Optimal-Degree Polynomial Approximations for Exponentials and Gaussian Kernel Density Estimation

Amol Aggarwal[*] Josh Alman[†]

May 13, 2022

Abstract

For any real numbers $B \geq 1$ and $\delta \in (0, 1)$ and function $f : [0, B] \to \mathbb{R}$, let $d_{B;\delta}(f) \in \mathbb{Z}_{>0}$ denote the minimum degree of a polynomial $p(x)$ satisfying $\big|p(x) - f(x)\big| < \delta$ for each $x \in [0, B]$. In this paper, we provide precise asymptotics for $d_{B;\delta}(e^{-x})$ and $d_{B;\delta}(e^{x})$ in terms of both B and δ, improving both the previously known upper bounds and lower bounds. In particular, we show that

$$d_{B;\delta}(e^{-x}) = \Theta\left(\max\left\{\sqrt{B\log(\delta^{-1})}, \frac{\log(\delta^{-1})}{\log(B^{-1}\log(\delta^{-1}))}\right\}\right), \text{ and}$$

$$d_{B;\delta}(e^{x}) = \Theta\left(\max\left\{B, \frac{\log(\delta^{-1})}{\log(B^{-1}\log(\delta^{-1}))}\right\}\right),$$

and we explicitly determine the leading coefficients in most parameter regimes.

Polynomial approximations for e^{-x} and e^{x} have applications to the design of algorithms for many problems, including in scientific computing, graph algorithms, machine learning, and statistics. Our degree bounds show both the power and limitations of these algorithms.

We focus in particular on the Batch Gaussian Kernel Density Estimation problem for n sample points in $\Theta(\log n)$ dimensions with error $\delta = n^{-\Theta(1)}$. We show that the running time one can achieve depends on the square of the diameter of the point set, B, with a transition at $B = \Theta(\log n)$ mirroring the corresponding transition in $d_{B;\delta}(e^{-x})$:

- When $B = o(\log n)$, we give the first algorithm running in time $n^{1+o(1)}$.
- When $B = \kappa \log n$ for a small constant $\kappa > 0$, we give an algorithm running in time $n^{1+O(\log\log\kappa^{-1}/\log\kappa^{-1})}$. The $\log\log\kappa^{-1}/\log\kappa^{-1}$ term in the exponent comes from analyzing the behavior of the leading constant in our computation of $d_{B;\delta}(e^{-x})$.
- When $B = \omega(\log n)$, we show that time $n^{2-o(1)}$ is necessary assuming SETH.

[*] Department of Mathematics at Columbia University and Institute for Advanced Study. amolaggarwal@math.columbia.edu. Partially supported by NSF grants DGE-1144152 and DMS-1664619, a Harvard Merit/Graduate Society Term-time Research Fellowship, and a Clay Research Fellowship.

[†] Department of Computer Science at Columbia University. josh@cs.columbia.edu. Partially supported by a Harvard Michael O. Rabin postdoctoral fellowship.

1 Introduction

Polynomial approximations of important functions play a key role in many areas of computer science and mathematics. We measure the extent to which a function can be approximated by a degree d polynomial as follows.

Definition 1.1. For any real numbers $B \geq 1$ and $\delta \in (0,1)$, and function $f : [0,B] \to \mathbb{R}$, let $d_{B;\delta}(f) \in \mathbb{Z}_{>0}$ denote the minimum degree of a non-constant polynomial $p(x)$ satisfying

$$\sup_{x \in [0,B]} \left| p(x) - f(x) \right| < \delta.$$

Past work in polynomial approximation theory has typically focused on the case when $B = O(1)$; see, for example, [Tim94, Chapter 7]. However, recent computer science applications have motivated studying the setting where both B and δ^{-1} are growing simultaneously. Indeed, in algorithmic applications, both the magnitude of the input to the function f and the tolerance for error can scale with the size of the input to the problem.

In this paper, we focus specifically[1] on the functions e^x and e^{-x}. As we will discuss more shortly, polynomial approximations for these functions appear naturally in computational problems throughout scientific computing, graph algorithms, machine learning, statistics, and many other areas. Precisely determining $d_{B;\delta}(e^{-x})$ and $d_{B;\delta}(e^x)$ is particularly important since in a number of algorithmic applications, such as the batch Gaussian Kernel Density Estimation that we discuss in Section 1.2 below, these quantities appear in the exponent of the input size in the running time. In these settings, logarithmic or even constant factors can be the difference between a fast or a trivially slow running time (see especially Sections 1.2.1 and 1.2.3 below). The standard framework of approximation theory (e.g., [Pow67, Tim94, Tre13]) can be used to deduce bounds on $d_{B;\delta}$ that are typically suboptimal, often losing (at least) such logarithmic factors, especially in the regime when B is large.

Our main results are tight asymptotics, including the exact leading constant in most parameter regimes (see Remark 1.5 below), for both $d_{B;\delta}(e^{-x})$ and $d_{B;\delta}(e^x)$.

In what follows, we define the function

$$G(x) = \sqrt{x^2+1} + x \log\left(\sqrt{x^2+1} - x\right), \tag{1}$$

for each $x \in \mathbb{R}_{\geq 0}$.

Theorem 1.2 (Approximate degree of e^{-x})**.** *Let $B \geq 1$ and $\delta \in (0,1)$. Then,*

$$d_{B;\delta}(e^{-x}) = \Theta\left(\max\left\{\sqrt{B\log(\delta^{-1})}, \frac{\log(\delta^{-1})}{\log(B^{-1}\log(\delta^{-1}))}\right\}\right).$$

More precisely, we have the following asymptotics as $B + \delta^{-1}$ tends to ∞.

1. *If $B = o\big(\log(\delta^{-1})\big)$, then $d_{B;\delta}(e^{-x}) = \left(\dfrac{\log\left(\delta^{-1}\right)}{\log\left(B^{-1}\log(\delta^{-1})\right)}\right)\big(1+o(1)\big)$.*

2. *If $B = 2r\log(\delta^{-1})$ for fixed $r > 0$, then $d_{B;\delta}(e^{-x}) = \big(\nu r + o(1)\big)\log(\delta^{-1})$, where $\nu = \nu(r) > 0$ is the unique positive solution*[2] *to the equation $G(\nu) = 1 - r^{-1}$.*

[1] We mention, however, that the method used in this paper is quite general and is expected to more broadly apply for functions f whose Taylor series coefficients decay sufficiently quickly.

[2] The uniqueness of this solution (and the ones to be mentioned below) follows from the facts that $G(0) = 1$, $\lim_{z\to\infty} G(z) = -\infty$, and $G'(z) < 0$ for $z > 0$.

3. *If $B = \omega\big(\log(\delta^{-1})\big)$ and $B \leq \delta^{-o(1)}$, then $d_{B;\delta}(e^{-x}) = \big(1 + o(1)\big)\sqrt{B\log(\delta^{-1})}$.*

4. *If $B \geq \delta^{-\Omega(1)}$, then $d_{B;\delta}(e^{-x}) = \Theta\left(\sqrt{B\log\left(\delta^{-1}\right)}\right)$.*

Theorem 1.3 (Approximate degree of e^x)**.** *Let $B \geq 1$ and $\delta \in (0, 1)$. Then,*

$$d_{B;\delta}(e^x) = \Theta\left(\max\left\{B, \frac{\log(\delta^{-1})}{\log(B^{-1}\log(\delta^{-1}))}\right\}\right).$$

More precisely, we have the following asymptotics as $B + \delta^{-1}$ tends to ∞.

1. *If $B = o\big(\log(\delta^{-1})\big)$, then $d_{B;\delta}(e^x) = \left(\frac{\log\left(\delta^{-1}\right)}{\log\left(B^{-1}\log(\delta^{-1})\right)}\right)\big(1 + o(1)\big)$.*

2. *If $B = 2r\log(\delta^{-1})$ for fixed $r > 0$, then $d_{B;\delta}(e^x) = \big(\mu r + o(1)\big)\log(\delta^{-1})$, where $\mu = \mu(r) > 0$ is the unique positive solution to the equation $G(\mu) = -1 - r^{-1}$.*

3. *If $B = \omega\big(\log\left(\delta^{-1}\right)\big)$, then $d_{B;\delta}(e^x) = \frac{z_* B}{2}\big(1 + o(1)\big)$, where $z_* \approx 2.2334$ denotes the unique positive solution to the equation $G(z_*) = -1$.*

Remark 1.4. Polynomials achieving the degree upper bounds stated in Theorems 1.2 and 1.3 can be constructed in poly(d) time, with coefficients which are rational numbers with poly(d)-bit integer numerators and denominators, where d is the degree.

Remark 1.5. In the fourth case of Theorem 1.2, we do not determine the leading constant $A = A(B;\delta)$ for which $d_{B;\delta}(e^{-x}) = \big(A + o(1)\big)\sqrt{B\log\left(\delta^{-1}\right)}$; as we will see below, when $\delta = o(1)$, we only bound it between $\frac{1}{2} \leq A \leq 1$. It is unclear to us whether or not this constant would admit a concise description in this parameter regime, especially in the case when δ is fixed as B tends to ∞. In all other parameter regimes of Theorem 1.2, and in every case of Theorem 1.3, we determine the exact leading constant.[3]

Previous bounds. The question of providing tight bounds on $d_{B;\delta}(e^{-x})$ was posed in works of Orecchia, Sachdeva, and Vishnoi [OSV12, Sections 4 and 7], and Sachdeva and Vishnoi [SV14, Section 5]. They were motivated by algorithmic applications, as [OSV12] showed how upper bounds on $d_{B;\delta}(e^{-x})$ can be used to design faster algorithms for the Balanced Separator problem from spectral graph theory. They gave an upper bound of $d_{B;\delta}(e^{-x}) \leq O(\sqrt{\max\{\log(\delta^{-1}), B\}} \cdot \log^{3/2}(\delta^{-1}))$, and a lower bound of $d_{B;\delta}(e^{-x}) \geq \frac{1}{2}\sqrt{B}$. Later, [SV14] improved the upper bound to $d_{B;\delta}(e^{-x}) \leq O(\sqrt{\max\{\log(\delta^{-1}), B\}} \cdot \log^{1/2}(\delta^{-1}))$ (noting that such a bound was also implicit in [HL97]). Theorem 1.2 provides precise asymptotics for $d_{B;\delta}(e^{-x})$, thereby answering the above question.

In particular, Theorem 1.2 shows that the prior upper bound could be improved by a logarithmic factor in some parameter regimes, but was otherwise asymptotically tight. For the Balanced Separator problem studied by [OSV12], where the running time depends polynomially on $d_{B;\delta}(e^{-x})$, this rules out a big improvement without a new approach. For other applications where the running time has an exponential dependence on $d_{B;\delta}(e^{-x})$, our improvements have more significant implications. For instance, as we discuss below in Section 1.2.1, in some parameter regimes of the batch Gaussian Kernel Density Estimation problem, our Theorem 1.2 yields a near linear time algorithm,

[3] It is quickly verified that the constants ν, μ, and z_* from Theorem 1.2 and Theorem 1.3 satisfy $2r^{-1/2} \leq \nu \leq \max\left\{r^{-1}, e\right\}$ and $z_* \leq \mu \leq \max\left\{r^{-1}, e\right\}$ for all $r > 0$.

whereas applying instead the prior bound of [SV14] would only yield a trivial quadratic running time.

We are unaware of prior work which specifically bounded $d_{B;\delta}(e^x)$, although one could apply standard results on Chebyshev interpolation (such as [Tre13, Theorem 8.2]) with some work to yield a bound $d_{B;\delta}(e^x) \geq \Omega(\max\{B, \log(\delta^{-1})\})$.

Phase transitions. In fact, Theorem 1.2 and Theorem 1.3 indicate that the dependence of the optimal degrees $d_{B;\delta}(e^{-x})$ and $d_{B;\delta}(e^x)$ on the parameters B and δ is quite intricate. First, their orders of magnitudes both exhibit transitions depending on the relative sizes of B and $\log(\delta^{-1})$. For example, when $B = \omega\big(\log(\delta^{-1})\big)$, Theorem 1.2 shows that $d_{B;\delta}(e^{-x})$ exhibits square root dependence on both $\log(\delta^{-1})$ and B, but when $B = o\big(\log(\delta^{-1})\big)$ it exhibits nearly linear dependence on $\log(\delta^{-1})$ and only logarithmic dependence on B. Second, in the "critical regime" $B = 2r\log(\delta^{-1})$, Theorem 1.2 shows that $d_{B;\delta}(e^{-x}) = \Theta\big(\log(\delta^{-1})\big)$, whose implicit constant is obtained by solving the transcendental equation $G(z) = 1 - r^{-1}$. A similar transition (with a transcendental leading constant in the critical regime) is shown for the approximating degrees of e^x in Theorem 1.3, but with the qualitative difference that $d_{B;\delta}(e^x)$ is linear in B (to leading order) and independent of δ, for $B = \omega\big(\log(\delta^{-1})\big)$.

