

Energy Clustering

Guilherme França* and Joshua T. Vogelstein†

Johns Hopkins University

Abstract

Energy statistics was proposed by Székely in the 80's inspired by the Newtonian gravitational potential from classical mechanics, and it provides a non-parametric hypothesis test for equality of distributions. It was further generalized from Euclidean spaces to metric spaces of strong negative type, and more recently, a connection with reproducing kernel Hilbert spaces (RKHS) was established. Here we consider the problem of clustering data from an energy statistics theory perspective. We provide a precise mathematical formulation yielding a quadratically constrained quadratic program (QCQP) in the associated RKHS. We show that this QCQP is equivalent to kernel k -means optimization problem, once the kernel is fixed. Thus, our results imply a first principles derivation of kernel k -means from energy statistics. However, energy statistics fixes a family of standard kernels. Furthermore, we also consider a weighted version of energy statistics, making connection to graph partitioning problems. To find local optimizers of such QCQP we consider an iterative algorithm based on Hartigan's method, which in this case has the same computational cost as kernel k -means algorithm, based on Lloyd's method, but usually with better clustering quality. We provide carefully designed numerical experiments showing the superiority of the proposed method compared to kernel k -means, standard k -means, and Gaussian mixture models in a variety of settings.

* guifranca@gmail.com

† jovo@jhu.edu

I. INTRODUCTION

Energy statistics [1] is based on a notion of statistical potential energy between probability distributions, in close analogy to Newton’s gravitational potential in classical mechanics. It provides a non-parametric hypothesis test for equality of distributions which is achieved under minimum energy. When probability distributions are different the statistical potential energy diverges as sample size increases, while tends to a nondegenerate limit distribution when probability distributions are equal. Energy statistics has been applied to several goodness-of-fit hypothesis tests, multi-sample tests of equality of distributions, analysis of variance [2], nonlinear dependence tests through distance covariance and distance correlation, which generalizes the Pearson correlation coefficient, and hierarchical clustering [3] by extending Ward’s method of minimum variance. Moreover, an application of energy statistics to clustering was already proposed [4], which in part motivated this paper. We refer the reader to [1], and references therein, for an overview of energy statistics theory and its applications.

In its original formulation, energy statistics has a compact representation in terms of expectations of pairwise Euclidean distances, providing straightforward empirical estimates. The notion of distance covariance was further generalized from Euclidean spaces to metric spaces of strong negative type [5]. Furthermore, the missing link between energy distance based tests and kernel based tests has been recently resolved [6], where a unifying framework establishing an equivalence between generalized energy distances to maximum mean discrepancies (MMD), which are distances between embeddings of distributions in reproducing kernel Hilbert spaces (RKHS), was established. This equivalence immediately relates energy statistics to kernel methods, often used in machine learning, and form the basis of our approach.

Clustering has such a long history in machine learning, making it impossible to mention all important contributions in a short space. Perhaps, the most used method is k -means [7–9], which is based on Lloyd’s heuristic [7] of assigning a data point to the cluster with closest center. The only statistical information about each cluster comes from its mean, making it sensitive to outliers. Nevertheless, k -means works very well when data is linearly separable in Euclidean space. Gaussian mixture models (GMM) is another very common approach, providing more flexibility than k -means, however, it still makes strong assumptions about

the distribution of the data.

To account for nonlinearities, kernel methods were introduced [10, 11]. A Mercer kernel [12] is used to implicitly map data points to a RKHS, then clustering can be performed in the associated Hilbert space by using its inner product. However, the kernel choice remains the biggest challenge since there is no principled theory to construct a kernel for a given dataset, and usually a kernel introduces hyperparameters that need to be carefully chosen. The well-known kernel k -means optimization problem is nothing but k -means in the feature space [11]. Furthermore, kernel k -means algorithm [13, 14] is still based on Lloyd’s heuristic [7] of grouping points that are closer to a cluster center, now in feature space. We refer the reader to [15] for a survey of clustering methods.

Although clustering from energy statistics was considered in [4], the precise optimization problem behind this approach remains elusive, as well as the connection with kernel methods. The main theoretical contribution of this paper is to fill this gap. Since the statistical potential energy is minimum when distributions are equal, the principle behind clustering is to maximize the statistical energy, enforcing probability distributions associated to each cluster to be different from one another. We provide a precise mathematical formulation to this statement, leading to a quadratically constrained quadratic program (QCQP) in the associated RKHS. Our results immediately establish the connection with kernel methods, showing that this QCQP is equivalent to kernel k -means optimization problem. The equivalence between kernel k -means, spectral clustering, and graph partitioning problems is well-known [13, 14]. We demonstrate how these relations arise from a weighted version of energy statistics.

Our main algorithmic contribution is to use Hartigan’s method [16] to find local solutions of the above mentioned QCQP, which is NP-hard in general. Hartigan’s method was also used in [4]. More importantly, its advantages over Lloyd’s method was already demonstrated in some simple settings [17, 18], but apparently this method did not receive the deserved attention. To the best of our knowledge, Hartigan’s method was not previously employed together with kernel methods. We provide a full kernel based Hartigan’s algorithm for clustering, where the kernel is fixed by energy statistics. We make clear the advantages of this proposal versus Lloyd’s method, which kernel k -means is based upon and will also be used to solve our QCQP. We show that both algorithms have the same time complexity, but Hartigan’s method in kernel spaces offer several advantages. Furthermore, in the exam-

ples considered in this paper, it also provides superior performance compared to a spectral clustering relaxation of the QCQP.

Our numerical results provide compelling evidence that Hartigan’s method applied to energy statistics based clustering, or *energy clustering* for short, is more accurate and robust than Lloyd’s method. Our experiments also put in evidence the flexibility of energy clustering, which provides a family of default kernels, showing that it is able to perform accurately on data coming from very different distributions, contrary to k -means and GMM for instance. More specifically, our algorithms for energy clustering perform closely to k -means and GMM on normally distributed data, but on the other hand, performs considerably better than k -means and GMM on data that is not normally distributed. It also performs better than k -means and GMM in high dimensions, even on Gaussian settings. However, it performs worse than GMM, and more closely to k -means, for highly unbalanced clusters with normal distributions.

Our work is organized as follows. In section II, we introduce the necessary background on energy statistics and RKHS. Section III contains the main theoretical results of this paper, where we consider a clustering theory based on energy statistics leading to a QCQP, which is NP-hard in general. We also show the equivalence to kernel k -means optimization problem. In Section IV, we generalize these results to a weighted version of energy statistics, providing connections to graph partitioning problems. In Section V, we consider the simple case of a two-class problem in one dimension, where we propose an algorithm which requires no random initialization. In section VI, we consider Lloyd’s and Hartigan’s methods and propose iterative algorithms to solve the QCQP. Section VII contains some carefully designed numerical experiments showing that, in a variety of settings, energy clustering based on Hartigan’s algorithm outperform k -means, GMM, kernel k -means, and a version of spectral clustering. Our final conclusions are presented in Section VIII.

II. BACKGROUND ON ENERGY STATISTICS AND RKHS

In this section we introduce the main concepts from energy statistics and its relation to RKHS which form the basis of our work. For more details we refer the reader to [1] and [6].

Consider random variables in \mathbb{R}^D such that $X, X' \stackrel{iid}{\sim} P$ and $Y, Y' \stackrel{iid}{\sim} Q$, where P and Q

are cumulative distribution functions with finite first moments. The quantity

$$\mathcal{E}(P, Q) \equiv 2\mathbb{E}\|X - Y\| - \mathbb{E}\|X - X'\| - \mathbb{E}\|Y - Y'\|, \quad (1)$$

called *energy distance* [1], is rotationally invariant and nonnegative, $\mathcal{E}(P, Q) \geq 0$, where equality to zero holds if and only if $P = Q$. Above, $\|\cdot\|$ denotes the Euclidean norm in \mathbb{R}^D . Energy distance provides a characterization of equality of distributions, and $\mathcal{E}^{1/2}$ is a metric on the space of distributions.

The energy distance can be generalized as, for instance,

$$\mathcal{E}_\alpha(P, Q) \equiv 2\mathbb{E}\|X - Y\|^\alpha - \mathbb{E}\|X - X'\|^\alpha - \mathbb{E}\|Y - Y'\|^\alpha \quad (2)$$

where $0 < \alpha \leq 2$. This quantity is also nonnegative, $\mathcal{E}_\alpha(P, Q) \geq 0$. Furthermore, for $0 < \alpha < 2$ we have that $\mathcal{E}_\alpha(P, Q) = 0$ if and only if $P = Q$, while for $\alpha = 2$ we have $\mathcal{E}_2(P, Q) = 2\|\mathbb{E}(X) - \mathbb{E}(Y)\|^2$ which shows that equality to zero only requires equality of the means, and thus $\mathcal{E}_2(P, Q) = 0$ does not imply equality of distributions.

The energy distance can be even further generalized. Let $X, Y \in \mathcal{X}$ where \mathcal{X} is an arbitrary space endowed with a *semimetric of negative type* $\rho : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$, which is required to satisfy

$$\sum_{i,j=1}^n c_i c_j \rho(X_i, X_j) \leq 0, \quad (3)$$

where $X_i \in \mathcal{X}$ and $c_i \in \mathbb{R}$ such that $\sum_{i=1}^n c_i = 0$. Then, \mathcal{X} is called a *space of negative type*. We can thus replace $\mathbb{R}^D \rightarrow \mathcal{X}$ and $\|X - Y\| \rightarrow \rho(X, Y)$ in the definition (1), obtaining the generalized energy distance

$$\mathcal{E}(P, Q) \equiv 2\mathbb{E}\rho(X, Y) - \mathbb{E}\rho(X, X') - \mathbb{E}\rho(Y, Y'). \quad (4)$$

For spaces of negative type there exists a Hilbert space \mathcal{H} and a map $\varphi : \mathcal{X} \rightarrow \mathcal{H}$ such that $\rho(X, Y) = \|\varphi(X) - \varphi(Y)\|_{\mathcal{H}}^2$. This allows us to compute quantities related to probability distributions over \mathcal{X} in the associated Hilbert space \mathcal{H} . Even though the semimetric ρ may not satisfy the triangle inequality, $\rho^{1/2}$ does since it can be shown to be a proper metric.

There is an equivalence between energy distance, commonly used in statistics, and distances between embeddings of distributions in RKHS, commonly used in machine learning. This equivalence was established in [6]. Let us first recall the definition of RKHS. Let \mathcal{H} be a Hilbert space of real-valued functions over \mathcal{X} . A function $K : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ is a reproducing kernel of \mathcal{H} if it satisfies the following two conditions:

1. $h_x \equiv K(\cdot, x) \in \mathcal{H}$ for all $x \in \mathcal{X}$.
2. $\langle h_x, f \rangle_{\mathcal{H}} = f(x)$ for all $x \in \mathcal{X}$ and $f \in \mathcal{H}$.

In other words, for any $x \in \mathcal{X}$ and any function $f \in \mathcal{H}$, there is a unique $h_x \in \mathcal{H}$ that reproduces $f(x)$ through the inner product of \mathcal{H} . If such a *kernel* function K exists, then \mathcal{H} is called a RKHS. The above two properties immediately imply that K is symmetric and positive definite. Indeed, notice that $\langle h_x, h_y \rangle = h_y(x) = K(x, y)$, and by definition $\langle h_x, h_y \rangle^* = \langle h_y, h_x \rangle$, but since the inner product is real we have $\langle h_y, h_x \rangle = \langle h_x, h_y \rangle$, or equivalently $K(y, x) = K(x, y)$. Moreover, for any $w \in \mathcal{H}$ we can write $w = \sum_{i=1}^n c_i h_{x_i}$ where $\{h_{x_i}\}_{i=1}^n$ is a basis of \mathcal{H} . It follows that $\langle w, w \rangle_{\mathcal{H}} = \sum_{i,j=1}^n c_i c_j K(x_i, x_j) \geq 0$, showing that the kernel is positive definite. If G is a matrix with elements $G_{ij} = K(x_i, x_j)$ this is equivalent to G being positive semidefinite, i.e. $v^\top G v \geq 0$ for any vector $v \in \mathbb{R}^n$.

