



An optimal transport approach for selecting a representative subsample with application in efficient kernel density estimation

Jingyi Zhang, Cheng Meng, Jun Yu, Mengrui Zhang, Wenxuan Zhong & Ping Ma

To cite this article: Jingyi Zhang, Cheng Meng, Jun Yu, Mengrui Zhang, Wenxuan Zhong & Ping Ma (2022): An optimal transport approach for selecting a representative subsample with application in efficient kernel density estimation, Journal of Computational and Graphical Statistics, DOI: [10.1080/10618600.2022.2084404](https://doi.org/10.1080/10618600.2022.2084404)

To link to this article: <https://doi.org/10.1080/10618600.2022.2084404>

 View supplementary material 

 Accepted author version posted online: 06 Jun 2022.

 Submit your article to this journal 

 Article views: 31

 View related articles 

 View Crossmark data 

An optimal transport approach for selecting a representative subsample with application in efficient kernel density estimation

Jingyi Zhang

Center for Statistical Science, Tsinghua University

Cheng Meng*

Center for Applied Statistics, Institute of Statistics and Big Data, Renmin University of China

Jun Yu

School of Mathematics and Statistics, Beijing Institute of Technology

Mengrui Zhang, Wenxuan Zhong, and Ping Ma†

Department of Statistics, University of Georgia

*Joint first author

†Corresponding author pingma@uga.edu

Abstract

Subsampling methods aim to select a subsample as a surrogate for the observed sample. Such methods have been used pervasively in large-scale data analytics, active learning, and privacy-preserving analysis in recent decades. Instead of model-based methods, in this paper, we study model-free subsampling methods, which aim to identify a subsample that is not confined by model assumptions. Existing model-free subsampling methods are usually built upon clustering techniques or kernel tricks. Most of these methods suffer from either a large computational burden or a theoretical weakness. In particular, the theoretical weakness is that the empirical distribution of the selected subsample may not necessarily converge to the population distribution. Such computational and theoretical limitations hinder the broad applicability of model-free subsampling methods in practice. We propose a novel model-free subsampling method by utilizing optimal transport techniques. Moreover, we develop an efficient subsampling algorithm that is adaptive to the unknown probability density function. Theoretically,

we show the selected subsample can be used for efficient density estimation by deriving the convergence rate for the proposed subsample kernel density estimator. We also provide the optimal bandwidth for the proposed estimator. Numerical studies on synthetic and real-world datasets demonstrate the performance of the proposed method is superior.

Keywords: Subsampling; Optimal transport; Star discrepancy; Density estimation; Inverse transform sampling

1 Introduction

A subsampling problem can be described as follows: given a d -dimensional sample $\{\mathbf{x}_i\}_{i=1}^n$ generated from an unknown probability distribution, the goal is to take a subsample $\{\mathbf{x}_i^*\}_{i=1}^r$, $r \ll n$, as a surrogate for the original sample. In recent decades, the subsampling problem has drawn great attention in machine learning, statistics, and computer science. For example, subsampling methods are used pervasively in optimal design/active learning problems, where in a large sample of unlabeled data, the goal is to select an informative subsample to label (Settles 2012). Consider privacy-preserving analysis as another example. In some applications, subsampling methods have the potential to enhance data security (Nissim et al. 2007, Li et al. 2012). Specifically, a carefully selected subset of data can reveal little confidential information (Shu et al. 2015). Last but not least, subsampling methods are also widely applied in algorithm design to alleviate the computational burden in large-scale data analysis (Tsai et al. 2015, Zhou et al. 2017).

Many existing subsampling methods are model-based methods, which assume predictors and responses, if any, follow a postulated model. These methods aim to select an informative subsample that benefits model-fitting and prediction. Various models have been considered in subsampling problems, including linear regression (Drineas et al. 2006, 2011, Ma et al. 2014, 2015, Ma & Sun 2015, Wang et al. 2017, Meng et al. 2017, Zhang et al. 2018, Ma et al. 2020, Li & Meng 2020), generalized linear regression (Wang et al. 2018, Ai

et al. 2021b, Yu et al. 2020), l_p regression (Dasgupta et al. 2009), quantile regression (Ai et al. 2021), streaming time series model (Xie et al. 2019), Gaussian mixture model (Feldman et al. 2011), nonparametric regression (Meng et al. 2020a, 2021), among others (Bardenet et al. 2017, Quiroz et al. 2018, Yu & Wang 2022). While model-based subsampling methods have already yielded impressive achievements, the key to the success of these methods highly depends on the correct model specification. Nevertheless, in practice, model specification is a trial and error process, and a postulated model for the data could be misspecified. For example, in supervised learning, we start with a high dimensional model with numerous features; and by using model selection, we may end up with a low dimensional model with parsimonious features. In another instance, we may start with a linear regression model for a continuous response; and by discretizing the response, we may end up with a classification model. Model-based subsampling methods, however, may result in subsamples hampering such dynamic processes of model specification (Tsao & Ling 2012). Consequently, in scenarios when the model may be misspecified or in the stage of exploratory analysis, more preferred methods are model-free subsampling methods, which can identify a subsample that is not confined by model assumptions.

Recently, there have been emerging model-free subsampling methods, which aim to select a representative subsample that can capture the overall patterns of the observed sample. These methods can be divided into two classes: clustering-based approaches and kernel-based approaches. Clustering-based approaches, which are usually used in unsupervised learning methods, include k -medoids method (Kaufman & Rousseeuw 1987, Park & Jun 2009), k -center method (Feder & Greene 1988), and Wasserstein barycenter method (Agueh & Carlier 2011, Cuturi & Doucet 2014). The k -medoids method is closely related to the k -means algorithm, and the k -center method is used extensively in fast multipole methods (Greengard & Strain 1991, White et al. 1994, Yang et al. 2003, Lee & Gray 2009). The Wasserstein barycenter method aims to find

the barycenter of a set of empirical probability measures under the optimal transport metric, and such a barycenter itself can be regarded as a representative subsample. Despite wide applications of these subsampling methods, the empirical distributions of the selected subsamples, yielded by these clustering-based approaches, may not resemble the probability distribution of the original sample. That is, as the subsample size increases, the probability distributions of the subsample identified by these methods may not necessarily converge to the true probability distribution. To address such a limitation, researchers developed kernel-based approaches, which aim to select a subsample that can effectively approximate the population distribution. These approaches include the kernel herding method (Chen & Zhang 2014), the coresnet for kernel density estimation (Phillips 2013, Zheng et al. 2013, 2017), and the support point method (Mak & Joseph 2018). Despite the theoretical benefits, one limitation of these kernel-based approaches is that they may result in a large computational burden in large-scale data analysis.

To overcome the computational and theoretical limitations of the aforementioned methods, we propose a novel model-free subsampling method that is computationally efficient and enjoys nice theoretical properties. The proposed method combines the techniques of optimal transport and space-filling designs. In particular, we first transform the observed sample to be uniformly distributed on a hypercube using optimal transport techniques (Villani 2008, Peyré et al. 2019), then select a set of data points that can effectively represent the uniform distribution using space-filling designs (Owen 2003, Fang et al. 2005). The desired subsample is the one corresponding to the selected data points. The idea is analogous to an inverse procedure of the inverse transform sampling technique, which transforms a uniformly distributed sample to a sample that follows an arbitrary probability density function. Theoretically, we show the proposed subsample kernel density estimator converges to the true probability density function under mild conditions. Moreover, we show the proposed estimator converges faster than the estimator based on a randomly selected

subsample, suggesting the proposed method can be utilized for efficient density estimation. We also provide the optimal bandwidth for the proposed estimator. Numerically, utilizing projection-based optimal transport methods (Pitie et al. 2005, Rabin et al. 2011), the computational cost for the proposed method is at the order of $O(n \log(n) d^2)$ for a d -dimensional sample of size n . The proposed method thus is scalable to datasets with large n and moderate d . Numerical studies on synthetic and real-world datasets demonstrate the superior performance of the proposed method in comparison with mainstream competitors. The proposed method is implemented in an R package, named SPARTAN.

