Contextures: Representations from Contexts

Runtian Zhai [1] **Kai Yang** [2] **Burak Varıcı** [1] **Che-Ping Tsai** [1] **Zico Kolter** [1] **Pradeep Ravikumar** [1]

Abstract

Despite the empirical success of foundation models, we do not have a systematic characterization of the *representations* that these models learn. In this paper, we establish the contexture theory. It shows that a large class of representation learning methods can be characterized as learning from the association between the input and a *context variable*. Specifically, we show that many popular methods aim to approximate the top-d singular functions of the expectation operator induced by the context, in which case we say that the representation *learns the contexture*. We demonstrate the generality of the contexture theory by proving that representation learning within various learning paradigms—supervised, self-supervised, and manifold learning—can all be studied from such a perspective. We prove that representations that learn the contexture are optimal on those tasks that are compatible with the context. One important implication of our theory is that once the model is large enough to approximate the top singular functions, scaling up the model size yields diminishing returns, so further improvement requires better contexts. To this end, we study how to evaluate a context without knowing the downstream tasks. We propose a metric and show by experiments that it correlates well with the actual performance of the encoder on many real datasets.

1. Introduction

Representation learning underpins the modern deep learning revolution, leading up to the remarkable recent successes of foundation models (Bommasani et al., 2021). But a critical question that has remained unanswered to a satisfactory extent is: why are these models learning anything useful, or perhaps even what representations are these models learning? Unlike classical statistical learning theory, where there is no mystery regarding the statistical estimand, the very target itself is unclear in representation learning. For example, what are the representations that BERT (Devlin et al., 2019)—trained to do cloze tests—is learning, and why are they useful in understanding the sentiment of user reviews on Netflix? What representations do deep neural networks learn, and why are they useful if they cause neural collapse (Papyan et al., 2020), where deep representations could collapse to a few clusters? Do the many different self-supervised learning methods (Balestriero et al., 2023) all learn similar or disparate representations?

The responses to these questions are often muddled. Many analyses are conflated with the mystery of deep learning generalization—the ability of large neural networks to learn function approximations that generalize to unseen points. However, this is a different problem from the mechanism of representation learning. Our focus is on what representation learning (or "pretraining" in the context of foundation models) aims to capture, and why it can be applied to tasks completely different from the objectives used to train the representations. Another way this question is muddled is by recourse to scaling. A popular viewpoint argues that even if the encoder performs poorly on one task, increasing the model size while keeping everything else the same could allow better performance to "emerge" (Wei et al., 2022). However, substantial evidence suggests that certain abilities cannot emerge solely from scaling. Additional training signals, such as alignment (Ouyang et al., 2022), are necessary.

The above questions are naturally interesting to learning theorists, but why should the broader machine learning community care about understanding the mechanism of representation learning, if empirical success seems to be always achievable with existing approaches by scaling up the model size, an empirical observation known as *scaling laws* (Kaplan et al., 2020)? This is because sustainable success or progress is not always guaranteed. Ilya Sutskever recently remarked that "pretraining as we know it will end" (Sutskever, 2024), largely because the current pretraining paradigm is producing diminishing returns. Understanding what representations are learned by foundation models is crucial for designing future generations of pretraining methods, and this is how this field can make scientific progress.

[1] Carnegie Mellon University, Pittsburgh, PA, USA [2] Peking University, Beijing, China. Correspondence to: Runtian Zhai <rzhai@alumni.cmu.edu>.

Proceedings of the 42^{nd} International Conference on Machine Learning, Vancouver, Canada. PMLR 267, 2025. Copyright 2025 by the author(s).

In this work, we establish **the contexture theory**, which provides a unified lens for inspecting a large class of representation learning methods. The central argument of this theory is that representations are learned from the association between the input X and a context variable A. This framework is general enough to encompass a wide variety of learning paradigms, as we demonstrate in Section 3.

Now, suppose we are given a context variable A along with X, how should we learn the representation? In Section 4 we prove that the optimal method is to approximate the top singular functions of the expectation operator induced by the context, in which case we say that the encoder "**learns the contexture**". Such an encoder is optimal as long as the task is **compatible** with the context, and in Section 4 we define a quantitative measurement of such compatibility.

Our theory implies that the main consequence of enlarging the model is that the learned representation space will be brought closer to the span of the top-d singular functions of the expectation operator, which we empirically verify in Section 4.2. Once the two spaces are close enough, further scaling will yield diminishing returns. We envision that future breakthroughs in pretraining require *context scaling*, where better contexts are learned from data, not heuristics.

In Section 5 we study how to evaluate contexts. This is a prerequisite for context scaling because if we cannot even determine which contexts are good, then we cannot create better contexts. The key takeaway is that for a context to be useful (meaning that it can lead to good representations), the *association* between X and A should be moderate—neither too strong nor too weak. For example, if $A = X$, then their association is the strongest; if A is independent of X, then their association is the weakest. However, neither context is useful because A does not provide additional information about X. In Section 5.2, we propose a quantitative measurement of context usefulness that can be efficiently estimated and does not require knowledge of the downstream task. We also empirically verify that the metric correlates with the performance of the encoder on many real datasets.

In one sentence, our key contribution is clarity on the target of a large class of representation learning methods—the singular functions of the expectation operator. We do not discuss the numerical aspect of approximating these functions, which requires an expressive model architecture and a good optimizer, and such analyses are left to future work.

1.1. Examples of Representation Learning Methods

Supervised learning is the simplest way to learn representations. For example, neural networks pretrained on ImageNet (Russakovsky et al., 2015) were very popular in the early days of the deep learning boom (Huh et al., 2016). Specifically, one uses the output of an intermediate layer, typically the one before the last linear layer, as the representation. However, it has never been fully explained why the penultimate layer works so well across disparate tasks.

Self-supervised learning (SSL) is currently the most common way of learning representations. There are two main types of SSL: multi-view learning and masked autoencoders. Multi-view learning includes contrastive learning (Oord et al., 2018; Chen et al., 2020) and non-contrastive learning (Grill et al., 2020; Zbontar et al., 2021; Bardes et al., 2022). Masked autoencoders have wide applications, including language (Devlin et al., 2019; Radford et al., 2019), vision (He et al., 2022), videos (Gupta et al., 2023), and more.

Manifold learning is a classical method that aims to capture the geometry of the data. Examples such as locally linear embedding (LLE) (Roweis & Saul, 2000) and Laplacian eigenmaps (Belkin & Niyogi, 2003) formulate manifold learning as node representation learning on a graph, where connected nodes should have similar embeddings.

2. Definitions and Examples

Let $\mathcal{X}$ be the *input space*. Pretraining aims to learn a feature encoder $\Phi : \mathcal{X} \to \mathbb{R}^d$. We call $\Phi(x)$ the *embedding* of x, and d the embedding dimension. Let $P_{\mathcal{X}}$ be the data distribution. In this work, we assume $P_{\mathcal{X}}$ to be fixed.

The central argument of the contexture theory is that representations are learned from the association between two random variables: the input $X \in \mathcal{X}$ and a context variable $A \in \mathcal{A}$. $\mathcal{A}$ is called the context space. Let $P^+(x, a)$ be the joint distribution of X and A, with marginal distributions $P_{\mathcal{X}}$ and $P_{\mathcal{A}}$. Let $L^2(P_{\mathcal{X}})$ be the L^2 function space *w.r.t.* $P_{\mathcal{X}}$, with inner product $\langle f_1, f_2 \rangle_{P_{\mathcal{X}}} = \mathbb{E}_{X \sim P_{\mathcal{X}}}[f_1(X) f_2(X)]$ and norm $\|f\|_{P_{\mathcal{X}}} = \langle f, f \rangle_{P_{\mathcal{X}}}^{1/2}$. Define $L^2(P_{\mathcal{A}}), \langle \cdot, \cdot \rangle_{P_{\mathcal{A}}}, \|\cdot\|_{P_{\mathcal{A}}}$ for $P_{\mathcal{A}}$ similarly.

2.1. Examples of Contexts

1. **Labels** are a common type of context. They can take different forms, such as discrete categories in classification, continuous values in regression, or text captions of images. Labels may be obtained from human annotators or in pseudo-forms, such as clusters or teacher models. Typically, labels are provided as compatible pairs sampled from the joint distribution $P^+(x, a)$.

2. **Random transformations** generate different views of the same data point. Common transformations include adding random noise to inputs, as seen in diffusion models and denoising autoencoders, or randomly corrupting/masking inputs, as in SimCLR and masked autoencoder. These transformations are typically defined by domain experts and are specified through a predefined conditional distribution $P^+(a \mid x)$.

3. **Graphs** provide locality information about the inputs. The edge weights represent the similarity between two inputs. In this case, we have $\mathcal{A} = \mathcal{X}$ with the conditional distribution $P^+(a \mid x)$ proportional to the edge values between x and a. See Section 3.3.
4. **Features** are predefined or pretrained mappings from $\mathcal{X}$ to a vector space, which we denote by $a = \Omega(x)$. This encompasses teacher models that provide stochastic pretrained feature encoders. In contrast to previous instances, here $P^+(a \mid x)$ is directly described via a reparameterization or structural equation for a in terms of x.

2.2. Induced Kernels and the Expectation Operator

The joint distribution P^+ fully determines the association between X and A. It induces the positive-pair kernel (Johnson et al., 2023) and the dual kernel (Zhai et al., 2024).

Definition 2.1. The **positive-pair kernel** k_A^+ and its **dual kernel** k_X^+ are defined as

$$k_A^+(a, a') = \frac{P^+(a, a')}{P_{\mathcal{A}}(a)P_{\mathcal{A}}(a')} = \frac{\int P^+(a|x)P^+(a'|x)dP_{\mathcal{X}}(x)}{P_{\mathcal{A}}(a)P_{\mathcal{A}}(a')};$$
$$k_X^+(x, x') = \frac{P^+(x, x')}{P_{\mathcal{X}}(x)P_{\mathcal{X}}(x')} = \frac{\int P^+(x|a)P^+(x'|a)dP_{\mathcal{A}}(a)}{P_{\mathcal{X}}(x)P_{\mathcal{X}}(x')}.$$

The kernels are density ratios between joints and marginals for A and X, respectively. They capture how more likely (a, a') or (x, x') appear together than independently given P^+. P^+ also induces the following expectation operator. Intuitively, given a function $g \in L^2(P_{\mathcal{A}})$, the operator computes the expectation of $g(A)$ conditioned on any x.

Definition 2.2. The **expectation operator** $T_{P^+} : L^2(P_{\mathcal{A}}) \to L^2(P_{\mathcal{X}})$ is defined as for all $g \in L^2(P_{\mathcal{A}})$,

$$(T_{P^+}g)(x) = \int g(a)P^+(a \mid x)da = \mathbb{E}[g(A) \mid x].$$

Its adjoint operator $T_{P^+}^* : L^2(P_{\mathcal{X}}) \to L^2(P_{\mathcal{A}})$, which satisfies $\langle f, T_{P^+}g\rangle_{P_{\mathcal{X}}} = \langle T_{P^+}^* f, g\rangle_{P_{\mathcal{A}}}$ ($\forall f \in L^2(P_{\mathcal{X}}), g \in L^2(P_{\mathcal{A}})$), is given by $\left(T_{P^+}^* f\right)(a) = \int f(x)\frac{P^+(a \mid x)P_{\mathcal{X}}(x)}{P_{\mathcal{A}}(a)}dx = \mathbb{E}[f(X) \mid a]$.

Now we discuss the spectral properties of these operators. Define the kernel integral operator $T_{k_A^+} : L^2(P_{\mathcal{A}}) \to L^2(P_{\mathcal{A}})$ as $(T_{k_A^+}g)(a) = \int g(a')k_A^+(a, a')dP_{\mathcal{A}}(a')$. Define the other operator $T_{k_X^+} : L^2(P_{\mathcal{X}}) \to L^2(P_{\mathcal{X}})$ similarly. It is easy to see that $T_{k_A^+} = T_{P^+}^* T_{P^+}$, and $T_{k_X^+} = T_{P^+}T_{P^+}^*$.

We call $\lambda \in \mathbb{R}$ an eigenvalue of $T_{k_A^+}$ with eigenfunction $\nu \in L^2(P_{\mathcal{A}})$, if $T_{k_A^+}\nu = \lambda\nu$. Suppose $T_{k_A^+}$ is a Hilbert-Schmidt integral operator. Then, we can order its eigenvalues by $1 = \lambda_0 \ge \lambda_1 \ge \cdots \ge 0$, and the corresponding eigenfunctions $\nu_0, \nu_1, \cdots$ form an orthonormal basis (ONB) of $L^2(P_{\mathcal{A}})$. Here $\lambda_i \le 1$ because of Jensen's inequality, and $\nu_0 \equiv 1$ is always an eigenfunction of $T_{k_A^+}$ with $\lambda_0 = 1$. Similarly, we can order the eigenvalues of $T_{k_X^+}$ by $1 = \kappa_0 \ge \kappa_1 \ge \cdots \ge 0$, and the eigenfunctions $\mu_0, \mu_1, \cdots$ form an ONB of $L^2(P_{\mathcal{X}})$, where $\mu_0 \equiv 1$. We also have the following result.

Lemma 2.3 (Duality property, Zhai et al. (2024), Proposition 1)**.** *For all i, we have $\lambda_i = \kappa_i \in [0, 1]$. And if $\lambda_i > 0$, then we have $\mu_i = \lambda_i^{-\frac{1}{2}} T_{P^+}\nu_i$, and $\nu_i = \lambda_i^{-\frac{1}{2}} T_{P^+}^*\mu_i$.*

We call $s_i = \lambda_i^{\frac{1}{2}}$ a **singular value** of T_{P^+}, associated with left **singular function** $\mu_i \in L^2(P_{\mathcal{X}})$ and right singular function $\nu_i \in L^2(P_{\mathcal{A}})$. Since $\mu_0 \equiv 1$ and $\nu_0 \equiv 1$, all other μ_i (ν_i) must have zero mean as they are orthogonal to μ_0 (ν_0). Now, we can spectrally decompose P^+ as follows.

Lemma 2.4. *The spectral decomposition of $P+$ is $P^+(x, a) = \sum_i s_i\mu_i(x)\nu_i(a)P_{\mathcal{X}}(x)P_{\mathcal{A}}(a)$.*

Proof. $\forall i$, $\left\langle \frac{P^+(x,a)}{P_{\mathcal{X}}(x)P_{\mathcal{A}}(a)}, \nu_i \right\rangle_{P_{\mathcal{A}}} = \int P^+(a \mid x)\nu_i(a)da = (T_{P^+}\nu_i)(x) = \left(\lambda_i^{\frac{1}{2}}\mu_i\right)(x) = s_i\mu_i(x)$. Since $(\nu_i)_{i \ge 0}$ is an ONB, we have $\frac{P^+(x,a)}{P_{\mathcal{X}}(x)P_{\mathcal{A}}(a)} = \sum_{i=0}^{\infty} s_i\mu_i(x)\nu_i(a)$. □

The first key result of the contexture theory is that the optimal d-dimensional representation should recover the linear space spanned by the top-d singular functions $\mu_1, \cdots, \mu_d$. We say that such a representation learns the contexture. Note that the constant function $\mu_0 \equiv 1$ is excluded, as it does not need to be learned—there is no benefit in allocating a dimension to encode something already universally present.

Definition 2.5. A d-dimensional encoder $\Phi = [\phi_1, \cdots, \phi_d]$ **learns the contexture** of P^+, if there exists a set of top-d singular functions $\{\mu_1, \cdots, \mu_d\}$ of T_{P^+}, such that $\text{span}\{\phi_1, \cdots, \phi_d\} = \text{span}\{\mu_1, \cdots, \mu_d\}$. We also say that Φ **extracts the top-d eigenspace** of $T_{k_X^+}$.

If the multiplicity of s_d is more than 1, then Φ recovering the span of any top-d singular functions suffices. The intuition why learning the contexture is ideal is that such a representation keeps the most variance of the context, analogous to principal component analysis (PCA) in the finite-dimensional case. Suppose $\mathcal{X}$ and $\mathcal{A}$ are both finite sets. Let $N = |\mathcal{X}|$ and $M = |\mathcal{A}|$. Then, a function $f \in L^2(P_{\mathcal{X}})$ is a vector in $\mathbb{R}^N$, $g \in L^2(P_{\mathcal{A}})$ is a vector in $\mathbb{R}^M$, and T_{P^+} is essentially a matrix $\boldsymbol{T} \in \mathbb{R}^{N \times M}$. The goal is learning a d-dimensional embedding $\boldsymbol{E} \in \mathbb{R}^{N \times d}$ for the N samples in $\mathcal{X}$. PCA states that we should use the top-d left singular vectors of $\boldsymbol{T}$ as $\boldsymbol{E}$, or the top-d eigenvectors of $\boldsymbol{T}\boldsymbol{T}^\top$, because they maximize the explained variance. Similarly, functional spaces are essentially infinite-dimensional vector spaces, so the d-dimensional encoder that preserves the most variance of T_{P^+} consists of the top-d left singular functions of T_{P^+}, or the top-d eigenfunctions of $T_{k_X^+} = T_{P^+}T_{P^+}^*$.

3. Learning the Contexture

In this section, we show that every example method in Section 1.1 does one of the following:

(i) Extracts the top-d eigenspace of $T_{k_X^+} = T_{P^+}T_{P^+}^*$ (learns the contexture of P^+), that is, recovering the span of $\mu_1, \cdots, \mu_d$ (excluding $\mu_0 \equiv 1$);

(ii) Extracts the top-d eigenspace of $T_{P^+}\Lambda T_{P^+}^*$, where Λ is the integral operator of a kernel $k_\Lambda(a, a')$, that is $(\Lambda g)(a) = \int g(a')k_\Lambda(a, a')dP_{\mathcal{A}}(a')$. k_Λ is called the **loss kernel**, which is defined by the loss function used in the objective. Since the constant function is not necessarily the top eigenfunction of $T_{P^+}\Lambda T_{P^+}^*$, in this case, we do not exclude any eigenfunction.

Notation: For $f \in L^2(P_{\mathcal{X}})$, denote its mean by $\bar{f} = \mathbb{E}_{P_{\mathcal{X}}}[f(X)]$, and its centered version by $\tilde{f} = f - \bar{f}$. The same notation is used for multi-dim functions and random variables, when the distribution is clear from context. The covariance matrix of $\Phi : \mathcal{X} \to \mathbb{R}^d$, denoted by $\mathrm{Cov}_{P_{\mathcal{X}}}[\Phi]$, is a $d \times d$ matrix $\boldsymbol{C}$ where $\boldsymbol{C}[i, j] = \left\langle \tilde{\phi}_i, \tilde{\phi}_j \right\rangle_{P_{\mathcal{X}}}$.

3.1. Supervised Learning: Label Context

Let A be the label of X. The mean squared error (MSE) is

$$\mathcal{R}(\Phi) = \min_{\boldsymbol{W}, \boldsymbol{b}} \mathop{\mathbb{E}}_{(X,A)\sim P^+} \left[\| \boldsymbol{W}\Phi(X) + \boldsymbol{b} - A \|_2^2 \right], \qquad (1)$$

where $\Phi(X)$ is the output of the layer before the last linear layer in a neural network, and $\boldsymbol{b}$ denotes the bias. If $\boldsymbol{b}$ can be an arbitrary vector, then the linear layer is biased; if $\boldsymbol{b} = \boldsymbol{0}$ is fixed, then the linear layer is unbiased. For classification where A is one-hot, we have the following result.