To our knowledge, this is the first appearance of a transition arising when one simultaneously scales B and δ in the context of polynomial approximation theory. Indeed, as mentioned previously, prior works in this direction typically analyzed the case $B = O(1)$, where transitions like these are not visible. As we will explain in Section 1.1 below, these behaviors for $d_{B;\delta}(e^{-x})$ and $d_{B;\delta}(e^x)$ will have algorithmic interpretations. For example, we will see that the estimates on $d_{B;\delta}(e^{-x})$ provided in Theorem 1.2 imply a fine-grained computational phase transition for Gaussian Kernel Density Estimation in certain parameter regimes.

Previous methods. In the theoretical computer science literature, proofs of upper and lower bounds on the approximate degree $d_{B;\delta}(f)$ of a function f had been typically based on two distinct arguments [NS94, Shi02, AS04, Amb05, OSV12, SV14, BT15]. Upper bounds were often shown by providing an explicit polynomial approximation for f, usually given by (a truncation of) the expansion of f in the basis of Chebyshev polynomials. Lower bounds were typically shown by making use of an estimate, such as Markov Brothers' inequality, that constrains the maximum derivative of a bounded polynomial in terms of its degree. Both ideas are archetypes of classical approximation theory; see [Tim94, Chapters 2 and 4].

Our methods. As above, to upper bound $d_{B;\delta}(f)$ we will explicitly provide an approximating polynomial for f, obtained from the Chebyshev expansion of its rescale $f_B(x) = f\big(\frac{B}{2}(1-x)\big)$ (whose domain is now $[-1, 1]$). However, derivative-degree estimates such as Markov's inequality that prior works used to lower bound $d_{B;\delta}(f)$ usually become insensitive to the tolerance parameter δ once it passes below a (typically non-optimal) threshold. Thus, they will not suffice for our purposes of pinpointing the precise asymptotic behavior of $d_{B;\delta}(f)$.

We therefore proceed differently, by instead again making use of the Chebyshev expansion of $f_B(x)$. In particular, we use the orthogonality of the Chebsyhev polynomials to lower bound the minimal distance from f_B to a polynomial of degree d in terms of the series coefficients of f_B when expanded in the Chebyshev basis; see Proposition 2.2 below. Thus, bounds on these series coefficients can be used to bound $d_{B;\delta}(f)$. This idea was also ubiquitous in the traditional theory and practice of approximating polynomials; for instance, it was very fruitful in proving the classical sharp estimates [Tim94, Chapter 7.8 (22)] on $d_{B;\delta}(e^{-x})$ and $d_{B;\delta}(e^{-x})$ when $B = O(1)$.

However, to our understanding, this idea has not been implemented before in our context where

B and δ scale jointly (either in the computer science or approximation theory literature). In this setting, we must study the limiting behaviors for the high-degree coefficients in the Chebyshev expansion f_B, simultaneously as the degree and as B tend to ∞; see Proposition 2.4 below. This analysis becomes more involved than in the case $B = O(1)$, as it should in order to give rise to the intricate asymptotic phenomena described in Theorem 1.2 and Theorem 1.3. In particular, the phase transitions observed in those results can be traced to corresponding phase transitions for these series coefficients, given in Lemma 2.6 below.

1.1 Algorithmic Applications

Polynomial approximations with low error have numerous applications throughout algorithm design and complexity theory; see, for instance, the introduction of the survey by Sachdeva and Vishnoi [SV14] for an overview. The quantities $d_{B;\delta}(e^{-x})$ and $d_{B;\delta}(e^{x})$, in particular, play a central role in many algorithms due to the prevalence of exponential functions. Some examples include:

- **Approximating matrix exponentials.** Given a matrix A and a vector v, approximate $e^A \cdot v$. One of the most common algorithms in theory and in practice for this problem is the Lanczos method [Lan50], whose running time is bounded by $O(d \cdot m_A + d^2)$ [MMS18], where m_A is the amount of time required to do a matrix-vector multiplication by A, and $d = d_{B;\delta}(e^x)$ is the approximate degree which we compute in Theorem 1.3 with $B = ||A||$ and δ is the desired approximation error parameter.

- **Finding balanced separators in graphs.** The aforementioned work by Orecchia, Sachdeva and Vishnoi [OSV12] uses polynomial approximations of e^{-x} in a way similar to the Lanczos method to give fast, practical algorithms for the Balanced Separator problem.[4]

- **Estimating softmax.** Many "multinomial classification" problems in natural language processing and other areas make use of the *softmax* function to convert vectors representing the different classes into estimated probabilities. Given n vectors $w_1, \dots, w_n \in \mathbb{R}^m_{\geq 0}$, an index $i \in [n]$ and a sample vector $h \in \mathbb{R}^m_{\geq 0}$, softmax is defined as

$$\operatorname{softmax}(h, i, w_1, \dots, w_n) := \frac{e^{\langle w_i, h\rangle}}{\sum_{j=1}^{n} e^{\langle w_j, h\rangle}}.$$

  Training models in these applications frequently requires many softmax computations, and so approximations of softmax which are faster to compute are often used [CGA15]. Replacing the exponentials in softmax by the optimal polynomial approximations we give in Theorem 1.3 can be used to more quickly compute such approximations [JCG$^+$17, NSGH14].

- **Kernel methods.** Polynomial approximations for e^{-x} have been used to design faster sketching and estimation techniques for Gaussian kernels, including in a number of recent algorithms; see e.g. [YDGD03, LLM$^+$19, ACSS20, AKK$^+$20]. In Section 1.2 below, we show a new application along these lines to batch Gaussian Kernel Density Estimation.

1.2 Gaussian Kernel Density Estimation

Kernel Density Estimation (KDE) is one of the most common methods for non-parametric estimation of the density of an unknown distribution $\mathcal{D}$. Given a set $P \subset \mathbb{R}^m$ of samples from $\mathcal{D}$, along

[4] They also give a faster algorithm in some special cases using *rational approximations* of e^{-x}.

with a weight $w_y \in \mathbb{R}$ for each $y \in P$, the kernel density function (KDF) of P at a point $x \in \mathbb{R}^m$ is given by

$$KDF_P(x) := \sum_{y \in P} w_y \cdot k(x, y),$$

where $k : \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}$ is a carefully chosen *kernel function.* In most applications, one would like to compute KDF_P at many points x. Perhaps the most commonly studied kernel function is the *Gaussian kernel* $k(x, y) = e^{-\|x-y\|_2^2}$. This motivates the question:

Problem 1.6 (Batch Gaussian KDE)**.** Given as input $2n$ points $x^{(1)}, \dots, x^{(n)}, y^{(1)}, \dots, y^{(n)} \in \mathbb{R}^m$ which implicitly define the matrix $K \in \mathbb{R}^{n \times n}$ by $K[i, j] = e^{-\|x^{(i)}-y^{(j)}\|_2^2}$, as well as a vector $w \in \mathbb{R}^n$, and an error parameter $\delta > 0$, compute an approximation to $K \cdot w$, meaning, output a vector $v \in \mathbb{R}^n$ such that $\|K \cdot w - v\|_\infty \leq \delta \cdot \|w\|_1$.

Polynomial method algorithm. This problem can be solved by using a polynomial approximation to e^{-x} in order to construct a low-rank approximation to the matrix K, as follows. Let $B \geq 1$ and $\delta \in (0, 1)$ denote real numbers, and suppose $p(z)$ is a univariate polynomial of degree $d \geq d_{B;\delta}(e^{-x})$ such that

$$\sup_{z \in [0,B]} \left| p(z) - e^{-z} \right| \leq \delta.$$

Thus, for $x, y \in \mathbb{R}^m$ with $\|x-y\|_2^2 \leq B$, the polynomial $p(\sum_{\ell=1}^m (x_\ell - y_\ell)^2)$ outputs a value within an additive δ of $e^{-\|x-y\|_2^2}$. Hence, to solve Batch Gaussian KDE, it suffices to output the vector $\tilde{K} \cdot w$, where $\tilde{K} \in \mathbb{R}^{n \times n}$ is the matrix given by $\tilde{K}[i, j] = p(\sum_{\ell=1}^m (x_\ell^{(i)} - y_\ell^{(j)})^2)$. By a standard argument, the rank of $\tilde{K}$ is at most the number of monomials in the expansion of $p(\sum_{\ell=1}^m (x_\ell - y_\ell)^2)$, which is bounded above by $M \leq \binom{2d+2m}{2d}$, and the corresponding low rank expression for $\tilde{K}$ can be found in time $O(n \cdot M \cdot m)$.

In other words, whenever $M < n^{o(1)}$, we can solve Batch Gaussian KDE in deterministic $n^{1+o(1)}$ time in this way (see also [ACSS20, Section 5.3] where this approach was previously laid out). Theorem 1.2 characterizes exactly when this is possible in terms of m (the dimension of the points), B (the square of the diameter of the point set), and δ (the error parameter):

Corollary 1.7. *For any positive integer $m < n^{o(1)}$, and real numbers $B \geq 1$ and $\delta \in (0, 1)$, define $d = d_{B;\delta}(e^{-x})$ as in Theorem 1.2. Then, batch Gaussian KDE can be solved in deterministic time $n^{1+o(1)}$ whenever $\binom{2d+2m}{2d} < n^{o(1)}$. Similarly, if $\binom{2d+2m}{2d} < n^c$ for some constant $0 < c < 1$, then batch Gaussian KDE can be solved in truly subquadratic deterministic time $n^{1+c+o(1)}$.*

1.2.1 Comparison with prior work.

The previous best known algorithm for Batch Gaussian KDE is due to recent work of Charikar and Siminelakis [CS17], which showed how to solve this problem in randomized time $\delta^{-2} n^{1+o(1)} \cdot (\log n)^{O(B^{2/3})}$ for any dimension $m < n^{o(1)}$. Their algorithm achieves randomized running time $n^{1+o(1)}$ whenever $\delta^{-1} < n^{o(1)}$, $B < o((\log n / \log\log n)^{3/2})$, and $m < n^{o(1)}$.

Focusing on the setting[5] where $m = O(\log n)$, Corollary 1.7 achieves deterministic running time $n^{1+o(1)}$ in all the same parameter settings as the previous algorithm, and also new settings including:

[5] Often, depending on the desired error guarantees, one can reduce to roughly this case using dimensionality reduction like the Johnson–Lindenstrauss lemma.

- When $B = o(\log^2 n)$ and $\delta^{-1} < n^{o(\log n/B)}$ (slightly improving the parameter B), or
- When $B = o(\log n)$ and $\delta^{-1} = n^{\Theta(1)}$ (considerably improving the parameter δ).

In particular, the latter setting enables us to take δ to depend polynomially in n, while still retaining an $n^{1+o(1)}$ running time.

The above parameter regimes are also where our upper bound on $d_{B;\delta}(e^{-x})$ from Theorem 1.2 logarithmically improves on the one given in [SV14]. This improvement was in fact necessary for our application to KDE, as the estimate from [SV14] was $d_{B;\delta}(e^{-x}) = \Omega(\log n)$ in these settings, which would give a running time of $n \cdot \binom{2d+2m}{2d} \geq n^{1+\Omega(1)}$, as opposed to our near-linear one.

Interestingly, our algorithm and that of [CS17] take approaches which rely on very different properties of the kernel function k. Charikar and Siminelakis's algorithm uses a clever Locality-Sensitive Hashing-based approach, and also works well for other kernels with efficient hash functions, whereas our approach instead requires k to have a low-degree polynomial approximation. Other popular algorithmic techniques for KDE, such as the Fast Multipole Method [GR87], or core-sets [AHPV$^+$05, Phi13], lead to $n^{1+\Omega(1)}$ running times in the high-dimensional $d = \Omega(\log n)$, low-error $\varepsilon < n^{-\Theta(1)}$ setting; see [Sym19, Section 1.3.2] for an overview of these known approaches.

1.2.2 SETH lower bound.

To complement Corollary 1.7, we also show a fine-grained lower bound, that assuming the Strong Exponential Time Hypothesis (SETH), when $m = \Theta(\log n)$ and $\delta^{-1} = n^{\Theta(1)}$, one cannot achieve running time $n^{1+o(1)}$ when $B = \Omega(\log n)$.

Proposition 1.8. *Assuming SETH, for every $q > 0$, there are constants $\alpha, \beta, \kappa > 0$ such that Batch Gaussian KDE in dimension $m = \alpha \log n$ and error $\delta = n^{-\beta}$ for input points whose diameter squared is at most $B = \kappa \log n$ requires time $\Omega(n^{2-q})$.*

The proof of Proposition 1.8 is a slight modification of a similar lower bound of Backurs, Indyk, and Schmidt [BIS17], which relates Gaussian KDE to nearest neighbor search (for which SETH lower bounds are already known [Rub18]).

To summarize, in the natural setting where $m = \Theta(\log n)$ and $\delta^{-1} = n^{\Theta(1)}$:

- Our algorithm using the polynomial method achieves running time $n^{1+o(1)}$ when $B = o(\log n)$.
- Assume SETH. It is not possible to improve our algorithm to achieve running time $n^{1+o(1)}$ when $B = \Theta(\log n)$. Moreover, if $B = \omega(\log n)$, then no algorithm achieves running time faster than $n^{2-o(1)}$.

