A Family of Distributions of Random Subsets for Controlling Positive and Negative Dependence

Takahiro Kawashima
ZOZO Research

Hideitsu Hino
Institute of Statistical Mathematics
RIKEN AIP

Abstract

Positive and negative dependence are fundamental concepts that characterize the attractive and repulsive behavior of random subsets. Although some probabilistic models are known to exhibit positive or negative dependence, it is challenging to seamlessly bridge them with a practicable probabilistic model. In this study, we introduce a new family of distributions, named the discrete kernel point process (DKPP), which includes determinantal point processes and parts of Boltzmann machines. We also develop some computational methods for probabilistic operations and inference with DKPPs, such as calculating marginal and conditional probabilities and learning the parameters. Our numerical experiments demonstrate the controllability of positive and negative dependence and the effectiveness of the computational methods for DKPPs.

1 INTRODUCTION

Random subset selection from a ground set is often encountered in problems related to statistics and machine learning. One common problem is modeling the purchasing behavior of customers; buying items from a ground set of products can be seen as the occurrence of a random subset. To go beyond the independent selection of items, we should consider a probabilistic model on the powerset of the ground set. Positive and negative dependence are fundamental concepts that characterize probabilistic models of random subsets. If the model has positive dependence, an attractive force is present and similar items tend to appear in a random subset. Conversely, if the model has negative dependence, a repulsive force occurs and random subsets are likely to contain diverse items.

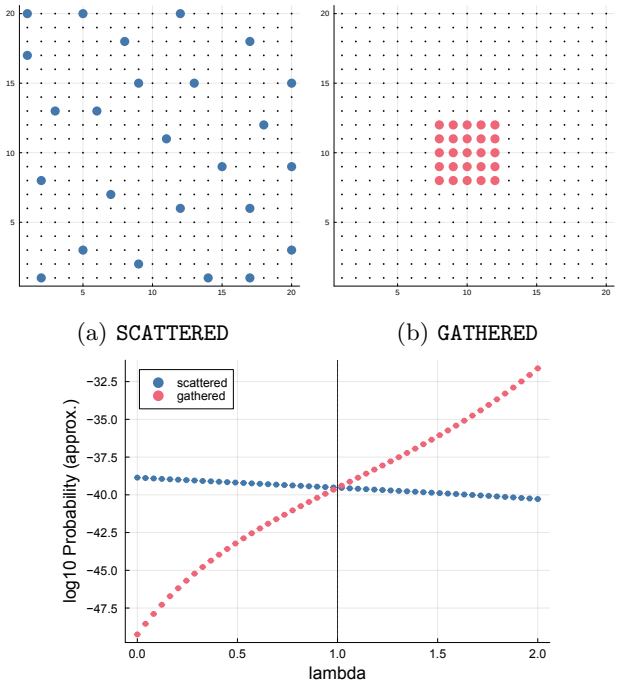


Figure 1: Two examples of subsets on the grid and their conditional probabilities $\log_{10} P(\mathcal{A} \mid |\mathcal{A}| = k)$ with slight change in function that determines DKPPs. See Section 6 for more details.

Log-supermodularity and log-submodularity are representative characterizations of positive and negative dependence (Fortuin et al., 1971; Pemantle, 2000; Borcea et al., 2009). For example, ferromagnetic Ising models and determinantal point processes (DPPs) (Macchi, 1975; Borodin and Rains, 2005) are log-supermodular and log-submodular probabilistic models, respectively. Since the review article by Kulesza and Taskar (2012), DPPs have gained increasing attention in the machine learning community owing to their diversity-

Proceedings of the 28th International Conference on Artificial Intelligence and Statistics (AISTATS) 2025, Mai Khao, Thailand. PMLR: Volume 258.

promoting property.

One of the reasons that makes DPPs popular is their parameterization by a kernel matrix, which consists of pairwise similarities of items. Since kernel matrices can be constructed from any features of items, this parameterization is suitable for various machine learning and statistical problems. Indeed, DPPs are now applied to widespread problems including image search (Kulesza and Taskar, 2011), document summarization (Gillenwater et al., 2012; Dupuy and Bach, 2018), recommender systems (Wilhelm et al., 2018), randomized numerical linear algebra (Derezinski et al., 2020b; Dereziński and Mahoney, 2021), experimental design (Derezinski et al., 2020a; Dereziński et al., 2022), and counterfactual explanation (Mothilal et al., 2020). On the other hand, diversity is not a general prescription for such problems. Recommending similar glasses to a customer who previously bought certain glasses may not be effective, but it may be different with socks. Otherwise, we may want to adjust the strength of the repulsive force.

In this paper, we propose a new family of distributions for random subsets, discrete kernel point processes (DKPPs), by generalizing DPPs. DKPPs are determined by a kernel matrix and a scalar function on $\mathbb{R}_{\geq 0}$. Similarly to DPPs, the kernel matrix provides the pairwise similarity of items. Furthermore, the presence of the scalar function enables us to control positive and negative dependence. Figure 1 shows the actual behavior of DKPPs (see Section 6 for details). Attractive and repulsive forces are flexibly determined by changing a parameter λ. We also develop computational methods for evaluating marginal and conditional probabilities and learning DKPPs for practical use. Furthermore, we conduct an experiment on repulsive and attractive subset acquisition to demonstrate the applicability of the DKPPs.

1.1 Related Works

There are a few previous works that aim to develop a family of distributions to control positive and negative dependence. The closest ones to our study are immanantal point processes (Diaconis and Evans, 2000) and α-DPPs (Vere-Jones, 1997), in which the determinants in DPPs are generalized to the immanant and α-determinant, respectively. Although both immanantal point processes and α-DPPs include permanental point processes (i.e., behave like bosons with positive dependence (Macchi, 1975)), the computational issue is not resolved; even computing the permanent is #P-hard (Valiant, 1979). For negative dependence only, Mariet et al. (2018) considered the exponentiated strong Rayleigh distributions that can control the negative dependence and developed an approximate sampler for them.

Iyer and Bilmes (2015) introduced the submodular and log-submodular point processes as the family of distributions on a powerset. Although they mainly discussed how to handle these distributions, such as probabilistic operations and parameter learning, they did not focus on the control of positive and negative dependence.

2 PRELIMINARY

2.1 Supermodular and Submodular Functions

Let $\mathcal{Y} = \{1, \dots, N\}$ be a finite ground set with N items. A set function $f : 2^{\mathcal{Y}} \to \mathbb{R}$ is said to be submodular if $f(\mathcal{S}) + f(\mathcal{T}) \geq f(\mathcal{S} \cup \mathcal{T}) + f(\mathcal{S} \cap \mathcal{T})$ holds for every $\mathcal{S}, \mathcal{T} \subseteq \mathcal{Y}$. A set function f is said to be supermodular if $-f$ is submodular and modular if f satisfies both submodularity and supermodularity.

For a continuous model with a probability density function p, the log-concavity of p is considered rather than concavity (An, 1996; Borzadaran and Borzadaran, 2011). In a similar spirit, we will focus on log-supermodular and log-submodular probability functions later; the set function f has log-supermodularity if $\log f$ is supermodular, and the same applies to log-submodularity. If f is log-submodular and log-supermodular, f is said to be a log-modular function.

The multilinear extension of the set function f is one approach to extend the discrete f to a continuous function and was first introduced for submodular maximization problems (Calinescu et al., 2007; Chekuri et al., 2014).

Definition 2.1. Let $f : 2^{\mathcal{Y}} \to \mathbb{R}$ be a set function. The multilinear extension of f, denoted by $\tilde{f} : [0,1]^N \to \mathbb{R}$, is defined by

$$\tilde{f}(\boldsymbol{q}) \coloneqq \sum_{\mathcal{A} \subseteq \mathcal{Y}} f(\mathcal{A}) \prod_{i \in \mathcal{A}} q_i \prod_{i \notin \mathcal{A}} (1 - q_i), \tag{1}$$

where $\boldsymbol{q} \in [0,1]^N$.

Suppose that $Q_{\boldsymbol{q}}$ is the probability function of N independent Bernoulli trials with the parameters $\{q_i\}$:

$$Q_{\boldsymbol{q}}(\boldsymbol{\xi}) \coloneqq \prod_{i=1}^{N} Q_{q_i}(\xi_i) \coloneqq \prod_{i=1}^{N} \mathrm{Bernoulli}(\xi_i; q_i), \tag{2}$$

for $\boldsymbol{\xi} \in \{0,1\}^N$. Then, the multilinear extension (1) can also be written as

$$\tilde{f}(\boldsymbol{q}) = \mathbb{E}_{\boldsymbol{\xi} \sim Q_{\boldsymbol{q}}}[f(\mathcal{A}_{\boldsymbol{\xi}})],$$

where $\mathcal{A}_{\boldsymbol{\xi}} \coloneqq \{i \in \mathcal{Y} : \xi_i = 1\}$.

2.2 Positive and Negative Dependence

Throughout this paper, we consider probability distributions on $2^{\mathcal{Y}}$, which assign occurrence probabilities to every subset $\mathcal{A} \subseteq \mathcal{Y}$. Such distributions are equivalent to those of N-dimensional random binary vectors and discrete point processes on $\mathcal{Y}$.

Positive and negative dependence are essential concepts that characterize distributions on $2^{\mathcal{Y}}$. Generally, the probability function $P : 2^{\mathcal{Y}} \to [0, 1]$ exhibits positive dependence when a random subset $\mathcal{A} \sim P$ tends to contain similar elements. For example, consider a ferromagnetic Ising model. In this model, closely located spins tend to align in the same direction, with similarity determined by their distance on the grid. This scenario illustrates positive dependence. Conversely, an antiferromagnetic Ising model leads to a random subset with diverse elements, as adjacent spins tend to have opposite directions. This is an example of negative dependence.

Log-supermodularity and log-submodularity of the probability function P are representative characterizations of positive and negative dependence, respectively. For an intuitive understanding, we consider two singletons $\mathcal{S} = \{i\}$ and $\mathcal{T} = \{j\}$ such that $i \neq j$. If P is log-submodular, the inequality $P(\{i, j\}) \leq ZP(\{i\})P(\{j\}) \propto P(\{i\})P(\{j\})$ holds, where Z is the normalizing constant of P. The co-occurrence of i and j is upper bounded by a constant multiple of the probability that each appears alone, which implies the negative dependence. The same applies to the positive dependence. The log-supermodularity of the probability function P is a sufficiently strong condition. Indeed, the Fortuin—Kasteleyn—Ginibre (FKG) inequality states that log-supermodularity leads to other major characterizations (Fortuin et al., 1971). On the other hand, the log-submodularity of P is a relatively weak condition for the negative dependence induced by other characterizations but not vice versa (Pemantle, 2000; Borcea et al., 2009). However, log-submodular distributions are often used to model diverse random subsets (Iyer and Bilmes, 2015; Tschiatschek et al., 2016; Djolonga et al., 2018) because of their tractability. In Subsection 3.1, we also use log-supermodularity and log-submodularity to control positive and negative dependence.

2.3 Operator Monotonicity and Convexity

Let $\phi : \mathbb{R} \to \mathbb{R}$ be a function. Given the $N \times N$ Hermitian matrix $\boldsymbol{X}$ that can be diagonalized as $\boldsymbol{X} = \boldsymbol{U}\mathrm{diag}(\lambda_1, \ldots, \lambda_N)\boldsymbol{U}^*$, we also regard ϕ as a matrix operator such that $\phi : \boldsymbol{X} \mapsto \boldsymbol{U}\mathrm{diag}(\phi(\lambda_1), \ldots, \phi(\lambda_N))\boldsymbol{U}^*$. Note that a matrix logarithm and a matrix exponential are special cases of ϕ. Now, we can define the monotonicity of ϕ as an operator.

Definition 2.2. The function ϕ is said to be operator monotone if $\boldsymbol{A} \succeq \boldsymbol{B}$ implies $\phi(\boldsymbol{A}) \succeq \phi(\boldsymbol{B})$ for all $n \in \mathbb{N}$ and for all the $n \times n$ Hermitian matrices $\boldsymbol{A}, \boldsymbol{B}$. The function ϕ is operator antitone if $-\phi$ is operator monotone.

Here, we use $\succeq$ for positive semidefinite ordering. Since ordinary monotonicity is the special case of $n = 1$ in Definition 2.2, operator monotonicity is much stronger than ordinary monotonicity. Indeed, $\phi(x) = x^p$ is operator monotone on $[0, \infty)$ for $p \in [0, 1]$, but not for $p = 2$ (see Bhatia, 1997, Chapter 5).

We can also define operator convexity and concavity.

Definition 2.3. The function ϕ is said to be operator convex if

$$t\phi(\boldsymbol{A}) + (1-t)\phi(\boldsymbol{B}) \succeq \phi(t\boldsymbol{A} + (1-t)\boldsymbol{B}),\ t \in [0, 1]$$

holds for all $n \in \mathbb{N}$ and for all the Hermitian matrices $\boldsymbol{A}, \boldsymbol{B}$. A function ϕ is operator concave if $-\phi$ is operator convex.

Let ϕ' be a function and ϕ be a primitive of ϕ'. As with ordinary monotone functions, ϕ is operator convex if ϕ' is operator monotone. We denote $\boldsymbol{X}[\mathcal{A}] \coloneqq (X_{ij})_{i,j \in \mathcal{A}}$ for $\mathcal{A} \subseteq \mathcal{Y}$. Friedland and Gaubert (2013) obtained an interesting result that bridges operator monotonicity/antitonicity and supermodularity/submodularity.