The Moore-Aronszajn theorem [19] establishes the converse of the above paragraph. For every symmetric and positive definite function $K : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$, there is an associated RKHS \mathcal{H}_K with reproducing kernel K . The map $\varphi : x \mapsto h_x \in \mathcal{H}_K$ is called the *canonical feature map*. Given a kernel K , this theorem enables us to define an embedding of a probability measure P into the RKHS as follows: $P \mapsto h_P \in \mathcal{H}_K$ such that $\int f(x) dP(x) = \langle f, h_P \rangle$ for all $f \in \mathcal{H}_K$, or alternatively, $h_P \equiv \int K(\cdot, x) dP(x)$. We can now introduce the notion of distance between two probability measures using the inner product of \mathcal{H}_K , which is called the maximum mean discrepancy (MMD) and is given by

$$\gamma_K(P, Q) \equiv \|h_P - h_Q\|_{\mathcal{H}_K}. \quad (5)$$

This can also be written as [20]

$$\gamma_K^2(P, Q) = \mathbb{E}K(X, X') + \mathbb{E}K(Y, Y') - 2\mathbb{E}K(X, Y) \quad (6)$$

where $X, X' \stackrel{iid}{\sim} P$ and $Y, Y' \stackrel{iid}{\sim} Q$. From the equality between (5) and (6) we also have

$$\langle h_P, h_Q \rangle_{\mathcal{H}_K} = \mathbb{E}K(X, Y). \quad (7)$$

Thus, in practice, we can estimate the inner product between embedded distributions by averaging the kernel function over sampled data.

The following important result shows that semimetrics of negative type and symmetric positive definite kernels are closely related [21]. Let $\rho : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ and $x_0 \in \mathcal{X}$ an arbitrary

but fixed point. Define

$$K(x, y) \equiv \frac{1}{2} [\rho(x, x_0) + \rho(y, x_0) - \rho(x, y)]. \quad (8)$$

Then, it can be shown that K is positive definite if and only if ρ is a semimetric of negative type. We have a family of kernels, one for each choice of x_0 . Conversely, if ρ is a semimetric of negative type and K is a kernel in this family, then

$$\begin{aligned} \rho(x, y) &= K(x, x) + K(y, y) - 2K(x, y) \\ &= \|h_x - h_y\|_{\mathcal{H}_K}^2 \end{aligned} \quad (9)$$

and the canonical feature map $\varphi : x \mapsto h_x$ is injective [6]. When these conditions are satisfied we say that the kernel K generates the semimetric ρ . If two different kernels generate the same ρ they are said to be equivalent kernels.

Now we can state the equivalence between energy distance \mathcal{E} and inner products on RKHS, which is one of the main results of [6]. If ρ is a semimetric of negative type and K a kernel that generates ρ , then replacing (9) into (4), and using (6), yields

$$\mathcal{E}(P, Q) = 2 [\mathbb{E} K(X, X') + \mathbb{E} K(Y, Y') - 2\mathbb{E} K(X, Y)] = 2\gamma_K^2(P, Q). \quad (10)$$

Due to (5) we can compute the energy distance $\mathcal{E}(P, Q)$ between two probability distributions using the inner product of \mathcal{H}_K .

Finally, let us recall the main formulas from energy statistics for the test statistic of equality of distributions [1]. Assume we have data $\mathbb{X} = \{x_1, \dots, x_n\}$ where $x_i \in \mathcal{X}$, and \mathcal{X} is a space of negative type. Consider a disjoint partition $\mathbb{X} = \bigcup_{j=1}^k \mathcal{C}_j$, with $\mathcal{C}_i \cap \mathcal{C}_j = \emptyset$. Each expectation in the generalized energy distance (4) can be computed through the function

$$g(\mathcal{C}_i, \mathcal{C}_j) \equiv \frac{1}{n_i n_j} \sum_{x \in \mathcal{C}_i} \sum_{y \in \mathcal{C}_j} \rho(x, y), \quad (11)$$

where $n_i = |\mathcal{C}_i|$ is the number of elements in partition \mathcal{C}_i . The *within energy dispersion* is defined by

$$W \equiv \sum_{j=1}^k \frac{n_j}{2} g(\mathcal{C}_j, \mathcal{C}_j), \quad (12)$$

and the *between-sample energy statistic* is defined by

$$S \equiv \sum_{1 \leq i < j \leq k} \frac{n_i n_j}{2n} [2g(\mathcal{C}_i, \mathcal{C}_j) - g(\mathcal{C}_i, \mathcal{C}_i) - g(\mathcal{C}_j, \mathcal{C}_j)], \quad (13)$$

where $n = \sum_{j=1}^k n_j$. Given a set of distributions $\{P_j\}_{j=1}^k$, where $x \in \mathcal{C}_j$ if and only if $x \sim P_j$, the quantity S provides a test statistic for equality of distributions [1]. When the sample size is large enough, $n \rightarrow \infty$, under the null hypothesis $H_0 : P_1 = P_2 = \dots = P_k$ we have that $S \rightarrow 0$, and under the alternative hypothesis $H_1 : P_i \neq P_j$ for at least two $i \neq j$, we have that $S \rightarrow \infty$. Note that this test does not make any assumptions about the distributions P_j , thus it is said to be non-parametric or distribution-free.

One can make a physical analogy by thinking that points $x \in \mathcal{C}_j$ form a massive body whose total mass is characterized by the distribution function P_j . The quantity S is thus a potential energy of the form $S(P_1, \dots, P_k)$ which measures how different the distribution of these masses are, and achieves the ground state $S = 0$ when all bodies have the same mass distribution. The potential energy S increases as bodies have different mass distributions.

III. CLUSTERING BASED ON ENERGY STATISTICS

This section contains the main theoretical results of this paper, where we formulate an optimization problem for clustering based on energy statistics and RKHS introduced in the previous section.

Due to the energy test statistic for equality of distributions, the obvious criterion for clustering data is to maximize S which makes each cluster as different as possible from the other ones. In other words, given a set of points coming from different probability distributions, the test statistic S should attain a maximum when each point is correctly classified as belonging to the cluster associated to its probability distribution. The following straightforward result shows that maximizing S is, however, equivalent to minimizing W which has a more convenient form.

Proposition 1. *Let $\mathbb{X} = \{x_1, \dots, x_n\}$ where each data point x_i lives in a space \mathcal{X} endowed with a semimetric $\rho : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ of negative type. For a fixed integer k , the partition $\mathbb{X} = \bigcup_{j=1}^k \mathcal{C}_j$, where $\mathcal{C}_i \cap \mathcal{C}_j = \emptyset$ for all $i \neq j$, maximizes the between-sample statistic S , defined in equation (13), if and only if*

$$\min_{\mathcal{C}_1, \dots, \mathcal{C}_k} W(\mathcal{C}_1, \dots, \mathcal{C}_k), \quad (14)$$

where the within energy dispersion W is defined by (12).

Proof. From (12) and (13) we have

$$\begin{aligned}
S + W &= \frac{1}{2n} \sum_{\substack{i,j=1 \\ i \neq j}}^k n_i n_j g(\mathcal{C}_i, \mathcal{C}_j) + \frac{1}{2n} \sum_{i=1}^k \left[n - \sum_{j \neq i=1}^k n_j \right] n_i g(\mathcal{C}_i, \mathcal{C}_i) \\
&= \frac{1}{2n} \sum_{i,j=1}^k n_i n_j g(\mathcal{C}_i, \mathcal{C}_j) = \frac{1}{2n} \sum_{x \in \mathbb{X}} \sum_{y \in \mathbb{X}} \rho(x, y) = \frac{n}{2} g(\mathbb{X}, \mathbb{X}).
\end{aligned} \tag{15}$$

Note that the right hand side of this equation only depends on the pooled data, so it is a constant independent of the choice of partition. Therefore, maximizing S over the choice of partition is equivalent to minimizing W . \square

For a given k , the clustering problem amounts to finding the best partition of the data by minimizing W . Notice that this is a hard clustering problem as partitions are disjoint. The optimization problem (14) based on energy statistics was already proposed in [4]. However, it is important to note that this is equivalent to maximizing S , which is the test statistic for equality of distributions. In this current form, the relation with kernels and other clustering methods is obscure. In the following, we show what is the explicit optimization problem behind (14) in the corresponding RKHS, establishing the connection with kernel methods.

Based on the relation between kernels and semimetrics of negative type, assume that the kernel $K : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ generates ρ . Define the Gram matrix

$$G \equiv \begin{pmatrix} K(x_1, x_1) & K(x_1, x_2) & \cdots & K(x_1, x_n) \\ K(x_2, x_1) & K(x_2, x_2) & \cdots & K(x_2, x_n) \\ \vdots & \vdots & \ddots & \vdots \\ K(x_n, x_1) & K(x_n, x_2) & \cdots & K(x_n, x_n) \end{pmatrix}. \tag{16}$$

Let $Z \in \{0, 1\}^{n \times k}$ be the label matrix, with only one nonvanishing entry per row, indicating to which cluster (column) each point (row) belongs to. This matrix satisfies $Z^\top Z = D$, where the diagonal matrix $D = \text{diag}(n_1, \dots, n_k)$ contains the number of points in each cluster. We also introduce the rescaled matrix $Y \equiv Z D^{-1/2}$. In component form they are given by

$$Z_{ij} \equiv \begin{cases} 1 & \text{if } x_i \in \mathcal{C}_j \\ 0 & \text{otherwise} \end{cases} \quad Y_{ij} \equiv \begin{cases} \frac{1}{\sqrt{n_j}} & \text{if } x_i \in \mathcal{C}_j \\ 0 & \text{otherwise} \end{cases}. \tag{17}$$

Throughout the paper, we use the notation $M_{i\bullet}$ to denote the i th row of a matrix M , and $M_{\bullet j}$ denotes its j th column. Our next result shows that the optimization problem (14) is NP-hard since it is a quadratically constrained quadratic program (QCQP) in the RKHS.

Proposition 2. *The optimization problem (14) is equivalent to*

$$\max_Y \text{Tr}(Y^\top G Y) \quad \text{s.t. } Y \geq 0, Y^\top Y = I, Y Y^\top e = e, \quad (18)$$

where $e = (1, 1, \dots, 1)^\top \in \mathbb{R}^n$ is the all-ones vector, and G is the Gram matrix (16).

