Transformers as Statisticians: Provable In-Context Learning with In-Context Algorithm Selection

Yu Bai[*]
Salesforce Research
yu.bai@salesforce.com

Fan Chen[*]
Massachusetts Institute of Technology
fanchen@mit.edu

Huan Wang
Salesforce Research
huan.wang@salesforce.com

Caiming Xiong
Salesforce Research
cxiong@salesforce.com

Song Mei[*]
UC Berkeley
songmei@berkeley.edu

Abstract

Neural sequence models based on the transformer architecture have demonstrated remarkable *in-context learning* (ICL) abilities, where they can perform new tasks when prompted with training and test examples, without any parameter update to the model. This work first provides a comprehensive statistical theory for transformers to perform ICL. Concretely, we show that transformers can implement a broad class of standard machine learning algorithms in context, such as least squares, ridge regression, Lasso, learning generalized linear models, and gradient descent on two-layer neural networks, with near-optimal predictive power on various in-context data distributions. Using an efficient implementation of in-context gradient descent as the underlying mechanism, our transformer constructions admit mild size bounds, and can be learned with polynomially many pretraining sequences.

Building on these "base" ICL algorithms, intriguingly, we show that transformers can implement more complex ICL procedures involving *in-context algorithm selection*, akin to what a statistician can do in real life—A *single* transformer can adaptively select different base ICL algorithms—or even perform qualitatively different tasks—on different input sequences, without any explicit prompting of the right algorithm or task. We both establish this in theory by explicit constructions, and also observe this phenomenon experimentally. In theory, we construct two general mechanisms for algorithm selection with concrete examples: pre-ICL testing, and post-ICL validation. As an example, we use the post-ICL validation mechanism to construct a transformer that can perform nearly Bayes-optimal ICL on a challenging task—noisy linear models with mixed noise levels. Experimentally, we demonstrate the strong in-context algorithm selection capabilities of standard transformer architectures.

1 Introduction

Large neural sequence models have demonstrated remarkable *in-context learning* (ICL) capabilities [12], where models can make accurate predictions on new tasks when prompted with training examples from the same task, in a zero-shot fashion without any parameter update to the model. A prevalent example is large language models based on the transformer architecture [84], which can perform a diverse range of tasks in context when trained on enormous text [12, 90]. Recent models

[*]Equal technical and directional contributions.
Code is available at https://github.com/allenbai01/transformers-as-statisticians.

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

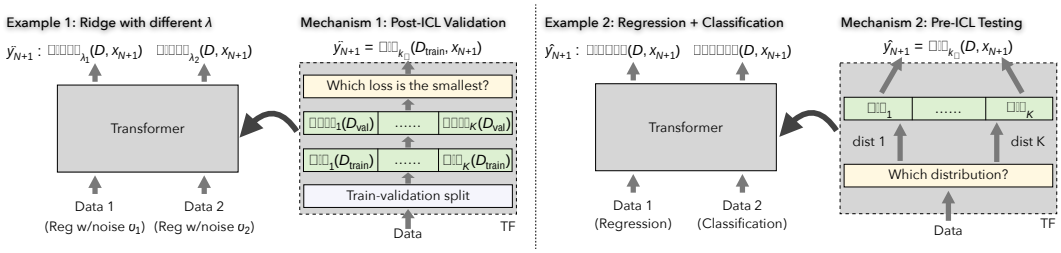

Figure 1: **Illustration of in-context algorithm selection, and two mechanisms constructed in our theory.** *Left, middle-left*: A single transformer can perform ridge regression with different 's on input sequences with different observation noise; we prove this by the **post-ICL validation** mechanism (Section 4.1). *Middle-right, right*: A single transformer can perform linear regression on regression data and logistic regression on classification data; we prove this via the **pre-ICL testing** mechanism (Section 4.2).

in this paradigm such as GPT-4 achieve surprisingly impressive ICL performance that makes them akin to a general-purpose agent in many aspects [65, 14]. Such strong capabilities call for better understandings, which a recent line of work tackles from various aspects [49, 94, 28, 72, 15, 57, 64].

Recent pioneering work of Garg et al. [31] proposes an interpretable and theoretically amenable setting for understanding ICL in transformers. They perform ICL experiments where input tokens are real-valued (input, label) pairs generated from standard statistical models such as linear models (and the sparse version), neural networks, and decision trees. Garg et al. [31] find that transformers can learn to perform ICL with prediction power (and fitted functions) matching standard machine learning algorithms for these settings, such as least squares for linear models, and Lasso for sparse linear models. Subsequent work further studies the internal mechanisms [2, 86, 18], expressive power [2, 32], and generalization [47] of transformers in this setting. However, these works only showcase simple mechanisms such as regularized regression [31, 2, 47] or gradient descent [2, 86, 18], which are arguably only a small subset of what transformers are capable of in practice; or expressing universal function classes not specific to ICL [89, 32]. This motivates the following question:

How do transformers learn in context beyond implementing simple algorithms?

This paper makes steps on this question by making two main contributions: (1) We **unveil a general mechanism—*in-context algorithm selection***—by which a *single* transformer can adaptively *select different "base" ICL algorithms* to use on *different ICL instances*, without any explicit prompting of the right algorithm to use in the input sequence. For example, a transformer may choose to perform ridge regression with regularization $_1$ on ICL instance 1, and $_2$ on ICL instance 2 (Figure 2); or perform regression on ICL instance 1 and classification on ICL instance 2 (Figure 5). This adaptivity allows transformers to achieve much stronger ICL performance than the base ICL algorithms. We both prove this in theory, and demonstrate this phenomenon empirically on standard transformer architectures. (2) Along the way, equally importantly, we present a comprehensive theory for ICL in transformers by establishing end-to-end quantitative guarantees for the **expressive power, in-context prediction performance, and sample complexity of pretraining**. These results add upon the recent line of work on the statistical learning theory of transformers [97, 89, 27, 39], and lay out a foundation for the intriguing special case where the *learning targets are themselves ICL algorithms*.

A detailed summary of our contributions is as follows.

- We prove that transformers can implement a broad class of standard machine learning algorithms in context, such as least squares, ridge regression, Lasso, convex risk minimization for learning generalized linear models (such as logistic regression), and gradient descent for two-layer neural networks (Section 3). Our constructions admit mild bounds on the number of layers, heads, and weight norms, and achieve near-optimal prediction power on many in-context data distributions.
- Technically, the above transformer constructions build on a new efficient implementation of in-context gradient descent (Appendix D), which could be broaderly applicable. For a broad class of smooth convex empirical risks over the in-context training data, we construct an $(L + 1)$-layer transformer that approximates L steps of gradient descent. Notably, the approximation error accumulates only *linearly* in L, utilizing a stability-like property of smooth convex optimization.
- We prove that transformers can perform in-context algorithm selection (Section 4). We construct two algorithm selection mechanisms: Post-ICL validation (Section 4.1), and Pre-ICL testing

(a) Noisy linear reg with noise $_1$ (b) Noisy linear reg with noise $_2$ (c) Task 1 vs. task 2 at token 20

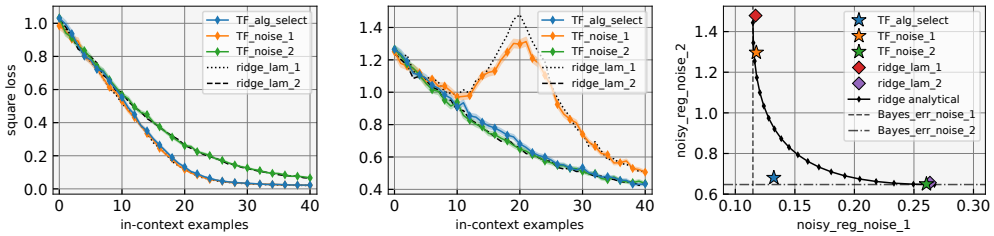

Figure 2: In-context algorithm selection on two separate noisy linear regression tasks with noise ($_1$; $_2$) = (0:1; 0:5). *(a,b)* A **single transformer** `TF_alg_select` **simultaneously approaches the performance of the two individual Bayes predictors** `ridge_lam_1` on task 1 and `ridge_lam_2` on task 2. *(c)* At token 20 (using example $f0; : : : ; 19g$ for training), `TF_alg_select` approaches the Bayes error on two tasks simultaneously, and **outperforms ridge regression with any fixed** . *(a,b,c)* Note that transformers pretrained on a single task (`TF_noise_1`, `TF_noise_2`) perform near-optimally on that task but suboptimally on the other task. More details about the setup and training method can be found in Appendix M.2.

(Section 4.2). For both mechanisms, we provide general constructions as well as concrete examples. Figure 1 provides a pictorial illustration of the two mechanisms.

- As a concrete application, using the post-ICL validation mechanism, we construct a transformer that can perform nearly Bayes-optimal ICL on noisy linear models with *mixed* noise levels (Section 4.1.1), a more complex task than those considered in existing work.
- We provide the first line of results for *pretraining* transformers to perform the various ICL tasks above, from polynomially many training sequences (Section 5 & Appendix K).
- Experimentally, we find that learned transformers indeed exhibit strong in-context algorithm selection capabilities in the settings considered in our theory (Section 6). For example, Figure 2 shows that a *single* transformer can approach the individual Bayes risks (the optimal risk among all possible algorithms) simultaneously on two noisy linear models with different noise levels.

Transformers as statisticians We humbly remark that the typical toolkit of a statistician contains much more beyond those covered in this work, including and not limited to inference, uncertainty quantification, and theoretical analysis. This work merely aims to show the algorithm selection capability of transformers, akin to what a statistician *can* do.

Related work Our work is intimately related to the lines of work on in-context learning, theoretical understandings of transformers, as well as other formulations for learning-to-learn such as meta-learning. Due to limited space, we discuss these related work in Appendix A.

2 Preliminaries

We consider a sequence of N input vectors $fh_i g_{i=1}^N$ R^D, written compactly as an input matrix $H = [h_1; : : : ; h_N]$ 2 $R^{D\times N}$, where each h_i is a column of H (also a *token*). Throughout this paper, we let $(t) := \mathrm{ReLU}(t) = \max ft; 0g$ denote the standard relu activation.

2.1 Transformers

We consider transformer architectures that process any input sequence H 2 $R^{D\times N}$ by applying (encoder-mode[2]) attention layers and MLP layers formally defined as follows.

Definition 1 (Attention layer)**.** *A (self-)attention layer with M heads is denoted as* Attn ( ) *with parameters* = $f(V_m; Q_m; K_m)g_{m\in[M]}$ $R^{D\times D}$. *On any input sequence* H 2 $R^{D\times N}$,

$$\text{Attn}(H) := H + \frac{1}{N}\sum_{m=1}^{M}(V_m H)\quad (Q_m H)^\top (K_m H)\ 2\ R^{D\times N}; \tag{1}$$

where : R ! R *is the ReLU function. In vector form,*

$$[\text{Attn}(H)]_i = h_i + \sum_{m=1}^{M}\frac{1}{N}\sum_{j=1}^{N}(hQ_m h_i; K_m h_j i)\ V_m h_j:$$

[2]Many of our results can be generalized to decoder-based architectures; see Appendix C for a discussion.

Above, (1) uses a normalized ReLU[3] activation $t \mapsto \mathrm{relu}(t)/N$ in place of the standard softmax activation; we remark this activation is also found to work well empirically in recent studies [78, 93].

Definition 2 (MLP layer)**.** *A (token-wise) MLP layer with hidden dimension D' is denoted as* $\mathrm{MLP}(\cdot)$ *with parameters* $= (W_1, W_2) \in \mathbb{R}^{D' \times D} \times \mathbb{R}^{D \times D'}$*. On any input sequence* $H \in \mathbb{R}^{D \times N}$*,*

$$\mathrm{MLP}(H) := H + W_2\,\mathrm{relu}(W_1 H),$$

where $\mathrm{relu} : \mathbb{R} \to \mathbb{R}$ *is the ReLU function. In vector form, we have* $\mathrm{MLP}(h_i) = h_i + W_2\,\mathrm{relu}(W_1 h_i)$.

We consider a transformer architecture with L transformer layers, each consisting of a self-attention layer followed by an MLP layer.

Definition 3 (Transformer)**.** *An L-layer transformer, denoted as* $\mathrm{TF}(\cdot)$*, is a composition of L self-attention layers each followed by an MLP layer:* $H^{(L)} = \mathrm{TF}(H^{(0)})$*, where* $H^{(0)} \in \mathbb{R}^{D \times N}$ *is the input sequence, and*

$$H^{(\ell)} = \mathrm{MLP}(\theta^{(\ell)}_{\mathrm{mlp}})\,\mathrm{Attn}(\theta^{(\ell)}_{\mathrm{attn}})(H^{(\ell-1)}) \quad ; \quad \ell \in \{1, \dots, L\}.$$

Above, the parameter $\theta = (\theta^{(1:L)}_{\mathrm{attn}}, \theta^{(1:L)}_{\mathrm{mlp}})$ *consists of the attention layers* $\theta^{(\ell)}_{\mathrm{attn}} = (V^{(\ell)}_m, Q^{(\ell)}_m, K^{(\ell)}_m)_{m \in [M^{(\ell)}]} \in \mathbb{R}^{D \times D}$ *and the MLP layers* $\theta^{(\ell)}_{\mathrm{mlp}} = (W^{(\ell)}_1, W^{(\ell)}_2) \in \mathbb{R}^{D^{(\ell)} \times D} \times \mathbb{R}^{D \times D^{(\ell)}}$*. We will frequently consider "*attention-only*" transformers with* $W^{(\ell)}_1, W^{(\ell)}_2 = 0$*, which we denote as* $\mathrm{TF}^0(\cdot)$ *for shorthand, with* $\theta = \theta^{(1:L)} := \theta^{(1:L)}_{\mathrm{attn}}$.

We additionally define the following norm of a transformer TF:

$$|||\mathrm{TF}||| := \max_{\ell \in [L]} \max_{m \in [M]} \Big\{ \|Q^{(\ell)}_m\|_{\mathrm{op}}, \|K^{(\ell)}_m\|_{\mathrm{op}} + \frac{1}{M} \sum_{m=1}^{M} \|V^{(\ell)}_m\|_{\mathrm{op}} + \|W^{(\ell)}_1\|_{\mathrm{op}} + \|W^{(\ell)}_2\|_{\mathrm{op}} \Big\}. \quad (2)$$

In (2), the choices of the operator norm and max/sums are for convenience only and not essential, as our results (e.g. for pretraining) depend only logarithmically on $|||\mathrm{TF}|||$.

2.2 In-context learning

In an in-context learning (ICL) instance, the model is given a dataset $D = \{(x_i, y_i)\}_{i \in [N]} \overset{\text{iid}}{\sim} P$ and a new test input $x_{N+1} \sim P_X$ for some data distribution P, where $\{x_i\}_{i \in [N]} \in \mathbb{R}^d$ are the input vectors, $\{y_i\}_{i \in [N]} \in \mathbb{R}$ are the corresponding labels (e.g. real-valued for regression, or $\{0, 1\}$-valued for binary classification), and x_{N+1} is the test input on which the model is required to make a prediction. Different from standard supervised learning, in ICL, each instance (D, x_{N+1}) is in general drawn from a different distribution P_j, such as a linear model with a new ground truth coefficient $w_{?,j} \in \mathbb{R}^d$. Our goal is to construct *fixed* transformer to perform ICL on a large set of P_j's.

We consider using transformers to perform ICL, in which we encode (D, x_{N+1}) into an input sequence $H \in \mathbb{R}^{D \times (N+1)}$. In our theory, we use the following format, where the first two rows contain (D, x_{N+1}) (zero at the location for y_{N+1}), and the third row contains fixed vectors $\{p_i\}_{i \in [N+1]}$ with ones, zeros, and indicator for being the train token (similar to a positional encoding vector):

$$H = \begin{bmatrix} x_1 & x_2 & \cdots & x_N & x_{N+1} \\ y_1 & y_2 & \cdots & y_N & 0 \\ p_1 & p_2 & \cdots & p_N & p_{N+1} \end{bmatrix} \in \mathbb{R}^{D \times (N+1)}, \quad p_i := \begin{bmatrix} 0_{D-(d+3)} \\ 1 \\ 1\{i < N+1\} \end{bmatrix} \in \mathbb{R}^{D-(d+1)}. \quad (3)$$

We will choose $D = \Theta(d)$, so that the hidden dimension of H is at most a constant multiple of d. We then feed H into a transformer to obtain the output $\hat{H} = \mathrm{TF}(H) \in \mathbb{R}^{D \times (N+1)}$ with the same shape, and *read out* the prediction $\hat{y}_{N+1}$ from the $(d+1, N+1)$-th entry of $\hat{H} = [\hat{h}_i]_{i \in [N+1]}$ (the entry corresponding to the missing test label): $\hat{y}_{N+1} = \mathrm{read}_y(\hat{H}) := (\hat{h}_{N+1})_{d+1}$. The goal is to predict

[3] For each query index i, the attention weights $\{\mathrm{softmax}(\langle Q_m h_i, K_m h_j \rangle / N)\}_{j \in [N]}$ is also a set of non-negative weights that sum to $O(1)$ (similar as a softmax probability distribution) in typical scenarios. Also, our approximation results can potentially be generalized to softmax attention e.g. using the technique of [32].

$\hat{y}_{N+1}$ that is close to $y_{N+1} \sim P_{y|x_{N+1}}$ measured by proper losses. We emphasize that we consider predicting only at the last token x_{N+1}, which is without much loss of generality.[4]

Miscellaneous setups We assume bounded features and labels throughout the paper (unless otherwise specified, e.g. when x_i is Gaussian): $\|x_i\|_2 \le B_x$ and $|y_i| \le B_y$ with probability one. We use the standard notation $X = [x_1^\top; \ldots; x_N^\top] \in \mathbb{R}^{N \times d}$ and $y = [y_1; \ldots; y_N] \in \mathbb{R}^N$ to denote the matrix of inputs and vector of labels, respectively. To prevent the transformer from blowing up on tail events, in all our results concerning (statistical) in-context prediction powers, we consider a clipped prediction $\hat{y}_{N+1} = \mathrm{read}_y(H) := \mathrm{clip}_R((H_{N+1})_{d+1})$, where $\mathrm{clip}_R(t) := \mathrm{Proj}_{[-R;R]}(t)$ is the standard clipping operator with (a suitably large) radius $R > 0$ that varies in different problems.

3 Basic in-context learning algorithms

We begin by constructing transformers that approximately implement a variety of standard machine learning algorithms in context, with mild size bounds and near-optimal prediction power on many standard in-context data distributions.

3.1 In-context ridge regression and least squares

Consider the standard ridge regression estimator over the in-context training examples D with regularization $\lambda > 0$ (reducing to least squares at $\lambda = 0$ and $N \ge d$):

$$w_{\mathrm{ridge}} := \arg\min_{w \in \mathbb{R}^d} \frac{1}{2N} \sum_{i=1}^N (\langle w, x_i \rangle - y_i)^2 + \frac{\lambda}{2} \|w\|_2^2. \tag{ICRidge}$$

We show that transformers can approximately implement (ICRidge) (proof in Appendix F.1).

Theorem 4 (Implementing in-context ridge regression)**.** *For any $\varepsilon > 0$, $\lambda \ge 0$ with $\kappa := \frac{\lambda + 1}{\lambda}$, $B_w > 0$, and $\varepsilon < B_x B_w / 2$, there exists an L-layer attention-only transformer TF^0 with*

$$L = d \cdot \log(B_x B_w / (2\varepsilon)) \cdot \kappa + 1, \quad \max_{\ell \in [L]} M^{(\ell)} \le 3, \quad \max_{\ell \in [L]} \|W^{(\ell)}\| \le 4R + 8(\lambda + 1)^{-1}, \tag{4}$$

(with $R := \max\{B_x B_w, B_y, 1\}$) such that the following holds. On any input data (D, x_{N+1}) such that the problem (ICRidge) is well-conditioned and has a bounded solution:

$$\lambda_{\min}(X^\top X / N) \ge \lambda \quad \text{and} \quad \lambda_{\max}(X^\top X / N) \le \kappa, \quad \|w_{\mathrm{ridge}}\|_2 \le B_w / 2, \tag{5}$$

TF^0 approximately implements (ICRidge): The prediction $\hat{y}_{N+1} = \mathrm{read}_y(\mathrm{TF}^0(H))$ satisfies

$$\hat{y}_{N+1} \approx \langle w_{\mathrm{ridge}}, x_{N+1} \rangle \pm \varepsilon. \tag{6}$$

Theorem 4 presents the first quantitative construction for end-to-end in-context ridge regression up to arbitrary precision, and improves upon Akyürek et al. [2] whose construction does not give (or directly imply) an explicit error bound like (6). Further, the bounds on the number of layers and heads in (4) are mild (constant heads and logarithmically many layers).

Near-optimal in-context prediction power for linear problems Combining Theorem 4 with standard analyses of linear regression yields the following corollaries (proofs in Appendix F.3 & F.4).

Corollary 5 (Near-optimal linear regression with transformers by approximating least squares)**.** *For any $N \ge \Theta(d)$, there exists an $O(\log(N/\varepsilon))$-layer transformer TF, such that on any P satisfying standard statistical assumptions for least squares (Assumption A), its ICL prediction $\hat{y}_{N+1}$ achieves*

$$\mathbb{E}_{(D; x_{N+1}; y_{N+1}) \sim P}[(\hat{y}_{N+1} - y_{N+1})^2] \le \inf_w \mathbb{E}_{(x,y) \sim P} (y - \langle w, x \rangle)^2 + \Theta(d^2/N).$$

Assumption A requires only generic tail properties such as sub-Gaussianity, and *not* realizability (i.e., P follows a true linear model); κ; above denote the covariance condition number and the

[4] Our constructions may be generalized to predicting at every token, by using a decoder architecture and potentially different input formats correspondingly (cf. Appendix C). Our theory focuses on predicting at the last token only, which simplifies the setting. Our experiments test both settings.

noise level therein. The $\widetilde{O}(d\ ^2/N)$ excess risk is known to be rate-optimal for linear regression [38], and Corollary 5 achieves this in context with a transformer with only logarithmically many layers.

Next, consider Bayesian linear models where each in-context data distribution $P = P^{\mathrm{lin}}_{\mathbf{w}_?}$ is drawn from a Gaussian prior $\ : \mathbf{w}_? \ \ N(0; \mathrm{I}_d/d)$, and $(\mathbf{x}; y) \ \ P^{\mathrm{lin}}_{\mathbf{w}_?}$ is sampled as $\mathbf{x} \ \ N(0; \mathrm{I}_d)$, $y = \langle \mathbf{w}_?; \mathbf{x} \rangle + N(0; \ ^2)$. It is a standard result that the Bayes estimator of y_{N+1} given $(D; \mathbf{x}_{N+1})$ is given by ridge regression (ICRidge): $\hat{y}^{\mathrm{Bayes}}_{N+1} := \langle \mathbf{w}_{\mathrm{ridge}}; \mathbf{x}_{N+1} \rangle$ with $\ = d\ ^2/N$. We show that transformers achieve nearly-Bayes risk for this problem, and we use

$$\mathrm{BayesRisk} \ := \mathsf{E}_{\mathbf{w}_? \sim \ ; (\mathcal{D}; \mathbf{x}_{N+1}; y_{N+1}) \sim P^{\mathrm{lin}}_{\mathbf{w}_?}} \left[\frac{1}{2} \left(\hat{y}^{\mathrm{Bayes}}_{N+1} \ \ y_{N+1} \right)^2 \right]$$

to denote the Bayes risk of this problem under prior $\ $.

Corollary 6 (Nearly-Bayes linear regression with transformers by approximating ridge regression)**.** *Under the Bayesian linear model above with* $N \ \ \max\{d/10; O(\log(1/\varepsilon))\}$*, there exists a* $L = O(\log(1/\varepsilon))$*-layer transformer such that* $\mathsf{E}_{\mathbf{w}_?; (\mathcal{D}; \mathbf{x}_{N+1}; y_{N+1})} \frac{1}{2} (\hat{y}_{N+1} \ \ y_{N+1})^2 \ \ \mathrm{BayesRisk} \ + \varepsilon$.

Generalized linear models In Appendix G, we extend the above results to generalized linear models [53] and show that transformers can approximate the corresponding convex risk minimization algorithm in context (which includes logistic regression for linear classification as an important special case), and achieve near-optimal excess risk under standard statistical assumptions.

3.2 In-context Lasso

Consider the standard Lasso estimator [82] which minimizes an ℓ_1-regularized linear regression loss $\hat{L}_{\mathrm{lasso}}$ over the in-context training examples D:

$$\mathbf{w}_{\mathrm{lasso}} := \arg\min_{\mathbf{w} \in \mathbb{R}^d} \hat{L}_{\mathrm{lasso}}(\mathbf{w}) = \frac{1}{2N} \sum_{i=1}^{N} (\langle \mathbf{w}; \mathbf{x}_i \rangle \ \ y_i)^2 + \ _N \|\mathbf{w}\|_1 : \qquad \text{(ICLasso)}$$

We show that transformers can also approximate in-context Lasso with a mild number of layers, and can perform sparse linear regression in standard sparse linear models (proofs in Appendix H).

Theorem 7 (Implementing in-context Lasso)**.** *For any* $_N \ \ 0, \ \ > 0$*,* $B_w > 0$*, and* $\varepsilon > 0$*, there exists a* L*-layer transformer* $\mathsf{TF} \ \ $ *with*

$$L = \ \ B_w^2/\varepsilon \ + 1; \quad \max_{\ell \in [L]} M^{(\ell)} \ \ 2; \quad \max_{\ell \in [L]} D^{(\ell)} \ \ 2d; \quad \|\mathsf{TF}\| \ \|\mathsf{TF}\| \ \ O\left(R + (1 + \ _N)^{\ -1}\right)$$

(where $R := \max\{B_x B_w; B_y; 1\}$*) such that the following holds. On any input data* $(D; \mathbf{x}_{N+1})$ *such that* $\ _{\max}(\mathbf{X}^\top \mathbf{X}/N) \ \ $ *and* $\|\mathbf{w}_{\mathrm{lasso}}\|_2 \ \ B_w/2$*,* $\mathsf{TF}\ (\mathrm{H}^{(0)})$ *approximately implements (ICLasso), in that it outputs* $\hat{y}_{N+1} = \langle \mathbf{x}_{N+1}; \hat{\mathbf{w}} \rangle$ *with* $\hat{L}_{\mathrm{lasso}}(\hat{\mathbf{w}}) \ \ \hat{L}_{\mathrm{lasso}}(\mathbf{w}_{\mathrm{lasso}}) \ \ \varepsilon$.

Theorem 8 (Near-optimal sparse linear regression with transformers by approximating Lasso)**.** *For any* $d; N \ \ 1; \ \ > 0; B_w^?; \ \ > 0$*, there exists a* $\widetilde{O}((B_w^?)^2/\ ^2 \ \ (1 + (d/N)))$*-layer transformer such that the following holds: For any* s *and* $N \ \ O(s \log(d/\ ))$*, suppose that* P *is an* s*-sparse linear model:* $\mathbf{x}_i \ \ N(0; \mathrm{I}_d)$*,* $y_i = \langle \mathbf{w}_?; \mathbf{x}_i \rangle + N(0; \ ^2)$ *for any* $\|\mathbf{w}_?\|_2 \ \ B_w^?$ *and* $\|\mathbf{w}_?\|_0 \ \ s$*, then with probability at least* $1 \ \ $ *(over the randomness of* D*), the transformer output* $\hat{y}_{N+1}$ *achieves*

$$\mathsf{E}_{(\mathbf{x}_{N+1}; y_{N+1}) \sim P} \ (\hat{y}_{N+1} \ \ y_{N+1})^2 \ \ \ ^2[1 + O(s \log(d/\ )/N)].$$

The $\widetilde{O}(s \log d/N)$ excess risk obtained in Theorem 8 is optimal up to log factors [62, 87]. We remark that Theorem 8 is not a direct corollary of Theorem 7; Rather, the bound on the number of layers in Theorem 8 requires a sharper convergence analysis of the (ICLasso) problem under sparse linear models (Appendix H.2), similar to [1].

3.3 Proof technique: In-context gradient descent

The constructions in Section 3.1 and 3.2 is built on the following result for approximating in-context (proximal) gradient descent on (regularized) convex losses.

Theorem 9 (ICGD; Informal version of Theorem D.1 & D.2)**.** *For a broad class of convex losses of form* $\mathbf{w} \ ? \ \frac{1}{N} \sum_{i=1}^{N} \ell(\mathbf{w}^\top \mathbf{x}_i; y_i) + R(\mathbf{w})$*, there exists an* L*-layer transformer that takes in any* $(D; \mathbf{w}^0)$ *and outputs* $\hat{\mathbf{w}}^L$ *such that* $\|\hat{\mathbf{w}}^L \ \ \mathbf{w}^L_{\{\mathrm{GD}; \mathrm{PGD}\}}\|_2 \ \ O(L\varepsilon)$*, by composing* L *identical layers each* $O(\varepsilon)$*-approximating a single step of GD (so that* $O(L\varepsilon)$ *is a* linear *error accumulation).*

Theorem 9 is established in two main steps:

- Approximating one-step of ICGD using one attention layer (Proposition E.1), which substantially generalizes that of von Oswald et al. [86] (which only does GD on square losses with a *linear* self-attention), and is simpler than the ones in Akyürek et al. [2] and Giannou et al. [32].
- Stacking L of the above layer to approximate L steps of ICGD. Done naively, the error accumulation of this stacking operation is exponential in L in the worst case. We utilize the stability of *convex* gradient descent (Lemma D.1) to obtain the *linear* in L error accumulation in Theorem 9.

In Appendix D.3, we also give results for *non-convex* GD on two-layer neural nets, though with a worse (exponential in L) error accumulation as expected.

4 In-context algorithm selection

We now show that transformers can perform various kinds of *in-context algorithm selection*, which allows them to implement more complex ICL procedures by adaptively selecting different "base" algorithms on different input sequences. We construct two general mechanisms: *Post-ICL validation*, and *Pre-ICL testing*; See Figure 1 for a pictorial illustration.

4.1 Post-ICL validation mechanism

In our first mechanism, post-ICL validation, the transformer begins by implementing a *train-validation split* $D = (D_{\text{train}}; D_{\text{val}})$, and running K *base* ICL algorithms on D_{train}. Let $f f_k g_{k\in[K]}$ $(\mathbb{R}^d \to \mathbb{R})$ denote the K learned predictors, and

$$\hat{L}_{\text{val}}(f) := \frac{1}{|\mathcal{D}_{\text{val}}|} \sum_{(x_i;y_i)\in\mathcal{D}_{\text{val}}} \ell(f(x_i); y_i) \tag{7}$$

denote the validation loss of any predictor f.

We show that (proof in Appendix I.1) a 3-layer transformer can output a predictor $\hat{f}$ that achieves nearly the smallest validation loss, and thus nearly optimal expected loss if $\hat{L}_{\text{val}}$ concentrates around the expected loss L. Below, the input sequence H uses a generalized positional encoding $p_i := [0_{D-(d+3)}; 1; t_i]$ in (3), where $t_i := 1$ for $i \in D_{\text{train}}$, $t_i := -1$ for $i \in D_{\text{val}}$, and $t_{N+1} := 0$.

Proposition 10 (In-context algorithm selection via train-validation split)**.** *Suppose that $\ell(\cdot;\cdot)$ in (7) is approximable by sum of relus (Definition D.1, which includes all C^3-smooth bivariate functions). Then there exists a 3-layer transformer TF that maps (defining $y'_i = y_i 1\{i < N+1\}$)*

$$h_i = [x_i; y'_i; *; f_1(x_i); \cdots; f_K(x_i); 0_{K+1}; 1; t_i] \to h'_i = [x_i; y'_i; *; \hat{f}(x_i); 1; t_i],\ i \in [N+1];$$

where the predictor $\hat{f}: \mathbb{R}^d \to \mathbb{R}$ is a convex combination of $f f_k : \hat{L}_{\text{val}}(f_k) \le \min_{k'\in[K]} \hat{L}_{\text{val}}(f_{k'}) + \varepsilon g$. As a corollary, for any convex risk $L : (\mathbb{R}^d \to \mathbb{R}) \to \mathbb{R}$, $\hat{f}$ satisfies

$$L(\hat{f}) \le \min_{k'\in[K]} L(f_{k'}) + \max_{k\in[K]} \hat{L}_{\text{val}}(f_k) - L(f_k) + \varepsilon.$$

Ridge regression with in-context regularization selection As an example, we use Proposition 10 to construct a transformer to perform in-context ridge regression with regularization selection according to the *unregularized* validation loss $\hat{L}_{\text{val}}(w) := \frac{1}{2|\mathcal{D}_{\text{val}}|} \sum_{(x_i;y_i)\in\mathcal{D}_{\text{val}}} (\langle w; x_i\rangle - y_i)^2$ (proof in Appendix I.2). Let $\lambda_1, \ldots, \lambda_K \ge 0$ be K fixed regularization strengths.

Theorem 11 (Ridge regression with in-context regularization selection)**.** *There exists a transformer with $O(\log(1/\varepsilon))$ layers and $O(K)$ heads such that the following holds: On any $(D; x_{N+1})$ well-conditioned (cf. (5)) for all $f_k g_{k\in[K]}$, it outputs $\hat{y}_{N+1} = \langle \hat{w}; x_{N+1}\rangle$, where*

$$\text{dist}\left(\hat{w}; \text{conv}\{w^{k}_{\text{ridge;train}} : \hat{L}_{\text{val}}(w^{k}_{\text{ridge;train}}) \le \min_{k'\in[K]} \hat{L}_{\text{val}}(w^{k'}_{\text{ridge;train}}) + \varepsilon\}\right) \le \varepsilon.$$

Above, $w_{\text{ridge;train}}$ denotes the solution to (ICRidge) on the training split D_{train}.

4.1.1 Nearly Bayes-optimal ICL on noisy linear models with mixed noise levels

We build on Theorem 11 to show that transformers can perform nearly Bayes-optimal ICL when data come from noisy linear models with a *mixture of K different noise levels* $_1; \dots; \ _K > 0$.

Concretely, consider the following data generating model, where we first sample $P = P_{\mathbf{w}?; \ k}$ from $k \quad 2 \quad ([K])$, $\mathbf{w}? \quad N(0; I_d/d)$, and then sample data $f(x_i; y_i)g_{i \in [N+1]} \overset{\text{iid}}{} P_{\mathbf{w}?; \ k}$ as

$$P_{\mathbf{w}?; \ k} : x_i \quad N(0; I_d); \quad y_i = hx_i; \mathbf{w}?i + N(0; \ ^2_k):$$

For any fixed $(N; d)$, consider the Bayes risk for predicting y_{N+1} under this model:

$$\text{BayesRisk} \ := \inf_{\mathcal{A}} E \ \ \tfrac{1}{2}(A(D)(x_{N+1}) \quad y_{N+1})^2 \ :$$

By standard Bayesian calculations, the above Bayes risk is attained when A is a certain *mixture of K ridge regressions* with regularization $_k = d \ ^2_k/N$; however, the mixing weights depend on D in a highly non-trivial fashion (see Appendix J.2 for a derivation). By using the post-ICL validation mechanism in Theorem 11, we construct a transformer that achieves nearly the Bayes risk.

Theorem 12 (Nearly Bayes-optimal ICL; Informal version of Theorem J.1)**.** *For sufficiently large $N; d$, there exists a transformer with $O(\log N)$ layers and $O(K)$ heads such that on the above model, it outputs a prediction $\hat{y}_{N+1}$ that is nearly Bayes-optimal:*

$$E \ \ \tfrac{1}{2}(y_{N+1} \quad \hat{y}_{N+1})^2 \quad \text{BayesRisk} \ + O \ \ (\log K/N)^{1/3} \ : \tag{8}$$

In particular, Theorem 12 applies in the *proportional setting* where $N; d$ are large and $N/d = \ (1)$ [22], in which case $\text{BayesRisk} \ = \ (1)$, and thus the transformer achieves vanishing excess risk relative to the Bayes risk at large N.