Theorem 2.4. *(Friedland and Gaubert, 2013) Suppose that ϕ is a real continuous function on the interval $\mathcal{E} \subset \mathbb{R}$ and that ϕ is a primitive of the operator monotone function ϕ' on $\mathcal{E}$. Then, for every $N \times N$ Hermitian matrix $\boldsymbol{X}$ whose eigenvalues are all in $\mathcal{E}$, the set function*

$$f : 2^{\mathcal{Y}} \to \mathbb{R} : \mathcal{A} \mapsto \mathrm{tr}\phi(\boldsymbol{X}[\mathcal{A}]) \tag{3}$$

is supermodular. If ϕ is a primitive of the operator antitone ϕ', the set function f is submodular.

Conceptually, Theorem 2.4 says that the operator convexity/concavity of ϕ corresponds to the supermodularity/submodularity of $f : \mathcal{A} \to \boldsymbol{X}[\mathcal{A}]$[1]. Note that the submodularity of the set function $\mathcal{A} \mapsto \log\det(\boldsymbol{X}[\mathcal{A}])$ is well known (Bach, 2013), and it is a special case of (3) with $\phi = \log$. Another example from Theorem 2.4 is $\phi : x \mapsto x^p$; it leads to a submodular f when $p \in [0, 1]$ and a supermodular f when $p \in [1, 2]$.

[1]This is not always true because there are operator convex functions that cannot be obtained as primitives of operator motonone functions, as stated by Friedland and Gaubert (2013).

3 DISCRETE KERNEL POINT PROCESSES

Definition 3.1. Let $\phi : \mathbb{R}_{\geq 0} \to \mathbb{R}$ be a continuous function and $\boldsymbol{L}$ be an $N \times N$ positive semidefinite Hermitian matrix. The probability function $P_\phi(\cdot; \boldsymbol{L}) : 2^{\mathcal{Y}} \to [0, 1]$ is called a discrete kernel point process (DKPP) if it is given by

$$P_\phi(\mathcal{A}; \boldsymbol{L}) = \frac{1}{Z_\phi(\boldsymbol{L})} \exp\left(\mathrm{tr}\phi(\boldsymbol{L}[\mathcal{A}])\right) =: \frac{\tilde{P}_\phi(\mathcal{A}; \boldsymbol{L})}{Z_\phi(\boldsymbol{L})},$$
$$\text{where} \quad Z_\phi(\boldsymbol{L}) \coloneqq \sum_{\mathcal{A} \subseteq \mathcal{Y}} \exp\left(\mathrm{tr}\phi(\boldsymbol{L}[\mathcal{A}])\right)$$

for every $\mathcal{A} \subseteq \mathcal{Y}$.

Let $\mathcal{C}(\mathbb{R}_{\geq 0}, \mathbb{R})$ denote the set of all functions from $\mathbb{R}_{\geq 0}$ to $\mathbb{R}$. The set

$$\mathcal{F}_{\mathrm{DKPP}} \coloneqq \{P_\phi(\cdot; \boldsymbol{L}) : \ \phi \in \mathcal{C}(\mathbb{R}_{\geq 0}, \mathbb{R}), \boldsymbol{L} \in \mathbb{H}^N_{\geq 0}\}$$

is a family of specific distributions on $2^{\mathcal{Y}}$. We readily see that DPPs belong to $\mathcal{F}_{\mathrm{DKPP}}$.

Proposition 3.2. *If $\phi = \log$, a DKPP $P_\phi(\cdot; \boldsymbol{L})$ is a DPP.*

Proof. In general, we have the identity $\mathrm{tr}\log \boldsymbol{X} = \log\det \boldsymbol{X}$ for every positive semidefinite $\boldsymbol{X}$. This leads to $P_\phi(\mathcal{A}; \boldsymbol{L}) \propto \exp(\mathrm{tr}\log \boldsymbol{L}[\mathcal{A}]) = \exp(\log\det \boldsymbol{L}[\mathcal{A}]) = \det \boldsymbol{L}[\mathcal{A}]$. □

One of the representative probabilistic models for a random binary vector $\boldsymbol{\xi} \in \{0, 1\}^N$ (equivalently random subsets) is the Boltzmann machine (Ackley et al., 1985). A fully visible Boltzmann machine with bias vectors $\boldsymbol{h} \in \mathbb{R}^N$ and symmetric connections $\boldsymbol{W}$ ($\mathrm{diag}(\boldsymbol{W}) = \boldsymbol{0}$, $W_{ij} = W_{ji}$) is modeled as

$$P_{\mathrm{BM}}(\boldsymbol{\xi}; \boldsymbol{h}, \boldsymbol{W}) \propto \exp\left(\sum_{i=1}^N h_i \xi_i + \sum_{i,j=1}^N W_{ij} \xi_i \xi_j\right). \quad (4)$$

In the Boltzmann machine (4), two-body interactions of $\boldsymbol{\xi}$ are captured by the quadratic term. We can see that a DKPP becomes a Boltzmann machine if ϕ is a quadratic function.

Proposition 3.3. *Suppose that $\phi(x) = ax^2 + bx + c$. Then, the DKPP $P_\phi(\cdot; \boldsymbol{L})$ is equivalent to the Boltzmann machine* (4) *with the parameters*

$$W_{ij} = \begin{cases} 0 & (i = j), \\ a|L_{ij}|^2 & (i \neq j), \end{cases} \quad (5)$$
$$h_i = aL_{ii}^2 + bL_{ii} + c,$$

for $i, j = 1, \ldots, N$.

The proof of Proposition 3.3 is in Appendix A. Note that although any values are allowed for W_{ij} in Boltzmann machines, the relation (5) implies that a DKPP with quadratic ϕ can only represent either all non-negative or all non-positive $\{W_{ij}\}_{i,j}$.

When $a = 0$, the connections W_{ij} vanish in (5), meaning all elements of $\boldsymbol{\xi}$ become independent. Formally, the following corollary holds as a special case of Proposition 3.2.

Corollary 3.4. *Suppose that $\phi(x) = bx + c$. Then, the DKPP $P_\phi(\cdot; \boldsymbol{L})$ is equivalent to N independent Bernoulli trials where the probability of success of the i-th trial is given by $p_i = \sigma(bL_{ii} + c)$, with $\sigma : \mathbb{R} \to (0, 1)$ being the logistic sigmoid function.*

Additionally, the following proposition also holds for affine ϕ.

Proposition 3.5. *A DKPP $P_\phi(\cdot; \boldsymbol{L})$ is log-modular if and only if ϕ is affine for all $x \in \mathbb{R}_{\geq 0}$.*

The proof of Proposition 3.5 is shown in Appendix A.

3.1 Positive and Negative Dependence of DKPPs

The behavior of a DKPP P_ϕ is fundamentally determined by ϕ, allowing control over positive and negative dependence by appropriately choosing ϕ. The following corollary follows directly from Theorem 2.4.

Corollary 3.6. *Let ϕ be a primitive of a function ϕ'. A DKPP $P_\phi(\cdot; \boldsymbol{L})$ is log-supermodular if ϕ' is operator monotone and log-submodular if ϕ' is operator antitone.*

An appropriate parameterization of ϕ enables a smooth transition between positive and negative dependence. For example, we consider the scaled Box–Cox transformation for ϕ:

$$\phi_{\beta,\lambda}(x) \coloneqq \begin{cases} \beta \log x & (\lambda = 0), \\ \dfrac{\beta(x^\lambda - 1)}{\lambda} & (\text{otherwise}), \end{cases} \quad (6)$$

with the hyperparameters $\lambda \in \mathbb{R}$ and $\beta \in \mathbb{R}_{>0}$. According to Corollary 3.6, a DKPP $P_{\phi_{\beta,\lambda}}$ exhibits negative dependence for $\lambda \in [0, 1]$ and positive dependence for $\lambda \in [1, 2]$. It reduces to a DPP for $\lambda = 0$ and a Boltzmann machine for $\lambda = 2$. Hereafter, we denote $\phi_\lambda \coloneqq \phi_{1,\lambda}$ for simplicity.

4 OPERATIONS AND INFERENCE OF DKPPs

4.1 Mode Exploration

The mode exploration $\arg\max_{\mathcal{A}\subseteq\mathcal{Y}} \log P_\phi(\mathcal{A};\boldsymbol{L})$ is one of the most fundamental problems regarding probability model over sets, but it is NP-hard for a general ϕ for DKPPs. If $P_\phi(\cdot;\boldsymbol{L})$ is log-supermodular, it becomes a submodular minimization problem. Although there are (strongly) polynomial-time combinatorial algorithms (Schrijver, 2000; Iwata et al., 2001; Orlin, 2009) for submodular minimization problems, their computational complexity is expensive (typically $\mathcal{O}(N^6)$ for function calls). The minimum-norm point algorithm (Fujishige et al., 2006; Fujishige and Isotani, 2011; Chakrabarty et al., 2014) is an alternative to such combinatorial algorithms. Although the minimum-point algorithm has weaker theoretical complexity, it usually performs better in practice.

When $P_\phi(\cdot;\boldsymbol{L})$ is log-submodular, the problem becomes a submodular maximization problem. Although submodular maximization problems are NP-hard, many approximation algorithms have been proposed. If $\log P_\phi(\cdot;\boldsymbol{L})$ is not only submodular but also monotone (i.e., $\log P_\phi(\mathcal{S};\boldsymbol{L}) \le \log P_\phi(\mathcal{T};\boldsymbol{L})$ holds for every $\mathcal{S}\subseteq\mathcal{T}$), the simple greedy algorithm provides a $1-1/e$ approximation (Nemhauser et al., 1978). However, we have no guarantee of monotonicity in general. For example, a DPP is a special case of DKPPs and has a submodular but non-monotone log-probability function. For general non-monotone submodular maximization problems, a deterministic 1/3-approximate algorithm and a randomized 1/2-approximate algorithm (in expectation) are proposed in (Buchbinder et al., 2012).

In constrained settings, submodular minimization problems are generally NP-hard (Garey and Johnson, 1979; Feige et al., 2001), and there are no algorithms with a polynomial approximation factor even for constraints with a cardinality lower bound (Svitkina and Fleischer, 2011). Nevertheless, some practical approximation techniques can be applied (Svitkina and Fleischer, 2011; Iyer et al., 2013). Conversely, there are approximate algorithms yielding constant factor approximations for submodular maximization problems under cardinality-constrained settings. Buchbinder et al. (2014) proposed efficient algorithms achieving the approximation factors in the range $[1/e+0.004, 1/2-o(1)]$ for the constraint $|\mathcal{A}|\le k$ and $[0.356, 1/2-o(1)]$ for $|\mathcal{A}|=k$, in expectation.

4.2 Sampling

Although direct sampling from a DKPP $P_\phi(\cdot;\boldsymbol{L})$ is difficult for a general ϕ, Markov chain Monte Carlo (MCMC) samplers are effective. Although many advanced MCMC samplers are not suitable for DKPPs owing to their discreteness, the recently proposed Langevin-like sampler (Zhang et al., 2022) may be a viable option. Classical MCMC samplers are also applicable. For the probability function P on the power-set 2^V, mixing times of the Metropolis–Hastings sampler (including the Gibb sampler) are studied both in the log-supermodular or log-submodular cases (Gotovos et al., 2015) and for general cases (Rebeschini and Karbasi, 2015).

4.3 Normalizing Constant and Expectation

The evaluation of the normalizing constant $Z_\phi(\boldsymbol{L})$ is also crucial, as it is required for calculating marginal probabilities. Several methods are developed for approximating or bounding $Z_\phi(\boldsymbol{L})$, including the mean-field approximation for set distributions (Djolonga et al., 2018) and the perturb-and-MAP method (Papandreou and Yuille, 2011; Hazan and Jaakkola, 2012; Balog et al., 2017). The L-field developed in (Djolonga and Krause, 2014; Djolonga et al., 2018) is also applicable to obtain the lower and upper bound of $\log Z_\phi(\boldsymbol{L})$ if the DKPP is log-submodular or log-supermodular.

The mean-field approximation is a basic method in statistical mechanics and Bayesian statistics to approximate target distributions. For the DKPP $P_\phi(\cdot;\boldsymbol{L})$, we can employ $Q_{\boldsymbol{q}}$ defined in (2) for the variational distribution and aim to minimize the Kullback–Leibler divergence (KLD) between the variational distribution and the DKPP. Then, we obtain the following iterative update rule for $i=1,\dots,N$ from the coordinate ascent:

$$q_i \leftarrow \sigma\left(\mathbb{E}_{\boldsymbol{\xi}_{\setminus i}\sim Q_{\boldsymbol{q}_{\setminus i}}}[f(i|\mathcal{A}_{\boldsymbol{\xi}_{\setminus i}})]\right), \tag{7}$$

$$\text{where}\quad Q_{\boldsymbol{q}_{\setminus i}}(\boldsymbol{\xi}_{\setminus i}) \coloneqq \prod_{j\neq i} \mathrm{Bernoulli}(\xi_j; q_j),$$

$$f(i|\mathcal{A}) \coloneqq \mathrm{tr}\phi(\boldsymbol{L}[\mathcal{A}\cup\{i\}]) - \mathrm{tr}\phi(\boldsymbol{L}[\mathcal{A}]).$$

Here, $\mathcal{A}_{\boldsymbol{\xi}_{\setminus i}}$ is defined as $\mathcal{A}_{\boldsymbol{\xi}_{\setminus i}} \coloneqq \{i\in\{1,\dots,i-1,i+1,\dots,N\} : \xi_i = 1\}$. The derivation of (7) is presented in Appendix B. The expectation in (7) can easily be approximated using Monte Carlo methods. Once the optimal $\boldsymbol{q}$ is found, the tightened evidence lower bound (ELBO) $L(\boldsymbol{q}) \coloneqq \mathbb{H}[Q_{\boldsymbol{q}}] + \mathbb{E}_{\boldsymbol{\xi}\sim Q_{\boldsymbol{q}}}[\mathrm{tr}\phi(\boldsymbol{L}[\mathcal{A}_{\boldsymbol{\xi}}])]$, which ensures $\log Z_\phi \ge L(\boldsymbol{q})$, can be computed ($\mathbb{H}[\cdot]$ means the entropy).

