Kernel k-Groups via Hartigan's Method

Guilherme França, Maria L. Rizzo and Joshua T. Vogelstein

Abstract—Energy statistics was proposed by Székely in the 80's inspired by Newton's gravitational potential in classical mechanics, and it provides a model-free hypothesis test for equality of distributions. In its original form, energy statistics was formulated in Euclidean spaces. More recently, it was generalized to metric spaces of negative type. In this paper, we consider a formulation for the clustering problem using a weighted version of energy statistics in spaces of negative type. We show that this approach leads to a quadratically constrained quadratic program in the associated kernel space, establishing connections with graph partitioning problems and kernel methods in unsupervised machine learning. To find local solutions of such an optimization problem, we propose an extension of Hartigan's method to kernel spaces. Our method has the same computational cost as kernel k-means algorithm, which is based on Lloyd's heuristic, but our numerical results show an improved performance, especially in high dimensions.

Index Terms—Clustering, Energy Statistics, Kernel Methods.

1 Introduction

Energy Statistics [1], [2] is based on a notion of statistical potential energy between probability distributions, in close analogy to Newton's gravitational potential in classical mechanics. 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. Thus, it provides a model-free hypothesis test for equality of distributions which is achieved under minimum energy.

Energy statistics has been applied to several goodness-of-fit hypothesis tests, multi-sample tests of equality of distributions, analysis of variance [3], nonlinear dependence tests through distance covariance and distance correlation [4], which generalizes the Pearson correlation coefficient, and hierarchical clustering by extending Ward's method of minimum variance [5]; see [1], [2] for an overview of energy statistics and its applications. Moreover, in Euclidean spaces, an application of energy statistics to clustering was recently proposed [6] and the method was named *k-groups*.

In its original formulation, energy statistics has a compact representation in terms of expectations of pairwise Euclidean distances, providing straightforward empirical estimates. More recently, the notion of distance covariance was further generalized from Euclidean spaces to metric spaces of negative type [7]. Furthermore, the link between energy distance based tests and kernel based tests has been recently established [8] through an asymptotic equivalence between generalized energy distances and maximum mean discrepancies (MMD), which are distances between embeddings of distributions in reproducing kernel Hilbert spaces (RKHS). Even more recently, generalized energy distances and kernel methods have been demonstrated to be exactly equivalent, for all finite samples [9]. This equivalence immediately relates energy statistics to kernel methods often used in machine learning and form the basis of our approach in this paper.

Clustering is an important unsupervised learning problem and has a long history in statistics and machine learning, making it impossible to mention all important contributions in a short space. Perhaps, the most used method is k-means [10]–[12], which is based on Lloyd's heuristic [10] of iteratively computing the means of each cluster and then assigning points to the cluster with closest center. The only statistical information about each cluster comes from its mean, making the method 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 [13], [14]. A Mercer kernel [15] 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. A well-known kernel based clustering method is kernel k-means, which is precisely k-means formulated in the feature space [14]. Furthermore, kernel k-means algorithm [16], [17] is still based on Loyd's heuristic. We refer the reader to [18] for a survey of clustering methods.

Besides Lloyd's approach to clustering there is an old heuristic due to Hartigan [19], [20] that goes as follows: for each data point, simply assign it to a cluster in an optimal way such that a loss function is minimized. While Lloyd's method only iterates if some cluster contains a point that is closer to the mean of another cluster, Hartigan's method

- *G. França is with the Mathematical Institute for Data Science (MINDS), Johns Hopkins University.*
  E-mail: guifranca@jhu.edu
- *J. T. Vogelstein is with the Center for Imaging Science, the Department of Biomedical Engineering and Institute for Computational Medicine, Johns Hopkins University.*
  E-mail: jovo@jhu.edu
- *M. L. Rizzo is with the Department of Mathematics and Statistics, Bowling Green State University.*
  E-mail: mrizzo@bgsu.edu

Manuscript received August 14, 2018; revised August 14, 2018.

may iterate even if that is not the case, and moreover, it takes into account the motion of the means resulting from the reassignments. In this sense, Hartigan's method may potentially escape local minima of Lloyd's method. In the Euclidean case, this was shown to be the case [21]. Moreover, the advantages of Hartigan's over Lloyd's method was verified empirically [21], [22]. However, although it was observed to be as fast as Lloyd's method, no complexity analysis was provided.

Contributions

Although k-groups considers clustering from energy statistics in the particular Euclidean case [6], the precise optimization problem behind this approach remains obscure, as well as the connection with other methods in machine learning. The main theoretical contribution of this paper is to fill these gaps, which we do in more generality. For instance, our approach is not limited to the Euclidean case but holds for general arbitrary spaces of negative type. Moreover, we also consider a weighted version of energy statistics. Our approach reveals connections between energy statistics based clustering and existing methods such as kernel k-means and graph partitioning problems.

Another contribution of this paper is to extend Hartigan's method to kernel spaces. To the best of our knowledge, such an extension was not previously considered. Since this approach was motivated by energy statistics and [6] considered the Euclidean case, we call the proposed method *kernel k-groups*. We show that kernel k-groups has the same complexity as kernel k-means algorithm, however, our numerical results provide compelling evidence that kernel k-groups is more accurate and robust, especially in high dimensions.

Using the standard kernel defined by energy statistics, our experiments illustrate that kernel k-groups is able to perform accurately on data coming from very different distributions, contrary to k-means and GMM, for instance. More specifically, our method performs closely to k-means and GMM on normally distributed data, while it is significantly better on data that is not normally distributed. Its superiority in high dimensions is striking, being more accurate than k-means and GMM even in Gaussian settings. We also illustrate the advantages of kernel k-groups on real data.

2 Review of 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 to [1] and [7], [8].

Consider random variables in $\mathbb{R}^D$ such that $X, X' \overset{iid}{\sim} P$ and $Y, Y' \overset{iid}{\sim} Q$, 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* $\rho : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$, 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 $\mathbb{R}^D$ by $\mathcal{X}$ and $\|X-Y\|$ by $\rho(X,Y)$ 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 $\varphi : \mathcal{X} \to \mathcal{H}$ 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. Our energy clustering formulation, proposed in the next section, will be based on the generalized energy distance (4).

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 [8]. Let us first recall the definition of RKHS. Let $\mathcal{H}$ be a Hilbert space of real-valued functions over $\mathcal{X}$. A function $K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ 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. Defining the Gram matrix G with elements $G_{ij} = K(x_i, x_j)$, this is equivalent to $G = G^\top$ being positive semidefinite, i.e., $v^\top G v \geq 0$ for any vector $v \in \mathbb{R}^n$.

The Moore-Aronszajn theorem [23] establishes the converse of the above paragraph. For every symmetric and positive definite function $K : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$, 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

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

This can also be written as [24]

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

where $X, X' \overset{iid}{\sim} P$ and $Y, Y' \overset{iid}{\sim} Q$. From the equality between (5) and (6) we also have $\langle h_P, h_Q \rangle_{\mathcal{H}_K} = \mathbb{E}\, K(X,Y)$.

The following important result shows that semimetrics of negative type and symmetric positive definite kernels are closely related [25]. Let $\rho : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ and $x_0 \in \mathcal{X}$ an arbitrary but fixed point. Define

$$K(x,y) \equiv \tfrac{1}{2}\left[\rho(x,x_0) + \rho(y,x_0) - \rho(x,y)\right]. \tag{7}$$

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

$$\begin{aligned} \rho(x,y) &= K(x,x) + K(y,y) - 2K(x,y) \\ &= \|h_x - h_y\|^2_{\mathcal{H}_K} \end{aligned} \tag{8}$$

and the canonical feature map $\varphi : x \mapsto h_x$ is injective [8]. 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 the generalized energy distance (4) and inner products on RKHS, which is one of the main results of [8]. If ρ is a semimetric of negative type and K a kernel that generates ρ, then replacing (8) into (4), and using (6), yields

$$\begin{aligned} \mathcal{E}(P,Q) &= 2\left[\mathbb{E}\, K(X,X') + \mathbb{E}\, K(Y,Y') - 2\mathbb{E}\, K(X,Y)\right] \\ &= 2\gamma_K^2(P,Q). \end{aligned} \tag{9}$$

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 generalized energy statistics for the test statistic of equality of distributions [1]. Assume that we have data $\mathbb{X} = \{x_1, \ldots, x_n\}$, 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

$$g(\mathcal{C}_i, \mathcal{C}_j) \equiv \frac{1}{n_i n_j} \sum_{x \in \mathcal{C}_i} \sum_{y \in \mathcal{C}_j} \rho(x,y), \tag{10}$$

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

$$W \equiv \sum_{j=1}^k \frac{n_j}{2} g(\mathcal{C}_j, \mathcal{C}_j), \tag{11}$$

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

$$S \equiv \sum_{1 \le i < j \le k} \frac{n_i n_j}{2n} \left[2g(\mathcal{C}_i, \mathcal{C}_j) - g(\mathcal{C}_i, \mathcal{C}_i) - g(\mathcal{C}_j, \mathcal{C}_j)\right], \tag{12}$$

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, $n \to \infty$, under the null hypothesis $H_0 : P_1 = P_2 = \cdots = P_k$, we have that $S \to 0$, and under the alternative hypothesis $H_1 : P_i \neq P_j$ for at least two $i \neq j$, we have that $S \to \infty$.

