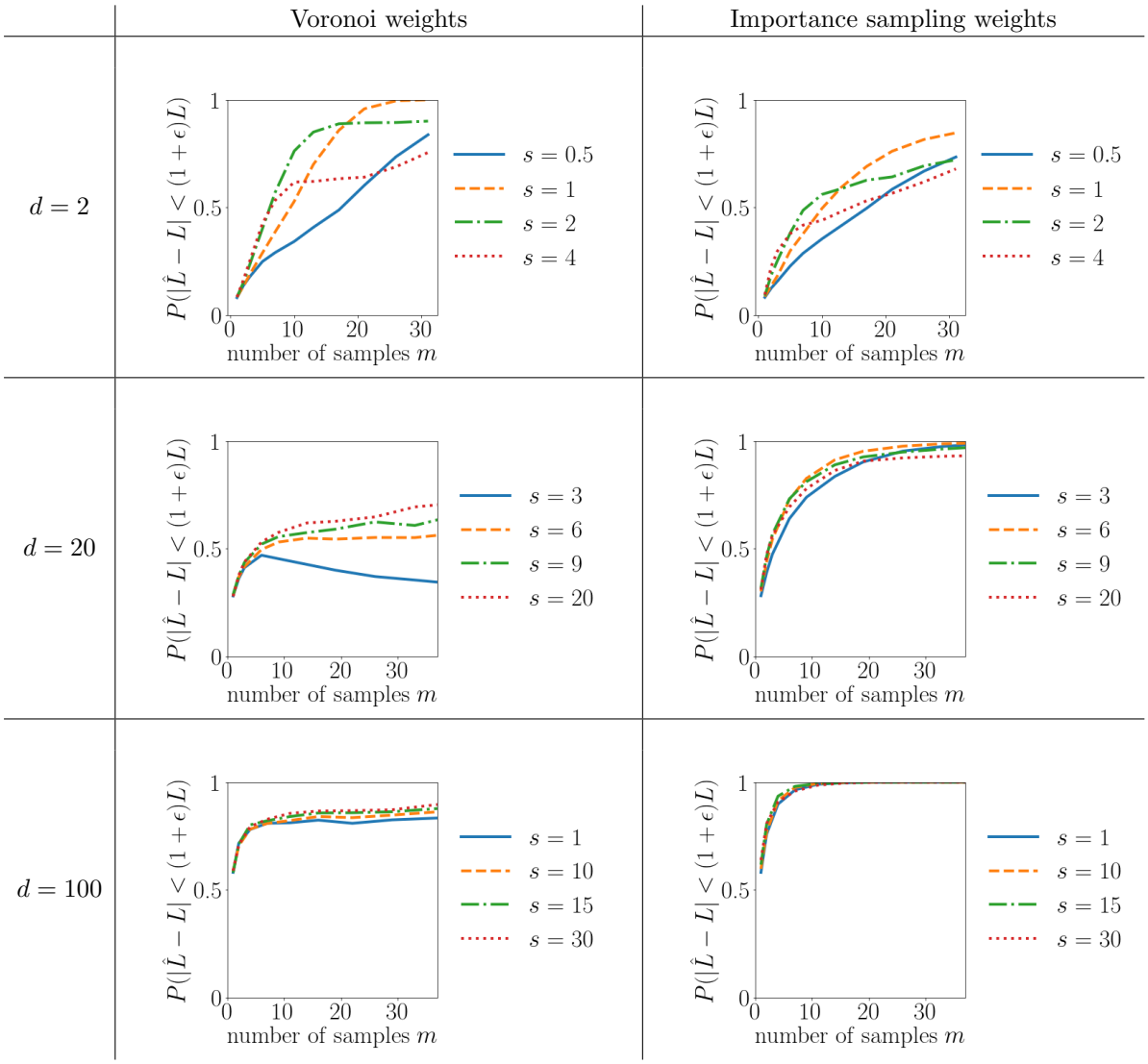

Figure 2. *Performance of* `m-DPP` *on the* 1*-means problem, versus the dimension* d*, the parameter* s *of the Gaussian kernel and the choice of weights (Voronoi or importance sampling weights) in the cost estimator.*

dataset. In the experiments of Fig. 1, the average interdistance corresponds to $s \simeq 1.5, 6.3$ and 14.0 for $d = 2, 20$ and 100 respectively, which give in fact a good order of magnitude for the choice of s. In the following, to simplify the discussion, we will sometimes set s to be the average interdistance, that we will denote by $\bar{s}$.

We pursue by comparing the performance of several methods in Fig. 3. One observes that the superior performance of the Voronoi estimator over the importance sampling estimator in low dimension d is verified for all methods. Moreover, as the dimension increases, all methods' performance converge to the performance of the uniform iid sampling method. Finally, `m-DPP` associated with Voronoi weights is competitive with $\mathtt{D}^2$ in low d; and, regardless of how one chooses the weights, our method has a clear edge over the sensitivity-based iid random sampling (the lower the dimension, the clearer the edge).

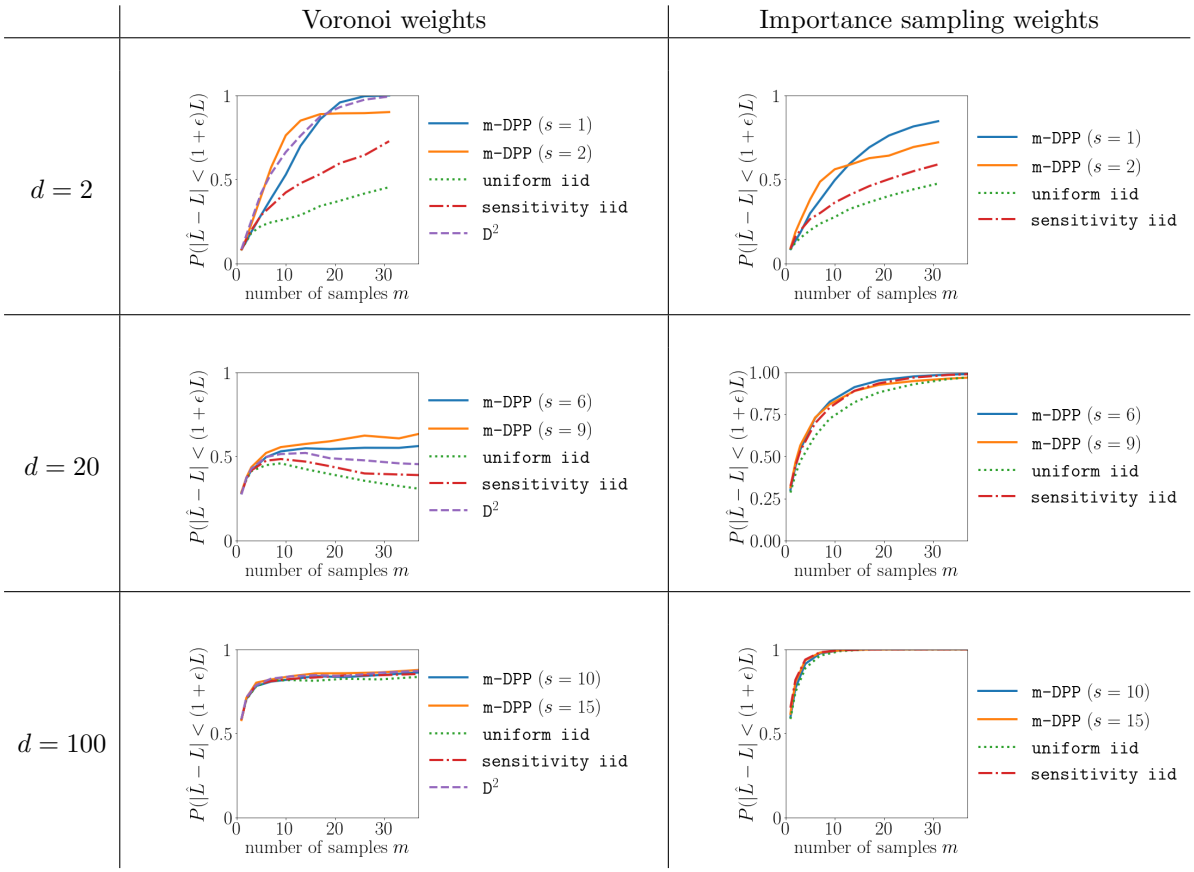

Figure 3. *Performance comparison of different sampling methods on the* 1*-means problem, versus the dimension* d *and the choice of weights (Voronoi or importance sampling weights) in the cost estimator.*

In order to clarify further discussion, we will from now on only discuss the importance sampling estimated cost. One should keep in mind that in low dimensions, Voronoi-based estimated costs usually perform well, but fail (sometimes drastically) as the dimension increases.

A natural question arises at this point: is the observed edge of `m-DPP` over `sensitivity iid` due to a better probability of inclusion of the point process? Or is it truly due to the negative correlations induced by the determinantal nature of our method? In fact, we compare in Fig. 1b and c the probability of inclusion for `sensitivity iid` versus `m-DPP`: they have a similar general behavior but are nevertheless quantitatively different. In Fig. 4, we compare `m-DPP` versus `matched iid` and `sensitivity iid`: the observed edge is clearly due to the negative correlations induced by the determinantal nature of our method. As expected from Corollary 4.2, the best inclusion probability is based on the sensitivity. Nevertheless, the figure shows that even if it is not set to its ideal value, one can still improve the performance by inducing negative correlations.