Theorem 3.1 (Proof in Appendix A.1)**.** *Let A be a one-hot random vector. Suppose the linear layer is unbiased, that is $\boldsymbol{b} = \boldsymbol{0}$. Then, Φ^* minimizes $\mathcal{R}(\Phi)$ if and only if it extracts the top-d eigenspace of $T_{P^+}\Lambda T_{P^+}^*$, where $k_\Lambda(a, a') = \mathbb{I}[a = a']$, or $(\Lambda g)(a) = g(a)P_{\mathcal{A}}(a)$. If all classes have the same size, then the top-d eigenfunctions of $T_{P^+}\Lambda T_{P^+}^*$ and $T_{P^+}T_{P^+}^*$ are the same.*

This theorem works for randomized labels, where each x can belong to multiple classes with certain probabilities. Kernel k_Λ stems from class imbalance; it puts more weights on larger classes. Indeed, in practice, smaller classes are harder to learn. To get rid of Λ, we can use the class-balanced risk (also known as importance weighting (Shimodaira, 2000)):

$$\mathcal{R}_{\mathrm{bal}}(\Phi) = \min_{\boldsymbol{W}, \boldsymbol{b}} \mathop{\mathbb{E}}_{(X,A)\sim P^+} \left[\frac{\| \boldsymbol{W}\Phi(X) + \boldsymbol{b} - A \|_2^2}{\sqrt{P_{\mathcal{A}}(A)}} \right].$$

Theorem 3.2 (Proof in Appendix A.2)**.** *Under the setting of Thm. 3.1, let the linear layer be biased. Then, Φ^* minimizes $\mathcal{R}_{\mathrm{bal}}(\Phi)$ if and only if it learns the contexture of P^+.*

Interestingly, this result can partially explain **neural collapse**. Papyan et al. (2020) empirically showed that when there are d classes of equal sizes and the label A is deterministic, a sufficiently trained deep representation will collapse to an equiangular tight frame (ETF) $\phi_1, \cdots, \phi_d$, which are defined as $\phi_i(x) = c(\mathbb{I}[x \text{ belongs to class } i] - d^{-1})$ for all $i \in [d]$ and some constant c. The span of $\phi_1, \cdots, \phi_d$ is the same as the top-d eigenspace of $T_{P^+}\Lambda T_{P^+}^*$. However, it cannot explain why $\phi_1, \cdots, \phi_d$ converge to the exact functions as above—it only proves that they will span the same space. To prove this, one needs to analyze the training dynamics of the specific optimizer, such as gradient-based methods, while our results are independent of the optimizer.

When the classes have different sizes, the dual kernel of $T_{P^+}\Lambda T_{P^+}^*$ is $k_X^+(x, x') = \mathbb{I}[x \text{ and } x' \text{ have the same label}]$. This is equivalent to the simplex-encoded labels interpolation (SELI) defined by Thrampoulidis et al. (2022, Definition 2), which generalizes neural collapse.

For a regression task where A is an arbitrary Euclidean vector, using the same objective Eqn. (1), a similar result can be proved. See Appendix A.3.

3.2. Self-supervised Learning: Transformation Context

Two major types of SSL are multi-view learning and denoising autoencoders. Multi-view learning independently samples two views A, A^+ from $P^+(\cdot|x)$ for every x. A^+ is called a positive sample of A. Then, one trains $\Psi : \mathcal{A} \to \mathbb{R}^d$ such that $\Psi(A) \approx \Psi(A^+)$. This Ψ is an encoder on $\mathcal{A}$ instead of $\mathcal{X}$, so at downstream we need to convert $\Psi(a)$ to $\Phi(x)$, which is typically done via the **average encoder**:

$$\Phi(x) = (T_{P^+}\Psi)(x) = \int \Psi(a)dP^+(a \mid x).$$

By Lemma 2.3, we have the following corollary.

Corollary 3.3. *Let $s_d > 0$. The average encoder Φ spanning the span of the left top-d singular functions of T_{P^+} is equivalent to Ψ spanning the span of the right top-d singular functions of T_{P^+}.*

Enforcing $\Psi(A) \approx \Psi(A^+)$ alone leads to the degenerate solution where Ψ gives the same embedding to all a. This is called the *feature collapse* problem. There are two solutions: contrastive learning and non-contrastive learning. Prior work by HaoChen et al. (2021); Johnson et al. (2023); Zhai et al. (2024) showed that the spectral contrastive loss $\mathcal{L}_{\mathrm{C}}$ and non-contrastive loss $\mathcal{L}_{\mathrm{N}}$ can learn the contexture of P^+. Let A^+ be a positive sample of A, and A^- be a negative sample drawn from $P_{\mathcal{A}}$ independently. Define

$$\mathcal{L}_{\mathrm{C}} = \mathbb{E}\left[-\left\langle \tilde{\Psi}(A), \tilde{\Psi}(A^+) \right\rangle + \frac{1}{2}\left\langle \tilde{\Psi}(A), \tilde{\Psi}(A^-) \right\rangle^2 \right];$$

$$\mathcal{L}_{\mathrm{N}} = \mathbb{E}\left[\left\| \Psi(A) - \Psi(A^+) \right\|_2^2 \right] \ \text{ s.t. } \ \mathrm{Cov}_{P_{\mathcal{A}}}[\Psi] = \boldsymbol{I},$$

where the (i,j)-th entry of $\mathrm{Cov}_{P_\mathcal{A}}[\Psi]$ is $\langle \psi_i, \psi_j \rangle_{P_\mathcal{A}}$, $i,j \in [d]$. Minimizing $\mathcal{L}_\mathrm{N}$ is a constrained optimization problem. The constraint $\mathrm{Cov}_{P_\mathcal{A}}[\Psi] = \boldsymbol{I}$ is called the **orthonormality constraint**. It makes sure that Ψ must be rank-d, so that it cannot be a constant function on $\mathcal{A}$.

Theorem 3.4 (Proof in Appendix A.4)**.** *Ψ^* minimizes $\mathcal{L}_\mathrm{C}$ or $\mathcal{L}_\mathrm{N}$ if and only if $\tilde{\Phi}^* = T_{P^+}\tilde{\Psi}^*$ learns the contexture.*

For denoising autoencoders, a similar result can be proved. See Appendix A.5.

3.3. Node Representation Learning: Graph Context

Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be an undirected graph, where edge (u,v) has a weight $w(u,v)$ such that $w(u,v) = w(v,u) \geq 0$. Let the degree of node u be $d(u) = \sum_{v\in\mathcal{V}} w(u,v)$, and $d_{\text{sum}} = \sum_v d(v)$. Let $P_\mathcal{X}(u) = \frac{d(u)}{d_{\text{sum}}}$ be a distribution on $\mathcal{V}$, and $P_w(u,v) = \frac{w(u,v)}{d_{\text{sum}}}$ be a distribution on $\mathcal{E}$. Define $P^+(v|u) = \frac{w(u,v)}{d(u)}$. The following problem with a similar orthonormality constraint learns the contexture of P^+.

$$\begin{aligned} \underset{\Phi:\mathcal{X}\to\mathbb{R}^d}{\text{minimize}} \quad & \frac{1}{2}\mathbb{E}_{(u,v)\sim P_w}\left[\|\Phi(u)-\Phi(v)\|_2^2\right] \\ \text{s.t.} \quad & \mathrm{Cov}_{P_\mathcal{X}}[\Phi] = \boldsymbol{I}. \end{aligned} \tag{2}$$

Theorem 3.5 (Proof in Appendix A.6)**.** *Let Φ^* be any solution to Eqn. (2) (so that for any constant c, $\Phi^* + c$ is also a solution). Then, $\tilde{\Phi}^*$ learns the contexture of P^+.*

4. Optimality of the Contexture

So far, we have shown that commonly used learning objectives can learn the contexture. Now, we address the question of *why* and *when* learning the contexture is optimal. Arguably no representations can be good for all downstream tasks, but we show that a feature encoder that learns the contexture is optimal for the class of tasks that are *compatible* with the context. This provides a quantitative characterization of when a task respects the human prior knowledge the context incorporates. Interestingly, as we detail in the sequel, this has intriguing implications for scaling laws.

4.1. Compatibility

The ultimate evaluation of an encoder is its performance on relevant downstream tasks. Most downstream tasks, such as prediction, clustering, and segmentation, can be associated with a target function $f^* \in L^2(P_\mathcal{X})$. For example, multi-class classification can be associated with multiple one-vs-all labeling functions. Moreover, if we are fitting a linear predictor on top of Φ, then the mean and variance of f^* do not matter because we can change the weight and bias of the predictor accordingly. Thus, we can assume that f^* is normalized, that is, it has zero mean and unit variance.

We say that a context P^+ and a task f^* are compatible, if P^+ can help us learn a good predictor for f^*. Formally, consider the scenario where we have a corrupted training set $\{(a_i, y_i)\}$, where $a_i \sim P^+(\cdot \mid x_i)$ and $y_i = f^*(x_i)$. That is, we cannot see the original samples x_i, but can only see the corrupted samples a_i. To learn a predictor on this training set, we can train a predictor $\hat{g} : \mathcal{A} \to \mathcal{Y}$, and then use $\hat{f} = \mathbb{E}[\hat{g}(A) \mid x]$. At test time, given input x, we can draw $a \sim P^+(\cdot \mid x)$ and then output the average of $g^*(a)$. For this procedure to work, two conditions are necessary:

- There exists $g^* \in L^2(P_\mathcal{A})$ s.t. $f^*(x) = \mathbb{E}[g^*(A) \mid x]$.
- This g^* has a low $\mathrm{Var}[g^*(A) \mid x]$ on average over x.

If $\mathrm{Var}[g^*(A) \mid x]$ is high, then $g^*(a)$ will be far away from $y = f^*(x)$, so fitting $\hat{g}$ on (a,y) will not work. Based on these insights, we define compatibility as follows.

Definition 4.1. The **compatibility** with P^+ of any non-zero $f \in L^2(P_\mathcal{X})$ is defined as

$$\rho(f,P^+) = \max_{g\in L^2(P_\mathcal{A}), g\neq \mathbf{0}} \frac{\left\langle \tilde{f}, T_{P^+} g\right\rangle_{P_\mathcal{X}}}{\left\|\tilde{f}\right\|_{P_\mathcal{X}} \|g\|_{P_\mathcal{A}}} \in [0,1].$$

For further insight, let $f = \sum_i u_i \mu_i$ and $g = \sum_i v_i \nu_i$. Then, $\rho(f,P^+) = \max_{v_i} \frac{\sum_{i\geq1} s_i u_i v_i}{\sqrt{\sum_{i\geq1}u_i^2}\sqrt{\sum_i v_i^2}} = \sqrt{\frac{\sum_{i\geq1}s_i^2u_i^2}{\sum_{i\geq1}u_i^2}}$ by Cauchy-Schwarz inequality (the optimal v_i satisfy $v_0 = 0$ and $v_i \propto s_i u_i$ for $i \geq 1$). For any $\epsilon > 0$, we define the class of $(1-\epsilon)$-compatible tasks as

$$\mathcal{F}_\epsilon(P^+) = \left\{ f \in L^2(P_\mathcal{X}) : \mathbb{E}[f] = 0, \rho(f,P^+) \geq 1-\epsilon \right\}.$$

This class of tasks satisfies the two conditions, *i.e.* we can find a g^* with low variance $\mathrm{Var}[g^*(A) \mid x]$:

Theorem 4.2 (Proof in Appendix B.1)**.** *For any $f^* \in \mathcal{F}_\epsilon(P^+)$, there exists a $g^* \in L^2(P_\mathcal{A})$ such that $f^*(x) = \mathbb{E}[g^*(A) \mid x]$, and g^* satisfies*

$$\mathop{\mathbb{E}}_{X\sim P_\mathcal{X}} \mathop{\mathbb{E}}_{A,A'\sim P^+(\cdot\mid X)} \left[(g^*(A)-g^*(A'))^2\right] \leq 4\epsilon \|g^*\|_{P_\mathcal{A}}^2.$$

Now that we have a class of tasks compatible with P^+, we evaluate Φ by its worst-case approximation error on $\mathcal{F}_\epsilon(P^+)$. The most common way to evaluate Φ is to fit a linear predictor on top, also called a **linear probe**, which is the focus of our attention (other methods for using Φ include fitting a small neural network on top, using a kernel method, or using KNN). Specifically, the worst-case approximation error of Φ on $\mathcal{F} \subset L^2(P_\mathcal{X})$ is the maximum error of the optimal linear probe in estimating any function in $\mathcal{F}$. In this work, we focus on the L_2 error.

Definition 4.3. Let $\mathcal{F} \subset L^2(P_{\mathcal{X}})$ be a function class where $f \in \mathcal{F} \Rightarrow \alpha f \in \mathcal{F}$ for all $\alpha \in \mathbb{R}$. The **worst-case approximation error** of $\Phi : \mathcal{X} \to \mathbb{R}^d$ on $\mathcal{F}$ is defined as

$$\operatorname{err}(\Phi; \mathcal{F}) = \max_{f \in \mathcal{F}(P^+),\ \|f\|_{P_{\mathcal{X}}} = 1} \operatorname{err}(\Phi, f),$$

$$\text{where} \quad \operatorname{err}(\Phi, f) = \min_{\boldsymbol{w} \in \mathbb{R}^d,\ b \in \mathbb{R}} \left\| \boldsymbol{w}^\top \Phi + b - f \right\|_{P_{\mathcal{X}}}^2 .$$

The following key result shows that the Φ that minimizes $\operatorname{err}(\Phi; \mathcal{F}_\epsilon(P^+))$ over all d-dimensional encoders must recover the linear space spanned by the $\mu_1, \cdots, \mu_d$. Here μ_0 is excluded since the bias b in the linear predictor implicitly contains μ_0.

Theorem 4.4 (Proof in Appendix B.2)**.** *Suppose* $1 - s_1 \le \epsilon \le 1 - \sqrt{\frac{s_1^2 + s_2^2}{2}}$. *For any* d*, among all* $\Phi = [\phi_1, \cdots, \phi_d]$ *where* $\phi_i \in L^2(P_{\mathcal{X}})$ *,* Φ *minimizes* $\operatorname{err}(\Phi; \mathcal{F}_\epsilon(P^+))$ *if and only if it learns the contexture of* T_{P^+}*. The error is given by*

$$\min_{\Phi : \mathcal{X} \to \mathbb{R}^d,\ \phi_i \in L^2(P_{\mathcal{X}})} \operatorname{err}\big(\Phi; \mathcal{F}_\epsilon(P^+)\big) = \frac{s_1^2 - (1-\epsilon)^2}{s_1^2 - s_{d+1}^2}.$$

Conversely, for any d*-dimensional encoder* Φ *and any* $\epsilon > 0$*, there exists* $f \in L^2(P_{\mathcal{X}})$ *such that* $\rho(f, P^+) = 1 - \epsilon$*, and* $\operatorname{err}(\Phi, f) \ge \frac{s_1^2 - (1-\epsilon)^2}{s_1^2 - s_{d+1}^2}$.

This result has two parts. First, we show that if f^* is compatible ($f^* \in \mathcal{F}_\epsilon(P^+)$), the optimal encoder achieves low error on f^*. Second, we ask what if f^* is incompatible. We cannot claim that no Φ works for f^*—if one knows f^* a priori, then one can set $\phi_1 = f^*$ to achieve zero error. Instead, we show that for any Φ, there exists an f with the same compatibility as f^* for which Φ performs poorly. Therefore, compatibility reflects whether a context is suitable for a task.

Evaluating an arbitrary encoder. The above result bounds the approximation error of the encoder that learns the contexture. We can also bound the approximation error of an arbitrary encoder. See Appendix C.

4.2. Implications for Neural Scaling Laws

Scaling laws (Kaplan et al., 2020) state that the performance of large neural networks grows with their size. Moreover, models of different architectures learn highly aligned representations when scaled up. Huh et al. (2024) thus proposed the platonic representation hypothesis that scaling makes representations more aligned with an underlying *reality*, though they did not formally define this reality.

The contexture theory provides a new perspective on the role of scaling. The function class from which the feature encoders Φ are trained is a subset of $L^2(P_{\mathcal{X}})$, and as the model gets larger, the class approaches $L^2(P_{\mathcal{X}})$. This suggests that scaling brings the learned representation closer to the span of the top-d singular functions of T_{P^+}, explaining why big models learn aligned representations. The key difference is that contexts are designed by humans and thus are more subjective than the so-called "underlying reality".

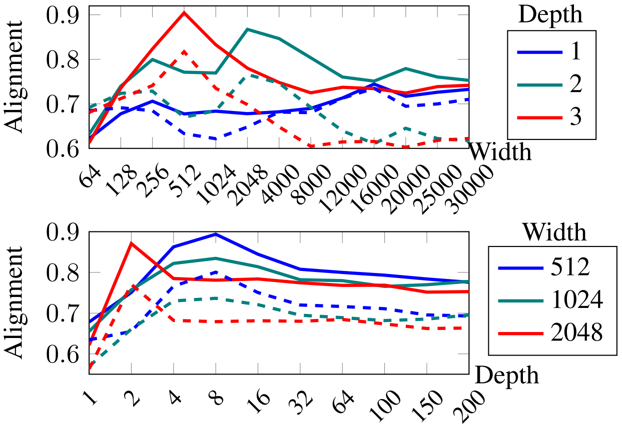


Figure 1. Alignment between the learned representation and the top-d eigenfunctions of $T_{k_{\mathcal{X}}^+}$ on the `abalone` dataset. Solid curves: CCA. Dashed curves: mutual KNN. Depth here means the number of hidden layers.

We substantiate this extrapolation with an experiment on the `abalone` dataset from OpenML. We use KNN ($K = 30$) as context, where $\mathcal{A} = \mathcal{X}$, and P^+ maps x to one of its K nearest neighbors equiprobably. We compare two d-dimensional representations with $d = 128$ learned in the following two ways: (i) Kernel PCA to obtain the exact top-d eigenfunctions of $T_{k_{\mathcal{X}}^+}$; (ii) non-contrastive learning ($\mathcal{L}_N$ in Theorem 3.4) implemented with VICReg (Bardes et al., 2022). For (ii), we use a fully-connected neural network with Tanh activation, skip connections, and AdamW optimizer (Kingma & Ba, 2015; Loshchilov & Hutter, 2017). We use the same number of training epochs for every model. For each width and depth, we run the experiments 15 times with different random initializations, and report the average alignment. See Appendix E for more details.

We measure the alignment between the two representations using the canonical-correlation analysis (CCA) metric R^2_{CCA}, and the mutual KNN metric with 10 neighbors like Huh et al. (2024). We center and whiten the representations (making the covariance identity) when using mutual KNN. CCA is invariant to all invertible linear transformations on Φ, which is ideal because such transformations do not affect the performance of the downstream linear probe, since one can adjust $\boldsymbol{W}$ and $\boldsymbol{b}$ of the linear probe accordingly. We do not use linear CKA proposed by Kornblith et al. (2019) because it is only invariant to orthogonal transformations.

Figure 1 plots the alignment between the exact top-d eigenfunctions and the learned deep representation while varying the depth and width of the neural network. When they are chosen optimally, the CCA can be as high as 0.9, and the mutual KNN can be higher than 0.8. Note that these alignment metric values are very high. For example, in Huh et al. (2024), the mutual KNN metric value is usually below 0.2. Hence, the representation learned by the neural network is highly aligned with the top-d eigenfunctions.

The top plot studies neural networks with increasing widths. We observe that when the neural network is relatively narrow, increasing the width improves alignment. However, once the neural network is sufficiently wide, further increasing the width may have a negative effect. For example, when the depth is 3, the alignment is the highest when the width is 512, and the alignment becomes lower when the network is wider than 512. Since increasing the width can only make the function class of Φ larger, this phenomenon is not due to the expressivity of the neural network. We hypothesize that it arises from optimization difficulty, that is larger models are harder to train effectively. Consequently, with the same number of pretraining steps, a larger model will be farther away from the minima, leading to a reduced alignment.

The bottom plot studies neural networks with increasing depths, and a similar trend is observed. When the network is shallow, increasing the depth improves the alignment. However, once the network is sufficiently deep, further increasing the depth may have a negative effect.

In summary, we draw two conclusions from this experiment: (i) the representation learned by a large neural network is highly aligned with the top-d eigenfunctions; (ii) once the neural network is large enough, further increasing its size does not increase the alignment. Hence, we argue that once the model is large enough such that Φ is already highly aligned with the top-d eigenfunctions, further increasing the model size inevitably yields diminishing returns.

