

Finer-Grained Hardness of Kernel Density Estimation

Josh Alman  

Department of Compute Science, Columbia University, New York, NY, USA

Yunfeng Guan  

Department of Compute Science, Columbia University, New York, NY, USA

Abstract

In batch Kernel Density Estimation (KDE) for a kernel function $f : \mathbb{R}^m \times \mathbb{R}^m \rightarrow \mathbb{R}$, we are given as input $2n$ points $x^{(1)}, \dots, x^{(n)}, y^{(1)}, \dots, y^{(n)} \in \mathbb{R}^m$ in dimension m , as well as a vector $v \in \mathbb{R}^n$. These inputs implicitly define the $n \times n$ kernel matrix K given by $K[i, j] = f(x^{(i)}, y^{(j)})$. The goal is to compute a vector $w \in \mathbb{R}^n$ which approximates Kw , i.e., with $\|Kw - v\|_\infty < \varepsilon \|w\|_1$. For illustrative purposes, consider the Gaussian kernel $f(x, y) := e^{-\|x-y\|_2^2}$. The classic approach to this problem is the famous Fast Multipole Method (FMM), which runs in time $n \cdot O(\log^m(\varepsilon^{-1}))$ and is particularly effective in low dimensions because of its exponential dependence on m . Recently, as the higher-dimensional case $m \geq \Omega(\log n)$ has seen more applications in machine learning and statistics, new algorithms have focused on this setting: an algorithm using discrepancy theory, which runs in time $O(n/\varepsilon)$, and an algorithm based on the polynomial method, which achieves inverse polynomial accuracy in almost linear time when the input points have bounded square diameter $B < o(\log n)$.

A recent line of work has proved fine-grained lower bounds, with the goal of showing that the “curse of dimensionality” arising in FMM is necessary assuming the Strong Exponential Time Hypothesis (SETH). Backurs et al. [NeurIPS 2017] first showed the hardness of a variety of Empirical Risk Minimization problems including KDE for Gaussian-like kernels in the case with high dimension $m = \Omega(\log n)$ and large scale $B = \Omega(\log n)$. Alman et al. [FOCS 2020] later developed new reductions in roughly this same parameter regime, leading to lower bounds for more general kernels, but only for very small error $\varepsilon < 2^{-\log^{\Omega(1)}(n)}$.

In this paper, we refine the approach of Alman et al. to show new lower bounds in all parameter regimes, closing gaps between the known algorithms and lower bounds. For example:

- In the setting where $m = C \log n$ and $B = o(\log n)$, we prove Gaussian KDE requires $n^{2-o(1)}$ time to achieve additive error $\varepsilon < \Omega(m/B)^{-m}$, matching the performance of the polynomial method up to low-order terms.
- In the low dimensional setting $m = o(\log n)$, we show that Gaussian KDE requires $n^{2-o(1)}$ time to achieve ε such that $\log \log(\varepsilon^{-1}) > \tilde{\Omega}(\log n)/m$, matching the error bound achievable by FMM up to low-order terms. To our knowledge, no nontrivial lower bound was previously known in this regime.

Our approach also generalizes to any parameter regime and any kernel. For example, we achieve similar fine-grained hardness results for any kernel with slowly-decaying Taylor coefficients such as the Cauchy kernel. Our new lower bounds make use of an intricate analysis of the “counting matrix”, a special case of the kernel matrix focused on carefully-chosen evaluation points. As a key technical lemma, we give a novel approach to bounding the entries of its inverse by using Schur polynomials from algebraic combinatorics.

2012 ACM Subject Classification Theory of computation \rightarrow Problems, reductions and completeness

Keywords and phrases Kernel Density Estimation, Fine-Grained Complexity, Schur Polynomials

Digital Object Identifier 10.4230/LIPIcs.CCC.2024.35

Funding Supported in part by a grant from the Simons Foundation (Grant Number 825870 JA) and a Google Research Scholar award.

Acknowledgements We would like to thank Amol Aggarwal for constructive discussions on Schur polynomials, and anonymous reviewers for helpful suggestions.



© Josh Alman and Yunfeng Guan;
licensed under Creative Commons License CC-BY 4.0
39th Computational Complexity Conference (CCC 2024).

Editor: Rahul Santhanam; Article No. 35; pp. 35:1–35:21

Leibniz International Proceedings in Informatics



Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany



1 Introduction

In computational statistics and learning theory, many applications reduce to solving the following problem: Given a set of points $X \subseteq \mathbb{R}^m$ sampled from some unknown distribution \mathcal{D} , estimate the probability density at a query point $y \in \mathbb{R}^m$ (or multiple such points $Y \subseteq \mathbb{R}^m$). This problem is known as the density estimation problem and has attracted interest in theoretical computer science in recent years. One of the most common methods for this problem is Kernel Density Estimation (KDE), which tries to approximate the distribution by a sum of kernel functions centered at each data point x . More concretely, if a kernel function $k : \mathbb{R}^m \times \mathbb{R}^m \rightarrow [0, 1]$ is appropriately picked, then the *Kernel Density* $\text{KDF}(y) := \frac{1}{n} \sum_{x \in X} k(x, y)$ is a reasonably good approximation of \mathcal{D} at point y .¹ In this work, we will focus on the popular class of radial kernels, i.e., kernels of the form $K(x, y) = f(\|x - y\|_2^2)$ for some function $f : \mathbb{R}_{\geq 0} \rightarrow [0, 1]$. Some prominent radial kernels include the Gaussian kernel with $f(u) = e^{-u}$, Rational Quadratic kernel with $f(u) = 1/(1 + u)^\sigma$ for constant σ , and t -Student kernel with $f(u) = 1/(1 + u^\rho)$ for constant ρ .

By virtue of its excellent statistical properties, Kernel Density Estimation has found numerous applications in computational statistics for tasks like mean estimation, classification, and outlier detection; see, for instance, the surveys [18, 13] and [24, Chapter 1]. The prevalence of kernel methods in machine learning has also led to many new applications of Kernel Density Estimation [22]. One popular recent example is in “attention computation”, the time bottleneck in computations involving transformers and other large language models; this is known to be essentially equivalent to Kernel Density Estimation with the Gaussian kernel [3]².

In light of its wide applicability, tremendous effort has also been put into designing efficient algorithms for the KDE problem. The straightforward algorithm shows that for a given y , $\text{KDF}(y)$ can be computed in $O(n)$ time, assuming f is efficiently computable.³ If one aims at an exact result, this simple algorithm seems to be optimal. However, a number of advances starting from the celebrated Fast Multipole Method [14] have successfully brought the running time down to $n^{o(1)}$ in various parameter regimes (after preprocessing the set X), provided an approximation to the Kernel Density suffices. Before introducing these algorithmic ideas, we first give a formal definition of the (approximate) KDE problem. For simplicity, we present here the batched version, which asks to compute the KDF value for a collection of query points simultaneously. (We also compare the batched version and the data structure version in Section 1.5.)

► **Definition 1** ((Approximate) Kernel Density Estimation $\text{KDE}_f(n, m, \varepsilon, B)$). *Let $f : [0, B] \rightarrow [0, 1]$ be a real function and define the (kernel) function $k(x, y) = f(\|x - y\|_2^2)$.*

Suppose we are given as input $2n$ points $x^{(1)}, \dots, x^{(n)}, y^{(1)}, \dots, y^{(n)} \in \mathbb{R}^m$ with the guarantee that $\|x^{(i)} - y^{(j)}\|_2^2 \leq B$ for all $i, j \in [n]$. Define the kernel matrix $K \in [0, 1]^{n \times n}$ by $K[i, j] = f(\|x^{(i)} - y^{(j)}\|_2^2)$ for $i, j \in [n]$. Then given additionally a vector $u \in \mathbb{R}^n$, one needs to output a vector $v \in \mathbb{R}^n$ such that $\|v - Ku\|_\infty \leq \varepsilon \|u\|_1$.

¹ For example, when $m = 1$ and $k(x, y) = \mathbb{1}[|x - y| \leq 1]$, the KDF is the histogram of the dataset X .

² The parameters of KDE correspond directly to the parameters in attention: n is the number of “tokens” in the query sequence, and m is the dimension of the vectors which encode tokens.

³ In this work we always assume $f(\|x - y\|_2^2)$ can be exactly computed in $O(1)$ time. Most of the previous works adopt this assumption.

1.1 Algorithms

As a computational problem, KDE has its running time dependent on four parameters: the number of data points / query points n , the dimension of vectors m , the *additive* error of approximation ε , and the maximum pairwise (square) distance B . Multiple parameter regimes arise from their interplay, and the KDE problem tends to have rather different behavior across the regimes. Of course, the choice of the kernel function can also substantially change the complexity of the problem; in the following discussion we take the Gaussian kernel as a running example.

Low dimensional KDE: $m = o(\log n)$. The first nontrivial algorithm for the KDE problem is Greengard and Roklin's Fast Multipole Method [14]. In this method, one partitions the space into bounded regions, and then Taylor expands the kernel around centers of these regions. In this way a truncation of the Taylor series yields a approximation of high accuracy when the data points lie in distant cells. For the Gaussian kernel, the Fast Multipole Method runs in time $O(n \log^{O(m)}(n/\varepsilon))$, which is exceptionally good in low dimensions (e.g., $m = O(1)$ in the original physical context⁴ of [15]). However, in higher dimensional regimes, this approach suffers from an exponential dependence on m , which is inherent in the (deterministic) space partitioning procedure (essentially building a quad-tree), and shared by other tree-based methods.

Moderate dimensional KDE: $m = \Theta(\log n)$. One way to avoid the exponential dependence on m of the Fast Multipole Method is to use the method of polynomial approximation directly, without combining it with space partitioning. Alman and Aggarwal [1] pinned down the optimal degree of a polynomial that approximates $f(u) = e^{-Bu}$ over $u \in [0, 1]$ by analyzing its Chebyshev truncation. A polynomial approximation of e^{-Bu} then allows one to approximate the Gaussian kernel matrix using a matrix consisting of only polynomial entries. Such a matrix admits a decomposition as a product of two matrices of dimension $n \times n^{o(1)}$, for which the matrix-vector product can be performed efficiently. As a result, they gave an algorithm for Gaussian KDE running in $n^{1+o(1)}$ time when $m = O(\log n)$, $B = o(\log n)$ and $\varepsilon = 1/\text{poly}(n)$.

High dimensional KDE: $m = \omega(\log n)$. Another technique often used in the study of KDE is random sampling. For example, to compute $\frac{1}{n} \sum_{x \in X} k(x, y)$, one can sample a subset $S \subseteq X$ so that $\frac{1}{|S|} \sum_{x \in S} k(x, y)$ is a close approximation to $\text{KDF}(y)$ when $|S|$ is sufficiently large. Simple calculation shows that $|S| = O(\log n/\varepsilon^2)$ suffices, and a $\tilde{O}(n/\varepsilon^2)$ time (randomized) algorithm follows. We note that this algorithm has no dependence on m and works for arbitrarily high dimensions. This folklore random sampling algorithm stood unchallenged until recently Phillips and Tai [20] devised an $O(n/\varepsilon)$ -time algorithm by building a small coresets based on discrepancy theory. They show that a clever subsampling scheme yields a smaller $S \subseteq X$ which has the same accuracy as its counterpart when used in the random sampling algorithm. We in addition note that much effort [10, 11, 5, 7, 8, 9] has been dedicated to the *relative* error setting, combining sampling schemes with techniques from high-dimensional geometry (such as hashing-based space partitioning). See Section 1.5 below where we compare the additive error and relative error settings and survey these algorithms in more detail.

