Specialization-generalization transition in exemplar-based in-context learning

Chase Goddard
Joseph Henry Laboratories of Physics
Princeton University
cgoddard@princeton.edu

Lindsay M. Smith
Joseph Henry Laboratories of Physics
Princeton University
lindsay.smith@princeton.edu

Vudtiwat Ngampruetikorn*
School of Physics
University of Sydney
vudtiwat.ngampruetikorn@sydney.edu.au

David J. Schwab*
Initiative for the Theoretical Sciences
The Graduate Center, CUNY
dschwab@gc.cuny.edu

Abstract

In-context learning (ICL) is a striking behavior seen in pretrained transformers that allows models to generalize to unseen tasks after seeing only a few examples. We investigate empirically the conditions necessary on the pretraining distribution for ICL to emerge. Previous work has focused on the number of distinct tasks necessary in the pretraining distribution – here, we use a different notion of task diversity to study the emergence of ICL in transformers trained on linear functions. We find that as task diversity increases, transformers undergo a transition from a specialized solution, which exhibits ICL only within the pretraining distribution, to a solution which generalizes out of distribution to the entire task space. We also investigate the nature of the solutions learned by the transformer on both sides of the transition, and observe similar transitions in nonlinear regression problems.

1 Introduction

The ability of transformers [1] to do few-shot learning from examples seen in their context is a striking phenomenon exhibited by modern machine learning models [2] called *in-context learning* (ICL). ICL has been extensively studied [3–6] and enables models to solve certain new tasks without re-training. Of particular interest is how the ability for transformers to perform ICL arises from pretraining: What conditions must be met in order for ICL to emerge?

Prior work [3, 4] has focused on understanding how the number of tasks in the pretraining distribution affects the ability of the model to generalize to new tasks not present during pretraining. Here, we ask a related but distinct question: If a model is pretrained only on tasks from a *subset* of the full task space, what conditions are necessary for it to generalize to the rest of the space? This question prompts us to consider a more general notion of *task diversity* – a pretraining distribution with K tasks that are more different in character should be considered more "diverse" than another distribution with K tasks that are similar to each other. Sampling a distribution with many similar tasks has the potential to induce the model towards a more specialized ICL solution that performs well only on novel tasks within its pretraining distribution. However, we observe that transformers trained to do ICL of linear functions undergo a transition from a specialized solution to one that generalizes over the full task space as we increase the degree of task diversity. This phenomenon of out-of-distribution *task generalization* sheds new light on in-context learning behavior.

*DJS and VN contributed to this work equally.

Workshop on Scientific Methods for Understanding Deep Learning, NeurIPS 2024.

Contributions:

- We train transformers to exhibit ICL of linear functions with weight vectors drawn from a subset of the unit hypersphere. As the size of this subset increases, we observe a transition from specialized models, which perform well only on the training portion of the hypersphere, to models that generalize out of task distribution to the entire hypersphere.
- We show empirically that label noise shifts the location of this specialization-generalization transition; the transformer must be trained on tasks from a larger subset of the hypersphere in order to generalize to the whole sphere.
- We investigate the nature of the solutions found by our transformers, and find that specialized solutions outperform optimal Bayesian solutions to the regression problem on small numbers of examples. In contrast, transformers that generalize to the entire hypersphere exhibit performance similar to known optimal solutions.
- We investigate the ICL performance of models as test tasks move off the unit hypersphere. We observe good performance for tasks within the hypersphere, but our models fail to generalize to tasks far outside the hypersphere.
- We show that specialization-generalization transitions also occur in nonlinear regression problems, suggesting that the phenomenon may be a general feature of ICL in transformers.