Proof. From (9), (11), and (12) we have

$$W(\mathcal{C}_1, \dots, \mathcal{C}_k) = \frac{1}{2} \sum_{j=1}^k \frac{1}{n_j} \sum_{x, y \in \mathcal{C}_j} \rho(x, y) = \sum_{j=1}^k \sum_{x \in \mathcal{C}_j} \left(K(x, x) - \frac{1}{n_j} \sum_{y \in \mathcal{C}_j} K(x, y) \right). \quad (19)$$

Note that the first term is global so it does not contribute to the optimization problem. Therefore, minimizing (19) is equivalent to

$$\max_{\mathcal{C}_1, \dots, \mathcal{C}_k} \sum_{j=1}^k \frac{1}{n_j} \sum_{x, y \in \mathcal{C}_j} K(x, y). \quad (20)$$

But

$$\sum_{x, y \in \mathcal{C}_j} K(x, y) = \sum_{p=1}^n \sum_{q=1}^n Z_{pj} Z_{qj} G_{pq} = (Z^\top G Z)_{jj}, \quad (21)$$

where we used the definitions (16) and (17). Thus, the objective function in (20) is equal to $\text{Tr}(D^{-1} Z^\top G Z)$. Now we can use the cyclic property of the trace, and by the definition of the matrix Z in (17), we obtain the following integer programming problem:

$$\max_Z \text{Tr} \left((Z D^{-1/2})^\top G (Z D^{-1/2}) \right) \quad \text{s.t. } Z_{ij} \in \{0, 1\}, \sum_{j=1}^k Z_{ij} = 1, \sum_{i=1}^n Z_{ij} = n_j. \quad (22)$$

Now we write this in terms of the matrix $Y = Z D^{-1/2}$. The objective function immediately becomes $\text{Tr}(Y^\top G Y)$. Notice that the above constraints imply that $Z^\top Z = D$, where $D = \text{diag}(n_1, \dots, n_k)$, which in turn gives $D^{-1/2} Y^\top Y D^{-1/2} = D$, or $Y^\top Y = I$. Also, every entry of Y is positive by definition, $Y \geq 0$. Now it only remains to show the last constraint in (18), which comes from the last constraint in (22). In matrix form this reads $Z^\top e = D e$. Replacing $Z = Y D^{1/2}$ we have $Y^\top e = D^{1/2} e$. Multiplying this last equation on the left by Y , and noticing that $Y D^{1/2} e = Z e = e$, we finally obtain $Y Y^\top e = e$. Therefore, the optimization problem (22) is equivalent to (18). \square

Based on Proposition 2, to group data $\mathbb{X} = \{x_1, \dots, x_n\}$ into k clusters we first compute the Gram matrix G and then solve the optimization problem (18) for $Y \in \mathbb{R}^{n \times k}$. The i th row of Y will contain a single nonzero element in some j th column, indicating that

$x_i \in \mathcal{C}_j$. This optimization problem is nonconvex, and also NP-hard, thus a direct approach is computational prohibitive even for small datasets. However, one can find approximate solutions by relaxing some of the constraints, or obtaining a relaxed SDP version of it. For instance, the relaxed problem

$$\max_Y \text{Tr}(Y^\top G Y) \quad \text{s.t. } Y^\top Y = I \quad (23)$$

has a well-known closed form solution $Y^* = UR$, where the columns of $U \in \mathbb{R}^{n \times k}$ contain the top k eigenvectors of G corresponding to the k largest eigenvalues $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_k$, and $R \in \mathbb{R}^{k \times k}$ is an arbitrary orthogonal matrix. The resulting optimal objective function assumes the value $\max \text{Tr}(Y^{*\top} G Y^*) = \sum_{i=1}^k \lambda_i$.

Spectral clustering is based on the above approach, where one further normalizes the rows of Y^* , i.e. $Y^* \rightarrow \tilde{Y}$ where $\tilde{Y}_{ij} \equiv Y_{ij}^* / \|Y_{i\bullet}^*\|$. The vectors $y_i \equiv \tilde{Y}_{i\bullet} \in \mathbb{R}^k$, which are the rows of matrix \tilde{Y} , lie on the surface of the k -dimensional unit sphere. The set $\{y_i\}_{i=1}^n$ is a spectral embedding of the original set $\{x_i\}_{i=1}^n$ onto an eigen-subspace of the kernel matrix G . After performing this embedding, one groups the points $\{y_i\}_{i=1}^n$ into k clusters $\{\tilde{\mathcal{C}}_1, \dots, \tilde{\mathcal{C}}_k\}$ using any clustering algorithm, such as k -means. This yields the final solution where the original point x_i belongs to cluster \mathcal{C}_j if and only if y_i belongs to $\tilde{\mathcal{C}}_j$. A procedure on these lines was proposed in the seminal paper [22], which considers the eigen-decomposition of the matrix $D^{-1/2} G D^{-1/2}$ instead of G , and also where G is an affinity matrix with the modification that its diagonal elements are set to zero $G_{ii} = 0$, and $D = \text{diag}(d_1, \dots, d_n)$ is the degree matrix, with $d_i \equiv \sum_{j=1}^n G_{ij}$. We consider a slightly modification of this procedure, which we call \mathcal{E}^S -clustering to emphasize that it is a spectral method whose kernel comes from energy statistics, by computing the eigen-decomposition of GD^{-1} instead. This method is described in Algorithm 1 and will be used as a comparison to the other iterative algorithms that will be proposed later.

Note that the energy clustering problem (18) is valid for data living in an *arbitrary* space of negative type, where a semimetric ρ , and thus the kernel K , are assumed to be known. Standard energy statistics in Euclidean space fixes a family of choices $\rho_\alpha(x, y) = \|x - y\|^\alpha$, with the corresponding kernel $K_\alpha(x, y) = \frac{1}{2}(\|x\|^\alpha + \|y\|^\alpha - \|x - y\|^\alpha)$, where we set $x_0 = 0$ in (8) and $0 < \alpha \leq 2$. The same would be valid for data living in a more general metric space (\mathcal{X}, ρ) , where the corresponding semimetric ρ fixes the kernel. In any case, the energy clustering formulation does not make assumptions about the distribution of the data,

Algorithm 1 \mathcal{E}^S -clustering is an exact spectral method to solve the relaxed problem (23).

input Gram matrix $G \in \mathbb{R}^{n \times n}$, and number of clusters k

output label matrix $Z \in \{0, 1\}^{n \times k}$

- 1: set the diagonal of G to zero: $G_{ii} \leftarrow 0$ for $i = 1, \dots, n$
 - 2: $D \leftarrow \text{diag}(d_1, \dots, d_n)$ where $d_i \leftarrow \sum_{j=1}^n G_{ij}$
 - 3: solve the generalized eigenvalue problem $Gv_i = \lambda_i Dv_i$, where $i = 1, \dots, n$ with eigenvalues, and corresponding eigenvectors, obeying the ordering $\lambda_1 \geq \lambda_2 \geq \dots \lambda_n$
 - 4: $Y \leftarrow (Dv_1 | \dots | Dv_k) \in \mathbb{R}^{n \times k}$ has the top k eigenvectors (multiplied by D) as columns
 - 5: normalize the rows Y , i.e. $\tilde{Y}_{ij} \leftarrow Y_{ij} / \left(\sum_{j=1}^n |Y_{ij}|^2 \right)^{1/2}$
 - 6: apply k -means on $\{y_i\}_{i=1}^n$, where $y_i = \tilde{Y}_{i\bullet}$ is the i th row of \tilde{Y}
 - 7: $Z_{ij} \leftarrow 1$ if y_i is in cluster j , and $Z_{ij} \leftarrow 0$ otherwise
-

contrary to k -means and GMM, for example. In practice, however, the clustering quality strongly depend on the choice of a suitable ρ , which measures the similarity between data points, and is equivalent to choosing an appropriate kernel. However, if prior information is available to choose an appropriate ρ , it can be conveniently incorporated in the optimization problem (18).

Relation to Kernel k -Means

One may wonder how energy clustering relates to the well-known kernel k -means problem¹ which is extensively used in machine learning. For a positive semidefinite Gram matrix G , as defined in (16), there exists a map $\varphi : \mathcal{X} \rightarrow \mathcal{H}_K$ such that $K(x, y) = \varphi(x)^\top \varphi(y)$. The kernel k -means optimization problem, in feature space, is defined by

$$\min_{\mathcal{C}_1, \dots, \mathcal{C}_k} \left\{ J(\mathcal{C}_1, \dots, \mathcal{C}_k) \equiv \sum_{j=1}^k \sum_{x \in \mathcal{C}_j} \|\varphi(x) - \varphi(\mu_j)\|^2 \right\} \quad (24)$$

where $\mu_j = \frac{1}{n_j} \sum_{x \in \mathcal{C}_j} x$ is the mean of cluster \mathcal{C}_j in the ambient space. Notice that the above objective function is strongly tied to the idea of minimizing distances between points and

¹ When we refer to kernel k -means problem we mean specifically the optimization problem (24), which should not be confused with kernel k -means algorithm that is just one possible recipe to solve (24).

cluster centers, which arises from k -means objective function based on Lloyd's method [7]. It is known [13, 14] that problem (24) can be cast into a trace maximization in the same form as (18). The next result makes this explicit, showing that (14) and (24) are actually equivalent.

Proposition 3. *For a fixed kernel, the clustering optimization problem (14) based on energy statistics is equivalent to the kernel k -means optimization problem (24), and both are equivalent to (18).*

Proof. Notice that $\|\varphi(x) - \varphi(\mu_j)\|^2 = \varphi(x)^\top \varphi(x) - 2\varphi(x)^\top \varphi(\mu_j) + \varphi(\mu_j)^\top \varphi(\mu_j)$, therefore

$$J = \sum_{j=1}^k \sum_{x \in \mathcal{C}_j} \left(K(x, x) - \frac{2}{n_j} \sum_{y \in \mathcal{C}_j} K(x, y) + \frac{1}{n_j^2} \sum_{y, z \in \mathcal{C}_j} K(y, z) \right). \quad (25)$$

The first term is global so it does not contribute to the optimization problem. Notice that the third term gives $\sum_{x \in \mathcal{C}_j} \frac{1}{n_j^2} \sum_{y, z \in \mathcal{C}_j} K(y, z) = \frac{1}{n_j} \sum_{y, z \in \mathcal{C}_j} K(y, z)$, which is the same as the second term. Thus, problem (24) is equivalent to

$$\max_{\mathcal{C}_1, \dots, \mathcal{C}_k} \sum_{j=1}^k \frac{1}{n_j} \sum_{x, y \in \mathcal{C}_j} K(x, y) \quad (26)$$

which is exactly the same as (20) from the energy statistics formulation. Therefore, once the kernel K is fixed, the function W given by (12) is the same as J in (24). The remaining of the proof proceeds as already shown in the proof of Proposition 2, leading to the optimization problem (18). \square

The above result shows that kernel k -means problem is equivalent to the clustering problem formulated in the energy statistics framework, when operating on the same kernel. Notice, however, that energy statistics theory is valid for arbitrary semimetric spaces of negative type, fixing the kernel function in the associated RKHS, which is guaranteed to be positive definite.

As shown in [13, 14], kernel k -means, spectral clustering, and graph partitioning problems such as ratio association, ratio cut, and normalized cut are all equivalent to a QCQP of the form (18). Thus one can use kernel k -means algorithm to solve these problems as well. This correspondence involves a weighted version of problem (18), that we demonstrate in the following from the perspective of energy statistics.

IV. CLUSTERING BASED ON WEIGHTED ENERGY STATISTICS

We generalize the formulas from energy statistics to incorporate weights associated to each data point. Let $w(x)$ be a weight function associated to point $x \in \mathcal{X}$. We can generalize (11) as follows:

$$g(\mathcal{C}_i, \mathcal{C}_j) \equiv \frac{1}{s_i s_j} \sum_{x \in \mathcal{C}_i} \sum_{y \in \mathcal{C}_j} w(x)w(y)\rho(x, y), \quad s_i \equiv \sum_{x \in \mathcal{C}_i} w(x). \quad (27)$$

Now we replace this function in the formulas (12) and (13), with $n_i \rightarrow s_i$ and $n \rightarrow s$, where $s = \sum_{j=1}^k s_j$. With these changes, Proposition 1 remains the unaltered, so the clustering problem becomes

$$\min_{\mathcal{C}_1, \dots, \mathcal{C}_k} \left\{ W(\mathcal{C}_1, \dots, \mathcal{C}_k) \equiv \sum_{j=1}^k \frac{s_j}{2} g(\mathcal{C}_j, \mathcal{C}_j) \right\} \quad (28)$$

where now g is given by (27). Let us define the following matrices and vector:

$$Y_{ij} \equiv \begin{cases} \frac{1}{\sqrt{s_j}} & \text{if } x_i \in \mathcal{C}_j \\ 0 & \text{otherwise} \end{cases}, \quad \mathcal{W} \equiv \text{diag}(w_1, \dots, w_n), \quad H \equiv \mathcal{W}^{1/2} Y, \quad \omega \equiv \mathcal{W}e, \quad (29)$$

where $w_i = w(x_i)$ and $e \in \mathbb{R}^n$ is the all-ones vector. We can now show the analogous of Proposition 2 to the case of problem (28).

Proposition 4. *The weighted energy clustering given by problem (28) is equivalent to*

$$\max_H \text{Tr} \{ H^\top (\mathcal{W}^{1/2} G \mathcal{W}^{1/2}) H \} \quad \text{s.t. } H \geq 0, H^\top H = I, H H^\top \omega = \omega, \quad (30)$$

where G is the Gram matrix (16), $\omega = (w_1, \dots, w_n)^\top$ contains the weights of each point, and $\mathcal{W} = \text{diag}(\omega)$.