To be complete, we still need to discuss the impact of two variables: the number of random Fourier features r used in our method, and the percentage of outliers q in the data. In the following, we set s to $\bar{s}$, the average interdistance. Fig. 5 shows the impact of the choice of r on performances: as expected, as r increases, performance increases, and as d increases, performances become more sensitive to the choice of r. The impact of the choice of r is nevertheless very limited: setting r to a multiple of m has been a safe choice in all our experiments. Finally, Fig. 6 shows the impact of the percentage of outliers q on performances. Empirically, we see here that outliers have a smaller impact

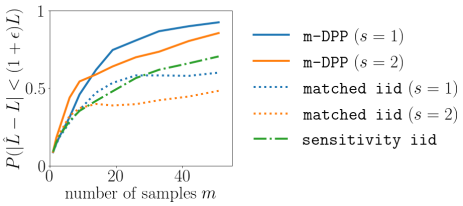

Figure 4. *Performance comparison on the* 1*-means problem. We compare* `m-DPP` *versus* `matched iid` *and* `sensitivity iid`*.*

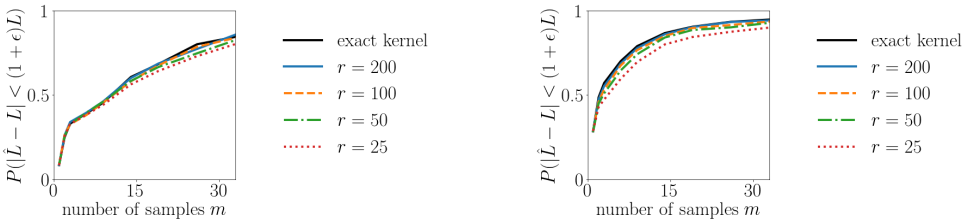

Figure 5. *Performance comparison of* `m-DPP` *on the* 1*-means problem versus the number of RFF* r*, for* $d=2$ *(left) and* $d=20$ *(right). For readability's sake, the* y*-axis of both figures are in logarithmic scale.*

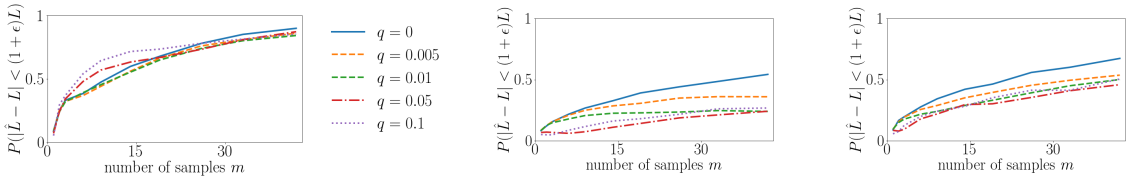

Figure 6. *Performance comparison of* `m-DPP` *(left),* `uniform iid` *(middle) and* `senisitivity iid` *(right) on the* 1*-means problem versus the percentage of outliers* q*.*

on DPP sampling than on uniform or sensitivity-based iid sampling.

We conclude this first well-controlled experimental section by summarizing the observed behaviors:

- `m-DPP` outperforms the current state of the art `sensitivity iid`, even in the 1-means case, where sensitivities do not need to be estimated but may be computed exactly.
- As the dimension increases, the edge over iid sampling decreases.
- The best choice of parameter s of the Gaussian kernel in our method is still an open problem. Empirically, a good order of magnitude is the average interdistance of the datapoints. Ideally, nevertheless, s should increase as m, the number of wanted samples, decreases.
- Regarding the number of RFFs r, setting r to a few times m is sufficient.
- Regarding the impact of outliers. Our theorems are not well suited to outliers (due to the proof techniques used); nevertheless, in practice, we see that outliers are not an issue in our method: they even have a smaller impact on our method's performances than on other methods.
- Replacing weights by Voronoi weights yields in general better results, but only in low dimension. As the dimension increases, the Voronoi cost estimator fails (sometimes drastically).

6.2.2. Experiments on non-Gaussian data: the case of spectral features

Spectral features. Given a graph of N nodes where $\mathsf{W} \in \mathbb{R}^{N\times N}$ is the adjacency matrix (*i.e.*, $\mathsf{W}_{ij}=1$ if nodes i and j are connected, and 0 ortherwise), a standard problem consists in partitioning the nodes in k communities, *i.e.*, sets of nodes more connected to themselves than to other nodes of

Figure 7. *Examples of SBM spectral features $\boldsymbol{x}_i$, here with $k = 2$. Each colour corresponds to one block of the SBM. On the left, for an "easy" classification task ($\zeta = \zeta_c/4$), and on the right, for a harder setting ($\zeta = \zeta_c/2$).*

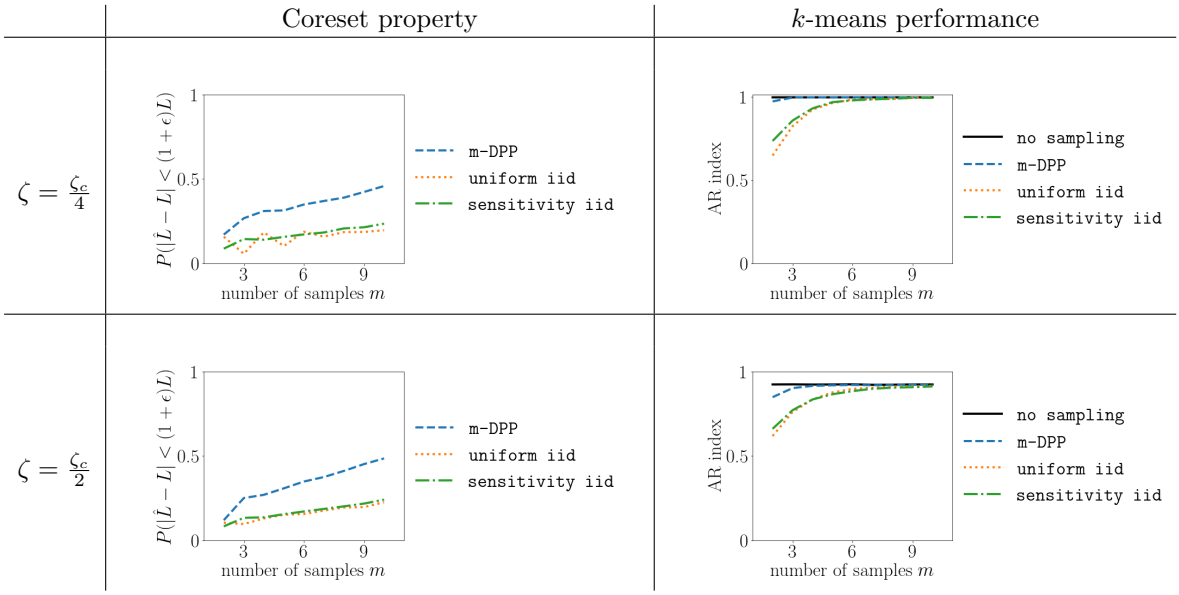

Figure 8. *Performance comparison of different methods on the k-means problem for spectral features of balanced SBM graphs (here, $k = 2$). Left: testing the coreset property. Right: the Adjusted Rand index between the partition recovered by k-means on the weighted subsets and the ground truth partition of the SBM. ζ quantifies the difficulty of the classification task (see text): the lower it is, the easier the classification task.*

the graph [36]. A classical algorithm to solve efficiently this problem is the so-called spectral clustering algorithm [37]:

- Define the normalized Laplacian matrix $\mathsf{L} = \mathsf{I} - \mathsf{D}^{-\frac{1}{2}}\mathsf{W}\mathsf{D}^{-\frac{1}{2}} \in \mathbb{R}^{N\times N}$ where I is here the identity matrix in dimension N, and $\mathsf{D} \in \mathbb{R}^{N\times N}$ is a diagonal matrix with $\mathsf{D}_{ii} = d_i = \sum_j \mathsf{W}_{ij}$ the degree of node i.
- Compute via Arnoldi iterations or a similar algorithm the k first eigenvectors of L: $(\boldsymbol{u}_1, \ldots, \boldsymbol{u}_k)$.
- Associate to each node i a (spectral) feature vector $\boldsymbol{x}_i \in \mathbb{R}^k$: $\forall l = 1, \ldots, k \quad x_i(l) = u_l(i)$.
- Normalize all feature vectors: $\boldsymbol{x}_i \leftarrow \boldsymbol{x}_i / \|\boldsymbol{x}_i\|_2$.
- Run k-means on all such normalized spectral features.

An extensive literature exists on spectral clustering and it has shown to be a very successful unsupervised classification algorithm in many situations [38].

The Stochastic Block Model (SBM). We consider random community-structured graphs drawn from the SBM, a classical class of structured random graphs (see for instance [39]). We first look at graphs with k communities of same size N/k. In the SBM, the probability of connection between any two nodes i and j is q_1 if they are in the same community, and q_2 otherwise. One can show that

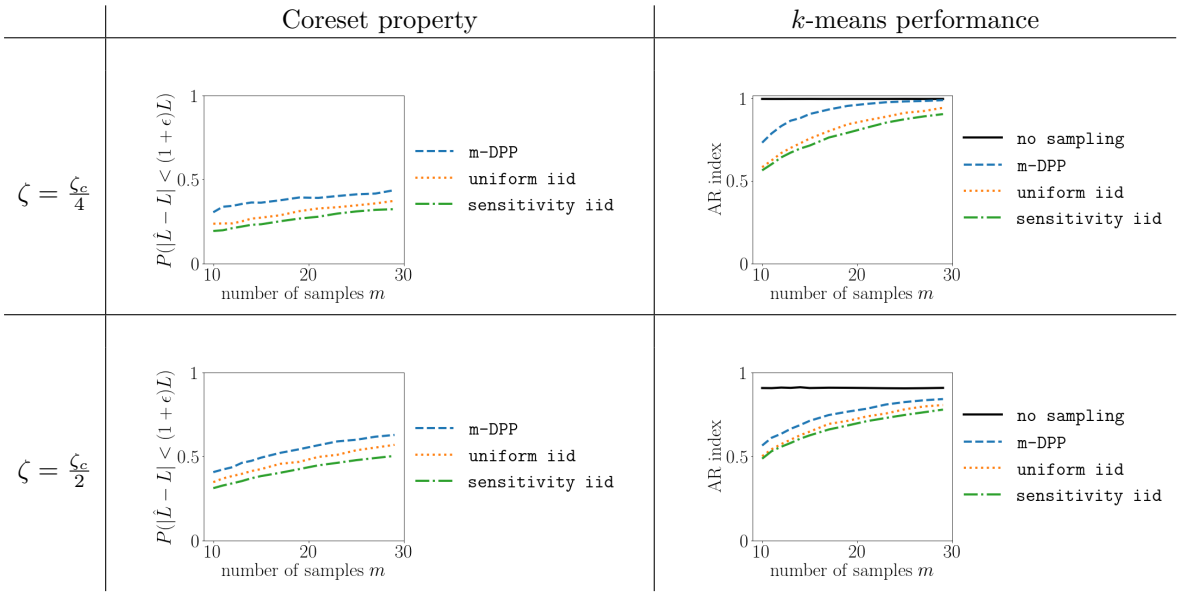

Figure 9. *Same as Fig. 8 but with* $k = 10$.

the average degree reads $c = q_1\left(\frac{N}{k} - 1\right) + q_2\left(N - \frac{N}{k}\right)$. Thus, instead of providing the probabilities (q_1, q_2), one may characterize a SBM by considering $(\zeta = \frac{q_2}{q_1}, c)$. The larger ζ, the fuzzier the community structure, the harder the classification task. In fact, authors in [40] show that above the critical value $\zeta_c = (c - \sqrt{c})/(c + \sqrt{c}(k-1))$, community structure becomes undetectable in the large N limit. In the following, we set $N = 1000$ and $c = 16$; k and ζ will vary. Note that spectral features $\boldsymbol{x}_i$ are not Gaussian and, in fact, do not fall into any classical data model (see Fig. 7 to visualize instances of SBM spectral features with $k = 2$). They are thus interesting candidates to test k-means algorithms.