2 Training setup and task distribution geometry

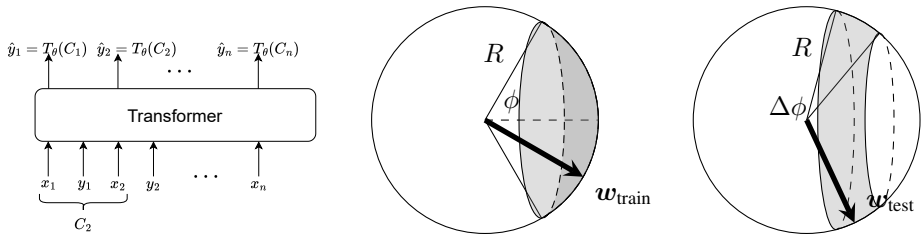


Figure 1: **Testing ICL via task similarity. A:** The transformer takes as input a sequence of pairs $\{x_i, y_i\}_{i=1}^n$ and is trained to predict y_k from a context $C_k = \{x_1, y_1, \ldots, x_k\}$. The elements x_i and y_i are related linearly by a task w^T: $y_i = w^T x_i + \epsilon_i$. **B:** The training tasks w_{train} are drawn from a hyperspherical cap with half-angle ϕ. Notice that $\phi = 180^\circ$ therefore corresponds to the entire hypersphere. **C:** The test tasks w_{test} are drawn from a hyperspherical band of width $\Delta\phi$.

ICL of linear functions: We investigate the ability of transformers to perform in-context learning of linear functions, when tasks are drawn from distributions with varying levels of *task diversity*, i.e. from hyperspherical caps of varying half-angles. We define a *task* to be a vector $w \in \mathbb{R}^d$, and the transformer takes as input a sequence of up to n pairs $\{x_1, y_1, \ldots, x_n\}$, where $y_i = w^T x_i + \epsilon_i$. Here, $x_i \sim \mathcal{N}(0, I_d)$ and $\epsilon_i \sim \mathcal{N}(0, \sigma^2)$.

Pretraining task distribution: We define a family of *task distributions* parameterized by $\phi \in [0, \pi]$ (See Fig 1B). We take $S^{d-1}(\phi)$ to be a section of the surface of the hypersphere in d dimensions, i.e. $S^{d-1}(\phi) = \{w \in S^{d-1} \mid \text{angle}(w, v) \leq \phi\}$, with $v \in \mathbb{R}^d$ a fixed vector. We then define the task distribution family $p_\phi(w) \equiv \text{Unif}(S^{d-1}(\phi))$.

Pretraining: During pretraining, the transformer T_θ is optimized to minimize the mean squared error (MSE) between a *context* of data $C_k \equiv \{x_1, y_1, \ldots, x_k\}$ and the target y_k. During pretraining, the tasks w are drawn i.i.d. for each context from $p_\phi(w)$. We use AdamW [7] to optimize the MSE,

$$L_{\text{train}}(\theta) = \mathbb{E}_{w \sim p_\phi}\left[\frac{1}{n}\sum_{k=1}^{n}\left(T_\theta(C_k) - y_k\right)^2\right]. \tag{1}$$

Test task distribution: We evaluate the performance of the transformer over a family of task distributions parameterized by $\phi, \Delta\phi \in [0, \pi]$ (See Fig 1C). We define the hyperspherical band starting at angle ϕ with width $\Delta\phi$ to be the set $B^{d-1}(\phi, \Delta\phi) = \{w \in S^{d-1} \mid \phi \leq \text{angle}(w, v) \leq \phi + \Delta\phi\}$, with v some fixed vector. The test task distribution is then uniform over this set: $p_{\phi,\Delta\phi}(w) = \text{Unif}(B^{d-1}(\phi, \Delta\phi))$.

Evaluation: We evaluate models by computing the MSE between the full context C_n and the final target y_n. During test time, we draw w i.i.d. for each context from $p_{\phi,\Delta\phi}$:

$$L_{\text{test}}(\theta) = \mathbb{E}_{w \sim p_{\phi,\Delta\phi}}\left[(T_\theta(C_n) - y_n)^2\right] \tag{2}$$

3 Experimental Results

In all experiments, we study $d = 10$ dimensional regression with $n = 50$ examples in each context. We use a GPT-2 style transformer [8] with learned positional embeddings, a hidden dimension of $d_h = 128$, 10 layers, and 8 attention heads. We use a learned linear embedding to map x_i and y_i to the hidden dimension $d_h = 128$. The target values y_i are padded with $d - 1$ zeroes.

