Pretrain–Test Task Alignment Model for In-Context Learning by Linear Attention

Mary I. Letey[a,b], **Yue M. Lu**[a], **Cengiz Pehlevan**[a,b,c]

[a]The John A. Paulson School of Engineering and Applied Sciences, Harvard University
[b]The Kempner Institute for the Study of Natural and Artificial Intelligence, Harvard University
[c]Center for Brain Science, Harvard University
maryletey@fas.harvard.edu, yuelu@seas.harvard.edu, cpehlevan@seas.harvard.edu

Abstract In-context learning (ICL) allows pretrained neural networks, particularly transformers, to solve new tasks from examples presented in-context, without updating model weights. Our work develops a theory of task similarity to explain and predict generalization performance in ICL. We use an exactly solvable model of ICL in linear transformers, building on the work of [1], to derive an exact expression for ICL error, from which we propose an alignment measure that describes how train-test task mismatch affects ICL error. Our results offer a unified theoretical and empirical framework for understanding task alignment in in-context learning. We refer the interested reader to the full version of this work [2].

Setup We consider an in-context regression task: the input to the model is a sequence of the form $\{\boldsymbol{x}_1, y_1, \boldsymbol{x}_2, y_2, \ldots, \boldsymbol{x}_\ell, y_\ell, \boldsymbol{x}_{\ell+1}\}$, and the required output is the matching $y_{\ell+1}$. This input is called a context, and ℓ the context length. We consider an approximately linear relationship between $\boldsymbol{x} \in \mathbb{R}^d$ and y: for noise ϵ_i and task vector $\boldsymbol{w} \in \mathbb{R}^d$, have $y_i = \langle \boldsymbol{x}_i, \boldsymbol{w} \rangle + \epsilon_i$.

We will study the performance of the linear self-attention block [3] on this in-context regression task. The input to the linear self-attention model is an embedding matrix Z of the context sequence

$$Z = \begin{bmatrix} \boldsymbol{x}_1 & \boldsymbol{x}_2 & \ldots & \boldsymbol{x}_\ell & \boldsymbol{x}_{\ell+1} \\ y_1 & y_2 & \ldots & y_\ell & 0 \end{bmatrix} \in \mathbb{R}^{(d+1)\times(\ell+1)},$$

where 0 in the lower-right corner is a placeholder token for the $y_{\ell+1}$ we wish to predict. The output of a linear-attention block [3, 4, 5] is given by

$$A = Z + VZ(KZ)^\top (QZ)/\ell \tag{1}$$

for value matrix $V \in \mathbb{R}^{(d+1)\times(d+1)}$ and key, query matrices K, Q such that $K^\top Q \in \mathbb{R}^{(d+1)\times(d+1)}$. The final prediction of the linear attention for $y_{\ell+1}$ given context information Z is $\hat{y}(Z) = A_{d+1,\ell+1}$. See Section A for more detailed setup of this predictor.

The model is pretrained on n such contexts Z^μ to minimise the MSE between $\hat{y}(Z^\mu)$ and the true label $y^\mu_{\ell+1}$. The pretraining distribution is, for $i \in [\ell], \mu \in [n]$,

$$\boldsymbol{x}_i^\mu \sim_{\text{i.i.d.}} \mathcal{N}(0, I_d/d)\,, \qquad \epsilon_i^\mu \sim_{\text{i.i.d.}} \mathcal{N}(0, \rho)\,, \qquad \boldsymbol{w}^\mu \sim_{\text{unif}} \{\boldsymbol{w}_1\,, \cdots, \boldsymbol{w}_k\} \tag{2}$$

where $\boldsymbol{w}_j \sim_{\text{i.i.d.}} \mathcal{N}(0, C_{\text{train}})$ for $j \in [k]$. In this way, we control both the structure of the task distribution using C_{train} as well as the amount k of actually unique tasks seen during pretraining. We then wish to test the pretrained model on a general task to see if the model can genuinely perform in-context regression. The test distribution is then

$$\boldsymbol{x}_i^{\text{test}} \sim_{\text{i.i.d.}} \mathcal{N}(0, I_d/d)\,, \qquad \epsilon_i^{\text{test}} \sim_{\text{i.i.d.}} \mathcal{N}(0, \rho)\,, \qquad \boldsymbol{w}^{\text{test}} \sim_{\text{i.i.d.}} \mathcal{N}(0, C_{\text{test}})\,. \tag{3}$$