Results. For different values of ζ and k, we generate 1000 such SBM graphs from which we sample subsets according to different methods. We test both the coreset property (as before) and the k-means performance on the weighted subset compared to the k-means performed on all data. We plot in Fig. 8 (resp. Fig. 9) the results obtained for $k = 2$ (resp. $k = 10$). Note that in this case, we have no explicit formula for the sensitivity such that for `sensitivity iid`, we use the bi-criteria approximation scheme provided in [16] (Alg. 2). Here again, we see how our method outperforms iid sampling schemes, even in difficult classification contexts (for instance when $\zeta = \zeta_c/2$: even with all the data, k-means' performance saturates at an AR index of 0.9). Moreover, as the dimension increases (here $d = k$), performances of all methods tend to uniformize. Surprisingly, `uniform iid` performs as well ($k = 2$) and even outperforms ($k = 10$) `sensitivity iid`. We believe this is due to approximation errors of the bi-criteria scheme used to find upper bounds of the sensitivity. Also, in this balanced case (communities have the same number of nodes), uniform sampling is in fact a good option. We will now see how this changes in the unbalanced case.

The unbalanced case. In the unbalanced case, ζ_c is no longer a recovery threshold, but we may still use ζ as a marker of difficulty of the recovery task. We set ζ to $\zeta_c/4$ and perform the same experiments as previously with $k = 2$ blocks of unbalanced size. Results are shown in Fig. 10. For a fixed ζ, the more unbalanced, the more difficult the recovery task. Also, the more unbalanced, the better is `sensitivity iid` compared to `uniform iid`. Nevertheless, `m-DPP` shows an edge over all iid methods in all tested configurations.

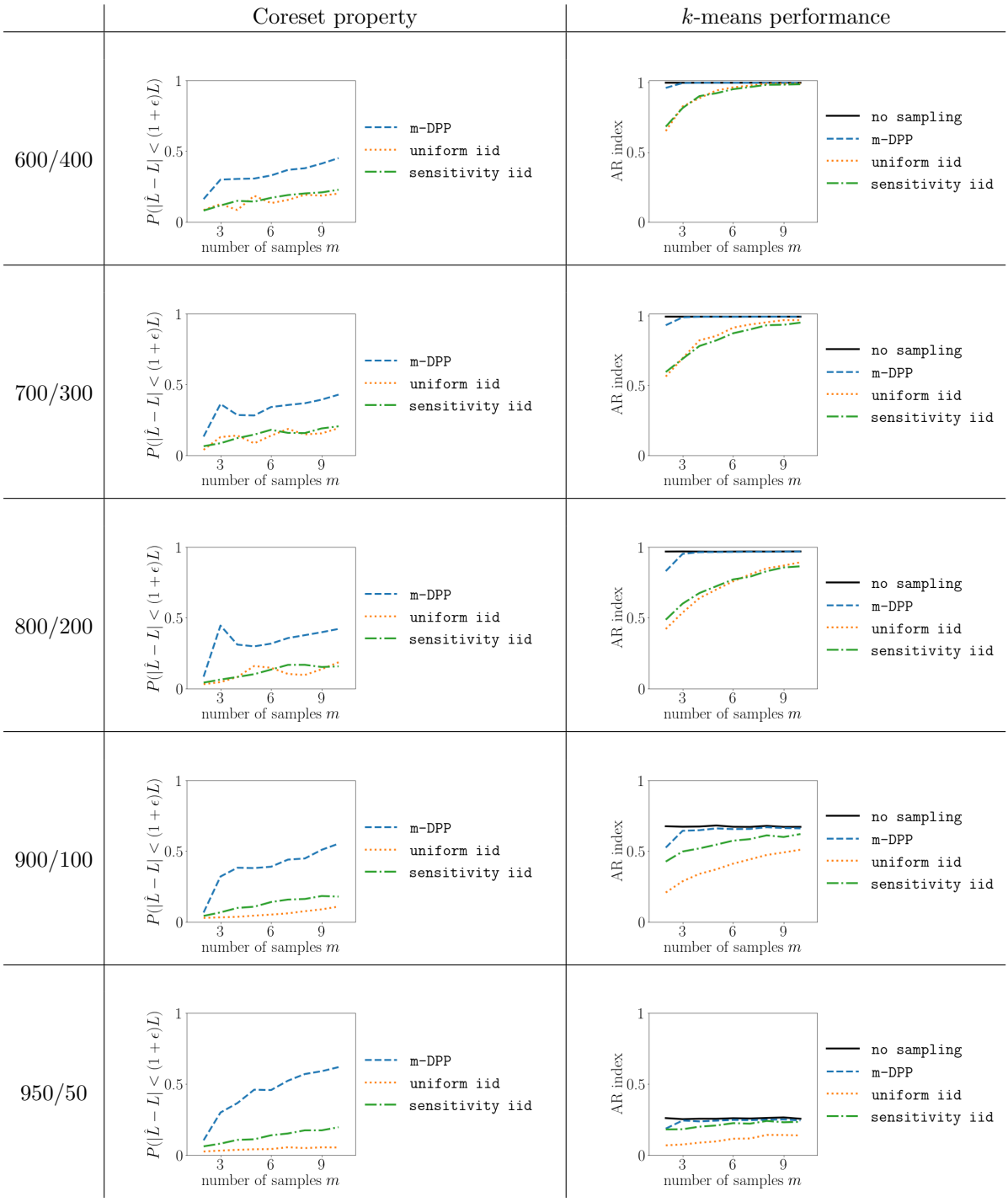

Figure 10. *Same as Fig. 8 but with a fixed $\zeta = \zeta_c/4$ and a varying level of balance within the sizes of the $k = 2$ communities. N_1/N_2 means one community with N_1 nodes and the other with N_2 nodes.*

6.2.3. Experiments on two real world datasets

The MNIST dataset. We perform a first experiment on the MNIST dataset [41] that consists in $7 \cdot 10^4$ images of handwritten digits (from 0 to 9) for which the ground truth is known. The classical

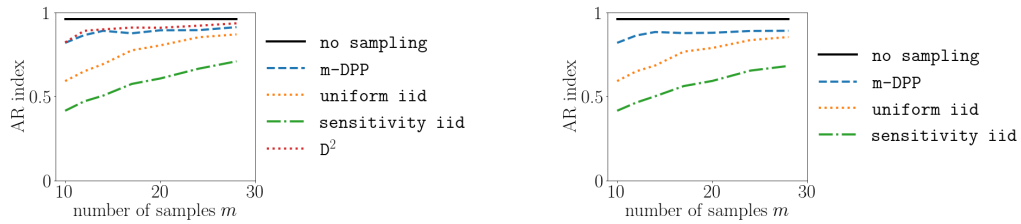

Figure 11. *Classification performance on the MNIST dataset obtained with different sampling methods versus the result obtained without sampling, using Voronoi weights (left), or importance sampling weights (right).*

associated machine learning goal is to classify them in 10 classes (one for each digit). To do so, we pre-process the data in the following unsupervised way. We consider all images and extract SIFT descriptors [42] for each image. We then use FLANN [43] to compute a κ-nearest neighbor graph (with κ=10) based on these descriptors. We finally run the spectral clustering algorithm with $k = 10$ to find the 10 classes corresponding to each digit, as explained in Section 6.2.2. The k-means step is thus the last step of the overall processing. We compare results obtained with different sampling methods versus the results obtained without sampling in Fig. 11 (bottom). For `m-DPP`, several values of s were tried, and we show here the result obtained for $s = 2.2$. Also, a number $r = 200$ of Fourier features were used. We see that, in the Voronoi weight setting, `m-DPP` is competitive with `D`^{2}. Moreover, `uniform iid` outperforms `sensitivity iid` certainly due to approximation errors of the bi-criteria procedure and to the fact that the data is balanced (there are more or less $7 \cdot 10^3$ instances of each digit in the dataset), thus favoring uniform sampling. Finally, `m-DPP` outperforms once again the iid random sampling techniques. Note that the methods' classification performance is remarkable. Without sampling, the overall classification performance in terms of AR index with the ground truth is 0.95. With only ~ 20 samples, `m-DPP` reaches a performance of ~ 0.9!

The US Census dataset. We also perform experiments on the 1990 US Census dataset[4], that consists in $N = 2458285$ surveyed person, and $d = 68$ categorical attributes such as age, income, etc. The data was pre-processed by a series of operation detailed on its download webpage. In our experiments, and in order to limit memory usage, we perform experiments on the first $N = 5 \cdot 10^5$ instances of the data. As there is no ground truth in this dataset to compare to, we arbitrarily decide $k = 15$ classes, and show solely the coreset property of the samples obtained via different methods. For `m-DPP`, s was set to 70 (the mean interdistance estimated on 1000 randomly chosen pairs of datapoints), and a number $r = 25$ of Fourier features was chosen. Experiments were done with s ranging from $s = 30$ to $s = 140$ with no qualitative change in performance (not shown). Fig. 12 shows the results of the experiments. We see that `m-DPP` outperforms all other methods, in both Voronoi and importance sampling settings. In this example, note that `sensitivity iid` outperforms `uniform iid` probably due to the fact that the 15 potential classes are unbalanced.