During pretraining, we train 12 models over pretraining distributions $p_\phi(w)$ for $\phi \in [15^\circ, 180^\circ]$ in 15° increments. We observe that repeated runs, with different initializations and trained on data generated from different sampled tasks $w \sim p_\phi$, yield consistent results (see Fig 5).

3.1 Observation of a specialization-generalization transition

We show the results from evaluating the 12 models on the test task distributions $p_{\phi,\Delta\phi}(w)$ in Fig 2A. We pick $\Delta\phi = 5^\circ$ and $\phi \in [0^\circ, 5^\circ, \ldots 175^\circ]$. For models with a pretraining task distribution with $\phi < 90^\circ$, we observe good test performance only within the portion of the hypersphere covered by the pretraining distribution, and performance degrades outside of this range. However, for models trained over $p_\phi(w)$ with $\phi > 90^\circ$, we see essentially perfect performance across the entire hypersphere. This occurs despite the fact that these models were trained using only data generated from a *subset* of the full task space. Note that even before the transition, models trained on a cap with $\phi \geq 45^\circ$ exhibit out-of-distribution task generalization – they outperform simply picking the weight vector in the training distribution closest to the test weight vector (see dashed lines in 2A).

One may think that this transition arises solely from geometric considerations owing to the high-dimensional nature of the task hypersphere (since $\phi = 90^\circ$ corresponds to half the hypersphere, where most of its volume concentrates as d becomes large). However, in Fig 2B, we see that this cannot be the only cause of the transition. For noisy regression with $\sigma^2 = 0.25$, we see that the transition now occurs around $\phi = 120^\circ$, which plays no special role in high dimensional geometry.

Comparison to ordinary least squares We now investigate the solutions learned by the transformer on both sides of the transition. In Fig 2C, we compare the performance of the transformer and ordinary least squares (OLS), solid and dashed curves, respectively. For short contexts, the specialized solution which is learned for $\phi < 90^\circ$ outperforms OLS within the task distribution. For $\phi > 90^\circ$, the performance of the transformer is similar to OLS.

Beyond the unit hypersphere What happens to the generalization ability of the model as the radius of the task distribution changes? We train several models on data generated from tasks on the surface of the unit hypersphere, and evaluate them on tasks drawn from spheres of varying radii. Each model is trained on tasks from $p_\phi(w)$ and evaluated on the equivalent distribution (with the same ϕ) on a hypersphere with a different radius. In Fig 2D, we observe that for $\phi > 45^\circ$ the model is able to generalize perfectly to tasks with $R < 1$, despite being trained only on tasks with $R = 1$.

3.2 Investigating the interplay between the two forms of task diversity

In order to examine the effect of both forms of task diversity (number of tasks & task similarity), we train 120 models with task similarity ϕ and number of tasks N in the set: $(\phi, N) \in \{15^\circ, 30^\circ, \ldots, 180^\circ\} \times \{2^2, 2^3, \ldots, 2^{11}\}$. In Fig 3A, we plot the resulting *in task-distribution* loss:

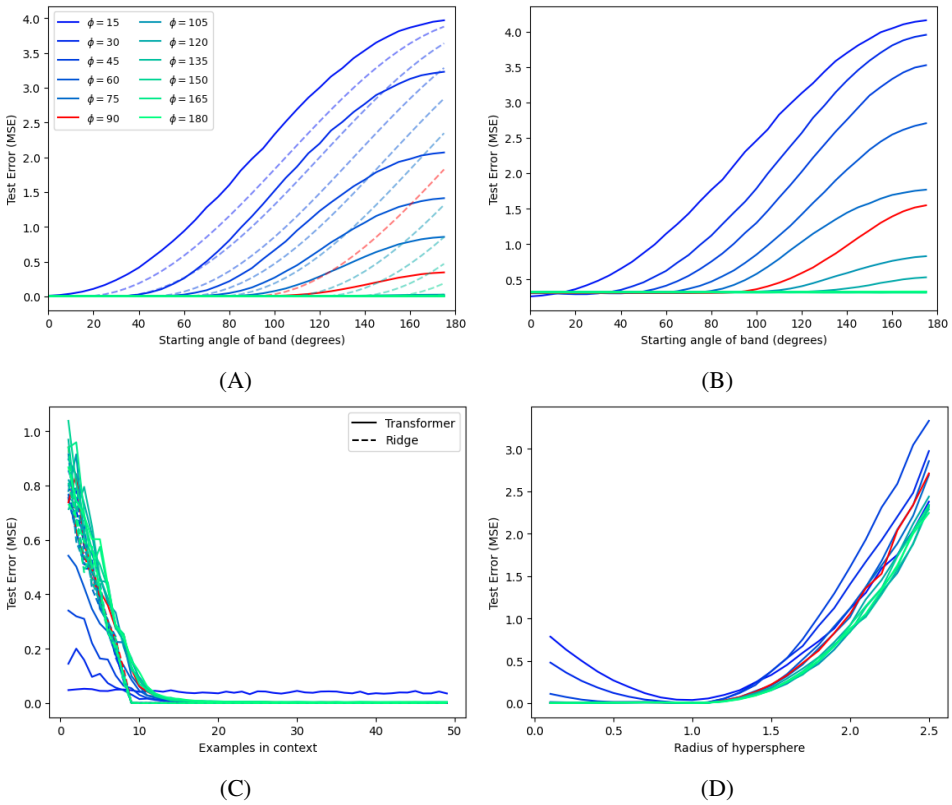
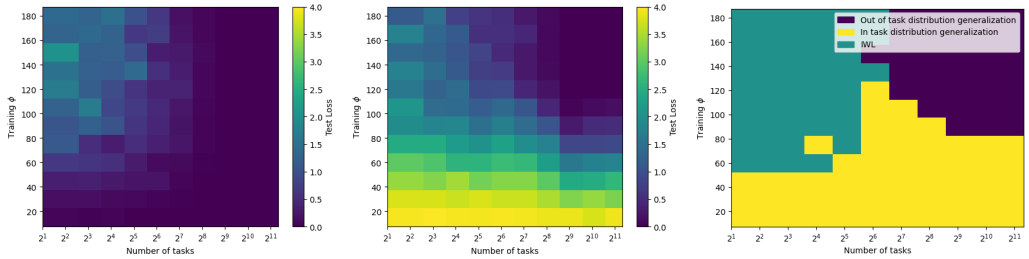


Figure 2: **ICL undergoes a transition between a specialized solution and a general-purpose solution. A:** Test error in $\Delta\phi = 5^\circ$ bands (see Fig 1) for transformers pretrained to do in-context learning of linear functions with pretraining task distributions $p_\phi(w)$. For distributions with $\phi < 90^\circ$, the transformer learns a specialized solution that performs well on unseen tasks drawn from the $p_\phi(w)$, but fails for tasks outside this distribution. However, for pretraining distributions with $\phi > 90^\circ$, the transformer learns a solution that performs well for all ϕ. Here, the noise $\sigma^2 = 0$. Dashed lines show the MSE if the weight vector in $p_\phi(w)$ closest to the test weight vector is used as the predictor. **B:** With $\sigma^2 = 0.25$, we still observe a transition from a specialized to a generic solution, but the transition point has moved to $\phi = 120^\circ$. **C:** We evaluate the models in task-distribution for varying context lengths, and plot the performance of the transformer (solid) and ordinary least squares (dashed) for the same data. For low context length, the specialized solution learned by models with $\phi < 90^\circ$ outperforms OLS. For $\phi = 15^\circ$, the specialized solution is worse than OLS for large context length. **D:** The test error for tasks drawn uniformly from subsets of a hypersphere of radius R, when a model is pretrained on tasks taken only from subsets of the unit hypersphere. When $\phi > 45^\circ$, the model generalizes perfectly to tasks with $R < 1$, despite being pretrained with $R = 1$.