This substantially strengthens the results of Akyürek et al. [2], who empirically find that transformers can achieve nearly Bayes risk under any *fixed* noise level. By contrast, Theorem 12 shows that a *single* transformer can achieve nearly Bayes risk even under a mixture of K noise levels, with quantitative guarantees. Also, our proof in fact gives a stronger guarantee: The transformer approaches the *individual Bayes risks on all K noise levels simultaneously* (in addition to the overall Bayes risk for $k \quad$ as in Theorem 12). We demonstrate this empirically in Section 6 (cf. Figure 3b & 2).

Exact Bayes predictor vs. Post-ICL validation mechanism As BayesRisk is the theoretical lower bound for the risk of any possible ICL algorithm, Theorem 12 implies that our transformer performs similarly as the exact Bayes estimator[5]. Notice that our construction builds on the (generic) post-ICL validation mechanism, rather than a direct attempt of approximating the exact Bayes predictor, whose structure may vary significantly case-by-case. This highlights post-ICL validation as a promising mechanism for approximating the Bayes predictor on broader classes of problems beyond noisy linear models, which we leave as future work.

Generalized linear models with adaptive link function selection As another example of the post-ICL validation mechanism, we construct a transformer that can learn a generalized linear model with adaptively chosen link function for the particular ICL instance; see Theorem J.2.

4.2 Pre-ICL testing mechanism

In our second mechanism, pre-ICL testing, the transformer runs a *distribution testing* procedure on the input sequence to determine the right ICL algorithm to use. While the test (and thus the mechanism itself) could in principle be general, we focus on cases where the test amounts to computing some simple summary statistics of the input sequence.

To showcase pre-ICL testing, we consider the toy problem of selecting between in-context regression and in-context classification, by running the following *binary type check* on the input labels $fy_i g_{i \in [N]}$.

$$^{\text{binary}}(D) = \frac{1}{N} \sum_{i=1}^{N} \quad (y_i); \qquad (y) := \begin{cases} 1; & y \ 2 \ f0; 1g; \\ 0; & y \ \not{2} \ [\ \ ''; ''] \ [\ [1 \quad ''; 1 + ''] ; \\ \text{linear interpolation}; & \text{otherwise}: \end{cases}$$

[5] By the Bayes risk decomposition for square loss, (8) implies that $E[(\hat{y}_{N+1} \quad \hat{y}^{\text{Bayes}}_{N+1})^2] \quad O((\log K/N)^{1/3})$.

Lemma 13. *There exists a single attention layer with 6 heads that implements* $^{\text{binary}}$ *exactly.*

Using this test, we construct a transformer that performs logistic regression when labels are binary, and linear regression with high probability if the label admits a continuous distribution.

Proposition 14 (Adaptive regression or classification; Informal version of Proposition I.4)**.** *There exists a transformer with* $O(\log(1/\varepsilon))$ *layers such that the following holds: On any* D *such that* $y_i \in \{0,1\}$*, it outputs* $\hat{y}_{N+1}$ *that* ε*-approximates the prediction of in-context logistic regression.*

By contrast, for any distribution P *whose marginal distribution of* y *is not concentrated around* $\{0,1\}$*, with high probability (over* D*),* $\hat{y}_{N+1}$ ε*-approximates the prediction of in-context least squares.*

The proofs can be found in Appendix I.3. We additionally show that transformers can implement more complex tests such as a *linear correlation test*, which can be useful in certain scenarios such as "confident linear regression" (predict only when the signal-to-noise ratio is high); see Appendix I.4.

5 Analysis of pretraining

Building on the expressivity results in Section 3 & 4, we provide the first line of polynomial sample complexity results for *pretraining* transformers to perform ICL (including with in-context algorithm selection). We begin by providing a generic generalization guarantee for pretraining transformers.

Consider the pretraining ERM problem (TF-ERM), which minimizes the pretraining risk $\hat{L}_{\text{icl}}(\theta)$ over n pretraining sequences. Let $L_{\text{icl}}(\theta)$ denote the corresponding population risk.

Theorem 15 (Generalization of transformers; Informal version of Theorem K.1)**.** *The solution* $\hat{\theta}$ *to (TF-ERM) over transformers with* L *layers,* M *heads per layer, and hidden dimension* D' *satisfies*

$$L_{\text{icl}}(\hat{\theta}) \quad \inf L_{\text{icl}}(\theta) + \tilde{O}\left(\sqrt{\frac{L^2(MD^2 + DD')}{n}}\right);$$

Theorem 15 builds on standard uniform concentration analysis via chaining (Proposition B.4). Combining Theorem 15 with the in-context linear regression construction in Theorem 4 gives the following end-to-end result on the excess in-context prediction risk of trained transformers.

Theorem 16 (Pretraining transformers for in-context linear regression; Informal version of Theorem K.2)**.** *Under Assumption A and* $N \quad \tilde{O}(d)$*, the solution* $\hat{\theta}$ *to (TF-ERM) with* $L = O(\log(N/\varepsilon))$ *layers,* $M = 3$ *heads,* $D' = 0$ *(attention-only as in Theorem 4) achieves small excess ICL risk over the best linear predictor* $w_P^* := \mathbb{E}_P[xx^\top]^{-1}\mathbb{E}_P[xy]$ *for each* P*:*

$$L_{\text{icl}}(\hat{\theta}) \quad \mathbb{E}_{P\sim} \; \mathbb{E}_{(x,y)\sim P} \; \frac{1}{2}(y \quad \langle w_P^*, x_i\rangle)^2 \quad \tilde{O}\left(\sqrt{\frac{^2 d^2}{n}} + \frac{d}{N}^2\right);$$

See Appendix K.2 for similar results in several additional settings.

6 Experiments

We test our theory by studying the ICL and in-context algorithm selection capabilities of transformers, using the encoder-based architecture in our theoretical constructions (Definition 3). Due to limited space, additional experimental details can be found in Appendix M.1. Results with a decoder architecture as in [31, 47] (including the setup of Figure 2) can be found in Appendix M.2.

Training data distributions and evaluation We train a 12-layer transformer, with two modes for the training sequence (instance) distribution . In the "base" mode, similar to [31, 2, 86, 47], we sample the training instances from *one* of the following base distributions (tasks), where we first sample $P = P_{w_?}$ by sampling $w_? \quad N(0, I_d/d)$, and then sample $\{(x_i, y_i)\}_{i\in[N+1]} \overset{\text{iid}}{\sim} P_{w_?}$ as $x_i \overset{\text{iid}}{\sim} N(0, I_d)$, and y_i from one of the following models studied in Section 3:

1. Linear model: $y_i = \langle w_?, x_i\rangle$;
2. Noisy linear model: $y_i = \langle w_?, x_i\rangle + \quad z_i$, where $\quad > 0$ is a fixed noise level, and $z_i \quad N(0, 1)$.

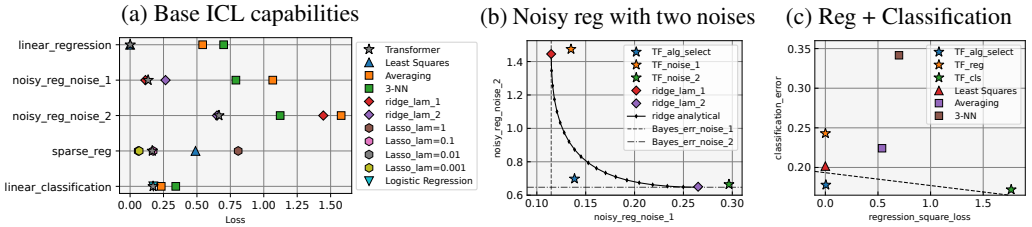

Figure 3: ICL capabilities of the transformer architecture used in our theoretical constructions. *(a)* On five representative base tasks, transformers approximately match the best baseline algorithm for each task, when pretrained on the corresponding task. *(b,c)* A **single transformer** `TF_alg_select` **simultaneously approaches the performance of the strongest baseline algorithm** on two separate tasks: *(b)* noisy linear regression with two different noise levels $2\ f0{:}1;\ 0{:}5g$, and *(c)* adaptively selecting between regression and classification.

3. Sparse linear model: $y_i = \langle \mathsf{w}_?;\ \mathsf{x}_i \rangle$ with $k\mathsf{w}_?k_0 \quad s$, where $\mathsf{s} < d$ is a fixed sparsity level, and in this case we sample $\mathsf{w}_?$ from a special prior supported on s-sparse vectors;
4. Linear classification model: $y_i = \mathsf{sign}(\langle \mathsf{w}_?;\ \mathsf{x}_i \rangle)$.

These base tasks have been empirically investigated by Garg et al. [31], though we remark that our architecture (used in our theory) differs from theirs in several aspects, such as encoder-based architecture instead of decoder-based, and ReLU activation instead of softmax. All experiments use $d = 20$. We choose $2\ f\ _1;\ _2g = f0{:}1;\ 0{:}5g$ and $N = 20$ for noisy linear regression, $s = 3$ and $N = 10$ for sparse linear regression, and $N = 40$ for linear regression and linear classification.

In the "mixture" mode, is the uniform *mixture of two or more base distributions*. We consider two representative mixture modes studied in Section 4:

- Linear model + linear classification model;
- Noisy linear model with four noise levels $2\ f0{:}1;\ 0{:}25;\ 0{:}5;\ 1g$.

Transformers trained with the mixture mode will be evaluated on *multiple* base distributions simultaneously. When the base distributions are sufficiently diverse, a transformer performing well on all of them will *likely* be performing some level of in-context algorithm selection. We evaluate transformers against standard machine learning algorithms in context (for each task respectively) as baselines.

Results Figure 3a shows the ICL performance of transformers on five base tasks, within each the transformer is trained on the same task. Transformers match the best baseline algorithm in four out of the five cases, except for the sparse regression task where the Transformer still outperforms least squares and matches Lasso with some choices of (thus utilizing sparsity to some extent). This demonstrates the strong ICL capability of the transformer architecture considered in our theory.

Figure 3b & 3c examine the in-context algorithm selection capability of transformers, on noisy linear regression with two different noise levels (Figure 3b), and regression + classification (Figure 3c). In both figures, the transformer trained in the mixture mode (`TF_alg_select`) approaches the best baseline algorithm on both tasks simultaneously. By contrast, transformers trained in the base mode for one of the tasks perform well on that task but behave suboptimally on the other task as expected. The existence of `TF_alg_select` showcases a single transformer that performs well on multiple tasks simultaneously (and thus has to perform in-context algorithm selection to some extent), supporting our theoretical results in Section 4.

7 Conclusion

This work shows that transformers can perform complex in-context learning procedures with strong in-context algorithm selection capabilties, by both explicit theoretical constructions and experiments. We believe our work opens up many exciting directions, such as (1) more mechanisms for in-context algorithm selection; (2) Bayes-optimal ICL on other problems by either the post-ICL validation mechanism or new approaches; (3) understanding the internal workings of transformers performing in-context algorithm selection; (4) other mechanisms for implementing complex ICL procedures beyond in-context algorithm selection; (5) further statistical analyses, e.g. of pretraining. Besides, this work focuses on the transformer architecture; alternative sequence-to-sequence architectures (such as RNNs) are beyond our scope but would be interesting directions for future work.

Acknowledgment

The authors would like to thank Tengyu Ma and Jason D. Lee for the many insightful discussions. S. Mei is supported in part by NSF DMS-2210827 and NSF CCF-2315725.

References

[1] A. Agarwal, S. Negahban, and M. J. Wainwright. Fast global convergence rates of gradient methods for high-dimensional statistical recovery. *Advances in Neural Information Processing Systems*, 23, 2010.

[2] E. Akyürek, D. Schuurmans, J. Andreas, T. Ma, and D. Zhou. What learning algorithm is in-context learning? investigations with linear models. *arXiv preprint arXiv:2211.15661*, 2022.

[3] J. L. Ba, J. R. Kiros, and G. E. Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

[4] F. Bach. Breaking the curse of dimensionality with convex neural networks. *The Journal of Machine Learning Research*, 18(1):629–681, 2017.

[5] Y. Bai, M. Chen, P. Zhou, T. Zhao, J. Lee, S. Kakade, H. Wang, and C. Xiong. How important is the train-validation split in meta-learning? In *International Conference on Machine Learning*, pages 543–553. PMLR, 2021.

[6] Y. Bai, S. Mei, H. Wang, and C. Xiong. Don't just blame over-parametrization for over-confidence: Theoretical analysis of calibration in binary classification. In *International Conference on Machine Learning*, pages 566–576. PMLR, 2021.

[7] J. Baxter. A model of inductive bias learning. *Journal of artificial intelligence research*, 12: 149–198, 2000.

[8] A. Beck and M. Teboulle. Gradient-based algorithms with applications to signal recovery. *Convex optimization in signal processing and communications*, pages 42–88, 2009.

[9] S. Bengio, Y. Bengio, J. Cloutier, and J. Gescei. On the optimization of a synaptic learning rule. In *Optimality in Biological and Artificial Networks?*, pages 281–303. Routledge, 2013.

[10] S. Bhattamishra, K. Ahuja, and N. Goyal. On the ability and limitations of transformers to recognize formal languages. *arXiv preprint arXiv:2009.11264*, 2020.

[11] S. Bhattamishra, A. Patel, and N. Goyal. On the computational power of transformers and its implications in sequence modeling. *arXiv preprint arXiv:2006.09286*, 2020.

[12] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[13] S. Bubeck. Convex optimization: Algorithms and complexity. *Foundations and Trends® in Machine Learning*, 8(3-4):231–357, 2015.

[14] S. Bubeck, V. Chandrasekaran, R. Eldan, J. Gehrke, E. Horvitz, E. Kamar, P. Lee, Y. T. Lee, Y. Li, S. Lundberg, et al. Sparks of artificial general intelligence: Early experiments with gpt-4. *arXiv preprint arXiv:2303.12712*, 2023.

[15] S. Chan, A. Santoro, A. Lampinen, J. Wang, A. Singh, P. Richemond, J. McClelland, and F. Hill. Data distributional properties drive emergent in-context learning in transformers. *Advances in Neural Information Processing Systems*, 35:18878–18891, 2022.

[16] L. Chen, K. Lu, A. Rajeswaran, K. Lee, A. Grover, M. Laskin, P. Abbeel, A. Srinivas, and I. Mordatch. Decision transformer: Reinforcement learning via sequence modeling. *Advances in neural information processing systems*, 34:15084–15097, 2021.

[17] K. Chua, Q. Lei, and J. D. Lee. How fine-tuning allows for effective meta-learning. *Advances in Neural Information Processing Systems*, 34:8871–8884, 2021.

[18] D. Dai, Y. Sun, L. Dong, Y. Hao, Z. Sui, and F. Wei. Why can gpt learn in-context? language models secretly perform gradient descent as meta optimizers. *arXiv preprint arXiv:2212.10559*, 2022.

[19] G. Denevi, C. Ciliberto, D. Stamos, and M. Pontil. Incremental learning-to-learn with statistical guarantees. *arXiv preprint arXiv:1803.08089*, 2018.

[20] G. Denevi, C. Ciliberto, D. Stamos, and M. Pontil. Learning to learn around a common mean. *Advances in Neural Information Processing Systems*, 31, 2018.

[21] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[22] E. Dobriban and S. Wager. High-dimensional asymptotics of prediction: Ridge regression and classification. *The Annals of Statistics*, 46(1):247–279, 2018.

[23] L. Dong, S. Xu, and B. Xu. Speech-transformer: a no-recurrence sequence-to-sequence model for speech recognition. In *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 5884–5888. IEEE, 2018.

[24] Q. Dong, L. Li, D. Dai, C. Zheng, Z. Wu, B. Chang, X. Sun, J. Xu, and Z. Sui. A survey for in-context learning. *arXiv preprint arXiv:2301.00234*, 2022.

[25] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[26] S. S. Du, W. Hu, S. M. Kakade, J. D. Lee, and Q. Lei. Few-shot learning via learning the representation, provably. *arXiv preprint arXiv:2002.09434*, 2020.

[27] B. L. Edelman, S. Goel, S. Kakade, and C. Zhang. Inductive biases and variable creation in self-attention mechanisms. In *International Conference on Machine Learning*, pages 5793–5831. PMLR, 2022.

[28] N. Elhage, N. Nanda, C. Olsson, T. Henighan, N. Joseph, B. Mann, A. Askell, Y. Bai, A. Chen, T. Conerly, et al. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, 2021.

[29] C. Finn, P. Abbeel, and S. Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International conference on machine learning*, pages 1126–1135. PMLR, 2017.

[30] C. Finn, A. Rajeswaran, S. Kakade, and S. Levine. Online meta-learning. In *International Conference on Machine Learning*, pages 1920–1930. PMLR, 2019.

[31] S. Garg, D. Tsipras, P. S. Liang, and G. Valiant. What can transformers learn in-context? a case study of simple function classes. *Advances in Neural Information Processing Systems*, 35:30583–30598, 2022.

[32] A. Giannou, S. Rajput, J.-y. Sohn, K. Lee, J. D. Lee, and D. Papailiopoulos. Looped transformers as programmable computers. *arXiv preprint arXiv:2301.13196*, 2023.

[33] M. Hahn. Theoretical limitations of self-attention in neural sequence models. *Transactions of the Association for Computational Linguistics*, 8:156–171, 2020.

[34] S. Hochreiter, A. S. Younger, and P. R. Conwell. Learning to learn using gradient descent. In *Artificial Neural Networks—ICANN 2001: International Conference Vienna, Austria, August 21–25, 2001 Proceedings 11*, pages 87–94. Springer, 2001.

[35] N. Hollmann, S. Müller, K. Eggensperger, and F. Hutter. Tabpfn: A transformer that solves small tabular classification problems in a second. *arXiv preprint arXiv:2207.01848*, 2022.

[36] T. Hospedales, A. Antoniou, P. Micaelli, and A. Storkey. Meta-learning in neural networks: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 44(9):5149–5169, 2021.

[37] J. Hron, Y. Bahri, J. Sohl-Dickstein, and R. Novak. Infinite attention: Nngp and ntk for deep attention networks. In *International Conference on Machine Learning*, pages 4376–4386. PMLR, 2020.

[38] D. Hsu, S. M. Kakade, and T. Zhang. Random design analysis of ridge regression. In *Conference on learning theory*, pages 9–1. JMLR Workshop and Conference Proceedings, 2012.

[39] S. Jelassi, M. E. Sander, and Y. Li. Vision transformers provably learn spatial structure. *arXiv preprint arXiv:2210.09221*, 2022.

[40] K. Ji, J. D. Lee, Y. Liang, and H. V. Poor. Convergence of meta-learning with task-specific adaptation over partial parameters. *Advances in Neural Information Processing Systems*, 33: 11490–11500, 2020.

[41] J. Jumper, R. Evans, A. Pritzel, T. Green, M. Figurnov, O. Ronneberger, K. Tunyasuvunakool, R. Bates, A. vZídek, A. Potapenko, et al. Highly accurate protein structure prediction with alphafold. *Nature*, 596(7873):583–589, 2021.

[42] S. M. Kakade, V. Kanade, O. Shamir, and A. Kalai. Efficient learning of generalized linear and single index models with isotonic regression. *Advances in Neural Information Processing Systems*, 24, 2011.

[43] M. Khodak, M.-F. F. Balcan, and A. S. Talwalkar. Adaptive gradient-based meta-learning methods. *Advances in Neural Information Processing Systems*, 32, 2019.

[44] L. Kirsch and J. Schmidhuber. Meta learning backpropagation and improving it. *Advances in Neural Information Processing Systems*, 34:14122–14134, 2021.

[45] L. Kirsch, J. Harrison, J. Sohl-Dickstein, and L. Metz. General-purpose in-context learning by meta-learning transformers. *arXiv preprint arXiv:2212.04458*, 2022.

[46] K. Li and J. Malik. Learning to optimize. *arXiv preprint arXiv:1606.01885*, 2016.

[47] Y. Li, M. E. Ildiz, D. Papailiopoulos, and S. Oymak. Transformers as algorithms: Generalization and implicit model selection in in-context learning. *arXiv preprint arXiv:2301.07067*, 2023.

[48] B. Liu, J. T. Ash, S. Goel, A. Krishnamurthy, and C. Zhang. Transformers learn shortcuts to automata. *arXiv preprint arXiv:2210.10749*, 2022.

[49] J. Liu, D. Shen, Y. Zhang, B. Dolan, L. Carin, and W. Chen. What makes good in-context examples for gpt-3? *arXiv preprint arXiv:2101.06804*, 2021.

[50] Y. Lu, M. Bartolo, A. Moore, S. Riedel, and P. Stenetorp. Fantastically ordered prompts and where to find them: Overcoming few-shot prompt order sensitivity. *arXiv preprint arXiv:2104.08786*, 2021.

[51] A. Madani, B. McCann, N. Naik, N. S. Keskar, N. Anand, R. R. Eguchi, P.-S. Huang, and R. Socher. Progen: Language modeling for protein generation. *arXiv preprint arXiv:2004.03497*, 2020.

[52] A. Maurer, M. Pontil, and B. Romera-Paredes. The benefit of multitask representation learning. *Journal of Machine Learning Research*, 17(81):1–32, 2016.

[53] P. McCullagh. *Generalized linear models*. Routledge, 2019.

[54] S. Mei, Y. Bai, and A. Montanari. The landscape of empirical risk for nonconvex losses. *The Annals of Statistics*, 46(6A):2747–2774, 2018.

[55] S. Min, M. Lewis, H. Hajishirzi, and L. Zettlemoyer. Noisy channel language model prompting for few-shot text classification. *arXiv preprint arXiv:2108.04106*, 2021.

[56] S. Min, M. Lewis, L. Zettlemoyer, and H. Hajishirzi. Metaicl: Learning to learn in context. *arXiv preprint arXiv:2110.15943*, 2021.

[57] S. Min, X. Lyu, A. Holtzman, M. Artetxe, M. Lewis, H. Hajishirzi, and L. Zettlemoyer. Rethinking the role of demonstrations: What makes in-context learning work? *arXiv preprint arXiv:2202.12837*, 2022.

[58] N. Mishra, M. Rohaninejad, X. Chen, and P. Abbeel. A simple neural attentive meta-learner. *arXiv preprint arXiv:1707.03141*, 2017.

[59] S. Müller, N. Hollmann, S. P. Arango, J. Grabocka, and F. Hutter. Transformers can do bayesian inference. *arXiv preprint arXiv:2112.10510*, 2021.

[60] T. Nagler. Statistical foundations of prior-data fitted networks. *arXiv preprint arXiv:2305.11097*, 2023.

[61] D. K. Naik and R. J. Mammone. Meta-neural networks that learn by learning. In *[Proceedings 1992] IJCNN International Joint Conference on Neural Networks*, volume 1, pages 437–442. IEEE, 1992.

[62] S. N. Negahban, P. Ravikumar, M. J. Wainwright, and B. Yu. A unified framework for high-dimensional analysis of m-estimators with decomposable regularizers. 2012.

[63] Y. Nesterov. *Lectures on convex optimization*, volume 137. Springer, 2018.

[64] C. Olsson, N. Elhage, N. Nanda, N. Joseph, N. DasSarma, T. Henighan, B. Mann, A. Askell, Y. Bai, A. Chen, et al. In-context learning and induction heads. *arXiv preprint arXiv:2209.11895*, 2022.

[65] OpenAI. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

[66] N. Parikh, S. Boyd, et al. Proximal algorithms. *Foundations and trends® in Optimization*, 1 (3):127–239, 2014.

[67] J. Pérez, J. Marinković, and P. Barceló. On the turing completeness of modern neural network architectures. *arXiv preprint arXiv:1901.03429*, 2019.

[68] A. Radford, K. Narasimhan, T. Salimans, I. Sutskever, et al. Improving language understanding by generative pre-training. 2018.

[69] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.

[70] A. Raventos, M. Paul, F. Chen, and S. Ganguli. The effects of pretraining task diversity on in-context learning of ridge regression. In *ICLR 2023 Workshop on Mathematical and Empirical Understanding of Foundation Models*, 2023.

[71] S. Ravi and H. Larochelle. Optimization as a model for few-shot learning. In *International conference on learning representations*, 2017.

[72] Y. Razeghi, R. L. Logan IV, M. Gardner, and S. Singh. Impact of pretraining term frequencies on few-shot reasoning. *arXiv preprint arXiv:2202.07206*, 2022.

[73] S. Reed, K. Zolna, E. Parisotto, S. G. Colmenarejo, A. Novikov, G. Barth-Maron, M. Gimenez, Y. Sulsky, J. Kay, J. T. Springenberg, et al. A generalist agent. *arXiv preprint arXiv:2205.06175*, 2022.

[74] O. Rubin, J. Herzig, and J. Berant. Learning to retrieve prompts for in-context learning. *arXiv preprint arXiv:2112.08633*, 2021.

[75] A. Santoro, S. Bartunov, M. Botvinick, D. Wierstra, and T. Lillicrap. Meta-learning with memory-augmented neural networks. In *International conference on machine learning*, pages 1842–1850. PMLR, 2016.

[76] N. Saunshi, A. Gupta, and W. Hu. A representation learning perspective on the importance of train-validation splitting in meta-learning. In *International Conference on Machine Learning*, pages 9333–9343. PMLR, 2021.

[77] J. Schmidhuber. *Evolutionary principles in self-referential learning, or on learning how to learn: the meta-meta-... hook*. PhD thesis, Technische Universität München, 1987.

[78] K. Shen, J. Guo, X. Tan, S. Tang, R. Wang, and J. Bian. A study on relu and softmax in transformer. *arXiv preprint arXiv:2302.06461*, 2023.

[79] C. Snell, R. Zhong, D. Klein, and J. Steinhardt. Approximating how single head attention learns. *arXiv preprint arXiv:2103.07601*, 2021.

[80] J. Snell, K. Swersky, and R. Zemel. Prototypical networks for few-shot learning. *Advances in neural information processing systems*, 30, 2017.

[81] S. Thrun and L. Pratt. *Learning to learn*. Springer Science & Business Media, 2012.

[82] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288, 1996.

[83] N. Tripuraneni, M. Jordan, and C. Jin. On the theory of transfer learning: The importance of task diversity. *Advances in neural information processing systems*, 33:7852–7862, 2020.

[84] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[85] R. Vershynin. *High-dimensional probability: An introduction with applications in data science*, volume 47. Cambridge university press, 2018.

[86] J. von Oswald, E. Niklasson, E. Randazzo, J. Sacramento, A. Mordvintsev, A. Zhmoginov, and M. Vladymyrov. Transformers learn in-context by gradient descent. *arXiv preprint arXiv:2212.07677*, 2022.

[87] M. J. Wainwright. *High-dimensional statistics: A non-asymptotic viewpoint*, volume 48. Cambridge university press, 2019.

[88] X. Wang, S. Yuan, C. Wu, and R. Ge. Guarantees for tuning the step size using a learning-to-learn approach. In *International Conference on Machine Learning*, pages 10981–10990. PMLR, 2021.

[89] C. Wei, Y. Chen, and T. Ma. Statistically meaningful approximation: a case study on approximating turing machines with transformers. *arXiv preprint arXiv:2107.13163*, 2021.

[90] J. Wei, Y. Tay, R. Bommasani, C. Raffel, B. Zoph, S. Borgeaud, D. Yogatama, M. Bosma, D. Zhou, D. Metzler, et al. Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682*, 2022.

[91] J. Wei, J. Wei, Y. Tay, D. Tran, A. Webson, Y. Lu, X. Chen, H. Liu, D. Huang, D. Zhou, et al. Larger language models do in-context learning differently. *arXiv preprint arXiv:2303.03846*, 2023.

[92] G. Weiss, Y. Goldberg, and E. Yahav. Thinking like transformers. In *International Conference on Machine Learning*, pages 11080–11090. PMLR, 2021.

[93] M. Wortsman, J. Lee, J. Gilmer, and S. Kornblith. Replacing softmax with relu in vision transformers. *arXiv preprint arXiv:2309.08586*, 2023.

[94] S. M. Xie, A. Raghunathan, P. Liang, and T. Ma. An explanation of in-context learning as implicit bayesian inference. *arXiv preprint arXiv:2111.02080*, 2021.

[95] S. Yao, B. Peng, C. Papadimitriou, and K. Narasimhan. Self-attention networks can process bounded hierarchical languages. *arXiv preprint arXiv:2105.11115*, 2021.

[96] C. Ying, T. Cai, S. Luo, S. Zheng, G. Ke, D. He, Y. Shen, and T.-Y. Liu. Do transformers really perform badly for graph representation? *Advances in Neural Information Processing Systems*, 34:28877–28888, 2021.

[97] C. Yun, S. Bhojanapalli, A. S. Rawat, S. J. Reddi, and S. Kumar. Are transformers universal approximators of sequence-to-sequence functions? *arXiv preprint arXiv:1912.10077*, 2019.

[98] Y. Zhang, A. Backurs, S. Bubeck, R. Eldan, S. Gunasekar, and T. Wagner. Unveiling transformers with lego: a synthetic reasoning task. *arXiv preprint arXiv:2206.04301*, 2022.

[99] Y. Zhang, B. Liu, Q. Cai, L. Wang, and Z. Wang. An analysis of attention via the lens of exchangeability and latent variable models. *arXiv preprint arXiv:2212.14852*, 2022.

[100] Z. Zhao, E. Wallace, S. Feng, D. Klein, and S. Singh. Calibrate before use: Improving few-shot performance of language models. In *International Conference on Machine Learning*, pages 12697–12706. PMLR, 2021.

[101] X. Zuo, Z. Chen, H. Yao, Y. Cao, and Q. Gu. Understanding train-validation split in meta-learning with neural networks. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=JVlyfHEEm0k.

A Related work

In-context learning The in-context learning (ICL) capability of large language models (LLMs) has gained significant attention since demonstrated on GPT-3 Brown et al. [12]. A number of subsequent empirical studies have contributed to a better understanding of the capabilities and limitations of ICL in LLM systems, which include but are not limited to [49, 55, 56, 50, 100, 74, 72, 28, 45, 91]. For an overview of ICL, see the survey by Dong et al. [24] which highlights some key findings and advancements in this direction.

A line of recent work investigates why and how LLMs perform ICL [94, 31, 86, 2, 18, 32, 47, 70]. In particular, Xie et al. [94] propose a Bayesian inference framework explaining how ICL works despite formatting differences between training and inference distributions. Garg et al. [31] show empirically that transformers could be trained from scratch to perform ICL of linear models, sparse linear models, two-layer neural networks, and decision trees. Li et al. [47] analyze the generalization error of trained ICL transformers from a stability viewpoint. They also experimentally show that transformers could perform "in-context model selection" (conceptually similar to in-context algorithm selection considered in this work) in specific tasks and presented related theoretical hypotheses. However, they do not provide concrete mechanisms or constructions for in-context model selection. A recent work [99] shows that pretrained transformers can perform Bayesian inference in latent variable models, which may also be interpreted as a mechanism for ICL. Our experimental findings extend these results by unveiling and demonstrating the in-context algorithm selection capabilities of transformers.

Closely related to our theoretical results are [86, 2, 18, 32], which show (among many things) that transformers can perform ICL by simulating gradient descent. However, these results do not provide quantitative error bounds for simulating multi-step gradient descent, and only handle linear regression models or their simple variants. Among these works, Akyürek et al. [2] showed that transformers can implement learning algorithms for linear models based on gradient descent and closed-form ridge regression; it also presented preliminary evidence that learned transformers perform ICL similar to Bayes-optimal ridge regression. Our work builds upon and substantially extends this line of work by (1) providing a more efficient construction for in-context gradient descent; (2) providing an end-to-end theory with additional results for pretraining and statistical power; (3) analyzing a broader spectrum of ICL algorithms, including least squares, ridge regression, Lasso, convex risk minimization for generalized linear models, and gradient descent on two-layer neural networks; and (4) constructing more complex ICL procedures using in-context algorithm selection.

When in-context data are generated from a prior, the Bayes risk is a theoretical lower bound for the risk of any possible ICL algorithm, including transformers. Xie et al. [94], Akyürek et al. [2] observe

that learned transformers behave closely to the Bayes predictor on a variety of tasks such as hidden Markov models [94] and noisy linear regression with a fixed noise level [2, 47]. Using the in-context algorithm selection mechanism (more precisely the post-ICL validation mechanism), we show that transformers can perform nearly-Bayes optimal ICL in noisy linear models with mixed noise levels (a strictly more challenging task than considered in [2, 47]), with both concrete theoretical guarantees (Section 4.1.1) and empirical evidence (Figure 2 & 3b). Complementary to these works, a line of work on "prior-data fitted networks" [59, 60, 35] also empirically demonstrates the Bayesian optimality of transformers in various settings. Our expressivity results support these empirical findings and are applicable beyond the Bayesian setting, e.g. for providing frequentist in-context prediction guarantees for transformers.

Transformers and its theory The transformer architecture, introduced by [84], has revolutionized natural language processing and been adopted in most of the recently developed large language models such as BERT and GPT [68, 21, 12]. Broaderly, transformers have demonstrated remarkable performance in many other fields of artificial intelligence such as computer vision, speech, graph processing, reinforcement learning, and biological applications [23, 25, 51, 69, 96, 16, 41, 73, 65, 14]. Towards a better theoretical understanding, recent work has studied the capabilities [97, 67, 37, 95, 11, 98, 48], limitations [33, 10], and internal workings [28, 79, 92, 27, 64] of transformers.

We remark that the transformer architecture used in our theoretical constructions differs from the standard one by replacing the softmax activation (in the attention layers) with a (normalized) ReLU function. Transformers with ReLU activations is experimentally studied in the recent work of Shen et al. [78], who find that they perform as well as the standard softmax activation in many NLP tasks.

Meta-learning Training models (such as transformers) to perform ICL can be viewed as an approach for the broader problem of learning-to-learn or meta-learning [77, 61, 81]. A number of other approaches has been studied extensively for this problem, including (and not limited to) training a meta-learner on how to update the parameters of a downstream learner [9, 46], learning parameter initializations that quickly adapt to downstream tasks [29, 71], learning latent embeddings that allow for effective similarity search [80]. Most relevant to the ICL setting are approaches that directly take as input examples from a downstream task and a query input and produce the corresponding output [34, 58, 75, 44]. For a comprehensive overview, see the survey [36].

Theoretical aspects of meta-learning have received significant recent interest [7, 52, 26, 83, 19, 30, 43, 40, 88, 20, 5, 76, 17, 101]. In particular, [52, 26, 83] analyzed the benefit of multi-task learning through a representation learning perspective, and [88, 20, 5, 76, 101] studied the statistical properties of learning the parameter initialization for downstream tasks.

Techniques We build on various existing techniques from the statistics and learning theory literature to establish our approximation and generalization guarantees for transformers. For the approximation component, we rely on a technical result of Bach [4] on the approximation power of ReLU networks. We use this result to show that transformers can approximate gradient descent (GD) on a broad range of loss functions, substantially extending the results of [86, 2, 18] who primarily consider the square loss. The recent work of Giannou et al. [32] also approximates GD with general loss functions by transformers, though using a different technique of forcing the softmax activations to act as sigmoids. Our analyses of Lasso and generalized linear models build on [87, 62, 1, 54]. Our generalization bound for transformers (used in our pretraining results) build on a chaining argument [87].

B Technical tools

Additional notation for proofs We say a random variable X is ^{2}-sub-Gaussian (or $\mathsf{SG}(\ )$ interchangeably) if $\mathsf{E}[\exp(X^2 = \ ^2)]$ 2. A random vector $\mathsf{x}\ 2\ \mathsf{R}^d$ is ^{2}-sub-Gaussian if $h\mathsf{v}; \mathsf{x}i$ is ^{2}-sub-Gaussian for all $k\mathsf{v}k_2 = 1$. A random variable X is K-sub-Exponential (or $\mathsf{SE}(K)$ interchangeably) if $\mathsf{E}[\exp(jXj=K)]$ 2.