5. Context Evaluation

How to create better contexts is a challenging problem. In this section, we take a first step by studying *when* a context is useful and *how* to efficiently evaluate its usefulness. The key result is that the usefulness of a context is largely determined by the association level between X and A, and **a useful context should have a moderate association**. The association level affects the decay rate of the singular values. We propose a usefulness metric that only depends on the singular values. Then, we empirically verify that this metric has a strong correlation with the actual performance of the encoder on many real datasets. As such, the proposed metric can help practitioners to select among various pretraining methods or hyperparameter settings efficiently.

5.1. The Effect of Context Association

A useful context should provide sufficient training signals that are easy for the model to capture. If the association between the X and A of a context is too weak, then the signals will be insufficient. If the association is too strong, then capturing the signals will be too hard. The association level affects the spectrum of the context—the stronger the association, the slower the decay of the singular values.

Case 1: Weak association. Consider the extreme case where A is independent of X. This context is clearly useless because it provides no information. In this case, only the trivial singular function $\mu_0 \equiv 1$ has a positive singular value; all the other singular values are 0. When X and A are nearly independent, $k_X^+(x, x')$ is very close to 1, which causes the singular values to decay too fast. Formally, we have:

Lemma 5.1 (Proof in Appendix G.1)**.** *When* $|k_X^+(x, x') - 1| < \epsilon$ *for all* $x, x' \in \mathcal{X}$*, we have* $\sum_{i>0} s_i^2 < \epsilon$*.*

In Appendix F, we empirically verify that low association leads to a small $|k_X^+(x, x') - 1|$ for all x, x'. In such a scenario, $\mathcal{F}_\epsilon(P^+)$ is a very small set, so very few tasks are compatible with and can benefit from the context.

Case 2: Strong association. The context $A = X$ is useless. Contexts whose singular values decay too slow are bad because (i) for pretraining, there are non-smooth singular functions with large singular values, which are hard to learn; (ii) for downstream, a larger d is needed, as more singular functions have non-trivial contributions, and it leads to a higher sample complexity. In Appendix F, we empirically verify that kernel k_X^+ has a high Lipschitz constant when the association is strong, meaning that the kernel is non-smooth and thus the singular functions are non-smooth.

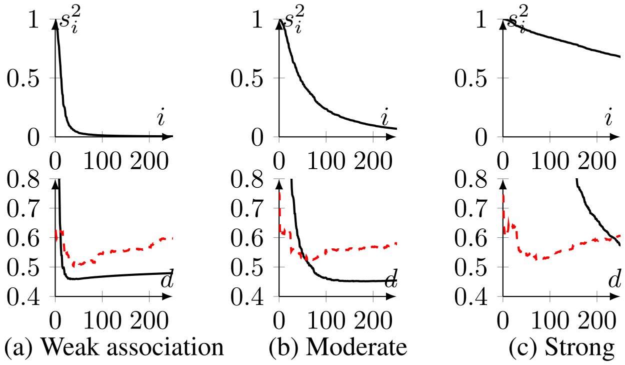
5.2. Task-agnostic Evaluation of Contexts

A good measurement of context usefulness should be task-agnostic, because we would like the pretrained encoder to be transferable to a variety of tasks, which we might not know at pretrain time. Note that for any task-agnostic metric, one can adversarially create a task for which the metric fails, so there is no universal task-agnostic metric. However, a metric can still be very useful if it provides guidance for most real tasks. To this end, we propose the following metric:

$$\tau_d = \frac{1}{1 - s_{d+1}^2} + \beta \, \frac{\sum_{i=1}^{d} s_i^2}{\sum_{i=1}^{d_0} s_i^2}, \qquad \tau = \min_d \tau_d, \tag{3}$$

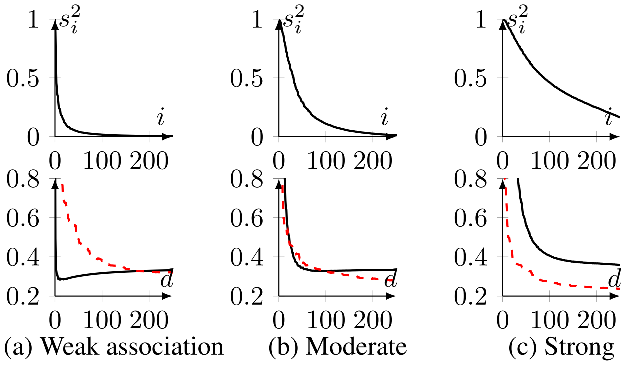
where $\beta > 0$ is a parameter, and d_0 is the maximum embedding dimension we consider. Typically d_0 ranges from 512 to 8192. We choose $\beta = 1$ and $d_0 = 512$ in our experiments. τ_d is a proxy of the prediction error when the embedding dimension is d. Thus, the d that minimizes τ_d can be viewed as the predicted optimal embedding dimension, and τ evaluates the context when d is chosen optimally.

Metric derivation. Let the target function be $f^* = f_0 + f_1$, where $\langle f_0, f_1 \rangle_{P_\mathcal{X}} = 0$, f_1 is compatible with the



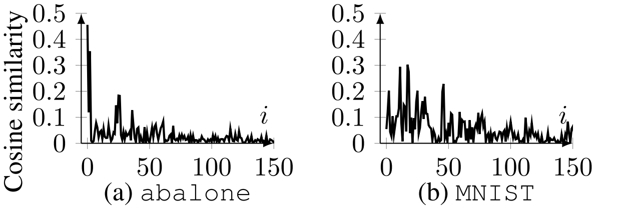
(a) Weak association (b) Moderate (c) Strong

Figure 2. Metric illustration on `abalone`. **Top row:** context spectra. **Bottom row:** black solid curves are τ_d divided by 6; red dashed curves are the actual downstream prediction error. We divide τ_d by 6 to fit it in the same plot.



(a) Weak association (b) Moderate (c) Strong

Figure 3. Metric illustration on `MNIST`, similar to Figure 2.



(a) `abalone` (b) `MNIST`

Figure 4. Comparison of the downstream task between `abalone` and `MNIST`. The y-axis is the cosine similarity between the downstream task and the i-th eigenfunction.

context, and f_0 is not compatible with the context. The prediction error can then be decomposed into (i) the approximation error of f_1; (ii) the approximation error of f_0; (iii) the estimation error. Theorem 4.4 bounds component (i) by $\frac{s_1^2-(1-\epsilon)^2}{s_1^2-s_{d+1}^2}$. In practice, s_1 is usually very close to 1, and we simplify this bound to the first term of Eqn. (3), up to a constant factor. For component (ii), stronger associations imply that more tasks are compatible with the context, reducing this approximation error. Thus, this component should be negatively correlated with $\sum_{i=1}^{d_0} s_i^2$. Component (iii), the estimation error, increases with stronger associations, since higher association typically requires a larger d, and thus greater sample complexity. Based on the results in Zhai et al. (2024), this component can be essentially understood as positively correlated with $\sum_{i=1}^{d} s_i^2$. The second term in Eqn. (3) combines the contributions from components (ii) and (iii), and is designed to be bounded by 1. This metric can be efficiently estimated. It only requires the top-d_0 eigenfunctions of $T_{k_X^+}$, which can be estimated in $O(m^3)$ time using a random subset of $m = \Theta(d_0 \log d_0)$ samples. See Appendix D for details.

We now conduct an experiment that examines τ_d on two datasets. First, we use the `abalone` dataset and KNN as the context, similar to Section 4.2. We adjust the association level by changing K: $K = 150$ (weak), $K = 30$ (moderate) and $K = 5$ (strong). We obtain the exact eigenvalues and eigenfunctions of $T_{k_X^+}$ using kernel PCA. In Figure 2, we plot the spectra of the three contexts in the top row. Then, in the bottom row, we compare τ_d against the prediction error of the linear probe under different d. We can see that when the association is weak or moderate, τ_d first decreases and then increases, which tracks the actual error. However, when the association is too strong, τ_d monotonically decreases with d, and it cannot track the actual error.

Second, we use the `MNIST` dataset. The context is random cropping with crop ratio α. We adjust the association level by changing α: $\alpha = 0.5$ (weak), $\alpha = 0.2$ (moderate) and $\alpha = 0.05$ (strong). Since kernel PCA is not scalable to datasets as large as MNIST, we instead train a LeNet (LeCun et al., 1998) using the non-contrastive learning objective ($\mathcal{L}_N$ in Theorem 3.4) and the AdamW optimizer. Then, we estimate the top eigenvalues using the method in Appendix D. The downstream task is a binary classification task—whether the digit is greater than 4. After pretraining, a linear probe is fit on top of Φ using ridge regression. The result is plotted in Figure 3.

Unlike `abalone`, on `MNIST` the downstream error monotonically decreases with d. This disparity stems from the difference between the two downstream tasks. To demonstrate this, in Figure 4 we plot the cosine similarity between the target function f^* and the estimated i-th eigenfunction on the two datasets. We can see that the variance of f^* on `abalone` is mostly concentrated on the top-5 eigenfunctions, with the first cosine similarity being almost 0.5. In contrast, the variance of f^* on `MNIST` is more scattered, and the cosine similarity is still close to 0.1 for the 150-th eigenfunction. Consequently, having a large d on `abalone` will have a little impact on the approximation error but will increase the estimation error significantly. On the other hand, having a larger d on `MNIST` will decrease the approximation error more than it increases the estimation error, which is why the total error monotonically decreases with d. The takeaway from this experiment is that, while a context with moderate association is generally good, its effectiveness ultimately depends on the specific downstream task.

The implication is that no evaluation metric would universally work for all contexts and downstream tasks. However, a metric would still be useful if it correlates well with the actual error in most scenarios, and thus can provide insights into choosing the right context and the right hyperparameters, such as the mask or crop ratio.

5.3. Empirical Verification

Now we examine if our metric correlates with the encoder's performance on real datasets. In practice, the performance is influenced by many factors. To create a setting where all factors but the context are controlled, we let the encoder be the exact top-d singular functions obtained by kernel PCA.

Each dataset is randomly split into a pretrain set, a downstream labeled set, and a test set. The downstream linear predictor is fit via ridge regression. Hyperparameter grid search is conducted at both encoder learning and downstream stages. The evaluation metric is the mean squared error. Let err_d be the actual prediction error when Φ is d-dimensional. We test d up to $d_0 = 512$. Let d^* be the one that minimizes err_d. We use four types of contexts:

- RBF kernels: $k(x, a) = \exp(-\gamma \|x - a\|^2)$. We define P^+ as $P^+(a \mid x) \propto k(x, a)$ for each x.
- KNN: $P^+(a \mid x) = K^{-1}$ if a is a KNN of x, else 0.
- RBF_{mask}: First, randomly mask 20% of the features, and then apply RBF kernels to the other features. Specifically, we randomly draw 50 masks, and use the average of P^+ over all masks as the context.
- KNN_{mask}: 20% random masking and then apply KNN.

For each of these contexts, $\mathcal{A} = \mathcal{X}$. For each type, we use 35 contexts by adjusting γ for RBF kernels and K for KNN. By doing so, we adjust the association level between X and A. We make sure that contexts in every type range from very weak to very strong association. We do not use masking alone because its dual kernel is hard to estimate.

In Table 1 we report the correlation between τ and err_{d^*} over all 140 contexts from the 4 types on 28 classification (Cls) and regression (Reg) datasets from OpenML (Vanschoren et al., 2013) widely used in machine learning research. The most common metric is the Pearson correlation, but it can only detect linear correlations, while the correlation between τ and err_{d^*} is not necessarily linear. Thus, we also report the distance correlation (Székely et al., 2007) that can detect non-linear correlations, but it cannot tell if the correlation is positive or negative because it is always non-negative.

The median reported in the table shows that on more than half of the datasets, there is a Pearson correlation of over 0.5, which is in general considered a strong correlation. The distance correlation is even higher. As expected, the metric does not work on all datasets. For example, the Pearson correlation is very negative on brazilian_houses and fifa. To understand the failure modes, in Appendix G we do more analysis on the datasets where our metric fails.

Table 1. Correlation between τ and err_{d^*} on all 4 types of contexts (clipped). Full results reported in Table 3 in Appendix G.

Dataset	Size (↑)	#Feat.	Type	Pearson	Dist.
diabetes	768	8	Cls	0.737	0.740
Moneyball	1232	14	Reg	0.680	0.650
yeast	1269	8	Cls	0.221	0.256
splice	3190	60	Cls	0.831	0.801
abalone	4177	8	Reg	0.028	0.470
mushroom	8124	22	Cls	0.185	0.340
pumadyn32nh	8192	32	Reg	0.938	0.961
SpeedDating	8378	120	Cls	0.590	0.656
grid_stability	10000	12	Reg	0.925	0.911
brazilian_houses	10692	9	Reg	-0.290	0.563
fifa	19178	28	Reg	-0.349	0.663
kings_county	21613	21	Reg	0.842	0.882
cps88wages	28155	6	Reg	0.250	0.479
Mean on 28 datasets				0.431	0.611
Median on 28 datasets				0.587	0.659

6. Conclusion

We advance the science of representation learning by articulating the target of representation learning—the top singular functions of a particular operator induced by a context, which we term *contextures*. We further prove that such a representation is optimal because it minimizes the worst-case approximation on the class of tasks compatible with the context. We show that most representation learning approaches could be cast as estimating contextures, and empirically verified that large neural networks can learn the top singular functions well. We further analyze when a context can be useful, relating that to its spectrum, and proposed a task-agnostic usefulness metric that correlates well with the encoder's performance on real datasets.

Our analysis has three limitations, which lead to three open problems. First, our analysis focused on the minimizers of the objectives. However, Cohen et al. (2021) showed that deep models trained by popular gradient methods do not find the minimizers, but instead oscillate around the edge of stability. The open problem is how this phenomenon affects our results. Second, we did not discuss the impact of the inductive bias of the model architecture, such as the translation invariance of convolutional neural networks. Such inductive biases can affect the context and, therefore, the encoder. We pose how to integrate the effect of these biases into our theory as an open problem. Third, our theory assumes that $P_{\mathcal{X}}$ is fixed. In practice, however, there is always a data distribution shift from upstream to downstream. Refining our theory to handle such distribution shifts is an exciting direction for future work.

Acknowledgements

We thank Rattana Pukdee, Hugo Contant, Chenhao Zhang and Zihao Ye for their feedback on this paper. Kai Yang did this work at Carnegie Mellon University, under the PKU-CMU undergraduate research program. We acknowledge the support of NSF via IIS-2211907, ONR via N00014-23-1-2368, AFRL via FA8750-23-2-1015, and DARPA via HR00112020006.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

Code and Data Availability

The code for this paper can be found at https://colab.research.google.com/drive/1GdJ0Yn-PKiKfkZIwUuon3WpTpbNWEtAO?usp=sharing. The data can be downloaded from OpenML.

References

Balestriero, R., Ibrahim, M., Sobal, V., Morcos, A. S., Shekhar, S., Goldstein, T., Bordes, F., Bardes, A., Mialon, G., Tian, Y., Schwarzschild, A., Wilson, A. G., Geiping, J., Garrido, Q., Fernandez, P., Bar, A., Pirsiavash, H., LeCun, Y., and Goldblum, M. A cookbook of self-supervised learning. *arxiv:2304.12210*, 2023.

Bardes, A., Ponce, J., and LeCun, Y. VICReg: Variance-invariance-covariance regularization for self-supervised learning. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=xm6YD62D1Ub.

Baudat, G. and Anouar, F. Generalized discriminant analysis using a kernel approach. *Neural computation*, 12(10): 2385–2404, 2000.

Belkin, M. and Niyogi, P. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural computation*, 15(6):1373–1396, 2003.

Bommasani, R., Hudson, D. A., Adeli, E., Altman, R., Arora, S., von Arx, S., Bernstein, M. S., Bohg, J., Bosselut, A., Brunskill, E., et al. On the opportunities and risks of foundation models. *arXiv:2108.07258*, 2021.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. A simple framework for contrastive learning of visual representations. In *Proc. International Conference on Machine Learning*, July 2020.

Cohen, J., Kaur, S., Li, Y., Kolter, J. Z., and Talwalkar, A. Gradient descent on neural networks typically occurs at the edge of stability. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=jh-rTtvkGeM.

Devlin, J., Chang, M., Lee, K., and Toutanova, K. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT, Volume 1 (Long and Short Papers)*, Minneapolis, MN, June 2019. Association for Computational Linguistics.

Grill, J.-B., Strub, F., Altché, F., Tallec, C., Richemond, P., Buchatskaya, E., Doersch, C., Avila Pires, B., Guo, Z., Gheshlaghi Azar, M., Piot, B., Kavukcuoglu, K., Munos, R., and Valko, M. Bootstrap your own latent - a new approach to self-supervised learning. In *Proc. Advances in Neural Information Processing Systems*, December 2020.

Gupta, A., Wu, J., Deng, J., and Li, F.-F. Siamese masked autoencoders. In *Advances in Neural Information Processing Systems*, New Orleans, LA, December 2023.

HaoChen, J. Z., Wei, C., Gaidon, A., and Ma, T. Provable guarantees for self-supervised deep learning with spectral contrastive loss. In *Proc. Advances in Neural Information Processing Systems*, December 2021.

He, K., Chen, X., Xie, S., Li, Y., Dollár, P., and Girshick, R. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, New Orleans, LA, June 2022.

Huh, M., Agrawal, P., and Efros, A. A. What makes imagenet good for transfer learning? *arXiv:1608.08614*, 2016.

Huh, M., Cheung, B., Wang, T., and Isola, P. Position: The platonic representation hypothesis. In *Proc. International Conference on Machine Learning*, Vienna, Austria, July 2024.

Jing, L., Vincent, P., LeCun, Y., and Tian, Y. Understanding dimensional collapse in contrastive self-supervised learning. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=YevsQ05DEN7.

Johnson, D. D., Hanchi, A. E., and Maddison, C. J. Contrastive learning can find an optimal basis for approximately view-invariant functions. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=AjC0KBjiMu.

Kaplan, J., McCandlish, S., Henighan, T., Brown, T. B., Chess, B., Child, R., Gray, S., Radford, A., Wu, J., and Amodei, D. Scaling laws for neural language models. *arXiv:2001.08361*, 2020.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. In Bengio, Y. and LeCun, Y. (eds.), *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. URL http://arxiv.org/abs/1412.6980.

Kornblith, S., Norouzi, M., Lee, H., and Hinton, G. Similarity of neural network representations revisited. In *Proc. International Conference on Machine Learning*, Long Beach, CA, June 2019.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Liu, Q., Lu, H., and Ma, S. Improving kernel fisher discriminant analysis for face recognition. *IEEE transactions on circuits and systems for video technology*, 14(1):42–49, 2004.

Loshchilov, I. and Hutter, F. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Mika, S., Ratsch, G., Weston, J., Scholkopf, B., and Mullers, K.-R. Fisher discriminant analysis with kernels. In *Neural networks for signal processing IX: Proceedings of the 1999 IEEE signal processing society workshop (cat. no. 98th8468)*, pp. 41–48. Ieee, 1999.

Oord, A. v. d., Li, Y., and Vinyals, O. Representation learning with contrastive predictive coding. *arXiv:1807.03748*, 2018.

Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., et al. Training language models to follow instructions with human feedback. In *Proc. Advances in Neural Information Processing Systems*, New Orleans, LA, December 2022.

Paninski, L. Estimation of entropy and mutual information. *Neural computation*, 15(6):1191–1253, 2003.

Papyan, V., Han, X., and Donoho, D. L. Prevalence of neural collapse during the terminal phase of deep learning training. *Proceedings of the National Academy of Sciences*, 117(40):24652–24663, 2020.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I., et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

Roweis, S. T. and Saul, L. K. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290 (5500):2323–2326, 2000.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115: 211–252, 2015.

Shawe-Taylor, J., Williams, C. K., Cristianini, N., and Kandola, J. On the eigenspectrum of the gram matrix and the generalization error of kernel-PCA. *IEEE Transactions on Information Theory*, 51(7):2510–2522, 2005.