the loss for a test angle between 0° and 5° (these test angles are *always* in the training task distribution). We see that models with low N and large ϕ perform poorly in-distribution, suggesting that the density of tasks may be important. See Section A.1 for a partial explanation as to how some models trained with a small number of tasks (bottom left in Fig 3A) appear to generalize well. In Fig 3B, we plot the resulting *out of task-distribution* loss, corresponding to test angles 175° to 180°. We see that models with small ϕ perform poorly, and observe a diagonal boundary dividing models that generalize well and those that do not, suggesting interplay between these two forms of task diversity. In 3C, we summarize these results as a phase diagram, depicting three distinct phases:

1. Good generalization both in- and out-of-task-distribution (top right).
2. Good in-task-distribution generalization, poor out-of-task-distribution generalization (bottom).
3. Poor generalization both in- and out-of-task-distribution (top left); the model exhibits only in-weights learning (IWL).

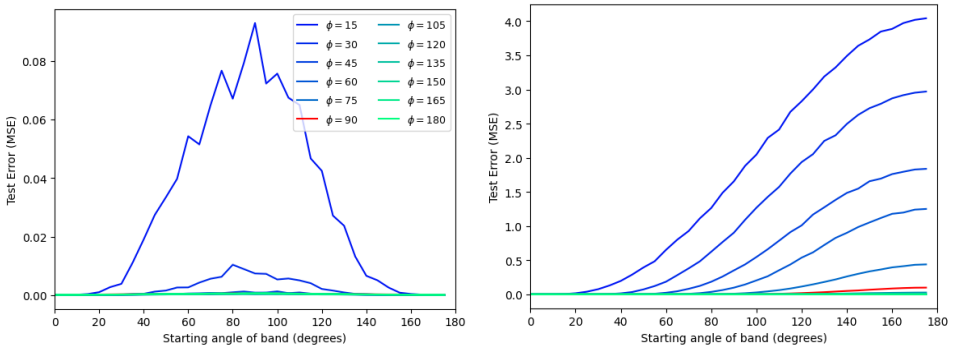


(A) (B) (C)

Figure 3: **Two-axis phase diagrams. A:** Phase diagram for in task-distribution test loss. **B:** Phase diagram for out of task-distribution test loss. **C:** Three phases of in-context learning. In constructing the phase diagram, we set the threshold for high and low generalization losses to 0.5.

3.3 Nonlinear regression

We now change the mapping between input and label for the regression to be a nonlinear function of the weights. Specifically, we consider $y_i = w_2^T \, \mathrm{ReLU}(W_1 x_i)$, with $x_i, w_2 \in \mathbb{R}^d$ and $W_1 \in \mathbb{R}^{d \times d}$. We choose $d = 3$ so that the model has 12 parameters. In Fig 4, we see that specialization-generalization transitions still occur, and investigate two ways of choosing the parameters. In Fig 4A, we pick the full 12-dimensional parameter vector $\theta = \{\mathrm{vec}(W_1), w_2\}$ from the surface of S^{11}. This choice induces a bias towards $\|w_2\| \ll 1$ for angles ϕ near the 'poles' ($v = (\pm 1, \vec{0})^T$). This bias is relaxed, however, when $\phi \sim 90°$, near the equator of the sphere. This leads to nonmonotonic behavior (Fig 4A) – the tasks near the poles are more similar to each other than to those near the equator. In contrast, in Fig 4B, we pick from two separate hyperspheres: $\mathrm{vec}(W_1) \in S^8$ and $w_2 \in S^2$. This choice leads to a qualitatively similar transition to those we see in the linear case.



(A) (B)

Figure 4: **Specialization-generalization transitions in *nonlinear* regression. A:** All parameters in the nonlinear model (a small one-hidden-layer network) are drawn from the same hypersphere. The transition occurs at $\phi \approx 45°$. **B:** The parameters in the nonlinear model are drawn separately from a different hypersphere for each layer in the model. The transition occurs at $\phi \approx 90°$.