⁴ The Fast Multipole Method was originally introduced to solve the n -body problem from physics.

1.2 Lower bounds

Despite the great variety of algorithms developed over the years, we still lack a comprehensive understanding of the complexity of the KDE problem. In this work, our goal is to complement the distinct algorithms targeting different parameter regimes with (nearly) matching running time lower bounds, and explain how the complexity of KDE is affected by parameters.

The first and most influential known lower bound was developed by Backurs, Indyk and Schmidt [6]. This work establishes a reduction from the Bichromatic Closest Pair (BCP) problem to a collection of empirical risk minimization problems including KDE. As BCP is a standard hard problem in fine-grained complexity assuming the Strong Exponential Time Hypothesis (SETH), the reduction leads to conditional lower bounds on the running time of KDE. To explain in detail, we first give the formal statement of the BCP problem.

► **Problem 2** (Bichromatic Closest Pair). *Hamming-BCP(n, m): Given two sets $X = \{x^{(1)}, \dots, x^{(n)}\}, Y = \{y^{(1)}, \dots, y^{(n)}\} \in \{0, 1\}^m$, compute $\min_{i, j \in [n]} \|x^{(i)} - y^{(j)}\|_2^2$.*

We then observe that terms in the KDE result $\|K \times \mathbf{1}\|_1$ can be grouped according to the pairwise distance between $x^{(i)}$ and $y^{(j)}$.

$$\|K \times \mathbf{1}\|_1 = \sum_{i=1}^n \sum_{j=1}^n f(\|x^{(i)} - y^{(j)}\|_2^2) = \sum_{p=0}^m f(p) \cdot \#\{(i, j) \in [n]^2 : \|x^{(i)} - y^{(j)}\|_2^2 = p\}.$$

This identity provides a way of extracting minimum pairwise distance from the KDE result. Indeed, if $\min_{i, j} \|x^{(i)} - y^{(j)}\|_2^2 \geq p+1$, then $\|K \times \mathbf{1}\|_1 \leq n^2 f(p+1)$ (assuming f is decreasing); otherwise, there exists a pair $(x^{(i)}, y^{(j)})$ with $\|x^{(i)} - y^{(j)}\|_2^2 \leq p$ and $\|K \times \mathbf{1}\|_1 \geq f(p)$. In this way, if f is decreasing quickly enough, one can decide whether $\min_{i, j} \|x^{(i)} - y^{(j)}\|_2^2 \leq p$ based on a sufficiently accurate approximation of $\|K \times \mathbf{1}\|_1$.

This relatively simple reduction gives strong running time lower bounds in the moderate/high dimensional regime. In the original paper, [6] shows that when $m = \Omega(\log n)$ and $B = \Omega(\log n)$, it requires $n^{2-o(1)}$ time to approximate the Gaussian KDE to $\varepsilon = 2^{-\text{poly} \log n}$. This result is later improved by Alman and Aggarwal [1] (also in [11] for KDE with relative error), who combine the same reduction with the hardness result of approximate BCP [21] and show that even approximating to accuracy $\varepsilon = 1/\text{poly}(n)$ requires $n^{2-o(1)}$ time.

However, we note that this reduction relies on a strong premise that $f(p)/f(p+1) > n^2$, i.e., the kernel function has *rapid decay*. (The variant in [1] relies on a similar condition.) This fails to hold for many kernels of interest, including all smooth kernels (such as the Rational Quadratic kernels and t -Student kernels) and small-scale Gaussian kernels with small $B < o(\log n)$.

To get around this barrier, Alman, Chu, Schild and Song [2] extends the reduction of [6], by solving *multiple* KDE instances and, roughly speaking, combining their answers to extract *more* information about the pairwise distances. Concretely, we define the *distance vector* (in terms of two sets of points) and *counting matrix* (in terms of the function f) as follows.

► **Definition 3.** *Let $X = \{x^{(1)}, \dots, x^{(n)}\}, Y = \{y^{(1)}, \dots, y^{(n)}\} \in \{0, 1\}^m$ be two sets of points. We define the distance vector $w = [\#\{(i, j) : \|x^{(i)} - y^{(j)}\|_2^2 = p\}]_{p \in [m]}$, and for $\alpha_1, \dots, \alpha_m \in [0, 1]$ define the counting matrix $M = [f(\alpha_\ell \cdot p)]_{\ell, p \in [m]}$.*

Consider the matrix-vector multiplication $M \times w$. By the same argument as in the reduction of [6], we observe each entry in $\tilde{w} := M \times w$ can be computed by some KDE instance $K_\ell \times \mathbf{1}$ (K_ℓ is associated with appropriately scaled X and Y):

$$\tilde{w}[\ell] = \sum_{i=1}^n \sum_{j=1}^n f(\|\sqrt{\alpha_\ell} x^{(i)} - \sqrt{\alpha_\ell} y^{(j)}\|_2^2) = \sum_{p=0}^m f(\alpha_\ell p) \cdot \#\{(i, j) : \|x^{(i)} - y^{(j)}\|_2^2 = p\}.$$

Once \tilde{w} is obtained from KDE subroutines, one can then approximate the distance vector by simply computing $M^{-1} \times \tilde{w}$. If this approximation to w has (additive) error bounded by $1/3$, then a subsequent rounding step yields the exact distance vector, and automatically a solution to the BCP problem.

To analyze this reduction, we define a key quantity

$$\tau(M) := \max_{0 \neq v \in \mathbb{R}^m} \frac{\|M^{-1}v\|_\infty}{\|v\|_\infty}.$$

Suppose $\|K_\ell \times \mathbf{1}\|_\infty \leq \varepsilon \|\mathbf{1}\|_1$ for all $\ell \in [m]$. Then

$$\|M^{-1}\tilde{w} - w\|_\infty = \|M^{-1}(\tilde{w} - Mw)\|_\infty \leq \tau(M) \cdot \max_{\ell \in [m]} \|K_\ell \times \mathbf{1}\|_1 \leq \tau(M) \cdot n^2 \varepsilon.$$

Therefore, any algorithm that approximates KDE instances to accuracy $\varepsilon = 1/(3n^2\tau(M))$ in $n^{2-\Omega(1)}$ time amounts to a truly subquadratic BCP algorithm and refutes SETH. (In this paper below, we slightly modify this approach to improve the dependence on n in ε from quadratic to linear; see Section 1.4 for more details.)

To complete the hardness result, it remains to bound the quantity $\tau(M)$ as a function of m and B . [2] makes a generic statement relating $\tau(M)$ to the approximability of f by low-degree polynomials. In particular, it is shown that for all three kernels – Rational Quadratic kernel, t -Student kernel and small-scale Gaussian kernel – performing $\Omega(\log n)$ -dimensional KDE requires $n^{2-o(1)}$ time to achieve accuracy $\varepsilon = 2^{-\text{poly} \log n}$.

The proof of [2] for this bound on $\tau(M)$ is highly technical, and requires new tools from analysis and linear algebra. For the sake of coherence, we defer an overview of the details to Section 1.4. A main weakness of [2] is that it only gives hardness for very small ε , which is an inherent consequence of their approach to bounding $\tau(M)$. One of the main technical contributions of our paper, which we discuss in more detail shortly, is an improved approach to bounding $\tau(M)$ which yields exponentially better bounds on ε for many kernel functions of interest.

1.3 Our contribution

In this work we give stronger negative results for the KDE problem, and pin down its complexity in each parameter regime. We mainly focus on the Gaussian kernel, and on two of the most used smooth kernels – the Rational Quadratic kernel and the t -Student kernel. That said, our approach is general and would apply to any other kernel of interest after some calculations (See Section 1.5 for further discussion). To give a unified presentation, we formulate both the positive and negative results as upper bounds and lower bounds on $1/\varepsilon$. More specifically, we will answer the following questions.

► **Question 4.** *Fix a kernel function f and a parameter regime determined by $m = o(\log n)$ (resp. $\Theta(\log n)$, $\omega(\log n)$) and $B = o(\log n)$ (resp. $\Omega(\log n)$). What is the range of $1/\varepsilon$ achievable in $n^{1+o(1)}$ time? What is the range of $1/\varepsilon$ that requires $n^{2-o(1)}$ time?*

For simplicity, in the following discussion we understand “Easy” as being achievable in $n^{1+o(1)}$ time, and understand “Hard” as requiring $n^{2-o(1)}$ time.

Gaussian kernel. In the regime $m = \Theta(\log n)$, $B = o(\log n)$, the polynomial method [1] gives the best known positive result: Gaussian KDE is Easy when $1/\varepsilon < (m/B)^{o(m)}$. The best known negative result due to [2] establishes the Hardness of KDE when $1/\varepsilon > 2^{\text{poly} \log n}$. In this work we improve the negative result and show that the polynomial method is optimal up to a low-order $2^{O(m)}$ factor in $1/\varepsilon$.

► **Theorem 5.** *Assuming SETH, for every $q \in (0, 1)$, there exist $C_1, C_2 > 0$ such that when $m > C_1 \log n$ and $1/\varepsilon > (m/B)^m \cdot C_2^m$, GaussianKDE(n, m, B, ε) cannot be solved in $O(n^{2-q})$ time.*

In the low dimensional regime⁵, $c^{\log^* n} < m < o(\log n)/(\log \log n)$, the Fast Multipole Method has stood unchallenged for over three decades. Using this method, Gaussian KDE is Easy when $\log \log(1/\varepsilon) < o(\log n)/m$. It is natural to conjecture that a substantial improvement is impossible. However, to the best of our knowledge, no previous hardness result was known for Gaussian kernel KDE in this regime.⁶ In this work we give the first negative result against the Fast Multipole Method, and in particular show that the $\log \log(1/\varepsilon)$ achieved by the Fast Multipole Method is optimal up to a roughly logarithmic factor in m .

► **Theorem 6.** *Assuming SETH, for every $q > 0$, there exist $C_1, C'_1, C_2 > 0$ such that when $C_1^{\log^* n} < m < C'_1(\log n)/(\log \log n)$ and $\log \log(1/\varepsilon) > (\log n)/m \cdot (\log m) \cdot C_2^{\log^* n}$, GaussianKDE(n, m, B, ε) cannot be solved in $O(n^{2-q})$ time.*

Apart from the two major improvements above, our techniques also lead to new results in other regimes. As a straightforward corollary of Theorem 5, we show $1/\varepsilon > ((\log n)/B)^{\Omega(\log n)}$ is Hard for high-dimensional $m = \omega(\log n)$, small-scale $B = o(\log n)$ regime, improving the $1/\varepsilon > 2^{\text{poly} \log n}$ bound in [2]. For the large scale regime, we note the hardness result in [1] requires $(B/\log n)$ to tend to infinity alongside $(m/\log n)$. This inherent dependence between B and m is an inevitable consequence of the rapid decay condition. In comparison, as our new reduction is free of such restrictions, new hardness results can be developed as well in the regime where $B = \Theta(\log n)$ is fixed and only $m/\log n$ tends to infinity.

We summarize all the known upper and lower bounds for Gaussian KDE in Table 1.