1.2.3 Critical regime behavior from the leading constant in Theorem 1.2.

Thus, assuming SETH (and under the setting $m = \Theta(\log n)$ and $\delta^{-1} = n^{\Theta(1)}$), the complexity of Batch Gaussian KDE exhibits a transition mirroring the one exhibited by $d_{B;\delta}(e^{-x})$ in Theorem 1.2. More specifically, as B goes from $o(\log n)$ to $\omega(\log n)$, this complexity transitions from $n^{1+o(1)}$ to $n^{2-o(1)}$. In the "critical regime" where $B = \kappa \log n$ for some fixed $\kappa > 0$, this suggests that its complexity should grow as $n^{1+\varphi(\kappa)}$, for some non-decreasing function $\varphi : \mathbb{R}_{>0} \to \mathbb{R}_{>0}$ satisfying $\lim_{\kappa \to 0} \varphi(\kappa) = 0$ and $\lim_{\kappa \to \infty} \varphi(\kappa) = 1$. It would be fascinating to better understand more precise behavior of this function φ. Does it continuously transition from 0 to 1 as κ increases, or does it admit a sudden "jump" at a specific threshold value for κ?

While these questions remain open, we can use our asymptotics for $d_{B;\delta}(e^{-x})$ to provide bounds on $\varphi(\kappa)$ for small κ. In particular, the below corollary implies that $\varphi(\kappa) = O\big(\log\log\kappa^{-1}/\log\kappa^{-1}\big)$. Our derivation of the term $\log\log\kappa^{-1}/\log\kappa^{-1}$ appearing in the exponent makes use of the leading constant ν (defined in the second part of Theorem 1.2) for the asymptotics of $d_{B;\delta}(e^{-x})$. Indeed, this is the case of Corollary 1.7 where $d = d_{B;\delta}(e^{-x}) = \Theta(\log n)$ and $m = O(\log n)$, and so $\binom{2d+2m}{2d} = n^{\Theta(1)}$, where the leading constant in $d_{B;\delta}(e^{-x})$ determines the value of the $\Theta(1)$.

Corollary 1.9. *Fix constants $\alpha, \beta > 0$, and suppose that $m = \alpha \log n$ and $\delta = n^{-\beta}$. If $B = \kappa \log n$ for some $\kappa < \frac{1}{2}$, then Batch Gaussian KDE can be solved in time $O(n^{1+c\log\log\kappa^{-1}/\log\kappa^{-1}})$, where $c = c(\alpha,\beta) > 0$ only depends on α and β.*

Prior work has shown similar "critical regime" behavior for other problems with SETH-based lower bounds. The Orthogonal Vectors problem for n vectors in dimension $\kappa \log n$ for large κ can be solved in time $n^{2-1/O(\log\kappa)}$ [AWY14, CW16], whereas the problem in dimension $\omega(\log n)$ requires time $n^{2-o(1)}$ assuming SETH. The Batch Hamming Nearest Neighbors problem for n vectors in dimension $\kappa \log n$ for large κ can be solved in time $n^{2-1/\tilde{O}(\sqrt{\kappa})}$ [AW15, ACW16], whereas the problem in dimension $\omega(\log n)$ also requires time $n^{2-o(1)}$ assuming SETH. Interestingly, these algorithms make use of variants on the polynomial method using *probabilistic* polynomials, whereas we make use of approximate polynomials here.

2 Proof Overview

In this section we outline the proofs of Theorem 1.2 and Theorem 1.3, which will be established in detail in Section 3 below. To that end, we will use the Chebyshev polynomials, which are defined as follows; see Section 3 for a more thorough explanation of its properties.

Definition 2.1. Fix an integer $d \geq 0$. Let $\mathcal{P}_d \subset \mathbb{R}[x]$ denote the set of single-variable polynomials $p(x)$ with $\deg p \leq d$, and define the degree d *monic Chebyshev polynomial* $Q_d(x) \in \mathcal{P}_d$ as follows. Set $Q_0(x) = 1$ and, for each $d \geq 1$, define $Q_d(x)$ by imposing that

$$Q_d(\cos\theta) = 2^{1-d}\cos(d\theta), \qquad \text{for each } \theta \in [0, 2\pi]. \tag{2}$$

It is well understood in the literature that smooth functions f are typically well-approximated by polynomials obtained by truncating the series expansion of f in the basis of Chebyshev polynomials; see, for instance, [Tre13, (15.5), (15.8)] for more precise formulations of this statement.

In particular, the following proposition provides a version of this statement that will be more useful for our purposes. It provides upper and lower bounds on the optimal error of a polynomial approximation $p(x)$ of a function $f : [-1,1] \to \mathbb{R}$ in terms of its Chebyshev expansion coefficients. Both bounds in this result are known; the lower bound follows from the L^2-orthogonality of the Chebyshev polynomials, and the upper bound follows from (2). Still, we provide a short and self-contained proof of the below proposition in Section 3.3.

Proposition 2.2. *Let $a_0, a_1, \ldots \in \mathbb{R}$ satisfy $\sum_{j=0}^{\infty} |a_j| < \infty$. Then, the absolutely convergent series $f : [-1,1] \to \mathbb{R}$ defined by $f(x) = \sum_{j=0}^{\infty} 2^{j-1} a_j Q_j(x)$ satisfies*

$$\left(\frac{1}{2} \sum_{k=D}^{\infty} \frac{a_k^2}{k} \right)^{1/2} \leq \inf_{p \in \mathcal{P}_{D-1}} \sup_{x \in [-1,1]} \big| p(x) - f(x) \big| \leq \sum_{j=D}^{\infty} |a_j|, \tag{3}$$

for any integer $D \geq 1$.

In particular, (3) provides nearly matching upper and lower bounds (up to a factor of $(2D)^{1/2}$) if the coefficients $\{a_j\}$ decay sufficiently quickly. Indeed, then the left and right sides of that inequality are asymptotically governed by their leading terms a_D.

As stated, Proposition 2.2 only applies for approximating polynomials on the interval $[-1,1]$. However, we would like to approximate e^{-x} and e^x on $[0,B]$, for some $B \geq 1$. Therefore, we rescale by first setting $\lambda = \frac{B}{2}$, and then by observing that to approximate e^{-x} (or e^x) on $[0,B]$ it suffices to approximate $e^{\lambda x-\lambda}$ (or $e^{\lambda x+\lambda}$, respectively) on $[-1,1]$.

We will show that the coefficients of these latter functions, when written in the Chebyshev basis, indeed decay quickly (with an explicit rate, dependent on λ), and then apply Proposition 2.2. We can in fact compute these coefficients exactly, by first expressing e^{-x} and e^x through their Taylor series, and then by changing basis from the monomials x^n to the Chebyshev polynomials. This yields the following (known) lemma, whose short proof will be recalled in Section 4.1 below.

Lemma 2.3. *For any real numbers $\lambda > 0$ and $x \in [-1,1]$, we have that*

$$e^{-\lambda x-\lambda} = \sum_{v=0}^{\infty} 2^{v-1} A_{v,\lambda} Q_v(x), \qquad e^{\lambda x+\lambda} = \sum_{v=0}^{\infty} 2^{v-1} B_{v,\lambda} Q_v(x), \tag{4}$$

where for any integer $v \geq 0$ we have set

$$A_{v,\lambda} = 2e^{-\lambda}(-1)^v \sum_{n-v\in 2\mathbb{Z}_{\geq 0}} \frac{\lambda^n}{2^n n!}\binom{n}{\frac{n-v}{2}}, \qquad B_{v,\lambda} = 2e^{\lambda} \sum_{n-v\in 2\mathbb{Z}_{\geq 0}} \frac{\lambda^n}{2^n n!}\binom{n}{\frac{n-v}{2}}. \tag{5}$$

We next apply a saddle point analysis to obtain precise asymptotics for $A_{v,\lambda}$ and $B_{v,\lambda}$, as $\lambda + v$ tends to ∞. In particular, the following proposition shows that these coefficients decay exponentially in v, with an explicit rate function given by $\Psi_{v,\lambda}$ in (6). This exact form of this rate function will eventually serve as the source of the phase transitions for $d_{B;\delta}(e^{-x})$ and $d_{B;\delta}(e^x)$ explained in Theorem 1.2 and Theorem 1.3, respectively. Indeed, one might already observe that $\Psi_{v,\lambda} = \lambda G\big(\frac{v}{\lambda}\big)$, where we recall the function $G(x)$ from those results. The below proposition will be stated a bit informally; we refer to Proposition 4.1 below for the more precise formulation needed for our purposes.

Proposition 2.4. *Recall the quantities $A_{v,\lambda}$ and $B_{v,\lambda}$ from (5) for any integer $v \geq 0$ and real number $\lambda \geq \frac{1}{2}$. Denote*

$$\Psi_{v,\lambda} = \sqrt{v^2+\lambda^2} + v \log\left(\frac{\sqrt{v^2+\lambda^2}-v}{\lambda}\right). \tag{6}$$

As $v + \lambda$ tends to ∞, we have that

$$(-1)^v A_{v,\lambda} = (\lambda+v)^{O(1)} \exp\big(\Psi_{v,\lambda} - \lambda\big); \qquad B_{v,\lambda} = (\lambda+v)^{O(1)} \exp\big(\Psi_{v,\lambda} + \lambda\big).$$

Combining Proposition 2.2 and Proposition 2.4, we obtain the following corollary, which provides nearly sharp bounds on the error one can achieve for a degree d polynomial approximation of e^{-x} and e^x on $[0,B]$. Once again, the below proposition will be stated a bit informally, and we refer to Proposition 4.5 below for a more precise formulation.

Corollary 2.5. *Let $d \geq 1$ be an integer, and let $B \geq 1$ be a real number. Set $\lambda = \frac{B}{2}$ and recall Ψ from (6). As $\lambda + d$ tends to ∞, we have that*

$$\inf_{p\in\mathcal{P}_d} \sup_{x\in[0,B]} \big|p(x) - e^{-x}\big| = (\lambda+d)^{O(1)} \exp(\Psi_{d,\lambda} - \lambda), \tag{7}$$

$$\inf_{p\in\mathcal{P}_d} \sup_{x\in[0,B]} \big|p(x) - e^{x}\big| = (\lambda+d)^{O(1)} \exp(\Psi_{d,\lambda} + \lambda). \tag{8}$$

Now Theorem 1.2 and Theorem 1.3 will follow from an explicit analysis of (7) and (8), respectively. For the purposes of this outline, we will omit the remaining details of analyzing the asymptotics of these expressions (referring to Section 4 for a more detailed exposition). However, let us briefly explain how the transitions from $B = \omega\big(\log(\delta^{-1})\big)$ to $B = o\big(\log(\delta^{-1})\big)$ arise, for example in Theorem 1.2.

They ultimately follow from the fact that the function $\Psi_{v,\lambda} - \lambda$ behaves differently depending on whether $v = O(\lambda)$ or $v = \Omega(\lambda)$, as different terms in the definition of $\Psi_{v,\lambda}$ are dominant in each of these settings; this is stated more precisely through the following lemma, which will be established in Section 4.2 below.

Lemma 2.6. *Let $\lambda \geq \frac{1}{2}$ be a real number, $v \geq 0$ be an integer, denote $\kappa = \frac{v}{\lambda}$, and recall the function $\Psi_{v,\lambda}$ from (6).*

1. *If $v \leq 2\lambda$ (that is, $\kappa \leq 2$) then $\Psi_{v,\lambda} - \lambda = -\dfrac{v^2}{2\lambda}\big(1 + O(\kappa)\big)$.*
2. *If $v \geq 2\lambda$ (that is, $\kappa \geq 2$) then $\Psi_{v,\lambda} - \lambda = -v \log\left(\dfrac{v}{\lambda}\right)\Big(1 + O\big((\log\kappa)^{-1}\big)\Big)$.*

In particular, the first part of Lemma 2.6 gives rise to the first part of Theorem 1.2, and the second part of Lemma 2.6 gives rise to the third part of Theorem 1.2.

Finally, we need one last tool to prove the fourth part of Theorem 1.2. In Theorem 1.3 as well as the first three parts of Theorem 1.2, our degree lower bound ultimately followed by finding a large coefficient in the Chebyshev expansion of $e^{\lambda x + \lambda}$ or $e^{\lambda x - \lambda}$ and then applying Proposition 2.2. However, when B (and hence λ) is very large compared to the desired error, the Chebyshev expansion of $e^{\lambda x - \lambda}$ actually has no sufficiently large coefficients.

We instead take a different approach in this last case. We observe that any polynomial p satisfying the bound $\sup_{x\in[0,B]} \left|p(x) - e^{-x}\right| < \delta$ must have,

- $|p(x)| \leq 2\delta$ in the entire interval $x \in [\log(\delta^{-1}), B]$, and
- $p(0) \geq 1 - \delta$.