Shimodaira, H. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of Statistical Planning and Inference*, 90(2):227–244, 2000.

Sutskever, I. Test of time award talk: Sequence to sequence learning with neural networks. Advances in Neural Information Processing Systems, 2024.

Székely, G. J., Rizzo, M. L., and Bakirov, N. K. Measuring and testing dependence by correlation of distances. *The Annals of Statistics*, 35(6):2769–2794, 2007. ISSN 00905364. URL http://www.jstor.org/stable/25464608.

Thrampoulidis, C., Kini, G. R., Vakilian, V., and Behnia, T. Imbalance trouble: Revisiting neural-collapse geometry. In *Proc. Advances in Neural Information Processing Systems*, New Orleans, LA, December 2022.

Vanschoren, J., van Rijn, J. N., Bischl, B., and Torgo, L. Openml: Networked science in machine learning. *SIGKDD Explorations*, 15(2):49–60, 2013. doi: 10.1145/2641190.2641198. URL http://doi.acm.org/10.1145/2641190.2641198.

Wei, J., Tay, Y., Bommasani, R., Raffel, C., Zoph, B., Borgeaud, S., Yogatama, D., Bosma, M., Zhou, D., Metzler, D., Chi, E. H., Hashimoto, T., Vinyals, O., Liang, P., Dean, J., and Fedus, W. Emergent abilities of large language models. *Transactions on Machine Learning Research*, 2022. ISSN 2835-8856. URL https://openreview.net/forum?id=yzkSU5zdwD. Survey Certification.

Yarotsky, D. Optimal approximation of continuous functions by very deep relu networks. In *Conference on Learning Theory*, pp. 639–649, Stockholm, Sweden, July 2018.

Zbontar, J., Jing, L., Misra, I., LeCun, Y., and Deny, S. Barlow twins: Self-supervised learning via redundancy reduction. In *Proc. International Conference on Machine Learning*, pp. 12310–12320, July 2021.

Zhai, R., Liu, B., Risteski, A., Kolter, Z., and Ravikumar, P. Understanding augmentation-based self-supervised representation learning via rkhs approximation and regression. In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=Ax2yRhCQr1.

A. Proofs for Section 3

A.1. Proof of Theorem 3.1

Theorem 3.1. Let A be a one-hot random vector. Suppose the linear layer is unbiased, that is $\boldsymbol{b} = \boldsymbol{0}$. Then, Φ^* minimizes $\mathcal{R}(\Phi)$ if and only if it extracts the top-d eigenspace of $T_{P^+} \Lambda T_{P^+}^*$, where $k_\Lambda(a, a') = \mathbb{I}[a = a']$, or $(\Lambda g)(a) = g(a) P_{\mathcal{A}}(a)$. If all classes have the same size, then the top-d eigenfunctions of $T_{P^+} \Lambda T_{P^+}^*$ and $T_{P^+} T_{P^+}^*$ are the same.

The following lemma will be very useful in the proof.

Lemma A.1. *$T_{P^+} \Lambda T_{P^+}^*$ is the integral kernel operator of the following kernel*

$$k(x, x') = \iint k_\Lambda(a, a') P^+(a|x) P^+(a'|x') da da'.$$

Proof. By definition, we have

$$(T_{P^+}^* h)(a') = \int h(x') P^+(x'|a') dx'.$$

Thus we can get

$$\begin{aligned}
(\Lambda T_{P^+}^* h)(a) &= \int (T_{P^+}^* h)(a') k_\Lambda(a, a') P_{\mathcal{A}}(a') da' \\
&= \iint h(x') P^+(x'|a') k_\Lambda(a, a') P_{\mathcal{A}}(a') dx' da' \\
&= \iint h(x') P^+(a'|x') k_\Lambda(a, a') P_{\mathcal{X}}(x') dx' da'.
\end{aligned}$$

This implies that

$$\begin{aligned}
(T_{P^+} \Lambda T_{P^+}^* h)(x) &= \int (\Lambda T_{P^+}^* h)(a) P^+(a|x) da \\
&= \iiint h(x') k_\Lambda(a, a') P^+(a|x) P^+(a'|x') P_{\mathcal{X}}(x') da da' dx' \\
&= \int h(x') k(x, x') P_{\mathcal{X}}(x') dx',
\end{aligned}$$

as desired. □

Then, we finish the proof of Theorem 3.1.

Proof. For any fixed Φ, define

$$\mathcal{R}(\Phi, \boldsymbol{W}) = \mathbb{E}_{P^+} \left[\| A - \boldsymbol{W} \Phi(X) \|_2^2 \right] = \mathop{\mathbb{E}}_{X \sim P_{\mathcal{X}}} \mathop{\mathbb{E}}_{A \sim P^+(\cdot|X)} \left[\| A - \boldsymbol{W} \Phi(X) \|_2^2 \right].$$

Assuming, without loss of generality, that $\mathbb{E}_{X \sim P_{\mathcal{X}}}[\Phi_i \Phi_j] = \delta_{ij}$; otherwise one can perform Gram-Schmidt process on Φ_i and change the value of $\boldsymbol{W}$ respectively. Thus, it amounts to minimize

$$\begin{aligned}
\mathcal{R}(\Phi, \boldsymbol{W}) &= \mathop{\mathbb{E}}_{X \sim P_{\mathcal{X}}} \mathop{\mathbb{E}}_{A \sim P^+(\cdot|X)} \left[\| A - \boldsymbol{W} \Phi(X) \|_2^2 \right] \\
&= \mathop{\mathbb{E}}_{X \sim P_{\mathcal{X}}} \| \boldsymbol{W} \Phi(X) \|_2^2 - 2 \mathop{\mathbb{E}}_{X \sim P_{\mathcal{X}}} \mathop{\mathbb{E}}_{A \sim P^+(\cdot|X)} \langle A, \boldsymbol{W} \Phi(X) \rangle + \mathop{\mathbb{E}}_{A \sim P_{\mathcal{A}}} \| A \|_2^2 \\
&= \| \boldsymbol{W} \|_F^2 - 2 \mathop{\mathbb{E}}_{X \sim P_{\mathcal{X}}} \mathop{\mathbb{E}}_{A \sim P^+(\cdot|X)} \langle A, \boldsymbol{W} \Phi(X) \rangle + \mathop{\mathbb{E}}_{A \sim P_{\mathcal{A}}} \| A \|_2^2.
\end{aligned}$$

Denote $\boldsymbol{W} = (w_{ij})_{1 \le i \le d_A, 1 \le j \le d}$. We have

$$\frac{\partial \mathcal{R}}{\partial w_{ij}} = 2 w_{ij} - 2 \mathop{\mathbb{E}}_{X \sim P_{\mathcal{X}}} \mathop{\mathbb{E}}_{A \sim P^+(\cdot|X)} [A_i \Phi_j(X)],$$

which implies that for a fixed Φ, the optimal $\boldsymbol{W}$ that minimizes this loss should satisfy

$$w_{ij} = \mathop{\mathbb{E}}_{X \sim P_{\mathcal{X}}} \mathop{\mathbb{E}}_{A \sim P^+(\cdot|X)} [A_i \Phi_j(X)].$$

Combining the minimizer of $\boldsymbol{W}$ with $\mathcal{R}$ and notice that $\mathbb{E}_{A \sim P_{\mathcal{A}}} \|A\|_2^2$ is a constant, it suffices to **maximize**

$$\begin{aligned}
F(\Phi) &= \sum_{i,j} \left[\mathop{\mathbb{E}}_{X \sim P_{\mathcal{X}}} \mathop{\mathbb{E}}_{A \sim P^+(\cdot|X)} A_i \Phi_j(X) \right]^2 \\
&= \int \sum_j \Phi_j(x_1)\Phi_j(x_2) \langle a_1, a_2 \rangle P_{\mathcal{X}}(x_1) P^+(a_1|x_1) P_{\mathcal{X}}(x_2) P^+(a_2|x_2) dx_1 da_1 dx_2 da_2 \\
&= \iint \sum_j \Phi_j(x_1)\Phi_j(x_2) \hat{k}(x_1, x_2) P_{\mathcal{X}}(x_1) P_{\mathcal{X}}(x_2) dx_1 dx_2,
\end{aligned}$$

where

$$\begin{aligned}
\hat{k}(x_1, x_2) &= \iint \langle a_1, a_2 \rangle P^+(a_1|x_1) P^+(a_2|x_2) da_1 da_2 \qquad (4) \\
&= \iint \mathbb{I}[a_1 = a_2] P^+(a_1|x_1) P^+(a_2|x_2) da_1 da_2.
\end{aligned}$$

Thus Φ^* is a minimizer of $\mathcal{R}(\Phi)$ if Φ^* extracts the top-d eigenfunctions of $\hat{k}(x_1, x_2)$. Combining with Lemma A.1 yields that $k_\Lambda(a, a') = \mathbb{I}[a = a']$. Furthermore, we have $(\Lambda g)(a) = \int g(a') k_\Lambda(a, a') dP_{\mathcal{A}}(a') = g(a) P_{\mathcal{A}}(a)$, as desired.

If all classes have the same size, we have $P_{\mathcal{A}}(a) \equiv c \in (0, 1)$ where c is a constant. Thus $(\Lambda g)(a) = g(a) P_{\mathcal{A}}(a) = c g(a)$, which implies that $T_{P^+} \Lambda T^*_{P^+} = c T_{P^+} T^*_{P^+}$. This concludes that $T_{P^+} \Lambda T^*_{P^+}$ and $T_{P^+} T^*_{P^+}$ share the same top-d eigenfunctions. □

A.2. Proof of Theorem 3.2

Theorem 3.2. Under the setting of Theorem 3.1, let the linear layer be biased. Then, Φ^* minimizes $\mathcal{R}_{\text{bal}}(\Phi)$ if and only if it learns the contexture of P^+.

Proof. For any fixed Φ, define

$$\mathcal{R}(\Phi, \boldsymbol{W}) = \mathbb{E}_{P^+} \left[\frac{1}{\sqrt{P_{\mathcal{A}}(A)}} \|A - \boldsymbol{W}\Phi(X)\|_2^2 \right] = \mathop{\mathbb{E}}_{X \sim P_{\mathcal{X}}} \mathop{\mathbb{E}}_{A \sim P^+(\cdot|X)} \left[\frac{1}{\sqrt{P_{\mathcal{A}}(A)}} \|A - \boldsymbol{W}\Phi(X)\|_2^2 \right].$$

Assuming, without loss of generality,

$$\mathop{\mathbb{E}}_{X \sim P_{\mathcal{X}}} \mathop{\mathbb{E}}_{A \sim P^+(\cdot|X)} \left[\frac{1}{\sqrt{P_{\mathcal{A}}(A)}} \Phi_i \Phi_j \right] = \delta_{ij};$$

otherwise we can perform Gram-Schmidt process on Φ_i and change the value of $\boldsymbol{W}$ respectively. Thus it suffices to minimize

$$\begin{aligned}
\mathcal{R}(\Phi, \boldsymbol{W}) &= \mathop{\mathbb{E}}_{X \sim P_{\mathcal{X}}} \mathop{\mathbb{E}}_{A \sim P^+(\cdot|X)} \left[\frac{1}{\sqrt{P_{\mathcal{A}}(A)}} \|A - \boldsymbol{W}\Phi(X)\|_2^2 \right] \\
&= \mathop{\mathbb{E}}_{X \sim P_{\mathcal{X}}} \mathop{\mathbb{E}}_{A \sim P^+(\cdot|X)} \left[\frac{1}{\sqrt{P_{\mathcal{A}}(A)}} \|\boldsymbol{W}\Phi(X)\|_2^2 \right] \\
&\quad - 2 \mathop{\mathbb{E}}_{X \sim P_{\mathcal{X}}} \mathop{\mathbb{E}}_{A \sim P^+(\cdot|X)} \left\langle \frac{A}{\sqrt{P_{\mathcal{A}}(A)}}, \boldsymbol{W}\Phi(X) \right\rangle + \mathop{\mathbb{E}}_{A \sim P_{\mathcal{A}}} \left[\frac{\|A\|_2^2}{\sqrt{P_{\mathcal{A}}(A)}} \right] \\
&= \|\boldsymbol{W}\|_F^2 - 2 \mathop{\mathbb{E}}_{X \sim P_{\mathcal{X}}} \mathop{\mathbb{E}}_{A \sim P^+(\cdot|X)} \left\langle \frac{A}{\sqrt{P_{\mathcal{A}}(A)}}, \boldsymbol{W}\Phi(X) \right\rangle + \mathop{\mathbb{E}}_{A \sim P_{\mathcal{A}}} \left[\frac{\|A\|_2^2}{\sqrt{P_{\mathcal{A}}(A)}} \right].
\end{aligned}$$

Denote $\boldsymbol{W} = (w_{ij})_{1 \le i \le d_A, 1 \le j \le d}$. We have

$$\frac{\partial \mathcal{R}}{\partial w_{ij}} = 2w_{ij} - 2 \mathop{\mathbb{E}}_{X \sim P_{\mathcal{X}}} \mathop{\mathbb{E}}_{A \sim P^+(\cdot|X)} \left[\frac{A_i}{\sqrt{P_{\mathcal{A}}(A)}} \Phi_j(X) \right],$$

which implies that for a fixed Φ, the minimizer of $\boldsymbol{W}$ satisfies

$$w_{ij} = \mathop{\mathbb{E}}_{X \sim P_{\mathcal{X}}} \mathop{\mathbb{E}}_{A \sim P^+(\cdot|X)} \left[\frac{A_i}{\sqrt{P_{\mathcal{A}}(A)}} \Phi_j(X) \right].$$

Combining the minimizer of $\boldsymbol{W}$ with $\mathcal{R}$, it suffices to maximize

$$\mathcal{R}' = \sum_{i,j} \left[\mathop{\mathbb{E}}_{X \sim P_{\mathcal{X}}} \mathop{\mathbb{E}}_{A \sim P^+(\cdot|X)} \frac{A_i}{\sqrt{P_{\mathcal{A}}(A)}} \Phi_j(X) \right]^2 = \iint \sum_j \Phi_j(x_1)\Phi_j(x_2)\hat{k}(x_1,x_2)P_{\mathcal{X}}(x_1)P_{\mathcal{X}}(x_2)dx_1dx_2,$$

where

$$\begin{aligned}\hat{k}(x_1,x_2) &= \iint \frac{\langle a_1, a_2 \rangle}{\sqrt{P_{\mathcal{A}}(a_1)P_{\mathcal{A}}(a_2)}} P^+(a_1|x_1)P^+(a_2|x_2)da_1da_2 \\ &= \iint \frac{\mathbb{I}[a_1 = a_2]}{\sqrt{P_{\mathcal{A}}(a_1)P_{\mathcal{A}}(a_2)}} P^+(a_1|x_1)P^+(a_2|x_2)da_1da_2 \\ &= \int \frac{P^+(a|x_1)P^+(a|x_2)}{P_{\mathcal{A}}(a)} dy.\end{aligned}$$

Thus, Φ^* is a minimizer of $\mathcal{R}(\Phi)$ if Φ^* extracts the top-d eigenfunctions of $\hat{k}(x_1,x_2)$. Combining with Definitions 2.1 and 2.5 yields the desired results. □

A.3. Result for Regression

For regression where A is an arbitrary Euclidean vector, using the same objective as Eqn. (1), we can prove the following result.

Theorem A.2. Φ^* *minimizes Eqn.* (1) *if and only if* Φ^* *extracts the top-d eigenspace of* $T_{P^+}\Lambda T^*_{P^+}$. *If the linear layer is unbiased (*$\boldsymbol{b} = \boldsymbol{0}$*), then* $k_\Lambda(a,a') = \langle a, a' \rangle$*; if it is biased (*$\boldsymbol{b}$ *can be arbitrary), then* $k_\Lambda(a,a') = \langle \tilde{a}, \tilde{a'} \rangle$.

Remark A.3. Kernel $k_\Lambda(a,a') = \langle a, a' \rangle$ is called the **linear kernel** on $\mathcal{A}$, and $k_\Lambda(a,a') = \langle \tilde{a}, \tilde{a'} \rangle$ is called the **centered linear kernel** *w.r.t.* $P_{\mathcal{A}}$. Theorem 3.1 is a special case of Theorem A.2.

Proof. For the unbiased linear model, the proof is similar to that of Theorem 3.1. Combining Eqn. (4) and Lemma A.1 yields the desired result.

Next, we consider a biased linear model. For a variable z, we denote $\bar{z} = \mathbb{E}[Z]$, and $\tilde{z} = z - \mathbb{E}[Z]$ as its centered version.

For any fixed Φ, define

$$\begin{aligned}\mathcal{R}(\Phi, \boldsymbol{W}, \boldsymbol{b}) &= \mathop{\mathbb{E}}_{X \sim P_{\mathcal{X}}} \mathop{\mathbb{E}}_{A \sim P^+(\cdot|X)} \left[\| A - \boldsymbol{W}\Phi(X) - \boldsymbol{b} \|_2^2 \right] \\ &= \mathop{\mathbb{E}}_{X \sim P_{\mathcal{X}}} \mathop{\mathbb{E}}_{A \sim P^+(\cdot|X)} \left[\| A - \boldsymbol{W}\Phi(X) - \boldsymbol{b} \|_2^2 \right] \\ &= \mathop{\mathbb{E}}_{X \sim P_{\mathcal{X}}} \mathop{\mathbb{E}}_{A \sim P^+(\cdot|X)} \left[\left\| \tilde{A} - \boldsymbol{W}\tilde{\Phi}(X) - \hat{\boldsymbol{b}} \right\|_2^2 \right] \\ &= \mathop{\mathbb{E}}_{X \sim P_{\mathcal{X}}} \mathop{\mathbb{E}}_{A \sim P^+(\cdot|X)} \left[\left\| \tilde{A} - \boldsymbol{W}\tilde{\Phi}(X) \right\|_2^2 \right] + \left\| \hat{\boldsymbol{b}} \right\|_2^2\end{aligned}$$

where $\hat{\boldsymbol{b}} = \boldsymbol{W}\mathbb{E}_{X \sim P_{\mathcal{X}}}[\Phi(X)] - \mathbb{E}_{A \sim P_{\mathcal{A}}}[A] + \boldsymbol{b}$. Thus, for any fixed Φ, $\boldsymbol{W}$, the optimal $\boldsymbol{b} = \mathbb{E}_{A \sim P_{\mathcal{A}}}[A] - \boldsymbol{W}\mathbb{E}_{X \sim P_{\mathcal{X}}}[\Phi(X)]$.