■ **Table 1** Summary of known results for Gaussian KDE, incorporating our new Theorems 5 and 6. The new hardness in high dimensions follows from our Theorem 5 and a straightforward reduction from moderate to high dimension. The stated hardness results for large scale and moderate or high dimension were previously known [6, 1], although we improve the constant C in these cases.

	small scale $B = o(\log n)$	large scale $B = \Omega(\log n)$
low dimension $c^{\log^* n} < m < o\left(\frac{\log n}{\log \log n}\right)$	Easy: $\log \log(1/\varepsilon) < o((\log n)/m)$ Hard (new): $\log \log(1/\varepsilon) > \tilde{\Omega}(\log n)/m$	
moderate dimension $m = C \log n$	Easy: $1/\varepsilon < (m/B)^{o(m)}$ Hard (new): $1/\varepsilon > \Omega(m/B)^m$	Easy: $1/\varepsilon < n^{1-q}$ Hard: $1/\varepsilon > n^C$ for some $C > 1$
high dimension $m > \omega(\log n)$	Easy: $1/\varepsilon < n^{1-q}$ Hard (new): $1/\varepsilon > ((\log n)/B)^{\Omega(\log n)}$	Easy: $1/\varepsilon < n^{1-q}$ Hard: $1/\varepsilon > n^C$ for some $C > 1$

Rational Quadratic kernel and t -Student kernel. For the Rational Quadratic kernel $f(x) = 1/(1+x)^\sigma$ and t -Student kernel $f(x) = 1/(1+x^\rho)$ parameterized by absolute constants $\sigma, \rho \geq 1$, we give similar lower bound results. In the moderate to high dimensional regime $d = \Omega(\log n)$, the best known negative result is again that $1/\varepsilon = 2^{\text{poly} \log n}$ is Hard by [2]. We show that this can be improved to $1/\varepsilon = \text{poly}(n)$. This parallels the improvement

⁵ We use \log^* to denote the very slowly-growing iterated logarithm function.

⁶ For non-Lipschitz kernels, some hardness results for very low error in low dimensions were established in [2].

by [1] for the large-scale Gaussian kernel from $1/\varepsilon = 2^{\text{poly} \log n}$ to $1/\varepsilon = \text{poly}(n)$ (and we recall that these kernels do not decrease quickly enough to prove this using the approach of [6, 1]).

► **Theorem 7.** *For Rational Quadratic kernel $f(x) = 1/(1+x)^\sigma$ and t -Student kernel $f(x) = 1/(1+x^\rho)$ parameterized by absolute constants $\sigma, \rho \geq 1$, the following holds. Assuming SETH, for every $q > 0$, there exists $C_1, C_2 > 0$ such that if $m > C_1 \log n, 1/\varepsilon > n^{C_2}$, then $\text{KDE}_f(n, m, B = 1, \varepsilon)$ cannot be solved in $O(n^{2-q})$. Here C_2 is dependent on σ or ρ .*

This result implies that KDE for Rational Quadratic kernel and t -Student kernel are strictly harder than Gaussian KDE: for $B = O(1)$ and $m = \Theta(\log n)$, Gaussian KDE is Easy when $1/\varepsilon < m^{o(m)}$ whereas KDE for Rational Quadratic kernel and t -Student kernel are Hard for $1/\varepsilon > 2^{\Omega(m)}$. Interestingly, this is in sharp contrast to the phenomenon in the study of KDE with relative error, where KDE for smooth kernels are seemingly easier to solve. We discuss this difference in detail in Section 1.5.

In the low dimensional regime, we also prove negative results against the Fast Multipole Method. For both Rational Quadratic kernel and t -Student kernel, the Fast Multipole Method has similar bound $O(n \log^{O(m)}(n/\varepsilon))$ on running time. We complement this algorithm with a matching lower bound up to a $\tilde{O}(\log m)$ factor in $\log \log(1/\varepsilon)$.

► **Theorem 8.** *For Rational Quadratic kernel $f(x) = 1/(1+x)^\sigma$ and t -Student kernel $f(x) = 1/(1+x^\rho)$ parameterized by absolute constants $\sigma, \rho \geq 1$, the following holds. Assuming SETH, for every $q > 0$, there exist $C_1, C'_1, C_2 > 0$ such that when $C_1^{\log^* n} < m < C'_1(\log n)/(\log \log n)$ and $\log \log(1/\varepsilon) > (\log n)/m \cdot (\log m) \cdot C_2^{\log^* n}$, $\text{KDE}_f(n, m, B = 1, \varepsilon)$ cannot be solved in $O(n^{2-q})$ time. Here C_2 is dependent on σ or ρ .*

1.4 Techniques

As discussed in Section 1.2, [2] established an upper bound on $1/\varepsilon$ which hinges on a key quantity $\tau(M)$, and the central ingredient of their reduction is a bound on this quantity. Therefore we start by sketching the proof [2] developed for this bound. By definition,

$$\tau(M) = \max_{\|v\|_\infty=1} \|M^{-1}v\|_\infty \leq \max_{\|v\|_\infty=1} \max_{s \in [m]} \sum_{t=1}^m |M^{-1}[s, t]| |v[t]| \leq m \max_{s, t \in [m]} |M^{-1}[s, t]|.$$

By Cramer's rule, we write $M^{-1}[s, t] = \frac{\det(M_{t-}^{s-})}{\det(M)}$, where M_{t-}^{s-} is the matrix obtained by removing the s -th row and t -th column of M . Thus it suffices to bound $\det(M)$ and $\det(M_{t-}^{s-})$. We here make use of a common matrix decomposition technique in the study of the polynomial method. If f has Taylor series $f(x) = \sum_{k=0}^{\infty} \frac{f^{(k)}(0)}{k!} x^k$ convergent over $[0, 1]$, then

$$\det(M) = \det \left[\sum_{k=0}^{\infty} \frac{f^{(k)}(0)}{k!} \alpha_s^k \beta_t^k \right]_{s, t \in [m]} = \det \left(\left[\alpha_s^k \cdot \frac{f^{(k)}(0)}{k!} \right]_{m \times \mathbb{N}} \times \left[\beta_t^k \right]_{\mathbb{N} \times m} \right).$$

One tool for computing determinants of the form $\det(A \times B)$, where A and B are rectangular matrices, is the Cauchy-Binet formula (See Section 2.3 for details), which gives

$$\det(M) = \sum_{0 \leq n_1 < \dots < n_m} \left(\prod_{k=1}^m \frac{f^{(n_k)}(0)}{n_k!} \right) \det \left[\alpha_s^{n_k} \right]_{s, k \in [m]} \det \left[\beta_t^{n_k} \right]_{t, k \in [m]}. \quad (1)$$

Observing that the determinants involved are effectively $(m!)$ -term polynomials in α and β , [2] then views the entire sum as a power series and applies a standard (yet technical) analysis to derive a bound on $\tau(M)$.

In this work, we extend this approach in four aspects.

Direction 1: Schur polynomials. First, we improve on the analysis of the series (1). Although this series is a rather concrete representation of $\det(M)$, the analysis of the $(m!)$ -term polynomials and infinite sum is still a strenuous task and tends not to lead to tight bounds. In this work, we make further inspections of the structure of series (1) and observe that all the determinants $\det[\alpha_s^{n_k}]_{s,k \in [m]}, \det[\beta_t^{n_k}]_{t,k \in [m]}$ have a special structure – they are known as *generalized Vandermonde matrices*. Such matrices have been extensively studied in Algebraic Combinatorics under the name *Schur polynomials*, and are central to the theory of symmetric polynomials. For the classical theory and application of Schur polynomials we point to Chapter 7 of [23]. In recent years, Schur polynomials have also found various applications in computer science, such as in quantum computation [16, 19] and geometric complexity theory [17].

One key property of Schur polynomials lies in its two equivalent definitions. The algebraic definition by Cauchy establishes connection between Schur polynomials and generalized Vandermonde matrices, while the combinatorial definition by Littlewood gives a concrete specification of the coefficients of Schur polynomials.

► **Definition 9** (Schur polynomials). *Let $m > 0$ be an integer and $\lambda_1 \leq \dots \leq \lambda_m$ be positive integers. We define the Schur polynomial s_λ on variables (u_1, \dots, u_m) by*

$$s_\lambda(u) = s_\lambda(u_1, \dots, u_m) = \frac{\det[u_i^{\lambda_k + (k-1)}]_{i,k \in [m]}}{\prod_{j>i} (u_j - u_i)}. \tag{Cauchy}$$

Equivalently, the Schur polynomial $s_\lambda(u)$ can be defined by

$$s_\lambda(u) = \sum_{T \in \mathcal{T}} \prod_{i=1}^m u_i^{t(T)_i}, \tag{Littlewood}$$

where \mathcal{T} is the set of all semi-standard Young tableaux of shape λ on alphabet $[m]$, and $t(T) \in \mathbb{N}^m$ is the type of T . (See Section 4 for the definition of Young tableaux and associated parameters.)

Based on this property, we are able to represent $\det[\alpha_s^{n_k}]$ and $\det[\beta_t^{n_k}]$ by “neater” polynomials whose nonzero coefficients are uniformly 1, and whose monomials can be enumerated using some combinatorial objects. One can thereby obtain stronger bounds through more straightforward analysis. Moreover, many known results regarding the asymptotic behavior of Schur polynomials also turn out useful in providing guidance on proof strategies.

Direction 2: Special counting matrices. We also observe that for many kernels of interest, the counting matrix M itself has a special structure. For the t -Student kernel $f = 1/(1 + x^\rho)$, its associated counting matrix $M_f = [1/(1 + \alpha_s^\rho \beta_t^\rho)]_{s,t \in [m]}$ is known as a (scaled) Cauchy matrix, as is M_{t-}^{s-} . For the Gaussian kernel $f = e^{-x}$, the associated counting matrix $M_f = [e^{\alpha_s \beta_t}]_{s,t \in [m]}$ is a Vandermonde matrix, and M_{t-}^{s-} is (a relatively simple example of) a generalized Vandermonde matrix. Closed-form formulas are known for determinants of such special matrices. One may thus get around the Cauchy-Binet expansion and bounds can be deduced via a direct argument.

Direction 3: Grouping vector pairs. Regarding the reduction per se, we show the reduction in [2] can be modified so that $\varepsilon = 1/(3n\tau(M))$ suffices, in contrast to the aforementioned $\varepsilon = 1/(3n^2\tau(M))$ lower bound. The main idea is to perform the reduction on $\{x^{(i)}\} \times Y$ for each $x^{(i)} \in X$ separately. Note that for fixed $i \in [n]$,

$$(K \times \mathbf{1})[i] = \sum_{j \in [n]} f(\|x^{(i)} - y^{(j)}\|_2^2) = \sum_{p=0}^m f(p) \cdot \#\{j : \|x^{(i)} - y^{(j)}\|_2^2 = p\}.$$

By the same argument as before, one now recovers the components $w^i = [\#\{j \in [m] : \|x^{(i)} - y^{(j)}\|_2^2 = p\}]_{p \in [m]}$ of w for respective $i \in [m]$. We note, in this new reduction, it suffices to approximate single entries $(K \times \mathbf{1})[i]$, which is arguably a simpler task than approximating $\|K \times \mathbf{1}\|_1$, as an error n times as large may accumulate in the latter case. Formalizing this idea in Section 3, we successfully shave a factor of n .

This improvement on the dependence of n in addition raises an interesting question. In the high dimensional regime, [20] showed that for any positive definite kernel f , one can compute $\text{KDE}_f(n, m, \varepsilon, B)$ in truly subquadratic time with accuracy $1/\varepsilon < n^{1-\delta}$ for any constant $\delta > 0$. Against this algorithm, our improved reduction produces a lower bound for $1/\varepsilon > \Omega(n) \cdot \tau(M)$, leaving a gap of simply $\tau(M)$. This brings within reach a potential tightness result: the optimality of [20] can be (in part) established as long as one can bound $\tau(M) = O(1)$ for some positive definite kernel f . Such a result is arguably hard to obtain from the previous lower bounds by either [6] or [2].