2 Preliminaries

2.1 Star discrepancy and space-filling designs

The proposed method is developed upon the notion of star discrepancy, which is a classical metric that measures the discrepancy between a set of discrete data points and the uniform distribution on the unit hypercube $[0, 1]^d$, denoted by $U[0, 1]^d$ (Niederreiter 1992, Fang & Wang 1993, Fang et al. 2005). Let $1\{\cdot\}$ be the indicator function and $\mathbf{a} = (a_1, \dots, a_d) \in [0, 1]^d$ be a vector. Let $[\mathbf{0}, \mathbf{a}] = \prod_{j=1}^d [0, a_j]$ be a hyper-rectangle and $\mathcal{U}_r = \{\mathbf{u}_i\}_{i=1}^r$ be a set of r data points in $[0, 1]^d$. We introduce the definition of the star discrepancy in the following.

Definition 1. *Given $\mathcal{U}_r$ and a hyper-rectangle $[\mathbf{0}, \mathbf{a}]$, $\mathbf{a} \in [0, 1]^d$, the corresponding local discrepancy is defined as, $D(\mathcal{U}_r, \mathbf{a}) = \left| \frac{1}{r} \sum_{i=1}^r 1\{\mathbf{u}_i \in [\mathbf{0}, \mathbf{a}]\} - \prod_{j=1}^d a_j \right|$. The star discrepancy is defined as*

$$D^*(\mathcal{U}_r) = \sup_{\mathbf{a} \in [0, 1]^d} D(\mathcal{U}_r, \mathbf{a}).$$

Definition 1 suggests a set of data points $\mathcal{U}_r$, which can effectively represent $U[0, 1]^d$, has a small value of $D^*(\mathcal{U}_r)$, and vice versa. There exist methods that

generate design points via directly minimizing the star discrepancy, and these methods are called uniform design methods (Fang et al. 2005). Despite wide applications, most of these methods are computationally expensive and are not scalable to a design with a large number of points. To alleviate such a computational burden, methods yielding a set of design points with a relatively small star discrepancy could be used as alternatives for uniform design methods. These alternatives include space-filling design methods (Wu & Hamada 2011, Fang et al. 2005) and low-discrepancy sequences (Owen 2003, Lemieux 2009, Dick et al. 2013, Leobacher & Pillichshammer 2014). The former aims to generate a set of design points that spread out over the domain as uniformly as possible. The latter sequentially generates the design points, which achieve an asymptotically fast decay rate respecting the star discrepancy. Consequently, these methods provide powerful tools to generate a set of representative design points in terms of $U[0,1]^d$.

We now discuss the theoretical property of space-filling designs and low-discrepancy sequences in terms of the star discrepancy (Owen 2003). For a Sobol sequence $\mathcal{S}_r = \{s_i\}_{i=1}^r$, a representative of low-discrepancy sequences, $D^*(\mathcal{S}_r)$ converges to zero at the rate of $O(\log(r)^d / r)$. In other words, the convergence rate of $D^*(\mathcal{S}_r)$ is of the order $O(r^{-(1-\delta)})$ for an arbitrary small $\delta > 0$ and fixed d , as r goes to infinity. For comparison, when a set of data points $\mathcal{X}_r = \{x_i\}_{i=1}^r$ is randomly generated from $U[0,1]^d$, the convergence rate of $D^*(\mathcal{X}_r)$ is of the order $O((\log \log(r) / r)^{1/2})$, which is much slower than $O(r^{-(1-\delta)})$ (Chung 1949). By adopting a method which is no worse than the Sobol sequence, in this paper, we always assume the star discrepancy $D^*(\mathcal{S}_r)$ converges to zero with the rate $O(r^{-(1-\delta)})$. There also exist some space-filling designs that can achieve a potentially faster convergence rate in terms of star discrepancy (Fang et al. 2005).

Utilizing space-filling design techniques, we propose a simple algorithm to select a representative subsample from a sample that is generated from $U[0,1]^d$. Let

$\{u_i\}_{i=1}^n$ be such a sample. The proposed algorithm, summarized in Algorithm 1, combines space-filling design techniques and the one-nearest-neighbor approximation.

Algorithm 1 Select a representative subsample from a sample generated from $U[0,1]^d$.

Step 1. Generate a set of space-filling design points $\{s_i\}_{i=1}^r \in [0,1]^d$

Step 2. For $i = 1$ to r

Select the nearest neighbor for s_i from $\{u_i\}_{i=1}^n$ using the Euclidean distance

Let u_i^* be the selected data point

Step 3. The final subsample is given by $\mathcal{U}_r^* = \{u_i^*\}_{i=1}^r$

Lemma 1 below, which is first stated in Meng et al. (2020a), characterizes the approximation error of the subsample selected by Algorithm 1. This lemma suggests the selected subsample can effectively approximate the design points in the sense that their corresponding star discrepancies are almost at the same order under certain conditions.

Lemma 1. *Let $\mathcal{S}_r = \{s_i\}_{i=1}^r \in [0,1]^d$ be a set of design points which satisfy $D^*(\mathcal{S}_r) = O(r^{-(1-\delta)})$ for any arbitrary small $\delta > 0$, as $r \rightarrow \infty$. Suppose d is fixed, when $r = O(n^{1/d})$, as $n \rightarrow \infty$, we have $D^*(\mathcal{U}_r^*) = O_p(r^{-(1-\delta)})$.*

Algorithm 1 can be extended to the case that the cumulative distribution function F of the samples is non-uniform when $d = 1$. The idea is analogous to the classical inverse transform sampling method (Devroye 1986, Mosegaard & Tarantola 1995). Let $\{x_i\}_{i=1}^n \in \mathbb{R}$ be the observed sample, we first calculate $\{F(x_i)\}_{i=1}^n$, from which, we then select a subsample $\{F(x_i^*)\}_{i=1}^r$ using Algorithm 1. Notice that the transformed sample is uniformly distributed on $[0,1]$; thus, the selected subsample is relatively representative of $U[0,1]$. Finally, the desired subsample is given by $\{x_i^*\}_{i=1}^r$. Although this simple strategy works well in practice, a limitation of such a strategy is that it is inapplicable when $d \geq 2$. To

overcome the limitation, we introduce the optimal transport map, which serves as a surrogate for F in multivariate cases. This idea is similar to the one in Chernozhukov et al. (2017), where the authors used the optimal transport map to extend the concepts of quantiles and ranks from one-dimensional samples to multivariate samples. Analogously, in this paper, we use the optimal transport map to extend the technique of inverse transform sampling from one-dimensional cases to high-dimensional cases.

2.2 Optimal transport maps

Optimal transport maps have been extensively used as a standard technique to transform one probability distribution to another. Recently, such maps have received a significant attention in machine learning and computer science (Ferradans et al. 2014, Rabin et al. 2014, Su et al. 2015, Courty et al. 2017, Meng et al. 2020b, Peyré et al. 2019), due to its close relationship with generative models, including generative adversarial nets (Goodfellow et al. 2014), the “decoder” network in variational autoencoders (Kingma & Welling 2013), among others.

Instead of introducing the general definition of the optimal transport map, we now present a specific map of our interest, and we refer to Villani (2008), Peyré et al. (2019), Zhang et al. (2021) for more details. Let u be the uniform probability distribution on $[0, 1]^d$. Let p_X and $\Omega \subseteq \mathbb{R}^d$ be the probability distribution and the domain of the random variable X , respectively. Let $\#$ be the push-forward operator, such that for all measurable $B \subset \Omega$, we have $\phi_{\#}(p_X)(B) = p_X(\phi^{-1}(B))$. Among all the maps $\phi : \Omega \rightarrow [0, 1]^d$ such that $\phi_{\#}(p_X) = u$ and $\phi_{\#}^{-1}(u) = p_X$, the optimal transport map ϕ^* of our interest is the one that minimizes the L_2 cost, $\int_{\Omega} \|X - \phi(X)\|^2 dp_X$, where $\|\cdot\|$ denotes the Euclidean norm. We focus on L_2 cost in this paper for simplicity and it is possible to consider other costs as long as the optimal transport map exists. For the L_2 cost, as a special case, when $\Omega = \mathbb{R}$ and $d = 1$, it is known that ϕ^* is equivalent to the cumulative distribution function F

(Villani 2008). This fact motivates us to use the ϕ^* as a surrogate for F in high-dimensional cases.