Assuming, without loss of generality, $\mathbb{E}_{X \sim P_{\mathcal{X}}}[\tilde{\Phi}_i\tilde{\Phi}_j] = \delta_{ij}$; otherwise we can perform Gram-Schmidt process on $\tilde{\Phi}_i$ and change the value of $\boldsymbol{W}$ respectively. Thus, it suffices to minimize

$$\begin{aligned}
\hat{\mathcal{R}}(\Phi, \boldsymbol{W}) &= \mathop{\mathbb{E}}_{X \sim P_{\mathcal{X}}} \mathop{\mathbb{E}}_{A \sim P^+(\cdot|X)} \left[\left\| \tilde{A} - \boldsymbol{W} \tilde{\Phi}(X) \right\|_2^2 \right] \\
&= \mathop{\mathbb{E}}_{X \sim P_{\mathcal{X}}} \left\| \boldsymbol{W} \tilde{\Phi}(X) \right\|_2^2 - 2 \mathop{\mathbb{E}}_{X \sim P_{\mathcal{X}}} \mathop{\mathbb{E}}_{A \sim P^+(\cdot|X)} \left\langle \tilde{Y}, \boldsymbol{W} \tilde{\Phi}(X) \right\rangle + \mathop{\mathbb{E}}_{A \sim P_{\mathcal{A}}} \left\| \tilde{A} \right\|_2^2 \\
&= \| \boldsymbol{W} \|_F^2 - 2 \mathop{\mathbb{E}}_{X \sim P_{\mathcal{X}}} \mathop{\mathbb{E}}_{A \sim P^+(\cdot|X)} \left\langle \tilde{A}, \boldsymbol{W} \tilde{\Phi}(X) \right\rangle + \mathop{\mathbb{E}}_{A \sim P_{\mathcal{A}}} \left\| \tilde{A} \right\|_2^2 .
\end{aligned}$$

Denote $\boldsymbol{W} = (w_{ij})_{1 \le i \le d_y, 1 \le j \le d}$. We have

$$\frac{\partial \hat{\mathcal{R}}}{\partial w_{ij}} = 2 w_{ij} - 2 \mathop{\mathbb{E}}_{X \sim P_{\mathcal{X}}} \mathop{\mathbb{E}}_{A \sim P^+(\cdot|X)} \left[\tilde{A}_i \tilde{\Phi}_j(X) \right],$$

which implies that for a fixed Φ, the minimizer of $\boldsymbol{W}$ satisfies

$$w_{ij} = \mathop{\mathbb{E}}_{X \sim P_{\mathcal{X}}} \mathop{\mathbb{E}}_{A \sim P^+(\cdot|X)} \left[\tilde{A}_i \tilde{\Phi}_j(X) \right].$$

Combining the minimizer of $\boldsymbol{W}$ with $\hat{\mathcal{R}}$ and notice that $\mathbb{E}_{A \sim P_{\mathcal{A}}} \left\| \tilde{A} \right\|_2^2$ is a constant, it suffices to maximize

$$\begin{aligned}
\hat{\mathcal{R}}' &= \sum_{i,j} \left[\mathop{\mathbb{E}}_{X \sim P_{\mathcal{X}}} \mathop{\mathbb{E}}_{A \sim P^+(\cdot|X)} \tilde{A}_i \tilde{\Phi}_j(X) \right]^2 \\
&= \int \sum_j \tilde{\Phi}_j(x_1) \tilde{\Phi}_j(x_2) \langle \tilde{a}_1, \tilde{a}_2 \rangle P_{\mathcal{X}}(x_1) P^+(a_1|x_1) P_{\mathcal{X}}(x_2) P^+(a_2|x_2) dx_1 da_1 dx_2 da_2 \\
&= \iint \sum_j \tilde{\Phi}_j(x_1) \tilde{\Phi}_j(x_2) \hat{k}(x_1, x_2) P_{\mathcal{X}}(x_1) P_{\mathcal{X}}(x_2) dx_1 dx_2,
\end{aligned}$$

where

$$\hat{k}(x_1, x_2) = \iint \langle \tilde{a}_1, \tilde{a}_2 \rangle P^+(a_1|x_1) P^+(a_2|x_2) da_1 da_2.$$

Notice that

$$\begin{aligned}
\iint \hat{k}(x_1, x_2) P_{\mathcal{X}}(x_1) P_{\mathcal{X}}(x_2) dx_1 dx_2 &= \int \langle \tilde{a}_1, \tilde{a}_2 \rangle P^+(x_1, a_1) P^+(x_2, a_2) dx_1 da_1 dx_2 da_2 \\
&= \int \langle \tilde{a}_1, \tilde{a}_2 \rangle P_{\mathcal{A}}(a_1) P_{\mathcal{A}}(a_2) da_1 da_2 = 0,
\end{aligned}$$

thus Φ^* is a minimizer of $\mathcal{R}(\Phi)$ if $\tilde{\Phi}^*$ extracts the top-d eigenfunctions of $\hat{k}(x_1, x_2)$. Combining with Lemma A.1 yields the desired results. □

A.4. Proof of Theorem 3.4

Theorem 3.4. Ψ^* minimizes $\mathcal{L}_{\mathrm{C}}$ or $\mathcal{L}_{\mathrm{N}}$ if and only if $\tilde{\Phi}^* = T_{P^+} \tilde{\Psi}^*$ learns the contexture.

Proof. (i) The spectral contrastive loss is

$$\mathcal{L}_{\mathrm{C}}(\Psi) = \mathop{\mathbb{E}}_{X \sim P_{\mathcal{X}}} \mathop{\mathbb{E}}_{A, A^+ \sim P^+(\cdot|X)} \left[- \left\langle \tilde{\Psi}(A), \tilde{\Psi}(A^+) \right\rangle + \frac{1}{2} \mathop{\mathbb{E}}_{A^- \sim P_{\mathcal{A}}} \left[\left\langle \tilde{\Psi}(A), \tilde{\Psi}(A^-) \right\rangle^2 \right] \right].$$

Suppose $\psi_i = \sum_{j \ge 0} c_{ij} \nu_j$ where ν_j is the ONB of $L^2(P_{\mathcal{A}})$ in Lemma 2.3. Since ν_j is the ONB of $L^2(P_{\mathcal{A}})$ and $\nu_0 \equiv 1$, we can get for $j \ge 1$, $\mathbb{E}_{P_{\mathcal{A}}}[\nu_j(a)] = \delta_{0,j} = 0$. Thus we can get $\tilde{\psi}_i = \psi_i - \mathbb{E}[\psi_i] = \sum_{j \ge 1} c_{ij} \nu_j$.

Denote matrix $\boldsymbol{C} = (c_{ij})_{1 \le i \le d, j \ge 1}$, matrix $\boldsymbol{B} = (b_{ij}) := \boldsymbol{C}^\top \boldsymbol{C}$, and matrix $\boldsymbol{D} = \operatorname{diag}(s_1^2, s_2^2, \cdots)$ where s_i is the singular value of T_{P^+}. We have

$$\begin{aligned}
&\mathop{\mathbb{E}}_{X \sim P_{\mathcal{X}}} \mathop{\mathbb{E}}_{A, A^+ \sim P^+(\cdot|X)} \left[\left\langle \tilde{\Psi}(A), \tilde{\Psi}(A^+) \right\rangle \right] \\
= &\iiint \left\langle \tilde{\Psi}(a), \tilde{\Psi}(a^+) \right\rangle P^+(a|x) P^+(a^+|x) P_{\mathcal{X}}(x) dx da da^+ \\
= &\int \left\langle \int \tilde{\Psi}(a) P^+(a|x) dy, \int \tilde{\Psi}(a^+) P^+(a^+|x) da^+ \right\rangle p(x) dx \\
= &\int \left\langle T_{P^+} \tilde{\Psi}(x), T_{P^+} \tilde{\Psi}(x) \right\rangle p(x) dx = \| T_{P^+} \tilde{\Psi} \|_{P_{\mathcal{X}}}^2 \\
= &\sum_i s_i^2 b_{ii};
\end{aligned}$$

and

$$\begin{aligned}
\mathop{\mathbb{E}}_{A, A^- \sim P_{\mathcal{A}}} \left[\left\langle \tilde{\Psi}(A), \tilde{\Psi}(A^-) \right\rangle^2 \right] &= \iint \left[\sum_{i=1}^d \tilde{\psi}_i(a) \tilde{\psi}_i(a^-) \right]^2 dP_{\mathcal{A}}(a) dP_{\mathcal{A}}(a^-) \\
&= \sum_{1 \le i, j \le d} \left[\int \tilde{\psi}_i(a) \tilde{\psi}_j(a) dP_{\mathcal{A}}(a) \right]^2 \\
&= \sum_{i,j} b_{ij}^2.
\end{aligned}$$

Thus, we have

$$\mathcal{L}_{\mathrm{C}}(\Psi) = -\sum_i s_i^2 b_{ii} + \frac{1}{2} \sum_{i,j} b_{ij}^2 = \| \boldsymbol{B} - \boldsymbol{D} \|_F^2 - \| \boldsymbol{D} \|_F^2.$$

So if suffices to minimize $\| \boldsymbol{B} - \boldsymbol{D} \|_F^2$ where $\operatorname{rank}(\boldsymbol{B}) \le d$. By Eckart-Young-Mirsky Theorem, we know the minimizer of $\boldsymbol{B}$ is $\boldsymbol{B}^* = \operatorname{diag}(s_1^2, \cdots, s_d^2)$. Thus the minimizer of $\boldsymbol{C}$ should be $\boldsymbol{C}^* = \boldsymbol{U} \operatorname{diag}(s_1, \cdots, s_d)$ where $\boldsymbol{U} \in \mathbb{R}^{d \times d}$ is an orthonormal matrix. This indicates the minimizer $\tilde{\Psi}^*$ extracts the top-d singular functions of T_{P^+}, and $\tilde{\Phi}^*$ learns the contexture of P^+.

(ii) The non-contrastive loss is

$$\begin{aligned}
\mathcal{L}_{\mathrm{N}}(\Psi) &= \mathop{\mathbb{E}}_{X \sim P_{\mathcal{X}}} \mathop{\mathbb{E}}_{A, A^+ \sim P^+(\cdot|X)} \left[-\left\langle \tilde{\Psi}(A), \tilde{\Psi}(A^+) \right\rangle \right]; \\
\mathcal{L}'_{\mathrm{N}}(\Psi) &= \mathop{\mathbb{E}}_{X \sim P_{\mathcal{X}}} \mathop{\mathbb{E}}_{A, A^+ \sim P^+(\cdot|X)} \left[\left\| \Psi(A) - \Psi(A^+) \right\|_2^2 \right],
\end{aligned}$$

where $\operatorname{Cov}_{P_{\mathcal{A}}}[\Psi] = \boldsymbol{I}$. Since for any Ψ,

$$\mathcal{L}'_{\mathrm{N}}(\Psi) - \mathcal{L}_N(\Psi) = 2 \mathop{\mathbb{E}}_{A \sim P_{\mathcal{A}}} \left[\left\| \tilde{\Psi}(A) \right\|_2^2 \right] = 2d$$

is a constant, thus Ψ^* minimizes $\mathcal{L}_{\mathrm{N}}(\Psi) \iff \Psi^*$ minimizes $\mathcal{L}'_{\mathrm{N}}(\Psi)$.

Suppose $\psi_i = \sum_{j \ge 0} c_{ij} \nu_j$ where ν_j is the ONB of $L^2(P_{\mathcal{A}})$ in Lemma 2.4. Since $\mathbb{E}_{P_{\mathcal{A}}}[\nu_j(a)] = \delta_{0,j}$, we can get $\tilde{\psi}_i = \psi_i - \mathbb{E}[\psi_i] = \sum_{j \ge 1} c_{ij} \nu_j$.

We now consider the minimizer of $\mathcal{L}_{\mathrm{N}}(\Psi)$. By the calculation in (i), we obtain

$$\mathcal{L}_{\mathrm{N}}(\Psi) = -\mathop{\mathbb{E}}_{X \sim P_{\mathcal{X}}} \mathop{\mathbb{E}}_{A, A^+ \sim P^+(\cdot|X)} \left[\left\langle \tilde{\Psi}(A), \tilde{\Psi}(A^+) \right\rangle \right] = -\| T_{P^+} \tilde{\Psi} \|_{P_{\mathcal{X}}}^2 = -\sum_i s_i^2 b_{ii}.$$

By $\mathbb{E}_{P_{\mathcal{A}}} \left[\tilde{\psi}_i \tilde{\psi}_j \right] = \delta_{ij}$, we have

$$\sum_i b_{ii} = \sum_{i,j} c_{ij}^2 = d.$$

Since ν_i is an ONB of $L^2(P_{\mathcal{A}})$, $\tilde{\psi}_1, \cdots, \tilde{\psi}_d$ are orthogonal, we have

$$b_{ii} = \sum_{j=1}^d c_{ji}^2 = \sum_{j=1}^d \left\langle \tilde{\psi}_j, \nu_i \right\rangle_{P_{\mathcal{A}}}^2 \le \|\nu_i\|_{P_{\mathcal{A}}}^2 = 1. \tag{5}$$

Thus, we conclude that

$$\mathcal{L}_{\mathrm{N}}(\Psi) + \sum_{i=1}^d s_i^2 = \sum_{i=1}^d s_i^2(1 - b_{ii}) - \sum_{i>d} s_i^2 b_{ii} \ge \sum_{i=1}^d s_d^2(1 - b_{ii}) - \sum_{i>d} s_d^2 b_{ii} = 0,$$

which implies that $\mathcal{L}_{\mathrm{N}}(\Psi) \ge -\sum_{i=1}^d s_i^2$. To attain equality, we will have $b_{ii} = 1$ for $i = 1, \cdots, d$, and $b_{ii} = 0$ for $i \ge d+1$. By Eqn. (5), we can know Ψ^* extracts the span of $\nu_1, \cdots, \nu_d$, indicating that $\tilde{\Psi}^*$ extracts the top-d singular functions of T_{P^+} and $\tilde{\Phi}^*$ learns the contexture of P^+.

□

A.5. Result for Denoising Autoencoders

For denoising autoencoders, suppose $\mathcal{X} \subseteq \mathbb{R}^{d_X}$. Then, consider minimizing the following objective:

$$\mathcal{R}(\Psi) = \min_{\boldsymbol{W} \in \mathbb{R}^{d_X \times d},\, \boldsymbol{b} \in \mathbb{R}^{d_X}} \mathop{\mathbb{E}}_{(X,A) \sim P^+} \left[\|\boldsymbol{W}\Psi(A) + \boldsymbol{b} - X\|_2^2 \right]. \tag{6}$$

Theorem A.4. *Let Ψ^* be any minimizer of Eqn. (6). Then, $\tilde{\Psi}^*$ extracts the top-d eigenspace of $T_{P^+}^* \Lambda T_{P^+}$, where Λ is the integral operator of $k_\Lambda(x, x') = \langle \tilde{x}, \tilde{x}' \rangle$ if $\boldsymbol{b}$ can be an arbitrary vector, or $k_\Lambda(x, x') = \langle x, x' \rangle$ if $\boldsymbol{b} = \boldsymbol{0}$. Consequently, $\tilde{\Phi}^* = T_{P^+} \tilde{\Psi}^*$ extracts the top-d eigenspace of $T_{P^+} T_{P^+}^* \Lambda$.*

Proof. The proof is the same as Theorem A.2. □

A.6. Proof of Theorem 3.5

Theorem 3.5. Let Φ^* be any solution to Eqn. (2) (so that for any constant c, $\Phi^* + c$ is also a solution). Then, $\tilde{\Phi}^*$ learns the contexture of P^+.

Proof. Without loss of generality, suppose $\bar{\Phi} = \boldsymbol{0}$. We have

$$(T_{P^+} f)(u) = \sum_v f(v) \frac{w(u,v)}{d(u)}; \quad \langle T_{P^+} f, g \rangle_{P_{\mathcal{X}}} = \sum_{u,v} f(u) g(v) \frac{w(u,v)}{d_{\text{sum}}} = \langle f, T_{P^+} g \rangle_{P_{\mathcal{X}}},$$

which implies that T_{P^+} is self-adjoint. Therefore, the eigenfunctions of T_{P^+} are the same as those of $T_{P^+}^* T_{P^+}$, with square root eigenvalues.

For the objective of Eqn. (2), we have

$$\begin{aligned} \frac{1}{2} \mathbb{E}_{(u,v) \sim P_w} \left[\|\Phi(u) - \Phi(v)\|_2^2 \right] &= \mathop{\mathbb{E}}_{(u,v) \sim P_w} \left[\|\Phi(u)\|_2^2 - \langle \Phi(u), \Phi(v) \rangle \right] \\ &= \sum_{i=1}^d \left(\|\phi_i\|_{P_{\mathcal{X}}}^2 - \langle \phi_i, T_{P^+} \phi_i \rangle_{P_{\mathcal{X}}} \right) \\ &= d - \sum_{i=1}^d \langle \phi_i, T_{P^+} \phi_i \rangle_{P_{\mathcal{X}}}. \end{aligned}$$

Note that (u, v) and (v, u) can be drawn from P_w with equal probability. We conclude that Φ extracts the top-d eigenfunctions of T_{P^+}, which are the same as the top-d eigenfunctions of $T_{P^+}^* T_{P^+}$, or the top-d singular functions of T_{P^+}. This implies that $\tilde{\Phi}$ learns the contexture of T_{P^+}. □

B. Proofs for Section 4

B.1. Proof of Theorem 4.2

Theorem 4.2. For any $f^* \in \mathcal{F}_\epsilon(P^+)$, there exists a $g^* \in L^2(P_\mathcal{A})$ such that $f^*(x) = \mathbb{E}[g^*(A) \mid x]$, and g^* satisfies

$$\mathop{\mathbb{E}}_{X \sim P_\mathcal{X}} \mathop{\mathbb{E}}_{A, A' \sim P^+(\cdot \mid X)} \left[(g^*(A) - g^*(A'))^2 \right] \le 4\epsilon \|g^*\|_{P_\mathcal{A}}^2 . \tag{7}$$

Proof. Let $g^* = \sum s_i u_i \nu_i$. We have already explained that if $f^* \in \mathcal{F}_\epsilon(P^+)$, i.e., $\mathbb{E}[f^*] = 0$ and $\rho(f^*, P^+) \ge 1 - \epsilon$, then it must satisfy the condition *w.r.t.* g^*:

$$\frac{\langle f^*, T_{P^+} g^* \rangle_{P_\mathcal{X}}}{\|f^*\|_{P_\mathcal{X}} \|g^*\|_{P_\mathcal{A}}} \ge 1 - \epsilon.$$

For Eqn. (7), we have $P(A'|A = a) = \int P^+(A'|X = x) P^+(x|A' = a) dx$, where $P^+(x|a) = \frac{P^+(a|x) P_\mathcal{X}(x)}{P_\mathcal{A}(a)}$ by Bayes rule. Then, using Definition 2.1 we have $P(A'|A = a) = k_A^+(a, a') P_\mathcal{A}(a')$, which implies that

$$\begin{aligned} &\mathbb{E}_{X \sim P_\mathcal{X}} \mathbb{E}_{A, A' \sim P^+(\cdot|X)} [g^*(A) g^*(A')] = \mathbb{E}_{A \sim P_\mathcal{A}} \mathbb{E}_{A' \sim P(\cdot|A)} [g^*(A) g^*(A')] \\ = &\mathbb{E}_A \left[g^*(A) \int g^*(a') P(a'|A) da' \right] = \mathbb{E}_A \left[g^*(A) \int g^*(a') k_A^+(a, a') P_\mathcal{A}(a') da' \right] = \left\langle g^*, T_{k_A^+} g^* \right\rangle_{P_\mathcal{A}} . \end{aligned}$$

Since $T_{k_A^+} g^* = T_{P^+}^* T_{P^+} g^* = \sum s_i^3 u_i \nu_i$, Eqn. (7) is equivalent to $\sum (s_i^2 - s_i^4) u_i^2 \le 2\epsilon \sum s_i^2 u_i^2$. Meanwhile, we have $\sum s_i^2 u_i^2 \ge (1 - \epsilon)^2 \sum u_i^2 \ge (1 - 2\epsilon) \sum u_i^2$. By Cauchy-Schwarz inequality, we have $(\sum s_i^4 u_i^2)(\sum u_i^2) \ge (\sum s_i^2 u_i^2)^2 \ge (1 - 2\epsilon)(\sum u_i^2)(\sum s_i^2 u_i^2)$, which proves Eqn. (7). □

B.2. Proof of Theorem 4.4

Theorem 4.4. Suppose $1 - s_1 \le \epsilon \le 1 - \sqrt{\frac{s_1^2 + s_2^2}{2}}$. For any d, among all $\Phi = [\phi_1, \cdots, \phi_d]$ where $\phi_i \in L^2(P_\mathcal{X})$, Φ minimizes err$(\Phi; \mathcal{F}_\epsilon(P^+))$ if and only if it learns the contexture of T_{P^+}. The error is given by

$$\min_{\Phi : \mathcal{X} \to \mathbb{R}^d, \; \phi_i \in L^2(P_\mathcal{X})} \operatorname{err}\big(\Phi; \mathcal{F}_\epsilon(P^+)\big) = \frac{s_1^2 - (1 - \epsilon)^2}{s_1^2 - s_{d+1}^2}.$$

Conversely, for any d-dimensional encoder Φ and any $\epsilon > 0$, there exists $f \in L^2(P_\mathcal{X})$ such that $\rho(f, P^+) = 1 - \epsilon$, and err$(\Phi, f) \ge \frac{s_1^2 - (1 - \epsilon)^2}{s_1^2 - s_{d+1}^2}$.