B.1 Concentration inequalities

Lemma B.1. *Let* $\mathsf{N}(0; \mathsf{I}_d=d)$. *Then we have*

$$\mathsf{P}\ k\ k_2^2 \quad (1 + \ )^2 \qquad e^{-d\ ^2=2}:$$

Lemma B.2 (Theorem 6.1 of [87]). *Let $X = [X_{ij}] \in \mathbb{R}^{n\times d}$ be a Gaussian random matrix with $X_{ij} \sim N(0,1)$. Let $s_{\min}(X)$ and $s_{\min}(X)$ be the minimum and maximum singular value of X, respectively. Then we have*

$$\mathbb{P}\left\{s_{\max}(X) \ge \sqrt{n}\left(1+\sqrt{d/n}+t\right)\right\} \le e^{-nt^2/2};$$
$$\mathbb{P}\left\{s_{\min}(X) \le \sqrt{n}\left(1-\sqrt{d/n}\right)\right\} \le e^{-nt^2/2}.$$

The following lemma is a standard result of covariance concentration, see e.g. [85, Theorem 4.6.1].

Lemma B.3. *Suppose that $x_1, \ldots, x_N$ are independent d-dimensional K-sub-Gaussian random vectors. Then as long as $N \ge C_0 d$, with probability at least $1 - \exp(-N/C_0)$ we have*

$$\left\|\frac{1}{N}\sum_{i=1}^{N} x_i x_i^\top - I\right\|_{\mathrm{op}} \le 8K^2;$$

where C_0 is a universal constant.

Lemma B.4. *For random matrix $X = [x_{ij}] \in \mathbb{R}^{N\times d}$ with $x_{ij} \overset{\text{iid}}{\sim} N(0,1)$ and $\varepsilon = [\varepsilon_i] \in \mathbb{R}^N$ with $\varepsilon_i \overset{\text{iid}}{\sim} N(0, \sigma^2)$, it holds that*

$$\mathbb{P}\left\{\|X^\top \varepsilon\|_\infty \ge \sqrt{8N\sigma^2 \log(2d/\delta)}\right\} \le \delta + \exp(-N/2).$$

Proof. We consider $u_j := [x_{ij}]_i \in \mathbb{R}^N$, then $\|X^\top \varepsilon\|_\infty = \max_{i\in[d]} |\langle u_j, \varepsilon\rangle|$. Notice that the random variables $\langle u_1, \varepsilon\rangle, \ldots, \langle u_d, \varepsilon\rangle$ are independent $N(0, k\varepsilon k_2^2)$, and hence

$$\mathbb{P}\left\{\max_{i\in[d]} |\langle u_j, \varepsilon\rangle| \ge t\varepsilon\right\} \le 2d \exp\left(-\frac{t^2}{2k\varepsilon k_2^2}\right).$$

Further, by Lemma B.1, $\mathbb{P}(k\varepsilon k_2 \ge 2\sqrt{N}) \le \exp(-N/2)$. Taking $t = \sqrt{8N\sigma^2 \log(2d/\delta)}$ completes the proof. □

B.2 Approximation theory

For any signed measure μ over a space W, let $\mathrm{TV}(\mu) := \int_W |d\mu(w)| \in [0, \infty]$ denote its total measure. Recall $\sigma(\cdot) = \mathrm{ReLU}(\cdot)$ is the standard relu activation, and $B_\infty^k(R) = [-R, R]^k$ denotes the standard ℓ_∞ ball in $\mathbb{R}^k$ with radius $R > 0$.

Definition B.1 (Sufficiently smooth k-variable function). *We say a function $g : \mathbb{R}^k \to \mathbb{R}$ is (R, C_ℓ)-smooth, if for $s = s(k-1)/2\rfloor + 2$, g is a C^s function on $B_\infty^k(R)$, and*

$$\sup_{x\in B_1^k(R)} \|\nabla^j g(x)\|_\infty = \sup_{x\in B_1^k(R)} \max_{j_1,\ldots,j_j\in[k]} \left|\frac{\partial^j g(x)}{\partial x_{j_1}\cdots\partial x_{j_j}}\right| \le L_j$$

for all $j \in \{0, 1, \ldots, s\}$, with $\max_{0\le j\le s} L_j R^j \le C_\ell$.

The following result for expressing smooth functions as a random feature model with relu activation is adapted from Bach [4, Proposition 5].

Lemma B.5 (Expressing sufficiently smooth functions by relu random features). *Suppose function $g : R^k \to \mathbb{R}$ is (R, C_ℓ) smooth. Then there exists a signed measure μ over $W = \{w \in \mathbb{R}^{k+1} : \|w\|_1 = 1\}$ such that*

$$g(x) = \int_W \frac{1}{R}(w^\top [x; R])_+ \, d\mu(w), \qquad \forall x \in X$$

and $\mathrm{TV}(\mu) \le C(k) C_\ell$, where $C(k) < \infty$ is a constant that only depends on k.

Lemma B.6 (Uniform finite-neuron approximation). *Let X be a space equipped with a distance function $d_X(\cdot,\cdot) : X \times X \to \mathbb{R}_{\ge 0}$. Suppose function $g : X \to \mathbb{R}$ is given by*

$$g(x) = \int_W \phi(x; w) \, d\mu(w),$$

where $(\cdot\,;\cdot): \mathcal{X} \times \mathcal{W} \to [-B, B]$ *is* L*-Lipschitz (in* $d_{\mathcal{X}}$*) in the first argument, and* ν *is a signed measure over* $\mathcal{W}$ *with finite total measure* $A = \mathrm{TV}(\nu) < 1$*. Then for any* $\varepsilon > 0$*, there exists* $w_1, \ldots, w_K \in \mathcal{W}$ *with* $K = O(A^2 B^2 \log N(\mathcal{X}; d_{\mathcal{X}}; \frac{\varepsilon}{3AL})/\varepsilon^2)$*, such that*

$$\sup_{x \in \mathcal{X}} \left| g(x) - \frac{A}{K} \sum_{i=1}^{K} \sigma_i (x; w_i) \right| \le \varepsilon;$$

where $N(\mathcal{X}; d_{\mathcal{X}}; \frac{\varepsilon}{3AL})$ *denotes the* $(\frac{\varepsilon}{3AL})$*-covering number of* $\mathcal{X}$ *in* $d_{\mathcal{X}}$*.*

Proof. Let $\sigma(w) := \mathrm{sign}(d\nu(w)) \in \{\pm 1\}$ denote the sign of the density $d\nu(w)$. We have

$$g(x) = A \int_{\mathcal{W}} \sigma(w)\, \sigma(x; w)\, \frac{|d\nu(w)|}{A}. \tag{9}$$

Note that $|d\nu(w)|/A$ is the density of a probability distribution over $\mathcal{W}$. Thus for any $x \in \mathcal{X}$, as long as $K \ge O(A^2 B^2 \log(1/\delta)/\varepsilon^2)$, we can sample $w_1, \ldots, w_K \overset{\text{iid}}{\sim} |d\nu(\cdot)|/A$, and obtain by Hoeffding's inequality that with probability at least $1 - \delta$,

$$\left| g(x) - \frac{A}{K} \sum_{i=1}^{K} \sigma(w_i)\, \sigma(x; w_i) \right| \le \varepsilon.$$

Let $N(\frac{\varepsilon}{3AL}) := N(\mathcal{X}; d_{\mathcal{X}}; \frac{\varepsilon}{3AL})$ for shorthand. By union bound, as long as $K \ge O(A^2 B^2 \log(N(\frac{\varepsilon}{3AL})/\delta)/\varepsilon^2)$, we have with probability at least $1 - \delta$ that for every $\hat{x}$ in the covering set corresponding to $N(\frac{\varepsilon}{3AL})$,

$$\left| g(\hat{x}) - \frac{A}{K} \sum_{i=1}^{K} \sigma(w_i)\, \sigma(\hat{x}; w_i) \right| \le \varepsilon/3.$$

Taking $\delta = 1/2$ (for which $K = O(A^2 B^2 \log N(\frac{\varepsilon}{3AL})/\varepsilon^2)$), by the probabilistic method, there exists a deterministic set $\{w_i\}_{i \in [K]} \subseteq \mathcal{W}$ and $f_i := \sigma(w_i)$ $g_{i \in [K]} \in \{\pm 1\}$ such that the above holds.

Next, note that both g (by (9)) and the function $x \mapsto \frac{A}{K} \sum_{i=1}^{K} \sigma(w_i)\, \sigma(x; w_i)$ are (AL)-Lipschitz. Therefore, for any $x \in \mathcal{X}$, taking $\hat{x}$ to be the point in the covereing set with $d_{\mathcal{X}}(x; \hat{x}) \le \frac{\varepsilon}{3AL}$, we have

$$\begin{aligned}
&\left| g(x) - \frac{A}{K} \sum_{i=1}^{K} \sigma(w_i)\, \sigma(x; w_i) \right| \\
&\le |g(x) - g(\hat{x})| + \left| g(\hat{x}) - \frac{A}{K} \sum_{i=1}^{K} \sigma(w_i)\, \sigma(\hat{x}; w_i) \right| + \left| \frac{A}{K} \sum_{i=1}^{K} \sigma(w_i)\, \sigma(\hat{x}; w_i) - \frac{A}{K} \sum_{i=1}^{K} \sigma(w_i)\, \sigma(x; w_i) \right| \\
&\le AL \cdot \frac{\varepsilon}{3AL} + \frac{\varepsilon}{3} + AL \cdot \frac{\varepsilon}{3AL} = \varepsilon.
\end{aligned}$$

This proves the lemma. □

Proposition B.1 (Approximating smooth k-variable functions)**.** *For any* $\varepsilon_{\text{approx}} > 0$*,* $R \ge 1$*,* $C_\ell > 0$*, we have the following: Any* $(R; C_\ell)$*-smooth function (Definition B.1)* $g : \mathbb{R}^k \to \mathbb{R}$ *is* $(\varepsilon_{\text{approx}}; R; M; C)$*-approximable by sum of relus (Definition D.1) with* $M \le C(k) C_\ell^2 \log(1 + C_\ell/\varepsilon_{\text{approx}})/\varepsilon_{\text{approx}}^2$ *and* $C \le C(k) C_\ell$*, where* $C(k) > 0$ *is a constant that depends only on* k*. In other words, there exists*

$$f(z) = \sum_{m=1}^{M} c_m\, (a_m^\top [z; 1]) \quad \text{with} \quad \sum_{m=1}^{M} |c_m| \le C; \qquad \max_{m \in [M]} \|a_m\|_1 \le 1;$$

such that $\sup_{z \in [-R; R]^k} |f(z) - g(z)| \le \varepsilon_{\text{approx}}$*.*

Proof. As function $g : \mathrm{B}_\infty^k(R) \to \mathbb{R}$ is $(R; C_\cdot)$-smooth, we can apply Lemma B.5 to obtain that there exists a signed measure $\int$ over $W := \{w \in \mathbb{R}^{k+1} : \|w\|_1 = 1\}$ such that

$$g(z) = \int_W \frac{1}{R} (w^\top [z; R]) d\tau(w); \qquad \forall z \in [-R; R]^k;$$

and $\tau = \mathrm{TV}(\tau) \le C(k) C_\cdot$ where $C(k) > 0$ denotes a constant depending only on k.

We now apply Lemma B.6 to approximate the above random feature by finitely many neurons. Let $x := [z; R] \in X := [-R; R]^k \times \{R\}$. Then, the function $\phi(x; w) := \frac{1}{R} (w^\top x) = (\frac{1}{R} w^\top [z; R])$ is bounded by $B = 1$ and $(1/R)$-Lipschitz in x (in the standard ℓ_∞-distance). Further, we have $\log N(X; \|\cdot\|_\infty; \frac{\varepsilon_{\text{approx}}}{3A/R}) \le O(k \log(1 + A/\varepsilon_{\text{approx}}))$. We can thus apply Lemma B.6 to obtain that, for

$$M = O\left(kA^2 \log(1 + A/\varepsilon_{\text{approx}})/\varepsilon_{\text{approx}}^2\right) = C(k) C_\cdot^2 \log(1 + C_\cdot/\varepsilon_{\text{approx}})/\varepsilon_{\text{approx}}^2;$$

there exists $\{a_m\}_{m \in [M]} \subseteq \{\pm 1\}$ and $W = \{w_m\}_{m \in [M]} \subseteq W := \{w \in \mathbb{R}^{k+1} : \|w\|_1 = 1\}$ such that

$$\sup_{z \in [-R; R]^2} |g(z) - f_{a,W}(z)| \le \varepsilon_{\text{approx}};$$

where (recalling $z = [s; t]$)

$$f_{a,W}(z) = \frac{A}{M} \sum_{m=1}^{M} a_m \frac{1}{R} w_m^\top [z; R] = \sum_{m=1}^{M} \underbrace{\frac{A}{M} a_m}_{c_m} \Big[\underbrace{\frac{1}{R} w_{m;1:k}; w_{m;k+1}}_{\tilde{a}_m}\Big]^\top [z; 1]).$$

Note that we have $\sum_{m=1}^{M} |c_m| = A \le C(k) C_\cdot$, and $\|\tilde{a}_m\|_1 \le \|w_m\|_1 = 1$. This is the desired result. □

B.3 Optimization

The following convergence result for minimizing a smooth and strongly convex function is standard from the convex optimization literature, see e.g. Bubeck [13, Theorem 3.10].

Proposition B.2 (Gradient descent for smooth and strongly convex functions)**.** *Suppose $L : \mathbb{R}^d \to \mathbb{R}$ is α-strongly convex and β-smooth for some $0 < \alpha \le \beta$. Then, the gradient descent iterates $w_{\text{GD}}^{t+1} := w_{\text{GD}}^t - \eta \nabla L(w_{\text{GD}}^t)$ with learning rate $\eta = 1/\beta$ and initialization $w_{\text{GD}}^0 \in \mathbb{R}^d$ satisfies for any $t \ge 1$,*

$$\|w_{\text{GD}}^t - w^\star\|_2^2 \le \exp(-t/\kappa) \|w_{\text{GD}}^0 - w^\star\|_2^2;$$

$$L(w_{\text{GD}}^t) - L(w^\star) \le \frac{\beta}{2} \exp(-t/\kappa) \|w_{\text{GD}}^0 - w^\star\|_2^2;$$

where $\kappa := \beta/\alpha$ is the condition number of L, and $w^\star := \arg\min_{w \in \mathbb{R}^d} L(w)$ is the minimizer of L.

The following convergence result of proximal gradient descent (PGD) on convex composite minimization problem is also standard, see e.g. [8].

Proposition B.3 (Proximal gradient descent for convex function)**.** *Suppose $L = f + h$, $f : \mathbb{R}^d \to \mathbb{R}$ is convex and β-smooth for some $\beta > 0$, $h : \mathbb{R}^d \to \mathbb{R}$ is a simple convex function. Then, the proximal gradient descent iterates $w_{\text{PGD}}^{t+1} := \mathrm{prox}_{\eta h}(w_{\text{PGD}}^t - \eta \nabla f(w_{\text{PGD}}^t))$ with learning rate $\eta = 1/\beta$ and initialization $w_{\text{GD}}^0 \in \mathbb{R}^d$ satisfies the following for any $t \ge 1$:*

1. *$\{L(w_{\text{PGD}}^t)\}$ is a decreasing sequence.*
2. *For any minimizer $w^\star \in \arg\min_{w \in \mathbb{R}^d} L(w)$,*

$$L(w_{\text{GD}}^{t+1}) - L(w^\star) \le \frac{\beta}{2} \|w_{\text{PGD}}^t - w^\star\|_2^2 - \|w_{\text{PGD}}^{t+1} - w^\star\|_2^2;$$

and hence $\{\|w_{\text{PGD}}^t - w^\star\|_2^2\}$ is also a decreasing sequence.

3. *For $k \ge 1; t \ge 0$, it holds that*

$$L(w_{\text{GD}}^{t+k}) - L(w^\star) \le \frac{\beta}{2k} \|w_{\text{PGD}}^t - w^\star\|_2^2;$$

B.4 Uniform convergence

The following result is shown in [87, Section 5.6].

Theorem B.1. *Suppose that* $\psi : [0; +1) \to [0; +1)$ *is a convex, non-decreasing function that satisfies* $\psi(x+y) \geq \psi(x) + \psi(y)$. *For any random variable* X, *we consider the Orlicz norm induced by* ψ: $\|X\|_\psi := \inf\{K > 0 : \mathbb{E}[\psi(|X|/K)] \leq 1\}$.

Suppose that $\{X_\theta\}_{\theta \in \Theta}$ *is a zero-mean random process indexed by* $\theta \in \Theta$ *such that* $\|X_\theta - X_{\theta'}\|_\psi \leq d(\theta, \theta')$ *for some metric* d *on the space* Θ. *Then it holds that*

$$\mathbb{P}\left[\sup_{\theta, \theta' \in \Theta} |X_\theta - X_{\theta'}| \leq 8(J + t)\right] \geq 1 - \frac{1}{\psi(t/D)} \quad \forall t \geq 0,$$

where D *is the diameter of the metric space* (Θ, d), *and the generalized Dudley entropy integral* J *is given by*

$$J := \int_0^D \psi^{-1}(N(\varepsilon, \Theta, d))\, d\varepsilon,$$

where $N(\varepsilon, \Theta, d)$ *is the* ε*-covering number of* (Θ, d).

As a corollary of Theorem B.1, we have the following result.

Proposition B.4 (Uniform concentration bound by chaining)**.** *Suppose that* $\{X_\theta\}_{\theta \in \Theta}$ *is a zero-mean random process given by*

$$X_\theta := \frac{1}{N} \sum_{i=1}^N f(z_i, \theta) - \mathbb{E}_z[f(z, \theta)],$$

where $z_1, \ldots, z_N$ *are i.i.d samples from a distribution* $\mathbb{P}_z$ *such that the following assumption holds:*

(a) The index set Θ *is equipped with a distance* d *and diameter* D. *Further, assume that for some constant* A, *for any ball* B' *of radius* r *in* Θ, *the covering number admits upper bound* $\log N(\varepsilon, B', d) \leq d \log(2Ar/\varepsilon)$ *for all* $0 < \varepsilon \leq 2r$.

(b) For any fixed $\theta \in \Theta$ *and* z *sampled from* $\mathbb{P}_z$, *the random variable* $f(z, \theta)$ *is a* $\mathrm{SG}(B^0)$*-sub-Gaussian random variable.*

(c) For any $\theta, \theta' \in \Theta$ *and* z *sampled from* $\mathbb{P}_z$, *the random variable* $f(z, \theta) - f(z, \theta')$ *is a* $\mathrm{SG}(B^1 d(\theta, \theta'))$*-sub-Gaussian random variable.*

Then with probability at least $1 - \delta$, *it holds that*

$$\sup_{\theta \in \Theta} |X_\theta| \leq C B^0 \sqrt{\frac{d \log(2A\Theta/\delta) + \log(1/\delta)}{N}},$$

where C *is a universal constant, and we denote* $\Theta = 1 + B^1 D / B^0$.

Furthermore, if we replace the SG *in assumption (b) and (c) by* SE, *then with probability at least* $1 - \delta$, *it holds that*

$$\sup_{\theta \in \Theta} |X_\theta| \leq C B^0 \left(\sqrt{\frac{d \log(2A\Theta/\delta) + \log(1/\delta)}{N}} + \frac{d \log(2A\Theta/\delta) + \log(1/\delta)}{N} \right).$$

Proof. Fix a $D_0 \in (0, D]$ to be specified later. We pick a $(D_0/2)$-covering Θ_0 of Θ so that $\log |\Theta_0| \leq d \log(2AD/D_0)$. Then, by the standard uniform covering of independent sub-Gaussian random variables, we have with probability at least $1 - \delta/2$,

$$\sup_{\theta \in \Theta_0} |X_\theta| \leq C B^0 \sqrt{\frac{d \log(2AD/D_0) + \log(2/\delta)}{N}}.$$

Assume that $_0 = f\ _1; \quad ; \ _n g$. For each $j\ 2\ [n]$, we consider $\ _j$ is the ball centered at $\ _j$ of radius D_0 in $(\ ; \ )$. Then $\ 2\ \ _j$ has diameter D_0 and admits covering number bound $\log N(\ _j; \ )$ $d \log(AD_0=\ )$. Hence, we can apply Theorem B.1 with the process $fX\ \ g_{\ \in\ _j}$, then

$$= \ _2; \qquad kX \quad X\ _0k \quad \frac{B^1}{\sqrt{N}}\ (\ ; \ ');$$

and a simple calculation yields

$$\mathrm{P}\left[\sup_{;\ ^0\in\ _j} jX \quad X\ _0j \quad C'B^1D_0\left(\sqrt{\frac{d\log(2A)}{N}} + t\right)\right] \quad 2\exp(\ Nt^2)\ 8t \quad 0:$$

Therefore, we can let $t \quad \sqrt{\log(2n=\ )=N}$ in the above inequality and taking the union bound over $j\ 2\ [n]$, and hence with probability at least $1 \quad =2$, it holds that for all $j\ 2\ [n]$,

$$\sup_{;\ ^0\in\ _j} jX \quad X\ _0j \quad C'B^1D_0\sqrt{\frac{2d\log(2AD=D_0) + \log(4=\ )}{N}}:$$

Notice that for each $\ 2\ ;$ there exists $j\ 2\ [n]$ such that $\ 2\ \ _j$, and hence

$$jX\ j \quad X\ _j\ + \ X \quad X\ _j\ :$$

Thus, with probability at least $1 \quad$, it holds

$$\sup_{\in} jX\ j \quad \sup_{\in\ _0} jX\ j + \sup_j \sup_{\in\ _j} X \quad X\ _j \quad C''(B_0 + B^1D_0)\sqrt{\frac{d\log(2AD=D_0) + \log(2=\ )}{N}}:$$

Taking $D_0 = D=$ completes the proof of SG case.

We next consider the SE case. The idea is the same as the SG case, but in this case we need to consider the following Orlicz-norm:

$$_N(t) := \exp\left(\frac{Nt^2}{t+1}\right) \quad 1:$$

Then Bernstein's inequality of SE random variables yields

$$kX \quad X\ _0k_{\ _N} \quad C_0B^1\ (\ ; \ ')$$

for some universal constant C_0. Therefore, we can repeat the argument above to deduce that with probability at least $1 \quad$, it holds

$$\sup_{\in} jX\ j \quad C''(B_0 + B^1D_0)\left(\sqrt{\frac{d\log(2AD=D_0) + \log(2=\ )}{N}} + \frac{d\log(2AD=D_0) + \log(2=\ )}{N}\right):$$

Taking $D_0 = D=$ completes the proof. □

B.5 Useful properties of transformers

The following result can be obtained immediately by "joining" the attention heads and MLP layers of two single-layer transformers.

Proposition B.5 (Joining parallel single-layer transformers)**.** *Suppose that* $P_1 : \mathrm{R}^{(D_0+D_1)\times N} \ ! \ \mathrm{R}^{D_1\times N}; P_2 : \mathrm{R}^{(D_0+D_2)\times N} \ ! \ \mathrm{R}^{D_2\times N}$ *are two sequence-to-sequence functions that are implemented by single-layer transformers, i.e. there exists* $_1; \ _2$ *such that*

$$\mathrm{TF}_{\ _1}:\mathrm{H}_1 = \left[\begin{matrix}\mathrm{h}_i^{(0)}\\ \mathrm{h}_i^{(1)}\end{matrix}\right]_{1\le i\le N} \ 2\ \mathrm{R}^{(D_0+D_1)\times N} \ 7! \ \begin{matrix}\mathrm{H}^{(0)}\\ P_1(\mathrm{H}_1)\end{matrix};$$

$$\mathrm{TF}_{\ _2}:\mathrm{H}_2 = \left[\begin{matrix}\mathrm{h}_i^{(0)}\\ \mathrm{h}_i^{(2)}\end{matrix}\right]_{1\le i\le N} \ 2\ \mathrm{R}^{(D_0+D_2)\times N} \ 7! \ \begin{matrix}\mathrm{H}^{(0)}\\ P_2(\mathrm{H}_2)\end{matrix}:$$

Then, there exists $\quad$ *such that for* H' *that takes form* $h'_i = [h_i^{(0)}; h_i^{(1)}; h_i^{(2)}]$*, with* $h_i^{(0)} \in \mathbb{R}^{D_0}; h_i^{(1)} \in \mathbb{R}^{D_1}; h_i^{(2)} \in \mathbb{R}^{D_2}$*, we have*

$$\mathrm{TF}\ \ :H' = \begin{bmatrix} h_i^{(0)} \\ h_i^{(1)} \\ h_i^{(2)} \end{bmatrix}_{1\le i\le N} \in \mathbb{R}^{(D_0+D_1+D_2)\times N} \mapsto \begin{bmatrix} H^{(0)} \\ P_1(H_1) \\ P_2(H_2) \end{bmatrix}:$$

Further, $\quad$ *has at most* $M \quad M_1 + M_2$ *heads,* $D' \quad D'_1 + D'_2$ *hidden dimension in its MLP layer, and norm bound* $\|\ \| \quad \|\ _1\| + \|\ _2\|$*.*

Proposition B.6 (Joining parallel multi-layer transformers)**.** *Suppose that* $P_1 : \mathbb{R}^{(D_0+D_1)\times N} \to \mathbb{R}^{D_1\times N}; P_2 : \mathbb{R}^{(D_0+D_2)\times N} \to \mathbb{R}^{D_2\times N}$ *are two sequence-to-sequence functions that are implemented by multi-layer transformers, i.e. there exists* $_1;\ _2$ *such that*

$$\mathrm{TF}\ _1 :H_1 = \begin{bmatrix} h_i^{(0)} \\ h_i^{(1)} \end{bmatrix}_{1\le i\le N} \in \mathbb{R}^{(D_0+D_1)\times N} \mapsto \begin{bmatrix} H^{(0)} \\ P_1(H_1) \end{bmatrix};$$

$$\mathrm{TF}\ _2 :H_2 = \begin{bmatrix} h_i^{(0)} \\ h_i^{(2)} \end{bmatrix}_{1\le i\le N} \in \mathbb{R}^{(D_0+D_2)\times N} \mapsto \begin{bmatrix} H^{(0)} \\ P_2(H_2) \end{bmatrix}:$$

Then, there exists $\quad$ *such that for* H' *that takes form* $h'_i = [h_i^{(0)}; h_i^{(1)}; h_i^{(2)}]$*, with* $h_i^{(0)} \in \mathbb{R}^{D_0}; h_i^{(1)} \in \mathbb{R}^{D_1}; h_i^{(2)} \in \mathbb{R}^{D_2}$*, we have*

$$\mathrm{TF}\ \ :H' = \begin{bmatrix} h_i^{(0)} \\ h_i^{(1)} \\ h_i^{(2)} \end{bmatrix}_{1\le i\le N} \in \mathbb{R}^{(D_0+D_1+D_2)\times N} \mapsto \begin{bmatrix} H^{(0)} \\ P_1(H_1) \\ P_2(H_2) \end{bmatrix}:$$

Further, $\quad$ *has at most* $L \quad \max\{L_1; L_2\}$ *layers,* $\max_{\ell\in[L]} M^{(\ell)} \quad \max_{\ell\in[L]} M_1^{(\ell)} + M_2^{(\ell)}$ *heads,* $\max_{\ell\in[L]} D^{(\ell)} \quad \max_{\ell\in[L]} D_1^{(\ell)} + D_2^{(\ell)}$ *hidden dimension in its MLP layer (understanding the size of the empty layers as 0), and norm bound* $\|\ \| \quad \|\ _1\| + \|\ _2\|$*.*

Proof. When $L_1 = L_2$ ($_1$ and $_2$ have the same number of layers), the result follows directly by applying Proposition B.5 repeatedly for all L_1 layers and the definition of the norm (2).

If (without loss of generality) $L_1 < L_2$, we can augment $_1$ to L_2 layers by adding $(L_2 \quad L_1)$ layers with zero attention heads, and zero MLP hidden dimension (note that this does not change M_1, D'_1, and $\|\ _1\|$). Due to the residual structure, the transformer maintains the output $P_1(H_1)$ throughout layer $L_1 + 1, \ldots, L_2$, and it reduces to the case $L_1 = L_2$. □

C Extension to decoder-based architecture

Here we briefly discuss how our theoretical results can be adapted to decoder-based architectures (henceforth decoder TFs). Adopting the setting as in Section 2, we consider a sequence of N input vectors $\{h_i\}_{i=1}^N \quad \mathbb{R}^D$, written compactly as an input matrix $H = [h_1, \ldots, h_N] \in \mathbb{R}^{D\times N}$. Recall that $(t) := \mathrm{ReLU}(t) = \max\{t, 0\}$ denotes the standard relu activation.

C.1 Decoder-based transformers

Decoder TFs are the same as encoder TFs, except that the attention layers are replaced by masked attention layers with a specific decoder-based (causal) attention mask.

Definition C.1 (Masked attention layer)**.** *A masked attention layer with* M *heads is denoted as* $\mathrm{MAttn}(\ )$ *with parameters* $= \{(V_m, Q_m, K_m)\}_{m\in[M]} \quad \mathbb{R}^{D\times D}$*. On any input sequence* $H \in \mathbb{R}^{D\times N^0}$ *with* $N' \quad N$*,*

$$\mathrm{MAttn}(H) := H + \sum_{m=1}^{M} (V_m H) \quad (\mathrm{MSK}_{1:N^0,1:N^0}) \quad (Q_m H)^\top (K_m H) \quad \in \mathbb{R}^{D\times N^0}; \tag{10}$$

where $\odot$ *denotes the entry-wise (Hadamard) product of two matrices, and* $\mathsf{MSK} \in \mathbb{R}^{N \times N}$ *is the mask matrix given by*

$$\mathsf{MSK} = \begin{bmatrix} 1 & 1/2 & 1/3 & \cdots & 1/N \\ 0 & 1/2 & 1/3 & \cdots & 1/N \\ 0 & 0 & 1/3 & \cdots & 1/N \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1/N \end{bmatrix}.$$

In vector form, we have

$$\mathsf{h}'_i = [\mathsf{Attn}(\mathsf{H})]_i = \mathsf{h}_i + \sum_{m=1}^{M} \frac{1}{i} \sum_{j=1}^{i} \langle \mathsf{Q}_m \mathsf{h}_i, \mathsf{K}_m \mathsf{h}_j \rangle \mathsf{V}_m \mathsf{h}_j.$$

Notice that standard masked attention definitions use the pre-activation additive masks (with mask value $-\infty$) [84]. The post-activation multiplicative masks we use is equivalent to the pre-activation additive masks, and the modified presentation is for notational convenience. We also use a normalized ReLU activation $t \mapsto \mathrm{ReLU}(t)/i$ in place of the standard softmax activation to be consistent with Definition 1. Note that the normalization $1/i$ is to ensure that the attention weights $\{\langle \mathsf{Q}_m \mathsf{h}_i, \mathsf{K}_m \mathsf{h}_j \rangle / i\}_{j \in [i]}$ is a set of non-negative weights that sum to $O(1)$. The motivation of masked attention layer is to ensure that, when processing a sequence of tokens, the computations at any token do not see any later token.

We next define the decoder-based transformers with $L \geq 1$ transformer layers, each consisting of a masked attention layer (c.f. Definition C.1) followed by an MLP layer (c.f. Definition 2). This definition is similar to the definition of encoder-based transformers (c.f., Definition 3), except that we replace the attention layers by masked attention layers.

Definition C.2 (Decoder-based Transformer)**.** *An* L*-layer decoder-based transformer, denoted as* $\mathsf{DTF}(\theta)$*, is a composition of* L *self-attention layers each followed by an MLP layer:* $\mathsf{H}^{(L)} = \mathsf{DTF}_\theta(\mathsf{H}^{(0)})$*, where* $\mathsf{H}^{(0)} \in \mathbb{R}^{D \times N}$ *is the input sequence, and*

$$\mathsf{H}^{(\ell)} = \mathsf{MLP}^{(\ell)}_{\theta_{\mathtt{mlp}}}\left(\mathsf{MAttn}^{(\ell)}_{\theta_{\mathtt{mattn}}}\left(\mathsf{H}^{(\ell-1)}\right)\right), \quad \ell \in \{1, \ldots, L\}.$$

Above, the parameter $\theta = (\theta^{(1:L)}_{\mathtt{mattn}}, \theta^{(1:L)}_{\mathtt{mlp}})$ *is the parameter consisting of the attention layers* $\theta^{(\ell)}_{\mathtt{mattn}} = \{(\mathsf{V}^{(\ell)}_m, \mathsf{Q}^{(\ell)}_m, \mathsf{K}^{(\ell)}_m)\}_{m \in [M^{(\ell)}]} \in \mathbb{R}^{D \times D}$ *and the MLP layers* $\theta^{(\ell)}_{\mathtt{mlp}} = (\mathsf{W}^{(\ell)}_1, \mathsf{W}^{(\ell)}_2) \in \mathbb{R}^{D^{(\ell)} \times D} \times \mathbb{R}^{D \times D^{(\ell)}}$*. We will frequently consider "*attention-only*" decoder-based transformers with* $\mathsf{W}^{(\ell)}_1, \mathsf{W}^{(\ell)}_2 = 0$*, which we denote as* $\mathsf{DTF}^0(\theta)$ *for shorthand, with* $\theta = \theta^{(1:L)} := \theta^{(1:L)}_{\mathtt{mattn}}$*.*

We also use (2) to define the norm of DTF_θ.

C.2 In-context learning with decoder-based transformers

We consider using decoder-based TFs to perform ICL. We encode (D, x_{N+1}), which follows the generating rule as described in Section 2.2, into an input sequence $\mathsf{H} \in \mathbb{R}^{D \times (2N+1)}$. In our theory, we use the following format, where the first two rows contain (D, x_{N+1}) which alternating between $[\mathsf{x}_i; 0] \in \mathbb{R}^{d+1}$ and $[0_{d \times 1}; y_i] \in \mathbb{R}^{d+1}$ (the same setup as adopted in [31, 2]); The third row contains fixed vectors $\{\mathsf{p}_i\}_{i \in [N+1]}$ with ones, zeros, the example index, and indicator for being the covariate token (similar to a positional encoding vector):

$$\mathsf{H} = \begin{bmatrix} \mathsf{x}_1 & 0 & \cdots & \mathsf{x}_N & 0 & \mathsf{x}_{N+1} \\ 0 & y_1 & \cdots & 0 & y_N & 0 \\ \mathsf{p}_1 & \mathsf{p}_2 & \cdots & \mathsf{p}_{2N-1} & \mathsf{p}_{2N} & \mathsf{p}_{2N+1} \end{bmatrix}; \quad \mathsf{p}_i := \begin{bmatrix} 0_{D-(d+4)} \\ \mathrm{di/2e} \\ 1 \\ \mathrm{mod}(i+1, 2) \end{bmatrix} \in \mathbb{R}^{D-(d+1)}. \tag{11}$$

(11) is different from out input format (3) for encoder-based TFs. The main difference is that (x_i, y_i) are in different tokens in (11), whereas (x_i, y_i) are in the same token in (3). The reason for the former (i.e., different tokens in decoder) is that we want to avoid every $[\mathsf{x}_i; 0]$ token seeing the information of y_i, since we will evaluate the loss at every token. The reason for the latter (i.e., the same token in encoder) is for presentation convenience: since we only evaluate the loss at the last token, it is not necessary to alternate between $[\mathsf{x}_i; 0]$ and $[0; y_i]$ to avoid information leakage.