3 The Clustering Problem Formulation

This section contains our main theoretical results. First, we generalize the previous formulas from energy statistics by introducing weights associated to data points. Second, we formulate an optimization problem for clustering in the associated RKHS, making connection with kernel methods in machine learning.

Let $w(x)$ be a weight function associated to point $x \in \mathcal{X}$ and define

$$g(\mathcal{C}_i, \mathcal{C}_j) \equiv \frac{1}{s_i s_j} \sum_{x \in \mathcal{C}_i} \sum_{y \in \mathcal{C}_j} w(x) w(y) \rho(x,y), \tag{13}$$

where

$$s_i \equiv \sum_{x \in \mathcal{C}_i} w(x), \qquad s \equiv \sum_{j=1}^k s_j. \tag{14}$$

The weighted version of the within energy dispersion and between-sample energy statistic are thus given by

$$W \equiv \sum_{j=1}^k \frac{s_j}{2} g(\mathcal{C}_j, \mathcal{C}_j), \tag{15}$$

$$S \equiv \sum_{1 \le i < j \le k} \frac{s_i s_j}{2s} \left[2g(\mathcal{C}_i, \mathcal{C}_j) - g(\mathcal{C}_i, \mathcal{C}_i) - g(\mathcal{C}_j, \mathcal{C}_j)\right]. \tag{16}$$

Note that if $w(x) = 1$ for every x we recover the previous formulas.

Due to the test statistic for equality of distributions, the obvious criterion for clustering data is to maximize S in (16), 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 result shows that maximizing S is, however, equivalent to minimizing W in (15).

Lemma 1. *Let $\mathbb{X} = \{x_1, \ldots, x_n\}$ where each data point x_i lives in a space $\mathcal{X}$ endowed with a semimetric $\rho : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ of negative type. For a fixed integer k, the partition $\mathbb{X} = \bigcup_{j=1}^k \mathcal{C}_j^\star$, where $\mathcal{C}_i^\star \cap C_j^\star = \emptyset$ for all $i \neq j$, maximizes the between-sample statistic S, defined in equation* (16)*, if and only if*

$$\{\mathcal{C}_1^\star, \ldots, \mathcal{C}_k^\star\} = \underset{\mathcal{C}_1, \ldots, C_k}{\arg\min}\, W(\mathcal{C}_1, \ldots, \mathcal{C}_k), \tag{17}$$

where the within energy dispersion W is defined by (15)*.*

Proof. From (15) and (16) we have that

$$\begin{aligned} &S + W \\ &= \frac{1}{2s} \sum_{\substack{i,j=1 \\ i \neq j}}^{k} s_i s_j g(\mathcal{C}_i, \mathcal{C}_j) + \frac{1}{2s} \sum_{i=1}^{k} \left[s - \sum_{\substack{j=1 \\ j \neq i}}^{k} s_j \right] s_i g(\mathcal{C}_i, \mathcal{C}_i) \\ &= \frac{1}{2s} \sum_{i,j=1}^{k} s_i s_j g(\mathcal{C}_i, \mathcal{C}_j) \\ &= \frac{1}{2s} \sum_{x \in \mathbb{X}} \sum_{y \in \mathbb{X}} w(x) w(y) \rho(x, y) \\ &= \frac{s}{2} g(\mathbb{X}, \mathbb{X}). \end{aligned} \tag{18}$$

Since $g(\mathbb{X}, \mathbb{X})$ is independent of the choice of partition, $\max_{\{\mathcal{C}_i\}} S = -\max_{\{\mathcal{C}_i\}} W = \min_{\{\mathcal{C}_i\}} W$, as claimed. □

For a given k, the clustering problem amounts to finding the best partitioning of the data by minimizing W. In the current form of problem (17), the relationship with other clustering methods or kernel spaces is totally obscure. In the following, we demonstrate what is the explicit optimization problem behind (17) in the corresponding RKHS, which establishes the connection with kernel methods.

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

$$G \equiv \begin{pmatrix} K(x_1, x_1) & \cdots & K(x_1, x_n) \\ \vdots & \ddots & \vdots \\ K(x_n, x_1) & \cdots & K(x_n, x_n) \end{pmatrix}. \tag{19}$$

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 $D = \operatorname{diag}(n_1, \ldots, n_k)$ contains the number of points in each cluster. We also introduce the rescaled matrix Y below. In component form they are

$$Z_{ij} \equiv \begin{cases} 1 & \text{if } x_i \in \mathcal{C}_j \\ 0 & \text{otherwise} \end{cases}, \qquad Y_{ij} \equiv \begin{cases} \frac{1}{\sqrt{s_j}} & \text{if } x_i \in \mathcal{C}_j \\ 0 & \text{otherwise} \end{cases}. \tag{20}$$

Throughout the paper, we use the notation $M_{i\bullet}$ to denote the ith row of a matrix M, and $M_{\bullet j}$ denotes its jth column. We also define the following:

$$\mathcal{W} \equiv \operatorname{diag}(w_1, \ldots, w_n), \quad H \equiv \mathcal{W}^{1/2} Y, \quad \omega \equiv \mathcal{W} e, \tag{21}$$

where $w_i = w(x_i)$ is the weight associated to point x_i, and $e = (1, \ldots, 1)^\top \in \mathbb{R}^n$ is the all-ones vector.

Our next result shows that the optimization problem (17) is NP-hard since it is a quadratically constrained quadratic program (QCQP) in the associated RKHS.

Theorem 2. *The optimization problem* (17) *is equivalent to*

$$\begin{aligned} &\max_{H} \operatorname{Tr}\left[H^\top \left(\mathcal{W}^{1/2} G \mathcal{W}^{1/2} \right) H \right] \\ &\text{such that } H \geq 0,\ H^\top H = I,\ H H^\top \omega = \omega, \end{aligned} \tag{22}$$

where G is the Gram matrix (19) *and the other quantities are defined in* (21).

Proof. From (8), (13), and (15) we have

$$\begin{aligned} W &= \sum_{j=1}^{k} \frac{1}{2 s_j} \sum_{x,y \in \mathcal{C}_j} w(x) w(y) \rho(x, y) \\ &= \sum_{j=1}^{k} \sum_{x \in \mathcal{C}_j} \left[w(x) K(x, x) - \frac{1}{s_j} \sum_{y \in \mathcal{C}_j} w(x) w(y) K(x, y) \right]. \end{aligned} \tag{23}$$

Note that the first term is global so it does not contribute to the optimization problem. Therefore, problem (17) becomes

$$\max_{\mathcal{C}_1, \ldots, \mathcal{C}_k} \sum_{j=1}^{k} \frac{1}{s_j} \sum_{x,y \in C_j} w(x) w(y) K(x, y). \tag{24}$$

Using the definitions (20) and (21), the previous objective function can be written as

$$\begin{aligned} &\sum_{j=1}^{k} \frac{1}{s_j} \sum_{p=1}^{n} \sum_{q=1}^{n} w_p w_q Z_{pj} Z_{qj} G_{pq} \\ &\quad = \sum_{j=1}^{k} \sum_{p=1}^{n} \sum_{q=1}^{n} \frac{Z_{jp}^\top \sqrt{w_p}}{\sqrt{s_j}} w_p^{1/2} G_{pq} w_q^{1/2} \frac{\sqrt{w_q} Z_{qj}}{\sqrt{s_j}} \\ &\quad = \sum_{j=1}^{k} \left(H^\top \mathcal{W}^{1/2} G \mathcal{W}^{1/2} H \right)_{jj} \\ &\quad = \operatorname{Tr}\left[H^\top \mathcal{W}^{1/2} G \mathcal{W}^{1/2} H \right]. \end{aligned} \tag{25}$$