Proof. **Necessity:** Since span(Φ) is at most rank-d, thus there exists $f_1 \in \text{span}\{\mu_1, \cdots, \mu_{d+1}\}$ with $\|f_1\|_{P_\mathcal{X}} = 1$ that is orthogonal to span(Φ). Thus there exists $f_1, f_2 \in \text{span}\{\mu_1, \cdots, \mu_{d+1}\}$ with $\|f_1\|_{P_\mathcal{X}} = \|f_2\|_{P_\mathcal{X}} = 1$, f_1 is orthogonal to span(Φ) and $f_2 \in \text{span}(\Phi)$ (thus $f_1 \perp f_2$), and $\mu_1 \in \text{span}\{f_1, f_2\}$. Suppose $\mu_1 = \alpha_1 f_1 + \alpha_2 f_2$ (without loss of generosity, assuming $\alpha_1, \alpha_2 \in [0, 1]$) and denote $f_0 = \alpha_2 f_1 - \alpha_1 f_2$. Then $\|f_0\|_{P_\mathcal{X}} = 1$ and $\langle \mu_1, f_0 \rangle_{P_\mathcal{X}} = 0$. Since $f_1, f_2 \in \text{span}\{\mu_1, \cdots, \mu_{d+1}\}$, we have $f_0 \in \text{span}\{\mu_2, \cdots, \mu_{d+1}\}$ and thus $\mathbb{E}[f_0] = 0$.

Consider $f = \beta_1 \mu_1 + \beta_2 f_0$ where $\beta_1^2 + \beta_2^2 = 1$, $\beta_1, \beta_2 \in [0, 1]$. Denote $f = \sum_{i \ge 1} u_i \mu_i$, we can get $\sum_i u_i^2 = 1$ and

$$\beta_2^2 \le \frac{s_1^2 - (1 - \epsilon)^2}{s_1^2 - s_{d+1}^2} \implies \sum_{i \ge 1} s_i^2 u_i^2 \ge s_1^2 \beta_1^2 + s_{d+1}^2 \beta_2^2 = s_1^2 - (s_1^2 - s_{d+1}^2) \beta_2^2 \ge (1 - \epsilon)^2 \sum_i u_i^2.$$

Obviously, $f \in \mathcal{F}(P^+)$. We have

$$f = \beta_1 \mu_1 + \beta_2 f_0 = \beta_1 (\alpha_1 f_1 + \alpha_2 f_2) + \beta_2 (\alpha_2 f_1 - \alpha_1 f_2) = (\alpha_1 \beta_1 + \alpha_2 \beta_2) f_1 + (\alpha_2 \beta_1 - \alpha_1 \beta_2) f_2.$$

By the definition of f_1, f_2 we can know the approximation error for f is $(\alpha_1 \beta_1 + \alpha_2 \beta_2)^2$. We can show $F(\alpha_1) = \alpha_1 \beta_1 + \alpha_2 \beta_2 = \alpha_1 \beta_1 + \sqrt{1 - \alpha_1^2} \beta_2$ ($\alpha_1 \in [0, 1]$) first increases then decreases when $\beta_1, \beta_2 \in [0, 1]$. Thus $F(\alpha_1)^2 \ge \min\{F(0)^2, F(1)^2\} = \min\{\beta_1^2, \beta_2^2\}$. Take $\beta_2^2 = \frac{s_1^2 - (1 - \epsilon)^2}{s_1^2 - s_{d+1}^2} \le \frac{1}{2}$, we can get for f, the approximation error is always at least $\frac{s_1^2 - (1 - \epsilon)^2}{s_1^2 - s_{d+1}^2}$.

To attain equality, we must have $\sum_{i \geq 1} s_i^2 u_i^2 = s_1^2 \beta_1^2 + s_{d+1}^2 \beta_2^2$. This implies that $f_1 = \mu_{d+1}$, indicating that $\operatorname{span}(\phi_1, \cdots, \phi_d) = \operatorname{span}(\mu_1, \cdots, \mu_d)$. Thus Φ learns the contexture of T_{P^+}.

Furthermore, denote $f_0 = k_2 \mu_2 + \cdots + k_{d+1} \mu_{d+1}$ and consider $f = \beta_1 \mu_1 + \beta_2 f_0$ where $\beta_1^2 + \beta_2^2 = 1, \beta_1, \beta_2 \in [0, 1]$. By the definition of f_0 and f, we have $\|f\|_{P_{\mathcal{X}}} = 1$ and $f = \beta_1 \mu_1 + \beta_2 \mu_2 = \beta_1 \mu_1 + \beta_2 k_2 \mu_2 + \cdots + \beta_2 k_{d+1} \mu_{d+1}$. Thus

$$\rho^2(f, P^+) = s_1^2 \beta_1^2 + \beta_2^2 \sum_{i=2}^{d+1} s_i^2 k_i^2 = s_1^2 - \left(s_1^2 - \sum_{i=2}^{d+1} s_i^2 k_i^2 \right) \beta_2^2.$$

Take

$$\beta_2^2 = \frac{s_1^2 - (1-\epsilon)^2}{s_1^2 - \left(\sum_{i=2}^{d+1} s_i^2 k_i^2 \right)} \leq \frac{s_1^2 - (1-\epsilon)^2}{s_1^2 - s_2^2} \leq \frac{s_1^2 - \frac{s_1^2 + s_2^2}{2}}{s_1^2 - s_2^2} = \frac{1}{2},$$

we have $\rho(f, P^+) = 1 - \epsilon$. Similarly, the approximation error for f is

$$(\alpha_1 \beta_1 + \alpha_2 \beta_2)^2 \geq \min\{\beta_1^2, \beta_2^2\} = \beta_2^2 = \frac{s_1^2 - (1-\epsilon)^2}{s_1^2 - \left(\sum_{i=2}^{d+1} s_i^2 k_i^2 \right)} \geq \frac{s_1^2 - (1-\epsilon)^2}{s_1^2 - s_{d+1}^2}.$$

Sufficiency: For any $f \in \mathcal{F}(P^+)$ with $\|f\|_{P_{\mathcal{X}}} = 1$ and $\mathbb{E}[f] = 0$, denote $f = \sum_{i \geq 1} u_i \mu_i$ where $\sum_{i \geq 1} u_i^2 = 1$. Obviously we have $(1-\epsilon)^2 \leq \sum_{i \geq 1} s_i^2 u_i^2 \leq 1$. Notice that when $\operatorname{span}(\phi_1, \cdots, \phi_d) = \operatorname{span}(\mu_1, \cdots, \mu_d)$ since Φ learns the contexture of T_{P^+}, the approximation of f will be $\sum_{i \geq d+1} u_i^2 := A$. By the given conditions, we have

$$(1-\epsilon)^2 \leq \sum_{i \geq 1} s_i^2 u_i^2 \leq s_1^2 \sum_{i=1}^{d} u_i^2 + s_{d+1}^2 \sum_{i \geq d+1} u_i^2 = s_1^2 - (s_1^2 - s_{d+1}^2) A,$$

and this implies that

$$A = \min_{\boldsymbol{w} \in \mathbb{R}^d, \, b \in \mathbb{R}} \left\| \boldsymbol{w}^\top \Phi + b - f \right\|_{P_{\mathcal{X}}}^2 \leq \frac{s_1^2 - (1-\epsilon)^2}{s_1^2 - s_{d+1}^2}.$$

When $u_1^2 = 1 - \frac{s_1^2 - (1-\epsilon)^2}{s_1^2 - s_{d+1}^2}, u_{d+1}^2 = \frac{s_1^2 - (1-\epsilon)^2}{s_1^2 - s_{d+1}^2}$, the equality holds. Thus the approximation error reaches its lower bound when Φ learns the contexture of T_{P^+}. □

C. Evaluating an Arbitrary Encoder

Given a context that is compatible with the task, the encoder that learns the contexture is optimal. Now, what about an arbitrary encoder Φ? Is it possible to bound its worst-case approximation error on the class of compatible tasks? To derive such a bound, two key objects are necessary: the induced RKHS and the ratio trace. They were originally defined in (Zhai et al., 2024) for self-supervised learning, and here we extend them to a broader scope.

Denote the range of $T_{P^+}^*$ by $R(T_{P^+}^*) = \left\{ T_{P^+}^* f \;\middle|\; f \in L^2(P_{\mathcal{X}}) \right\}$.

Definition C.1. The **induced RKHS** of P^+, denoted by $\mathcal{H}_{P^+}$, is the Hilbert space $R(T_{P^+}^*)$ with the inner product given by $\left\langle T_{P^+}^* f_1, T_{P^+}^* f_2 \right\rangle_{\mathcal{H}_{P^+}} = \langle f_1, f_2 \rangle_{P_{\mathcal{X}}}$.

An alternative formula is that for any $h_1, h_2 \in \mathcal{H}_{P^+}$ where $h_1 = \sum u_i \nu_i$ and $h_2 = \sum v_i \nu_i$, there is $\langle h_1, h_2 \rangle_{\mathcal{H}_{P^+}} = \sum \frac{u_i v_i}{s_i^2}$.

Proposition C.2. *The induced RKHS $\mathcal{H}_{P^+}$ has the following properties:*

(i) k_A^+ is the reproducing kernel, such that $h(a) = \left\langle h, k_A^+(a, \cdot) \right\rangle_{\mathcal{H}_{P^+}}$ for all $h \in \mathcal{H}_{P^+}$.

(ii) $\mathcal{H}_{P^+}$ is isometric to $\operatorname{span}\{\mu_i : s_i > 0\}$*, which is a subspace of $L^2(P_{\mathcal{X}})$.*

(iii) $f^* \in \mathcal{F}_\epsilon(P^+)$ is equivalent to $h^* = T_{P^+}^* f^*$ satisfying the following ***isometry property****:*

$$(1-\epsilon) \left\| \tilde{h}^* \right\|_{\mathcal{H}_{P^+}} \leq \left\| \tilde{h}^* \right\|_{P_{\mathcal{A}}} \leq \left\| \tilde{h}^* \right\|_{\mathcal{H}_{P^+}}. \tag{8}$$

Proof. For any $h \in \mathcal{H}_{P^+}$ where $h = T_{P^+}^* f$ and $f = \sum u_i \mu_i$, we have

$$\left\langle h, k_A^+(a,\cdot) \right\rangle_{\mathcal{H}_{P^+}} = \left\langle \sum s_i u_i \nu_i, \sum s_i^2 \nu_i(a) \nu_i \right\rangle_{\mathcal{H}_{P^+}} = \sum s_i u_i \nu_i(a) = h(a),$$

which proves (i). (ii) is obvious. Regarding (iii), recall that $f^* = \sum u_i \mu_i \in \mathcal{F}_\epsilon(P^+)$ is equivalent to $\sum_{i \geq 1} s_i^2 u_i^2 \geq (1-\epsilon)^2 \sum_{i \geq 1} u_i^2$, and this is $\left\| \tilde{h}^* \right\|_{P_{\mathcal{A}}} \geq (1-\epsilon) \left\| \tilde{h}^* \right\|_{\mathcal{H}_{P^+}}$. Meanwhile, it is obvious that $\left\| \tilde{h}^* \right\|_{P_{\mathcal{A}}} \leq \left\| \tilde{h}^* \right\|_{\mathcal{H}_{P^+}}$ always holds. □

Definition C.3. Define covariance matrices $\boldsymbol{C}_\Phi = \mathrm{Cov}_{P_{\mathcal{X}}}[\Phi]$, and $\boldsymbol{B}_\Phi = \mathrm{Cov}_{P_{\mathcal{A}}}[T_{P^+}^* \Phi]$. If $\boldsymbol{C}_\Phi$ is invertible, then the **ratio trace** of Φ *w.r.t.* P^+ is defined as $\mathrm{RT}(\Phi; P^+) = \mathrm{RT}(\phi_1, \cdots, \phi_d; P^+) := \mathrm{Tr}(\boldsymbol{C}_\Phi^{-1} \boldsymbol{B}_\Phi)$; otherwise, let $\Phi' = [\phi_{i_1}, \cdots, \phi_{i_t}]$ be the maximal linearly independent subset of $[\phi_1, \cdots, \phi_d]$, and define the ratio trace of Φ the same as the ratio trace of Φ'.

The ratio trace of any Φ essentially measures how well Φ is aligned with the contexture of P^+. Multiplying Φ by any invertible matrix does not change its ratio trace. If Φ learns the contexture, then its ratio trace is $s_1^2 + \cdots + s_d^2$, which can be easily shown by setting $\phi_i = \mu_i$. In fact, this is the maximum ratio trace of any d-dimensional encoder.

Lemma C.4. *Suppose $\phi_1, \cdots, \phi_d$ are orthonormal and all have zero mean. Then, we have*

$$\|T_{P^+}^* \phi_1\|_{P_{\mathcal{A}}}^2 + \cdots + \|T_{P^+}^* \phi_d\|_{P_{\mathcal{A}}}^2 \leq s_1^2 + \cdots + s_d^2.$$

Proof. Let $\phi_i = \sum_{j \geq 1} q_{ij} \mu_j$ for $i \in [d]$. Then, $\boldsymbol{Q} = (q_{ij})$ is a matrix with d orthonormal rows and infinitely many columns. It is easy to see that the left-hand side is equal to $\mathrm{Tr}(\boldsymbol{Q}\boldsymbol{D}\boldsymbol{Q}^\top)$, where $\boldsymbol{D} = \mathrm{diag}\{s_1^2, s_2^2, \cdots\}$. Let $\boldsymbol{q}_j$ be the j-th column of $\boldsymbol{Q}$. For all $j \in [d]$, there is $\sum_{i=1}^j \boldsymbol{q}_i^\top \boldsymbol{q}_i \leq j$; and for any $j > d$, there is $\sum_{i=1}^j \boldsymbol{q}_i^\top \boldsymbol{q}_i \leq d$. Thus, using Abel transformation, we have

$$\mathrm{Tr}(\boldsymbol{Q}\boldsymbol{D}\boldsymbol{Q}^\top) = \mathrm{Tr}(\boldsymbol{D}\boldsymbol{Q}^\top \boldsymbol{Q}) = \sum_{j=1}^\infty s_j^2 \boldsymbol{q}_j^\top \boldsymbol{q}_j = \sum_{j=1}^\infty \left(\sum_{i=1}^j \boldsymbol{q}_i^\top \boldsymbol{q}_i \right) \left(s_j^2 - s_{j+1}^2 \right) \leq \sum_{j=1}^d s_j^2,$$

as desired. □

The ratio trace induces a key quantity in the approximation error bound called the trace gap, which reflects the gap between Φ and the top-d singular functions. The larger the trace gap is, the larger the approximation error will be. A simple definition is $s_1^2 + \cdots + s_{d+1}^2 - \mathrm{RT}(\Phi; P^+)$, whose lower bound s_{d+1}^2 can be achieved by the top-d singular functions, the optimal encoder. However, there is an issue with this definition. For example, consider an encoder with $d = 1000$. It learns the top-10 singular functions, but the other 990 dimensions are complete noise that has zero contribution to $\mathrm{RT}(\Phi; P^+)$. The approximation error of this encoder should be no higher than that of the top-10 singular functions, because adding more dimensions will never make the approximation error higher. However, if d becomes larger and $\mathrm{RT}(\Phi; P^+)$ stays the same, then $s_1^2 + \cdots + s_{d+1}^2 - \mathrm{RT}(\Phi; P^+)$ will become larger, so this quantity does not correlate with the approximation error in this scenario. The following definition fixes this issue.

Definition C.5. For any linearly independent $f_1, \cdots, f_{d'} \in L^2(P_{\mathcal{X}})$, denote $F = [f_1, \cdots, f_{d'}]$, $\boldsymbol{C}_F = \mathrm{Cov}_{P_{\mathcal{X}}}[F]$, and $\boldsymbol{B}_F = \mathrm{Cov}_{P_{\mathcal{A}}}[F]$. The **trace gap** of Φ *w.r.t.* P^+ is defined as

$$\mathrm{TG}(\Phi; P^+) := \inf_{d' \leq d} \inf_{f_1, \cdots, f_{d'}} \left\{ s_1^2 + \cdots + s_{d'+1}^2 - \mathrm{Tr}(\boldsymbol{C}_F^{-1} \boldsymbol{B}_F) \right\}.$$

Obviously, this definition of trace gap is upper bounded by $s_1^2 + \cdots + s_{d+1}^2 - \mathrm{RT}(\Phi; P^+)$. It solves the issue in the previous example because having completely noisy dimensions does not affect the trace gap. The following result bounds the approximation error.

Theorem C.6. *Suppose* $\mathrm{TG}(\Phi; P^+) < s_1^2$, *and* $\epsilon > 1 - s_1$. *Then,*

$$\mathrm{err}(\Phi; \mathcal{F}_\epsilon(P^+)) \leq \frac{s_1^2 - (1-\epsilon)^2 + s_1 \mathrm{TG}(\Phi; P^+)}{s_1^2 - \mathrm{TG}(\Phi; P^+)^2}.$$

Remark C.7. This bound is fairly tight. If Φ learns the contexture, then by Theorem 4.4 we have $\mathrm{err}(\Phi; \mathcal{F}_\epsilon(P^+)) = \frac{s_1^2 - (1-\epsilon)^2}{s_1^2 - s_{d+1}^2}$, and $\mathrm{TG}(\Phi; P^+) = s_{d+1}$. Compared to this exact formula, the above upper bound only has an extra $s_1 \mathrm{TG}(\Phi; P^+)$ term in the numerator.

Proof. Let $f_1, \cdots, f_{d'}$ be the functions that minimize $s_1^2 + \cdots + s_{d'+1}^2 - \operatorname{Tr}(\boldsymbol{C}_F^{-1} \boldsymbol{B}_F)$. Without loss of generality, assume that $f_1, \cdots, f_{d'}$ have zero mean and are orthonormal. Let $\mathcal{F} = \operatorname{span}\{f_1, \cdots, f_{d'}\}$, and $\mathcal{H} = \operatorname{span}\left\{T_{P^+}^* f_1, \cdots, T_{P^+}^* f_{d'}\right\}$. For any $f \in \mathcal{F}_\epsilon(P^+)$ with $\|f\|_{P_\mathcal{X}} = 1$, let $h = T_{P^+}^* f \in \mathcal{H}_{P^+}$, and let $f_\mathcal{F}$ be the projection of f onto $\mathcal{F}$. Since $\operatorname{err}(\Phi; \mathcal{F}_\epsilon(P^+))$ is upper bounded by $\|f - f_\mathcal{F}\|_{P_\mathcal{X}}^2$, it suffices to show that $\|f - f_\mathcal{F}\|_{P_\mathcal{X}}^2$ is upper bounded by the right-hand side.

