Estimating the Spectral Moments of the Kernel Integral Operator from Finite Sample Matrices

Chanwoo Chun[1,4] **SueYeon Chung**[2,4] **Daniel D. Lee** [3,4]
[1] Weill Cornell Medical College [2] New York University [3] Cornell Tech [4] Flatiron Institute

Abstract

Analyzing the structure of sampled features from an input data distribution is challenging when constrained by limited measurements in both the number of inputs and features. Traditional approaches often rely on the eigenvalue spectrum of the sample covariance matrix derived from finite measurement matrices; however, these spectra are sensitive to the size of the measurement matrix, leading to biased insights. In this paper, we introduce a novel algorithm that provides unbiased estimates of the spectral moments of the kernel integral operator in the limit of infinite inputs and features from finitely sampled measurement matrices. Our method, based on dynamic programming, is efficient and capable of estimating the moments of the operator spectrum. We demonstrate the accuracy of our estimator on radial basis function (RBF) kernels, highlighting its consistency with the theoretical spectra. Furthermore, we showcase the practical utility and robustness of our method in understanding the geometry of learned representations in neural networks.

1 INTRODUCTION

A primary objective of statistical inference in machine learning is to accurately estimate the characteristics of a high-dimensional distribution based on finite samples. For example, consider a Gaussian process with an unknown covariance where only a limited set of sample functions is drawn from the process. In many cases, we cannot observe the functions themselves, but rather their noisy evaluations at sampled input points (Williams and Rasmussen, 2006). What can we infer about the underlying process from such a sampled set of functions and input points? Similarly, in large-scale neural networks, we aim to understand the characteristics of the neural feature representations as both the number of features and input points grow infinitely large (Cho and Saul, 2009; Mei et al., 2018; Chung et al., 2018; Cohen et al., 2020; Canatar et al., 2021, 2024). The central question of this work is how to estimate the spectral properties of the underlying infinite process when both features and evaluation points are finitely sampled.

A matrix can be constructed from sampled measurements of the process, where each row corresponds to an individual input sample and each column is a sampled feature. It is common practice to analyze the eigenvalue spectrum of the sample covariance matrix derived from this finite measurement matrix. However, this spectrum is biased, leading to inaccurate insights into the underlying structure. Therefore, prior work aims to correct this bias under the assumption that the rows are sampled (Kong and Valiant, 2017). However, in the setup where both rows and columns are sampled, this method produces biased estimates.

In our model, we consider the measurement matrix as a sampled submatrix of a larger underlying matrix where both the number of rows and columns approach infinity. A kernel integral operator can be defined as the expected covariance of this larger matrix and we study how to accurately infer the spectral properties of this operator, in particular its spectral moments. We propose a novel, computationally efficient algorithm based upon dynamic programming, to estimate the spectral moments of the underlying kernel operator from a finite measurement matrix.

In the following, we first describe a mathematical framework relating a finite measurement matrix to the spectrum of a kernel integral operator (Bach, 2017). We show how the naive estimators of the spectral mo-

Proceedings of the 28[th] International Conference on Artificial Intelligence and Statistics (AISTATS) 2025, Mai Khao, Thailand. PMLR: Volume 258.

ments based upon the finite covariance matrix are biased. Then we introduce our method for estimating the spectral moments by averaging appropriate products of non-repeating cycles in the measurement matrix. Our method employs a recursive procedure that is computationally efficient, polynomial in the size of the matrix and the order of the moments. We demonstrate the accuracy of our method with the radial basis function (RBF) kernel operator, where a direct comparison to the theoretical spectrum and to other estimation methods is possible. We also demonstrate inferring the eigenvalues of kernel integral operators from our moment estimates using an existing algorithm. Finally, we show how our estimates can be used to analyze the learning dynamics of a rectified linear unit (ReLU) neural network during feature learning, showing how networks of different widths can be related by their kernel operators.

2 KERNEL OPERATOR

2.1 Kernel as expectation

We model the entries of a $P \times Q$ measurement matrix $[\Phi_{i\alpha}]$ as arising from the following stochastic process. Each row $i \in \{1, ..., P\}$ is characterized by a latent input variable x_i drawn independently from a probability measure $\rho_{\mathcal{X}}(x)$ over a latent space $\mathcal{X}$, and each column $\alpha \in \{1, ..., Q\}$ is characterized by a latent variable w_α drawn independently from a probability measure $\rho_{\mathcal{W}}(w)$ over a latent space $\mathcal{W}$. In random feature networks, for instance, x_i and w_α can be seen as an input pattern and neural weights respectively. The $(i\alpha)$-th entry of the matrix $[\Phi_{i\alpha}]$ is produced by a function ϕ that maps the pair (x_i, w_α) to a real number:

$$\Phi_{i\alpha} = \phi(x_i, w_\alpha). \tag{1}$$

A kernel function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ over input pairs can then be defined as the expected value of the product of ϕ over the features (Bach, 2017):

$$k(x, x') = \int d\rho_{\mathcal{W}}(w)\, \phi(x, w)\phi(x', w). \tag{2}$$

We assume the function ϕ is square-integrable with respect to both $\rho_{\mathcal{X}}$ and $\rho_{\mathcal{W}}$, so the kernel function is positive-definite and bounded. The tuple $(\phi, \rho_{\mathcal{X}}, \rho_{\mathcal{W}})$ uniquely defines a generative process for the measurement matrices and corresponding kernel.

2.2 Kernel integral operator

Now consider the integral operator $T_k : \mathcal{L}^2(\mathcal{X}, \rho_{\mathcal{X}}) \to \mathcal{L}^2(\mathcal{X}, \rho_{\mathcal{X}})$ (Cucker and Smale, 2002):

$$T_k f \coloneqq \int d\rho_{\mathcal{X}}(x)\, k(\cdot, x) f(x). \tag{3}$$

Since ϕ is square-integrable, T_k is a trace class operator. Therefore T_k is a compact, bounded, and self-adjoint linear operator, whose spectrum consists of a countable, non-increasing sequence of eigenvalues $\{\lambda_l \geq 0\}_{l=1}^{\infty}$. The corresponding eigenfunctions are defined implicitly as $\lambda_l e_l = T_k e_l$ where $e_l \in \mathcal{L}^2(\mathcal{X}, \rho_{\mathcal{X}})$ and $\{e_l\}_{l=1}^{\infty}$ forms an orthonormal set. Since the product of a bounded operator and trace class operator is also trace class, $\mathrm{tr}\, T_k^n$ is well-defined for any positive integer n. We define the spectral n-th moment $m(n)$ as the sum over the n-th powers of the eigenvalues:

$$m(n) \coloneqq \sum_{l=1}^{\infty} \lambda_l^n \equiv \mathrm{tr} T_k^n. \tag{4}$$

The moments can also be written as the expectation over a product of kernel functions,

$$m(n) = \int \prod_{j=1}^{n} d\rho_{\mathcal{X}}(x_j) \prod_{j=1}^{n} k(x_j, x_{j+1}) \tag{5}$$

with the constraint that $x_{n+1} = x_1$. The moments $\{m(n)\}_{n=1}^{\infty}$ uniquely determine the spectrum of T_k as well as the kernel covariance operator via the Stieltjes transform (see Appendix). Various methods to estimate the spectral moments $m(n)$ of T_k from a finite measurement matrix $[\Phi_{i\alpha}]$ are investigated in this work.

2.3 Naive estimator

Although the rows and columns of the measurement matrix Φ are sampled independently, the matrix coefficients will be correlated due to similarities in the sampled inputs and features. The conventional approach to analyze the spectral structure of the measurement matrix is to form the Gram matrix $K \in \mathbb{R}^{P \times P}$ that represents the similarity between input rows:

$$K_{ij} = \frac{1}{Q} \sum_{\alpha=1}^{Q} \phi(x_i, w_\alpha)\phi(x_j, w_\alpha) = \frac{1}{Q} \sum_{\alpha=1}^{Q} \Phi_{i\alpha}\Phi_{j\alpha}. \tag{6}$$

The matrix K is positive semi-definite and its moments are given by $\mathrm{tr} K^n$. If the similarity between different inputs x_i and x_j is $\mathcal{O}(1)$, then we expect the trace $\mathrm{tr} K^n$ to scale as $\mathcal{O}(P^n)$. We normalize the traces by this scaling factor to give the naive estimator $\hat{m}_0(n)$:

$$\hat{m}_0(n) = \mathrm{tr}\left[\left(\frac{K}{P}\right)^n\right]. \tag{7}$$

In the limit of large P and Q, the naive estimates will converge to the moments of the kernel integral operator, $\hat{m}_0(n) \to m(n)$. For $n = 1$, $\hat{m}_0(1)$ is the sample variance of $\phi(x, w)$ and is an unbiased estimate

of $m(1)$. However, for $n > 1$ and finite P and Q, $\hat{m}_0(n)$ is a biased estimate of $m(n)$. To understand why $\hat{m}_0(n)$ is biased, consider the expected value of the second moment estimate:

$$\begin{aligned}\langle \hat{m}_0(2) \rangle_\Phi &= \frac{1}{P^2}\frac{1}{Q^2}\sum_{i,j}^{P}\sum_{\alpha,\beta}^{Q}\langle \Phi_{i\alpha}\Phi_{j\alpha}\Phi_{j\beta}\Phi_{i\beta}\rangle \\ &= \frac{1}{P^2Q^2}\sum_{i\neq j}^{P}\sum_{\alpha\neq\beta}^{Q}\langle \Phi_{i\alpha}\Phi_{j\alpha}\Phi_{j\beta}\Phi_{i\beta}\rangle \\ &+ \frac{1}{P^2Q^2}\sum_{i}^{P}\sum_{\alpha\neq\beta}^{Q}\langle \Phi_{i\alpha}^2\Phi_{i\beta}^2\rangle + \frac{1}{P^2Q^2}\sum_{i\neq j}^{P}\sum_{\alpha}^{Q}\langle \Phi_{i\alpha}^2\Phi_{j\alpha}^2\rangle \\ &+ \frac{1}{P^2Q^2}\sum_{i}^{P}\sum_{\alpha}^{Q}\langle \Phi_{i\alpha}^4\rangle . \quad (8)\end{aligned}$$

The second term in this expansion contains the connected product $\left\langle \Phi_{i\alpha}^2\Phi_{i\beta}^2 \right\rangle$ whose expected value is $\int d\rho_{\mathcal{X}}(x)\, k(x,x)^2$, and differs from the second moment of the kernel integral operator. This term is order $\mathcal{O}\left(\frac{1}{P}\right)$ and introduces a finite sampling bias into the estimate $\hat{m}_0(2)$. Similarly, the third and fourth terms in the expansion will give rise to bias terms of order $\mathcal{O}\left(\frac{1}{Q}\right)$ and $\mathcal{O}\left(\frac{1}{PQ}\right)$ respectively.

This analysis generalizes to all higher moments for $n > 1$. The naive estimate $\hat{m}_0(n)$ derived from the sample Gram matrix contains biased terms of order $\mathcal{O}\left(\frac{1}{P}+\frac{1}{Q}\right)$.

3 RELATED WORK

3.1 Random matrix theory

Random matrix theory analyzes the spectral characteristics of ensembles of Wishart matrices. The basic theory considers a Wishart matrix formed by taking the covariance of a large $P \times Q$ random matrix $[\Phi_{i\alpha}]$, whose entries are independently sampled from the standard normal distribution, i.e. $\Phi_{i\alpha} \sim \mathcal{N}(0,1)$. The spectrum of the Wishart matrix converges to a well-defined limit as $P, Q \to \infty$ when the ratio $\frac{P}{Q}$ is fixed. This limiting spectral distribution differs from that of an identity covariance matrix and is known as the Marchenko-Pastur distribution (Marchenko and Pastur, 1967).

Within our framework, a measurement matrix with independent and identically distributed (i.i.d.) normal entries is generated by taking $x, w \in \mathbb{R}^d$ with $\rho_{\mathcal{X}} = \mathcal{N}(0, I_{d\times d})$, $\rho_{\mathcal{W}} = \mathcal{N}\left(0, \frac{1}{d} I_{d\times d}\right)$ and bilinear map $\phi(x,w) = x^\top w$. When d approaches infinity and is much larger than P and Q, each element $\Phi_{i\alpha}$ becomes an independent standard normal random variable.

In this case, the moments of the kernel integral operator are $m(n) = d^{-(n-1)}$. For large $d \gg P, Q$, the spectrum of the sample Gram matrix and the corresponding naive spectral moment estimates will be dominated by bias terms, and the leading order fully-connected bias terms are the same terms that give rise to the Marchenko-Pastur distribution. In the next section, we will see how to better estimate the spectral moments from measurement matrices by eliminating the bias terms.

3.2 Estimator for fully observed features

Kong and Valiant (2017) consider the problem where the inputs are sampled from an underlying distribution but the features are fully observed with finite cardinality d, e.g. $w_\alpha \in \{w_1, w_2, ..., w_d\}$. In their scenario, the measurements can be modeled as a $P \times d$ matrix $\bar{\Phi} \in \mathbb{R}^{P\times d}$; other work in the spectrum estimation literature also considers similar problem setup (Ledoit and Wolf, 2004; Burda et al., 2004; El Karoui, 2008; Khorunzhiy, 2008; Bhattacharjee et al., 2024). Kong and Valiant (2017) models the observed measurement matrix as a matrix sketching process: $\bar{\Phi}_{i\alpha} = \sum_{k=1}^{d} x_{ik} F_{k\alpha}$ where the coefficients x_{ik} are independently sampled from the standard normal distribution, and $F \in \mathbb{R}^{d\times d}$ is a deterministic matrix. Their method seeks to estimate the spectral moments of $S \coloneqq \frac{1}{d} F F^\top$, i.e. $m_{\mathrm{KV}}(n) = \mathrm{tr}\,(S^n)$ in order to obtain the spectrum of S.