Direction 4: Low-dimensional BCP. To extend the negative result to the low-dimensional regime, we combine our main reduction with a variant of the BCP problem. Williams [25] and Chen [12] showed that the BCP problem for vectors with *integer entries* remains hard even in extremely low dimensions $d = 2^{O(\log^* n)}$. (See Section 2.1 for a formal statement). With slight modification, our main reduction can use KDE subroutines to recover the distance vector for not only datapoints in $\{0, 1\}^m$ but those in \mathbb{Z}^m (though a larger counting matrix is required). A hardness result for KDE in low dimensions thus follows from similar analysis. Moreover, looking into the proofs of [25] and [12], we notice that they effectively showed a stronger trade-off between the dimension of vectors and magnitude of vector entries. Translated into the setting of KDE, this is a trade-off between dimension m and approximation error ε .

1.5 Discussion

Additive vs. Relative Error. In the previous discussion we focused mainly on the KDE problem with additive error. In recent years, much effort has also been dedicated to algorithm design in the setting with relative error, primarily in the moderate to high dimensional regime $d = \Omega(\log n)$. In this setting, the running time of KDE algorithms normally depends not only on the relative error parameter ε_R but also on a lower bound of the kernel value $\mu = \min_{x \in [0, B]} f(x)$. The folklore random sampling algorithm runs in time $O(n\varepsilon_R^{-2}\mu^{-1})$. For the Gaussian kernel, Charikar and Siminelakis [10] made the first major improvement by designing a $O(n\varepsilon_R^{-2}\mu^{-0.5})$ -time algorithm using a LSH-based Importance Sampling scheme. Later Charikar et al. [8] presented an improved implementation of Importance sampling that achieves $O(n\varepsilon_R^{-2}\mu^{-0.173})$ running time. For smooth kernels (including the Rational Quadratic kernel and t -Student kernel), the first non-trivial improvement was due to Backurs et al. [5], who presented an algorithm running in $n\varepsilon_R^{-2}\text{poly log}(\mu^{-1})$ time using tree-based space partitioning techniques. Recently, Charikar, Kapralov and Waingarten [9] combined this result with the discrepancy based sampling scheme by Phillips and Tai [20] and achieved $n\varepsilon_R^{-1}\text{poly log}(\mu^{-1})$ running time, improving the dependence on ε_R .

Interestingly, our new lower bounds in Section 1.3 exhibits a sharp contrast between the additive and relative error setting. The best known KDE algorithms stated above suggest that KDE for smooth kernels are likely easier than that for Gaussian-like kernels in the

35:10 Finer-Grained Hardness of Kernel Density Estimation

relative error setting. However, comparing between Theorem 5 and Theorem 7, we observe an opposite trend. For example, for $B = O(1)$ and $m = \Theta(\log n)$, Gaussian KDE is Easy when $1/\varepsilon < m^{o(m)}$ whereas KDE for Rational Quadratic kernel and t -Student kernel are Hard even for $1/\varepsilon = 2^{\Omega(m)}$. This difference suggests that the discrepancy between two formulations is likely inherent and they should be treated with respective care.

Dynamic vs. Batched KDE. As we see from the reduction of [6] and [2], the BCP problem naturally reduces to the batched version of KDE, and thereby we take batched KDE as the primary formulation in this work for simplicity. In the literature, the KDE problem is equally often phrased in its dynamic version, e.g., in [10, 20, 9]. In the dynamic KDE problem for kernel $k(x, y)$, one is given a dataset $X \subset \mathbb{R}^m$ and a vector $w \in \mathbb{R}^n$, and asked to design a data structure \mathcal{A} that preprocesses X and outputs an approximation to the Kernel Density $\sum_{x \in X} k(x, q)w[x]$ for each query point $q \in \mathbb{R}^m$. Given a KDE data structure, one can easily build a batched KDE algorithm in time $T(\text{preprocessing}) + n \cdot T(\text{query})$. Hence any hardness result proved for batched KDE automatically holds for dynamic KDE as well. It is not clear whether there is a reduction in the reverse direction, and it remains an open problem to determine whether the batched version is strictly easier. Nonetheless, all the known algorithms including the Fast Multipole method, Polynomial method and sampling-based methods, are data structures or can be modified to solve the dynamic problem.

2 Preliminaries

2.1 SETH and known hard problems

We now introduce several variants of the Bichromatic Closest Pair problem.

► **Problem 10** (Hamming (Exact) Bichromatic Closest Pair). *Hamming-BCP(n, m): Given two sets $A = \{x^{(1)}, \dots, x^{(n)}\}, B = \{y^{(1)}, \dots, y^{(n)}\} \subset \{0, 1\}^m$, compute $\min_{i, j \in [n]} \|x^{(i)} - y^{(j)}\|_2^2$.*

► **Theorem 11** ([4]). *Assuming SETH, for every $q \in (0, 1)$, there exists $C > 0$ such that if $m > C \log n$, then Hamming-BCP(n, m) cannot be solved in time $O(n^{2-q})$ for any constant $q > 0$.*

Similarly, one can define the Hamming approximate BCP problem and its decision version.

► **Problem 12** (Hamming Approximate BCP). *Hamming-Apx-BCP(n, m, μ): Given two sets A, B as in Problem 10 as well as $\mu \in \mathbb{R}_+$, output $d \in \mathbb{R}$ such that $\min_{i, j \in [n]} \|x^{(i)} - y^{(j)}\|_2^2 \leq d \leq (1 + \mu) \min_{i, j \in [n]} \|x^{(i)} - y^{(j)}\|_2^2$.*

► **Theorem 13** ([21]). *Assuming SETH, for every $q > 0$, there exist $C > 0, \mu > 0$ such that if $m > C \log n$, then Hamming-Apx-BCP(n, m, μ) cannot be solved in time $O(n^{2-q})$ for any constant $q > 0$.*

In the low dimensional regime $m = o(\log n)$, we consider the hardness of the ℓ_2 BCP problem.

► **Problem 14** (ℓ_2 (Exact) Bichromatic Closest Pair). *ℓ_2 -BCP(n, m, D): Given two sets $A = \{x^{(1)}, \dots, x^{(n)}\}, B = \{y^{(1)}, \dots, y^{(n)}\} \in \mathbb{Z}^m$ such that $\max_{i, j \in [n]} \|x^{(i)} - y^{(j)}\|_2^2 \leq D$, compute $\min_{i, j \in [n]} \|x^{(i)} - y^{(j)}\|_2^2$.*

► **Theorem 15** ([12]). *Assuming SETH, for every $q > 0$, there exists $C_1, C_2 > 0$ such that if $m > C_1^{\log^* n}$ and $D > m^{C_2^{\log^* n} \cdot (\log n)/m}$, then ℓ_2 -BCP(n, m, D) cannot be solved in time $O(n^{2-q})$ for any constant $q > 0$.*

To unify the hardness results for BCP in different dimension regimes, we view the Hamming BCP problem as an ℓ_2 BCP problem with $D = m$. Formally, we combine Theorem 13 and Theorem 15 as follows.

► **Theorem 16.** *Assuming SETH, for every $q > 0$, there exists $C > 0$ such that ℓ_2 -BCP(n, m, D) cannot be solved in time $O(n^{2-q})$ for any constant $q > 0$ if either of the following holds: (1) $m > C \log n, D = m$, or (2) $m > C^{\log^* n}, D > m^{C^{\log^* n} \cdot (\log n)/m}$.*

2.2 Kernels of interest

In this work we focus primarily on three kernels $k(x, y) = f(\|x - y\|_2^2)$:

- Gaussian kernel $f(x) = e^{-x}$;
- Rational Quadratic kernel $f(x) = 1/(1 + x)^\sigma$ for $\sigma \geq 1$ a parameter;
- t -Student kernel $f(x) = 1/(1 + x^\rho)$ for $\rho \geq 1$ a parameter.⁷

Rational Quadratic kernel and t -Student kernel are two typical kernels with mild decay (as opposed to the rapid decay of Gaussian kernel). This property is abstracted by Backurs et al. [5] in the definition of a smooth kernel. Here we only focus on decreasing radial kernels.

► **Definition 17 (smooth kernel).** *A decreasing radial kernel $k(x, y) = f(\|x - y\|_2^2)$ is (L, t) smooth if for any $0 < a < b$, $\frac{f(a)}{f(b)} \leq L \left(\frac{b}{a}\right)^t$.*

By calculation one can verify that Rational Quadratic kernel and t -Student kernel are respectively $(1, 1)$ - and $(1, \rho)$ -smooth.

Positive definite kernels. Most commonly-studied kernels are positive definite. We will find that our lower bound approach takes a particularly nice form for such kernels.

► **Definition 18 (Positive definite kernel).** *A kernel $k : \mathbb{R}^m \times \mathbb{R}^m \rightarrow \mathbb{R}$ is positive definite if for any n points $x_1, \dots, x_n \in \mathbb{R}^m$, the Gram matrix $G = [k(x_i, x_j)]_{i, j \in [n]}$ is always positive definite.*

For radial kernels, we have the following concise characterization of positive definite kernels.

► **Definition 19.** *Let $f : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}$ be a real function. We say f is absolutely monotone if $G \in C^\infty(\mathbb{R}_{\geq 0})$ and $f^{(k)}(t)$ for all $k \in \mathbb{N}$ and $t \geq 0$, and we say f is completely monotone if $G \in C^\infty(\mathbb{R}_{\geq 0})$ and $(-1)^k \cdot f^{(k)}(t) \geq 0$ for all $k \in \mathbb{N}$ and $t \geq 0$.*

► **Theorem 20 (Schoenberg's characterization).** *Let $f : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}$ be a real function. Then the kernel $k(x, y) = f(\|x - y\|_2^2)$ is positive definite if and only if f is completely monotone on $\mathbb{R}_{\geq 0}$.*

2.3 Tools from linear algebra

As in prior work, we will make use of the Cauchy-Binet formula.

► **Lemma 21 (Cauchy-Binet formula).** *Let $k > 0$ be an integer, and for functions $A : [k] \times \mathbb{N} \rightarrow \mathbb{R}$ and $B : \mathbb{N} \times [k] \rightarrow \mathbb{R}$, define the matrix $C \in \mathbb{R}^{k \times k}$ by, for $i, j \in [k]$, $C[i, j] = \sum_{\ell=0}^\infty A[i, \ell]B[\ell, j]$. If the sum converges absolutely for all i, j , then*

$$\det(C) = \sum_{1 \leq \ell_1 < \dots < \ell_k} \det(A[\ell_1, \dots, \ell_k]) \cdot \det(B[\ell_1, \dots, \ell_k]).$$

Here $A[\ell_1, \dots, \ell_k]$ (resp. $B[\ell_1, \dots, \ell_k]$) is the $k \times k$ matrix obtained from A (resp. B) by taking the columns (resp. rows) ℓ_1, \dots, ℓ_k .

⁷ t in the name of the kernel in principle should be the name of the parameter. We here use ρ as parameter while keeping the name t -Student kernel unchanged.