Let $\alpha^2 = \|f_\mathcal{F}\|_{P_\mathcal{X}}^2$, and $\beta^2 = \|f - f_\mathcal{F}\|_{P_\mathcal{X}}^2$, where α and β are non-negative. Then, $\alpha^2 + \beta^2 = \|f\|_{P_\mathcal{X}}^2 = 1 = \|h\|_{\mathcal{H}_{P^+}}^2$. The isometry property says that $(1-\epsilon)^2(\alpha^2 + \beta^2) \le \|h\|_{P_\mathcal{A}}^2$. Let $f - f_\mathcal{F} = \beta f_0$ where $\|f_0\|_{P_\mathcal{X}} = 1$. Let $h_\mathcal{F} = T_{P^+}^* h_\mathcal{F}$ and $h_0 = T_{P^+}^* f_0$. Then, we have $\|h_\mathcal{F}\|_{P_\mathcal{A}}^2 \le s_1^2 \|f_\mathcal{F}\|_{P_\mathcal{X}}^2 = s_1^2 \alpha^2$. Meanwhile, since f_0 is orthogonal to $f_1, \cdots, f_{d'}$, by Lemma C.4 we have $\left\|T_{P^+}^* f_0\right\|_{P_\mathcal{A}}^2 + \left\|T_{P^+}^* f_1\right\|_{P_\mathcal{A}}^2 + \cdots + \left\|T_{P^+}^* f_{d'}\right\|_{P_\mathcal{A}}^2 \le s_1^2 + \cdots + s_{d'+1}^2$, which implies that $\left\|T_{P^+}^* f_0\right\|_{P_\mathcal{A}}^2 \le s_1^2 + \cdots + s_{d'+1}^2 - \operatorname{Tr}(\boldsymbol{C}_F^{-1} \boldsymbol{B}_F^{-1})$. Let $\tau = \operatorname{TG}(\Phi; P^+)$. Then, we have

$$\|h\|_{P_\mathcal{A}}^2 = \|h_\mathcal{F} + \beta h_0\|_{P_\mathcal{A}}^2 \le \|h_\mathcal{F}\|_{P_\mathcal{A}}^2 + \beta^2 \|h_0\|_{P_\mathcal{A}}^2 + 2\beta \|h_\mathcal{F}\|_{P_\mathcal{A}} \|h_0\|_{P_\mathcal{A}} \le s_1^2 \alpha^2 + \tau^2 \beta^2 + 2 s_1 \tau \alpha \beta.$$

Thus, we have $(1-\epsilon)^2(\alpha^2 + \beta^2) \le s_1^2 \alpha^2 + \tau^2 \beta^2 + 2 s_1 \tau \alpha \beta$, which implies that $(s_1^2 - \tau^2)\beta^2 \le [s_1^2 - (1-\epsilon)^2](\alpha^2 + \beta^2) + 2 s_1 \tau \alpha \beta \le [s_1^2 - (1-\epsilon)^2 + s_1 \tau](\alpha^2 + \beta^2)$, as desired. □

Connection to Fisher discriminant analysis. Fisher discriminant analysis (Mika et al., 1999; Baudat & Anouar, 2000; Liu et al., 2004), or more generally linear discriminant analysis (LDA), is a classical method of learning linear classifiers in statistics. Here we show that Fisher discriminant analysis has a strong connection to the contexture theory. Suppose $\mathcal{X} \subseteq \mathbb{R}^{d_\mathcal{X}}$. Fisher discriminant analysis defines the following **between-class covariance matrix** $\boldsymbol{S}_B \in \mathbb{R}^{d_\mathcal{X} \times d_\mathcal{X}}$ and **within-class covariance matrix** $\boldsymbol{S}_W \in \mathbb{R}^{d_\mathcal{X} \times d_\mathcal{X}}$:

$$\boldsymbol{S}_B = \iint \left\{ (\mathbb{E}[X \mid A = a_1] - \mathbb{E}[X \mid A = a_2])(\mathbb{E}[X \mid A = a_1] - \mathbb{E}[X \mid A = a_2])^\top \right\};$$
$$\boldsymbol{S}_W = \int \mathbb{E}_{P^+}\left[(X - \mathbb{E}[X \mid A = a])(X - \mathbb{E}[X \mid A = a])^\top \,\middle|\, A = a \right] dP_\mathcal{A}(a).$$

In the original formulation of Fisher discriminant analysis, A is the label of X. Here we extend it to a general context variable. Consider a linear encoder $\Phi(x) = \boldsymbol{W}x$, where $\boldsymbol{W} \in \mathbb{R}^{d \times d_\mathcal{X}}$. Then, one solves the following optimization problem to find $\boldsymbol{W}$:

$$\underset{\boldsymbol{W} \in \mathbb{R}^{d \times d_\mathcal{X}}}{\text{maximize}} \; J(\boldsymbol{W}) = \operatorname{Tr}\left[\left(\boldsymbol{W} \boldsymbol{S}_B \boldsymbol{W}^\top \right) \left(\boldsymbol{W} \boldsymbol{S}_W \boldsymbol{W}^\top \right)^{-1} \right] \quad \text{s.t.} \quad \boldsymbol{W} \boldsymbol{S}_W \boldsymbol{W}^\top \text{ is invertible.}$$

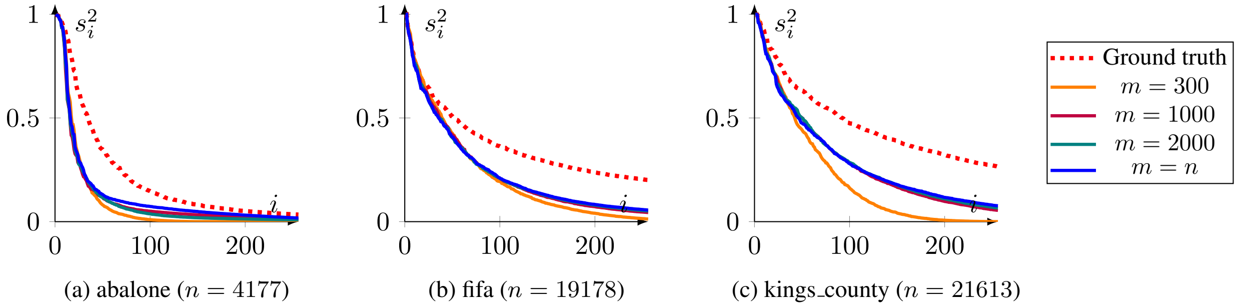
Here, $J(\boldsymbol{W})$ is called the **Fisher discriminant**. Define $\Psi(a) = \mathbb{E}_{P^+}[\boldsymbol{W}X | A = a]$. Then, we can see that

$$\boldsymbol{W} \boldsymbol{S}_B \boldsymbol{W}^\top = \iint (\Psi(a_1) - \Psi(a_2))(\Psi(a_1) - \Psi(a_2))^\top dP_\mathcal{A}(a_1) dP_\mathcal{A}(a_2);$$
$$\boldsymbol{W} \boldsymbol{S}_W \boldsymbol{W}^\top = \int \mathbb{E}_{P^+}\left[(\Phi(X) - \Psi(a))(\Phi(X) - \Psi(a))^\top \,\middle|\, A = a \right] dP_\mathcal{A}(a).$$

Let $\boldsymbol{C}_\Phi = \mathbb{E}[\tilde{\Phi}(X)\tilde{\Phi}(X)^\top]$ and $\boldsymbol{B}_\Phi = \mathbb{E}[\tilde{\Psi}(A)\tilde{\Psi}(A)^\top]$. Then, we have

$$\begin{aligned} \boldsymbol{W} \boldsymbol{S}_B \boldsymbol{W}^\top &= 2\left\{ \mathbb{E}\left[\Psi(A)\Psi(A)^\top\right] - \bar{\Psi}\bar{\Psi}^\top \right\} = 2\mathbb{E}\left[\tilde{\Psi}(A)\tilde{\Psi}(A)^\top \right] = 2\boldsymbol{B}_\Phi; \\ \boldsymbol{W} \boldsymbol{S}_W \boldsymbol{W}^\top &= \int \mathbb{E}_{P^+}\left[\Phi(X)\Phi(X)^\top - \Psi(a)\Psi(a)^\top \mid A = a \right] dP_\mathcal{A}(a) \\ &= \mathbb{E}\left[\Phi(X)\Phi(X)^\top\right] - \mathbb{E}\left[\Psi(A)\Psi(A)^\top\right] \\ &= \mathbb{E}\left[\tilde{\Phi}(X)\tilde{\Phi}(X)^\top \right] - \mathbb{E}\left[\tilde{\Psi}(A)\tilde{\Psi}(A)^\top \right] = \boldsymbol{C}_\Phi - \boldsymbol{B}_\Phi. \end{aligned}$$

Therefore, $J(\boldsymbol{W}) = 2\operatorname{Tr}[(\boldsymbol{C}_\Phi - \boldsymbol{B}_\Phi)^{-1} \boldsymbol{B}_\Phi]$, which is very similar to the ratio trace defined in Definition C.3. Recall that an encoder that learns the contexture maximizes the ratio trace. A well-known result is that $J(\boldsymbol{W})$ is maximized when $\boldsymbol{W}$ consists of the top-d eigenvectors of $\boldsymbol{S}_W^{-1} \boldsymbol{S}_B$. Hence, Fisher discriminant analysis is almost equivalent to contexture learning under the constraint that the encoder must be linear.



(a) abalone ($n = 4177$) (b) fifa ($n = 19178$) (c) kings_county ($n = 21613$)

Figure 5. Estimating the eigenvalues using a random subset of m samples, from n total training samples.

D. Efficient Estimation of the Context Usefulness Metric

To estimate the metric τ_d defined in Eqn. (3), it suffices to estimate the top-d_0 eigenvalues of the context. This can be efficiently done with the following procedure:

(i) Train an encoder Φ whose output dimension is at least d_0 to learn the contexture with a random subset of m samples.

(ii) Estimate the covariance matrix $\boldsymbol{C}_\Phi \in \mathbb{R}^{d\times d} = \mathrm{Cov}_{P_\mathcal{X}}[\Phi]$ with Monte Carlo.

(iii) Estimate $\boldsymbol{B}_\Phi \in \mathbb{R}^{d\times d}$, where $\boldsymbol{B}_\Phi[i,j] = \left\langle \tilde{\phi}_i, T_{k_X^+}\tilde{\phi}_j \right\rangle_{P_\mathcal{X}}$, with Monte Carlo.

(iv) Solve the generalized eigenvalue problem $\boldsymbol{B}_\Phi \boldsymbol{v} = \lambda \boldsymbol{C}_\Phi \boldsymbol{v}$. The eigenvalues $\lambda_1 \geq \cdots \geq \lambda_{d_0}$ are estimates of $s_1^2, \cdots, s_{d_0}^2$. Moreover, let $\boldsymbol{Q} = [\boldsymbol{v}_1, \cdots, \boldsymbol{v}_d]$ where $(\boldsymbol{v}_i)$ are the orthonormal eigenvectors corresponding to (λ_i). Let Φ^* be the normalized version of $\tilde{\Phi}\boldsymbol{Q}$ such that each dimension is scaled to unit variance. Then, $\phi_1^*, \cdots, \phi_{d_0}^*$ are estimates of $\mu_1, \cdots, \mu_{d_0}$.

If we only need to estimate the eigenvalues, then (Shawe-Taylor et al., 2005) showed that for any fixed d, the sum $s_1^2 + \cdots + s_d^2$ can be estimated with low error using $\Theta(d)$ *i.i.d.* samples. By union bound, all $s_1^2, \cdots, s_{d_0}^2$ can be estimated with low error using $m = \Theta(d_0 \log d_0)$ *i.i.d.* samples. However, if we want to estimate the eigenfunctions as well, then usually we need to use the entire training set.

Let us demonstrate this method on 3 real datasets from OpenML (Vanschoren et al., 2013): `abalone`, `fifa`, and `kings_county`. We use KNN with $K = 60$ as context, where $\mathcal{A} = \mathcal{X}$, and $P^+(x'|x) = K^{-1}$ if x' is a K-nearest neighbor of x and 0 otherwise. For this context, we can exactly compute k_X^+, and thus we can obtain the exact eigenvalues (ground truth) using kernel PCA. Meanwhile, we pretrain Φ with one of the variational objectives using a random subset of m samples, and estimate the eigenvalues using the post-hoc approach. Then, we compare the estimation with the ground truth.

We use a 2-layer wide Tanh-activated neural network with embedding dimension $d = 512$ and hidden dimension 20,000 as Φ. We train the model through non-contrastive learning with the orthonormality constraint implemented by VICReg (Bardes et al., 2022), and AdamW (Kingma & Ba, 2015; Loshchilov & Hutter, 2017) as the optimizer. We vary m and compare the estimated top-d_0 eigenvalues with the ground truth, where $d_0 = 256$. The estimated eigenvalues and the ground truth are plotted in Figure 5. From the plots, we observe that the eigenvalues estimated by our estimation method decay faster than the ground truth, even if the full dataset is used. We hypothesize that the main reason is that even though we use a very wide neural network, its function class is still a subset of $L^2(P_\mathcal{X})$. Consequently, the inductive bias of the model architecture has an impact on the encoder, and therefore the learned contexture can be viewed as a mixture of the inductive bias and the original KNN context. This mixture causes the eigenvalues to decay faster, which explains the observation in Figure 5. Another reason is related to optimization. Since the model is non-convex, gradient methods cannot find the minima of the objective.

The average estimation error of the top-256 eigenvalues is reported in Table 2. The error is defined as $\frac{1}{d_0}\sum_{i=1}^{d_0} |\hat{\lambda}_i - s_i^2|$, where $\hat{\lambda}_i$ is the estimated eigenvalue. The table shows that when $m \in [600, 1000] \approx [0.5 d_0 \log d_0, 0.7 d_0 \log d_0]$, the

Dataset	$m = 100$	$m = 300$	$m = 600$	$m = 1000$	$m = 2000$	Full dataset
abalone	0.157	0.124	0.088	0.104	0.110	0.088
fifa	0.218	0.151	0.137	0.134	0.133	0.131
kings_county	0.278	0.264	0.190	0.183	0.177	0.177

Table 2. Average estimation error of the top-256 eigenvalues.

performance is comparable to using the full dataset, which verifies the theoretical result of (Shawe-Taylor et al., 2005). The estimation error is not zero even if the full dataset is used due to the aforementioned reasons. In summary, the post-hoc method can estimate the eigenvalues using a small subset of samples, but the estimated eigenvalues decay faster than the ground truth.

E. Scaling Law Experiment Details

Here we provide a more detailed description of the experiment setting in Section 4.2.

Experiment overview. The purpose of this experiment is to examine whether a large neural network can learn the contexture well, and whether scaling up the model size makes the learned representation more aligned to the top-d eigenfunctions. We compare two encoders. The first encoder is obtained via kernel PCA on the dual kernel, so it consists of the exact top-d eigenfunctions. The second encoder is obtained via training a large neural network to optimize an objective that can learn the contexture. Then, we compute the representational alignment of these two encoders. The most classical metric is the canonical-correlation analysis (CCA) metric R^2_{CCA}, which is invariant under invertible linear transformations to the encoders. (Kornblith et al., 2019) proposed a variant called linear CKA, which is only invariant under orthogonal transformations. In our setting, since we only care about the span of $\phi_1, \cdots, \phi_d$, we would like the metric to be invariant under all invertible transformations, which is why we use CCA. In addition, we also use the mutual KNN metric with 10 neighbors proposed by (Huh et al., 2024), which measures the intersection over union (IoU) of nearest neighbors between the two representations. This metric is not invariant under invertible linear transformations, so we whiten the two representations such that their covariance matrices are both identities.

Setup. We use the abalone dataset from OpenML, and split the dataset into a pretrain set, a downstream train set and a downstream test set by 70%-15%-15%. We use K-nearest neighbors (KNN) with $K = 30$ as the context. The embedding dimension is set to $d = 128$. For the second encoder, we train a fully-connected neural network with Tanh activation and skip connections for a sufficient number of steps with full-batch AdamW, and vary the depth and width of the network so that we can study their effect on the alignment. Here, “depth” refers to the number of hidden layers—for example, a 2-layer neural network has depth 1. For each width and depth, we run the experiments 15 times with different random initializations and report the average alignment.

In our experiments, we observe the **dimension collapse** problem (Jing et al., 2022)—if we set the output dimension of the neural network to be d, then the rank of the learned representation will usually be less than d, meaning that it can only extract the top-d' eigenspace for some $d' < d$. (Jing et al., 2022) proved that the training dynamics of self-supervised learning can cause this problem, that is, a large neural network trained with a gradient method cannot find the exact minima, but will find a low-rank solution instead.

To fix this issue, we set the output dimension of the neural network to be $d_1 = 512 > d$. After we obtain the d_1-dimensional encoder, similar to Appendix D we estimate the matrices $\boldsymbol{C}_\Phi$ and $\boldsymbol{B}_\Phi$, and solve the generalized eigenvalue problem $\boldsymbol{B}_\Phi \boldsymbol{v} = \lambda \boldsymbol{C}_\Phi \boldsymbol{v}$. Let $\boldsymbol{V} = [\boldsymbol{v}_1, \cdots, \boldsymbol{v}_d] \in \mathbb{R}^{d_1 \times d}$ be the top-d eigenvectors; then, we use $\tilde{\Phi}\boldsymbol{V}$ as the d-dimensional representation. In other words, we use the 128 principal components of the 512-dimensional embedding.

F. More on Context Evaluation

In this section, we offer guidance for practitioners on identifying contexts with weak or strong associations with inputs. We then show that both excessively weak and overly strong associations degrade downstream performance and demonstrate that the proposed quantitative measurements accurately capture association strength in our controlled experiments. Moreover, we provide full experimental results that complements Table 1. Finally, we provide proofs for the lemmas in Section 5.1.

F.1. Quantitative measurements for level of association

While mutual information captures mutual dependence between random variables, estimating it from samples remains a long-standing challenge as it requires the joint density function to be known (Paninski, 2003). To address this, we propose alternative metrics that are computationally more tractable.

- **Decay rate (all association):** As we mentioned, the decay rate of the singular values $(s_i)_{i \geq 0}$ reflects the strength of association. To estimate the decay rate λ, we assume the singular values decay exponentially and fit the regression model $s_i^2 = \exp(-\lambda i)$. When λ is large, it indicates a fast decay rate and context has low association. Conversely, when λ is low, it implies a slow decay and highly associated context.
- **Expected kernel deviation (weak association):** Since the kernel values $k_A^+(a, a')$ can be close to 1 for the contexts with low association, we propose using the expected absolute deviation from 1, i.e. $\mathbb{E}_{x,x' \sim P_{\mathcal{X}}} \left[|k_X^+(x, x') - 1| \right]$, as a measure to indicate the weak association setting. Given samples $x_1, \cdots, x_n \in \mathcal{X}$, we use Monte Carlo sampling to approximate it, i.e.
$$\frac{1}{n^2} \sum_{i=1}^{n} \sum_{j=1}^{n} |k_X^+(x_i, x_j) - 1|.$$
- **Lipschitz constant (strong association):** We empirically measure the Lipschitz constant of k_X^+ for detecting contexts with strong association. Specifically, given samples $x_1, \cdots, x_n \in \mathcal{X}$, we use the following estimation:
$$L_{k_X^+} = \sup_{x,y,z \in \mathcal{X}} \frac{|k_X^+(z, x) - k_X^+(z, y)|}{||x - y||_2} \approx \max_{1 \leq i < j \leq n, 1 \leq k \leq n} \frac{|k_X^+(x_k, x_i) - k_X^+(x_k, x_j)|}{||x_i - x_j||_2}.$$

We note that the first metric requires estimating singular values $(s_i^2)_{i \geq 0}$ and the last two metrics rely on estimating kernel values $k_X^+(x, x')$. For estimating singular values, we employ the same technique as the task agnostic metric τ in Eqn. (3). Estimating kernel values, on the other hand, necessitates training an encode Φ and approximating $k_X^+(x, x')$ as $\Phi(x)^\top \Phi(x')$, which may require a large training set. Thus, decay rate measurements are preferred due to their simpler estimation process.

Additionally, for the non-smooth kernel k_X^+, we have the following lemma showing that we may have the singular functions could be non-smooth and are difficult to estimate.