Theoretical Results Because of the simplicity of the setup, we can explicitly find the finite-sample optimised parameters of the linear attention model, and then compute an exact expression for the average ICL test error on tests sampled from (3) by the model pretrained on (2). The full expression is given in Eq. (26) in terms of token dimension d, context length ℓ, pretraining batch size n, task

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

diversity k, pretraining task distribution C_{train}, and testing task distribution C_{test}. We highlight **two key observations** that result from our formula for ICL error. Note in what follows that $\kappa := k/d$.

The first observation we highlight is that the full expression for ICL error (26) nicely decomposes into a more interpretable form, $e_{\text{ICL}}(C_{\text{train}}, C_{\text{test}}) = e_{\text{scalar}} + e_{\text{align}}(C_{\text{train}}, C_{\text{test}})$, with this alignment error term defined by Eq (28). This alignment measure captures information about the covariances C_{train} and C_{test}, but also captures how much of C_{train} can actually be learned given finite context length ℓ and finite task diversity κ: depending on how much of C_{train} can actually be resolved, it will "align" differently with C_{test}.

We validate our theory on tasks with various covariance structures, including power-law spectra and low-rank covariances. As shown in Figure 1, the theory-derived alignment measure strongly correlates with ICL generalization error, even for nonlinear architectures (panel B).

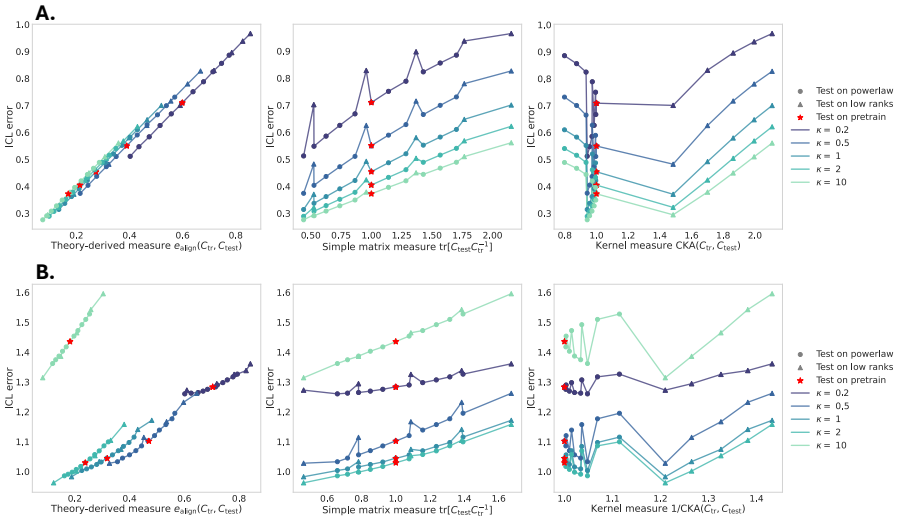

Figure 1: ICL error as a function of various task similarity measures. Panel A shows theoretical ICL error from Eq. (26); Panel B shows ICL error of a trained 2-layer transformer with softmax attention and MLP layers. From left to right, the alignment measures are our derived measure $e_{\text{align}}(C_{\text{train}}, C_{\text{test}})$, a matrix measure $\text{tr}[C_{\text{test}}C_{\text{train}}^{-1}]$, and a kernel-based measure [6]. The colours correspond to equivalent values of task diversity. The points show different C_{test} matrices with C_{train} fixed. Our measure e_{align} achieves the best correlation with ICL: the Spearman coefficients (measuring monotonicity, averaged over the different κs) for realistic architecture panel B are **0.99** (ours), 0.96, and 0.40 from left to right.

The second observation is that it is not always optimal to pretrain on the test distribution (i.e. choosing $C_{\text{train}} = C_{\text{test}}$). Figure 2 gives an example of this, showing how changing the spectral power of C_{train} relative to spectral power of C_{test} affects ICL performance. Specifically, take $C_{\text{test}} \propto \text{diag}(1^{-p}, ...d^{-p})$ and $C_{\text{train}} \propto \text{diag}(1^{-q}, ...d^{-q})$ such that $\text{tr}(C_{\text{test}}) = d = \text{tr}(C_{\text{train}})$. We fix p and vary q. For low task diversity (low κ), ICL error can be improved (green) by increasing q relative to p.