We then feed H into a decoder TF to obtain the output $\hat{H} = \mathrm{DTF}(H) \in \mathbb{R}^{D\times(2N+1)}$ with the same shape, and *read out* the prediction $\hat{y}_{N+1}$ from the $(d+1, 2N+1)$-th entry of $\hat{H} = [\hat{h}_i]_{i\in[2N+1]}$ (the entry corresponding to the last missing test label): $\hat{y}_{N+1} = \mathrm{read}_y(\hat{H}) := (\hat{h}_{2N+1})_{d+1}$. The goal is to predict $\hat{y}_{N+1}$ that is close to $y_{N+1} \sim P_{y|x_{N+1}}$ measured by proper losses.

The benefit of using the decoder architecture is that, during the pre-training phase, one can construct the training loss function by using all the predictions $\{\hat{y}_j\}_{j\in[N+1]}$, where $\hat{y}_j$ gives the $(d+1, 2j-1)$-th entry of $\hat{H} = [\hat{h}_i]_{i\in[2N+1]}$ for each $j \in [N+1]$ (the entry corresponding to the missing test label of the $2j-1$'th token): $\hat{y}_j = \mathrm{read}_{y,j}(\hat{H}) := (\hat{h}_{2j-1})_{d+1}$. Given a loss function $\ell : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$ associated to a single response, the training loss associated to the whole input sequence can be defined by $\ell(H) = \sum_{j=1}^{N+1} \ell(y_j, \hat{y}_j)$. This potentially enables less training sequences in the pre-training stage, and some generalization bound analysis justifying this benefit was provided in [47].

C.3 Results

We discuss how our theoretical results upon encoder TFs can be converted to those of the decoder TFs. Taking the implementation of (ICGD) (a key mechanism that enables most basic ICL algorithms such as ridge regression; cf. Appendix D.1) as an example, this conversion is enabled by the following facts: (a) the input format (11) of decoders can be converted to the input format (3) of encoders by a 2-layer decoder TF; (b) the encoder TF that implements (ICGD) with input format (3), by a slight parameter modification, can be converted to a decoder TF that implements the (ICGD) algorithm with a converted input format.

Input format conversion Despite the difference between the input format (11) and (3), we show that there exists a 2-layer decoder TF that can convert the input format (11) to format (3). The proof can be found in Appendix C.4.

Proposition C.1 (Input format conversion)**.** *There exists a 2-layer decoder TF* DTF *with* 3 *heads per layer, hidden dimension* 2 *and sufficiently large* 1 *such that upon taking input* H *of format* (11)*, it outputs* $\hat{H} = \mathrm{DTF}(H)$ *with*

$$\hat{H} = \begin{bmatrix} x_1 & x_1 & \cdots & x_N & x_N & x_{N+1} \\ 0 & y_1 & \cdots & 0 & y_N & 0 \\ p_1 & p_2 & \cdots & p_{2N-1} & p_{2N} & p_{2N+1} \end{bmatrix}. \tag{12}$$

In particular, format (12) contains format (3) as a submatrix, by restricting to the $\{1, 2, \cdots, D-1, D\}$ *rows and* $\{2, 4, \cdots, 2N-2, 2N, 2N+1\}$ *columns.*

Generalization TF constructions to decoder architecture The construction in Theorem D.1 can be generalized to using the input format (12) along with a decoder TF, by using the scratch pad within the last token to record the gradient descent iterates. Further, if we slightly change the normalization in MSK from $1/i$ to $1/((i-1) \vee 1)$, then the same construction performs (ICGD) (with training examples $\{1, \cdots, j\}$) at every token $i = 2j+1$ (corresponding to predicting at x_{j+1}). Building on this extension, all our constructions in Section 3 and Section 4.2 can be generalized to decoder TFs.

C.4 Proof of Proposition C.1

For the simplicity of presentation, we write $c_i = \lceil i/2 \rceil$; $t_i = \mathrm{mod}(i+1, 2)$, $u_i = h_i[1:d] \in \mathbb{R}^{d+1}$ be the vector of first d entries of h_i [6], and let $v_i = h_i[d+1]$ be the $(d+1)$-th entry of h_i. With such notations, the input sequence $H = [h_i]_i$ can be compactly written as

$$h_i = [u_i; v_i; 0_{D-d-4}; c_i; 1; t_i].$$

In the following, we construct the desired $\theta = (\theta^{(1)}, \theta^{(2)})$ as follows.

Step 1: construction of $\theta^{(1)} = (\theta^{(1)}_{\mathtt{mattn}}, \theta^{(1)}_{\mathtt{mlp}})$, so that $\mathsf{MLP}_{\theta^{(1)}_{\mathtt{mlp}}} \circ \mathsf{MAttn}_{\theta^{(1)}_{\mathtt{mattn}}}$ maps

$$h_i \xrightarrow{\mathsf{MAttn}_{\theta^{(1)}_{\mathtt{mattn}}}} h'_i = [u_i; v_i; 0_{D-d-6}; t_i(c_i^2 + 0.5); t_i c_i; c_i; 1; t_i]$$

[6] In other words, when $2 \cdot i$, $u_i = x_{(i-1)/2}$; when $2 \nmid i$, $u_i = 0_d$.

$$\overset{\text{MLP}}{\underset{\text{mlp}}{(1)}}: \quad \mathsf{h}_i^{(1)} = [\mathsf{u}_i; v_i; \mathsf{0}_{D-d-6}; t_i c_i^2; t_i c_i; c_i; 1; t_i].$$

For $m \in \{0; 1\}$, we define matrices $\mathsf{Q}_m^{(1)}, \mathsf{K}_m^{(1)}, \mathsf{V}_m^{(1)} \in \mathbb{R}^{D \times D}$ such that

$$\mathsf{Q}_0^{(1)}\mathsf{h}_i = \mathsf{Q}_1^{(1)}\mathsf{h}_i = \begin{bmatrix} t_i \\ 0 \end{bmatrix}; \qquad \mathsf{K}_0^{(1)}\mathsf{h}_j = \mathsf{K}_1^{(1)}\mathsf{h}_j = \begin{bmatrix} c_j \\ 0 \end{bmatrix}; \qquad \mathsf{V}_0^{(1)}\mathsf{h}_j = \begin{bmatrix} \mathsf{0}_{D-4} \\ 3c_j \\ 0_3 \end{bmatrix}; \qquad \mathsf{V}_1^{(1)}\mathsf{h}_j = \begin{bmatrix} \mathsf{0}_{D-3} \\ 2 \\ 0_2 \end{bmatrix};$$

for all i, j. By the structure of h_i, these matrices indeed exist, and further it is straightforward to check that they have norm bounds

$$\max_m \left\| \mathsf{Q}_m^{(1)} \right\|_{\text{op}} \le 1; \quad \max_m \left\| \mathsf{K}_m^{(1)} \right\|_{\text{op}} \le 1; \quad \sum_m \left\| \mathsf{V}_m^{(1)} \right\|_{\text{op}} \le 5.$$

Now, for every i,

$$\frac{1}{i} \sum_{j=1}^{i} \sum_{m \in \{0;1\}} \left\langle \mathsf{Q}_m^{(1)}\mathsf{h}_i, \mathsf{K}_m^{(1)}\mathsf{h}_j \right\rangle \mathsf{V}_m^{(1)}\mathsf{h}_j = \frac{1}{i} \sum_{j=1}^{i} t_i \, [\mathsf{0}_{D-4}; 3c_j^2; 2c_j; 0; 0].$$

Notice that $t_i \neq 0$ only when $2 \mid i$, we then compute for $i = 2k$ that

$$\sum_{j=1}^{i} 3c_j^2 = 3 \, \frac{k(k-1)(2k-1)}{3} + 3k^2 = 2k^3 + k; \qquad \sum_{j=1}^{i} 2c_j = 2 \, k(k-1) + 2k = 2k^2.$$

Therefore, the $\overset{(1)}{\text{mattn}} = \{(\mathsf{Q}_m^{(1)}, \mathsf{K}_m^{(1)}, \mathsf{V}_m^{(1)}) \in \mathbb{R}^{D \times D}\}_{m \in \{0;1\}}$ we construct above is indeed the desired attention layer. The existence of the desired $\overset{(1)}{\text{mlp}}$ is clear, and $\overset{(1)}{\text{mlp}} = (\mathsf{W}_1^{(1)}, \mathsf{W}_2^{(1)})$ can further be chosen so that $\|\mathsf{W}_1^{(1)}\|_{\text{op}} \le 1, \|\mathsf{W}_2^{(1)}\|_{\text{op}} \le 1$.

Step 2: construction of $\overset{(2)}{}$. For every $m \in \{-1; 0; 1\}$, we define matrices $\mathsf{Q}_m^{(2)}, \mathsf{K}_m^{(2)}, \mathsf{V}_m^{(2)} \in \mathbb{R}^{D \times D}$ such that

$$\mathsf{Q}_0^{(2)}\mathsf{h}_i^{(1)} = \mathsf{Q}_1^{(2)}\mathsf{h}_i^{(1)} = \mathsf{Q}_{-1}^{(2)}\mathsf{h}_i^{(1)} = \begin{bmatrix} t_i c_i^2 \\ t_i c_i \\ 0 \end{bmatrix};$$

$$\mathsf{K}_0^{(2)}\mathsf{h}_j^{(1)} = \begin{bmatrix} 1 \\ c_j \\ 0 \end{bmatrix}; \qquad \mathsf{K}_1^{(2)}\mathsf{h}_j^{(1)} = \begin{bmatrix} 1 \\ (c_j + 1) \\ 0 \end{bmatrix}; \qquad \mathsf{K}_1^{(2)}\mathsf{h}_j^{(1)} = \begin{bmatrix} 1 \\ (c_j - 1) \\ 0 \end{bmatrix};$$

$$\mathsf{V}_0^{(2)}\mathsf{h}_j^{(1)} = \begin{bmatrix} 4\mathsf{u}_j \\ \mathsf{0}_{D-d} \end{bmatrix}; \qquad \mathsf{V}_1^{(2)}\mathsf{h}_j^{(1)} = \mathsf{V}_{-1}^{(2)}\mathsf{h}_j^{(1)} = \begin{bmatrix} 2\mathsf{u}_j \\ \mathsf{0}_{D-d} \end{bmatrix};$$

for all i, j. By the structure of $\mathsf{h}_i^{(1)}$, these matrices indeed exist, and further it is straightforward to check that they have norm bounds

$$\max_m \left\| \mathsf{Q}_m^{(2)} \right\|_{\text{op}} \le 1; \quad \max_m \left\| \mathsf{K}_m^{(2)} \right\|_{\text{op}} \le 2; \quad \sum_m \left\| \mathsf{V}_m^{(2)} \right\|_{\text{op}} \le 8.$$

Now, for every i, j, we have

$$\begin{aligned}
&\sum_{m \in \{-1;0;1\}} \left\langle \mathsf{Q}_m^{(2)}\mathsf{h}_i^{(1)}, \mathsf{K}_m^{(2)}\mathsf{h}_j^{(1)} \right\rangle \mathsf{V}_m^{(2)}\mathsf{h}_j^{(1)} \\
&= -2 \, t_i c_i^2 - t_i c_i c_j + t_i c_i^2 - t_i c_i (c_j + 1) + t_i c_i^2 - t_i c_i (c_j - 1) \cdot 2[\mathsf{u}_j; \mathsf{0}_{D-d}] \\
&= \{-2 \, (c_i - c_j) + ((c_i - c_j) - 1) + ((c_i - c_j) + 1)\} \, 2c_i t_i [\mathsf{u}_j; \mathsf{0}_{D-d}] \\
&= \mathsf{I}(c_i = c_j) \cdot 2c_i t_i [\mathsf{u}_j; \mathsf{0}_{D-d}];
\end{aligned}$$

where the last equality follows from the fact that

$$-2 \, (x) + (x - 1) + (x + 1) = \begin{cases} 0, & x \le -1 \text{ or } x \ge 1; \\ x + 1, & x \in [-1, 0]; \\ 1 - x, & x \in [0, 1]. \end{cases}$$

Therefore,

$$\frac{1}{i}\sum_{j=1}^{i}\sum_{m\in\{-1;0;1\}}\left\langle Q_m^{(2)}h_i^{(1)}; K_m^{(2)}h_j^{(1)}\right\rangle V_m^{(2)}h_j^{(1)} = \frac{1}{i}\sum_{j=1}^{i} 2\mathbb{I}(c_i = c_j)c_i t_i[u_j; 0_{D-d}]$$

$$= \begin{cases}[x_k; 0_{D-d}]; & i = 2k\\ 0_D; & \text{otherwise}\end{cases}.$$

Therefore, the $\mathrm{attn}^{(2)} = f(Q_m^{(2)}; K_m^{(2)}; V_m^{(2)} \in \mathbb{R}^{D\times D})g_{m\in\{-1;0;1\}}$ we construct above maps

$$h_i^{(1)} \to h_i'' = [x_{\lceil i=2\rceil}; v_i; 0_{D-d-6}; t_i c_i^2; t_i c_i; c_i; 1; t_i].$$

Finally, we only need to take a MLP layer $\mathrm{mlp}^{(2)} = (W_1^{(2)}; W_2^{(2)})$ with hidden dimension 2 that maps

$$h_i'' \to h_i^{(2)} = [x_{\lceil i=2\rceil}; v_i; 0_{D-d-6}; 0; 0; c_i; 1; t_i];$$

which clearly exists and can be chosen so that $\|W_1^{(2)}\|_{\mathrm{op}} \le 1; \|W_2^{(2)}\|_{\mathrm{op}} \le 1$.

Combining the two steps above, we complete the proof of Proposition C.1. □

D Mechanism: In-context gradient descent

Technically, the constructions in Section 3.1-3.2 rely on a new efficient construction for transformers to implement in-context gradient descent and its variants, which we present as follows. We begin by presenting the result for implementing (vanilla) gradient descent on convex empirical risks.

Compact notation of input We will often use shorthand $y_i' \in \mathbb{R}$ defined as $y_i' = y_i$ for $i \in [N]$ and $y_{N+1}' = 0$ to simplify our notation, with which the input sequence $H \in \mathbb{R}^{D\times(N+1)}$ can be compactly written as $h_i = [x_i; y_i'; p_i] = [x_i; y_i'; 0_{D-d-3}; 1; t_i]$ for $i \in [N+1]$, where $t_i := 1\{i < N+1\}$ is the indicator for the training examples.

D.1 Gradient descent on convex empirical risk

Let $\ell(\cdot;\cdot) : \mathbb{R}^2 \to \mathbb{R}$ be a loss function. Let $L_N(w) := \frac{1}{N}\sum_{i=1}^{N}\ell(w^\top x_i; y_i)$ denote the empirical risk with loss function ℓ on dataset $\{(x_i; y_i)\}_{i\in[N]}$, and

$$w_{\mathrm{GD}}^{t+1} := w_{\mathrm{GD}}^{t} - \eta\nabla L_N(w_{\mathrm{GD}}^{t}) \tag{ICGD}$$

denote the gradient descent trajectory on L_N with initialization $w_{\mathrm{GD}}^0 \in \mathbb{R}^d$ and learning rate $\eta > 0$.

We require the partial derivative of the loss $\partial_s\ell : (s; t) \mapsto \partial_s\ell(s; t)$ (as a bivariate function) to be approximable by a sum of relus, defined as follows.

Definition D.1 (Approximability by sum of relus)**.** *A function $g : \mathbb{R}^k \to \mathbb{R}$ is $(\varepsilon_{\mathrm{approx}}; R; M; C)$-*approximable by sum of relus*, if there exists a "$(M; C)$-sum of relus" function*

$$f_{M;C}(z) = \sum_{m=1}^{M} c_m\,(a_m^\top[z; 1])_+ \quad \text{with} \quad \sum_{m=1}^{M}|c_m| \le C;\ \max_{m\in[M]}\|a_m\|_1 \le 1;\ a_m \in \mathbb{R}^{k+1}; c_m \in \mathbb{R};$$

such that $\sup_{z\in[-R;R]^k}|g(z) - f_{M;C}(z)| \le \varepsilon_{\mathrm{approx}}$.

Definition D.1 is known to contain broad class of functions. For example, any mildly smooth k-variate function is approximable by a sum of relus for any $(\varepsilon_{\mathrm{approx}}; R)$, with mild bounds on $(M; C)$ (Proposition B.1, building on results of Bach [4]). Also, any function that is a $(M; C)$-sum of relus itself (which includes all piecewise linear functions) is by definition $(0; 1; M; C)$-approximable by sum of relus.

We show that L steps of (ICGD) can be approximately implemented by an $(L+1)$-layer transformer.

Theorem D.1 (Convex ICGD)**.** *Fix any $B_w > 0$, $L > 1$, $\eta > 0$, and $\varepsilon < B_w\eta/(2L)$. Suppose that*

1. *The loss $\ell(\cdot;\cdot)$ is convex in the first argument;*
2. *$\partial_s\ell$ is $(\varepsilon; R; M; C)$-approximable by sum of relus with $R = \max\{B_x B_w; B_y; 1\}$.*

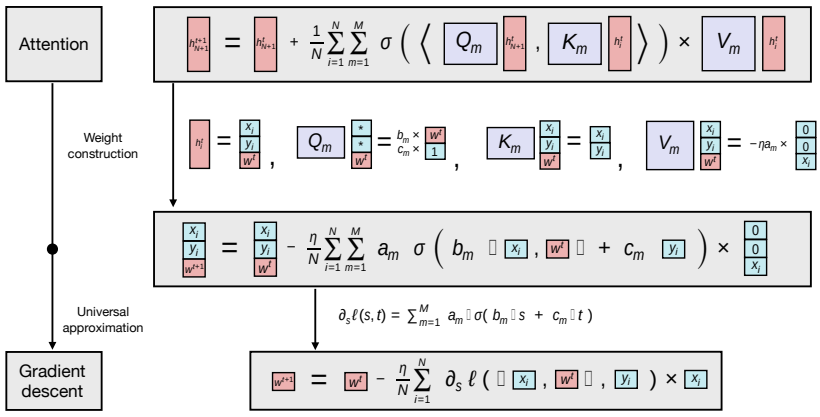

Figure 4: Illustration of our main mechanism for implementing basic ICL algorithms: One attention layer implements a single (ICGD) iterate (Proposition E.1 & Theorem D.1). Top: the attention mechanism as in Definition 1. Bottom: A single (ICGD) iterate. Middle: Linear algebraic illustration of the attention layer for implementing a GD update.

Then, there exists an attention-only transformer TF^0 *with* $(L+1)$ *layers,* $\max_{\ell\in[L]} M^{(\ell)} \le M$ *heads within the first* L *layers, and* $M^{(L+1)} = 2$ *such that for* any input data $(D; \mathsf{x}_{N+1})$ *such that*

$$\sup_{\|\mathbf{w}\|_2\le B_w} \max\big(r^2 \mathcal{L}_N(\mathbf{w})\big) \le 2 = ; \quad \forall \mathbf{w}^\star \in \arg\min_{\mathbf{w}\in\mathbb{R}^d} \mathcal{L}_N(\mathbf{w}) \text{ such that } \|\mathbf{w}^\star\|_2 \le B_w = 2;$$

$\mathsf{TF}^0(\mathsf{H}^{(0)})$ *approximately implements (ICGD) with initialization* $\mathbf{w}_{\mathrm{GD}}^0 = 0$*:*

1. *(Parameter space) For every* $\ell \in [L]$*, the* ℓ*-th layer's output* $\mathsf{H}^{(\ell)} = \mathsf{TF}^0_{(1:\ell)}(\mathsf{H}^{(0)})$ *approximates* ℓ *steps of (ICGD): We have* $\mathsf{h}_i^{(\ell)} = [\mathsf{x}_i; y_i'; \mathbf{w}^\ell; 0_{D-2d-3}; 1; t_i]$ *for every* $i \in [N+1]$*, where*

$$\|\mathbf{w}^\ell - \mathbf{w}_{\mathrm{GD}}^\ell\|_2 \le \varepsilon'(\ell, B_x).$$

   Note that the bound scales as $O(\ell)$*, a* linear *error accumulation.*

2. *(Prediction space) The final output* $\mathsf{H}^{(L+1)} = \mathsf{TF}^0(\mathsf{H}^{(0)})$ *approximates the prediction of* L *steps of (ICGD): We have* $\mathsf{h}_{N+1}^{(L+1)} = [\mathsf{x}_{N+1}; \hat{y}_{N+1}; \mathbf{w}^L; 0_{D-2d-3}; 1; t_i]$*, where* $\hat{y}_{N+1} = \langle \mathbf{w}^L, \mathsf{x}_{N+1}\rangle$ *so that*

$$|\hat{y}_{N+1} - \langle \mathbf{w}_{\mathrm{GD}}^L, \mathsf{x}_{N+1}\rangle| \le \varepsilon'(L, B_x^2).$$

Further, the transformer admits norm bound $\|\mathsf{TF}\| \le 2 + R + 2C$*.*

The proof can be found in Appendix E.2. Theorem D.1 substantially generalizes that of von Oswald et al. [86] (which only does GD on square losses with a *linear* self-attention), and is simpler than the ones in Akyürek et al. [2] and Giannou et al. [32]. See Figure 4 for a pictorial illustration of the basic component of the construction, which implements a single step of gradient descent using a single attention layer (Proposition E.1).

Technically, we utilize the stability of *convex* gradient descent as in the following lemma (proof in Appendix E.3) to obtain the *linear* error accumulation in Theorem D.1; the error accumulation will become *exponential* in L in the non-convex case in general; see Lemma D.3(b).

Lemma D.1 (Composition of error for approximating convex GD)**.** *Suppose* $f : \mathbb{R}^d \to \mathbb{R}$ *is a convex function. Let* $\mathbf{w}^\star \in \arg\min_{\mathbf{w}\in\mathbb{R}^d} f(\mathbf{w})$*,* $R \ge 2\|\mathbf{w}^\star\|_2$*, and assume that* $r f$ *is* L_f*-smooth on* $\mathbb{B}_2^d(R)$*. Let sequences* $\{\mathbf{w}^\ell\}_{\ell\ge 0} \subset \mathbb{R}^d$ *and* $\{\mathbf{w}_{\mathrm{GD}}^\ell\}_{\ell\ge 0} \subset \mathbb{R}^d$ *be given by* $\mathbf{w}^0 = \mathbf{w}_{\mathrm{GD}}^0 = 0$*,*

$$\mathbf{w}^{\ell+1} = \mathbf{w}^\ell - r f(\mathbf{w}^\ell) + \varepsilon'; \quad \|\varepsilon'\|_2 \le \varepsilon;$$
$$\mathbf{w}_{\mathrm{GD}}^{\ell+1} = \mathbf{w}_{\mathrm{GD}}^\ell - r f(\mathbf{w}_{\mathrm{GD}}^\ell);$$

for all $\ell \ge 0$*. Then as long as* $\varepsilon \le L_f$*, for any* $0 \le L \le R/(2\varepsilon)$*, it holds that* $\|\mathbf{w}^L - \mathbf{w}_{\mathrm{GD}}^L\|_2 \le L\varepsilon$ *and* $\|\mathbf{w}^L\|_2 \le \frac{R}{2} + L\varepsilon \le R$*.*

D.2 Proximal gradient descent for regularized convex losses

Proximal gradient descent (PGD) is a variant of gradient descent that is suitable for minimizing regularized risks [66], in particular those with a non-smooth regularizer such as the ℓ_1 norm. In this section, we show that transformers can approximate PGD with similar quantitative guarantees as for GD in Appendix D.1.

Let $\ell(\cdot;\cdot) : \mathbb{R}^2 \to \mathbb{R}$ be a loss function. Let $\hat{L}_N(w) := \frac{1}{N}\sum_{i=1}^N \ell(w^\top x_i; y_i) + R(w)$ denote the regularized empirical risk with loss function ℓ on dataset $\{(x_i; y_i)\}_{i\in[N]}$ and regularizer R. To minimize $\hat{L}_N$, we consider the proximal gradient descent trajectory on $\hat{L}_N$ with initialization $w_{\mathrm{GD}}^0 = 0 \in \mathbb{R}^d$ and learning rate $\eta > 0$:

$$w_{\mathrm{PGD}}^{t+1} := \mathrm{prox}_{\eta\mathcal{R}}\left(w_{\mathrm{PGD}}^t - \eta \nabla \hat{L}_N^0(w_{\mathrm{PGD}}^t)\right); \tag{ICPGD}$$

where we denote $\hat{L}_N^0(w) := \frac{1}{N}\sum_{i=1}^N \ell(w^\top x_i; y_i)$.

To approximate (ICPGD) by transformers, in addition to the requirement on the loss ℓ as in Theorem D.1, we additionally require the the proximal operator $\mathrm{prox}_{\eta\mathcal{R}}(\cdot)$ to be approximable by an MLP layer (as a vector-valued analog of Definition D.1) defined as follows.

Definition D.2 (Approximability by MLP)**.** *An operator $P : \mathbb{R}^d \to \mathbb{R}^d$ is $(\varepsilon; R; D; C)$-approximable by MLP, if there exists a there exists a MLP $\theta_{\mathrm{mlp}} = (W_1; W_2) \in \mathbb{R}^{D\times d} \times \mathbb{R}^{d\times D}$ with hidden dimension D, $\|W_1\|_{\mathrm{op}} + \|W_2\|_{\mathrm{op}} \le C'$, such that $\sup_{\|w\|_2\le R} \|P(w) - \mathrm{MLP}_{\theta_{\mathrm{mlp}}}(w)\|_2 \le \varepsilon$.*

The definition above captures the proximal operator $\mathrm{prox}_{\mathcal{R}}$ for a broad class of regularizers, such as the (commonly-used) L_1 and L_2 regularizer listed in the following proposition, for all of which one can directly check that they can be exactly implemented by an MLP as stated below.

Proposition D.1 (Proximal operators for commonly-used regularizers)**.** *For regularizer R in $\{\lambda\|\cdot\|_1; \frac{\lambda}{2}\|\cdot\|_2^2; \mathbb{I}_{B_1(B)}(\cdot)\}$, the operator $\mathrm{prox}_{\mathcal{R}} : \mathbb{R}^d \to \mathbb{R}^d$ is* exactly *approximable by MLP. More concretely, we have*

1. *For $R = \lambda\|\cdot\|_1$, $\mathrm{prox}_{\mathcal{R}}$ is $(0; +1; 4d; 4 + 2\lambda)$-approximable by MLP.*
2. *For $R = \frac{\lambda}{2}\|\cdot\|_2^2$, $\mathrm{prox}_{\mathcal{R}}$ is $(0; +1; 2d; 2 + 2\lambda)$-approximable by MLP.*
3. *For $R = \mathbb{I}_{B_1(B)}(\cdot)$, $\mathrm{prox}_{\mathcal{R}} = \mathrm{Proj}_{B_1(B)}$ is $(0; +1; 2d; 2 + 2B)$-approximable by MLP.*

Theorem D.2 (Convex ICPGD)**.** *Fix any $B_w > 0$, $L > 1$, $\eta > 0$, and $\varepsilon + \varepsilon' \le B_w/(2L)$. Suppose that*

1. *The loss $\ell(\cdot;\cdot)$ is convex in the first argument;*
2. *$\partial_s\ell$ is $(\varepsilon; R; M; C)$-approximable by sum of relus with $R = \max\{B_x B_w; B_y; 1\}$.*
3. *R convex, and the proximal operator $\mathrm{prox}_{\eta\mathcal{R}}(w)$ is $(\varepsilon'; R'; D'; C')$-approximable by MLP with $R' = \sup_{\|w\|_2\le B_w}\|w^+\|_2 + \varepsilon$.*

Then there exists a transformer TF with $(L+1)$ layers, $\max_{\ell\in[L]} M^{(\ell)}$ M heads within the first L layers, $M^{(L+1)} = 2$, and hidden dimension D' such that, for any input data $(D; X_{N+1})$ *such that*

$$\sup_{\|w\|_2\le B_w} \max_i \|\nabla \hat{L}_N(w)\|_2 = 0; \qquad \exists w^\star \in \arg\min_{w\in\mathbb{R}^d} \hat{L}_N(w) \text{ such that } \|w^\star\|_2 \le B_w/2;$$

$\mathsf{TF}(H^{(0)})$ *approximately implements (ICGD):*

1. *(Parameter space) For every $\ell \in [L]$, the ℓ-th layer's output $H^{(\ell)} = \mathsf{TF}_{(1:\ell)}(H^{(0)})$ approximates ℓ steps of (ICGD): We have $h_i^{(\ell)} = [x_i; y_i'; w^\ell; 0_{D-2d-3}; 1; t_i]$ for every $i \in [N+1]$, where*
$$\|w^\ell - w_{\mathrm{PGD}}^\ell\|_2 \le \ell(\varepsilon + \varepsilon')(L B_x).$$
2. *(Prediction space) The final output $H^{(L+1)} = \mathsf{TF}(H^{(0)})$ approximates the prediction of L steps of (ICGD): We have $h_{N+1}^{(L+1)} = [x_{N+1}; \hat{y}_{N+1}; w^L; 0_{D-2d-3}; 1; t_i]$, where $\hat{y}_{N+1} = \langle w^L; x_{N+1}\rangle$ so that*
$$\hat{y}_{N+1} \approx \langle w_{\mathrm{PGD}}^L; x_{N+1}\rangle \pm (\varepsilon + \varepsilon')(2L B_x^2).$$

Further, the weight matrices have norm bounds $\|W\| \le 3 + R + 2C + C'$.

The proof of Theorem D.2 is essentially similar to the proof of Theorem D.1, using the following generalized version of Lemma D.1.

Lemma D.2 (Composition of error for approximating convex PGD)**.** *Suppose* $f : \mathbb{R}^d \to \mathbb{R}$ *is a convex function and* R *is a convex regularizer. Let* $w^\star \in \arg\min_{w \in \mathbb{R}^d} f(w) + R(w)$, $R \ge 2\|w^\star\|_2$, *and assume that* f *is* L_f*-smooth on* $\mathsf{B}_2^d(R)$. *Let sequences* $\{v_t\}_{t \ge 0} \subset \mathbb{R}^d$ *and* $\{w_{\mathrm{GD}}^t\}_{t \ge 0} \subset \mathbb{R}^d$ *be given by* $v^0 = w_{\mathrm{GD}}^0 = 0$,

$$\begin{cases} v^{t+1} = \mathrm{prox}_{\mathcal{R}}\left[v^t - \eta \nabla f(v^t) + \varepsilon^t\right], & \|\varepsilon^t\|_2 \le \varepsilon, \\ w_{\mathrm{GD}}^{t+1} = \mathrm{prox}_{\mathcal{R}}\left[w_{\mathrm{GD}}^t - \eta \nabla f(w_{\mathrm{GD}}^t)\right]. \end{cases}$$

for all $t \ge 0$. *Then as long as* $\eta \le 2/L_f$, *for any* $0 \le L \le R/(2\varepsilon)$, *it holds that* $\|v^L - w_{\mathrm{GD}}^L\|_2 \le L\varepsilon$ *and* $\|v^L\|_2 \le \frac{R}{2} + L\varepsilon \le R$.

The proof of the above lemma is done by utilizing the non-expansiveness of the PGD operator $w \mapsto \mathrm{prox}_{\mathcal{R}}(w - \eta \nabla f(w))$ and otherwise following the same arguments as for Lemma D.1.

D.3 Gradient descent on two-layer neural networks

We now move beyond the convex setting by showing that transformers can implement gradient descent on two-layer neural networks in context.

Suppose that the prediction function $\mathsf{pred}(x; w) := \sum_{k=1}^K u_k r(v_k^\top x)$ is given by a two-layer neural network, parameterized by $w = [v_k, u_k]_{k \in [K]} \in \mathbb{R}^{K(d+1)}$. Consider the empirical risk minimization problem:

$$\min_{w \in \mathcal{W}} \widehat{L}_N(w) := \frac{1}{2N} \sum_{i=1}^N \ell(\mathsf{pred}(x_i; w); y_i) = \frac{1}{2N} \sum_{i=1}^N \ell\left(\sum_{k=1}^K u_k r(v_k^\top x_i); y_i\right); \tag{13}$$

where W is a bounded domain. For the sake of simplicity, in the following discussion we assume that $\mathsf{Proj}_{\mathcal{W}}$ can be *exactly* implemented by a MLP layer (e.g. $W = \mathsf{B}_\infty(R_w)$ for some $R_w > 0$).

Theorem D.3 (Approximate ICGD on two-layer NNs)**.** *Fix any* $B_v, B_u > 0$, $L \ge 1$, $\eta > 0$, *and* $\varepsilon > 0$. *Suppose that*

1. *Both the activation function* r *and the loss function* ℓ *is* C^4*-smooth;*
2. W *is a closed domain such that* $W \supset \{w = [v_k, u_k]_{k \in [K]} \in \mathbb{R}^{K(d+1)} : \|v_k\|_2 \le B_v, |u_k| \le B_u\}$, *and* $\mathsf{Proj}_{\mathcal{W}} = \mathsf{MLP}_{\mathtt{mlp}}$ *for some MLP layer* $\mathtt{mlp}$ *with hidden dimension* D_w *and* C_w *activations;*

Then there exists a $(2L)$*-layer transformer* TF *with*

$$\max_{\ell \in [2L]} M^{(\ell)} \le \mathcal{O}(\varepsilon^{-2}), \quad \max_{\ell \in [2L]} D^{(\ell)} \le \mathcal{O}(\varepsilon^{-2}) + D_w, \quad \|W\| \le O(1 + \eta) + C_w;$$

where $O(\cdot)$ *hides the constants that depend on* K, *the radius parameters* B_x, B_y, B_u, B_v *and the smoothness of* r *and* ℓ, *such that for* any input data (D, x_{N+1}) *such that input sequence* $H^{(0)} \in \mathbb{R}^{D \times (N+1)}$ *takes form* (3), $\mathsf{TF}(H^{(0)})$ *approximately implements in-context gradient descent on risk (13): For every* $t \in [L]$, *the* $2t$*-th layer's output* $h_i^{(2t)} = [x_i; y_i'; v^t; 0; 1; t_i]$ *for every* $i \in [N+1]$, *and*

$$v^t = \mathrm{Proj}_{\mathcal{W}}\left[v^{t-1} - \eta \nabla \widehat{L}_N(v^{t-1}) + \varepsilon^{t-1}\right), \quad v^0 = 0; \tag{14}$$

where $\|\varepsilon^{t-1}\|_2 \le \varepsilon$ *is an error term.*

As a direct corollary, the transformer constructed above can approximate the true gradient descent trajectory $\{w_{\mathrm{GD}}^t\}_{t \ge 0}$ on (16), defined as $w_{\mathrm{GD}}^0 = 0$ and $w_{\mathrm{GD}}^{t+1} = w_{\mathrm{GD}}^t - \eta \nabla \widehat{L}_N(w_{\mathrm{GD}}^t)$ for all $t \ge 0$.

Corollary D.1 (Approximating multi-step ICGD on two-layer NNs)**.** *For any $L \geq 1$, under the same setting as Theorem D.3, the $(2L)$-layer transformer TF there approximates the true gradient descent trajectory $\{\mathbf{w}_{\mathrm{GD}}^{(\ell)}\}_{\ell \geq 0}$: For the intermediate iterates $\{\mathbf{w}^{(\ell)}\}_{\ell \in [L]}$ considered therein, we have*

$$\|\mathbf{w}^{(\ell)} - \mathbf{w}_{\mathrm{GD}}^{(\ell)}\|_2 \leq L_f^{-1}(1 + L_f)^{\ell}\varepsilon;$$

where $L_f = \sup_{\mathbf{w} \in \mathcal{W}} \|\nabla^2 \widehat{L}_N(\mathbf{w})\|_{\mathrm{op}}$ denotes the smoothness of $\widehat{L}_N$ within $\mathcal{W}$.

Remark on error accumulation Note that in Corollary D.1, the error accumulates *exponentially* in ℓ rather than linearly as in Theorem D.1. This is as expected, since gradient descent on non-convex objectives is inherently unstable at a high level (a slight error added upon each step may result in a drastically different trajectories); technically, this happens as the stability-like property Lemma D.1 no longer holds for the non-convex case.