7. Conclusion

In this work, we introduced a new random sampling method based on DPPs to build coresets. Different from sensitivity-based iid random sampling, our method introduces negative correlations between samples due to its determinantal nature. Also, different from D^2 sampling, also known to be repulsive, our method is tractable in the sense that marginal probabilities are known and importance sampling schemes can be used. Our theoretical results may be summarized in two points. Firstly, Thms 3.1 and 3.5 provide coreset guarantees in function of the point process' probabilities of inclusion, that is: the diagonal elements of the marginal kernel K parametrizing the DPP. These guarantees are not stronger than the iid case and are in fact similar: they both show that the ideal marginal probabilities are proportional to the sensitivity. Nevertheless, these results do not take into account the off-diagonal

[4]downloaded from `https://archive.ics.uci.edu/ml/datasets/US+Census+Data+(1990)`

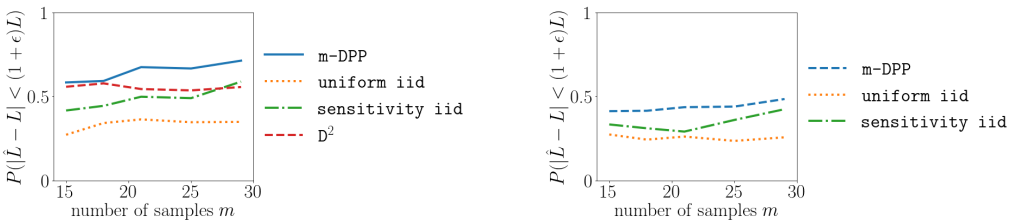

Figure 12. *Performance of different sampling methods on the US Census dataset. Left: performance using Voronoi weights. Right: performance using importance sampling weights.*

elements of K coding for the repulsion within the sampled subsets and are in fact verified for any choice of off-diagonal elements (provided K stays SDP with eigenvalues between 0 and 1). This leads to the second point: given that these off-diagonal elements offer extra degrees of freedom and due to a simple variance argument (Thm 3.6), we show that DPP-based random sampling will necessarily provide better performance than its iid counterpart. On the theoretical side, additional work is required to specify precisely the minimum number of required samples guaranteeing the coreset property. We expect that further research on concentration properties of DPPs, involving not only the diagonal elements of K but also its off-diagonal elements, should enable to move forward in this direction.

We applied our general coreset theorems to the ubiquitous k-means problem. Given a dataset, the ideal marginal kernel K adapted to the k-means problem is untractable and we thus propose a heuristics via random Fourier features and the Gaussian kernel in order to efficiently sample a DPP that has the desirable properties to sample coresets (if not provably, at least quantitatively). To sample a subset of size m, our heuristics runs in $\mathcal{O}(Nm^2)$. This is more expensive than the sensitivity-based iid strategy (that runs in $\mathcal{O}(Ndm)$), especially as the number of samples m increases; but empirically provides better results both regarding the coreset property and the k-means performance in classification tasks, on different artificial and real-world datasets.

. Acknowledgments

This work was partly funded by the ANR GenGP (ANR-16-CE23-0008), the LabEx PERSYVAL-Lab (ANR-11-LABX-0025-01), the CNRS PEPS I3A (Project RW4SPEC), the Grenoble Data Institute (ANR-15-IDEX-02) and the LIA CNRS/Melbourne Univ Geodesic.

Appendix A - Proof of Theorem 3.1

Proof. The theorem consists in proving that Eq. (6) is true. We follow a classical proof scheme from compressed sensing [44], in four steps:

1) we first use concentration arguments for a given $\theta \in \Theta$.
2) we then build an ϵ-net paving the space of parameters.
3) via the union bound, we obtain the result for all θ in the ϵ-net.
4) via the Lipschitz property of f, we obtain the desired result for all $\theta \in \Theta$.

Step 1 (Concentration around $\theta \in \Theta$) For a given $\theta \in \Theta$, we have the following concentration result [30]: $\forall \epsilon \in (0,1), \forall \delta \in (0,1)$:

$$\mathbb{P}\left(\left|\frac{\hat{L}}{L} - 1\right| \geqslant \epsilon\right) = \mathbb{P}\left(\left|\hat{L} - L\right| \geqslant \epsilon L\right) \leqslant \delta, \tag{43}$$

provided that:

$$\mu \geqslant \frac{16}{\epsilon^2}\left(\epsilon C + 2C^2\right) \log \frac{5}{\delta}, \tag{44}$$

with $C = \max_i \frac{f(x_i,\theta)}{L\bar{\pi}_i}$, where $\bar{\pi}_i$ is a shorthand for π_i/μ.

Using the same concentration results, we also have:

$$\forall(\epsilon,\delta) \in (0,1)^2,\ \mathbb{P}\left(\left|\frac{\sum_i \frac{\epsilon_i}{\pi_i}}{N} - 1\right| \geqslant \epsilon\right) \leqslant \delta, \tag{45}$$

provided that:

$$\mu \geqslant \frac{16}{\epsilon^2 N \bar{\pi}_{\min}}\left(\epsilon + \frac{2}{N\bar{\pi}_{\min}}\right)\log\frac{5}{\delta}, \tag{46}$$

where $\bar{\pi}_{\min} = \min_i \bar{\pi}_i$.

Step 2 (ϵ'-net of Θ) Consider $\Gamma_{\epsilon'} = (\theta_1^*, \dots, \theta_n^*)$ the smallest subset of Θ such that balls of radius ϵ' centered around the elements in $\Gamma_{\epsilon'}$ cover Θ. $\Gamma_{\epsilon'}$ is called an ϵ'-net of Θ and $n = |\Gamma_{\epsilon'}|$ its covering number. The covering property entails that:

$$\forall\theta \in \Theta \quad \exists\theta^* \in \Gamma_{\epsilon'} \ \text{ s.t. } \ d_\Theta(\theta,\theta^*) \leqslant \epsilon'. \tag{47}$$

Step 3. (Union bound) Write $\delta' = \delta/2n$. From step 1, we know that, $\forall\theta^* \in \Gamma_{\epsilon'}$:

$$\mathbb{P}\left(\left|\frac{\hat{L}}{L} - 1\right| \geqslant \epsilon\right) \leqslant \delta' \tag{48}$$

provided that:

$$\mu \geqslant \frac{16}{\epsilon^2}\left(\epsilon C + 2C^2\right)\log\frac{5}{\delta'}. \tag{49}$$

From the union bound, we have:

$$\mathbb{P}\left(\forall\theta^* \in \Gamma_{\epsilon'}, \quad \left|\frac{\hat{L}}{L} - 1\right| \leqslant \epsilon\right) \geqslant 1 - \sum_{\theta^*\in\Gamma}\delta' = 1 - \frac{\delta}{2}, \tag{50}$$

provided that:

$$\mu \geqslant \frac{16}{\epsilon^2}\max_{\theta^*\in\Gamma_{\epsilon'}}\left(\epsilon C + 2C^2\right)\log\frac{10n}{\delta}. \tag{51}$$

Given that $\bar{\pi}_i$ will *in fine* be independent of θ (as we want the coreset property to be true for all $\theta \in \Theta$),

$$\max_{\theta^*\in\Gamma_{\epsilon'}} C = \max_{\theta^*\in\Gamma_{\epsilon'}}\max_i \frac{f(x_i,\theta)}{L\bar{\pi}_i} \tag{52}$$

$$= \max_i \frac{1}{\bar{\pi}_i}\max_{\theta^*\in\Gamma_{\epsilon'}}\frac{f(x_i,\theta)}{L} \tag{53}$$

$$\leqslant \max_i \frac{1}{\bar{\pi}_i}\max_{\theta\in\Theta}\frac{f(x_i,\theta)}{L} = \max_i\frac{\sigma_i}{\bar{\pi}_i}, \tag{54}$$

where we see how the sensitivity σ_i naturally arises in the proof. Eq. (54) entails that Eq. (51) is verified if $\mu \geqslant \mu_1$ with

$$\mu_1 = \frac{16}{\epsilon^2}\left(\epsilon\max_i\frac{\sigma_i}{\bar{\pi}_i} + 2\left(\max_i\frac{\sigma_i}{\bar{\pi}_i}\right)^2\right)\log\frac{10n}{\delta}. \tag{55}$$

Write $\delta'' = \delta/2$. From Eq. (45), we have:

$$\mathbb{P}\left(\left|\frac{\sum_i \frac{\epsilon_i}{\pi_i}}{N} - 1\right| \geqslant \epsilon\right) \leqslant \delta'', \tag{56}$$

provided that $\mu \geqslant \mu_2$ with

$$\mu_2 = \frac{16}{\epsilon^2 N\bar{\pi}_{\min}}\left(\epsilon + \frac{2}{N\bar{\pi}_{\min}}\right)\log\frac{10}{\delta}. \tag{57}$$

We have (with the union bound again):

$$\mathbb{P}\left(\left|\frac{\sum_i \frac{\epsilon_i}{\pi_i}}{N} - 1\right| \leqslant \epsilon \quad \text{AND} \quad \forall \theta^* \in \Gamma_{\epsilon'}, \quad \left|\frac{\hat{L}}{L} - 1\right| \leqslant \epsilon\right)$$

$$\geqslant 1 - \delta/2 - \delta'' = 1 - \delta, \tag{58}$$

provided that:

$$\mu \geqslant \max(\mu_1, \mu_2). \tag{59}$$

Step 4 (Continuity argument) Suppose that $\mu \geqslant \max(\mu_1^*, \mu_2^*)$ with μ_1^*, μ_2^* as defined in the theorem. The result of step 3 with $\epsilon \leftarrow \epsilon/2$ states that, with probability at least $1 - \delta$, one has:

$$\left|\frac{\sum_i \frac{\epsilon_i}{\pi_i}}{N} - 1\right| \leqslant \frac{\epsilon}{2} \quad \text{AND} \quad \forall \theta^* \in \Gamma_{\epsilon'}, \left|\frac{\hat{L}}{L} - 1\right| \leqslant \frac{\epsilon}{2}. \tag{60}$$