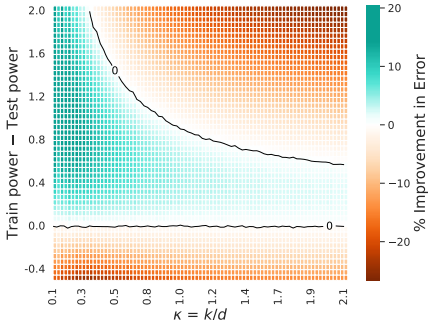

Figure 2: Percent improvement in ICL error against task spectral power difference and pretraining task diversity.

Conclusion We present a theoretical framework for understanding and predicting task alignment and spectral bias in in-context learning. We derive a matrix-based alignment score between pretraining and test task covariances, that accurately tracks generalization error even in nonlinear architectures. We also highlight that our formula shows ICL performance can be improved by changing the training task distribution in cases of low amounts of task data.

References

[1] Yue M. Lu, Mary Letey, Jacob A. Zavatone-Veth, Anindita Maiti, and Cengiz Pehlevan. Asymptotic theory of in-context learning by linear attention. *Proceedings of the National Academy of Sciences*, 122(28):e2502599122, 2025.

[2] Mary I. Letey, Jacob A. Zavatone-Veth, Yue M. Lu, and Cengiz Pehlevan. Pretrain-test task alignment governs generalization in in-context learning, 2025.

[3] Sinong Wang, Belinda Z. Li, Madian Khabsa, Han Fang, and Hao Ma. Linformer: Self-attention with linear complexity, 2020.

[4] Zhuoran Shen, Mingyuan Zhang, Haiyu Zhao, Shuai Yi, and Hongsheng Li. Efficient attention: Attention with linear complexities. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 3531–3539, 2021.

[5] Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. Transformers are RNNs: Fast autoregressive transformers with linear attention. In *International conference on machine learning*, pages 5156–5165. PMLR, 2020.

[6] Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited, 2019.

[7] Ruiqi Zhang, Spencer Frei, and Peter L. Bartlett. Trained transformers learn linear models in-context. *Journal of Machine Learning Research*, 25(49):1–55, 2024.

[8] Jingfeng Wu, Difan Zou, Zixiang Chen, Vladimir Braverman, Quanquan Gu, and Peter Bartlett. How many pretraining tasks are needed for in-context learning of linear regression? In *The Twelfth International Conference on Learning Representations*, 2024.

A Detailed Setup

This setup will follow [1], but differs in how the task structures and context lengths are modeled more generally, leading to more interpretable results highlighted above.

ICL of linear regression We consider an in-context regression task: the input to the model is a sequence of the form $\{x_1, y_1, x_2, y_2, \ldots, x_\ell, y_\ell, x_{\ell+1}\}$, and the required output is the matching $y_{\ell+1}$. This input is called a context, and ℓ the context length. The actual relationship we will consider between x and y will be approximately linear,

$$y_i = \langle x_i, w \rangle + \epsilon_i \tag{4}$$

for noise ϵ_i and task vector w. Effectively the model needs to form an estimate of w using $\{x_1, y_1, x_2, y_2, \ldots, x_\ell, y_\ell\}$ and then apply it to $x_{\ell+1}$ to estimate $y_{\ell+1}$.

Linear self-attention We will study the performance of the linear self-attention block [3] on this in-context regression task. The input to the linear self-attention model is an embedding matrix Z made up of our context sequence. Here, following the convention of [7, 8, 3], we chose the particular embedding of $\{x_1, y_1, x_2, y_2, \ldots, x_\ell, y_\ell, x_{\ell+1}\}$ to be

$$Z = \begin{bmatrix} x_1 & x_2 & \ldots & x_\ell & x_{\ell+1} \\ y_1 & y_2 & \ldots & y_\ell & 0 \end{bmatrix} \in \mathbb{R}^{(d+1)\times(\ell+1)}, \tag{5}$$

where 0 in the lower-right corner is a placeholder token for the $y_{\ell+1}$ we wish to predict. The output of a linear-attention block [4, 5, 3] is given by