Proof. Replacing (9) and eliminating the global terms which do not contribute, the optimization problem (28) becomes

$$\max_{\mathcal{C}_1, \dots, \mathcal{C}_k} \sum_{j=1}^k \frac{1}{s_j} \sum_{x \in \mathcal{C}_j} \sum_{y \in \mathcal{C}_j} w(x)w(y)K(x, y). \quad (31)$$

This objective function can be written as

$$\begin{aligned} \sum_{j=1}^k \frac{1}{s_j} \sum_{p=1}^n \sum_{q=1}^n w_p w_q Z_{pj} Z_{qj} G_{pq} &= \sum_{j=1}^k \sum_{p=1}^n \sum_{q=1}^n \frac{Z_{jp}^\top \sqrt{w_p}}{\sqrt{s_j}} w_p^{1/2} G_{pq} w_q^{1/2} \frac{\sqrt{w_q} Z_{qj}}{\sqrt{s_j}} \\ &= \sum_{j=1}^k (H^\top \mathcal{W}^{1/2} G \mathcal{W}^{1/2} H)_{jj} \\ &= \text{Tr} (H^\top \mathcal{W}^{1/2} G \mathcal{W}^{1/2} H). \end{aligned} \quad (32)$$

To obtain the constraints, note that $H_{ij} \geq 0$ by definition, and

$$(H^\top H)_{ij} = \sum_{\ell=1}^n Y_{\ell i} \mathcal{W}_{\ell\ell} Y_{\ell j} = \frac{1}{\sqrt{s_i} \sqrt{s_j}} \sum_{\ell=1}^n w_\ell Z_{\ell i} Z_{\ell j} = \frac{\delta_{ij}}{s_i} \sum_{\ell=1}^n w_\ell Z_{\ell i} = \delta_{ij}, \quad (33)$$

therefore $H^\top H = I$. This is a constraint on the rows of H . To obtain a condition on its columns observe that

$$(H^\top H)_{pq} = \sqrt{w_p w_q} \sum_{j=1}^k \frac{Z_{pj} Z_{qj}}{s_j} = \begin{cases} \frac{\sqrt{w_p w_q}}{s_i} & \text{if both } x_p, x_q \in \mathcal{C}_i \\ 0 & \text{otherwise.} \end{cases} \quad (34)$$

Therefore, $(H^\top H \mathcal{W}^{1/2})_{pq} = \sqrt{w_p} w_q s_i^{-1}$ if both points x_p and x_q belong to the same cluster, which we denote by \mathcal{C}_i for some $i \in \{1, \dots, k\}$, and $(H^\top H \mathcal{W}^{1/2})_{pq} = 0$ otherwise. Thus, the p th line of this matrix is nonzero only on entries corresponding to points that are in the same cluster as x_p . If we sum over the columns of this line we obtain $\sqrt{w_p} s_i^{-1} \sum_{q=1}^n w_q Z_{qi} = \sqrt{w_p}$, or equivalently $HH^\top \mathcal{W}^{1/2} e = \mathcal{W}^{1/2} e$, which gives the constraint $HH^\top \omega = \omega$. \square

Connection with Graph Partitioning

The relation between kernel k -means and graph partitioning problems is known [13, 14]. For conciseness, we repeat a similar analysis due to the relation of these problems to energy statistics and RKHS, which provides a different perspective.

Consider a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{A})$ where \mathcal{V} is the set of vertices, \mathcal{E} the set of edges, and \mathcal{A} is an affinity matrix of the graph that measures the similarities between pairs of nodes. Thus, $\mathcal{A}_{ij} \neq 0$ if $(i, j) \in \mathcal{E}$, and $\mathcal{A}_{ij} = 0$ otherwise. We also associate weights to every vertex, $w_i = w(i)$ for $i \in \mathcal{V}$, and let $s_j = \sum_{i \in \mathcal{C}_j} w_i$, where $\mathcal{C}_j \subseteq \mathcal{V}$ is one partition of \mathcal{V} . Let

$$\text{links}(\mathcal{C}_\ell, \mathcal{C}_m) \equiv \sum_{i \in \mathcal{C}_\ell, j \in \mathcal{C}_m} \mathcal{A}_{ij}. \quad (35)$$

We want to partition the set of vertices \mathcal{V} into k disjoint subsets, $\mathcal{V} = \bigcup_{j=1}^k \mathcal{C}_j$. The generalized ratio association problem is given by

$$\max_{\mathcal{C}_1, \dots, \mathcal{C}_k} \sum_{j=1}^k \frac{\text{links}(\mathcal{C}_j, \mathcal{C}_j)}{s_j} \quad (36)$$

and maximizes the within cluster association. The generalized ratio cut problem

$$\min_{\mathcal{C}_1, \dots, \mathcal{C}_k} \sum_{j=1}^k \frac{\text{links}(\mathcal{C}_j, \mathcal{V} \setminus \mathcal{C}_j)}{s_j} \quad (37)$$

minimizes the cut between clusters. These two problems are equivalent, in analogous way as minimizing (12) is equivalent to maximizing (13) as shown in Proposition 1. Here this is due to the equality $\text{links}(\mathcal{C}_j, \mathcal{V} \setminus \mathcal{C}_j) = \text{links}(\mathcal{C}_j, \mathcal{V}) - \text{links}(\mathcal{C}_j, \mathcal{C}_j)$. Several graph partitioning methods [23–26] can be seen as a particular case of (36) or (37).

Consider the ratio association problem (36), whose objective function can be written as

$$\sum_{j=1}^k \frac{1}{s_j} \sum_{p \in \mathcal{C}_j} \sum_{q \in \mathcal{C}_j} \mathcal{A}_{pq} = \sum_{j=1}^k \sum_{p=1}^n \sum_{q=1}^n \frac{Z_{jp}^\top}{\sqrt{s_j}} \mathcal{A}_{pq} \frac{Z_{qj}}{\sqrt{s_j}} = \text{Tr} (Y^\top \mathcal{A} Y), \quad (38)$$

with Z defined in (17) and Y in (29). Therefore, the ratio association problem can be written in the form (30) as

$$\max_H \text{Tr} (H^\top \mathcal{W}^{-1/2} \mathcal{A} \mathcal{W}^{-1/2} H) \quad \text{s.t. } H \geq 0, H^\top H = I, HH^\top \omega = \omega. \quad (39)$$

This is exactly the same problem as weighted energy clustering with $G = \mathcal{W}^{-1} \mathcal{A} \mathcal{W}^{-1}$. Assuming this matrix is positive semidefinite, this generates a semimetric (9) for graphs given by

$$\rho(i, j) = \frac{\mathcal{A}_{ii}}{w_i^2} + \frac{\mathcal{A}_{jj}}{w_j^2} - \frac{2\mathcal{A}_{ij}}{w_i w_j} \quad \text{or} \quad \rho(i, j) = -\frac{2\mathcal{A}_{ij}}{w_i w_j} \quad (40)$$

for vertices $i, j \in \mathcal{V}$, and where in the second equation we assume the graph has no self-loops, i.e. $\mathcal{A}_{ii} = 0$. Using (40) in the energy statistics formulation allows one to make inference on graphs. Above, the weight $w_i = w(i)$ of node $i \in \mathcal{V}$ can be, for instance, its degree $w_i = d(i)$.

V. TWO-CLASS PROBLEM IN ONE DIMENSION

Before stating a general algorithm to solve the optimization problem (18) we first consider the simplest possible case which is one-dimensional data and a two-class problem. This will be useful to test energy clustering on a simple setting.

Fixing $\rho(x, y) = |x - y|$ according to the standard energy distance, we can actually compute the function (11) in $\mathcal{O}(n \log n)$ and minimize W directly. This is done by noting that

$$\begin{aligned} |x - y| &= (x - y) \mathbb{1}_{x \geq y} - (x - y) \mathbb{1}_{x < y} \\ &= x (\mathbb{1}_{x \geq y} - \mathbb{1}_{x < y}) + y (\mathbb{1}_{y > x} - \mathbb{1}_{y \leq x}) \end{aligned} \quad (41)$$

where we have the indicator function defined by $\mathbb{1}_A = 1$ if A is true, and $\mathbb{1}_A = 0$ otherwise.

Let \mathcal{C} be a partition with n elements. Using the above distance we have

$$g(\mathcal{C}, \mathcal{C}) = \frac{1}{n^2} \sum_{x \in \mathcal{C}} \sum_{y \in \mathcal{C}} x (\mathbb{1}_{x \geq y} + \mathbb{1}_{y > x} - \mathbb{1}_{x \geq y} - \mathbb{1}_{x < y}). \quad (42)$$

Algorithm 2 \mathcal{E}^{1D} -clustering algorithm to find local solutions to the optimization problem (14) for a two-class problem in one dimension.

input data \mathbb{X}

output label matrix Z

- 1: sort \mathbb{X} obtaining $\tilde{\mathbb{X}} = [x_1, \dots, x_n]$
 - 2: **for** $j \in [1, \dots, n]$ **do**
 - 3: $\tilde{\mathcal{C}}_{1,j} \leftarrow [x_i : i = 1, \dots, j]$, and $\tilde{\mathcal{C}}_{2,j} \leftarrow [x_i : i = j + 1, \dots, n]$
 - 4: $W^{(j)} \leftarrow W(\tilde{\mathcal{C}}_{1,j}, \tilde{\mathcal{C}}_{2,j})$, from (44)
 - 5: **end for**
 - 6: $j^* \leftarrow \arg \min_j W^{(j)}$
 - 7: $Z_{j\bullet} \leftarrow (1, 0)$ if $j \leq j^*$, and $Z_{j\bullet} \leftarrow (0, 1)$ otherwise, for $j = 1, \dots, n$
-

The sum over y can be eliminated since each term in the parenthesis is simply counting the number of elements in \mathcal{C} that satisfy the condition of the indicator function. Assuming that we first order the data in \mathcal{C} , obtaining $\tilde{\mathcal{C}} = [x_j \in \mathcal{C} : x_1 \leq x_2 \leq \dots \leq x_n]$, we get

$$g(\tilde{\mathcal{C}}, \tilde{\mathcal{C}}) = \frac{2}{n^2} \sum_{\ell=1}^n (2\ell - 1 - n) x_\ell. \quad (43)$$

Note that the cost of computing $g(\tilde{\mathcal{C}}, \tilde{\mathcal{C}})$ is $\mathcal{O}(n)$ and the cost of sorting the data is at the most $\mathcal{O}(n \log n)$. Assuming that each partition is ordered, $\mathbb{X} = \bigcup_{j=1}^k \tilde{\mathcal{C}}_j$, the within energy dispersion can be written explicitly as

$$W(\tilde{\mathcal{C}}_1, \dots, \tilde{\mathcal{C}}_k) = \sum_{j=1}^k \sum_{\ell=1}^{n_j} \frac{2\ell - 1 - n_j}{n_j} x_\ell. \quad (44)$$

For a two-class problem we can use the formula (44) to cluster the data through a simple algorithm as follows. We first order the entire dataset, $\mathbb{X} \rightarrow \tilde{\mathbb{X}}$. Then we compute (44) for each possible split of $\tilde{\mathbb{X}}$ and pick the point which gives the minimum value of W . This procedure is described in Algorithm 2 and called \mathcal{E}^{1D} -clustering. Note that this algorithm is exact in the sense that it does not involve any random initialization, however, it only works for one-dimensional data with Euclidean distance. The total complexity of \mathcal{E}^{1D} -clustering is $\mathcal{O}(n \log n + n^2) = \mathcal{O}(n^2)$.