We now look for the maximum value of ϵ' such that Eq. (60) implies the following desired result:

$$\forall \theta \in \Theta, \qquad \left|\frac{\hat{L}}{L} - 1\right| \leqslant \epsilon. \tag{61}$$

Consider $\theta \in \Theta$. By the covering property of $\Gamma_{\epsilon'}$, we have:

$$\exists \theta^* \in \Gamma_{\epsilon'} \text{ s.t. } d_\Theta(\theta, \theta^*) \leqslant \epsilon'. \tag{62}$$

Moreover, as f is γ-Lipschitz, $\forall x_i \in \mathcal{X}$:

$$|f(x_i, \theta) - f(x_i, \theta^*)| \leqslant \gamma \; d_\Theta(\theta, \theta^*) \leqslant \gamma\epsilon'. \tag{63}$$

Thus, using Eqs. (60) and (63):

$$\hat{L}(\mathcal{X}, \theta) \leqslant \hat{L}(\mathcal{X}, \theta^*) + \gamma\epsilon' \sum_i \frac{\epsilon_i}{\pi_i} \tag{64}$$

$$\leqslant (1 + \frac{\epsilon}{2})(L(\mathcal{X}, \theta^*) + N\gamma\epsilon'). \tag{65}$$

Also, using Eq. (63) again:

$$L(\mathcal{X}, \theta^*) \leqslant L(\mathcal{X}, \theta) + N\gamma\epsilon'. \tag{66}$$

Thus:

$$\hat{L}(\mathcal{X}, \theta) \leqslant (1 + \frac{\epsilon}{2})L(\mathcal{X}, \theta) + 2N\gamma\epsilon'(1 + \frac{\epsilon}{2}). \tag{67}$$

Similarly, for the lower bound, one obtains:

$$(1 - \frac{\epsilon}{2})(L(\mathcal{X}, \theta) - 2N\gamma\epsilon') \leqslant \hat{L}(\mathcal{X}, \theta) \tag{68}$$

In order for Eqs (67) and (68) to imply Eq.(61), we need:

$$2N\gamma\epsilon'(1 + \frac{\epsilon}{2}) \leqslant \frac{\epsilon}{2} L(\mathcal{X}, \theta), \tag{69}$$

i.e.:

$$\epsilon' \leqslant \frac{\epsilon L(\mathcal{X}, \theta)}{4N\gamma(1 + \frac{\epsilon}{2})} \leqslant \frac{\epsilon L(\mathcal{X}, \theta)}{6N\gamma}. \tag{70}$$

In order for this condition to be true for all θ, we choose:

$$\epsilon' = \frac{\epsilon \min_{\theta \in \Theta} L(\mathcal{X}, \theta)}{6N\gamma} = \frac{\epsilon L^{\text{opt}}}{6N\gamma} = \frac{\epsilon \langle f \rangle_{\text{opt}}}{6\gamma}. \tag{71}$$

Concluding the proof. Consider $\mathcal{S}$ a sample from a DPP with kernel K, marginal probabilities of inclusion $\mathsf{K}_{ii} = \pi_i$ and normalized marginal probabilities $\bar{\pi}_i = \pi_i/\mu$. Consider $\epsilon \in (0, 1)$ and $\delta \in (0, 1)$. Define ϵ' as in Eq. (71) and Γ the set of centers of the n balls of radius ϵ' covering the parameter space. We showed that if $\mu \geqslant \max(\mu_1^*, \mu_2^*)$, then $\mathcal{S}$ is an ϵ-coreset with probability at least $1 - \delta$. □

Appendix B - Proof of two Lemmas

Lemma B.1. *In the 1-means problem (the k-means problem with $k = 1$), and supposing without loss of generality that the data is centered (i.e.: $\sum_j x_j = 0$), we have:*

$$\sigma_i = \frac{1}{N}\left(1 + \frac{\|x_i\|^2}{v}\right), \tag{72}$$

where $v = \frac{1}{N}\sum_{x\in\mathcal{X}} \|x\|^2$.

Proof. By definition:

$$\frac{1}{\sigma_i} = \min_c \frac{\sum_x \|x-c\|^2}{\|x_i - c\|^2}.$$

Consider $\mathcal{S}(x_i, R)$ the sphere centered on x_i and radius $R \geqslant 0$. We have that:

$$\min_c \;=\; \min_{R\geqslant 0}\; \min_{c\in\mathcal{S}}$$

We thus have:

$$\frac{1}{\sigma_i} = \min_{R\geqslant 0} \frac{1}{R^2} \min_{c\in\mathcal{S}} \sum_x \|x-c\|^2.$$

Writing $x - c = x - x_i - (c - x_i)$, we may write

$$\sum_x \|x-c\|^2 = NR^2 + \sum_x \|x-x_i\|^2 - 2R\left\|\sum_x x - x_i\right\| \cos\theta,$$

with θ the angle formed by $\sum_x x - x_i$ and $c - x_i$. As the minimum is sought for c on the sphere, the angle θ may take any value, such that the minimum is always attained with θ s.t. $\cos\theta = 1$. We finally obtain:

$$\frac{1}{\sigma_i} = N + \min_{R\geqslant 0} \frac{1}{R^2}\left(\sum_x \|x-x_i\|^2 - 2R\left\|\sum_x x - x_i\right\|\right).$$

Studying analytically the function $f(R) = \frac{a-2bR}{R^2}$, its minimum is attained for $R^* = \frac{a}{b}$ and $f(R^*) = -\frac{b^2}{a}$, such that:

$$\frac{1}{\sigma_i} = N - \frac{||\sum_x x - x_i||^2}{\sum_x \|x-x_i\|^2}.$$

Supposing without loss of generality that the data is centered, *i.e.*: $\sum_x x = 0$ and denoting $v = \frac{1}{N}\sum_x \|x\|^2$, we have:

$$\frac{1}{\sigma_i} = N - \frac{N^2\|x_i\|^2}{Nv + N\|x_i\|^2}.$$

Inverting this equation yields:

$$\begin{aligned}\sigma_i &= \frac{v + \|x_i\|^2}{Nv + N\|x_i\|^2 - N\|x_i\|^2} \\ &= \frac{1}{N}\left(1 + \frac{\|x_i\|^2}{v}\right)\end{aligned}$$

□

Lemma B.2. *In the k-means problem, $N\sigma_{min} \geqslant 1$.*

Proof. Consider $\theta^{\text{opt}} = (c_1^{\text{opt}}, \dots, c_k^{\text{opt}})$ the optimal solution of k-means and $\{\mathcal{V}_1, \mathcal{V}_2, \dots, \mathcal{V}_k\}$ their associated Voronoi sets. Consider $x_i \in \mathcal{X}$ and suppose, without loss of generality that $x_i \in \mathcal{V}_1$. Also, for any $x \in \mathcal{X}$, we denote by $c(x) = \text{argmin}_{c \in \theta} \|x - c\|^2$. We have:

$$\begin{aligned}\frac{1}{\sigma_i} &= \min_{c_1,\dots,c_k} \frac{\sum_{x\in\mathcal{X}} \|x - c(x)\|^2}{\|x_i - c(x_i)\|^2} \\ &= \min_{c_1,\dots,c_k} \frac{\sum_{x\in\mathcal{V}_1} \|x - c(x)\|^2}{\|x_i - c(x_i)\|^2} + \sum_{j=2}^{k} \frac{\sum_{x\in\mathcal{V}_j} \|x - c(x)\|^2}{\|x_i - c(x_i)\|^2}\end{aligned}$$

Given that, by definition of $c(x)$, $\forall j,\ \|x - c(x)\|^2 \leqslant \|x - c_j\|^2$, we have:

$$\frac{1}{\sigma_i} \leqslant \min_{c_1,\dots,c_k} \frac{\sum_{x\in\mathcal{V}_1} \|x - c_1\|^2}{\|x_i - c(x_i)\|^2} + \sum_{j=2}^{k} \frac{\sum_{x\in\mathcal{V}_j} \|x - c_j\|^2}{\|x_i - c(x_i)\|^2}$$

To further bound this quantity, let us constrain the domain over which the minimum is sought. Consider $\mathcal{B}(x_i, R)$ the ball centered on x_i and radius $R \geqslant 0$. Consider $\mathcal{S}(x_i, R)$ its surface (*i.e.*, the associated sphere). We have that:

$$\min_{c_1,\dots,c_k} \ \leqslant \ \min_{R\geqslant 0} \ \min_{c_1\in\mathcal{S}, (c_2,\dots,c_k)\notin\mathcal{B}}$$

Given this restricted search space, we have: $c(x_i) = c_1$ and $\|x_i - c_1\|^2 = R^2$, and thus:

$$\begin{aligned}\frac{1}{\sigma_i} \ \leqslant \ \min_{R\geqslant 0} \frac{1}{R^2} \min_{c_1\in\mathcal{S}} \Bigg(&\sum_{x\in\mathcal{V}_1} \|x - c_1\|^2 \\ &+ \min_{(c_2,\dots,c_k)\notin\mathcal{B}} \sum_{j=2}^{k} \sum_{x\in\mathcal{V}_j} \|x - c_j\|^2 \Bigg)\end{aligned}$$