$$A = Z + \frac{1}{\ell} V Z (KZ)^\top (QZ) \tag{6}$$

for value matrix $V \in \mathbb{R}^{(d+1)\times(d+1)}$ and key, query matrices K, Q such that $K^\top Q \in \mathbb{R}^{(d+1)\times(d+1)}$. Following the positional encoding in (5), the final prediction of the linear attention for $y_{\ell+1}$ given context information Z is

$$\hat{y} = A_{d+1,\ell+1}. \tag{7}$$

Pretraining data We pretrain the linear attention architecture on n sample sequences of the above form, i.e. for $\mu = 1, ..., n$, the μth sample sequence $\{x_1, y_1, x_2, y_2, \ldots, x_\ell, y_\ell, x_{\ell+1}\}$ related by the approximate linear mapping from (4), $y_i^\mu = \langle x_i^\mu, w^\mu \rangle + \epsilon_i^\mu$, where now w^μ is the task vector corresponding to the μth sample context.

We make the following statistical assumptions about this pretraining data, and will denote a sample from this distribution by $\mathcal{P}_{\text{train}}$:

$$\begin{aligned} x_i^\mu &\sim_{\text{i.i.d.}} \mathcal{N}(0, I_d/d) && \text{for } i \in [\ell], \mu \in [n] \\ \epsilon_i^\mu &\sim_{\text{i.i.d.}} \mathcal{N}(0, \rho) && \text{for } i \in [\ell], \mu \in [n] \\ w^\mu &\sim_{\text{unif}} \{w_1, \cdots, w_k\} && \text{for } w_j \sim_{\text{i.i.d.}} \mathcal{N}(0, C_{\text{train}}),\ j \in [k]. \end{aligned} \tag{8}$$

Let's parse the task distribution: For $\mu = 1, ..., n$, the task vector w^μ for the μth sample context is uniformly sampled from the set $\{w_1, \cdots, w_k\}$, sampled from $\mathcal{N}(0, C_{\text{train}})$ before sampling the n contexts. Thus k controls the task diversity in the pretraining dataset; the same task vector from $\{w_1, \cdots, w_k\}$ could be repeated within the n contexts. Note that the tokens and label noise here are sampled precisely the same as in [1]; the key difference here is that the task covariance C_{train} is now general.

The parameters of the linear attention module are learned from n samples of input sequences,

$$\{x_1^\mu, y_1^\mu, \ldots, x_{\ell+1}^\mu, y_{\ell+1}^\mu\}, \qquad \mu = 1, \ldots, n. \tag{9}$$

Parameter reduction Work [1] has shown that the output $\hat{y} = A_{d+1,\ell+1}$ of the model can be reduced as follows: Expand attention and value matrices as

$$V = \begin{bmatrix} V_{11} & v_{12} \\ v_{21}^\top & v_{22} \end{bmatrix}, \quad M = \begin{bmatrix} M_{11} & m_{12} \\ m_{21}^\top & m_{22} \end{bmatrix} \equiv K^\top Q\,.$$

Expanding these expressions and the particular form of Z gives the predictor as

$$\hat{y} = \frac{1}{\ell} x_{\ell+1} \cdot \left(v_{22} M_{11}^\top \sum_{i=1}^{\ell} y_i x_i + v_{22} m_{21} \sum_{i=1}^{\ell} y_i^2 + M_{11}^\top \sum_{i=1}^{\ell+1} x_i x_i^\top v_{21} + m_{21} \sum_{i=1}^{\ell} y_i x_i^\top v_{21} \right), \tag{10}$$

Within this expression, it is argued in [1] that the final two terms depending on v_{21} can be removed without affecting the performance of the estimator: the first, depending on $x_i x_i^\top v_{21}$, does not contain any task information, and thus does not help us estimate w. The final, depending on $y_i x_i^\top v_{21}$ provides only a one dimensional projection of x and w, and so for large dimensional tokens, does not effectively contribute to good estimate of w either. For this reason, we set $v_{21} = 0$. Reference [7] shows that this choice of parameter initialization is stable under SGD, further validating this assumption. With this simplification, we can rewrite the simplified model's output as

$$\hat{y} = \langle \Gamma, H_Z \rangle \tag{11}$$

for parameter matrix

$$\Gamma \equiv v_{22} \begin{bmatrix} M_{11}^\top / d & m_{21} \end{bmatrix} \in \mathbb{R}^{d \times (d+1)}. \tag{12}$$

and data matrix

$$H_Z \equiv x_{\ell+1} \begin{bmatrix} \frac{d}{\ell} \sum_{i \le \ell} y_i x_i^\top & \frac{1}{\ell} \sum_{i \le \ell} y_i^2 \end{bmatrix} \in \mathbb{R}^{d \times (d+1)}, \tag{13}$$

from input sequence Z.