4 Discussion and future work

We propose another "axis" to task diversity, distinct from the task diversity measure in [3] (the number of pretraining tasks). This other axis of task diversity, based on subsets of the task space, accounts for the similarity present between tasks. Depending on the level of task diversity present during pretraining, we have shown that transformers learn either a specialized solution that fails to generalize out-of-task-distribution, or a generic solution with good performance across the entire task space.

Our experiments open a new direction for understanding how general-purpose models are able to solve unseen tasks using only a few examples in their context: We show empirically that transformers

can learn to do ICL over much more of the task space than they are trained on. Understanding the generality of this behavior may help explain why language models are able to perform well on ICL tasks not present in their pretraining distribution. Although our experiments here are limited by their focus on relatively simple functions as the ICL task, we believe investigations into specialization-generalization transitions for more complex tasks are a promising direction for future study. Building trust in LLMs is an important challenge with positive societal impacts, and understanding the degree and nature of task generalization via ICL takes a step towards this goal.

Acknowledgments and Disclosure of Funding

LMS is supported by the National Science Foundation Graduate Research Fellowship Program under Grant No. DGE-2039656. VN acknowledges research funds from the University of Sydney. DJS was partially supported by a Simons Fellowship in the MMLS, a Sloan Fellowship, and the National Science Foundation, through the Center for the Physics of Biological Function (PHY-1734030).

References

[1] Ashish Vaswani et al. "Attention is All you Need". In: *Advances in Neural Information Processing Systems*. Ed. by I. Guyon et al. Vol. 30. Curran Associates, Inc., 2017. URL: https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

[2] Tom Brown et al. "Language Models are Few-Shot Learners". In: *Advances in Neural Information Processing Systems*. Ed. by H. Larochelle et al. Vol. 33. Curran Associates, Inc., 2020, pp. 1877–1901. URL: https://proceedings.neurips.cc/paper_files/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf.

[3] Allan Raventos et al. "Pretraining task diversity and the emergence of non-Bayesian in-context learning for regression". In: *Thirty-seventh Conference on Neural Information Processing Systems*. 2023. URL: https://openreview.net/forum?id=BtAz4a5xDg.

[4] Yue M. Lu et al. *Asymptotic theory of in-context learning by linear attention*. 2024. arXiv: 2405.11751 [stat.ML]. URL: https://arxiv.org/abs/2405.11751.

[5] Shivam Garg et al. *What Can Transformers Learn In-Context? A Case Study of Simple Function Classes*. en. arXiv:2208.01066 [cs]. Aug. 2023. URL: http://arxiv.org/abs/2208.01066 (visited on 05/10/2024).

[6] Stephanie C. Y. Chan et al. *Data Distributional Properties Drive Emergent In-Context Learning in Transformers*. 2022. arXiv: 2205.05055 [cs.LG]. URL: https://arxiv.org/abs/2205.05055.

[7] Ilya Loshchilov and Frank Hutter. "Fixing Weight Decay Regularization in Adam". In: *CoRR* abs/1711.05101 (2017). arXiv: 1711.05101. URL: http://arxiv.org/abs/1711.05101.

[8] Alec Radford et al. *Language Models are Unsupervised Multitask Learners*. Tech. rep. OpenAI, 2019.

[9] Adam Paszke et al. "PyTorch: An Imperative Style, High-Performance Deep Learning Library". In: *Advances in Neural Information Processing Systems*. Vol. 32. Curran Associates, Inc., 2019. URL: https://proceedings.neurips.cc/paper_files/paper/2019/hash/bdbca288fee7f92f2bfa9f7012727740-Abstract.html (visited on 09/18/2024).

A Appendix / supplemental material

Further training details: All code was written in Python using the PyTorch library [9]. All models were trained for 58,000 steps using a batch size of 128 and a constant learning rate of 3×10^{-4}. All models were converged at the end of training. All models were trained on a single GPU, either a MIG GPU with 10GB of memory or an A100 with 40GB of memory, and took $\sim$ 3hrs to train.