Now it remains to obtain the constraints. Note that $H_{ij} \geq 0$ by definition, and

$$\begin{aligned} (H^\top H)_{ij} &= \sum_{\ell=1}^{n} Y_{\ell i} \mathcal{W}_{\ell\ell} Y_{\ell j} \\ &= \frac{1}{\sqrt{s_i} \sqrt{s_j}} \sum_{\ell=1}^{n} w_\ell Z_{\ell i} Z_{\ell j} \\ &= \frac{\delta_{ij}}{s_i} \sum_{\ell=1}^{n} w_\ell Z_{\ell i} \\ &= \delta_{ij} \end{aligned} \tag{26}$$

where $\delta_{ij} = 1$ if $i = j$ and $\delta_{ij} = 0$ if $i \neq j$ is the Kronecker delta. Therefore, $H^\top H = I$. This is a constraint on the rows of H. To obtain a constraint on its columns, observe that

$$\begin{aligned} \left(H^\top H \right)_{pq} &= \sqrt{w_p w_q} \sum_{j=1}^{k} \frac{Z_{pj} Z_{qj}}{s_j} \\ &= \begin{cases} \dfrac{\sqrt{w_p w_q}}{s_i} & \text{if both } x_p, x_q \in \mathcal{C}_i, \\ 0 & \text{otherwise.} \end{cases} \end{aligned} \tag{27}$$

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 $i \in \{1, \ldots, k\}$, and $(H^\top H \mathcal{W}^{1/2})_{pq} = 0$ otherwise. Thus, the pth 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

$$H H^\top \mathcal{W}^{1/2} e = \mathcal{W}^{1/2} e. \tag{28}$$

From (21) this gives $H H^\top \omega = \omega$, finishing the proof. □

The optimization problem (22) is nonconvex, besides being NP-hard, thus a direct approach is computationally prohibitive even for small datasets. However, one can find approximate solutions by relaxing some of the constraints. For instance, consider the relaxed problem

$$\max_{H} \operatorname{Tr}\left[H^\top \widetilde{G}\, H\right] \qquad \text{such that } H^\top H = I, \tag{29}$$

where $\widetilde{G} \equiv \mathcal{W}^{1/2}\, G\, \mathcal{W}^{1/2}$. This problem has a well-known closed form solution $H^\star = UR$, where the columns of $U \in \mathbb{R}^{n\times k}$ contain the top k eigenvectors of $\widetilde{G}$ corresponding to the k largest eigenvalues, $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_k$, and $R \in \mathbb{R}^{k\times k}$ is an arbitrary orthogonal matrix. The resulting optimal objective function assumes the value $\max \operatorname{Tr}\left[H^{\star\top} \widetilde{G}\, H^\star\right] = \sum_{i=1}^k \lambda_i$. Spectral clustering is based on this approach, where one further normalize the rows of $H^\star$, then cluster the resulting rows as data points using any clustering method such as k-means. A procedure on these lines was proposed in the seminal papers [26], [27].

3.1 Connection with Graph Partitioning

We now show how graph partitioning problems are related to the energy statistics formulation leading to problem (22).

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 which 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

$$\operatorname{links}(\mathcal{C}_\ell, \mathcal{C}_m) \equiv \sum_{\substack{i\in\mathcal{C}_\ell \\ j\in\mathcal{C}_m}} A_{ij}. \tag{30}$$

Our goal is 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

$$\max_{\mathcal{C}_i,\ldots,\mathcal{C}_k} \sum_{j=1}^k \frac{\operatorname{links}(\mathcal{C}_j, \mathcal{C}_j)}{s_j} \tag{31}$$

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

$$\min_{\mathcal{C}_i,\ldots,\mathcal{C}_k} \sum_{j=1}^k \frac{\operatorname{links}(\mathcal{C}_j, \mathcal{V}\setminus\mathcal{C}_j)}{s_j} \tag{32}$$

minimizes the cut between clusters. Both problems (31) and (32) are equivalent, in analogous way as minimizing (15) is equivalent to maximizing (16), as shown in Lemma 1. Here, this equivalence is a consequence of the equality $\operatorname{links}(\mathcal{C}_j, \mathcal{V}\setminus\mathcal{C}_j) = \operatorname{links}(\mathcal{C}_j, \mathcal{V}) - \operatorname{links}(\mathcal{C}_j, \mathcal{C}_j)$. Several graph partitioning methods [26], [28]–[30] can be seen as a particular case of problems (31) or (32).

Consider the ratio association problem (31), whose objective function can be written as

$$\begin{aligned}\sum_{j=1}^k \frac{1}{s_j} \sum_{p\in\mathcal{C}_j}\sum_{q\in\mathcal{C}_j} \mathcal{A}_{pq} &= \sum_{j=1}^k \sum_{p=1}^n \sum_{q=1}^n \frac{Z_{jp}^\top}{\sqrt{s_j}}\, \mathcal{A}_{pq}\, \frac{Z_{qj}}{\sqrt{s_j}} \\ &= \operatorname{Tr}\left[Y^\top \mathcal{A}\, Y\right],\end{aligned} \tag{33}$$

where we recall that Z is defined in (20) and Y is defined in (21). Therefore, the ratio association problem can be written in the form (22), i.e.,

$$\begin{aligned}&\max_{H} \operatorname{Tr}\left[H^\top \mathcal{W}^{-1/2} \mathcal{A}\, \mathcal{W}^{-1/2} H\right] \\ &\text{such that } H \geq 0,\, H^\top H = I,\, HH^\top \omega = \omega.\end{aligned} \tag{34}$$

This is exactly the same as (22) with $G = \mathcal{W}^{-1}\mathcal{A}\mathcal{W}^{-1}$. Assuming that this matrix is positive semidefinite, this generates a semimetric (8) for graphs given by

$$\rho(i,j) = \frac{\mathcal{A}_{ii}}{w_i^2} + \frac{\mathcal{A}_{jj}}{w_j^2} - \frac{2\mathcal{A}_{ij}}{w_i w_j} \tag{35}$$

for vertices $i, j \in \mathcal{V}$. If we assume the graph has no self-loops we must replace $\mathcal{A}_{ii} = 0$ above. The weight of node $i \in \mathcal{V}$ can be, for instance, its degree $w_i = w(i) = d(i)$.

3.2 Connection with Kernel k-Means

We now show that kernel k-means optimization problem [16], [17] is also related to the previous energy statistics formulation to clustering. To be precise, we consider a weighted generalization of kernel k-means.

For a positive semidefinite Gram matrix G, as defined in (19), there exists a map $\varphi : \mathcal{X} \to \mathcal{H}_K$ such that

$$K(x,y) = \langle \varphi(x), \varphi(y)\rangle. \tag{36}$$

Define the weighted mean of cluster $\mathcal{C}_j$ as

$$\mu_j = \frac{1}{s_j} \sum_{x\in\mathcal{C}_j} w(x)x. \tag{37}$$

Disregarding the first global term in (23), note that the second term, $-\frac{1}{s_j}\sum_{x,y\in\mathcal{C}_j} w(x)w(y)K(x,y)$, is equal to

$$\begin{aligned}&\frac{1}{s_j^2} \sum_{x,y,z\in\mathcal{C}_j} \langle w(y)\varphi(y), w(z)\varphi(z)\rangle \\ &\qquad - \frac{2}{s_j} \sum_{x,y\in\mathcal{C}_j} \langle w(x)\varphi(x), w(y)\varphi(y)\rangle,\end{aligned} \tag{38}$$

which using (37) becomes

$$\begin{aligned}&\sum_{x\in\mathcal{C}_j} \left\{\langle\varphi(\mu_j),\varphi(\mu_j)\rangle - 2\langle w(x)\varphi(x), \varphi(\mu_j)\rangle\right\} \\ &\quad = \sum_{x\in\mathcal{C}_j} \left\{\|w(x)\varphi(x) - \varphi(\mu_j)\|^2 - \|w(x)\varphi(x)\|^2\right\}.\end{aligned} \tag{39}$$

Therefore, minimizing W in (23) is equivalent to

$$\min_{\mathcal{C}_1,\ldots,\mathcal{C}_k} \left\{ J(\{\mathcal{C}_j\}) \equiv \sum_{j=1}^k \sum_{x\in\mathcal{C}_j} \|w(x)\varphi(x) - \varphi(\mu_j)\|^2 \right\}. \tag{40}$$

Problem (40) is obviously equivalent to problem (22). When $w(x) = 1$ for all x, problem (40) corresponds to kernel k-means problem [16], [17]. Thus, the result (40) shows that the previous energy statistics formulation to clustering is equivalent to a weighted modification of kernel k-means[1]. One must note, however, that energy statistics fixes the kernel through (7).