Now, one may show, for all $j = 2, \dots, k$, that:

$$\sum_{x\in\mathcal{V}_j} \|x - c_j\|^2 = \sum_{x\in\mathcal{V}_j} \left\|x - c_j^{\text{opt}}\right\|^2 + \#\mathcal{V}_j \left\|c_j - c_j^{\text{opt}}\right\|^2,$$

due to the fact that $c_j^{\text{opt}} = \frac{1}{\#\mathcal{V}_j} \sum_{x\in\mathcal{V}_j} x$. Given that the minimum of $\left\|c_j - c_j^{\text{opt}}\right\|^2$ is necessarily smaller than R^2:

$$\min_{c_j\notin\mathcal{B}} \sum_{x\in\mathcal{V}_j} \|x - c_j\|^2 \leqslant \sum_{x\in\mathcal{V}_j} \left\|x - c_j^{\text{opt}}\right\|^2 + \#\mathcal{V}_j R^2,$$

such that:

$$\begin{aligned}\frac{1}{\sigma_i} &\leqslant \min_{R\geqslant 0} \frac{1}{R^2} \min_{c_1\in\mathcal{S}} \left(\sum_{x\in\mathcal{V}_1} \|x - c_1\|^2 + \alpha + (N - \#\mathcal{V}_1)R^2 \right) \\ &= N - \#\mathcal{V}_1 + \min_{R\geqslant 0} \frac{1}{R^2} \min_{c_1\in\mathcal{S}} \left(\sum_{x\in\mathcal{V}_1} \|x - c_1\|^2 + \alpha \right)\end{aligned}$$

with $\alpha = L^{\text{opt}\setminus\mathcal{V}}$ the optimal $(k-1)$-means cost on $\mathcal{X}\setminus\mathcal{V}$. Writing $x - c_1 = x - x_i - (c_1 - x_i)$, we may decompose $\sum_{x\in\mathcal{V}_1} \|x - c_1\|^2$ in $R^2\#\mathcal{V}_1 + \sum_{x\in\mathcal{V}_1} \|x - x_i\|^2 - 2R \left\|\sum_{x\in\mathcal{V}_1} x - x_i\right\| \cos\theta$, with θ the angle formed by $\sum_{x\in\mathcal{V}_1} x - x_i$ and $c_1 - x_i$. As the minimum is sought for c_1 on the sphere, the angle θ may take any value, such that the minimum is always attained with θ s.t. $\cos\theta = 1$. We finally obtain, denoting $\forall x \in \mathcal{V}_1,\ y = x - x_i$:

$$\frac{1}{\sigma_i} \leqslant N + \min_{R\geqslant 0} \frac{1}{R^2} \left(\sum_{x\in\mathcal{V}_1} \|y\|^2 - 2R \left\| \sum_{x\in\mathcal{V}_1} y \right\| + \alpha \right).$$

Studying analytically the function $f(R) = \frac{a-2bR+\alpha}{R^2}$, its minimum is attained for $R^* = \frac{a+\alpha}{b}$ and $f(R^*) = -\frac{b^2}{a+\alpha}$, such that:

$$\frac{1}{\sigma_i} \leqslant N - \frac{||\sum_{x\in\mathcal{V}_1} y||^2}{\sum_{x\in\mathcal{V}_1} \|y\|^2 + \alpha} \leqslant N.$$

This is true for all i, and in particular for $\sigma_{\min}$. □

Appendix C - The issue of outliers

Corollary 3.3 is applicable to cases where $\sigma_{\max}$ is not too large. In fact, in order for $\alpha\sigma_i$ to be smaller than π_i, and thus smaller than 1 as π_i is a probability, α should always be set inferior to $\frac{1}{\sigma_{\max}}$. Now, if $\sigma_{\max}$ is so large that $\frac{1}{\sigma_{\max}} \leqslant \frac{32}{\epsilon^2}(\epsilon + 4\mathfrak{S}) \log \frac{10n}{\delta}$, then, even by setting β to its minimum value 1, there is no admissible α verifying both conditions (26) and (27). Large values of σ_i means strong outliers[5]. A simple workaround in this case is to separate the data in two: $\mathcal{X}_o = \{x_i \text{ s.t. } \sigma_i > \sigma^*\}$ the set of outliers and $\bar{\mathcal{X}} = \{x_i \text{ s.t. } \sigma_i \leqslant \sigma^*\}$ the others, where σ^* is the threshold sensitivity over which a data point is considered as an outlier (it is set in the following). The initial cost L may also be separated in two: $L = L_o + \bar{L}$ where

$$L_o = \sum_{x\in\mathcal{X}_o} f(x,\theta) \qquad \text{and} \qquad \bar{L} = \sum_{x\in\bar{\mathcal{X}}} f(x,\theta). \tag{73}$$

Let us write $\bar{\sigma}_i$ the sensitivity of data point i in $\bar{\mathcal{X}}$ and $\bar{\mathfrak{S}} = \sum_{x\in\bar{\mathcal{X}}} \bar{\sigma}_i$. Let us choose σ^* to be the largest value in $[1/N, 1]$ for which $\frac{1}{\bar{\sigma}_{\max}} \geqslant \frac{32}{\epsilon^2}(\epsilon + 4\bar{\mathfrak{S}}) \log \frac{10n}{\delta}$ is verified. One can thus apply the corollary to $\bar{\mathcal{X}}$ to obtain $\bar{\mathcal{S}}$ such that:

$$\forall\theta \in \Theta \qquad (1-\epsilon)\bar{L}(\bar{\mathcal{X}},\theta) \leqslant \hat{\bar{L}}(\bar{\mathcal{S}},\theta) \leqslant (1+\epsilon)\bar{L}(\bar{\mathcal{X}},\theta).$$

Trivially, one may add to $\bar{\mathcal{S}}$ all outliers in $\mathcal{X}_o$ and associate to each of them a weight 1 in the estimated cost. The resulting set $\mathcal{S}$ is thus necessarily a coreset for all datapoints:

$$\forall\theta \in \Theta \qquad (1-\epsilon)L \leqslant (1-\epsilon)\bar{L} + L_o \leqslant \hat{L} = \hat{\bar{L}} + L_o \leqslant (1+\epsilon)\bar{L} + L_o \leqslant (1+\epsilon)L.$$

The number of required samples is thus the number required for $\bar{\mathcal{S}}$ to be a coreset for $\bar{\mathcal{X}}$ plus the number of outliers in $\mathcal{X}_o$: $\mathcal{O}(|\mathcal{X}_o| + \frac{\bar{\mathfrak{S}}}{\epsilon^2}(\epsilon + \bar{\mathfrak{S}}) \log \frac{n}{\delta})$. The exact value of σ^* is application and data dependent. In general, we expect it to be $\mathcal{O}(1)$, such that the number of outliers $|\mathcal{X}_o|$ may be considered as a constant and the number of required samples is of the order $\mathcal{O}(\frac{\mathfrak{S}}{\epsilon^2}(\epsilon + \mathfrak{S}) \log \frac{n}{\delta})$.

Appendix D - Implementation

D.1. Approximating the kernel via Random Fourier Features

In order to approximate L in time linear in N, we rely on random Fourier features (RFF) [33]. We briefly recall the RFF framework in the following.

Let us write κ the Gaussian kernel that we use: $\kappa(\boldsymbol{t}) = \exp(-\boldsymbol{t}^2/s^2)$. Its Fourier transform is:

$$\hat{\kappa}(\boldsymbol{\omega}) = \int_{\mathbb{R}^d} \kappa(\boldsymbol{t}) \exp^{-i\boldsymbol{\omega}^\intercal \boldsymbol{t}} \mathrm{d}\boldsymbol{t}. \tag{74}$$

It has real values as κ is symmetrical. One may write:

$$\kappa(\boldsymbol{x},\boldsymbol{y}) = \kappa(\boldsymbol{x}-\boldsymbol{y}) = \frac{1}{Z}\int_{\mathbb{R}^d} \hat{\kappa}(\boldsymbol{\omega}) \exp^{i\boldsymbol{\omega}^\intercal(\boldsymbol{x}-\boldsymbol{y})} \mathrm{d}\boldsymbol{\omega}, \tag{75}$$

where, in order to ensure that $\kappa(\boldsymbol{x},\boldsymbol{x}) = 1$:

$$Z = \int_{\mathbb{R}^d} \hat{\kappa}(\boldsymbol{\omega}) \mathrm{d}\boldsymbol{\omega}. \tag{76}$$

[5]sensitivities have indeed been shown to be good outlierness indicators [26]

According to Bochner's theorem, and due to the fact that κ is positive-definite, $\hat{\kappa}/Z$ is a valid probability density function. $\kappa(\boldsymbol{x}, \boldsymbol{y})$ may thus be interpreted as the expected value of $\exp^{i\boldsymbol{\omega}^\intercal(\boldsymbol{x}-\boldsymbol{y})}$ provided that $\boldsymbol{\omega}$ is drawn from $\hat{\kappa}/Z$:

$$\kappa(\boldsymbol{x}, \boldsymbol{y}) = \mathbb{E}_{\boldsymbol{\omega}}\left(\exp^{i\boldsymbol{\omega}^\intercal(\boldsymbol{x}-\boldsymbol{y})}\right) \tag{77}$$

The distribution $\hat{\kappa}/Z$ from which ω should be drawn from may be shown to be $\mathcal{N}(\omega; 0, 2/s^2)$, where $\mathcal{N}(x; \mu, v)$ is the normal law:

$$\mathcal{N}(x; \mu, v) = \frac{1}{\sqrt{2v\pi}} \exp^{-\frac{(x-\mu)^2}{2v}}. \tag{78}$$

In practice, we draw r random Fourier vectors from $\hat{\kappa}/Z$:

$$\Omega_r = (\boldsymbol{\omega}_1, \ldots, \boldsymbol{\omega}_r).$$

For each data point $\boldsymbol{x}_j$, we define a column feature vector associated to Ω_r:

$$\boldsymbol{\psi}_j = \frac{1}{\sqrt{r}}[\cos(\boldsymbol{\omega}_1^\intercal \boldsymbol{x}_j)| \cdots |\cos(\boldsymbol{\omega}_r^\intercal \boldsymbol{x}_j)|\sin(\boldsymbol{\omega}_1^\intercal \boldsymbol{x}_j)| \cdots |\sin(\boldsymbol{\omega}_r^\intercal \boldsymbol{x}_j)]^\intercal \in \mathbb{R}^{2r}, \tag{79}$$

and call $\Psi = (\boldsymbol{\psi}_1| \cdots |\boldsymbol{\psi}_N) \in \mathbb{R}^{2r \times N}$ the RFF matrix. Other embeddings are possible in the RFF framework, but this one was shown to be the most appropriate to the Gaussian kernel [45]. As r increases, $\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j)$ concentrates around its expected value: $\boldsymbol{\psi}_i^\intercal \boldsymbol{\psi}_j \simeq \kappa(\boldsymbol{x}_i, \boldsymbol{x}_j)$. The Gaussian kernel matrix is thus approximated via:

$$\mathsf{L} \simeq \Psi^\intercal \Psi. \tag{80}$$

Computing the RFF matrix requires $\mathcal{O}(Nrd)$ operations.