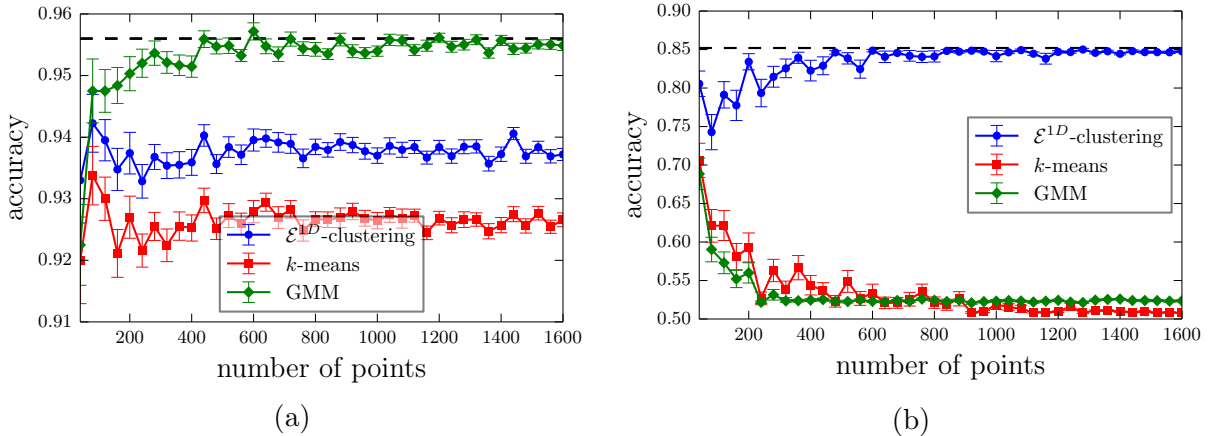


FIG. 1. \mathcal{E}^{1D} -clustering versus k -means and GMM in one dimension. The two clusters have the same number of points, and we use 50 Monte Carlo runs. We plot the average accuracy (45) versus the total number of points, where error bars are standard error. The dashed line indicates Bayes accuracy. (a) Data distributed as (46), where the optimal accuracy is ≈ 0.956 . (b) Data from (47) where the optimal accuracy is ≈ 0.852 .

Assuming the true label matrix Z is available, a direct measure of how different the estimated matrix \hat{Z} is from Z , up to label permutations, is given by

$$\text{accuracy}(\hat{Z}) \equiv \max_{\sigma} \frac{1}{n} \sum_{i=1}^n \sum_{j=1}^k \hat{Z}_{i\sigma(j)} Z_{ij} \quad (45)$$

where σ is a permutation of the k cluster groups. The accuracy is always between $[0, 1]$, where 1 corresponds to all points correctly clustered, and 0 to all points wrongly clustered. For a balanced two-class problem the value $1/2$ correspond to chance.

We now consider two simple experiments where we sample the same number of points for each cluster. We plot the average accuracy (45) versus the total number of points, with error bars indicating standard error. The data is clustered using \mathcal{E}^{1D} -clustering algorithm, GMM through EM algorithm, and the standard k -means algorithm. For k -means and GMM we use the implementations from the well-known *scikit-learn* library in Python [27], where k -means is initialized through k -means++ procedure [28], and GMM is initialized with the output of k -means. We choose to run both algorithms 5 times with different initializations and the answer with best objective function value is returned. Notice that \mathcal{E}^{1D} -clustering does not require any random initialization so we only run it once. For each fixed number of points we use 50 Monte Carlo runs. In Fig. 1a we have the results for data sampled

from two normal distributions,

$$x \sim \mathcal{N}(\mu_i, \sigma_i^2) \quad \text{with } \mu_1 = 0, \sigma_1 = 1 \text{ and } \mu_2 = 5, \sigma_2 = 2. \quad (46)$$

In this case the optimal accuracy obtained from Bayes classification error is ≈ 0.956 , indicated by the dashed line in the plot. The three methods perform closely, with a slight advantage of GMM, as expected since it corresponds to the true model of the data, and \mathcal{E}^{1D} -clustering performs slightly better than k -means. On the other hand, in Fig. 1b we consider two clusters with lognormal distributions,

$$\log x \sim \mathcal{N}(\mu_i, \sigma_i^2) \quad \text{with } \mu_1 = 0, \sigma_1 = 0.3 \text{ and } \mu_2 = -1.5, \sigma_2 = 1.5. \quad (47)$$

The optimal classification accuracy from Bayes error is ≈ 0.852 . We can now see that \mathcal{E}^{1D} -clustering is still very accurate, while GMM and k -means basically cluster at chance. The two simple experiments of Fig. 1 illustrate how energy statistics does not make assumptions about the distributions of the data, providing more flexibility and being able to accurately cluster data with very different distributions.

VI. ITERATIVE ALGORITHMS FOR ENERGY CLUSTERING

In this section we introduce an iterative algorithm to find a local maximizer of the optimization problem (18). Due to Proposition 3 we can also find an approximate solution by the well-known kernel k -means algorithm based on Lloyd's heuristic (see [13, 14]), which for convenience will also be restated in the present context.

Consider the optimization problem written in the form (20) as follows:

$$\max_{\{\mathcal{C}_1, \dots, \mathcal{C}_k\}} \left\{ Q = \sum_{j=1}^k \frac{Q_j}{n_j} \right\}, \quad Q_j \equiv \sum_{x, y \in \mathcal{C}_j} K(x, y), \quad (48)$$

where Q_j represents an internal energy cost of cluster \mathcal{C}_j , and Q is the total energy cost where each Q_j is weighted by the inverse of the number of points in \mathcal{C}_j . For a data point x_i we denote its own energy cost with the entire cluster \mathcal{C}_ℓ by

$$Q_\ell(x_i) \equiv \sum_{y \in \mathcal{C}_\ell} K(x_i, y) = G_{i\bullet} \cdot Z_{\bullet\ell}, \quad (49)$$

where we recall that $G_{i\bullet}$ ($G_{\bullet i}$) denotes the i th row (column) of matrix G .

Lloyd's Method for Energy Clustering

To optimize the kernel k -means objective function (25) we remove the global term and define the function

$$J^{(\ell)}(x_i) \equiv \frac{1}{n_\ell^2} Q_\ell - \frac{2}{n_\ell} Q_\ell(x_i). \quad (50)$$

We are thus solving

$$\min_Z \sum_{i=1}^n \sum_{\ell=1}^k Z_{i\ell} J^{(\ell)}(x_i). \quad (51)$$

One possible strategy is to assign x_i to cluster \mathcal{C}_{j^*} according to

$$j^* = \arg \min_{\ell=1, \dots, k} J^{(\ell)}(x_i). \quad (52)$$

This is done for every data point x_i and repeated until convergence, i.e. until no new assignments are made. The entire procedure is described in Algorithm 3, which we name \mathcal{E}^L -clustering to emphasize that we are optimizing the within energy function W based on Lloyd's method [7]. It can be shown that this algorithm converges provided G is positive semidefinite.

Obviously, \mathcal{E}^L -clustering is precisely kernel k -means algorithm [13, 14] but written more concisely and with the kernel induced by energy statistics. Indeed, recalling that $K(x, y) = \langle \varphi(x), \varphi(y) \rangle$ where $\varphi : \mathcal{X} \rightarrow \mathcal{H}_K$ is the feature map, we have from (50) that

$$J^{(\ell)}(x_i) = \langle \varphi(\mu_\ell), \varphi(\mu_\ell) \rangle - 2\langle \varphi(x_i), \varphi(\mu_\ell) \rangle = \|\varphi(x_i) - \varphi(\mu_\ell)\|^2 - \|\varphi(x_i)\|^2, \quad (53)$$

where $\mu_\ell = \frac{1}{n_\ell} \sum_{x \in \mathcal{C}_\ell} x$ is the mean of cluster \mathcal{C}_ℓ . Therefore, $\min_\ell J^{(\ell)}(x_i) = \min_\ell \|\varphi(x_i) - \varphi(\mu_\ell)\|^2$, i.e. we are assigning x_i to the cluster with closest center (in feature space), which is the familiar Lloyd's heuristic approach that kernel k -means is based upon.

To check the complexity of \mathcal{E}^L -clustering, notice that to compute the second term of $J^{(\ell)}(x_i)$ in (50) requires $\mathcal{O}(n_\ell)$ operations, and although the first term requires $\mathcal{O}(n_\ell^2)$ it only needs to be computed once outside loop through data points (step 1 of Algorithm 3). Therefore, the time complexity of \mathcal{E}^L -clustering is $\mathcal{O}(nk \max_\ell n_\ell) = \mathcal{O}(kn^2)$. For a sparse Gram matrix G having n' nonzero elements this complexity can be further reduced to $\mathcal{O}(kn')$.

Hartigan's Method for Energy Clustering

We now consider Hartigan's method [16] applied to the optimization problem in the form (48), which gives a local solution to the QCQP defined in (18). The method is based in

Algorithm 3 \mathcal{E}^L -clustering is Lloyd's method for energy clustering, which is precisely kernel k -means algorithm, with the kernel induced by energy statistics. This procedure finds local solutions to the problem (18).

input number of clusters k , Gram matrix G , initial label matrix $Z \leftarrow Z_0$

output label matrix Z

```

1:  $q \leftarrow (Q_1, \dots, Q_k)^\top$  have the costs of each cluster, defined in (48)
2:  $n \leftarrow (n_1, \dots, n_k)^\top$  have the number of points in each cluster
3: repeat
4:   for  $i = 1, \dots, n$  do
5:     let  $j$  be such that  $x_i \in \mathcal{C}_j$ 
6:      $j^* \leftarrow \arg \min_{\ell=1, \dots, k} J^{(\ell)}(x_i)$ , where  $J^{(\ell)}(x_i)$  is defined in (50)
7:     if  $j^* \neq j$  then
8:       move  $x_i$  to  $\mathcal{C}_{j^*}$ :  $Z_{ij} \leftarrow 0$  and  $Z_{ij^*} \leftarrow 1$ 
9:       update  $n$ :  $n_j \leftarrow n_j - 1$  and  $n_{j^*} \leftarrow n_{j^*} + 1$ 
10:      update  $q$ :  $q_j \leftarrow q_j - 2Q_j(x_i)$  and  $q_{j^*} \leftarrow q_{j^*} + 2Q_{j^*}(x_i)$ 
11:    end if
12:  end for
13: until convergence

```

computing the maximum change in the total cost function Q when moving each data point to another cluster. More specifically, suppose point x_i is currently assigned to cluster \mathcal{C}_j yielding a total cost function denoted by $Q^{(j)}$. Moving x_i to cluster \mathcal{C}_ℓ yields another total cost function denoted by $Q^{(\ell)}$. We are interested in computing the maximum cost change $\Delta Q^{j \rightarrow \ell}(x_i) \equiv Q^{(\ell)} - Q^{(j)}$, for $\ell \neq j$. From (48), by explicitly writing the costs related to these two cluster we obtain

$$\Delta Q^{j \rightarrow \ell}(x_i) = \frac{Q_\ell^+}{n_\ell + 1} + \frac{Q_j^-}{n_j - 1} - \frac{Q_j}{n_j} - \frac{Q_\ell}{n_\ell} \quad (54)$$

where Q_ℓ^+ denote the cost of the new ℓ th cluster with the point x_i added to it, and Q_j^- is the cost of new j th cluster with x_i removed from it. Noting that $Q_\ell^+ = Q_\ell + 2Q_\ell(x_i) + G_{ii}$

and $Q_j^- = Q_j - 2Q_j(x_i) + G_{ii}$, we get the formula

$$\Delta Q^{j \rightarrow \ell}(x_i) = \frac{1}{n_j - 1} \left[\frac{Q_j}{n_j} - 2Q_j(x_i) + G_{ii} \right] - \frac{1}{n_\ell + 1} \left[\frac{Q_\ell}{n_\ell} - 2Q_\ell(x_i) - G_{ii} \right]. \quad (55)$$

Therefore, if $\Delta Q^{j \rightarrow \ell}(x_i) > 0$ we get closer to a maximum of (48) by moving x_i to \mathcal{C}_ℓ , otherwise we keep x_i in \mathcal{C}_j .

We thus propose an algorithm that goes as follows. We start with an initial configuration for the label matrix Z , then for each point x_i we compute the cost of moving it to another cluster \mathcal{C}_ℓ , i.e. $\Delta Q^{j \rightarrow \ell}(x_i)$ for $\ell = 1, \dots, k$ with $\ell \neq j$, where j denotes the index of its current partition, $x \in \mathcal{C}_j$. Hence, we choose

$$j^* = \arg \max_{\ell=1, \dots, k \mid \ell \neq j} \Delta^{j \rightarrow \ell}(x_i). \quad (56)$$

If $\Delta Q^{j \rightarrow j^*}(x_i) > 0$ we move x_i to cluster \mathcal{C}_{j^*} , otherwise we keep x_i in its original cluster \mathcal{C}_j . This process is repeated until no points are assigned to new clusters. The entire procedure is explicitly described in Algorithm 4, which we denote \mathcal{E}^H -clustering to emphasize that it is based on Hartigan's method. This method automatically ensures that the objective function is monotonically increasing at each iteration, and consequently the algorithm converges in a finite number of steps.