To obtain the desired optimal transport map that maps the observed sample to be uniformly distributed on $[0,1]^d$, we propose to first generate a synthetic sample from $U[0,1]^d$, then calculate the optimal transport map from the observed sample to the synthetic sample. One can utilize the auction algorithm or the refined auction algorithm to calculate such a map (Bertsekas 1992, Schuhmacher et al. 2020). Despite the effectiveness, the auction algorithm has an average computational cost of the order $O(n^2)$, and thus it may incur an enormous computational cost when n is large. To alleviate the computational burden, in practise, we propose to approximate the optimal transport map ϕ^* using projection-based methods (Pitié et al. 2007, Bonneel et al. 2015, Rabin et al. 2011, Meng et al. 2019, Zhang et al. Just accepted). These methods tackle the problem of estimating a d -dimensional optimal transport map iteratively by breaking down the problem into a series of subproblems. Each of the subproblems involves finding a one-dimensional optimal transport map between the projected samples, and such a subproblem can be easily solved through sorting algorithms.

3 Main algorithm

We develop a novel subsampling method named SPARTAN, which integrates space-filling design techniques and optimal transport methods. The proposed method works as follows. First, we transform the observed sample, denoted by $\{x_i\}_{i=1}^n$, to be uniformly distributed on $[0,1]^d$. We achieve this goal by utilizing the empirical optimal transport map. Here, the empirical optimal transport map is also called the optimal matching between two discrete distributions, such that each of them have n atoms and each atom has weight $1/n$. We use such an empirical optimal transport map as a surrogate of the optimal transport map between the underlying population density function of the observed sample and the uniform distribution. We then select a set of data points of size r from the transformed

sample using Algorithm 1. The subsample corresponding to the selected data points is the final output. We summarize the algorithm below.

Algorithm 2 Space-filling after optimal transport (SPARTAN)

- Step 1.* Generate a synthetic random sample $\{\mathbf{u}_i\}_{i=1}^n$ from $U[0,1]^d$
- Step 2.* Calculate the empirical optimal transport map, denoted by $\hat{\phi}$, that maps the observed sample $\{\mathbf{x}_i\}_{i=1}^n$ to the synthetic sample $\{\mathbf{u}_i\}_{i=1}^n$
- Step 3.* Calculate the transformed sample $\{\hat{\phi}(\mathbf{x}_i)\}_{i=1}^n$
- Step 4.* Select a set of data points $\{\hat{\phi}(\mathbf{x}_i^*)\}_{i=1}^r$ from $\{\hat{\phi}(\mathbf{x}_i)\}_{i=1}^n$ using Algorithm 1
- Step 5.* The final subsample is given by $\{\mathbf{x}_i^*\}_{i=1}^r$.

Figure 1 illustrates Algorithm 2 using a toy example. A two-dimensional synthetic sample of size 1000, marked as grey dots, is shown in Fig. 1(a). We first transform the sample to be uniformly distributed on $[0,1]^2$ using the projection pursuit Monge map method (Meng et al. 2019), shown in Fig. 1(b). We then generate 32 design points using a space-filling design method (Owen 2003, Fang et al. 2005). The design points are marked as triangles in Fig. 1(c). Next, for each design point, we search for its nearest neighbor, labeled as black dots in Fig. 1(c). Finally, the subsample corresponding to the selected data points, marked as black dots in Fig. 1(d), gives the desired subsample.

The computational cost for Algorithm 2 mainly incurs in Step 2 and Step 4. In particular, we use a projection-based method to approximate the desired optimal transport map in Step 2, requiring a computational cost of the order $O(n \log(n) d^2)$ (Pitié et al. 2007, Bonneel et al. 2015, Meng et al. 2019). Step 4 includes two sub-steps: generating the design points and searching the corresponding nearest neighbors. The design points can be generated beforehand; thus, the computation time for generating these points is not considered here. For searching the nearest neighbors, we opt to use the k -d tree method, whose computation cost is at the order of $O(n \log(n))$ (Bentley 1975, Wald &

Havran 2006). In sum, the overall computational complexity for Algorithm 1 is at the order of $O(n \log(n) d^2)$.

Figure 2 visualizes the subsamples (black dot) selected by the proposed method (lower row) compared with the subsamples selected by the random subsampling method (upper row). The two-dimensional samples (grey dots) are generated from three different distributions: the standard Gaussian distribution (left column), a mixture Gaussian distribution (middle column), and a mixture beta distribution (right column). From plots in the left column, one can observe that the randomly selected subsample is far from symmetric. From plots in the middle and the right columns, one can see that some peaks in the probability distribution are largely overlooked by the random subsampling method. We observe that the subsamples identified by the proposed method have a more robust and appealing visual representation of the corresponding probability distribution in all the cases.

4 Theoretical results

In this section, we study the theoretical properties of the subsamples obtained in Algorithm 2. In particular, we develop an asymptotic theory concerning the rates of convergence of the estimated density to the true density as the sample size goes to infinity. The rates are calculated in terms of the point-wise mean squared error (MSE) that defined as $\text{MSE}(\hat{p}(z)) = E\{\hat{p}(z) - p(z)\}^2$, where $z \in \mathbb{R}^d$, $\hat{p}$ is the density estimator and p is the true density. The density is estimated using the widely-used kernel density estimation method. Throughout this paper, we consider the Gaussian kernel. The extension of the main theorem to other kernel functions is straightforward, as long as such a kernel function satisfies some regularity conditions, which are relegated to the Supplementary Material. A more in-depth discussion on different choices of kernel functions can be found in Scott (2015). To avoid trivial cases, we consider the case that $d \geq 2$ in this section. Without lose of generality, we assume the points $\{x_i\}$ are distinct, and the points $\{u_i\}$ are distinct. In such cases, the optimal transport map in Step 2 of

Algorithm 2 is a one-to-one map from $\{\mathbf{x}_i\}_{i=1}^n$ to $\{\mathbf{u}_i\}_{i=1}^n$. Let p be the probability density function to be estimated. Two widely-used regularity conditions for p are required in kernel density estimation,

- Condition (a). $\partial^2 p(z) / \partial z_j^2$ is absolutely continuous, for $j = 1, \dots, d$;
- Condition (b). $\partial^3 p(z) / \partial z_j^3$ is square-integrable, for $j = 1, \dots, d$.

Let $\mathbf{x} \in \mathbb{R}^{n \times d}$ be the sample matrix, where the (i, j) -th element is x_{ij} , and $\mathbf{x}^* \in \mathbb{R}^{r \times d}$ be the subsample matrix, where the (i, j) -th element is x_{ij}^* . Let $h > 0$ be the bandwidth and $K : \mathbb{R} \rightarrow \mathbb{R}$ be a kernel function. For any $z \in \mathbb{R}^d$, the full-sample product kernel density estimator can be written as

$$\hat{p}(z) = \sum_{i=1}^n \left[\prod_{j=1}^d K \left\{ (z_j - x_{ij}) / h \right\} / h \right] / n. \quad (1)$$

Equation (1) can be generalized to a more general multivariate kernel density estimator. In particular, for a $d \times d$ nonsingular bandwidth matrix $\mathbf{H}$ and a multivariate kernel function $\mathcal{K} : \mathbb{R}^d \rightarrow \mathbb{R}$, a general multivariate kernel estimator can be written as

$$\hat{p}_{general}(z) = \frac{1}{n |\mathbf{H}|} \sum_{i=1}^n \left[\mathcal{K} \left\{ \mathbf{H}^{-1/2} (z - \mathbf{x}_i) \right\} \right]. \quad (2)$$

It is apparent that Equation (2) is equivalent to Equation (1) when $\mathbf{H} = h \mathbf{I}_d$, where $\mathbf{I}_d$ is the identity matrix. Let $\mathcal{K}$ be the Gaussian kernel in Equation (2), it is equivalent to choose $\mathcal{K} = \mathcal{N}(\mathbf{0}, \Sigma)$ with $\mathbf{H} = \mathbf{I}_d$, or to choose $\mathcal{K} = \mathcal{N}(\mathbf{0}, \mathbf{I}_d)$ with $\mathbf{H} = \Sigma^{1/2}$ in Equation (1). Consequently, with a properly chosen kernel function, one can reformulate a general multivariate kernel estimator to a product kernel density estimator. We thus only focus on the product kernel density estimator in this section without loss of generality.

Analogous to Equation (1), the density estimator $\hat{p}^*(z)$ that computed from the subsample can be written as

$$\hat{p}^*(z) = \sum_{i=1}^r \left[\prod_{j=1}^d K \left\{ (z_j - x_{ij}^*) / h \right\} / h \right] / r.$$

We derive the convergence rate for the mean squared error for the proposed subsample estimator. The results are summarized in Theorem 1 below, and the proof is relegated to Appendix.

