Continuous LWE and Its Applications

by

Min Jae Song

A dissertation submitted in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

Department of Computer Science

New York University

May, 2023

Professor Joan Bruna

Professor Oded Regev

Acknowledgments

My PhD journey has been fun and fulfilling. I would first and foremost like to thank my amazing advisors, Joan Bruna and Oded Regev. I have learned immensely from them over the past 5 years. Oded taught me that most problems, no matter how complicated they may seem at first, can always be simplified. One just needs to take a deep breath, ask the right questions, and find the right angle to view them. His emphasis on clarity and conciseness will always occupy my mind. Joan taught me to keep an open mind, be optimistic, and stay curious. His remarkable stamina, demonstrated by his endless curiosity and his ability to continuously work on intricate proofs for hours on end, continues to be a constant source of inspiration for me. I could go on about the countably infinite lessons I've learned from them (I am an avid note-taker), but to keep things concise, I will simply say that it was a true blessing to have had them as my advisors.

I thank my co-authors for the numerous hours of joy and confusion, and few minutes of epiphanies we had together. I thank Yi Tang for being confused about lattices and LWE together and Clayton Sanford for being confused about kernel methods and uniform convergence of loss landscapes together. I'm glad that our confusions have been somewhat alleviated and resulted in papers. I also thank my relatively senior co-authors Alberto Bietti, Alex Wein, and Ilias Zadik for generously sharing their deep insights and expertise. A few pieces of insights they would casually drop during meetings have saved me weeks of "hitting the books".

I thank my friends at NYU for the many chats, laughs, and cups of espresso we had together. Special thanks my cohort Ishan Agarwal, David Brandfonbrener, Mark Goldstein, Aahlad Manas

Puli, Aaron Zweig who have been my support group from the beginning, my office mates Adriel Saporta, Raghav Singhal, Mukund Sudarshan, Mimee Xu, Aaron Zweig, and our espresso (and pizza) connoisseur, Alfredo Canziani. I also thank my fellow NYU Koreans, Kyunghyun Cho, Jason Lee, Taegyun Kim, and Young Kun Ko, for their support and the fun conversations we had exclusively in Korean,

I thank the people I've met and befriended at conferences and workshops, especially the Simons CCSI Workshop and COLT2022. Special thanks to Alex Wein, Eren Kizildag, Navid Ardeshir, and Clayton Sanford for the many laughters and interesting discussions we had together at those exotic locations. I also thank my ML-NYC co-conspirators, Dave Blei, Joan Bruna, David Brandfonbrener, Claudia Shi, and Keyon Vafa, for the many talks, bars, and dinners we enjoyed together.

I thank my thesis committee Joan Bruna, Oded Regev, Jonathan Niles-Weed, Subhash Khot, and Alex Wein. It was an honor to share my work with you.

Lastly, I thank my family, both my immediate family and my in-laws, for their unwavering support. They are the bedrock on which my existence stands. Special thanks to my wife who has been my biggest supporter throughout this entire journey. Every day with you has been a blessing and it will continue to be so in the future. Churro, our corgi, has been my second biggest supporter, although definitely the furriest one.

Abstract

Efficiently extracting useful information from high-dimensional data is a major challenge in machine learning (ML). Oftentimes, the challenge comes not from a lack of data, but from its high dimensionality and computational constraints. For instance, when data exhibits a low-dimensional structure, one could in principle exhaustively search over all candidate structures, and obtain estimators with strong statistical guarantees. Of course, such brute-force approach is prohibitively expensive in high dimensions, necessitating the need for computationally efficient alternatives. When our problem, however, *persistently* eludes efficient algorithms, we may find ourselves asking the following perplexing question: is the failure due to our lack of algorithmic ingenuity or is the problem just too hard? Is there a *gap* between what we can achieve statistically and what we can achieve computationally?

This thesis is one attempt at answering such questions on the *computational complexity of statistical inference*. We provide results of both positive and negative nature on the complexity of canonical learning problems by establishing connections between ML and lattice-based cryptography. The *continuous learning with errors* (CLWE) problem, which can be seen as a continuous variant of the well-known learning with errors (LWE) problem from lattice-based cryptography, lies at the center of this fruitful connection.

In the first part of this thesis, we show that CLWE enjoys essentially the same average-case hardness guarantees as LWE. This result has several important applications. For example, it shows that estimating the density of high-dimensional Gaussian mixtures is computationally

hard, and gives rise to "backdoored" Gaussian distributions that can be used to plant undetectable backdoors in ML models and construct novel public-key encryption schemes.

Next, we focus on the "backdoored" Gaussian distributions, which we refer to as *Gaussian pancakes*, and the problem of distinguishing these distributions from the standard Gaussian. We provide evidence for the hardness of this distinguishing problem based on a reduction from CLWE and lower bounds against restricted classes of algorithms, such as algorithms that compute low-degree polynomials of the observations.

Finally, we end on a positive note by showing that the Lenstra-Lenstra-Lovász (LLL) algorithm, commonly used in computational number theory and lattice-based cryptography, has surprising implications for *noiseless* inference. In particular, we show that LLL solves both CLWE and Gaussian pancakes in the noiseless setting, showing that noise is necessary for hardness. This strengthens the analogy between LWE and CLWE since LWE is also easy in the noiseless case. A minor modification of our LLL-based method in fact surpasses sum-of-squares and approximate message passing algorithms, two methods often conjectured to be optimal among polynomial-time algorithms, on other noiseless problems such as Gaussian clustering and Gaussian phase retrieval.

Contents

List of Figures

1 Introduction

The fundamental idea of statistics is that useful information can be accrued from individual small bits of data. No one [sample] try by itself tells us much, but together the data speak.

Bradley Efron[1]

Statistical inference starts with the assumption that some unknown probability distribution $\mathcal{P}$ over a domain $\mathcal{X}$ has produced the observed data $X \in \mathcal{X}$. Our goal as learners is to perform an "inversion" by *inferring* (or equivalently, *learning*) properties of the unknown $\mathcal{P}$ using X and potentially a priori knowledge to better position ourselves for the future. To make things concrete, consider the *hypothesis testing* setup, in which the goal is to decide[2] whether X has been drawn from some distribution $\mathcal{Q}$ or an alternative distribution $\mathcal{P}$, given the promise that the truth lies within these two choices. This is one of the simplest settings that captures the essence of (frequentist) statistical inference, and thus has long been a central subject of study in classical statistics [Wal50; LRC05; Leh11; LeC12].

As is the case in many other theoretical disciplines, significant efforts in statistics have been devoted to the pursuit of *optimality*. After all, when we have a procedure that works, a natural

[1] [Efr82].

[2] In CS, such problems are indeed called *decision* problems. Statisticians call them *detection* problems when the alternate distribution $\mathcal{P}$ has some "planted" structure. In this case, deciding between $\mathcal{P}$ and $\mathcal{Q}$ can be seen as detecting whether X exhibits the planted structure.

follow-up question is "can it be better?"[3] The concept of optimality is typically defined with respect to some performance measure and various resource constraints. The go-to choice of performance measure in this thesis is Type I + Type II error (i.e., equally weighted errors), which is equivalent to the concept of *advantage* in computer science and cryptography.[4] Type I error is the probability that our decision rule $\Psi : \mathcal{X} \to \{0, 1\}$, i.e., any $\{0, 1\}$-valued function on $\mathcal{X}$, outputs $\Psi(X) = 1$ ("X is from $\mathcal{P}$") when in truth $X \sim \mathcal{Q}$; Type II error is the probability that $\Psi(X) = 0$ ("X is from $\mathcal{Q}$") when in fact $X \sim \mathcal{P}$.

The choice of resource constraints in defining optimality is where modern statistics and machine learning departs from classical statistics (see, e.g., a brief survey on "constrained statistical minimax" by Wainwright [Wai14] and also [Wai19, Chapter 1]). In classical statistics, the dimensionality of the data is typically assumed to be fixed, and the primary resource constraint is the number of available samples or, more generally, a signal-to-noise ratio (SNR) which quantifies statistical discrepancy between $\mathcal{P}$ and $\mathcal{Q}$. Thus, a fundamental question in this field is "What is the minimum number of samples or SNR required to distinguish $\mathcal{P}$ and $\mathcal{Q}$ with high probability?"

Modern datasets tend to be massive, both in terms the number of samples *and* the dimensionality of each sample. For instance, a single photo shot via iPhone 13 has dimension roughly 12×10^6, when represented naively, and Google's Open Images dataset [KRAU+20], one of the largest publicly available image datasets, contains about 9×10^6 images. Consequently, the focus of theoretical and practical efforts has shifted towards the high-dimensional setting in which *both* the dimensionality and number of samples grow simultaneously. With the data dimension no longer fixed, the *computational complexity* of inference algorithms becomes a key resource constraint as well. A statistically-optimal algorithm whose running time scales exponentially in the data dimension would be practically useless to any computationally-bounded being.

[3] Again, we quote Efron: *Optimality results are a mark of scientific maturity.* [Efr98]

[4] More precisely, advantage is equal to 1 − (Type I + Type II error). A random guess which doesn't even look at X achieves 0 advantage.

1.1 Statistical-to-Computational Gaps

The inclusion of computation as a resource opens up a wealth of intriguing theoretical phenomena. One of the most intriguing is the appearance of (conjectured) *statistical-to-computational gaps* (stat-to-comp gaps), gaps between what is achievable statistically (i.e., with infinite time) and what is achievable computationally (i.e., in polynomial time). In the context of hypothesis testing, this means that the observed data X may contain sufficient information for a computationally-unbounded learner to correctly decide whether X is drawn from $\mathcal{P}$ or $\mathcal{Q}$ with high probability, but no efficient algorithm can extract such information. Indeed, more than a decade of research has identified stat-to-comp gaps in many natural inference problems, such as sparse principal component analysis [BR13], submatrix detection [MW15], and sparse linear regression [ZWJ14]. We refer the reader to the surveys [ZK16; BPW18; BB20; WX21; Gam21; KWB22; DK23] and doctoral theses [Hop18; Kun22] for a thorough overview of the literature, and diverse perspectives ranging from information theory and theoretical computer science to statistical physics.

The existence of stat-to-comp gaps is not particularly surprising in itself; public-key cryptography, for example, is precisely built on the existence of such gaps. Our ability to securely transmit private text messages and financial information over public channels relies on the assumption that the encryption scheme generates two distributions (each encoding a single bit 0 or 1, respectively) which cannot be distinguished by any computationally-bounded adversary, but can be easily distinguished using the secret key [KL20, Chapter 10]. What is surprising, however, is the fact that these gaps arise in *natural* and *canonical* inference problems, without any deliberate intent to create them. In light of this surprise, one may then ask the following question:

Can we get cryptographic primitives from natural inference problems?

In fact, the analogy to cryptography unlocks a question in the opposite direction as well:

Can we prove hardness of inference problems using cryptographic assumptions?

In this thesis, we provide positive answers to both questions by introducing a new problem called *continuous learning with errors* (CLWE) and studying its computational complexity. A key contribution of this thesis is showing that hardness of natural inference problems, in particular CLWE and its sibling *Gaussian pancakes*, can be based on *worst-case* hardness assumptions from lattice-based cryptography [MR09]. The proof of this fact comes in the form of a reduction: any algorithm that can achieve "very modest" statistical performance on CLWE can be used to solve *any* instance of lattice problems which are widely believed to be hard.

This has opened up new avenues for showing computational hardness in the context of statistical inference. Previous work on stat-to-comp gaps either show lower bounds against restricted classes of algorithms, such as sum-of-squares (SoS) [BHKK+19], low-degree polynomials [KWB22], and statistical query (SQ) algorithms [Kea98; FGRV+17; DKS17], or assume the average-case hardness of well-known problems such as planted clique and reduce from them [BR13; MW15; BB20]. Our result provides one of the few and rare instances in which hardness of an inference problem is based on well-studied worst-case problems. In fact, prior to our work, experts had expressed skepticism about whether such an approach, namely reduction from worst-case problems, could be successful (see, e.g., [Hop18, Section 1.1] and [BB20, Section 1]).

Another remarkable feature of CLWE and Gaussian pancakes is that they remain hard even when the number of samples is unbounded,[5] which is in contrast to previously studied problems that undergo "impossible-hard-easy" phase transitions depending on the number of observed samples, such as planted vector recovery [MW21] and tensor PCA [RM14; HKPR+17; DH22]. This feature makes CLWE and Gaussian pancakes particularly well-suited for cryptographic applications. Indeed, a follow-up work has demonstrated this by constructing public key cryptosystems based on Gaussian pancakes [BNHR22].

[5] Note, however, that the algorithm's run time immediately puts a restriction on the number of samples it can see.

1.2 Continuous LWE and Gaussian Pancakes

We now briefly define CLWE and Gaussian pancakes. Precise definitions can be found in their respective chapters (Chapters 3 and 4). The CLWE *distribution* with secret $\boldsymbol{u} \in \mathbb{S}^{n-1}$, frequency $\gamma > 0$, and noise level $\beta > 0$ consists of samples of the form $(\boldsymbol{y}, z)$, where $\boldsymbol{y} \in \mathbb{R}^n$ is drawn from the standard Gaussian, $z = \gamma\langle \boldsymbol{y}, \boldsymbol{u}\rangle + e \pmod 1$, and e is drawn independently from the Gaussian distribution of variance β^2. The CLWE *problem* asks one to decide whether the observed samples $(\boldsymbol{y}_i, z_i)$ are from a CLWE distribution with some secret direction $\boldsymbol{u} \in \mathbb{S}^{n-1}$, or the null distribution in which z_i's are independent of $\boldsymbol{y}_i$ and uniformly distributed on $[0, 1]$ (see Figure 1.1). We denote this problem by $\mathrm{CLWE}_{\beta,\gamma}$. The name "continuous" LWE comes from the well-known *learning with errors* (LWE) problem from lattice-based cryptography. The definition of LWE and analogies between the two problems will be given in Chapter 3.

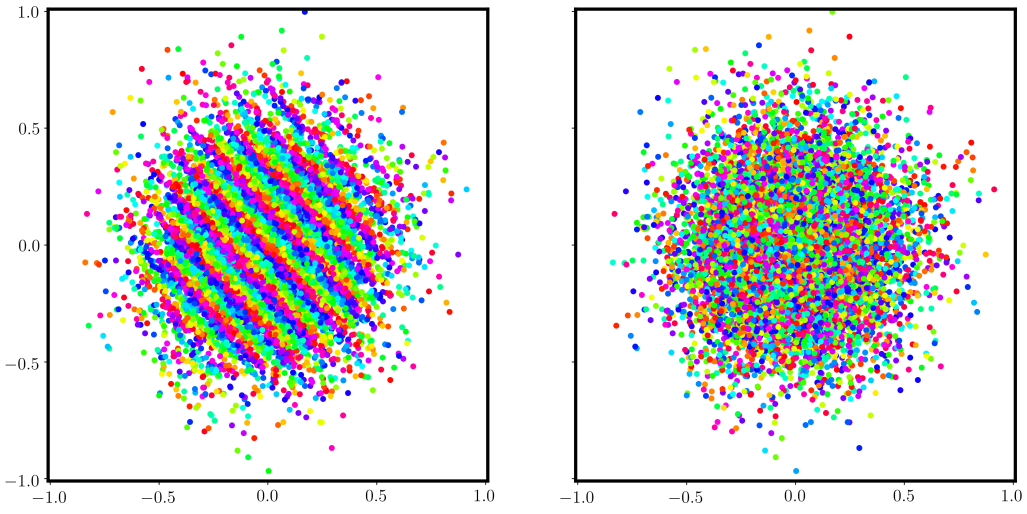


Figure 1.1: Scatter plot of 2D CLWE samples (left) and samples from the null distribution (right). Colors indicate the values of z. Note the periodic structure of the CLWE distribution along the secret direction.

The Gaussian pancakes *distribution* with secret $\boldsymbol{u} \in \mathbb{S}^{n-1}$, spacing $\gamma > 0$, and thickness $\beta > 0$ is a distribution which is a (noisy) discrete Gaussian supported on $(1/\gamma)\mathbb{Z}$ along the direction $\boldsymbol{u}$ and

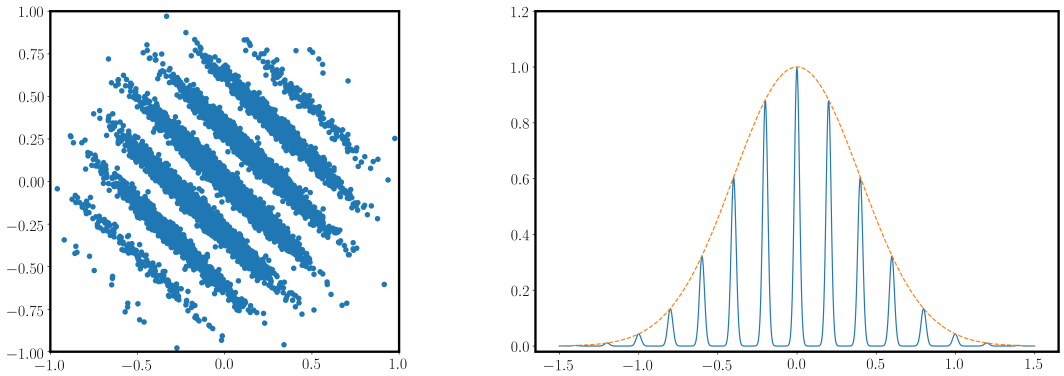


Figure 1.2: Left: Scatter plot of 2D Gaussian pancakes samples. Right: Unnormalized probability densities of Gaussian pancakes (blue) and Gaussian (orange) along the hidden direction.

is standard Gaussian in the remaining $n-1$ directions. β controls the thickness of each "pancake" (see Figure 1.2). The Gaussian pancakes *problem* asks one to decide whether the observed samples are from a Gaussian pancakes distribution with some secret direction $\boldsymbol{u} \in \mathbb{S}^{n-1}$, or the standard Gaussian. Gaussian pancakes can be seen as siblings of CLWE distributions since they can be obtained by conditioning CLWE samples on $z \approx 0$. For this reason, we also refer to Gaussian pancakes as *homogeneous* CLWE (hCLWE) since the samples $\boldsymbol{x}$ satisfy $\gamma\langle \boldsymbol{x}, \boldsymbol{u}\rangle \approx 0 \pmod 1$ and denote the problem by $\mathrm{hCLWE}_{\beta,\gamma}$.

1.3 Summary of Contributions

Results presented in this thesis are collected from three published works [BRST21; SZB21; ZSWB22], and a paper in preparation with Oded Regev and Alex Wein [RSW23]. Some passages have been taken verbatim from the original sources.

Continuous LWE (Chapter 3). We show that CLWE enjoys essentially the same average-case hardness guarantees as LWE. That is, its hardness can be based on fundamental worst-case lattice problems. More precisely, we prove the following theorem.

Theorem 1.1 (Informal version of Theorem 3.15)**.** *Let $n \in \mathbb{N}$, $\beta = \beta(n) \in (0,1)$ and $\gamma = \gamma(n) \geq 2\sqrt{n}$ such that γ/β is polynomially bounded. If there exists an efficient algorithm that solves* $\mathrm{CLWE}_{\beta,\gamma}$*, then there exists an efficient quantum algorithm that approximates worst-case lattice problems to within polynomial factors.*

Our proof follows the blueprint of the original reduction for LWE [Reg05], but with a modification in an intermediate step that gives rise to CLWE instead of LWE. It is worth noting that recent work by [GVV22] has established a direct classical reduction from LWE to CLWE, which allows for the use of the well-developed machinery of LWE in the context of CLWE.

As an example application, we show that hardness of CLWE immediately implies hardness of learning "cosine neurons", a family of functions which has previously been shown to be hard to learn for restricted classes of algorithms. Our result shows that the hardness holds against *any* polynomial-time algorithm, assuming worst-case lattice problems are hard.

Gaussian pancakes (Chapter 4). We show hardness of the Gaussian pancakes problem using a variety of techniques. We first show that CLWE reduces to the Gaussian pancakes problem, resulting in the following highly analogous theorem.

Theorem 1.2 (Informal version of Theorem 4.9)**.** *Let $n \in \mathbb{N}$, $\beta = \beta(n) \in (0,1)$ and $\gamma = \gamma(n) \geq 2\sqrt{n}$ such that γ/β is polynomially bounded. If there exists an efficient algorithm that solves* $\mathrm{hCLWE}_{\beta,\gamma}$*, then there exists an efficient quantum algorithm that approximates worst-case lattice problems to within polynomial factors.*

We corroborate the above theorem with lower bounds against restricted classes of algorithms, such as statistical query algorithms and low-degree polynomials. These alternative approaches provide additional insights into the Gaussian pancakes problem, allowing for an analysis of its complexity beyond the parameters covered by the reduction-based approach.

Hardness of Gaussian pancakes has several important applications. For example, it shows that estimating the density of high-dimensional Gaussian mixtures is computationally hard. More-

over, a standard hybrid argument (see, e.g., [Gol08, Chapter 3]) shows that the problem remains hard for Gaussian "baguettes", i.e., when one has more than 1 discrete directions. These applications will be covered in Chapter 4 as well. We remark that follow-up work has used Gaussian pancakes to plant undetectable backdoors in ML models [GKVZ22] and construct novel public-key encryption schemes [BNHR22].

Lattice-based methods for noiseless inference (Chapter 5). We show that the Lenstra-Lenstra-Lovász (LLL) algorithm, commonly used in computational number theory and lattice-based cryptography, has surprising implications for *noiseless* inference. In particular, we show that LLL solves both CLWE and Gaussian pancakes in the noiseless setting, which is in stark contrast with the hardness that occurs in the noisy setting.

Theorem 1.3 (Informal version of Theorem 5.8)**.** *Let $n \in \mathbb{N}$ and $\gamma = \mathrm{poly}(n)$. There is a polynomial-time algorithm which uses $n+1$ noiseless CLWE_γ (or hCLWE_γ) samples and exactly outputs the secret direction $\boldsymbol{u}$ (up to a sign flip) with probability $1 - \exp(-\Omega(n))$.*

We remark that minor modifications to our LLL algorithm in fact surpasses sum-of-squares and approximate message passing algorithms, two methods often conjectured to be optimal among polynomial-time algorithms, on other noiseless problems such as Gaussian clustering and Gaussian phase retrieval [SZB21; ZSWB22; DK22]. These results highlight the crucial but subtle role of noise and hidden algebraic structure in the onset of statistical-to-computational gaps.

2 Preliminaries

Books may be linearly ordered but our minds are not.

Paul Halmos[1]

2.1 Hypothesis Testing

We define hypothesis testing problems with respect to a sequence of distribution pairs $(\mathcal{P}_n, \mathcal{Q}_n)_{n\in\mathbb{N}}$, where $\mathcal{P}_n$ and $\mathcal{Q}_n$ are distributions on $\mathbb{R}^N$. Here, $N = N(n)$ is the size of the problem instance that scales with n. In this work, we set $N = nm$, with n denoting the dimension of each sample and m the number of samples.[2] Typically, the planted distribution $\mathcal{P}_n$ is generated by a two-step procedure. First, sample the planted signal $\boldsymbol{u}$ uniformly from $\mathcal{S}_n$, the set of possible signals. Then, sample i.i.d. samples from $P_{\boldsymbol{u}}$, the conditional distribution given $\boldsymbol{u}$. For example, in the Gaussian pancakes problem, $\boldsymbol{u}$ is the secret direction, $\mathcal{S}_n$ can either be the unit sphere $\mathbb{S}^{n-1}$ or the Boolean hypercube $\{\pm 1/\sqrt{n}\}^n$. The null distribution $\mathcal{Q}_n$ is usually chosen to be a distribution with nice analytical properties, e.g., the standard Gaussian.

Hypothesis testing for planted structures is defined as follows. The statistician is given $m = m(n)$ i.i.d. samples drawn from an unknown distribution D with the promise that either $D \in \{P_{\boldsymbol{u}}\}_{\boldsymbol{u}\in\mathcal{S}_n}$ or $D = Q_n$, and the goal is to decide between the two cases. The quantity of interest

[1] [Hal13].

[2] In cryptography, the index n is referred to as the *security parameter* [Gol04; KL20].

associated with a decision rule is the *advantage* (terminology from cryptography [Gol04; KL20]).

Definition 2.1 (Advantage)**.** We define the *advantage* of a decision rule $\Psi : \mathcal{X} \to \{0, 1\}$ solving the decision problem of distinguishing two distributions $\mathcal{P}$ and $\mathcal{Q}$ over the domain $\mathcal{X}$ by

$$\mathsf{Adv}(\Psi) = \left| \Pr_{x \sim \mathcal{P}}[\Psi(x) = 1] - \Pr_{x \sim \mathcal{Q}}[\Psi(x) = 1] \right| .$$

Note that in statistical parlance, $\mathsf{Adv}(\Psi)$ is equivalent to Type I + Type II error since

$$\mathsf{Adv}(\Psi) = 1 - (\text{Type I error} + \text{Type II error}) .$$

We are interested in distribution pairs $(\mathcal{P}_n, \mathcal{Q}_n)$ which cannot be distinguished by any efficiently computable decision rule. To formalize this, we define the notion of computational indistinguishability.

Definition 2.2 (Computational Indistinguishability)**.** We say a sequence of distribution pairs $(\mathcal{P}_n, \mathcal{Q}_n)$ over a common domain $\mathcal{X}_n$ are *computationally indistinguishable* if for any efficiently computable decision rule $\Psi_n : \mathcal{X}_n \to \{0, 1\}$,

$$|\mathcal{P}_n(\Psi_n) - \mathcal{Q}_n(\Psi_n)| = \mathrm{negl}(n) . \tag{2.1}$$

Definition 2.3 (Statistical distance)**.** For two distributions $\mathcal{P}$ and $\mathcal{Q}$ over $\mathbb{R}^n$ with density functions ϕ_1 and ϕ_2, respectively, we define the *statistical distance* between them as

$$\Delta(\mathcal{P}, \mathcal{Q}) = \frac{1}{2} \int_{\mathbb{R}^n} |p(\boldsymbol{x}) - q(\boldsymbol{x})| d\boldsymbol{x} .$$

We denote the statistical distance by $\Delta(p, q)$ if the density functions are specified. Moreover, for random variables $X_1 \sim \mathcal{P}$ and $X_2 \sim \mathcal{Q}$, we also denote $\Delta(X_1, X_2) = \Delta(\mathcal{P}, \mathcal{Q})$. One important fact is that applying (possibly a randomized) function cannot increase statistical distance, i.e., for

any pair of random vectors X, Y and any function f,

$$\Delta(f(X), f(Y)) \leq \Delta(X, Y) \ .$$

2.2 Lattices and Discrete Gaussians

Lattices. A *lattice* is a discrete additive subgroup of $\mathbb{R}^n$. Unless specified otherwise, we assume all lattices are full rank, i.e., their linear span is $\mathbb{R}^n$. For an n-dimensional lattice Λ, a set of linearly independent vectors $B = \{\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n\}$ is called a *basis* of Λ if Λ is generated by taking integer combinations of B. The *determinant* of a lattice Λ with basis B is defined as $\det(\Lambda) = |\det(B)|$.

The *dual lattice* of a lattice Λ, denoted by Λ^*, is defined as

$$\Lambda^* = \{\boldsymbol{y} \in \mathbb{R}^n \mid \langle \boldsymbol{x}, \boldsymbol{y} \rangle \in \mathbb{Z} \text{ for all } \boldsymbol{x} \in \Lambda\} \ .$$

If B is a basis of Λ then $(B^\top)^{-1}$ is a basis of Λ^*. In particular, $\det(\Lambda^*) = \det(\Lambda)^{-1}$.

Discrete Gaussians. We define $\rho_r : \mathbb{R}^n \to \mathbb{R}$, the *Gaussian function* of width $r > 0$, by

$$\rho_s(\boldsymbol{x}) = \exp\left(-\frac{1}{2} \cdot \frac{\|\boldsymbol{x}\|^2}{r^2}\right) \ .$$

We denote the Gaussian mass of a lattice $\Lambda \subset \mathbb{R}^n$ by $\rho_s(\Lambda) = \sum_{\boldsymbol{x} \in \Lambda} \rho_s(\boldsymbol{x})$.

Definition 2.4 (Discrete Gaussian)**.** Let $\Lambda \subset \mathbb{R}^n$ be a lattice, $\boldsymbol{c} \in \mathbb{R}^n$, and $r > 0$. The *discrete Gaussian* $D_{\boldsymbol{c}+\Lambda,r}$ on coset $\boldsymbol{c} + \Lambda$ of width r is a discrete distribution supported on $\boldsymbol{c} + \Lambda$ with probability mass function proportional to ρ_r.

For $\boldsymbol{c} = \boldsymbol{0}$, we simply denote the discrete Gaussian distribution on lattice Λ with width r by $D_{\Lambda,r}$. We omit the subscript r when $r = 1$ and simply write D_Λ. Furthermore, we denote the centered continuous Gaussian of variance s^2 by D_s. We remark that the discrete Gaussian is

not the same as the rounded Gaussian, which is given by first sampling $x \sim \mathcal{N}(0,1)$ and then rounding x to the nearest lattice point.[3]

Claim 2.5 ([Pei10, Fact 2.1]). *For any $r_1, r_2 > 0$ and vectors $\boldsymbol{x}, \boldsymbol{c}_1, \boldsymbol{c}_2 \in \mathbb{R}^n$, let $r_0 = \sqrt{r_1^2 + r_2^2}$, $r_3 = r_1 r_2 / r_0$, and $\boldsymbol{c}_3 = (r_3/r_1)^2 \boldsymbol{c}_1 + (r_3/r_2)^2 \boldsymbol{c}_2$. Then*

$$\rho_{r_1}(\boldsymbol{x} - \boldsymbol{c}_1) \cdot \rho_{r_2}(\boldsymbol{x} - \boldsymbol{c}_2) = \rho_{r_0}(\boldsymbol{c}_1 - \boldsymbol{c}_2) \cdot \rho_{r_3}(\boldsymbol{x} - \boldsymbol{c}_3) \ .$$

Fourier analysis. We briefly review basic tools of Fourier analysis required later on. The Fourier transform of a function $f : \mathbb{R}^n \to \mathbb{C}$ is defined by

$$\hat{f}(\boldsymbol{y}) = \int_{\mathbb{R}^n} f(\boldsymbol{x}) e^{-2\pi i \langle \boldsymbol{x}, \boldsymbol{y} \rangle} d\boldsymbol{x} \ .$$

An elementary property of the Fourier transform is that if $f(\boldsymbol{y}) = g(\boldsymbol{y} + \boldsymbol{c})$ for some $\boldsymbol{c} \in \mathbb{R}^n$, then $\hat{f}(\boldsymbol{y}) = e^{2\pi i \langle \boldsymbol{c}, \boldsymbol{y} \rangle} \hat{g}(\boldsymbol{y})$. Another important fact is that the Fourier transform of a Gaussian is also a Gaussian. More precisely, $\hat{\rho}_s = (s\sqrt{2\pi})^n \rho_{1/(2\pi s)}$. We also exploit the Poisson summation formula stated below. Note that we denote by $f(S) = \sum_{\boldsymbol{x} \in S} f(\boldsymbol{x})$ for any function f and any discrete set S.

Lemma 2.6 (Poisson summation formula). *For any lattice Λ and any function f,*[4]

$$f(\Lambda) = \det(\Lambda^*) \cdot \widehat{f}(\Lambda^*) \ .$$

Smoothing parameter. An important lattice parameter induced by discrete Gaussian which will repeatedly appear in our work is the *smoothing parameter*, defined as follows.

Definition 2.7 (Smoothing parameter). Let $\Lambda \subset \mathbb{R}^n$ be a lattice and let $\varepsilon > 0$. We define the

[3] In fact, it can be shown that the statistical distance between the discrete Gaussian and the rounded Gaussian on $(1/\gamma)\mathbb{Z}$, where $\gamma = \mathrm{poly}(n)$, is at least $\Omega(1/\gamma^3)$. See discussion in [GPV08, Section 4] for more details.

[4] To be precise, f must satisfy some niceness conditions; this will always hold in our applications.

smoothing parameter $\eta_\varepsilon(\Lambda)$ by

$$\eta_\varepsilon(\Lambda) = \inf\{s \mid \rho_{1/(2\pi s)}(\Lambda^* \setminus \{\mathbf{0}\}) \leq \varepsilon\} .$$

Intuitively, this parameter is the width beyond which the discrete Gaussian distribution "behaves" like a continuous Gaussian. This is formalized in the lemmas below.

Lemma 2.8 ([Reg09, Claim 3.9])**.** *For any lattice* $\Lambda \subset \mathbb{R}^n$*, vector* $\boldsymbol{c} \in \mathbb{R}^n$*, and* $r, s > 0$ *satisfying* $rs/t \geq \eta_\varepsilon(\Lambda)$ *for some* $\varepsilon < 1/2$*, where* $t = \sqrt{r^2 + s^2}$*, the statistical distance between* $Q_{\boldsymbol{c}+\Lambda,r} + Q_s$ *and* Q_t *is at most* 4ε*.*

Lemma 2.8 states that if we take a sample from $Q_{\Lambda,r}$ and add continuous Gaussian noise Q_s to the sample, the resulting distribution is statistically close to $Q_{\sqrt{r^2+s^2}}$, which is precisely what one would expect from adding two *continuous* Gaussian random variables of variance r^2 and s^2, respectively. Unless specified otherwise, we always assume ε is negligibly small in n, say $\varepsilon = \exp(-\omega(\log n))$. The following is a useful upper bound on the smoothing parameter $\eta_\varepsilon(\Lambda)$.

Lemma 2.9 ([MP12, Lemma 2.3])**.** *Let* $\Lambda \subset \mathbb{R}^n$ *be a lattice with (ordered) basis* $B = \{\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n\}$*, let* $\tilde{B}$ *be the Gram-Schmidt orthogonalization of* B*, and let* $\varepsilon > 0$*. Then,*

$$\eta_\varepsilon(\Lambda) \leqslant \|\tilde{B}\| \cdot \sqrt{\ln(2n(1 + 1/\varepsilon))/\pi} .$$

In particular, for any $\omega(\sqrt{\log n})$ *function, there is a negligible* $\varepsilon = \varepsilon(n)$ *for which* $\eta_\varepsilon(\Lambda) \leqslant \|\tilde{B}\| \cdot \omega(\sqrt{\log n})$*.*

Specializing to the one-dimensional lattice $(1/\gamma)\mathbb{Z}$, we have the following useful corollary.

Corollary 2.10 (Smoothing parameter for $(1/\gamma)\mathbb{Z}$)**.** *Let* $\gamma = \gamma(n)$*. There exists some constant* $c > 0$ *such that* $\eta_\varepsilon((1/\gamma)\mathbb{Z}) \leqslant 1$ *for any* $\varepsilon \leqslant \exp(-c\gamma^2)$*.*

3 | Continuous LWE

Great talking to you yesterday! I would be interested to see if we can prove a computational hardness results based on LWE for their distributions.

Oded Regev[1]

The Learning with Errors (LWE) problem has served as a foundation for many lattice-based cryptographic schemes [MR09; Pei16]. Informally, LWE asks one to solve noisy random linear equations. To be more precise, the goal is to find a secret vector $\boldsymbol{s} \in \mathbb{Z}_q^n$ given polynomially many samples of the form $(\boldsymbol{a}_i, b_i)$, where $\boldsymbol{a}_i \in \mathbb{Z}_q^n$ is uniformly chosen and $b_i \approx \langle \boldsymbol{a}_i, \boldsymbol{s} \rangle \pmod q$. In the absence of noise, LWE can be efficiently solved using Gaussian elimination. However, LWE is known to be hard assuming hardness of worst-case lattice problems, such as the gap shortest vector problem (GapSVP) or shortest independent vectors problem (SIVP), in the sense that there is a polynomial-time quantum reduction from these worst-case lattice problems to LWE [Reg05].