After noting that the naive estimator based upon the Gram matrix $\bar{K} \coloneqq \frac{1}{d}\bar{\Phi}\bar{\Phi}^\top$ with spectral moments $\hat{m}_0(n) = \mathrm{tr}(\hat{K}^n)$ is biased, the following simple unbiased estimator is proposed:

$$\hat{m}'_{\mathrm{KV}}(n) = \prod_{l=1}^{n} \bar{K}_{i_l i_{l+1}} \quad (9)$$

where the product indices $i_l \in \{1, \ldots, P\}$ are disjoint, $i_l \neq i_k$ for all $l \neq k$, except for the trace constraint $i_1 = i_{n+1}$. The proof that this simple estimator is unbiased relies on the assumption that $\bar{\Phi}_{i\alpha}$ is zero-mean, which is not a requirement for our model. With only a single realization of $\{i_l\}_{l=1}^n$, $\hat{m}'_{\mathrm{KV}}(n)$, the variance of the estimate is high. It would be optimal to average over all possible realizations of $\{i_l\}_{l=1}^n$, but there is no known computationally efficient algorithm to perform the sum for large n. Thus, the authors propose averaging over sets of increasing indices, i.e. $i_1 < i_2 < \ldots < i_n$. This leads to their estimator which considers the trace of the following matrix product:

$$\hat{m}_{\mathrm{KV}}(n) = \frac{\mathrm{tr}\left(\bar{K}_{\mathrm{up}}^{n-1}\bar{K}\right)}{\binom{P}{n}}. \quad (10)$$

$\bar{K}_{\text{up}}$ is the upper triangular matrix formed from $\bar{K}$, e.g. the diagonal and lower triangular entries are set to zero.

The estimator $\hat{m}_{\text{KV}}(n)$ can be used with the measurement matrix Φ in our problem setup in two ways. One is setting $\bar{\Phi} \leftarrow \Phi$ which is equivalent to treating Φ as if all features are observed. The other is to set $\bar{\Phi} \leftarrow \Phi^\top$, which is equivalent to the assumption that all the inputs are observed but the features are sampled. We refer to the former estimator as $\hat{m}_{\text{KV-row}}(n)$ and the latter as $\hat{m}_{\text{KV-col}}(n)$. When both inputs and features are not fully observed, we will see that both of these approaches result in biased estimates.

4 UNBIASED ESTIMATION OF SPECTRAL MOMENTS AND EIGENVALUE RECOVERY

4.1 Unbiased estimator

We saw that for estimating the second moment, $\hat{m}_0(2)$, only the first term in (8) consisting of products of matrix coefficients with disjoint indices is unbiased. This observation can be generalized to higher spectral moments of the kernel integral operator. An elementary unbiased estimator for $m(n)$ is given by

$$\hat{m}'(n) = \prod_{l=1}^{n} \Phi_{i_l \alpha_l} \Phi_{i_{l+1} \alpha_l} \tag{11}$$

where the indices $i_l \in \{1, \ldots, P\}$ are disjoint, $i_l \neq i_k$ for all $l \neq k$, except for the trace constraint $i_1 = i_{n+1}$. Similarly, the feature indices $\alpha_l \in \{1, \ldots, Q\}$ should also be disjoint, $\alpha_l \neq \alpha_k$ for all $l \neq k$. Since there is no overlap in the indices, the expected value of $\hat{m}'(n)$ is the product of the expected values of the kernel functions:

$$\left\langle \prod_{l=1}^{n} \Phi_{i_l \alpha_l} \Phi_{i_{l+1} \alpha_l} \right\rangle_{\Phi} = \prod_{l=1}^{n} \left\langle \Phi_{i_l \alpha_l} \Phi_{i_{l+1} \alpha_l} \right\rangle_{\Phi} \tag{12}$$

$$= \prod_{i=1}^{n} \left\langle k(x_i, x_{i+1}) \right\rangle_{x_i, x_{i+1}}. \tag{13}$$

This is equivalent to the definition of $m(n)$, showing that $\hat{m}'(n)$ is an unbiased estimator of $m(n)$.

As shown in Figure 1, one can view the product in $\hat{m}'(n)$ as forming a cyclic path over the coefficients of matrix Φ by first choosing a starting coefficient Φ_{i_1,α_1}, multiplying it with a distinct coefficient Φ_{i_2,α_1} on the same column, then multiplying with another coefficient Φ_{i_2,α_2} on the same row as the previous one, and so on until returning to the starting coefficient, creating a product with a total of $2n$ distinct coefficients.

In order to reduce the variance of this estimate, we can consider averaging $\prod_{l=1}^{n} \Phi_{i_l \alpha_l} \Phi_{i_{l+1} \alpha_l}$ over all possible cyclic paths $\{i_l\}_{l=1}^{n}$ and $\{\alpha_l\}_{l=1}^{n}$ of non-overlapping indices:

$$\hat{m}^*(n) = \frac{\sum_{\substack{\alpha_1 \neq \ldots \neq \alpha_n \\ i_1 \neq \ldots \neq i_n}} \prod_{l=1}^{n} \Phi_{i_l \alpha_l} \Phi_{i_{l+1} \alpha_l}}{\prod_{i=0}^{n-1} (P-i)(Q-i)}, \tag{14}$$

with the constraint that the final input index is the same as the initial input index $i_{n+1} = i_1$.

This sum can be performed for small moments n. For example, with $n = 2$, we have the straightforward expression:

$$c\hat{m}^*(2) = \hat{m}_0(2) - \sum_{i=1}^{P} \frac{K_{ii}^2}{P^2} - \sum_{\alpha=1}^{Q} \frac{\tilde{K}_{\alpha\alpha}^2}{Q^2} + \sum_{i=1}^{P} \sum_{\alpha=1}^{Q} \frac{\Phi_{i\alpha}^4}{P^2 Q^2} \tag{15}$$

where $c \coloneqq \frac{(P-1)(Q-1)}{PQ}$, $K_{ij} = \frac{1}{Q} \sum_{\alpha=1}^{Q} \Phi_{i\alpha} \Phi_{j\alpha}$ and $\tilde{K}_{\alpha\beta} = \frac{1}{P} \sum_{i=1}^{P} \Phi_{i\alpha} \Phi_{i\beta}$.

Unfortunately, summing over all disjoint index sets is computationally inefficient for larger n, with apparent complexity $\mathcal{O}(P^n Q^n)$. Instead, similar to the approach by Kong and Valiant (2017), we can average over cyclic paths where both sets of indices are increasing, i.e. $1 \leq i_1 < i_2 < \ldots < i_n \leq P$ and $1 \leq \alpha_1 < \alpha_2 < \ldots < \alpha_n \leq Q$ (see Figure 1b):

$$\hat{m}(n) = \frac{1}{\binom{P}{n}\binom{Q}{n}} \sum_{\substack{1 \leq i_1 < i_2 < \ldots < i_n \leq P \\ 1 \leq \alpha_1 < \alpha_2 < \ldots < \alpha_n \leq Q}} \prod_{l=1}^{n} \Phi_{i_l \alpha_l} \Phi_{i_{l+1} \alpha_l} \tag{16}$$

where $i_{n+1} = i_1$, and the combinatorial product $\binom{P}{n}\binom{Q}{n}$ is the total number of terms in the sum. Graphically, the paths over increasing index sets create stair-like cyclic paths in the matrix Φ as shown in Figure 1c.

4.2 Dynamic programming algorithm

Here we show how to efficiently compute $\hat{m}(n)$ for higher moments via a recursive dynamic programming algorithm. The estimate $\hat{m}(n)$ can be written in terms of the partial sums:

$$\hat{m}(n) = \frac{1}{\binom{P}{n}\binom{Q}{n}} \sum_{h=1}^{P-n+1} \sum_{j=n}^{Q} \sum_{i=h+n-1}^{P} S_{ij}^{(h)}[n] \Phi_{hj}. \tag{17}$$

where the sums are defined as:

$$S_{ab}^{(h)}[n] = \sum_{\substack{h < i_2 < \ldots < i_{n-1} < a \\ 1 \leq \alpha_1 < \ldots < \alpha_{n-1} < b}} \left(\prod_{l=1}^{n-1} \Phi_{i_l \alpha_l} \Phi_{i_{l+1} \alpha_l} \right) \Phi_{i_n \alpha_n} \tag{18}$$

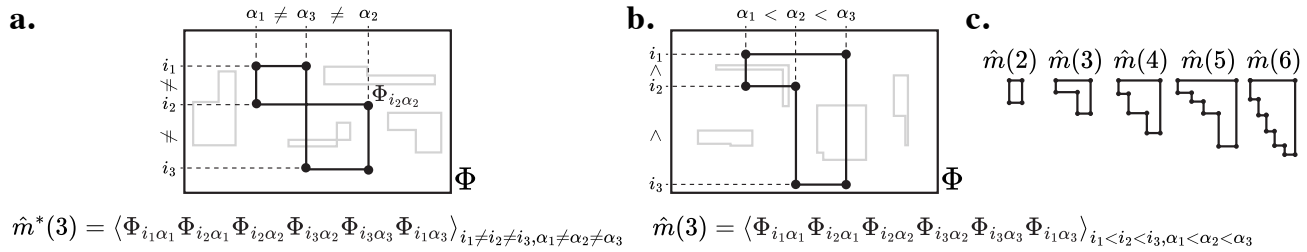

Figure 1: Visual illustration of the calculation of the unbiased estimator. **a.** For computing $\hat{m}^*(3)$, one can select matrix entries such that the entries create a cyclic path of 6 turns without revisiting rows and columns more than twice, and average over all possible such paths. **b.** Our method limits the cyclic paths to only increasing indices. **c.** Example paths for $\hat{m}(2), \ldots, \hat{m}(6)$.

Algorithm 1 Computation of $\hat{m}(n)$ for $n = 2$ to $n_{\max}$

Require: $\Phi \in \mathbb{R}^{P\times Q}$, $n_{\max}$

1: **for** $h \leftarrow 1$ **to** P **do**
2: Initialize S as a $P \times Q$ zero matrix.
3: Set $S_{hi} \leftarrow P\Phi_{hi}\ \forall i \in [1, Q]$
4: **for** $n \leftarrow 2$ **to** $n_{\max}$ **do**
5: Update
$$S_{ab} \leftarrow \frac{n^2 \sum_{l=h+n-2}^{a-1} \sum_{k=n-1}^{b-1} S_{lk}\Phi_{ak}\Phi_{ab}}{(P-n+1)(Q-n+1)}$$
$\forall a \in [h+n-1, P], \forall b \in [n, Q]$.
6: Compute
$$\hat{m}^{(h)}(n) \leftarrow \frac{1}{PQ} \sum_{i=h+n-1}^{P} \sum_{j=n}^{Q} S_{ij}\Phi_{hj}.$$
7: **end for**
8: **end for**
9: Get $\hat{m}(n) \leftarrow \frac{1}{P} \sum_{h=1}^{P-n+1} \hat{m}^{(h)}(n)\ \forall n \in [2, n_{\max}]$.

The indices h, a, b in $S_{ab}^{(h)}$ correspond to the indices i_1, i_n, and α_n in the estimator expression (16). Next, we note that $S_{ab}^{(h)}$ can be written recursively:

$$S_{ab}^{(h)}[n+1] = \sum_{k=n}^{b-1} \sum_{l=h+n-1}^{a-1} S_{lk}^{(h)}[n]\Phi_{ak}\Phi_{ab}. \qquad (19)$$

In this manner, we can iteratively compute $S^{(h)}[n] \to S^{(h)}[n+1]$ to obtain all the partial sums for the spectral moment estimation.

This computation is also memory efficient. The partial sums $S_{lk}^{(h)}[n]$ can be stored as the (lk)-th element of a $P \times Q$ matrix $\left[S_{lk}^{(h)}[n]\right]$ and the computation can be performed in place. The algorithm first initializes the matrix $S^{(h)}[1]$ by setting its h-th row to match the h-th row of Φ, with the rest of the elements initialized to 0. Then $S^{(h)}[2]$ is computed via (19) to obtain $\hat{m}(2)$ with (17). This procedure can be repeated to get $\hat{m}(n)$ for all n ranging from 2 to the desired $n_{\max}$. The estimate for $m(1)$ is the same as the naive estimate $\hat{m}_0(1)$ which is unbiased. Pseudo-code of our recursive algorithm to compute the spectral moment estimates is provided in Algorithm 1.

The computational complexity of our algorithm is $\mathcal{O}(nP^2Q)$, or $\mathcal{O}(nPQ^2)$ if the algorithm is performed on the matrix transpose $\Phi^\top$. In practice, $S^{(h)}[n]$ should also be normalized at each of step of the recursion to prevent any overflow in the calculation; this normalization is included in the description of Algorithm 1. Additionally, step 5 in Algorithm 1 can readily be vectorized with a simple modification to the cumulative summation subroutine. To further improve the accuracy of the estimates, the rows and columns of Φ can first be permuted and the same algorithm can be performed to compute additional cyclic path sums to reduce the variance of the spectral moment estimates.

4.3 Noisy measurements

Our estimator $\hat{m}(n)$ is unbiased even in the presence of independent noise, when the noise is injected into the generative process as $\Phi_{i\alpha} = \phi(x_i, w_\alpha) + \gamma(x_i, w_\alpha, \epsilon_{i\alpha})$, assuming: $\epsilon_{i\alpha}$ is sampled independently from some probability measure ρ_ϵ across all (i, α); $\langle \gamma(x, w, \epsilon)\rangle_\epsilon = 0$; and $\langle \gamma(x, w, \epsilon)^2\rangle_\epsilon < \infty$.