This parameter-reduced version of linear attention lends itself to analytical study. When referring to the linear attention model in the remainder of this paper, we mean this reduced parameter version.

Model pretraining The parameters of the linear attention module are learned from n samples of input sequences,

$$\{x_1^\mu, y_1^\mu, \ldots, x_{\ell+1}^\mu, y_{\ell+1}^\mu\}, \qquad \mu = 1, \ldots, n. \tag{14}$$

We estimate model parameters by minimizing MSE loss on next-output prediction with ridge regularization, giving

$$\Gamma^* = \arg\min_{\Gamma} \sum_{\mu=1}^{n} \left(y_{\ell+1}^\mu - \langle \Gamma, H_{Z^\mu} \rangle \right)^2 + \frac{n}{d} \lambda \|\Gamma\|_{\mathrm{F}}^2, \tag{15}$$

where $\lambda > 0$ is a regularization parameter, and H_{Z^μ} is defined by (13), containing the μth context. Here $\|\|_F$ is the Frobenius norm. The solution to the optimization problem in (15) can be expressed explicitly as

$$\mathrm{vec}(\Gamma^*) = \left(\frac{n}{d} \lambda I + \sum_{\mu=1}^{n} \mathrm{vec}(H_{Z^\mu}) \mathrm{vec}(H_{Z^\mu})^\top \right)^{-1} \sum_{\mu=1}^{n} y_{\ell+1}^\mu \mathrm{vec}(H_{Z^\mu}). \tag{16}$$

Evaluation We wish to understand the performance and behavior of this estimator Γ^*, which is pretrained on data from $\mathcal{P}_{\mathrm{train}}$, when tested on new data. Namely, we will analyze the average performance of an estimator $\hat{y} = \langle \Gamma, H_Z \rangle$ as a function of given fixed Γ under the MSE loss, which is the natural loss for a regression test. We have

$$e(\Gamma) = \mathbb{E}_{\mathcal{P}_{\mathrm{test}}} \left[(y_{\ell+1} - \langle \Gamma, H_Z \rangle)^2 \right]. \tag{17}$$

The distribution $\mathcal{P}_{\mathrm{test}}$ refers to the test distribution, detailing how to sample tokens x, tasks w, and noise ϵ at test time. We will consider two different testing regimes: the *in-context learning (ICL)* test, where the model sees new tasks w, and the *in-distribution generalization (IDG)* (or *in-weights*) test, where the model sees the exact same tasks used in training $\{w_1, \cdots, w_k\}$, where each $w_j \sim_{\text{i.i.d.}} \mathcal{N}(0, C_{\mathrm{train}})$. Explicitly, we define these test distributions and corresponding error functions as

$$e_{\mathrm{IDG}}(\Gamma) = \mathbb{E}_{\mathcal{P}_{\mathrm{IDG}}} \left[(y_{\ell+1} - \langle \Gamma, H_Z \rangle)^2 \right] \tag{18}$$

$$\mathcal{P}_{\mathrm{IDG}} := x_i^\mu \sim_{\text{i.i.d.}} \mathcal{N}(0, I_d/d), \quad w^\mu \sim_{\text{unif}} \{w_1, \cdots, w_k\}, \quad \epsilon_i^\mu \sim_{\text{i.i.d.}} \mathcal{N}(0, \rho)$$
$$\text{for } i \in [\ell], \mu \in [n]$$

and

$$e_{\text{ICL}}(\Gamma) = \mathbb{E}_{\mathcal{P}_{\text{ICL}}}\left[(y_{\ell+1} - \langle \Gamma, H_Z \rangle)^2\right] \tag{19}$$
$$\mathcal{P}_{\text{ICL}} := x_i^\mu \sim_{\text{i.i.d.}} \mathcal{N}(0, I_d/d)\,, \quad w^\mu \sim_{\text{i.i.d.}} \mathcal{N}(0, C_{\text{test}})\,, \quad \epsilon_i^\mu \sim_{\text{i.i.d.}} \mathcal{N}(0, \rho)$$
$$\text{for } i \in [\ell_{\text{test}}]\,, \mu \in [n]$$