1. One should not confuse kernel k-means optimization problem, given by (40), with kernel k-means *algorithm*. We will discuss two approaches to solve (40), or equivalently (22). One is based on Lloyd's method, which leads to kernel k-means algorithm, and the other is based on Hartigan's method, which leads to a new algorithm.

4 ITERATIVE ALGORITHMS

In this section, we introduce two iterative algorithms to solve the optimization problem (22). The first is based on Lloyd's method, while the second is based on Hartigan's method.

Consider the optimization problem (24) written as

$$\max_{\{\mathcal{C}_1,\ldots,\mathcal{C}_k\}} \left\{ Q = \sum_{j=1}^{k} \frac{Q_j}{s_j} \right\},\ Q_j \equiv \sum_{x,y\in\mathcal{C}_j} w(x)w(y)K(x,y), \quad (41)$$

where Q_j represents an internal cost of cluster $\mathcal{C}_j$, and Q is the total cost where each Q_j is weighted by the inverse of the sum of weights of the points in $\mathcal{C}_j$. For a data point x_i, we denote its cost with cluster $\mathcal{C}_\ell$ by

$$Q_\ell(x_i) \equiv \sum_{y\in\mathcal{C}_\ell} w(x_i)w(y)K(x_i,y) = (\mathcal{W}G\mathcal{W})_{i\bullet}\cdot Z_{\bullet\ell}, \quad (42)$$

where we recall that $M_{i\bullet}$ ($M_{\bullet i}$) denotes the ith row (column) of matrix M.

4.1 Weighted Kernel k-Means Algorithm

Using the definitions (41) and (42), the the optimization problem (40) can be written as

$$\min_{Z} \sum_{i=1}^{n}\sum_{\ell=1}^{k} Z_{i\ell} J^{(\ell)}(x_i) \quad (43)$$

where

$$J^{(\ell)}(x_i) \equiv \frac{1}{s_\ell^2} Q_\ell - \frac{2}{s_\ell} Q_\ell(x_i). \quad (44)$$

A possible strategy to solve (43) is to assign x_i to cluster $\mathcal{C}_{j^\star}$ according to

$$j^\star = \arg\min_{\ell=1,\ldots,k} J^{(\ell)}(x_i). \quad (45)$$

This should be 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 1. It can be shown that this algorithm converges when G is positive semidefinite.

To see that the above procedure is indeed kernel k-means [16], [17], based on Lloyd's heuristic [10], note that from (40) and (44) we have

$$\min_{\ell} J^{(\ell)}(x_i) = \min_{\ell} \|w(x_i)\varphi(x_i) - \varphi(\mu_\ell)\|^2. \quad (46)$$

Therefore, we are assigning x_i to the cluster with closest center, in the feature space. When $w(x) = 1$ for all x, the above method is exactly kernel k-means algorithm.

To check the complexity of Algorithm 1, note that the second term in (44) 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). Therefore, the time complexity of Algorithm 1 is $\mathcal{O}(nk\max_\ell n_\ell) = \mathcal{O}(kn^2)$. For a sparse Gram matrix G, having $\tilde{n}$ nonzero elements, this can be further reduced to $\mathcal{O}(k\tilde{n})$.

Algorithm 1 Weighted version of kernel k-means algorithm to find local solutions to the optimization problem (22).

input $k, G, \mathcal{W}, Z \leftarrow Z_0$
output Z
1: $q \leftarrow (Q_1,\ldots,Q_k)^\top$ (see (41))
2: $s \leftarrow (s_1,\ldots,s_k)^\top$
3: **repeat**
4: **for** $i = 1,\ldots,n$ **do**
5: let j be such that $x_i \in \mathcal{C}_j$
6: $j^\star \leftarrow \arg\min_{\ell=1,\ldots,k} J^{(\ell)}(x_i)$ (see (44))
7: **if** $j^\star \neq j$ **then**
8: $Z_{ij} \leftarrow 0$
9: $Z_{ij^\star} \leftarrow 1$
10: $s_j \leftarrow s_j - \mathcal{W}_{ii}$
11: $s_{j^\star} \leftarrow s_{j^\star} + \mathcal{W}_{ii}$
12: $q_j \leftarrow q_j - 2Q_j(x_i)$ (see (42))
13: $q_{j^\star} \leftarrow q_{j^\star} + 2Q_{j^\star}(x_i)$ (see (42))
14: **end if**
15: **end for**
16: **until** convergence

Algorithm 2 Kernel k-groups algorithm, based on Hartigan's method, to find local solutions to problem (22).

input $k, G, \mathcal{W}, Z \leftarrow Z_0$
output Z
1: $q \leftarrow (Q_1,\ldots,Q_k)^\top$ (see (41))
2: $s \leftarrow (s_1,\ldots,s_k)^\top$
3: **repeat**
4: **for** $i = 1,\ldots,n$ **do**
5: let j be such that $x_i \in \mathcal{C}_j$
6: $j^\star \leftarrow \arg\max_{\ell=1,\ldots,k\,|\,\ell\neq j} \Delta Q^{j\to\ell}(x_i)$ (see (50))
7: **if** $\Delta Q^{j\to j^\star}(x_i) > 0$ **then**
8: $Z_{ij} \leftarrow 0$
9: $Z_{ij^\star} \leftarrow 1$
10: $s_j \leftarrow s_j - \mathcal{W}_{ii}$
11: $s_{j^\star} \leftarrow s_{j^\star} + \mathcal{W}_{ii}$
12: $q_j \leftarrow q_j - 2Q_j(x_i) + (\mathcal{W}G\mathcal{W})_{ii}$ (see (42))
13: $q_{j^\star} \leftarrow q_{j^\star} + 2Q_{j^\star}(x_i) + (\mathcal{W}G\mathcal{W})_{ii}$ (see (42))
14: **end if**
15: **end for**
16: **until** convergence

4.2 Kernel k-Groups Algorithm

We now consider Hartigan's method [19], [20] applied to the optimization problem in the form (41), which gives a local solution to (22). The method is based in computing the maximum change in the total cost function Q when moving each data point to another cluster. More specifically, suppose that 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 $\mathcal{C}_\ell$ yields another total cost function denoted by $Q^{(\ell)}$. We are interested in computing the maximum change $\Delta Q^{(j\to\ell)}(x_i) \equiv Q^{(\ell)} - Q^{(j)}$, for $\ell \neq j$. From (41), by explicitly writing the costs related to these two cluster we obtain

$$\Delta Q^{(j\to\ell)}(x_i) = \frac{Q_\ell^+}{s_\ell + w_i} + \frac{Q_j^-}{s_j - w_i} - \frac{Q_\ell}{s_\ell} - \frac{Q_j}{s_j}, \quad (47)$$

where Q_ℓ^+ denote the cost of the new ℓth cluster with the

point x_i added to it, and Q_j^- is the cost of new jth cluster with x_i removed from it. Recall also that $w_i = w(x_i)$ is the weight associated to point x_i. Noting that

$$Q_\ell^+ = Q_\ell + 2Q_\ell(x_i) + (\mathcal{W}G\mathcal{W})_{ii}, \tag{48}$$
$$Q_j^- = Q_j - 2Q_j(x_i) + (\mathcal{W}G\mathcal{W})_{ii}, \tag{49}$$

we obtain

$$\Delta Q^{(j\to\ell)}(x_i) = \frac{1}{s_j - w_i}\left[\frac{w_i}{s_j}Q_j - 2Q_j(x_i) + (\mathcal{W}G\mathcal{W})_{ii}\right] - \frac{1}{s_\ell + w_i}\left[\frac{w_i}{s_\ell}Q_\ell - 2Q_\ell(x_i) - (\mathcal{W}G\mathcal{W})_{ii}\right]. \tag{50}$$

Therefore, we compute

$$j^\star = \underset{\ell=1,\ldots,k \,|\, \ell\neq j}{\arg\max}\ \Delta Q^{(j\to\ell)}(x_i) \tag{51}$$

and if $\Delta Q^{j\to j^\star}(x_i) > 0$ we move x_i to cluster $\mathcal{C}_{j^\star}$, 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 described in Algorithm 2, which we call kernel k-groups. This method is a generalization of the k-groups with first variations proposed in [6], which only considers the Euclidean case, and also an extension of Hartigan's method to kernel spaces.