We can also handle the case when the noise $\epsilon_{i\alpha}$ is correlated across row or column entries of $[\Phi_{i\alpha}]$ with a simple modification to our estimator. Multiple measurements can be taken over the same set of inputs and features, $\{x_i\}_{i=1}^P$ and $\{w_\alpha\}_{\alpha=1}^Q$, and can be used to form separate trial measurements of the matrix $\Phi^{(t)}$ with t indexing different trials with independent noise across trials. With measurement samples from only two trials $\Phi^{(1)}$ and $\Phi^{(2)}$, unbiased estimates of the moments can be obtained by alternating trial measurements in the product terms: $\hat{m}'_{\text{alt}}(n) = \prod_{l=1}^n \Phi_{i_l\alpha_l}^{(1)} \Phi_{i_{l+1}\alpha_l}^{(2)}$. This procedure can be easily extended with additional trials to further denoise the spectral moment estimates, and the corresponding modifications to our dynamic

programming algorithm are straightforward (see Appendix for more details).

For the rest of the theoretical and experimental analyses in the main text, we assume the measurement is noise-free.

4.4 Variance of $\hat{m}(n)$

Here we bound the variance of our estimator $\hat{m}(n)$ and derive a probabilistic guarantee for its accuracy. The following lemma gives an upper bound when $\hat{m}(n)$ is computed from a $P \times Q$ measurement matrix.

Theorem 1. *Suppose $\phi \in \mathcal{L}^4(\rho_{\mathcal{X}} \otimes \rho_{\mathcal{W}})$. Then variance of $\hat{m}(n)$ satisfies*

$$\operatorname{Var}(\hat{m}(n)) \leq \left(\frac{1}{P} + \frac{1}{Q}\right) f(n), \tag{20}$$

where

$$f(n) = n^2 \operatorname{Var}\left(\prod_{i=1}^{n} \phi(x_i, w_i)\, \phi(x_{i+1}, w_i)\right). \tag{21}$$

Applying Chebyshev's inequality yields the following guarantee on the absolute error:

Corollary 1. *Suppose $\phi \in \mathcal{L}^4(\rho_{\mathcal{X}} \otimes \rho_{\mathcal{W}})$. Then for any $\delta \in (0, 1)$, with probability at least $1 - \delta$ we have*

$$\left|\hat{m}(n) - m(n)\right| \leq \sqrt{\frac{f(n)}{\delta}\left(\frac{1}{P} + \frac{1}{Q}\right)}. \tag{22}$$

Our estimator is also strongly consistent. The complete proofs of Theorem 1 and estimator consistency are provided in the Appendix.

4.5 Eigenvalues from Moments

Kong and Valiant (2017) presents a linear programming method to recover a finite sequence of eigenvalues from estimated spectral moments. The algorithm takes as input the moments $\{\hat{m}(k)\}_{k=1}^{n}$, the number d of eigenvalues to recover, and an upper bound b on the eigenvalues. It approximates the spectral density p with a discrete distribution $\hat{p}$ defined on points $\{s_i\}_{i=1}^{T} \subset [0, b]$, and then returns the $(d{+}1)$st quantiles of $\hat{p}$ as the eigenvalue estimates.

To compute $\hat{p}$, the algorithm minimizes

$$\min_{\{\hat{p}_i\}_{i=1}^{T}} \sum_{k=1}^{n} \left| \hat{m}(k) - \sum_{i=1}^{T} \hat{p}_i\, s_i^k \right|, \tag{23}$$

subject to $\sum_{i=1}^{T} \hat{p}_i = 1$ and $\hat{p}_i \geq 0$. This optimization is readily solved using linear programming.

Assuming the kernel integral operator has finite rank d, we obtain the following error bound for the recovered eigenvalues.

Corollary 2. *Suppose the kernel integral operator has rank $d < \infty$. Then, the expected total absolute error of recovered eigenvalues $\{\lambda_i\}_{i=1}^{d}$ via the method in Kong and Valiant (2017) applied to our moments $\{\hat{m}(k)\}_{k=1}^{n}$ is bounded by*

$$\left\langle \sum_{i=1}^{d} \left|\lambda_i - \hat{\lambda}_i\right| \right\rangle \leq$$
$$bd\left(c\, 3^n n^2 \left(\sqrt{\left(\frac{1}{P} + \frac{1}{Q}\right)\frac{f(n)}{n^2}} + dc'\right) + \frac{c''}{n} + \frac{1}{d}\right), \tag{24}$$

where c, c', and c'' are positive constants.

The proof is provided in the Appendix.

5 RBF KERNEL OPERATOR

In this section, we describe the spectrum of the kernel integral operator for the radial basis function (RBF) kernel function with multivariate Gaussian inputs. We then confirm that our method estimates the correct spectral moments and compare its performance to other methods.

5.1 Random Fourier features

Rahimi and Recht (2007) and Rudi and Rosasco (2017) describe the following process using random Fourier features. Consider a two-layer neural network with an input layer and a feature layer with a sinusoidal nonlinearity. Given an input pattern $x \in \mathbb{R}^d$ in the input layer, the value of a random feature in the feature layer is defined by weights $w \in \mathbb{R}^d$ and phase shift $b \in \mathbb{R}$ as:

$$\phi(x, (w, b)) = \sqrt{2} \sin\left(w^\top x + b\right) \tag{25}$$

$$w \sim \mathcal{N}\left(0, \Sigma^{-1}\right) \qquad b \sim \mathcal{U}\left(0, 2\pi\right) \tag{26}$$

so that w is a random weight vector sampled from a multivariate normal distribution $\mathcal{N}\left(0, \Sigma^{-1}\right)$, and b is a random phase shift sampled from the uniform distribution $\mathcal{U}$. The similarity between two inputs x and y is given by taking the expectation over all possible features and is equivalent to the RBF kernel:

$$k(x, y) = e^{-\frac{1}{2}(x-y)^\top \Sigma^{-1}(x-y)}. \tag{27}$$

In the following, we also assume that the inputs x and y are sampled from an input distribution $\rho_{\mathcal{X}}(x)$ described by the multivariate normal distribution $\mathcal{N}(0, \Sigma_x)$.

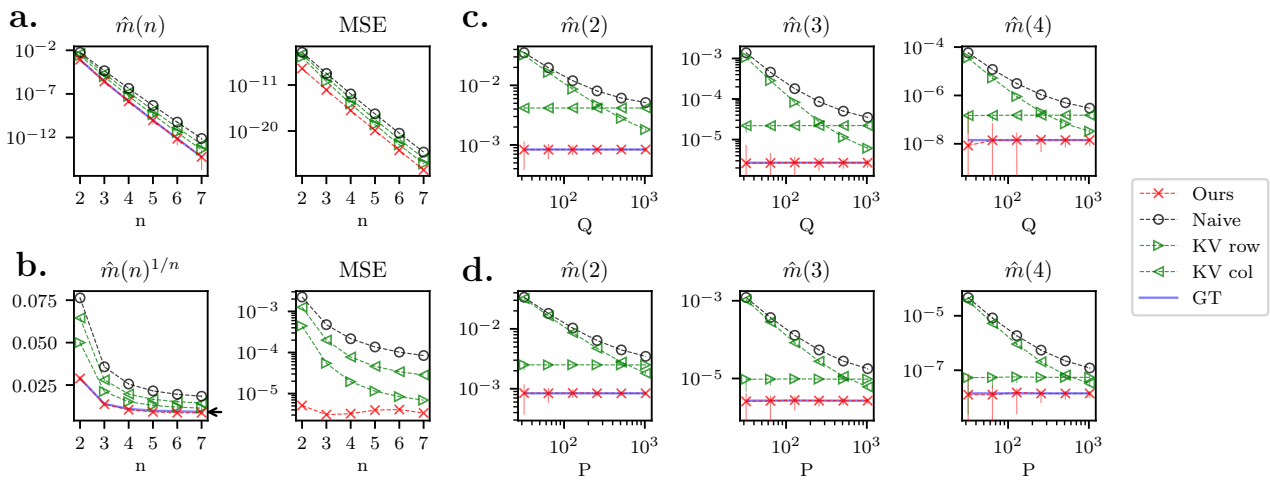

Figure 2: Estimated RBF moments for $d = 5$, $\Sigma_x = I_{d\times d}$, $\Sigma = 0.25 I_{d\times d}$. Our estimator $\hat{m}$ is labeled as "Ours", the two versions of Kong and Valiant estimators $\hat{m}_{\text{KV-row}}$ and $\hat{m}_{\text{KV-col}}$ are labeled as "KV-row" and "KV-col" respectively, the naive estimator $\hat{m}_0$ is labeled as "naive", and the analytic ground truth moments m are labeled as "GT". **a.** $P = 300$ and $Q = 600$. Left: The $\hat{m}(n)$ values for various estimators, with n ranging from 2 to 7. Right: The MSE between $\hat{m}(n)$ and $m(n)$. **b.** The same as **a.**, but for $\hat{m}^{1/n}(n)$. The black arrow indicates the value of the operator norm. **c.** P is fixed to 300, and Q is varied. **d.** Q is fixed to 600, and P is varied. Bars indicate a 50% confidence interval.

5.2 Spectrum of the RBF kernel

The spectrum of the RBF kernel operator has been described before for $d = 1$, or when the kernel and input covariances are isotropic, or can be simultaneously factorized in Zhu et al. (1997); Williams and Seeger (2000); Williams and Rasmussen (2006); Canatar et al. (2021). The spectrum of the general kernel integral operator for arbitrary kernel and input covariances is described in terms of the $d \times d$ positive-definite matrix $\Sigma_x \Sigma^{-1}$. Let $\{\eta_i\}_{i=1}^d$ be the eigenvalues of $\Sigma_x \Sigma^{-1}$ and let $u := \{u_i\}_{i=1}^d$ be a multiset of d natural numbers. Then the following is an eigenvalue of the kernel integral operator for all u:

$$\lambda_u = \prod_{i=1}^{d} \left(\eta_i^{1+u_i} \varphi_{\eta_i}^{1+2u_i}\right)^{-1} \tag{28}$$

where the scalar function $\varphi_z = \frac{1+\sqrt{1+4z}}{2z}$. The largest kernel operator eigenvalue, e.g. the spectral norm of the operator, is obtained when $u_i = 0$ for all i. The corresponding eigenfunctions are given by products of generalized Hermite polynomials with degree u_i and a common multivariate Gaussian function (see Appendix).

The n-th spectral moments can then be computed as

$$m(n) = \prod_{i=1}^{d} \frac{1}{\eta_i^n \varphi_{\eta_i}^n - \varphi_{\eta_i}^{-n}}. \tag{29}$$

Note that $m(1) = 1$ regardless of the choice of kernel parameters.

A particularly interesting case is when $\Sigma = \Sigma_x$ so that $\eta_i = 1$ for all i. In this case, the kernel operator eigenvalues are powers of the golden ratio φ_1, i.e. $\lambda_t = \varphi_1^{-t}$ for $t \in \{1, 2, \ldots\}$, where the t-th eigenvalue has multiplicity $\binom{t+d-1}{t}$. The moments are given by $m(n) = \left(\frac{1}{\phi_1^n - \phi_1^{-n}}\right)^d$, and decrease exponentially for $n \geq 2$ to zero for larger d.

5.3 Numerical estimation results

We consider $P \times Q$ measurement matrices $[\Phi_{i\alpha}]$ taking P samples of inputs patterns $\{x_i\}_{i=1}^P$ and Q samples of weights and phase shifts $\{(w_\alpha, b_\alpha)\}_{\alpha=1}^Q$. The naive estimator $\hat{m}_0(n)$, Kong and Valiant's estimators $\hat{m}_{\text{KV-row}}(n)$ and $\hat{m}_{\text{KV-col}}(n)$, and our method, $\hat{m}(n)$, are all applied to estimate the spectral moments of the operator and compared with the analytical formula in (29). As shown in Figure 2a, there is excellent agreement between our estimates (red dotted lines) and the true moments (blue solid lines). Our estimator achieves the smallest mean-squared error, showing that it is both unbiased and has low variance (see Appendix for detailed bias-variance analysis of the various estimators along with their performance in the presence of independent and correlated noise).

The difference between the estimators is more pro-

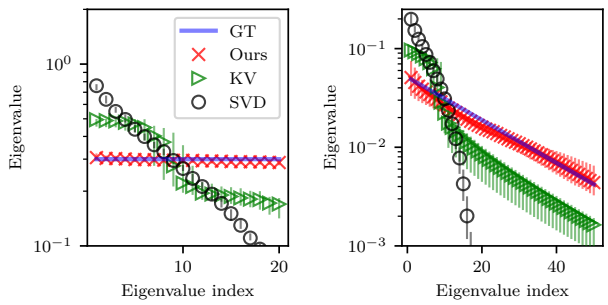

Figure 3: The reconstructed eigenvalues of two generative processes. "GT" refers to ground truth eigenvalues. "SVD" is the singular values of the empirical Gram matrix. "KV" is the eigenvalues reconstructed from $\{\hat{m}_{\mathrm{KV}}(n)\}_{n=1}^{10}$. "Ours" is the eigenvalues reconstructed from $\{\hat{m}(n)\}_{n=1}^{10}$. Left: finite-rank linear generative process whose true eigenvalue is 0.3 with multiplicity 20. $P = Q = 100$. Right: Random Fourier feature generative process whose true ith eigenvalue is $(\eta\varphi_\eta)^{-i}$ with $\eta = 400$. $P = Q = 20$.

nounced when comparing the n-th root of the estimated n-th moment, i.e. $m(n)^{1/n}$. $m(2)^{1/2}$ is the standard deviation of the spectrum, and $\lim_{n\to\infty} m(n)^{1/n}$ gives the operator norm, the largest eigenvalue of the kernel integral operator. Note that even though $\hat{m}(n)$ is unbiased, the n-th root of the estimate, $\hat{m}(n)^{1/n}$, will be a biased estimate of $m(n)^{1/n}$. Nevertheless, we observe that our estimates of the rooted moment achieve remarkably small MSE ($< 10^{-5}$, Figure 2b right). We also observe that our rooted moment estimate accurately recovers the operator norm (Figure 2b left, compare $\hat{m}(7)$ with the black arrow).