Remark D.1. *How many random features r should we choose? Firstly, note that the entry-wise concentration of $\Psi^\intercal\Psi$ around its expected value L is controlled by a multiplicative error ϵ provided that $r \geqslant \mathcal{O}(d/\epsilon^2)$ [33]. Thus, r should at least be of the order of the dimension d. Note also that,* in fine*, our goal is to obtain in average μ samples from a DPP with L-ensemble $\Psi^\intercal\Psi$. The maximum number of samples of such a DPP is the rank of Ψ, such that r should necessarily be chosen larger than μ. Finally, in our setting of k-means, we know that in the best scenario $\mu = \mathcal{O}(dk\mathfrak{S}^2) \geqslant d$. In the following, we thus set r to be a few times μ.*

D.2. Fast sampling of DPPs

In order to sample a DPP from a L-ensemble given its eigenvectors $\{\boldsymbol{u}_k\}$ and eigenvalues λ_k, one may follow Alg. 1 of [8], originally from [46]. This algorithm runs in $\mathcal{O}(N\mu^3)$ in average. The limiting step of the overall sampling algorithm is the $\mathcal{O}(N^3)$ cost of the diagonalisation of L. Thankfully, the RFFs not only provide us with an approximation of L in linear time, it also provides us with a dual representation, *i.e.*, a representation of L in the form

$$\mathsf{L} = \Psi^\intercal \Psi. \tag{81}$$

Thus, we may circumvent the prohibitive diagonalization cost of L and only diagonalize its dual form:

$$\mathsf{C} = \Psi\Psi^\intercal \in \mathbb{R}^{2r \times 2r}, \tag{82}$$

costing only $\mathcal{O}(Nr^2) = \mathcal{O}(N\mu^2)$ (time to compute C from Ψ and to compute the low-dimensional diagonalization). C's eigendecomposition yields:

$$\mathsf{C} = \mathsf{V}\mathsf{D}\mathsf{V}^\intercal, \tag{83}$$

with $\mathsf{V} = (\boldsymbol{v}_1| \ldots |\boldsymbol{v}_{2r})$ the orthonormal basis of eigenvectors and D the diagonal matrix of eigenvalues such that $0 \leqslant \nu_1 \leqslant \ldots \leqslant \nu_{2r}$.

Note that all eigenvectors associated to non-zero eigenvalues of L can be recovered from C's eigendecomposition (see, e.g., Proposition 3.1 in [8]). More precisely, if $\boldsymbol{v}_k$ is an eigenvector of C associated to eigenvalue ν_k, then:

$$\boldsymbol{u}_k = \frac{1}{\sqrt{\nu_k}} \Psi^\intercal \boldsymbol{v}_k \tag{84}$$

is a normalized eigenvector of L associated to the same eigenvalue.

In the case of such a dual representation, two standard approaches are used in the literature: 1) either follow Alg. 1 of [8] with the reconstructed eigenvectors $\mathsf{U} = \Psi^\intercal \mathsf{V} \mathsf{D}^{-1/2}$ as inputs, running in $\mathcal{O}(N\mu^3)$; 2) or follow Alg. 3 of [8] with the dual eigendecomposition $\{\boldsymbol{v}_k\}$ and $\{\nu_k\}$ as inputs, running in $\mathcal{O}(Nr\mu^2 + r^2\mu^3)$. Both approaches are nevertheless suboptimal and we show in [47] that the first (resp. second) one has an equivalent formulation running in $\mathcal{O}(N\mu^2)$ (resp. $\mathcal{O}(N\mu r)$). In this paper, we work with the following sampling strategy, given the dual eigendecomposition $\{\boldsymbol{v}_k\}$ and $\{\nu_k\}$:

i/ Sample eigenvectors. Draw N Bernoulli variables with parameters $\nu_k/(1+\nu_k)$: for $k = 1, \dots, 2r$, add k to the set of sampled indices $\mathcal{J}$ with probability $\nu_k/(1+\nu_k)$. We generically denote by J the number of elements in $\mathcal{J}$. Note that the expected value of J is μ.

ii/ Run Alg. 2 to sample a J-DPP with projective L-ensemble $\mathsf{P} = \mathsf{W}\mathsf{W}^\intercal$ where $\mathsf{W} \in \mathbb{R}^{N \times J}$ concatenates all the reconstructed eigenvectors $\boldsymbol{u}_k = \frac{1}{\sqrt{\nu_k}} \Psi^\intercal \boldsymbol{v}_k$ such that $k \in \mathcal{J}$.

Algorithm 2 Efficient J-DPP sampling algorithm with projective L-ensemble $\mathsf{P} = \mathsf{W}\mathsf{W}^\intercal$

Input: $\mathsf{W} \in \mathbb{R}^{N \times J}$ such that $\mathsf{W}^\intercal \mathsf{W} = \mathsf{I}_J$

Write $\forall i,\ \boldsymbol{y}_i = \mathsf{W}^\intercal \boldsymbol{\delta_i} \in \mathbb{R}^J$.

$\mathcal{S} \leftarrow \emptyset$

Define $\boldsymbol{p} \in \mathbb{R}^N : \forall i, \quad p(i) = \|\boldsymbol{y}_i\|^2$

for $n = 1, \dots, J$ **do**:

- Draw s_n with proba $\mathbb{P}(s) = p(s) / \sum_i p(i)$
- $\mathcal{S} \leftarrow \mathcal{S} \cup \{s_n\}$
- Compute $\boldsymbol{f}_n = \boldsymbol{y}_{s_n} - \sum_{l=1}^{n-1} \boldsymbol{f}_l (\boldsymbol{f}_l^\intercal \boldsymbol{y}_{s_n}) \in \mathbb{R}^J$
- Normalize $\boldsymbol{f}_n \leftarrow \boldsymbol{f}_n / \sqrt{\boldsymbol{f}_n^\intercal \boldsymbol{y}_{s_n}}$
- Update $\boldsymbol{p} : \forall i \quad p(i) \leftarrow p(i) - (\boldsymbol{f}_n^\intercal \boldsymbol{y}_i)^2$

end for

Output: $\mathcal{S}$ of size J.

The runtime of this strategy given the dual eigendecomposition is $\mathcal{O}(N\mu^2)$. Also, for a proof that this strategy does sample from a DPP with L-ensemble $\mathsf{L} = \Psi^\intercal \Psi$ we refer the reader to our technical report [47].

D.3. Fast sampling of m-DPPs

In the experiments, we will only provide results for m-DPP sampling. In fact, results are easier to compare with classical i.i.d. coreset methods when the number of samples is fixed and not random. Given the eigendecomposition of the dual representation C, one samples a m-DPP via the following two steps (we refer once again to [47] for a proof):

i/ Sample m eigenvectors. Draw $2r$ Bernoulli variables with parameters $\nu_k/(1+\nu_k)$ under the constraint that exactly m variables should be equal to one. Call $\mathcal{J}$ the set of indices thus drawn: $|\mathcal{J}| = J = m$.

ii/ Run Alg. 2 to sample a J-DPP with projective L-ensemble $\mathsf{P} = \mathsf{W}\mathsf{W}^\intercal$ where $\mathsf{W} \in \mathbb{R}^{N \times J}$ concatenates all the reconstructed eigenvectors $\boldsymbol{u}_k = \frac{1}{\sqrt{\nu_k}} \Psi^\intercal \boldsymbol{v}_k$ such that $k \in \mathcal{J}$.

The only difference with a usual DPP is in the first step, where the N Bernoulli variables are not drawn independently anymore, but under constraint that exactly m of them should be equal to one. To do so, one may follow Alg. 8 of [8] which runs in $\mathcal{O}(Nm)$. Step ii/ runs in $\mathcal{O}(Nm^2)$, such that the overall cost of sampling a m-DPP given the dual eigendecomposition is also $\mathcal{O}(Nm^2)$. Alg. 8 of [8]

makes use of elementary polynomials. Given the eigenvalues of C, $\{\nu_i\}$, the n-th order associated elementary polynomial reads:

$$e_n(\nu_1, \ldots, \nu_{2r}) = \sum_{\mathcal{J} \subseteq \{1,2,\ldots,2r\} \text{ s.t. } |\mathcal{J}|=n} \prod_{j \in \mathcal{J}} \nu_j \qquad \in \mathbb{R}. \tag{85}$$

As r increases, these polynomials become less and less stable to compute and Alg. 8 of [8] fails in many practical situations due to numerical precision errors as m becomes too large. In order to avoid these errors, we follow the saddle-point approximation method detailed in [48]. This method has the additional advantage of providing very accurate approximations of the probabilities of inclusion of the m-DPP (that are exactly written as a ratio of elementary polynomials and thus also vulnerable to numerical instability). We in fact need these marginals as the weight of each sample in the importance sampling estimator is the inverse of its probability of inclusion.

References

[1] P. Agarwal, S. Har-Peled, and K. Varadarajan, "Geometric approximation via coresets. Combinatorial and Computational Geometry (JE Goodman, J. Pach, and E. Welzl, eds.)," *Math. Sci. Research Inst. Pub., Cambridge*, vol. 6, p. 42, 2005.

[2] S. Har-Peled, *Geometric approximation algorithms*. American Mathematical Soc., 2011, no. 173.

[3] K. L. Clarkson, "Coresets, sparse greedy approximation and the Frank-Wolfe algorithm," in *Proceedings of the 19th Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 922–931.

[4] M. Langberg and L. J. Schulman, "Universal -approximators for integrals," in *Proceedings of the twenty-first annual ACM-SIAM symposium on Discrete Algorithms*. SIAM, 2010, pp. 598–607.