Continuous LWE (CLWE), which we introduced in Section 1.2, can be seen as a continuous analogue of LWE, where equations in $\mathbb{Z}_q^n$ are replaced with ones in $\mathbb{R}^n$ (recall Figure 1.1). This analogy provides a natural motivation to study CLWE, as it allows us to extend our understanding of LWE to the continuous domain. Our main result is that CLWE enjoys hardness guarantees

[1] Email from Oded to Joan and me on Feb 19, 2019. "Their" refers to [BLPR19], who conjectured that distinguishing Gaussian pancakes from the standard Gaussian is hard for *any* polynomial-time algorithm and suggested leveraging cryptographic assumptions to prove this. Our motivation for defining Continuous LWE and proving its hardness was largely influenced by this open problem.

similar to those of LWE. More precisely, we show the following theorem.

Theorem 3.1 (Informal). *Let $n \in \mathbb{N}$, $\beta = \beta(n) \in (0,1)$ and $\gamma = \gamma(n) \geq 2\sqrt{n}$ such that the ratio γ/β is polynomially bounded. If there exists an efficient algorithm that solves $\mathrm{CLWE}_{\beta,\gamma}$, then there exists an efficient quantum algorithm that approximates worst-case lattice problems to within polynomial factors.*

The continuous nature of CLWE makes it a promising candidate for establishing connections between statistical inference problems, which are naturally defined in the continuous domain, and worst-case lattice problems. Indeed, formulated as a supervised learning problem, CLWE can be seen as a generalized linear model (GLM) [NW72], a class of functions extensively studied in the context of high-dimensional inference (see e.g., [GZ17; BKMM+19; MAB20]). GLMs have the form $f(\boldsymbol{y}) = \phi(\langle \boldsymbol{u}, \boldsymbol{y} \rangle)$, where $\boldsymbol{u} \in \mathbb{R}^n$ is an unknown parameter and the univariate function $\phi : \mathbb{R} \to \mathbb{R}$ (called the *link* function) is known. Parameterizing the model with $\boldsymbol{u} \in \mathbb{S}^{n-1}$, CLWE is essentially a GLM with the sequence of link functions $\phi_n(t) = \gamma t \pmod 1$.[2]

Connections to other GLMs can be established by applying different 1-periodic functions to labels of CLWE samples. One notable example is the class of *cosine neurons* which are functions defined by $f(\boldsymbol{y}) = \cos(2\pi\gamma\langle \boldsymbol{u}, \boldsymbol{y} \rangle)$. Previous work [SVWX17; Sha18] has shown that learning cosine neurons over Gaussian input distributions is hard for restricted classes of algorithms, such as gradient-based methods. In Section 3.4, we show that the hardness extends to *any* polynomial-time learning algorithm if small noise is added to the labels. Since cosine neurons can be approximated by one-hidden-layer neural networks, our result implies that learning (noisy) neural networks over the Gaussian distribution is hard. This adds to the growing literature on hardness of learning neural networks over Gaussian distributions [GGJK+20; DKKZ20; DV21; CGKM22]. We list additional motivation for studying CLWE below.

[2] This is somewhat non-standard. In GLMs, link functions are typically fixed, i.e., $\phi_n = \phi$ for all $n \in \mathbb{N}$.

Cryptographic applications. Given the wide range of cryptographic applications of LWE [Reg05; MR09; Pei16], it is only natural to expect that CLWE would also be useful for cryptography. CLWE's clean and symmetric definition should make it a better fit for some applications; its continuous nature, however, might require a discretization step due to efficiency considerations.

Algorithms for lattice problems. Another motivation to study CLWE is as a possible approach to finding quantum algorithms for lattice problems. Indeed, our reduction, just like the reduction to LWE [Reg05], can be interpreted in an algorithmic way: in order to quantumly solve worst-case lattice problems, "all" we have to do is solve CLWE (classically or quantumly). The elegant geometric nature of CLWE and its connection to Gaussian pancakes (see Chapter 4) opens up a new toolbox of techniques that can potentially be used for solving lattice problems, such as sum-of-squares-based techniques and algorithms for learning mixtures of Gaussians [MV10].

Looking ahead, we note that the simple moment-based subexponential time algorithm from Section 4.5 demonstrates the usefulness of CLWE as an algorithmic target. Even though this does not imply subexponential time algorithms for lattice problems (since Theorem 1.1 requires $\gamma > \sqrt{n}$), it is interesting to contrast this algorithm with an analogous algorithm for LWE by Arora and Ge [AG11]. The two algorithms have the same running time (where γ is replaced by the absolute noise αq in the LWE samples), and both rely on related techniques (moments in our case, powering in Arora-Ge's), yet the Arora-Ge algorithm is technically more involved than our rather trivial algorithm (which just amounts to computing the empirical covariance matrix). We interpret this as an encouraging sign that CLWE might be a better algorithmic target than LWE.

Analogy with LWE. As mentioned above, there are non-trivial differences between CLWE and LWE, especially in terms of possible algorithmic approaches. However, there is undoubtedly also strong similarity between the two. In terms of parameters, the γ parameter in CLWE (density of layers) plays the role of the absolute noise level αq in LWE. And the β parameter in CLWE plays the role of the relative noise parameter α in LWE. Using this correspondence between the

parameters, the hardness proved for CLWE in Theorem 1.1 is essentially identical to the one proved for LWE in [Reg05]. The similarity extends even to the noiseless case, where $\alpha = 0$ in LWE and $\beta = 0$ in CLWE. In particular, there is an efficient LLL-based algorithm for solving noiseless CLWE, which is analogous to Gaussian elimination for noiseless LWE (see Chapter 5).

3.1 Technical Overview

Broadly speaking, our proof follows the iterative structure of the original LWE hardness proof [Reg05] (in fact, one might say most of the ingredients for CLWE were already present in that 2005 paper!). We also make use of some recent techniques, such as a way to reduce to decision problems directly [PRS17].

In more detail, as in previous work, our main theorem boils down to solving the following problem: we are given a $\mathrm{CLWE}_{\beta,\gamma}$ oracle and polynomially many samples from $D_{L,r}$, the discrete Gaussian distribution on L of width r,[3] and our goal is to solve $\mathrm{BDD}_{L^*,\gamma/r}$, which is the problem of finding the closest vector in the dual lattice L^* given a vector $\boldsymbol{t}$ that is within distance γ/r of L^*. (It is known that $\mathrm{BDD}_{L^*,1/r}$ can be efficiently solved even if all we are given is polynomially many samples from $D_{L,r}$, without any need for an oracle [AR05]; the point here is that the CLWE oracle allows us to extend the decoding radius from $1/r$ to γ/r.) Once this is established, the main theorem follows from previous work [PRS17; Reg05]. Very briefly, the resulting BDD solution is used in a quantum procedure to produce discrete Gaussian samples that are shorter than the ones we started with. This process is then repeated, until eventually we end up with the desired short discrete Gaussian samples. We remark that this process incurs a $\sqrt{n}$ loss in the Gaussian width (Lemma 3.18), and the reason we require $\gamma \geqslant 2\sqrt{n}$ is to overcome this loss.

We now explain how we solve the above problem. For simplicity, assume for now that we have a *search* CLWE oracle that recovers the secret exactly. (Our actual reduction is stronger and

[3] We actually require samples from D_{L,r_i} for polynomially many r_i's satisfying $r_i \geq r$, see Section 3.3.

only requires a *decision* CLWE oracle.) Let the given BDD instance be $\boldsymbol{a} + \boldsymbol{u}$, where $\boldsymbol{a} \in L^*$ and $\|\boldsymbol{u}\| = \gamma/r$. We will consider the general case of $\|\boldsymbol{u}\| \leqslant \gamma/r$ in Section 3.3. The main idea is to generate CLWE samples whose secret is essentially the desired BDD solution $\boldsymbol{u}$, which would then complete the proof. To begin, take a sample from the discrete Gaussian distribution $\boldsymbol{y} \sim D_{L,r}$ (as provided to us) and consider the inner product

$$\langle \boldsymbol{y}, \boldsymbol{a} + \boldsymbol{u} \rangle = \langle \boldsymbol{y}, \boldsymbol{u} \rangle \pmod 1 ,$$

where the equality holds since $\langle \boldsymbol{y}, \boldsymbol{a} \rangle \in \mathbb{Z}$ by definition. The $(n+1)$-dimensional vector $(\boldsymbol{y}, \langle \boldsymbol{y}, \boldsymbol{u} \rangle \bmod 1)$ is almost a CLWE sample (with parameter γ since $\gamma = r\|\boldsymbol{u}\|$ is the width of $\langle \boldsymbol{y}, \boldsymbol{u} \rangle$) — the only problem is that in CLWE the $\boldsymbol{y}$'s need to be distributed according to a standard Gaussian, but here the $\boldsymbol{y}$'s are distributed according to a *discrete* Gaussian over L. To complete the transformation into bonafide CLWE samples, we add Gaussian noise of appropriate variance to both $\boldsymbol{y}$ and $\langle \boldsymbol{y}, \boldsymbol{u} \rangle$ (and rescale $\boldsymbol{y}$ so that it is distributed according to the standard Gaussian distribution). We then apply the search $\mathrm{CLWE}_{\beta,\gamma}$ oracle on these CLWE samples to recover $\boldsymbol{u}$ and thereby solve $\mathrm{BDD}_{L^*,\gamma/r}$.

As mentioned previously, our main result actually uses a *decision* CLWE oracle, which does not recover the secret $\boldsymbol{u}$ immediately. Working with this decision oracle requires some care. To that end, our proof will incorporate the "oracle hidden center" finding procedure from [PRS17], the details of which can be found in Section 3.3.3.

3.2 Preliminaries

Chapter-specific notation. In this chapter, we instead denote $\rho_s(\boldsymbol{x}) = \exp(-\pi\|\boldsymbol{x}/s\|^2)$. This rescaling is standard in lattice-based cryptography. In fact, the isotropic Gaussian of covariance $I_n/(2\pi)$ (rather than I_n) is often referred to as the *standard* Gaussian in the literature. For lattice

$L \subset \mathbb{R}^n$, $r > 0$, and vector $\boldsymbol{y} \in \mathbb{R}^n$ we denote the discrete Gaussian by $D_{\boldsymbol{y}+L,r}$. For $\boldsymbol{y} = \boldsymbol{0}$, we simply denote the discrete Gaussian distribution on lattice L with width r by $D_{L,r}$. Abusing notation, we denote the n-dimensional *continuous* Gaussian distribution with zero mean and isotropic variance $r^2/(2\pi)$ by $D_{\mathbb{R}^n,r}$. Finally, we omit the subscript r when $r = 1$.

3.2.1 Worst-case lattice problems

GapSVP and SIVP are among the main computational problems on lattices and are believed to be computationally hard (even with quantum computation) for polynomial approximation factor $\alpha(n)$. We also define two additional problems, DGS and BDD.

Definition 3.2 (GapSVP)**.** For an approximation factor $\alpha = \alpha(n)$, an instance of GapSVP_α is given by an n-dimensional lattice L and a number $d > 0$. In YES instances, $\lambda_1(L) \leq d$, whereas in NO instances, $\lambda_1(L) > \alpha \cdot d$.

Definition 3.3 (SIVP)**.** For an approximation factor $\alpha = \alpha(n)$, an instance of SIVP_α is given by an n-dimensional lattice L. The goal is to output a set of n linearly independent lattice vectors of length at most $\alpha \cdot \lambda_n(L)$.

Definition 3.4 (DGS)**.** For a function φ that maps lattices to non-negative reals, an instance of DGS_φ is given by a lattice L and a parameter $r \geq \varphi(L)$. The goal is to output an independent sample whose distribution is within negligible statistical distance of $D_{L,r}$.

Definition 3.5 (BDD)**.** For an n-dimensional lattice L and distance bound $d > 0$, an instance of $\mathrm{BDD}_{L,d}$ is given by a vector $\boldsymbol{t} = \boldsymbol{w} + \boldsymbol{u}$, where $\boldsymbol{u} \in L$ and $\|\boldsymbol{w}\| \leq d$. The goal is to output $\boldsymbol{w}$.

3.2.2 Bounds on the smoothing parameter

Lemma 3.6 ([PRS17, Lemma 2.5])**.** *For any n-dimensional lattice L, real $\varepsilon > 0$, and $r \geq \eta_\varepsilon(L)$, the statistical distance between $D_{\mathbb{R}^n,r} \bmod L$ and the uniform distribution over $\mathbb{R}^n/L$ is at most $\varepsilon/2$.*

The following are some useful upper and lower bounds on the smoothing parameter $\eta_\varepsilon(L)$.

Lemma 3.7 ([PRS17, Lemma 2.6]). *For any n-dimensional lattice L and $\varepsilon = \exp(-c^2 n)$,*

$$\eta_\varepsilon(L) \leq c\sqrt{n}/\lambda_1(L^*) \ .$$

Lemma 3.8 ([MR07, Lemma 3.3]). *For any n-dimensional lattice L and $\varepsilon > 0$,*

$$\eta_\varepsilon(L) \leq \sqrt{\frac{\ln(2n(1+1/\varepsilon))}{\pi}} \cdot \lambda_n(L) \ .$$

Lemma 3.9 ([Reg05, Claim 2.13]). *For any n-dimensional lattice L and $\varepsilon > 0$,*

$$\eta_\varepsilon(L) \geq \sqrt{\frac{\ln 1/\varepsilon}{\pi}} \cdot \frac{1}{\lambda_1(L^*)} \ .$$

3.2.3 Learning with errors

We now define the learning with errors (LWE) problem. This definition will not be used in the sequel, and is included for completeness. Let n and q be positive integers, and $\alpha > 0$ an error rate. We denote the quotient ring of integers modulo q as $\mathbb{Z}_q = \mathbb{Z}/q\mathbb{Z}$ and quotient group of reals modulo the integers as $\mathbb{T} = \mathbb{R}/\mathbb{Z} = [0, 1)$.

Definition 3.10 (LWE distribution). For integer $q \geqslant 2$ and vector $\boldsymbol{s} \in \mathbb{Z}_q^n$, the *LWE distribution* $P_{\boldsymbol{s},\alpha,q}$ over $\mathbb{Z}_q^n \times \mathbb{T}$ is sampled by independently choosing uniformly random $\boldsymbol{a} \in \mathbb{Z}_q^n$ and $e \sim D_{\mathbb{R},\alpha}$, and outputting $(\boldsymbol{a}, (\langle \boldsymbol{a}, \boldsymbol{s} \rangle / q + e) \bmod 1)$.

Definition 3.11. For an integer $q = q(n) \geq 2$ and error parameter $\alpha = \alpha(n) > 0$, the average-case decision problem $\mathrm{LWE}_{q,\alpha}$ is to distinguish the following two distributions over $\mathbb{Z}_q^n \times \mathbb{T}$: (1) the LWE distribution $P_{\boldsymbol{s},\alpha,q}$ for some uniformly random $\boldsymbol{s} \in \mathbb{Z}_q^n$ (which is fixed for all samples), or (2) the uniform distribution.

3.2.4 Continuous learning with errors

We now define the CLWE distribution, which is the central subject of our analysis.

Definition 3.12 (CLWE distribution)**.** For unit vector $\boldsymbol{u} \in \mathbb{R}^n$ and parameters $\beta, \gamma > 0$, define the *CLWE distribution* $P_{\boldsymbol{u},\beta,\gamma}$ over $\mathbb{R}^{n+1}$ to have density at $(\boldsymbol{y}, z)$ proportional to

$$\rho(\boldsymbol{y}) \cdot \sum_{k \in \mathbb{Z}} \rho_\beta(z + k - \gamma\langle \boldsymbol{y}, \boldsymbol{u}\rangle) \ .$$

Equivalently, a sample $(\boldsymbol{y}, z)$ from the CLWE distribution $P_{\boldsymbol{u},\beta,\gamma}$ is given by the $(n{+}1)$-dimensional vector $(\boldsymbol{y}, z)$ where $\boldsymbol{y} \sim D_{\mathbb{R}^n}$ and $z = (\gamma\langle \boldsymbol{y}, \boldsymbol{u}\rangle + e) \bmod 1$ where $e \sim D_{\mathbb{R},\beta}$. The vector $\boldsymbol{u}$ is the hidden direction, γ is the density of layers, and β is the noise added to each equation.

Definition 3.13. For parameters $\beta, \gamma > 0$, the average-case decision problem $\mathrm{CLWE}_{\beta,\gamma}$ is to distinguish the following two distributions over $\mathbb{R}^n \times \mathbb{T}$: (1) the CLWE distribution $P_{\boldsymbol{u},\beta,\gamma}$ for some uniformly random unit vector $\boldsymbol{u} \in \mathbb{R}^n$ (which is fixed for all samples), or (2) $D_{\mathbb{R}^n} \times U$.

Note that $\mathrm{CLWE}_{\beta,\gamma}$ is defined as an average-case problem. We could have equally well defined them to be worst-case problems, requiring the algorithm to distinguish the distributions for *all* hidden directions $\boldsymbol{u} \in \mathbb{R}^n$. The following claim shows that the two formulations are equivalent.

Claim 3.14. *For any $\beta, \gamma > 0$, there is a polynomial-time reduction from worst-case* $\mathrm{CLWE}_{\beta,\gamma}$ *to (average-case)* $\mathrm{CLWE}_{\beta,\gamma}$.

Proof. Given CLWE samples $\{(\boldsymbol{y}_i, z_i)\}_{i=1}^{\mathrm{poly}(n)}$ from $P_{\boldsymbol{u},\beta,\gamma}$, we apply a random rotation R, giving us samples of the form $\{(R\boldsymbol{y}_i, z_i)\}_{i=1}^{\mathrm{poly}(n)}$. Since the standard Gaussian is rotationally invariant and $\langle \boldsymbol{y}, \boldsymbol{u}\rangle = \langle R\boldsymbol{y}, R^T\boldsymbol{u}\rangle$, the rotated CLWE samples are distributed according to $P_{R^T\boldsymbol{u},\beta,\gamma}$. Since R is a random rotation, the random direction $R^T\boldsymbol{u}$ is uniformly distributed on the sphere. □

3.3 Hardness of CLWE

3.3.1 LWE background and reduction overview

In this section, we give an overview of the quantum reduction from worst-case lattice problems to CLWE. Our goal is to show that we can efficiently solve worst-case lattice problems, in particular GapSVP and SIVP, using an oracle for CLWE (and with quantum computation). We first state our main theorem, which was stated informally as Theorem 1.1 in the introduction.

Theorem 3.15. *Let $\beta = \beta(n) \in (0,1)$ and $\gamma = \gamma(n) \geq 2\sqrt{n}$ be such that γ/β is polynomially bounded. Then there is a polynomial-time quantum reduction from $\mathrm{DGS}_{2\sqrt{n}\eta_\varepsilon(L)/\beta}$ to $\mathrm{CLWE}_{\beta,\gamma}$.*

Using standard reductions from GapSVP and SIVP to DGS (see, e.g., [Reg05, Section 3.3]), our main theorem immediately implies the following corollary.

Corollary 3.16. *Let $\beta = \beta(n) \in (0,1)$ and $\gamma = \gamma(n) \geq 2\sqrt{n}$ such that γ/β is polynomially bounded. Then, there is a polynomial-time quantum reduction from SIVP_α and GapSVP_α to $\mathrm{CLWE}_{\beta,\gamma}$ for some $\alpha = \tilde{O}(n/\beta)$.*

Based on previous work, to prove Theorem 3.15, it suffices to prove the following lemma, which is the goal of this section.

Lemma 3.17. *Let $\beta = \beta(n) \in (0,1)$ and $\gamma = \gamma(n) \geq 2\sqrt{n}$ such that $q = \gamma/\beta$ is polynomially bounded. There exists a probabilistic polynomial-time (classical) algorithm with access to an oracle that solves $\mathrm{CLWE}_{\beta,\gamma}$, that takes as input a lattice $L \subset \mathbb{R}^n$, parameters β, γ, and $r \geq 2q \cdot \eta_\varepsilon(L)$, and $\mathrm{poly}(n)$ many samples from the discrete Gaussian distribution D_{L,r_i} for $\mathrm{poly}(n)$ parameters $r_i \geq r$ and solves $\mathrm{BDD}_{L^*,d}$ for $d = \gamma/(\sqrt{2}r)$.*

In other words, we can implement an oracle for $\mathrm{BDD}_{L^*,\gamma/(\sqrt{2}r)}$ using polynomially many discrete Gaussian samples and the CLWE oracle as a sub-routine. The proof of Lemma 3.17 will

be given in Section 3.3.2 (which is the novel contribution) and Section 3.3.3 (which mainly follows [PRS17]).

In the rest of this subsection, we briefly explain how Theorem 3.15 follows from Lemma 3.17. This derivation is already implicit in past work [PRS17; Reg05], and is included here mainly for completeness. Readers familiar with the reduction may skip directly to Section 3.3.2.

The basic idea is to start with samples from a very wide discrete Gaussian (which can be efficiently sampled) and then iteratively sample from narrower discrete Gaussians, until eventually we end up with short discrete Gaussian samples, as required (see Figure 3.1). Each iteration consists of two steps: the first classical step is given by Lemma 3.17, allowing us to solve BDD on the dual lattice; the second step is quantum and is given in Lemma 3.18 below, which shows that solving BDD leads to sampling from narrower discrete Gaussian.

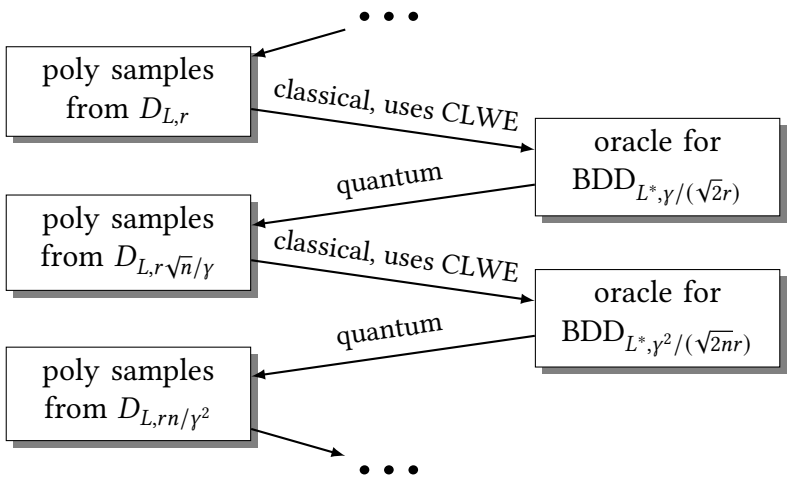


Figure 3.1: Two iterations of the reduction.

Lemma 3.18 ([Reg05, Lemma 3.14])**.** *There exists an efficient quantum algorithm that, given any n-dimensional lattice L, a number $d < \lambda_1(L^*)/2$, and an oracle that solves $\mathrm{BDD}_{L^*,d}$, outputs a sample from $D_{L,\sqrt{n}/(\sqrt{2}d)}$.*

Similar to [PRS17], there is a subtle requirement in Lemma 3.17 that we need discrete Gaussian samples from several different parameters $r' \geq r$. However, this is a non-issue since an oracle

for $\mathrm{BDD}_{L^*,\gamma/(\sqrt{2}r)}$ also solves $\mathrm{BDD}_{L^*,\gamma/(\sqrt{2}r')}$ for any $r' \geqslant r$, so Lemma 3.18 in fact allows us to efficiently sample from $D_{L,r'\sqrt{n}/\gamma}$ for any $r' \geqslant r$.

3.3.2 CLWE samples from BDD

In this subsection we prove Lemma 3.19, showing how to generate CLWE samples from the given BDD instance using discrete Gaussian samples. In the next subsection we will show how to solve the BDD instance by applying the decision CLWE oracle to these samples, thereby completing the proof of Lemma 3.17.

Lemma 3.19. *There is an efficient algorithm that takes as input an n-dimensional lattice L, a vector $\boldsymbol{u}+\boldsymbol{a}$ where $\boldsymbol{a} \in L^*$, reals $r, s_1, s_2 > 0$ such that $rs_1/\sqrt{\|\boldsymbol{u}\|^2(rs_1/s_2)^2 + t^2} \geq \eta_\varepsilon(L)$ for some $\varepsilon < \frac{1}{2}$ and $t = \sqrt{r^2+s_1^2}$, and samples from $D_{L,r}$, and outputs samples that are within statistical distance 8ε of the CLWE distribution $P_{\boldsymbol{u}',\beta,\gamma}$ for $\boldsymbol{u}' = \boldsymbol{u}/\|\boldsymbol{u}\|$, $\beta = \|\boldsymbol{u}\|\sqrt{(rs_1/t)^2 + (s_2/\|\boldsymbol{u}\|)^2}$ and $\gamma = \|\boldsymbol{u}\|r^2/t$.*

Proof. We start by describing the algorithm. For each $\boldsymbol{x}$ from the given samples from $D_{L,r}$, do the following. First, take the inner product $\langle \boldsymbol{x}, \boldsymbol{u}+\boldsymbol{a}\rangle$, which gives us

$$\langle \boldsymbol{x}, \boldsymbol{u}+\boldsymbol{a}\rangle = \langle \boldsymbol{x}, \boldsymbol{u}\rangle \bmod 1 \ .$$

Appending this inner product modulo 1 to the sample $\boldsymbol{x}$, we get $(\boldsymbol{x}, \langle \boldsymbol{x}, \boldsymbol{u}\rangle \bmod 1)$. Next, we "smooth out" the lattice structure of $\boldsymbol{x}$ by adding Gaussian noise $\boldsymbol{g} \sim D_{\mathbb{R}^n,s_1}$ to $\boldsymbol{x}$ and $e \sim D_{\mathbb{R},s_2}$ to $\langle \boldsymbol{x}, \boldsymbol{u}\rangle$ (modulo 1). Then, we have

$$(\boldsymbol{x}+\boldsymbol{g}, (\langle \boldsymbol{x}, \boldsymbol{u}\rangle + e) \bmod 1) \ . \tag{3.1}$$

Finally, we normalize the first component by t so that its marginal distribution has unit width,

giving us

$$((\boldsymbol{x}+\boldsymbol{g})/t, (\langle \boldsymbol{x}, \boldsymbol{u}\rangle + e) \bmod 1) , \tag{3.2}$$

which the algorithm outputs.

Our goal is to show that the distribution of (3.2) is within statistical distance 8ε of the CLWE distribution $P_{\boldsymbol{u}',\beta,\gamma}$, given by

$$(\boldsymbol{y}', (\gamma\langle \boldsymbol{y}', \boldsymbol{u}'\rangle + e') \bmod 1) ,$$

where $\boldsymbol{y}' \sim D_{\mathbb{R}^n}$ and $e' \sim D_{\mathbb{R},\beta}$. Because applying a function cannot increase statistical distance (specifically, dividing the first component by t and taking mod 1 of the second), it suffices to show that the distribution of

$$(\boldsymbol{x}+\boldsymbol{g}, \langle \boldsymbol{x}, \boldsymbol{u}\rangle + e) , \tag{3.3}$$

is within statistical distance 8ε of that of

$$(\boldsymbol{y}, (r/t)^2\langle \boldsymbol{y}, \boldsymbol{u}\rangle + e') , \tag{3.4}$$

where $\boldsymbol{y} \sim D_{\mathbb{R}^n,t}$ and $e' \sim D_{\mathbb{R},\beta}$. First, observe that by Lemma 2.8, the statistical distance between the marginals on the first component (i.e., between $\boldsymbol{x}+\boldsymbol{g}$ and $\boldsymbol{y}$) is at most 4ε. It is therefore sufficient to bound the statistical distance between the second components conditioned on any fixed value $\overline{\boldsymbol{y}}$ of the first component. Conditioned on the first component being $\overline{\boldsymbol{y}}$, the second component in (3.3) has the same distribution as

$$\langle \boldsymbol{x}+\boldsymbol{h}, \boldsymbol{u}\rangle \tag{3.5}$$

where $\boldsymbol{h} \sim D_{\mathbb{R}^n, s_2/\|\boldsymbol{u}\|}$, and the second component in (3.4) has the same distribution as

$$\langle (r/t)^2 \overline{\boldsymbol{y}} + \boldsymbol{h}', \boldsymbol{u} \rangle \tag{3.6}$$

where $\boldsymbol{h}' \sim D_{\mathbb{R}^n, \beta/\|\boldsymbol{u}\|}$.

By Claim 3.20 below, conditioned on $\boldsymbol{x}+\boldsymbol{g} = \overline{\boldsymbol{y}}$, the distribution of $\boldsymbol{x}$ is $(r/t)^2\overline{\boldsymbol{y}} + D_{L-(r/t)^2\overline{\boldsymbol{y}}, rs_1/t}$. Therefore, by Lemma 2.8, the conditional distribution of $\boldsymbol{x}+\boldsymbol{h}$ given $\boldsymbol{x}+\boldsymbol{g} = \overline{\boldsymbol{y}}$ is within statistical distance 4ε of that of $(r/t)^2\overline{\boldsymbol{y}}+\boldsymbol{h}'$. Since statistical distance cannot increase by applying a function (inner product with $\boldsymbol{u}$ in this case), (3.5) is within statistical distance 4ε of (3.6). Hence, the distribution of (3.3) is within statistical distance 8ε of that of (3.4). □

Claim 3.20. *Let $\boldsymbol{y} = \boldsymbol{x} + \boldsymbol{g}$, where $\boldsymbol{x} \sim D_{L,r}$ and $\boldsymbol{g} \sim D_{\mathbb{R}^n, s}$. Then, the conditional distribution of $\boldsymbol{x}$ given $\boldsymbol{y} = \overline{\boldsymbol{y}}$ is $(r/t)^2\overline{\boldsymbol{y}} + D_{L-(r/t)^2\overline{\boldsymbol{y}}, rs/t}$ where $t = \sqrt{r^2+s^2}$.*

Proof. Observe that $\boldsymbol{x}$ conditioned on $\boldsymbol{y} = \overline{\boldsymbol{y}}$ is a discrete random variable supported on L. The probability of $\boldsymbol{x}$ given $\boldsymbol{y} = \overline{\boldsymbol{y}}$ is proportional to

$$\rho_r(\boldsymbol{x}) \cdot \rho_s(\overline{\boldsymbol{y}} - \boldsymbol{x}) = \rho_t(\overline{\boldsymbol{y}}) \cdot \rho_{rs/t}(\boldsymbol{x} - (r/t)^2\overline{\boldsymbol{y}}) \propto \rho_{rs/t}(\boldsymbol{x} - (r/t)^2\overline{\boldsymbol{y}}) \,,$$

where the equality follows from Claim 2.5. Hence, the conditional distribution of $\boldsymbol{x} - (r/t)^2\boldsymbol{y}$ given $\boldsymbol{y} = \overline{\boldsymbol{y}}$ is $D_{L-(r/t)^2\overline{\boldsymbol{y}}, rs/t}$. □

3.3.3 Solving BDD with the CLWE oracle

In this subsection, we complete the proof of Lemma 3.17. We first give some necessary background on the Oracle Hidden Center Problem (OHCP) [PRS17]. The problem asks one to search for a "hidden center" $\boldsymbol{u}^*$ using a decision oracle whose acceptance probability depends only on the distance to $\boldsymbol{u}^*$. The problem's precise statement is as follows.

Definition 3.21 (OHCP)**.** For parameters $\varepsilon, \delta \in [0,1)$ and $\zeta \geq 1$, the $(\varepsilon, \delta, \zeta)$-*OHCP* is an approximate search problem that tries to find the "hidden" center $\boldsymbol{u}^*$. Given a scale parameter $d > 0$ and access to a randomized oracle $O : \mathbb{R}^n \times \mathbb{R}^{\geq 0} \to \{0,1\}$ such that its acceptance probability $p(\boldsymbol{u}, t)$ only depends on $\exp(t)\|\boldsymbol{u}-\boldsymbol{u}^*\|$ for some (unknown) "hidden center" $\boldsymbol{u}^* \in \mathbb{R}^n$ with $\delta d \leq \|\boldsymbol{u}^*\| \leq d$ and for any $\boldsymbol{u} \in \mathbb{R}^n$ with $\|\boldsymbol{u} - \boldsymbol{u}^*\| \leq \zeta d$, the goal is to output $\boldsymbol{u}$ s.t. $\|\boldsymbol{u} - \boldsymbol{u}^*\| \leq \varepsilon d$.

Notice that OHCP corresponds to our problem since we want to solve BDD, which is equivalent to finding the "hidden" offset vector $\boldsymbol{u}^*$, using a decision oracle for $\mathrm{CLWE}_{\beta,\gamma}$. The acceptance probability of the $\mathrm{CLWE}_{\beta,\gamma}$ oracle will depend on the distance between our guess $\boldsymbol{u}$ and the true offset $\boldsymbol{u}^*$. For OHCP, we have the following result from [PRS17].

Lemma 3.22 ([PRS17], Proposition 4.4)**.** *There is a poly(κ, n)-time algorithm that takes as input a confidence parameter $\kappa \geq 20\log(n+1)$ (and the scale parameter $d > 0$) and solves $(\exp(-\kappa), \exp(-\kappa), 1+1/\kappa)$-OHCP in dimension n except with probability $\exp(-\kappa)$, provided that the oracle O corresponding to the OHCP instance satisfies the following conditions. For some $p(\infty) \in [0,1]$ and $t^* \geqslant 0$,*

1. *$p(\mathbf{0}, t^*) - p(\infty) \geq 1/\kappa$;*
2. *$|p(\mathbf{0}, t) - p(\infty)| \leq 2\exp(-t/\kappa)$ for any $t \geq 0$; and*
3. *$p(\boldsymbol{u}, t)$ is κ-Lipschitz in t for any $\boldsymbol{u} \in \mathbb{R}^n$ such that $\|\boldsymbol{u}\| \leq (1 + 1/\kappa)d$.*

Furthermore, each of the algorithm's oracle calls takes the form $O(\cdot, i\Delta)$ for some $\Delta < 1$ that depends only on κ and n and $0 \leq i \leq \mathrm{poly}(\kappa, n)$.

The main idea in the proof of Lemma 3.22 is performing a guided random walk with advice from the decision oracle O. The decision oracle O rejects a random step with high probability if it increases the distance $\|\boldsymbol{u} - \boldsymbol{u}^*\|$. Moreover, there is non-negligible probability of decreasing the distance by a factor $\exp(1/n)$ unless $\log\|\boldsymbol{u} - \boldsymbol{u}^*\| \leq -\kappa$. Hence, with sufficiently many steps, the random walk will reach $\widehat{\boldsymbol{u}}$, a guess of the hidden center, which is within $\exp(-\kappa)$ distance to $\boldsymbol{u}^*$ with high probability.

Our goal is to show that we can construct an oracle O satisfying the above conditions using an oracle for $\mathrm{CLWE}_{\beta,\gamma}$. Then, it follows from Lemma 3.22 that BDD with discrete Gaussian samples can be solved using an oracle for CLWE. We first state some lemmas useful for our proof. Lemma 3.23 is Babai's closest plane algorithm and Lemma 3.24 is an upper bound on the statistical distance between two one-dimensional Gaussian distributions.

Lemma 3.23 ([LLL82a; Bab86]). *For any n-dimensional lattice L, there is an efficient algorithm that solves $\mathrm{BDD}_{L,d}$ for $d = 2^{-n/2} \cdot \lambda_1(L)$.*

Lemma 3.24 ([DMR18, Theorem 1.3]). *For all $\mu_1, \mu_2 \in \mathbb{R}$, and $\sigma_1, \sigma_2 > 0$, we have*

$$\Delta\big(\mathcal{N}(\mu_1, \sigma_1), \mathcal{N}(\mu_2, \sigma_2)\big) \leq \frac{3|\sigma_1^2 - \sigma_2^2|}{2\max(\sigma_1^2, \sigma_2^2)} + \frac{|\mu_1 - \mu_2|}{2\max(\sigma_1, \sigma_2)},$$

where $\mathcal{N}(\mu, \sigma)$ denotes the Gaussian distribution with mean μ and standard deviation σ.