Notice here that we've introduced two different context lengths: ℓ for the IDG distribution, which is the same context length as the pretraining setup $\mathcal{P}_{\text{train}}$ given by (2), and ℓ_{test} which is the context length at testing time. This allows us to later explore the effect of pretraining and testing on sequences of different context lengths.

We assume that both task covariance matrices C_{train} and C_{test} are well-behaved in high dimensions, specifically that $\text{tr}[C_{\text{train}}], \text{tr}[C_{\text{test}}] = \Theta(1)$. This ensures the task signals are not over or under amplified as $d \to \infty$.

Given this setup, to understand the performance of our model on both ICL and IDG tasks, we will need to evaluate these expressions for the pretrained attention matrix Γ^* given in (16). In Section B we will give explicitly formulas for $e_{\text{IDG}}(\Gamma^*)$ and $e_{\text{ICL}}(\Gamma^*)$, i.e. the in-distribution and in-context performance of the optimal pretrained parameters Γ^* from (16).

B Theoretical Results

We ultimately wish to obtain a closed-form expression for the in-distribution and in-context performance of the model's parameters Γ^*, as measured by e_{IDG} and e_{ICL} defined in (18) and (19). These equations currently express the errors as averages over the general testing distributions $\mathcal{P}_{\text{IDG}}$ and $\mathcal{P}_{\text{ICL}}$. The main result of this paper will be a deterministic formula for these errors $e_{\text{ICL}}(\Gamma^*)$ and $e_{\text{IDG}}(\Gamma^*)$ at optimal model parameters Γ^* from (16).

Joint scaling limit We consider a joint asymptotic limit [1] in which the input dimension d, the pretraining dataset size n, the context length ℓ, and the number of task vectors in the training set k, go to infinity together such that

$$\frac{\ell}{d} \equiv \alpha = \Theta(1), \quad \frac{k}{d} \equiv \kappa = \Theta(1), \quad \frac{n}{d^2} \equiv \tau = \Theta(1). \tag{20}$$

As we are allowing mismatch between pretraining context length ℓ and test-time context length ℓ_{test}, we will thus write

$$\alpha_{\text{train}} = \frac{\ell}{d}\,, \qquad \alpha_{\text{test}} = \frac{\ell_{\text{test}}}{d}\,.$$

These main scaling parameters can be understand intuitively as follows (note that this explanation overlaps with intuition given in [1]).

- α: Context length ℓ should scale linearly with token dimension d. This is familiar from standard linear regression, and intuitively we can think about this from the perspective of how many samples $x_1, ..., x_\ell$ it would take to accurately estimate the statistical structure of the tokens. Accurately estimating the sample covariance $\hat{\Sigma}_x$ is key for good ICL performance, as without this, the model cannot decouple tasks from tokens.
- κ: Task diversity k should scale linearly with token dimension d. Here a similar intuition applies, as we need to correctly estimate the statistics of the tasks we see during pretraining to be able to have a good estimate of the full task manifold.
- τ: number of pretraining samples n should scale quadratically with token dimension d. This can be understood by considering the view of the linear attention module itself: there are d^2 free parameters (i.e. Γ) and n data points $Z \to y_{\ell+1}$ that the model sees during pretraining.

Pretraining task quantities Before presenting our formula for ICL and IDG error, we must first define various task-distribution quantities, which asymptotically only depend on the pretraining covariance C_{train} and the task diversity scaling parameter κ. These quantities effectively tell us how

well we can reconstruct C_{train} from the k-sample pretraining task covariance

$$R_k = \frac{1}{k}\sum_{j\in[k]} w_j w_j^\top .$$

We will define the following deterministic quantities, given in terms of high-dimensional limits of R_k.