A.1 Two stages of specialization

In Fig 6A, we compare the training loss of a transformer trained normally on data with $\phi = 45^\circ$ to a transformer trained on modified data. To modify the data, we zero out all components of x_i

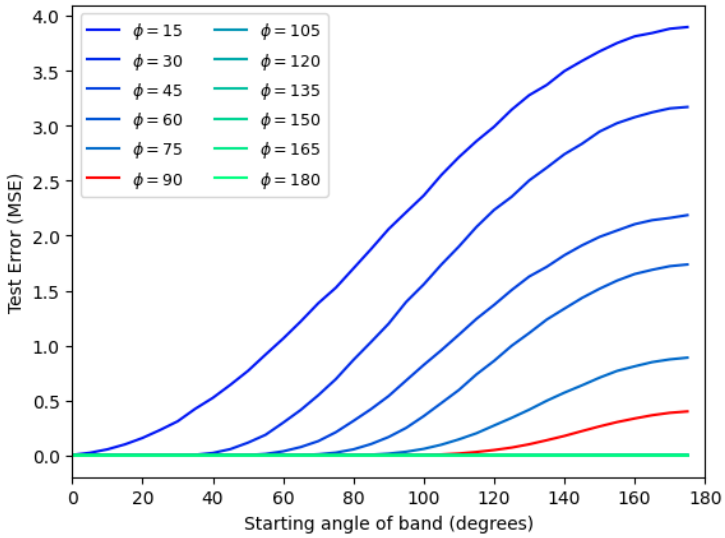
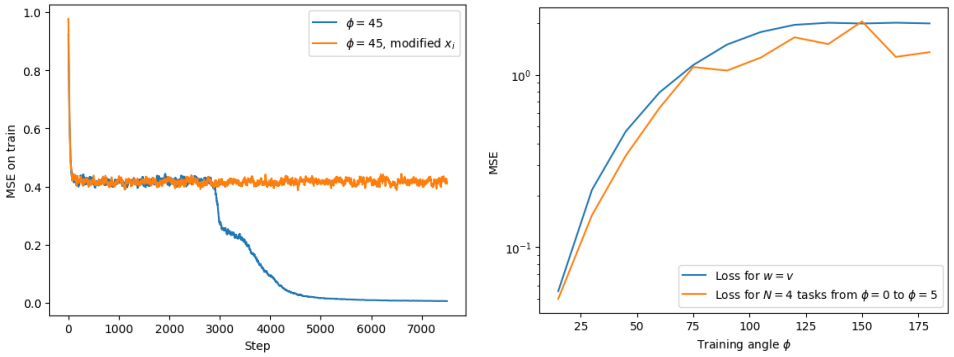


Figure 5: A second run of Fig 2A, with a different initialization and sampling of $w \sim p_\phi$.



(A) (B)

Figure 6: **Transformers undergo two stages of specialization during training: A:** For small ϕ, the transformer quickly learns a solution that only takes into account the component of x_i in the direction of the vector v forming the center of the hyperspherical cap. *Blue:* A transformer trained normally on training data with $\phi = 45°$. *Orange:* A transformer trained on data with the components of x_i perpendicular to v zeroed out. The training loss is smoothed with an exponential moving average for clarity of visualization. **B:** For low task number, the in-task distribution loss (*orange*) tracks the loss for a regression weight vector $w = v$ (*blue*).

that are perpendicular to the vector v defining the center of the hyperspherical training cap. We see that during early stages of training, the transformer trained on unmodified data performs similarly to the transformer trained on modified data, suggesting that early in training, transformers trained to do linear regression only take into account the component of x_i parallel to v. Later in training, the unmodified transformer learns to take into account other directions in the training data. This suggests that there are two distinct specialized solutions learned by transformers when ϕ is small.

In Fig 6B, we compare the in-task-distribution test loss for a model with the regression weight vector $w = v$ with the in-task distribution loss for models trained on a low number of tasks. We see that these two quantities closely track each other, suggesting that transformers are able to learn the first stage of specialized solution even when the number of tasks is low.