Now, we prove Lemma 3.17, restated below.

Lemma 3.17. *Let $\beta = \beta(n) \in (0,1)$ and $\gamma = \gamma(n) \geq 2\sqrt{n}$ such that $q = \gamma/\beta$ is polynomially bounded. There exists a probabilistic polynomial-time (classical) algorithm with access to an oracle that solves $\mathrm{CLWE}_{\beta,\gamma}$, that takes as input a lattice $L \subset \mathbb{R}^n$, parameters β, γ, and $r \geq 2q \cdot \eta_\varepsilon(L)$, and $\mathrm{poly}(n)$ many samples from the discrete Gaussian distribution D_{L,r_i} for $\mathrm{poly}(n)$ parameters $r_i \geq r$ and solves $\mathrm{BDD}_{L^*,d}$ for $d = \gamma/(\sqrt{2}r)$.*

Proof. Let $d' = (1 - 1/(2n)) \cdot d$. By [LM09, Corollary 2], it suffices to solve $\mathrm{BDD}_{L^*,d'}$. Let $\kappa = \mathrm{poly}(n)$ with $\kappa \geq 8qn\ell$ be such that the advantage of our $\mathrm{CLWE}_{\beta,\gamma}$ oracle is at least $1/\kappa$, where $\ell \geq 1$ is the number of samples required by the oracle.

Given as input a lattice $L \subset \mathbb{R}^n$, a parameter $r \geq 2q \cdot \eta_\varepsilon(L)$, samples from D_{L,r_i} for $1 \leq i \leq \mathrm{poly}(n)$, and a BDD instance $\boldsymbol{u}^* + \boldsymbol{a}$ where $\boldsymbol{a} \in L^*$ and $\|\boldsymbol{u}^*\| \leq d'$, we want to recover $\boldsymbol{u}^*$. Without loss of generality, we can assume that $\|\boldsymbol{u}^*\| \geq \exp(-n/2) \cdot \lambda_1(L^*) \geq (2q/r) \cdot \exp(-n/2)$

(Lemma 3.9), since we can otherwise find $\boldsymbol{u}^*$ efficiently using Babai's closest plane algorithm (Lemma 3.23).

We will use the CLWE oracle to simulate an oracle $O : \mathbb{R}^n \times \mathbb{R}^{\geqslant 0} \to \{0,1\}$ such that the probability that $O(\boldsymbol{u}, t)$ outputs 1 ("accepts") only depends on $\exp(t)\|\boldsymbol{u} - \boldsymbol{u}^*\|$. Our oracle O corresponds to the oracle in Definition 3.21 with $\boldsymbol{u}^*$ as the "hidden center". We will use Lemma 3.22 to find $\boldsymbol{u}^*$.

On input $(\boldsymbol{u}, t)$, our oracle O receives ℓ independent samples from $D_{L,\exp(t)r}$. Then, we generate CLWE samples using the procedure from Lemma 3.19. The procedure takes as input these ℓ samples, the vector $\boldsymbol{a} + \boldsymbol{u}^* - \boldsymbol{u}$ where $\boldsymbol{a} \in L^*$, and parameters $\exp(t)r, \exp(t)s_1, s_2$. Our choice of s_1 and s_2 will be specified below. Note that the CLWE oracle requires the "hidden direction" $(\boldsymbol{u} - \boldsymbol{u}^*)/\|\boldsymbol{u} - \boldsymbol{u}^*\|$ to be uniformly distributed on the unit sphere. To this end, we apply the worst-to-average case reduction from Claim 3.14. Let $S_{\boldsymbol{u},t}$ be the resulting CLWE distribution. Our oracle O then calls the $\mathrm{CLWE}_{\beta,\gamma}$ oracle on $S_{\boldsymbol{u},t}^{\ell}$ and outputs 1 if and only if it accepts.

Using the oracle O, we can run the procedure from Lemma 3.22 with confidence parameter κ and scale parameter d'. The output of this procedure will be some approximation $\widehat{\boldsymbol{u}}$ to the oracle's "hidden center" with the guarantee that $\|\widehat{\boldsymbol{u}} - \boldsymbol{u}^*\| \leq \exp(-\kappa)d'$. Finally, running Babai's algorithm on the vector $\boldsymbol{a} + \boldsymbol{u}^* - \widehat{\boldsymbol{u}}$ will give us $\boldsymbol{u}^*$ exactly since

$$\|\widehat{\boldsymbol{u}} - \boldsymbol{u}^*\| \leq \exp(-\kappa)d \leq \beta \exp(-\kappa)/\eta_\varepsilon(L) \leq 2^{-n}\lambda_1(L^*) \,,$$

where the last inequality is from Lemma 3.7.

The running time of the above procedure is clearly polynomial in n. It remains to check that our oracle O (1) is a valid instance of $(\exp(-\kappa), \exp(-\kappa), 1 + 1/\kappa)$-OHCP with hidden center $\boldsymbol{u}^*$ and (2) satisfies all the conditions of Lemma 3.22. First, note that $S_{\boldsymbol{u},t}$ will be negligibly close in

statistical distance to the CLWE distribution with parameters

$$\beta' = \sqrt{(\exp(t)\|\boldsymbol{u} - \boldsymbol{u}^*\|)^2 s_1'^2 + s_2^2}\,,$$
$$\gamma' = \exp(t)\|\boldsymbol{u} - \boldsymbol{u}^*\| r'\,,$$

where $r' = r^2/\sqrt{r^2 + s_1^2}$ and $s_1' = rs_1/\sqrt{r^2 + s_1^2}$ as long as r, s_1, s_2 satisfy the conditions of Lemma 3.19. Then, we set $s_1 = r/(\sqrt{2}q)$ and choose s_2 such that

$$s_2^2 = \beta^2 - (s_1'/r')^2\gamma^2 = \beta^2 - (s_1/r)^2\gamma^2 = \beta^2/2\,.$$

Lemma 3.19 requires $rs_1/\sqrt{r^2\|\boldsymbol{u} - \boldsymbol{u}^*\|^2(s_1/s_2)^2 + r^2 + s_1^2} \geq \eta_\varepsilon(L)$. We know that $r \geq 2q \cdot \eta_\varepsilon(L)$ and $s_1 \geq \sqrt{2} \cdot \eta_\varepsilon(L)$, so it remains to determine a sufficient condition for the aforementioned inequality. Observe that for any $\boldsymbol{u}$ such that $\|\boldsymbol{u} - \boldsymbol{u}^*\| \leq d$, the condition $s_2 \geq 2d \cdot \eta_\varepsilon(L)$ is sufficient. Since $r \geq 2(\gamma/\beta) \cdot \eta_\varepsilon(L)$, this translates to $s_2 \geq \beta/(\sqrt{2})$. Hence, the transformation from Lemma 3.19 will output samples negligibly close to CLWE samples for our choice of s_1 and s_2 as long as $\|\boldsymbol{u} - \boldsymbol{u}^*\| \leq d$ (beyond the BDD distance bound d').

Since $S_{\boldsymbol{u},t}$ is negligibly close to the CLWE distribution, the acceptance probability $p(\boldsymbol{u}, t)$ of O only depends on $\exp(t)\|\boldsymbol{u} - \boldsymbol{u}^*\|$. Moreover, by assumption $\|\boldsymbol{u}^*\| \geq \exp(-n/2) \cdot (2q/r) \geq \exp(-\kappa)d'$. Hence, O, κ, d' correspond to a valid instance of $(\exp(-\kappa), \exp(-\kappa), 1 + 1/\kappa)$-OHCP with "hidden center" $\boldsymbol{u}^*$.

Next, we show that $p(\boldsymbol{u}, t)$ of O satisfies all three conditions of Lemma 3.22 with $p(\infty)$ taken to be the acceptance probability of the CLWE oracle on samples from $D_{\mathbb{R}^n} \times U$. Item 1 of Lemma 3.22 follows from our assumption that our $\mathrm{CLWE}_{\beta,\gamma}$ oracle has advantage $1/\kappa$, and by our choice of r, s_1, and s_2, when $t^* = \log(\gamma/(\|\boldsymbol{u}^*\|r')) > \log(\sqrt{2})$, the generated CLWE samples satisfy $\gamma'(t^*) = \gamma$ and $\beta'(t^*) = \beta$. Hence, $p(\boldsymbol{0}, t^*) - p(\infty) \geq 1/\kappa$.

We now show that Item 2 holds, which states that $|p(\boldsymbol{0}, t) - p(\infty)| \leq 2\exp(-t/\kappa)$ for any

$t > 0$. We will show that $S_{\mathbf{0},t}$ converges exponentially fast to $D_{\mathbb{R}^n} \times U$ in statistical distance. Let $f(\boldsymbol{y}, z)$ be the probability density of $S_{\mathbf{0},t}$. Then,

$$\begin{aligned}\Delta(S_{\mathbf{0},t}, D_{\mathbb{R}^n} \times U) &= \frac{1}{2} \int |f(z|\boldsymbol{y}) - U(z)| \rho(\boldsymbol{y}) d\boldsymbol{y} dz \\ &= \frac{1}{2} \int \left(\int |f(z|\boldsymbol{y}) - U(z)| dz \right) \rho(\boldsymbol{y}) d\boldsymbol{y} \,.\end{aligned}$$

Hence, it suffices to show that the conditional density of z given $\boldsymbol{y}$ for $S_{\mathbf{0},t}$ converges exponentially fast to the uniform distribution on $\mathbb{T}$. Notice that the conditional distribution of z given $\boldsymbol{y}$ is the Gaussian distribution with width parameter $\beta' \geq \exp(t)\|\boldsymbol{u}^*\| r/(2q) \geq \exp(t - n/2)$, where we have used our assumption that $\|\boldsymbol{u}^*\| \geq (2q/r) \cdot \exp(-n/2)$. By Lemma 3.7 applied to $\mathbb{Z}$, we know that β' is larger than $\eta_\varepsilon(\mathbb{Z})$ for $\varepsilon = \exp(-\exp(2t - n))$. Hence, one sample from this conditional distribution is within statistical distance ε of the uniform distribution by Lemma 3.6. By the triangle inequality applied to ℓ samples,

$$\Delta\left(S_{\mathbf{0},t}^{\ell}, (D_{\mathbb{R}^n} \times U)^{\ell}\right) \leq \min(1, \ell \exp(-\exp(2t - n))) \leq 2\exp(-t/\kappa) \,,$$

where in the last inequality, we use the the fact that we can choose κ to be such that $2\exp(-t/\kappa) \geq 1$ unless $t \geq \kappa/2$. And when $t \geq \kappa/2 \geq 4qn\ell$, we have $\ell \exp(-\exp(2t - n)) \ll \exp(-t/\kappa)$.

It remains to verify Item 3, which states that $p(\boldsymbol{u}, t)$ is κ-Lipschitz in t for any $\|\boldsymbol{u}\| \leq (1 + 1/\kappa)d' \leq d$. We show this by bounding the statistical distance between $S_{\boldsymbol{u},t_1}$ and $S_{\boldsymbol{u},t_2}$ for $t_1 \geq t_2$. With a slight abuse in notation, let $f_{t_i}(\boldsymbol{y}, z)$ be the probability density of $S_{\boldsymbol{u},t_i}$ and let (β_i, γ_i) be the corresponding CLWE distribution parameters. For simplicity, also denote the hidden direction by $\boldsymbol{u}' = (\boldsymbol{u} - \boldsymbol{u}^*)/\|\boldsymbol{u} - \boldsymbol{u}^*\|$. Then,

$$
\begin{aligned}
\Delta(f_{t_1}, f_{t_2}) &= \frac{1}{2}\int \left(\int |f_{t_1}(z|\boldsymbol{y}) - f_{t_2}(z|\boldsymbol{y})|dz\right)\rho(\boldsymbol{y})d\boldsymbol{y} \\
&= \int \Delta\left(\mathcal{N}(\gamma_1\langle \boldsymbol{y}, \boldsymbol{u}'\rangle, \beta_1/\sqrt{2\pi}), \mathcal{N}(\gamma_2\langle \boldsymbol{y}, \boldsymbol{u}'\rangle, \beta_2/\sqrt{2\pi})\right)\rho(\boldsymbol{y})d\boldsymbol{y} \\
&\le \frac{1}{2}\int \left(3(1-(\beta_2/\beta_1)^2) + \sqrt{2\pi}(\gamma_1-\gamma_2)/\beta_1 \cdot |\langle \boldsymbol{y}, \boldsymbol{u}'\rangle|\right)\cdot \rho(\boldsymbol{y})d\boldsymbol{y} \qquad (3.7) \\
&\le \mathop{\mathbb{E}}_{\boldsymbol{y}\sim\rho}[M(\boldsymbol{y})]\cdot\left(1-\exp(-2(t_1-t_2))\right) \text{ where } M(\boldsymbol{y}) = \frac{1}{2}\left(3 + 2\sqrt{\pi}q\cdot|\langle \boldsymbol{y}, \boldsymbol{u}'\rangle|\right) \\
&\le \mathop{\mathbb{E}}_{\boldsymbol{y}\sim\rho}[M(\boldsymbol{y})]\cdot 2(t_1-t_2) \qquad (3.8) \\
&\le (\kappa/\ell)\cdot(t_1-t_2)\,, \qquad (3.9)
\end{aligned}
$$

where (3.7) follows from Lemma 3.24, (3.8) uses the fact that $1-\exp(-2(t_1-t_2)) \le 2(t_1-t_2)$, and (3.9) uses the fact that $\mathbb{E}_{\boldsymbol{y}\sim\rho}[M(\boldsymbol{y})] \le 4q \le \kappa/(2\ell)$. Using the triangle inequality over ℓ samples, the statistical distance between $S^{\ell}_{\boldsymbol{u},t_1}$ and $S^{\ell}_{\boldsymbol{u},t_2}$ is at most

$$
\min(1, \ell\cdot(\kappa/\ell)(t_1-t_2)) \le \kappa(t_1-t_2)\,.
$$

Therefore, $p(\boldsymbol{u}, t)$ is κ-Lipschitz in t. □

3.4 Hardness of Learning Cosine Neurons

A *cosine neuron* is a function with the form $f(x) = \cos(2\pi\gamma\langle \boldsymbol{u}, \boldsymbol{x}\rangle)$, where we refer to $\boldsymbol{u} \in \mathbb{S}^{n-1}$ as its hidden direction, and γ as its frequency. Such functions have already been investigated by previous works [SVWX17; Sha18; SSS17] in the context of lower bounds for learning neural networks. More precisely, [SVWX17] has shown that SQ algorithms whose queries are restricted to be Lipschitz cannot weakly learn cosine neurons over isotropic Gaussian inputs, and [SSS17; Sha18] have shown that gradient-based algorithms fail as well.

For these hard constructions, the frequency γ is taken to scale polynomially with the dimension d. Note that as the univariate function $\cos(2\pi\gamma t)$ is $O(\gamma)$-Lipschitz, the function f is well-approximated by one-hidden-layer ReLU network of $\mathrm{poly}(\gamma)$-width on any compact set. Hence, understanding the hardness of learning such functions is an unavoidable step towards understanding the hardness of learning one-hidden-layer ReLU networks.

We show that weakly learning the cosine neuron class over the standard Gaussian with small label noise is hard. The proof is based on a simple reduction from CLWE. Our result therefore extends the hardness of learning such functions from a restricted family of algorithms, such as gradient-based algorithms or SQ, to *all* polynomial-time algorithms by leveraging cryptographic assumptions. We remark that our result can be extended to *any* 1-periodic "link" function ϕ beyond $\cos(2\pi\cdot)$. At the heart of our reduction is the following elementary equality.

$$\cos(2\pi(t \mod 1)) = \cos(2\pi t) \ . \tag{3.10}$$

Theorem 3.25. *Let $n \in \mathbb{N}$, $\gamma = \gamma(n) \geqslant 2\sqrt{n}$ and $\tau = \tau(n) \in (0,1)$ be such that $\gamma/\tau = \mathrm{poly}(n)$, $\beta = \beta(n)$ be such that $\beta/\tau = \omega(\sqrt{\log n})$. Then, a polynomial-time algorithm that weakly learns the cosine neuron class $\mathcal{F}_\gamma$ under β-bounded adversarial noise implies a polynomial-time quantum algorithm for $O(d/\tau)$-GapSVP.*

Proof sketch. We reduce $\mathrm{CLWE}_{\tau,\gamma}$ to the problem of weakly learning the function class $\mathcal{F}_\gamma^\phi$. Let ϕ be any 1-periodic, κ-Lipschitz link function (e.g., $\phi(z) = \cos(2\pi z)$ is 2π-Lipschitz). Denote by $D_{\boldsymbol{u}}^\phi$ the distribution of $(\boldsymbol{y}, \phi(\gamma\langle \boldsymbol{y}, \boldsymbol{u}\rangle))$ induced by $\boldsymbol{y} \sim \mathcal{N}(0, I_n)$.

Now, $\mathrm{CLWE}_{\tau,\gamma}$ is the problem of distinguishing the distribution $P_{\boldsymbol{u},\tau,\gamma}$ which outputs samples of the form $(\boldsymbol{y}, z)$ where $z = \gamma\langle \boldsymbol{u}, \boldsymbol{y}\rangle + \xi, \boldsymbol{y} \sim \mathcal{N}(0, I_n), \xi \sim \mathcal{N}(0, \tau^2)$, for some hidden direction $\boldsymbol{u} \in \mathbb{S}^{n-1}$, from the null distribution Q_n which outputs $(\boldsymbol{y}, z)$ where $\boldsymbol{y} \sim \mathcal{N}(0, I_n)$ but $z \sim \mathcal{U}[-1/2, 1/2]$ independent from $\boldsymbol{y}$. Notice that (similar to Eq (3.10)) the 1-periodicity and the Lipschitzness of

ϕ implies that for any $\gamma \geqslant 0$, $\boldsymbol{u} \in \mathbb{S}^{n-1}$, $\boldsymbol{y} \in \mathbb{R}^n$, and $\xi \in \mathbb{R}$,

$$\phi(z_i) = \phi(\gamma\langle \boldsymbol{u}, \boldsymbol{y}\rangle + \xi \mod 1) = \phi(\gamma\langle \boldsymbol{u}, \boldsymbol{y}\rangle + \xi) = \phi(\gamma\langle \boldsymbol{u}, \boldsymbol{y}\rangle) + \tilde{\xi}' \,, \tag{3.11}$$

for some $\tilde{\xi} \in [-\kappa|\xi|, \kappa|\xi|]$. Using Eq. (3.11) one can then directly use m CLWE samples with Gaussian random noise, say, $(\boldsymbol{y}_i, z_i)$, and transform them into m samples from $D^{\phi}_{\boldsymbol{u}}$ corrupted with bounded *adversarial* noise by $\kappa\tau \leqslant \beta$, by simply considering the samples $(\boldsymbol{y}_i, \phi(z_i))$, $i = 1, 2, \ldots, m$.

Let us suppose now we have a learning algorithm that weakly learns the function class $\mathcal{F}^{\phi}_{\gamma}$ with β-bounded adversarial noise. Then we can draw m samples from $P_{\boldsymbol{u},\tau,\gamma}$, transform them as above into samples from $D^{\phi}_{\boldsymbol{u}}$, run the (robust) learning algorithm on $D^{\phi}_{\boldsymbol{u}}$, and finally obtain a hypothesis $g = g(\boldsymbol{y}_i, \phi(z_i))$ that weakly learns the function class $\mathcal{F}^{\phi}_{\gamma}$. On the other hand, samples from Q_n have labels z_i independent of $\boldsymbol{y}_i$ and therefore are completely uninformative for the learning problem of interest. In particular, one can never hope to achieve weak learning of the function class $\mathcal{F}^{\phi}_{\gamma}$, using the hypothesis function $g = g(\boldsymbol{y}_i, \phi(z_i))$ on the samples $(\boldsymbol{y}_i, z_i)$ now generated from Q_n. This difference is quantified by the loss of the hypothesis $\mathcal{L}_D(h)$ which in case $D = P_{\boldsymbol{u},\tau,\gamma}$, is smaller by an $\Omega(1)$ additive factor from the trivial loss, while in the case in case $D = Q_n$ it is lower bounded by the trivial loss. This property is what allows us to indeed distinguish between $P_{\boldsymbol{u},\tau,\gamma}$ and Q_n, and therefore solve $\mathrm{CLWE}_{\beta,\gamma}$ and complete the reduction. □

4 | Gaussian Pancakes

In the summer of 1930 ... I had been stricken by an acute attack of a disease which at irregular intervals afflicts all mathematicians and, for that matter, all scientists: I became obsessed by a problem.

Mark Kac[1]

Gaussian pancakes can be motivated from many angles, underscoring its importance as a fundamental and canonical problem in statistical inference. From a statistician's perspective, Gaussian pancakes is closely related to the problem of learning high-dimensional mixtures of Gaussians, which is a classical problem in statistics and machine learning [Pea94]. In fact, the original motivation for considering distributions with "parallel pancakes" comes from an attempt to establish lower bounds for estimating the density of Gaussian mixtures [DKS17]. In Section 4.2.1, we show that estimating the density of Gaussian pancakes reduces to the distinguishing task. Hence, our result on the hardness of Gaussian pancakes implies that no algorithm can learn k-mixtures of n-dimensional Gaussians in $\mathrm{poly}(n, k)$ time. We note that the computational complexity of density estimation for Gaussian mixtures was an open problem prior to our work [BRST21] (see e.g., [Moi18, Open Question 7.1] and [Dia16, Open Problem 1.7.2]), though the influential work of [DKS17] had previously established hardness against SQ algorithms.

[1] [Kac87, Prologue].

Formulated as a search (or estimation) problem of *finding* the secret direction $\boldsymbol{u} \in \mathbb{S}^{n-1}$, Gaussian pancakes can also be seen as a special instance of non-Gaussian component analysis (NGCA), which is the canonical problem of finding "interesting" (i.e., non-Gaussian) directions of high-dimensional data [BKSS+06]. A standard method in statistics precisely designed to uncover interesting directions of the data is *projection pursuit* [FT74; Hub85; JS87], which finds such directions by optimizing certain functionals (called the *projection index*) of low-dimensional projections of the data. Thus, the Gaussian pancakes problem provides valuable insights into the complexity of NGCA and intractability of projection pursuit-based estimators for high-dimensional data.

We remark that [DKS17; DH22] have previously shown that NGCA is hard for restricted classes of algorithms such as SQ and low-degree polynomials. In fact, their hard instance can be seen as a close cousin of the Gaussian pancakes we consider, in which discrete Gaussians are replaced with discrete distributions (with *unequally* spaced support) carefully constructed to match the low-order moments of the standard Gaussian exactly. We also note the work of [NR22], who proposed a projection pursuit-based estimator that achieves optimal statistical rate for estimating Wasserstein distances between the more general *spiked transport models* (i.e., distributions that only differ in a low-dimensional subspace) and presented evidence, in the form of an SQ lower bound, that there are fundamental obstructions to computing this estimator efficiently.

From a cryptographer's perspective, Gaussian pancakes are "backdoored" Gaussians which can be used to probabilistically encrypt bits [GM84]. For example, one can encrypt 0's with i.i.d. samples from the standard Gaussian and 1's with i.i.d. samples from the Gaussian pancakes distribution with secret direction $\boldsymbol{u} \in \mathbb{S}^{n-1}$, and decrypt by projecting the samples along $\boldsymbol{u}$ and checking which ones are close to $(1/\gamma)\mathbb{Z}$. Hardness of distinguishing Gaussian pancakes guarantees that no polynomial-time adversary can decrypt the samples without knowing the secret $\boldsymbol{u}$. Follow-up works by cryptographers have used Gaussian pancakes to construct novel public-key cryptosystems [BNHR22] and plant undetectable backdoors in machine learning models [GKVZ22], thereby highlighting the value of Gaussian pancakes in cryptography.

Numerous connections to canonical problems and growing range of applications provide ample motivation for studying the complexity of Gaussian pancakes. Our first result shows that distinguishing Gaussian pancakes is as hard as worst-case lattice problems via a straightforward reduction from CLWE (see Section 4.2). More precisely, we show

Theorem 4.1 (Informal). *Let $n \in \mathbb{N}$, $\beta = \beta(n) \in (0,1)$ and $\gamma = \gamma(n) \geq 2\sqrt{n}$ such that the ratio γ/β is polynomially bounded. If there exists an efficient algorithm that solves* $\mathrm{hCLWE}_{\beta,\gamma}$[2]*, then there exists an efficient quantum algorithm that approximates worst-case lattice problems to within polynomial factors.*

Next, we investigate the hardness of Gaussian pancakes beyond the parameters β and γ covered by Theorem 4.1 through the lens of SQ algorithms and low-degree polynomials. Both results indicate that unless one uses many samples ($m \approx \exp(\tilde{O}(\gamma^2))$), in which the spectrum of the empirical covariance is sufficient for distinguishing (see Section 4.5), algorithms require $\exp(\tilde{\Omega}(n))$ time to distinguish Gaussian pancakes. Understanding the precise theorem statements requires background knowledge of each computation model, so we defer them to their respective sections (Sections 4.3 and 4.4). It is worth mentioning that the "low-degree hardness" of Gaussian pancakes applies even to the *noiseless* setting, in which the pancakes have 0 thickness. Therefore, the Gaussian pancakes problem serves as yet another instance where low degree polynomials fail to pick up on highly informative, but brittle, structures in the data.

Below, we provide additional background and motivation for our results.

Hardness of learning Gaussian mixtures. Efficient algorithms are known for learning Gaussian mixtures if the components are guaranteed to be sufficiently well separated (e.g., [Das99; VW02; AK05; DS07; BV08; RV17; HL18; KSS18; DKS18]). Without such strong separation requirements, it is known that efficiently *recovering* the individual components of a mixture (technically known as "parameter estimation") is in general impossible [MV10]; intuitively, this exponential

[2]Recall that we denote the distinguishing problem by hCLWE.

(in the number of components) information theoretic lower bound holds because the Gaussian components "blur into each other", despite being mildly separated pairwise.

This leads to the question of whether there exists an efficient algorithm that can learn mixtures of Gaussians without strong separation requirements, not in the above parameter estimation sense (which is impossible), but rather in the much weaker density estimation sense, where the goal is merely to output an approximation of the given distribution's density function. See [Dia16, Open Problem 1.7.2] for the precise statement and [DKS17] where a super-polynomial lower bound for density estimation is shown in the SQ model [Kea98; FGRV+17]. Our work provides a negative answer to this open question, showing that learning Gaussian mixtures is computationally hard even if the goal is only to output an estimate of the density (see Proposition 4.11). It is worth noting that our hard instance has almost non-overlapping components, i.e., the pairwise statistical distance between distinct Gaussian components is essentially 1, a property shared by the SQ hard instance of [DKS17]. This property is useful for controlling the decryption error in the public key encryption scheme proposed by [BNHR22].

Statistical-to-computational gaps and cryptography. Cryptography and statistical-to-computational gaps share a rich history, dating back to at least the first formalization of *computational indistinguishability* [GM84]. Extensive studies (see e.g., [GGM86; Kha93; KV94; DGR97; Ser99; KS09; SST12; DV21]) on how computational constraints affect statistical performance have been carried out on the Boolean hypercube $\{0,1\}^n$, the native domain of computational complexity. A standard template for such hardness results is to first start with a cryptographic primitive exhibiting *average-case* hardness (e.g., pseudorandom functions), use it to construct an artificial hard learning problem, and then show that it reduces to some natural learning problem.

In this regard, the Gaussian pancakes problem is exceptional in several ways. Firstly, it admits a direct reduction from *worst-case* lattice problems, which demonstrates its hardness in certain parameter regimes. This hardness is sufficient for constructing public key cryptosystems [BNHR22],

as previously mentioned. Moreover, the problem of distinguishing "backdoored" Gaussians from the "true" Gaussian is a canonical and particularly relevant task, given that Gaussians are commonly used for neural network initialization. In addition, Gaussian pancakes distributions have nice analytical properties which can be leveraged to study the complexity of the problem through alternate notions of hardness, such as SQ and low-degree hardness, beyond the regimes covered by the reduction.

Low-degree predictions for cryptography. Our analysis of the Gaussian pancakes problem using the *low-degree method* [KWB22] hints at the possibility of extending the method to "predict" whether a distinguishing problem is hard enough to serve as a cryptographic primitive (i.e., rules out non-negligible advantage). However, our results also suggest that refinements to the *low-degree conjecture* (originally from [Hop18, Hypothesis 2.1.5]; see also [KWB22, Section 4.2.4]) are necessary to get reliable predictions of hardness. Since the Gaussian pancakes problem is known to be hard based on reductions from worst-case lattice problems and LWE [BRST21; GVV22], it can serve as an important theoretical testbed for assessing the soundness of any future refinements to the conjecture.

4.1 Preliminaries

Chapter-specific notation. We denote $\rho_s(\boldsymbol{x}) = \exp(-\pi\|\boldsymbol{x}/s\|^2)$ in Section 4.2, following the tradition of lattice-based cryptography. For the other sections, we default to $\rho_s(\boldsymbol{x}) = \exp(-\|\boldsymbol{x}/s\|^2/2)$, following the tradition of statistics and probability theory. Differences in the scaling of the Gaussian density function only affect the normalization constants of related distributions.

Adapting the hypothesis testing notation (see Chapter 2) to our setup, we denote by Q_n the standard Gaussian distribution on $\mathbb{R}^n$. We use the caligraphic notation $\mathcal{Q}_n$ to denote the joint distribution of m i.i.d. standard Gaussian vectors in $\mathbb{R}^n$, i.e., $\mathcal{Q}_n = Q_n^{\otimes m}$. Assuming the parameters β

and γ are clear from context, we denote by $P_{\boldsymbol{u}}$ the (single-sample) Gaussian pancakes distribution with parameters β, γ and secret direction $\boldsymbol{u} \in \mathbb{S}^{n-1}$. We use the caligraphic notation $\mathcal{P}_n$ to denote the distribution induced by the two-step data generation process for Gaussian pancakes, in which $\boldsymbol{u}$ is first drawn uniformly from $\mathbb{S}^{n-1}$ and m i.i.d. samples are drawn from $P_{\boldsymbol{u}}$. Finally, we denote by A_n the discrete Gaussian of unit width supported on $(1/\gamma)\mathbb{Z}$, where $\gamma = \gamma(n)$.

We now define the Gaussian pancakes distribution (also called hCLWE) in a way that allows for a clear correspondence to the CLWE parameters (Definition 4.2). This definition will be used in Section 4.2 and Section 4.3. For the other sections, we use an alternate definition (Definition 4.4) of Gaussian pancakes which simplifies how the distribution parameters change with respect to the Ornstein-Uhlenbeck noise operator (see Definition 4.5). This will simplify the analysis for univariate projections of $P_{\boldsymbol{u}}$.

Definition 4.2 (Gaussian pancakes distribution)**.** For any $\boldsymbol{u} \in \mathbb{S}^{n-1}$ and parameters $\beta, \gamma > 0$, define the *Gaussian pancakes distribution* $P_{\boldsymbol{u},\beta,\gamma}$ over $\mathbb{R}^n$ to have density at $\boldsymbol{x} \in \mathbb{R}^n$ proportional to

$$\rho(\boldsymbol{x}) \cdot \sum_{k \in \mathbb{Z}} \rho_\beta(k - \gamma\langle \boldsymbol{x}, \boldsymbol{u}\rangle) \ . \tag{4.1}$$

We sometimes refer to the Gaussian pancakes distribution as the *homogeneous CLWE distribution* to emphasize its connection to CLWE. The distribution can be equivalently be expressed as a mixture of Gaussians. To see this, notice that Eq. (4.1) is equal to

$$\sum_{k \in \mathbb{Z}} \rho_{\sqrt{\beta^2+\gamma^2}}(k) \cdot \rho(\mathrm{Proj}_{\boldsymbol{u}^\perp}(\boldsymbol{x})) \cdot \rho_{\beta/\sqrt{\beta^2+\gamma^2}}\left(\langle \boldsymbol{x}, \boldsymbol{u}\rangle - \frac{\gamma}{\beta^2+\gamma^2}k\right) , \tag{4.2}$$

where $\mathrm{Proj}_{\boldsymbol{u}^\perp}$ denotes the projection to the subspace orthogonal to $\boldsymbol{u}$. Hence, $P_{\boldsymbol{u},\beta,\gamma}$ can be viewed as a mixture of Gaussian components of width $\beta/\sqrt{\beta^2+\gamma^2}$ (which is roughly β/γ for $\beta \ll \gamma$) in the secret direction, and unit width in the orthogonal space. The components are equally spaced, with a separation of $\gamma/(\beta^2+\gamma^2)$ between them (which is roughly $1/\gamma$ for $\beta \ll \gamma$).

If $\rho(\boldsymbol{x}) = \exp(-\pi\|\boldsymbol{x}\|^2)$, then integral of (4.2) (and also (4.1)) over all $\boldsymbol{x} \in \mathbb{R}^n$ is

$$Z = \frac{\beta}{\sqrt{\beta^2+\gamma^2}} \cdot \rho\left(\frac{1}{\sqrt{\beta^2+\gamma^2}}\mathbb{Z}\right). \tag{4.3}$$

This is easy to see since the integral over $\boldsymbol{x}$ of the product of the last two ρ terms in (4.2) is $\beta/\sqrt{\beta^2+\gamma^2}$ independently of k.

Definition 4.3. For parameters $\beta, \gamma > 0$, the average-case decision problem $\mathrm{hCLWE}_{\beta,\gamma}$ is to distinguish the following two distributions over $\mathbb{R}^n$: (1) the Gaussian pancakes distribution $H_{\boldsymbol{w},\beta,\gamma}$ for some uniformly random unit vector $\boldsymbol{w} \in \mathbb{R}^n$ (which is fixed for all samples), or (2) $D_{\mathbb{R}^n}$.

For the alternate definition of Gaussian pancakes below, we denote $\rho(\boldsymbol{x}) = \exp(-\|\boldsymbol{x}\|^2/2)$.

Definition 4.4 (Alternate definition of Gaussian pancakes)**.** For any $\boldsymbol{u} \in \mathbb{S}^{n-1}$ and $\gamma > 0$, we define the *Gaussian pancakes distribution* $P_{\boldsymbol{u}}$ over $\mathbb{R}^n$ to be a distribution with (mixed) density proportional to

$$P_{\boldsymbol{u}}(\boldsymbol{x}) \propto A(\langle \boldsymbol{x}, \boldsymbol{u}\rangle) \cdot \rho(\boldsymbol{x} - \langle \boldsymbol{x}, \boldsymbol{u}\rangle \boldsymbol{u}), \tag{4.4}$$

where A denotes the discrete Gaussian on $(1/\gamma)\mathbb{Z}$ of unit width.