Corollary D.1 is a simple implication of Theorem D.3 and Part (b) of the following convergence and trajectory closeness result for inexact gradient descent. For any closed convex set $\mathcal{W} \subseteq \mathbb{R}^d$, any function $f : \mathcal{W} \to \mathbb{R}$, and any initial point $\mathbf{w} \in \mathcal{W}$, let

$$G_{\mathcal{W},\eta}^{f}(\mathbf{w}) := \frac{\mathbf{w} - \mathrm{Proj}_{\mathcal{W}}(\mathbf{w} - \eta \nabla f(\mathbf{w}))}{\eta}$$

denote the gradient mapping at $\mathbf{w}$ with step size η, a standard measure of stationarity in constrained optimization [63]. Note that $G_{\mathcal{W},\eta}^{f}(\mathbf{w}) = \nabla f(\mathbf{w})$ when $\mathbf{w} - \eta \nabla f(\mathbf{w}) \in \mathcal{W}$ (so that the projection does not take effect).

Lemma D.3 (Convergence and trajectory closeness of inexact GD)**.** *Suppose $f : \mathcal{W} \to \mathbb{R}$, where $\mathcal{W} \subseteq \mathbb{R}^d$ is a convex closed domain and ∇f is L_f-Lipschitz on $\mathcal{W}$. Let sequence $\{\mathbf{w}^{(\ell)}\}_{\ell \geq 0} \subseteq \mathbb{R}^d$ be given by $\mathbf{w}^{(0)} = \mathbf{w}^0$,*

$$\mathbf{w}^{(\ell+1)} = \mathrm{Proj}_{\mathcal{W}}\left(\mathbf{w}^{(\ell)} - \eta\left(\nabla f(\mathbf{w}^{(\ell)}) + \varepsilon^{(\ell)}\right)\right), \qquad \|\varepsilon^{(\ell)}\|_2 \leq \varepsilon;$$

for all $\ell \geq 0$. Then the following holds.

(a) As long as $\eta \leq 1/L_f$, for all $L \geq 1$,

$$\min_{\ell \in [L-1]} \left\|G_{\mathcal{W},\eta}^{f}(\mathbf{w}^{(\ell)})\right\|_2^2 \leq \frac{1}{L}\sum_{\ell=0}^{L-1} \left\|G_{\mathcal{W},\eta}^{f}(\mathbf{w}^{(\ell)})\right\|_2^2 \leq \frac{8(f(\mathbf{w}^0) - \inf_{\mathbf{w} \in \mathcal{W}} f(\mathbf{w}))}{\eta L} + 10\varepsilon^2.$$

(b) Let the sequences $\{\mathbf{w}_{\mathrm{GD}}^{(\ell)}\}_{\ell \geq 0} \subseteq \mathbb{R}^d$ and be given by $\mathbf{w}_{\mathrm{GD}}^{0} = \mathbf{w}^0$ and $\mathbf{w}_{\mathrm{GD}}^{(\ell+1)} = \mathrm{Proj}_{\mathcal{W}}(\mathbf{w}_{\mathrm{GD}}^{(\ell)} - \eta \nabla f(\mathbf{w}_{\mathrm{GD}}^{(\ell)}))$. Then it holds that

$$\|\mathbf{w}^{(\ell)} - \mathbf{w}_{\mathrm{GD}}^{(\ell)}\|_2 \leq L_f^{-1}(1 + L_f)^{\ell}\varepsilon; \qquad \forall \ell \geq 0.$$

E Proofs for Section D

E.1 Approximating a single GD step

Proposition E.1 (Approximating a single GD step by a single attention layer)**.** *Let $\ell(\cdot;\cdot) : \mathbb{R}^2 \to \mathbb{R}$ be a loss function such that $\partial_1 \ell$ is $(\varepsilon; R; M; C)$-approximable by sum of relus with $R = \max\{B_x B_w; B_y; 1\}$. Let $\widehat{L}_N(\mathbf{w}) := \frac{1}{N}\sum_{i=1}^{N} \ell(\mathbf{w}^\top \mathbf{x}_i; y_i)$ denote the empirical risk with loss function ℓ on dataset $\{(\mathbf{x}_i; y_i)\}_{i \in [N]}$.*

Then, for any $\varepsilon > 0$, there exists an attention layer $\mathsf{F} = \{(\mathbf{Q}_m; \mathbf{K}_m; \mathbf{V}_m)\}_{m \in [M]}$ with M heads such that, for any input sequence that takes form $\mathbf{h}_i = [\mathbf{x}_i; y_i'; \mathbf{w}; \mathbf{0}_{D-2d-3}; 1; t_i]$ with $\|\mathbf{w}\|_2 \leq B_w$, it gives output $\mathbf{h}_i = [\mathrm{Attn}(\mathbf{H})]_i = [\mathbf{x}_i; y_i'; \mathbf{w}^+; \mathbf{0}_{D-2d-3}; 1; t_i]$ for all $i \in [N+1]$, where

$$\|\mathbf{w}^+ - (\mathbf{w} - \eta \nabla \widehat{L}_N(\mathbf{w}))\|_2 \leq \varepsilon(\eta + B_x).$$

Further, $\|\mathbf{w}^+\| \leq 2 + R + 2\eta C$.

Proof of Proposition E.1. As $\ell_s(\cdot)$ is (ε, R, M, C)-approximable by sum of relus, there exists a function $f : [-R, R]^2 \to \mathbb{R}$ of form

$$f(s,t) = \sum_{m=1}^{M} c_m \, (a_m s + b_m t + d_m) \quad \text{with} \quad \sum_{m=1}^{M} |c_m| \le C, \; |a_m| + |b_m| + |d_m| \le 1, \forall m \in [M],$$

such that $\sup_{(s,t)\in[-R,R]^2} |f(s,t) - \ell_s(s,t)| \le \varepsilon$.

Next, for every $m \in [M]$, we define matrices $Q_m, K_m, V_m \in \mathbb{R}^{D\times D}$ such that

$$Q_m h_i = \begin{bmatrix} a_m w \\ b_m \\ d_m \\ 2 \\ 0 \end{bmatrix}; \quad K_m h_j = \begin{bmatrix} x_j \\ y_j' \\ 1 \\ -R(1-t_j) \\ 0 \end{bmatrix}; \quad V_m h_j = \frac{(N+1)\, c_m}{N} \begin{bmatrix} 0_d \\ 0 \\ x_j \\ 0_{D-2d-1} \end{bmatrix}$$

for all $i, j \in [N+1]$. As the input has structure $h_i = [x_i; y_i'; w; 0_{D-2d-3}; 1; t_i]$, these matrices indeed exist, and further it is straightforward to check that they have norm bounds

$$\max_{m\in[M]} \|Q_m\|_{\mathrm{op}} \le 3, \quad \max_{m\in[M]} \|K_m\|_{\mathrm{op}} \le 2 + R, \quad \sum_{m\in[M]} \|V_m\|_{\mathrm{op}} \le 2C.$$

Consequently, $\|H\| \le 2 + R + 2C$.

Now, for every $i, j \in [N+1]$, we have

$$\begin{aligned} (h Q_m h_i, K_m h_j i) &= a_m w^\top x_j + b_m(1-t_j) y_j + d_m - 2Rt_j \\ &= a_m w^\top x_j + b_m y_j + d_m - 1\{t_j = 1\}; \end{aligned}$$

where the last equality follows from the bound

$$a_m w^\top x_j + b_m(1-t_j) y_j + d_m \le |a_m| B_x B_w + R \le 2R; \tag{15}$$

so that the above relu equals 0 if $t_j \le 0$. Therefore,

$$\begin{aligned} &\sum_{m=1}^{M} (h Q_m h_i, K_m h_j i) V_m h_j \\ &= \sum_{m=1}^{M} c_m \left[a_m w^\top x_j + b_m y_j + d_m - \frac{(N+1)}{N} 1\{t_j = 0\} \right] [0_{d+1}; x_j; 0_2] \\ &= f(w^\top x_j, y_j) - \frac{(N+1)}{N} 1\{t_j = 0\} [0_{d+1}; x_j; 0_{D-2d-1}]. \end{aligned}$$

Thus letting the attention layer $\ell = f(V_m, Q_m, K_m)_{m\in[M]}$, we have

$$\begin{aligned} \tilde{h}_i = [\mathrm{Attn}_\ell(H)]_i &= h_i + \frac{1}{N+1} \sum_{j=1}^{N+1} \sum_{m=1}^{M} (h Q_m h_i, K_m h_j i) V_m h_j \\ &= h_i - \frac{1}{N} \sum_{j=1}^{N} f(w^\top x_j, y_j)[0_{d+1}; x_j; 0_2] \\ &= [x_i; y_i; w; 1; t_i] - \frac{1}{N} \sum_{j=1}^{N} \ell_s'(w^\top x_j, y_j)[0_{d+1}; x_j; 0_{D-2d-1}] + [0_{d+1}; \varepsilon; 0_{D-2d-1}] \\ &\qquad \underbrace{\qquad\qquad\qquad\qquad\qquad\qquad}_{[0_{d+1}; -\nabla \hat{L}_N(w); 0_{D-2d-1}]} \\ &= [x_i; y_i; w^+ + \varepsilon; 0_{D-2d-3}; 1; t_i]; \end{aligned}$$

where the error vector $\varepsilon \in \mathbb{R}^d$ satisfies

$$\|\varepsilon\|_2 = \left\| \frac{1}{N} \sum_{j=1}^{N} f(w^\top x_j, y_j) - \ell_s'(w^\top x_j, y_j) \, x_j \right\|_2$$

$$\frac{1}{N}\sum_{j=1}^{N} f(w^\top x_j; y_j) \quad \partial_s \ell(w^\top x_j; y_j) \quad k x_j k_2$$

$$\frac{1}{N} \quad N \quad " \quad B_x = " \quad (\quad B_x):$$

This is the desired result. □

E.2 Proof of Theorem D.1

We first prove part (a), which requires constructing the first L layers of . Note that by our precondition $L \quad B_w=(2")$.

By our precondition, the partial derivative of the loss $\partial_s \ell$ is $(";R;M;C)$-approximable by sum of relus. Therefore we can apply Proposition E.1 to obtain that, there exists a single attention layer $^{(1)} = f(Q_m; K_m; V_m)g_{m\in[M]}$ with M heads (and norm bounds specified in Proposition E.1), such that for any w with $kwk_2 \quad B_w$, the attention layer $\mathrm{Attn}_{(1)}$ maps the input $h_i = [x_i; y'_i; w; 0_{D-2d-3}; 1; t_i]$ to output $h'_i = [x_i; y'_i; \tilde{w}; 0_{D-2d-3}; 1; t_i]$ for all $i \ 2 \ [N+1]$, where

$$\tilde{w} \quad w \quad r \frac{b}{L_N(w)} \quad _2 \quad " \quad (\quad B_x) =: "':$$

Consider the L-layer transformer $^{1:L} = (^{(1)}; : : : ; ^{(1)})$ which stacks the same attention layer $^{(1)}$ for L times, and for the given input $h_i^{(0)} = [x_i; y'_i; w^0; 0_{D-2d-3}; 1; t_i]$, its ℓ-th layer's output $h_i^{(\ell)} = [x_i; y'_i; \tilde{w}^\ell; 0_{D-2d-3}; 1; t_i]$.

We now inductively show that $\tilde{w}^\ell _2 \quad B_w$ and $\tilde{w}^\ell \quad w^\ell_{GD} _2 \quad \ell"$ for all $\ell \ 2 \ [L]$. The base case of $\ell = 0$ is trivial. Suppose the claim holds for ℓ. Then for $\ell + 1 \quad L \quad B_w=(2")$, the sequence $f\tilde{w}^i g_{i\le\ell+1}$ and $fw^i_{GD}g_{i\le\ell+1}$ satisfies the precondition of the error composition lemma (Lemma D.1) with error bound $"$, from which we obtain $\tilde{w}^{\ell+1} _2 \quad B_w$ and

$$\tilde{w}^{\ell+1} \quad w^{\ell+1}_{GD} _2 \quad (\ell+1)"':$$

This finishes the induction, and gives the following approximation guarantee for all $\ell \ 2 \ [L]$:

$$\tilde{w}^\ell \quad w^\ell_{GD} _2 \quad \ell"' \quad " \quad (L \quad B_x);$$

which proves part (a).

We now prove part (b), which requires constructing the last attention layer $^{(L+1)}$. Recall $h_i^{(L)} = [x_i; y'_i; \tilde{w}^L; 0_{D-2d-3}; 1; t_i]$ for all $i \ 2 \ [N+1]$. We construct a 2-head attention layer $^{(L+1)} = f(Q_m^{(L+1)}; K_m^{(L+1)}; V_m^{(L+1)})g_{m=1;2}$ such that for every $i; j \ 2 \ [N+1]$,

$$Q_1^{(L+1)}h_i^{(L)} = [x_i; 0_{D-d}]; \quad K_1^{(L+1)}h_j^{(L)} = [\tilde{w}^L; 0_{D-d}]; \quad V_1^{(L+1)}h_j^{(L)} = [0_d; 1; 0_{D-d-1}];$$
$$Q_2^{(L+1)}h_i^{(L)} = [x_i; 0_{D-d}]; \quad K_2^{(L+1)}h_j^{(L)} = [\quad \tilde{w}^L; 0_{D-d}]; \quad V_2^{(L+1)}h_j^{(L)} = [0_d; \quad 1; 0_{D-d-1}]:$$

Note that the weight matrices have norm bound

$$\max_{i=1;2} \ Q_i^{(L+1)} \quad _{\mathrm{op}} \quad 1; \quad \max_{i=1;2} \ K_i^{(L+1)} \quad _{\mathrm{op}} \quad 1; \quad \sum_{i=1} \ V_i^{(L+1)} \quad _{\mathrm{op}} \quad 2:$$

Then we have

$$\begin{aligned}
h_{N+1}^{(L+1)} &= h_{N+1}^{(L)} + \frac{1}{N+1}\sum_{j=1}^{N+1}\sum_{m=1} \left\langle Q^{(L+1)}h_{N+1}^{(L)}; K^{(L+1)}h_j^{(L)} \right\rangle V^{(L+1)}h_j^{(L)} \\
&= [x_i; 0; \tilde{w}^L; 0_{D-2d-3}; 1; 1] + \quad (\quad \tilde{w}^L; x_{N+1}) \quad (\quad \tilde{w}^L; x_{N+1}) \quad [0_d; 1; 0_{D-d-1}] \\
&\overset{(i)}{=} [x_i; 0; \tilde{w}^L; 0_{D-2d-3}; 1; 1] + [0_d; \quad \tilde{w}^L; x_{N+1} \quad ; 0_{D-d-1}] \\
&= [x_i; \underbrace{\quad \tilde{w}^L; x_{N+1} \quad}_{\text{}}; \tilde{w}^L; 0_{D-2d-3}; 1; 1];
\end{aligned}$$

Above, (i) uses the identity $t = (t) \quad (t)$. Further by part (a) we have

$$\hat{p}_{N+1}(w_{GD}^{L}; x_{N+1}) = \hat{w}^{L}(w_{GD}^{L}; x_{N+1}) \quad " (L B_x^2):$$

This proves part (b), and also finishes the proof Theorem D.1 where the overall $(L+1)$-layer attention-only transformer is given by TF^0 with

$$= (\underbrace{^{(1)}; \ldots; ^{(1)}}_{L \text{ times}}; ^{(L+1)}):$$

□

E.3 Proof of Lemma D.1

As f is a convex, L_f smooth function on $B_2^d(R)$, the mapping $T : w \mapsto w \quad r f(w)$ is non-expansive in $k k_2$: Indeed, for any $w; w' \in B_2^d(R)$ we have

$$\begin{aligned} kT(w) \quad T(w')k_2 &= kw \quad r f(w) \quad (w' \quad r f(w'))k_2^2 \\ &= kw \quad w'k_2^2 \quad 2\langle w \quad w'; r f(w) \quad r f(w')\rangle + {}^2 k r f(w) \quad r f(w')k_2^2 \\ &\overset{(i)}{} kw \quad w'k_2^2 \quad 2 =L_f \quad {}^2 k r f(w) \quad r f(w')k_2^2 \overset{(ii)}{} kw \quad w'k_2^2: \end{aligned}$$

Above, (i) uses the property $\langle w \quad w'; r f(w) \quad r f(w')\rangle \quad \frac{1}{L_f} k r f(w) \quad r f(w')k_2^2$ for smooth convex functions [63, Theorem 2.1.5]; (ii) uses the precondition that $2=L_f$.

The lemma then follows directly by induction on L. The base case of $L = 0$ follows directly by assumption that $\hat{w}^0 = w_{GD}^0 \in B_2^d(R=2)$. Suppose the claim holds for iterate L. For iterate $L+1 \quad R=(2")$, we have

$$\begin{aligned} \hat{w}^{L+1} \quad w_{GD}^{L+1}{}_2 &= T(\hat{w}^{L}) + "^{L} \quad T(w_{GD}^{L})_2 \\ &\quad T(\hat{w}^{L}) \quad T(w_{GD}^{L})_2 + "^{L}{}_2 \\ &\overset{(i)}{} \hat{w}^{L} \quad w_{GD}^{L}{}_2 + " \overset{(ii)}{} (L+1)": \end{aligned}$$

Above, (i) uses the non-expansiveness, and (ii) uses the inductive hypothesis. Similarly, by our assumption $w^? = T(w^?)$,

$$\hat{w}^{L+1} \quad w^?{}_2 = T(\hat{w}^{L}) + "^{L} \quad T(w^?)_2 \quad \hat{w}^{L} \quad w^?{}_2 + "^{L}{}_2 \quad \frac{R}{2} + (L+1)" \quad R:$$

This finishes the induction. □

E.4 Convex ICGD with ℓ_2 regularization

In the same setting as Theorem D.1, consider the ICGD dynamics over an ℓ_2-regularized empirical risk:

$$w_{GD}^{t+1} := w_{GD}^{t} \quad r \hat{p}_N(w_{GD}^{t}) \qquad \text{(ICGD-}\ell_2\text{)}$$

with initialization $w_{GD}^0 \in R^d$ and learning rate > 0, where $\hat{p}_N(w) := \hat{p}_N(w) + \frac{}{2} kwk_2^2$ denotes the ℓ_2-regularized empirical risk.

Corollary E.1 (Convex ICGD with ℓ_2 regularization)**.** *Fix any $B_w > 0$, $L > 1$, > 0, and $" < B_x B_w$. Suppose the loss $\ell(\,;\,)$ is convex in the first argument, and $\partial_s \ell$ is $(";R;M;C)$-approximable by sum of relus with $R = \max\{B_x B_w; B_y; 1\}$.*

Then, there exists an attention-only transformer TF^0 with $(L+1)$ layers, $\max_{\ell\in[L]} M^{(\ell)} \quad M+1$ heads within the first L layers, and $M^{(L+1)} = 2$ such that for any input data $(D; x_{N+1})$ *with*

$$\sup_{\|w\|_2 \le B_w} \max(r^2 \hat{p}_N(w)) \quad 2^{-1}; \quad \exists w^? \in \arg\min_{w\in R^d} \hat{p}_N(w) \text{ such that } kw^?k_2 \quad B_w=2;$$

$TF^0(H^{(0)})$ *approximately implements (ICGD-ℓ_2):*

1. *(Parameter space) For every* $\grave{} \in [L]$*, the* $\grave{}$*-th layer's output* $H^{(\grave{})} = TF_{(1:\grave{})}(H^{(0)})$ *approximates* $\grave{}$ *steps of (ICGD-*$\grave{}_2$*): We have* $h_i^{(\grave{})} = [x_i; y_i'; w^{\grave{}}; 0_{D-2d-3}; 1; t_i]$ *for every* $i \in [N+1]$*, where*

$$w^{\grave{}} \quad w_{GD}^{\grave{}} \quad _2 \quad " \quad (2L \quad B_x):$$

2. *(Prediction space) The final output* $H^{(L+1)} = TF(H^{(0)})$ *approximates the prediction of* L *steps of (ICGD-*$\grave{}_2$*): We have* $h_{N+1}^{(L+1)} = [x_{N+1}; \hat{y}_{N+1}; w^L; 0_{D-2d-3}; 1; 0]$*, where*

$$\hat{y}_{N+1} \quad w_{GD}^{L}; x_{N+1} \quad " \quad (2L \quad B_x^2):$$

Further, the transformer admits norm bound $\|\| \quad \|\| \quad 2 + R + (2C + \quad)$ *:*

Proof. This construction is the same as in the proof of Theorem D.1, except that within each layer $\grave{} \in [L]$, we add one more attention head $(Q^{(\grave{})}; K^{(\grave{})}; V^{(\grave{})}) \quad R^{D\times D}$ which when acting on its input $h_i^{(\grave{}-1)} = [\ ;\ ; w^{\grave{}-1}; 1;\ ]$ gives

$$Q^{(\grave{})}h_i^{(\grave{}-1)} = \begin{matrix} 1 \\ 0_{D-1} \end{matrix}; \quad K^{(\grave{})}h_j^{(\grave{}-1)} = \begin{matrix} 1 \\ 0_{D-1} \end{matrix}; \quad V^{(\grave{})}h_j^{(\grave{}-1)} = \begin{bmatrix} 0_{d+1} \\ w^{\grave{}-1} \\ 0_2 \end{bmatrix}$$

for all $i; j \in [N+1]$. Note that $\|Q^{(\grave{})}\|_{op} = \|K^{(\grave{})}\|_{op} = 1$, and $\|V^{(\grave{})}\|_{op} = \quad$. Further, it is straightforward to check that the output of this attention head on every $h_i^{(\grave{})}$ is

$$\frac{1}{N+1}\sum_{j=1}^{N+1} \left\langle Q^{(\grave{})}h_i^{(\grave{}-1)}; K^{(\grave{})}h_j^{(\grave{}-1)} \right\rangle V^{(\grave{})}h_j^{(\grave{}-1)} = \begin{bmatrix} 0_{d+1} \\ w^{\grave{}-1} \\ 0_2 \end{bmatrix}.$$

Adding this onto the original output of the $\grave{}$-th layer exactly implements the gradient of the regularizer $w \mapsto \frac{1}{2}\|w\|_2^2$. The rest of the proof follows by repeating the argument of Theorem D.1, and combining the norm bound for the additional attention head here with the norm bound therein. □

E.5 Proof of Theorem D.3

We only need to prove the following single-step version of Theorem D.3.

Proposition E.2. *Under the assumptions of Theorem D.3, there exists a 2-layer transformer* TF *with the same bounds on the number of heads, hidden dimension and the norm, such that for* any input data $(D; x_{N+1})$ *and* any $w \in R^d$*,* TF *maps*

$$h_i = [x_i; y_i'; w; 0; 1; t_i] \quad ! \quad h_i' = [x_i; y_i'; w^+; 0; 1; t_i];$$

where

$$w^+ = \mathrm{Proj}_{\mathcal{W}} \quad w \quad r\hat{L}_N(w) + "(w) \ ; \quad \|"(w)\|_2 \quad ":$$

Before we present the formal (and technical) proof of Proposition E.2, we first provide some intuitions. To begin with, we first note that

$$r_w\hat{L}_N(w) = \frac{1}{N}\sum_{i=1}^{N} \partial_1 \ell(\mathrm{pred}(x_i; w); y_i) \quad r_w \mathrm{pred}(x_i; w); \tag{16}$$

where $\partial_1 \ell$ is the partial derivative of ℓ with respect to the first component, and

$$r_w \mathrm{pred}(x_i; w) = \begin{bmatrix} u_1 & r'(\langle v_1; x_i\rangle) & x_i \\ & r(\langle v_1; x_i\rangle) & \\ & \vdots & \\ u_K & r'(\langle v_K; x_i\rangle) & x_i \\ & r(\langle v_K; x_i\rangle) & \end{bmatrix} \in R^{K(d+1)}. \tag{17}$$

Therefore, the basic idea is that we can use an attention layer to approximate $(x_i; w) \mapsto \mathrm{pred}(x_i; w)$, then use an MLP layer to implement $(\mathrm{pred}(x_i; w); y_i'; t_i) \mapsto 1\{i < N+1\}\, \ell_1'(\mathrm{pred}(x_i; w); y_i)$, and then use an attention layer to compute the gradient descent step $w \mapsto w - r L_N(w)$, and finally use an MLP layer to implement the projection into W.

Based on the observations above, we now present the proof of Proposition E.2.

Proof of Proposition E.2. We write $D_0 = d + 1 + K(d+1)$ be the length of the vector $[x_i; y_i; w]$. We also define

$$B_r := \max_{|t| \le B_x B_u} |r(t)|; \qquad B_g := \max_{|t| \le K B_r, |y| \le B_y} |\ell_t'(t; y)|.$$

Let us fix $\varepsilon_r; \varepsilon_p; \varepsilon_\ell > 0$ that will be specified later in proof (see (18)). By our assumption and Proposition B.1, the following facts hold.

(1) The function $r(t)$ is $(\varepsilon_r; R_1; M_1; C_1)$ for $R_1 = \max\{B_x B_u; 1\}$, $M_1 = \Theta(C_1^2 \varepsilon_r^{-2})$, where C_1 depends only on R_1 and the C^2-smoothness of r. Therefore, there exists

$$\bar{r}(t) = \sum_{m=1}^{M_1} c_m^1 \sigma(\langle a_m^1, [t; 1]\rangle) \text{ with } \sum_{m=1}^{M_1} |c_m^1| \le C_1; \quad \|a_{m\,1}^1\| \le 1; \ \forall m \in [M_1];$$

such that $\sup_{t \in [-R_1, R_1]} |r(t) - \bar{r}(t)| \le \varepsilon_r$.

(2) The function $(t; y) \mapsto \ell_1'(t; y)$ is $(\varepsilon_\ell; R_2; M_2; C_2)$ for $R_2 = \max\{K B_r; B_y; 1\}$ $M_2 = \Theta(C_2^2 \varepsilon_\ell^{-2})$, where C_2 depends only on R_2 and the C^3-smoothness of ℓ_1'. Therefore, there exists

$$g(t; y) = \sum_{m=1}^{M_2} c_m^2 \sigma(\langle a_m^2, [t; y; 1]\rangle) \text{ with } \sum_{m=1}^{M_2} |c_m^2| \le C_2; \quad \|a_{m\,1}^2\| \le 1; \ \forall m \in [M_2];$$

such that $\sup_{(t;y) \in [-R_2; R_2]^2} |g(t; y) - \ell_1'(t; y)| \le \varepsilon_\ell$.

(3) The function $(s; t) \mapsto s - r'(t)$ is $(\varepsilon_p; R_3; M_3; C_3)$ for $R_3 = \max\{B_x B_u; B_g B_u; 1\}$, $M_3 = \Theta(C_3^2 \varepsilon_p^{-2})$, where C_3 depends only on R_3 and the C^3-smoothness of r'. Therefore, there exists

$$P(s; t) = \sum_{m=1}^{M_3} c_m^3 \sigma(\langle a_m^3, [s; t; 1]\rangle) \text{ with } \sum_{m=1}^{M_3} |c_m^3| \le C_3; \quad \|a_{m\,1}^3\| \le 1; \ \forall m \in [M_3];$$

such that $\sup_{(s;t) \in [-R_3; R_3]^2} |P(s; t) - s + r'(t)| \le \varepsilon_p$.

In the following, we proceed to construct the desired transformer step by step.

Step 1: construction of $\mathrm{attn}^{(1)}$. We consider the matrices $\{Q_{k;m}^{(1)}; K_{k;m}^{(1)}; V_{k;m}^{(1)}\}_{k \in [K]; m \in [M_1]}$ so that for all $i; j \in [N+1]$, we have

$$Q_{k;m}^{(1)} h_i = \begin{bmatrix} a_m^1[1] & x_i \\ a_m^1[2] & \\ 0 & \end{bmatrix}; \quad K_{k;m}^{(1)} h_j = \begin{bmatrix} \varepsilon_\ell v_k \\ 1 \\ 0 \end{bmatrix}; \quad V_{k;m}^{(1)} h_j = c_m^1 u_k e_{D_0+1}.$$

As the input has structure $h_i = [x_i; y_i'; w; 0; 1; t_i]$, these matrices indeed exist, and further it is straightforward to check that they have norm bounds

$$\max_{k;m} \|Q_{k;m}^{(1)}\|_{\mathrm{op}} \le 1; \quad \max_{k;m} \|K_{k;m}^{(1)}\|_{\mathrm{op}} \le 1; \quad \sum_{k;m} \|V_{k;m}^{(1)}\|_{\mathrm{op}} \le C_1.$$

A simple calculation shows that

$$\sum_{m \in [M_1]; k \in [K]} \langle Q_{k;m}^{(1)} h_i, K_{k;m}^{(1)} h_j \rangle V_{k;m}^{(1)} h_j = \sum_{k=1}^{K} u_k \bar{r}(\langle v_k, x_i \rangle) e_{D_0+1}.$$

For simplicity, we denote $\overline{\mathrm{pred}}(x; w) := \sum_{k=1}^{K} u_k \bar{r}(\langle v_k; x_i \rangle)$ in the following analysis. Thus, letting the attention layer $\Phi^{(1)}_{\mathrm{attn}} = f(V^{(1)}_{k;m}; Q^{(1)}_{k;m}; K^{(1)}_{k;m})g_{(k;m)}$, we have

$$\mathrm{Attn}_{\Phi^{(1)}_{\mathrm{attn}}} : h_i \mapsto h_i^{(0:5)} = [x_i; y'_i; w; \overline{\mathrm{pred}}(x_i; w); 0; 1; t_i].$$

Step 2: construction of $\Phi^{(1)}_{\mathrm{mlp}}$. We pick matrices $W_1; W_2$ so that W_1 maps

$$W_1 h_i^{(0:5)} = \begin{bmatrix} a^2_m[1] \ \overline{\mathrm{pred}}(x_i; w) + a^2_m[2] \ y'_i + a^2_m[3] \ R_2(1 - t_i) \end{bmatrix}_{m \in [M_2]} \in \mathbb{R}^{M_2};$$

and $W_2 \in \mathbb{R}^{D \times M_3}$ with entries being $(W_2)_{(j;m)} = c^2_m 1\{j = D_0 + 2\}$. It is clear that $\|W_1\|_{\mathrm{op}} \le R_2 + 1, \|W_2\|_{\mathrm{op}} \le C_2$. Then we have

$$\begin{aligned} W_2 \ (W_1 h_i^{(0:5)}) &= \sum_{m \in [M_3]} a^2_m; [\overline{\mathrm{pred}}(x_i; w); y'_i; 1] \ R_2(1 - t_j) \ c^2_m e_{D_0+2} \\ &= 1\{t_j = 1\} \ g(\overline{\mathrm{pred}}(x_i; w); y'_i) \ e_{D_0+2}. \end{aligned}$$

In the following, we abbreviate $g_i = 1\{t_j = 1\} \ g(\overline{\mathrm{pred}}(x_i; w); y'_i)$. Hence, Φ_{mlp} maps

$$\mathrm{MLP}_{\Phi_{\mathrm{mlp}}} : h_i^{(0:5)} \mapsto h_i^{(1)} = [x_i; y'_i; w; \overline{\mathrm{pred}}(x_i; w); g_i; 0; 1; t_i].$$

By the definition of the function g, for each $i \in [N]$,

$$|g_i - \ell(\mathrm{pred}(x_i; w); y_i)| \le \varepsilon + B_u L_\ell \varepsilon_r;$$

where $L_\ell := \max_{|t| \le K B_r; |y| \le B_y} |\ell''_{tt}(t; y)|$ is the smoothness of ℓ. Also, $g_{N+1} = 0$ by definition.

Step 3: construction of $\Phi^{(2)}_{\mathrm{attn}}$. We consider the matrices $\{Q^{(2)}_{k;1;m}; K^{(2)}_{k;1;m}; V^{(2)}_{k;1;m}\}_{k \in [K]; m \in [M_3]}$ so that for all $i; j \in [N + 1]$, we have

$$Q^{(2)}_{k;1;m} h_i^{(1)} = \begin{bmatrix} a^3_m[1] & u_k \\ a^3_m[2] & v_k \\ a^3_m[3] & 0 \end{bmatrix}; \quad K^{(2)}_{k;1;m} h_j^{(1)} = \begin{bmatrix} g_j \\ x_j \\ 1 \\ 0 \end{bmatrix}; \quad V^{(2)}_{k;1;m} h_j^{(1)} = \frac{(N+1) \ c^3_m}{N} \begin{bmatrix} 0_{k(d+1)} \\ x_j \\ 0 \end{bmatrix}.$$

We further consider the matrices $\{Q^{(2)}_{k;2;m}; K^{(2)}_{k;2;m}; V^{(2)}_{k;2;m}\}_{k \in [K]; m \in [M_1]}$ so that for all $i; j \in [N+1]$, we have

$$Q^{(2)}_{k;2;m} h_i^{(1)} = \begin{bmatrix} a^1_m[1] & v_k \\ a^1_m[2] & 0 \end{bmatrix}; \quad K^{(2)}_{k;2;m} h_j^{(1)} = \begin{bmatrix} x_j \\ 1 \\ 0 \end{bmatrix}; \quad V^{(2)}_{k;2;m} h_j^{(1)} = \frac{(N+1) \ c^1_m}{N} \begin{bmatrix} 0_{k(d+1)+d} \\ g_j \\ 0 \end{bmatrix}.$$

By the structure of the input $h_i^{(1)}$, these matrices indeed exist, and further it is straightforward to check that they have norm bounds

$$\max_{(k;w;m)} \|Q^{(2)}_{k;w;m}\|_{\mathrm{op}} \le 1; \quad \max_{(k;w;m)} \|K^{(2)}_{k;w;m}\|_{\mathrm{op}} \le 1; \quad \sum_{(k;w;m)} \|V^{(2)}_{k;w;m}\|_{\mathrm{op}} \le 2 \ C_1 + 2 \ C_3.$$

Furthermore, a simple calculation shows that

$$g(w) =: \frac{1}{N+1} \sum_{i=1}^{N+1} \sum_{(k;w;m)} \left\langle Q^{(2)}_{k;w;m} h_i; K^{(2)}_{k;w;m} h_j \right\rangle V^{(2)}_{k;w;m} h_j = \frac{1}{N} \sum_{j=1}^{N+1} \begin{bmatrix} 0_{d+1} \\ P(u_1 g_j; \langle v_1; x_j \rangle) \ x_j \\ \bar{r}(\langle v_1; x_j \rangle) \ g_j \\ \vdots \\ P(u_K g_j; \langle v_K; x_j \rangle) \ x_j \\ \bar{r}(\langle v_K; x_j \rangle) \ g_j \\ 0 \end{bmatrix};$$

where the summation is taken over all possibilities of the tuple $(k; w; m)$, i.e. over the union of $[K] \times \{1\} \times [M_3]$ and $[K] \times \{2\} \times [M_1]$.