3 Main reduction from BCP

Most of our new lower bounds are based on the reduction below from Hamming or ℓ_2 Exact Bichromatic Closest Pair to KDE. This reduction generalizes a framework developed in [2] to accommodate different parameter regimes. In this section we will give the outline of the reduction and establish an upper bound on $1/\varepsilon$ determined by the quantity $\tau(M)$.

► **Definition 22.** Let $D > 0$ be an integer. Fix vectors $\alpha, \beta \in \mathbb{R}^D$ and $c \in \mathbb{R}$. Suppose $c + \alpha_\ell \beta_p \in [0, 1]$ for all $\ell, p \in [D]$. Then for function $f : [0, 1] \rightarrow \mathbb{R}$, the counting matrix $M = M(D; f, \alpha, \beta)$ is a $D \times D$ matrix defined by $M[\ell, p] = f(c + \alpha_\ell \beta_p)$, $\ell, p \in [D]$.

► **Theorem 23.** Let $\alpha \in \mathbb{R}^D$ be a fixed vector and $\beta \in \mathbb{R}^D$ be the identity vector defined by $\beta_p = p$. If $\text{KDE}_f(n, m, \varepsilon)$ can be solved in $T(n, m, \varepsilon)$ time, then ℓ_2 -BCP(n, m, D) with $m = n^{o(1)}$, $D = n^{o(1)}$ can be solved in $T(n, m + 1, (3n\tau(M))^{-1}) \cdot n^{o(1)} + n^{1+o(1)}$ time, where M is the $D \times D$ counting matrix associated with f, α, β and $\tau(M) = \max_{0 \neq b \in \mathbb{R}^D} \frac{\|M^{-1}b\|_\infty}{\|b\|_\infty}$.

Proof. Given two sets $X = \{x^{(1)}, \dots, x^{(n)}\}, Y = \{y^{(1)}, \dots, y^{(n)}\} \subseteq \mathbb{Z}^m$ such that $\max_{i, j \in [n]} \|x^{(i)} - y^{(j)}\|_2^2 \leq D$, let $W \in \mathbb{N}^{D \times n}$ denote the distance count matrix defined by $W[p, i] = \#\{j \in [n] : \|x^{(i)} - y^{(j)}\|_2^2 = p\}$. Then the matrix product $U = M \times W$ gives

$$\begin{aligned} U[\ell, i] &= \sum_{p=1}^D M[\ell, p] \cdot W[p, i] = \sum_{p=1}^D f(c + \alpha_\ell \beta_p) \sum_{j \in [n]} \mathbb{1}[\|x^{(i)} - y^{(j)}\|_2^2 = p] \\ &= \sum_{j \in [n]} f\left(c + \alpha_\ell \cdot \beta \left[\|x^{(i)} - y^{(j)}\|_2^2\right]\right). \end{aligned} \quad (2)$$

We note that when β is the identity vector, the summation (2) can be formulated as a KDE instance. More specifically, let $\tilde{x}_\ell^{(i)}, \tilde{y}_\ell^{(j)} \in \mathbb{R}^{m+1}$ be defined by $\tilde{x}_\ell^{(i)}[k] = \sqrt{\alpha(\ell)}x^{(i)}[k]$ if $k \in [m]$ and $\tilde{x}_\ell^{(i)}[m+1] = \sqrt{c}$; $\tilde{y}_\ell^{(j)}[k] = \sqrt{\alpha(\ell)}y^{(j)}[k]$ if $k \in [m]$ and $\tilde{y}_\ell^{(j)}[m+1] = 0$. Then $U[\ell, i] = \sum_{j \in [n]} f(\|\tilde{x}_\ell^{(i)} - \tilde{y}_\ell^{(j)}\|_2^2)$. Therefore we have the following algorithm for ℓ_2 -BCP(n, m, D) using KDE_f as a subroutine. On input $X = \{x^{(1)}, \dots, x^{(n)}\}, Y = \{y^{(1)}, \dots, y^{(n)}\}$ such that $\max_{i, j \in [n]} \|x^{(i)} - y^{(j)}\|_2^2 \leq D$:

1. For $\ell \in [D]$, construct vectors $\tilde{x}_\ell^{(i)}, \tilde{y}_\ell^{(j)}, i, j \in [n]$. Then approximate the ℓ -th row of U : $\hat{U}[\ell] \approx U[\ell] = K_\ell \times \mathbf{1}$ using the KDE_f oracle.
2. Compute $\hat{W} = M^{-1} \times \hat{U}$, and round each entry to the closest integer.

We claim that if we call the KDE_f subroutine with $\varepsilon = (3n\tau(M))^{-1}$, then the distance count matrix W is exactly recovered after the rounding step. Indeed, for fixed $i \in [n]$, letting $W[\cdot, i], U[\cdot, i]$ respectively denote the i -th column of matrix W and U , we have

$$\|\hat{W}[\cdot, i] - W[\cdot, i]\|_\infty = \|M^{-1}(\hat{U}[\cdot, i] - U[\cdot, i])\|_\infty \leq \tau(M) \cdot \|\hat{U}[\cdot, i] - U[\cdot, i]\|_\infty.$$

If the KDE_f subroutine guarantees that $\|\hat{U}[\ell] - U[\ell]\|_\infty \leq (3n\tau(M))^{-1} \cdot \|\mathbf{1}\|_1 = (3\tau(M))^{-1}$ for all $\ell \in \{1, \dots, D\}$, then the entry-wise difference between \hat{W} and W is bounded by $1/3$.

The D calls to the subroutine then take in total $D \cdot T(n, m, (3n\tau(M))^{-1})$ time. It takes in addition $D \cdot O(nm) = n^{1+o(1)}$ operations to construct the vectors and $O(D^\omega)$ time for step 2.⁸ ◀

⁸ We always assume $f(x)$ can be exactly computed in constant time for any $x \in [0, 1]$.

We now combine the reduction above and the hardness result of BCP in Theorem 16. The hardness of the KDE problem follows.

► **Theorem 24.** *Assuming SETH, for every $q > 0$, there exists $C \geq 0$ such that $\text{KDE}_f(n, m, (3n\tau(M))^{-1})$ cannot be solved in time $O(n^{2-q})$ for any constant $q > 0$. if either of the following holds: (1) $m > C \log n, D = m$ or (2) $m > C^{\log^* n}, D > m^{C^{\log^* n} \cdot (\log n)/m}$. Here M is the $D \times D$ counting matrix in Definition 22.*

4 Schur polynomials

Let \mathbb{F} be a field. For matrices $A, B \in \mathbb{F}^{m \times n}$, let $A \circ B$ denote the Hadamard product of A and B , i.e. the $m \times n$ matrix with entries $(A \circ B)[i, j] = A[i, j] \cdot B[i, j], i \in [m], j \in [n]$. For matrix $A \in \mathbb{F}^{m \times n}$ we define the Hadamard powers $A^{\circ 1} = A$ and $A^{\circ(k+1)} = A^{\circ k} \circ A$ for integer $k \geq 1$. Similarly one can define the Hadamard product and Hadamard power for vectors $u \in \mathbb{F}^m$. Moreover, given a vector $u \in \mathbb{F}^m$ and a tuple $r = (r_1, \dots, r_n) \in \mathbb{N}^n$, we denote by $u^{\circ r}$ the $m \times n$ matrix

$$\left(\begin{array}{c|c|c|c} u^{\circ r_1} & u^{\circ r_2} & \dots & u^{\circ r_n} \\ \hline \vdots & \vdots & \ddots & \vdots \\ \hline u_m^{\circ r_1} & u_m^{\circ r_2} & \dots & u_m^{\circ r_n} \end{array} \right) = \begin{pmatrix} u_1^{r_1} & u_1^{r_2} & \dots & u_1^{r_n} \\ \vdots & \vdots & \ddots & \vdots \\ u_m^{r_1} & u_m^{r_2} & \dots & u_m^{r_n} \end{pmatrix}.$$

In particular, when $m = n$, we call $u^{\circ r}$ a generalized Vandermonde matrix. This is a natural generalization of the (usual) Vandermonde matrix $u^{\circ \delta}$ associated with the tuple $\delta = (0, 1, \dots, m-1)$.

The concept of Schur polynomials was first proposed by Cauchy and defined as the ratio of two generalized Vandermonde determinants. In what follows we denote by $\mathbb{N}_{<}^m = \{x \in \mathbb{N}^m : x_1 < \dots < x_m\}$ the set of m -tuples composed of distinct entries in ascending order, and define $\mathbb{N}_{\leq}^m = \{x \in \mathbb{N}^m : x_1 \leq \dots \leq x_m\}$ similarly.

► **Definition 25** (Cauchy's definition of Schur polynomials). *Let $m > 0$ be an integer and $\lambda = (\lambda_1, \dots, \lambda_m) \in \mathbb{N}_{\leq}^m$ an integer tuple. We define the Schur polynomial on variables (u_1, \dots, u_m) by*

$$s_\lambda(u) = s_\lambda(u_1, \dots, u_m) = \frac{\det(u^{\circ(\lambda+\delta)})}{\det(u^{\circ \delta})} = \frac{\det(u^{\circ(\lambda+\delta)})}{V(u)}.$$

Here $V(u) = \det(u^{\circ \delta}) = \prod_{1 \leq i < j \leq m} (u_j - u_i)$ is the Vandermonde determinant.

It is known that Schur polynomial has an equivalent definition by Littlewood using Young tableaux.

► **Definition 26.** *Let $m > 0$ be an integer, and $\lambda = (\lambda_1, \dots, \lambda_m) \in \mathbb{N}_{\leq}^m$ an integer tuple. A semi-standard Young tableau (SSYT) of shape λ on alphabet $[m]$ is a left-aligned two-dimensional rectangular array T of cells, with λ_i cells in the i -th row ($i \in [m]$), from bottom to top, such that*

- each cell in T is assigned with an entry from $1, \dots, m$;
- entries weakly decrease in each row, from left to right;
- entries strictly decrease in each column, from top to bottom.

Moreover, for a SSYT T , we define the type $\mathfrak{t}(T) = (t_1, \dots, t_m) \in \mathbb{N}^m$ of T , where t_j is the number of cells in T assigned with entry $j \in [m]$.

35:14 Finer-Grained Hardness of Kernel Density Estimation

► **Definition 27** (Littlewood's definition of Schur polynomials). *Let $m > 0$ be an integer. For integer tuple $\lambda = (\lambda_1, \dots, \lambda_m) \in \mathbb{N}_{\leq}^m$, the Schur polynomial $s_\lambda(u)$ can be equivalently defined by*

$$s_\lambda(u) = \sum_{T \in \mathcal{T}} u^{t(T)},$$

where \mathcal{T} is the set of all SSYT of shape λ on alphabet $[m]$.

5 Direct calculation of $\tau(M)$: Gaussian kernel and t -Student kernel

5.1 Gaussian kernel

In this section we show upper bounds for

$$\tau(M) = \max_{b \in \mathbb{R}^D: b \neq 0} \frac{\|M^{-1}b\|_\infty}{\|b\|_\infty} \leq D \cdot \max_{s, t \in [D]} |M^{-1}[s, t]|$$

where M is the counting matrix associated with the Gaussian kernel $f(x) = e^{-Bx}$.⁹ By standard facts in linear algebra, we can rewrite $|M^{-1}[s, t]| = \left| \frac{\det(M_{t-}^{s-})}{\det(M)} \right|$ with M_{t-}^{s-} the submatrix of M consisting of all entries but those in row s or column t .