The complexity analysis of \mathcal{E}^H -clustering is the following. Computing the Gram matrix G requires $\mathcal{O}(Dn^2)$ operations, where D is the dimension of each data point and n is the data size. However, both algorithms \mathcal{E}^L - and \mathcal{E}^H -clustering assume that G is given. There are more efficient methods to compute G , specially if it is sparse, but we will not consider this further and just assume that G is given. The computation of each cluster cost Q_j has complexity $\mathcal{O}(n_j^2)$, and overall to compute q we have $\mathcal{O}(n_1^2 + \dots + n_k^2) = \mathcal{O}(k \max_j n_j^2)$. These operations only need to be performed a single time. For each point x_i we need to compute $Q_j(x_i)$ once, which is $\mathcal{O}(n_j)$, and we need to compute $Q_\ell(x_i)$ for each $\ell \neq j$. The cost of computing $Q_\ell(x_i)$ is $\mathcal{O}(n_\ell)$, thus the cost of step 6 in Algorithm 4 is $\mathcal{O}(k \max_\ell n_\ell)$ for $\ell = 1, \dots, k$. For the entire dataset this gives a time complexity of $\mathcal{O}(nk \max_\ell n_\ell) = \mathcal{O}(kn^2)$. Note that this is the same cost as in \mathcal{E}^L -clustering, or kernel k -means algorithm. Again, if G is sparse this can be reduced to $\mathcal{O}(kn')$ where n' is the number of nonzero entries of G .

In the following we mention some important known results about Hartigan's method.

Theorem 5 (Telgarsky-Vattani [17]). *Hartigan's method has the cost function strictly decreasing in each iteration. Moreover, if $n > k$ then*

Algorithm 4 \mathcal{E}^H -clustering is Hartigan's method for energy clustering. This algorithm finds local solutions to the optimization problem (18).

input number of clusters k , Gram matrix G , initial label matrix $Z \leftarrow Z_0$

output label matrix Z

```

1:  $q \leftarrow (Q_1, \dots, Q_k)^\top$  have the energy costs of each cluster, defined in (48)
2:  $n \leftarrow (n_1, \dots, n_k)^\top$  have the number of points in each cluster
3: repeat
4:   for  $i = 1, \dots, n$  do
5:     let  $j$  be such that  $x_i \in \mathcal{C}_j$ 
6:      $j^* \leftarrow \arg \max_{\ell=1, \dots, k \mid \ell \neq j} \Delta Q^{j \rightarrow \ell}(x_i)$  using (55)
7:     if  $\Delta Q^{j \rightarrow j^*}(x_i) > 0$  then
8:       move  $x_i$  to  $\mathcal{C}_{j^*}$ :  $Z_{ij} \leftarrow 0$  and  $Z_{ij^*} \leftarrow 1$ 
9:       update  $n$ :  $n_j \leftarrow n_j - 1$  and  $n_{j^*} \leftarrow n_{j^*} + 1$ 
10:      update  $q$ :  $q_j \leftarrow q_j - 2Q_j(x_i) + G_{ii}$  and  $q_{j^*} \leftarrow q_{j^*} + 2Q_{j^*}(x_i) + G_{ii}$ 
11:    end if
12:  end for
13: until convergence

```

1. the resulting partition has no empty clusters, and
2. the resulting partition has distinct means.

Neither of these two conditions are guaranteed to be satisfied by Lloyd's method, and consequently by \mathcal{E}^L -clustering algorithm. The next result indicates that Hartigan's method can potentially escape local optima of Lloyd's method.

Theorem 6 (Telgarsky-Vattani [17]). *The set of local optima of Hartigan's method is a (possibly strict) subset of local optima of Lloyd's method.*

The above theorem implies that \mathcal{E}^L -clustering cannot improve on a local optima of \mathcal{E}^H -clustering. On the other hand, \mathcal{E}^H -clustering might improve on a local optima of \mathcal{E}^L -clustering. Lloyd's method forms Voronoi partitions, while Hartigan's method groups data in regions formed by the intersection of spheres called circlonoi cells. It can be shown that

the circlonoi cells are contained within a smaller volume of a Voronoi cell, and this excess volume grows exponentially with the dimension of \mathcal{X} [17, Theorems 2.4 and 3.1]. Points in this excess volume force Hartigan’s method to iterate, contrary to Lloyd’s method. Therefore, Hartigan’s can escape local optima of Lloyd’s. Moreover, this improvement should be more prominent as dimension increases. Also, the improvement grows as the number of clusters k increases. The empirical results of (author?) [17] show that an implementation of Hartigan’s method has comparable execution time to an implementation of Lloyd’s method, but no explicit complexity was provided. We show that both \mathcal{E}^L - and \mathcal{E}^H -clustering have the same time complexity. To the best of our knowledge, Hartigan’s method was not previously considered together with kernels, as we are proposing in \mathcal{E}^H -clustering.

In (author?) [18], Hartigan’s method was applied to k -means problem with any Bregman divergence. It was shown that the number of Hartigan’s local optima is upper bounded by $\mathcal{O}(1/k)$ [18, Proposition 5.1]. In addition, it was provided examples where *any* initial partition correspond to a local optima of Lloyd’s method, while the number of local optima in Hartigan’s method is small and correspond to true partitions of the data. Empirically, the number of Hartigan’s local optima was considerably smaller than the number of Lloyd’s local optima.

The above results indicate that Hartigan’s method provides several advantages over Lloyd’s method, a fact that will also be supported by our numerical experiments in the next section where \mathcal{E}^H -clustering outperforms of \mathcal{E}^L -clustering (kernel k -means) in several settings, specially in high dimensions.

VII. NUMERICAL EXPERIMENTS

The main goal of this section is threefold. First, we want to compare energy clustering, based on the kernels provided by the standard energy distance \mathcal{E}_α , to k -means and GMM. Second, we want to compare \mathcal{E}^H -clustering, based on Hartigan’s method, to \mathcal{E}^L -clustering, based on Lloyd’s method. We also compare to \mathcal{E}^S -clustering which is an exact solution to the relaxed problem. Third, we want to illustrate the flexibility provided by energy clustering which do not make assumptions about the distribution of the data, being able to cluster accurately in different settings while keeping the same standard kernel.

In the experiments below we fix the semimetric according to the traditional energy dis-

tance, $\alpha = 1$, and the point $x_0 = 0$ is chosen in the corresponding kernel (8). We thus have

$$\rho(x, y) = \|x - y\| \quad \text{and} \quad K(x, y) = \frac{1}{2} (\|x\| + \|y\| - \|x - y\|). \quad (57)$$

We shall consider other kernels as well, but (57) will be the standard kernel for energy statistics and will always be present in the experiments as a reference.

We briefly mention that we compared \mathcal{E}^H -clustering, as described in Algorithm 4, to \mathcal{E}^{1D} -clustering, described in Algorithm 2, for several univariate distributions. Both perform almost indistinguishable regarding the clustering quality. However, we omit these results since we will analyse more interesting scenarios in high dimensions.

From the results of [17], summarized in the end of the previous section, we expect the improvement of Hartigan's over Lloyd's method to be more accentuated in high dimensions. Thus, we analyze how the algorithms degrade as the number of dimensions increase while keeping the number of points in each cluster fixed. Consider two clusters with multivariate normal distributions given by

$$x \in \mathcal{C}_i \sim \mathcal{N}(\mu_i, \Sigma_i) \quad (i = 1, 2),$$

$$\mu_1 = \underbrace{(0, \dots, 0)}_{\times D}^\top, \quad \mu_2 = 0.7 \times \underbrace{(1, \dots, 1)}_{\times 10} \underbrace{(0, \dots, 0)}_{\times (D-10)}^\top, \quad \Sigma_1 = \Sigma_2 = I_D. \quad (58)$$

Note that the Bayes error is fixed as D increases, yielding an optimal accuracy of ≈ 0.86 . For each D we generate 100 Monte Carlo runs, sampling 100 points for each cluster. We apply each algorithm to the resulting dataset and compute the accuracy (45). The algorithms \mathcal{E}^H - and \mathcal{E}^L -clustering both use the standard kernel (57), as well as the spectral \mathcal{E}^S -clustering method², and we initialize the algorithms with the initialization of k -means++ [28]. For k -means and GMM we use the implementation of *scikit-learn* library [27], where k -means is also initialized with k -means++, while GMM is initialized with the output of k -means. Every algorithm is ran 5 times, returning the answer with best objective function value. We plot the average accuracy versus D , with error bars indicating standard error. The results are shown in Fig. 2a. We can see that \mathcal{E}^H - and \mathcal{E}^S -clustering have practically the same performance, which is much higher than \mathcal{E}^L -clustering (kernel k -means). Moreover, \mathcal{E}^H -clustering outperforms k -means and GMM, where the improvement is noticeable specially

² We compared our implementation of spectral clustering provided in Algorithm 1 with the spectral clustering routine provided by *scikit-learn* library [27], which is based on [24]. In the few cases considered the algorithms had similar performance.

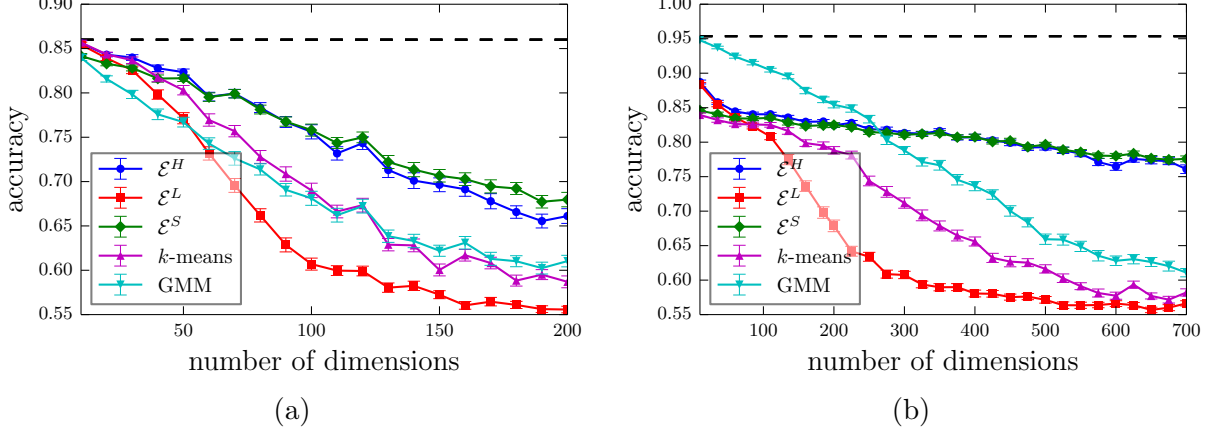


FIG. 2. Comparison of energy clustering algorithms to k -means and GMM in high dimensional Gaussian settings. We plot the mean accuracy versus the number of dimensions, with error bars indicating standard error from 100 Monte Carlo runs. (a) Data distributed as in (58), with Bayes accuracy ≈ 0.86 , over the range $D \in [10, 200]$. (b) Data distributed as in (59), with Bayes accuracy ≈ 0.95 , over the range $D \in [10, 700]$.

in high dimensions. Note that in this setting k -means and GMM are consistent estimators, however, energy clustering degrades much less as dimension increases.