Theorem 1. Suppose p satisfies Conditions (a) and (b). Moreover, suppose p has a compact convex domain $\Omega \subset \mathbb{R}^d$, and there exists a constant $c \geq 1$ for which $c^{-1} \leq p(x) \leq c$ for any $x \in \Omega$. When $d \geq 2$, $r = O(n^{1/d})$, as $n \rightarrow \infty$ and $h \rightarrow 0$, for any arbitrary small $\delta > 0$, we have

$$\text{MSE}(\hat{p}^*(z)) = O\left(\frac{1}{r^{2(1-\delta)} h^{d+2}}\right) + O(h^4).$$

In particular, if $h = O(r^{-2(1-\delta)/(d+6)})$, we have

$$\text{MSE}(\hat{p}^*(z)) = O(r^{-8(1-\delta)/(d+6)}). \quad (3)$$

Theorem 1 shows the proposed subsample estimator converges to the true probability density function. Moreover, Theorem 1 indicates the proposed subsampling method can be used for efficient density estimation. Specifically, let $\mathbf{X}^+ \in \mathbb{R}^{r \times d}$ be a randomly selected subsample matrix, and $\hat{p}^+(z)$ be the corresponding subsample estimator. According to Theorem 6.4 of Scott (2015), as $r = o(n)$ and $n \rightarrow \infty$, when $h = O(r^{-1/(4+d)})$, $\text{MSE}(\hat{p}^+(z))$ achieves the optimal convergence rate $O(r^{-4/(d+4)})$ for any $z \in \Omega$. Such a convergence rate is much slower than the convergence rate in Equation (3). Consequently, Theorem 1 indicates one can approximate the probability density function p more efficiently using the proposed subsample kernel density estimator, compared with the counterpart based on a randomly selected subsample.

Consider the bandwidth h , or generally, the bandwidth matrix $\mathbf{H} \in \mathbb{R}^{d \times d}$. In practice, one can determine the value of $\mathbf{H}$ through the plug-in approach or the

cross-validation approach (Duong & Hazelton 2003, Chacón & Duong 2010, Scott 2015). One limitation of these approaches, however, is that they may result in a computational burden for the sample with moderate or large n . To combat the computational burden, we opt to determine the value of $\mathbf{H}$ using the general Scott's rule (Scott 2015), which suggests to use $\mathbf{H} = r^{-1/(d+4)} \times \Sigma^{1/2}$ for a subsample kernel density estimator that based on a subsample of size r . Here, Σ is the empirical variance-covariance matrix for the observed sample. Analogously, as suggested by Theorem 1, we also consider using $\mathbf{H} = r^{-2/(d+6)} \times \Sigma^{1/2}$ for the proposed estimator. Consider the essential condition in Theorem 1, which requires the domain of p to be compact convex. Empirically, we find the proposed estimator still works reasonably well when such a condition does not hold, as shown in the following section.

5 Simulation Results

To evaluate the proposed subsampling method, we compare it with three mainstream competitors in terms of the estimation accuracy of the kernel density estimator. The competitors include the uniform subsampling method, also called the random subsampling method, the k -medoids method, and the support point method (Mak & Joseph 2018). We use the projection-pursuit Monge map method (Meng et al. 2019) for approximating the optimal transport map in Algorithm 2. All the methods are implemented in $\mathbb{R}$, and all the parameters are set as default.

For each subsampling method, we first calculate the subsample kernel density estimator $\hat{p}(x)$, then evaluate the accuracy of which using the Hellinger distance (Li et al. 2016), defined as $1 - \sum_{i=1}^n \sqrt{\hat{p}(x_i) / p(x_i)} / n$, where $\{x_i\}_{i=1}^n$ is an independent testing dataset generate from the same probability density function as the training sample. Empirically, we find other metrics, like the mean squared error considered in Theorem 1, also yield similar performance. For the kernel density estimator, we use the Gaussian kernel and the general Scott's rule (Scott 2015) to determine the bandwidth matrix. In particular, for all the subsample estimator,

the bandwidth matrix $\mathbf{H} = r^{-1/(d+4)} \times \Sigma^{1/2}$, where Σ is the empirical variance-covariance matrix. For the proposed method, we also consider the cases that $\mathbf{H} = r^{-2/(d+6)} \times \Sigma^{1/2}$, according to Theorem 1. The standard errors are calculated through a hundred replicates. In each replicate, we generate a synthetic training sample with $n = 10^4$ from $d = \{2, 5, 10, 20\}$ and each of the following three probability density functions,

- D1: A Gaussian distribution $\mathcal{N}(\mathbf{0}, \Sigma)$, where $\Sigma_{ij} = 0.5^{|i-j|}$, $i, j = 1, \dots, d$;
- D2: A mixture Gaussian distribution
- $\mathcal{N}(\mathbf{1}, \Sigma) / 4 + \mathcal{N}(-\mathbf{1}, \Sigma) / 4 + \mathcal{N}(\mathbf{0}, \Sigma) / 2$, where $\Sigma = 0.8^{|i-j|}$, $i, j = 1, \dots, d$.
- D3: A mixture t -distribution, whose degree-of-freedom equals 8, 10, and 12,
- $t(\mathbf{0}, \Sigma, 8) / 3 + t(\mathbf{0}, \Sigma, 10) / 3 + t(\mathbf{0}, \Sigma, 12) / 3$, where $\Sigma = 0.8^{|i-j|}$, $i, j = 1, \dots, d$.

Figure 3 shows the Hellinger distance versus different r under various settings. Each row represents a particular data distribution D1–D3, and each column represents a particular d . We use crosses to denote the uniform subsampling method (UNIF), hollow circles to denote the K-medoids method (KM), hollow triangles to denote the support point method (SP), solid circles to denote the proposed method (SPARTAN), and solid triangles to denote the proposed method with $\mathbf{H} = r^{-2/(d+6)} \times \Sigma^{1/2}$ (SPARTAN*).

Three significant observations can be made from Fig. 3. We first observe that the K-medoids method performs worse than the uniform subsampling method in almost all cases. Moreover, the support point method outperforms the uniform subsampling method in all cases. We also observe the Hellinger distance yielded by these two methods do not converge to zero in some cases. Such an observation can be attributed to the fact that the probability distribution of the subsample identified by these two methods may not necessarily converge to the true probability distribution.

Second, we observe the Hellinger distance yielded by the proposed method decreases as r increases. Moreover, the proposed method outperforms the uniform subsampling method in all cases. These observations are consistent with Theorem 1, which indicates the proposed subsample estimator converges to the true probability density function and is more efficient than the estimator corresponding to the uniform subsampling method.

Third, we observe the proposed estimator with $\mathbf{H} = r^{-1/(d+4)} \times \Sigma^{1/2}$ outperforms the other three competitors in most of the cases. As the same bandwidth matrices are applied in all these estimators, such a comparison is fair. Consequently, the aforementioned observation suggests the subsample identified by the proposed subsampling method is more representative of the observed sample than the subsamples selected by the other three methods. We also observe the proposed estimator with $\mathbf{H} = r^{-2/(d+6)} \times \Sigma^{1/2}$ consistently outperforms the one with $\mathbf{H} = r^{-1/(d+4)} \times \Sigma^{1/2}$. This observation is consistent with Theorem 1, which suggests $h = O(r^{-2(1-\delta)/(d+6)})$ yields the smallest upper bound of the asymptotic integrated mean squared error for the proposed estimator.

6 Real data example

6.1 Density estimation

Throughout this section, we consider the banknote authentication dataset, which is extracted from images that were taken from 1372 genuine and forged banknotes. Wavelet transform was used to extract four features from images ². To evaluate the performance of the proposed subsampling method, we compare it with other competitors in terms of the accuracy of the kernel density estimation and the prediction accuracy in active learning. A brief introduction to active learning will be given later.

We first visualize the banknote authentication dataset and the subsample selected by the proposed method. In Fig.4, the lower diagonal panels show the

scatter plots for each pair of the predictors. We select a subsample of size fifty, and the scatter plots for such a subsample are shown in the upper panels of Fig.4. The heat maps are obtained using kernel density estimation. We observe the selected subsample has an appealing visual representation of the original sample.