$$\mathcal{M}_\kappa(\nu) \equiv \lim_{d\to\infty} \operatorname{tr}\left[(R_k+\nu I_d)^{-1}\right], \qquad (R_k+\nu I_d)^{-1} \simeq F \tag{21}$$

$$\mathcal{M}'_\kappa(\nu) \equiv -\lim_{d\to\infty} \operatorname{tr}\left[(R_k+\nu I_d)^{-2}\right], \quad (R_k+\nu I_d)^{-2} \simeq -F' \equiv -\frac{\mathrm{d}}{\mathrm{d}\nu}F \tag{22}$$

where

$$F \equiv \left(\left(1-\frac{1}{\kappa}+\frac{\nu}{\kappa}\mathcal{M}_\kappa(\nu)\right)C_{\text{train}}+\nu I_d\right)^{-1} \tag{23}$$

Intuitively, these give us information about how much signal in C_{train} can be recovered after a finite number k of pretraining samples, filtered by noise level ν. As $\nu\to 0$ and $\kappa\to\infty$, we limit to full recovery of the original distribution as R_k approaches C_{train}.

Renormalized ridges For convenience write $c_{\text{train}} \equiv \operatorname{tr}[C_{\text{train}}]$, $c_{\text{test}} \equiv \operatorname{tr}[C_{\text{test}}]$. We further define renormalized ridge $\tilde{\lambda}$ self-consistently as the nonnegative root of

$$\tilde{\lambda}\mathcal{M}_\kappa\left(\frac{c_{\text{train}}+\rho}{\alpha_{\text{train}}}+\tilde{\lambda}\right)-\frac{\lambda\tau}{\tilde{\lambda}}+\tau-1=0 \tag{24}$$

and

$$\nu \equiv \frac{\rho+c_{\text{train}}}{\alpha_{\text{train}}}+\tilde{\lambda}. \tag{25}$$

We see here that α modulates how well the model can infer the statistics of the tokens. This is why it appears in expressions like $\nu=\frac{\rho+c_{\text{train}}}{\alpha_{\text{tr}}}$ or $\frac{\rho+c_{\text{test}}}{\alpha_{\text{test}}}$. These are effective noise-to-signal terms familiar from standard linear regression, where the optimal ridge regularization parameter balances the variance due to label noise (ρ) with the estimation error from having finite data. At infinite sample size, the optimal ridge is simply ρ, the label noise. However, at finite sample size, the effective regularization is increased due to the finite-sample estimation error, and becomes $\frac{\rho+\sigma_w^2}{\alpha}$, where σ_w^2 characterizes task variability or complexity. In the same way, the model has to resolve the statistics of the tokens (x) over samples (ℓ, measured by α), and so these same ridge terms familiar from linear regression appear in our formula.

We finally have the following asymptotic error characterizations. We validate that these analytical formulas indeed match simulations of finite sample Γ^* error in Figure 3.

Result 1 ***Asymptotic ICL and IDG errors at optimal Γ^*.*** *Let $\mathcal{M}_\kappa(\nu), \mathcal{M}'_\kappa(\nu), F, F', \tilde{\lambda}$, and ν be the quantities defined by (21),(22),(24), and (25). We then have the deterministic equivalent of ICL error (19) as*

$$\begin{aligned} e_{\text{ICL}}(C_{\text{train}}, C_{\text{test}}) \simeq \rho &+ \frac{\rho+c_{\text{test}}}{\alpha_{\text{test}}}\left(1+(q-2\nu)\mathcal{M}_\kappa+(q\tilde{\lambda}-\nu^2)\mathcal{M}'_\kappa\right) \\ &+ q\operatorname{tr}\left[C_{\text{test}}F\right]+(q\tilde{\lambda}-\nu^2)\operatorname{tr}\left[C_{\text{test}}F'\right]. \end{aligned} \tag{26}$$

where $q \equiv e_{\text{IDG}}(\Gamma^*)/\tau$ is defined in terms of the deterministic equivalent of IDG error (18)

$$e_{\text{IDG}}(C_{\text{train}}) \simeq \tau\frac{\rho+\nu-\nu^2\mathcal{M}_\kappa(\nu)-\tilde{\lambda}(1-2\nu\mathcal{M}_\kappa(\nu)-\nu^2\mathcal{M}'_\kappa(\nu))}{\tau-(1-2\tilde{\lambda}\mathcal{M}_\kappa(\nu)-\tilde{\lambda}^2\mathcal{M}'_\kappa(\nu))}. \tag{27}$$