We now allow the diagonal covariance of the second Gaussian distribution to have different values by choosing $(\Sigma_2)_{ii} \stackrel{iid}{\sim} \text{Uni}(1, 5)$ for $i = 1, \dots, 10$, and $(\Sigma_2)_{ii} = 1$ for $i = 11, \dots, D$. For instance, we consider the fixed setting

$$x \in \mathcal{C}_i \sim \mathcal{N}(\mu_i, \Sigma_i) \quad (i = 1, 2),$$

$$\mu_1 = \underbrace{(0, \dots, 0)}_{\times D}^\top, \quad \mu_2 = \underbrace{(1, \dots, 1)}_{\times 10} \underbrace{(0, \dots, 0)}_{\times (D-10)}^\top, \quad \Sigma_1 = I_D, \quad \Sigma_2 = \begin{pmatrix} \tilde{\Sigma}_{10} & 0 \\ 0 & I_{D-10} \end{pmatrix}, \quad (59)$$

$$\tilde{\Sigma}_{10} = \text{diag}(1.367, 3.175, 3.247, 4.403, 1.249, 1.969, 4.035, 4.237, 2.813, 3.637),$$

as we vary D . Note that there is nothing special about the numbers in $\tilde{\Sigma}_{10}$, it is simply a random choice of sampling $\text{Uni}(1, 5)$ ten times, and one could pick any other choices as well. The Bayes error is kept fixed yielding an optimal accuracy ≈ 0.95 . In Fig. 2b we show the results of energy clustering, k -means, and GMM. Note that GMM performs better in low dimensions, since it is a consistent estimator. However, \mathcal{E}^H and \mathcal{E}^S degrade much less as dimensions increase, performing better than k -means and GMM in high dimensions.

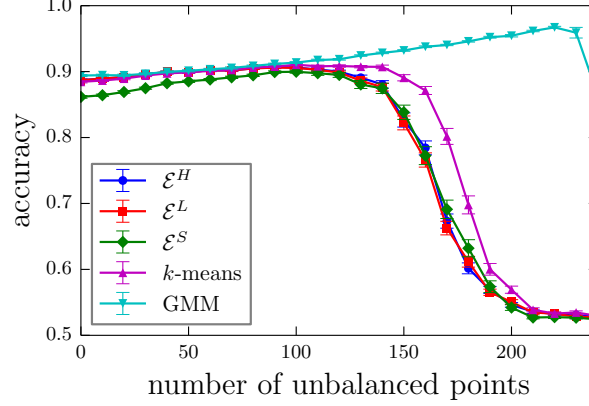


FIG. 3. Comparison of energy clustering algorithms to k -means and GMM on unbalanced clusters. The data is distributed as (60), where we vary $m \in [0, 240]$, and in each case we do 100 Monte Carlo runs showing the average accuracy with standard error.

Consider the effect of having unbalanced clusters according to

$$x \in \mathcal{C}_i \sim \frac{n_i}{N} \mathcal{N}(\mu_i, \Sigma_i) \quad (i = 1, 2), \quad \mu_1 = (0, 0, 0, 0)^\top, \quad \mu_2 = 1.5 \times (1, 1, 0, 0)^\top, \\ \Sigma_1 = I_4, \quad \Sigma_2 = \left(\begin{array}{c|c} \frac{1}{2}I_2 & 0 \\ \hline 0 & I_2 \end{array} \right), \quad n_1 = N - m, \quad n_2 = N + m, \quad N = 250. \quad (60)$$

We then increase $m \in [0, 240]$ making the clusters progressively more unbalanced. We plot the average accuracy over 100 Monte Carlo runs for each m , with error bars indicating standard error. The results are shown in Fig. 3. For highly unbalanced clusters we see that GMM performs better than the other methods, which have basically similar performance. Based on this experiment, an interesting problem would be to extend \mathcal{E}^H -clustering algorithm to account for highly unbalanced clusters.

Besides the standard kernel from energy statistics (57), consider the following two other semimetrics with their respective generating kernels:

$$\rho_{1/2}(x, y) = \|x - y\|^{1/2} \quad K_{1/2}(x, y) = \frac{1}{2} (\|x\|^{1/2} + \|y\|^{1/2} - \|x - y\|^{1/2}), \quad (61)$$

$$\rho_e(x, y) = 2 - 2e^{-\frac{1}{2}\|x-y\|} \quad K_e(x, y) = e^{-\frac{1}{2}\|x-y\|}. \quad (62)$$

The kernel $K_{1/2}(x, y)$ corresponds to the standard energy distance with $\alpha = 1/2$. We sample data from the following normal distribution in 20 dimensions:

$$x \in \mathcal{C}_i \sim \mathcal{N}(\mu_i, \Sigma_i) \quad (i = 1, 2), \\ \mu_1 = (\underbrace{0, \dots, 0}_{\times 20})^\top, \quad \mu_2 = \frac{1}{2}(\underbrace{1, \dots, 1}_5, \underbrace{0, \dots, 0}_{15})^\top, \quad \Sigma_1 = \frac{1}{2}I_{20}, \quad \Sigma_2 = I_{20}. \quad (63)$$

The optimal accuracy based on Bayes classification error is ≈ 0.90 . We sample an equal number of points for each cluster which is progressively increased, $n \in [5, 200]$. The results over 100 Monte Carlo trials are shown in Fig. 4a, where we compare \mathcal{E}^H -clustering, with different kernels, to k -means and GMM. Note that for small n all methods are superior than GMM, which slowly catches up and tend to optimal Bayes, as expected since it is a consistent estimator in this case. Note also that \mathcal{E}^H -clustering with kernel K_e is as accurate as GMM for large number of points, however, it is superior for small number of points. Still for the same setting, in Fig. 4b we show the difference in accuracy provided by \mathcal{E}^H minus \mathcal{E}^L and \mathcal{E}^S , when using the kernel K_e . Note that \mathcal{E}^H was always superior than the other methods (there would be points negative values on the y -axis otherwise).

Now consider the same parameters as in (63) but with lognormal distributions,

$$\log x \in \mathcal{C}_i \sim \mathcal{N}(\mu_i, \Sigma_i) \quad (i = 1, 2). \quad (64)$$

The same previous experiment is shown in Fig. 4c. Energy clustering still performs accurately, with any of the utilized kernels, providing better results than k -means and GMM on this non-normal data. The kernel K_e still provides the best results for small number of points, but its performance is eventually achieved by $K_{1/2}$, indicating that $\alpha \approx 1/2$ in the standard energy distance should be more appropriate for skewed distributions. Still for the same setting, in Fig. 4d we show the difference in accuracy provided by \mathcal{E}^H versus \mathcal{E}^L and \mathcal{E}^S , for the kernel K_e . Again, the accuracy provided by \mathcal{E}^H is higher than the other methods, although not much higher than \mathcal{E}^S in this example. The two experiments of Fig. 4 illustrate how energy statistics clustering does not make assumptions about the distribution of the data, performing well in different settings with the same kernel.

Consider the following choices of semimetric and corresponding generating kernel:

$$\rho_\alpha(x, y) = \|x - y\|^\alpha \quad K_\alpha(x, y) = \frac{1}{2} (\|x\|^\alpha + \|y\|^\alpha - \|x - y\|^\alpha), \quad (65)$$

$$\tilde{\rho}_\sigma(x, y) = 2 - 2e^{-\frac{\|x-y\|}{2\sigma}} \quad \tilde{K}_\sigma(x, y) = e^{-\frac{\|x-y\|}{2\sigma}}, \quad (66)$$

$$\hat{\rho}_\sigma(x, y) = 2 - 2e^{-\frac{\|x-y\|^2}{2\sigma^2}} \quad \hat{K}_\sigma(x, y) = e^{-\frac{\|x-y\|^2}{2\sigma^2}}. \quad (67)$$

In Fig. 5a–c we have complex two dimensional datasets. The two parallel cigars in (a) have 200 points each. The concentric circles in (b) and (c) have 400 points for each class. We apply \mathcal{E}^H -clustering with different choices of kernels in (65)–(67). We also consider the best

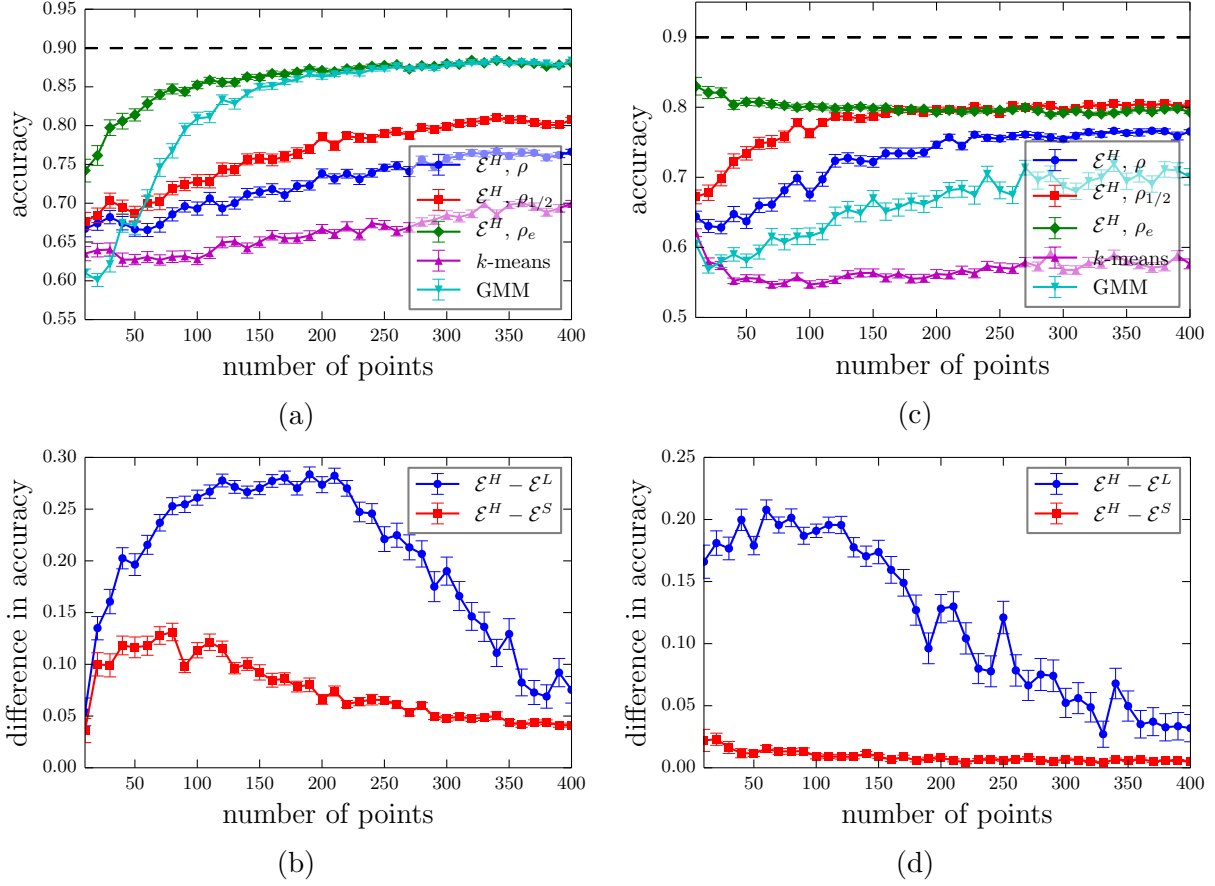


FIG. 4. \mathcal{E}^H -clustering with different kernels versus k -means and GMM. In both settings Bayes accuracy is ≈ 0.9 . We show average accuracy (error bars are standard error) versus number of points for 100 Monte Carlo trials. (a,b) Normal distribution (63). (c,d) Lognormal distribution (64). The plots in (c) and (d) consider the difference in accuracy between \mathcal{E}^H versus \mathcal{E}^L and \mathcal{E}^S -clustering, with semimetric ρ_e .