Note that Algorithm 2 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 Algorithm 2 is the following. The computation of each cluster cost Q_j has complexity $\mathcal{O}(n_j^2)$, and overall to compute q we have $\mathcal{O}(n_1^2+\cdots+n_k^2) = \mathcal{O}(k\max_j n_j^2)$. 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 2 is $\mathcal{O}(k\max_\ell n_\ell)$ for $\ell = 1,\ldots,k$. For the entire dataset this gives a complexity of $\mathcal{O}(nk\max_\ell n_\ell) = \mathcal{O}(kn^2)$. Note that this is the same cost as in kernel k-means algorithm. Again, if G is sparse this can be reduced to $\mathcal{O}(k\tilde{n})$ where $\tilde{n}$ is the number of nonzero entries of G.

5 Numerical Experiments

The main goal of this section is twofold. First, to illustrate that in Euclidean spaces with the standard metric of energy statistics, as defined by the energy distance (1), the clustering method implemented by kernel k-groups is more flexible and in general more accurate than k-means and GMM. Second, we want to compare kernel k-groups with kernel k-means and spectral clustering when these methods operate on the same kernel.

We consider the metrics

$$\rho_\alpha(x,y) = \|x-y\|^\alpha, \tag{52}$$
$$\widetilde{\rho}_\sigma(x,y) = 2 - 2e^{-\frac{\|x-y\|}{2\sigma}}, \tag{53}$$
$$\widehat{\rho}_\sigma(x,y) = 2 - 2e^{-\frac{\|x-y\|^2}{2\sigma^2}}, \tag{54}$$

which define the corresponding kernels through (7), where we always fix $x_0 = 0$. We use ρ_α by default, unless otherwise specified. We consider the weights associated to data points to be $w(x) = 1$ for all x, so that $\mathcal{W} = I$ in Algorithms 1 and 2. For k-means, GMM and spectral clustering we use the implementations of *scikit-learn* library [31], where k-means is initialized with k-means++ [32], and GMM is initialized with the output of k-means, making it more robust and preventing it from breaking in high dimensions. The spectral clustering implementation of scikit-learn is based on [26]. Kernel k-means is implemented according to Algorithm 1 while kernel k-groups follows Algorithm 2. Both will also be initialized with k-means++, unless specified otherwise. We run every algorithms 5 times with different initializations and then we choose the result with the best objective function value. We evaluate the clustering quality by the accuracy defined as

$$\text{accuracy}(\hat{Z}) \equiv \max_\pi \frac{1}{n}\sum_{i=1}^n\sum_{j=1}^k \hat{Z}_{i\pi(j)}Z_{ij}, \tag{55}$$

where $\hat{Z}$ is the predicted label matrix, Z is the ground truth label matrix, and π is a permutation of $\{1,2,\ldots,k\}$. Thus, the accuracy corresponds to the fraction of correctly classified data points, and it is always between $[0,1]$. For each setting, we show the average accuracy over 100 Monte Carlo trials (we ommit the error bars since they are too small to be visible in our experiments).

5.1 Synthetic Experiments

We first consider one-dimensional data for a two-class problem. We compare kernel k-groups with k-means and GMM, as illustrated in Fig. 1. The left panels show a mixture of Gaussians, and the right panels show a mixture of log Gaussians (see caption for details). Notice that in the kernel density estimation plots for lognormal distribution, only kernel k-groups was able to distinguish between the two classes. The accuracy results for both density estimation cases are in Table 1. We remark that kernel k-groups in this one-dimensional example performed the same as the exact deterministic Algorithm 3 introduced in the Appendix.

TABLE 1
Accuracy results for the density estimation of Fig. 1d–e.

Method	normal	lognormal
kmeans	0.778	0.520
GMM	**0.887**	0.542
kernel k-groups	0.807	**0.846**

Next, we analyze how the algorithms degrade as the number of dimensions increase. Consider data from the Gaussian mixture

$$x \overset{iid}{\sim} \tfrac{1}{2}\mathcal{N}(\mu_1,\Sigma_1) + \tfrac{1}{2}\mathcal{N}(\mu_2,\Sigma_2), \quad \Sigma_1 = \Sigma_2 = I_D,$$
$$\mu_1 = (\underbrace{0,\ldots,0}_{\times D})^\top, \qquad \mu_2 = 0.7(\underbrace{1,\ldots,1}_{\times 10},\underbrace{0,\ldots,0}_{\times(D-10)})^\top. \tag{56}$$

The Bayes error is fixed as D increases giving an optimal accuracy of ≈ 0.86. We sample 200 points on each trial. A scatter plot of the last two dimensions that contains signal in μ_2 is shown in Fig. 2a. The clustering results are shown in Fig. 3a. We see that kernel k-groups and spectral clustering

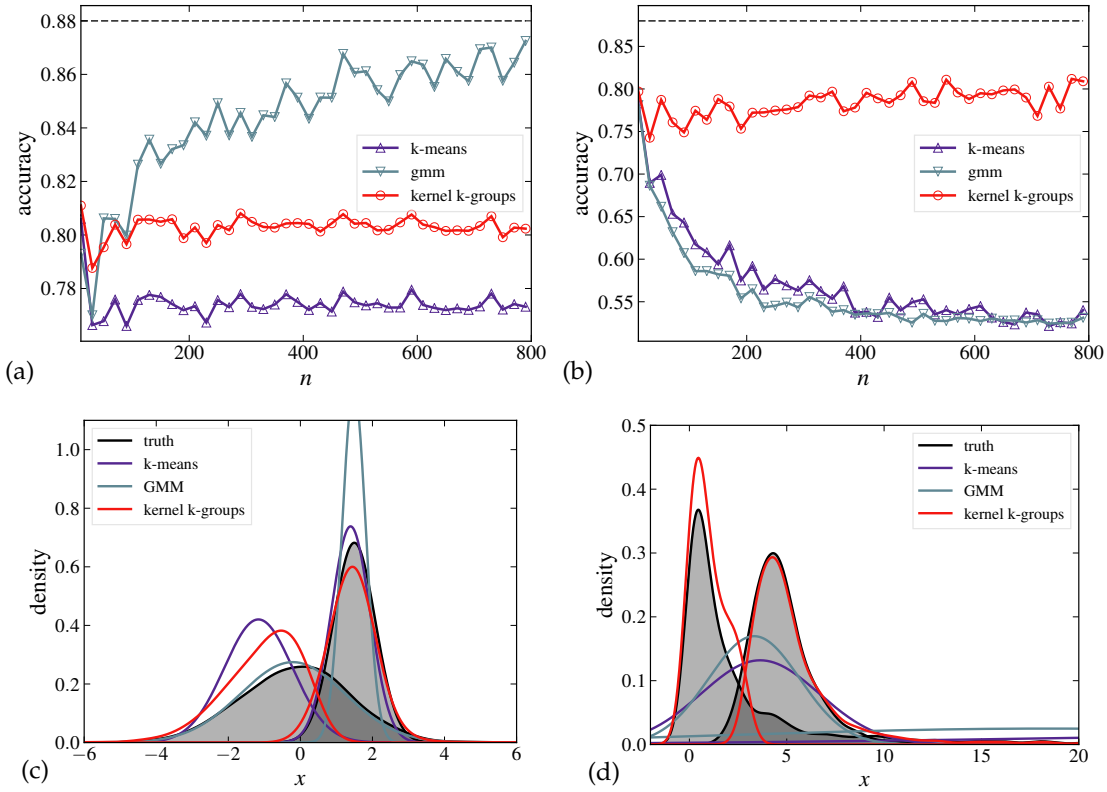

Fig. 1. Clustering one-dimensional data for a two-class problem. (a) Data normally distributed as $x \overset{iid}{\sim} (1/2)\mathcal{N}(0, 1.5) + (1/2)\mathcal{N}(1.5, 0.3)$. (b) Data following lognormal distributions as $x \overset{iid}{\sim} (1/2)e^{\mathcal{N}(0,1.5)} + (1/2)e^{\mathcal{N}(1.5,0.3)}$. In both cases we plot the average accuracy versus the total number of points (error bars are too small to be visible). (c) For the same distribution as in item (a), we sample 2000 points, cluster them with the three methods, then perform a kernel density estimation for kernel k-groups, since this is a model-free method, while for k-means and GMM we show the estimated Gaussian for each class. The clustering accuracy for each method is in Table 1. (d) Exactly the same experiment but for the distribution in item (b). Note that only kernel k-groups is able to distinguish between the two classes in this example.

have close performance, being superior to kernel k-means, k-means, and GMM. The improvement is noticeable in higher dimensions.