It is known (see Fact 3.1 below) that the polynomial of lowest degree achieving these two properties must in fact be a (rescaled) Chebyshev polynomial. We then show that a Chebyshev polynomial requires degree $\Omega(\sqrt{B \log(\delta^{-1})})$ to realize these properties, from which the desired result follows.

3 Preliminaries

3.1 Notation

For a nonnegative integer d, we write $\mathcal{P}_d$ to denote the set of polynomials $p : \mathbb{R} \to \mathbb{R}$ with real coefficients of degree at most d. For a Boolean predicate P, we write

$$\mathbf{1}_P = \begin{cases} 1 & \text{if } P \text{ is true,} \\ 0 & \text{if } P \text{ is false.} \end{cases}$$

All logarithms in this paper are assumed to have base e, and we similarly write $\exp(x) := e^x$.

3.2 Chebyshev Polynomials

In this paper we make heavy use of the Chebyshev polynomials. Chebyshev polynomials appear prominently throughout polynomial approximation theory, and have been used in numerous other areas of theoretical computer science, including in Boolean function analysis and quantum computing; see e.g., [BT21]. Here we define them and give some of their well-known properties which will be important in our proofs. We refer the reader to [MH03] for more details.

The degree d *monic Chebyshev polynomial* $Q_d(x) \in \mathcal{P}_d$ is defined in many equivalent ways:

Definition 1 Set $Q_0(x) = 1$ and, for each $d \geq 1$, define $Q_d(x)$ by imposing that, for each $\theta \in [0, 2\pi]$, we have $Q_d(\cos\theta) = 2^{1-d}\cos(d\theta)$.

Definition 2 $Q_0(x) = 1$, $Q_1(x) = x$, and for $d \geq 2$ we have $Q_d(x) = x \cdot Q_{d-1}(x) - \frac{1}{2}Q_{d-2}(x)$.

Definition 3 $Q_0(x) = 1$ and for each $d \geq 1$ we have $Q_d(x) = 2^{1-d} \cdot \sum_{k=0}^{\lfloor d/2 \rfloor} \binom{d}{2k}(x^2-1)^k x^{d-2k}$.

In particular, $Q_d(x)$ is an even function when d is even, and an odd function when d is odd. From Definition 1, one observes several simple properties of $Q_d(x)$ for all $d > 0$ for $x \in [-1, 1]$:

- For all $x \in [-1, 1]$, we have $2^{d-1} \cdot Q_d(x) \in [-1, 1]$.
- All d roots of $Q_d(x)$ lie in $[-1, 1]$, and they lie at the points $x = \cos\big(\pi(2k+1)/2d\big)$ for each integer $0 \leq k < d$.
- On the interval $[-1, 1]$, the extrema of Q_d are located at the points $x = \cos(\pi k/d)$ for each integer $0 \leq k \leq d$. $Q_d(x)$ alternates between the values 2^{1-d} and -2^{1-d} at these extrema, starting at $Q_d(\cos(0)) = Q_d(1) = 2^{1-d}$.

Outside of the interval $[-1, 1]$, it is well-known that the Chebyshev polynomials exhibit useful extremal properties.

Fact 3.1. *For every integer $d > 0$, every polynomial $p(x) \in \mathcal{P}_d$ of degree d such that $2^{d-1} \cdot p(x) \in [-1, 1]$ for all $x \in [-1, 1]$, and every real $x' \notin [-1, 1]$, we have $|Q_d(x')| \geq |p(x')|$.*

Proof. Assume to the contrary that there is an $x' \notin [-1, 1]$ such that $|Q_d(x')| < |p(x')|$. By rescaling p by a factor in the range $(|Q_d(x')|/|p(x')|, 1)$, we can further assume that $2^{d-1} \cdot p(x) \in (-1, 1)$ for all $x \in [-1, 1]$. By the symmetry of $Q_d(x)$ (and by negating p if necessary), we may also assume without loss of generality that $x' > 1$ and that $p(x') > Q_d(x') > 0$ are positive.

Define the difference polynomial $g(x) = p(x) - Q_d(x)$, which has degree at most d. Consider the $d + 2$ points $x_{-1} > x_0 > x_1 > x_2 > \cdots > x_d \in \mathbb{R}$ given by

- $x_{-1} = x'$, and
- $x_k = \cos(\pi k/d)$ for each integer $0 \leq k \leq d$.

We have $g(x_{-1}) = p(x') - Q_d(x') > 0$ by assumption. For even $k \geq 0$ we have $Q_d(x_k) = 2^{1-d}$ and $|p(x_k)| < 2^{1-d}$, and so $g(x_k) < 0$. Similarly, for odd $k > 0$ we have $g(x_k) > 0$. Hence, $g(x)$ alternates signs at least $d + 2$ times in the interval $[x_d, x_{-1}]$, meaning it has at least $d + 1$ roots in that interval, a contradiction. □

In fact, $Q_d(1 + \varepsilon)$ for small $\varepsilon > 0$ is very closely approximated by an exponential in $\sqrt{\varepsilon}$:

Fact 3.2. *For any $\varepsilon > 0$ we have $Q_d(1 + \varepsilon) = 2^{-d} \cdot e^{d\sqrt{2\varepsilon}(1+O(\sqrt{\varepsilon}))}$.*

Proof. Extending Definition 1 of $Q_d(x)$ to $x \notin [-1,1]$, we find that

$$Q_d(x) = 2^{-d}\left(\left(x-\sqrt{x^2-1}\right)^d + \left(x+\sqrt{x^2-1}\right)^d\right), \qquad \text{for each } x \text{ with } |x|>1.$$

For $x = 1+\varepsilon$ with $\varepsilon > 0$, we thus get

$$Q_d(1+\varepsilon) = 2^{-d}\left(\left(1-\sqrt{2\varepsilon}+O(\varepsilon)\right)^d + \left(1+\sqrt{2\varepsilon}+O(\varepsilon)\right)^d\right) = 2^{-d}e^{d\sqrt{2\varepsilon}(1+O(\sqrt{\varepsilon}))},$$

as desired. □

3.3 Chebyshev Expansion Coefficients

In this section we prove Proposition 2.2, which shows how the coefficients of a function f written in the basis of Chebyshev polynomials can be used to bound how well f can be approximated by low-degree polynomials.

Proof of Proposition 2.2. Observe for any $d \in \mathbb{Z}_{\geq 0}$ and $x \in [-1,1]$ that $\left|Q_d(x)\right| \leq 2^{1-d}$, which follows from (2) after setting $x = \cos\theta$. This implies the absolute convergence of $f(x)$ for $x \in [-1,1]$, since $\sum_{j=0}^{\infty} |a_j| < \infty$. So, it remains to establish (3).

To establish the upper bound there, define

$$H_D(x) = \sum_{j=0}^{D-1} 2^{j-1} a_j Q_j(x). \tag{9}$$

Since $\left|Q_d(x)\right| \leq 2^{1-d}$ for each $x \in [-1,1]$, we have that

$$\sup_{x\in[-1,1]} \left|H_D(x) - f(x)\right| = \sup_{x\in[-1,1]} \left|\sum_{j=D}^{\infty} 2^{j-1} a_j Q_j(x)\right| \leq \sum_{j=D}^{\infty} |a_j|, \tag{10}$$

which proves the upper bound in (3).

To establish the lower bound, fix $p \in \mathcal{P}_{D-1}$, and let $c_0, c_1, \ldots \in \mathbb{R}$ satisfy

$$p(x) = \sum_{j=0}^{D-1} 2^{j-1} c_j Q_j(x), \quad \text{and } c_j = 0 \text{ for } j \geq D.$$

Then, applying (2), we obtain

$$p(\cos\theta) - f(\cos\theta) = \sum_{j=0}^{\infty} 2^{j-1}(a_j - c_j) Q_j(\cos\theta) = \frac{a_0 - c_0}{2} + \sum_{j=1}^{\infty} (a_j - c_j)\cos(j\theta). \tag{11}$$

Define $b_0, b_1, \ldots \in \mathbb{R}$ by setting $b_0 = \frac{a_0 - c_0}{2}$ and $b_j = a_j - c_j$ for $j > 0$, and observe that

$$\int_0^{2\pi} \cos(j\theta)\cos(k\theta)d\theta = 0, \qquad \text{for } j \neq k,$$

and

$$\int_0^{2\pi} \cos^2(k\theta)d\theta = \frac{1}{|k|}\int_0^{2\pi} \cos^2(\theta)d\theta = \pi|k|^{-1}\mathbf{1}_{k\neq 0} + (2\pi)\mathbf{1}_{k=0},$$

it follows that

$$\int_0^{2\pi} \left|p(\cos\theta) - f(\cos\theta)\right|^2 = \int_0^{2\pi} \left(\sum_{j=0}^{\infty} b_j \cos(j\theta) \right)^2 d\theta$$
$$= \sum_{j=0}^{\infty} \sum_{k=0}^{\infty} \int_0^{2\pi} b_j b_k \cos(j\theta)\cos(k\theta) d\theta = 2\pi b_0^2 + \pi \sum_{k=1}^{\infty} \frac{b_k^2}{k}.$$

Thus, since $b_j = a_j$ for $j \geq D$ (since $c_j = 0$ for $j \geq D$), we deduce

$$\sup_{x\in[-1,1]} \left|p(x) - f(x)\right|^2 = \sup_{\theta\in[0,2\pi]} \left|p(\cos\theta) - f(\cos\theta)\right|^2 \geq \frac{1}{2\pi} \int_0^{2\pi} \left|p(\cos\theta) - f(\cos\theta)\right|^2 d\theta$$
$$\geq \frac{1}{2} \sum_{k=D}^{\infty} \frac{a_k^2}{k},$$

which yields the proposition. □

Finally, we recall a well-known [MH03] identity expressing a monomial as an explicit linear combination of Chebyshev polynomials.

Lemma 3.3 ([MH03, (2.14)])**.** *For any integer $n \geq 0$, we have that*

$$x^n = \sum_{k=0}^{\lfloor n/2 \rfloor} 2^{-2k} \binom{n}{k} Q_{n-2k}(x).$$

4 Degree Bounds for Polynomial Approximations

4.1 Estimating $A_{v,\lambda}$ and $B_{v,\lambda}$

In this section we analyze $A_{v,\lambda}$ and $B_{v,\lambda}$ from (5). We begin with the proof of Lemma 2.3.

Proof of Lemma 2.3. We only establish the first statement in (4), as the proof of the second is entirely analogous. To that end, first using the series expansion for $e^{-z} = \sum_{n=0}^{\infty} \frac{(-z)^n}{n!}$ and then applying Lemma 3.3, yields

$$e^{-\lambda x - \lambda} = e^{-\lambda} \sum_{n=0}^{\infty} \frac{(-\lambda)^n}{n!} x^n = e^{-\lambda} \sum_{n=0}^{\infty} \frac{(-\lambda)^n}{n!} \sum_{k=0}^{\lfloor n/2 \rfloor} 2^{-2k} \binom{n}{k} Q_{n-2k}(x).$$

Then, by setting $v = n - 2k$, we obtain

$$e^{-\lambda x - \lambda} = e^{-\lambda} \sum_{v=0}^{\infty} Q_v(x) \sum_{k=0}^{\infty} \frac{(-\lambda)^{v+2k}}{(v+2k)!} 2^{-2k} \binom{v+2k}{k} = \sum_{v=0}^{\infty} 2^{v-1} A_{v,\lambda} Q_v(x),$$

from which we deduce the lemma. □

We next have the following proposition that more precisely formulates Proposition 2.4.

Proposition 4.1. *There exist constants $C, c > 0$ such that the following holds. For any integer $v \geq 0$ and real number $\lambda \geq \frac{1}{2}$, recall the quantities $A_{v,\lambda}$ and $B_{v,\lambda}$ from* (5) *and $\Psi_{\lambda,v}$ from* (6).

1. *For any v and λ as above, we have that*

$$c(v+\lambda)^{-1} \exp\big(\Psi_{v,\lambda} - \lambda\big) \le (-1)^v A_{v,\lambda} \le C(v+\lambda) \exp\big(\Psi_{v,\lambda} - \lambda\big);$$
$$c(v+\lambda)^{-1} \exp\big(\Psi_{v,\lambda} + \lambda\big) \le B_{v,\lambda} \le C(v+\lambda) \exp\big(\Psi_{v,\lambda} + \lambda\big).$$

2. *If $v \le \lambda$, then*

$$c(v+\lambda)^{-1} \exp\big(\Psi_{v,\lambda} - \lambda\big) \le (-1)^v A_{v,\lambda} \le C\lambda^{-1/2} \exp\big(\Psi_{v,\lambda} - \lambda\big).$$

To to establish the above result, observe that the two quantities $A_{v,\lambda}$ and $B_{v,\lambda}$ are quite similar, in that they both involve certain sum given by

$$E_{v,\lambda} = \sum_{n-v \in 2\mathbb{Z}_{\ge 0}} \frac{\lambda^n}{2^n n!} \binom{n}{\frac{n-v}{2}} .a \tag{12}$$

We therefore require the following proposition that estimates $E_{v,\lambda}$. Observe its second statement slightly improves the upper bound in its first statement for λ large (which will be useful in analyzing $d_{B;\delta}$ in the regime of large B below).