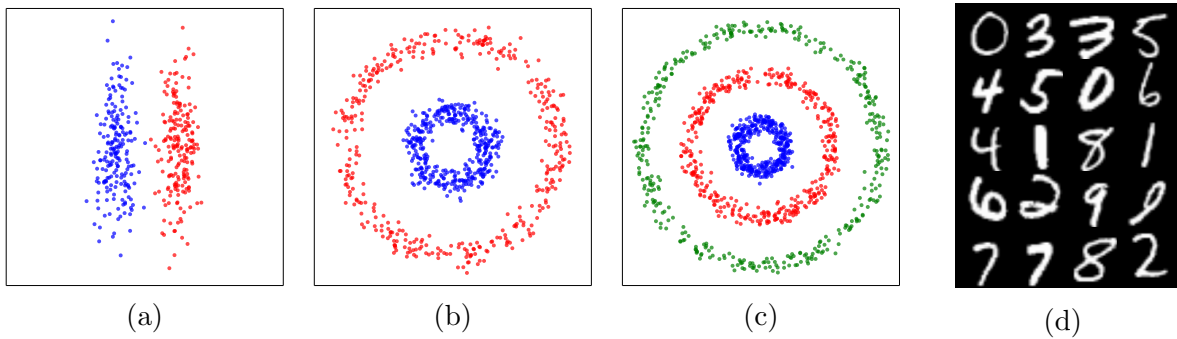


FIG. 5. (a) Parallel cigars. (b) Two concentric circles with noise. (c) Three concentric circles with noise. (d) MNIST handwritten digits. Clustering results are in Table I and Table II.

		<i>Fig. 5a</i>		<i>Fig. 5b</i>		<i>Fig. 5c</i>
\mathcal{E}^H	ρ_1	0.705 ± 0.065	ρ_1	0.521 ± 0.005	ρ_1	0.393 ± 0.020
	$\rho_{1/2}$	0.952 ± 0.048	$\rho_{1/2}$	0.522 ± 0.004	$\rho_{1/2}$	0.486 ± 0.040
	$\tilde{\rho}_2$	0.9987 ± 0.0008	$\tilde{\rho}_1$	0.778 ± 0.075	$\tilde{\rho}_2$	0.666 ± 0.007
	$\hat{\rho}_2$	0.956 ± 0.020	$\hat{\rho}_1$	1.0 ± 0.0	$\hat{\rho}_2$	0.676 ± 0.002
\mathcal{E}^S	$\tilde{\rho}_2$	0.557 ± 0.014	$\hat{\rho}_1$	0.732 ± 0.002	$\hat{\rho}_2$	0.364 ± 0.004
<i>k-means</i>		0.550 ± 0.011		0.522 ± 0.004		0.368 ± 0.005
<i>GMM</i>		0.903 ± 0.064		0.595 ± 0.011		0.465 ± 0.030

TABLE I. Clustering data from Fig. 5. We show average accuracy over 10 Monte Carlo runs.

<i>Class Subset</i>	$\{0, 1, 2\}$	$\{0, 1, \dots, 4\}$	$\{0, 1, \dots, 6\}$	$\{0, 1, \dots, 8\}$	
\mathcal{E}^H -clustering	ρ_1	0.937 ± 0.006	0.873 ± 0.025	0.731 ± 0.016	0.687 ± 0.016
	$\rho_{1/2}$	0.939 ± 0.006	0.874 ± 0.027	0.722 ± 0.017	0.647 ± 0.017
	$\tilde{\rho}_\sigma$	0.939 ± 0.006	0.847 ± 0.031	0.695 ± 0.023	0.657 ± 0.014
	$\hat{\rho}_\sigma$	0.933 ± 0.005	0.891 ± 0.009	0.759 ± 0.011	0.704 ± 0.011
\mathcal{E} -spectral	$\hat{\rho}_\sigma$	0.823 ± 0.015	0.769 ± 0.012	0.678 ± 0.014	0.649 ± 0.018
	σ	10.34	10.41	10.41	10.37
<i>k-means</i>		0.927 ± 0.004	0.878 ± 0.010	0.744 ± 0.008	0.695 ± 0.012
<i>GMM</i>		0.952 ± 0.005	0.839 ± 0.015	0.694 ± 0.010	0.621 ± 0.009

TABLE II. Clustering MNIST data from Fig. 5d. For each subset of digits we perform 10 Monte Carlo runs, sampling 100 points for each class.

kernel choice for each example for the spectral \mathcal{E}^S -clustering. Moreover, we consider *k*-means and GMM. We perform 10 Monte Carlo runs for each example. The results are in Table I. For (a) we initialize all algorithms with *k*-means++, and for (b) and (c) we initialize at random. \mathcal{E}^H has superior performance in every example, and in particular better than the spectral \mathcal{E}^S . For the data in Fig. 5a the semimetrics ρ_1 and $\rho_{1/2}$ are still able to provide accurate results, however, for the examples in (b) and (c) the kernel choice is more sensitive.

Next, we consider the infamous MNIST handwritten digits as illustrated in Fig. 5d. Each data point is an 8-bit gray scale image forming a 784-dimensional vector corresponding to the digits $\{0, 1, \dots, 9\}$. Besides the kernel (65), we consider the Gaussian kernel (67) with

$$\sigma^2 = \frac{1}{n^2} \sum_{i,j=1}^n \|x_i - x_j\|^2, \quad (68)$$

computed from a separate training set. We consider subsets of $\{0, 1, \dots, 9\}$, sampling 100 points for each class. The results are shown in Table II, where kernels and parameters

are indicated. \mathcal{E}^H -clustering performs slightly better than k -means and GMM, however the difference is not considerable. Unsupervised clustering on MNIST without any feature extraction is not trivial. For instance, the same experiment was performed in [29] where a low-rank transformation is learned then subsequently used in subspace clustering, providing very accurate results. It would be interesting to explore analogous methods for learning a better representation of the data and subsequently apply \mathcal{E}^H -clustering.

VIII. DISCUSSION

We considered clustering from the perspective of energy statistics, which provides a non-parametric test for equality of distributions. The mathematical formulation of energy clustering reduces to a QCQP in the associated RKHS, as demonstrated in Proposition 2. We showed that the optimization problem is equivalent to kernel k -means, once the kernel is fixed; see Proposition 3. Energy statistics, however, fixes a family of standard kernels in Euclidean space, and more general kernels on semimetric spaces of negative type can also be obtained. We also considered a weighted version of energy statistics, whose clustering formulation establishes connections with graph partitioning. We proposed the iterative \mathcal{E}^H -clustering algorithm based on Hartigan's method, which was compared to kernel k -means algorithm based on Lloyd's heuristic. Both have the same time complexity, however, numerical and theoretical results provide compelling evidence that \mathcal{E}^H -clustering is more robust with a superior performance, specially in high dimensions. Furthermore, energy clustering, with standard kernels from energy statistics, outperformed k -means and GMM on several settings, illustrating the flexibility of the proposed method which is distribution-free. In many settings, the iterative \mathcal{E}^H -clustering also surpassed \mathcal{E}^S -clustering, which is a spectral relaxation of the QCQP.

It would be interesting to formally demonstrate cases where energy clustering is a consistent estimator. A soft version of energy clustering is also an interesting extension. Finally, kernel methods can benefit from sparsity and fixed-rank approximations of the Gram matrix, and there is plenty of room to make \mathcal{E}^H -clustering algorithm more scalable.

ACKNOWLEDGMENTS

We would like to thank Carey Priebe for discussions. This work was supported by NIH TRA grant.

- [1] G. J. Székely and M. L. Rizzo. Energy Statistics: A Class of Statistics Based on Distances. *Journal of Statistical Planning and Inference*, 143:1249–1272, 2013.
- [2] M. L. Rizzo and G. J. Székely. DISCO Analysis: A Nonparametric Extension of Analysis of Variance. *The Annals of Applied Statistics*, 4(2):1034–1055, 2010.
- [3] G. J. Székely and M. L. Rizzo. Hierarchical Clustering via Joint Between-Within Distances: Extending Ward’s Minimum Variance Method. *Journal of Classification*, 22(2):151–183, 2005.
- [4] S. Li. k -Groups: A Generalization of k -Means by Energy Distance. PhD Thesis, Bowling Green State University, 2015.
- [5] R. Lyons. Distance Covariance in Metric Spaces. *The Annals of Probability*, 41(5):3284–3305, 2013.
- [6] D. Sejdinovic, B. Sriperumbudur, A. Gretton, and K. Fukumizu. Equivalence of Distance-Based and RKHS-Based Statistic in Hypothesis Testing. *The Annals of Statistics*, 41(5):2263–2291, 2013.
- [7] S. P. Lloyd. Least Squares Quantization in PCM. *IEEE Transactions on Information Theory*, 28(2):129–137, 1982.
- [8] J. B. MacQueen. Some Methods for Classification and Analysis of Multivariate Observations. In *Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability*, volume 1, pages 281–297. University of California Press, 1967.
- [9] E. Forgy. Cluster Analysis of Multivariate Data: Efficiency versus Interpretability of Classification. *Biometrics*, 21(3):768–769, 1965.
- [10] B. Schölkopf, A. J. Smola, and K. R. Müller. Nonlinear Component Analysis as a Kernel Eigenvalue Problem. *Neural Computation*, 10:1299–1319, 1998.
- [11] M. Girolami. Kernel Based Clustering in Feature Space. *Neural Networks*, 13(3):780–784, 2002.
- [12] J. Mercer. Functions of Positive and Negative Type and their Connection with the Theory of Integral Equations. *Proceedings of the Royal Society of London*, 209:415–446, 1909.

- [13] I. S. Dhillon, Y. Guan, and B. Kulis. Kernel K-means: Spectral Clustering and Normalized Cuts. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '04, pages 551–556, New York, NY, USA, 2004. ACM.
- [14] I. S. Dhillon, Y. Guan, and B. Kulis. Weighted Graph Cuts without Eigenvectors: A Multilevel Approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(11):1944–1957, 2007.
- [15] M. Filippone, F. Camastra, F. Masulli, and S. Rovetta. A Survey of Kernel and Spectral Methods for Clustering. *Pattern Recognition*, 41:176–190, 2008.
- [16] J. A. Hartigan and M. A. Wong. Algorithm AS 136: A k -Means Clustering Algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 28(1):100–108, 1979.
- [17] M. Telgarsky and A. Vattani. Hartigan’s Method: k -Means Clustering without Voronoi. In *Proceedings of the 13th International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 9, pages 313–319. JMLR, 2010.
- [18] N. Slonim, E. Aharoni, and K. Crammer. Hartigan’s k -Means versus Lloyd’s k -Means — Is it Time for a Change? In *Proceedings of the 20th International Conference on Artificial Intelligence*, pages 1677–1684. AAI Press, 2013.
- [19] N. Aronszajn. Theory of Reproducing Kernels. *Transactions of the American Mathematical Society*, 68(3):337–404, 1950.
- [20] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. Smola. A Kernel Two-Sample Test. *Journal of Machine Learning Research*, 13:723–773, 2012.
- [21] C. Berg, J. P. R. Christensen, and P. Ressel. *Harmonic Analysis on Semigroups: Theory of Positive Definite and Related Functions*. Graduate Text in Mathematics 100. Springer, New York, 1984.
- [22] A. Y. Ng, M. I. Jordan, and Y. Weiss. On Spectral Clustering: Analysis and an Algorithm. In *Advances in Neural Information Processing Systems*, volume 14, pages 849–856, Cambridge, MA, 2001. MIT Press.
- [23] B. Kernighan and S. Lin. An Efficient Heuristic Procedure for Partitioning Graphs. *The Bell System Technical Journal*, 49(2):291–307, 1970.
- [24] J. Shi and J. Malik. Normalized Cut and Image Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.
- [25] P. Chan, M. Schlag, and J. Zien. Spectral k -Way Ratio Cut Partitioning. *IEEE Transactions*

- on Computer-Aided Design of Integrated Circuits and Systems*, 13:1088–1096, 1994.
- [26] S. X. Yu and J. Shi. Multiclass Spectral Clustering. In *Proceedings Ninth IEEE International Conference on Computer Vision*, volume 1, pages 313–319, 2003.
- [27] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.
- [28] D. Arthur and S. Vassilvitskii. k -means++: The Advantage of Careful Seeding. In *Proceedings of the Eighteenth annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1027–1035, Philadelphia, PA, USA, 2007. Society for Industrial and Applied Mathematics.
- [29] Q. Qui and G. Sapiro. Learning Transformations for Clustering and Classification. *Journal of Machine Learning Research*, 16:187–225, 2015.