Still for a two-class Gaussian mixture as in (56), we now choose different numbers for the diagonal covariance Σ_2. We have $\Sigma_1 = I_D$, $\mu_1 = (0, \dots, 0)^\top \in \mathbb{R}^D$, $\mu_2 = (1, \dots, 1, 0, \dots, 0)^T \in \mathbb{R}^D$, with signal in the first 10 dimensions, and

$$\begin{aligned} \Sigma_2 &= \left(\begin{array}{c|c} \widetilde{\Sigma}_{10} & 0 \\ \hline 0 & I_{D-10} \end{array}\right), \\ \widetilde{\Sigma}_{10} &= \operatorname{diag}(1.367, 3.175, 3.247, 4.403, 1.249, \\ &\qquad 1.969, 4.035, 4.237, 2.813, 3.637). \end{aligned} \tag{57}$$

We simply chose 10 numbers uniformly at random on the interval $[1, 5]$ and other choice would give analogous results. Bayes accuracy is fixed at ≈ 0.95. In Fig 2b we show a scatter plot of the 9th and 10th dimension. From Fig. 3b we see that all the methods are similarly accurate in low dimensions, but they quickly degenerate as the number of dimensions increase, except kernel k-groups which is much more stable.

Now, consider $x \overset{iid}{\sim} \frac{1}{2}\mathcal{N}(\mu_1, \Sigma_1) + \frac{1}{2}\mathcal{N}(\mu_2, \Sigma_2)$ with

$$\begin{aligned} &2\Sigma_1 = \Sigma_2 = I_{20} \\ &\mu_1 = (\underbrace{0, \dots, 0}_{\times 20})^\top, \quad \mu_2 = \tfrac{1}{2}(\underbrace{1, \dots, 1}_{5}, \underbrace{0, \dots, 0}_{15})^\top. \end{aligned} \tag{58}$$

Bayes accuracy is ≈ 0.90. A scatter plot of the 4th and 5th dimensions is shown in Fig. 2c. We increase the sample size $n \in [10, 400]$ and show the accuracy versus n in Fig. 3c. We compare kernel k-groups, with different metrics, to k-means and GMM. We also use the best metric in this example for spectral clustering. We notice a superior performance of kernel k-groups compared to the other methods.

To consider non-normal data, we sample from the log-normal mixture $x \overset{iid}{\sim} (1/2)e^{\mathcal{N}(\mu_1,\Sigma_1)} + (1/2)e^{\mathcal{N}(\mu_2,\Sigma_2)}$ with the same parameters as in (58). The optimal Bayes accuracy is still ≈ 0.9. A scatter plot is in Fig. 2d and the results are shown in Fig. 3d. We use exactly the same metrics as in the normal mixture of Fig. 3c to illustrate that the proposed method still performs accurately.

Finally, we show a limitation of kernel k-groups, which is shared between all the other methods except for GMM. For highly unbalanced clusters, k-means, spectral clustering, kernel k-means and kernel k-groups all degenerate more quickly than GMM. A scatter plot of the first two dimensions is shown in Fig. 2e and the clustering results are in Fig. 3e, where we generate data according to

$$\begin{aligned} &x \overset{iid}{\sim} \frac{n_1}{2N}\mathcal{N}(\mu_1, \Sigma_1) + \frac{n_1}{2N}\mathcal{N}(\mu_1, \Sigma_1), \\ &\mu_1 = (0, 0, 0, 0)^\top, \quad \mu_2 = 1.5 \times (1, 1, 0, 0)^\top, \\ &\Sigma_1 = I_4, \quad \Sigma_2 = \left(\begin{array}{c|c} \frac{1}{2}I_2 & 0 \\ \hline 0 & I_2 \end{array}\right), \\ &n_1 = N - m, \quad n_2 = N + m, \quad N = 300. \end{aligned} \tag{59}$$

We then increase $m \in [0, 240]$ making the clusters progres-

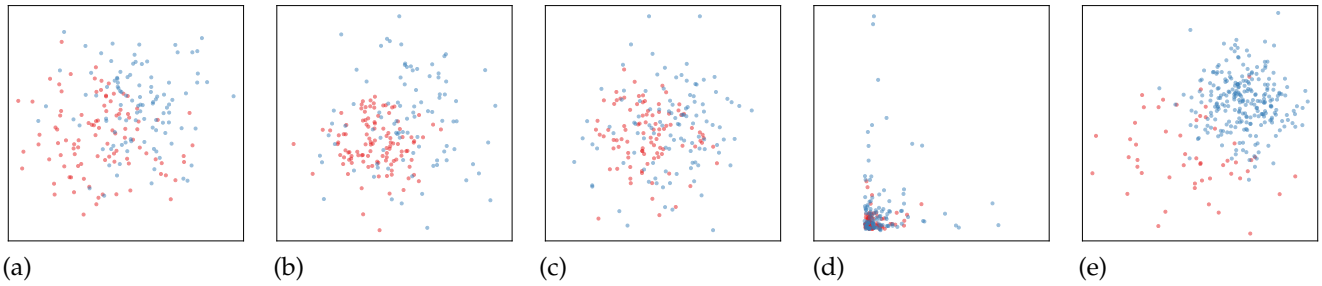

Fig. 2. Scatter plot of the last two dimensions where μ_2 has signal. Each plot has 200 points total. (a) Data distributed as in (56). (b) Data distributed as in (57). (c) Data distributed as in (58). (d) Parameters as in (58) but for lognormal mixture. (e) Data from (59) with $N = 300$ and $m = 200$.