Our estimator can be compared to the other estimators as both P and Q are varied. If we fix P and vary Q, $\hat{m}_0(n)$ and $\hat{m}_{\text{KV-col}}(n)$ are biased even in the limit of large Q, due to the small sampling of P (Figure 2c). On the other hand, our estimator and $\hat{m}_{\text{KV-col}}(n)$ asymptotically converge to the true moments at large Q. Similarly, varying P for fixed Q (Figure 2) shows that only our estimator and $\hat{m}_{\text{KV-row}}(n)$ asymptotically converges to the true moments. Our estimator is the only unbiased estimator across all finite P and Q.

6 EIGENVALUE RECONSTRUCTION

Here we demonstrate the recovery of the eigenvalues of kernel integral operators from finite samples by combining our moment estimator $\hat{m}$ with the moment-to-spectrum algorithm of Kong and Valiant (2017).

In the first example, consider the generative process

$$\phi(x, w) = \sqrt{0.3}\, x^\top w, \quad \rho_{\mathcal{X}} = \rho_{\mathcal{W}} = \mathcal{N}(0, I_{d\times d}). \quad (30)$$

The corresponding kernel operator has finite rank d with all nonzero eigenvalues equal to 0.3. As shown in Figure 3 (Left), our estimator $\hat{m}$ yields an eigenvalue spectrum that closely matches the true spectrum, while the reconstruction based on $\hat{m}_{\mathrm{KV}}$ is significantly biased.

In the second example, we recover the eigenvalues of the RBF kernel derived from finite random Fourier features. Here, the kernel operator has a countably infinite number of exponentially decaying eigenvalues. Figure 3 (Right) shows that the largest $d = 50$ eigenvalues are accurately recovered using our estimator $\hat{m}$, whereas other methods fail. Note that in this case, the estimation is sensitive to the choice of parameters d and T.

These examples confirm that our moment-based approach effectively reconstructs the eigenvalue spectrum of kernel integral operators from finite measurements.

7 RELU KERNEL MOMENTS DURING FEATURE LEARNING

Here we show how our estimator can be used to analyze the neural representation of varying widths in a neural network during feature learning. Specifically, a single-hidden layer neural network with ReLU activation, trained on the Fashion-MNIST dataset with Adam-SGD (Xiao et al., 2017; Kingma, 2014) is considered. To enable efficient feature learning, the maximal-update parameterization (μP) as proposed by Yang et al. (2022) is utilized. As the width becomes large, the trained network approaches the mean-field limit where each feature becomes an i.i.d. random variable in the learned weight distribution, and consequently remains exchangeable after training (Mei et al., 2018; Yang et al., 2022; Vyas et al., 2024; Bordelon and Pehlevan, 2022; Seroussi et al., 2023; Yang et al., 2023).

The measurement matrix Φ is constructed from a network with Q neurons in the hidden layer, and each row of Φ represents the Q-dimensional hidden layer activations from one of P input images. Φ is normalized so that it can be compared across different matrix sizes by dividing by the standard deviation of its entries. The spectral moments of the hidden layer representations for networks with widths ranging from 32 to 1024 are shown in Figure 4. The naive spectral moment estimates diverge across the different network widths; in contrast, our estimator produces consistent spectral moments across the entire range of widths. This

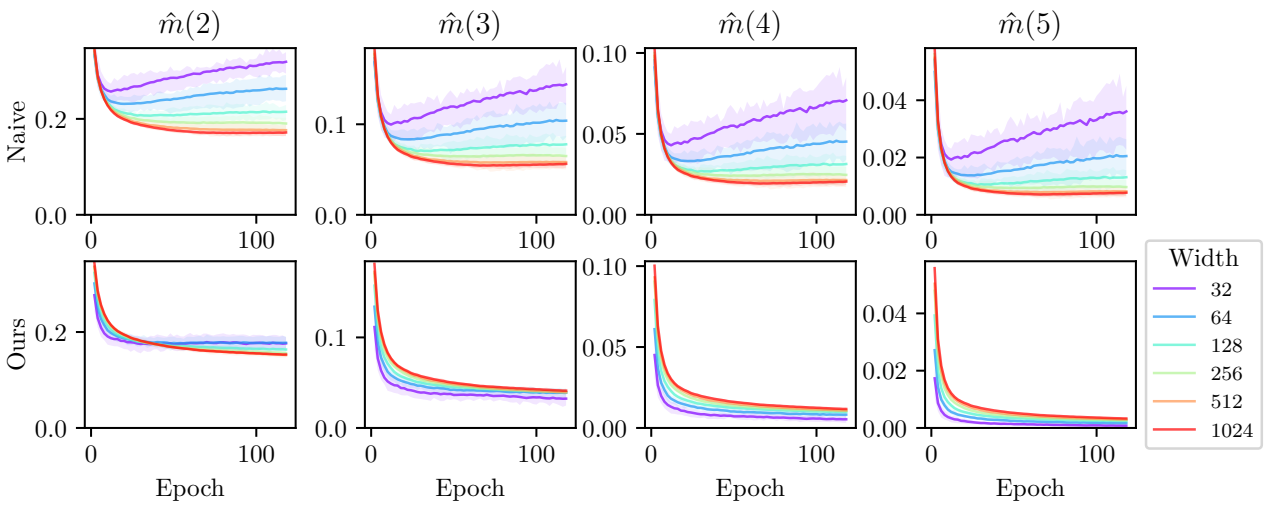

Figure 4: The estimated spectral moments during training of single hidden layer ReLU neural networks. Top row: networks of different widths have dramatically different naive estimates $\hat{m}_0(n)$ of the operator moments. Bottom row: estimates using the unbiased estimator $\hat{m}(n)$ is similar across all widths. Results were obtained from networks trained from 29 random initializations. Shades indicate a 50% confidence interval.

indicates that predictions of learning dynamics from feature learning theories can be applied to understand the behavior of neural networks across a wide range of sizes. We defer exploration of how feature learning theories such as mean-field theory can be used to model our spectral moment estimates to future work.

8 DISCUSSION

We have shown that conventional methods for analyzing the spectrum of a measurement matrix with finitely sampled inputs and features are biased. As an alternative, we propose an unbiased method for estimating the spectral moments of the kernel integral operator from finite measurement matrices. Our method is computationally efficient and results in accurate moment and eigenvalue estimates, as demonstrated in numerical experiments with the RBF kernel where analytic results for the true operator spectrum are known.

Our estimator can be used to gain geometrical insight into measurement matrices of varying sizes. For example, we can accurately estimate the effective dimension of the kernel operator T_k. By considering $\mathrm{tr}\left[T_k \left(T_k + \lambda I\right)^{-1}\right]$ whose Taylor series include weighted sums of the spectral moments (Caponnetto and De Vito, 2007; Bach, 2013), a soft count of the number of eigenvalues above a threshold λ can be obtained. This quantity has been widely used to study prediction performance in kernel ridge regression.

The spectral moment estimates can also be employed in conjunction with methods to approximate kernels with sampled features to reduce computational complexity (Rahimi and Recht, 2007; Rudi and Rosasco, 2017; Rudi et al., 2024). For example, kernel approximation has recently been used in state-of-the-art transformer models by replacing the softmax attention with a kernel (Peng et al., 2021). The decay of the kernel integral operator spectrum can be used to gauge the accuracy of the kernel approximation.

We showed how our estimator can be used to quantify the learning dynamics of neural networks during training. The kernel operator can be related to the Hessian of a quadratic objective function (Dieuleveut et al., 2017; Pedregosa and Scieur, 2020; Sagun et al., 2017; Martin and Mahoney, 2021) which is valuable for understanding gradient-based learning dynamics. Other recent work proposes spectral estimates to understand convergence rates (Dieuleveut et al., 2017) as well as accelerating the optimization (Pedregosa and Scieur, 2020). Thus, there are many potential avenues for future exploration where unbiased and efficient estimates of the spectral characteristics of the kernel integral operator will be valuable.

Acknowledgements

We thank Jonathan D. Victor and Abdulkadir Canatar for their valuable feedback on the project. This work was supported by the Simons Foundation.

References

Bach, F. (2013). Sharp analysis of low-rank kernel matrix approximations. In *Conference on learning theory*, pages 185–209. PMLR.

Bach, F. (2017). On the equivalence between kernel quadrature rules and random feature expansions. *Journal of machine learning research*, 18(21):1–38.

Bhattacharjee, R., Dexter, G., Drineas, P., Musco, C., and Ray, A. (2024). Sublinear time eigenvalue approximation via random sampling. *Algorithmica*, pages 1–66.

Bordelon, B. and Pehlevan, C. (2022). Self-consistent dynamical field theory of kernel evolution in wide neural networks. *Advances in Neural Information Processing Systems*, 35:32240–32256.

Burda, Z., Görlich, A., Jarosz, A., and Jurkiewicz, J. (2004). Signal and noise in correlation matrix. *Physica A: Statistical Mechanics and its Applications*, 343:295–310.

Canatar, A., Bordelon, B., and Pehlevan, C. (2021). Spectral bias and task-model alignment explain generalization in kernel regression and infinitely wide neural networks. *Nature communications*, 12(1):2914.

Canatar, A., Feather, J., Wakhloo, A., and Chung, S. (2024). A spectral theory of neural prediction and alignment. *Advances in Neural Information Processing Systems*, 36.

Caponnetto, A. and De Vito, E. (2007). Optimal rates for the regularized least-squares algorithm. *Foundations of Computational Mathematics*, 7:331–368.

Cho, Y. and Saul, L. (2009). Kernel methods for deep learning. *Advances in neural information processing systems*, 22.

Chung, S., Lee, D. D., and Sompolinsky, H. (2018). Classification and geometry of general perceptual manifolds. *Physical Review X*, 8(3):031003.

Cohen, U., Chung, S., Lee, D. D., and Sompolinsky, H. (2020). Separability and geometry of object manifolds in deep neural networks. *Nature communications*, 11(1):746.

Cucker, F. and Smale, S. (2002). On the mathematical foundations of learning. *Bulletin of the American mathematical society*, 39(1):1–49.

Dieuleveut, A., Flammarion, N., and Bach, F. (2017). Harder, better, faster, stronger convergence rates for least-squares regression. *Journal of Machine Learning Research*, 18(101):1–51.

El Karoui, N. (2008). Spectrum estimation for large dimensional covariance matrices using random matrix theory.

Janson, S. (2018). Renewal theory for asymmetric U-statistics. *Electronic Journal of Probability*, 23(none):1 – 27.

Khorunzhiy, O. (2008). Estimates for moments of random matrices with gaussian elements. *Séminaire de probabilités XLI*, pages 51–92.

Kingma, D. P. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kong, W. and Valiant, G. (2017). Spectrum estimation from samples. *The Annals of Statistics*, 45(5).

Korolyuk, V. S. and Borovskich, Y. V. (2013). *Theory of U-statistics*, volume 273. Springer Science & Business Media.

Ledoit, O. and Wolf, M. (2004). A well-conditioned estimator for large-dimensional covariance matrices. *Journal of multivariate analysis*, 88(2):365–411.

Marchenko, V. A. and Pastur, L. A. (1967). Distribution of eigenvalues for some sets of random matrices. *Matematicheskii Sbornik*, 114(4):507–536.

Martin, C. H. and Mahoney, M. W. (2021). Implicit self-regularization in deep neural networks: Evidence from random matrix theory and implications for learning. *Journal of Machine Learning Research*, 22(165):1–73.

Mei, S., Montanari, A., and Nguyen, P.-M. (2018). A mean field view of the landscape of two-layer neural networks. *Proceedings of the National Academy of Sciences*, 115(33):E7665–E7671.

Pedregosa, F. and Scieur, D. (2020). Acceleration through spectral density estimation. In *International Conference on Machine Learning*, pages 7553–7562. PMLR.

Peng, H., Pappas, N., Yogatama, D., Schwartz, R., Smith, N. A., and Kong, L. (2021). Random feature attention. *arXiv preprint arXiv:2103.02143*.

Rahimi, A. and Recht, B. (2007). Random features for large-scale kernel machines. *Advances in neural information processing systems*, 20.

Rudi, A., Marteau-Ferey, U., and Bach, F. (2024). Finding global minima via kernel approximations. *Mathematical Programming*, pages 1–82.

Rudi, A. and Rosasco, L. (2017). Generalization properties of learning with random features. *Advances in neural information processing systems*, 30.

Sagun, L., Evci, U., Guney, V. U., Dauphin, Y., and Bottou, L. (2017). Empirical analysis of the hessian of over-parametrized neural networks. *arXiv preprint arXiv:1706.04454*.

Seroussi, I., Naveh, G., and Ringel, Z. (2023). Separation of scales and a thermodynamic description of

feature learning in some cnns. *Nature Communications*, 14(1):908.

Vyas, N., Atanasov, A., Bordelon, B., Morwani, D., Sainathan, S., and Pehlevan, C. (2024). Feature-learning networks are consistent across widths at realistic scales. *Advances in Neural Information Processing Systems*, 36.

Williams, C. and Seeger, M. (2000). The effect of the input density distribution on kernel-based classifiers. In *ICML '00 Proceedings of the Seventeenth International Conference on Machine Learning*, pages 1159–1166. Morgan Kaufmann Publishers Inc.

Williams, C. K. and Rasmussen, C. E. (2006). *Gaussian processes for machine learning*, volume 2. MIT press Cambridge, MA.

Xiao, H., Rasul, K., and Vollgraf, R. (2017). Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms.

Yang, A. X., Robeyns, M., Milsom, E., Anson, B., Schoots, N., and Aitchison, L. (2023). A theory of representation learning gives a deep generalisation of kernel methods. In *International Conference on Machine Learning*, pages 39380–39415. PMLR.

Yang, G., Hu, E. J., Babuschkin, I., Sidor, S., Liu, X., Farhi, D., Ryder, N., Pachocki, J., Chen, W., and Gao, J. (2022). Tensor programs v: Tuning large neural networks via zero-shot hyperparameter transfer. *arXiv preprint arXiv:2203.03466*.