Definition 4.5 (β-smoothed discrete Gaussian)**.** Let $\beta \in [0,1]$. Let A be the discrete Gaussian of unit width on $(1/\gamma)\mathbb{Z}$. We define the *β-smoothed discrete Gaussian* A^β to be the distribution induced by applying the *Ornstein-Uhlenbeck noise operator* to A. In other words, $\tilde{x} \sim A^\beta$ is given by

$$\tilde{x} \overset{\mathrm{d}}{=} \sqrt{1-\beta^2}x + \beta y, \text{ where } x \sim A,\ y \sim \mathcal{N}(0,1).$$

Remark 4.6 (univariate projections)**.** *Let $\tau \in (0,1)$ and let $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{S}^{n-1}$ be such that $\langle \boldsymbol{u}, \boldsymbol{v}\rangle = \tau$. If $\boldsymbol{x}$ is a random sample from $P_{\boldsymbol{u}}$, then the marginal distribution of $\langle \boldsymbol{x}, \boldsymbol{v}\rangle$ is $A^{\sqrt{1-\tau^2}}$.*

Claim 4.7. *Let $\beta \in (0,1), \gamma > 0$. Then, the density of A^β at $x \in \mathbb{R}$ is given by*

$$A^\beta(x) = \frac{1}{\sqrt{2\pi}}\rho(x) \cdot \frac{1}{\beta\rho((1/\gamma)\mathbb{Z})} \sum_{z \in (1/\gamma\sqrt{1-\beta^2})\mathbb{Z}} \rho_{\beta/\sqrt{1-\beta^2}}(x-z) \ . \tag{4.5}$$

Equivalently, we can express it in the Gaussian mixture form.

$$A^\beta(x) = \frac{1}{\beta\sqrt{2\pi}} \cdot \frac{1}{\rho((1/\gamma)\mathbb{Z})} \sum_{\tilde{z} \in (\sqrt{1-\beta^2}/\gamma)\mathbb{Z}} \rho_{\sqrt{1-\beta^2}}(\tilde{z}) \cdot \rho_\beta(x-\tilde{z}) \ . \tag{4.6}$$

Proof. Let $x \sim A$ and $y \sim \mathcal{N}(0,1)$. We write out the formal density of $\sqrt{1-\beta^2}x + \beta y$ as a convolution, using Dirac deltas to express A.

$$\begin{aligned}
\sqrt{1-\beta^2}A * \beta\mathcal{N}(0,1) &= \frac{1}{\rho((1/\gamma)\mathbb{Z})} \sum_{z \in (\sqrt{1-\beta^2}/\gamma)\mathbb{Z}} \exp\left(-\frac{z^2}{2(1-\beta^2)}\right)\delta(x-z) * \frac{1}{\sqrt{2\pi}\beta}\exp\left(-\frac{y^2}{2\beta^2}\right) \\
&= \frac{1}{\sqrt{2\pi}\beta\rho((1/\gamma)\mathbb{Z})} \sum_{z \in (\sqrt{1-\beta^2}/\gamma)\mathbb{Z}} \exp\left(-\frac{z^2}{2(1-\beta^2)}\right)\exp\left(-\frac{(x-z)^2}{2\beta^2}\right) \\
&= \frac{1}{\sqrt{2\pi}}\exp\left(-\frac{x^2}{2}\right)\frac{1}{\beta\rho((1/\gamma)\mathbb{Z})} \sum_{z \in (1/\gamma\sqrt{1-\beta^2})\mathbb{Z}} \exp\left(-\frac{1}{2}\cdot\frac{(x-z)^2}{\beta^2/(1-\beta^2)}\right),
\end{aligned}$$

where the last line comes from straightforward algebraic manipulation.

$$\begin{aligned}
(1/(1-\beta^2) + 1/\beta^2)z^2 - (2x/\beta^2)z + x^2/\beta^2 &= \frac{1}{\beta^2(1-\beta^2)}\Big(z - (1-\beta^2)x\Big)^2 + (x^2/\beta^2)(1-(1-\beta^2)) \\
&= \frac{1-\beta^2}{\beta^2}\Big(x - z/(1-\beta^2)\Big)^2 + x^2 \ .
\end{aligned}$$

□

4.2 Reduction-based Hardness of Gaussian Pancakes

We now present a reduction from CLWE to Gaussian pancakes which establishes its hardness based on worst-case lattice problems. We refer to the Gaussian pancakes problem as hCLWE to simplify notation and emphasize its connection to CLWE. The main step of the reduction is to transform CLWE samples to hCLWE samples using rejection sampling (Lemma 4.8).

Let $W_{\boldsymbol{u},\beta,\gamma}$ denote the CLWE distribution for $\mathrm{CLWE}_{\beta,\gamma}$ and consider the samples $(\boldsymbol{y}, z) \sim W_{\boldsymbol{u},\beta,\gamma}$. If we condition $\boldsymbol{y}$ on $z = 0 \pmod 1$ then we get exactly samples $\boldsymbol{y} \sim P_{\boldsymbol{u},\beta,\gamma}$ for $\mathrm{hCLWE}_{\beta,\gamma}$. However, this approach is impractical as $z = 0 \pmod 1$ happens with probability 0. Instead we condition $\boldsymbol{y}$ on $z \approx 0 \pmod 1$ somehow. One can imagine that the resulting samples $\boldsymbol{y}$ will still have a "wavy" probability density in the direction of $\boldsymbol{u}$ with spacing $1/\gamma$, which accords with the picture of homogeneous CLWE. To avoid throwing away too many samples, we will do rejection sampling with some small "window" $\delta = 1/\mathrm{poly}(n)$. Formally, we have the following lemma.

Lemma 4.8. *There is a* $\mathrm{poly}(n, 1/\delta)$*-time probabilistic algorithm that takes as input a parameter* $\delta \in (0, 1)$ *and samples from* $W_{\boldsymbol{u},\beta,\gamma}$*, and outputs samples from* $P_{\boldsymbol{u},\sqrt{\beta^2+\delta^2},\gamma}$.

Proof. Without loss of generality assume that $\boldsymbol{u} = \boldsymbol{e}_1$. By definition, the probability density of sample $(\boldsymbol{y}, z) \sim W_{\boldsymbol{u},\beta,\gamma}$ is

$$p(\boldsymbol{y}, z) = \frac{1}{\beta} \cdot \rho(\boldsymbol{y}) \cdot \sum_{k \in \mathbb{Z}} \rho_\beta(z + k - \gamma y_1) \ .$$

Let $g : \mathbb{T} \to [0, 1]$ be the function $g(z) = g_0(z)/M$, where $g_0(z) = \sum_{k\in\mathbb{Z}} \rho_\delta(z + k)$ and $M = \sup_{z\in\mathbb{T}} g_0(z)$. We perform rejection sampling on the samples $(\boldsymbol{y}, z)$ with acceptance probability $\Pr[\mathrm{accept}|\boldsymbol{y}, z] = g(z)$. We remark that $g(z)$ is efficiently computable (see [BLPR+13, Section

5.2]). The probability density of outputting $\boldsymbol{y}$ and accept is

$$\begin{aligned}\int_{\mathbb{T}} p(\boldsymbol{y},z)g(z)dz &= \frac{\rho(\boldsymbol{y})}{\beta M}\cdot\int_{\mathbb{T}}\sum_{k_1,k_2\in\mathbb{Z}}\rho_\beta(z+k_1-\gamma y_1)\rho_\delta(z+k_2)dz\\ &= \frac{\rho(\boldsymbol{y})}{\beta M}\cdot\int_{\mathbb{T}}\sum_{k,k_2\in\mathbb{Z}}\rho_{\sqrt{\beta^2+\delta^2}}(k-\gamma y_1)\rho_{\beta\delta/\sqrt{\beta^2+\delta^2}}\Big(z+k_2+\frac{\delta^2(k-\gamma y_1)}{\beta^2+\delta^2}\Big)dz\\ &= \frac{\delta}{M\sqrt{\beta^2+\delta^2}}\cdot\rho(\boldsymbol{y})\cdot\sum_{k\in\mathbb{Z}}\rho_{\sqrt{\beta^2+\delta^2}}(k-\gamma y_1)\ ,\end{aligned}$$

where the second equality follows from Claim 2.5. This shows that the conditional distribution of $\boldsymbol{y}$ upon acceptance is indeed $P_{\boldsymbol{e}_1,\sqrt{\beta^2+\delta^2},\gamma}$. Moreover, a byproduct of this calculation is that the expected acceptance probability is $\Pr[\text{accept}] = Z\delta/(M\sqrt{\beta^2+\delta^2})$, where, according to Eq. (4.3),

$$\begin{aligned}Z &= \sqrt{\frac{\beta^2+\delta^2}{\beta^2+\delta^2+\gamma^2}}\cdot\rho_{\sqrt{\beta^2+\delta^2+\gamma^2}}(\mathbb{Z})\\ &= \sqrt{\beta^2+\delta^2}\cdot\rho_{1/\sqrt{\beta^2+\delta^2+\gamma^2}}(\mathbb{Z})\\ &\geqslant \sqrt{\beta^2+\delta^2}\ ,\end{aligned}$$

and the second equality uses Lemma 2.6. Observe that

$$\begin{aligned}g_0(z) &= \sum_{k\in\mathbb{Z}}\rho_\delta(z+k)\\ &\leq 2\cdot\sum_{k=0}^{\infty}\rho_\delta(k)\\ &< 2\cdot\sum_{k=0}^{\infty}\exp(-\pi k) < 4\end{aligned}$$

since $\delta < 1$, implying that $M \leqslant 4$. Therefore, $\Pr[\text{accept}] \geqslant \delta/4$, and so the rejection sampling procedure has $\mathrm{poly}(n, 1/\delta)$ expected running time. □

The above lemma reduces CLWE to homogeneous CLWE with slightly worse parameters.

Hence, homogeneous CLWE is as hard as CLWE. Specifically, combining Theorem 3.15 (with β taken to be $\beta/\sqrt{2}$) and Lemma 4.8 (with δ also taken to be $\beta/\sqrt{2}$), we obtain the following corollary.

Corollary 4.9 (Hardness of hCLWE)**.** *For any $\beta = \beta(n) \in (0,1)$ and $\gamma = \gamma(n) \geq 2\sqrt{n}$ such that γ/β is polynomially bounded, there is a polynomial-time quantum reduction from* $\mathrm{DGS}_{2\sqrt{2n}\eta_\varepsilon(L)/\beta}$ *to* $\mathrm{hCLWE}_{\beta,\gamma}$.

We again note that other standard worst-case lattice problems, such as GapSVP and SIVP, can be reduced to DGS (see, e.g., [Reg09, Section 3.3]).

4.2.1 Hardness of Density Estimation for Gaussian Mixtures

We rule out $\mathrm{poly}(n,k)$ time algorithms for estimating the density of k-mixtures of n-dimensional Gaussians by reducing hCLWE to the density estimation problem. This answers an open question regarding its computational complexity [Dia16; Moi18]. We first formally define the density estimation problem for Gaussian mixtures.

Definition 4.10 (Density estimation of Gaussian mixtures)**.** Let $\mathcal{G}_{n,k}$ be the family of k-mixtures of n-dimensional Gaussians. The problem of *density estimation* for $\mathcal{G}_{n,k}$ is the following. Given $\delta > 0$ and sample access to an unknown $P \in \mathcal{G}_{n,k}$, with probability $2/3$, output a hypothesis distribution D (in the form of an evaluation oracle) such that $\Delta(P,D) \leqslant \delta$.

For our purposes, we fix the precision parameter δ to a very small constant, say, $\delta = 10^{-3}$. Now we show a reduction from $\mathrm{hCLWE}_{\beta,\gamma}$ to the problem of density estimation for Gaussian mixtures. Corollary 4.9 shows that $\mathrm{hCLWE}_{\beta,\gamma}$ is hard for $\gamma \geqslant 2\sqrt{n}$ (assuming worst-case lattice problems are hard). Hence, by taking $\gamma = 2\sqrt{n}$ and $g(n) = O(\log n)$ in Proposition 4.11, we rule out the possibility of a $\mathrm{poly}(n,k)$-time density estimation algorithm for $\mathcal{G}_{n,k}$ under the same hardness assumption.

Proposition 4.11. *Let $\beta = \beta(n) \in (0, 1/32)$, $\gamma = \gamma(n) \geqslant 1$, and $g(n) \geqslant 4\pi$. For $k = 2\gamma\sqrt{g(n)/\pi}$, if there is an $\exp(g(n))$-time algorithm that solves density estimation for $\mathcal{G}_{n,2k+1}$, then there is a $O(\exp(g(n)))$-time algorithm that solves $\mathrm{hCLWE}_{\beta,\gamma}$.*

Proof. We apply the (hypothetical) density estimation algorithm $\mathcal{A}$ to the unknown given distribution D. As we will show below, with constant probability, it outputs a density estimate f that satisfies $\Delta(f, D) < 2\delta = 2 \cdot 10^{-3}$ (and this is even though $P_{\boldsymbol{u},\beta,\gamma}$ has infinitely many components). We then test whether $D = Q_n$ or not using the following procedure. We repeat the following procedure $r = 1/(6\sqrt{\delta})$ times. We draw $\boldsymbol{x} \sim Q_n$ and check whether the following holds

$$\frac{f(\boldsymbol{x})}{D(\boldsymbol{x})} \in [1 - \sqrt{\delta}, 1 + \sqrt{\delta}] \ . \tag{4.7}$$

where D denotes the density of Q_n. We output $D = Q_n$ if Eq. (4.7) holds for all m independent trials and $D = P_{\boldsymbol{u},\beta,\gamma}$ otherwise. Since $\Delta(P_{\boldsymbol{u},\beta,\gamma}, Q_n) > 1/2$ (Claim 4.12), it is not hard to see that this test solves $\mathrm{hCLWE}_{\beta,\gamma}$ with probability at least $2/3$ (see [RS09, Observation 24] for a closely related statement). Moreover, the total running time is $O(\exp(g(n))$ since this test uses a constant number of samples.

If $D = Q_n$, it is obvious that $\mathcal{A}$ outputs a close density estimate with constant probability since $Q_n \in \mathcal{G}_{n,2k+1}$. It remains to consider the case $P = P_{\boldsymbol{u},\beta,\gamma}$. To this end, we observe that $P_{\boldsymbol{u},\beta,\gamma}$ is close to a $(2k+1)$-mixture of Gaussians. Indeed, by Claim 4.13 below,

$$\Delta(P_{\boldsymbol{u},\beta,\gamma}, P_k) \leqslant 2\exp(-\pi \cdot k^2/(\beta^2 + \gamma^2)) < 2\exp(-\pi \cdot k^2/(2\gamma^2)) \ ,$$

where P_k is the distribution given by truncating $P_{\boldsymbol{u},\beta,\gamma}$ to the $(2k+1)$ central mixture components. Hence, the statistical distance between the joint distribution of $\exp(g(n))$ samples from $P_{\boldsymbol{u},\beta,\gamma}$ and

that of $\exp(g(n))$ samples from P_k is bounded by

$$2\exp(-\pi \cdot k^2/(2\gamma^2)) \cdot \exp(g(n)) = 2\exp(-g(n)) \leqslant 2\exp(-4\pi) \ .$$

Since the two distributions are statistically close, a standard argument shows that $\mathcal{A}$ will output f satisfying $\Delta(f, P_{\boldsymbol{u},\beta,\gamma}) \leqslant \Delta(f, P_k) + \Delta(P_k, P_{\boldsymbol{u},\beta,\gamma}) < 2\delta$ with constant probability. □

Claim 4.12. *Let $\beta = \beta(n) \in (0, 1/32)$ and $\gamma = \gamma(n) \geqslant 1$. Then,*

$$\Delta(P_{\boldsymbol{u},\beta,\gamma}, Q_n) > 1/2 \ .$$

Proof. Let $\gamma' = \sqrt{\beta^2 + \gamma^2} > \gamma$. Let $\boldsymbol{y} \in \mathbb{R}^n$ be a random vector distributed according to $P_{\boldsymbol{u},\beta,\gamma}$. Using the Gaussian mixture form of (4.2), we observe that $\langle \boldsymbol{y}, \boldsymbol{u} \rangle \bmod \gamma/\gamma'^2$ is distributed according to $D_{\beta/\gamma'} \bmod \gamma/\gamma'^2$. Since statistical distance cannot increase by applying a function (inner product with $\boldsymbol{u}$ and then applying the modulo operation in this case), it suffices to lower bound the statistical distance between $D_{\beta/\gamma'} \bmod \gamma/\gamma'^2$ and $D \bmod \gamma/\gamma'^2$, where D denotes the 1-dimensional standard Gaussian.

By Chernoff, for all $\zeta > 0$, at least $1 - \zeta$ mass of $D_{\beta/\gamma'}$ is contained in $[-a \cdot (\beta/\gamma'), a \cdot (\beta/\gamma')]$, where $a = \sqrt{\log(1/\zeta)}$. Hence, $D_{\beta/\gamma'} \bmod \gamma/\gamma'^2$ is at least $1 - 2a\beta\gamma'/\gamma - \zeta$ far in statistical distance from the uniform distribution over $\mathbb{R}/(\gamma/\gamma'^2)\mathbb{Z}$, which we denote by U. Moreover, by Lemma 3.6 and Lemma 3.7, $D \bmod \gamma/\gamma'^2$ is within statistical distance $\varepsilon/2 = \exp(-\gamma'^4/\gamma^2)/2$ from U. Therefore,

$$\begin{aligned}
\Delta(D_{\beta/\gamma'} \bmod \gamma/\gamma'^2, D \bmod \gamma/\gamma'^2) &\geqslant \Delta(D_{\beta/\gamma'} \bmod \gamma/\gamma'^2, U) - \Delta(U, D \bmod \gamma/\gamma'^2) \\
&\geqslant 1 - 2a\beta\gamma'/\gamma - \zeta - \varepsilon/2 \\
&> 1 - 2\sqrt{2}a\beta - \zeta - \exp(-\gamma^2)/2 \qquad (4.8) \\
&> 1/2 \ ,
\end{aligned}$$

where we set $\zeta = \exp(-2)$ and use the fact that $\beta \leqslant 1/32$ and $\gamma \geqslant 1$ in (4.8). □

Claim 4.13. *Let $\beta = \beta(n) \in (0,1), \gamma = \gamma(n) \geqslant 1$, and $k \in \mathbb{Z}^+$. Then,*

$$\Delta(P_{\boldsymbol{u},\beta,\gamma}, P_k) \leqslant 2\exp(-\pi \cdot k^2/(\beta^2+\gamma^2)) ,$$

where P_k is the distribution given by truncating $P_{\boldsymbol{u},\beta,\gamma}$ to the central $(2k+1)$ mixture components.

Proof. We express $P_{\boldsymbol{u},\beta,\gamma}$ in its Gaussian mixture form given in Eq. (4.2) and define a random variable X taking on values in $\mathbb{Z}$ such that the probability of $X = i$ is equal to the probability that a sample comes from the i-th component in $P_{\boldsymbol{u},\beta,\gamma}$. Then, we observe that P_k is the distribution given by conditioning on $|X| \leqslant k$. Since X is a discrete Gaussian random variable with distribution $D_{\mathbb{Z},\sqrt{\beta^2+\gamma^2}}$, we observe that $\Pr[|X| > k] \leqslant \varepsilon := 2\exp(-\pi \cdot k^2/(\beta^2+\gamma^2))$ by [MP12, Lemma 2.8]. Since conditioning on an event of probability $1-\varepsilon$ cannot change the statistical distance by more than ε, we have

$$\Delta(P_{\boldsymbol{u},\beta,\gamma}, P_k) \leqslant \varepsilon .$$

□

4.2.2 Hardness of Gaussian baguettes

We now generalize the hardness result to the setting where the distribution has $\ell \geqslant 1$ hidden directions. This follows from a standard hybrid argument.

Definition 4.14 (Gaussian baguettes distribution)**.** For $0 \leqslant \ell \leqslant n$, matrix $W \in \mathbb{R}^{n\times\ell}$ with orthonormal columns $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_\ell$, and $\beta, \gamma > 0$, define the *ℓ-hCLWE distribution* $P_{\boldsymbol{u},\beta,\gamma}$ over $\mathbb{R}^n$ to

have density at $\boldsymbol{x}$ proportional to

$$\rho(\boldsymbol{x}) \cdot \prod_{i=1}^{\ell} \sum_{k \in \mathbb{Z}} \rho_\beta(k - \gamma \langle \boldsymbol{x}, \boldsymbol{u}_i \rangle) \ .$$

Note that the $\ell = 0$ corresponds to Q_n regardless of β and γ.

Definition 4.15. For parameters $\beta, \gamma > 0$ and $1 \leqslant \ell \leqslant n$, the average-case decision problem $\mathrm{hCLWE}_{\beta,\gamma}^{(\ell)}$ is to distinguish the following two distributions over $\mathbb{R}^n$: (1) the ℓ-hCLWE distribution $P_{\boldsymbol{u},\beta,\gamma}$ for some matrix $W \in \mathbb{R}^{n \times \ell}$ (which is fixed for all samples) with orthonormal columns chosen uniformly from the set of all such matrices, or (2) Q_n.

Lemma 4.16. *For any $\beta, \gamma > 0$ and positive integer $\ell = \ell(n)$ such that $\ell \leqslant n$ and $n - \ell = \Omega(n^c)$ for some constant $c > 0$, if there exists an efficient algorithm that solves $\mathrm{hCLWE}_{\beta,\gamma}^{(\ell)}$ with non-negligible advantage, then there exists an efficient algorithm that solves $\mathrm{hCLWE}_{\beta,\gamma}$ with non-negligible advantage.*

Proof. Suppose $\mathcal{A}$ is an efficient algorithm that solves $\mathrm{hCLWE}_{\beta,\gamma}^{(\ell)}$ with non-negligible advantage in dimension n. Then consider the following algorithm $\mathcal{B}$ that uses $\mathcal{A}$ as an oracle and solves $\mathrm{hCLWE}_{\beta,\gamma}$ in dimension $n' = n - \ell + 1$.

1. Input: n'-dimensional samples, drawn from either $\mathrm{hCLWE}_{\beta,\gamma}$ or $Q_{n'}$;
2. Choose $0 \leqslant i \leqslant \ell - 1$ uniformly at random;
3. Append $\ell - 1 = n - n'$ coordinates to the given samples, where the first i appended coordinates are drawn from $P_{I_i,\beta,\gamma}$ (with I_i denoting the rank-i identity matrix) and the rest of the coordinates are drawn from $Q_{\ell-i-1}$;
4. Rotate the augmented samples using a uniformly random rotation from the orthogonal group $O(n)$;

5. Call $\mathcal{A}$ with the samples and output the result.

As $n = O(n'^{1/c})$, $\mathcal{B}$ is an efficient algorithm. Moreover, the samples passed to $\mathcal{A}$ are effectively drawn from either $\mathrm{hCLWE}^{(i+1)}_{\beta,\gamma}$ or $\mathrm{hCLWE}^{(i)}_{\beta,\gamma}$. Therefore the advantage of $\mathcal{B}$ is at least $1/m$ fraction of the advantage of $\mathcal{A}$, which would be non-negligible (in terms of n, and thus also in terms of n') as well. □

Combining Corollary 4.9 and Lemma 4.16, we obtain the following corollary.

Corollary 4.17. *For any $\beta = \beta(n) \in (0,1)$ and $\gamma = \gamma(n) \geq 2\sqrt{n}$ such that γ/β is polynomially bounded, and positive integer $\ell = \ell(n)$ such that $\ell \leqslant n$ and $n - \ell = \Omega(n^c)$ for some constant $c > 0$, there is a polynomial-time quantum reduction from $\mathrm{DGS}_{2\sqrt{2n}\eta_\varepsilon(L)/\beta}$ to $\mathrm{hCLWE}^{(\ell)}_{\beta,\gamma}$.*

4.3 SQ Hardness of Gaussian Pancakes

Statistical Query (SQ) algorithms [Kea98] are a restricted class of algorithms that are only allowed to query expectations of functions of the input distribution without directly accessing individual samples. To be more precise, SQ algorithms access the input distribution indirectly via the $\mathrm{STAT}(\tau)$ oracle, which given a query function f and data distribution D, returns a value contained in the interval $\mathbb{E}_{x\sim D}[f(x)] + [-\tau, \tau]$ for some precision parameter τ.

We prove SQ hardness of distinguishing Gaussian pancakes distributions from the standard Gaussian. In particular, we show that SQ algorithms that solve Gaussian pancakes require super-polynomial number of queries even with super-polynomial precision. This is formalized in Theorem 4.18.

Theorem 4.18. *Let $\beta = \beta(n) \in (0,1)$ and $\gamma = \gamma(n) \geq \sqrt{2}$. Then, any (randomized) SQ algorithm with precision $\tau \geq 4 \cdot \exp(-\pi \cdot \gamma^2/4)$ that successfully solves $\mathrm{hCLWE}_{\beta,\gamma}$ with probability $\eta > 1/2$ requires at least $(2\eta - 1) \cdot \exp(cn) \cdot \tau^2\beta^2/(4\gamma^2)$ statistical queries of precision τ for some constant $c > 0$.*

Note that when $\gamma = \Omega(\sqrt{n})$ and $\gamma/\beta = \mathrm{poly}(n)$, even exponential precision $\tau = \exp(-O(n))$ results in a query lower bound that grows as $\exp(\tilde{\Omega}(n))$. This establishes an unconditional hardness result for SQ algorithms in the parameter regime $\gamma = \Omega(\sqrt{n})$, which is consistent with our computational hardness result based on worst-case lattice problems. The uniform spacing in Gaussian pancakes gives us tight control over their pairwise correlation (see definition in (4.9)), which leads to a simple proof of the SQ lower bound.

We first provide some necessary background on the SQ framework. We denote by $\mathcal{B}(\mathcal{U}, Q)$ the decision problem in which the input distribution D either equals Q or belongs to $\mathcal{U}$, and the goal of the algorithm is to identify whether $D = Q$ or $D \in \mathcal{U}$. For our purposes, D will be the standard Gaussian Q_n and $\mathcal{U}$ will be a finite set of Gaussian pancakes. Abusing notation, we denote by $D(x)$ the density of D. Following [FGRV+17], we define the *pairwise correlation* between two distributions P_1, P_2 relative to Q as

$$\chi_Q(P_1, P_2) := \mathbb{E}_{\boldsymbol{x}\sim Q}\left[\left(\frac{P_1(\boldsymbol{x})}{Q(\boldsymbol{x})} - 1\right)\cdot\left(\frac{P_2(\boldsymbol{x})}{Q(\boldsymbol{x})} - 1\right)\right] = \mathbb{E}_{\boldsymbol{x}\sim Q}\left[\frac{P_1(\boldsymbol{x})P_2(\boldsymbol{x})}{Q(\boldsymbol{x})^2}\right] - 1\ . \tag{4.9}$$

Lemma 4.19 below establishes a lower bound on the number of statistical queries required to solve $\mathcal{B}(\mathcal{U}, D)$ in terms of pairwise correlation between distributions in $\mathcal{U}$.

Lemma 4.19 ([FGRV+17, Lemma 3.10])**.** *Let Q be a distribution and $\mathcal{U}$ be a set of distributions both over a domain X such that for any $P_1, P_2 \in \mathcal{U}$*

$$|\chi_Q(P_1, P_2)| \le \begin{cases} \delta & \text{if } P_1 = P_2 \\ \varepsilon & \text{otherwise} \end{cases}\ .$$

Let $\tau \geqslant \sqrt{2\varepsilon}$. Then, any (randomized) SQ algorithm that solves $\mathcal{B}(\mathcal{U}, Q)$ with success probability $\eta > 1/2$ requires at least $(2\eta - 1)\cdot|\mathcal{U}|\cdot\tau^2/(2\delta)$ queries to $\mathrm{STAT}(\tau)$.

The following proposition establishes a tight upper bound on the pairwise correlation be-

tween Gaussian pancakes. To deduce Theorem 4.18 from Lemma 4.19 and Proposition 4.20, we take a set of unit vectors $\mathcal{U}$ such that any two distinct vectors $\boldsymbol{u}, \boldsymbol{v} \in \mathcal{U}$ satisfy $|\langle \boldsymbol{u}, \boldsymbol{v} \rangle| \leq 1/\sqrt{2}$, and identify it with the set of Gaussian pancakes $\{P_{\boldsymbol{u},\beta,\gamma}\}_{\boldsymbol{u}\in\mathcal{U}}$. A standard probabilistic argument shows that such a $\mathcal{U}$ can be as large as $\exp(\Omega(n))$, which proves Theorem 4.18.

Proposition 4.20. *Let $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^n$ be unit vectors and let $P_{\boldsymbol{u}}, P_{\boldsymbol{v}}$ be n-dimensional Gaussian pancakes with parameters $\gamma \geq 1, \beta \in (0,1)$, and hidden direction $\boldsymbol{v}$ and $\boldsymbol{u}$, respectively. Then, for any $\alpha \geqslant 0$ that satisfies $\gamma^2(1-\alpha^2) \geqslant 1$,*

$$|\chi_Q(P_{\boldsymbol{u}}, P_{\boldsymbol{v}})| \leq \begin{cases} 2(\gamma/\beta)^2 & \text{if } \boldsymbol{u} = \boldsymbol{v} \\ 8\exp(-\pi \cdot \gamma^2(1-\alpha^2)) & \text{if } |\langle \boldsymbol{u}, \boldsymbol{v} \rangle| \leq \alpha \end{cases}.$$

Proof. We will show that computing $\chi_Q(P_{\boldsymbol{u}}, P_{\boldsymbol{v}})$ reduces to evaluating the Gaussian mass of two lattices L_1 and L_2 defined below. Then, we will tightly bound the Gaussian mass using Lemma 2.6 and Lemma 3.8, which will result in upper bounds on $|\chi_Q(P_{\boldsymbol{u}}, P_{\boldsymbol{v}})|$. We define L_1 and L_2 by specifying their bases B_1 and B_2, respectively.

$$B_1 = \frac{1}{\sqrt{\beta^2+\gamma^2}} \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix},$$
$$B_2 = \frac{1}{\sqrt{\beta^2+\gamma^2}} \begin{pmatrix} 1 & 0 \\ -\frac{\alpha\gamma^2}{\zeta\sqrt{\beta^2+\gamma^2}} & \frac{\sqrt{\beta^2+\gamma^2}}{\zeta} \end{pmatrix},$$

where $\zeta = \sqrt{(\beta^2+\gamma^2) - \alpha^2\gamma^4/(\beta^2+\gamma^2)}$. Then the basis of the dual lattice L_1^* and L_2^* is B_1^{-T} and B_2^{-T}, respectively. Note that $\lambda_2(L_1)^2 = 1/(\beta^2+\gamma^2)$ and that the two columns of B_2 have the same

norm, and so

$$\lambda_2(L_2)^2 \le \frac{1}{\beta^2+\gamma^2} \cdot \max\left\{1 + \frac{\alpha^2\gamma^4}{\zeta^2(\beta^2+\gamma^2)}, \frac{\beta^2+\gamma^2}{\zeta^2}\right\}$$
$$= \frac{1}{\zeta^2} \tag{4.10}$$
$$\le \frac{1}{\gamma^2(1-\alpha^2)} \,. \tag{4.11}$$

Now define the density ratio $T(t) := A(t)/Q(t)$, where Q is the standard Gaussian and A is the marginal distribution of Gaussian pancakes with parameters β, γ along the hidden direction. We immediately obtain

$$T(t) = \frac{1}{Z}\sum_{k\in\mathbb{Z}} \rho_{\beta/\gamma}(t - k/\gamma) \,, \tag{4.12}$$

where $Z = \int_{\mathbb{R}} \rho(t) \cdot \sum_{k\in\mathbb{Z}} \rho_{\beta/\gamma}(t-k/\gamma)dt$. By Eq. (4.3), Z is given by

$$Z = \frac{\beta}{\sqrt{\beta^2+\gamma^2}} \cdot \rho\left(\frac{1}{\sqrt{\beta^2+\gamma^2}}\mathbb{Z}\right) .$$

Moreover, we can express Z^2 in terms of the Gaussian mass of (L_1) as

$$Z^2 = \frac{\beta^2}{\beta^2+\gamma^2} \cdot \rho(L_1) \,.$$

$\chi_Q(P_{\boldsymbol{u}}, P_{\boldsymbol{v}})$ can be expressed in terms of $T(t)$ as

$$\chi_Q(P_{\boldsymbol{u}}, P_{\boldsymbol{v}}) = \mathop{\mathbb{E}}_{\boldsymbol{x}\sim Q}\left[T(\langle \boldsymbol{x}, \boldsymbol{u}\rangle) \cdot T(\langle \boldsymbol{x}, \boldsymbol{v}\rangle)\right] - 1 \,. \tag{4.13}$$

Without loss of generality, assume $\boldsymbol{u} = \boldsymbol{e}_1$ and $\boldsymbol{v} = \alpha\boldsymbol{e}_1 + \xi\boldsymbol{e}_2$, where $\xi = \sqrt{1-\alpha^2}$. We first compute the pairwise correlation for $\boldsymbol{u} \neq \boldsymbol{v}$. For notational convenience, we denote by $\varepsilon = 8 \cdot \exp(-\pi \cdot \gamma^2(1-\alpha^2))$.

$$
\begin{aligned}
\chi_Q(P_{\boldsymbol{u}}, P_{\boldsymbol{v}}) + 1 &= \mathop{\mathbb{E}}_{\boldsymbol{x}\sim Q}\left[T(x_1)\cdot T(\alpha x_1 + \xi x_2)\right] \\
&= \frac{1}{Z^2}\sum_{k,\ell\in\mathbb{Z}}\int\int \rho_\beta(\gamma x_1 - k)\cdot\rho_\beta((\gamma\alpha x_1 + \gamma\xi x_2) - \ell)\cdot\rho(x_1)\cdot\rho(x_2)dx_1dx_2 \\
&= \frac{1}{Z^2}\cdot\frac{\beta}{\sqrt{(\gamma\xi)^2+\beta^2}}\sum_{k,\ell\in\mathbb{Z}}\int \rho_\beta(\gamma x_1 - k)\cdot\rho(x_1)\cdot\rho_{\sqrt{1+\beta^2/(\gamma\xi)^2}}(\ell/(\gamma\xi) - (\alpha/\xi)x_1)dx_1 \\
&= \frac{1}{Z^2}\cdot\frac{\beta}{\sqrt{(\gamma\xi)^2+\beta^2}}\cdot\frac{\beta\sqrt{(\gamma\xi)^2+\beta^2}}{\zeta\sqrt{\beta^2+\gamma^2}}\sum_{k,\ell\in\mathbb{Z}}\rho_{\sqrt{\beta^2+\gamma^2}}(k)\cdot\rho_\zeta\left(\ell - \gamma^2\alpha\cdot k/(\beta^2+\gamma^2)\right) \\
&= \frac{\sqrt{\beta^2+\gamma^2}}{\zeta}\cdot\frac{\sum_{k,\ell\in\mathbb{Z}}\rho_{\sqrt{\beta^2+\gamma^2}}(k)\cdot\rho_\zeta\left(\ell - \gamma^2\alpha\cdot k/(\beta^2+\gamma^2)\right)}{\rho(L_1)} \\
&= \frac{\sqrt{\beta^2+\gamma^2}}{\zeta}\cdot\frac{\rho(L_2)}{\rho(L_1)} \\
&= \frac{\sqrt{\beta^2+\gamma^2}}{\zeta}\cdot\frac{\det(L_2^*)}{\det(L_1^*)}\cdot\frac{\rho(L_2^*)}{\rho(L_1^*)} \\
&= \frac{\rho(L_2^*)}{\rho(L_1^*)} \qquad (4.14) \\
&\in \left[\frac{1}{1+\varepsilon}, 1+\varepsilon\right],
\end{aligned}
$$

In (4.14), we used the Poisson summation formula (Lemma 2.6). The last line follows from (4.11) and Lemma 3.8, which implies that for any 2-dimensional lattice L satisfying $\lambda_2(L) \le 1$,

$$
\rho(L^* \setminus \{\mathbf{0}\}) \le 8\exp(-\pi/\lambda_2(L)^2)\ . \qquad (4.15)
$$

Now consider the case $\boldsymbol{u} = \boldsymbol{v}$. Using (4.10), we get an upper bound $\lambda_2(L_2) \le 1/\beta$ when $\alpha = 1$.