Proposition 4.2. *There exist constants $C, c > 0$ such that the following holds. For any integer $v \ge 0$ and real number $\lambda \ge \frac{1}{2}$, recall the quantities $E_{v,\lambda}$ and $\Psi_{v,\lambda}$ from* (12) *and* (6)*, respectively.*

1. *If $v \ge \lambda$, then $c(v+\lambda)^{-1} \exp\big(\Psi_{v,\lambda}\big) \le E_{v,\lambda} \le C(v+\lambda) \exp\big(\Psi_{v,\lambda}\big)$.*
2. *If $v \le \lambda$, then $c(v+\lambda)^{-1} \exp\big(\Psi_{v,\lambda}\big) \le E_{v,\lambda} \le C\lambda^{-1/2} \exp\big(\Psi_{v,\lambda}\big)$.*

Proposition 2.4 now follows directly from Proposition 4.2.

Proof of Proposition 4.1 Assuming Proposition 4.2. Given Proposition 4.2, Proposition 4.1 follows from the facts that $(-1)^v A_{v,\lambda} = 2e^{-\lambda} E_{v,\lambda}$ and $B_{v,\lambda} = 2e^{\lambda} E_{v,\lambda}$. □

We must thus establish Proposition 4.2, to which end we begin with the following lemma.

Lemma 4.3. *For any integer $v \ge 1$ and real number $\lambda \ge \frac{1}{2}$, we have that*

$$E_{v,\lambda} = \Theta\Bigg(\sum_{n-v \in 2\mathbb{Z}_{\ge 0}} \big(n^2 - v^2 + n\big)^{-1/2} \exp\big(F(n)\big) \Bigg), \tag{13}$$

where for any real number $n \ge v$ we have defined $F(n) = F(n)$ by

$$F(n) = n \log \lambda - n \log 2 + n - \Big(\frac{n-v}{2}\Big) \log\Big(\frac{n-v}{2}\Big) - \Big(\frac{n+v}{2}\Big) \log\Big(\frac{n+v}{2}\Big). \tag{14}$$

Proof. The explicit form (12) for $E_{v,\lambda}$ and the Stirling estimate $n! = \Theta\big((n+1)^{n+1/2} e^{-n}\big)$, which holds uniformly in $n \ge 0$, together imply

$$E_{v,\lambda} = \Theta\Bigg(\sum_{n-v \in 2\mathbb{Z}_{\ge 0}} \big((n+2)^2 - v^2\big)^{-1/2} \exp\big(F(n)\big) \Bigg).$$

From this, we deduce the lemma since $(n+2)^2 - v^2 = \Theta(n^2 - v^2 + n)$, uniformly in $n \ge v$. □

The right side of (13) will be dominated by the terms near which F is maximized, so we next perform a critical point analysis on F.

Lemma 4.4. *Fix an integer $v \geq 1$ and a real number $\lambda \geq \frac{1}{2}$, and set $n_0 = n_0(v, \lambda) = \sqrt{v^2 + \lambda^2}$. There exist constants $c, C > 0$ (independent of v and λ) such that the following holds.*

1. *The function $F(n)$ is maximized at $n = n_0$, and $F(n_0) = \Psi_{v,\lambda}$ (recall (6)).*
2. *For $z \geq 2\lambda$, we have that $F(n_0 + z) \leq F(n_0) - cz$.*
3. *For at least one choice of $m \in \big\{ \lfloor n_0 \rfloor, \lceil n_0 \rceil \big\}$, we have that $F(m) \geq F(n_0) - C$.*
4. *If $v \leq \lambda$, then for any $z \in [v - n_0, 2\lambda]$ we have that $F(n_0 + z) \leq F(n_0) - c\lambda^{-1} z^2$.*

Proof. Using the explicit form (14) for $F(n)$, we deduce for $n \geq v$ that

$$F'(n) = \log \lambda - \frac{1}{2} \log(n^2 - v^2), \quad \text{and} \quad F''(n) = \frac{n}{v^2 - n^2} \in \Big[-\frac{1}{n}, 0 \Big], \tag{15}$$

which implies that $F'(n_0) = 0$ and that F is maximized at n_0. Upon insertion into (14) (and recalling (6)), we also find that

$$F(n_0) = \sqrt{v^2 + \lambda^2} + v \log \bigg(\frac{\sqrt{v^2 + \lambda^2} - v}{\lambda} \bigg) = \Psi_{v,\lambda},$$

which verifies the first statement of the lemma.

To establish the second, first observe since $F''(n) \leq 0$ and $n_0^2 - v^2 = \lambda^2$ that

$$F'(n_0 + \lambda) = \log \lambda - \frac{1}{2} \log \big((n_0 + \lambda)^2 - v^2 \big) \leq \log \lambda - \frac{1}{2} \log(n_0^2 + \lambda^2 - v^2) = -\frac{\log 2}{2}. \tag{16}$$

Thus, we deduce for $z \geq 2\lambda$ that

$$F(n_0 + z) - F(n_0) \leq F(n_0 + z) - F(n_0 + \lambda) \leq (z - \lambda) F'(n_0 + \lambda) \leq \frac{\log 2}{2} (\lambda - z) \leq -\frac{z \log 2}{4},$$

where in the first inequality we used the fact that F is maximized at n_0; in the second we used the fact that $F''(n) \leq 0$; in the third we used (16); and in the fourth we used the fact that $z - \lambda \geq \frac{z}{2}$. This verifies the second statement of the lemma.

To show the third part of the lemma, we separately consider the cases when $v \leq \lambda^2$ and $v \geq \lambda^2$. In the former situation $v \leq \lambda^2$, we select $m = \lceil n_0 \rceil$. Since $F'''(n) = (n^2 + v^2)(n^2 - v^2)^{-2} \geq 0$, we then have for for $n \in [n_0, m]$ that

$$F'(n) \geq (n - n_0)^2 F''(n_0) \geq -\frac{(n - n_0)^2}{n_0} \geq -\frac{1}{n_0} \geq \frac{n_0}{\lambda^2},$$

using the last second identity in (15). Since $n_0 \leq \lambda + v \leq 3\lambda^2$ (due to the facts that $\lambda \geq \frac{1}{2}$ and $v \leq \lambda^2$), it follows that $F'(n) \geq -\frac{1}{3}$ for $n \in [n_0, m]$, which implies that $F(m) \geq F(n_0) - \frac{1}{3}$ if $v \leq \lambda^2$.

Now instead suppose that $v \geq \lambda^2$, in which case we take $m = \lfloor n_0 \rfloor = v$. Then, using the second identity in (15), we obtain

$$F(n_0) - F(v) = \int_0^{n_0 - v} F'(v + z) dz = -\frac{1}{2} \int_0^{n_0 - v} \log \bigg(\frac{2vz + z^2}{\lambda^2} \bigg) dz \leq -\frac{1}{2} \int_0^{n_0 - v} \log \bigg(\frac{vz}{\lambda^2} \bigg) dz.$$

Now set $r = \frac{\lambda^2}{v}$, and observe that $n_0 - v \leq r \leq 1$. This yields

$$F(m) \geq F(n_0) + \frac{1}{2}\int_0^r \log\left(\frac{z}{r}\right)dz = F(n_0) + \frac{r}{2}\int_0^1 (\log y)dy = F(n_0) - \frac{r}{2} \geq F(n_0) - 1,$$

where in the first equality we changed variables $z = ry$. This verifies the third part of the lemma.

To establish the fourth part of the lemma, let us first consider the case when $z \in [v - n_0, 0]$. Then, since $F'(n_0) = 0$ and $F'''(n) = (n^2 + v^2)(n^2 - v^2)^{-2} \geq 0$ for each $n \geq v$, we have that

$$F(n_0 + z) \leq F(n_0) + \frac{z^2 F''(n_0)}{2} = F(n_0) - \frac{z^2(\lambda^2 + v^2)^{1/2}}{2\lambda^2}, \quad \text{for } z \in [v - n_0, 0],$$

where in the last equality we used the second identity in (15) for $F'(n)$ (and the fact that $n_0 = \sqrt{\lambda^2 + v^2}$). Thus, we deduce that

$$F(n_0 + z) \leq F(n_0) - \frac{z^2}{2\lambda}, \quad \text{for } z \in [v - n_0, 0]. \tag{17}$$

Next we consider the case when $z \in [0, 2\lambda]$. In this case, the second identity in (15) implies for each $m \in [n_0, n_0 + 2\lambda]$ that

$$F''(m) = \frac{m}{v^2 - m^2} \leq -\frac{\lambda}{m^2} \leq -\frac{1}{16\lambda},$$

where in the last inequality we used the fact that $m \leq n_0 + 2\lambda \leq 4\lambda$ (as $v \leq \lambda$ and $n_0 = \sqrt{v^2 + \lambda^2}$). Thus, it follows from the fact that $F'(n_0) = 0$ that

$$F(n_0 + z) \leq F(n_0) - \frac{z^2}{32\lambda}, \quad \text{for each } z \in [0, 2\lambda]. \tag{18}$$

Now the fourth statement of the lemma follows from (17) and (18). □

Now we can establish Proposition 4.2.

Proof of Proposition 4.2. We begin by establishing the lower bound on $E_{v,\lambda}$, simultaneously in both cases $v \geq \lambda$ and $v \leq \lambda$. To that end, we first apply Lemma 4.3 and then use the third part of Lemma 4.4 to bound the sum on the right side of (13) its summand corresponding to a suitable choice of index $m \in \big\{\lfloor n_0 \rfloor, \lceil n_0 \rceil\big\}$. This yields constants $c_1, c_2 > 0$ such that

$$E_{v,\lambda} \geq c_1\big(\lambda^2 + v^2\big)^{-1/2} \exp\big(-F(m)\big) \geq c_1(\lambda + v)^{-1} \exp\big(F(n_0) - c_2\big),$$

which implies the lower bound on $E_{v,\lambda}$ (in either case $v \geq \lambda$ or $v \leq \lambda$), since $F(n_0) = \Psi_{\lambda,v}$.

To establish the upper bound in the case $v \geq \lambda$, observe that Lemma 4.3; the fact that $F(n) \leq F(n_0) = \Psi_{v,\lambda}$ for each $n \in [v, n_0 + 2\lambda]$ (by the first part of Lemma 4.4); and the existence of a constant $c > 0$ such that $F(n_0 + z) \leq F(n_0) - cz$ for $z \geq 2\lambda$ (by the second part of Lemma 4.4) together yield a constant $C_1 > 0$ such that

$$E_{v,\lambda} \leq C_1(n_0 + 2\lambda)\exp(\Psi_{v,\lambda}) \sum_{z=2\lambda}^{\infty} e^{-cz} \leq 2c^{-1}C_1(n_0 + 2\lambda)\exp(\Psi_{v,\lambda}).$$

This establishes the upper bound on $E_{v,\lambda}$ when $v \geq \lambda$.

In the latter case $v \leq \lambda$, we proceed as above but additionally use the facts that $F(n_0 + z) \leq F(n_0) - c\lambda^{-1}z^2$ for $z \in [v - n_0, 2\lambda]$ (by the fourth part of Lemma 4.4) to deduce for some constant $C_2 > 0$ that

$$\begin{aligned}
E_{v,\lambda} &\leq C_1 \exp(\Psi_{v,\lambda})\Bigg(\sum_{z=v-n_0}^{2\lambda} \big((n_0+z)^2 - v^2 + n_0 + z\big)^{-1/2} e^{-cz^2/\lambda} + \sum_{z=2\lambda}^{\infty} e^{-cz} \Bigg) \\
&\leq C_1 \exp(\Psi_{v,\lambda})\Bigg(\sum_{|z|\leq\lambda/4} \big((n_0+z)^2 - v^2 + n_0 + z\big)^{-1/2} e^{-cz^2/\lambda} + \sum_{|z|\geq\lambda/4} e^{-cz^2/\lambda} + \sum_{z=2\lambda}^{\infty} e^{-cz} \Bigg) \\
&\leq C_1 \exp(\Psi_{v,\lambda})\Bigg(12\lambda^{-1} \sum_{|z|\leq\lambda/4} e^{-cz^2/\lambda} + 35c^{-1}e^{-2\lambda} \Bigg) \leq C_2 \lambda^{-1/2} \exp(\Psi_{v,\lambda}),
\end{aligned}$$

where in the third inequality we used the fact that $(n_0 + z)^2 - v^2 \geq \frac{\lambda^2}{144}$ for $|z| \leq \frac{\lambda}{4}$ (as $n_0 = \sqrt{\lambda^2 + v^2} \geq \frac{\lambda}{3} + v$ for $\lambda \geq v$). This establishes the upper bound on $E_{v,\lambda}$ when $v \leq \lambda$. □

4.2 Estimates for the Minimum Polynomial Approximation Error

In this section we establish the following proposition, which is the variant of Corollary 2.5 that will be useful for our purposes.