For density estimation, we consider three competitors, as mentioned in the previous section. Same as the settings stated in the previous section, we used the Gaussian kernel for kernel density estimators and the general Scott's rule to determine the value of the bandwidth matrix. All the parameters are set as the same as the ones we used in the previous section. We replicated the experiment twenty times. In each replication, the dataset is randomly divided into the training set and the testing set of equal sizes. We first calculate the full sample kernel density estimator using the testing set, denoted by $\hat{p}_{full}$. For each subsample kernel density estimator, we then evaluate its estimation accuracy through the empirical Hellinger distance, defined as $1 - \sum_{i=1}^n \sqrt{\hat{p}(x_i) / \hat{p}_{full}(x_i) / n}$, where $\{x_i\}_{i=1}^n$ represents the testing set. This empirical Hellinger distance is not a formal distance and thus may have negative values, as we will see later. Nevertheless, the empirical Hellinger distance can be used as a surrogate for the true Hellinger distance since a small value of the empirical Hellinger distance is associated with a small value of the true Hellinger distance, intuitively.

The left panel of Fig. 5 shows the empirical Hellinger distance versus different subsample sizes r . The standard error bars are obtained from one hundred replicates. We observe that the uniform subsampling method consistently outperforms the K-medoids method. We then observe that the proposed method and the support point method perform similarly, and both have better performance than the uniform subsampling method. Finally, we observe the proposed estimator with $\mathbf{H} = r^{-2/(d+6)} \times \Sigma^{1/2}$, as guided by Theorem 1, gives the

best result. All these observations are consistent with the findings in the previous section.

6.2 Active learning

We now consider the task of active learning, which aims to make an accurate prediction, with the number of labeled training data points as small as possible (Krogh & Vedelsby 1995, Cohn et al. 1996). These approaches are essential for numerous sophisticated supervised learning tasks, where the labeled instances are challenging, time-consuming, or expensive to obtain. Take speech recognition as an example; accurate labeling of speech utterances is extremely time-consuming and requires trained linguists. It is reported that annotation at the level of the phoneme can take 400 times longer than the actual audio (Settles 2012). In general, active learning approaches select the data points (also termed as the query points) iteratively and interactively. In each iteration, one query the oracle to obtain the label at a new query point, based on certain criteria. It is known that a representative subsample is potentially associated with an accurate prediction in active learning (Settles 2012).

The proposed subsampling method can be cast as an active learning approach. In particular, we generate the Sobol sequence (Owen 2003) in Algorithm 1 and select the query points sequentially in Algorithm 2. To evaluate the performance of the proposed method, we compare it with the following baseline methods: (1) random sampling (RANDOM), (2) query by committee (COMMITTEE), which select query points that maximize the disagreement among different models (Settles 2012), and (3) margin-based method (MARGIN) which choose query points that lie on the margin of the decision line (Schohn & Cohn 2000).

We replicate the experiment a hundred times on the banknote dataset. In each replication, the dataset is randomly divided into the training set and the testing set of equal sizes. We evaluate the classification model by its mean classification accuracy on the testing set. The classification accuracy is defined as

$(TP + FN) / n$, where n denotes the size of the testing set, and TP and FN denote true positive and false negative, respectively. We use the support vector machine, implemented by the R package `e1071` (Meyer et al. 2015)), for classification in the active learning. The RBF kernel with default parameters is applied. The size of query points ranges from 10 to 200. For the committee method and the margin-based method, which require several initial labeled data points as input, ten data points are randomly selected and labeled.

The right panel of Fig. 5 shows the mean classification accuracy of different active learning methods versus different numbers of query points. The vertical bars represent the standard errors. These bars, however, are almost invisible due to extremely small values of standard errors. We observe the proposed method consistently outperforms all the competitors. We attribute such an observation to the fact that the proposed method selects a representative subsample in a sequential way, resulting in a more accurate prediction in active learning.

7 Discussion

In this paper, we proposed a novel model-free subsampling method, utilizing the space-filling design and optimal transport techniques. The proposed algorithm is efficient and can be adaptive to the unknown probability density function. Theoretically, we show the proposed subsample kernel density estimator converges to the true probability density function under mild conditions. The order for the optimal smoothing parameter for the proposed kernel density estimator is also derived. The superior performance of the proposed method over mainstream competitors was justified by various numerical experiments.

In this paper, we mainly focus on using the unit cube as the target distribution due to mathematical simplicity. In practise, it is possible to consider standard Gaussian distribution instead. Specifically, we could generate the random sample from the standard Gaussian distribution in Algorithm 2, and use the Gaussian

Sobol sequence instead of the space-filling design points in Algorithm 1. The other steps remain the same. Empirical results show such a scheme may lead to slightly better performance. The proposed method has the potential to be applied to many large-sample applications, including but not limited to nonparametric regression, kernel methods, and low-rank approximation of matrices. This work may speed up these researches with theoretical guarantees.

Acknowledgment

The authors thank the associate editor and two anonymous reviewers for provided helpful comments on earlier drafts of the manuscript. The authors would like to acknowledge the support from National Key R&D Program of China (No. 2021YFA1001300), National Natural Science Foundation of China Grant No.12101606, No.12001042, the U.S. National Science Foundation under grants DMS-1903226, DMS-1925066, the U.S. National Institute of Health under grant R01GM122080, and Beijing Institute of Technology research fund program for young scholars.

Conflict of Interest

The authors report there are no competing interests to declare.

Notes

¹ One exception is that when all the covariates of the sample are independent with each other, in which case one can directly calculate the multivariate cumulative distribution function as the product of all the one-dimensional marginal cumulative distribution function. Nevertheless, independent covariates are rarely the case in practice.

² The dataset can be downloaded from
<https://archive.ics.uci.edu/ml/datasets/banknote+authentication>

References

Agueh, M. & Carlier, G. (2011), 'Barycenters in the Wasserstein space', *SIAM Journal on Mathematical Analysis* **43**(2), 904–924.

Ai, M., Wang, F., Yu, J. & Zhang, H. (2021b), 'Optimal subsampling for large-scale quantile regression', *Journal of Complexity* **62**, 101512.

Ai, M., Yu, J., Zhang, H. & Wang, H. (2021), 'Optimal subsampling algorithms for big data regressions', *Statistica Sinica* **31**, 1–24.

Bardenet, R., Doucet, A. & Holmes, C. (2017), 'On markov chain Monte Carlo methods for tall data', *The Journal of Machine Learning Research* **18**(1), 1515–1557.

Bentley, J. L. (1975), 'Multidimensional binary search trees used for associative searching', *Communications of the ACM* **18**(9), 509–517.

Bertsekas, D. P. (1992), 'Auction algorithms for network flow problems: A tutorial introduction', *Computational optimization and applications* **1**(1), 7–66.

Bonneel, N., Rabin, J., Peyré, G. & Pfister, H. (2015), 'Sliced and Radon Wasserstein barycenters of measures', *Journal of Mathematical Imaging and Vision* **51**(1), 22–45.

Chacón, J. E. & Duong, T. (2010), 'Multivariate plug-in bandwidth selection with unconstrained pilot bandwidth matrices', *Test* **19**(2), 375–398.

Chen, C. P. & Zhang, C.-Y. (2014), 'Data-intensive applications, challenges, techniques and technologies: A survey on big data', *Information Sciences* **275**, 314–347.

Chernozhukov, V., Galichon, A., Hallin, M. & Henry, M. (2017), 'Monge–Kantorovich depth, quantiles, ranks and signs', *The Annals of Statistics* **45**(1), 223–256.

- Chung, K.-L. (1949), 'An estimate concerning the Kolmogoroff limit distribution', *Transactions of the American Mathematical Society* **67**(1), 36–50.
- Cohn, D. A., Ghahramani, Z. & Jordan, M. I. (1996), 'Active learning with statistical models', *Journal of artificial intelligence research* **4**, 129–145.
- Courty, N., Flamary, R., Tuia, D. & Rakotomamonjy, A. (2017), 'Optimal transport for domain adaptation', *IEEE transactions on pattern analysis and machine intelligence* **39**(9), 1853–1865.
- Cuturi, M. & Doucet, A. (2014), Fast computation of wasserstein barycenters, in 'International conference on machine learning', PMLR, pp. 685–693.
- Dasgupta, A., Drineas, P., Harb, B., Kumar, R. & Mahoney, M. W. (2009), 'Sampling algorithms and coresets for l_p regression', *SIAM Journal on Computing* **38**(5), 2060–2078.
- Devroye, L. (1986), Sample-based non-uniform random variate generation, in 'Proceedings of the 18th conference on Winter simulation', ACM, pp. 260–265.
- Dick, J., Kuo, F. Y. & Sloan, I. H. (2013), 'High-dimensional integration: the quasi-Monte Carlo way', *Acta Numerica* **22**, 133–288.
- Drineas, P., Kannan, R. & Mahoney, M. W. (2006), 'Fast Monte Carlo algorithms for matrices I: Approximating matrix multiplication', *SIAM Journal on Computing* **36**(1), 132–157.
- Drineas, P., Mahoney, M. W., Muthukrishnan, S. & Sarlós, T. (2011), 'Faster least squares approximation', *Numerische mathematik* **117**(2), 219–249.
- Duong, T. & Hazelton, M. (2003), 'Plug-in bandwidth matrices for bivariate kernel density estimation', *Journal of Nonparametric Statistics* **15**(1), 17–30.