It follows that $\lambda_2((\beta/\gamma)L_2) \leqslant 1/\gamma \leqslant 1$. Hence,

$$\begin{aligned}
\chi_Q(P_v, P_v) + 1 &= \frac{\sqrt{\beta^2+\gamma^2}}{\zeta} \cdot \frac{\rho(L_2)}{\rho(L_1)} \\
&\leq \frac{\sqrt{\beta^2+\gamma^2}}{\zeta} \cdot \frac{\rho((\beta/\gamma)L_2)}{\rho(L_1)} \\
&= \frac{\sqrt{\beta^2+\gamma^2}}{\zeta} \cdot \frac{\det((\gamma/\beta)L_2^*)}{\det(L_1^*)} \cdot \frac{\rho((\gamma/\beta)L_2^*)}{\rho(L_1^*)} \\
&= \frac{\gamma^2}{\beta^2} \cdot \frac{\rho((\gamma/\beta)L_2^*)}{\rho(L_1^*)} \qquad (4.16) \\
&\leq 2(\gamma/\beta)^2 \,. \qquad (4.17)
\end{aligned}$$

where we used Lemma 2.6 in (4.16) and in (4.17), we used (4.15) and the fact that $\lambda_2((\beta/\gamma)L_2) \leq 1$ to deduce $\rho((\gamma/\beta)L_2^* \setminus \{\mathbf{0}\}) \leq 1$. □

4.4 Low-Degree Hardness of Gaussian Pancakes

The *low-degree method* is a framework for predicting the computational hardness of hypothesis testing problems by computing the L^2-norm of the low-degree likelihood ratio (LDLR) with respect to the null distribution. More precisely, let L_n denote the likelihood ratio between the planted distribution $\mathcal{P}_n$ and null distribution $\mathcal{Q}_n$, and let $L_n^{\leqslant D}$ denote its orthogonal projection onto the subspace $\mathcal{V}^{\leqslant D}$ of (total) degree-D polynomials in $L^2(\mathcal{Q}_n)$. The low-degree method "predicts" the computational hardness of distinguishing $(\mathcal{P}_n, \mathcal{Q}_n)$ by computing bounds on $\|L_n^{\leqslant D}\|_{\mathcal{Q}_n}$. This method is typically used to predict hardness of *strong* detection (i.e., advantage tends to 1 asymptotically), and one predicts hardness if $\|L^{\leqslant D}\|_Q = O(1)$ for degree sequences satisfying $D = \omega(\log n)$. Upper bounds on $\|L_n^{\leqslant D}\|_{\mathcal{Q}_n}$, where L_n is the likelihood ratio for the Gaussian pancakes distinguishing problem and $D = D(n)$, is the main technical result of this section.

There are several motivations behind analyzing the L^2-norm of the low-degree likelihood ratio. One motivation is its connection to the second moment method for contiguity [LeC12], which

is a standard approach for establishing *statistical* indistinguishability between distributions. The low-degree method serves as a computational analogue of this method, where degree-D polynomials $f : \mathbb{R}^{n\times m} \to \mathbb{R}$ are viewed as proxies for decision rules $\Psi : \mathbb{R}^{n\times m} \to \{0,1\}$ that can be computed in $n^{O(D)}$ time. Strictly speaking, formal guarantees of the low-degree method are rather weak [KWB22, Section 4]. However, despite its formal limitations, the low-degree methods's success in predicting computational thresholds consistent with other evidence of hardness, its wide applicability[3], and its near equivalence with other lower bounds [BBHL+20; MW22] provide strong support for the reliability of its predictions. We refer the reader to [KWB22; Kun22; BAHS+22] for more details.

Following an extension of the framework proposed in [BAHS+22], which viewed $\|L^{\leqslant D}\|_Q = 1+o(1)$[4] as evidence that *weak* detection (i.e., advantage tends to $\Omega(1)$) is hard for $n^{O(D)}$ time algorithms, we consider the possibility of extending the analogy to *non-negligible* detection by viewing bounds of the form $\|L^{\leqslant D}\|_Q = 1 + \mathrm{negl}(n)$ as evidence that achieving even non-negligible advantage is hard. This motivation arises from the potential application of the low-degree method in predicting the run-times required to break cryptographic primitives, such as CLWE and Gaussian pancakes. Our results suggest that refinements are necessary to extend the low-degree method to this setting. In particular, an important aspect that requires further investigation is determining the appropriate amount of noise to introduce into "noiseless" problems to align with the requirements of the low-degree conjecture. The low-degree conjecture, as discussed in [KWB22], only applies to "noisy" problems. However, when constant Ornstein-Uhlenbeck noise is added to the Gaussian pancakes problem, it becomes information-theoretically impossible.

Our main low-degree lower bound is given by Theorem 4.21. This lower bound applies to any γ and even to *noiseless* Gaussian pancakes. This is in contrast to all known superpolynomial SQ lower bounds for Gaussian pancakes, which require at least sub-exponential noise [DKS17;

[3] For example, it can also be used to predict hardness of estimation problems [SW22].
[4] $\|L^{\leqslant D}\|_Q$ is trivially lower bounded by 1 for any $D \in \mathbb{N}$.

BRST21]. It is also worth mentioning that hardness based worst-to-average case reductions requires $\gamma \geqslant 2\sqrt{n}$ in addition to inverse polynomial noise. Thus, our low-degree lower bound provides insight into the hardness landscape of the Gaussian pancakes problem beyond parameter regimes studied using other techniques.

Theorem 4.21 (Total degree lower bound). *Let $n \in \mathbb{N}$, $D = D(n) \in \mathbb{N}$, $\gamma = \gamma(n) > \sqrt{2\pi}$ be a real number, and $m = m(n) \in \mathbb{N}$ such that $m \leqslant \exp(2\gamma^2)/(2\gamma^2)$. Then, there exist universal constants $C_1, C_2 > 0$ such that*

$$\left\|L^{\leqslant D}\right\|_Q^2 \leqslant \exp(m \cdot \exp(-C_1\gamma^2)) + 2D(4mD^2/\gamma)^{D/\gamma^2} \exp(-C_2 n) \ .$$

In particular, if $\log m = o(\gamma^2)$ and $D(\log m + 2\log D - \log \gamma) = o(\gamma^2 n)$, then

$$\|L^{\leqslant D}\|_Q^2 = 1 + \mathrm{negl}(n) \ .$$

The following lemma reduces the problem of bounding $\|L^{\leqslant D}\|_Q$ to computing $\mathbb{E}_{x\sim A}[h_k(x)]$, where $h_k : \mathbb{R} \to \mathbb{R}$ is the normalized k-th Hermite polynomial.

Lemma 4.22 (LDLR norm [MW21, Lemma 6.4]). *Let $\{h_k\}_{k\in\mathbb{N}}$ be the normalized Hermite polynomials (so that $\mathbb{E}_{z\sim\mathcal{N}(0,1)}[h_k(z)h_\ell(z)] = \mathbb{1}_{k=\ell}$). Then,*

$$\|L^{\leqslant D}\|^2 = \sum_{t=0}^{D} \mathop{\mathbb{E}}_{\boldsymbol{u},\boldsymbol{v}\sim\mu} [\langle \boldsymbol{u}, \boldsymbol{v}\rangle^t] \sum_{\substack{\alpha\in\mathbb{N}^m \\ |\alpha|=t}} \prod_{i=1}^{m} \left(\mathop{\mathbb{E}}_{x\sim A} [h_{\alpha_i}(x)] \right)^2 , \tag{4.18}$$

where $\boldsymbol{u}, \boldsymbol{v}$ are drawn independently from the prior distribution μ on $\mathbb{S}^{n-1}$.

4.4.1 Expectation of Hermite polynomials over discrete Gaussians

To compute the LDLR norm given by Lemma 4.22, we give tight bounds on $\mathbb{E}_{x\sim A}[h_k(x)]$ by exploiting the lattice structure of the support of A and using the Poisson summation formula (see

Lemma 2.6). Note that odd-degree Hermite polynomials are of secondary importance since their expectation evaluates to 0 for any distribution on $\mathbb{R}$ that is symmetric with respect to 0.

Our tight bounds on the expectation of Hermite polynomials over the univariate discrete Gaussian of unit width provide the following two useful insights: 1) if the degree k is "small" ($k \leqslant \gamma^2$), then $|\mathbb{E}_{x\sim A}[h_k(x)]|$ has negligible magnitude (Lemma 4.26) and 2) if the degree k is "large" ($k \geqslant \gamma^2$), then $|\mathbb{E}_{x\sim A}[h_k(x)]| = O(1/k^{1/4})$. Furthermore, this upper bound is tight for k that are in "sync" with the spacing parameter γ of A (Lemma 4.27 and Lemma 4.28). Such "synchronization" between k and γ^2 is expected since the Hermite *function* $h_k(x)\exp(-x^2/4)$ roughly behaves like the cosine function with frequency $\Theta(\sqrt{k})$. Numerical illustrations for Lemma 4.27 can be found in Figures 4.1 and 4.2.

Lemma 4.23 (Stirling bounds)**.** *Let $k \in \mathbb{Z}_+$. Then,*

$$\sqrt{2\pi k}(k/e)^k \leqslant k! \leqslant e\sqrt{k}(k/e)^k \ .$$

Claim 4.24. *Let $k \in \mathbb{N}$ and let $f(x) = h_k(x)\exp(-x^2/2)$. Then, its Fourier transform is given by*

$$\hat{f}(y) = \sqrt{2\pi}(-i)^k \frac{(2\pi y)^k}{\sqrt{k!}} \exp(-2\pi^2 y^2) \ . \tag{4.19}$$

Proof. We denote by H_k, the unnormalized k-th Hermite polynomial. Note that $H_k(x) = h_k(x)\sqrt{k!}$. By Rodrigues' formula,

$$H_k(x)\exp(-x^2/2) = (-1)^k \frac{d^k}{dx^k}\exp(-x^2/2) \ . \tag{4.20}$$

Using integration by parts, we observe that its Fourier transform is equal to $(2\pi iy)^k$ times the Fourier transform of $(-1)^k\exp(-x^2/2)$, which is $\sqrt{2\pi}(-1)^k\exp(-2\pi^2y^2)$. It follows that

$$\mathcal{F}\{H_k(x)\exp(-x^2/2)\} = \sqrt{2\pi}(-2\pi iy)^k\exp(-2\pi^2y^2) \ .$$

□

Lemma 4.25 (Expectation of Hermite polynomials)**.** *Let $\gamma \geqslant 1$ be a real number and let A be the discrete Gaussian of unit width supported on $(1/\gamma)\mathbb{Z}$. Then, for any $k \in 2\mathbb{N}$,*

$$\frac{1}{2}\left|\sum_{y\in\gamma\mathbb{Z}} C_k \cdot y^k \exp(-2\pi^2 y^2)\right| \leqslant \left|\mathop{\mathbb{E}}_{x\sim A}[h_k(x)]\right| \leqslant \left|\sum_{y\in\gamma\mathbb{Z}} C_k \cdot y^k \exp(-2\pi^2 y^2)\right| , \tag{4.21}$$

where $C_k = \frac{(-2\pi i)^k}{\sqrt{k!}}$.

Proof. Using the Poisson summation formula (Lemma 2.6), and Claim 4.24, we have

$$\begin{aligned}\left|\mathop{\mathbb{E}}_{x\sim A}[h_k(x)]\right| &= \frac{1}{\rho_{\sqrt{2\pi}}((1/\gamma)\mathbb{Z})}\left|\sum_{x\in(1/\gamma)\mathbb{Z}} h_{2k}(x)\exp(-x^2/2)\right| \\ &= \frac{\gamma}{\rho_{\sqrt{2\pi}}((1/\gamma)\mathbb{Z})}\left|\sum_{y\in\gamma\mathbb{Z}} \sqrt{2\pi}C_k y^k \exp(-2\pi^2 y^2)\right| . \end{aligned} \tag{4.22}$$

Using the loose bound $\rho_{\sqrt{2\pi}}((1/\gamma)\mathbb{Z}) \in (\gamma\sqrt{2\pi}, 2\gamma\sqrt{2\pi})$ in Eq. (4.22), the result follows. □

Lemma 4.26 (Upper bound for low-degree)**.** *Let $\gamma \geqslant \sqrt{2}$ be a real number and let $k \in \mathbb{Z}_+$ be a positive integer satisfying $k \leqslant \gamma^2$. Then,*

$$\left|\mathop{\mathbb{E}}_{x\sim A}[h_k(x)]\right| \leqslant \exp(-\gamma^2) . \tag{4.23}$$

Proof. If k is odd, then $\mathbb{E}_{x\sim A}[h_k(x)] = 0$, which trivially satisfies (4.23). Now suppose k is even and define $g(t) = \log(a_{t+1}/a_t)$, where a_t denotes the t-th term (to the right of 0) in the sum $\sum_{y\in\gamma\mathbb{Z}} C_k y^k \exp(-2\pi^2 y^2)$ and $C_k = \frac{(-2\pi i)^k}{\sqrt{k!}}$. The function $g(t)$ represents the decay rate of consecutive terms in the infinite sum. Then,

$$g(t) = k\log(1+1/t) - 2\pi^2\gamma^2\cdot(2t+1) \leqslant k/t - 4\pi^2\gamma^2 t - 2\pi^2\gamma^2 .$$

Also, for any $t \geqslant 1$

$$g'(t) = -\frac{k/t^2}{1+1/t} - 4\pi^2\gamma^2 = -\frac{k}{t^2+t} - 4\pi^2\gamma^2 < 0 \ .$$

If $k \leqslant 2\pi^2\gamma^2$, then $g(1) \leqslant k - 6\pi^2\gamma^2 \leqslant -4\pi^2\gamma^2$. This implies that the terms in the sum decay geometrically at a rate faster than $\exp(-4\pi^2\gamma^2) < 2^{-36}$ after the first term. Hence, the first term dominates the sum is dominant. To be more precise, this implies $|\mathbb{E}_{x\sim A}[h_k(x)]|$ is upper bounded by, say $3|C_k|\gamma^k \exp(-2\pi^2\gamma^2)$. Now we let $2 \leqslant k \leqslant \gamma^2$ (if $\gamma^2 \leqslant 2$, then it must be the case that $k = 1$ for which the expectation is 0). Then,

$$\begin{aligned}
|C_k|\gamma^k \exp(-2\pi^2\gamma^2) &\leqslant \frac{1}{\sqrt{k!}}(2\pi\gamma)^k \exp(-2\pi^2\gamma^2) \\
&\leqslant \frac{1}{(2\pi k)^{1/4}} \left(\frac{4\pi^2\gamma^2}{k/e}\right)^{k/2} \exp(-2\pi^2\gamma^2) \\
&\leqslant \left(4\pi^2 e\right)^{\gamma^2/2} \exp(-2\pi^2\gamma^2) \qquad (4.24) \\
&\leqslant \exp(\gamma^2(\log 2\pi + 1/2)) \exp(-2\pi^2\gamma^2) \\
&\leqslant \exp(-\gamma^2(2\pi^2 - 5/2)) \ ,
\end{aligned}$$

where we used a lower bound on the factorial (Lemma 4.23) and the fact that $2\pi \leqslant e^2$ in the last line. Eq. (4.24) follows from the fact that $1/(2\pi k)^{1/4} < 1$ and that $(4\pi^2\gamma^2 e/k)^{k/2}$ is increasing in the interval $k \in [2, \gamma^2]$.

Hence, $\mathbb{E}_{x\sim A}[h_k(x)] \leqslant 3 \cdot \exp(-\gamma^2(2\pi^2 - 5/2)) \leqslant \exp(-\gamma^2)$. □

Lemma 4.27 below states that the function mapping $s \in \mathbb{Z}$ to the s-th term of the sum in Eq. (4.21) behaves like a "delta" function, which has most of its mass concentrated on the $\lceil\sqrt{k/(2\pi^2\gamma^2)}\rfloor$-th term. See Figure 4.3 for a numerical illustration.

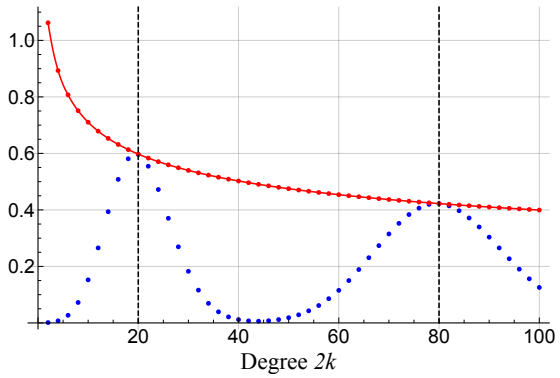


Figure 4.1: Plot of the expectation of Hermite polynomials $h_{2k}(x)$ over a 1D discrete Gaussian with $\gamma = \sqrt{20/(4\pi^2)}$. We can see that $|\mathbb{E}_{x\sim A}[h_{2k}(x)]|$ (Blue) lies below the upper bound $2.01/((4\pi k)^{1/4})$ (Red), and that the upper bound is tight when $\sqrt{2k/(4\pi^2\gamma^2)} \in \mathbb{N}$.

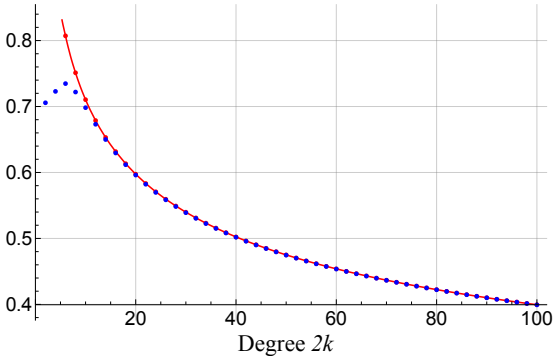


Figure 4.2: Plot of the expectation of Hermite polynomials where the spacing parameter γ is set to $\gamma(k) = \sqrt{2k/(4\pi^2)}$, so that it is in "sync" with the degree k of the Hermite polynomial. We can see that the $2.01/((4\pi k)^{1/4})$ upper bound (Red) tightly tracks $|\mathbb{E}_{x\sim A}[h_{2k}(x)]|$ (Blue).

Lemma 4.27 (Upper bound for high-degree)**.** *Let $\gamma \geqslant \sqrt{2}$ be a real number and let $k \in \mathbb{N}$. Then,*

$$\left|\mathop{\mathbb{E}}_{x\sim A}[h_k(x)]\right| \leqslant \frac{2.01}{(2\pi k)^{1/4}} \,. \tag{4.25}$$

Furthermore, if $k \in 2\mathbb{Z}_+$ and $\sqrt{k/(4\pi^2\gamma^2)} \in \mathbb{Z}_+$, this upper bound is tight. That is,

$$\mathop{\mathbb{E}}_{x\sim A}[h_k(x)] = (-1)^{k/2}\frac{2}{(2\pi k)^{1/4}}\left(1 + o_\gamma(1)\right) \,. \tag{4.26}$$

Proof. If k is odd, then by the expecation of $h_k(x)$ over A evaluates to 0, which satisfies Eq. (4.25) trivially. Hence, for the remainder of the proof we assume $k \in 2\mathbb{Z}_+$. Define the logarithm of the t-th term (to the right of 0) in the sum

$$r(t) := \log((t\gamma)^k \cdot \exp(-2\pi^2\gamma^2(t\gamma)^2)) = k\log(t\gamma) - 2\pi^2\gamma^2(t\gamma)^2 \,.$$

Since log is an increasing function, $r(t)$ is maximized when the term inside the logarithm is

maximized. By differentiating r with respect to t, we have

$$r'(t) = k/t - 4\pi^2\gamma^2 t \ ,$$

Hence, $r(t)$ achieves its maximum at $\sqrt{k/(4\pi^2\gamma^2)}$. Now let a_t denote the "t-th" term in the sum and again define $g(t) := \log(a_{t+1}/a_t)$. Then,

$$g(t) = k\log(1+1/t) - 2\pi^2\gamma^2 \cdot (2t+1) \leqslant k/t - 4\pi^2\gamma^2 t - 2\pi^2\gamma^2 = r'(t) - 2\pi^2\gamma^2 \ . \tag{4.27}$$

Now denote $s = \sqrt{k/(4\pi^2\gamma^2)}$. By Eq. (4.27), after the s-th term, the terms decay supergeometrically at a rate faster than $\exp(-2\pi^2\gamma^2)$. We also show that the terms decay at a rate faster than $\exp(-2\pi^2\gamma^2)$ in the other direction away from the "s-th" term. Note that $\log(a_{t-1}/a_t) = -g(t-1)$. Using the fact that $x/(1+x) \leqslant \log(1+x)$, we have for $t > 1$

$$g(t) \geqslant \frac{k/t}{1+1/t} - 2\pi^2\gamma^2(2t+1) = \frac{k}{t+1} - 2\pi^2\gamma^2(2t+1) \ .$$

It follows that for $t < \sqrt{k/(4\pi^2\gamma^2)}$

$$-g(t-1) \leqslant 2\pi^2\gamma^2(2t-1) - k/t < -2\pi^2\gamma^2 \ .$$

Thus, the terms decay at a rate faster than $\exp(-2\pi^2\gamma^2)$ in the other direction away from s as well. To account for rounding error for fractional s, consider $\tilde{g}(t) = \log(a_{t+1/2}/a_t)$. Then, for $t \geqslant s$,

$$\tilde{g}(t) = k\log(1+1/(2t)) - 2\pi^2\gamma^2(t+1/4) \leqslant r'(t)/2 - \pi^2\gamma^2/2 \leqslant -\pi^2\gamma^2/2 \ .$$

Also, as before, $\log(a_{t-1/2}/a_t) = -\tilde{g}(t-1/2)$. Then, a similar calculation shows that for $t \leqslant s$,

$$-\tilde{g}(t-1/2) \leqslant 2\pi^2\gamma^2(t-1/4) - k/(2t) = -r'(t)/2 - \frac{\pi^2\gamma^2}{2} \leqslant -\frac{\pi^2\gamma^2}{2} .$$

To finally establish Eq. (4.25), we observe that

$$\begin{aligned} |C_k| \cdot (s\gamma)^k \cdot \exp(-2\pi^2\gamma^2 s^2) &= \frac{(2\pi)^k}{\sqrt{k!}} \left(\frac{k}{4\pi^2}\right)^{k/2} \exp(-k/2) \\ &\leqslant \frac{(4\pi^2)^{k/2}}{(2\pi k)^{1/4}} \cdot \frac{e^{k/2}}{k^{k/2}} \cdot \left(\frac{k}{4\pi^2}\right)^{k/2} \cdot \exp(-k/2) \\ &= \left(\frac{1}{2\pi k}\right)^{1/4} , \end{aligned}$$

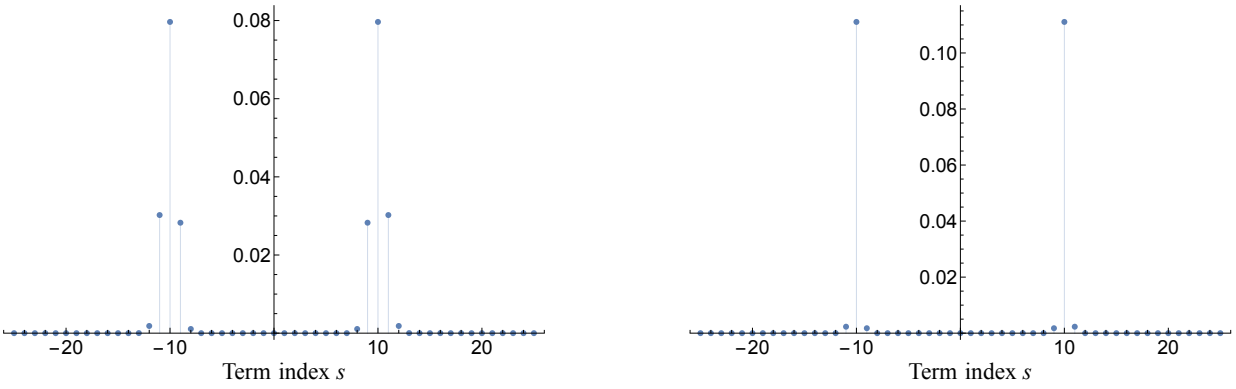
where $C_k = \frac{(-2\pi i)^k}{\sqrt{k!}}$, and we used Lemma 4.23, in particular the inequality $k! \geqslant (\sqrt{2\pi k})(k/e)^k$, in the second-to-last line.

Putting everything together, for $k \in 2\mathbb{Z}_+$ it holds

$$\begin{aligned} \left| \mathop{\mathbb{E}}_{x\sim A}[h_k(x)] \right| &\leqslant \sum_{t\in\mathbb{Z}} |C_k| (t\gamma)^k \exp(-2\pi^2(t\gamma)^2) \\ &= 2\sum_{t=1}^{\infty} |C_k| (t\gamma)^k \exp(-2\pi^2(t\gamma)^2) \\ &\leqslant 2|C_k| (s\gamma)^k \exp(-2\pi^2\gamma^2 s^2) \left(1 + e^{-\pi^2\gamma^2/2} + 2\sum_{r=1}^{\infty} e^{-2\pi^2\gamma^2 r}\right) \\ &\leqslant \frac{2.01}{(2\pi k)^{1/4}} , \end{aligned}$$

where in the last inequality, we used the fact that $e^{-\pi^2\gamma^2/2} + 2\sum_{r=1}^{\infty} e^{-2\pi^2\gamma^2 r} \leqslant 3e^{-\pi^2} \leqslant 0.005$ for $\gamma \geqslant \sqrt{2}$.

If $k \in 2\mathbb{Z}_+$ and $s \in \mathbb{Z}_+$, then using again Lemma 4.23 and the fact that $\rho_{\sqrt{2\pi}}((1/\gamma)\mathbb{Z}) = \gamma\sqrt{2\pi}(1+$



(a) $\gamma = 1/\sqrt{4\pi^2}, k = 100.$ **(b)** $\gamma = 2/\sqrt{4\pi^2}, k = 400.$

Figure 4.3: Plot of $C_k(s\gamma)^k \exp(-2\pi^2(s\gamma)^2)$ according to the term index $s \in \mathbb{Z}$ for various γ and k. Note that the plot behaves like a "delta" function supported on the index $s = \pm\sqrt{k/(4\pi^2\gamma^2)}$.

$o_\gamma(1)$),

$$\left|\mathop{\mathbb{E}}_{x\sim A}[h_k(x)]\right| \geqslant 2|C_k|(s\gamma)^k \exp(-2\pi^2\gamma^2 s^2)(1+o_\gamma(1))$$
$$\geqslant \frac{2}{(4\pi k)^{1/4}}(1+o_\gamma(1)) \ .$$

The above lower bound, together with (4.25), establishes Eq. (4.26). □

Lemma 4.28 (Dominant term for fractional γ)**.** *Let $\gamma \geqslant \sqrt{2}$ be a real number and let $k \in 2\mathbb{Z}_+$ be an even number such that $k = \lceil s^2(4\pi^2\gamma^2)\rceil$ or $k = \lfloor s^2(4\pi^2\gamma^2)\rfloor$ for some $s \in \mathbb{Z}_+$. Then,*

$$\mathop{\mathbb{E}}_{x\sim A}[h_k(x)] = \Theta_\gamma(1/\sqrt{s\gamma}) \ . \tag{4.28}$$

Proof. The upper bound $O(1/\sqrt{s\gamma})$ is immediate from Lemma 4.27. Hence, it suffices to show a lower bound of $\Omega(1/\sqrt{s\gamma})$. Let $k = s^2(4\pi^2\gamma^2) + \alpha$ (or $k = s^4(4\pi^2\gamma^2) - \alpha$), where $\alpha \in [0,1)$. Then, the maximizing "fractional" index $\tilde{s}$ can be expressed as

$$\tilde{s} = \sqrt{\frac{k}{4\pi^2\gamma^2}} = \sqrt{s^2 + \alpha/(4\pi^2\gamma^2)} \in [s, s\cdot e^{\alpha/(4\pi^2\gamma^2 s^2)}] \ .$$

We now analyze the magnitude of the s-th term relative to the fractional "$\tilde{s}$-th" term.

$$\begin{aligned}\log \frac{a_{\tilde{s}}}{a_s} &\leqslant k\alpha/(4\pi^2\gamma^2 s^2) \\ &= \alpha\left(1 + \alpha/(4\pi^2\gamma^2 s^2)\right) \\ &= \alpha(1 + o_\gamma(1)) \\ &= O_\gamma(\alpha) \ .\end{aligned}$$

Since $\alpha \in [0, 1)$, by Lemma 4.27, we have

$$\mathop{\mathbb{E}}_{x\sim A}[h_k(x)] = \Omega_\gamma(a_{\tilde{s}}) = \Omega_\gamma\left(\frac{1}{\sqrt{s\gamma}}\right) \ .$$

□

4.4.2 Low-degree lower bound for Gaussian pancakes

We now use the bounds on the expectation of univariate Hermite polynomials to prove our main low-degree lower bound. The key lemma is Lemma 4.29. We assume for the moment that the discrete distribution A is smoothed by small Ornstein-Uhlenbeck noise, and thus has a density with respect to the Lebesgue measure on $\mathbb{R}$. Recall that we denote by $P_{\boldsymbol{u}}$ the density of the planted distribution (of a *single* sample) conditioned on the direction $\boldsymbol{u} \in \mathbb{S}^{n-1}$, and $\bar{P}_{\boldsymbol{u}}(\boldsymbol{x}) = P_{\boldsymbol{u}}(\boldsymbol{x})/Q(\boldsymbol{x})$.

Lemma 4.29 (Upper bounds on coefficients of φ_D)**.** *Let $\gamma \geqslant \sqrt{2}$ be a real number and $m \in \mathbb{Z}_+$ be such that $m \leqslant \exp(2\gamma^2)/(2\gamma^2)$. If the total degree bound $t \in \mathbb{Z}_+$ satisfies $t \leqslant \gamma^2$, then*

$$\sum_{\substack{\alpha\in\mathbb{N}^m\setminus\{\mathbf{0}\}\\ \|\alpha\|_1\leqslant t}} \prod_{i=1}^{m}\left(\mathop{\mathbb{E}}_{x\sim A}[h_{2\alpha_i}(x)]\right)^2 \leqslant 2m\gamma^2\exp(-2\gamma^2) \ .$$

On the other hand, if $t > \gamma^2$, then

$$\sum_{\substack{\alpha \in \mathbb{N}^m \setminus \{\mathbf{0}\} \\ \|\alpha\|_1 = t}} \prod_{i=1}^{m} \left(\mathop{\mathbb{E}}_{x \sim A} [h_{2\alpha_i}(x)] \right)^2 \leqslant 3 \left(\frac{4mt}{\gamma} \right)^{t/\gamma^2} . \tag{4.29}$$

Proof. Given a non-zero degree vector $\alpha \in \mathbb{N}^m \setminus \{\mathbf{0}\}$, we split the indices of its support into two sets: 1) the set L of low degree indices which satisfy $2\alpha_i \leqslant \gamma^2$, and 2) the set H of "high" degree indices satisfying $2\alpha_i > \gamma^2$. Then,

$$\sum_{\substack{\alpha \in \mathbb{N}^m \\ \|\alpha\|_1 = t}} \prod_{i=1}^{m} \left(\mathop{\mathbb{E}}_{x \sim A} [h_{2\alpha_i}(x)] \right)^2 = \sum_{s=0}^{m} \sum_{\substack{L \subseteq [m] \\ \|L\|_1 \leqslant t \\ \|L\|_0 = s}} \sum_{\substack{H \subseteq [m] \setminus L \\ \|H\|_1 = t - \|L\|_1}} \prod_{i \in L} \left(\mathop{\mathbb{E}}_{x \sim A} [h_{2\alpha_i}(x)] \right)^2 \cdot \prod_{i \in H} \left(\mathop{\mathbb{E}}_{x \sim A} [h_{2\alpha_i}(x)] \right)^2$$

We show an upper bound on the RHS. Let $\alpha \in \mathbb{N}^m$ be a non-zero vector and let $\sum_{i \in L} 2\alpha_i = \ell$. If $\|L\|_0 = s$, we observe that $\ell \leqslant s\gamma^2$ since $\alpha_i \leqslant \gamma^2$ for any $i \in L$. By Claim 4.26, if $\|L\|_0 = s$ we have

$$\prod_{i \in L} \left(\mathop{\mathbb{E}}_{x \sim A} [h_{2\alpha_i}(x)] \right)^2 \leqslant \exp\left(-2\gamma^2 s\right) ,$$

If $t \leqslant \gamma^2$, then L is non-empty and

$$\begin{aligned} \sum_{s=1}^{m} \sum_{\substack{L \subseteq [m] \\ \|L\|_1 \leqslant t, \|L\|_0 = s}} \prod_{i \in L} \left(\mathop{\mathbb{E}}_{x \sim A} [h_{2\alpha_i}(x)] \right)^2 &\leqslant \sum_{s=1}^{m} \binom{m}{s} \gamma^{2s} \exp\left(-2\gamma^2 s\right) \qquad (4.30) \\ &\leqslant \sum_{s=1}^{m} \left(m\gamma^2 \exp\left(-2\gamma^2\right) \right)^s \\ &\leqslant 2m\gamma^2 \exp(-2\gamma^2) , \end{aligned}$$

where in (4.30) we used the fact that the number of subsets $L \subseteq [m]$ satisfying $\|L\|_0 = s$ is loosely upper bounded by $\binom{m}{s}\gamma^2$ since the number of possible values each α_i can take for each $i \in L$ is

less than γ^2. In addition, we used the assumption that $m\gamma^2 \exp(-2\gamma^2) \leqslant 1/2$ and bounded the sum by the geometric series of decay rate $1/2$ in the last line. This establishes the Lemma for the case $t \leqslant \gamma^2$.

Now we establish the bound for $t > \gamma^2$. From Lemma 4.27, it follows that for any $L \subset [m]$ and $t \geqslant 1$,

$$\begin{aligned}
\sum_{\substack{H\subseteq[m]\setminus L \\ \|H\|_1=d-\ell}} \prod_{i\in H} \left(\mathop{\mathbb{E}}_{x\sim A}[h_{2\alpha_i}(x)] \right)^2 &\leqslant \sum_{\substack{H\subseteq[m] \\ \|H\|_1=t-\ell}} \prod_{i\in H} \frac{1}{\sqrt{\pi\alpha_i}} \\
&= \sum_{r=1}^{\lfloor (t-\ell)/\gamma^2 \rfloor} \sum_{\substack{H\subseteq[m] \\ \|H\|_1=t-\ell, \\ \|H\|_0=r}} \prod_{i\in H} \frac{1}{\sqrt{\pi\alpha_i}} \\
&\leqslant \sum_{r=1}^{\lfloor (t-\ell)/\gamma^2 \rfloor} \binom{m}{r} \binom{(t-\ell)+r-1}{r-1} \left(\frac{2}{\gamma}\right)^r \\
&\leqslant \sum_{r=1}^{\lfloor (t-\ell)/\gamma^2 \rfloor} \binom{m}{r} \binom{2t-1}{r-1} \left(\frac{2}{\gamma}\right)^r \\
&\leqslant \sum_{r=1}^{\lfloor (t-\ell)/\gamma^2 \rfloor} \binom{m}{r} \binom{2t}{r} \left(\frac{2}{\gamma}\right)^r \\
&\leqslant \sum_{r=1}^{\lfloor (t-\ell)/\gamma^2 \rfloor} \left(\frac{4mt}{\gamma}\right)^r \\
&\leqslant 2\left(\frac{4mt}{\gamma}\right)^{\lfloor (t-\ell)/\gamma^2 \rfloor},
\end{aligned}$$

where we used the fact that $t > \gamma^2$ in the last line.