Proposition 4.5. *There exist constants $C, c > 0$ such that the following holds. Let $d \geq 1$ be an integer, and let $B \geq 1$ be a real number. Set $\lambda = \frac{B}{2}$ and recall Ψ from (6).*

1. *For any B and d as above, we have*

$$c(d+\lambda)^{-3/2} \exp(\Psi_{d,\lambda} - \lambda) \leq \inf_{p \in \mathcal{P}_d} \sup_{x \in [0,B]} \big|p(x) - e^{-x}\big| \leq C(d+\lambda)^2 \exp(\Psi_{d,\lambda} - \lambda); \tag{19}$$

$$c(d+\lambda)^{-3/2} \exp(\Psi_{d,\lambda} + \lambda) \leq \inf_{p \in \mathcal{P}_d} \sup_{x \in [0,B]} \big|p(x) - e^{x}\big| \leq C(d+\lambda)^2 \exp(\Psi_{d,\lambda} + \lambda). \tag{20}$$

2. *If $B \geq 2d$, then we have that*

$$c(d+\lambda)^{-3/2} \exp(\Psi_{d,\lambda} - \lambda) \leq \inf_{p \in \mathcal{P}_d} \sup_{x \in [0,B]} \big|p(x) - e^{-x}\big| \leq C \exp(\Psi_{d,\lambda} - \lambda). \tag{21}$$

We will establish Proposition 4.5 as a consequence of Proposition 2.2 and Proposition 4.1. However, before doing so, it will be useful to obtain some properties for $\Psi_{v,\lambda}$. Therefore, we first prove Lemma 2.6.

Proof of Lemma 2.6. First observe that

$$\Psi_{v,\lambda} - \lambda = \lambda\Big(\sqrt{\kappa^2+1} - 1 + \kappa \log\big(\sqrt{\kappa^2+1} - \kappa\big) \Big).$$

In particular, if $\kappa \leq 2$ then using the series expansions

$$\sqrt{z^2+1} = 1 + \frac{z^2}{2} + O(z^3), \quad \text{and} \quad \log(1+z) = z + O(z^2), \quad \text{valid for } z \in [0,2],$$

we obtain that

$$\Psi_{v,\lambda} = \lambda\bigg(\frac{\kappa^2}{2} + O(\kappa^3) + \kappa \log\Big(1 - \kappa + O(\kappa^2)\Big) \bigg) = -\frac{\kappa^2 \lambda}{2}\big(1 + O(\kappa)\big).$$

If instead $\kappa \geq 2$ then using the series expansion

$$\sqrt{z^2+1} = z + \frac{1}{2z} + O(z^{-2}), \quad \text{and} \quad \log(z^{-1}+z^{-2}) = -\log z - O(z^{-1}), \quad \text{valid for } |z| \geq 2,$$

we deduce that

$$\Psi_{v,\lambda} = \lambda\Bigg(\kappa + O\Big(\frac{1}{\kappa}\Big) + \kappa \log\bigg(\frac{1}{2\kappa} + O\Big(\frac{1}{\kappa^2}\Big)\bigg)\Bigg) = -\lambda\kappa\log\kappa\Big(1 + O\big((\log\kappa)^{-1}\big)\Big).$$

This establishes the lemma. □

Now we can establish Proposition 4.5.

Proof of Proposition 4.5. First observe that by rescaling (namely, replacing x with $\lambda(x+1)$ or $-\lambda(x+1)$), we have that

$$\begin{aligned} \inf_{p\in\mathcal{P}_d} \sup_{x\in[0,B]} \big|p(x) - e^x\big| &= \inf_{p\in\mathcal{P}_d} \sup_{x\in[-1,1]} \big|p(x) - e^{\lambda x+\lambda}\big|; \\ \inf_{p\in\mathcal{P}_d} \sup_{x\in[0,B]} \big|p(x) - e^{-x}\big| &= \inf_{p\in\mathcal{P}_d} \sup_{x\in[-1,1]} \big|p(x) - e^{-\lambda-\lambda x}\big|. \end{aligned} \tag{22}$$

Therefore, Proposition 2.2 and the definitions of $A_{v,\lambda}$ and $B_{v,\lambda}$ from (5), together yield

$$\inf_{p\in\mathcal{P}_d} \sup_{x\in[0,B]} \big|p(x) - e^{-x}\big| \geq (2d)^{-1/2}\big|A_{v,\lambda}\big|; \qquad \inf_{p\in\mathcal{P}_d} \sup_{x\in[0,B]} \big|p(x) - e^{x}\big| \geq (2d)^{-1/2}\big|B_{v,\lambda}\big|. \tag{23}$$

Thus, the lower bounds on the minimal error for e^{-x} and e^x in both of the cases listed in the proposition follow from (23) and the lower bounds on $|A_{v,\lambda}|$ and $|B_{v,\lambda}|$ from the first part of Proposition 4.1.

Now let us establish the upper bounds in this proposition; in what follows, $C > 0$ will denote a constant (uniform in d and λ) that might change between appearances. We first show (20), to which end, observe that (22), Proposition 2.2, and the upper bound for $B_{v,\lambda}$ from the first part of Proposition 4.1 together yield

$$\inf_{p\in\mathcal{P}_d} \sup_{x\in[0,B]} \big|p(x) - e^{-x}\big| \leq \sum_{v=d}^{\infty} B_{v,\lambda} \leq C \sum_{v=d}^{\infty} (v+\lambda)\exp(\Psi_{v,\lambda}+\lambda). \tag{24}$$

Next, observe from the second part of Lemma 2.6 that

$$\Psi_{v,\lambda} + \lambda \leq -v, \qquad \text{for } v > C\lambda. \tag{25}$$

Further observe that

$$\Psi_{v,\lambda} \text{ is decreasing in } v \geq 0 \text{ for fixed } \lambda, \tag{26}$$

since

$$\frac{\partial}{\partial v}\Psi_{v;\lambda} = \log\big(\sqrt{\kappa^2+1} - \kappa\big) \leq 0, \qquad \text{for } \kappa = \frac{v}{\lambda} \geq 0. \tag{27}$$

From (24), (25), and (26), it follows that

$$\inf_{p\in\mathcal{P}_d}\sup_{x\in[0,B]}\left|p(x)-e^{x}\right| \le C(d+\lambda)\exp\big(\Psi_{d,\lambda}+\lambda\big)\left(d+C\lambda+\sum_{v\ge d+C\lambda}(v+\lambda)e^{-v}\right)$$
$$\le C(d+\lambda)^2\exp\big(\Psi_{d,\lambda}+\lambda\big).$$

This establishes (20); the proof of (19) is omitted as it is entirely analogous.

Now let us establish the improved upper bound on the minimum error in the case when $B\ge 2d$. To that end, we as before apply (22), Proposition 2.2, and the upper bound for $A_{v,\lambda}$ from the second part of Proposition 4.1 to obtain

$$\inf_{p\in\mathcal{P}_d}\sup_{x\in[0,B]}\left|p(x)-e^{-x}\right| \le C\sum_{v=d}^{\infty}\lambda^{-1/2}\exp(\Psi_{v,\lambda}-\lambda). \tag{28}$$

By (27), we have that $\Psi_{v;\lambda}<0$ for $v>0$ and moreover that

$$\frac{\partial}{\partial v}\Psi_{v;\lambda}=\log\left(1-\frac{v}{\lambda}+O\Big(\frac{v^2}{\lambda^2}\Big)\right)=O\left(\frac{v^2}{\lambda^2}\right)-\frac{v}{\lambda},\qquad \text{for } v\le 10\lambda;$$
$$\frac{\partial}{\partial v}\Psi_{v;\lambda}=\log\left(\frac{\lambda}{2v}+O\Big(\frac{\lambda^2}{v^2}\Big)\right)=\left(1+O\Big(\frac{\lambda}{v}\Big)\right)\log\left(\frac{\lambda}{2v}\right),\qquad \text{for } v\ge 10\lambda.$$

In particular, there exist constants $c_1,c_2>0$ such that for $v,\lambda\ge\frac{1}{2}$ we have $\frac{\partial}{\partial v}\Psi_{v;\lambda}\le -c_1$ if $v\ge c_2\lambda$ and $\frac{\partial}{\partial v}\Psi_{v;\lambda}\le-\frac{v}{2\lambda}$ for $v\le c_2\lambda$. Thus,

$$\Psi_{v;\lambda}-\Psi_{d;\lambda}\le-\frac{c_2}{4\lambda}(v-d)^2,\quad \text{for } v\le c_2\lambda;\qquad \Psi_{v;\lambda}-\Psi_{d;\lambda}\le C^{-1}(d-v),\quad \text{for } v\ge c_2\lambda,$$

and so

$$\sum_{v=d}^{\infty}\exp(\Psi_{v,\lambda}-\lambda)\le C\lambda^{1/2}\exp(\Psi_{d,\lambda}-\lambda). \tag{29}$$

The upper bound in (21) now follows from (28) and (29). □

4.3 Proofs of Theorem 1.2 and Theorem 1.3

In this section we establish Theorem 1.2 and Theorem 1.3. Recalling the function $G(x)$ and the quantity $\Psi_{v,\lambda}$ from (6) from these statements, both of these proofs will use the fact that

$$\Psi_{v,\lambda}=\lambda G\Big(\frac{v}{\lambda}\Big). \tag{30}$$

We begin with the proof of Theorem 1.3.

Proof of Theorem 1.3. Set $\lambda=\frac{B}{2}$. Observe that there exists a constant $c_1>0$ such that $G'(z)<-c_1$ whenever $z\in[z_*-c_1,z_*+c_1]$. Thus, $G(x)+1>c_1(z-x_*)$, and so (30) yields for some constant $c_2>0$ that

$$(d+\lambda)^{-1}\exp(\Psi_{d,\lambda}+\lambda)>(d+\lambda)^{-1}\exp(c_2\lambda^{1/2})>10,$$

if $d < z_*\lambda - \lambda^{1/2}$ and λ is sufficiently large. Thus, by the lower bound in Proposition 4.5, for any λ sufficiently large and $\delta \le \frac{1}{2}$ we have

$$d_{B;\delta} \ge \big(z_* + o(1)\big)\lambda = \big(z_* + o(1)\big)\frac{B}{2}. \tag{31}$$

Now, assume first that $B = \omega\big(\log(\delta^{-1})\big)$. Then, the upper bound in Proposition 4.5 implies that $d(B;\delta) \le d$ if d satisfies $(d+\lambda)^2 \exp\big(\Psi_{d,\lambda} + \lambda\big) < \delta$. Since $G(z_*) = 0$ and $G'(z_*) < 0$, there exists a constant $C > 0$ such that

$$(d+\lambda)^2 \exp\big(\Psi_{d,\lambda} + \lambda\big) \le (d+\lambda)^2 e^{-C(d - z_*\lambda)}.$$

Since $\lambda = \frac{B}{2} = \omega\big(\log(\delta^{-1})\big)$ implies that $(d+\lambda)^2 \exp\big(\Psi_{d,\lambda} + \lambda\big) < \delta$ for $d = \big(z_* + o(1)\big)\lambda = \big(z_* + o(1)\big)\frac{B}{2}$. Hence, in this case $d_{B;\delta}(e^x) \le \big(z_* + o(1)\big)\frac{B}{2}$, which by (31) implies that $d_{B;\delta}(e^x) = \big(z_* + o(1)\big)\frac{B}{2}$.

Now assume that $B = \big(2r + o(1)\big)\log(\delta^{-1})$ for some fixed $r > 0$, so that $\lambda = \big(r + o(1)\big)\log(\delta^{-1})$. Suppose that $d < \mu'\lambda$ for some $\mu' < \mu(r)$. Then, $G(\mu') + 1 > -r^{-1}$ and so we have again using (30) (and the fact that G is decreasing) that there would exist a constant $c_3 > 0$ such that

$$\begin{aligned}(d+\lambda)^{-3/2} \exp\big(\Psi_{d,\lambda} + \lambda\big) &= (d+\lambda)^{-3/2} \exp\left(\lambda G\Big(\frac{d}{\lambda} + \lambda\Big)\right)\\ &\ge (d+\lambda)^{-3/2} \exp\Big(\lambda\big(G(\mu') + 1\big)\Big)\\ &\ge (d+\lambda)^{-3/2} \exp\big((c_3 - r^{-1}\lambda\big) = \delta(d+\lambda)^{-1} e^{(c_3 - o(1))\lambda} > \delta.\end{aligned}$$

Thus, the lower bound in Proposition 4.5 implies that $d_{B;\delta}(e^x) \ge \big(\mu + o(1)\big)\lambda = \big(\mu r + o(1)\big)\log(\delta^{-1})$.