Our proposed method for evaluating the normalizing constant $Z_\phi(\boldsymbol{L})$ of a DKPP is the combination of mean-field approximation and importance sampling.

For any set function g and a proposal distribution Q on $2^{\mathcal{Y}}$, the expectation $\mathbb{E}_{\mathcal{A}\sim P_\phi}[g(\mathcal{A})]$ can be evaluated as the weighted mean over Q since

$$\mathbb{E}_{\mathcal{A}\sim P_\phi}[g(\mathcal{A})] = \mathbb{E}_{\mathcal{A}\sim Q}[w(\mathcal{A})g(\mathcal{A})], \tag{8}$$

where $w(\mathcal{A}) \coloneqq P_\phi(\mathcal{A};\boldsymbol{L})/Q(\mathcal{A})$. For unnormalized $\tilde{P}_\phi$,

$$1 = \mathbb{E}_{\mathcal{A}\sim P_\phi}[1] = \frac{1}{Z_\phi(\boldsymbol{L})}\mathbb{E}_{\mathcal{A}\sim Q}\left[\frac{\tilde{P}_\phi(\mathcal{A};\boldsymbol{L})}{Q(\mathcal{A})}\right]$$
$$\iff Z_\phi(\boldsymbol{L}) = \mathbb{E}_{\mathcal{A}\sim Q}\left[\frac{\tilde{P}_\phi(\mathcal{A};\boldsymbol{L})}{Q(\mathcal{A})}\right] \tag{9}$$

yields the sampling-based method for evaluating the normalizing constant. If our goal is to obtain (8), the approximated $Z_\phi(\boldsymbol{L})$ can be plugged into the weight $w(\mathcal{A}) = P_\phi(\mathcal{A};\boldsymbol{L})/Q(\mathcal{A}) = \tilde{P}_\phi(\mathcal{A};\boldsymbol{L})/(Z_\phi(\boldsymbol{L})Q(\mathcal{A}))$.

When evaluating (9), i.e., $g(\mathcal{A}) \equiv 1$, we employ the variational distribution $Q_{\boldsymbol{q}}$ obtained by iterating (7) as the proposal distribution. In general, $Q^*(\mathcal{A}) \propto |g(\mathcal{A})|P_\phi(\mathcal{A};\boldsymbol{L})$ minimizes $\mathrm{Var}_{\mathcal{A}\sim Q}[w(\mathcal{A})g(\mathcal{A})]$ (Rubinstein and Kroese, 2008). Because this choice of the proposal distribution is the solution of $\arg\min_{Q_{\boldsymbol{q}}} \mathrm{KL}(Q_{\boldsymbol{q}}\|Q^*) = \arg\min_{Q_{\boldsymbol{q}}} \mathrm{KL}(Q_{\boldsymbol{q}}\|P_\phi(\cdot;\boldsymbol{L}))$, it is the reasonable choice for evaluating (9).

4.4 Marginal and Conditional Probabilities

Consider recommending k items as a subset of size k. The customer may already have items in the basket; then conditional probabilities naturally arise. Marginal probabilities are also important because they define conditional probabilities. We introduce some computational techniques to evaluate marginal and conditional probabilities of random subset models.

Marginal Probability

Let $\mathcal{A}_{\mathrm{sub}}$ and $\mathcal{A}_{\mathrm{sup}}$ be given subsets $\mathcal{Y}$ such that $\mathcal{A}_{\mathrm{sub}} \subseteq \mathcal{A}_{\mathrm{sup}}$. We consider approximating the marginal probability $\mathbb{P}(\mathcal{A}_{\mathrm{sub}} \subseteq \mathcal{A} \subseteq \mathcal{A}_{\mathrm{sup}})$ when $\mathcal{A}$ is a random subset following a probabilistic function on $2^{\mathcal{Y}}$. A straightforward way to evaluate this marginal probability is to use the importance sampling as in (8):

$$\begin{aligned}\mathbb{P}(\mathcal{A}_{\mathrm{sub}} \subseteq \mathcal{A} \subseteq \mathcal{A}_{\mathrm{sup}}) &= \mathbb{E}_{P_\phi}[\mathbb{1}(\mathcal{A}_{\mathrm{sub}} \subseteq \mathcal{A} \subseteq \mathcal{A}_{\mathrm{sup}})] \\ &= \mathbb{E}_Q[w(\mathcal{A})\mathbb{1}(\mathcal{A}_{\mathrm{sub}} \subseteq \mathcal{A} \subseteq \mathcal{A}_{\mathrm{sup}})].\end{aligned} \tag{10}$$

The Monte Carlo method for (10) is unbiased for $\mathbb{P}(\mathcal{A}_{\mathrm{sub}} \subseteq \mathcal{A} \subseteq \mathcal{A}_{\mathrm{sup}})$, but the critical issue remains: most Monte Carlo samples yield zero when $|\mathcal{A}_{\mathrm{sup}}\backslash\mathcal{A}_{\mathrm{sub}}|$ is small owing to the indicator function. If we use Monte Carlo samples on $2^{\mathcal{A}_{\mathrm{sup}}\backslash\mathcal{A}_{\mathrm{sub}}}$ instead of $2^{\mathcal{Y}}$, the number of wasted samples could be reduced. The following proposition realizes this.

Proposition 4.1. *Let P be a probability function on $2^{\mathcal{Y}}$ and $\boldsymbol{\xi}$ be a random vector following the independent Bernoulli trials $Q_{\boldsymbol{q}}$ defined in* (2)*. Then,*

$$\mathbb{E}_{\boldsymbol{\xi}}\left[\frac{P(\mathcal{A}_{\boldsymbol{\xi}})}{\prod_{i\in\mathcal{A}_{\mathrm{sup}}\backslash\mathcal{A}_{\mathrm{sub}}} Q_{q_i}(\xi_i)}\middle|\begin{matrix}\xi_i = 1 & (i \in \mathcal{A}_{\mathrm{sub}}) \\ \xi_j = 0 & (j \in \mathcal{Y}\backslash\mathcal{A}_{\mathrm{sup}})\end{matrix}\right] = \mathbb{P}(\mathcal{A}_{\mathrm{sub}} \subseteq \mathcal{A} \subseteq \mathcal{A}_{\mathrm{sup}}) \tag{11}$$

and

$$\mathrm{Var}_{\boldsymbol{\xi}}\left[\frac{P(\mathcal{A}_{\boldsymbol{\xi}})}{\prod_{i\in\mathcal{A}_{\mathrm{sup}}\backslash\mathcal{A}_{\mathrm{sub}}} Q_{q_i}(\xi_i)}\middle|\begin{matrix}\xi_i = 1 & (i \in \mathcal{A}_{\mathrm{sub}}) \\ \xi_j = 0 & (j \in \mathcal{Y}\backslash\mathcal{A}_{\mathrm{sup}})\end{matrix}\right] \leq \mathrm{Var}_{\boldsymbol{\xi}}[w(\mathcal{A}_{\boldsymbol{\xi}})\mathbb{1}(\mathcal{A}_{\mathrm{sub}} \subseteq \mathcal{A}_{\boldsymbol{\xi}} \subseteq \mathcal{A}_{\mathrm{sup}})]$$

hold, where $w(\mathcal{A}_{\boldsymbol{\xi}}) = P(\mathcal{A}_{\boldsymbol{\xi}})/Q_{\boldsymbol{q}}(\boldsymbol{\xi})$.

The proof of Proposition 4.1 mainly depends on the tower properties of expectation and variance (the detailed proof is in Appendix A). Such variance reduction techniques are called Rao–Blackwellization and are commonly used in computational statistics (Casella and Robert, 1996; Doucet et al., 2000). Since only $\{\xi_i\}_{i\in\mathcal{A}_{\mathrm{sup}}\backslash\mathcal{A}_{\mathrm{sub}}}$ are required as Monte Carlo samples owing to the independence of the proposal distribution $Q_{\boldsymbol{q}}$, and the indicator function no longer appears in (11), we can expect a significant reduction in variance when evaluating the marginal probability. Rao–Blackwellization is effective in estimating marginal probabilities other than $\mathbb{P}(\mathcal{A}_{\mathrm{sub}} \subseteq \mathcal{A} \subseteq \mathcal{A}_{\mathrm{sup}})$.

Proposition 4.2. *Let $\mathcal{A}$ be a random subset following the probability function $P : 2^{\mathcal{Y}} \to [0,1]$. Then, we have*

$$\mathbb{P}(|\mathcal{A}| = k) = \binom{N}{k}\mathbb{E}_{\mathcal{A}^k\sim Q^k}[P(\mathcal{A}^k)], \tag{12}$$

where Q^k is the uniform distribution on $\{\mathcal{A} \subseteq \mathcal{Y} : |\mathcal{A}| = k\}$.

The proof of 4.2 is also shown in Appendix A. Proposition 4.2 provides an accurate Monte Carlo method to evaluate the marginal probability $\mathbb{P}(|\mathcal{A}| = k)$ for a given $k \in \{1,\dots,N\}$. Simply take Monte Carlo samples consisting of uniformly randomly chosen k items in $\mathcal{Y}$ and evaluate (12).

Conditional Probability

The conditional probability

$$P_\phi(\mathcal{A}|\mathcal{A}_{\mathrm{sub}} \subseteq \mathcal{A} \subseteq \mathcal{A}_{\mathrm{sup}};\boldsymbol{L}) = \frac{P_\phi(\mathcal{A};\boldsymbol{L})}{\mathbb{P}(\mathcal{A}_{\mathrm{sub}} \subseteq \mathcal{A} \subseteq \mathcal{A}_{\mathrm{sup}})} \tag{13}$$

can be accurately estimated by approximating the marginal probability $\mathbb{P}(\mathcal{A}_{\mathrm{sub}} \subseteq \mathcal{A} \subseteq \mathcal{A}_{\mathrm{sup}})$ using

Proposition 4.1. Note that the approximation of the normalizing constants within (11) is not required if we are only interested in the conditional probability because they cancel out in the numerator and denominator in (13). The same applies to $P_\phi(\mathcal{A}||\mathcal{A}| = k; \boldsymbol{L})$.

5 LEARNING DKPPs

Since the kernel matrix $\boldsymbol{L}$ of DKPPs indicates the pairwise similarity for each $(i, j) \in \mathcal{Y} \times \mathcal{Y}$, we can build it directly when the items have feature vectors. Otherwise, we may need to learn the kernel matrix $\boldsymbol{L}$ from observations.

Let $\mathcal{A}_1, \dots, \mathcal{A}_M \in \mathcal{Y}$ be the observed data series with M samples. The simplest way to learn $\boldsymbol{L}$ is through the maximum likelihood estimation (MLE). For $\phi = \log$, say for DPPs, several algorithms have been developed to learn $\boldsymbol{L}$ through MLE (Gillenwater et al., 2014; Mariet and Sra, 2015, 2016; Gartrell et al., 2017; Dupuy and Bach, 2018; Kawashima and Hino, 2023). However, the term $\log Z_\phi(\boldsymbol{L})$ makes it difficult to learn $\boldsymbol{L}$ directly for general ϕ.

As an alternative to MLE, we propose using ratio matching, which was developed by (Hyvärinen, 2007) as a discrete extension of score matching (Hyvärinen, 2005). According to the prescription of the ratio matching, the objective function to minimize is

$$J(\boldsymbol{L}) \coloneqq \frac{1}{M} \sum_{m,n} g\left(\exp(\mathrm{tr}\phi(\boldsymbol{L}[\mathcal{A}_m]) - \mathrm{tr}\phi(\boldsymbol{L}[\mathcal{A}_m^{\bar{n}}]))\right)^2, \tag{14}$$

$$\text{where} \quad g(x) \coloneqq \frac{1}{1+x}, \ \mathcal{A}^{\bar{n}} \coloneqq \begin{cases} \mathcal{A}\backslash\{n\} & (n \in \mathcal{A}), \\ \mathcal{A} \cup \{n\} & (n \notin \mathcal{A}), \end{cases}$$

for DKPPs. Although we cannot evaluate the exponent term in (14) more efficiently than by direct computation for general ϕ, we can employ stochastic gradient descent (SGD) because J has a summation form. Let $\Omega \subseteq \{(m, n) : m = 1, \dots, M, \ n = 1, \dots, N\}$ be the minibatch and $\kappa \coloneqq \max\{|\mathcal{A}_1|, \dots, |\mathcal{A}_M|\}$. The time complexity of computing the gradient $\partial J / \partial \boldsymbol{L}$ is $\mathcal{O}(|\Omega|\kappa^3 + |\Omega|N^2)$, which no longer depends on the sample size M and the number of items N, except for the term $|\Omega|N^2$ owing to $|\Omega|$ additions of $N \times N$ matrices (see Appendix C for derivation). This ensures the good scalability of the algorithm. Optionally, the variance reduction technique for SGD of the ratio matching can be applied (Liu et al., 2022).

6 EXPERIMENTS

All experiments were performed on a MacBook Pro (2023, macOS 14.3) with an Apple M3 Pro chip and 36GB RAM.