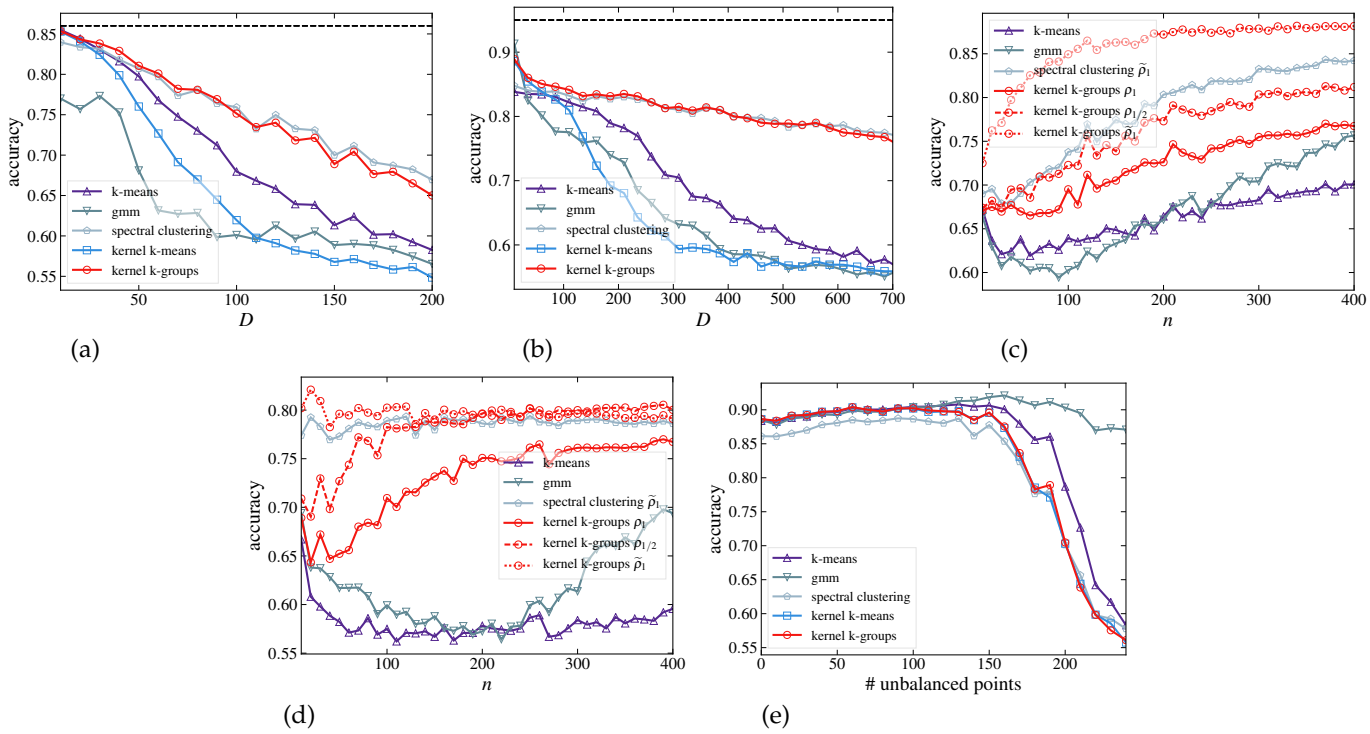

Fig. 3. Clustering results associated to the data illustrated in Fig. 2. For each experiment we perform 100 Monte Carlo runs and show the average accuracy. We ommit error bars since they are too small to be visible. (a) High dimensional Gaussian mixture according to (56). The dashed line is Bayes accuracy ≈ 0.86. We use the metric ρ_1 in (52), which is standard in energy statistics. (b) High dimensional Gaussian mixture according to (57). Bayes accuracy ≈ 0.95. We use ρ_1 in (52). (c) Gaussian mixture with parameters (58). We increase the number of sampled points in each trial. We use different metrics; see (52)–(54). Here, kernel k-groups is more accurate than spectral clustering. (d) Same experiment as in Fig. 3c but with a lognormal mixture with parameters (58). Again, kernel k-groups is more accurate than alternatives. The plot suggests that neither of these methods are consistent on this example since Bayes accuracy is ≈ 0.90. (e) Comparison between clustering methods on unbalanced clusters. The data is normally distributed as (59) where we vary $m \in [0, 240]$. We use the standard metric ρ_1 (see (52)) from energy statistics.

sively more unbalanced. 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 kernel k-groups to account for unbalanced clusters.

In Fig. 4 we show examples of two-dimensional datasets whose clustering results are shown in Table 2. For kernel k-means and kernel k-groups we initialize at random. We see that both methods perform closely, with higher accuracy than the other ones.

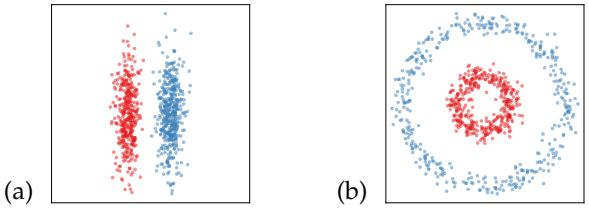

Fig. 4. (a) Data distributed as $x \overset{iid}{\sim} (1/2)\mathcal{N}(\mu_1, \Sigma_1) + (1/2)\mathcal{N}(\mu_1, \Sigma_1)$, $\mu_1 = (0, 0)^\top$, $\mu_2 = (6.5, 0)^\top$ and $\Sigma_1 = \Sigma_2 = \mathrm{diag}(1, 20)$. We sample 800 points. (b) Concentric circles with radius $r_1 = 1$ and $r_2 = 3$, with noise $0.2 \cdot \mathcal{N}(0, I_2)$. We sample 800 points with probability $1/2$ for each class.

5.2 Real Data Experiment

We consider the dermatology dataset [33], [34] which has 366 data points, each with 34 attributes where 33 are linear valued and one is categorical. There are 8 data points with missing entries in the "age" column. We complete the missing entries with the mean of the entire column, and

TABLE 2
Clustering the data from Fig. 4. For spectral clustering, kernel k-means and kernel k-groups we use the metric $\widetilde{\rho}_2$ (see (53)) for the data in Fig. 4a, while $\widehat{\rho}_1$ (see (54)) for the data in Fig. 4b. We have 800 points on each trial and 30 Monte Carlo runs for both datasets.

Method	Fig. 4a Accuracy	Fig. 4a SEM	Fig. 4b Accuracy	Fig. 4b SEM
kmeans	0.533	0.005	0.521	0.003
GMM	0.929	0.029	0.533	0.004
spectral clustering	0.577	0.010	0.725	0.003
kernel k-means	**1.000**	0.000	**1.000**	0.000
kernel k-groups	**1.000**	0.000	**1.000**	0.000

TABLE 3
Clustering the dermatology dataset of [33], [34] with kernel k-groups using the metric $\rho_{1/2}$ (see (52)) from energy statistics. The table below should be compared with Table 2 of [5], for which our results are slightly more accurate. See also Table 4 below for clustering metrics. The classes in the vertical indicates the ground truth and the classes in the horizontal correspond to the classification obtained by kernel k-groups. We show the estimated number of points for each class.

class	1	2	3	4	5	6	# cases
1	112	0	0	0	0	0	112
2	0	50	0	11	0	0	61
3	0	0	72	0	0	0	72
4	0	2	0	47	0	0	49
5	0	0	0	1	51	0	52
6	0	0	0	0	0	20	20
total	112	52	72	59	51	20	366

then we normalize the entire dataset to zero mean and unit variance. There are a total of 6 classes, and this is a challenging clustering problem. We refer the reader to [33], [34] for a complete description of the dataset, and also to [5] where this dataset was previously analyzed. In Table 3 we show the results of kernel k-groups using the metric (52) with $\alpha = 1/2$. We show the number of points assigned to each class while indicating the actual class that points belong to. In Table 4 we show this experiment using several clustering methods, and we also compare with the results from [5] and [6] on this same data. Our results provide an improvement in comparison to all the other methods and also the analysis of [5].

6 Conclusion

We proposed a formulation to clustering based on a weighted version of energy statistics, valid for arbitrary spaces of negative type. Our 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, and also to several graph partitioning problems.

We extended Hartigan's method to kernel spaces and proposed Algorithm 2, which we called kernel k-groups. This method was compared to kernel k-means and spectral clustering, besides k-means and GMM. Our numerical results show a superior performance of the proposed method, specially in high dimensions. We stress that kernel k-groups has the same complexity of kernel k-means[2].

2. An implementation of unweighted k-groups is publicly available in the energy package for R [35].

TABLE 4
For the dataset [33], [34] (see also Table 3) we show the accuracy (55) and the adjusted Rand index (aRand) of several methods. In [5] the authors obtained $\mathrm{aRand} = 0.9195$ using an energy statistics based method, while [6] obtains $\mathrm{aRand} = 0.9188$ where points with missing entries are removed. Below we complete the missing entries with the mean. If we remove the points with missing entries, kernel k-groups provides an improvement of $\mathrm{accuracy} = 0.9637$ and $\mathrm{aRand} = 0.9396$.

Method	Accuracy	aRand
kmeans	0.713	0.690
GMM	0.877	0.840
spectral clustering	0.954	0.912
kernel k-means	0.751	0.851
kernel k-groups	**0.962**	**0.936**

Kernel k-groups suffers a limitation shared by kernel k-means and spectral clustering which involves highly unbalanced clusters. An interesting problem that we leave open is to extend the method to such situations. Finally, kernel methods can benefit from sparsity and fixed-rank approximations of the Gram matrix, and there is plenty of room to make kernel k-groups more scalable.

Appendix

Two-Class Problem in One Dimension

Here we consider the simplest possible case which is one-dimensional data and a two-class problem. We propose an algorithm that does not depend on initialization. We used this simple scheme to compare with kernel k-groups given in algorithm 2. Both algorithms have the same clustering performance in the one-dimensional examples that we tested.