Zhu, H., Williams, C. K. I., Rohwer, R., and Morciniec, M. (1997). Gaussian regression and optimal finite dimensional linear models. Technical report, Birmingham.

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes]
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [Yes]
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [Yes]

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. [Yes]
   (b) Complete proofs of all theoretical results. [Yes]
   (c) Clear explanations of any assumptions. [Yes]

3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [Yes]
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes]
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Yes]
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Yes]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. [Yes]
   (b) The license information of the assets, if applicable. [Not Applicable]
   (c) New assets either in the supplemental material or as a URL, if applicable. [Not Applicable]
   (d) Information about consent from data providers/curators. [Not Applicable]
   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Not Applicable]

5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots. [Not Applicable]
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Not Applicable]
   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Not Applicable]

A Kernel integral operator moments and Stieltjes transform

We can easily see that the sequence of spectral moments $\{\sum_{i=1}^{\infty}\lambda_i^n\}_{n=1}^{\infty}$ uniquely determines the non-zero eigenvalues $\{\lambda_i\}_{i=1}^{\infty}$ of a self-adjoint trace-class operator. Consider the following Stieltjes transform where z is in the complex plane:

$$g(z) = \sum_{i=1}^{\infty}\frac{\lambda_i}{z-\lambda_i} \tag{A1}$$

Its Taylor series for z^{-1} near zero is

$$g(z) = \sum_{i=1}^{\infty}\lambda_i z^{-1} + \lambda_i^2 z^{-2} + \lambda_i^3 z^{-3} + \lambda_i^4 z^{-4} + \mathcal{O}\left(z^{-5}\right) \tag{A2}$$

which is equivalent to

$$g(z) = m(1)z^{-1} + m(2)z^{-2} + m(3)z^{-3} + m(4)z^{-4} + \ldots \tag{A3}$$

with moments $m(n) = \sum_{i=1}^{\infty}\lambda_i^n$. The moments $\{m(n)\}_{n=1}^{\infty}$ uniquely define the complex meromorphic function $g(z)$, and the eigenvalues $\{\lambda_i\}_{i=1}^{\infty}$ can then be determined by the location of the poles of $g(z)$. Therefore, the operator moments uniquely determine the non-zero operator eigenvalues.

Newton's identities can also be used to express the relationship between operator moments and the characteristic equation for the eigenvalues of finite rank d operators. Consider the characteristic polynomial

$$f(\lambda) = \prod_{i=1}^{d}(\lambda - \lambda_i) \tag{A4}$$

with roots at the eigenvalues of the operator, i.e. $\{\lambda_i\}_{i=1}^{d}$. The function $f(\lambda)$ can be decomposed in decreasing orders of λ with coefficients consisting of the elementary symmetric polynomials of the roots. Newton's identities recursively relate these coefficients with the power sums of λ_i, which are identical to the spectral moments: $\{m(n)\}_{n=1}^{d}$. Thus, the spectral moments uniquely determine the characteristic polynomial of the eigenvalues.

B Kernel covariance operator

Let $\rho_{\mathcal{X}}(x)$ and $\rho_{\mathcal{W}}(w)$ be probability measures over latent spaces $\mathcal{X}$ and $\mathcal{W}$ respectively. The map $\phi : \mathcal{X}\times\mathcal{W} \to \mathbb{R}$ is square-integrable with respect to both $\rho_{\mathcal{X}}$ and $\rho_{\mathcal{W}}$ and determines the $(i\alpha)$-th coefficient of the measurement matrix $[\Phi_{i\alpha}]$ by $\Phi_{i\alpha} = \phi(x_i, w_\alpha)$. Let $\psi(x) \coloneqq \phi(x,\cdot) : \mathcal{W} \to \mathbb{R}$. Then the completion of the linear space of the functions $\{\psi(x)\}_{x\in\mathcal{X}}$ is $\mathcal{F} \subset \mathcal{L}^2(\mathcal{W}, \rho_{\mathcal{W}})$, with the inner product

$$\langle f|f'\rangle_{\mathcal{F}} = \int d\rho_{\mathcal{W}}(w)\, f(w)f'(w) \tag{A5}$$

where $f, f' \in \mathcal{F}$. Now consider the covariance operator $T_c : \mathcal{F} \to \mathcal{F}$,

$$T_c \coloneqq \int d\rho_{\mathcal{X}}(x)\ |\psi(x)\rangle\langle\psi(x)| \tag{A6}$$

$\mathcal{F}$ is also a reproducing kernel Hilbert space (RKHS), so the Riesz representation theorem implies that for all $w \in \mathcal{W}$, there exists a unique evaluation function $\varphi(w) \in \mathcal{F}$ such that $f(w) = \langle f|\varphi(w)\rangle_{\mathcal{F}}, \forall f \in \mathcal{F}$. The map ϕ can then be expressed as an inner product in the RKHS, $\phi(x,w) \equiv \langle\psi(x)|\varphi(w)\rangle_{\mathcal{F}}$. Define the frame operator $S : \mathcal{F} \to \mathcal{F}$:

$$S = \int d\rho_{\mathcal{W}}(w)\ |\varphi(w)\rangle\langle\varphi(w)| \tag{A7}$$

Then S is equivalent to the identity operator, since $\langle f|S|f'\rangle_{\mathcal{F}} = \langle f|f'\rangle_{\mathcal{F}}$. The set of $|\varphi(w)\rangle$ is a tight Parseval frame in $\mathcal{F}$.

Now we can interpret what it means to compose T_c:

$$T_c^n = \int\prod_{l=1}^{n} d\rho_{\mathcal{X}}(x_l)\ |\psi(x_1)\rangle\langle\psi(x_1)|\psi(x_2)\rangle\cdots\langle\psi(x_{n-1})|\psi(x_n)\rangle\langle\psi(x_n)| \tag{A8}$$

the trace of which is

$$\mathrm{tr}T_c^n = \int \prod_{l=1}^{n} d\rho_{\mathcal{X}}(x_l) \prod_{l=1}^{n} \langle \psi(x_l)|\psi(x_{l+1})\rangle \tag{A9}$$

with the trace constraint $x_{n+1} = x_1$. Since S is the identity operator on $\mathcal{F}$, $\langle\psi(x)|\psi(x')\rangle$ is equivalent to $\langle\psi(x)|S|\psi(x')\rangle$, so

$$\langle \psi(x)|\psi(x')\rangle \equiv \int d\rho_{\mathcal{W}}(w)\ \langle\psi(x)|\varphi(w)\rangle_{\mathcal{F}} \langle\psi(x')|\varphi(w)\rangle_{\mathcal{F}}\,. \tag{A10}$$

Therefore, the trace becomes

$$\mathrm{tr}T_c^n = \int \prod_{l=1}^{n} d\rho_{\mathcal{X}}(x_l) d\rho_{\mathcal{W}}(w_l) \prod_{l=1}^{n} \phi(x_l, w_l)\phi(x_{l+1}, w_l). \tag{A11}$$

Now we can relate T_c to T_k defined in the main text. With the definition of the kernel function and $T_k : \mathcal{L}^2(\mathcal{X}, \rho_{\mathcal{X}}) \to \mathcal{L}^2(\mathcal{X}, \rho_{\mathcal{X}})$, we see that

$$\mathrm{tr}T_c^n = \int \prod_{i=1}^{n} d\rho_{\mathcal{X}}(x_i) \prod_{i=1}^{n} k(x_i, x_{i+1}) \equiv \mathrm{tr}T_k^n. \tag{A12}$$

C Proof of estimator consistency

Consider the sets of increasing index sequences:

$$\{i_1, i_2, \ldots, i_n\} \subset \{1, 2, \ldots, P\}, \quad \text{where} \quad i_1 < i_2 < \cdots < i_n \tag{A13}$$

and similarly for $\{\alpha_1, \alpha_2, \ldots, \alpha_n\} \subset \{1, 2, \ldots, Q\}$, where $\alpha_1 < \alpha_2 < \cdots < \alpha_n$. There are $\binom{P}{n}$ such sequences for $\{i_l\}$ and $\binom{Q}{n}$ for $\{\alpha_l\}$.

Define

$$\hat{m}(n) = \frac{1}{\binom{P}{n}\binom{Q}{n}} \sum_{1\le i_1<\cdots<i_n\le P} \sum_{1\le\alpha_1<\cdots<\alpha_n\le Q} \prod_{l=1}^{n} \Phi_{i_l\alpha_l}\Phi_{i_{l+1}\alpha_l} \tag{A14}$$

where $i_{n+1} = i_1$. The estimator $\hat{m}(n)$ is an unbiased estimator of

$$m(n) \coloneqq \int \prod_{l=1}^{n} d\rho_{\mathcal{X}}(x_l) d\rho_{\mathcal{W}}(w_l) \prod_{l=1}^{n} \phi(x_l, w_l)\phi(x_{l+1}, w_l). \tag{A15}$$

Theorem 2. *$\hat{m}(n)$ is a strongly consistent estimator of $m(n)$, namely*

$$\hat{m}(n) \xrightarrow{a.s.} m(n) \tag{A16}$$

as $P, Q \to \infty$.

Proof. Define the function

$$h(\{w_\alpha\} \mid \{x_i\}) = \frac{1}{\binom{P}{n}\binom{Q}{n}} \sum_{1\le i_1<\cdots<i_n\le P} \sum_{1\le\alpha_1<\cdots<\alpha_n\le Q} \prod_{l=1}^{n} \phi(x_{i_l}, w_{\alpha_l})\phi(x_{i_{l+1}}, w_{\alpha_l}), \tag{A17}$$

with $i_{n+1} = i_1$.

For each fixed $\{x_i\}$, consider h as a U-statistic with asymmetric kernel in the variables $\{w_\alpha\}$. The U-statistic kernel function is

$$h_{\mathrm{kernel}}(\{w_{\alpha_l}\}) = \prod_{l=1}^{n} \phi(x_{i_l}, w_{\alpha_l})\phi(x_{i_{l+1}}, w_{\alpha_l}). \tag{A18}$$

We first need to show that h_{kernel} is absolutely integrable with respect to $\rho_{\mathcal{W}}^{\otimes n}$. Applying Hölder's inequality and using the square-integrability of ϕ, we have

$$\begin{aligned}\int_{\mathcal{W}} d\rho_{\mathcal{W}}(w)\ \left|\phi(x_{i_l}, w)\phi(x_{i_{l+1}}, w)\right| &\leq \left(\int_{\mathcal{W}} d\rho_{\mathcal{W}}(w)\,\phi^2(x_{i_l}, w)\right)^{1/2}\left(\int_{\mathcal{W}} d\rho_{\mathcal{W}}(w)\,\phi^2(x_{i_{l+1}}, w)\right)^{1/2}\\ &< \infty.\end{aligned}$$

Since the integrals are finite for all x_{i_l} and $x_{i_{l+1}}$, the function h_{kernel} is in $\mathcal{L}^1$.

By the strong law of large numbers for U-statistics with absolutely integrable asymmetric kernels (Janson, 2018), it follows that, as $Q \to \infty$,

$$h(\{w_\alpha\} \mid \{x_i\}) \xrightarrow{\text{a.s.}} g(\{x_i\}), \tag{A19}$$

where

$$g(\{x_i\}) = \frac{1}{\binom{P}{n}} \sum_{1 \leq i_1 < \cdots < i_n \leq P} \prod_{l=1}^{n} k(x_{i_l}, x_{i_{l+1}}). \tag{A20}$$

Recall that k is defined by

$$k(x, y) = \int_{\mathcal{W}} d\rho_{\mathcal{W}}(w)\,\phi(x, w)\phi(y, w). \tag{A21}$$

Next, consider $g(\{x_i\})$ as a U-statistic over the variables $\{x_i\}$. The corresponding U-statistic kernel function is

$$g_{\text{kernel}}(\{x_{i_l}\}) = \prod_{l=1}^{n} k(x_{i_l}, x_{i_{l+1}}). \tag{A22}$$

Note g_{kernel} is absolutely integrable. Applying the strong law of large numbers for U-statistics (Korolyuk and Borovskich, 2013), it follows that, as $P \to \infty$,

$$g(\{x_i\}) \xrightarrow{\text{a.s.}} m(n), \tag{A23}$$

where

$$m(n) = \int \prod_{l=1}^{n} d\rho_{\mathcal{X}}(x_l) \prod_{l=1}^{n} k(x_l, x_{l+1}) \tag{A24}$$

with $x_{n+1} = x_1$. Combining the two results for almost sure convergence and using the independence between $\{x_i\}$ and $\{w_\alpha\}$, we conclude that, as $P, Q \to \infty$,

$$\hat{m}(n) \xrightarrow{\text{a.s.}} m(n). \tag{A25}$$

□

D Variance of $\hat{m}$

In this section we sketch a derivation for an upper bound on the variance of our estimator $\hat{m}(n)$ (see Theorem 1). To motivate our derivation, consider a simpler setup. Suppose we have two sets

$$\mathcal{S}_X \coloneqq \{X_j\}_{j=1}^{A} \quad \text{and} \quad \mathcal{S}_W \coloneqq \{W_j\}_{j=1}^{B}, \tag{A26}$$

with elements sampled from distributions μ_X and μ_W, respectively. We assume that while elements within $\mathcal{S}_X$ or $\mathcal{S}_W$ may be correlated, the sets themselves are independent. Now define

$$V \coloneqq \frac{1}{AB} \sum_{i=1}^{A} \sum_{j=1}^{B} F(X_i, W_j), \tag{A27}$$

where $F : \mathcal{S}_X \times \mathcal{S}_W \to \mathbb{R}$. Our goal is to bound the variance of V by an expression of the form

$$\text{Var}(V) \leq C\,\epsilon(A, B), \tag{A28}$$

with C an absolute constant and $\epsilon(A, B)$ a function decreasing as A and B increase.