Algorithm 1: Subset Acquiring Algorithm

Input: DKPP $P_\phi(\cdot; \boldsymbol{L})$ and the subset size k
Output: $\mathcal{A}$
Initialize $\mathcal{A} \leftarrow \{\}, \ \mathcal{B} \leftarrow \{1, \dots, N\}$;
for $t = 1$ *to* k **do**
  Sample $j \in \mathcal{B}$ with probability proportional to weights $w_i = \tilde{P}_\phi(\mathcal{A} \cup \{i\}|\boldsymbol{L}), \ i \in \mathcal{B}$;
  Update $\mathcal{A} \leftarrow \mathcal{A} \cup \{i\}, \ \mathcal{B} \leftarrow \mathcal{B}\backslash\{j\}$;
end

6.1 Controllability of Positive and Negative Dependence

To verify that DKPPs can flexibly control positive and negative dependence, we experiment on the behavior of the conditional probability $P_\phi(\mathcal{A} \mid |\mathcal{A}| = k; \boldsymbol{L})$. We prepare a 20×20 grid of points in $\mathbb{R}^2$ as the ground set $\mathcal{Y}$, so that $N = 20^2 = 400$. The kernel matrix $\boldsymbol{L}$ is constructed from the Gaussian kernel with unit bandwidth. Then, we create two types of realization on $2^\mathcal{Y}$: `SCATTERED` and `GATHERED`, both containing $k = |\mathcal{A}| = 25$ items. `SCATTERED` is obtained by applying an approximate algorithm for submodular maximization (Buchbinder et al., 2014) to a DPP, whereas `GATHERED` is created manually. We evaluate the conditional probabilities $P_{\phi_\lambda}(\mathcal{A} \mid |\mathcal{A}| = k; \boldsymbol{L})$ where ϕ_λ is the Box–Cox transformation (6). We execute 30 trials of Monte Carlo approximations using (12) with 1,000 samples to estimate the conditional probabilities.

The results are shown in Figure 1. The error bars are present but nearly invisible owing to the remarkably small approximation variance. As expected, the probability of `SCATTERED` decreases monotonically and the probability of `GATHERED` increases monotonically with λ. Interestingly, the probabilities of `SCATTERED` and `GATHERED` reverse at $\lambda = 1$. This result aligns exactly with the theory; the switch between positive and negative dependence occurs at $\lambda = 1$, as explained in Subsection 3.1.

6.2 Subset Acquisition

Attractive and repulsive subset acquisition may be the most direct application of the DKPPs. We randomly pick $N = 2{,}000$ images of the MNIST dataset (Lecun et al., 1998) with containing 200 instances from each digit class. Then, the Gaussian kernel with the median heuristic is applied to make the kernel matrix $\boldsymbol{L} \in \mathbb{R}^{2{,}000 \times 2{,}000}$. Algorithm 1 is the heuristic algorithm to acquire subsets from a DKPP. Given the kernel matrix, we run the algorithm to acquire subsets of size 5 from $P_{\phi_{\beta,\lambda}}$ and measure the number of dis-

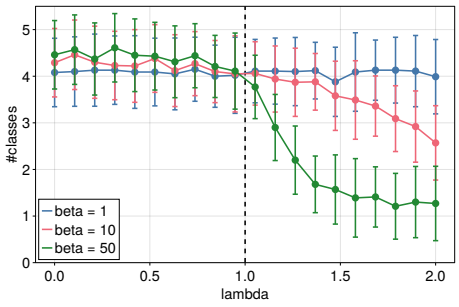


Figure 2: Number of distinct classes within the acquired subsets.

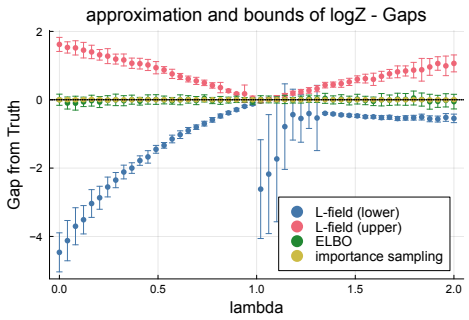


Figure 3: Evaluated gaps $\log Z_\phi^{\text{approx}}(\boldsymbol{L}) - \log Z_\phi(\boldsymbol{L})$.

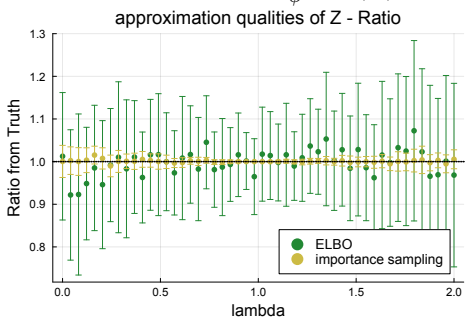


Figure 4: Evaluated ratios $Z_\phi^{\text{approx}}(\boldsymbol{L})/Z_\phi(\boldsymbol{L})$.

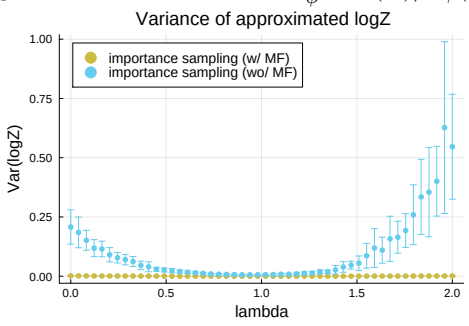


Figure 5: Evaluated variances of importance sampling with and without mean-field approximation.

tinct digit classes within each subset. A larger value of this metric indicates more repulsive subset acquisition, while a smaller value suggests more attractive.

Figure 2 shows the metric values for $\beta \in \{1, 10, 50\}$, $\lambda \in [0, 2]$ with the mean and standard deviations of 100 trials. Although the values are not significantly different in $\beta = 1$, it smoothly decreases as λ increases for $\beta = 10$; the dependency control is realized. We can find the dependency is more intensified for $\beta = 50$. Additionally, we visually show the acquired instances in Appendix D.

Notably this procedure involves no learning process at all. The entire process is forward-only, demonstrating the advantage of the DKPP's clear and practicable parameterization via the kernel matrix and the dependency controlling function ϕ.

6.3 Approximating the Normalizing Constant

We assess some approximating or bounding methods for $\log Z_\phi(\boldsymbol{L})$. We use the Box–Cox transformation (6) for ϕ and the hyperparameter $\lambda \in [0, 2]$. We calculate the mean-field ELBOs and demonstrate the importance sampling described in Subsection 4.3. Additionally, we apply the L-field (Djolonga and Krause, 2014; Djolonga et al., 2018) to obtain both the upper and lower bounds of $\log Z_\phi(\boldsymbol{L})$ since the DKPP is log-submodular or log-supermodular for such λ. For all experiments reported in this subsection, we ran 30 trials and showed the means and standard deviations in the figures.

First, we set $N = 16$ and obtain the kernel parameters from $\boldsymbol{L} \sim \text{Wishart}(\boldsymbol{L}; N, \boldsymbol{I})/N$ for each hyperparameter $\lambda \in [0, 2]$. Then, we evaluate the gaps from the ground truth $\log Z_{\phi_\lambda}^{\text{approx}}(\boldsymbol{L}) - \log Z_{\phi_\lambda}(\boldsymbol{L})$ for each pair of $(\lambda, \boldsymbol{L})$. Figure 3 shows the results. Although the ELBO and importance sampling seem to yield good estimates of the normalizing constant, the gaps of the L-fields become larger as λ moves away from 1. Figure 4 shows the more detailed behavior of the ELBO and proposed importance sampling through the ratio $Z_{\phi_\lambda}^{\text{approx}}(\boldsymbol{L})/Z_{\phi_\lambda}(\boldsymbol{L})$. Although the ELBO constructs a theoretical lower bound of the normalizing constant, it can take larger values than the ground truth owing to the evaluation error of the Monte Carlo approximation for the multilinear extension. We can see that the importance sampling achieves a better estimator than the ELBO, and the variance of the importance sampling also increases as λ moves away from 1, since the mean-field approximation for the proposal distribution becomes less accurate.

As an ablation study, we compare the importance sampling with and without the mean-field approximation for the proposal distribution. We set $N = 64$ and evaluate $\text{Var}(\log Z_{\phi_\lambda}^{\text{approx}}(\boldsymbol{L}))$ for each λ. We use $q_1 = \cdots = q_N = 0.5$ for the without-mean-field pro-

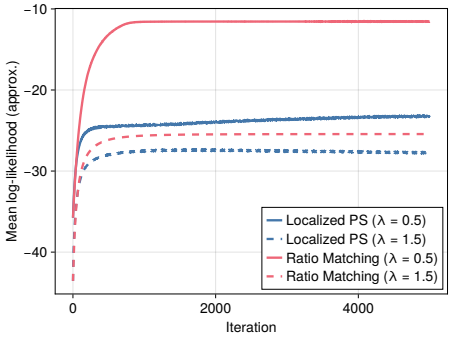


Figure 6: Learning curves with `media`. The LPSD takes 7.925×10^{-2} s per iteration and ratio matching takes 2.174×10^{-3} s per iteration.

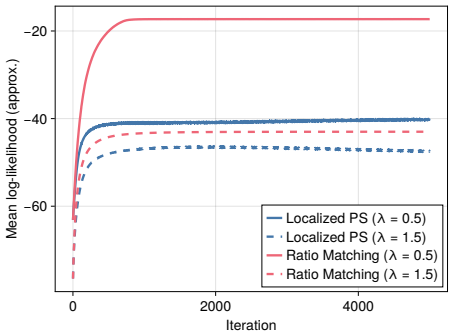


Figure 7: Learning curves with `apparel`. The LPSD takes 6.091×10^{-1} s per iteration and ratio matching takes 4.155×10^{-3} s per iteration.

posal. As shown in Figure 5, the proposal distributions with the mean-field approximation significantly reduce the variance of the importance sampling.

6.4 Learning the Kernel Parameter

We demonstrate kernel learning methods on the Amazon Baby Registry dataset (Gillenwater et al., 2014). The dataset contains 13 categories of childcare products, and we use `media` ($N = 58$, $M = 5{,}904$) and `apparel` ($N = 100, M = 14{,}970$) categories. We compare the ratio matching introduced in Section 5 with the localized pseudo-spherical divergence (LPSD) (Takenouchi and Kanamori, 2015, 2017), which can measure the discrepancy between an empirical distribution and a probability function with finite support while avoiding the calculation of the normalizing constant. We set the Box–Cox transformation hyperparameter $\lambda = 0.5, 1.5$ and the minibatch size for ratio matching $|\Omega| = 100$. Figures 6 and 7 show the learning curves. The normalizing constants are evaluated approximately by importance sampling. Overall, ratio matching performs better in both convergence and computational time.

7 CONCLUSION

In this study, we introduced DKPPs, a new family of distributions for random subsets that can seamlessly control positive and negative dependence. We developed methods for the probabilistic operations and inference in DKPPs for practical use, including the evaluation of expectations, the normalizing constant, and marginal and conditional probabilities. We also proposed an efficient learning method based on ratio matching. Empirically, we demonstrated the controllability of positive and negative dependence in DKPPs and conducted a subset acquisition experiment as the representative application. The effectiveness of the proposed computational algorithms was also shown numerically.

Although in this paper, we mainly focused on conceptual and practical motivation, many theoretical questions remain open. How do DKPPs connect to point processes on continuous spaces like the marginal kernel representation of DPPs? When do stronger conditions than log-submodularity hold as negative dependence in a DKPP? Furthermore, we will seek an understanding of how broad the DKPP family is as a set of distributions.

Acknowledgements

We thank Satoshi Kuriki, Keisuke Yano, Yuki Saito, Yuki Hirakawa, Takuya Furusawa, and anonymous reviewers for helpful comments. This work was supported by JST CREST Grant Number JPMJCR2015, JSPS KAKENHI Grant Numbers JP22H03653 and 23H04483, and JST the establishment of university fellowships towards the creation of science technology innovation Grant Number JPMJFS2136.

References

Ackley, D. H., Hinton, G. E., and Sejnowski, T. J. (1985). A learning algorithm for boltzmann machines. *Cognitive Science*, 9(1):147–169.

An, M. (1996). Log-concave Probability Distributions: Theory and Statistical Testing. Game Theory and Information, University Library of Munich, Germany.

Bach, F. (2013). *Learning with Submodular Functions: A Convex Optimization Perspective.*

Balog, M., Tripuraneni, N., Ghahramani, Z., and Weller, A. (2017). Lost Relatives of the Gumbel Trick. In *Proceedings of the 34th International Conference on Machine Learning*, pages 371–379.

Bhatia, R. (1997). *Matrix Analysis*, volume 169 of *Graduate Texts in Mathematics*.

Borcea, J., Brändén, P., and Liggett, T. M. (2009). Negative Dependence and the Geometry of Polynomials. *Journal of the American Mathematical Society*, 22(2):521–567.

Borodin, A. and Rains, E. M. (2005). Eynard–Mehta Theorem, Schur Process, and Their Pfaffian Analogs. *Journal of Statistical Physics*, 121(3):291–317.

Borzadaran, G. M. and Borzadaran, H. M. (2011). Log-concavity property for some well-known distributions. *Surveys in Mathematics and its Applications*, 6:203–219.

Buchbinder, N., Feldman, M., Naor, J., and Schwartz, R. (2012). A Tight Linear Time (1/2)-Approximation for Unconstrained Submodular Maximization. In *2012 IEEE 53rd Annual Symposium on Foundations of Computer Science*, pages 649–658.