Fang, K.-T., Li, R. & Sudjianto, A. (2005), *Design and modeling for computer experiments*, CRC Press.

Fang, K.-T. & Wang, Y. (1993), *Number-theoretic methods in statistics*, CRC Press.

Feder, T. & Greene, D. (1988), Optimal algorithms for approximate clustering, *in* 'Proceedings of the twentieth annual ACM symposium on Theory of computing', ACM, pp. 434–444.

Feldman, D., Faulkner, M. & Krause, A. (2011), Scalable training of mixture models via coresets, *in* 'Advances in neural information processing systems', pp. 2142–2150.

Ferradans, S., Papadakis, N., Peyré, G. & Aujol, J.-F. (2014), 'Regularized discrete optimal transport', *SIAM Journal on Imaging Sciences* **7**(3), 1853–1882.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. & Bengio, Y. (2014), Generative adversarial nets, *in* 'Advances in neural information processing systems', pp. 2672–2680.

Greengard, L. & Strain, J. (1991), 'The fast Gauss transform', *SIAM Journal on Scientific and Statistical Computing* **12**(1), 79–94.

Kaufman, L. & Rousseeuw, P. (1987), *Clustering by means of medoids*, North-Holland.

Kingma, D. P. & Welling, M. (2013), 'Auto-encoding variational bayes', *arXiv preprint arXiv:1312.6114*.

Krogh, A. & Vedelsby, J. (1995), Neural network ensembles, cross validation, and active learning, *in* 'Advances in neural information processing systems', pp. 231–238.

Lee, D. & Gray, A. G. (2009), Fast high-dimensional kernel summations using the Monte Carlo multipole method, *in* 'Advances in Neural Information Processing Systems', pp. 929–936.

Lemieux, C. (2009), *Monte Carlo and quasi-Monte Carlo sampling*, Springer, New York.

Leobacher, G. & Pillichshammer, F. (2014), *Introduction to quasi-Monte Carlo integration and applications*, Springer.

Li, D., Yang, K. & Wong, W. H. (2016), Density estimation via discrepancy based adaptive sequential partition, *in* 'Advances in Neural Information Processing Systems', pp. 1091–1099.

Li, N., Qardaji, W. & Su, D. (2012), On sampling, anonymization, and differential privacy or, k-anonymization meets differential privacy, *in* 'Proceedings of the 7th ACM Symposium on Information, Computer and Communications Security', ACM, pp. 32–33.

Li, T. & Meng, C. (2020), 'Modern subsampling methods for large-scale least squares regression', *International Journal of Cyber-Physical Systems (IJCPS)* 2(2), 1–28.

Ma, P., Mahoney, M. W. & Yu, B. (2015), 'A statistical perspective on algorithmic leveraging', *The Journal of Machine Learning Research* 16(1), 861–911.

Ma, P., Mahoney, M. & Yu, B. (2014), A statistical perspective on algorithmic leveraging, *in* 'International Conference on Machine Learning', pp. 91–99.

Ma, P. & Sun, X. (2015), 'Leveraging for big data regression', *Wiley Interdisciplinary Reviews: Computational Statistics* 7(1), 70–76.

Ma, P., Zhang, X., Xing, X., Ma, J. & Mahoney, M. W. (2020), 'Asymptotic analysis of sampling estimators for randomized numerical linear algebra algorithms', *The 23rd International Conference on Artificial Intelligence and Statistics*. 2020.

Mak, S. & Joseph, V. R. (2018), 'Support points', *The Annals of Statistics* **46**(6A), 2562–2592.

Meng, C., Ke, Y., Zhang, J., Zhang, M., Zhong, W. & Ma, P. (2019), Large-scale optimal transport map estimation using projection pursuit, *in* 'Advances in Neural Information Processing Systems', pp. 8116–8127.

Meng, C., Wang, Y., Zhang, X., Mandal, A., Zhong, W. & Ma, P. (2017), Effective statistical methods for big data analytics, *in* 'Handbook of research on applied cybernetics and systems science', IGI Global, pp. 280–299.

Meng, C., Yu, J., Chen, Y., Zhong, W. & Ma, P. (2021), 'Smoothing splines approximation using hilbert curve basis selection', *Journal of Computational and Graphical Statistics* (just-accepted), 1–26.

Meng, C., Yu, J., Zhang, J., Ma, P. & Zhong, W. (2020b), 'Sufficient dimension reduction for classification using principal optimal transport direction', *Advances in Neural Information Processing Systems* **33**.

Meng, C., Zhang, X., Zhang, J., Zhong, W. & Ma, P. (2020a), 'More efficient approximation of smoothing splines via space-filling basis selection', *Biometrika* **107**, 723–735.

Meyer, D., Dimitriadou, E., Hornik, K., Weingessel, A. & Leisch, F. (2015), 'e1071: misc functions of the department of statistics, probability theory group (formerly: E1071), tu wien. r package version 1.6-7'.

Mosegaard, K. & Tarantola, A. (1995), 'Monte Carlo sampling of solutions to inverse problems', *Journal of Geophysical Research: Solid Earth* **100**(B7), 12431–12447.

Niederreiter, H. (1992), *Random number generation and quasi-Monte Carlo methods*, SIAM.

Nissim, K., Raskhodnikova, S. & Smith, A. (2007), Smooth sensitivity and sampling in private data analysis, *in* 'Proceedings of the thirty-ninth annual ACM symposium on Theory of computing', ACM, pp. 75–84.

Owen, A. B. (2003), 'Quasi-Monte Carlo sampling', *Monte Carlo Ray Tracing: Siggraph* **1**, 69–88.

Park, H.-S. & Jun, C.-H. (2009), 'A simple and fast algorithm for k-medoids clustering', *Expert systems with applications* **36**(2), 3336–3341.

Peyré, G., Cuturi, M. et al. (2019), 'Computational optimal transport', *Foundations and Trends® in Machine Learning* **11**(5-6), 355–607.

Phillips, J. M. (2013), ϵ -samples for kernels, *in* 'Proceedings of the twenty-fourth annual ACM-SIAM symposium on Discrete algorithms', SIAM, pp. 1622–1632.

Pitié, F., Kokaram, A. C. & Dahyot, R. (2005), N-dimensional probability density function transfer and its application to color transfer, *in* 'Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on', Vol. 2, IEEE, pp. 1434–1439.

Pitié, F., Kokaram, A. C. & Dahyot, R. (2007), 'Automated colour grading using colour distribution transfer', *Computer Vision and Image Understanding* **107**(1-2), 123–137.

Quiroz, M., Kohn, R., Villani, M. & Tran, M.-N. (2018), 'Speeding up mcmc by efficient data subsampling', *Journal of the American Statistical Association*.

Rabin, J., Ferradans, S. & Papadakis, N. (2014), Adaptive color transfer with relaxed optimal transport, *in* '2014 IEEE International Conference on Image Processing (ICIP)', IEEE, pp. 4852–4856.

Rabin, J., Peyré, G., Delon, J. & Bernot, M. (2011), Wasserstein barycenter and its application to texture mixing, *in* 'International Conference on Scale Space and Variational Methods in Computer Vision', Springer, pp. 435–446.

Schohn, G. & Cohn, D. (2000), Less is more: Active learning with support vector machines, *in* 'ICML', Citeseer, pp. 839–846.

Schuhmacher, D., Bähre, B., Gottschlich, C., Hartmann, V., Heinemann, F. & Schmitzer, B. (2020), *transport: Computation of Optimal Transport Plans and Wasserstein Distances*. R package version 0.12-2.