By our definition, we have $|P(s,t) - sr'(t)| \le \varepsilon_p$ for all $s,t \in [-R_3, R_3]$. Therefore, for each $i \in [N], k \in [K]$,

$$|P(u_k g_j; h(v_k, x_j i)) - \ell(\mathrm{pred}(x_j; w), y_j) - u_k r'(h(v_k, x_j i))| \le \varepsilon_p + |g_j - \ell(\mathrm{pred}(x_i; w), y_i)| \le |u_k| \cdot |r'(h(v_k, x_j i))| \le \varepsilon_p + B_u L_r(\varepsilon_\xi + B_u L_\xi \varepsilon_r);$$

where $L_r := \max_{|t| \le B_x B_u} |r'(t)|$ is the upper bound of r'. Similarly, for each $i \in [N], k \in [K]$, we have

$$|\bar{r}(h(v_k, x_j i)) - g_j - r(h(v_k, x_j i)) + \ell(\mathrm{pred}(x_j; w), y_j)| \le 2B_g \varepsilon_r + 2B_r(\varepsilon_\xi + B_u L_\xi \varepsilon_r).$$

As for the case $i = N+1$, we have $g_{N+1} = 0$ and $|P(u_k g_{N+1}; h(v_k, x_{N+1} i))| \le \varepsilon_p$ for each $k \in [K]$ by defintion. Combining these estimations and using (16) and (17), we can conclude that

$$\|\nabla^{-1} g(w) + r \hat{L}_N(w)\|_2 \le \sqrt{K} B_x [\varepsilon_p + B_u L_r(\varepsilon_\xi + B_u L_\xi \varepsilon_r)] + 2\sqrt{K}[B_g \varepsilon_r + B_r(\varepsilon_\xi + B_u L_\xi \varepsilon_r)].$$

Thus, to ensure $\|\nabla^{-1} g(w) + r \hat{L}_N(w)\|_2 \le \varepsilon$, we only need to choose $\varepsilon_p, \varepsilon_\xi, \varepsilon_r$ as

$$\varepsilon_p = \frac{\varepsilon}{3\sqrt{K} B_x}; \qquad \varepsilon_\xi = \frac{\varepsilon}{9\sqrt{K}\max\{B_r, L_r B_x B_u\}}; \qquad \varepsilon_r = \frac{\varepsilon}{15\sqrt{K}\max\{B_g, L_\xi B_r B_u, L_r L_\xi B_x B_r B_u^2 g\}}. \tag{18}$$

Thus, letting the attention layer $\theta^{(2)}_{\mathrm{attn}} = f(V^{(2)}_{k;w;m}, Q^{(2)}_{k;w;m}, K^{(2)}_{k;w;m})_{g_{(k;w;m)}}$, we have

$$\mathrm{Attn}_{\theta^{(2)}_{\mathrm{attn}}} : h_i^{(1)} \mapsto h_i^{(1:5)} = [x_i, y_i', w + g(w), \overline{\mathrm{pred}}(x_i; w), g_i, 0, 1, t_i].$$

Step 4: construction of $\theta^{(2)}_{\mathrm{mlp}}$. We only need to pick $\theta^{(2)}_{\mathrm{mlp}}$ so that it maps

$$h_i^{(1:5)} = [x_i, y_i', w + g(w), \overline{\mathrm{pred}}(x_i; w), g_i, 0, 1, t_i] \xrightarrow{\mathrm{MLP}^{(2)}_{\mathrm{mlp}}} h_i^{(2)} = [x_i, y_i', \mathrm{Proj}_{\mathcal{W}}(w - g(w)), 0, 0, 0, 1, t_i].$$

By our assumption on the map $\mathrm{Proj}_{\mathcal{W}}$, this is easy.

Combining the four steps above and taking $\theta = (\theta^{(1)}_{\mathrm{attn}}, \theta^{(1)}_{\mathrm{mlp}}, \theta^{(2)}_{\mathrm{attn}}, \theta^{(2)}_{\mathrm{mlp}})$ completes the proof. □

E.6 Proof of Lemma D.3

For every $\ell \ge 0$, define the intermediate iterates (before projection)

$$w^{\ell+\frac{1}{2}} := w^\ell - r\nabla f(w^\ell) + \varepsilon^\ell; \quad w_{GD}^{\ell+\frac{1}{2}} := w_{GD}^\ell - r\nabla f(w_{GD}^\ell);$$

so that $w^{\ell+1} = \mathrm{Proj}_{\mathcal{W}}(w^{\ell+\frac{1}{2}})$ and $w_{GD}^{\ell+1} = \mathrm{Proj}_{\mathcal{W}}(w_{GD}^{\ell+\frac{1}{2}})$.

We first prove part (a). We begin by deriving a relation between $\|w^{\ell+1} - w^\ell\|_2^2$ and $\|G^f_{\mathcal{W},r}(w^\ell)\|_2^2$.

Let $v^{\ell+\frac{1}{2}} := w^\ell - r\nabla f(w^\ell)$ and $v^{\ell+1} := \mathrm{Proj}_{\mathcal{W}}(v^{\ell+\frac{1}{2}})$ denote the *exact* projected gradient iterate starting from w^ℓ. We have

$$\begin{aligned}\|w^{\ell+1} - w^\ell\|_2^2 &\overset{(i)}{\le} \frac{1}{2}\|v^{\ell+1} - w^\ell\|_2^2 + \frac{1}{2}\|v^{\ell+1} - w^{\ell+1}\|_2^2 \overset{(ii)}{\le} \frac{1}{2}\|v^{\ell+1} - w^\ell\|_2^2 + \frac{1}{2}\|v^{\ell+\frac{1}{2}} - w^{\ell+\frac{1}{2}}\|_2^2 \\ &\overset{(iii)}{=} \frac{r^2}{2}\|G^f_{\mathcal{W},r}(w^\ell)\|_2^2 + \frac{r^2}{2}\|G^f_{\mathcal{W},r}(w^\ell)\|_2^2 = r^2\|G^f_{\mathcal{W},r}(w^\ell)\|_2^2.\end{aligned} \tag{19}$$

Above, (i) uses the inequality $\|a+b\|_2^2 \le \frac{1}{2}\|a\|_2^2 + \|b\|_2^2$; (ii) uses the fact that projection to a convex set is a non-expansion; (iii) uses the definition of the gradient mapping.

By the L_f-smoothness of f within W, we have

$$f(w^{\ell+1}) - f(w^\ell) \le -r\nabla f(w^\ell)^\top (w^{\ell+1} - w^\ell) + \frac{L_f}{2}\|w^{\ell+1} - w^\ell\|_2^2$$

$$= \left\langle \frac{\mathbf{w}_t - \mathbf{w}_{t+\frac{1}{2}}}{\eta_t}, \mathbf{w}_{t+1} - \mathbf{w}_t \right\rangle + \frac{L_f}{2}\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2$$

$$\overset{(i)}{=} \left\langle \frac{\mathbf{w}_t - \mathbf{w}_{t+1}}{\eta_t}, \mathbf{w}_{t+1} - \mathbf{w}_t \right\rangle + \left\langle \nabla_t, \mathbf{w}_t - \mathbf{w}_{t+1} \right\rangle + \frac{L_f}{2}\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2$$

$$= \left(\frac{1}{\eta_t} + \frac{L_f}{2}\right)\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2 + \frac{1}{4}\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2 + \|\nabla_t\|_2^2$$

$$\overset{(ii)}{\leq} \frac{1}{4}\|\mathbf{w}_{t+1} - \mathbf{w}_t\|_2^2 + \|\nabla_t\|_2^2$$

$$\overset{(iii)}{\leq} \frac{1}{4}\cdot 2\left(\|\mathbf{w}_{t+1} - \mathbf{w}_{GD}^{t+1}\|_2^2 + \|\mathbf{w}_t - \mathbf{w}_{GD}^t\|_2^2\right) + \|\nabla_t\|_2^2$$

$$\leq \frac{1}{8}\|\mathbf{w}_{t+1} - \mathbf{w}_{GD}^{t+1}\|_2^2 + \frac{5}{4}\epsilon^2.$$

Above, (i) uses the property $\left\langle \mathbf{w}_{t+1} - \mathbf{w}_{t+\frac{1}{2}}, \mathbf{w}_{t+1} - \mathbf{w}_t \right\rangle \geq 0$ of the projection $\mathbf{w}_{t+1} = \mathrm{Proj}_{\mathcal{W}}(\mathbf{w}_{t+\frac{1}{2}})$ (using $\mathbf{w}_t \in \mathcal{W}$); (ii) uses $L_f/2 \leq 1/(2\eta_t)$ by our choice of $\eta_t \leq 1/L_f$; (iii) uses (19).

Rearranging and summing the above over $t = 0, \ldots, L-1$, we obtain

$$\frac{1}{8}\sum_{t=0}^{L-1}\|\mathbf{w}_{t+1} - \mathbf{w}_{GD}^{t+1}\|_2^2 \leq f(\mathbf{w}^0) - f(\mathbf{w}^L) + \frac{5L}{4}\epsilon^2.$$

Dividing both sides by $L/8$ yields part (a).

Next, we prove part (b). Let $C := 1 + L_f$. We prove by induction that

$$\|\mathbf{w}_t - \mathbf{w}_{GD}^t\|_2 \leq \frac{C^t - 1}{C - 1}\epsilon \tag{20}$$

for all $t \geq 0$. The base case of $t = 0$ follows by definition that $\mathbf{w}^0 = \mathbf{w}_{GD}^0 = \mathbf{w}^0$. Suppose the result holds for t. Then for $t+1$, we have

$$\|\mathbf{w}_{t+1} - \mathbf{w}_{GD}^{t+1}\|_2 \overset{(i)}{=} \|\mathbf{w}_{t+\frac{1}{2}} - \mathbf{w}_{GD}^{t+\frac{1}{2}}\|_2 = \|\mathbf{w}_t - \eta_t\nabla f(\mathbf{w}_t) - \mathbf{w}_{GD}^t + \eta_t\nabla f(\mathbf{w}_{GD}^t)\|_2$$

$$\overset{(ii)}{\leq} C\|\mathbf{w}_t - \mathbf{w}_{GD}^t\|_2 + \epsilon \overset{(iii)}{\leq} C\cdot\frac{C^t - 1}{C - 1}\epsilon + \epsilon = \frac{C^{t+1} - 1}{C - 1}\epsilon.$$

Above, (i) uses again the non-expansiveness of the convex projection $\mathrm{Proj}_{\mathcal{W}}$; (ii) uses the fact that the operator $\mathbf{w} \mapsto \mathbf{w} - \eta_t\nabla f(\mathbf{w})$ is $(1 + \eta_t L_f) = C$-Lipschitz; and (iii) uses the inductive hypothesis. This proves the case for $t+1$ and thus finishes the induction. We can further relax (20) into

$$\|\mathbf{w}_t - \mathbf{w}_{GD}^t\|_2 \leq \frac{C^t}{1 + \eta_t L_f - 1}\epsilon = \eta_t^{-1}(1 + \eta_t L_f)^t\epsilon.$$

This proves part (b). □

F Proofs for Section 3.1

F.1 Proof of Theorem 4

Fix $\alpha, \beta > 0$ with $\beta := \frac{\alpha}{\alpha+4}$, and $B_w > 0$, and consider any in-context data D such that the precondition of Theorem 4 holds. Let

$$L_{\mathrm{ridge}}(\mathbf{w}) := \frac{1}{2N}\sum_{i=1}^{N}(\langle \mathbf{w}, \mathbf{x}_i\rangle - y_i)^2 + \frac{\beta}{2}\|\mathbf{w}\|_2^2$$

denote the ridge regression loss in (ICRidge), so that $w_{\text{ridge}} = \arg\min_{w \in \mathbb{R}^d} L_{\text{ridge}}(w)$. It is a standard result that $\nabla^2 L_{\text{ridge}}(w) = X^\top X/N + \lambda I_d$, so that L_{ridge} is $(\lambda + \cdot)$-strongly convex and $(\lambda + \cdot)$-smooth over $\mathbb{R}^d$.

Consider the gradient descent algorithm on the ridge loss

$$w_{\text{GD}}^{t+1} = w_{\text{GD}}^t - \eta \nabla L_{\text{ridge}}(w_{\text{GD}}^t)$$

with initialization, learning rate, and number of steps

$$w_{\text{GD}}^0 := 0_d; \quad \eta := \frac{1}{\lambda + \cdot}; \quad T := \lceil 2 \log \frac{B_x B_w}{2\varepsilon} \rceil :$$

By standard convergence results for strongly convex and smooth functions (Proposition B.2), we have for all $t \geq 1$ that

$$\|w_{\text{GD}}^t - w_{\text{ridge}}\|_2^2 \leq \exp\left(-\frac{t}{\cdot}\right) \|w_{\text{GD}}^0 - w_{\text{ridge}}\|_2^2 = \exp\left(-\frac{t}{\cdot}\right) \|w_{\text{ridge}}\|_2^2 :$$

Further, we have

$$\|w_{\text{GD}}^T - w_{\text{ridge}}\|_2 \leq \exp\left(-\frac{T}{2}\right) \|w_{\text{ridge}}\|_2 \leq \frac{2\varepsilon}{B_x B_w} \cdot \frac{B_w}{2} \leq \frac{\varepsilon}{B_x}: \tag{21}$$

It remains to construct a transformer to approximate w_{GD}^T. Notice that the problem (ICRidge) corresponds to an ℓ_2-regularized ERM with the square loss $\ell(s,t) := \frac{1}{2}(s - t)^2$, whose partial derivative $\partial_s \ell(s,t) = s - t$ is exactly a sum of two relus:

$$\partial_s \ell(s,t) = 2 \cdot \mathrm{ReLU}((s-t)/2) - 2 \cdot \mathrm{ReLU}(-(s-t)/2):$$

In particular, this shows that $\partial_s \ell(s,t)$ is $(0, R, 2, 4)$-approximable for any $R > 0$, in particular for $R = \max\{B_x B_w, B_y, 1\}$.

Therefore, we can apply Corollary E.1 with the square loss ℓ, learning rate η, regularization strength λ and accuracy parameter $\varepsilon = 0$ to obtain that there exists an attention-only transformer TF^0 with $(T+1) := L$ layers such that the final output $h_{N+1}^{(L)} = [x_{N+1}, \hat{y}_{N+1}, \cdot]$ with

$$\hat{y}_{N+1} - \langle w_{\text{GD}}^T, x_{N+1} \rangle = 0; \tag{22}$$

and number of heads $M^{(\ell)} = 3$ for all $\ell \in [L-1]$ (can be taken as 2 in the unregularized case $\lambda = 0$ directly by Theorem D.1), and $M^{(L)} = 2$. Further, $\cdot$ admits norm bound $\|\cdot\| \leq 2 + R + \frac{8+\lambda}{\eta}$ $3R + 8(\lambda + \cdot)^{-1} + 1 \leq 4R + 8(\lambda + \cdot)^{-1}$.

Combining (21) and (22), we obtain that

$$\hat{y}_{N+1} - \langle w_{\text{ridge}}, x_{N+1} \rangle = \langle w_{\text{GD}}^T - w_{\text{ridge}}, x_{N+1} \rangle \leq (\varepsilon/B_x) \cdot B_x = \varepsilon:$$

Further, we have $\mathrm{read}_w(h_i^T) = w_{\text{GD}}^T$ for all $i \in [N+1]$, where $\mathrm{read}_w(h) := h_{(d+2):(2d+1)}$ (cf. Corollary E.1), so that $\|\mathrm{read}_w(h_i^T) - w_{\text{ridge}}\|_2 \leq \varepsilon/B_x$ as shown above. This finishes the proof. □

F.2 Statistical analysis of in-context least squares

Consider the standard least-squares algorithm A_{LS} and least-squares estimator $\hat{w}_{\text{LS}} \in \mathbb{R}^d$ defined as

$$A_{\text{LS}}(D)(x_{N+1}) := \langle \hat{w}_{\text{LS}}, x_{N+1} \rangle; \quad \hat{w}_{\text{LS}} = (X^\top X)^{-1} X^\top y \in \mathbb{R}^d: \tag{ICLS}$$

For any distribution P over $(x,y) \in \mathbb{R}^d \times \mathbb{R}$ and any estimator $w \in \mathbb{R}^d$, let

$$L_P(w) := \mathbb{E}_{(x',y') \sim P} \left[\frac{1}{2} (\langle w, x' \rangle - y')^2 \right]$$

denote the expected risk of w over a new test example $(x', y') \sim P$.

Assumption A (Well-posedness for learning linear predictors)**.** *We say a distribution P on $\mathbb{R}^d \times \mathbb{R}$ is well-posed for learning linear predictors, if $(x,y) \sim P$ satisfies*

(1) $\|x\|_2 \le B_x$ *and* $|y| \le B_y$ *almost surely;*

(2) The covariance $\Sigma_P := \mathbb{E}_P[xx^\top]$ *satisfies* $\lambda_{\min} I_d \preceq \Sigma_P \preceq \lambda_{\max} I_d$*, with* $0 < \lambda_{\min} \le \lambda_{\max}$*, and* $\kappa := \lambda_{\max}/\lambda_{\min}$*.*

(3) The whitened vector $\Sigma_P^{-1/2} x$ *is* K^2*-sub-Gaussian for some* $K \ge 1$*.*

(4) The best linear predictor $w_P^\star := \mathbb{E}_P[xx^\top]^{-1}\mathbb{E}_P[xy]$ *satisfies* $\|w_P^\star\|_2 \le B_w^\star$*.*

(5) We have $\mathbb{E}[(y - \langle x, w_P^\star\rangle)^2 | x] \le \sigma^2$ *with probability one (over* x*).*

Further, we say P *is well-posed with* canonical parameters *if*

$$B_x = \Theta(\sqrt{d}); \quad B_y = \Theta(1); \quad B_w^\star = \Theta(1); \quad \sigma = O(1); \quad \kappa_{\max} = \Theta(1); \quad K = \Theta(1); \tag{23}$$

where $\Theta(\cdot)$ *and* $O(\cdot)$ *only hides absolute constants.*

The following result bounds the excess risk of least squares under Assumption A with a clipping operation on the predictor; the clipping allows the result to only depend on the second moment of the noise (cf. Assumption A(5)) instead of e.g. its sub-Gaussianity, and also makes the result convenient to be directly translated to a result for transformers.

Proposition F.1 (Guarantees for in-context least squares)**.** *Suppose distribution* P *satisfies Assumption A. Then as long as* $N \ge O(dK^4 \log(1/\delta))$*, we have the following:*

(a) The (clipped) least squares predictor achieves small expected excess risk (fast rate) over the best linear predictor: For any clipping radius $R \ge B_y$*,*

$$\mathbb{E}_{\mathcal{D}; x_{N+1}, y_{N+1} \sim P} \frac{1}{2}(\mathrm{clip}_R(\langle \hat{w}_{\mathrm{LS}}; x_{N+1}\rangle) - y_{N+1})^2 \le \underbrace{\inf_{w \in \mathbb{R}^d} L_P(w)}_{L_P(w_P^\star)} + O\left(R^2 \cdot \frac{d}{N}\right). \tag{24}$$

(b) We have $P(E_{\mathrm{cov}} \cap E_w) \ge 1 - 1/10$*, where*

$$E_{\mathrm{cov}} = E_{\mathrm{cov}}(D) := \frac{1}{2} I_d \preceq \Sigma_P^{-1/2} \hat{\Sigma} \Sigma_P^{-1/2} \preceq 2 I_d; \tag{25}$$

$$E_w = E_w(D) := \left\{ \|\hat{w}_{\mathrm{LS}}\|_2 \le B_w^\star + \sqrt{\frac{80 d^2}{N \lambda_{\min}}} \right\}. \tag{26}$$

Proof. We first show $P(E_{\mathrm{cov}}) \ge 1 - 1/20$. Let $\hat{\Sigma} := \frac{1}{N}\sum_{i=1}^N x_i x_i^\top$, and let the whitened covariance and noise variables be denoted as

$$\tilde{x}_i = \Sigma_P^{-1/2} x_i; \quad \tilde{\Sigma} := \frac{1}{N}\sum_{i=1}^N \tilde{x}_i \tilde{x}_i^\top = \Sigma_P^{-1/2} \hat{\Sigma} \Sigma_P^{-1/2}.$$

Also let $z_i := y_i - \langle x_i, w_P^\star\rangle$ denote the "noise" variables. Note that

$$E_{\mathrm{cov}} = \frac{1}{2} I_d \preceq \tilde{\Sigma} \preceq 2 I_d$$

is exactly a covariance concentration of the whitened vectors $\{\tilde{x}_i\}_{i \in [N]}$. Recall that $\mathbb{E}[\tilde{x}_i \tilde{x}_i^\top] = I_d$, and $\tilde{x}_i$ are K^2-sub-Gaussian by assumption. Therefore, we can apply [85, Theorem 4.6.1], we have with probability at least $1 - 1/10$ that

$$\left\| \tilde{\Sigma} - I_d \right\|_{\mathrm{op}} \le O\left(K^2 \max\left\{ \sqrt{\frac{d + \log(1/\delta)}{N}}, \frac{d + \log(1/\delta)}{N} \right\}\right).$$

Setting $N \ge O(K^4(d + \log(1/\delta)))$ ensures that the right-hand side above is at most $1/2$, on which event we have

$$\frac{1}{2} I_d \preceq \tilde{\Sigma} \preceq \frac{3}{2} I_d \preceq 2 I_d; \tag{27}$$

i.e. E_{cov} holds. This shows that $\mathsf{P}(E_{\text{cov}}^c) = 0$.

Next, we show (24). Using E_{cov}, we decompose the risk as

$$
\begin{aligned}
&\mathsf{E}\,\frac{1}{2}(\text{clip}_R(\langle \hat{\mathbf{w}}_{\text{LS}}, \mathbf{x}_{N+1}\rangle) - y_{N+1})^2 \\
&= \mathsf{E}\,\frac{1}{2}(\text{clip}_R(\langle \hat{\mathbf{w}}_{\text{LS}}, \mathbf{x}_{N+1}\rangle) - y_{N+1})^2 \mathbf{1}\{E_{\text{cov}}\} + \mathsf{E}\,\frac{1}{2}(\text{clip}_R(\langle \hat{\mathbf{w}}_{\text{LS}}, \mathbf{x}_{N+1}\rangle) - y_{N+1})^2 \mathbf{1}\{E_{\text{cov}}^c\} \\
&\overset{(i)}{\le} \mathsf{E}\,\frac{1}{2}(\langle \hat{\mathbf{w}}_{\text{LS}}, \mathbf{x}_{N+1}\rangle - y_{N+1})^2 \mathbf{1}\{E_{\text{cov}}\} + 2R^2 \quad (= 20) \\
&\overset{(ii)}{=} \mathsf{E}_{\mathcal{D};\mathbf{x}_{N+1}}\,\frac{1}{2}(\langle \hat{\mathbf{w}}_{\text{LS}} - \mathbf{w}_{\mathsf{P}}^*, \mathbf{x}_{N+1}\rangle)^2 \mathbf{1}\{E_{\text{cov}}\} + \mathsf{E}_{\mathbf{x}_{N+1};y_{N+1}}\,\frac{1}{2}(\langle \mathbf{w}_{\mathsf{P}}^*, \mathbf{x}_{N+1}\rangle - y_{N+1})^2 \mathbf{1}\{E_{\text{cov}}\} + O(R^2) \\
&\quad \mathsf{E}_{\mathcal{D}}\,\frac{1}{2}\|\hat{\mathbf{w}}_{\text{LS}} - \mathbf{w}_{\mathsf{P}}^*\|_{\mathsf{P}}^2 \mathbf{1}\{E_{\text{cov}}\} + \underbrace{\mathsf{E}_{\mathbf{x}_{N+1};y_{N+1}}\,\frac{1}{2}(\langle \mathbf{w}_{\mathsf{P}}^*, \mathbf{x}_{N+1}\rangle - y_{N+1})^2}_{L_{\mathsf{P}}(\mathbf{w}_{\mathsf{P}}^*)} + O(R^2).
\end{aligned}
\tag{28}
$$

Above, (i) follows by assumption that $|y_{N+1}| \le B_y \le R$ almost surely, so that removing the clipping can only potentially increase the distance in the first term, and the square loss is upper bounded by $\frac{1}{2}(2R)^2$ almost surely in the second term; (ii) follows by the fact that $\mathsf{E}_{\mathbf{x}_{N+1};y_{N+1}}[\langle \hat{\mathbf{w}}_{\text{LS}} - \mathbf{w}_{\mathsf{P}}^*, \mathbf{x}_{N+1}\rangle (\langle \mathbf{w}_{\mathsf{P}}^*, \mathbf{x}_{N+1}\rangle - y_{N+1})] = 0$ by the definition of $\mathbf{w}_{\mathsf{P}}^*$, as well as the fact that $\mathbf{1}\{E_{\text{cov}}\}$ is independent of $(\mathbf{x}_{N+1}, y_{N+1})$.

It thus remains to bound $\mathsf{E}_{\mathcal{D}}\left[\frac{1}{2}\|\hat{\mathbf{w}}_{\text{LS}} - \mathbf{w}_{\mathsf{P}}^*\|_{\mathsf{P}}^2 \mathbf{1}\{E_{\text{cov}}\}\right]$. Note that on the event E_{cov}, we have

$$
\mathsf{P}^{1/2}\left(\mathsf{P}^{-1} - \mathsf{P}^{-1/2}\right)^{-1}\mathsf{P}^{1/2} = \left(\mathsf{P}^{-1/2}\left(\mathsf{P}^{-1} - \mathsf{P}^{-1/2}\right)^{-1}\mathsf{P}^{-1/2}\right)^{-1} \succeq 2\mathbf{I}_d.
$$

Therefore,

$$
\begin{aligned}
&\frac{1}{2}\|\hat{\mathbf{w}}_{\text{LS}} - \mathbf{w}_{\mathsf{P}}^*\|_{\mathsf{P}}^2 \mathbf{1}\{E_{\text{cov}}\} = \frac{1}{2}\,(\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top\mathbf{y} - \mathbf{w}_{\mathsf{P}}^*\Big)^\top \mathsf{P}\,(\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top\mathbf{y} - \mathbf{w}_{\mathsf{P}}^*\Big)\mathbf{1}\{E_{\text{cov}}\} \\
&= \frac{1}{2}\boldsymbol{\varepsilon}^\top\mathbf{X}(\mathbf{X}^\top\mathbf{X})^{-1}\,\mathsf{P}(\mathbf{X}^\top\mathbf{X})^{-1}\mathbf{X}^\top\boldsymbol{\varepsilon}\;\mathbf{1}\{E_{\text{cov}}\} \\
&= \frac{1}{2N^2}\boldsymbol{\varepsilon}^\top\mathbf{X}\,\mathsf{P}^{-1/2}\left(\mathsf{P}^{1/2}\left(\frac{\mathbf{X}^\top\mathbf{X}}{N} - \mathsf{P}\right)\mathsf{P}^{1/2}\right)^{-1}\mathsf{P}^{-1/2}\mathbf{X}^\top\boldsymbol{\varepsilon}\;\mathbf{1}\{E_{\text{cov}}\} \\
&\le \frac{2}{N^2}\left\|\mathsf{P}^{-1/2}\mathbf{X}^\top\boldsymbol{\varepsilon}\right\|_2^2\mathbf{1}\{E_{\text{cov}}\} = \frac{2}{N^2}\left\|\sum_{i=1}^N \xi_i z_i\right\|_2^2\mathbf{1}\{E_{\text{cov}}\} \le \frac{2}{N^2}\left\|\sum_{i=1}^N \xi_i z_i\right\|_2^2.
\end{aligned}
$$

Note that $\mathsf{E}[\xi_i z_i] = \mathsf{P}^{-1/2}\mathsf{E}[\mathbf{x}_i(y_i - \langle \mathbf{w}_{\mathsf{P}}^*, \mathbf{x}_i\rangle)] = 0$. Therefore, taking expectation on the above (over $\mathcal{D}$), we get

$$
\mathsf{E}_{\mathcal{D}}\,\frac{1}{2}\|\hat{\mathbf{w}}_{\text{LS}} - \mathbf{w}_{\mathsf{P}}^*\|_{\mathsf{P}}^2 \mathbf{1}\{E_{\text{cov}}\} \le \frac{2}{N^2}\mathsf{E}\left[\left\|\sum_{i=1}^N \xi_i z_i\right\|_2^2\right] = \frac{2}{N}\mathsf{E}\left[\|\xi_1 z_1\|_2^2\right] = \frac{2}{N}\mathsf{E}\left[z_1^2\mathbf{x}_1^\top\mathsf{P}^{-1}\mathbf{x}_1\right] \tag{29}
$$

$$
\overset{(i)}{\le} \frac{2\,\|\mathsf{P}\|}{N}\mathsf{E}\left[\mathbf{x}_1^\top\mathsf{P}^{-1}\mathbf{x}_1\right] = \frac{2d\,\|\mathsf{P}\|}{N}. \tag{30}
$$

Above, (i) follows by conditioning on $\mathbf{x}_1$ and using Assumption A(5). Combining with (28), we obtain

$$
\mathsf{E}\,\frac{1}{2}(\text{clip}_R(\langle \hat{\mathbf{w}}_{\text{LS}}, \mathbf{x}_{N+1}\rangle) - y_{N+1})^2 \le L_{\mathsf{P}}(\mathbf{w}_{\mathsf{P}}^*) + O\left(R^2 + \frac{d\,\|\mathsf{P}\|}{N}\right).
$$

This proves (24).

Finally, we show $\mathsf{P}(E_{\text{cov}} \setminus E_w) = 1 - o(1) = 1 - o_d(1)$. Using (29) and $\|\mathsf{P}\| \le \mathsf{P}_{\max}\mathbf{I}_d$ by assumption, we get

$$
\mathsf{E}\left[\|\hat{\mathbf{w}}_{\text{LS}} - \mathbf{w}_{\mathsf{P}}^*\|_2^2 \mathbf{1}\{E_{\text{cov}}\}\right] \le \frac{4d\,\|\mathsf{P}\|}{N} \le \frac{4d\,\mathsf{P}_{\max}}{N}.
$$

Therefore, using an argument similar to Chebyshev's inequality,

$$\begin{aligned}
\mathrm{P}(E_{\mathrm{cov}} \setminus E_w^c) &= \mathrm{E}\big[1\{E_{\mathrm{cov}}\} \quad 1\{\|\mathbf{w}_{\mathrm{LS}}\|_2 > \tfrac{1}{20}\sqrt{\tfrac{4d^2}{N_{\min}}} + B_w^2\}\big] \\
&\quad \mathrm{E}\big[1\{E_{\mathrm{cov}}\} \quad 1\{\|\mathbf{w}_{\mathrm{LS}} \quad \mathbf{w}_{\mathrm{P}}^2\|_2 > \tfrac{1}{20}\sqrt{\tfrac{4d^2}{N_{\min}}}\}\big] \\
&\quad \mathrm{E}\left[1\{E_{\mathrm{cov}}\} \quad \frac{\|\mathbf{w}_{\mathrm{LS}} \quad \mathbf{w}_{\mathrm{P}}^2\|_2}{\frac{1}{20}\sqrt{\frac{4d^2}{N_{\min}}}}\right] \quad =20.
\end{aligned}$$

This implies that

$$\mathrm{P}(E_{\mathrm{cov}} \setminus E_w) = \mathrm{P}(E_{\mathrm{cov}}) \quad \mathrm{P}(E_{\mathrm{cov}} \setminus E_w^c) \quad 1 \quad =20 \quad =20 \quad 1 \quad =10.$$

This is the desired result. □

F.3 Proof of Corollary 5

The proof follows by first checking the well-conditionedness of the data D (cf. (5)) with high probability, then invoking Theorem 4 (for approximation least squares) and Proposition F.1 (for the statistical power of least squares).

First, as P satisfies Assumption A, by Proposition F.1, as long as $N \quad O(K^4(d + \log(1/\ )))$, we have with probability at least $1 \quad /10$ that event $E_{\mathrm{cov}} \setminus E_w$ holds. On this event, we have

$$\begin{aligned}
&\frac{1}{2}\ _{\min}\mathrm{I}_d \quad \frac{1}{2}\ _{\mathrm{P}} \quad \mathbf{b} = \mathbf{X}^\top\mathbf{X}/N \quad 2\ _{\mathrm{P}} \quad 2\ _{\max}\mathrm{I}_d, \\
&\|\mathbf{w}_{\mathrm{LS}}\|_2 \quad B_w/2 := O\Big(B_w^2 + \sqrt{\frac{d^2}{N_{\min}}}\Big),
\end{aligned}$$

and thus the dataset D is well-conditioned (in the sense of (5)) with parameters $\ = \ _{\min}/2$, $\ = 2\ _{\max}$, and B_w defined as above. Note that the condition number of $\mathbf{b}$ is upper bounded by $\ = \ = 4\ _{\max}/\ _{\min} \quad 4\ $, where $\ $ is the upper bound on the condition number of $\ _{\mathrm{P}}$ as in Assumption A(c).

Define parameters

$$\varepsilon = \sqrt{\frac{d^2}{N}}; \qquad = \frac{d^2}{B_y^2 N} \wedge 1. \tag{31}$$

Note that $B_w \quad O(B_w^2 + \sqrt{B_y^2/\ _{\min}})$ by the above choice of $\ $.

We can thus apply Theorem 4 in the unregularized case ($\ = 0$) to obtain that, there exists a transformer $\ $ with $\max_{\ell\in[L]} M^{(\ell)} \quad 3, \|\ \| \quad 4R + 4/\ _{\max}$ (with $R = \max\{B_x B_w; B_y; 1\}$), and number of layers

$$L \quad O\left(\log\frac{B_x B_w}{\varepsilon} \quad O\left(\log\left(B_x\sqrt{\frac{N}{d^2}}\left(B_w^2 + \sqrt{\frac{B_y^2}{\ _{\min}}}\right)\right)\right)\right);$$

such that on $E_{\mathrm{cov}} \setminus E_w$ (so that D is well-conditioned), we have (choosing the clipping radius in $\mathrm{read}_y(\ ) = \mathrm{clip}_{B_y}(\mathrm{read}_y(\ ))$ to be B_y):

$$\mathrm{read}_y(\mathrm{TF}^0(H)) \quad \mathrm{clip}_{B_y}(\langle \mathbf{w}_{\mathrm{LS}}; \mathbf{x}_{N+1}\rangle) \qquad \mathrm{read}_y(\mathrm{TF}^0(H)) \quad \langle \mathbf{w}_{\mathrm{LS}}; \mathbf{x}_{N+1}\rangle \quad \varepsilon = \sqrt{\frac{d^2}{N}}. \tag{32}$$

We now bound the excess risk of the above transformer. Combining Proposition F.1 and (32), we have

$$\mathrm{E} \quad \mathrm{read}_y(\mathrm{TF}^0(H)) \quad y_{N+1}\ ^2$$

$$= \mathrm{E}\quad \mathrm{read}_y(\mathrm{TF}^0(\mathrm{H}))\quad y_{N+1}\ ^2 1\{E_{\mathrm{cov}} \setminus E_w\} \;+\; \mathrm{E}\quad \mathrm{read}_y(\mathrm{TF}^0(\mathrm{H}))\quad y_{N+1}\ ^2 1\{(E_{\mathrm{cov}} \setminus E_w)^c\}$$

$$2\mathrm{E}\quad \mathrm{read}_y(\mathrm{TF}^0(\mathrm{H}))\quad \mathrm{clip}_{B_y}(\langle \hat{w}_{\mathrm{LS}}, x_{N+1}\rangle)\ ^2 1\{E_{\mathrm{cov}} \setminus E_w\}$$

$$+\ 2\mathrm{E}\quad \mathrm{clip}_{B_y}(\langle \hat{w}_{\mathrm{LS}}, x_{N+1}\rangle)\quad y_{N+1}\ ^2 1\{E_{\mathrm{cov}} \setminus E_w\} \;+\; 2B_y^2 \quad =10$$

$$\overset{(i)}{} \quad 2''^2 + L_{\mathrm{P}}(w_{\mathrm{P}}^?) + O\quad B_y^2 \;+\; \frac{d}{N}^2 \;+\; O\quad B_y^2$$

$$L_{\mathrm{P}}(w_{\mathrm{P}}^?) + O\quad B_y^2 \;+\; \frac{d}{N}^2 \qquad O\quad \frac{d}{N}^2 \;:$$

Above, (i) uses the approximation guarantee (32) as well as Proposition F.1(a) (with clipping radius B_y). This proves the desired excess risk guarantee.