Buchbinder, N., Feldman, M., Naor, J. S., and Schwartz, R. (2014). Submodular Maximization with Cardinality Constraints. In *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1433–1452.

Calinescu, G., Chekuri, C., Pál, M., and Vondrák, J. (2007). Maximizing a Submodular Set Function Subject to a Matroid Constraint (Extended Abstract). In *Integer Programming and Combinatorial Optimization*, pages 182–196.

Casella, G. and Robert, C. P. (1996). Rao-Blackwellisation of Sampling Schemes. *Biometrika*, 83(1):81–94.

Chakrabarty, D., Jain, P., and Kothari, P. (2014). Provable Submodular Minimization using Wolfe' s Algorithm. In *Advances in Neural Information Processing Systems*, volume 27.

Chekuri, C., Vondrák, J., and Zenklusen, R. (2014). Submodular Function Maximization via the Multilinear Relaxation and Contention Resolution Schemes. *SIAM Journal on Computing*, 43(6):1831–1879.

Derezinski, M., Liang, F., and Mahoney, M. (2020a). Bayesian Experimental Design Using Regularized Determinantal Point Processes. In *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, pages 3197–3207.

Derezinski, M., Liang, F. T., Liao, Z., and Mahoney, M. W. (2020b). Precise expressions for random projections: Low-rank approximation and randomized Newton. In *Advances in Neural Information Processing Systems*, volume 33, pages 18272–18283.

Dereziński, M. and Mahoney, M. W. (2021). Determinantal Point Processes in Randomized Numerical Linear Algebra. *Notices of the American Mathematical Society*, 68(01):1.

Dereziński, M., Warmuth, M. K., and Hsu, D. (2022). Unbiased Estimators for Random Design Regression. *Journal of Machine Learning Research*, 23(167):1–46.

Diaconis, P. and Evans, S. N. (2000). Immanants and Finite Point Processes. *Journal of Combinatorial Theory, Series A*, 91(1):305–321.

Djolonga, J., Jegelka, S., and Krause, A. (2018). Provable Variational Inference for Constrained Log-Submodular Models. In *Advances in Neural Information Processing Systems*, volume 31.

Djolonga, J. and Krause, A. (2014). From MAP to Marginals: Variational Inference in Bayesian Submodular Models. In *Advances in Neural Information Processing Systems*, volume 27.

Doucet, A., de Freitas, N., Murphy, K. P., and Russell, S. J. (2000). Rao-blackwellised particle filtering for dynamic bayesian networks. In *Conference on Uncertainty in Artificial Intelligence*.

Dupuy, C. and Bach, F. (2018). Learning Determinantal Point Processes in Sublinear Time. In *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*, pages 244–257.

Feige, U., Peleg, D., and Kortsarz, G. (2001). The Dense k -Subgraph Problem. *Algorithmica*, 29(3):410–421.

Fortuin, C. M., Ginibre, J., and Kasteleyn, P. W. (1971). Correlation inequalities on some partially ordered sets. *Communications in Mathematical Physics*, 22(2):89–103.

Friedland, S. and Gaubert, S. (2013). Submodular spectral functions of principal submatrices of a hermitian matrix, extensions and applications. *Linear Algebra and its Applications*, 438(10):3872–3884.

Fujishige, S., Hayashi, T., and Isotani, S. (2006). The Minimum-Norm-Point Algorithm Applied to Submodular Function Minimization and Linear Programming.

Fujishige, S. and Isotani, S. (2011). A submodular function minimization algorithm based on the minimum-norm base. *Pacific Journal of Optimization*, 7(1):3–17.

Garey, M. R. and Johnson, D. S. (1979). *Computers and intractability : a guide to the theory of NP-completeness*.

Gartrell, M., Paquet, U., and Koenigstein, N. (2017). Low-Rank Factorization of Determinantal Point

Processes. *Proceedings of the AAAI Conference on Artificial Intelligence*, 31(1).

Gillenwater, J., Kulesza, A., and Taskar, B. (2012). Near-Optimal MAP Inference for Determinantal Point Processes. In *Advances in Neural Information Processing Systems*, volume 25.

Gillenwater, J. A., Kulesza, A., Fox, E., and Taskar, B. (2014). Expectation-Maximization for Learning Determinantal Point Processes. In *Advances in Neural Information Processing Systems*, volume 27.

Gotovos, A., Hassani, H., and Krause, A. (2015). Sampling from Probabilistic Submodular Models. In *Advances in Neural Information Processing Systems*, volume 28.

Hazan, T. and Jaakkola, T. S. (2012). On the partition function and random maximum a-posteriori perturbations. In *Proceedings of the 29th International Conference on Machine Learning, ICML 2012*.

Hyvärinen, A. (2005). Estimation of Non-Normalized Statistical Models by Score Matching. *Journal of Machine Learning Research*, 6(24):695–709.

Hyvärinen, A. (2007). Some extensions of score matching. *Computational Statistics & Data Analysis*, 51(5):2499–2512.

Iwata, S., Fleischer, L., and Fujishige, S. (2001). A combinatorial strongly polynomial algorithm for minimizing submodular functions. *Journal of the ACM*, 48(4):761–777.

Iyer, R. and Bilmes, J. (2015). Submodular Point Processes with Applications to Machine learning. In *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics*, pages 388–397.

Iyer, R., Jegelka, S., and Bilmes, J. (2013). Fast semidifferential-based submodular function optimization. In *Proceedings of the 30th International Conference on International Conference on Machine Learning - Volume 28*, ICML'13, pages III–855–III–863.

Kawashima, T. and Hino, H. (2023). Minorization-Maximization for Learning Determinantal Point Processes. *Transactions on Machine Learning Research*.

Kulesza, A. and Taskar, B. (2011). K-Dpps: Fixed-Size Determinantal Point Processes. In *International Conference on Machine Learning*.

Kulesza, A. and Taskar, B. (2012). Determinantal Point Processes for Machine Learning. *Foundations and Trends® in Machine Learning*, 5(2-3):123–286.

Lecun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.

Liu, M., Liu, H., and Ji, S. (2022). Gradient-Guided Importance Sampling for Learning Binary Energy-Based Models. In *The Eleventh International Conference on Learning Representations*.

Macchi, O. (1975). The Coincidence Approach to Stochastic Point Processes. *Advances in Applied Probability*, 7(1):83–122.

Mariet, Z. and Sra, S. (2015). Fixed-Point Algorithms for Learning Determinantal Point Processes. In *Proceedings of the 32nd International Conference on Machine Learning*, pages 2389–2397.

Mariet, Z. E. and Sra, S. (2016). Kronecker Determinantal Point Processes. In *Advances in Neural Information Processing Systems*, volume 29.

Mariet, Z. E., Sra, S., and Jegelka, S. (2018). Exponentiated Strongly Rayleigh Distributions. In *Advances in Neural Information Processing Systems*, volume 31.

Mothilal, R. K., Sharma, A., and Tan, C. (2020). Explaining machine learning classifiers through diverse counterfactual explanations. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, FAT* '20, pages 607–617.

Nemhauser, G. L., Wolsey, L. A., and Fisher, M. L. (1978). An analysis of approximations for maximizing submodular set functions—I. *Mathematical Programming*, 14(1):265–294.

Orlin, J. B. (2009). A faster strongly polynomial time algorithm for submodular function minimization. *Mathematical Programming*, 118(2):237–251.

Papandreou, G. and Yuille, A. L. (2011). Perturb-and-MAP random fields: Using discrete optimization to learn and sample from energy models. In *2011 International Conference on Computer Vision*, pages 193–200.

Pemantle, R. (2000). Towards a theory of negative dependence. *Journal of Mathematical Physics*, 41(3):1371–1390.

Rebeschini, P. and Karbasi, A. (2015). Fast Mixing for Discrete Point Processes. In *Proceedings of The 28th Conference on Learning Theory*, pages 1480–1500.

Rubinstein, R. Y. and Kroese, D. P. (2008). *Simulation and the Monte Carlo Method*. Wiley Series in Probability and Statistics. 2nd ed edition.

Schölkopf, B., Smola, A., and Müller, K.-R. (1997). Kernel principal component analysis. In *Artificial Neural Networks — ICANN'97*, volume 1327, pages 583–588.

Schrijver, A. (2000). A Combinatorial Algorithm Minimizing Submodular Functions in Strongly Polynomial Time. *Journal of Combinatorial Theory, Series B*, 80(2):346–355.

Svitkina, Z. and Fleischer, L. (2011). Submodular Approximation: Sampling-based Algorithms and Lower Bounds. *SIAM Journal on Computing*, 40(6):1715–1737.

Takenouchi, T. and Kanamori, T. (2015). Empirical Localization of Homogeneous Divergences on Discrete Sample Spaces. In *Advances in Neural Information Processing Systems*, volume 28.

Takenouchi, T. and Kanamori, T. (2017). Statistical Inference with Unnormalized Discrete Models and Localized Homogeneous Divergences. *Journal of Machine Learning Research*, 18(56):1–26.

Tschiatschek, S., Djolonga, J., and Krause, A. (2016). Learning Probabilistic Submodular Diversity Models Via Noise Contrastive Estimation. In *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, pages 770–779.

Valiant, L. G. (1979). The complexity of computing the permanent. *Theoretical Computer Science*, 8(2):189–201.

Vere-Jones, D. (1997). Alpha-permanents and their applications to multivariate gamma, negative binomial and ordinary binomial distributions. *New Zealand J. Math*, 26(1):125–149.

Wilhelm, M., Ramanathan, A., Bonomo, A., Jain, S., Chi, E. H., and Gillenwater, J. (2018). Practical Diversified Recommendations on YouTube with Determinantal Point Processes. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, pages 2165–2173.

Zhang, R., Liu, X., and Liu, Q. (2022). A Langevin-like Sampler for Discrete Distributions. In *Proceedings of the 39th International Conference on Machine Learning*, pages 26375–26396.

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes]
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [No]
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [Yes]

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. [Yes]
   (b) Complete proofs of all theoretical results. [Yes]
   (c) Clear explanations of any assumptions. [Yes]

3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [Yes]
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes]
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Yes]
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Yes]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. [Yes]
   (b) The license information of the assets, if applicable. [Not Applicable]
   (c) New assets either in the supplemental material or as a URL, if applicable. [Not Applicable]
   (d) Information about consent from data providers/curators. [Yes]
   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Not Applicable]

5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots. [Not Applicable]
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Not Applicable]
   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Not Applicable]

A PROOFS

A.1 Proof of Proposition 3.3

Proof. Let $\mathcal{A} \subseteq \mathcal{Y}$ be a random subset following the DKPP $P_\phi(\mathcal{A}; \boldsymbol{L})$, and let $\boldsymbol{z} = (z_1, \dots, z_N)^\top$ denote the indicator vector of $\mathcal{A}$:

$$z_i = \begin{cases} 0 & \text{if } i \notin \mathcal{A}, \\ 1 & \text{if } i \in \mathcal{A}, \end{cases}$$

for $i = 1, \dots, N$. Then, we obtain

$$\begin{aligned}
P_\phi(\mathcal{A}; \boldsymbol{L}) &\propto \exp \mathrm{tr} \phi(\boldsymbol{L}[\mathcal{A}]) \\
&= \exp \mathrm{tr} \left(a\boldsymbol{L}[\mathcal{A}]\boldsymbol{L}[\mathcal{A}] + b\boldsymbol{L}[\mathcal{A}] + c\boldsymbol{I} \right) \\
&= \exp \left(a \mathrm{tr} \boldsymbol{L}[\mathcal{A}]\boldsymbol{L}[\mathcal{A}] + b \mathrm{tr} \boldsymbol{L}[\mathcal{A}] + c|\mathcal{A}| \right) \\
&= \exp \left(a \|\boldsymbol{L}[\mathcal{A}]\|_F^2 + b \mathrm{tr} \boldsymbol{L}[\mathcal{A}] + c|\mathcal{A}| \right) \\
&= \exp \left(a \sum_{i,j \in \mathcal{A}} |L_{ij}|^2 + \sum_{i \in \mathcal{A}} (bL_{ii} + c) \right) \\
&= \exp \left(a \sum_{i,j=1}^N |L_{ij}|^2 z_i z_j + \sum_{i=1}^N (bL_{ii} + c) z_i \right) \\
&= \exp \left(a \sum_{i \neq j} |L_{ij}|^2 z_i z_j + a \sum_{i=1}^N L_{ii}^2 z_i^2 + \sum_{i=1}^N (bL_{ii} + c) z_i \right) \\
&= \exp \left(a \sum_{i \neq j} |L_{ij}|^2 z_i z_j + a \sum_{i=1}^N L_{ii}^2 z_i + \sum_{i=1}^N (bL_{ii} + c) z_i \right) \\
&= \exp \left(a \sum_{i \neq j} |L_{ij}|^2 z_i z_j + \sum_{i=1}^N (aL_{ii}^2 + bL_{ii} + c) z_i \right). \qquad \text{(A.1)}
\end{aligned}$$

By comparing (A.1) and (4), we prove the proposition. □

A.2 Proof of Proposition 3.5

Proof.
[ϕ is affine $\implies$ DKPP is log-modular]

We denote the diagonal matrix with the eigenvalues of $\boldsymbol{L}[\mathcal{A}]$ by $\boldsymbol{\Lambda}_\mathcal{A}$. By letting $\phi(x) = bx + c$ from the assumption, we find that the log-likelihood of the DKPP is

$$\begin{aligned}
\log P_\phi(\mathcal{A}; \boldsymbol{L}) = \mathrm{tr} \phi(\boldsymbol{L}[\mathcal{A}]) - \log Z_\phi(\boldsymbol{L}) &= \mathrm{tr}(b\boldsymbol{\Lambda}_\mathcal{A} + c\boldsymbol{I}) - \log Z_\phi(\boldsymbol{L}) \\
&= b \mathrm{tr} \boldsymbol{\Lambda}_\mathcal{A} + c|\mathcal{A}| - \log Z_\phi(\boldsymbol{L}) \\
&= b \mathrm{tr} \boldsymbol{L}_\mathcal{A} + c|\mathcal{A}| - \log Z_\phi(\boldsymbol{L}) \\
&= \sum_{i \in \mathcal{A}} (bL_{ii} + c) - \log Z_\phi(\boldsymbol{L}).
\end{aligned}$$

This is a modular function for $\mathcal{A} \subseteq \mathcal{Y}$.