URL: <https://cran.r-project.org/package=transport>

Scott, D. W. (2015), *Multivariate Density Estimation: Theory, Practice, and Visualization*, John Wiley & Sons.

Settles, B. (2012), 'Active learning', *Synthesis Lectures on Artificial Intelligence and Machine Learning* 6(1), 1–114.

Shu, X., Yao, D. & Bertino, E. (2015), 'Privacy-preserving detection of sensitive data exposure', *IEEE transactions on information forensics and security* 10(5), 1092–1103.

Su, Z., Wang, Y., Shi, R., Zeng, W., Sun, J., Luo, F. & Gu, X. (2015), 'Optimal mass transport for shape matching and comparison', *IEEE transactions on pattern analysis and machine intelligence* 37(11), 2246–2259.

Tsai, C.-W., Lai, C.-F., Chao, H.-C. & Vasilakos, A. V. (2015), 'Big data analytics: a survey', *Journal of Big data* 2(1), 21.

Tsao, M. & Ling, X. (2012), 'Subsampling method for robust estimation of regression models', *Open Journal of Statistics* 2(03), 281.

Villani, C. (2008), *Optimal transport: old and new*, Springer Science & Business Media.

Wald, I. & Havran, V. (2006), On building fast kd -trees for ray tracing, and on doing that in $O(n \log n)$, in 'Interactive Ray Tracing 2006, IEEE Symposium on', IEEE, pp. 61–69.

Wang, H., Zhu, R. & Ma, P. (2018), 'Optimal subsampling for large sample logistic regression', *Journal of the American Statistical Association* 113(522), 829–844.

Wang, Y., Yu, A. W. & Singh, A. (2017), 'On computationally tractable selection of experiments in measurement-constrained regression models', *Journal of Machine Learning Research* 18(143), 1–41.

White, C. A., Johnson, B. G., Gill, P. M. & Head-Gordon, M. (1994), 'The continuous fast multipole method', *Chemical physics letters* 230(1-2), 8–16.

Wu, C. J. & Hamada, M. S. (2011), *Experiments: planning, analysis, and optimization*, John Wiley & Sons.

Xie, R., Wang, Z., Bai, S., Ma, P. & Zhong, W. (2019), Online decentralized leverage score sampling for streaming multidimensional time series, in 'The 22nd International Conference on Artificial Intelligence and Statistics', pp. 2301–2311.

Yang, C., Duraiswami, R., Gumerov, N. A. & Davis, L. (2003), Improved fast Gauss transform and efficient kernel density estimation, in 'null', IEEE, p. 464.

Yu, J. & Wang, H. (2022), 'Subdata selection algorithm for linear model discrimination', *Statistical Papers* pp. 1–24.

Yu, J., Wang, H., Ai, M. & Zhang, H. (2020), 'Optimal distributed subsampling for maximum quasi-likelihood estimators with massive data', *Journal of the American Statistical Association* pp. 1–12.

Zhang, J., Ma, P., Zhong, W. & Meng, C. (Just accepted), 'Projection-based techniques for high-dimensional optimal transport problems', *Wiley Interdisciplinary Reviews: Computational Statistics* p. e1587.

Zhang, J., Zhong, W. & Ma, P. (2021), 'A review on modern computational optimal transport methods with applications in biomedical research', *Modern Statistical Methods for Health Research* pp. 279–300.

Zhang, X., Xie, R. & Ma, P. (2018), Statistical leveraging methods in big data, *in* 'Handbook of Big Data Analytics', Springer, pp. 51–74.

Zheng, Y., Jests, J., Phillips, J. M. & Li, F. (2013), Quality and efficiency for kernel density estimates in large data, *in* 'Proceedings of the 2013 ACM SIGMOD International Conference on Management of Data', ACM, pp. 433–444.

Zheng, Y., Ou, Y., Lex, A. & Phillips, J. M. (2017), Visualization of big spatial data using coresets for kernel density estimates, *in* '2017 IEEE Visualization in Data Science (VDS)', IEEE, pp. 23–30.

Zhou, L., Pan, S., Wang, J. & Vasilakos, A. V. (2017), 'Machine learning on big data: Opportunities and challenges', *Neurocomputing* **237**, 350–361.

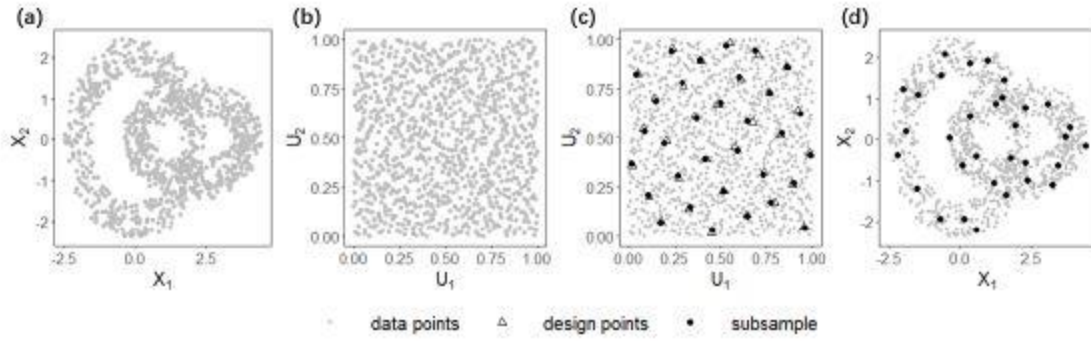


Fig. 1 Illustration for Algorithm 2. The two-dimensional sample, marked as gray dots in panel (a), is first transformed to be uniformly distributed on $[0, 1]^2$, shown in panel (b). We then generate a set of space-filling design points, marked as triangles, and search for the nearest neighbor for each of them, marked by black dots in panel (c). Panel (d) shows the subsample corresponding to the selected data points.

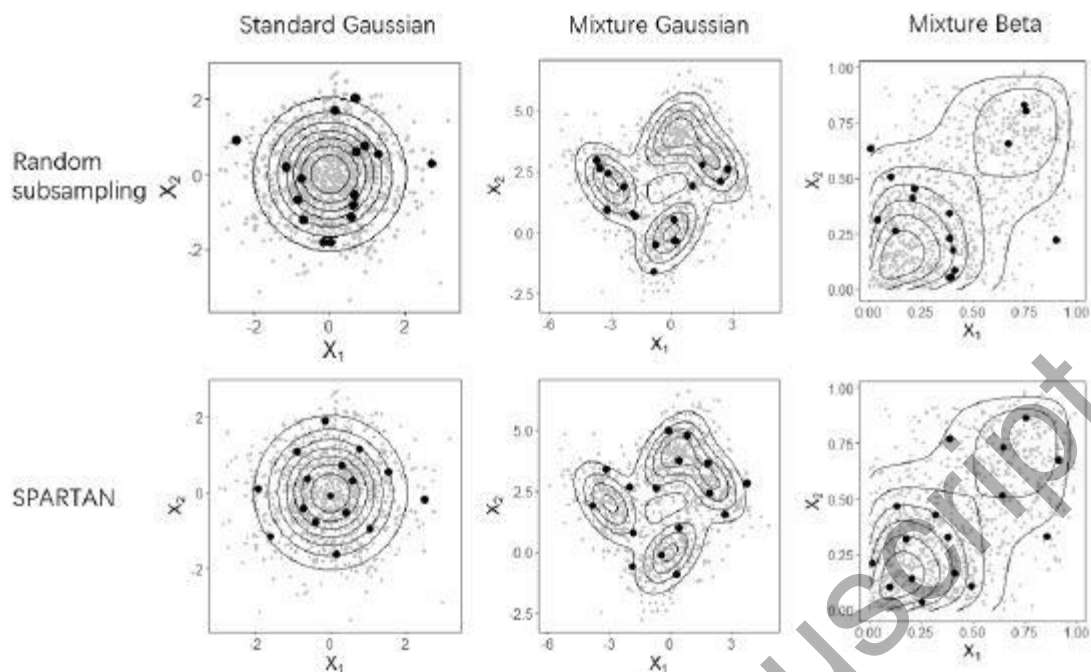


Fig. 2 Subsamples (black dots) selected by the proposed method (lower) versus randomly selected subsamples (upper). Contours (black) are superimposed. One can observe the proposed method selects subsamples that have more appealing visual representation of the corresponding population.