For the simplicity of analysis, we study the counting matrix M of dimension $(D+1)$. A slight modification of the main reduction can employ such a matrix to recover the distance matrix with a redundant row: $W[p, i] = \#\|x^{(i)} - y^{(j)}\|_2^2 = p, i \in [n], p = \{0, 1, \dots, D\}$.

► **Theorem 28.** *Let $\alpha \in \mathbb{R}^{D+1}$ be a fixed vector and $\beta \in \mathbb{R}^{D+1}$ be the identity vector. Let M be the $(D+1) \times (D+1)$ counting matrix associated with Gaussian kernel $f = e^{-Bx}$ and α, β . Then there exists a vector α such that $\tau(M) \leq \left(\frac{5e}{1 - e^{-B/D}} \right)^D$.*

Proof. Let $x_i = \exp(-B\alpha_i)$, then $M = [x_i^j : i, j \in \{0, \dots, D\}]$ is a Vandermonde matrix with $|\det(M)| = |\det(x^{\circ(0, \dots, D)})| = V(x)$. On the other hand, we observe that M_{t-}^{s-} can be viewed as a generalized Vandermonde matrix. Making use of both the algebraic and combinatorial definition of Schur polynomials, we have¹⁰

$$|\det(M_{t-}^{s-})| = |\det((x^{s-})^{\circ(0, \dots, t-1, t+1, \dots, D)})| = V(x^{s-}) \sum_{R \subseteq \{0, \dots, D\} \setminus \{s\}} \prod_{r \in R} x^r,$$

in which x^{s-} denotes the vector $(x_0, \dots, x_{s-1}, x_{s+1}, \dots, x_D)$ and $\sum_{R \subseteq \{0, \dots, D\} \setminus \{s\}} \prod_{r \in R} x^r \leq \prod_{i \in \{0, \dots, D\} \setminus \{s\}} (1 + x_i) \leq 2^D$. Therefore

$$\max_{s, t} |\det(M_{t-}^{s-})| \leq 2^D \cdot \max_s V(x^{s-}) = 2^D \cdot V(x) \left(\min_s \prod_{i \in [D] \setminus \{s\}} |x_i - x_s| \right)^{-1}.$$

We now pick $0 = \alpha_0 < \dots < \alpha_{D-1} < \alpha_D = 1/D$ such that $|x_{i+1} - x_i| = |\exp(-B\alpha_{i+1}) - \exp(-B\alpha_i)| = \frac{1}{D}(1 - e^{-B/D})$. Let $r = \frac{1}{D}(1 - e^{-B/D})$, then

⁹ By a straightforward reduction, $\text{KDE}_f(n, m, \varepsilon, B)$ is equivalent to $\text{KDE}_g(n, m, \varepsilon, B = 1)$ where $g = f(Bx)$.

¹⁰ The Schur polynomial here in fact equals an elementary symmetric polynomial.

$$\begin{aligned} \prod_{i \in \{0, \dots, D\} \setminus \{s\}} |x_i - x_s| &= \left(\prod_{i=0}^{s-1} (s-i)r \right) \left(\prod_{i=s+1}^D (i-s)r \right) \\ &= r^D \cdot s!(D-s)! \geq r^D \cdot \left(\frac{D}{2e} \right)^{D/2.2} = \left(\frac{1 - e^{-B/D}}{2e} \right)^D. \end{aligned}$$

Combining the calculations together, we have

$$\tau(M) \leq D \cdot \max_{s,t} \left| \frac{\det(M_{t-}^{s-})}{\det(M)} \right| \leq D \cdot 2^D \left(\min_s \prod_{i \in [D] \setminus \{s\}} |x_i - x_s| \right)^{-1} \leq \left(\frac{5e}{1 - e^{-B/D}} \right)^D. \blacktriangleleft$$

The following hardness result for Gaussian KDE therefore arises from a combination of the general KDE hardness Theorem 24 and the bound on $\tau(M)$ above.

► **Theorem 29** (Hardness of Gaussian KDE). *Let $f(x) = e^{-Bx}$ be the Gaussian kernel with $B \geq 1$. Then assuming SETH, for every $q > 0$, there exists $C_1, C'_1, C_2, C_3, C'_3, C_4 \geq 0$ such that $\text{KDE}_f(n, m, \varepsilon)$ cannot be solved in time $O(n^{2-q})$ if either of the following holds:*

- (1) $m > C_1 \log n, B < C'_1 \log n, 1/\varepsilon > (C_2 m/B)^m$, or
- (2) $C_3 \log^* n < m < C'_3 (\log n)/(\log \log n), \log \log(1/\varepsilon) > (\log n)/m \cdot (\log m) \cdot C_4^{\log^* n}$.

Proof. For regime (1), first pick appropriate C'_1 so that $B/m < 1$.

$$3n\tau(M) = 3n \left(\frac{5e}{1 - e^{-B/m}} \right)^m \leq 3n \left(\frac{5e}{B/(em)} \right)^m = 3 \cdot 2^{C_1^{-1}m} \cdot \left(\frac{5e^2m}{B} \right)^m \leq \left(\frac{C_2m}{B} \right)^m$$

for some constant $C_2 > 0$. For regime (2), we have $B/D \leq 1$ and thus

$$3n \left(\frac{5e}{1 - e^{-B/D}} \right)^D \leq 3n \left(\frac{5e^2D}{B} \right)^D \leq 3n(5e^2D)^D.$$

If $m < (\log n)/(\log \log n)$, $\log D = (\log n)/m \cdot (\log m) \cdot C^{\log^* m} = (\log \log n)^2(1 - o(1)) \cdot C^{\log^* m} > \log \log n$. $\log(3n\tau(M)) \leq \log(3n) + D \log(5e^2D) < C'D \log(5e^2D)$. For some $C' > 0$. Thus, $\log \log(3n\tau(M)) \leq \log D + \log \log(5e^2D) \leq (\log n)/m \cdot (\log m) \cdot C_4^{\log^* m}$. ◀

5.2 t -Student kernel

For t -Student kernels $f(x) = 1/(1+x^p)$, we prove a similar bound on $\tau(M)$. The key observation here is that both M and M_{s-}^t are (scaled) Cauchy matrices. For vectors $a, b \in \mathbb{R}^n$, the Cauchy matrix associated with a, b is defined by $M[i, j] = 1/(a_i + b_j), i, j \in [n]$. The following closed-form formula is known for its determinant.

► **Theorem 30** (Cauchy determinant). *Let $n \geq 1$ be an integer. For $a, b \in \mathbb{R}^n$, let $M = [1/(a_i + b_j)]_{i,j \in [n]}$ be the associated Cauchy matrix. Then*

$$\det(M) = \frac{\prod_{1 \leq i < j \leq n} (a_i - a_j)(b_i - b_j)}{\prod_{i,j \in [n]} (a_i + b_j)}.$$

► **Corollary 31.** *For $a, b \in \mathbb{R}^n$ with $a_i, b_j \in (0, 1), \forall i, j \in [n]$, and $s, t \in [n]$, we have*

$$\left| \frac{\det[(1 + a_i b_j)^{-1}]_{i,j \in [n]}}{\det[(1 + a_i b_j)^{-1}]_{i \in [n]^{s-}, j \in [n]^{t-}}} \right| = \left| \frac{\prod_{i \in [n]^{s-}} (a_i - a_s) \prod_{j \in [n]^{t-}} (b_j - b_t)}{\prod_{i \in [n]} (1 + a_i b_t) \prod_{j \in [n]^{t-}} (1 + a_s b_j)} \right|.$$

Here $[n]^{s-} = [n] \setminus \{s\}, [n]^{t-} = [n] \setminus \{t\}$.

35:16 Finer-Grained Hardness of Kernel Density Estimation

Before proving the bound on $\tau(M)$, we gather several simple inequalities to be used.

► **Lemma 32.**

(I) Let $r \geq 1, a, b > 0$ be real numbers. Then $(a + b)^r \geq a^r + b^r$.

(II) Let $a, b \geq 0$ be integers. Then $a!b! \geq ((\lfloor \frac{a+b}{2} \rfloor)!)^2$.

Proof.

(I) Taking the derivative, one can show $f(x) = (1 + x)^r - x^r$ is increasing in x for $x > 0$. Thus $f(b/a) = (1 + b/a)^r - (b/a)^r \geq f(0) = 1$. The inequality follows.

(II) For $a = b = 0$, the inequality is trivial. If not, we have $\binom{a+b}{a} \leq \binom{a+b}{\lceil (a+b)/2 \rceil}$, $a!b! \geq (\lceil \frac{a+b}{2} \rceil)! (\lfloor \frac{a+b}{2} \rfloor)! \geq ((\lfloor \frac{a+b}{2} \rfloor)!)^2$. ◀

► **Theorem 33.** Let $\rho \geq 1$ be a real number. Let $\alpha \in \mathbb{R}^D$ be a fixed vector and $\beta \in \mathbb{R}^D$ be the identity vector. Let M be the $D \times D$ counting matrix associated with t -Student kernel $f(x) = 1/(1 + x^\rho)$ and α, β . Then there exists a vector α such that $\tau(M) \leq (7e)^{2\rho D}$.

Proof. The counting matrix is $M = [f(\alpha_i \beta_j)]_{i,j \in [D]} = \left[\frac{1}{1 + (\alpha_i \beta_j)^\rho} \right]_{i,j \in [D]}$. For simplicity we assume D is odd. Let $s, t \in [D]$. By corollary 31, we have

$$\left| \frac{\det(M)}{\det(M_{t-}^{s-})} \right| = \left| \frac{\prod_{i \in [D]^{s-}} (\alpha_i^\rho - \alpha_s^\rho) \prod_{j \in [D]^{t-}} (\beta_j^\rho - \beta_t^\rho)}{\prod_{i \in [D]} (1 + \alpha_i^\rho \beta_t^\rho) \prod_{j \in [D]^{t-}} (1 + \alpha_s^\rho \beta_j^\rho)} \right|.$$

We set $\alpha = \beta$ to be the scaled identity vector with $\alpha_i = \beta_i = i/D, \forall i \in [D]$. (By rescaling $\alpha_i = i/D^2, \beta_i = i$ one can make β the identity vector.) Then $\prod_{i \in [D]} (1 + \alpha_i^\rho \beta_t^\rho) \prod_{j \in [D]^{t-}} (1 + \alpha_s^\rho \beta_j^\rho) \leq 2^{2D-1}$,

$$\begin{aligned} \prod_{i \in [D]^{s-}} |\alpha_i^\rho - \alpha_s^\rho| &= \prod_{i=1}^{s-1} \frac{s^\rho - i^\rho}{D^\rho} \prod_{i=s+1}^D \frac{i^\rho - s^\rho}{D^\rho} \geq \prod_{i=1}^{s-1} \frac{(s-i)^\rho}{D^\rho} \prod_{i=s+1}^D \frac{(i-s)^\rho}{D^\rho} \\ &= \left(\frac{(s-1)!(D-s)!}{D^{D-1}} \right)^\rho \geq \left(\frac{(\lfloor \frac{D-1}{2} \rfloor)!}{D^{D-1}} \right)^\rho \geq \left(\frac{D-1}{2eD} \right)^{\rho(D-1)} \geq (3e)^{-\rho D}. \end{aligned}$$

Similarly, $\prod_{j \in [D]^{t-}} |\beta_j^\rho - \beta_t^\rho| \geq (3e)^{-\rho D}$. In conclusion, we have $\tau(M) \leq D \cdot \max_{s,t} \left| \frac{\det(M_{t-}^{s-})}{\det(M)} \right| \leq D \cdot 2^{2D} \cdot (3e)^{2\rho D} \leq (7e)^{2\rho D}$. ◀

Combining the bound on $\tau(M)$ above and the general KDE hardness Theorem 24, we obtain

► **Theorem 34** (Hardness results of t -Student KDE). Let $f(x) = 1/(1 + x^\rho)$ be a t -Student kernel parameterized by $\rho \geq 1$ an absolute constant. Then assuming SETH, for every $q > 0$, there exists $C_1, C_2, C_3, C'_3, C_4 \geq 0$ such that $\text{KDE}_f(n, m, \varepsilon)$ cannot be solved in time $O(n^{2-q})$ if either of the following holds:

(1) $m > C_1 \log n, 1/\varepsilon > n^{C_2}$, or

(2) $C_3 \log^* n < m < C'_3 (\log n) / (\log \log n)$, $\log \log(1/\varepsilon) > (\log n)/m \cdot (\log m) \cdot C_4^{\log^* n}$.