[DKPP is log-modular $\implies$ ϕ is affine]

The goal of this proof is show the affinity of the single-variable function $\phi : \mathbb{R}_{\geq 0} \to \mathbb{R}$. From the log-modularity of the DKPP, we have

$$\mathrm{tr} \phi(\boldsymbol{L}[\mathcal{S}]) + \mathrm{tr} \phi(\boldsymbol{L}[\mathcal{T}]) = \mathrm{tr} \phi(\boldsymbol{L}[\mathcal{S} \cup \mathcal{T}]) + \mathrm{tr} \phi(\boldsymbol{L}[\mathcal{S} \cap \mathcal{T}]) \qquad \text{(A.2)}$$

for every $\mathcal{S}, \mathcal{T} \subseteq \mathcal{Y}$. By defining $g(x) \coloneqq \phi(x) - \phi(0)$, we can represent ϕ as $\phi(x) = g(x) + \phi(0)$. Note that g is continuous because of the continuity of ϕ, assumed in Definition 3.1. Now, we take $\mathcal{S} = \{i\}, \mathcal{T} = \{j\}$ $(i \neq j)$ and the kernel matrix $\boldsymbol{L}$ arbitrarily to satisfy $L_{ij} = \sqrt{L_{ii}L_{jj}} \in \mathbb{R}$. Since $\boldsymbol{L}$ and ϕ are independent, this choice of $\boldsymbol{L}$ does not impose any restriction on ϕ.

The two eigenvalues of $\boldsymbol{L}[\mathcal{S} \cup \mathcal{T}] \in \mathbb{R}^{2\times 2}$ are given by $\lambda_1 = L_{ii} + L_{jj}$ and $\lambda_2 = 0$. The l.h.s. of (A.2) is

$$\mathrm{tr}\phi(\boldsymbol{L}[\mathcal{S}]) + \mathrm{tr}\phi(\boldsymbol{L}[\mathcal{T}]) = \phi(L_{ii}) + \phi(L_{jj}),$$

and the r.h.s. is

$$\mathrm{tr}\phi(\boldsymbol{L}[\mathcal{S} \cup \mathcal{T}]) + \mathrm{tr}\phi(\boldsymbol{L}[\mathcal{S} \cap \mathcal{T}]) = \mathrm{tr}\phi(\boldsymbol{L}[\mathcal{S} \cup \mathcal{T}]) = \phi(\lambda_1) + \phi(\lambda_2).$$

Therefore, we have the identity

$$\phi(L_{ii}) + \phi(L_{jj}) = \phi(\lambda_1) + \phi(\lambda_2). \tag{A.3}$$

By substituting $\phi(x) = g(x) + \phi(0)$ into (A.3), it becomes

$$\begin{aligned}
&\{g(L_{ii}) + \phi(0)\} + \{g(L_{jj}) + \phi(0)\} = \{g(\lambda_1) + \phi(0)\} + \{g(\lambda_2) + \phi(0)\} \\
&\quad \iff g(L_{ii}) + g(L_{jj}) = g(\lambda_1) + g(\lambda_2) = g(L_{ii} + L_{jj}) + g(0) = g(L_{ii} + L_{jj}).
\end{aligned}$$

Because L_{ii} and L_{jj} are arbitrary on $\mathbb{R}_{\geq 0}$, we have $g(x)+g(y) = g(x+y)$ for all $x, y \in \mathbb{R}_{\geq 0}$. From the continuity of g, this means that g must be linear and ϕ must be affine. □

A.3 Proof of Proposition 4.1

Proof. We consider the partition of $\mathcal{Y}$ given by $\mathcal{A}_{\mathrm{sub}}, \mathcal{Y}\backslash\mathcal{A}_{\mathrm{sup}}, \mathcal{A}_{\mathrm{sup}}\backslash\mathcal{A}_{\mathrm{sub}}$ and separate the random vector $\boldsymbol{\xi} = (\xi_1, \ldots, \xi_N)^\top$ by $\boldsymbol{\xi}^+ \coloneqq (\xi_i)_{i\in\mathcal{A}_{\mathrm{sub}}}, \boldsymbol{\xi}^- \coloneqq (\xi_i)_{i\in\mathcal{Y}\backslash\mathcal{A}_{\mathrm{sup}}}$, and $\boldsymbol{\xi}^\circ \coloneqq (\xi_i)_{i\in\mathcal{A}_{\mathrm{sup}}\backslash\mathcal{A}_{\mathrm{sub}}}$. For example, when $\mathcal{Y} = \{1,2,3,4,5\}, \mathcal{A}_{\mathrm{sub}} = \{1,2\}$, and $\mathcal{A}_{\mathrm{sup}} = \{1,2,4,5\} (\supseteq \mathcal{A}_{\mathrm{sub}})$, the separation of $\boldsymbol{\xi}$ becomes $\boldsymbol{\xi}^+ = (\xi_1, \xi_2)^\top, \boldsymbol{\xi}^- = (\xi_3)$, and $\boldsymbol{\xi}^\circ = (\xi_4, \xi_5)^\top$. We also denote $N^+ \coloneqq |\mathcal{A}_{\mathrm{sub}}|$, $N^- \coloneqq |\mathcal{Y}\backslash\mathcal{A}_{\mathrm{sup}}|$, and $N^\circ \coloneqq |\mathcal{A}_{\mathrm{sup}}\backslash\mathcal{A}_{\mathrm{sub}}|$ and $\boldsymbol{\xi}^+ \sim Q_{\boldsymbol{q}^+}, \boldsymbol{\xi}^- \sim Q_{\boldsymbol{q}^-}$, and $\boldsymbol{\xi}^\circ \sim Q_{\boldsymbol{q}^\circ}$, where $Q_{\boldsymbol{q}^+} \coloneqq \prod_{i\in\mathcal{A}_{\mathrm{sub}}} Q_{q_i}(\xi_i), Q_{\boldsymbol{q}^-} \coloneqq \prod_{i\in\mathcal{Y}\backslash\mathcal{A}_{\mathrm{sup}}} Q_{q_i}(\xi_i)$, and $Q_{\boldsymbol{q}^\circ} \coloneqq \prod_{i\in\mathcal{A}_{\mathrm{sup}}\backslash\mathcal{A}_{\mathrm{sub}}} Q_{q_i}(\xi_i)$. We denote the probability measure that induces $Q_{\boldsymbol{q}^+}$ by $\mathbb{Q}_{\boldsymbol{q}^+}$ and the same for $\mathbb{Q}_{\boldsymbol{q}^-}$ and $\mathbb{Q}_{\boldsymbol{q}^\circ}$.

As shown in (10), $\mathbb{E}_{\boldsymbol{\xi}\sim Q_{\boldsymbol{q}}}[w(\mathcal{A}_{\boldsymbol{\xi}})\mathbb{1}(\mathcal{A}_{\mathrm{sub}} \subseteq \mathcal{A}_{\boldsymbol{\xi}} \subseteq \mathcal{A}_{\mathrm{sup}})]$ is equal to $\mathbb{P}(\mathcal{A}_{\mathrm{sub}} \subseteq \mathcal{A} \subseteq \mathcal{A}_{\mathrm{sup}})$. The tower property of expectation states that $\mathbb{E}_X[f(X)] = \mathbb{E}_Y[\mathbb{E}_X[f(X)|Y]]$ generally holds for an arbitrary pair of random variables

(X, Y) and an arbitrary function f. By choosing $X \leftarrow \boldsymbol{\xi}$ and $Y \leftarrow \boldsymbol{\xi}^+$ (and $Y \leftarrow \boldsymbol{\xi}^-$), we have

$$
\begin{aligned}
\mathbb{P}(\mathcal{A}_{\mathrm{sub}} \subseteq \mathcal{A} \subseteq \mathcal{A}_{\mathrm{sup}}) &= \mathbb{E}_{\boldsymbol{\xi} \sim Q_{\boldsymbol{q}}}[w(\mathcal{A}_{\boldsymbol{\xi}}) \mathbb{1}(\mathcal{A}_{\mathrm{sub}} \subseteq \mathcal{A}_{\boldsymbol{\xi}} \subseteq \mathcal{A}_{\mathrm{sup}})] \\
&= \mathbb{E}_{\boldsymbol{\xi}^+}[\mathbb{E}_{\boldsymbol{\xi}^-, \boldsymbol{\xi}^\circ}[w(\mathcal{A}_{\boldsymbol{\xi}}) \mathbb{1}(\mathcal{A}_{\mathrm{sub}} \subseteq \mathcal{A}_{\boldsymbol{\xi}} \subseteq \mathcal{A}_{\mathrm{sup}}) | \boldsymbol{\xi}^+]] \\
&= \sum_{z_1 \in \{0,1\}} \cdots \sum_{z_{N^+} \in \{0,1\}} \mathbb{Q}_{\boldsymbol{q}^+}(\xi_1^+ = z_1, \ldots, \xi_{N^+}^+ = z_{N^+}) \\
&\qquad \times \mathbb{E}_{\boldsymbol{\xi}^-, \boldsymbol{\xi}^\circ}[w(\mathcal{A}_{\boldsymbol{\xi}}) \underbrace{\mathbb{1}(\mathcal{A}_{\mathrm{sub}} \subseteq \mathcal{A}_{\boldsymbol{\xi}} \subseteq \mathcal{A}_{\mathrm{sup}})}_{\text{takes 0 if not } \xi_1^+ = \cdots = \xi_{N^+}^+ = 1} | \boldsymbol{\xi}^+ = \boldsymbol{z}] \\
&= \mathbb{Q}_{\boldsymbol{q}^+}(\xi_1^+ = 1, \ldots, \xi_{N^+}^+ = 1) \mathbb{E}_{\boldsymbol{\xi}^-, \boldsymbol{\xi}^\circ}[w(\mathcal{A}_{\boldsymbol{\xi}}) \mathbb{1}(\mathcal{A}_{\mathrm{sub}} \subseteq \mathcal{A}_{\boldsymbol{\xi}} \subseteq \mathcal{A}_{\mathrm{sup}}) | \boldsymbol{\xi}^+ = \mathbf{1}] \\
&= Q_{\boldsymbol{q}^+}(\mathbf{1}) \mathbb{E}_{\boldsymbol{\xi}^-}[\mathbb{E}_{\boldsymbol{\xi}^\circ}[w(\mathcal{A}_{\boldsymbol{\xi}}) \mathbb{1}(\mathcal{A}_{\mathrm{sub}} \subseteq \mathcal{A}_{\boldsymbol{\xi}} \subseteq \mathcal{A}_{\mathrm{sup}}) | \boldsymbol{\xi}^+ = \mathbf{1}, \boldsymbol{\xi}^-]] \\
&= Q_{\boldsymbol{q}^+}(\mathbf{1}) \sum_{z_1 \in \{0,1\}} \cdots \sum_{z_{N^-} \in \{0,1\}} \mathbb{Q}_{\boldsymbol{q}^-}(\xi_1^- = z_1, \ldots, \xi_{N^-}^- = z_{N^-}) \\
&\qquad \times \mathbb{E}_{\boldsymbol{\xi}^\circ}[w(\mathcal{A}_{\boldsymbol{\xi}}) \underbrace{\mathbb{1}(\mathcal{A}_{\mathrm{sub}} \subseteq \mathcal{A}_{\boldsymbol{\xi}} \subseteq \mathcal{A}_{\mathrm{sup}})}_{\text{takes 0 if not } \xi_1^- = \cdots = \xi_{N^+}^- = 0} | \boldsymbol{\xi}^+ = \mathbf{1}, \boldsymbol{\xi}^- = \boldsymbol{z}] \\
&= Q_{\boldsymbol{q}^+}(\mathbf{1}) Q_{\boldsymbol{q}^-}(\mathbf{0}) \mathbb{E}_{\boldsymbol{\xi}^\circ}[w(\mathcal{A}_{\boldsymbol{\xi}}) \mathbb{1}(\mathcal{A}_{\mathrm{sub}} \subseteq \mathcal{A}_{\boldsymbol{\xi}} \subseteq \mathcal{A}_{\mathrm{sup}}) | \boldsymbol{\xi}^+ = \mathbf{1}, \boldsymbol{\xi}^- = \mathbf{0}] \\
&= Q_{\boldsymbol{q}^+}(\mathbf{1}) Q_{\boldsymbol{q}^-}(\mathbf{0}) \mathbb{E}_{\boldsymbol{\xi}^\circ}[w(\mathcal{A}_{\boldsymbol{\xi}}) | \boldsymbol{\xi}^+ = \mathbf{1}, \boldsymbol{\xi}^- = \mathbf{0}] \\
&= Q_{\boldsymbol{q}^+}(\mathbf{1}) Q_{\boldsymbol{q}^-}(\mathbf{0}) \mathbb{E}_{\boldsymbol{\xi}^\circ}\left[\frac{P(\mathcal{A}_{\boldsymbol{\xi}})}{Q_{\boldsymbol{q}}(\boldsymbol{\xi})} \middle| \boldsymbol{\xi}^+ = \mathbf{1}, \boldsymbol{\xi}^- = \mathbf{0} \right] \\
&= \mathbb{E}_{\boldsymbol{\xi}^\circ}\left[\frac{P(\mathcal{A}_{\boldsymbol{\xi}})}{Q_{\boldsymbol{q}^\circ}(\boldsymbol{\xi}^\circ)} \middle| \boldsymbol{\xi}^+ = \mathbf{1}, \boldsymbol{\xi}^- = \mathbf{0} \right],
\end{aligned}
$$

which is the first half of Proposition 4.1.