B.1 Train-Test decomposition of ICL error

Equation (26) admits an interesting decomposition: the structure of the test task distribution (i.e. the matrix C_{test}) only interacts with pretraining structure in this formula through the term

$$e_{\text{align}}(C_{\text{train}}, C_{\text{test}}) \equiv \operatorname{tr}\left[C_{\text{test}}\left(qF+(q\tilde{\lambda}-\nu^2)F'\right)\right]. \tag{28}$$

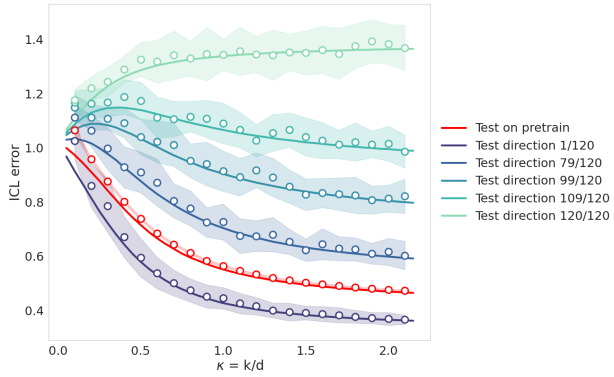

Figure 3: Theoretical e_{ICL} curves plotted against numerical simulations (dots) of Γ^* computed directly from sampled data. We choose C_{train} with uniform eigenvalue distribution, namely $C_{\mathrm{train}} \propto \mathrm{diag}([d, d-1, \cdots, 1])$ such that $c_{\mathrm{train}} = 1$. We compare to testing on the pretraining distribution, as well as testing on spiked rank-1 signals where $C_{\mathrm{test}}^i = \mathrm{diag}([0, \cdots, d, \cdots, 0])$ is spiked at index i. *Parameters:* $d = 120$, $\alpha = 2$, $\tau = 4$, $\rho = 0.01$. Shading represents $\pm$std of numerical simulations.

The e_{align} term can be understood as a very specific alignment measure between pretraining and testing, since the matrices F and F' recall depend only on C_{train}. This measure also depends on task diversity, context length, and sample size through κ, α, τ. The remaining terms of e_{ICL} can be grouped together as

$$e_{\mathrm{scalar}} \equiv \rho + \frac{\rho + c_{\mathrm{test}}}{\alpha_{\mathrm{test}}} \left(1 + (q - 2\nu)\mathcal{M}_\kappa + (q\tilde{\lambda} - \nu^2)\mathcal{M}'_\kappa\right). \tag{29}$$

Note that e_{scalar}, unlike e_{align}, contains no structural information about C_{test} and only depends on its trace c_{test}. The additive ρ term is simply the Bayes error coming from the label noise, while the second depends on the effective noise $(\rho + c_{\mathrm{test}})/\alpha_{\mathrm{test}}$ from the test contexts. We then have

$$e_{\mathrm{ICL}}(C_{\mathrm{train}}, C_{\mathrm{test}}) = e_{\mathrm{scalar}} + e_{\mathrm{align}}(C_{\mathrm{train}}, C_{\mathrm{test}}).$$

By comparing the small-κ and large-κ behaviour of the first term, i.e.

$$\lim_{\kappa \to 0} e_{\mathrm{scalar}} < \lim_{\kappa \to \infty} e_{\mathrm{scalar}}\,,$$

we see that this term is eventually increasing in κ. Conversely, the $\kappa \to 0$ limit of $e_{\mathrm{align}}(C_{\mathrm{train}}, C_{\mathrm{test}})$ may be larger or smaller than its $\kappa \to \infty$ limit. We see that the behaviour of e_{ICL} in κ will be a nontrivial interaction between these two terms, as seen in Figure 3: for highly-aligned training and testing distributions, e_{align} immediately wins over e_{scalar} leading to monotonic decrease in e_{ICL} in κ; for worse-aligned training and testing distributions, e_{ICL} can be nonmonotonic or even monotonically increasing in κ. This is intriguing, as one would naively expect additional task samples (i.e. higher κ) to always improve in-context performance, but this is not the case in general: whether or not additional task samples are helpful for ICL depends on the alignment of the pretraining and testing distributions.