Lemma F.1. *Let the Lipschitz constant for the positive pair kernel k_X^+ be $L_{k_X^+} = \sup_{x_1,x_2,x_3 \in \mathcal{X}} \frac{|k_X^+(x_3,x_1) - k_X^+(x_3,x_2)|}{||x_1 - x_2||_2}$ and the maximum Lipschitz constant of its eigenfunctions be $L_\mu = \max_{i \geq 1} \sup_{x_1,x_2 \in \mathcal{X}} \frac{|\mu_i(x_1) - \mu_i(x_2)|}{||x_1 - x_2||_2}$. Assume that all the eigenfunctions μ_i are bounded by c, i.e. $\mu_i(x) \leq c$ for all $i > 0$ and $x \in \mathcal{X}$. Then we have $L_{k_X^+} \leq c L_\mu \sum_{i=1} s_i^2$.*

Proof. We have

$$\begin{aligned}
L_{k_X^+} &= \sup_{x_1,x_2,x_3 \in \mathcal{X}} \frac{|k_X^+(x_3, x_1) - k_X^+(x_3, x_2)|}{\|x_1 - x_2\|_2} \\
&= \sup_{x_1,x_2,x_3 \in \mathcal{X}} \frac{|\sum_{i \geq 0} s_i^2 \mu_i(x_3) \mu_i(x_1) - \sum_{i \geq 0} s_i^2 \mu_i(x_3) \mu_i(x_2)|}{\|x_1 - x_2\|_2} \\
&= \sup_{x_1,x_2,x_3 \in \mathcal{X}} \frac{|\sum_{i \geq 0} s_i^2 \mu_i(x_3)(\mu_i(x_1) - \mu_i(x_2)|}{\|x_1 - x_2\|_2} \\
&= \sup_{x_1,x_2,x_3 \in \mathcal{X}} \frac{|\sum_{i > 0} s_i^2 \mu_i(x_3)(\mu_i(x_1) - \mu_i(x_2)|}{\|x_1 - x_2\|_2} \quad (\mu_0 \equiv 1) \\
&\leq \sup_{x_1,x_2,x_3 \in \mathcal{X}} \frac{\sum_{i > 0} s_i^2 \mu_i(x_3) |\mu_i(x_1) - \mu_i(x_2)|}{\|x_1 - x_2\|_2} \\
&\leq \sup_{x_3 \in \mathcal{X}} \sum_{i > 0} s_i^2 \mu_i(x_3) \sup_{x_1,x_2 \in \mathcal{X}} \frac{|\mu_i(x_1) - \mu_i(x_2)|}{\|x_1 - x_2\|_2} \\
&\leq \sup_{x_3 \in \mathcal{X}} \sum_{i > 0} s_i^2 \mu_i(x_3) L_\mu
\end{aligned}$$

$$\leq \sup_{x_3 \in \mathcal{X}} cL_\mu \sum_{i>0} s_i^2.$$

□

Assume that there exists a universal c that bounds all eigenfunctions. For a highly non-smooth kernel k_X^+ with high Lipschiz constant $L_{k_X^+}$, the lemma implies that we have either (i) smooth singular functions with large L_μ and slow decay in singular values with small $\sum_{i=1} s_i^2$, or (ii) non-smooth singular functions with large L_μ and fast decay in singular values with small $\sum_{i=1} s_i^2$. For (i), we need a larger d to approximate the kernel well, which leads to a higher downstream sample complexity. For (ii), the function approximation by neural networks becomes more difficult for non-smooth functions (Yarotsky, 2018).

F.1.1. Empirical Verification

Setup. We provide empirical evidence showing (1) downstream performance is worse for contexts with weak and strong associations, (2) the proposed quantitative measurements are positively correlated with the association level. To control the level of association, we use RBF kernels and KNN.

For the estimation of kernel value k_X^+, we use $k_X^+(x, x') = \int \frac{P^+(a|x)P^+(a|x')}{P_{\mathcal{A}}(a)} da$ since $P^+(a|x)$ can be efficiently computed for these contexts. The decay rate is estimated using a non-linear least squares approach to fit the regression model. For computing the expected kernel deviation, we utilize the entire training set. To estimate the Lipschitz constant, we restrict the sample size to $n = 1000$ for computational efficiency. Other experimental setups are the same as in Section 5.3.

Results. Figure 6 and Figure 7 illustrate the relationship between association level and both the linear probe error err_{d^*} and the decay rate λ for KNN and RBF contexts, respectively. The results show that err_{d^*} increases at both extremes, with most blue curves exhibiting a U-shape. This suggests that both weak and strong association levels lead to higher errors. Additionally, the red curve indicates that the decay rate λ increases as the association level strengthens, highlighting a strong correlation between association level and spectral decay, which is effectively captured by the estimated decay rate.

We report the relationship between association level and both the expected kernel deviation and the Lipschitz constant $L_{k_X^+}$ in Figure 8 for the KNN context and Figure 9 for the RBF context. The results show that contexts with low association exhibit small expected kernel deviations, while those with high association have large Lipschitz constants. These findings align with our theoretical developments in Section 5.1.

G. More Experiment Details and Results

See Table 3 for the full results.

G.1. Proof of Lemma 5.1

Proof. Define ${k_X^+}'(x, x') = k_X^+(x, x') - 1 = \sum_{i>0} s_i^2 \mu_i(x)\mu_i(x')$ that is the positive pair kernel without the trivial mode (s_0, μ_0), where the equality follows the definition of $T_{k_X^+}$. We also denote the corresponding kernel integral operator as $(T_{{k_X^+}'} g)(x) = \int g(x'){k_X^+}'(x, x')dP_{\mathcal{X}}(x')$. Then we have

$$\sum_{i>0} s_i^2 = \text{Tr}(T_{{k_X^+}'}) = \int {k_X^+}'(x, x')dP_{\mathcal{X}}(x') < \int \epsilon dP_{\mathcal{X}}(x') = \epsilon,$$

as desired. □

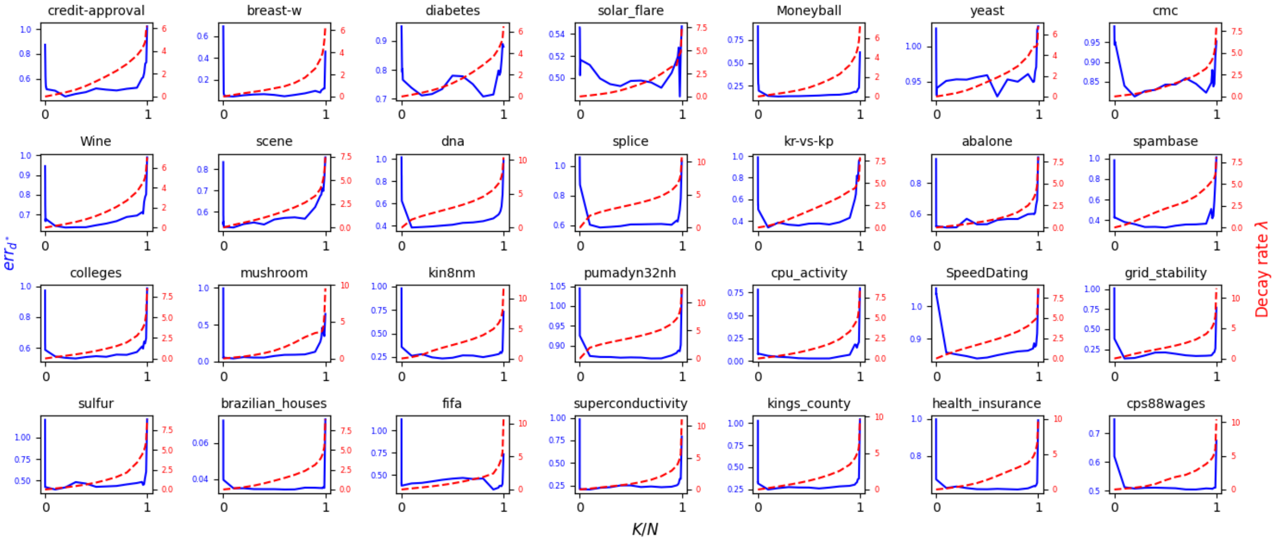


Figure 6. Association level vs err_{d^*} and the decay rate λ for the KNN context on the 28 datasets. A larger K/N indicates a lower association level, while a smaller K/N corresponds to a higher association level.

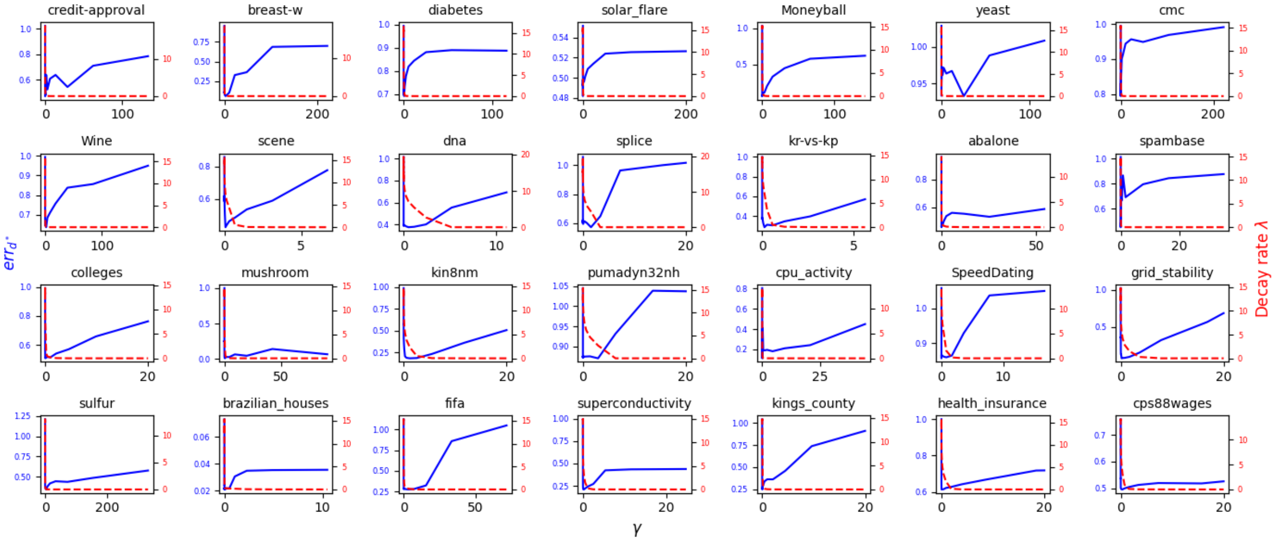


Figure 7. Association level vs err_{d^*} and the decay rate λ for the RBF context on the 28 datasets. A larger γ indicates a higher association level, while a smaller γ corresponds to a higher association level.

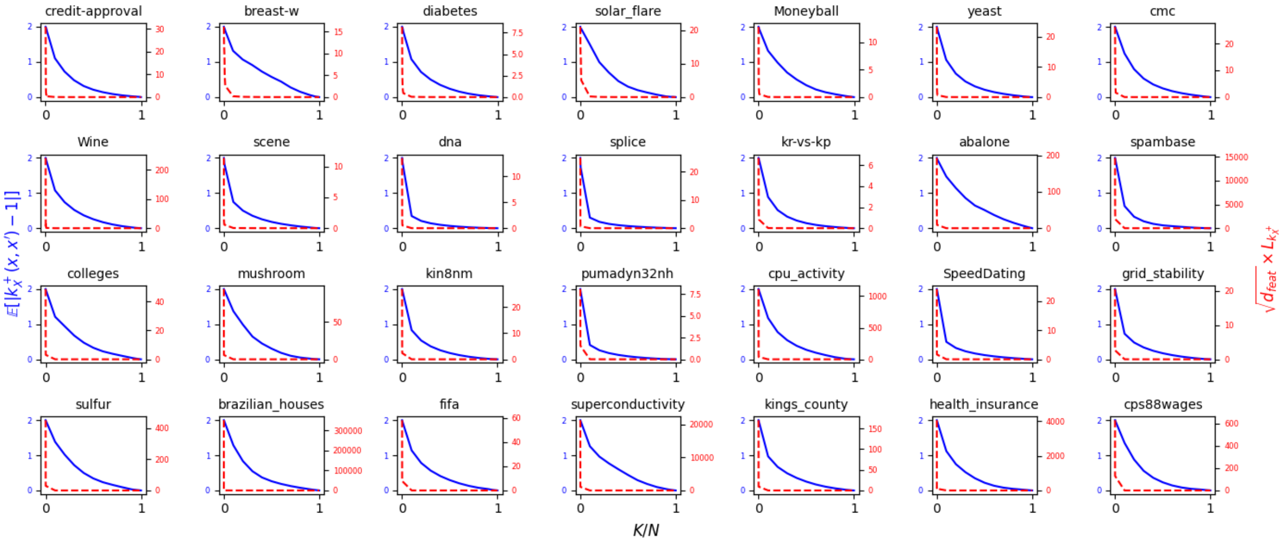


Figure 8. Association level vs the kernel deviation $\mathbb{E}_{x,x'\sim P_{\mathcal{X}}} |k_X^+(x,x') - 1|$ and the Lipschitz constant $L_{k_X^+}$ for the KNN context on the 28 datasets. A larger K/N indicates a lower association level, while a smaller K/N corresponds to a higher association level. We multiply $L_{k_X^+}$ by the input feature dimension d_{feat} to normalize the L_2 distance in the denominator.

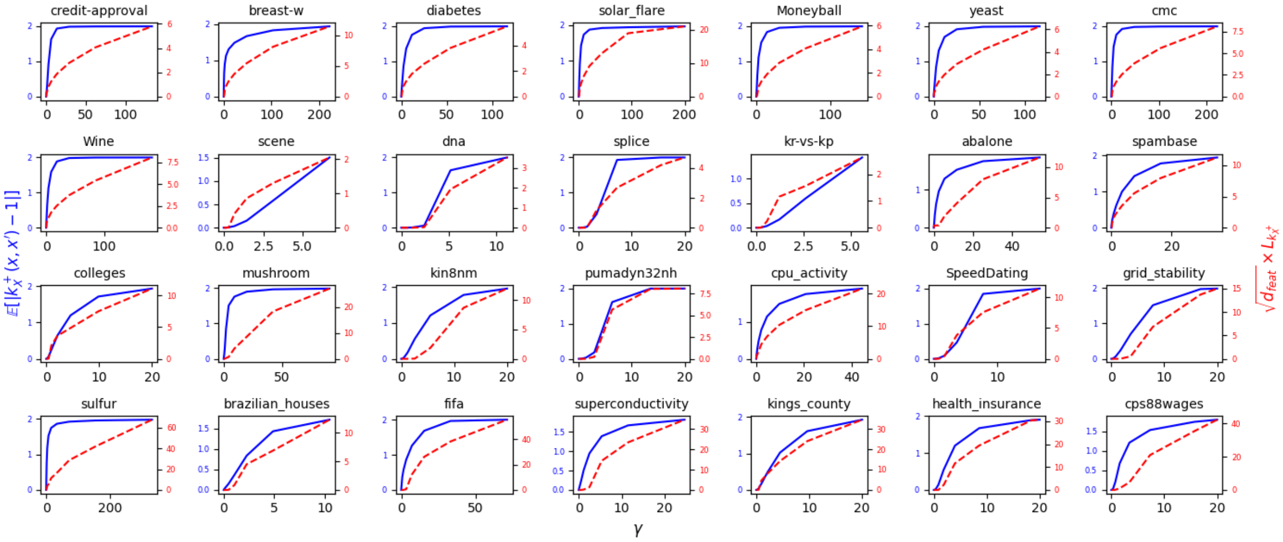
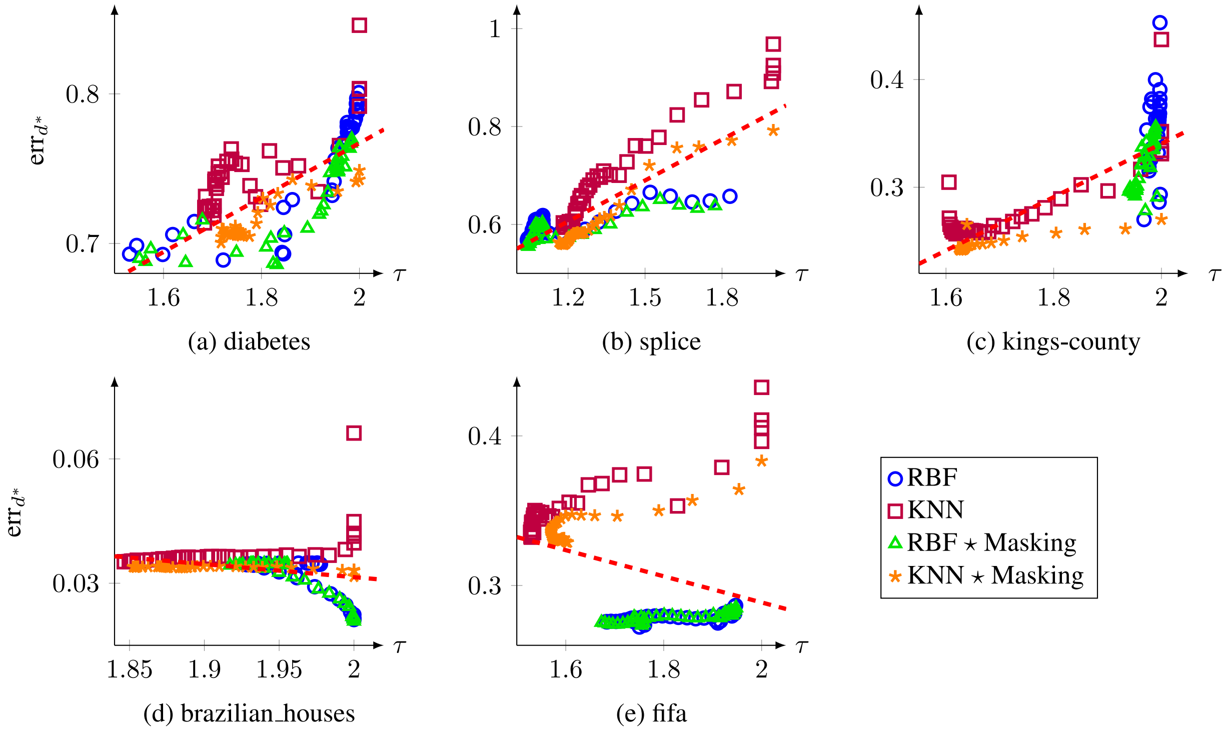


Figure 9. Association level vs the kernel deviation $\mathbb{E}_{x,x'\sim P_{\mathcal{X}}} |k_X^+(x,x') - 1|$ and the Lipschitz constant $L_{k_X^+}$ for the RBF context on the 28 datasets. A larger γ indicates a higher association level, while a smaller γ corresponds to a higher association level. We multiply $L_{k_X^+}$ by the input feature dimension d_{feat} to normalize the L_2 distance in the denominator.



(a) diabetes

(b) splice

(c) kings-county

(d) brazilian_houses

(e) fifa

Figure 10. Scatter plots of τ versus err_{d^*}. Dashed line: Linear fit.

Table 3. Correlation between τ and the actual error err_{d^*} on all 4 types of contexts.

Dataset	Size (↑)	#Feature	Type	Pearson	Distribution
credit-approval	690	15	Cls	0.583	0.683
breast-w	699	9	Cls	0.072	0.255
diabetes	768	8	Cls	0.737	0.740
solar_flare	1066	10	Reg	0.019	0.262
Moneyball	1232	14	Reg	0.680	0.650
yeast	1269	8	Cls	0.221	0.256
cmc	1473	9	Cls	0.867	0.860
Wine	1599	11	Reg	-0.084	0.212
scene	2407	299	Cls	0.608	0.685
dna	3186	180	Cls	0.881	0.843
splice	3190	60	Cls	0.831	0.801
kr-vs-kp	3196	36	Cls	0.543	0.512
abalone	4177	8	Reg	0.028	0.470
spambase	4601	57	Cls	0.775	0.858
colleges	7603	44	Reg	0.155	0.387
mushroom	8124	22	Cls	0.185	0.340
kin8nm	8192	8	Reg	0.805	0.760
pumadyn32nh	8192	32	Reg	0.938	0.961
cpu_activity	8192	21	Reg	0.709	0.825
SpeedDating	8378	120	Cls	0.590	0.656
grid_stability	10000	12	Reg	0.925	0.911
sulfur	10081	6	Reg	-0.180	0.487
brazilian_houses	10692	9	Reg	-0.290	0.563
fifa	19178	28	Reg	-0.349	0.663
superconductivity	21263	81	Reg	0.141	0.367
kings_county	21613	21	Reg	0.842	0.882
health_insurance	22272	11	Reg	0.601	0.749
cps88wages	28155	6	Reg	0.250	0.479
			Mean	0.431	0.611
			Median	0.587	0.659