Here C_1, C_3, C'_3 are absolute constants while C_2, C_4 are dependent on ρ .

Proof. Analogous to the proof of Theorem 29. ◀

6 Cauchy-Binet Expansion

To bound $\tau(M)$ for counting matrices that are associated with more general kernels, we establish the connection between the determinant $\det(M)$ and the Taylor coefficients of f via Cauchy-Binet Expansion, based on a framework given in [2]. Our improvement mainly derives from an observation that the terms in the expansion are in fact generalized Vandermonde determinants. The additional structure in the determinants allows for simplifications in analysis through Schur polynomials.

Define infinite matrices $A : [D] \times \mathbb{N} \rightarrow \mathbb{R}$, $\tilde{A} : [D] \times \mathbb{N} \rightarrow \mathbb{R}$, and $B : \mathbb{N} \times [D] \rightarrow \mathbb{R}$ by

$$A[i, k] = \alpha_i^k, \quad \tilde{A}[i, k] = \frac{f^{(k)}(c)}{k!} \alpha_i^k, \quad B[k, j] = \beta_j^k.$$

Suppose the Taylor series of $f(c+x)$ at $c \in [0, 1]$ is *absolutely convergent* whenever $c+x \in [0, 1]$. Then for $i, j \in [D]$, assuming $c + \alpha_i \beta_j \in [0, 1]$, we have

$$M[i, j] = f(c + \alpha_i \beta_j) = \sum_{k=0}^{\infty} \frac{f^{(k)}(c)}{k!} \alpha_i^k \beta_j^k = \sum_{k=0}^{\infty} \tilde{A}[i, k] B[k, j].$$

By Cauchy-Binet formula, we expand $\det(M)$ as follows.

$$\begin{aligned} \det(M) &= \sum_{\bar{n} \in \mathbb{N}_{\leq}^D} \det \tilde{A}[[D]; \bar{n}] \cdot \det B[\bar{n}; [D]] \\ &= \sum_{\lambda \in \mathbb{N}_{\leq}^D} \left(\prod_{i=1}^D \frac{f^{(\lambda_i + \delta_i)}(c)}{(\lambda_i + \delta_i)!} \right) \det(\alpha^{\circ(\lambda + \delta)}) \det(\beta^{\circ(\lambda + \delta)}) \quad (\lambda = \bar{n} - \delta) \\ &= V(\alpha) V(\beta) \sum_{\lambda \in \mathbb{N}_{\leq}^D} \left(\prod_{i=1}^D \frac{f^{(\lambda_i + \delta_i)}(c)}{(\lambda_i + \delta_i)!} \right) s_{\lambda}(\alpha) s_{\lambda}(\beta). \end{aligned} \tag{3}$$

The last step follows from the algebraic definition of Schur polynomials. Similarly,

$$\det(M_{t-}^{s-}) = V(\alpha^{s-}) V(\beta^{t-}) \sum_{\lambda \in \mathbb{N}_{\leq}^{D-1}} \left(\prod_{i=1}^{D-1} \frac{f^{(\lambda_i + \delta_i)}(c)}{(\lambda_i + \delta_i)!} \right) s_{\lambda}(\alpha^{s-}) s_{\lambda}(\beta^{t-}). \tag{4}$$

6.1 Absolutely monotonic kernels

At this point one may try bounding (3) and (4) using analytic properties of Schur polynomials. However, for a general function f , the arbitrary signs of Taylor coefficients largely complicate the analysis, making tight bounds out of reach. Hence we first study the easy case where $\tau(M)$ is associated with an *absolutely monotonic* kernel f , i.e., $f : [0, 1] \rightarrow [0, 1]$ satisfies $f^{(n)}(c) \geq 0$ for all $n \in \mathbb{N}$ and $c > 0$. We will shortly see that such kernels are in fact not artificial as all the positive definite kernels naturally reduce to this case.

We first use the following identity of Schur polynomials to “align” (3) and (4).

► **Proposition 35.** *If $\lambda_1 \neq 0$, then $s_{(\lambda_1, \dots, \lambda_n)}(\alpha_1 = 0, \alpha_2, \dots, \alpha_n) = 0$. If $\lambda_1 = 0$, then $s_{(\lambda_1, \dots, \lambda_n)}(\alpha_1 = 0, \alpha_2, \dots, \alpha_n) = s_{(\lambda_2, \dots, \lambda_n)}(\alpha_2, \dots, \alpha_n)$.*

Proof. By the combinatorial definition of Schur polynomials, we have $s_{(\lambda_1, \dots, \lambda_n)}(\alpha_1, \alpha_2, \dots, \alpha_n) = \sum_{T \in \mathcal{T}} \alpha^{t(T)}$, where \mathcal{T} is the set of all SSYT of shape $\lambda = (\lambda_1, \dots, \lambda_n)$ on alphabet $[n]$, and $t(T)$ denotes the type of SSYT T . Now set $\alpha_1 = 0$. If the letter 1 appears in a SSYT

T , i.e. $t(T)[1] > 0$, then the corresponding monomial vanishes when evaluated. Therefore the equation simplifies to $s_{(\lambda_1, \dots, \lambda_n)}(\alpha_1, \alpha_2, \dots, \alpha_n) = \sum_{T \in \mathcal{T}_1} \alpha^{t(T)}$ where \mathcal{T}_1 is the set of all SSYT of shape λ on alphabet $\{2, 3, \dots, n\}$.

However, if $\lambda_1 > 0$, no sequence of length n on alphabet $\{2, 3, \dots, n\}$ satisfies the (strictly) decreasing constraint in the first column of the tableau. In this case $s_{(\lambda_1, \dots, \lambda_n)}(\alpha_1, \alpha_2, \dots, \alpha_n) = 0$. If $\lambda_1 = 0$, then \mathcal{T}_1 is essentially the set of all SSYT of shape $\lambda' = (\lambda_2, \dots, \lambda_n)$ over alphabet $\{2, 3, \dots, n\}$. By definition, $s_{(\lambda_1, \dots, \lambda_n)}(\alpha_1, \alpha_2, \dots, \alpha_n) = \sum_{T \in \mathcal{T}_1} \alpha^{t(T)} = s_{(\lambda_2, \dots, \lambda_n)}(\alpha_2, \dots, \alpha_n)$. \blacktriangleleft

In consequence, if we fix $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_D)$ with $\alpha_1 = 0$, then

$$\begin{aligned} \det(M) &= V(\alpha)V(\beta) \sum_{\lambda \in \mathbb{N}_{\leq}^D} \left(\prod_{i=1}^D \frac{f(\lambda_i + \delta_i)(c)}{(\lambda_i + \delta_i)!} \right) s_\lambda(\alpha) s_\lambda(\beta) \\ &= V(\alpha)V(\beta)f(c) \sum_{\lambda \in \mathbb{N}_{\leq}^{D-1}} \left(\prod_{i=1}^{D-1} \frac{f(\lambda_i + \delta_{i+1})(c)}{(\lambda_i + \delta_{i+1})!} \right) s_\lambda(\alpha^{1-}) s_\lambda(\beta^{1-}). \end{aligned}$$

Here $\lambda^{1-}, \alpha^{1-}, \beta^{1-}$ are obtained by removing the first entry in the corresponding partition/vector λ, α, β . (In the last step we abuse the notation by renaming λ^{1-} as λ .) Meanwhile,

$$\max_{s,t} \det(M_{t-}^{s-}) \leq \left(\max_{s,t} V(\alpha^{s-}) V(\beta^{t-}) \right) \sum_{\lambda \in \mathbb{N}_{\leq}^{D-1}} \left(\prod_{i=1}^{D-1} \frac{f(\lambda_i + \delta_i)(c)}{(\lambda_i + \delta_i)!} \right) s_\lambda(\alpha^{1-}) s_\lambda(\beta^{1-}).$$

In what follows, we let

$$E_\lambda = \left(\prod_{i=1}^{D-1} \frac{f(\lambda_i + \delta_{i+1})(c)}{(\lambda_i + \delta_{i+1})!} \right) s_\lambda(\alpha^{1-}) s_\lambda(\beta^{1-}), \quad F_\lambda = \left(\prod_{i=1}^{D-1} \frac{f(\lambda_i + \delta_i)(c)}{(\lambda_i + \delta_i)!} \right) s_\lambda(\alpha^{1-}) s_\lambda(\beta^{1-})$$

denote the corresponding terms in the two sums. For absolutely monotonic function f , we have $F_\lambda > 0$ and $E_\lambda > 0$ for all $\lambda \in \mathbb{N}_{\leq}^{D-1}$. Thus,

$$\max_{s,t} \frac{\det(M_{t-}^{s-})}{\det(M)} \leq \frac{1}{f(c)} \max_{s,t} \frac{V(\alpha^{s-}) V(\beta^{t-})}{V(\alpha) V(\beta)} \max_{\lambda \in \mathbb{N}_{\leq}^{D-1}} \frac{F_\lambda}{E_\lambda}. \quad (5)$$

6.2 KDE hardness for positive definite kernels

To accommodate positive definite kernels, we slightly modify the reduction in Section 3. Let $k(x, y) = f(\|x - y\|_2^2)$ be a positive definite kernel. By Schoenberg's characterization (Theorem 20), we have $(-1)^k \cdot f^{(k)}(t) \geq 0$ for all $k \in \mathbb{N}, t \geq 0$. Then for the function $g(x) = f(1 - x)$, it holds $g^{(k)}(t) = (-1)^k \cdot f^{(k)}(1 - t) \geq 0$ for all $k \in \mathbb{N}, t \in [0, 1]$. Namely g is an absolutely monotonic kernel.