Let us fix $\rho(x, y) = |x - y|$ according to the standard energy distance. We also fix the weights $w(x) = 1$ for every data point x. We can thus compute the function (10) in $\mathcal{O}(n \log n)$ and minimize W directly. This is done by noting that

$$\begin{aligned} |x-y| &= (x-y)\mathbb{1}_{x\geq y} - (x-y)\mathbb{1}_{x<y} \\ &= x\left(\mathbb{1}_{x\geq y} - \mathbb{1}_{x<y}\right) + y\left(\mathbb{1}_{y>x} - \mathbb{1}_{y\leq x}\right) \end{aligned} \tag{60}$$

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

$$g\left(\mathcal{C}, \mathcal{C}\right) = \frac{1}{n^2} \sum_{x\in\mathcal{C}} \sum_{y\in\mathcal{C}} x\left(\mathbb{1}_{x\geq y} + \mathbb{1}_{y>x} - \mathbb{1}_{x\geq y} - \mathbb{1}_{x<y}\right). \tag{61}$$

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 $\widetilde{\mathcal{C}} = [x_j \in \mathcal{C} : x_1 \leq x_2 \leq \cdots \leq x_n]$, we get

$$g(\widetilde{\mathcal{C}}, \widetilde{\mathcal{C}}) = \frac{2}{n^2} \sum_{\ell=1}^{n} (2\ell - 1 - n) x_\ell. \tag{62}$$

Note that the cost of computing $g(\widetilde{\mathcal{C}}, \widetilde{\mathcal{C}})$ 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, $\mathbb{X} = \bigcup_{j=1}^{k} \widetilde{\mathcal{C}}_j$, the within energy dispersion can be written explicitly as

$$W(\widetilde{\mathcal{C}}_1, \ldots, \widetilde{\mathcal{C}}_k) = \sum_{j=1}^{k} \sum_{\ell=1}^{n_j} \frac{2\ell - 1 - n_j}{n_j} \, x_\ell. \qquad (63)$$

For a two-class problem we can use the formula (63) to cluster the data through a simple algorithm as follows. We first order the entire dataset, $\mathbb{X} \to \widetilde{\mathbb{X}}$. Then we compute (63) for each possible split of $\widetilde{\mathbb{X}}$ and pick the point which gives the minimum value of W. This procedure is described in Algorithm 3. Note that this algorithm is deterministic, however, it only works for one-dimensional data with Euclidean distance. Its total complexity is $\mathcal{O}(n \log n + n^2) = \mathcal{O}(n^2)$.

Algorithm 3 Clustering algorithm to find local solutions to the optimization problem (17) for a two-class problem in one dimension.

input data $\mathbb{X}$
output label matrix Z

sort $\mathbb{X}$ obtaining $\widetilde{\mathbb{X}} = [x_1, \ldots, x_n]$
for $j \in [1, \ldots, n]$ **do**
  $\widetilde{\mathcal{C}}_{1,j} \leftarrow [x_i : i = 1, \ldots, j]$
  $\widetilde{\mathcal{C}}_{2,j} \leftarrow [x_i : i = j+1, \ldots, n]$
  $W^{(j)} \leftarrow W(\widetilde{\mathcal{C}}_{1,j}, \widetilde{\mathcal{C}}_{2,j})$ (see (63))
end for
$j^\star \leftarrow \arg\min_j W^{(j)}$
for $j \in [1, \ldots, n]$ **do**
  if $j \leq j^\star$ **then**
    $Z_{j\bullet} \leftarrow (1, 0)$
  else
    $Z_{j\bullet} \leftarrow (0, 1)$
  end if
end for

ACKNOWLEDGMENTS

We would like to thank Carey Priebe for discussions. We would like to acknowledge the support of the Transformative Research Award (NIH #R01NS092474) and the Defense Advanced Research Projects Agencys (DARPA) SIMPLEX program through SPAWAR contract N66001-15-C-4041.

REFERENCES

[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] G. J. Székely and M. L. Rizzo. The Energy of Data. *Annu. Rev. Stat. Appl.*, 207:447–479, 2017.
[3] 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.
[4] G. J. Székely, M. L. Rizzo, and N. K. Bakirov. Measuring and testing dependence by correlation of distances. *Ann. Stat.*, 35(6):2769–2794, 2007.
[5] 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.
[6] S. Li and M. Rizzo. k-Groups: A Generalization of k-Means Clustering. arXiv:1711.04359 [stat.ME], 2017.
[7] R. Lyons. Distance Covariance in Metric Spaces. *The Annals of Probability*, 41(5):3284–3305, 2013.
[8] 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.
[9] C. Shen and J. T. Vogelstein. The exact equivalence of distance and kernel methods for hypothesis testing. arXiv:1806.05514 [stat.ML], 2018.
[10] S. P. Lloyd. Least Squares Quantization in PCM. *IEEE Transactions on Information Theory*, 28(2):129–137, 1982.
[11] 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.
[12] E. Forgy. Cluster Analysis of Multivariate Data: Efficiency versus Interpretabiliby of Classification. *Biometrics*, 21(3):768–769, 1965.
[13] B. Schölkopf, A. J. Smola, and K. R. Müller. Nonlinear Component Analysis as a Kernel Eigenvalue Problem. *Neural Computation*, 10:1299–1319, 1998.
[14] M. Girolami. Kernel Based Clustering in Feature Space. *Neural Networks*, 13(3):780–784, 2002.
[15] 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.
[16] 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.
[17] 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.
[18] M. Filippone, F. Camastra, F. Masulli, and S. Rovetta. A Survey of Kernel and Spectral Methods for Clustering. *Pattern Recognition*, 41:176–190, 2008.
[19] J. A. Hartigan. *Clustering Algorithms*. John Wiley & Sons, Inc., New York, NY, USA, 1975.
[20] 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.
[21] 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.
[22] 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.
[23] N. Aronszajn. Theory of Reproducing Kernels. *Transactions of the American Mathematical Society*, 68(3):337–404, 1950.
[24] 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.
[25] 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.
[26] J. Shi and J. Malik. Normalized Cust and Image Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.
[27] 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.
[28] B. Kernighan and S. Lin. An Efficient Heuristic Procedure for Partitioning Graphs. *The Bell System Technical Journal*, 49(2):291–307, 1970.
[29] 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.
[30] S. X. Yu and J. Shi. Multiclass Spectral Clustering. In *Proceedings Ninth IEEE International Conference on Computer Vision*, volume 1, pages 313–319, 2003.
[31] 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.
[32] 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.

[33] D. Dheeru and E. K. Taniskidou. UCI machine learning repository, 2017.

[34] H. A. Güvenir, G. Demiröza, and N. Ilterb. Learning differential diagnosis of erythemato-squamous diseases using voting feature intervals. *Artificial Intelligence in Medicine*, 13(3):147–165, 1998.

[35] M. Rizzo and G. Szekely. energy: E-Statistics: Multivariate Inference via the Energy of Data. R package version 1.7-5; https://CRAN.R-project.org/package=energy, 2018.

Guilherme França has a BS degree in physics and received both his MS and PhD in Theoretical Physics from the Institute of Theoretical Physics IFT-UNESP/ICTP-SAIFR (2012). He was a Postdoctoral Fellow with the High Energy Theory group in the Physics Department at Cornell University (2013–2015), working on topics of mathematical physics and non-perturbative methods in quantum field theory and statistical mechanics. He was a Postdoctoral Fellow in the Computer Science Department at Boston College (2015), working in optimization methods. Since 2016 he is a Postdoctoral Fellow at Johns Hopkins University, currently with the Mathematical Institute for Data Science (MINDS). His current research interests involve machine learning and optimization.

Maria L. Rizzo holds a BS and MA degree in Mathematics from the University of Toledo, a MS in Applied Statistics from BGSU, and a PhD from BGSU in 2002. After four years as Assistant Professor at Ohio University Department of Mathematics, she returned to BGSU in 2006, where she is currently Professor of Statistics in the Department of Mathematics and Statistics. Her primary research interests are centered on energy statistics, distance correlation and its applications, computational statistics, and related topics. In addition to teaching statistics, data science, and actuarial science, she enjoys writing textbooks about statistical computing and using R software. She has a forthcoming research monograph on energy statistics (joint with G. Szekely) and also other book writing projects under contract as well.

Joshua T. Vogelstein received a BS degree from the Department of Biomedical Engineering (BME) at Washington University in St. Louis, MO in 2002, a MS degree from the Department of Applied Mathematics and Statistics (AMS) at Johns Hopkins University (JHU) in Baltimore, MD in 2009, and a PhD degree from the Department of Neuroscience at JHU in 2009. He was a Postdoctoral Fellow in AMS/JHU from 2009 until 2011, at which time he was appointed an Assistant Research Scientist, and became a member of the Institute for Data Intensive Science and Engineering. He spent 2 years at Information Initiative at Duke University, before coming home to his current appointment as Assistant Professor in BME/JHU, and core faculty in both the Institute for Computational Medicine and the Center for Imaging Science, as well as a member of the Kavli Neuroscience Discovery Institute. He married his kindergarten sweetheart in the summer of 2014.

His primary research interest is to extend and fuse statistical machine learning and big data science to address the most important brain science and mental health questions of our time, particularly connectomics. His groups research has been featured in a number of prominent scientific and engineering journals and conferences including Nature, Nature Methods, Science, Cell, PNAS, Science Translational Medicine, JMLR, PAMI, Annals of Applied Statistics, NIPS, and SIAM Journal of Matrix Analysis and Applications. In 2011, he co-founded the Open Connectome Project which expanded in 2015 to be NeuroData, whose mission is to enable terascale neuroscience for everyone. All his works are conducted according to the highest standards of open science.