Putting everything together, for any $t \in \mathbb{Z}_+$ such that $t \geqslant \gamma^2$, we have

$$\begin{aligned}
\sum_{\substack{\alpha\in\mathbb{N}^m\\|\alpha|=t}} \prod_{i=1}^{m} \left(\mathop{\mathbb{E}}_{x\sim A}[h_{2\alpha_i}(x)]\right)^2 &= \sum_{s=0}^{m} \sum_{\substack{L\subseteq[m]\\\|L\|_0=s}} \sum_{\substack{H\subseteq[m]\setminus L\\\|H\|_1=t-\ell}} \prod_{i\in L} \left(\mathop{\mathbb{E}}_{x\sim A}[h_{2\alpha_i}(x)]\right)^2 \cdot \prod_{i\in H} \left(\mathop{\mathbb{E}}_{x\sim A}[h_{2\alpha_i}(x)]\right)^2 \\
&\leqslant \sum_{s=0}^{m} \sum_{\substack{L\subseteq[m]\\\|L\|_0=s}} \exp(-2\gamma^2 s) \cdot \sum_{\substack{H\subseteq[m]\setminus L\\\|H\|_1=t-\|L\|_1}} \prod_{i\in H} \left(\mathop{\mathbb{E}}_{x\sim A}[h_{2\alpha_i}(x)]\right)^2 \\
&\leqslant \exp(-2\gamma^2 m) + \sum_{s=0}^{m-1} \sum_{\substack{L\subseteq[m]\\\|L\|_0=s}} \exp(-2\gamma^2 s) \cdot 2\left(\frac{4mt}{\gamma}\right)^{\lfloor (t-\|L\|_1)/\gamma^2 \rfloor} \\
&\leqslant \exp(-2\gamma^2 m) + \sum_{s=0}^{m-1} \sum_{\substack{L\subseteq[m]\\\|L\|_0=s}} \exp(-2\gamma^2 s) \cdot 2\left(\frac{4mt}{\gamma}\right)^{\lfloor (t-s)/\gamma^2 \rfloor} \\
&\leqslant \exp(-2\gamma^2 m) + \sum_{s=0}^{m-1} \binom{m}{s} \gamma^{2s} \exp(-2\gamma^2 s) \cdot 2\left(\frac{4mt}{\gamma}\right)^{\lfloor (t-s)/\gamma^2 \rfloor} \qquad (4.31) \\
&\leqslant \exp(-2\gamma^2 m) + 2\left(\frac{4mt}{\gamma}\right)^{t/\gamma^2} \sum_{s=0}^{m-1} \left(m\gamma^2 \exp(-2\gamma^2)\right)^s \\
&\leqslant 3\left(\frac{4mt}{\gamma}\right)^{t/\gamma^2} .
\end{aligned}$$

where in (4.31) we again used the fact that the number of multisets $L \subseteq [m]$ satisfying $\|L\|_0 = s$ is upper bounded by $\binom{m}{s}\gamma^2$. In addition, we used the assumption $m\gamma^2 \exp(-2\gamma^2) \leqslant 1/2$ the last line.

□

Theorem 4.21 (Restated). *Let $n \in \mathbb{N}$, $D = D(n) \in \mathbb{N}$, $\gamma = \gamma(n) > \sqrt{2\pi}$ be a real number, and $m = m(n) \in \mathbb{N}$ such that $m \leqslant \exp(2\gamma^2)/(2\gamma^2)$. Then, there exist universal constants $C_1, C_2 > 0$ such that*

$$\left\|L^{\leqslant D}\right\|_Q^2 \leqslant \exp(m \cdot \exp(-C_1\gamma^2)) + 2D(4mD^2/\gamma)^{D/\gamma^2} \exp(-C_2 n) .$$

In particular, if $\log m = o(\gamma^2)$ *and* $D(\log m + 2\log D - \log \gamma) = o(\gamma^2 n)$*, then*

$$\|L^{\leqslant D}\|_Q^2 = 1 + \mathrm{negl}(n) \ .$$

Proof of Theorem 4.21. Let $\tau = \langle \boldsymbol{u}, \boldsymbol{v} \rangle$ denote the random variable given by random unit vectors $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{S}^{n-1}$ drawn uniformly and independently, and let ζ be the distribution of τ.

$$\begin{aligned} \left\|L^{\leqslant D}\right\|_{Q^{\otimes m}}^2 &= \mathop{\mathbb{E}}_{\tau \sim \zeta}[\varphi_D(\tau)] \\ &= \int_{|\tau| \leqslant \delta} \varphi_D(\tau) d\zeta(\tau) + \int_{|\tau| > \delta} \varphi_D(\tau) d\zeta(\tau) \ . \end{aligned}$$

For convenience, we write $R_1 = \int_{|\tau| \leqslant \delta} \varphi_D(\tau) d\zeta(\tau)$ and $R_2 = \int_{|\tau| > \delta} \varphi_D(\tau) d\zeta(\tau)$. We first upper bound R_1. We remark that $\varphi_{D_1}(\tau) \leqslant \varphi_{D_2}(\tau)$ whenever $D_1 \leqslant D_2$ since $\varphi_D(\tau)$ contains only even degree monomials with non-negative coefficients. For any $D \in \mathbb{N}$

$$\varphi_D(\langle \boldsymbol{u}, \boldsymbol{v} \rangle) = \langle (\bar{P}_{\boldsymbol{u}}^{\otimes m})^{\leqslant D}, (\bar{P}_{\boldsymbol{v}}^{\otimes m})^{\leqslant D} \rangle_{Q^{\otimes m}} = \langle \bar{P}_{\boldsymbol{u}}^{\leqslant D}, \bar{P}_{\boldsymbol{v}}^{\leqslant D} \rangle_Q^m \leqslant \langle \bar{P}_{\boldsymbol{u}}, \bar{P}_{\boldsymbol{v}} \rangle_Q^m \ .$$

Let $\delta \in (0,1)$ be a fixed constant (not dependent on n) such that $1 - \delta^2 \geqslant 2\pi/\gamma^2$. Then, by Lemma 4.20

$$\begin{aligned} R_1 &\leqslant \int_{|\langle \boldsymbol{u}, \boldsymbol{v} \rangle| \leqslant \delta} \langle \bar{P}_{\boldsymbol{u}}, \bar{P}_{\boldsymbol{v}} \rangle_Q^m d\zeta(\langle \boldsymbol{u}, \boldsymbol{v} \rangle) \\ &\leqslant \int_{|\langle \boldsymbol{u}, \boldsymbol{v} \rangle| \leqslant \delta} (1 + 8\exp(-\gamma^2(1-\delta^2)/2))^m d\zeta(\langle \boldsymbol{u}, \boldsymbol{v} \rangle) \\ &\leqslant (1 + 8\exp(-\gamma^2(1-\delta^2)/2))^m \\ &\leqslant \exp\left(m \exp(-O(\gamma^2)\right) \ , \end{aligned}$$

where in the last line we used the elementary inequality $1 + x \leqslant e^x$ which holds for all $x \in \mathbb{R}$.

Next, we upper bound R_2 by first upper bounding $\varphi_D(\tau)$ and then integrating over the region

$|\tau| > \delta$ with respect to measure ζ. Using Claim 4.29, we can bound $\varphi_D(\tau)$ on $\tau \in [0,1]$ as follows.

$$\begin{aligned}\varphi_D(\tau) &= \sum_{t=0}^{\lfloor D/2 \rfloor} \tau^{2t} \cdot \sum_{\substack{\alpha \in \mathbb{N}^m \\ |\alpha|=2t}} \prod_{i=1}^{m} \left(\mathop{\mathbb{E}}_{x \sim A} [h_{\alpha_i}(x)] \right)^2 \\ &\leqslant 1 + \sum_{t=1}^{\lfloor D/2 \rfloor} z^{2t} \cdot 3 \left(\frac{4mt}{\gamma} \right)^{2t/\gamma^2} \\ &\leqslant 2D \left(\frac{4mD}{\gamma} \right)^{D/\gamma^2} ,\end{aligned}$$

where in the last line we used the fact that $|\tau| \leqslant 1$ and that for any $t \geqslant 1$, it holds $2 \leqslant 4mt/\gamma \leqslant 4mD/\gamma$.

Hence,

$$\begin{aligned}R_2 &\leqslant 2D(4mD^2/\gamma)^{D/\gamma^2} \int \mathbb{I}[\tau > \delta] \cdot d\zeta(\tau) \\ &\leqslant 2D(4mD^2/\gamma)^{D/\gamma^2} \exp(-O(n)) .\end{aligned}$$

where the last line follows from standard bounds on the Beta distribution. Putting it all together,

$$\|L^{\leqslant D}\|^2 \leqslant \exp(m \cdot \exp(-O(\gamma^2))) + 2D(4mD^2/\gamma)^{D/\gamma^2} \exp(-O(n)) .$$

□

4.5 High-Sample Distinguisher

For $\gamma = o(\sqrt{n})$, the covariance matrix will reveal the discrete structure of Gaussian pancakes, which will lead to a subexponential time algorithm for the problem. This clarifies why the reduction for hCLWE does not extend beyond $\gamma \geq 2\sqrt{n}$.

We define *noiseless hCLWE distribution* $P_{\boldsymbol{u},\gamma}$ as $P_{\boldsymbol{u},\beta,\gamma}$ with $\beta = 0$. We begin with a claim that

will allow us to focus on the noiseless case.

Claim 4.31. *By adding Gaussian noise $D_{\mathbb{R}^n,\beta/\gamma}$ to $P_{\boldsymbol{u},\gamma}$ and then rescaling by a factor of $\gamma/\sqrt{\beta^2+\gamma^2}$, the resulting distribution is $P_{\boldsymbol{u},\tilde{\beta},\tilde{\gamma}}$, where $\tilde{\gamma} = \gamma/\sqrt{1+(\beta/\gamma)^2}$ and $\tilde{\beta} = \tilde{\gamma}(\beta/\gamma)$.*[5]

Proof. Without loss of generality, suppose $\boldsymbol{u} = \boldsymbol{e}_1$.

Let $z \sim P_{\boldsymbol{u},\gamma} + D_{\mathbb{R}^n,\beta/\gamma}$ and $\tilde{z} = \gamma z/\sqrt{\beta^2+\gamma^2}$. It is easy to verify that the marginals density of $\tilde{z}$ on subspace $\boldsymbol{e}_1^\perp$ will simply be ρ. Hence we focus on calculating the density of z_1 and $\tilde{z}_1$. The density can be computed by convolving the probability densities of $P_{\boldsymbol{u},\gamma}$ and $D_{\mathbb{R}^n,\beta/\gamma}$ as follows.

$$\begin{aligned} P_{\boldsymbol{u},\gamma} * D_{\mathbb{R}^n,\beta/\gamma}(z_1) &\propto \sum_{k\in\mathbb{Z}} \rho(k/\gamma)\cdot\rho_{\beta/\gamma}(z_1-k/\gamma) \\ &= \rho_{\sqrt{\beta^2+\gamma^2}/\gamma}(z_1)\cdot\sum_{k\in\mathbb{Z}}\rho_{\beta/\sqrt{\beta^2+\gamma^2}}\left(k/\gamma-\frac{\gamma^2}{\beta^2+\gamma^2}z_1\right) \\ &= \rho(\tilde{z}_1)\cdot\sum_{k\in\mathbb{Z}}\rho_{\tilde{\beta}}\left(k-\tilde{\gamma}\tilde{z}_1\right), \end{aligned}$$

where the second to last equality follows from Claim 2.5. This verifies that the resulting distribution is indeed $P_{\boldsymbol{w},\tilde{\beta},\tilde{\gamma}}$. □

Claim 4.31 implies that a Gaussian pancakes distribution with $\beta > 0$ is equivalent to a noiseless Gaussian pancakes with independent Gaussian noise added. We will first analyze the noiseless case and then derive the covariance of noisy (i.e., $\beta > 0$) case by adding independent Gaussian noise and rescaling.

Lemma 4.32. *Let $\Sigma \succ 0$ be the covariance matrix of the n-dimensional noiseless Gaussian pancakes $P_{\boldsymbol{u},\gamma}$ with $\gamma \geqslant 1$. Then,*

$$\left\|\Sigma - \frac{1}{2\pi}I_n\right\| \geq \gamma^2\exp(-\pi\gamma^2),$$

[5]Equivalently, in terms of the Gaussian mixture representation of Eq. (4.2), the resulting distribution has layers spaced by $1/\sqrt{\gamma^2+\beta^2}$ and of width $\beta/\sqrt{\gamma^2+\beta^2}$.

where $\|\cdot\|$ *denotes the spectral norm.*

Proof. Without loss of generality, let $\boldsymbol{u} = \boldsymbol{e}_1$. Then $P_{\boldsymbol{u},\gamma} = A \times Q_{n-1}$ where L is the one-dimensional lattice $(1/\gamma)\mathbb{Z}$. Then, $\Sigma = \mathrm{diag}(\mathbb{E}_{x\sim A}[x^2], \frac{1}{2\pi}, \ldots, \frac{1}{2\pi})$, so it suffices to show that

$$\left| \mathop{\mathbb{E}}_{x\sim A}[x^2] - \frac{1}{2\pi} \right| \geqslant \gamma^2 \exp(-\pi\gamma^2) \ .$$

Define $g(x) = x^2 \cdot \rho(x)$. The Fourier transform of ρ is itself; the Fourier transform of g is given by

$$\widehat{g}(y) = \left(\frac{1}{2\pi} - y^2\right)\rho(y) \ .$$

By definition and Poisson's summation formula (Lemma 2.6), we have

$$\begin{aligned} \mathop{\mathbb{E}}_{x\sim A}[x^2] &= \frac{g(L)}{\rho(L)} \\ &= \frac{\det(L^*)\cdot\widehat{g}(L^*)}{\det(L^*)\cdot\rho(L^*)} = \frac{\widehat{g}(L^*)}{\rho(L^*)} \ , \end{aligned}$$

where $L^* = ((1/\gamma)\mathbb{Z})^* = \gamma\mathbb{Z}$. Combining this with the expression for $\widehat{g}$, we have

$$\begin{aligned} \left| \mathop{\mathbb{E}}_{x\sim A}[x^2] - \frac{1}{2\pi} \right| &= \frac{\sum_{y\in L^*} y^2\rho(y)}{1 + \rho(L^* \setminus \{0\})} \\ &\geq \gamma^2 \exp(-\pi\gamma^2) \ , \end{aligned}$$

where we use the fact that for $\gamma \geqslant 1$,

$$\rho(\gamma\mathbb{Z} \setminus \{0\}) \leqslant \rho(\mathbb{Z} \setminus \{0\}) < 2\sum_{k=1}^{\infty} \exp(-\pi k) = \frac{2\exp(-\pi)}{1-\exp(-\pi)} < 1 \ .$$

□

Combining Claim 4.31 and Lemma 4.32, we get the following corollary for the noisy case.

Corollary 4.33. *Let $\Sigma \succ 0$ be the covariance matrix of n-dimensional Gaussian pancakes $P_{\boldsymbol{u},\beta,\gamma}$ with $\gamma \geqslant 1$ and $\beta > 0$. Then,*

$$\left\|\Sigma - \frac{1}{2\pi} I_n\right\| \geq \gamma^2 \exp(-\pi(\beta^2 + \gamma^2)) \,,$$

where $\|\cdot\|$ denotes the spectral norm.

Proof. Using Claim 4.31, we can view samples from $P_{\boldsymbol{u},\beta,\gamma}$ as samples from $P_{\boldsymbol{u},\gamma'}$ with independent Gaussian noise of width β'/γ' added and rescaled by $\gamma'/\sqrt{\beta'^2 + \gamma'^2}$, where β', γ' are given by

$$\begin{aligned} \beta' &= \beta\sqrt{1 + (\beta/\gamma)^2} \,, \\ \gamma' &= \sqrt{\beta^2 + \gamma^2} \,. \end{aligned}$$

Let Σ be the covariance of $P_{\boldsymbol{u},\beta,\gamma}$ and let Σ_0 be the covariance of $P_{\boldsymbol{u},\gamma'}$. Since the Gaussian noise added to $P_{\boldsymbol{u},\gamma'}$ is independent and $\beta'/\gamma' = \beta/\gamma$,

$$\Sigma = \frac{1}{1 + (\beta/\gamma)^2}\left(\Sigma_0 + \frac{(\beta/\gamma)^2}{2\pi} I_n\right) .$$

Hence,

$$\begin{aligned} \left\|\Sigma - \frac{1}{2\pi} I_n\right\| &= \frac{1}{1 + (\beta/\gamma)^2}\left\|\left(\Sigma_0 + \frac{(\beta/\gamma)^2}{2\pi} I_n\right) - \frac{1 + (\beta/\gamma)^2}{2\pi} I_n\right\| \\ &= \frac{1}{1 + (\beta/\gamma)^2}\left\|\Sigma_0 - \frac{1}{2\pi} I_n\right\| \\ &\geq \gamma^2 \exp(-\pi(\beta^2 + \gamma^2)) \,. \end{aligned}$$

where the last inequality follows from Lemma 4.32. □

We use the following lemma, which provides an upper bound on the error in estimating

the covariance matrix by samples. The sub-gaussian norm of a random variable Y is defined as $\|Y\|_{\psi_2} = \inf\{t > 0 \mid \mathbb{E}[\exp(Y^2/t^2)] \leq 2\}$ and that of an n-dimensional random vector $\boldsymbol{y}$ is defined as $\|\boldsymbol{y}\|_{\psi_2} = \sup_{\boldsymbol{u}\in\mathbb{S}^{n-1}} \|\langle \boldsymbol{y}, \boldsymbol{u}\rangle\|_{\psi_2}$.

Lemma 4.34 ([Ver18, Theorem 4.6.1])**.** *Let A be an $m \times n$ matrix whose rows A_i are independent, mean zero, sub-gaussian isotropic random vectors in $\mathbb{R}^n$. Then for any $u \geq 0$ we have*

$$\left\|\frac{1}{m}A^TA - I_n\right\| \leq K^2 \max(\delta, \delta^2) \quad \textit{where } \delta = C\left(\sqrt{\frac{n}{m}} + \frac{u}{\sqrt{m}}\right),$$

with probability at least $1 - 2e^{-u^2}$ for some constant $C > 0$. Here, $K = \max_i \|A_i\|_{\psi_i}$.

Combining Corollary 4.33 and Lemma 4.34, we have the following theorem for distinguishing Gaussian pancakes and Gaussian distribution.

Theorem 4.35. *Let $\gamma = n^{\varepsilon}$, where $\varepsilon < 1/2$ is a constant, and let $\beta = \beta(n) \in (0, 1)$. Then, there exists an $\exp(O(n^{2\varepsilon}))$-time algorithm that solves $\mathrm{hCLWE}_{\beta,\gamma}$.*

Proof. Our algorithm takes m samples from the unknown input distribution P and computes the sample covariance matrix $\Sigma_m = (1/m)X^TX$, where X's rows are the samples, and its eigenvalues $\mu_1, \ldots, \mu_n$. Then, it determines whether P is a Gaussian pancakes distribution or not by testing that

$$\left|\mu_i - \frac{1}{2\pi}\right| \leqslant \frac{1}{2} \cdot \gamma^2 \exp(-\pi(\beta^2 + \gamma^2)) \quad \text{for all } i \in [n] \ .$$

The running time of this algorithm is $O(n^2m) = \exp(O(n^{2\varepsilon}))$. To show correctness, we first consider the case $P = Q_n$. The standard Gaussian distribution satisfies the conditions of Lemma 4.34 (after rescaling by $1/(2\pi)$). Hence, the eigenvalues of Σ_m will be within distance $O(\sqrt{n/m})$ from $1/(2\pi)$ with high probability.

Now consider the case $P = P_{\boldsymbol{u},\beta,\gamma}$. We can assume $\boldsymbol{u} = \boldsymbol{e}_1$ without loss of generality since eigenvalues are invariant under rotations. Denote by $\boldsymbol{x}$ a random vector distributed according to

$P_{\boldsymbol{u},\beta,\gamma}$ and $\sigma^2 = \mathbb{E}_{\boldsymbol{x}\sim P_{\boldsymbol{u},\beta,\gamma}}[x_1^2]$. The covariance of P is given by

$$\Sigma = \begin{pmatrix} \sigma^2 & \mathbf{0} \\ \mathbf{0} & \frac{1}{2\pi} I_{n-1} \end{pmatrix} . \tag{4.32}$$

Now consider the sample covariance Σ_m of P and denote by $\sigma_m^2 = \boldsymbol{u}^T \Sigma_m \boldsymbol{u} = (1/m) \sum_{i=1}^m X_{i1}^2$. Since X_{i1}'s are sub-gaussian random variables [MP12, Lemma 2.8], $\sigma_m^2 - \sigma^2$ is a sum of m independent, mean-zero, sub-exponential random variables. For $m = \omega(n)$, Bernstein's inequality [Ver18, Corollary 2.8.3] implies that $|\sigma_m^2 - \sigma^2| = O(\sqrt{n/m})$ with high probability. By Corollary 4.33, we know that

$$\left|\sigma^2 - \frac{1}{2\pi}\right| \geqslant \gamma^2 \exp(-\pi(\beta^2 + \gamma^2)) .$$

Hence, if we choose $m = \exp(c\gamma^2)$ with some sufficiently large constant c, then Σ_m will have an eigenvalue that is noticeably far from $1/(2\pi)$ with high probability. □

5 | Lattice-Based Methods for Noiseless Inference

No problem whatever is completely exhausted. There remains always something to do.

George Polya[1]

We now show how LLL can be used to solve noiseless CLWE and hCLWE. In fact, we identify a new *class of problems* where the SoS hierarchy and low-degree lower bounds are provably bypassed by a polynomial-time algorithm. This class of problems is *not* based on linear equations, and the suggested optimal algorithm is *not* based on Gaussian elimination but on lattice basis reduction methods, which specifically seek to find a short non-zero vector in a lattice. Similar lattice-based methods have over the recent years been able to "close" various statistical-to-computational gaps [ZG18; AHSS17; SZB21], yet this is the first example we are aware of that they are able to close a gap suggested by SoS lower bounds.

The problems we analyze can be motivated from several angles in theoretical computer science and machine learning, and can be thought of as important special cases of well-studied problems such as NGCA. One important problem, other than CLWE or hCLWE, that can be solved

[1] [Pol04].

with LLL is the following: for a hidden unit vector $u \in \mathbb{R}^n$, we observe m independent samples

$$z_i \sim \mathcal{N}(x_i u, I_n - uu^\top), \quad i = 1, 2, \ldots, m, \tag{5.1}$$

where x_i are i.i.d. uniform ± 1, and the goal is to recover the hidden signs x_i and the hidden direction u (up to a global sign flip). Prior to our work, the best known poly-time algorithm required $n \gg d^2$ samples[2] [MW21]. Furthermore, this was believed to be unimprovable due to lower bounds against SoS algorithms and low-degree polynomials [MS16; Kun20; MRX20; GJJP+20; KB21; MW21; DDW21]. Nevertheless, we give a poly-time algorithm under the much weaker assumption $n \geqslant d+1$. In fact, this sample complexity is essentially optimal for the previous recovery problem (see [ZSWB22, Section 5]). Our result makes use of the Lenstra-Lenstra-Lovász (LLL) algorithm for lattice basis reduction [LLL82b], a powerful algorithm that has seen recent, arguably surprising, success in solving to information-theoretic optimality a few different "noiseless" statistical inference problems, some even in regimes where it was conjectured that no polynomial-time method works: linear regression with binary coefficients [ZG18; GKZ21], phase retrieval [AHSS17; SZB21], learning cosine neurons [SZB21], and CLWE [BRST21; SZB21][3]. Yet, to the best of our knowledge, this work is the first to establish the success of an LLL-based method in a regime where low-degree and SoS lower bounds both suggest computational intractability. This raises the question of whether LLL can "close" any other conjectured statistical-to-computational gaps. We believe that understanding the power and limitations of the LLL approach is an important direction for future research.

We also point out one weakness of the LLL approach: our algorithm is brittle to the specifics of the model, and relies on the observations being "noiseless" in some sense. For instance, our algorithm only solves the model in (5.1) because the x_i values lie *exactly* in ± 1 and the covariance $\Sigma = I - uu^\top$ has quadratic form $u^\top \Sigma u$ *exactly* equal to zero (or, similarly to other LLL applications

[2] Here are throughout, the notation $\gg$ hides logarithmic factors.
[3] in the exponentially-small noise regime

[ZG18], of exponentially small magnitude). If we were to perturb the model slightly, say by adding an inverse-polynomial amount of noise to the x_i's, our algorithm would break down because of the known non-robustness properties of the LLL algorithm. In fact, a noisy version (with inverse-polynomial noise) of one problem that we solve is the *homogeneous* Continuous Learning with Errors problem (hCLWE), which is provably hard based on the standard assumption [MR09, Conjecture 1.2] from lattice-based cryptography that certain worst-case lattice problems are hard against quantum algorithms [BRST21]. All existing algorithms for statistical problems based on LLL suffer from the same lack of robustness. In this sense, there is a strong analogy between LLL and the other known successful polynomial-time method for noiseless inference, namely the Gaussian elimination approach to learning parity: both exploit very precise algebraic structure in the problem and break down in the presence of even a small amount of noise.

As discussed above, our results "break" predictions of hardness based on SoS and low-degree lower bounds. Still, we believe that these types of lower bounds are interesting and meaningful, but some care should be taken when interpreting them. It is in fact already well-established that such lower bounds can sometimes be beaten on "noiseless" problems (a key example being Gaussian elimination). However, there are some subtleties in how "noiseless" should be defined here, and whether fundamental problems with statistical-to-computational gaps such as planted clique—which has implications for many other inference problems via average-case reductions (e.g., [BR13; MW15; HWX15; BBH18])—should be considered "noiseless." We discuss these issues further in Section 5.1.

Our main algorithmic result is informally stated as follows.

Theorem 5.1 (Informal). *Let $\gamma = \gamma(d)$ be polynomial in d. If $n = d + 1$ then there is an LLL-based algorithm for noiseless* hCLWE_γ *which terminates in polynomial time and outputs exactly, up to a global sign flip, both the correct labels $x_i \in (1/\gamma)\mathbb{Z}$ and the correct hidden direction $u \in \mathbb{S}^{d-1}$ with probability $1 - \exp(-\Omega(d))$.*

Problems	LDP LB	SoS LB	Our Results
Planted Vector (Rademacher)	$\tilde{\Omega}(d^2)$	$\tilde{\Omega}(d^{3/2})$	$d+1$
Gaussian Clustering (SNR $= \infty$)	$\tilde{\Omega}(d^2)$	$\tilde{\Omega}(d^{3/2})$	$d+1$
hCLWE (Noiseless)	-	-	$d+1$

Table 5.1: Sample complexity upper and lower bounds for *polynomial-time* exact recovery for Planted Rademacher Vector, Gaussian Clustering, and hCLWE.

5.1 Noiseless problems and SoS/low-degree lower bounds

SoS and low-degree lower bounds. The sum-of-squares (SoS) hierarchy [Par00; Las01] (see also surveys [BS16; RSS18; FKP19]) and low-degree polynomials [HS17; HKPR+17; Hop18] are two restricted classes of algorithms that are often studied in the context of stat-to-comp gaps. These are not the only two such frameworks, but we will focus on these two because our result "breaks" lower bounds in these two frameworks. SoS is a powerful hierarchy of semidefinite programming relaxations. Low-degree polynomial algorithms are simply multivariate polynomials in the entries of the input, of degree logarithmic in the input dimension; notably, these can capture all *spectral methods* (subject to some technical conditions), i.e., methods based on the leading eigenvalue/eigenvector of some matrix constructed from the input (see Theorem 4.4 of [KWB22]). Both SoS and low-degree polynomials have been widely successful at obtaining the best known algorithms for a wide variety of high-dimensional "planted" problems, where the goal is to recover a planted signal buried in noisy data. While there is no formal connection between SoS and low-degree algorithms, they are believed to be roughly equivalent in power [HKPR+17]. It is often informally conjectured that SoS and/or low-degree methods are as powerful as the best poly-time algorithms for "natural" high-dimensional planted problems (nebulously defined). As a result, lower bounds against SoS and/or low-degree methods are often considered strong evidence for inherent computational hardness of statistical problems.

Issue of noise-robustness. In light of the above, it is tempting to conjecture optimality of SoS and/or low-degree methods among all poly-time methods for a wide variety of statistical problems. While this conjecture seems to hold up for a surprisingly long and growing list of problems, there are, of course, limits to the class of problems for which this holds. As discussed previously, a well-known counterexample is the problem of learning parity (or the closely-related XOR-SAT problem), where Gaussian elimination succeeds in a regime where SoS and low-degree algorithms provably fail. This counterexample is often tossed aside by the following argument: "Gaussian elimination is a brittle algebraic algorithm that breaks down if a small amount of noise is added to the labels, whereas SoS/low-degree methods are more robust to noise and are therefore capturing the limits of poly-time *robust* inference, which is a more natural notion anyway. If we restrict ourselves to problems that are sufficiently *noisy* then SoS/low-degree methods should be optimal." However, we note that in our setting, SoS/low-degree methods are strictly suboptimal for a problem that *does* have plenty of Gaussian noise; the issue is that the signal and noise have a particular joint structure that preserves certain exact algebraic relationships in the data. This raises an important question: what exactly makes a problem "noisy" or "noiseless", and under what kinds of noise should we believe that SoS/low-degree methods are unbeatable? In the following, we describe one possible answer.

Low-degree conjecture. The "low-degree conjecture" of Hopkins [Hop18, Hypothesis 2.1.5] formalized one class of statistical problems for which low-degree polynomials are believed to be optimal among poly-time algorithms. These are certain *hypothesis testing* problems where the goal is to decide whether the input was drawn from a null (i.i.d. noise) distribution or a planted distribution (containing a planted signal). In our setting, one should imagine testing between n samples drawn from the model (5.1) and n samples drawn i.i.d. from $\mathcal{N}(0, I_d)$. Computational hardness of hypothesis testing generally implies hardness of the associated recovery/estimation/learning problem (which in our case is to recover x and u) as in Theorem 3.1 of [MW21].

The class of testing problems considered in Hopkins' conjecture has two main features: first, the problem should be highly symmetric, which is typical for high-dimensional statistical problems (although Hopkins' precise notion of symmetry does not quite hold for the problems we consider here). Second, and most relevant to our discussion, the problem should be *noise-tolerant*. More precisely, Hopkins' conjecture states that if low-degree polynomials fail to distinguish a null distribution $\mathbb{Q}$ from a planted distribution $\mathbb{P}$, then no poly-time algorithm can distinguish $\mathbb{Q}$ from *a noisy version of* $\mathbb{P}$. For our setting, the appropriate "noise operator" to apply to $\mathbb{P}$ (which was refined in [HW20]) is to replace each sample z_i by

$$\sqrt{1-\delta^2}z_i + \delta z_i'$$

where $z_i' \sim \mathcal{N}(0, I_d)$ independently from z_i, for an arbitrarily small constant $\delta > 0$. This has the effect of replacing x_i with $\sqrt{1-\delta^2}x_i + \delta\tilde{z}_i$ where $\tilde{z}_i \sim \mathcal{N}(0,1)$. This noise is designed to "defeat" brittle algorithms such as Gaussian elimination, and indeed our LLL-based algorithm is also expected to be defeated by this type of noise.

To summarize, the problem we consider here is *not* noise-tolerant in the sense of Hopkins' conjecture because the Gaussian noise depends on the signal (specifically, there is no noise in the direction of u) whereas Hopkins posits that the noise should be *oblivious* to the signal. Thus, in hindsight we should perhaps not be too surprised that LLL was able to beat SoS/low-degree for this problem. In other words, our result does not falsify the low-degree conjecture or the sentiment behind it (low-degree algorithms are optimal for noisy problems), with the caveat that one must be careful about the precise meaning of "noisy." We feel that this lesson carries an often-overlooked conceptual message that may have consequences for other fundamental statistical problems such as planted clique [Jer92; Kuč95].

5.2 Preliminaries

The key component of our algorithmic results is the LLL lattice basis reduction algorithm. The LLL algorithm receives as input d linearly independent vectors $v_1, \ldots, v_d \in \mathbb{Z}^d$ and outputs an integer linear combination of them with "small" ℓ_2 norm. Specifically, let us define the lattice generated by d *integer* vectors as simply the set of integer linear combination of these vectors.

The LLL algorithm solves a search problem called the *approximate* shortest vector problem (SVP) on a lattice L, given a basis of it.

Definition 5.2 (approximate SVP)**.** An instance of the algorithmic α-approximate SVP for a lattice $L \subseteq \mathbb{Z}^d$ is as follows. Given a lattice basis $v_1, \ldots, v_d \in \mathbb{Z}^d$ for the lattice L, find a vector $\widehat{x} \in L$, such that

$$\|\widehat{x}\|_2 \leq \alpha \cdot \mu(L) \ .$$

where $\mu(L) = \min_{x \in L, x \neq 0} \|x\|_2$.

The following theorem holds for the performance of the LLL algorithm, whose details can be found in [LLL82b].

Theorem 5.3 ([LLL82b])**.** *There is an algorithm (namely the LLL lattice basis reduction algorithm), which receives as input a basis for a lattice L given by $v_1, \ldots, v_d \in \mathbb{Z}^d$ which*

(1) returns a vector $v \in L$ satisfying $\|v\|_2 \leqslant 2^{d/2}\mu(L)$,

(2) terminates in time polynomial in d and $\log\left(\max_{i=1}^{d} \|v_i\|_\infty\right)$.

In this work, we use the LLL algorithm for an integer relation detection application, a problem which we formally define below.

Definition 5.4 (Integer relation detection)**.** An instance of the *integer relation detection problem* is as follows. Given a vector $b = (b_1, \ldots, b_k) \in \mathbb{R}^k$, find an $m \in \mathbb{Z}^k \setminus \{\mathbf{0}\}$, such that $\langle b, m \rangle := \sum_{j=1}^{k} b_j m_j = 0$. In this case, m is said to be an integer relation for the vector b.

To define our class of problems, we make use of the following two standard objects.

Definition 5.5 (Bernoulli–Rademacher vector)**.** We say that a random vector $v \in \mathbb{R}^n$ is a Bernoulli–Rademacher vector with parameter $\rho \in (0, 1]$ and write $v \sim \mathrm{BR}(n, \rho)$, if the entries of v are i.i.d. with

$$v_i = \begin{cases} 0 & \text{with probability } 1 - \rho, \\ 1/\sqrt{n\rho} & \text{with probability } \rho/2, \\ -1/\sqrt{n\rho} & \text{with probability } \rho/2. \end{cases}$$

5.3 The LLL-based algorithm

We now present the main contribution of this work, which is an LLL-based polynomial-time algorithm that provably solves the general problem defined in Model 5.7 with access to only $n = d + 1$ samples.

We deal formally with samples coming from d-dimensional Gaussians, which have as their mean some unknown multiple of an unknown unit vector $u \in \mathcal{S}^{d-1}$, and also some unknown covariance Σ which nullifies u and satisfies the following weak "separability" condition.

Assumption 5.6 (Weak separability of the spectrum)**.** *Fix a unit vector $u \in \mathcal{S}^{d-1}$. We say that a positive semi-definite $\Sigma \in \mathbb{R}^{d \times d}$ is u-weakly separable if for some constant $C > 0$ it holds that*

(a) $\Sigma u = 0$ *and,*

(b) All other eigenvalues of Σ *lie in the interval* $[d^{-C}, d^{C}]$.

Notice that in particular the canonical case $\Sigma = I - uu^\top$ is u-weakly separable as all eigenvalues of Σ are equal to one, besides the zero eigenvalue which has multiplicity one and eigenvector u.

Under the weak separability assumption we establish the following generic result.