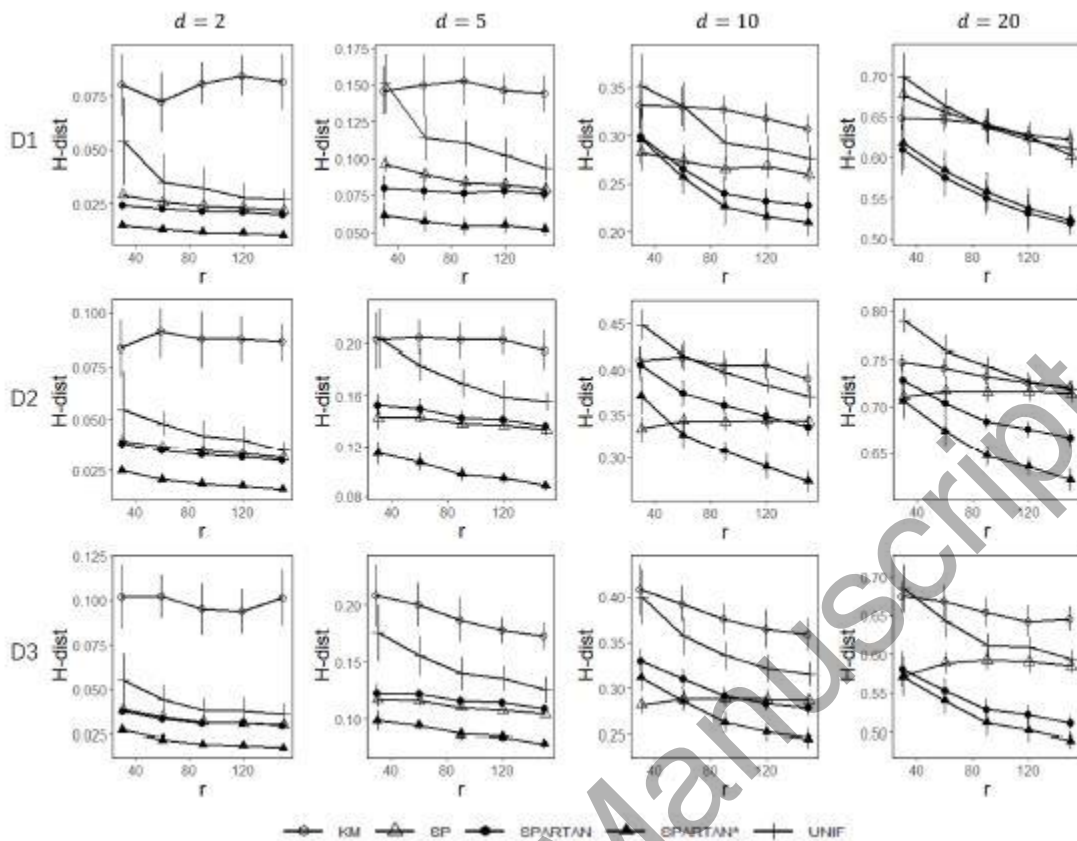


Fig. 3 Simulation under different d (from left to right) and different probability density functions (from upper to lower). The Hellinger distance (H-dist) are plotted versus different r . Vertical bars represent the standard errors.

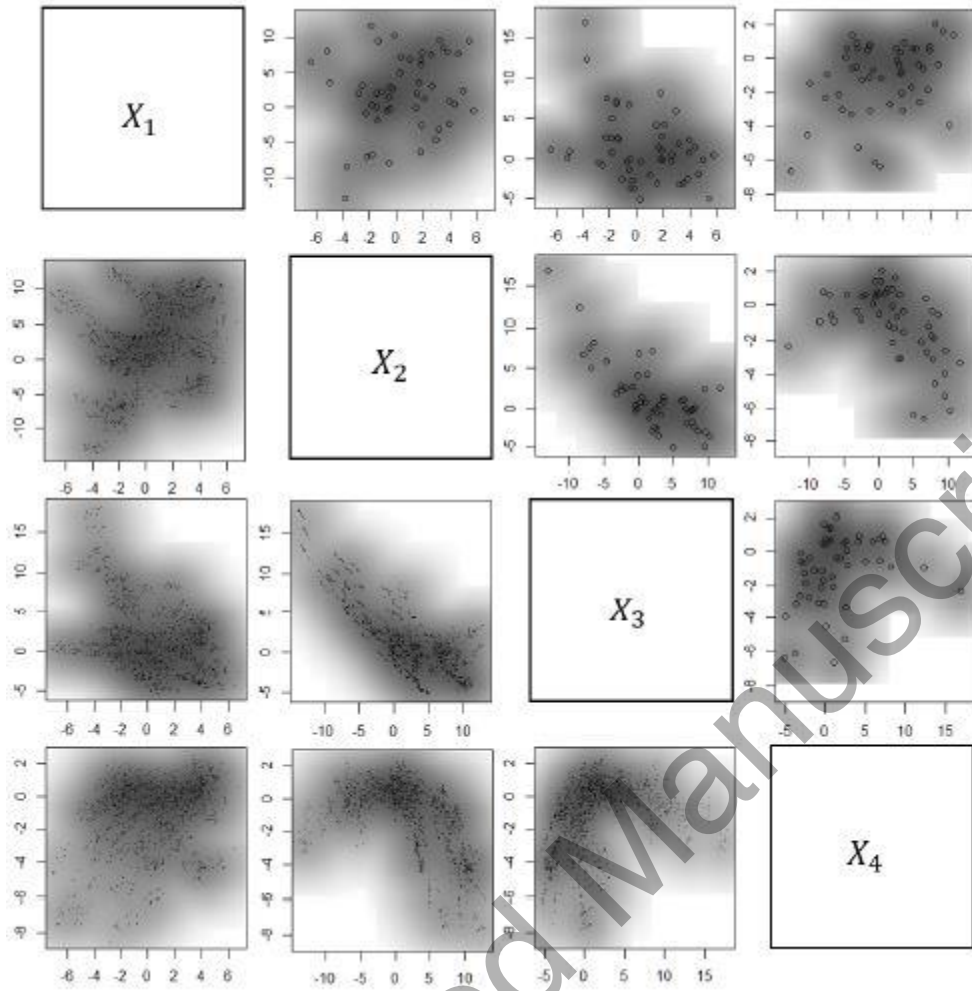


Fig. 4 Visualization of the banknote authentication dataset. The lower diagonal panels show the scatter plots for each pair of predictors. The upper diagonal panels show the scatter plots for the selected subsample using the proposed algorithm. The heat maps are obtained using kernel density estimation.

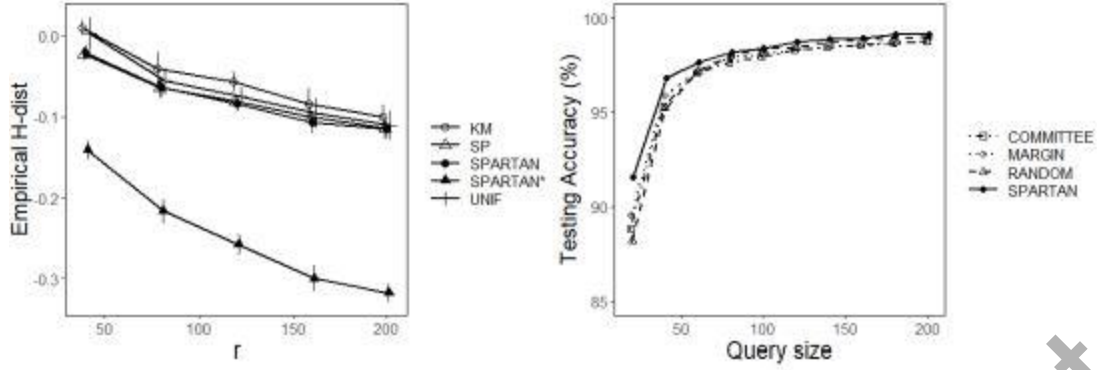


Fig. 5 Left: For the density estimation of the banknote authentication dataset, the empirical Hellinger distance (H-dist) is plotted versus different subsample sizes r . Right: For the active learning of the banknote authentication dataset, the testing accuracy is plotted versus different query sizes. Vertical bars represent the standard errors. In the right panel, the standard errors are tiny, and thus the error bars are almost invisible.