Similarly, if $d > \mu''\lambda$ for some $\mu'' > \mu(r)$, then there exists some constant $c_4 > 0$ such that

$$\begin{aligned}(d+\lambda)^2 \exp\big(\Psi_{d,\lambda} + \lambda\big) &\ge (d+\lambda)^2 \exp\Big(\lambda\big(G(\mu'') + 1\big)\Big)\\ &\ge (d+\lambda)^{-1} \exp\big(-\lambda(r^{-1} + c_4)\big) = \delta(d+\lambda)^2 e^{(o(1) - c_4)\lambda} < \delta,\end{aligned}$$

which implies by the upper bound in Proposition 4.5 that $d_{B;\delta}(e^x) \le \big(\mu r + o(1)\big)\log(\delta^{-1})$. Hence, $d_{B;\delta}(e^x) = \big(\mu r + o(1)\big)\log(\delta^{-1})$.

Now let us consider the final case $B = o\big(\log(\delta^{-1})\big)$. Suppose that

$$d = \frac{\gamma \log(\delta^{-1})}{\log(B^{-1}\log(\delta^{-1}))},$$

for some $\gamma \in (0,\infty)$ (bounded above and below). Then, $\frac{d}{\lambda} = \omega(1)$, and so the second part of Lemma 2.6 implies that

$$\Psi_{d,\lambda} = -d\log\Big(\frac{d}{\lambda}\Big)\big(1 + o(1)\big).$$

Hence, if $\gamma < 1$, then

$$\begin{aligned}&(d+\lambda)^{-3/2} \exp\big(\Psi_{d,\lambda} + \lambda\big)\\ &\quad= (d+\lambda)^{-3/2} \exp\left(-d\log\Big(\frac{d}{\lambda}\Big)\big(1 + o(1)\big)\right)\\ &\quad= (d+\lambda)^{-3/2} \exp\left(-\frac{\gamma\log(\delta^{-1})}{\log(B^{-1}\log(\delta^{-1}))}\log\Big(\frac{2\gamma B^{-1}\log(\delta^{-1})}{\log(B^{-1}\log(\delta^{-1}))}\Big)\big(1 + o(1)\big)\right)\\ &\quad= (d+\lambda)^{-3/2} \exp\Big(\big(\gamma + o(1)\big)\log(\delta^{-1})\Big) \ge \delta^{\gamma + o(1)}\big(\log(\delta^{-1})\big)^{-3/2} > \delta.\end{aligned}$$

Hence, the lower bound in (4.5) implies that

$$d_{B;\delta}(e^x) \geq \frac{\big(1+o(1)\big)\log(\delta^{-1})}{\log(B^{-1}\log(\delta^{-1}))},$$

The proof of the matching upper bound is entirely analogous and is therefore omitted. □

Next we establish Theorem 1.2. To that end, we begin with the following lemma that addresses the last part of that theorem, when $B \geq \delta^{-\Omega(1)}$.

Lemma 4.6. *For every real $\delta \in (0, 1/4)$ and $B \geq 1$ with $B > \omega(\log(\delta^{-1}))$ we have $d_{B;\delta}(e^{-x}) \geq (1/2 + o(1))\sqrt{B\log((2\delta)^{-1})}$.*

Proof. Let $p(x)$ be any polynomial satisfying $\sup_{x\in[0,B]}\left|p(x) - e^{-x}\right| < \delta$, and set $d = \deg p$. Let $x_0 = 0$, $x_1 = \log(\delta^{-1})$, and $x_2 = B$. It follows that $p(x_0) \geq 1 - \delta$, and that $p(x) \in [-\delta, 2\delta]$ for all $x \in [x_1, x_2]$. Let $a : \mathbb{R} \to \mathbb{R}$ be the linear function satisfying $a(1) = x_1$ and $a(-1) = x_2$, and let

$$x_0' := a^{-1}(x_0) = 1 + \frac{2(x_1 - x_0)}{x_2 - x_1} = 1 + \frac{2\log(\delta^{-1})}{B - \log(\delta^{-1})}.$$

Finally, define the polynomial $q(x) = \frac{p(a(x))}{2\delta}$, which also has degree d. It follows that $q(x) \in [-1, 1]$ for all $x \in [-1, 1]$, and that $q(x_0') \geq \frac{1-\delta}{2\delta}$.

Applying Fact 3.1 to q, we see that $|Q_d(x_0')| \geq 2^{1-d}q(x_0') \geq \frac{1-\delta}{2^d\delta}$. Furthermore, by Fact 3.2 we have that $|Q_d(x_0')| \leq 2^{-d}e^{(\sqrt{2}+o(1))d\sqrt{x_0'-1}}$. Combining the two bounds yields:

$$\frac{1-\delta}{2^d\delta} \leq 2^{-d}e^{(\sqrt{2}+o(1))d\sqrt{x_0'-1}} = 2^{-d}e^{(2+o(1))d\sqrt{\log(\delta^{-1})/(B-\log(\delta^{-1}))}}.$$

Taking logs of both sides and rearranging gives the desired result. □

Now we can establish Theorem 1.2.

Proof of Theorem 1.2. The proofs of the estimates on $d_{B;\delta}(e^{-x})$ in the first and second cases, when either $B = o\big(\log(\delta^{-1})\big)$ and $B = \Theta\Big(\log(\delta^{-1})\Big)$ are entirely analogous to those for $d_{B;\delta}(e^x)$ shown in Theorem 1.3 above. Therefore, they are omitted.

So, let us assume that $B = \omega\big(\log(\delta^{-1})\big)$, and let $d > 0$ be some integer with

$$d = \sqrt{\gamma B\log(\delta^{-1})} = \sqrt{2\gamma\lambda\log(\delta^{-1})},$$

where γ is uniformly bounded above and below. Observe since $B = \omega\big(\log(\delta^{-1})\big)$ that $d = o(B)$, and so Lemma 2.6 implies that

$$\Psi_{d,\lambda} - \lambda = -\frac{d^2}{2\lambda}\big(1 + o(1)\big).$$

Now, let us first approximate $d_{B;\delta}(e^{-x})$ by $\big(1 + o(1)\big)\sqrt{B\log(\delta^{-1})}$ in the regime where $B \leq \delta^{-o(1)}$. To lower bound it, suppose that $\gamma < 1$ (and is uniformly bounded away from 1). Then,

$$\begin{aligned}(\lambda + d)^{-3/2}\exp(\Psi_{d,\lambda} - \lambda) &\geq (\lambda + d)^{-3/2}\exp\Big(-\frac{d^2}{2\lambda}\big(1 + o(1)\big)\Big)\\ &\geq (\lambda + d)^{-3/2}\exp\Big(-\gamma\log(\delta^{-1})\Big) \geq \delta^{\gamma+o(1)},\end{aligned}$$

where in the last bound we used the fact that $d = o(\lambda)$ and that $\lambda = \frac{B}{2} \le \delta^{-o(1)}$. Hence, by the lower bound in the second part of Proposition 4.5, we find that $d_{B;\delta}(e^{-x}) \ge \big(1+o(1)\big)\sqrt{B\log(\delta^{-1})}$.

To upper bound $d_{B;\delta}(e^{-x})$ for $B \le \delta^{-o(1)}$, assume that $\gamma > 1$ (and is uniformly bounded away from 1). Then,

$$\begin{aligned}\exp(\Psi_{d,\lambda} - \lambda) &\le (\lambda+d)^2 \exp\Big(-\frac{d^2}{2\lambda}\big(1+o(1)\big)\Big)\\ &\le (\lambda+d)^2 \exp\Big(-\log(\delta^{-1})\big(\gamma - o(1)\big)\Big) \le (\lambda+d)^2\delta^{\gamma-o(1)} < \delta,\end{aligned}$$

and so again by Proposition 4.5 we deduce that $d_{B;\delta}(e^{-x}) \le \big(1+o(1)\big)\sqrt{B\log(\delta^{-1})}$. Together, these upper and lower bounds imply that $d_{B;\delta}(e^{-x}) = \big(1+o(1)\big)\sqrt{B\log(\delta^{-1})}$ when $B = \omega\big(\log(\delta^{-1})\big)$ and $B \le \delta^{-o(1)}$.

It remains to show that $d_{B;\delta}(e^{-x}) = \Theta\big(\sqrt{B\log(\delta^{-1})}\big)$ when $B \ge \delta^{-\Omega(1)}$. The lower bound (with implicit constant $\frac{1}{2} + o(1)$) was shown by Lemma 4.6, so we must verify the upper bound. To that end, we assume $\gamma > 1$ (uniformly bounded away from 1) and observe that $B \ge 2d$ for $B \ge \delta^{-\Omega(1)}$. Then, the upper bound from (21) applies; since the first part of Lemma 2.6

$$\exp(\Psi_{d,\lambda} - \lambda) \le \exp\Big(-\frac{d^2}{2\lambda}\big(1+o(1)\big)\Big) \le \exp\Big(-\log(\delta^{-1})\big(\gamma - o(1)\big)\Big) \le \delta^{\gamma-o(1)} \le \delta,$$

we deduce that $d_{B;\delta}(e^{-x}) \le \big(1+o(1)\big)\sqrt{B\log(\delta^{-1}))}$, which establishes the theorem. □

5 Applications to Batch Gaussian KDE

In this section we prove the statements given in Section 1.2 above about the Batch Gaussian KDE problem.

Proof of Corollary 1.7. Let $B \ge 1$ and $\delta \in (0,1)$ denote real numbers, and suppose $p(z)$ is a univariate polynomial of degree $d = d_{B;\delta}(e^{-x})$ such that

$$\sup_{z\in[0,B]} \big|p(z) - e^{-z}\big| \le \delta.$$

As discussed in the preamble to Corollary 1.7, it suffices to output the vector $\tilde{K}\cdot w$, where $\tilde{K} \in \mathbb{R}^{n\times n}$ is the matrix given by $\tilde{K}[i,j] = p(\sum_{\ell=1}^m (x_\ell^{(i)} - y_\ell^{(j)})^2)$.

For two points $x, y \in \mathbb{R}^m$, we have that $p(\sum_{\ell=1}^m (x_\ell - y_\ell)^2)$ is a polynomial of degree at most $2d$ in the $2m$ variables in $V := \{x_1, \ldots, x_m, y_1, \ldots, y_m\}$. Thus, letting $M_d := \{a : V \to \mathbb{Z}^{\ge 0} \mid \sum_{v\in V} a(v) \le 2d\}$, which has $|M_d| = \binom{2m+2d}{2d} = M$, we can write

$$p(\|x-y\|_2^2) = \sum_{a\in M_d} b_a \cdot \prod_{v\in V} v^{a(v)} \tag{32}$$

for appropriate coefficients $b_a \in \mathbb{R}$ which can all be computed in $O\left(m \cdot \binom{2m+2d}{2d}\right)$ time by expanding p.

It follows that the matrix $\tilde{K}$, whose entries are given by the expression (32), has rank at most $|M_d| = M$, and furthermore that the matrices $X, Y \in \mathbb{R}^{n\times M}$ such that $\tilde{K} = X \times Y^T$ can be computed in $O(n\cdot m\cdot M)$ time by evaluating all the monomials in M_G on the partial assignments of setting $x \leftarrow x^{(i)}$ for each $i \in [n]$ (to compute X) and setting $y \leftarrow y^{(j)}$ for each $j \in [n]$ (to compute Y). We can then, as desired, compute $\tilde{K}w = X(Y^T w)$ in time $O(n\cdot m\cdot M)$. □

Before proving Proposition 1.8, we give the necessary background about approximate nearest neighbor search.

Problem 5.1 (($1+\varepsilon$)-Approximate Hamming Nearest Neighbor)**.** For $\varepsilon > 0$, and positive integers n, m, given as input vectors $a_1, \dots, a_n, b_1, \dots, b_n \in \{0,1\}^m$, as well as an integer $t \in [0, m]$, one must:

- return 'true' if there are $i, j \in [n]$ such that $|a_i - b_j| \leq t$,
- return 'false' if, for every $i, j \in [n]$, we have $|a_i - b_j| > (1+\varepsilon) \cdot t$,

and one may return either 'true' or 'false' otherwise. (Here, $|a_i - b_j|$ denotes the Hamming distance between a_i and b_j.)

Theorem 5.2 ([Rub18])**.** *Assuming SETH, for every $q > 0$, there are $\varepsilon > 0$ and $C > 0$ such that $(1+\varepsilon)$-Approximate Hamming Nearest Neighbor in dimension $m = C \log n$ requires time $\Omega(n^{2-q})$.*

Proof of Proposition 1.8. For any $q > 0$, let $\varepsilon, C > 0$ be the corresponding constants from Theorem 5.2. We will prove that Batch Gaussian KDE requires $\Omega(n^{2-q})$ time when $m = C \log n$, $\delta = n^{-2/\varepsilon - 1}/4$, and $B = 2C \cdot c^{-1} \cdot \varepsilon^{-1} \log n$ for a constant c depending only on q that we will determine later. We will prove this by showing that $(1+\varepsilon)$-Approximate Hamming Nearest Neighbor in dimension m can be solved using $o(n^{2-q})$ time and one call to Batch Gaussian KDE with these parameters, which implies the desired result when combined with Theorem 5.2.