Finally, under the canonical choice of parameters (23), the bounds for $L; M; \|\ \|$ simplify to

$$L \quad O\quad \log\frac{N}{\quad};\quad \max_{`\in[L]} M^{(`)} \quad 3;\quad \|\ \| \quad O(\sqrt{d}); \tag{33}$$

and the requirement for N simplifies to $N \quad O(d + \log(1=\ )) = \tilde{O}(d)$ (as $K = \ (1)$). This proves the claim about the required N and L. □

F.4 Proof of Corollary 6

Fix parameters $\ ; \_ > 0$ to be specified later and a large universal constant C_0. Let us set

$$= \max\left\{0; 1=2\quad \sqrt{d=N}^{\,2}\right\};\qquad = 25;$$

$$B_w^? := 1 + 2\sqrt{\frac{\log(4=\ )}{d}};\qquad B_w = C_0(B_w^? + \ );$$

$$B_x = C_0\sqrt{d\log(N=\ )};\qquad B_y = C_0(B_w^? + \ )\sqrt{\log(N=\ )}:$$

Consider the following good events (below $'' = ['_i]_{i\in[N]}\ 2\ \mathrm{R}^N$ is given by $'_i = y_i \quad \langle w_?; x_i\rangle$)

$$E \ = \ \left\{\|w_?\|_2 \quad B_w^?; \|''\|_2 \quad 2\sqrt{N}\right\};$$

$$E_w = \qquad {}_{\min}(X^\top X=N) \qquad {}_{\max}(X^\top X=N) \qquad ;$$

$$E_b = \{8i\ 2\ [N];\ \|x_i\|_2 \quad B_x;\ |y_i| \quad B_y\};$$

$$E_{b;N+1} = \{\|x_{N+1}\|_2 \quad B_x;\ |y_{N+1}| \quad B_y\};$$

and we define $E := E \ \setminus E_w \setminus E_b \setminus E_{b;N+1}$. Under the event E, the problem (ICRidge) is well-conditioned and $\|w_{\mathrm{ridge}}\| \quad B_w=2$ (by Lemma F.1).

Therefore, Theorem 4 implies that for $\ = \frac{+}{+}$, there exists a $L = d2\ \log(B_w=\_)e + 1$-layer transformer with prediction $\hat{y}_{N+1} := \mathrm{read}_y(\mathrm{TF}^0(\mathrm{H}))$ (clipped by B_y), such that under the good event E, we have $\hat{y}_{N+1} = \mathrm{clip}_{B_y}(\langle x_{N+1}; \hat{w}\rangle)$ and $\|\hat{w} \quad w_{\mathrm{ridge}}\| \quad \_$.

In the following, we show that is indeed the desired transformer (when $\_$ and is suitably chosen). Notice that we have

$$\mathrm{E}(\hat{y}_{N+1} \quad y_{N+1})^2 = \mathrm{E}\ 1\{E\}(\hat{y}_{N+1} \quad y_{N+1})^2 \;+\; \mathrm{E}\ 1\{E^c\}(\hat{y}_{N+1} \quad y_{N+1})^2\ ;$$

and we analyze these two parts separately.

Prediction risk under good event E. We first note that

$$\mathrm{E}\ 1\{E\}(\hat{y}_{N+1} \quad y_{N+1})^2 \ = \mathrm{E}\left[1\{E\}(\mathrm{clip}_{B_y}(\langle x_{N+1}; \hat{w}\rangle) \quad y_{N+1})^2\right]$$
$$\mathrm{E}\ 1\{E\}(\langle x_{N+1}; \hat{w}\rangle \quad y_{N+1})^2\ ;$$

where the inequality is because $y_{N+1} \in [-B_y, B_y]$ under the good event E. Notice that by our construction, under the good event E, $\widehat{w} = \widehat{w}(D)$ depends only on the dataset D[7]. Therefore, we have $\|\widehat{w}(D) - w_{\text{ridge}}(D)\| \le \varepsilon'$ as long as the event $E_0 := E \cap E_w \cap E_b$ holds for $(w_\star, D)$. Thus, under E_0,

$$\begin{aligned}
\mathbb{E}\left[\mathbf{1}\{E\}(\langle x_{N+1}, \widehat{w}\rangle - y_{N+1})^2 \mid w_\star, D\right] &= \mathbb{E}\left[\mathbf{1}\{E\}(\langle x_{N+1}, \widehat{w}(D)\rangle - y_{N+1})^2 \mid w_\star, D\right] \\
&\le \mathbb{E}\left[(\langle x_{N+1}, \widehat{w}(D)\rangle - y_{N+1})^2 \mid w_\star, D\right] \\
&= \mathbb{E}\left[(\langle x_{N+1}, \widehat{w}(D)\rangle - \langle x_{N+1}, w_\star\rangle)^2 \mid w_\star, D\right] + \sigma^2 \\
&= \|\widehat{w}(D) - w_\star\|_2^2 + \sigma^2,
\end{aligned}$$

and we also have

$$\begin{aligned}
\|\widehat{w}(D) - w_\star\|_2^2 &\le \|w_{\text{ridge}} - w_\star\|_2^2 + 2\,\|w_{\text{ridge}} - w_\star\|_2\,\|\widehat{w}(D) - w_{\text{ridge}}\|_2 + \|\widehat{w}(D) - w_{\text{ridge}}\|_2^2 \\
&\le \|w_{\text{ridge}} - w_\star\|_2^2 + 2\varepsilon'\,\|w_{\text{ridge}} - w_\star\|_2 + \varepsilon'^2.
\end{aligned}$$

Recall that $\text{BayesRisk} = \mathbb{E}_{w_\star, \mathcal{D}}\|w_{\text{ridge}} - w_\star\|_2^2 + \sigma^2$. Note that $\text{BayesRisk} \ge 1 + \sigma^2$ by definition. Therefore, we can conclude that

$$\mathbb{E}\left[\mathbf{1}\{E\}(\widehat{y}_{N+1} - y_{N+1})^2\right] \le \text{BayesRisk} + 2\varepsilon' + \varepsilon'^2.$$

Prediction risk under bad event E^c. Notice that

$$\mathbb{E}\left[\mathbf{1}\{E^c\}(\widehat{y}_{N+1} - y_{N+1})^2\right] \le \sqrt{\mathbb{P}(E^c)\,\mathbb{E}[(\widehat{y}_{N+1} - y_{N+1})^4]}.$$

We can upper bound $\mathbb{P}(E^c) = \mathbb{P}(E^c \cup E_w^c \cup E_b^c \cup E_{b;N+1}^c)$ by Lemma B.1, Lemma B.2 and the sub-Gaussian tail bound:

$$\mathbb{P}(E^c) \le \frac{1}{2} + \exp(-N/8), \qquad \mathbb{P}(E_w^c) \le 2\exp(-N/8), \qquad \mathbb{P}(E_b^c \cup E_{b;N+1}^c) \le \frac{1}{4}.$$

Thus, as long as $N \ge 8\log(12/\delta)$, we have $\mathbb{P}(E^c) \le \delta$. Further, a simple calculation yields

$$\mathbb{E}(\widehat{y}_{N+1} - y_{N+1})^4 \le 8\mathbb{E}\widehat{y}_{N+1}^4 + 8\mathbb{E}y_{N+1}^4 \le 8B_y^2 + 8\mathbb{E}y_{N+1}^4.$$

Notice that $y_{N+1} \mid w_\star \sim \mathsf{N}(0, \|w_\star\|_2^2 + \sigma^2)$, hence $\mathbb{E}y_{N+1}^4 = 3\mathbb{E}(\|w_\star\|_2^2 + \sigma^2)^2 \le 3(3 + 2\sigma^2 + \sigma^4) \le 8B_y^4$. Thus, we can conclude that

$$\mathbb{E}\left[\mathbf{1}\{E^c\}(\widehat{y}_{N+1} - y_{N+1})^2\right] \le 4\delta B_y.$$

Choosing ε' and δ. Combining the inequalities above, we have

$$\mathbb{E}(\widehat{y}_{N+1} - y_{N+1})^2 \le \text{BayesRisk} + \left[2\varepsilon' \sqrt{2\,\text{BayesRisk}} + \varepsilon'^2 + 4\delta B_y\right].$$

To ensure $\frac{1}{2}\mathbb{E}(\widehat{y}_{N+1} - y_{N+1})^2 \le \text{BayesRisk} + \varepsilon$, we only need to take (ε', δ) so that the following constraints are satisfied:

$$\varepsilon' = \frac{1}{2}\min\left\{\varepsilon, \sqrt{\varepsilon}\right\}, \qquad 4\delta B_y \le \frac{\varepsilon}{2}, \qquad N \ge 8\log(12/\delta).$$

Therefore, it suffices to take $\Sigma = \frac{c_0}{\log^2(N)}\left(\frac{\varepsilon^2}{1+\sigma^2}\right)^2$ for some small constant c_0, then as long as

$$N \ge C\log\frac{\sigma^2 + 1}{\varepsilon} + C,$$

our choice of ε' and δ is feasible. Note that $\|\Sigma\| \le O(1 + \sigma^{-2})$, and hence under such choice of (ε', δ), we have $L = O(\log(1/\varepsilon))$ and $\|\Sigma\|/\|\Sigma\| = \Theta\left(\sqrt{d}\right)$. This is the desired result. □

[7] We need this, as on E^c, the transformer output at this location could in principle depend additionally on x_{N+1}, as (15) may not hold due to the potential unbounededness of its input. A similar fact will also appear in later proofs (for generalized linear models and Lasso).

Lemma F.1. *Under the event $E \cap E_w$, we have* $\|w_{\text{ridge}}\|_2 \le O(B_w^2 + \cdot)$.

Proof of Lemma F.1. By the definition of w_{ridge} and recall that $\lambda = d\sigma^2/N$, we have $w_{\text{ridge}} = (X^\top X + d\sigma^2 I_d)^{-1} X^\top y$.

Therefore, we only need to prove the following fact: for any $\lambda > 0$ and $\hat{b} = (X^\top X + d\lambda I_d)^{-1} X^\top y$, we have

$$\|\hat{b}\|_2 \le B_w^2 + 10\sigma(1 + \lambda^{-1/2}). \tag{34}$$

We now prove (34). Note that we have

$$\|\hat{b}\|_2 = \|(X^\top X + d\lambda I_d)^{-1} X^\top (X w_\star + \varepsilon)\|_2 \le \|B_1\|_{\text{op}} \|w_\star\|_2 + \|B_2\|_{\text{op}} \|\varepsilon\|_2$$

where $B_1 = X^\top X(X^\top X + d\lambda I_d)^{-1}$, $B_2 = (X^\top X + d\lambda I_d)^{-1} X^\top$. Note that $\|B_1\|_{\text{op}} \le 1$ clearly holds, and under E we also have $\|\varepsilon\|_2 \le 2\sigma\sqrt{N}$. Therefore, it remains to bound the term $\|B_2\|_{\text{op}}$.

Consider the SVD decomposition of $X = U \Sigma V^\top$, $\Sigma = \text{diag}(\sigma_1, \ldots, \sigma_d)$, and $U \in \mathbb{R}^{N\times d}, V \in \mathbb{R}^{d\times d}$ are orthonormal matrices. Then $B_2 = V(\Sigma^2 + d\lambda I_d)^{-1} \Sigma U^\top$, and hence

$$\|B_2\|_{\text{op}} = \|(\Sigma^2 + d\lambda I_d)^{-1} \Sigma\|_{\text{op}} = \max_i \frac{\sigma_i}{\sigma_i^2 + d\lambda}.$$

When $N \ge 36d$, we directly have $\|B_2\|_{\text{op}} \le \frac{1}{\sigma_{\min}(X^\top X)} \le \frac{2}{3(N-d)} \le \sqrt{3/N}$. Otherwise, we have $N < 36d$, and then for each $i \in [d]$, $\sigma_i \le \sqrt{\frac{1}{2}(d-N)\lambda}$ $\le \sqrt{3(N-d)\lambda/N} \le \sqrt{3\lambda}$. Hence, in this case we also have $\|B_2\|_{\text{op}} \le \max_i \sigma_i^{-1} \le 3N^{-1/2}$. Combining the both cases completes the proof of (34). □

G In-context learning of generalized linear models

As a natural generalization of linear regression, we now show that transformers can recover learn generalized linear models (GLMs) [53] (which includes logistic regression for linear classification as an important special case), by implementing the corresponding convex risk minimization algorithm in context, and achieve near-optimal excess risk under standard statistical assumptions.

Let $g : \mathbb{R} \to \mathbb{R}$ be a link function that is non-decreasing and C^2-smooth. We consider the following convex empirical risk minimization (ERM) problem

$$w_{\text{GLM}} := \arg\min_{w\in\mathbb{R}^d} \hat{L}_N(w) := \frac{1}{N}\sum_{i=1}^N \ell(\langle x_i, w\rangle; y_i); \tag{ICGLM}$$

where $\ell(t; y) := -yt + \int_0^t g(s)ds$ is the convex (integral) loss associated with g. A canonical example of (ICGLM) is logistic regression, in which $g(t) = \sigma_{\log}(t) := (1 + e^{-t})^{-1}$ is the sigmoid function, and the resulting $\ell(t; y) = \ell_{\log}(t; y) = -yt + \log(1 + e^t)$ is the logistic loss.

The following result (proof in Appendix G.1) shows that, as long as the empirical risk $\hat{L}_N$ satisfies strong convexity and bounded solution conditions (similar as in Theorem 4), transformers can approximately implement the ERM predictor $g(\langle x_{N+1}, w_{\text{GLM}}\rangle)$, with w_{GLM} given by (ICGLM).

Theorem G.1 (Implementing convex risk minimization for GLMs)**.** *For any $0 < \alpha < \beta$ with $\beta := \alpha$, $B_w > 0, B_x > 0$, $\sigma_w := L_g B_x^2/\lambda + 1$ and $\varepsilon < B_w/2$, there exists an attention-only transformer TF^0 with*

$$L = \Theta\big(\log(L_g B_w B_x/\varepsilon) + 1\big), \quad \max_{\ell\in[L]} M^{(\ell)} \le C_g^2 \sigma_w^{2L+2}, \quad \|\theta\| \le R^{L+1} C_g,$$

(where $L_g := \sup_t |g'(t)|$, $R := \max\{B_x B_w, B_y, 1\}$, and $C_g > 0$ is a constant that depends only on R and the C^2-smoothness of g within $[-R, R]$), such that the following holds. On any input data (D, x_{N+1}) such that

$$\lambda_{\min}(\nabla^2 \hat{L}_N(w)) \ge \alpha, \quad \lambda_{\max}(\nabla^2 \hat{L}_N(w)) \le \beta \quad \text{for all } w \in B_2(B_w), \quad \|w_{\text{GLM}}\|_2 \le B_w/2, \tag{35}$$

$\mathrm{TF}^0(H^{(0)})$ *approximately implements (ICGLM): We have* $h_{N+1}^{(L+1)} := [x_{N+1}; \hat{y}_{N+1}; w; 1; 1]$, *where*

$$|\hat{y}_{N+1} - g(\langle x_{N+1}, w_{\mathrm{GLM}}\rangle)| \le \varepsilon.$$

In Theorem G.1, the number of heads scales as $\widetilde{O}(1/\varepsilon^2)$ as opposed to $O(1)$ as in ridge regression (Theorem 4), due to the fact that the gradient of the loss is in general a smooth function that can be only *approximately* expressed as a sum-of-relus (cf. Definition D.1 & Lemma B.5) rather than exactly expressed as in the case for the square loss.

In-context prediction power We next show that (proof in Appendix G.2) the transformer constructed in Theorem G.1 achieves desirable statistical power if the in-context data distribution satisfies standard statistical assumptions for learning GLMs. Let $L_P(w) := \mathbb{E}_{(x,y)\sim P}[\ell(\langle w, x\rangle, y)]$ denote the corresponding population risk for any distribution P of (x, y). When P is *realizable* by a generalized linear model of link function g and parameter θ in the sense that $\mathbb{E}_P[y|x] = g(\langle \theta, x\rangle)$, it is a standard result that θ is indeed a minimizer of L_P [42] (see also [6, Appendix A.3]).

Theorem G.2 (Statistical guarantee for generalized linear models)**.** *For any fixed set of parameters defined in Assumption B, there exists a transformer with* $L = O(\log(N))$ *layers and* $\max_{\ell\in[L]} M^{(\ell)} \le \widetilde{O}(d^3 N)$, *such that for any distribution* P *satisfying Assumption B with those parameters, as long as* $N \ge O(d)$, *that outputs* $\hat{y}_{N+1} = \mathrm{read}_y(\mathrm{TF}(H))$ *and* $w = \mathrm{read}_w(\mathrm{TF}(H)) \in \mathbb{R}^d$ *(for another read-out function* read_w*) satisfying the following.*

(a) w *achieves small excess risk under the population loss, i.e. for the linear prediction* $\hat{y}^{\mathrm{lin}}_{N+1} := \langle x_{N+1}, w\rangle$,

$$\mathbb{E}_{(\mathcal{D};x_{N+1};y_{N+1})\sim P}\ \ell(\hat{y}^{\mathrm{lin}}_{N+1}; y_{N+1}) \le \min_{\theta} L_P(\theta) + O(d/N). \tag{36}$$

(b) (Realizable setting) If there exists a $\theta \in \mathbb{R}^d$ *such that under* P, $\mathbb{E}[y|x] = g(\langle \theta, x\rangle)$ *almost surely, then*

$$\mathbb{E}_{(\mathcal{D};x_{N+1};y_{N+1})\sim P}\ (\hat{y}_{N+1} - y_{N+1})^2 \le \mathbb{E}_{(x_{N+1};y_{N+1})\sim P}\ (g(\langle \theta, x_{N+1}\rangle) - y_{N+1})^2 + O(d/N); \tag{37}$$

or equivalently, $\mathbb{E}[(\hat{y}_{N+1} - \mathbb{E}[y_{N+1}|x_{N+1}])^2] \le O(d/N)$.

Above, $O(\cdot)$ hides constants that depend polynomially on the parameters in Assumption B. Similar as in Corollary 5, the $O(d/N)$ excess risk obtained here matches the optimal (fast) rate for typical learning problems with d parameters and N samples [87].

Assumption B (Well-posedness for learning GLMs)**.** *We assume that there is some* $B > 0$ *such that for any* $t \in [-B, B]$, $g'(t) \ge \gamma_g > 0$.

We also assume that for each $i \in [N+1]$, $(x_i; y_i)$ *is independently sampled from* P *such that the following holds.*

(a) Under the law $(x, y) \sim P$, *We have* $x \sim \mathrm{SG}(K_x)$, $y \sim \mathrm{SG}(K_y)$ *and* $g(\langle w, x\rangle) \sim \mathrm{SG}(K_y)$ $\forall w \in B_2(B_w)$.

(b) For some $\gamma_x > 0$, *it holds that*

$$\mathbb{E}[1\{|x^\top w| \le B\} \cdot xx^\top] \succeq \gamma_x I_d \quad \forall w \in B_2(B_w).$$

(c) For $\theta^\star = \arg\min L_P$, *it holds* $\|\theta^\star\|_2 \le B_w/4$.

Applying Theorem G.2 to logistic regression, we have the following result as a direct corollary. Below, the Gaussian input assumption is for convenience only and can be generalized to e.g. sub-Gaussian input.

Corollary G.1 (In-context logistic regression)**.** *Consider any in-context data distribution* P *satisfying*

$$x \sim N(0; I_d), \quad y \in \{0, 1\}, \quad \arg\min_{\theta\in\mathbb{R}^d} L_P(\theta) \in B_2(B_w^\star).$$

For the link function $g = \sigma_{\log}$ *and* $B_w^\star = O(1)$, *we can choose* $B_w, B, \gamma_g, L_g, \gamma_x, K_x, K_y = O(1)$ *so that Assumption B holds. In that case, when* $N \ge O(d)$, *there exists a transformer with* $L = O(\log(N))$ *layers, such that for any* P *considered above,*

(a) The estimation $\hat{w} = \mathrm{read}_w(\mathrm{TF}(H))$ *outputted by* *achieves excess risk bound (36).*

(b) (Realizable setting) Consider the logistic in-context data distribution

$$P^{\log}: \quad x \quad N(0; I_d); \qquad y|x \quad \text{Bernoulli}(g(\langle \beta, x\rangle)).$$

Then, for any distribution $P = P^{\log}$ *with* $\|\beta\|_2 \le B_w^?$, *the prediction* $\hat{y}_{N+1} = \mathrm{read}_y(\mathrm{TF}(H))$ *of* *additionally achieves the square loss excess risk (37).*

G.1 Proof of Theorem G.1

Let us fix parameters $\varepsilon_g > 0$ and $T > 0$ (that we specify later in proof).

Define $R = \max\{B_x B_w, B_y, 1\}$ and

$$C_g := \max_{i=0,1,2} R^i \max_{s\in[-B,B]} |g^{(i)}(s)| .$$

By Proposition B.1, g is (ε_g, M, R, C) with

$$C \le O(C_g), \qquad M \le O\left(C_g^2 \varepsilon_g^{-2} \log(1 + C_g \varepsilon_g^{-1})\right).$$

Therefore, we can invoke Theorem D.1 to obtain that, as long as $2T\varepsilon_g \le B_w$, there exists a T-layer attention-only transformer $^{(1:T)}$ with M heads per layer, such that for any input H of format (3) and satisfies (35), its last layer outputs $h_i^{(T)} = [x_i, y_i', \hat{w}^T, 0_{D-2d-3}, 1, t_i]$, such that

$$\|\hat{w}^T - w_{\mathrm{GD}}^T\|_2 \le \varepsilon_g (L^{-1} B_x),$$

where $\{w_{\mathrm{GD}}^\ell\}_{\ell\in[L]}$ is the sequence of gradient descent iterates with stepsize $^{-1}$ and initialization $w_{\mathrm{GD}}^0 = 0$. Notice that Proposition B.2 implies (with $:= = $)

$$\|w_{\mathrm{GD}}^T - w_{\mathrm{GLM}}\|_2 \le \exp(-T/(2\ )) \|w_{\mathrm{GLM}}\|_2 \le \exp(-T/(2\ )) \frac{B_w}{2} := \varepsilon_o.$$

Furthermore, we can show that (similar to the proof of Theorem D.1 (b)), there exists a single attention layer $^{(T+1)}$ with M heads such that it outputs $h_{N+1}^{(T+1)} = [x_{N+1}, \hat{y}_{N+1}, \hat{w}^T, 0_{D-2d-3}, 1, 0]$, where $\hat{y}_{N+1} \quad g(\langle x_{N+1}, \hat{w}^T\rangle) \quad \varepsilon_g$.

In the following, we show that for suitably chosen (T, ε_g), $= (^{(1:T)}, ^{(T+1)})$ is the desired transformer. First notice that its output $h_{N+1}^{(T+1)} = [x_{N+1}, \hat{y}_{N+1}, \hat{w}^T, 0_{D-2d-3}, 1, 0]$ satisfies

$$\begin{aligned}
|\hat{y}_{N+1} - g(\langle x_{N+1}, w_{\mathrm{GLM}}\rangle)| &\le |\hat{y}_{N+1} - g(\langle x_{N+1}, \hat{w}^T\rangle)| + |g(\langle x_{N+1}, \hat{w}^T\rangle) - g(\langle x_{N+1}, w_{\mathrm{GLM}}\rangle)| \\
&\le \varepsilon_g + L_g B_x \|\hat{w}^T - w_{\mathrm{GD}}^T\|_2 + L_g B_x \|w_{\mathrm{GD}}^T - w_{\mathrm{GLM}}\|_2 \\
&\le \varepsilon_g (1 + L_g B_x (L^{-1} B_x)) + L_g B_x \varepsilon_o.
\end{aligned}$$

Therefore, for any fixed $\varepsilon > 0$, we can take

$$T = \Theta\left(\log(L_g B_x B_w/\varepsilon)\right), \qquad \varepsilon_g = \frac{1}{2} \frac{\varepsilon}{1 + L^{-1}(L_g B_x^2)},$$

so that the we construct above ensures $|\hat{y}_{N+1} - g(\langle x_{N+1}, w_{\mathrm{GLM}}\rangle)| \le \varepsilon$ for any input H that satisfies (35). The upper bound on $\|\|$ follows immediately from Theorem D.1. □

G.2 Proof of Theorem G.2

We summarize some basic and useful facts about GLM in the following theorem. Its proof is presented in Appendix G.3 - G.6.

Theorem G.3. *Under Assumption B, the following statements hold with universal constant* C_0 *and constant* C_1, C_2 *that depend only on the parameters* $(K_x, K_y, B\ , B_w, \ _x, L_g, \ _g)$.

(a) As long as $N \geq C_1 d$*, the following event happens with probability at least* $1 - 2e^{-N/C_1}$*:*

$$E_w : \frac{1}{8} \gamma \kappa \leq \min(r^2 \hat{L}_N(w)) \leq \max(r^2 \hat{L}_N(w)) \leq 8 L_g K_x^2, \quad \forall w \in B_2(B_w).$$

(b) For any $\delta > 0$*, we have with probability at least* $1 - \delta$ *that*

$$\varepsilon_{\text{stat}} := \sup_{w \in B_2(B_w)} \left| r_w \hat{L}_N(w) - r_w \mathbb{E}[\hat{L}_N(w)] \right|_2 \leq C_0 K_x K_y \max\left(\sqrt{\frac{d + \log(1/\delta)}{N}}, \frac{d + \log(1/\delta)}{N} \right),$$

where we denote $\iota = \log(2 + L_g K_x^2 B_w / K_y)$.

(c) Condition on (a) holds and $N \geq C_2 d$*, the event* $E_r := \{\|w_{\text{GLM}}\|_2 \leq B_w/2\}$ *happens with probability at least* $1 - e^{-N/C_2}$.

(d) For any $w \in B_2(B_w)$*, it holds that*

$$L_p(w) - L_p(\theta) \leq \frac{4}{\gamma \kappa} \varepsilon_{\text{stat}}^2 + \left| r \hat{L}_N(w) \right|_2^2.$$

(e) (Realizable setting) As long as $w_{\text{GLM}} \in B_2(B_w)$*, it holds that*

$$\mathbb{E}_x(g(\langle x, w_{\text{GLM}} \rangle) - g(\langle x, \theta \rangle))^2 \leq \frac{L_g}{\gamma \kappa} \varepsilon_{\text{stat}}^2.$$

Therefore, we can set

$$\varepsilon = \frac{\gamma \kappa}{8}, \qquad \varepsilon_{\text{stat}} = 8 L_g K_x^2,$$
$$B_x = C_0 K_x \sqrt{d \log(N/\delta)}, \qquad B_y = C_0 K_y \sqrt{\log(N/\delta)}.$$

Consider the following good events

$$E_b = \{\forall i \in [N],\ \|x_i\|_2 \leq B_x,\ |y_i| \leq B_y\},$$
$$E_{b;N+1} = \{\|x_{N+1}\|_2 \leq B_x,\ |y_{N+1}| \leq B_y\},$$
$$E = E_r \cap E_w \cap E_b \cap E_{b;N+1}.$$

Under the event E and our choice of ε, $\varepsilon_{\text{stat}}$, the problem (ICGLM) is well-conditioned (i.e. (35) holds).

Theorem G.1 implies that there exists a transformer TF such that for any input H of the form (3), TF outputs $h'_{N+1} = [x_{N+1}; \hat{y}_{N+1}; w; 0_{D-2d-3}; 1; 0]$, such that the output is given by $\hat{y}_{N+1} = \text{read}_y(\mathsf{TF}(H)) = \text{clip}_{B_y}(\hat{y}_{N+1})$ and $\hat{w} = \text{read}_w(\mathsf{TF}(H)) := \text{Proj}_{B_2(B_w)}(w)$, and the following holds on the good event E:

(a) $\hat{y}_{N+1} = f_{\mathcal{D}}(x_{N+1})$, where $f_{\mathcal{D}} = A(D)$ is a predictor such that $|f_{\mathcal{D}}(x) - g(\langle x, w_{\text{GLM}} \rangle)| \leq \varepsilon$ for all $x \in B_2(B_x)$.

(b) $w = w(D) \in B_2(B_w)$ depends only on D (by the proof of Theorem G.1 and Theorem D.1), such that $\left| r \hat{L}_N(w) \right|_2 \leq \frac{\varepsilon}{L_g B_w}$.

In the following, we show that TF constructed above fulfills both (a) & (b) of Theorem G.2. The bounds on number of layers and heads and $\|\mathsf{TF}\|$ follows from plugging our choice of B_x, B_y in our proof of Theorem G.1.

Proof of Theorem G.2 (a). Notice that under the good event E, we have $\hat{w} = w = w(D)$ depends only on D. Then we have

$$\begin{aligned}
&\mathbb{E}_{(\mathcal{D}; x_{N+1}, y_{N+1})} \ell(\hat{y}^{\text{lin}}_{N+1}, y_{N+1}) \\
&= \mathbb{E}_{(\mathcal{D}; x_{N+1}, y_{N+1})} \mathbf{1}\{E\} \ell(\hat{y}^{\text{lin}}_{N+1}, y_{N+1}) + \mathbb{E}_{(\mathcal{D}; x_{N+1}, y_{N+1})} \mathbf{1}\{E^c\} \ell(\hat{y}^{\text{lin}}_{N+1}, y_{N+1}) \\
&= \mathbb{E}_{(\mathcal{D}; x_{N+1}, y_{N+1})} [\mathbf{1}\{E\} \ell(\langle x_{N+1}, w(D) \rangle, y_{N+1})] + \mathbb{E}_{(\mathcal{D}; x_{N+1}, y_{N+1})} \mathbf{1}\{E^c\} \ell(\hat{y}^{\text{lin}}_{N+1}, y_{N+1}).
\end{aligned}$$

Thus, we can consider $E_0 = E_r \cap E_w \cap E_b$, and then

$$\mathbb{E}_{(\mathcal{D}; x_{N+1}, y_{N+1})} [\mathbf{1}\{E\} \ell(\langle x_{N+1}, w(D) \rangle, y_{N+1})]$$

$$= \mathsf{E}_{(\mathcal{D};\mathbf{x}_{N+1};y_{N+1})}[1\{E_0\}g(\ell(\hat{f}(\mathbf{x}_{N+1};\hat{\mathbf{w}}(D)),y_{N+1})] \quad \mathsf{E}_{(\mathcal{D};\mathbf{x}_{N+1};y_{N+1})}[1\{E_0 \quad E\}g(\ell(\hat{f}(\mathbf{x}_{N+1};\hat{\mathbf{w}}(D)),y_{N+1})]$$
$$= \mathsf{E}_{(\mathcal{D};\mathbf{x}_{N+1};y_{N+1})}[1\{E_0\}gL_p(\hat{\mathbf{w}}(D))] \quad \mathsf{E}_{(\mathcal{D};\mathbf{x}_{N+1};y_{N+1})}[1\{E_0 \quad E\}g(\ell(\hat{f}(\mathbf{x}_{N+1};\hat{\mathbf{w}}(D)),y_{N+1})];$$

where the second equality follows from $L_p(\hat{\mathbf{w}}(D)) = \mathsf{E}_{(\mathbf{x}_{N+1};y_{N+1})|\mathcal{D}}(\ell(\hat{f}(\mathbf{x}_{N+1};\hat{\mathbf{w}}(D)),y_{N+1})$. Therefore,

$$\mathsf{E}_{(\mathcal{D};\mathbf{x}_{N+1};y_{N+1})}(\ell(\hat{y}^{\text{lin}}_{N+1};y_{N+1}) \quad \mathsf{E}_{\mathcal{D}}[1\{E_0\}gL_p(\hat{\mathbf{w}}(D))]$$
$$= \mathsf{E}_{(\mathcal{D};\mathbf{x}_{N+1};y_{N+1})}[1\{E^c\}g(\ell(\hat{y}^{\text{lin}}_{N+1};y_{N+1}) \quad \mathsf{E}_{(\mathcal{D};\mathbf{x}_{N+1};y_{N+1})}[1\{E_0 \quad E\}g(\ell(\hat{f}(\mathbf{x}_{N+1};\hat{\mathbf{w}}(D)),y_{N+1})]$$
$$2\sqrt{\mathsf{P}(E^c)\max\left\{\mathsf{E}(\ell(\hat{y}^{\text{lin}}_{N+1};y_{N+1})^4\,;\mathsf{E}[(\ell(\hat{f}(\mathbf{x}_{N+1};\hat{\mathbf{w}}(D)),y_{N+1})^4]\right\}} = O\left(\frac{B_\ell^2}{N^5}\right);$$

where the last line follows from Cauchy inequality and the fact $\mathsf{P}(E^c) = O(N^{-10})$, and B_ℓ is defined in Lemma G.1.