On the other hand, the tower property of variance states that $\mathrm{Var}_X[f(X)] = \mathbb{E}_Y[\mathrm{Var}_X[f(X)|Y]] + \mathrm{Var}_Y[\mathbb{E}_X[f(X)|Y]] \leq \mathrm{Var}_Y[\mathbb{E}_X[f(X)|Y]]$. This ensures the latter half of Proposition 4.1. □

A.4 Proof of Proposition 4.2

Proof. First, we define independent Bernoulli trials $Q_{\boldsymbol{q}}$ as in (2) with $q_1 = \cdots = q_N =: q$. By importance sampling, we have

$$
\mathbb{P}(|\mathcal{A}| = k) = \sum_{\mathcal{A}:|\mathcal{A}|=k} P(\mathcal{A}) = \mathbb{E}_{\mathcal{A} \sim P}[\mathbb{1}(|\mathcal{A}| = k)] = \mathbb{E}_{\boldsymbol{\xi} \sim Q_{\boldsymbol{q}}}[w(\mathcal{A}_{\boldsymbol{\xi}}) \mathbb{1}(|\mathcal{A}_{\boldsymbol{\xi}}| = k)] \tag{A.4}
$$

in the similar way to (10), where $w(\mathcal{A}_{\boldsymbol{\xi}}) = P(\mathcal{A}_{\boldsymbol{\xi}})/Q_{\boldsymbol{q}}(\boldsymbol{\xi})$ is the weight function. Then, we introduce a new random variable, $\zeta \coloneqq \sum_{i=1}^N \xi_i$, that follows the binomial distribution: $\xi \sim \mathrm{Bin}(N, q)$. Now, we consider the

Rao–Blackwellization of (A.4) by the auxiliary random variable ζ:

$$\begin{aligned}
\mathbb{P}(|\mathcal{A}| = k) &= \mathbb{E}_{\boldsymbol{\xi}}[w(\mathcal{A}_{\boldsymbol{\xi}})\mathbb{1}(|\mathcal{A}_{\boldsymbol{\xi}}| = k)] \\
&= \mathbb{E}_{\zeta}\left[\mathbb{E}_{\boldsymbol{\xi}}\left[w(\mathcal{A}_{\boldsymbol{\xi}})\mathbb{1}(|\mathcal{A}_{\boldsymbol{\xi}}| = k)\,\middle|\,\sum_{i=1}^{N}\xi_i = \zeta\right]\right] \\
&= \sum_{n=0}^{N}\mathbb{P}(\zeta = n)\mathbb{E}_{\boldsymbol{\xi}}\left[w(\mathcal{A}_{\boldsymbol{\xi}})\underbrace{\mathbb{1}(|\mathcal{A}_{\boldsymbol{\xi}}| = k)}_{\text{takes 0 if } \zeta \neq k}\,\middle|\,\sum_{i=1}^{N}\xi_i = n\right] \\
&= \mathbb{P}(\zeta = k)\mathbb{E}_{\boldsymbol{\xi}}\left[w(\mathcal{A}_{\boldsymbol{\xi}})\,\middle|\,\sum_{i=1}^{N}\xi_i = k\right] \\
&= \binom{N}{k}q^k(1-q)^{N-k}\mathbb{E}_{\boldsymbol{\xi}}\left[\frac{P(\mathcal{A}_{\boldsymbol{\xi}})}{Q_{\boldsymbol{q}}(\mathcal{A}_{\boldsymbol{\xi}})}\,\middle|\,\sum_{i=1}^{N}\xi_i = k\right] \\
&= \binom{N}{k}q^k(1-q)^{N-k}\mathbb{E}_{\boldsymbol{\xi}}\left[\frac{P(\mathcal{A}_{\boldsymbol{\xi}})}{q^k(1-q)^{N-k}}\,\middle|\,\sum_{i=1}^{N}\xi_i = k\right] \\
&= \binom{N}{k}\mathbb{E}_{\boldsymbol{\xi}}\left[P(\mathcal{A}_{\boldsymbol{\xi}})\,\middle|\,\sum_{i=1}^{N}\xi_i = k\right]. \qquad \text{(A.5)}
\end{aligned}$$

Because $\xi_1, \ldots, \xi_N$ are i.i.d. such that $\xi_i \sim \text{Bernoulli}(q)$, the conditional expectation in the r.h.s. of (A.5) equals to the expectation over the uniform distribution on $\{\mathcal{A} \subseteq \mathcal{Y} : |\mathcal{A}| = k\}$. □

B MEAN-FIELD APPROXIMATION

We derive the update rule of the mean-field approximation (7) for completeness. It is recommended to also refer to the thesis by Djolonga et al. (2018, Section 3) since the derivation is equivalent. Let $\boldsymbol{\xi} \in \{0,1\}^N$ denote a binary random vector, $f : 2^{\mathcal{Y}} \to \mathbb{R}$ be a set function, and $P : \boldsymbol{\xi} \mapsto Z^{-1}\exp(f(\mathcal{A}_{\boldsymbol{\xi}}))$ be a probabilistic function on $\{0,1\}^N$, or equivalently $2^{\mathcal{Y}}$. Now, we consider minimizing $\mathrm{KL}(Q_{\boldsymbol{q}}\|P)$, where $Q_{\boldsymbol{q}}$ is defined in (2). Given

$$\mathrm{KL}(Q_{\boldsymbol{q}}\|P) = \mathbb{E}_{\boldsymbol{\xi}\sim Q_{\boldsymbol{q}}}\left[\log\frac{Q_{\boldsymbol{q}}(\boldsymbol{\xi})}{P(\boldsymbol{\xi})}\right] = \mathbb{E}_{\boldsymbol{\xi}\sim Q_{\boldsymbol{q}}}[\log Q_{\boldsymbol{q}}(\boldsymbol{\xi})] - \mathbb{E}_{\boldsymbol{\xi}\sim Q_{\boldsymbol{q}}}[f(\mathcal{A}_{\boldsymbol{\xi}})] + \log Z,$$

the minimization of $\mathrm{KL}(Q_{\boldsymbol{q}}\|P)$ is equivalent to maximizing the ELBO, defined as

$$L(\boldsymbol{q}) \coloneqq -\mathbb{E}_{\boldsymbol{\xi}\sim Q_{\boldsymbol{q}}}[\log Q_{\boldsymbol{q}}(\boldsymbol{\xi})] + \mathbb{E}_{\boldsymbol{\xi}\sim Q_{\boldsymbol{q}}}[f(\mathcal{A}_{\boldsymbol{\xi}})]\ (= \mathbb{H}[Q_{\boldsymbol{q}}] + \tilde{f}(\boldsymbol{q})).$$

We solve the problem $\max_{\boldsymbol{q}} L(\boldsymbol{q})$ by using the coordinate ascent. The derivative $\frac{\partial L(\boldsymbol{q})}{\partial q_i}$ is

$$\begin{aligned}
\frac{\partial L(\boldsymbol{q})}{\partial q_i} &= \frac{\partial L(\boldsymbol{q})}{\partial q_i}\left\{\sum_{j=1}^{N}(-q_j\log q_j - (1-q_j)\log(1-q_j)) + \sum_{\mathcal{A}\subseteq\mathcal{Y}} f(\mathcal{A})\prod_{j\in\mathcal{A}} q_j \prod_{j\notin\mathcal{A}}(1-q_j)\right\} \\
&= \log\frac{1-q_i}{q_i} + \sum_{\mathcal{A}: i\in\mathcal{A}} f(\mathcal{A})\prod_{\substack{j\in\mathcal{A}\\ j\neq i}} q_j \prod_{j\notin\mathcal{A}}(1-q_j) - \sum_{\mathcal{A}: i\notin\mathcal{A}} f(\mathcal{A})\prod_{j\in\mathcal{A}} q_j \prod_{\substack{j\notin\mathcal{A}\\ j\neq i}}(1-q_j) \\
&= \log\frac{1-q_i}{q_i} + \sum_{\mathcal{A}\subseteq\mathcal{Y}\setminus\{i\}}[f(\mathcal{A}\cup\{i\}) - f(\mathcal{A})]\prod_{j\in\mathcal{A}} q_j \prod_{j\notin\mathcal{A}}(1-q_j) \\
&= \log\frac{1-q_i}{q_i} + \mathbb{E}_{\boldsymbol{\xi}_{\setminus i}\sim Q_{\boldsymbol{q}_{\setminus i}}}[f(i|\mathcal{A}_{\boldsymbol{\xi}_{\setminus i}})].
\end{aligned}$$

By solving the equation

$$\log\frac{1-q_i}{q_i} + \mathbb{E}_{\boldsymbol{\xi}_{\setminus i}\sim Q_{\boldsymbol{q}_{\setminus i}}}[f(i|\mathcal{A}_{\boldsymbol{\xi}_{\setminus i}})] = 0,$$

we obtain the update rule (7). We know $\mathrm{KL}(Q_{\boldsymbol{q}}\|P) \geq 0$, leading to the inequality $\log Z \geq L(\boldsymbol{q})$. Therefore, we can evaluate the tightened lower bound of $\log Z$ using the optimized $\boldsymbol{q}$.

C GRADIENT OF RATIO MATCHING

In this section, we derive the gradient of the loss function from ratio matching (14) in analytical form. By defining $u_{m,n} \coloneqq \exp(\mathrm{tr}\phi(\boldsymbol{L}[\mathcal{A}_m]) - \mathrm{tr}\phi(\boldsymbol{L}[\mathcal{A}_m^{\bar{n}}]))$, we obtain

$$\frac{\partial J(\boldsymbol{L})}{\partial \boldsymbol{L}} = \sum_{m,n} \frac{dg(u_{m,n})^2}{du_{m,n}} \frac{\partial u_{m,n}}{\partial \boldsymbol{L}} = -2 \sum_{m,n} \frac{g(u_{m,n})}{(1+u_{m,n})^2} \frac{\partial u_{m,n}}{\partial \boldsymbol{L}}. \tag{C.6}$$

Here, $\boldsymbol{U}_{\mathcal{A}}$ denotes the $N \times |\mathcal{A}|$ binary matrix such that $\boldsymbol{L}[\mathcal{A}] = \boldsymbol{U}_{\mathcal{A}}^\top \boldsymbol{L} \boldsymbol{U}_{\mathcal{A}}$. Then,

$$\frac{\partial}{\partial \boldsymbol{L}} \mathrm{tr}\phi(\boldsymbol{L}[\mathcal{A}]) = \boldsymbol{U}_{\mathcal{A}}^\top \phi'(\boldsymbol{L}[\mathcal{A}]) \boldsymbol{U}_{\mathcal{A}},$$

where ϕ' is the derivative of the univariate scalar function ϕ. Therefore, the remaining term in (C.6) becomes

$$\frac{\partial u_{m,n}}{\partial \boldsymbol{L}} = u_{m,n} (\boldsymbol{U}_{\mathcal{A}_m}^\top \phi'(\boldsymbol{L}[\mathcal{A}_m]) \boldsymbol{U}_{\mathcal{A}_m} - \boldsymbol{U}_{\mathcal{A}_m^{\bar{n}}}^\top \phi'(\boldsymbol{L}[\mathcal{A}_m^{\bar{n}}]) \boldsymbol{U}_{\mathcal{A}_m^{\bar{n}}}).$$

Consequently, the derivative we seek is

$$\frac{\partial J(\boldsymbol{L})}{\partial \boldsymbol{L}} = -2 \sum_{m,n} \frac{u_{m,n} g(u_{m,n})}{(1+u_{m,n})^2} (\boldsymbol{U}_{\mathcal{A}_m}^\top \phi'(\boldsymbol{L}[\mathcal{A}_m]) \boldsymbol{U}_{\mathcal{A}_m} - \boldsymbol{U}_{\mathcal{A}_m^{\bar{n}}}^\top \phi'(\boldsymbol{L}[\mathcal{A}_m^{\bar{n}}]) \boldsymbol{U}_{\mathcal{A}_m^{\bar{n}}}). \tag{C.7}$$

For evaluating the gradient (C.7), computing $u_{m,n}$ requires $\mathcal{O}(|\mathcal{A}_m|^3)$ time complexity, and $\boldsymbol{U}_{\mathcal{A}_m}^\top \phi'(\boldsymbol{L}[\mathcal{A}_m]) \boldsymbol{U}_{\mathcal{A}_m}$ also takes $\mathcal{O}(|\mathcal{A}_m|^3)$ because $\boldsymbol{U}_{\mathcal{A}_m}$ has only $|\mathcal{A}_m|$ non-zero elements. Computing $\boldsymbol{U}_{\mathcal{A}_m^{\bar{n}}}^\top \phi'(\boldsymbol{L}[\mathcal{A}_m^{\bar{n}}]) \boldsymbol{U}_{\mathcal{A}_m^{\bar{n}}}$ takes $\mathcal{O}((|\mathcal{A}_m|+1)^3) = \mathcal{O}(|\mathcal{A}_m|^3)$. By taking the complexity from the sum of $N \times N$ matrices into account, we obtain the whole complexity $\mathcal{O}(\sum_{(m,n)\in\Omega} |\mathcal{A}_m|^3 + |\Omega| N^2) = \mathcal{O}(|\Omega|(\kappa^3 + N^2))$ with the minibatch Ω.