Next we see how KDE_f can be related to $\tau(M_g)$. Let M be the $D \times D$ counting matrix associated with g , and let $W \in \mathbb{N}^{D \times n}$ be a re-indexed distance count matrix defined by $W[p, i] = \#\{j \in [n] : \|x^{(i)} - y^{(j)}\|_2^2 = D - p\}$. Then the matrix product $U = M \times W$ gives

$$\begin{aligned} U[\ell, i] &= \sum_{p=1}^D M[\ell, p] \cdot W[p, i] = \sum_{p=1}^D g(\alpha_\ell \beta_p) \sum_{j \in [n]} \mathbb{1} \left[\|x^{(i)} - y^{(j)}\|_2^2 = D - p \right] \\ &= \sum_{j \in [n]} g \left(c + \alpha_\ell \cdot \beta \left[D - \|x^{(i)} - y^{(j)}\|_2^2 \right] \right). \end{aligned} \quad (6)$$

If β is the identity vector, then

$$U[\ell, i] = \sum_{j \in [n]} g \left(c + \alpha_\ell \cdot (D - \|x^{(i)} - y^{(j)}\|_2^2) \right) = \sum_{j \in [n]} f \left(1 - c - \alpha_\ell D + \alpha_\ell \|x^{(i)} - y^{(j)}\|_2^2 \right).$$

Constructing vectors $\tilde{x}_\ell^{(i)}, \tilde{y}_\ell^{(j)} \in \mathbb{R}^{m+1}$ defined by $\tilde{x}_\ell^{(i)}[k] = \sqrt{\alpha(\ell)}x^{(i)}[k]$ if $k \in [m]$ and $\tilde{x}_\ell^{(i)}[m+1] = \sqrt{1-c-\alpha_\ell D}$; $\tilde{y}_\ell^{(j)}[k] = \sqrt{\alpha(\ell)}y^{(j)}[k]$ if $k \in [m]$ and $\tilde{y}_\ell^{(j)}[m+1] = 0$, we have $U[\ell, i] = \sum_{j \in [n]} f(\|\tilde{x}_\ell^{(i)} - \tilde{y}_\ell^{(j)}\|_2^2)$. In this way we show that the new product U can also be computed using KDE subroutines, and the reduction proceeds the same way as in Section 3 to recover the (re-indexed) distance count matrix. By similar analysis, we have the following hardness result relating the complexity of KDE_f to $\tau(M_g)$.

► **Theorem 36.** *Let $f : [0, 1] \rightarrow [0, 1]$ be a function and $g(x) = f(1-x)$. Then assuming SETH, for every $q > 0$, there exists $C \geq 0$ such that $\text{KDE}_f(n, m, (3n\tau(M))^{-1})$ cannot be solved in time $O(n^{2-q})$ for any constant $q > 0$. if either of the following holds: (1) $m > C \log n, D = m$ or (2) $m > C^{\log^* n}, D > m^{C^{\log^* n} \cdot (\log n)/m}$. Here M is the $D \times D$ counting matrix associated with function g .*

6.3 Hardness of Rational Quadratic kernel

For f, α, β of certain forms, we give explicit bounds on the two terms in (5). Regarding the ratio of Vandermonde determinants, we focus on $V(x^{s^-})/V(x)$ for scaled identity x defined by $x_p = p/D$.

► **Proposition 37.** *Let $\rho \geq 1$ be a real number and $x \in [0, 1]^D$ be the vector defined by $x_i = i/D$. Then for $s \in [D]$ we have $\frac{V(x^{s^-})}{V(x)} = \prod_{i \in [D] \setminus \{s\}} |x_i - x_s|^{-1} \leq (3e)^D$.*

Proof. $\prod_{i \in [D] \setminus \{s\}} |x_i - x_s| = \prod_{i=1}^{s-1} \frac{s-i}{D} \cdot \prod_{i=s+1}^D \frac{i-s}{D} = \frac{(s-1)!(D-s)!}{D^{D-1}} \geq \left(\frac{D-1}{2eD}\right)^{(D-1)} \geq (3e)^{-D}$. ◀

Hence for vectors α defined by $\alpha_\ell = \ell/D, \ell \in [D]$ and β defined by $\beta_p = p/D, p \in [D]$, we have $\max_{s,t} \frac{V(\alpha^{s^-})V(\beta^{t^-})}{V(\alpha)V(\beta)} \leq (3e)^{2D}$.

For the term involving Taylor coefficients, we fix a real number $\sigma \geq 1$ and focus on the absolutely monotonic function $f(x) = (2-x)^{-\sigma}$. By calculation, for $k \in \mathbb{N}$, $f^{(k)}(x) = (2-x)^{-(\sigma+k)} \prod_{i=0}^{k-1} (\sigma+i)$. Then for $c=0$, we have

$$\frac{F_\lambda}{E_\lambda} = \left(\prod_{i=1}^{D-1} \frac{f(\lambda_i + \delta_i)(0)}{(\lambda_i + \delta_i)!} \right) / \left(\prod_{i=1}^{D-1} \frac{f(\lambda_i + \delta_{i+1})(0)}{(\lambda_i + \delta_{i+1})!} \right) = \prod_{i=1}^{D-1} \left(2 \cdot \frac{1 + \lambda_i + \delta_i}{\sigma + \lambda_i + \delta_i} \right) \leq 2^{D-1}.$$

Combining the calculations, we obtain the following bound on $\tau(M)$.

► **Theorem 38.** *Let $\sigma \geq 1$ be a real number. Let $f : [0, 1] \rightarrow [0, 1]$ be the function $f(x) = (2-x)^{-\sigma}$, and let vectors $\alpha, \beta \in \mathbb{R}^D$ be defined by $\alpha_\ell = \ell/D, \beta_p = p/D$. Then the $D \times D$ counting matrix M associated with f, α, β has $\tau(M) \leq D \cdot 2^\sigma \cdot (3e)^{2D} \cdot 2^{D-1} \leq 2^\sigma (7e)^{2D}$.*

Combining the bound on $\tau(M)$ associated with $f(x) = (1+(1-x))^{-\sigma}$ with the hardness of positive definite KDE Theorem 36, we obtain

► **Theorem 39 (Hardness of Rational Quadratic KDE).** *Let $f(x) = 1/(1+x)^\sigma$ be a Rational Quadratic kernel parameterized by $\sigma \geq 1$ an absolute constant. Then assuming SETH, for every $q > 0$, there exists $C_1, C_2, C_3, C'_3, C_4 \geq 0$ such that $\text{KDE}_f(n, m, \varepsilon)$ cannot be solved in time $O(n^{2-q})$ for any constant $q > 0$. if either of the following holds:*

- (1) $m > C_1 \log n, 1/\varepsilon > n^{C_2}$, or
 (2) $C_3 \log^* n < m < C'_3(\log n)/(\log \log n), \log \log(1/\varepsilon) > (\log n)/m \cdot (\log m) \cdot C_4^{\log^* n}$.
 Here C_1, C_3, C'_3 are absolute constants while C_2, C_4 are dependent on σ .

Proof. Analogous to the proof of Theorem 29. ◀

References

- 1 Amol Aggarwal and Josh Alman. Optimal-Degree Polynomial Approximations for Exponentials and Gaussian Kernel Density Estimation. In *37th Computational Complexity Conference (CCC 2022)*, 2022.
- 2 Josh Alman, Timothy Chu, Aaron Schild, and Zhao Song. Algorithms and hardness for linear algebra on geometric graphs. In *2020 IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS)*, 2020. doi:10.1109/FOCS46700.2020.00057.
- 3 Josh Alman and Zhao Song. Fast attention requires bounded entries. In *NeurIPS*, 2023.
- 4 Josh Alman and Ryan Williams. Probabilistic polynomials and hamming nearest neighbors. *2015 IEEE 56th Annual Symposium on Foundations of Computer Science*, October 2015. doi:10.1109/focs.2015.18.
- 5 Arturs Backurs, Moses Charikar, Piotr Indyk, and Paris Siminelakis. Efficient density evaluation for smooth kernels. In *2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 615–626, 2018. doi:10.1109/FOCS.2018.00065.
- 6 Arturs Backurs, Piotr Indyk, and Ludwig Schmidt. On the fine-grained complexity of empirical risk minimization: Kernel methods and neural networks. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL: https://proceedings.neurips.cc/paper_files/paper/2017/file/635440afdfc39fe37995fed127d7df4f-Paper.pdf.
- 7 Arturs Backurs, Piotr Indyk, and Tal Wagner. Space and time efficient kernel density estimation in high dimensions. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, Red Hook, NY, USA, 2019. Curran Associates Inc.
- 8 Moses Charikar, Michael Kapralov, Navid Nouri, and Paris Siminelakis. Kernel density estimation through density constrained near neighbor search. In *2020 IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS)*, pages 172–183, 2020. doi:10.1109/FOCS46700.2020.00025.
- 9 Moses Charikar, Michael Kapralov, and Erik Waingarten. A quasi-monte carlo data structure for smooth kernel evaluations. In *Proceedings of the 2024 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 5118–5144, 2024. doi:10.1137/1.9781611977912.184.
- 10 Moses Charikar and Paris Siminelakis. Hashing-based-estimators for kernel density in high dimensions. In *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 1032–1043, 2017. doi:10.1109/FOCS.2017.99.
- 11 Moses Charikar and Paris Siminelakis. Multi-resolution hashing for fast pairwise summations. In *2019 IEEE 60th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 769–792, 2019. doi:10.1109/FOCS.2019.00051.
- 12 Lijie Chen. On the hardness of approximate and exact (bichromatic) maximum inner product. *Theory of Computing*, 16(4):1–50, 2020. doi:10.4086/toc.2020.v016a004.
- 13 Yen-Chi Chen. A tutorial on kernel density estimation and recent advances. *Biostatistics & Epidemiology*, 1(1):161–187, 2017.
- 14 L Greengard and V Rokhlin. A fast algorithm for particle simulations. *Journal of Computational Physics*, 73(2):325–348, 1987. doi:10.1016/0021-9991(87)90140-9.
- 15 Leslie Greengard and John A. Strain. The fast gauss transform. *SIAM J. Sci. Comput.*, 12:79–94, 1991. URL: <https://api.semanticscholar.org/CorpusID:3145209>.

- 16 Sean Hallgren, Alexander Russell, and Amnon Ta-Shma. Normal subgroup reconstruction and quantum computation using group representations. In *Proceedings of the Thirty-Second Annual ACM Symposium on Theory of Computing*, STOC '00, pages 627–635, New York, NY, USA, 2000. Association for Computing Machinery. doi:10.1145/335305.335392.
- 17 Christian Ikenmeyer and Greta Panova. Rectangular kronecker coefficients and plethysms in geometric complexity theory. In *2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 396–405, 2016. doi:10.1109/FOCS.2016.50.
- 18 Krikamol Muandet, Kenji Fukumizu, Bharath Sriperumbudur, Bernhard Schölkopf, et al. Kernel mean embedding of distributions: A review and beyond. *Foundations and Trends® in Machine Learning*, 10(1-2):1–141, 2017.
- 19 Ryan O'Donnell and John Wright. Quantum spectrum testing. In *Proceedings of the Forty-Seventh Annual ACM Symposium on Theory of Computing*, STOC '15, pages 529–538, New York, NY, USA, 2015. Association for Computing Machinery. doi:10.1145/2746539.2746582.
- 20 Jeff M. Phillips and Wai Ming Tai. Near-optimal coresets of kernel density estimates. *Discrete Comput. Geom.*, 63(4):867–887, June 2020. doi:10.1007/s00454-019-00134-6.
- 21 Aviad Rubinfeld. Hardness of approximate nearest neighbor search. *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, June 2018. doi:10.1145/3188745.3188916.
- 22 Bernhard Schölkopf and Alexander J Smola. *Learning with kernels: support vector machines, regularization, optimization, and beyond*. MIT press, 2002.
- 23 Richard Stanley. *Enumerative Combinatorics*. Cambridge Studies in Advanced Mathematics. Cambridge University Press, 2 edition, 2023.
- 24 Paraskevas Syminelakis. *Fast kernel evaluation in high dimensions: Importance sampling and near neighbor search*. PhD thesis, Stanford University, 2019.
- 25 Ryan Williams. On the difference between closest, furthest, and orthogonal pairs: Nearly-linear vs barely-subquadratic complexity. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, 2018.