Let $m = C \log n$, let $a_1, \dots, a_n, b_1, \dots, b_n \in \{0,1\}^m$ be the input vectors to $(1+\varepsilon)$-Approximate Hamming Nearest Neighbor, and let $t \in [0, m]$ be the target distance. First, if $t < c \log n$, we will simply brute-force for the answer in the following way: we store the vectors $b_1, \dots, b_n$ in a lookup table, then for each $i \in [n]$, we iterate over every vector $b' \in \{0,1\}^m$ which has Hamming distance at most t from a_i and check whether it is in the lookup table. The running time will be only $O\left(n \cdot \binom{m}{t}\right)$ when c is small enough. In particular,

$$\binom{m}{t} \leq \binom{C \log n}{c \log n} \leq n^{f(C,c)}$$

for some function $f : \mathbb{R}_{>0} \times \mathbb{R}_{>0} \to \mathbb{R}_{>0}$ with the property that, for any fixed $C > 0$, we have $\lim_{c \to 0} f(C, c) = 0$. We can thus pick a sufficiently small constant $c > 0$, depending only on q and C (which itself depends only on q) such that this entire brute-force takes $o(n^{2-q})$ time.

Henceforth, we assume that $t = c' \log n$ for some $C \geq c' > c$. Let $k = \sqrt{2(c'\varepsilon)^{-1}}$. Using our algorithm for Batch Gaussian KDE with the given parameters applied to the input points $ka_1, \dots, ka_n, kb_1, \dots, kb_n$ (i.e., the input points rescaled so they lie in $\{0, k\}^m$), and the weight vector $w = \vec{1} \in \mathbb{R}^n$, the all-1s vector, we get as output a vector $v \in \mathbb{R}^n$ such that, for all $i \in [n]$, we have the guarantee that

$$\left| v_i - \sum_{j=1}^{n} e^{-k^2 \cdot \|x_i - y_j\|_2^2} \right| < n^{-1/\varepsilon}/4.$$

Notice that, for $x_i, y_j \in \{0,1\}^m$, the quantity $\|x_i - y_j\|_2^2$ is equal to the Hamming distance $|x_i - y_j|$ between x_i and y_j. In particular, we have $\max_{i,j \in [n]} \|kx_i - ky_j\|_2^2 \leq k^2 m = (2Cc'^{-1}\varepsilon^{-1}) \log n \leq (2Cc^{-1}\varepsilon^{-1}) \log n = B$, so this was a valid application of our given algorithm.

Suppose first that there are an $i, j \in [n]$ such that $|x_i - y_j| \leq t$. It follows that $v_i \geq e^{-k^2 t} - n^{-2/\varepsilon}/4 = \frac{3}{4} n^{-2/\varepsilon}$.

Suppose second that, for all $i, j \in [n]$, we have $|x_i - y_j| > t(1+\varepsilon)$. It follows that, for all $i \in [n]$, we have $v_i \leq n \cdot e^{-k^2 t(1+\varepsilon)} + n^{-1/\varepsilon}/4 = n^{-2/\varepsilon - 1} + n^{-2/\varepsilon}/4 < \frac{3}{4} n^{-2/\varepsilon}$ for all $n > 2$.

Hence, by checking whether any entry of v is at least $\frac{3}{4}n^{-1/\varepsilon}$, we can distinguish the two cases, as desired. □

We conclude with the proof of Corollary 1.9.

Proof of Corollary 1.9. By Corollary 1.7, it suffices to show that there exists a constant $c = c(\alpha, \beta) > 0$ such that

$$\binom{2d+2m}{2d} \le n^{c \log\log \kappa^{-1} / \log \kappa^{-1}}, \tag{33}$$

for $m = \alpha \log n$ and $d = d_{B;\delta}(e^{-x})$, where $\delta = n^{-\beta}$ and $B = \kappa \log n$. In particular, $B = 2r \log(\delta^{-1})$, where $r = \frac{\kappa}{2\beta}$. Thus, Theorem 1.2 gives

$$2d = \big(\kappa\nu + o(1)\big) \log n, \tag{34}$$

where $\nu = \nu\big(\frac{\kappa}{2\beta}\big) > 0$ is the unique positive solution to $G(\nu) = 1 - r^{-1} = 1 - \frac{2\beta}{\kappa}$ (where G is given by (1)).

To establish (33), we must analyze the behavior of d and thus of ν. We claim that

$$\nu = \frac{2\beta}{\kappa \log \kappa^{-1}} + O\bigg(\frac{\log\log\kappa^{-1}}{\kappa(\log\kappa^{-1})^2}\bigg). \tag{35}$$

Let us quickly establish the corollary assuming (35). To that end, denote

$$x = \kappa\nu = \frac{2\beta}{\log\kappa^{-1}} + O\bigg(\frac{\log\log\kappa^{-1}}{(\log\kappa^{-1})^2}\bigg).$$

Then, (34) and a Taylor expansion together give

$$\begin{aligned}\log\binom{2m+2d}{2d} &= \big((2\alpha + x)\log(2\alpha + x) - 2\alpha\log(2\alpha) - x\log x + o(1)\big)\log n \\ &= \big(x + x\log(2\alpha + x) - x\log x + O(x^2)\big)\log n = x|\log x|\Bigg(1 + O\bigg(\frac{1}{|\log x|}\bigg)\Bigg)\log n,\end{aligned}$$

where the implicit constant in the error depends on α and β. Thus,

$$\binom{2m+2d}{d} \le n^{x|\log x| + O(x)} = n^{2\beta \log\log\kappa^{-1}/\log\kappa^{-1} + O(1/\log\kappa^{-1})},$$

which verifies (33) and thus the corollary.

It thus remains to verify (35). To that end, observe that

$$\frac{2\beta}{\kappa} = 1 - G(\nu) = 1 - \sqrt{\nu^2+1} - \nu\log\big(\sqrt{\nu^2+1} - \nu\big) = \nu\log\big(\sqrt{\nu^2+1} + \nu\big) - \sqrt{\nu^2+1} + 1. \tag{36}$$

In particular, this implies that

$$\nu\log\big(\sqrt{\nu^2+1} + \nu\big) \ge \frac{2\beta}{\kappa}, \qquad \text{so } \nu = \Omega\bigg(\frac{1}{\kappa\log\kappa^{-1}}\bigg),$$

where the implicit constant depends on β. The above lower bound enables us to Taylor expand the right side of (36). This gives

$$\nu\log\nu + \nu(\log 2 - 1) + O(\nu^{-1}) = \frac{2\beta}{\kappa},$$

from which (35) quickly follows. As mentioned above, this verifies the corollary. □

Acknowledgements

We would like to thank Alexei Borodin, Lijie Chen, Andrei Martinez-Finkelshtein, Sushant Sachdeva, Paris Siminelakis, Roald Trigub, Ryan Williams, and anonymous reviewers for helpful discussions and advice throughout this project.

References

[ACSS20] 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)*, pages 541–552, 2020.

[ACW16] Josh Alman, Timothy M Chan, and Ryan Williams. Polynomial representations of threshold functions and algorithmic applications. In *2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 467–476. IEEE, 2016.

[AHPV$^+$05] Pankaj K Agarwal, Sariel Har-Peled, Kasturi R Varadarajan, et al. Geometric approximation via coresets. *Combinatorial and computational geometry*, 52(1), 2005.

[AKK$^+$20] Thomas D Ahle, Michael Kapralov, Jakob BT Knudsen, Rasmus Pagh, Ameya Velingker, David P Woodruff, and Amir Zandieh. Oblivious sketching of high-degree polynomial kernels. In *Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 141–160. SIAM, 2020.

[Amb05] Andris Ambainis. Polynomial degree and lower bounds in quantum complexity: Collision and element distinctness with small range. *Theory of Computing*, 1(1):37–46, 2005.

[AS04] Scott Aaronson and Yaoyun Shi. Quantum lower bounds for the collision and the element distinctness problems. *Journal of the ACM (JACM)*, 51(4):595–605, 2004.

[AW15] Josh Alman and Ryan Williams. Probabilistic polynomials and hamming nearest neighbors. In *2015 IEEE 56th Annual Symposium on Foundations of Computer Science*, pages 136–150. IEEE, 2015.

[AWY14] Amir Abboud, Ryan Williams, and Huacheng Yu. More applications of the polynomial method to algorithm design. In *Proceedings of the twenty-sixth annual ACM-SIAM symposium on Discrete algorithms*, pages 218–230. SIAM, 2014.

[BIS17] Arturs Backurs, Piotr Indyk, and Ludwig Schmidt. On the fine-grained complexity of empirical risk minimization: Kernel methods and neural networks. In *Advances in Neural Information Processing Systems*, pages 4308–4318, 2017.

[BT15] Mark Bun and Justin Thaler. Dual lower bounds for approximate degree and markov–bernstein inequalities. *Information and Computation*, 243:2–25, 2015.

[BT21] Mark Bun and Justin Thaler. Guest column: Approximate degree in classical and quantum computing. *ACM SIGACT News*, 51(4):48–72, 2021.

[CGA15] Welin Chen, David Grangier, and Michael Auli. Strategies for training large vocabulary neural language models. *arXiv preprint arXiv:1512.04906*, 2015.

[CS17] 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. IEEE, 2017.

[CW16] Timothy M Chan and Ryan Williams. Deterministic apsp, orthogonal vectors, and more: Quickly derandomizing razborov-smolensky. In *Proceedings of the twenty-seventh annual ACM-SIAM symposium on Discrete algorithms*, pages 1246–1255. SIAM, 2016.

[GR87] Leslie Greengard and Vladimir Rokhlin. A fast algorithm for particle simulations. *Journal of computational physics*, 73(2):325–348, 1987.

[HL97] Marlis Hochbruck and Christian Lubich. On krylov subspace approximations to the matrix exponential operator. *SIAM Journal on Numerical Analysis*, 34(5):1911–1925, 1997.

[JCG$^+$17] Armand Joulin, Moustapha Cissé, David Grangier, Hervé Jégou, et al. Efficient softmax approximation for gpus. In *International Conference on Machine Learning*, pages 1302–1310. PMLR, 2017.

[Lan50] Cornelius Lanczos. *An iteration method for the solution of the eigenvalue problem of linear differential and integral operators*. United States Governm. Press Office Los Angeles, CA, 1950.

[LLM$^+$19] Jasper CH Lee, Jerry Li, Christopher Musco, Jeff M Phillips, and Wai Ming Tai. Finding the mode of a kernel density estimate. *arXiv preprint arXiv:1912.07673*, 2019.

[MH03] J. C. Mason and D. C. Handscomb. *Chebyshev polynomials*. Chapman & Hall/CRC, Boca Raton, FL, 2003.

[MMS18] Cameron Musco, Christopher Musco, and Aaron Sidford. Stability of the lanczos method for matrix function approximation. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1605–1624. SIAM, 2018.

[NS94] Noam Nisan and Mario Szegedy. On the degree of boolean functions as real polynomials. *Computational complexity*, 4(4):301–313, 1994.

[NSGH14] Peter Nilsson, Ateeq Ur Rahman Shaik, Rakesh Gangarajaiah, and Erik Hertz. Hardware implementation of the exponential function using taylor series. In *2014 NORCHIP*, pages 1–4. IEEE, 2014.

[OSV12] Lorenzo Orecchia, Sushant Sachdeva, and Nisheeth K Vishnoi. Approximating the exponential, the lanczos method and an o (m)-time spectral algorithm for balanced separator. In *Proceedings of the forty-fourth annual ACM symposium on Theory of computing*, pages 1141–1160, 2012.

[Phi13] Jeff M Phillips. ε-samples for kernels. In *Proceedings of the twenty-fourth annual ACM-SIAM symposium on Discrete algorithms*, pages 1622–1632. SIAM, 2013.

[Pow67] Michael JD Powell. On the maximum errors of polynomial approximations defined by interpolation and by least squares criteria. *The Computer Journal*, 9(4):404–407, 1967.

[Rub18] Aviad Rubinstein. Hardness of approximate nearest neighbor search. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, pages 1260–1268, 2018.

[Shi02] Yaoyun Shi. Approximating linear restrictions of boolean functions. In *Manuscript*. Citeseer, 2002.

[SV14] Sushant Sachdeva and Nisheeth K Vishnoi. Faster algorithms via approximation theory. *Foundations and Trends® in Theoretical Computer Science*, 9(2):125–210, 2014.

[Sym19] Paraskevas Syminelakis. *Fast Kernel Evaluation in High Dimensions: Importance Sampling and near Neighbor Search*. Stanford University, 2019.

[Tim94] A. F. Timan. *Theory of approximation of functions of a real variable*. Dover Publications, Inc., New York, 1994. Translated from the Russian by J. Berry, Translation edited and with a preface by J. Cossar, Reprint of the 1963 English translation.

[Tre13] Lloyd N. Trefethen. *Approximation theory and approximation practice*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2013.

[YDGD03] Changjiang Yang, Ramani Duraiswami, Nail A Gumerov, and Larry Davis. Improved fast gauss transform and efficient kernel density estimation. In *Computer Vision, IEEE International Conference on*, volume 2, pages 464–464. IEEE Computer Society, 2003.