Model 5.7 (Our general model). *Let $d, n \in \mathbb{N}$, known spacing level $a > 0$ satisfying $d^{-c} \leq a \leq d^c$ for some constant $c > 0$, and arbitrary $x_i \in \mathbb{Z} \cap [-2^d, 2^d]$, $i = 1, \ldots, n$. Consider also an arbitrary $u \in \mathcal{S}^{d-1}$ and an arbitrary unknown $\Sigma \in \mathbb{R}^{d \times d}$ which is u-weakly separable per Assumption 5.6. Conditional on u, Σ and $\{x_i\}_{i=1,\ldots,n}$, we then draw independent samples $z_1, \ldots, z_n \in \mathbb{R}^d$ where $z_i \sim \mathcal{N}((ax_i)u, \Sigma)$. The goal is to use z_i, $i = 1, \ldots, n$ to recover both $\{x_i\}_{i=1,\ldots,n}$ and u up to a global sign flip, with probability $1 - \exp(-\Omega(d))$ over the samples z_i, $i = 1, \ldots, n$.*

It is clear that noiseless Gaussian pancakes is a special instance of Model 5.7.

5.3.1 The algorithm and the main guarantee

In what follows, for some $N \in \mathbb{N}$ and $x \in \mathbb{R}$ we denote by $(x)_N := 2^{-N} \lfloor 2^N x \rfloor$ the truncation of x to its first N bits after zero.

Our proposed algorithm for solving Model 5.7 is described in Algorithm 1. Specifically we assume the algorithm receives $n = d + 1$ independent samples according to Model 5.7. As we see in the following theorem, the algorithm is able to recover exactly (up to a global sign flip) both the hidden direction u and the hidden labels x_i, $i = 1, \ldots, n$ in polynomial time.

Theorem 5.8. *Algorithm 1, given as input independent samples $(z_i)_{i=1,\ldots,d+1}$ from Model 5.7 with hidden direction u, covariance Σ, and true labels $\{x_i\}_{i=1,\ldots,d+1}$ satisfies the following with probability $1 - \exp(-\Omega(d))$: there exists $\varepsilon \in \{-1, 1\}$ such that the algorithm's outputs $\{\hat{x}_i\}_{i=1,\ldots,d+1}$ and $\hat{u} \in S^{d-1}$ satisfy*

$$\hat{x}_i = \epsilon x_i \quad \text{for } i = 1, \ldots, d+1$$
$$\text{and } \hat{u} = \epsilon u \ .$$

Algorithm 1: LLL-based algorithm for recovering $u, (x_i)_{i=1,\dots,d+1}$

Input: $n = d+1$ samples $z_i \in \mathbb{R}^d$, $i = 1, \dots, d+1$, spacing $a > 0$.
Output: Estimated labels $\hat{x}_i \in \mathbb{Z}$, $i = 1, \dots, d+1$ and unit vector $\hat{u} \in S^{d-1}$.

Construct a $d \times d$ matrix Z with columns $z_2, \dots, z_{d+1}$, and let $N = \lceil d^4 (\log d)^2 \rceil$.
if $\det(Z) = 0$ **then**
  return $\hat{u} = 0$ and output FAIL.
Compute $\lambda_1 = 1$ and $\lambda_i = \lambda_i(z_1, \dots, z_{d+1})$ given by $(\lambda_2, \dots, \lambda_{d+1})^\top = -Z^{-1} z_1$.
Set $M = 2^{2d}$ and $\tilde{v} = \left((\lambda_2)_N, \dots, (\lambda_{d+1})_N, 2^{-N}\right) \in \mathbb{R}^{d+1}$.
Output $(t_1, t_2) \in \mathbb{Z}^{d+1} \times \mathbb{Z}$ from running the LLL basis reduction algorithm on the lattice generated by the columns of the following $(d+2) \times (d+2)$ integer-valued matrix B,

$$B = \left(\begin{array}{c|c} M 2^N (\lambda_1)_N & M 2^N \tilde{v} \\ \hline 0_{(d+1)\times 1} & I_{(d+1)\times(d+1)} \end{array} \right).$$

$\hat{u}_0 \leftarrow \text{SolveLinearEquation}(u', Z^\top u' = a t_1)$.
if $\hat{u}_0 = 0$ **then**
  return $\hat{u} = 0$ and output FAIL.
Set $\hat{x}_i = (t_1)_i / \|\hat{u}_0\|_2$, $i = 1, \dots, d+1$.
return $\hat{x}_i$, $i = 1, \dots, d+1$ and $\hat{u}_0 / \|\hat{u}_0\|_2$ and output SUCCESS.

Moreover, Algorithm 1 terminates in $\text{poly}(d)$ *steps.*

We now provide intuition behind the algorithm's success. Note that for the unknown u and x_i it holds that

$$\langle z_i, u \rangle = a x_i \quad \text{for all } i = 1, \dots, d+1; . \tag{5.2}$$

In the first step, the algorithm checks a certain general-position condition on the received samples, which naturally is satisfied almost surely for our random data. In the following crucial three steps, the algorithm attempts to recover only the hidden integer labels x_i without learning u. To do this, it exploits a certain random integer linear relation that the labels x_i's satisfy which *importantly does not involve any information about the unknown* u, besides its existence. The key observation leading to this relation is the following. Since we have $d+1$ vectors z_i in a d-dimensional space, there exist scalars $\lambda_1, \dots, \lambda_{d+1}$ (depending on the z_i's) such that $\sum_{i=1}^{d+1} \lambda_i z_i = 0$.

These are exactly the λ_i's that the algorithm computes in the second step. Using them, observe that the following linear equation holds, due to (5.2),

$$\sum_{i=1}^{d+1} \lambda_i a x_i = \sum_{i=1}^{d+1} \lambda_i \langle z_i, u \rangle = \left\langle \sum_{i=1}^{d+1} \lambda_i z_i, u \right\rangle = \langle 0, u \rangle = 0 \,, \tag{5.3}$$

and therefore since $a > 0$ it gives the following integer linear equation

$$\sum_{i=1}^{d+1} \lambda_i x_i = 0. \tag{5.4}$$

Again note that the λ_i's can be computed from the given samples z_i, so this is an equation whose sole unknowns are the labels x_i. With this integer linear equation in mind, the algorithm in the following step employs the powerful LLL algorithm applied to an appropriate lattice. This application of the LLL is based on the breakthrough works of [Lag84; Fri86] for solving random subset-sum problems in polynomial-time, as well as its recent manifestations for solving various other noiseless inference settings such as binary regression [ZG18] and phase retrieval [AHSS17; SZB21]. To get some intuition for this connection, notice that in the case $x_i \in \{-1, 1\}$, (5.4) is really a (promise) subset-sum relation with weights λ_i and unknown subset $\{i : x_i = 1\}$ for which the corresponding λ_i's sum to $\frac{1}{2}\sum_{i=1}^{d+1} \lambda_i$. Now, after some careful technical work, including an appropriate truncation argument to work with integer-valued data, and various anti-concentration arguments such as the Carbery–Wright anticoncentration toolkit [CW01], one can show that the LLL step indeed recovers a constant multiple of the labels x_i, $i = 1, \ldots, d+1$ with probability $1 - \exp(-\Omega(d))$ (see also the next paragraph for more details on this). At this point, it is relatively straightforward to recover u using the linear equations (5.2).

Now we close by presenting the key technical lemma which ensures that LLL recovers the hidden labels x_i by finding a "short" vector in the lattice defined by the columns of the matrix B in Algorithm 1. Notice that if truncation at N bits was not present, that is we were "allowed"

to construct the lattice basis with the non-integer numbers λ_i instead of $(\lambda_i)_N$, then a direct calculation based on (5.4) would give that the hidden labels are embedded in an element of the lattice simply because we would have

$$B(0, x_1, \ldots, x_{d+1})^\top = (0, x_1, \ldots, x_{d+1})^\top.$$

As this "hidden vector" in the lattice is M-independent (and M is taken to be very large) this naturally suggests that this vector may be "short" compared to the others in the lattice. The following lemma states that with probability $1-\exp(-\Omega(d))$, this is indeed the case. The random lattice generated by the columns of B indeed does not contain any "spurious" short vectors other than the vector of the hidden labels and, naturally, its integer multiples. This implies that the LLL algorithm, despite its $2^{d/2}$ approximation ratio, will indeed return the integer relation that is "hidden in" the z_i's.

Lemma 5.9 (No spurious short vectors)**.** *Let $d \in \mathbb{N}$, $a \in [d^{-c}, d^c]$ for some constant $c > 0$ and $N = \lceil d^4(\log d)^2 \rceil$. Let $u \in S^{d-1}$ be an arbitrary unit vector, $\Sigma \in \mathbb{R}^{d\times d}$ an arbitrary unknown u-separable matrix, and let $x_i \in \mathbb{Z} \cap [-2^d, 2^d]$ for $i = 1, \ldots, d+1$ be arbitrary but not all zero. Moreover, let $\{z_i\}_{i=1,\ldots,d+1}$ be independent samples from $\mathcal{N}((ax_i)u, \Sigma)$, and let B be the matrix constructed in Algorithm 1 using $\{z_i\}_{i=1,\ldots,d+1}$ as input and N-bit precision. Then, with probability $1-\exp(-\Omega(d))$ over the samples, for any $t = (t_1, t_2) \in \mathbb{Z}^{d+1} \times \mathbb{Z}$ such that t_1 is not an integer multiple of $x = (x_1, \ldots, x_{d+1})$, the following holds:*

$$\|Bt\|_2 > 2^{2d}\ .$$

The proof of Lemma 5.9 is in Section 5.4.3.

5.4 Proof of Algorithm 1 correctness

5.4.1 Towards proving Theorem 5.8: auxiliary lemmas

We present here three auxiliary lemmas for proving Theorem 5.8 and Lemma 5.9. The first lemma establishes that given a small (in ℓ_2-norm) "approximate" integer relation between real numbers, one can appropriately truncate each real number to a sufficiently large number of bits, so that the truncated numbers satisfy a small (in ℓ_2-norm) integer relation between them. This lemma, which is an immediate implication of [SZB21, Lemma D.6], is important for the appropriate application of the LLL algorithm, which needs to receive integer-valued input. Recall that for a real number x we denote by $(x)_N$ its truncation to its first N bits after zero, i.e. $(x)_N := 2^{-N}\lfloor 2^N x \rfloor$.

Lemma 5.10 ("Rounding" approximate integer relations [SZB21, Lemma D.6])**.** *Let $d \in \mathbb{N}$ be a number and let $n \in \mathbb{N}$ be such that $n \leq C_0 d$ for some constant $C_0 > 0$. Moreover, suppose for some constant $C_1 > 0$, a (real-valued) vector $s \in \mathbb{R}^n$ satisfies $\langle m, s\rangle = 0$ for some $m \in \mathbb{Z}^n$. Then for some sufficiently large constant $C > 0$, if $N = \lceil d^4(\log d)^2 \rceil$, there is an $m' \in \mathbb{Z}^{n+1}$ which is equal to m in the first n coordinates, satisfies $\|m'\|_2 \leq Cd^{\frac{1}{2}}\|m\|_2$, and is an integer relation for the numbers $(s_1)_N, \ldots, (s_n)_N, 2^{-N}$.*

We need the following anticoncentration result.

Lemma 5.11 (Anticoncentration of misaligned integer combinations)**.** *Assume that $d^c > a > 1/d^c$ for some $c > 0$ constant. Let $u \in S^{d-1}$ be an arbitrary unit vector and let $x_1, \ldots, x_{d+1} \in \mathbb{Z}$ be an arbitrary sequence of integers, which are not all equal to zero. Now for a sequence of integers $t = (t_1, \ldots, t_{d+1}) \in \mathbb{Z}^{d+1}$, we define the (multi-linear) polynomial $P_t(z_1, \ldots, z_{d+1})$ in $d(d+1)$ variables by*

$$P_t(z_1, \ldots, z_{d+1}) = \det(Z)t_1 + \sum_{i=2}^{d+1} \det(Z_{-i})t_i \,, \tag{5.5}$$

where each $z_1, \ldots, z_{d+1}$ is assumed to have a d-dimensional vector form, Z denotes the $d \times d$ matrix with $z_2, \ldots, z_{d+1}$ as its columns, and each Z_{-i} for $i = 2, \ldots, d+1$ denotes the $d \times d$ matrix formed by swapping out the $(i-1)$-th column of Z with $-z_1$.

Suppose z_i's are drawn independently from $\mathcal{N}((ax_i)u, \Sigma)$ for some $u \in \mathcal{S}^{d-1}$ and $\Sigma \in \mathbb{R}^{d \times d}$ which is u-weakly separable per Assumption 5.6 and eigenvalues $0 = \lambda_1 < \lambda_2 \leq \lambda_3 \leq \ldots \leq \lambda_d$. Then, for any $t \in \mathbb{Z}^{d+1}$ it holds that

$$\mathbb{E}[P_t(z_1, \ldots, z_{d+1})] = 0 \tag{5.6}$$

and

$$\mathrm{Var}(P_t(z_1, \ldots, z_{d+1})) = (d-1)! a^{2d} \left(\prod_{i=2}^{d} \lambda_i \right)^2 \sum_{1 \leq i < j \leq d+1} (t_i x_j - t_j x_i)^2 \,. \tag{5.7}$$

Furthermore, for some universal constant $B > 0$ the following holds. If $t \neq cx$ for any $c \in \mathbb{R}$, where we denote $x = (x_1, \ldots, x_{d+1})$, then for any $\epsilon > 0$,

$$\mathbb{P}(|P_t(z_1, \ldots, z_{d+1})| \leq \epsilon) \leq B d^B \epsilon^{\frac{1}{d}}. \tag{5.8}$$

Proof. We first describe how (5.8) follows from (5.6) and (5.7). First, notice that under the assumption on the integer sequence $t_i, i = 1, \ldots, d+1$ not being a multiple of the sequence of integers $x_i, i = 1, \ldots, d+1$ it holds that for some $i, j = 1, \ldots, d+1, i \neq j$ with $(t_i x_j - t_j x_i)^2 \geq 1$. In particular, using (5.7) we have

$$\mathrm{Var}(P_t(z_1, \ldots, z_{d+1})) \geq (d-1)! a^{2d} \left(\prod_{i=2}^{d} \lambda_i \right)^2 .$$

But now notice that from Assumption 5.6 and $a > d^{-c}$, it holds for some constant $C' > 0$ that

$$a^{2d}\left(\prod_{i=2}^{d}\lambda_i\right)^2 \geq d^{-C'd}.$$

Hence, it holds that

$$\mathrm{Var}(P_t(z_1,\ldots,z_{d+1})) \geq d^{-C'd}.$$

Now we employ [MNV16, Theorem 1.4] (originally proved in [CW01]) which implies that for some universal constant $B > 0$, since our polynomial is multilinear and has degree $d+1$, it holds for any $\epsilon > 0$ that

$$\mathbb{P}\left(|P_t(z_1,\ldots,z_{d+1})| \leq \epsilon\sqrt{\mathrm{Var}(P_t(z_1,\ldots,z_{d+1}))}\right) \leq Bd\epsilon^{\frac{1}{d}}.$$

Using our lower bound on the variance we conclude the result.

Now we proceed with the mean and variance calculation. As this statement is about the first and second moment of P_t and the determinant operator is invariant up to basis transformations, we may assume without loss of generality that $u = e_1$, that is, u is equal to the first standard basis vector, and the remaining standard basis vectors are the remaining eigenvectors of Σ. Recall that z_i's are drawn in an independent fashion from $\mathcal{N}((ax_i)u, \Sigma)$. Hence for a sequence of i.i.d. $w_i \sim \mathcal{N}(0, I_{d-1}), i = 1,\ldots,d+1$ we may assume from now on that,

$$z_i = \begin{bmatrix} ax_i \\ \Lambda w_i \end{bmatrix} \tag{5.9}$$

for $\Lambda := \mathrm{diag}(\lambda_2,\ldots,\lambda_d)$.

Now let us define the $(d-1)\times(d-1)$ matrix W_{-j} for each $2 \leqslant j \leqslant d+1$ as the matrix formed

using $w_2, \ldots, w_{d+1}$ *except* w_j as its column vectors, and define functions $\psi_i : \mathbb{R}^{(d-1)\times(d-1)} \to \mathbb{R}$ for each $i = 2, \ldots, d+1$ to be the determinant of W_{-j} with the column corresponding to w_i swapped by $-w_1$. For instance, if $2 \leqslant i \neq j \leqslant d+1$, then

$$\psi_i(W_{-j}) := \det(w_2, \ldots, w_{i-1}, -w_1, w_{i+1}, \ldots, w_{j-1}, w_{j+1}, \ldots, w_{d+1}). \tag{5.10}$$

We abuse notation and also write

$$\psi_1(W_{-j}) := \det(w_2, \ldots, w_{j-1}, w_{j+1}, \ldots w_{d+1}) = \det(W_{-j}). \tag{5.11}$$

As the result is clearly a-homogeneous of degree $2d$ we assume in what follows that $a = 1$. Now by direct expansion along the first row of the corresponding matrices we have

$$\det(Z) = \sum_{j=2}^{d+1} (-1)^j x_j |\det(\Lambda)| \psi_1(W_{-j}) \,,$$

and for each $i \geq 2$,

$$\det(Z_{-i}) := (-1)^{i+1} x_1 |\det(\Lambda)| \psi_1(W_{-i}) + \sum_{j=2, j\neq i}^{d+1} (-1)^j x_j |\det(\Lambda)| \psi_i(W_{-j}) \,.$$

Since $d > 1$ and w_i are i.i.d. $\mathcal{N}(0, I_d)$ we can immediately conclude that for all $i \geq 1, j \geq 2, i \neq j$,

$$\mathbb{E}[\psi_i(W_{-j})] = 0.$$

Hence,

$$\mathbb{E}[P_t(z_1, \ldots, z_{d+1})] = t_1 \mathbb{E}[\det(Z)] + \sum_{i=2}^{d+1} t_i \mathbb{E}[\det(Z_{-i})] = 0.$$

Now we calculate the second moment of the polynomial. In what follows, we slightly abuse

notation and denote $Z_{-1} := Z$ for notational convenience. First, again by direct expansion of the determinant and the fact that w_i's for $i = 1, \ldots, d+1$ have i.i.d. standard Gaussian entries it holds by direct inspection that for all $i, j \in [d+1]$ with $i \neq j$,

$$\mathbb{E}[\psi_i(W_{-j})^2] = (d-1)! |\det(\Lambda)|^2 , \tag{5.12}$$

and unless $\{i, j\} = \{k, \ell\}$, it holds that

$$\mathbb{E}[\psi_i(W_{-j})\psi_k(W_{-\ell})] = 0. \tag{5.13}$$

We now calculate for $2 \leq i \neq j \leq d+1$ the term $\mathbb{E}[\psi_i(W_{-j})\psi_j(W_{-i})]$. We assume without loss of generality that $i < j$. Notice that for $\Pi_c \in \{0,1\}^{d-1 \times d-1}$, the permutation matrix corresponding to the cycle-permutation $c := (i-1, i, \ldots, j, j-1) \in \mathrm{Sym}([d-1])$, the matrix

$$(w_2, \ldots w_{i-1}, -w_1, w_{i+1}, \ldots, w_{j-1}, w_{j+1}, \ldots, w_{d+1}) ,$$

equals

$$\Pi_c(w_2, \ldots w_{i-1}, w_{i+1}, \ldots, w_{j-1}, -w_1, w_{j+1}, \ldots, w_{d+1}) .$$

Hence,

$$\psi_i(W_{-j})\psi_j(W_{-i}) = \det(\Pi_c)\psi_i^2(W_{-j}) = (-1)^{\mathrm{sgn}(c)}\psi_i^2(W_{-j}) = (-1)^{i-j+1}\psi_i^2(W_{-j}) .$$

In particular,

$$\mathbb{E}[\psi_i(W_{-j})\psi_j(W_{-i})] = (-1)^{i-j+1}(d-1)! |\det(\Lambda)|^2 . \tag{5.14}$$

Now using (5.12), (5.13), we have for each $1 \le i \le d+1$,

$$\mathbb{E}[\det(Z_{-i})^2] = (d-1)! \sum_{j=1, j\neq i}^{d+1} x_j^2 |\det(\Lambda)|^2 , \tag{5.15}$$

and using (5.12), (5.13), and (5.14) we have for all $i \neq j$ that

$$\mathbb{E}[\det(Z_{-i})\det(Z_{-j})] = -(d-1)! x_i x_j |\det(\Lambda)|^2 . \tag{5.16}$$

Hence, it holds that

$$\begin{aligned}
\mathbb{E}[P_t(z_1,\dots,z_{d+1})^2] &= |\det(\Lambda)|^2 \sum_{i,j=1}^{d+1} t_i t_j \mathbb{E}[\det(Z_{-i})\det(Z_{-j})] \\
&= |\det(\Lambda)|^2 \sum_{i=1}^{d+1} t_i^2 \mathbb{E}[\det(Z_{-i})^2] + \sum_{i,j=1, i\neq j}^{d+1} t_i t_j \mathbb{E}[\det(Z_{-i})\det(Z_{-j})] \\
&= (d-1)! |\det(\Lambda)|^2 \left(\sum_{i,j=1, i\neq j}^{d+1} t_i^2 x_j^2 - \sum_{i,j=1, i\neq j}^{d+1} t_i t_j x_i x_j \right) \\
&= (d-1)! |\det(\Lambda)|^2 \left(\sum_{1\le i<j\le d+1} (t_i x_j - t_j x_i)^2 \right) .
\end{aligned}$$

□

The following lemma establishes multiple structural properties of the $d+1$ samples.

Lemma 5.12. *Let $u \in S^{d-1}$ be an arbitrary unit vector and let $x_i \in \mathbb{Z} \cap [-2^d, 2^d]$ for $i = 1,\dots,d+1$ be arbitrary integers which are not all equal to zero. Let also spacing a with $d^{-c} < a < d^c$ for some $c > 0$ and Σ which is u-weakly separable per Assumption 5.6. We observe $d+1$ samples of the form z_i, where for each $i = 1,\dots,d+1$, z_i is an independent sample from $\mathcal{N}((ax_i)u, \Sigma)$. We denote by $Z \in \mathbb{R}^{d\times d}$ the (random) matrix with columns given by the d vectors $z_2,\dots,z_{d+1}$. The following properties hold.*

(1) The matrix Z is invertible almost surely.

(2) With probability $1 - \exp(-\Omega(d))$ over the z_i's,

$$\|Z^{-1}z_1\|_\infty = O(2^{2d^2}).$$

(3) With probability $1 - \exp(-\Omega(d))$ over the z_i's,

$$0 < |\det(Z)| = O(2^{d^2}).$$

Proof. For the fact that Z is invertible, consider its determinant, that is, the random variable $\det(Z)$. We claim that $\det(Z) \neq 0$ almost surely. Note that to prove this, by invariance of the determinant to the change of basis, we may assume without loss of generality that $u = e_1$, that is, u is the first standard basis vector, and the remaining standard basis vectors are the remaining eigenvectors of Σ. Under this assumption, for each $i = 1, \ldots, d+1$, we can write using Assumption 5.6

$$z_i = \begin{bmatrix} ax_i \\ \Lambda w_i \end{bmatrix},$$

where $\Lambda = \mathrm{diag}(\lambda_2, \ldots, \lambda_d)$ and w_i's are i.i.d. samples from $\mathcal{N}(0, I_{d-1})$. In other words, the first row of Z consists of $ax_2, \ldots, ax_{d+1}$, and the rest are coordinates of Λw_i, where each w_i is a vector with i.i.d. standard Gaussian entries. Now the result follows from the fact that since not all x_i are equal to zero and also none of the λ_i's are zero from Assumption 5.6, the determinant $\det(Z)$ with fixed $x_2, \ldots, x_{d+1}$ is a non-zero polynomial of the entries of $w_2, \ldots, w_{d+1}$. As all entries of w_i are distributed as i.i.d. standard Gaussians, the random polynomial $\det(Z)$ is almost surely non-zero [CT05].

For the second part, notice that by Cramer's rule for $i = 1, \ldots, d-1$, the i-th coordinate

of $Z^{-1}z_1$ equals the quantity $\lambda_{i+1}(Z) := \det(z_2, \ldots, z_i, -z_1, z_{i+1}, \ldots, z_{d+1})/\det(Z)$ almost surely. Hence, again by the rotational invariance property of the determinant operator, we may assume that $u = e_1$ and the remaining standard basis vectors are the remaining eigenvectors of Σ. Let $q^{(i)} \in \mathbb{Z}^{d+1}$ be an integer-valued vector such that $q^{(i)}_j = 1$ if $i = j$ and $q^{(i)}_j = 0$ otherwise. Now using the notation of Lemma 5.11 we have that $P_{q^{(i)}}(z_1, \ldots, z_{d+1}) = \det(Z_{-i})$. By applying the anticoncentration result from Lemma 5.11 for the polynomial $P_{q^{(1)}}(z_1, \ldots, z_{d+1})$ and $\epsilon = 2^{-d^2}$ we conclude that

$$|\det(Z)| = |P_{q^{(1)}}(z_1, \ldots, z_{d+1})| \geq 2^{-d^2} \tag{5.17}$$

with probability $1 - \exp(-\Omega(d))$. Furthermore, for all $i = 1, \ldots, d+1$ it holds that

$$\mathbb{E}[P_{q^{(i)}}(z_1, \ldots, z_{d+1})^2] = \mathrm{Var}(P_{q^{(i)}}(z_1, \ldots, z_{d+1})) = a^{2d} d! \|x\|_2 \leqslant a^{2d} |\det(\Lambda)|^2 2^{10d \log d} \|x\|_2^2 ,$$

where $x := (x_1, \ldots, x_{d+1})^\top$ where $\Lambda = \mathrm{diag}(\lambda_2, \ldots, \lambda_d)$ and $\lambda_i, i > 1$ are the non-zero eigenvalues of Σ per Assumption 5.6. Hence, by Markov's inequality, the fact that $a < d^c$, the Assumption 5.6 and a union bound over i, we have for all $i = 1, \ldots, d+1$ that

$$|P_{q^{(i)}}(z_1, \ldots, z_{d+1})| \leqslant 2^{d^2/2} \|x\|_2^2 \tag{5.18}$$

with probability $1 - \exp(-\Omega(d))$.

Combining Eq.(5.17) and Eq.(5.18), we conclude that for all $i = 2, \ldots, d$,

$$|\lambda_i(Z)| = |P_{q^{(i)}}(z_1, \ldots, z_{d+1})/P_{q^{(1)}}(z_1, \ldots, z_{d+1})| \leq 2^{3d^2/2} \|x\|_2^2$$

with probability $1 - \exp(-\Omega(d))$. Since $\|x\|_2^2 = O(2^{2d})$ we have $\|Z^{-1}z_1\|_\infty \leq 2^{3d^2/2} \|x\|_2^2 \leq 2^{2d^2}$ with probability $1 - \exp(-\Omega(d))$. This concludes the proof of the second part.

Finally, Eq.(5.18) for $i = 1$ and the fact $\|x\|_2^2 = O(2^{2d})$ imply

$$|\det(Z)| = |P_{q^{(1)}}(z_1, \ldots, z_{d+1})| \leqslant 2^{d^2} \tag{5.19}$$

with probability $1 - \exp(-\Omega(d))$. This concludes the proof of the third part. □

5.4.2 Proof of Theorem 5.8

We now proceed with the proof of the Theorem 5.8 using the lemmas from the previous sections.

Theorem 5.8 (Restated). *Algorithm 1, given as input independent samples $(z_i)_{i=1,\ldots,d+1}$ from Model 5.7 with hidden direction u, covariance Σ, and true labels $\{x_i\}_{i=1,\ldots,d+1}$ satisfies the following with probability $1 - \exp(-\Omega(d))$: there exists $\varepsilon \in \{-1, 1\}$ such that the algorithm's outputs $\{\hat{x}_i\}_{i=1,\ldots,d+1}$ and $\hat{u} \in S^{d-1}$ satisfy*

$$\hat{x}_i = \epsilon x_i \quad \text{for } i = 1, \ldots, d+1$$
$$\text{and } \hat{u} = \epsilon u \ .$$

Moreover, Algorithm 1 terminates in $\mathrm{poly}(d)$ *steps.*

Proof. We start with noticing that for an algorithm to recover u, x_i up a to global sign flip it suffices to recover the values of $\{x_i\}_{i=2,\ldots,d+1}$ up to a global non-zero constant multiple. Indeed, since we already know the value of z_i's, if we learn the x_i's up to a constant, call it $C > 0$, then we can solve the linear system of d (independent) equations and with d unknowns given by $\langle z_i, v\rangle = Cax_i = C\langle z_i, u\rangle, i = 2, \ldots, d+1$. Since by Lemma 5.12 the matrix Z, also formed in Algorithm 1, which is the $d \times d$ matrix with $z_2, \ldots, z_{d+1}$ as its column vectors, is invertible almost surely, one can indeed solve this linear system to recover $v = Cu$, that is the same constant C times u. Since u is assumed to be unit norm one can then recover the quantity $|C| = \|v\|_2$, which is the absolute value of the unknown constant. Hence one can output for some $\epsilon = C/|C| \in \{-1, 1\}$ the

estimated vector $Cu/|C| = \epsilon u$ and the estimated labels $Cx_i/|C| = \epsilon x_i, i = 1, \ldots, d+1$ which are indeed the hidden direction u and the true labels $x_i, i = 1, \ldots, d+1$ up to a global sign flip.

Now our proposed Algorithm 1 follows exactly this path: it first recovers a non-zero constant multiple of the x_i's (this is the values of the vector t_1 output by the LLL step) with probability $1 - \exp(-\Omega(d))$. Then it uses the simple procedure described above to output both the labels $x_i, i = 1, \ldots, d+1$ and u up to a global constant multiple. This second part comprises exactly the last steps of the algorithm after the LLL step. The main procedure of our algorithm therefore is to use an appropriate application of LLL to learn the exact values of x_i up to a global sign flip. We now analyze the success of the LLL step to recover a global constant multiple of the x_i's with probability $1 - \exp(-\Omega(d))$.

Now the algorithm does not terminate in the second step exactly because of the almost sure invertibility of the matrix Z, per Lemma 5.12. Let us now analyze the (random) lattice $L = L(B)$ generated by the basis B, which is constructed in the next step of Algorithm 1.

First, observe that the real numbers $\{\lambda_i\}_{i=1,2,\ldots,d+1}$ used in the top row of the lattice basis B, satisfy by definition

$$\sum_{i=1}^{d+1} \lambda_i z_i = 0 \ .$$

Hence, we conclude that since $\langle z_i, u \rangle = a x_i$ for the unknown direction $u \in S^{d-1}$ and spacing $a > 0$, it holds that

$$\sum_{i=1}^{d+1} \lambda_i a x_i = \sum_{i=1}^{d+1} \lambda_i \langle u, z_i \rangle = \langle u, \sum_{i=1}^{d+1} \lambda_i z_i \rangle = 0 \tag{5.20}$$

and therefore

$$\sum_{i=1}^{d+1} \lambda_i x_i = 0 \ . \tag{5.21}$$

We now show an upper bound on the shortest vector length of L, which we denote by $\mu(L)$. More precisely, we show that

$$\mu(L) = O(d2^d) \ .$$

To this end, define a real-valued vector $s \in \mathbb{R}^{d+1}$ with $s_i = \lambda_i$ for $i = 1, \ldots, d+1$, and also an integer-valued vector $m \in \mathbb{Z}^{d+1}$ with $m_i = x_i$ for $i = 1, \ldots, d+1$. Then, the integer relation (5.21) implies that $\langle s, m \rangle = 0$. Since $|x_i| \leq 2^d$ for all $i = 1, \ldots, d+1$ it also holds almost surely that $\|m\|_2 = \|x\|_2 \leq \sqrt{d}2^d$. By Lemma 5.10, for the bit-precision N chosen by Algorithm 1, there exists an integer $m'_{d+2} \in \mathbb{Z}$ such that $m' = (m, m'_{d+2}) \in \mathbb{Z}^{d+2}$ satisfies $\|m'\|_2 = O(d2^d)$ and is an integer relation for $(\lambda_1)_N, \ldots, (\lambda_{d+1})_N, 2^{-N}$.

Now define $b \in (2^{-N}\mathbb{Z})^{d+2}$ given by $b_i = (\lambda_i)_N$ for $i = 1, \ldots, d+1$, and $b_{d+2} = 2^{-N}$. Notice that $b_1 = (1)_N = 1$ and furthermore that the $\tilde{v}$ defined by the algorithm satisfies $\tilde{v} = (b_2, \ldots, b_{d+2})$. On top of this, we have that the m' defined in previous paragraph is an integer relation for b with $\|m'\|_2 = O(d2^d)$. Hence, $Bm' = (0, m')^\top$. It follows that $\mu(L) = O(d2^d)$ with probability $1 - \exp(-\Omega(d))$, since $\mu(L) \leqslant \|Bm'\|_2 = O(d2^d)$.

Recall from Theorem 5.3 that the LLL algorithm is guaranteed to return a lattice vector of ℓ_2-norm smaller than $2^{\frac{d+2}{2}}\mu(L)$. Now we employ Lemma 5.9 which combined with the fact that $2^{\frac{d+2}{2}}\mu(L) \leq 2^{2d}$ for sufficiently large d almost surely, allows us to conclude that the LLL algorithm returns a non-zero lattice vector $B(t_1, t_2)^\top$, where $t_1 \in \mathbb{Z}^{d+1}$ and $t_2 \in \mathbb{Z}$, such that t_1 is an integer multiple of $x = (x_1, \ldots, x_{d+1})$ with probability $1 - \exp(-\Omega(d))$. Hence, using t_1 the algorithm recovers a global non-zero constant multiple of the x_i's for $i = 1, \ldots, d+1$ with probability $1 - \exp(-\Omega(d))$.

For the termination time, it suffices to establish that the step using the LLL basis reduction algorithm can be performed in $\text{poly}(d)$ time. To ensure $\text{poly}(d)$ time for the LLL step, it suffices to show that the entries of the lattice basis B are not too large with probability $1 - \exp(-\Omega(d))$.

More precisely, the running time of LLL depends on the logarithm of the largest entry in B by Theorem 5.3. Clearly, N and $\log M$ are polynomial in d. Finally, direct inspection and Lemma 5.12 implies that the quantity $\log \|\lambda\|_\infty$, where $\lambda = (\lambda_1, \ldots, \lambda_{d+1})^\top$ is as defined in Algorithm 1, is polynomially bounded with probability $1 - \exp(-\Omega(d))$. This establishes the $\mathrm{poly}(d)$ running time of the LLL step. □

5.4.3 Proof of Lemma 5.9

We focus this section on proving the key technical Lemma 5.9. As mentioned above, the proof of the lemma is quite involved, and, potentially interestingly, it requires the use of anticoncentration properties of the coefficients λ_i, which are rational functions of the coordinates of x_i, as discussed in Lemma 5.11.

Lemma 5.9 (Restated). *Let $d \in \mathbb{N}$, $a \in [d^{-c}, d^c]$ for some $c > 0$ and $N = \lceil d^4 (\log d)^2 \rceil$. Let $u \in S^{d-1}$ be an arbitrary unit vector, $\Sigma \in \mathbb{R}^{d \times d}$ an arbitrary u-weakly separable matrix and let $x_i \in \mathbb{Z} \cap [-2^d, 2^d]$ for $i = 1, \ldots, d+1$ be arbitrary but not all zero. Moreover, let $\{z_i\}_{i=1,\ldots,d+1}$ be independent samples from $N((ax_i)u, \Sigma)$, and let B be the matrix constructed in Algorithm 1 using $\{z_i\}_{i=1,\ldots,d+1}$ as input and N-bit precision. Then, with probability $1 - \exp(-\Omega(d))$ over the samples, for any $t = (t_1, t_2) \in \mathbb{Z}^{d+1} \times \mathbb{Z}$ such that t_1 is not an integer multiple of $x = (x_1, \ldots, x_{d+1})$, the following holds:*

$$\|Bt\|_2 > 2^{2d} .$$

Proof of Lemma 5.9. Let $t = (t_1, t_2) \in \mathbb{Z}^{d+1} \times \mathbb{Z}$ be arbitrary non-zero integer coefficients. Our proof consists of characterizing integer coefficients t for which the corresponding lattice vector Bt is "short", that is,

$$\|Bt\|_2 \leqslant 2^{2d}. \tag{5.22}$$

In what follows, by a *short lattice vector* we refer to the condition (5.22).