Writing the variance we have

$$\mathrm{Var}(V) = \left(\frac{1}{AB}\right)^2 \sum_{i,j,k,l} \mathrm{Cov}\Big[F(X_i, W_j),\, F(X_k, W_l)\Big]. \tag{A29}$$

If we let G denote the number of nonzero covariance terms and note that $\mathrm{Var}\big(F(X, W)\big)$ is the largest term, then

$$\mathrm{Var}(V) \le \left(\frac{1}{AB}\right)^2 G\ \mathrm{Var}\big(F(X, W)\big). \tag{A30}$$

In our estimator, each element X_i is a tuple $\{x_{i_r}\}_{r=1}^n$ (with $i_1 < i_2 < \cdots < i_n$) and similarly each W_l is a tuple $\{w_{l_r}\}_{r=1}^n$ (with $l_1 < l_2 < \cdots < l_n$). Thus, the set $\mathcal{S}_X \times \mathcal{S}_W$ corresponds to all cyclic paths of length $2n$ (with the cyclic constraint $i_{n+1} = i_1$). In our case, we define

$$F(X_i, W_l) \coloneqq \prod_{r=1}^{n} \phi(x_{i_r}, w_{l_r})\, \phi(x_{i_{r+1}}, w_{l_r}), \tag{A31}$$

where $i_{n+1} = i_1$. Since the number of ways to choose an increasing sequence of indices is

$$A = \binom{P}{n} \quad \text{and} \quad B = \binom{Q}{n}, \tag{A32}$$

we have $V = \hat{m}(n)$. Moreover, one can show that the number of nonzero covariance terms is

$$G = \binom{P}{n}^2 \binom{Q}{n}^2 - \binom{P}{n}\binom{Q}{n}\binom{P-n}{n}\binom{Q-n}{n}. \tag{A33}$$

Thus,

$$\mathrm{Var}\big(\hat{m}(n)\big) \le \left(1 - \frac{\binom{P-n}{n}\binom{Q-n}{n}}{\binom{P}{n}\binom{Q}{n}}\right) \mathrm{Var}\left(\prod_{i=1}^{n} \phi(x_i, w_i)\, \phi(x_{i+1}, w_i)\right). \tag{A34}$$

It remains to simplify the factor

$$1 - \frac{\binom{P-n}{n}\binom{Q-n}{n}}{\binom{P}{n}\binom{Q}{n}}. \tag{A35}$$

Observe that

$$\frac{\binom{P-n}{n}}{\binom{P}{n}} = \prod_{i=0}^{n-1}\Big(1 - \frac{n}{P-i}\Big) \tag{A36}$$

and similarly for Q:

$$\frac{\binom{Q-n}{n}}{\binom{Q}{n}} = \prod_{i=0}^{n-1}\Big(1 - \frac{n}{Q-i}\Big). \tag{A37}$$

Using $1 - x \le e^{-x}$ (for $x \ge 0$) we obtain

$$\frac{\binom{P-n}{n}\binom{Q-n}{n}}{\binom{P}{n}\binom{Q}{n}} \le \exp\Big(-n^2\Big(\frac{1}{P} + \frac{1}{Q}\Big)\Big). \tag{A38}$$

Then, since $1 - e^{-x} \le x$ for $x \ge 0$,

$$1 - \frac{\binom{P-n}{n}\binom{Q-n}{n}}{\binom{P}{n}\binom{Q}{n}} \le n^2\Big(\frac{1}{P} + \frac{1}{Q}\Big). \tag{A39}$$

Thus, we finally obtain

$$\mathrm{Var}\big(\hat{m}(n)\big) \le n^2\Big(\frac{1}{P} + \frac{1}{Q}\Big)\ \mathrm{Var}\left(\prod_{i=1}^{n} \phi(x_i, w_i)\, \phi(x_{i+1}, w_i)\right). \tag{A40}$$

E Error in eigenvalue recovery

In Kong and Valiant (2017) the following relationship is established between the expected total absolute error in the recovered eigenvalues and the variance of a moment estimator. Specifically, if one recovers d eigenvalues $\{\hat{\lambda}_i\}_{i=1}^d$ from the estimated moments $\{\hat{m}(k)\}_{k=1}^n$ (using, for example, a moment–to–spectrum algorithm), then

$$\left\langle \sum_{i=1}^{d} \left| \lambda_i - \hat{\lambda}_i \right| \right\rangle \leq bd \left(C' 3^n n \left(\sqrt{\mathrm{Var}\big(\hat{m}(n)\big)} + dn\epsilon \right) + \frac{C}{n} + \frac{1}{d} \right), \tag{A41}$$

where C, C', and ϵ are positive constants (with ϵ capturing additional approximation error due to discretization of the spectral density), n denotes the largest order of the moments used in the reconstruction, b is the upper bound of the eigenvalues, and d is the (finite) rank of the operator. By substituting the variance bound derived above into this inequality, one obtains an explicit upper bound on the expected total absolute error in the recovered eigenvalues.

F Spectrum of the general RBF kernel with Gaussian input

F.1 Derivation of spectral moments

We study the kernel operator for the radial basis function (RBF) kernel

$$k(x, x') = e^{-\frac{1}{2}\left(x - x'\right)^\top \Sigma^{-1} \left(x - x'\right)} \tag{A42}$$

and input distribution $\rho_{\mathcal{X}} = \mathcal{N}(0, \Sigma_x)$.

Then the n-th spectral moment is moment is given by

$$m(n) = \int \prod_{i=1}^{n} d\rho_{\mathcal{X}}(x_i) \prod_{i=1}^{n} k(x_i, x_{i+1}) \tag{A43}$$

$$= \left((2\pi)^d \left| \Sigma_x \right| \right)^{-n/2} \int \prod_{i=1}^{n} dx_i \, \exp -\frac{1}{2} \left(\sum_{i=1}^{n-1} \left(x_i - x_{i+1} \right)^\top \Sigma^{-1} \left(x_i - x_{i+1} \right) + \sum_{i=1}^{n} x_i^\top \Sigma_x^{-1} x_i \right) \tag{A44}$$

where $x_{n+1} = x_1$. The integrand simplifies to

$$\exp -\frac{1}{2} \left[\sum_{i=1}^{n} x_i \left(2\Sigma^{-1} + \Sigma_x^{-1} \right) x_i - 2 \left(\sum_{i=1}^{n-1} x_i \Sigma^{-1} x_{i+1} + x_1 \Sigma^{-1} x_n \right) \right], \tag{A45}$$

which can be written as

$$\exp -\frac{1}{2} \bar{x}_n^\top \begin{pmatrix} D & -\Sigma^{-1} & 0 & \cdots & 0 & 0 & -\Sigma^{-1} \\ -\Sigma^{-1} & D & -\Sigma^{-1} & \cdots & 0 & 0 & 0 \\ 0 & -\Sigma^{-1} & D & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & D & -\Sigma^{-1} & 0 \\ 0 & 0 & 0 & \cdots & -\Sigma^{-1} & D & -\Sigma^{-1} \\ -\Sigma^{-1} & 0 & 0 & \cdots & 0 & -\Sigma^{-1} & D \end{pmatrix} \bar{x}_n \tag{A46}$$

where $D \coloneqq 2\Sigma^{-1} + \Sigma_x^{-1}$, and $\bar{x}_n \coloneqq \left[x_1^\top, \ldots, x_n^\top \right]^\top \in \mathbb{R}^{nd}$. Denoting the above block matrix as M, we have simplified the moment equation to

$$m(n) = \left((2\pi)^d \left| \Sigma_x \right| \right)^{-n/2} \int d\bar{x}_n \, \exp \left(-\frac{1}{2} \bar{x}_n^\top M \bar{x}_n \right). \tag{A47}$$

Solving the Gaussian integral gives

$$m(n) = \left(\det \Sigma_x^n \det M \right)^{-\frac{1}{2}}. \tag{A48}$$

In general, the determinant of a block circulant matrix:

$$M = \begin{pmatrix} R_0 & R_1 & \cdots & R_{n-1} \\ R_{n-1} & R_0 & \cdots & R_{n-2} \\ \vdots & \vdots & \ddots & \vdots \\ R_1 & R_2 & \cdots & R_0 \end{pmatrix}, \tag{A49}$$

is given by

$$\det M = \prod_{q=0}^{n-1} \det \left(\sum_{l=0}^{n-1} e^{2\pi i q l/n} R_l \right) \tag{A50}$$

After some algebra, the determinant of the block matrix becomes

$$\det M = \prod_{q=1}^{n} \det \left\{ 2 \left[1 - \cos\left(2\pi \frac{q}{n} \right) \right] \Sigma^{-1} + \Sigma_x^{-1} \right\}. \tag{A51}$$

Therefore,

$$m(n) = \left(\prod_{q=1}^{n} \det \left\{ 2 \left[1 - \cos\left(2\pi \frac{q}{n} \right) \right] \Sigma^{-1} \Sigma_x + I \right\} \right)^{-\frac{1}{2}}. \tag{A52}$$

We can simplify further. Let $\{\eta_i\}_{i=1}^d$ be the eigenvalues of $\Sigma^{-1}\Sigma_x$, which are the same as the eigenvalues of $\Sigma_x \Sigma^{-1}$. Note that η_i is always real, since Σ^{-1} and Σ_x are positive semi-definite, and the product of two positive semi-definite matrices has real eigenvalues. Then,

$$m(n) = \prod_{q=1}^{n} \prod_{i=1}^{d} \left[1 + 2\eta_i - 2\eta_i \cos\left(2\pi \frac{q}{n} \right) \right]^{-\frac{1}{2}}, \tag{A53}$$

which can be written using the law of cosines:

$$m(n) = \prod_{q=1}^{n} \prod_{i=1}^{d} \frac{1}{\eta_i} \left[\left(\frac{1 + \sqrt{1+4\eta_i}}{2\eta_i} \right)^2 + \left(\frac{-1 + \sqrt{1+4\eta_i}}{2\eta_i} \right)^2 - \frac{2}{\eta_i} \cos\left(2\pi \frac{q}{n} \right) \right]^{-\frac{1}{2}} \tag{A54}$$

Let us define $\phi_{\eta_i} = \frac{1+\sqrt{1+4\eta_i}}{2\eta_i}$, so that $\eta_i^{-1}\phi_{\eta_i}^{-1} = \frac{-1+\sqrt{1+4\eta_i}}{2\eta_i}$. Using these definitions, we can rewrite the above as

$$m(n) = \prod_{i=1}^{d} \phi_{\eta_i}^n \left\{ \prod_{q=1}^{n} \left[\left(\eta_i \phi_{\eta_i}^2 \right)^2 - 2\cos\left(2\pi \frac{q}{n} \right) \left(\eta_i \phi_{\eta_i}^2 \right) + 1 \right] \right\}^{-\frac{1}{2}} \tag{A55}$$

Now, the following identity can be used: $\prod_{q=1}^{n} \left(x^2 - 2\cos\left(2\pi \frac{q}{n}\right) x + 1 \right) = \left(x^n - 1\right)^2$. Therefore, we arrive at

$$m(n) = \prod_{i=1}^{d} \frac{1}{\eta_i^n \phi_{\eta_i}^n - \phi_{\eta_i}^{-n}}. \tag{A56}$$

F.2 Derivation of eigenvalues

Consider $f_u(x) = \prod_{i=1}^d H_{u_i}(x_i) e^{-\alpha_i x_i^2}$ where $H_c(x)$ is a c-th order polynomial with leading coefficient of 1. $u \coloneqq \{u_i\}_{i=1}^d$ is a multiset of d natural numbers. We require that $T_k f_u = \lambda_u f_u$, to solve for the eigenvalues and eigenfunctions. Let us first find an expression for $T_k f_u$:

$$\left[T_k f_u \right](y) = \frac{1}{\sqrt{(2\pi)^d \prod_{i=1}^d \eta_i}} e^{-\frac{1}{2}\sum_{i=1}^d y_i^2} \int \prod_{i=1}^{d} dx_i \prod_{i=1}^{d} H_{u_i}(x_i) \exp\left[\sum_{i=1}^{d} - \left(\frac{1}{2\eta_i} + \frac{1}{2} + \alpha_i \right) x_i^2 + x_i y_i \right] \tag{A57}$$

After some algebra, we arrive at

$$[T_k f_u](y) = \sqrt{\prod_{i=1}^{d} \frac{1}{2\alpha_i' \eta_i}} \left\langle \prod_{i=1}^{d} H_{u_i}(x_i) \right\rangle_{\mathcal{N}} e^{-\sum_{i=1}^{d} \left(\frac{1}{2} - \frac{1}{4\alpha_i'}\right) y_i^2} \tag{A58}$$

$$\mathcal{N}\left(\left\{ \frac{1}{2\alpha_i'} y_i \right\}_{i=1}^{d}, \operatorname{diag}\left(\left\{ \frac{1}{2\alpha_i'} \right\}_{i=1}^{d} \right) \right) \tag{A59}$$

where $\alpha_i' = \frac{1}{2\eta_i} + \frac{1}{2} + \alpha_i$, and $\operatorname{diag}\left(\{a_i\}_{i=1}^{d}\right)$ is a diagonal matrix whose i-th diagonal entry is a_i.