In practical scenarios, $\boldsymbol{V} \in \mathbb{R}^{N \times D}$ such that $\boldsymbol{L} = \boldsymbol{V}\boldsymbol{V}^\top$ is often learned to keep $\boldsymbol{L}$ positive (semi-)definite in learning steps. If $D < N$, the low-rank kernel matrix is obtained. Then, the gradient with respect to $\boldsymbol{V}$ becomes

$$\begin{aligned} \frac{\partial J(\boldsymbol{L})}{\partial \boldsymbol{V}} &= 2 \frac{\partial J(\boldsymbol{L})}{\partial \boldsymbol{L}} \boldsymbol{V} \\ &= -4 \sum_{m,n} \frac{u_{m,n} g(u_{m,n})}{(1+u_{m,n})^2} (\boldsymbol{U}_{\mathcal{A}_m}^\top \phi'(\boldsymbol{L}[\mathcal{A}_m]) \boldsymbol{U}_{\mathcal{A}_m} \boldsymbol{V} - \boldsymbol{U}_{\mathcal{A}_m^{\bar{n}}}^\top \phi'(\boldsymbol{L}[\mathcal{A}_m^{\bar{n}}]) \boldsymbol{U}_{\mathcal{A}_m^{\bar{n}}} \boldsymbol{V}). \end{aligned} \tag{C.8}$$

Since $\boldsymbol{U}_{\mathcal{A}_m} \boldsymbol{V}$ is the $|\mathcal{A}_m| \times D$ dense matrix, the time complexity of (C.8) is $\mathcal{O}(\sum_{(m,n)\in\Omega} (|\mathcal{A}_m|^3 + D|\mathcal{A}_m|^2) + |\Omega| N D) = \mathcal{O}(|\Omega|(\kappa^2 \max\{\kappa, D\} + ND))$. The term $\mathcal{O}(|\Omega| N D)$ arises from matrix additions, which still ensures scalability as M and/or N increases even if $D = N$.

D FURTHER EXPERIMENTS

D.1 Subset Acquiring Experiment

Here, we show additional results of the subset acquiring experiment in Section 6. As stated in Section 6, the kernel matrix $\boldsymbol{L}$ is constructed from the Gaussian kernel with the bandwidth parameter determined by the median heuristic. We can make the centered kernel matrix $\tilde{\boldsymbol{L}} \coloneqq (\boldsymbol{I} - \boldsymbol{1}\boldsymbol{1}^\top/N)\boldsymbol{L}(\boldsymbol{I} - \boldsymbol{1}\boldsymbol{1}^\top/N)$ and apply kernel principal component analysis (PCA) to $\tilde{\boldsymbol{L}}$ (Schölkopf et al., 1997). Figure 8 shows the first and second principal components of the MNIST obtained by the kernel. We can see that the kernel has a certain capability for class separation.

In Figures 9, 10, and 11, we show 10 randomly chosen acquired subsets for each $(\beta, \lambda) \in \{1, 10, 50\} \times \{0, 1, 2\}$. It is visually evident that the attractive power increases as λ and β become larger.

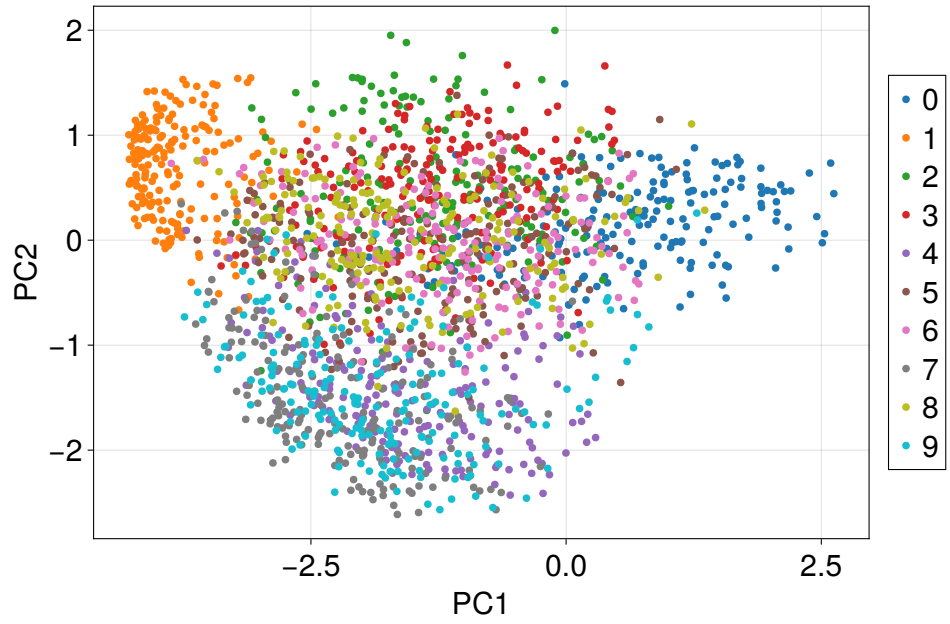
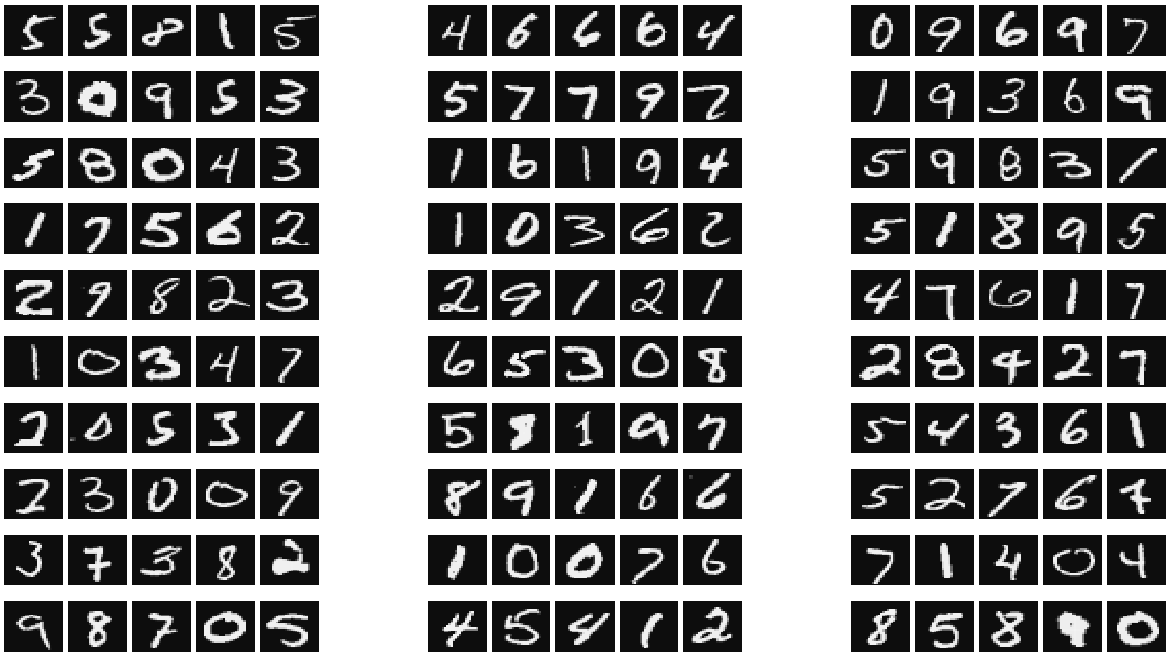
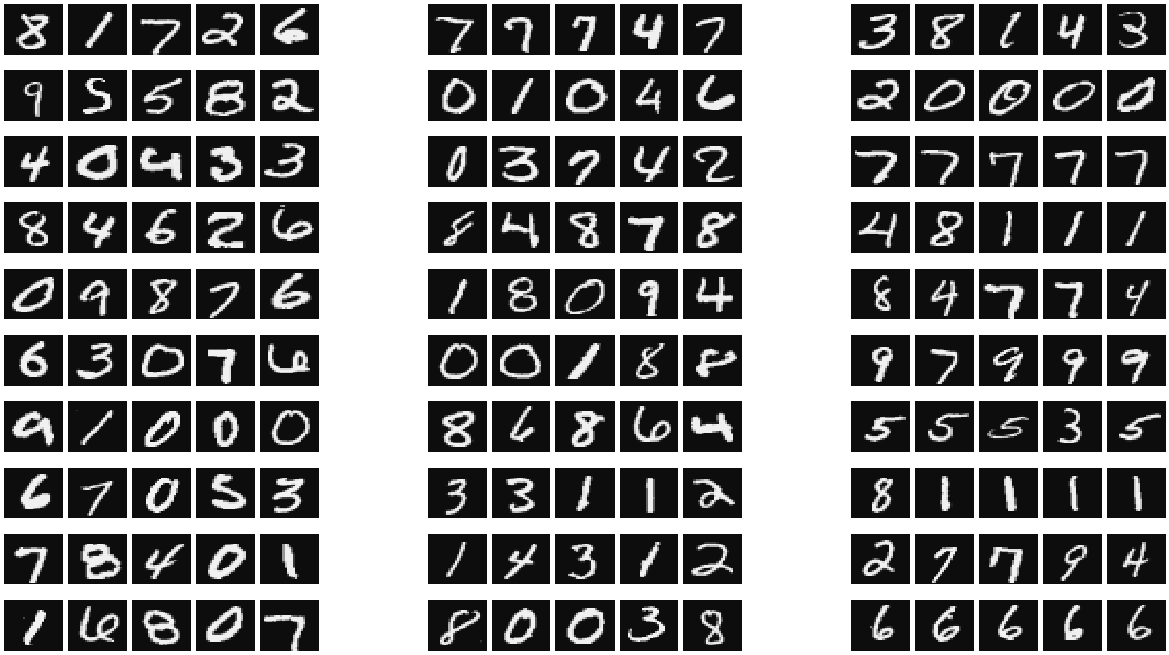


Figure 8: Kernel PCA for MNIST.



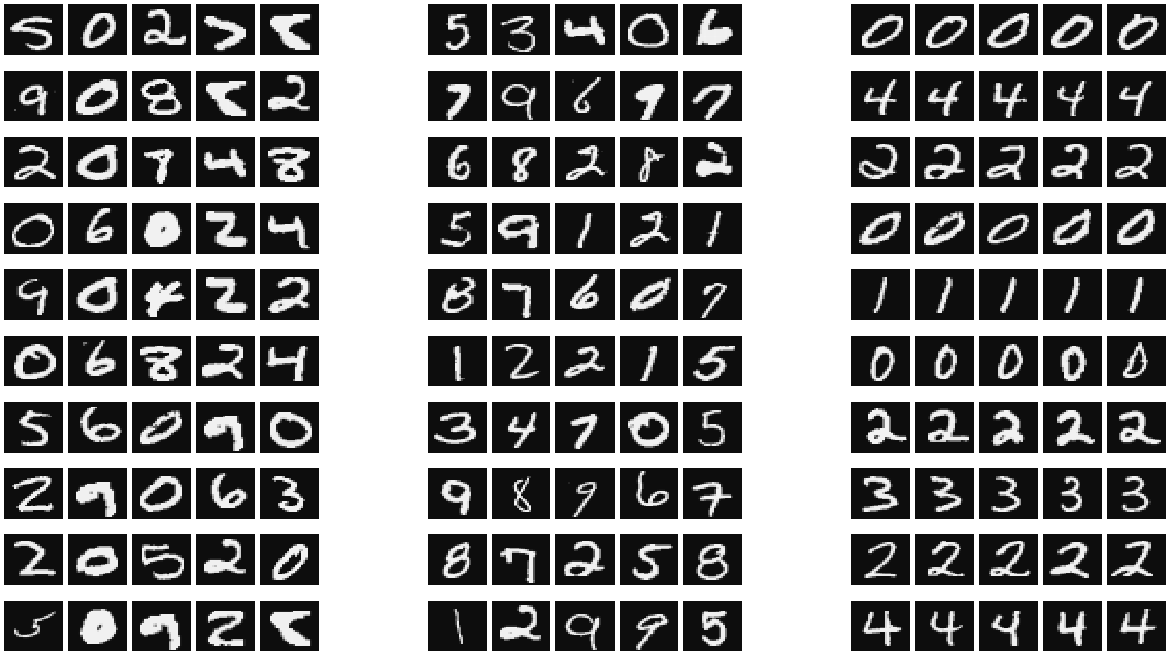
(a) $\lambda = 0, \beta = 1$ (b) $\lambda = 1, \beta = 1$ (c) $\lambda = 2, \beta = 1$

Figure 9: Examples of the acquired subsets of MNIST for $\beta = 1$.



(a) $\lambda = 0, \beta = 10$ (b) $\lambda = 1, \beta = 10$ (c) $\lambda = 2, \beta = 10$

Figure 10: Examples of the acquired subsets of MNIST for $\beta = 10$.



(a) $\lambda = 0, \beta = 50$ (b) $\lambda = 1, \beta = 50$ (c) $\lambda = 2, \beta = 50$

Figure 11: Examples of the acquired subsets of MNIST for $\beta = 50$.