We first show that with probability $1 - \exp(-\Omega(d))$, lattice vectors can only be short for integer coefficients contained in some bounded rectangle $\mathcal{R} \subset \mathbb{Z}^{d+2}$, which we define below (see Eq.(5.24)). Then, we apply our anticoncentration lemma (Lemma 5.11) and a union bound over a subset of $\mathcal{R}$ to conlcude that with probability $1 - \exp(-\Omega(d))$, the only short lattice vectors are ones whose integer coefficients satisfy $t_1 = cx$ for some $c \in \mathbb{Z}$.

To this end, we first observe that entries of the first row of B are elements of $M\mathbb{Z}$, as by direct inspection $(Bt)_1 = M(\sum_{i=1}^{d+1} \left(2^N (\lambda_i)_N\right) (t_1)_i + t_2)$. It follows that if t is not an integer relation for the numbers $(\lambda_1)_N, \ldots, (\lambda_{d+1})_N, 2^{-N}$, then $\|Bt\|_2 \geq M = 2^{2d}$. Hence, it suffices to restrict our attention to t's which are integer relations, that is,

$$\sum_{i=1}^{d+1} (\lambda_i)_N (t_1)_i + t_2 2^{-N} = 0 \ .$$

Note that it cannot be the case that $t_1 = 0$ since this implies, by the integer relation above, $t_2 = 0$, and therefore the pair $t = (t_1, t_2)$ are zero, a contradiction. Hence, from now on we restrict ourselves only to the case where $t_1 \neq 0$.

Let us denote by t' the vector t without the first coordinate $(t_1)_1$, i.e., $t' = ((t_1)_2, \ldots, (t_1)_{d+1}, t_2)$. Our second observation is that $\|Bt\|_2 \geqslant \|t'\|_\infty$ because of the use of the submatrix I_{d+1} in the definition of B. This implies that any short lattice vector Bt must satisfy $\|t'\|_\infty \leqslant 2^{2d}$. Moreover, since t is an integer relation and $\lambda_1 = 1$, we have

$$|(t_1)_1| = \left| \sum_{i=2}^{d+1} (\lambda_i)_N (t_1)_i + t_2 2^{-N} \right| \leqslant \|t'\|_\infty \left(\|\lambda\|_1 + 2^{-N} \right) \ . \tag{5.23}$$

Now in the notation of Lemma 5.12 we have $\lambda = -Z^{-1} z_1$. Hence using Lemma 5.12 and the elementary fact that $\|\lambda\|_1 \leqslant (d+1)\|\lambda\|_\infty$, it holds with probability $1 - \exp(-\Omega(d))$ that $\|\lambda\|_1 = O(2^{2d^2})$. It follows that, for sufficiently large d, any short lattice vector Bt must satisfy

$|(t_1)_1| \leqslant 2^{3d^2}$ with probability $1 - \exp(-\Omega(d))$. Hence, with probability $1 - \exp(-\Omega(d))$, every short vector Bt in the random lattice $L = L(B)$ has its integer coefficients t contained in $\mathcal{R}$, which is defined as

$$\mathcal{R} = \{(a, b) \in \mathbb{Z} \times \mathbb{Z}^{d+1} \ : \ |a| \leqslant 2^{3d^2}, \|b\|_\infty \leqslant 2^{2d}\} \ . \tag{5.24}$$

From $\mathcal{R}$, we also define $\mathcal{R}_1 \subset \mathbb{Z}^{d+1}$ such that $\mathcal{R}_1 = \{t_1 \in \mathbb{Z}^{d+1} \ : \ t = (t_1, t_2) \in \mathcal{R}\}$.

We now show using a union bound over $t \in \mathcal{R}$ that with probability $1-\exp(-\Omega(d))$, the only short lattice vectors in L are ones whose integer coefficients $t = (t_1, t_2)$ satisfy $t_1 = cx$ for some $c \in \mathbb{Z}$. First, observe that since $|t_2| \leq 2^{2d}$, the following inequality holds if t is an integer relation:

$$\left| \sum_{i=1}^{d+1} (\lambda_i)_N (t_1)_i \right| \leq 2^{2d} 2^{-N} \ .$$

Consider $\mathcal{T}$ the set of all $t_1 \in \mathbb{Z}^{d+1} \setminus \bigcup_{c \in \mathbb{R}} \{c(x_1, \ldots, x_{d+1})^\top\}$. To prove our result it suffices to prove that

$$\mathbb{P}\left(\bigcup_{t_1 \in \mathcal{T} \cap \mathcal{R}_1} \left\{ \left| \sum_{i=1}^{d+1} (\lambda_i)_N (t_1)_i \right| \leq 2^{2d}/2^N \right\} \right) \leq \exp(-\Omega(d))$$

for which, since for any x it holds $|x - (x)_N| \leq 2^{-N}$ and $\|t\|_1 = |(t_1)_1| + \|t'\|_\infty \leq 2^{4d^2}$ for sufficiently large d, it suffices to prove that for large d,

$$\mathbb{P}\left(\bigcup_{t_1 \in \mathcal{T} \cap \mathcal{R}_1} \left\{ \left| \sum_{i=1}^{d+1} \lambda_i (t_1)_i \right| \leq 2^{5d^2}/2^N \right\} \right) \leq \exp(-\Omega(d)) \ .$$

Using the polynomial notation of Lemma 5.11 (specifically, Eq.(5.5)), as well as the fact that by Cramer's rule λ_i are rational functions of the coordinates of z_i satisfying $\lambda_i \det(z_2, \ldots, z_{d+1}) =$

$\det(\ldots, z_{i-1}, -z_1, z_{i+1}, \ldots)$, it suffices to show

$$\mathbb{P}\left(\bigcup_{t_1 \in \mathcal{T} \cap \mathcal{R}_1} \{|P_{t_1}(z_1, \ldots, z_{d+1})| \le |\det(z_2, \ldots, z_{d+1})| 2^{5d^2}/2^N\} \right) \le \exp(-\Omega(d)) \ .$$

By Lemma 5.12, with probability $1 - \exp(-\Omega(d))$ there exists some constant $D > 0$ such that $\det(z_2, \ldots, z_{d+1}) \leqslant D2^{2d^2}$. Hence, it suffices to show

$$\mathbb{P}\left(\bigcup_{t_1 \in \mathcal{T} \cap \mathcal{R}_1} \{|P_{t_1}(z_1, \ldots, z_{d+1})| \le D2^{7d^2}/2^N\} \right) \le \exp(-\Omega(d)) \ .$$

Now since $N = \omega(d^2 \log d)$, it suffices to show, for sufficiently large d,

$$\mathbb{P}\left(\bigcup_{t_1 \in \mathcal{T} \cap \mathcal{R}_1} \{|P_{t_1}(z_1, \ldots, z_{d+1})| \le 2^{-\frac{N}{2}}\} \right) \le \exp(-\Omega(d)) \ .$$

By a union bound, it suffices to show

$$\sum_{t_1 \in \mathcal{T} \cap \mathcal{R}_1} \mathbb{P}\left(|P_t(z_1, \ldots, z_{d+1})| \le 2^{-\frac{N}{2}} \right) \le 2^{-\Omega(d)}. \tag{5.25}$$

Now the number of integer points t_1 with ℓ_∞ norm at most 2^{3d^2} is at most $2^{3d^2(d+1)}$, since there are at most 2^{3d^2} choices per coordinate. Furthermore, using the anticoncentration inequality (5.8) of Lemma 5.11, we have for any $t_1 \in \mathcal{T}$ that for some universal constant $B > 0$,

$$\mathbb{P}\left(|P_{t_1}(z_1, \ldots, z_{d+1})| \le 2^{-\frac{N}{2}} \right) \le Bd2^{-\frac{N}{2d}} \ .$$

Using the above to upper bound the left hand side of (5.25), we see that the sum is at most

$$Bd2^{3d^2(d+1)}2^{-\frac{N}{2d}} = \exp(O(d^3) - \Omega(N/d)) = \exp(-\Omega(d)) \ ,$$

where we used that $N/d = \Omega(d^3 \log d)$. This completes the proof. □

Bibliography

[AG11] Sanjeev Arora and Rong Ge. "New algorithms for learning in presence of errors". In: *ICALP*. ICALP'11. 2011, pp. 403–415 (page 16).

[AHSS17] Alexandr Andoni, Daniel Hsu, Kevin Shi, and Xiaorui Sun. "Correspondence retrieval". In: *COLT*. Vol. 65. Proceedings of Machine Learning Research. PMLR, 2017, pp. 105–126 (pages 76, 77, 86).

[AK05] Sanjeev Arora and Ravi Kannan. "Learning mixtures of separated nonspherical Gaussians". In: *Ann. Appl. Probab.* 15.1A (2005), pp. 69–92 (page 37).

[AR05] Dorit Aharonov and Oded Regev. "Lattice problems in NP ∩ CoNP". In: *J. ACM* 52.5 (2005), pp. 749–765 (page 17).

[Bab86] L Babai. "On Lovász' lattice reduction and the nearest lattice point problem". In: *Combinatorica* 6.1 (1986), pp. 1–13 (page 28).

[BAHS+22] Afonso S Bandeira, Ahmed El Alaoui, Samuel B Hopkins, Tselil Schramm, Alexander S Wein, and Ilias Zadik. "The Franz-Parisi Criterion and Computational Tradeoffs in High Dimensional Statistics". In: *arXiv preprint arXiv:2205.09727* (2022) (page 56).

[BB20] Matthew Brennan and Guy Bresler. "Reducibility and statistical-computational gaps from secret leakage". In: *Conference on Learning Theory*. PMLR. 2020, pp. 648–847 (pages 3, 4).

[BBH18] Matthew Brennan, Guy Bresler, and Wasim Huleihel. “Reducibility and computational lower bounds for problems with planted sparse structure”. In: *Conference On Learning Theory*. PMLR. 2018, pp. 48–166 (page 78).

[BBHL+20] Matthew Brennan, Guy Bresler, Samuel B Hopkins, Jerry Li, and Tselil Schramm. “Statistical query algorithms and low-degree tests are almost equivalent”. In: *arXiv preprint arXiv:2009.06107* (2020) (page 56).

[BHKK+19] Boaz Barak, Samuel Hopkins, Jonathan Kelner, Pravesh K Kothari, Ankur Moitra, and Aaron Potechin. “A nearly tight sum-of-squares lower bound for the planted clique problem”. In: *SIAM Journal on Computing* 48.2 (2019), pp. 687–735 (page 4).

[BKMM+19] Jean Barbier, Florent Krzakala, Nicolas Macris, Léo Miolane, and Lenka Zdeborová. “Optimal errors and phase transitions in high-dimensional generalized linear models”. In: *Proceedings of the National Academy of Sciences* 116.12 (2019), pp. 5451–5460 (page 15).

[BKSS+06] Gilles Blanchard, Motoaki Kawanabe, Masashi Sugiyama, Vladimir Spokoiny, Klaus-Robert Müller, and Sam Roweis. “In Search of Non-Gaussian Components of a High-Dimensional Distribution.” In: *Journal of Machine Learning Research* 7.2 (2006) (page 36).

[BLPR+13] Zvika Brakerski, Adeline Langlois, Chris Peikert, Oded Regev, and Damien Stehlé. “Classical hardness of learning with errors”. In: *STOC*. 2013, pp. 575–584 (page 43).

[BLPR19] Sebastien Bubeck, Yin Tat Lee, Eric Price, and Ilya Razenshteyn. “Adversarial examples from computational constraints”. In: *ICML*. Vol. 97. ICML ’19. 2019, pp. 831–840 (page 14).

[BNHR22] Andrej Bogdanov, Miguel Cueto Noval, Charlotte Hoffmann, and Alon Rosen. “Public-Key Encryption from Continuous LWE”. In: *Cryptology ePrint Archive* (2022) (pages 4, 8, 36, 38).

[BPW18] Afonso S Bandeira, Amelia Perry, and Alexander S Wein. "Notes on computational-to-statistical gaps: predictions using statistical physics". In: *Portugaliae Mathematica* 75.2 (2018), pp. 159–186 (page 3).

[BR13] Quentin Berthet and Philippe Rigollet. "Computational lower bounds for sparse PCA". In: *arXiv preprint arXiv:1304.0828* (2013) (pages 3, 4, 78).

[BRST21] Joan Bruna, Oded Regev, Min Jae Song, and Yi Tang. "Continuous LWE". In: *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*. 2021 (pages 6, 35, 39, 57, 77, 78).

[BS16] Boaz Barak and David Steurer. "Proofs, beliefs, and algorithms through the lens of sum-of-squares". In: *Course notes: http://www. sumofsquares. org/public/index. html* (2016) (page 79).

[BV08] Spencer Charles Brubaker and Santosh Vempala. "Isotropic PCA and affine-invariant clustering". In: *FOCS*. FOCS '08. 2008, pp. 551–560 (page 37).

[CGKM22] Sitan Chen, Aravind Gollakota, Adam Klivans, and Raghu Meka. "Hardness of noise-free learning for two-hidden-layer neural networks". In: *Advances in Neural Information Processing Systems* 35 (2022), pp. 10709–10724 (page 15).

[CT05] Richard Caron and Tim Traynor. "The zero set of a polynomial". In: *WSMR Report* (2005), pp. 05–02 (page 94).

[CW01] A. Carbery and James Wright. "Distributional and L^q norm inequalities for polynomials over convex bodies in $\mathbb{R}^n$". In: *Mathematical Research Letters* 8 (2001), pp. 233–248 (pages 86, 90).

[Das99] Sanjoy Dasgupta. "Learning mixtures of Gaussians". In: *FOCS*. FOCS '99. 1999, p. 634 (page 37).

[DDW21] Damek Davis, Mateo Diaz, and Kaizheng Wang. “Clustering a mixture of gaussians with unknown covariance”. In: *arXiv preprint arXiv:2110.01602* (2021) (page 77).

[DGR97] Scott Decatur, Oded Goldreich, and Dana Ron. “Computational sample complexity”. In: *Proceedings of the tenth annual conference on Computational learning theory*. 1997, pp. 130–142 (page 38).

[DH22] Rishabh Dudeja and Daniel Hsu. “Statistical-Computational Trade-offs in Tensor PCA and Related Problems via Communication Complexity”. In: *arXiv preprint arXiv:2204.07526* (2022) (pages 4, 36).

[Dia16] Ilias Diakonikolas. “Learning structured distributions”. In: *Handbook of Big Data*. 2016, pp. 267–284 (pages 35, 38, 45).

[DK22] Ilias Diakonikolas and Daniel Kane. “Non-gaussian component analysis via lattice basis reduction”. In: *Conference on Learning Theory*. PMLR. 2022, pp. 4535–4547 (page 8).

[DK23] Ilias Diakonikolas and Daniel M. Kane. *Algorithmic High-Dimensional Robust Statistics*. Cambridge university press Cambridge, 2023 (page 3).

[DKKZ20] Ilias Diakonikolas, Daniel M Kane, Vasilis Kontonis, and Nikos Zarifis. “Algorithms and sq lower bounds for pac learning one-hidden-layer relu networks”. In: *Conference on Learning Theory*. PMLR. 2020, pp. 1514–1539 (page 15).

[DKS17] Ilias Diakonikolas, Daniel M Kane, and Alistair Stewart. “Statistical query lower bounds for robust estimation of high-dimensional gaussians and gaussian mixtures”. In: *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*. IEEE. 2017, pp. 73–84 (pages 4, 35, 36, 38, 56).

[DKS18] Ilias Diakonikolas, Daniel M. Kane, and Alistair Stewart. “List-decodable robust mean estimation and learning mixtures of spherical Gaussians”. In: *STOC*. STOC 2018. 2018, pp. 1047–1060 (page 37).

[DMR18] Luc Devroye, Abbas Mehrabian, and Tommy Reddad. *The total variation distance between high-dimensional Gaussians*. 2018 (page 28).

[DS07] Sanjoy Dasgupta and Leonard Schulman. "A probabilistic analysis of EM for mixtures of separated, spherical Gaussians". In: *JMLR* 8 (2007), pp. 203–226 (page 37).

[DV21] Amit Daniely and Gal Vardi. "From local pseudorandom generators to hardness of learning". In: *Conference on Learning Theory*. PMLR. 2021, pp. 1358–1394 (pages 15, 38).

[Efr82] Bradley Efron. "Maximum likelihood and decision theory". In: *The Annals of Statistics* (1982), pp. 340–356 (page 1).

[Efr98] Bradley Efron. "R. A. Fisher in the 21st century". In: *Statistical Science* (1998), pp. 95–114 (page 2).

[FGRV+17] Vitaly Feldman, Elena Grigorescu, Lev Reyzin, Santosh S. Vempala, and Ying Xiao. "Statistical algorithms and a lower bound for detecting planted cliques". In: *J. ACM* 64.2 (2017) (pages 4, 38, 51).

[FKP19] Noah Fleming, Pravesh Kothari, and Toniann Pitassi. *Semialgebraic proofs and efficient algorithm design*. now the essence of knowledge, 2019 (page 79).

[Fri86] Alan M. Frieze. "On the Lagarias-Odlyzko Algorithm for the Subset Sum Problem". In: *SIAM J. Comput.* 15 (1986), pp. 536–539 (page 86).

[FT74] Jerome H Friedman and John W Tukey. "A projection pursuit algorithm for exploratory data analysis". In: *IEEE Transactions on computers* 100.9 (1974), pp. 881–890 (page 36).

[Gam21] David Gamarnik. "The overlap gap property: A topological barrier to optimizing over random structures". In: *Proceedings of the National Academy of Sciences* 118.41 (2021), e2108492118 (page 3).

[GGJK+20] Surbhi Goel, Aravind Gollakota, Zhihan Jin, Sushrut Karmalkar, and Adam Klivans. "Superpolynomial lower bounds for learning one-layer neural networks using gradient descent". In: *International Conference on Machine Learning*. PMLR. 2020, pp. 3587–3596 (page 15).

[GGM86] Oded Goldreich, Shafi Goldwasser, and Silvio Micali. "How to construct random functions". In: *Journal of the ACM (JACM)* 33.4 (1986), pp. 792–807 (page 38).

[GJJP+20] Mrinalkanti Ghosh, Fernando Granha Jeronimo, Chris Jones, Aaron Potechin, and Goutham Rajendran. "Sum-of-squares lower bounds for Sherrington-Kirkpatrick via planted affine planes". In: *2020 IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS)*. IEEE. 2020, pp. 954–965 (page 77).

[GKVZ22] Shafi Goldwasser, Michael P Kim, Vinod Vaikuntanathan, and Or Zamir. "Planting undetectable backdoors in machine learning models". In: *2022 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS)*. IEEE. 2022, pp. 931–942 (pages 8, 36).

[GKZ21] David Gamarnik, Eren C. Kızıldağ, and Ilias Zadik. "Inference in High-Dimensional Linear Regression via Lattice Basis Reduction and Integer Relation Detection". In: *IEEE Transactions on Information Theory* (2021), pp. 1–1 (page 77).

[GM84] Shafi Goldwasser and Silvio Micali. "Probabilistic encryption". In: *Journal of computer and system sciences* 28.2 (1984), pp. 270–299 (pages 36, 38).

[Gol04] Oded Goldreich. *Foundations of Cryptography*. Cambridge university press Cambridge, 2004 (pages 9, 10).

[Gol08] Oded Goldreich. "Computational complexity: a conceptual perspective". In: *ACM Sigact News* 39.3 (2008), pp. 35–39 (page 8).

[GPV08] Craig Gentry, Chris Peikert, and Vinod Vaikuntanathan. “Trapdoors for hard lattices and new cryptographic constructions”. In: *Proceedings of the fortieth annual ACM symposium on Theory of computing*. 2008, pp. 197–206 (page 12).

[GVV22] Aparna Gupte, Neekon Vafa, and Vinod Vaikuntanathan. “Continuous LWE is as Hard as LWE & Applications to Learning Gaussian Mixtures”. In: *arXiv preprint arXiv:2204.02550* (2022) (pages 7, 39).

[GZ17] David Gamarnik and Ilias Zadik. “High dimensional linear regression with binary coefficients: Mean squared error and a phase transition”. In: *Conference on Learning Theory (COLT)*. 2017 (page 15).

[Hal13] Paul Richard Halmos. *I want to be a mathematician: An automathography*. Springer Science & Business Media, 2013 (page 9).

[HKPR+17] Samuel B Hopkins, Pravesh K Kothari, Aaron Potechin, Prasad Raghavendra, Tselil Schramm, and David Steurer. “The power of sum-of-squares for detecting hidden structures”. In: *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*. IEEE. 2017, pp. 720–731 (pages 4, 79).

[HL18] Samuel B. Hopkins and Jerry Li. “Mixture models, robustness, and sum of squares proofs”. In: *STOC*. STOC 2018. 2018, pp. 1021–1034 (page 37).

[Hop18] Samuel Hopkins. “Statistical inference and the sum of squares method”. PhD thesis. Cornell University, 2018 (pages 3, 4, 39, 79, 80).

[HS17] Samuel B Hopkins and David Steurer. “Efficient bayesian estimation from few samples: community detection and related problems”. In: *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*. IEEE. 2017, pp. 379–390 (page 79).

[Hub85] Peter J Huber. “Projection pursuit”. In: *The annals of Statistics* (1985), pp. 435–475 (page 36).

[HW20] Justin Holmgren and Alexander S Wein. “Counterexamples to the low-degree conjecture”. In: *arXiv preprint arXiv:2004.08454* (2020) (page 81).

[HWX15] Bruce Hajek, Yihong Wu, and Jiaming Xu. “Computational lower bounds for community detection on random graphs”. In: *Conference on Learning Theory*. 2015, pp. 899–928 (page 78).

[Jer92] Mark Jerrum. “Large cliques elude the Metropolis process”. In: *Random Structures & Algorithms* 3.4 (1992), pp. 347–359 (page 81).

[JS87] M Chris Jones and Robin Sibson. “What is projection pursuit?” In: *Journal of the Royal Statistical Society: Series A (General)* 150.1 (1987), pp. 1–18 (page 36).

[Kac87] Mark Kac. *Enigmas of chance: an autobiography*. Univ of California Press, 1987 (page 35).

[KB21] Dmitriy Kunisky and Afonso S Bandeira. “A tight degree 4 sum-of-squares lower bound for the Sherrington–Kirkpatrick Hamiltonian”. In: *Mathematical Programming* 190.1 (2021), pp. 721–759 (page 77).

[Kea98] Michael Kearns. “Efficient noise-tolerant learning from statistical queries”. In: *J. ACM* 45.6 (1998), pp. 983–1006 (pages 4, 38, 50).

[Kha93] Michael Kharitonov. “Cryptographic hardness of distribution-specific learning”. In: *Proceedings of the twenty-fifth annual ACM symposium on Theory of computing*. 1993, pp. 372–381 (page 38).

[KL20] Jonathan Katz and Yehuda Lindell. *Introduction to modern cryptography*. CRC press, 2020 (pages 3, 9, 10).

[KRAU+20] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, Tom Duerig, and Vittorio Ferrari. “The Open Images Dataset V4: Unified image

classification, object detection, and visual relationship detection at scale". In: *IJCV* (2020) (page 2).

[KS09] Adam R Klivans and Alexander A Sherstov. "Cryptographic hardness for learning intersections of halfspaces". In: *Journal of Computer and System Sciences* 75.1 (2009), pp. 2–12 (page 38).

[KSS18] Pravesh K. Kothari, Jacob Steinhardt, and David Steurer. "Robust moment estimation and improved clustering via sum of squares". In: *STOC*. STOC 2018. 2018, pp. 1035–1046 (page 37).

[Kuč95] Luděk Kučera. "Expected complexity of graph partitioning problems". In: *Discrete Applied Mathematics* 57.2-3 (1995), pp. 193–212 (page 81).

[Kun20] Dmitriy Kunisky. "Positivity-preserving extensions of sum-of-squares pseudomoments over the hypercube". In: *arXiv preprint arXiv:2009.07269* (2020) (page 77).

[Kun22] Dmitriy Kunisky. "Spectral Barriers in Certification Problems". PhD thesis. New York University, 2022 (pages 3, 56).

[KV94] Michael Kearns and Leslie Valiant. "Cryptographic limitations on learning boolean formulae and finite automata". In: *Journal of the ACM (JACM)* 41.1 (1994), pp. 67–95 (page 38).

[KWB22] Dmitriy Kunisky, Alexander S Wein, and Afonso S Bandeira. "Notes on computational hardness of hypothesis testing: Predictions using the low-degree likelihood ratio". In: *Mathematical Analysis, its Applications and Computation: ISAAC 2019, Aveiro, Portugal, July 29–August 2*. Springer, 2022, pp. 1–50 (pages 3, 4, 39, 56, 79).

[Lag84] Jeffrey C Lagarias. "Knapsack public key cryptosystems and diophantine approximation". In: *Advances in cryptology*. Springer. 1984, pp. 3–23 (page 86).

[Las01] Jean B Lasserre. "Global optimization with polynomials and the problem of moments". In: *SIAM Journal on optimization* 11.3 (2001), pp. 796–817 (page 79).

[LeC12] Lucien LeCam. *Asymptotic methods in statistical decision theory*. Springer Science & Business Media, 2012 (pages 1, 55).

[Leh11] Erich L Lehmann. *Fisher, Neyman, and the creation of classical statistics*. Springer Science & Business Media, 2011 (page 1).

[LLL82a] A. K. Lenstra, H. W. Lenstra, and L. Lovász. "Factoring polynomials with rational coefficients". en. In: *Mathematische Annalen* 261.4 (1982), pp. 515–534 (page 28).

[LLL82b] Arjen Klaas Lenstra, Hendrik Willem Lenstra, and László Lovász. "Factoring polynomials with rational coefficients". In: *Mathematische Annalen* 261.4 (1982), pp. 515–534 (pages 77, 82).

[LM09] Vadim Lyubashevsky and Daniele Micciancio. "On bounded distance decoding, unique shortest vectors, and the minimum distance problem". In: *CRYPTO*. CRYPTO '09. 2009, pp. 577–594 (page 28).

[LRC05] Erich L Lehmann, Joseph P Romano, and George Casella. *Testing statistical hypotheses*. Vol. 3. Springer, 2005 (page 1).

[MAB20] Antoine Maillard, Gérard Ben Arous, and Giulio Biroli. "Landscape complexity for the empirical risk of generalized linear models". In: *Mathematical and Scientific Machine Learning*. PMLR. 2020, pp. 287–327 (page 15).

[MNV16] Raghu Meka, Oanh Nguyen, and Van Vu. "Anti-concentration for Polynomials of Independent Random Variables". In: *Theory of Computing* 12.11 (2016), pp. 1–17 (page 90).

[Moi18] Ankur Moitra. *Algorithmic aspects of machine learning*. Cambridge University Press, 2018 (pages 35, 45).

[MP12] Daniele Micciancio and Chris Peikert. “Trapdoors for Lattices: Simpler, Tighter, Faster, Smaller.” In: *Eurocrypt*. Vol. 7237. Springer. 2012, pp. 700–718 (pages 13, 48, 75).

[MR07] Daniele Micciancio and Oded Regev. “Worst-case to average-case reductions based on Gaussian measures”. In: *SIAM J. Comput.* 37.1 (2007), pp. 267–302 (page 20).

[MR09] Daniele Micciancio and Oded Regev. “Lattice-based cryptography”. In: *Post-quantum cryptography*. Springer, 2009, pp. 147–191 (pages 4, 14, 16, 78).

[MRX20] Sidhanth Mohanty, Prasad Raghavendra, and Jeff Xu. “Lifting sum-of-squares lower bounds: degree-2 to degree-4”. In: *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing*. 2020, pp. 840–853 (page 77).

[MS16] Andrea Montanari and Subhabrata Sen. “Semidefinite programs on sparse random graphs and their application to community detection”. In: *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing*. 2016, pp. 814–827 (page 77).

[MV10] Ankur Moitra and Gregory Valiant. “Settling the polynomial learnability of mixtures of Gaussians”. In: *FOCS*. FOCS ’10. 2010, pp. 93–102 (pages 16, 37).

[MW15] Zongming Ma and Yihong Wu. “Computational barriers in minimax submatrix detection”. In: *The Annals of Statistics* 43.3 (2015) (pages 3, 4, 78).

[MW21] Cheng Mao and Alexander S Wein. “Optimal spectral recovery of a planted vector in a subspace”. In: *arXiv preprint arXiv:2105.15081* (2021) (pages 4, 57, 77, 80).

[MW22] Andrea Montanari and Alexander S Wein. “Equivalence of Approximate Message Passing and Low-Degree Polynomials in Rank-One Matrix Estimation”. In: *arXiv preprint arXiv:2212.06996* (2022) (page 56).

[NR22] Jonathan Niles-Weed and Philippe Rigollet. “Estimation of Wasserstein distances in the Spiked Transport Model”. In: *Bernoulli* 28.4 (2022), pp. 2663–2688 (page 36).

[NW72] John Ashworth Nelder and Robert WM Wedderburn. “Generalized linear models”. In: *Journal of the Royal Statistical Society: Series A (General)* 135.3 (1972), pp. 370–384 (page 15).

[Par00] Pablo A Parrilo. *Structured semidefinite programs and semialgebraic geometry methods in robustness and optimization*. California Institute of Technology, 2000 (page 79).

[Pea94] Karl Pearson. “Contributions to the mathematical theory of evolution”. In: *Philosophical Transactions of the Royal Society of London. A* 185 (1894) (page 35).

[Pei10] Chris Peikert. “An efficient and parallel Gaussian sampler for lattices”. In: *CRYPTO*. 2010, pp. 80–97 (page 12).

[Pei16] Chris Peikert. “A decade of lattice cryptography”. In: *Foundations and Trends in Theoretical Computer Science* 10.4 (2016), pp. 283–424 (pages 14, 16).

[Pol04] George Polya. *How to solve it: A new aspect of mathematical method*. 246. Princeton university press, 2004 (page 76).

[PRS17] Chris Peikert, Oded Regev, and Noah Stephens-Davidowitz. “Pseudorandomness of ring-LWE for any ring and modulus”. In: *STOC*. STOC 2017. 2017, pp. 461–473 (pages 17–20, 23, 26, 27).

[Reg05] Oded Regev. “On lattices, learning with errors, random linear codes, and cryptography”. In: *STOC*. STOC ’05. 2005, pp. 84–93 (pages 7, 14, 16, 17, 20, 22, 23).

[Reg09] Oded Regev. “On lattices, learning with errors, random linear codes, and cryptography”. In: *Journal of the ACM (JACM)* 56.6 (2009), pp. 1–40 (pages 13, 45).

[RM14] Emile Richard and Andrea Montanari. “A statistical model for tensor PCA”. In: *Advances in neural information processing systems* 27 (2014) (page 4).

[RS09] Ronitt Rubinfeld and Rocco A. Servedio. “Testing monotone high-dimensional distributions”. In: *Random Structures & Algorithms* 34.1 (2009), pp. 24–44 (page 46).

[RSS18] Prasad Raghavendra, Tselil Schramm, and David Steurer. “High dimensional estimation via sum-of-squares proofs”. In: *Proceedings of the International Congress of Mathematicians: Rio de Janeiro 2018*. World Scientific. 2018, pp. 3389–3423 (page 79).

[RSW23] Oded Regev, Min Jae Song, and Alexander S. Wein. “On the Hardness of Homogeneous CLWE”. In: *In progress* (2023) (page 6).

[RV17] O. Regev and A. Vijayaraghavan. “On learning mixtures of well-separated Gaussians”. In: *FOCS*. 2017, pp. 85–96 (page 37).

[Ser99] Rocco A Servedio. “Computational sample complexity and attribute-efficient learning”. In: *Proceedings of the thirty-first annual ACM symposium on Theory of Computing*. 1999, pp. 701–710 (page 38).

[Sha18] Ohad Shamir. “Distribution-specific hardness of learning neural networks”. In: *The Journal of Machine Learning Research* 19.1 (2018), pp. 1135–1163 (pages 15, 32).

[SSS17] Shai Shalev-Shwartz, Ohad Shamir, and Shaked Shammah. “Failures of gradient-based deep learning”. In: *International Conference on Machine Learning*. PMLR. 2017, pp. 3067–3075 (page 32).

[SST12] Shai Shalev-Shwartz, Ohad Shamir, and Eran Tromer. “Using more data to speed-up training time”. In: *Artificial Intelligence and Statistics*. PMLR. 2012, pp. 1019–1027 (page 38).

[SVWX17] Le Song, Santosh Vempala, John Wilmes, and Bo Xie. “On the Complexity of Learning Neural Networks”. In: *Advances in Neural Information Processing Systems*. Ed. by I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett. Vol. 30. Curran Associates, Inc., 2017 (pages 15, 32).

[SW22] Tselil Schramm and Alexander S Wein. “Computational barriers to estimation from low-degree polynomials”. In: *The Annals of Statistics* 50.3 (2022), pp. 1833–1858 (page 56).

[SZB21] Min Jae Song, Ilias Zadik, and Joan Bruna. “On the Cryptographic Hardness of Learning Single Periodic Neurons”. In: *NeurIPS* (2021) (pages 6, 8, 76, 77, 86, 88).

[Ver18] R. Vershynin. *High-dimensional probability: an introduction with applications in data science*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, 2018 (pages 74, 75).

[VW02] Santosh Vempala and Grant Wang. “A spectral algorithm for learning mixtures of distributions”. In: *FOCS*. FOCS ’02. 2002, p. 113 (page 37).

[Wai14] Martin J Wainwright. “Constrained forms of statistical minimax: Computation, communication and privacy”. In: *Proceedings of the International Congress of Mathematicians*. 2014 (page 2).

[Wai19] Martin J Wainwright. *High-dimensional statistics: A non-asymptotic viewpoint*. Cambridge university press, 2019 (page 2).

[Wal50] Abraham Wald. *Statistical decision functions*. Wiley, 1950 (page 1).

[WX21] Yihong Wu and Jiaming Xu. “Statistical problems with planted structures: Information theoretical and computational limits”. In: *Information-Theoretic Methods in Data Science* 383 (2021), p. 13 (page 3).

[ZG18] Ilias Zadik and David Gamarnik. “High Dimensional Linear Regression using Lattice Basis Reduction”. In: *Advances in Neural Information Processing Systems*. Vol. 31. Curran Associates, Inc., 2018 (pages 76–78, 86).

[ZK16] Lenka Zdeborová and Florent Krzakala. “Statistical physics of inference: Thresholds and algorithms”. In: *Advances in Physics* 65.5 (2016), pp. 453–552 (page 3).

[ZSWB22] Ilias Zadik, Min Jae Song, Alexander S Wein, and Joan Bruna. "Lattice-based methods surpass sum-of-squares in clustering". In: *Conference on Learning Theory*. PMLR. 2022, pp. 1247–1248 (pages 6, 8, 77).

[ZWJ14] Yuchen Zhang, Martin J Wainwright, and Michael I Jordan. "Lower bounds on the performance of polynomial-time algorithms for sparse linear regression". In: *Conference on Learning Theory*. PMLR. 2014, pp. 921–948 (page 3).