Now the eigenvalue equation $[T_k f_u](y) = \lambda_u f_u(y)$ requires that

$$\sqrt{\prod_{i=1}^{d} \frac{1}{2\alpha_i' \eta_i}} \left\langle \prod_{i=1}^{d} H_{u_i}(x_i) \right\rangle_{\mathcal{N}} e^{-\sum_{i=1}^{d} \left(\frac{1}{2} - \frac{1}{4\alpha_i'}\right) y_i^2} = \lambda_u \prod_{i=1}^{d} H_{u_i}(y_i) e^{-\sum_{i=1}^{d} \alpha_i y_i^2} \tag{A60}$$

Equating the exponents $\frac{1}{2} - \frac{1}{4\alpha_i'} = \alpha_i$ gives

$$\alpha_i = \frac{-1 + \sqrt{1 + 4\eta_i}}{4\eta_i} \tag{A61}$$

Recall $\eta_i^{-1}\phi_{\eta_i}^{-1} = \frac{-1+\sqrt{1+4\eta_i}}{2\eta_i}$. Therefore $\sqrt{\frac{1}{2\alpha'\eta_i}} = \eta_i^{-1}\phi_{\eta_i}^{-1}$, which means the above eigenvalue equation simplifies:

$$\left(\prod_{i=1}^{d} \eta_i^{-1} \phi_{\eta_i}^{-1} \right) \left\langle \prod_{i=1}^{d} H_{u_i}(x_i) \right\rangle_{\mathcal{N}} = \lambda_u \prod_{i=1}^{d} H_{u_i}(y_i) \tag{A62}$$

$$\mathcal{N}\left(\left\{ \eta_i^{-1} \phi_{\eta_i}^{-2} y_i \right\}, \operatorname{diag}\left(\left\{ \eta_i^{-1} \phi_{\eta_i}^{-2} \right\} \right) \right). \tag{A63}$$

Since each dimension is independent,

$$\left(\prod_{i=1}^{d} \eta_i^{-1} \phi_{\eta_i}^{-1} \right) \prod_{i=1}^{d} \langle H_{u_i}(x_i) \rangle_{\mathcal{N}} = \lambda_u \prod_{i=1}^{d} H_{u_i}(y_i). \tag{A64}$$

Consider an example polynomial:

$$\langle H_{u_i}(x_i) \rangle_{\mathcal{N}} = \left\langle x_i^{u_i} + \mathcal{O}(x_i^{u_i - 1}) \right\rangle_{\mathcal{N}} = \left(\eta_i^{-1} \phi_{\eta_i}^{-2} \right)^{u_i} y^{u_i} + \mathcal{O}(y^{u_i - 1}). \tag{A65}$$

Showing only the leading order terms of the LHS and RHS of the eigenvalue equation, we see

$$\left(\prod_{i=1}^{d} \eta_i^{-1} \phi_{\eta_i}^{-1} \right) \left(\eta_i^{-1} \phi_{\eta_i}^{-2} \right)^{u_i} y^{u_i} + \mathcal{O}(y^{u_i - 1}) = \lambda_u y^{u_i} + \mathcal{O}(y^{u_i - 1}). \tag{A66}$$

Equating the coefficients of the leading order terms, we find that

$$\lambda_u = \prod_{i=1}^{d} \left(\eta_i^{1+u_i} \phi_{\eta_i}^{1+2u_i} \right)^{-1}. \tag{A67}$$

The same eigenvalues can be found via the Taylor series expansion of $m(n)$. Let us define $r_i = \eta_i^{-1}\phi_{\eta_i}^{-2}$ and $s = \prod_{i=1}^{d} \left(\eta_i^{-1} \phi_{\eta_i}^{-1} \right)$. Then,

$$m(n) = s^n \prod_{i=1}^{d} \frac{1}{1 - r_i^n} = s^n \prod_{i=1}^{d} \left(1 + r_i^n + r_i^{2n} + \cdots \right) \tag{A68}$$

$$= s^n \left(1 + r_1^n + r_1^{2n} + \cdots \right) \left(1 + r_2^n + r_2^{2n} + \cdots \right) \left(1 + r_3^n + r_3^{2n} + \cdots \right). \tag{A69}$$

Notice that expanding the above product yields a sum of terms of n-th degree: $\left(sr_1^a r_2^b r_3^c r_4^d \cdots\right)^n$ where $\{a, b, c, d, \ldots\}$ is a multiset of integers. This implies that

$$\lambda_u = s \prod_{i=1}^{d} r_i^{u_i}. \tag{A70}$$

Plugging in the definitions for r_i and s, we get

$$\lambda_u = \prod_{i=1}^{d} \left(\eta_i^{1+u_i} \phi_{\eta_i}^{1+2u_i}\right)^{-1} \tag{A71}$$

which agrees with the results of the earlier derivation.

G Bias and variance of kernel integral operator moment estimators (empirical)

In the numerical experiments with the radial basis function (RBF) kernel and Gaussian input distribution, we observe that our estimator achieves both the lowest bias and variance error, across all configurations that are tested.

Figure A1) shows the performance of the different estimators along with a detailed breakdown of the error in terms of the experimentally observed bias and variance.

H Moment estimation with noise

An unbiased estimate of the $m(n)$ can be obtained even when the measurements are corrupted by correlated noise by utilizing measurements over two or more trials. In these experiments, we add noise to the measurement matrix resulting in both row-correlated and column-correlated noise,

$$\left\langle \Phi_{i\alpha}^{(t)} \Phi_{j\beta}^{(t)} \right\rangle_t \propto \delta_{ij} + \delta_{\alpha\beta} \tag{A72}$$

where t is the trial index, and $\Phi_{i\alpha}^{(t)}$ is centered. As noted in the main text, the following product with alternation between two trials gives an unbiased estimate of $m(n)$ in the presence of the correlated noise:

$$\hat{m}'_{\text{alt-}\{1,2\}}(n) = \prod_{l=1}^{n} \Phi_{i_l \alpha_l}^{(1)} \Phi_{i_{l+1} \alpha_l}^{(2)} \tag{A73}$$

with the trace constraint $i_{n+1} = i_1$. Algorithm A1 details how to compute the estimator with two trial measurements. In general, when there are T total trials, we can write

$$\hat{m}'_{\text{alt-}\mathcal{T}}(n) = \prod_{l=1}^{n} \Phi_{i_l \alpha_l}^{(t_{2l-1})} \Phi_{i_{l+1} \alpha_l}^{(t_{2l})}. \tag{A74}$$

$\mathcal{T} = \{t_l\}_{l=1}^{2n}$ is a ordered multiset with cardinality $2n$ of the trial index set $\{1, 2, \ldots, T\}$. The necessary and sufficient condition for (A74) to be an unbiased estimator in the presence of correlated noise is that for all i and j where $|i - j| = 1$, and for $i = 1$ and $j = 2n$, t_i and t_j take distinct values. The algorithm for this case is presented in Algorithm A2. Alternatively, one can use algorithm A1 for (A73) by randomly selecting two distinct trials from $\{1, \ldots, T\}$ repeatedly and then averaging over the resulting estimates.

H.1 Numerical estimation results for noisy measurements

The cross-trial alternation method can also be applied to the naive as well as Kong and Valiant (2017) estimators to remove the effects of independent and correlated noise. We test the estimators with or without the cross-trial alternation, on the RBF kernel measurement data with no noise, independent noise, and correlated noise (Figure A2). These numerical tests confirm that with only two trials $T = 2$, our estimator is still unbiased even when the measurement matrix is corrupted by correlated noise (Figure A2e). It can also be seen that for independent noise, we only need one trial ($T = 1$) of the measurement matrix to obtain an unbiased estimate (Figure A2b). Our estimator achieves the lowest bias and variance error across nearly all configurations.

Algorithm A1 Computation of $\hat{m}_{\text{alt}}(n)$ for $n = 2$ to $n_{\max}$ when $T = 2$

Require: $\Phi^{(1)}, \Phi^{(2)} \in \mathbb{R}^{P\times Q}$, $n_{\max}$

for $h \leftarrow 1$ **to** P **do**
  Initialize S as a $P \times Q$ zero matrix.
  Set $S_{hi} \leftarrow P\Phi^{(1)}_{hi}$ $\forall i \in [1, Q]$
  for $n \leftarrow 2$ **to** $n_{\max}$ **do**
    Update $S_{ab} \leftarrow \dfrac{n^2 \sum_{l=h+n-2}^{a-1} \sum_{k=n-1}^{b-1} S_{lk}\Phi^{(2)}_{ak}\Phi^{(1)}_{ab}}{(P-n+1)(Q-n+1)}$ $\forall a \in [h+n-1, P]$, $\forall b \in [n, Q]$.
    Compute $\hat{m}^{(h)}_{\text{alt}}(n) \leftarrow \dfrac{1}{PQ}\sum_{i=h+n-1}^{P}\sum_{j=n}^{Q} S_{ij}\Phi^{(2)}_{hj}$.
  end for
end for
Get $\hat{m}_{\text{alt}}(n) \leftarrow \dfrac{1}{P}\displaystyle\sum_{h=1}^{P-n+1} \hat{m}^{(h)}_{\text{alt}}(n)$ $\forall n \in [2, n_{\max}]$.

Algorithm A2 Computation of $\hat{m}_{\text{alt}}(n)$ for $n = 2$ to $n_{\max}$ for arbitrary $T \geq 2$

Require: $\{\Phi^{(t)}\}_{t=1}^{T} \in \mathbb{R}^{P\times Q}$, $n_{\max}$,

for $h \leftarrow 1$ **to** P **do**
  Initialize S as a $P \times Q$ zero matrix.
  Set $S_{hi} \leftarrow P\Phi^{(t_1)}_{hi}$ $\forall i \in [1, Q]$.
  Choose $t_1 \in \{1, \ldots, T\}$
  for $n \leftarrow 2$ **to** $n_{\max}$ **do**
    Choose $t_{2n-2}, t_{2n-1} \in \{1, \ldots, T\}$ such that $t_{2n-3} \neq t_{2n-2}$ and $t_{2n-2} \neq t_{2n-1}$.
    Update $S_{ab} \leftarrow \dfrac{n^2 \sum_{l=h+n-2}^{a-1} \sum_{k=n-1}^{b-1} S_{lk}\Phi^{(t_{2n-2})}_{ak}\Phi^{(t_{2n-1})}_{ab}}{(P-n+1)(Q-n+1)}$ $\forall a \in [h+n-1, P]$, $\forall b \in [n, Q]$.
    Choose t_r such that $t_r \neq t_{2n-1}$ and $t_r \neq t_1$.
    Compute $\hat{m}^{(h)}_{\text{alt}}(n) \leftarrow \dfrac{1}{PQ}\sum_{i=h+n-1}^{P}\sum_{j=n}^{Q} S_{ij}\Phi^{(t_r)}_{hj}$.
  end for
end for
Get $\hat{m}_{\text{alt}}(n) \leftarrow \dfrac{1}{P}\displaystyle\sum_{h=1}^{P-n+1} \hat{m}^{(h)}_{\text{alt}}(n)$ $\forall n \in [2, n_{\max}]$.

I ReLU network learning implementation

The feature learning example in the main text is demonstrated with a single-hidden layer neural network. Each input $x \in \mathbb{R}^d$ is a flattened vector of Fashion-MNIST image's 28×28 pixels. The $w_i \in \mathbb{R}^d$ represents the weight vector for the i-th neuron in the hidden layer with a total N number of neurons. Let $a_j \in \mathbb{R}^N$ be the weight vector for the j-th neuron in the output layer with 10 neurons, each corresponding to one class in the Fashion-MNIST dataset. Explicitly, the value of the j-th output neuron can be written as:

$$y_j = \sum_{i=1}^{N} \phi(x, w_i) a_{ji} \tag{A75}$$

where $a_{ji} \in \mathbb{R}$ is an i-th element of the vector $a_j \in \mathbb{R}^N$, and

$$\phi(x, w) \coloneqq \max(x \cdot w, 0). \tag{A76}$$

We minimize the sum of the square of the difference between the 10-dimensional network output and the 10-dimensional one-hot vector that indicates the class membership. We use 10,000 images for training the network.

The weights are initialized according to the maximal update parameterization (μP) to ensure feature learning even when N is large (Yang et al., 2022). For the single-hidden layer neural network that will be trained with

Adam-SGD, the μP initialization is:

$$w_i \sim \mathcal{N}\left(0, \frac{1}{N} I_{d\times d}\right) \tag{A77}$$

$$a_j \sim \mathcal{N}\left(0, \frac{1}{N} I_{N\times N}\right) \tag{A78}$$

with independent sampling across all i's and j's. For proper gradient scaling during the backward pass, μP requires the following modification to the model:

$$y_j = \sum_{i=1}^{N} \frac{1}{\sqrt{N}} \phi(x, w_i) a_{ji} \tag{A79}$$

$$\phi(x, w) \coloneqq \sqrt{N} \max\left(x \cdot w, 0\right). \tag{A80}$$

which does not affect the forward pass. The learning rates are fixed to $\frac{01}{N}$. Note that this scaling is specific for Adam-SGD (Yang et al., 2022). We randomly sample 32 images from 10,000 training images for a mini-batch, and train each network for 120 epochs.

In the main text, we train networks with the following widths (N): 32, 64, 128, 256, 512, and 1024, and compute the spectral moments every even epoch. Each entry of the measurement matrix $\Phi \in \mathbb{R}^{P\times Q}$ in this case is obtained as:

$$\Phi_{i\alpha} = \frac{\phi(x_i, w_\alpha)}{\sqrt{\frac{1}{PQ}\sum_{j=1}^{P}\sum_{\beta=1}^{Q}\phi(x_j, w_\beta)^2}} \tag{A81}$$

for $P = 1000$ number of test images $\{x_i\}_{i=1}^{P}$, and Q number of neurons $\{w_\alpha\}_{\alpha=1}^{Q}$. In the main text, we showed the results when measuring all neurons $Q = N$ in the hidden layer. For all training, Quadro GV100 GPU is used.

Next, we explore the case where the measurement matrix consists of only partial observations of the neurons $Q < N$.

I.1 Estimation from partial measurements

Here we explore the following question: given a wide neural network (large N), how closely do the moment estimates from observing all neurons in the hidden layer $Q = N$ and the estimates from observing random subsamples of the neurons $Q < N$ match?