[5] O. Bachem, M. Lucic, and S. Lattanzi, "One-Shot Coresets: The Case of k-Clustering," *arXiv:1711.09649 [stat]*, Nov. 2017, arXiv: 1711.09649. [Online]. Available: http://arxiv.org/abs/1711.09649

[6] K. Chen, "On Coresets for k -Median and k -Means Clustering in Metric and Euclidean Spaces and Their Applications," *SIAM Journal on Computing*, vol. 39, no. 3, pp. 923–947, Jan. 2009. [Online]. Available: http://epubs.siam.org/doi/10.1137/070699007

[7] V. Braverman, D. Feldman, and H. Lang, "New Frameworks for Offline and Streaming Coreset Constructions," *arXiv preprint arXiv:1612.00889*, 2016. [Online]. Available: https://arxiv.org/abs/1612.00889

[8] A. Kulesza and B. Taskar, "Determinantal Point Processes for Machine Learning," *Foundations and Trends in Machine Learning*, vol. 5, no. 23, pp. 123–286, 2012. [Online]. Available: http://dx.doi.org/10.1561/2200000044

[9] A. Munteanu and C. Schwiegelshohn, "Coresets-Methods and History: A Theoreticians Design Pattern for Approximation and Streaming Algorithms," *KI - Knstliche Intelligenz*, Dec. 2017. [Online]. Available: http://link.springer.com/10.1007/s13218-017-0519-3

[10] S. Har-Peled and S. Mazumdar, "On coresets for k-means and k-median clustering," in *Proceedings of the thirty-sixth annual ACM symposium on Theory of computing*. ACM, 2004, pp. 291–300.

[11] S. Har-Peled and A. Kushal, "Smaller coresets for k-median and k-means clustering," in *Proceedings of the twenty-first annual symposium on Computational geometry*. ACM, 2005, pp. 126–134. [Online]. Available: http://dl.acm.org/citation.cfm?id=1064114

[12] M. Bdoiu and K. L. Clarkson, "Optimal core-sets for balls," *Computational Geometry*, vol. 40, no. 1, pp. 14–22, May 2008. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0925772107000454

[13] W. F. de la Vega, M. Karpinski, C. Kenyon, and Y. Rabani, "Approximation Schemes for Clustering Problems," in *Proceedings of the Thirty-fifth Annual ACM Symposium on Theory of Computing*, ser. STOC '03. New York, NY, USA: ACM, 2003, pp. 50–58. [Online]. Available: http://doi.acm.org/10.1145/780542.780550

[14] A. Kumar, Y. Sabharwal, and S. Sen, "Linear-time approximation schemes for clustering problems in any dimensions," *Journal of the ACM (JACM)*, vol. 57, no. 2, p. 5, 2010.

[15] D. Feldman and M. Langberg, "A unified framework for approximating and clustering data," in *Proceedings of the forty-third annual ACM symposium on Theory of computing*. ACM, 2011, pp. 569–578. [Online]. Available: http://dl.acm.org/citation.cfm?id=1993712

[16] O. Bachem, M. Lucic, and A. Krause, "Practical Coreset Constructions for Machine Learning," *arXiv:1703.06476 [stat]*, Mar. 2017, arXiv: 1703.06476. [Online]. Available: http://arxiv.org/abs/1703.06476

[17] P. Drineas and M. W. Mahoney, "Lectures on Randomized Numerical Linear Algebra," *arXiv:1712.08880 [cs, stat]*, Dec. 2017, arXiv: 1712.08880. [Online]. Available: http://arxiv.org/abs/1712.08880

[18] J. M. Phillips, "Coresets and Sketches," *arXiv:1601.00617 [cs]*, Jan. 2016, arXiv: 1601.00617. [Online]. Available: http://arxiv.org/abs/1601.00617

[19] D. P. Woodruff, "Sketching as a tool for numerical linear algebra," *Foundations and Trends in Theoretical Computer Science*, vol. 10, no. 12, pp. 1–157, 2014.

[20] M. W. Mahoney, "Randomized Algorithms for Matrices and Data," *Foundations and Trends in Machine Learning*, vol. 3, no. 2, pp. 123–224, 2011. [Online]. Available: http://dx.doi.org/10.1561/2200000035

[21] C. Boutsidis, A. Zouzias, M. W. Mahoney, and P. Drineas, "Randomized Dimensionality Reduction for k -Means Clustering," *IEEE Transactions on Information Theory*, vol. 61, no. 2, pp. 1045–1062, Feb. 2015. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6967844

[22] C. Boutsidis and A. Gittens, "Improved matrix algorithms via the Subsampled Randomized Hadamard Transform," *arXiv:1204.0062 [cs, math]*, Mar. 2012, arXiv: 1204.0062. [Online]. Available: http://arxiv.org/abs/1204.0062

[23] M. B. Cohen, S. Elder, C. Musco, C. Musco, and M. Persu, "Dimensionality Reduction for k-Means Clustering and Low Rank Approximation," *arXiv:1410.6801 [cs]*, Oct. 2014, arXiv: 1410.6801. [Online]. Available: http://arxiv.org/abs/1410.6801

[24] N. Keriven, N. Tremblay, Y. Traonmilin, and R. Gribonval, "Compressive K-means," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Mar. 2017, pp. 6369–6373.

[25] K. L. Clarkson and D. P. Woodruff, "Low Rank Approximation and Regression in Input Sparsity Time," *arXiv:1207.6365 [cs]*, Jul. 2012, arXiv: 1207.6365. [Online]. Available: http://arxiv.org/abs/1207.6365

[26] M. Lucic, O. Bachem, and A. Krause, "Linear-time Outlier Detection via Sensitivity," *arXiv:1605.00519 [cs, stat]*, May 2016, arXiv: 1605.00519. [Online]. Available: http://arxiv.org/abs/1605.00519

[27] M.-F. F. Balcan, S. Ehrlich, and Y. Liang, "Distributed k-means and k-median Clustering on General Topologies," in *Advances in Neural Information Processing Systems 26*, C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2013, pp. 1995–2003. [Online]. Available: http://papers.nips.cc/paper/5096-distributed-k-means-and-k-median-clustering-on-general-topologies.pdf

[28] K. Makarychev, Y. Makarychev, M. Sviridenko, and J. Ward, "A bi-criteria approximation algorithm for k Means," *arXiv:1507.04227 [cs]*, Jul. 2015, arXiv: 1507.04227. [Online]. Available: http://arxiv.org/abs/1507.04227

[29] S. Lloyd, "Least squares quantization in PCM," *IEEE Transactions on Information Theory*, vol. 28, no. 2, pp. 129–137, Mar. 1982.

[30] R. Pemantle and Y. Peres, "Concentration of Lipschitz functionals of determinantal and other strong Rayleigh measures," *Combinatorics, Probability and Computing*, vol. 23, no. 01, pp. 140–160, 2014.

[31] Y. Li, P. M. Long, and A. Srinivasan, "Improved Bounds on the Sample Complexity of Learning," *Journal of Computer and System Sciences*, vol. 62, no. 3, pp. 516 – 527, 2001. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0022000000917410

[32] S. A. Geer, *Empirical Processes in M-estimation*. Cambridge university press, 2000, vol. 6.

[33] A. Rahimi and B. Recht, "Random features for large-scale kernel machines," in *Advances in neural information processing systems*, 2008, pp. 1177–1184.

[34] D. Arthur and S. Vassilvitskii, "k-means++: The advantages of careful seeding," in *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*. Society for Industrial and Applied Mathematics, 2007, pp. 1027–1035.

[35] L. Hubert and P. Arabie, "Comparing partitions," *Journal of classification*, vol. 2, no. 1, pp. 193–218, 1985.

[36] S. Fortunato, "Community detection in graphs," *Physics Reports*, vol. 486, no. 3-5, pp. 75–174, 2010.

[37] A. Ng, M. Jordan, Y. Weiss, and others, "On spectral clustering: Analysis and an algorithm," *Advances in neural information processing systems*, vol. 2, pp. 849–856, 2002.

[38] U. von Luxburg, "A tutorial on spectral clustering," *Statistics and Computing*, vol. 17, no. 4, pp. 395–416, 2007.

[39] E. Abbe and C. Sandon, "Community Detection in General Stochastic Block models: Fundamental Limits and Efficient Algorithms for Recovery," in *Foundations of Computer Science (FOCS), 2015 IEEE 56th Annual Symposium on*. IEEE, 2015, pp. 670–688.

[40] A. Decelle, F. Krzakala, C. Moore, and L. Zdeborov, "Asymptotic analysis of the stochastic block model for modular networks and its algorithmic applications," *Phys. Rev. E*, vol. 84, no. 6, p. 066106, Dec. 2011.

[41] Y. LeCun, "The MNIST database of handwritten digits," *http://yann. lecun. com/exdb/mnist/*, 1998.

[42] A. Vedaldi and B. Fulkerson, "Vlfeat: an open and portable library of computer vision algorithms." ACM Press, 2010, p. 1469. [Online]. Available: http://dl.acm.org/citation.cfm?doid=1873951.1874249

[43] M. Muja and D. G. Lowe, "Scalable Nearest Neighbor Algorithms for High Dimensional Data," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 11, pp. 2227–2240, Nov. 2014.

[44] R. Baraniuk, M. Davenport, R. DeVore, and M. Wakin, "A Simple Proof of the Restricted Isometry Property for Random Matrices," *Constructive Approximation*, vol. 28, no. 3, pp. 253–263, Dec. 2008. [Online]. Available: http://link.springer.com/10.1007/s00365-007-9003-x

[45] D. J. Sutherland and J. Schneider, "On the Error of Random Fourier Features," *arXiv:1506.02785 [cs, stat]*, Jun. 2015, arXiv: 1506.02785. [Online]. Available: http://arxiv.org/abs/1506.02785

[46] J. B. Hough, M. Krishnapur, Y. Peres, and B. Virg, "Determinantal Processes and Independence," *Probability Surveys*, vol. 3, no. 0, pp. 206–229, 2006. [Online]. Available: http://projecteuclid.org/euclid.ps/1146832696

[47] N. Tremblay, S. Barthelme, and P.-O. Amblard, "Optimized Algorithms to Sample Determinantal Point Processes," *arXiv:1802.08471 [cs, stat]*, Feb. 2018, arXiv: 1802.08471. [Online]. Available: http://arxiv.org/abs/1802.08471

[48] S. Barthelme, P.-O. Amblard, and N. Tremblay, "Asymptotic Equivalence of Fixed-size and Varying-size Determinantal Point Processes," *arXiv:1803.01576 [math, stat]*, Mar. 2018, arXiv: 1803.01576. [Online]. Available: http://arxiv.org/abs/1803.01576