To check this, for each epoch of training a wide single-hidden layer neural network with $N = 1024$ hidden neurons using the above specifications, we estimate the moments from a measurement matrix Φ_{all} from all neurons $Q = N$ and the moments from another measurement matrix Φ_{sub} that observes a smaller set of neurons $Q = 128 < N$. The numerical results show that in every epoch, these two estimates match very closely (Figure A3). This means that we do not need to observe all neurons in a neural network to determine the spectral properties of the kernel operator. This can be particularly useful for large-scale networks, such as in state-of-the-art Transformer models, where computing the covariance matrix is highly memory and computationally intensive.

We also compare these moments to those from networks trained with smaller sizes ($N = 128$), and find that the resulting kernel moments estimates also match (Figure A4), as expected based upon the results from the main text.

J Code availability

The code for the estimators and generating all figures is publicly available on Github.

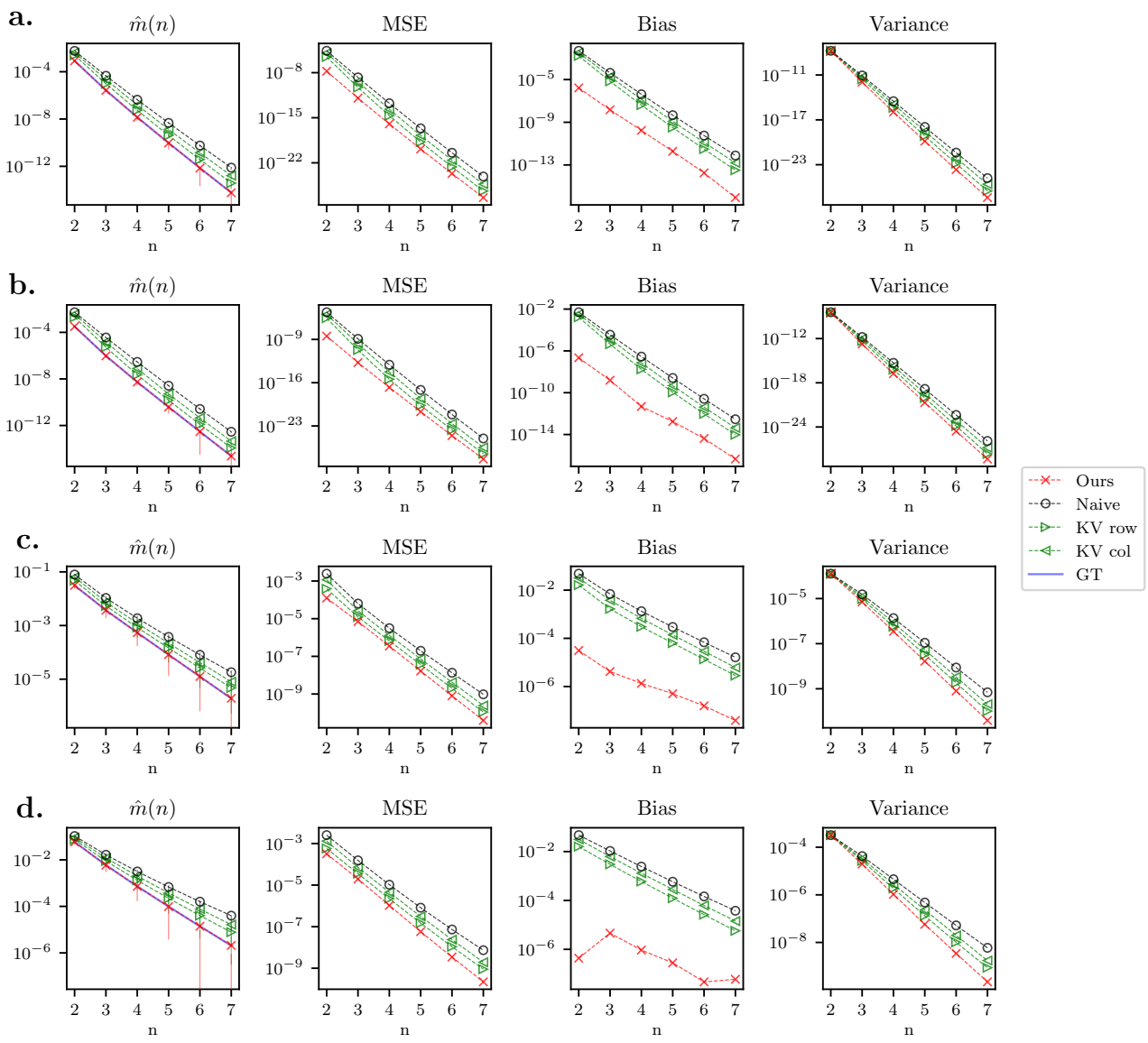

Figure A1: Performance of the estimators with the RBF kernel. Columns from left to right: the estimated moments; the mean-square error between the estimated moments and the ground true moments averaged over multiple samples of Φ's ($\langle \hat{m}(n) - m(n) \rangle_\Phi$); bias error ($\langle \hat{m}(n) \rangle - m(n)$); variance error ($\langle \hat{m}(n)^2 \rangle - \langle \hat{m}(n) \rangle^2$). **a.** $P = 300$, $Q = 600$, $d = 5$, $\Sigma_x = I_{d\times d}$, $\Sigma = 0.25 I_{d\times d}$. **b.** $P = 300$, $Q = 600$, $d = 10$, $\Sigma_x = I_{d\times d}$, $\Sigma = I_{d\times d}$. **c.** $P = 30$, $Q = 60$, $d = 10$, $\Sigma_x = I_{d\times d}$, $\Sigma = 4 I_{d\times d}$. **d.** $P = 30$, $Q = 60$, $d = 4$, $\Sigma_x = I_{d\times d}$, $\Sigma = 0.25 I_{d\times d}$.

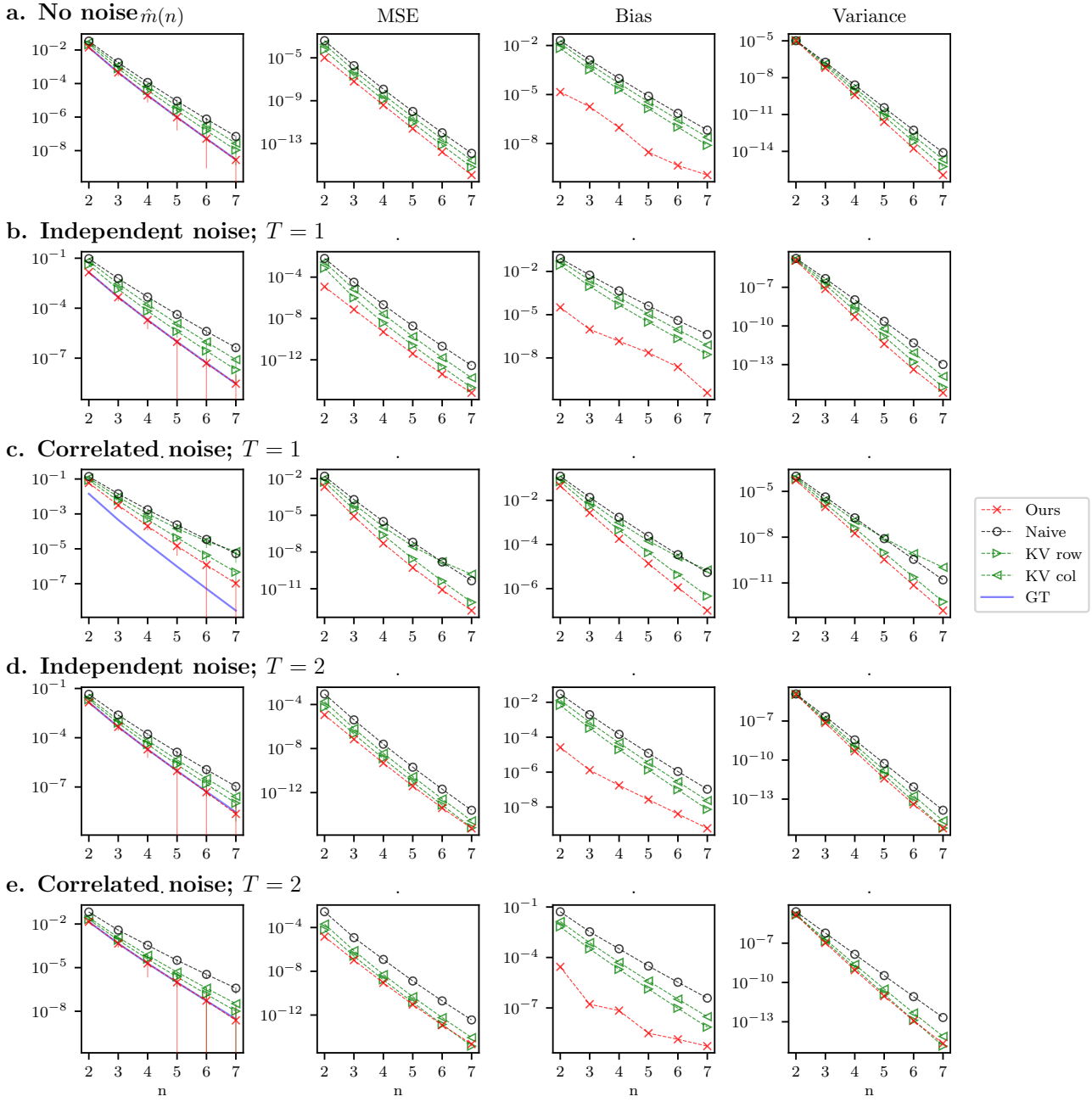

Figure A2: Performance of the estimators in the presence of independent or correlated noise for the RBF kernel. $P = 75$, $Q = 15$, $d = 3$, $\Sigma_x = I_{d\times d}$, $\Sigma = 0.25 I_{d\times d}$. Columns from left to right: the estimated moments; the mean-square error between the estimated moments and the ground true moments averaged over multiple samples of Φ's ($\langle \hat{m}(n) - m(n) \rangle_\Phi$); bias error ($\langle \hat{m}(n) \rangle - m(n)$); variance error ($\langle \hat{m}(n)^2 \rangle - \langle \hat{m}(n) \rangle^2$). **a.** No noise case. **b.** The data is corrupted by an additive independent noise sampled from the standard normal distribution. The number of trials is $T = 1$. **c.** The data is corrupted by an additive correlated noise (not independently) sampled from the standard normal distribution. For a given input i, the noise is correlated between entry $\Phi_{i\alpha}$ and $\Phi_{i\beta}$ for $|\alpha - \beta| > 10$. **d.** Estimators alternating between measurements from two trials with independent noise. **e.** Estimators alternating between measurements from two trials with correlated noise.

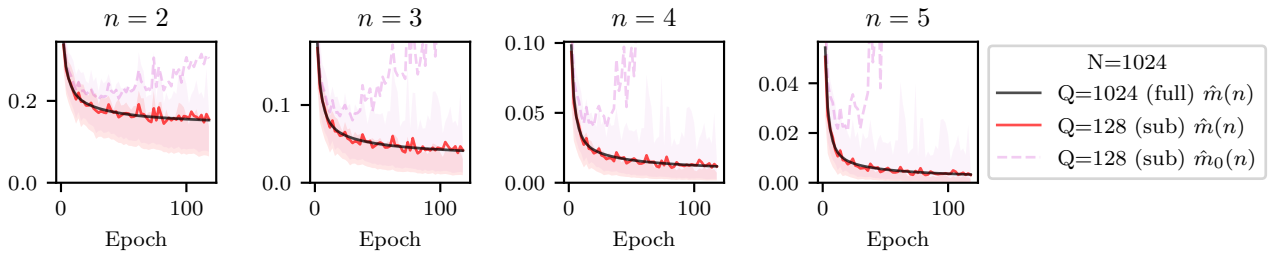

Figure A3: Single hidden layer neural network with $N = 1024$ neurons in the hidden layer. Each plot corresponds to a different moment order n. $P = 1000$ test images are used. Black line: mean value of our estimator $\hat{m}(n)$ applied to Φ with all neurons $Q = N$. Solid red line: mean value of our estimator $\hat{m}(n)$ applied to Φ with subsampled neurons $Q = 128 < N$. Dotted magenta line: mean value of the naive estimator $\hat{m}_0(n)$ applied to Φ with subsampled neurons. The shaded regions indicate a 50% confidence interval. For the naive estimator, the mean value falls outside the confidence interval.

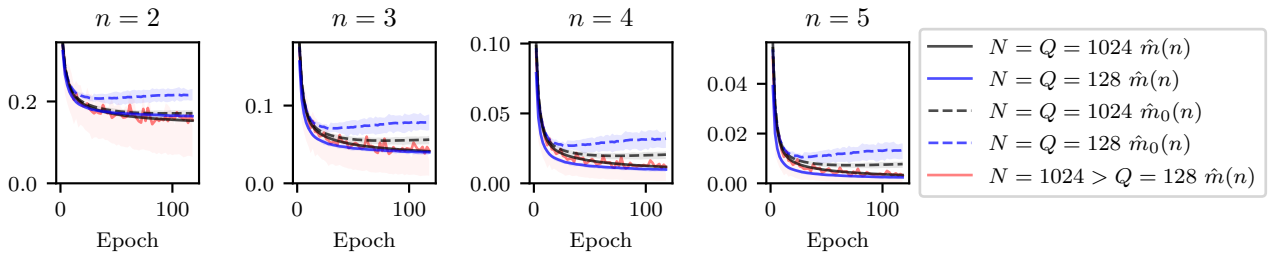

Figure A4: Single hidden layer neural networks trained with $N = 1024$ (black) and $N = 128$ (blue) hidden layer neurons. Each plot corresponds to a different moment order n. Each measurement matrix observes all neurons $Q = N$, except for one case (red) where $Q = 128$ neurons are randomly subsampled from $N = 1024$ neurons. $P = 1000$ test images are used. The solid lines are the values of our estimator $\hat{m}(n)$, and the dotted lines are the values of the naive estimator $\hat{m}_0(n)$. The shaded regions indicate a 50% confidence interval.