Notice that by Theorem G.3 (d), we have

$$\mathsf{E}_{\mathcal{D}}[1\{E_0\}g(L_p(\hat{\mathbf{w}}) \quad \inf L_p)] \quad \frac{4}{g_x}\left(\mathsf{E}[\varepsilon^2_{\text{stat}}] + \mathsf{E}\left[1\{E_0\}g\left(r\hat{L}_N(\hat{\mathbf{w}})\right)^2\right]\right);$$

and by Theorem G.3 (b) and taking integration over > 0, we have

$$\mathsf{E}[\varepsilon^2_{\text{stat}}] \quad O(1)\left(K_x^2K_y^2\left(\frac{d}{N} + \frac{d}{N}\right)^2\right):$$

Also, we have $\inf L_p = L_p(\ ^?) \quad B_\ell$ by Lemma G.1. Therefore, we can conclude that

$$\mathsf{E}_{(\mathcal{D};\mathbf{x}_{N+1};y_{N+1})}(\ell(\hat{y}^{\text{lin}}_{N+1};y_{N+1}) \quad \inf L_p + O(1)\left(\frac{K_x^2K_y^2}{g_x}\frac{d}{N} + \frac{K_x^4}{g_xB_w}\varepsilon^2 + \frac{B_\ell^2}{N^5}\right):$$

Taking $\varepsilon^2 \quad \frac{K_y^2}{B_wK_x^2}\frac{d}{N}$ completes the proof. □

Proof of Theorem G.2 (b). Similar to the proof of Corollary 6, we have

$$\mathsf{E}(\hat{y}_{N+1} \quad y_{N+1})^2 = \mathsf{E}\,1\{E\}g(\hat{y}_{N+1} \quad y_{N+1})^2 + \mathsf{E}\,1\{E^c\}g(\hat{y}_{N+1} \quad y_{N+1})^2$$
$$\mathsf{E}\,1\{E\}g(\hat{e}_{N+1} \quad y_{N+1})^2 + \sqrt{\mathsf{P}(E^c)\mathsf{E}(\hat{y}_{N+1} \quad y_{N+1})^4};$$

where the inequality follows from $y_{N+1}\ 2\ [\ B_y; B_y]$ on event E. For the first part, we have

$$\mathsf{E}\left[1\{E\}g(\hat{e}_{N+1} \quad y_{N+1})^2\right] = \mathsf{E}\left[1\{E\}g(f_{\mathcal{D}}(\mathbf{x}_{N+1}) \quad y_{N+1})^2\right]$$
$$\mathsf{E}_{\mathcal{D}}\left[1\{E_0\}g\left[\mathsf{E}_{(\mathbf{x};y)\sim\mathsf{P}}\left[1\{|x|_2 \quad B_x\}g(f_{\mathcal{D}}(\mathbf{x}) \quad y)^2\right]\right]\right];$$

where we use the fact that the conditional distribution of $(\mathbf{x}_{N+1};y_{N+1})|D$ agrees with P. Thus,

$$\mathsf{E}\left[1\{E\}g(\hat{e}_{N+1} \quad y_{N+1})^2\right] \quad \mathsf{E}_{(\mathbf{x};y)\sim\mathsf{P}}(g(h(\mathbf{x}; \ )) \quad y)^2$$
$$\mathsf{E}_{\mathcal{D}}\left[1\{E_0\}g \quad \mathsf{E}_{(\mathbf{x};y)\sim\mathsf{P}}1\{|x|_2 \quad B_x\}g(f_{\mathcal{D}}(\mathbf{x}) \quad y)^2 \quad \mathsf{E}_{(\mathbf{x};y)\sim\mathsf{P}}(g(h(\mathbf{x}; \ )) \quad y)^2\right]$$
$$\mathsf{E}_{\mathcal{D}}\left[1\{E_0\}g \quad \mathsf{E}_{\mathbf{x}}1\{|x|_2 \quad B_x\}g(f_{\mathcal{D}}(\mathbf{x}) \quad g(h(\mathbf{x}; \ ))^2\right]$$
$$2\mathsf{E}_{\mathcal{D}}\left[1\{E_0\}g\left[\mathsf{E}_{\mathbf{x}}1\{|x|_2 \quad B_x\}g(f_{\mathcal{D}}(\mathbf{x}) \quad g(h(\mathbf{x};\hat{\mathbf{w}}_{GLM}))^2\right] + 2\mathsf{E}_{\mathcal{D}}\left[1\{E_0\}g\left[\mathsf{E}_{\mathbf{x}}(g(h(\mathbf{x};\hat{\mathbf{w}}_{GLM}) \quad g(h(\mathbf{x}; \ ))^2\right]\right.\right.$$
$$2\varepsilon^2 + \frac{2L_g}{x_g}\mathsf{E}[\varepsilon^2_{\text{stat}}] \quad 2\varepsilon^2 + O(1)\frac{L_gK_x^2K_y^2}{x_g}\frac{d}{N}:$$

For the second part, we know $\mathsf{P}(E^c) = O(N^{-10})$ and

$$\mathsf{E}(\hat{y}_{N+1} \quad y_{N+1})^4 \quad 8\mathsf{E}\hat{y}^2_{N+1} + 8\mathsf{E}y^4_{N+1} = O(B_y^4):$$

In conclusion, we have

$$\mathsf{E}(\hat{y}_{N+1} \quad y_{N+1})^2 \quad \mathsf{E}(y_{N+1} \quad g(h(\mathbf{x}_{N+1}; \ )))^2 + 2\varepsilon^2 + O(1)\frac{L_gK_x^2K_y^2}{x_g}\frac{d}{N} + O\left(\frac{B_y^2}{N^5}\right):$$

Taking $\varepsilon^2 \quad \frac{L_gK_x^2K_y^2}{x_g}\frac{d}{N}$ completes the proof. □

Lemma G.1. *Suppose that* $\mathsf{x} \sim \mathrm{SG}(K_x)$, $y \sim \mathrm{SG}(K_y)$, *and* w *is a (possibly random) vector such that* $\|\mathsf{w}\|_2 \le B_w$. *Then*

$$\mathbb{E}\,\grave{}(h\mathsf{x}, \mathsf{w}i, y)^4)^{1/4} \le O\big(L_g K_x^2 B_w^2 d + K_x K_y B_w d\big) =: B\grave{}.$$

Proof. Notice that by our assumption, $|g(0)| \le 2K_y$. Therefore, by the definition of $\grave{}$,

$$|\grave{}(t, y)| = \Big|yt + \int_0^t g(s)ds\Big| \le |t(g(0) - y)| + \Big|\int_0^t (g(s) - g(0))ds\Big| \le |t|(2K_y + |y|) + 2L_g t^2.$$

The proof is then done by bounding the moment by $\mathbb{E}|y|^8 \le O(K_y^8)$ and $\mathbb{E}|h\mathsf{x}, \mathsf{w}i|^8 \le B_w^8 \mathbb{E}\|\mathsf{x}\|_2^8 \le O\big((\sqrt{d}B_w K_x)^8\big)$, which is standard (by utilizing the tail bound of sub-Gaussian/sub-Exponential random variable). □

G.3 Proof of Theorem G.3 (a)

We begin with the upper bound on $\lambda_{\max}(\nabla^2 \mathcal{L}_N(\mathsf{w}))$. By Lemma B.3, as long as $N \ge C_0 d$, the following event

$$E_{w;0} : \quad \Big\|\frac{1}{N}\sum_{i=1}^N \mathsf{x}_i \mathsf{x}_i^\top\Big\|_{\mathrm{op}} \le 8K^2.$$

happens with probability at least $1 - \exp(-N/C_0)$. By the assumption that $\sup|g'| \le L_g$, it is clear that when $E_{w;0}$ holds, we have $\lambda_{\max}(\nabla^2 \mathcal{L}_N(\mathsf{w})) \le 8L_g K_x^2$ $\forall \mathsf{w} \in \mathbb{R}^d$.

In the following, we analyze the quantity $\lambda_{\max}(\nabla^2 \mathcal{L}_N(\mathsf{w}))$. We have to invoke the following covering argument (see e.g. [85, Section 4.1.1]).

Lemma G.2. *Suppose that* $\mathcal{V}$ *is a* ε*-covering of* $\mathbb{S}^{d-1}$ *with* $\varepsilon \in [0, 1)$. *Then the following holds:*

1. *For any* $d \times d$ *symmetric matrix* A, $\|A\|_{\mathrm{op}} \le \frac{1}{1-2\varepsilon} \max_{\mathbf{v}\in\mathcal{V}} \mathbf{v}^\top A \mathbf{v}$ *and*
$$\lambda_{\min}(A) \ge \min_{\mathbf{v}\in\mathcal{V}} \mathbf{v}^\top A \mathbf{v} - 2\varepsilon\|A\|_{\mathrm{op}}$$
2. *For any vector* $\mathsf{x} \in \mathbb{R}^d$, $\|\mathsf{x}\|_2 \le \frac{1}{1-\varepsilon} \max_{\mathbf{v}\in\mathcal{V}} |h\mathbf{v}, \mathsf{x}i|$.

Notice that

$$\nabla^2 \mathcal{L}_N(\mathsf{w}) = \frac{1}{N}\sum_{i=1}^N g'(h\mathsf{w}, \mathsf{x}_i i)\mathsf{x}_i \mathsf{x}_i^\top \le \frac{1}{N}\sum_{i=1}^N {}_g\mathsf{I}(|h\mathsf{w}, \mathsf{x}_i i| \le B)\mathsf{x}_i \mathsf{x}_i^\top + \frac{1}{N}\sum_{i=1}^N \Big(g\Big(1 - \frac{|h\mathsf{w}, \mathsf{x}_i i|}{B}\Big)_+ \mathsf{x}_i \mathsf{x}_i^\top\Big).$$

Therefore, we can define $h(t) := (B - |t|)_+$ (which is a 1-Lipschitz function), and we have

$$\nabla^2 \mathcal{L}_N(\mathsf{w}) \le \frac{g}{B}\underbrace{\frac{1}{N}\sum_{i=1}^N h(h\mathsf{w}, \mathsf{x}_i i)\mathsf{x}_i \mathsf{x}_i^\top}_{=:A(\mathbf{w})}.$$

In the following, we pick a $\varepsilon_{\mathbf{v}}$-covering $\mathcal{V}$ of $\mathbb{S}^{d-1}$ such that $|\mathcal{V}| \le (3/\varepsilon_{\mathbf{v}})^d$ (we will specify $\varepsilon_{\mathbf{v}}$ later in proof). Then for any $\mathsf{w} \in \mathsf{B}_2(B_w)$,

$$\lambda_{\min}(A(\mathsf{w})) \ge \min_{\mathbf{v}\in\mathcal{V}} \mathbf{v}^\top A(\mathsf{w})\mathbf{v} - 2\varepsilon_{\mathbf{v}}\|A(\mathsf{w})\|_{\mathrm{op}}$$

By our definition of $A(\mathsf{w})$, we have (for any fixed B_{xv})

$$\min_{\mathbf{v}\in\mathcal{V}} \mathbf{v}^\top A(\mathsf{w})\mathbf{v} = \min_{\mathbf{v}\in\mathcal{V}} \frac{1}{N}\sum_{i=1}^N h(h\mathsf{w}, \mathsf{x}_i i)\, h\mathbf{v}, \mathsf{x}_i i^2$$

$$\min_{\mathbf{v}\in\mathcal{V}} \frac{1}{N}\sum_{i=1}^{N} h(\langle \mathbf{w}, \mathbf{x}_i\rangle) \min\left\{ \langle \mathbf{v}, \mathbf{x}_i\rangle^2, B_{xv}^2 \right\} \Big|_{=:U_{\mathbf{v}}(\mathbf{w})}$$

$$\min_{\mathbf{v}\in\mathcal{V}} \mathbb{E}[U_{\mathbf{v}}(\mathbf{w})] + \min_{\mathbf{v}\in\mathcal{V}} (U_{\mathbf{v}}(\mathbf{w}) - \mathbb{E}[U_{\mathbf{v}}(\mathbf{w})]).$$

By Lemma G.3, we can choose $B_{xv} = K_x(15 + \log(K_x^2/\varepsilon_x))$, and then $\mathbb{E}[U_{\mathbf{v}}(\mathbf{w})] \le 3B_{xv}\varepsilon_x/8$. Thus, combining the inequalities above, we can take $\varepsilon_{\mathbf{v}} = \frac{128K_x^2}{\varepsilon_x}$ in the following, so that under event $E_{w,0}$,

$$\min_{\mathbf{v}\in\mathcal{V}}(\hat{r}^2 \hat{p}_N(\mathbf{w})) \le \frac{\varepsilon_x}{8} + \frac{\varepsilon_x}{B_{xv}} \frac{B_{xv}}{16} \le \max_{\mathbf{v}\in\mathcal{V}} (\mathbb{E}[U_{\mathbf{v}}(\mathbf{w})] - U_{\mathbf{v}}(\mathbf{w})) + \frac{\varepsilon_x}{16}.$$

In the following, we consider the random process $\bar{U}_{\mathbf{v}}(\mathbf{w}) := U_{\mathbf{v}}(\mathbf{w}) - \mathbb{E}[U_{\mathbf{v}}(\mathbf{w})]$, which is zero-mean and indexed by $\mathbf{w} \in \mathrm{B}_2(B_w)$. For any fixed $\mathbf{v}$, consider applying Proposition B.4 to the random process $\{\bar{U}_{\mathbf{v}}(\mathbf{w})\}_{\mathbf{w}}$. We need to verify the preconditions:

(a) With norm $\rho(\mathbf{w}, \mathbf{w}') = \|\mathbf{w} - \mathbf{w}'\|_2$, $\log N(\mathrm{B}(\mathbf{w}, r), \varepsilon) \le d\log(2Ar/\varepsilon)$ with constant $A = 2$;

(b) Let $f(\mathbf{x}; \mathbf{w}) := h(\langle \mathbf{w}, \mathbf{x}_i\rangle) \min\left\{ \langle \mathbf{v}, \mathbf{x}_i\rangle^2, B_{xv}^2 \right\}$, then $|f(\mathbf{x}; \mathbf{w})| \le B_{xv}^2$ and hence in $\mathrm{SG}(CB_{xv}^2)$ for any random $\mathbf{x}$;

(c) For $\mathbf{w}, \mathbf{w}' \in W$, we have $|h(\langle \mathbf{w}, \mathbf{x}_i\rangle) - h(\langle \mathbf{w}', \mathbf{x}_i\rangle)| \le |\langle \mathbf{w} - \mathbf{w}', \mathbf{x}_i\rangle|$. Hence, because $\mathbf{x} \in \mathrm{SG}(K_x)$, the random variable $h(\langle \mathbf{w}, \mathbf{x}\rangle) - h(\langle \mathbf{w}', \mathbf{x}\rangle)$ is $\mathrm{SG}(CK_x\|\mathbf{w} - \mathbf{w}'\|_2)$, and the random variable $f(\mathbf{x}; \mathbf{w}) - f(\mathbf{x}; \mathbf{w}')$ is $\mathrm{SG}(CK_xB_{xv}^2\|\mathbf{w} - \mathbf{w}'\|_2)$.

Therefore, we can apply Proposition B.4 to obtain that with probability $1 - \varepsilon_0$, it holds

$$\sup_{\mathbf{w}} \bar{U}_{\mathbf{v}}(\mathbf{w}) \le C'B_{xv}^2 \sqrt{\frac{d\log(2\varepsilon_g) + \log(1/\varepsilon_0)}{N}},$$

where we denote $\varepsilon_g = 1 + K_xB_w/B$. Setting $\varepsilon_0 = \varepsilon/|\mathcal{V}|$ and taking the union bound over $\mathbf{v} \in V$, we obtain that with probability at least $1 - \varepsilon$,

$$\max_{\mathbf{v}\in\mathcal{V}} \sup_{\|\mathbf{w}\|_2 \le B_w} \bar{U}_{\mathbf{v}}(\mathbf{w}) \le C'B_{xv}^2 \sqrt{\frac{d\log(8\varepsilon_g/\varepsilon_{\mathbf{v}}) + \log(1/\varepsilon)}{N}},$$

where we use $\log|\mathcal{V}| \le d\log(4/\varepsilon_{\mathbf{v}})$. Therefore, we plug in the definition of $\varepsilon_{\mathbf{v}}$ and B_{xv} to deduce that, if we set

$$C_1 = \frac{16C'B_{xv}^2}{\varepsilon_x}^2 \log(8\varepsilon_g/\varepsilon_{\mathbf{v}}); \qquad \varepsilon_{\mathbf{v}} = \frac{128K_x^2}{\varepsilon_x}; \qquad B_{xv} = K_x(15 + \log(K_x^2/\varepsilon_x));$$

then as long as $N \ge C_1 d$, it holds $\max_{\mathbf{v}\in\mathcal{V}} \mathbb{E}[U_{\mathbf{v}}(\mathbf{w})] - U_{\mathbf{v}}(\mathbf{w}) \le \frac{\varepsilon_x B}{16}$ with probability at least $1 - \exp(-N/C_1)$. This is the desired result. □

Lemma G.3. *Under Assumption B, for $B_{xv} = K_x(15 + \log(K_x^2/\varepsilon_x))$, it holds*

$$\inf_{\mathbf{w}\in\mathrm{B}_2(B_w);\mathbf{v}\in\mathrm{S}^{d-1}} \mathbb{E}[1\{|\mathbf{x}^\top\mathbf{w}| \le B\} 2g(\mathbf{x}^\top\mathbf{v})^2 1\{|\mathbf{x}^\top\mathbf{v}| \le B_{xv}\}] \ge 3\varepsilon_x/4.$$

Proof. For any fixed $\mathbf{w} \in \mathrm{B}_2(B_w)$, $\mathbf{v} \in \mathrm{S}^{d-1}$,

$$\begin{aligned}
&\mathbb{E}[1\{|\mathbf{x}^\top\mathbf{w}| \le B\} 2g(\mathbf{x}^\top\mathbf{v})^2 1\{|\mathbf{x}^\top\mathbf{v}| \le B_{xv}\}] \\
&= \mathbb{E}[1\{|\mathbf{x}^\top\mathbf{w}| \le B\} 2g(\mathbf{x}^\top\mathbf{v})^2 g] - \mathbb{E}[1\{|\mathbf{x}^\top\mathbf{w}| \le B\} 2g(\mathbf{x}^\top\mathbf{v})^2 1\{|\mathbf{x}^\top\mathbf{v}| > B_{xv}\}] \\
&\ge \varepsilon_x - \mathbb{E}[(\mathbf{x}^\top\mathbf{v})^2 1\{|\mathbf{x}^\top\mathbf{v}| > B_{xv}\}].
\end{aligned}$$

Because $\mathbf{x} \in \mathrm{SG}(K_x)$, $\mathbf{x}^\top\mathbf{v} \in \mathrm{SG}(K_x)$, and a simple calculation yields

$$\mathbb{E}[(\mathbf{x}^\top\mathbf{v})^2 1\{|\mathbf{x}^\top\mathbf{v}| > tK_x\}] \le 2K_x^2(t^2 + 1)\exp(-t^2/2).$$

Taking $t = 15 + \log(K_x^2/\varepsilon_x)$ gives $\mathbb{E}[(\mathbf{x}^\top\mathbf{v})^2 1\{|\mathbf{x}^\top\mathbf{v}| > B_{xv}\}] \le \varepsilon_x/4$, which completes the proof. □

G.4 Proof of Theorem G.3 (b)

Notice that

$$r\,\underline{b}_N(\mathbf{w}) = \frac{1}{N}\sum_{i=1}^{N}(g(\langle \mathbf{w}, \mathbf{x}_i\rangle) - y_i)\mathbf{x}_i.$$

In the following, we pick a minimal $1/2$-covering of $\mathbb{S}^{d-1}$ (so $|\mathcal{V}| \le 5^d$). Then by Lemma G.2, it holds

$$\|r\,\underline{b}_N(\mathbf{w}) - \mathbb{E}[r\,\underline{b}_N(\mathbf{w})]\|_2 \le 2\max_{\mathbf{v}\in\mathcal{V}} \underbrace{\langle r\,\underline{b}_N(\mathbf{w}), \mathbf{v}\rangle - \mathbb{E}[\langle r\,\underline{b}_N(\mathbf{w}), \mathbf{v}\rangle]}_{=:X_{\mathbf{v}}(\mathbf{w})}.$$

Fix a $\mathbf{v} \in \mathbb{S}^{d-1}$ and set $\varepsilon' = \varepsilon/|\mathcal{V}|$. We proceed to bound $\sup_{\mathbf{w}} |X_{\mathbf{v}}(\mathbf{w})|$ by applying Proposition B.4 to the random process $\{X_{\mathbf{v}}(\mathbf{w})\}_{\mathbf{w}}$. We need to verify the preconditions:

(a) With norm $\gamma(\mathbf{w}, \mathbf{w}') = \|\mathbf{w} - \mathbf{w}'\|_2$, $\log N(\varepsilon, B_2(r), \gamma) \le d\log(2Ar/\varepsilon)$ with constant $A = 2$;

(b) For $\mathbf{z} = [\mathbf{x}, y]$, we let $f(\mathbf{z}; \mathbf{w}) := (g(\langle \mathbf{w}, \mathbf{x}\rangle) - y)\langle \mathbf{x}, \mathbf{v}\rangle$, then $f(\mathbf{z}; \mathbf{w}) - \mathrm{SE}(CK_xK_y)$ for any $\mathbf{w}$ by our assumption on $(\mathbf{x}, y)$;

(c) For $\mathbf{w}, \mathbf{w}' \in W$, we have $|g(\langle \mathbf{w}, \mathbf{x}\rangle) - g(\langle \mathbf{w}', \mathbf{x}\rangle)| \le L_g|\langle \mathbf{w} - \mathbf{w}', \mathbf{x}\rangle|$. Hence, because $\mathbf{x} \sim \mathrm{SG}(K_x)$, the random variable $g(\langle \mathbf{w}, \mathbf{x}\rangle) - g(\langle \mathbf{w}', \mathbf{x}\rangle)$ is sub-Gaussian in $\mathrm{SG}(K_xL_g\|\mathbf{w} - \mathbf{w}'\|_2)$. Thus, $f(\mathbf{z}; \mathbf{w}) - f(\mathbf{z}; \mathbf{w}')$ is sub-exponential in $\mathrm{SE}(CK_x^2L_g\|\mathbf{w} - \mathbf{w}'\|_2)$.

Therefore, we can apply Proposition B.4 to obtain that with probability $1 - \varepsilon_0$, it holds

$$\sup_{\mathbf{w}} |X_{\mathbf{v}}(\mathbf{w})| \le C'K_xK_y\left[\sqrt{\frac{d\log(2\varepsilon_y) + \log(1/\varepsilon_0)}{N}} + \frac{d\log(2\varepsilon_y) + \log(1/\varepsilon_0)}{N}\right],$$

where we denote $\varepsilon_y = 1 + L_gK_x^2B_w/K_y$. Setting $\varepsilon_0 = \varepsilon/|\mathcal{V}|$ and taking the union bound over $\mathbf{v} \in \mathcal{V}$, we obtain that with probability at least $1 - \varepsilon$,

$$\max_{\mathbf{v}\in\mathcal{V}} \sup_{\|\mathbf{w}\|_2\le B_w} |X_{\mathbf{v}}(\mathbf{w})| \le C'K_xK_y\left[\sqrt{\frac{d\log(10\varepsilon_y) + \log(1/\varepsilon)}{N}} + \frac{d\log(10\varepsilon_y) + \log(1/\varepsilon)}{N}\right].$$

This is the desired result. □

G.5 Proof of Theorem G.3 (c)

In the following, we condition on (a) holds, i.e. $\underline{b}_N$ is α-strongly-convex and β-smooth over $B_2(B_w)$ with $\alpha = \varepsilon_{\mathrm{stat}}/8$ and $\beta = 8L_gK_x^2$. We define

$$\widehat{\mathbf{w}} = \operatorname*{arg\,min}_{\mathbf{w}\in B_2(B_w)} \underline{b}_N(\mathbf{w}).$$

Then by standard convex analysis, we have

$$\alpha\|\widehat{\mathbf{w}} - \mathbf{w}^*\|_2^2 \le \langle r\,\underline{b}_N(\widehat{\mathbf{w}}) - r\,\underline{b}_N(\mathbf{w}^*), \widehat{\mathbf{w}} - \mathbf{w}^*\rangle \le \|r\,\underline{b}_N(\mathbf{w}^*)\|_2\|\widehat{\mathbf{w}} - \mathbf{w}^*\|_2.$$

Notice that $\|r\,\underline{b}_N(\mathbf{w}^*)\|_2 \le \varepsilon_{\mathrm{stat}}$, we can conclude that

$$\|\widehat{\mathbf{w}}\|_2 \le \|\mathbf{w}^*\|_2 + \frac{\varepsilon_{\mathrm{stat}}}{\alpha}.$$

Recall that we assume $\|\mathbf{w}^*\|_2 \le B_w/4$, we can then consider $E_s := \{\varepsilon_{\mathrm{stat}} < \alpha B_w/4\}$. Once E_s holds, our argument above yields $\|\widehat{\mathbf{w}}\|_2 < B_w$, which implies $r\,\underline{b}_N(\widehat{\mathbf{w}}) = 0$. Therefore, $\widehat{\mathbf{w}} = \arg\min_{\mathbf{w}\in\mathbb{R}^d} \underline{b}_N(\mathbf{w})$. Further, by Theorem G.3, we can set

$$C_2 := \max\left(2, \frac{32\beta K_xK_y}{\alpha B_w}^2; 2\frac{32\beta K_xK_y}{\alpha B_w}\right),$$

so that as long as $N \ge C_2d$, the event E_s holds with probability at least $1 - \exp(-N/C_2)$. This is the desired result. □

G.6 Proof of Theorem G.3 (d) & (e)

We first prove Theorem G.3 (d). Notice that

$$r^2 L_p(\mathsf{w}) = \mathsf{E}\, g'(\langle \mathsf{x}, \mathsf{w}\rangle)\mathsf{x}\mathsf{x}^\top \succeq \mathsf{E}\, \mu_g \mathbb{I}(|\langle \mathsf{x}, \mathsf{w}\rangle| \le B)\mathsf{x}\mathsf{x}^\top \succeq \mu_g \mu_x \mathbb{I}_d, \forall \mathsf{w} \in \mathsf{B}_2(B_w).$$

Therefore, L_p is $(\mu_g \mu_x)$-strongly-convex over $\mathsf{B}_2(B_w)$. Therefore, because $\mathsf{w}^\star \in \mathsf{B}_2(B_w)$ is the global minimum of L_p, it holds that for all $\mathsf{w} \in \mathsf{B}_2(B_w)$,

$$L_p(\mathsf{w}) - L_p(\mathsf{w}^\star) \le \frac{1}{2\mu_g \mu_x} \|\nabla L_p(\mathsf{w})\|_2^2.$$

By the definition of $\varepsilon_{\mathrm{stat}}$, $\|\nabla L_p(\mathsf{w})\|_2 \le \varepsilon_{\mathrm{stat}} + \|\nabla \hat{L}_N(\mathsf{w})\|_2$, and hence the proof of Theorem G.3 (d) is completed.

We next prove Theorem G.3 (e), where we assume that $\mathsf{E}[y|\mathsf{x}] = g(\langle \mathsf{x}, \mathsf{w}^\star\rangle)$ (which implies $\mathsf{w}^\star = \mathsf{w}^\star$ directly) and $\mathsf{w}_{\mathrm{GLM}} \in \mathsf{B}_2(B_w)$. Notice that

$$\nabla L_p(\mathsf{w}) = \mathsf{E}\left[\nabla \hat{L}_N(\mathsf{w})\right] = \mathsf{E}[(g(\langle \mathsf{x}, \mathsf{w}\rangle) - y)\mathsf{x}] = \mathsf{E}[(g(\langle \mathsf{x}, \mathsf{w}\rangle) - g(\langle \mathsf{w}^\star, \mathsf{x}\rangle))\mathsf{x}],$$

and hence

$$\begin{aligned}\langle \nabla L_p(\mathsf{w}_{\mathrm{GLM}}), \mathsf{w}_{\mathrm{GLM}} - \mathsf{w}^\star\rangle &= \mathsf{E}[(g(\langle \mathsf{x}, \mathsf{w}_{\mathrm{GLM}}\rangle) - g(\langle \mathsf{w}^\star, \mathsf{x}\rangle))(\langle \mathsf{x}, \mathsf{w}_{\mathrm{GLM}}\rangle - \langle \mathsf{w}^\star, \mathsf{x}\rangle)] \\ &\ge \frac{1}{L_g}\mathsf{E}\,(g(\langle \mathsf{x}, \mathsf{w}_{\mathrm{GLM}}\rangle) - g(\langle \mathsf{w}^\star, \mathsf{x}\rangle))^2.\end{aligned}$$

On the other hand, by the $(\mu_g \mu_x)$-strong-convexity of L_p over $\mathsf{B}_2(B_w)$, it holds that

$$\langle \nabla L_p(\mathsf{w}_{\mathrm{GLM}}), \mathsf{w}_{\mathrm{GLM}} - \mathsf{w}^\star\rangle \ge \frac{1}{\mu_g \mu_x}\|\nabla L_p(\mathsf{w}_{\mathrm{GLM}})\|_2^2.$$

Finally, using the definition of $\mathsf{w}_{\mathrm{GLM}}$, we have $\nabla \hat{L}_N(\mathsf{w}_{\mathrm{GLM}}) = 0$, and hence $\|\nabla L_p(\mathsf{w}_{\mathrm{GLM}})\|_2 \le \varepsilon_{\mathrm{stat}}$, which completes the proof of Theorem G.3 (e). □

H Proofs for Section 3.2

H.1 Proof of Theorem 7

Fix $\lambda_N \ge 0$, $\alpha > 0$ and $B_w > 0$, and consider any in-context data D such that the precondition of Theorem 7 holds. Recall that

$$L_{\mathrm{lasso}}(\mathsf{w}) := \frac{1}{2N}\sum_{i=1}^{N}(\langle \mathsf{w}, \mathsf{x}_i\rangle - y_i)^2 + \lambda_N \|\mathsf{w}\|_1$$

denotes the lasso regression loss in (ICLasso), so that $\mathsf{w}_{\mathrm{lasso}} = \arg\min_{\mathsf{w}\in\mathbb{R}^d} L_{\mathrm{lasso}}(\mathsf{w})$. We further write

$$\hat{L}_N^0(\mathsf{w}) := \frac{1}{2N}\sum_{i=1}^{N}(\langle \mathsf{w}, \mathsf{x}_i\rangle - y_i)^2, \qquad R(\mathsf{w}) := \lambda_N \|\mathsf{w}\|_1.$$

Note that $\nabla^2 \hat{L}_N^0(\mathsf{w}) = \mathsf{X}^\top \mathsf{X}/N$ and thus $\hat{L}_N^0$ is α-smooth over $\mathbb{R}^d$.

Consider the proximal gradient descent algorithm on the ridge loss

$$\mathsf{w}_{\mathrm{PGD}}^{t+1} = \mathrm{prox}_{\eta\mathcal{R}}\left[\mathsf{w}_{\mathrm{PGD}}^t - \eta \nabla \hat{L}_N^0(\mathsf{w}_{\mathrm{PGD}}^t)\right]$$

with initialization $\mathsf{w}_{\mathrm{PGD}}^0 := 0_d$, learning rate $\eta := \alpha^{-1}$, and number of steps T to be specified later. Similar to the proof of Theorem 4, we can construct a transformer to approximate $\mathsf{w}_{\mathrm{GD}}^T$. Consider $\ell(s; t) = \frac{1}{2}(s - t)^2$ and $R(\mathsf{w}) = \lambda_N \|\mathsf{w}\|_1$, then $\partial_s \ell(s; t)$ is $(0; +1; 2; 4)$-approximable by sum of relus (cf. Definition D.1), and $\mathrm{prox}_{\eta\mathcal{R}}$ is $(0; +1; 4d; 4 + 2\lambda_N)$-approximable by sum of relus (Proposition D.1). Therefore, we can apply Theorem D.2 with the square loss ℓ, regularizer R, learning rate η and accuracy parameter 0 to obtain that there exists a transformer TF with $(T + 1)$

layers, number of heads $M^{(\ell)} = 2$ for all $\ell \in [L]$, and hidden dimension $D' = 2d$, such that the final output $h_{N+1}^{(L)} = [x_{N+1}; \hat{y}_{N+1}; w_{\mathrm{PGD}}^T; \ldots]$ with $\hat{y}_{N+1} = \langle w_{\mathrm{PGD}}^T, x_{N+1} \rangle$. Further, the weight matrices have norm bounds $\|\cdot\| \le 10R + (8 + 2\cdot)^{-1}$.

By the standard convergence result for proximal gradient descent (Proposition B.3), we have for all $t \ge 1$ that

$$L_{\text{lasso}}(w_{\mathrm{PGD}}^t) - L_{\text{lasso}}(w_{\text{lasso}}) \le \frac{1}{2t} \|w_{\text{lasso}}\|_2^2.$$

Plugging in $\|w_{\text{lasso}}\|_2 \le B_w = 2$ and $T = L \cdot 1 = B_w^2/\varepsilon$ finishes the proof. □

H.2 Sharper convergence analysis of proximal gradient descent for Lasso

Collection of parameters Throughout the rest of this section, we consider fixed $N \ge 1$, $\delta_N = \sqrt{\frac{\log d}{N}}$ for ≥ 0, ≥ 0 fixed (and to be determined), fixed $0 < \le$, and fixed $B_w^\star > 0$. We write $:= = ;$ $_s := (B_w^\star)^2/\varepsilon^2$, and $\ell_N := -\frac{s \log d}{N}$.

Here we present a sharper convergence analysis on the proximal gradient descent algorithm for L_{lasso} under the following well-conditionedness assumption, which will be useful for proving Theorem 8 in the sequel.

Assumption C (Well-conditioned property for Lasso)**.** *We say the (ICLasso) problem is well-conditioned with sparsity s if the following conditions hold:*

1. *The $(\ ;\ )$-RSC condition holds:*

$$\frac{\|Xw\|_2^2}{N} \ge \|w\|_2^2 - \frac{\log d}{N}\|w\|_1^2, \quad \forall w \in \mathbb{R}^d. \tag{38}$$

   Further, $\max(X^\top X/N) \le$.

2. *The data (X, y) is "approximately generated from a s-sparse linear model": There* exists *a $w^\star \in \mathbb{R}^d$ such that $\|w^\star\|_2 \le B_w^\star, \|w^\star\|_0 \le s$ and for the residue $\varepsilon = y - Xw^\star$,*

$$\|X^\top \varepsilon\|_\infty \le \frac{1}{2} N \delta_N.$$

3. *It holds that $N \ge 32 - s \log d$ (i.e. $32\ell_N \le 1$).*

Assumption C1 imposes the standard restricted strong convexity (RSC) condition for the feature matrix $X \in \mathbb{R}^{N \times d}$, and Assumption C2 asserts that the data is approximately generated from a sparse linear model, with a bound on the L_∞ norm of the error vector $X^\top \varepsilon$. Assumption C is entirely deterministic in nature, and suffices to imply the following convergence result. In the proof of Theorem 8, we show that Assumption C is satisfied with high probability when data is generated from the standard sparse linear model considered therein.

Theorem H.1 (Sharper convergence guarantee for Lasso)**.** *Under Assumption C, for the PGD iterates $\{w^t\}_{t \ge 0}$ on loss function L_{lasso} with stepsize $= 1/$ and starting point $w^0 = 0$, we have $L_{\text{lasso}}(w^T) - L_{\text{lasso}}(w_{\text{lasso}}) \le \varepsilon$ for all*

$$T \ge C \frac{(B_w^\star)^2}{\varepsilon} + \log C \cdot \frac{(B_w^\star)^2}{\varepsilon} \cdot \frac{1}{\varepsilon} + \frac{\ell_N^2}{\varepsilon};$$

where C is a universal constant.

The proof can be found in Appendix H.4. Combining Theorem H.1 with the construction in Theorem 7, we directly obtain the following result as a corollary.

Theorem H.2 (In-context Lasso with transformers with sharper convergence)**.** *For any $N, d, s \ge 1$, $0 < $, ≥ 0, ≥ 0, there exists a L-layer transformer TF with*

$$L = C \cdot s + (\log(C \cdot s/\varepsilon) + \ell_N^2/\varepsilon) \cdot ; \quad \max_{\ell \in [L]} M^{(\ell)} \le 2; \quad \max_{\ell \in [L]} D^{(\ell)} \le 2d;$$

$$\|\| \|\| \quad 3 + R + (8 + 2 \quad _N) \quad ^{-1};$$

such that the following holds. On any input data $(D; x_{N+1})$ *such that the (ICLasso) problem satisfies Assumption C (which implies* $k w_{\text{lasso}} k_2 \quad B_w = 2$ *with* $B_w = 2B_w^? + \overline{\quad = \quad}$*),* $\mathrm{TF}\ (H^{(0)})$ *approximately implements (ICLasso), in that it outputs* $\hat{y}_{N+1} = \text{read}_y(\mathrm{TF}\ (H)) = h x_{N+1}; \hat{w} i$ *with*

$$\hat{b}_{\text{lasso}}(\hat{w}) \quad \hat{b}_{\text{lasso}}(w_{\text{lasso}}) \quad ":$$

H.3 Basic properties for Lasso

Lemma H.1 (Relaxed basic inequality)**.** *Suppose that Assumption C2 holds. Then it holds that*

$$k w \quad w_?k_1 \quad 4\sqrt{\bar{s}}\, k w \quad w_?k_2 + \frac{2}{N} \quad \hat{b}_{\text{lasso}}(w) \quad \hat{b}_{\text{lasso}}(w_?) \ ; \qquad 8w \ 2 \ \mathrm{R}^d:$$

As a corollary, $k w_{\text{lasso}} \quad w_?k_1 \quad 4\sqrt{\bar{s}}\, k w_{\text{lasso}} \quad w_?k_2$.

Proof. Let us first fix any $w \ 2 \ \mathrm{R}^d$. Denote $\quad = w \quad w_?$, and let $S = \text{supp}(w_?)$ be the set of indexes of nonzero entries of $w_?$. Then by definition, $y = X w_? + "$ and $jSj \quad s$, and hence

$$\begin{aligned}
&kX w \quad y k_2^2 \quad kX w_? \quad y k_2^2 = kX \qquad "k_2^2 - \sum \quad k"k_2^2 = kX \quad k_2^2 \quad 2"^\top X \ ; \\
&k w k_1 \quad k w_? k_1 = \sum_{j \in S} (j w[j]j \quad j w_?[j]j) + \sum_{j \notin S} j w[j]j \\
&\quad \sum_{j \in S} j w[j] \quad w_?[j]j + \sum_{j \notin S} j w[j]j = k \quad _{S^c} k_1 \quad k \quad _S k_1 :
\end{aligned}$$

Combining these inequalities, we obtain

$$\begin{aligned}
0 \quad \frac{1}{2N} kX \quad k_2^2 \quad \frac{"^\top X}{N} + \quad _N (k \quad _S k_1 \quad k \quad _{S^c} k_1) + \hat{b}_{\text{lasso}}(w) \quad \hat{b}_{\text{lasso}}(w_?) \\
\frac{N}{2} k \quad k_1 + \quad _N (k \quad _S k_1 \quad k \quad _{S^c} k_1) + \hat{b}_{\text{lasso}}(w) \quad \hat{b}_{\text{lasso}}(w_?) \qquad (39) \\
= \frac{N}{2} (3k \quad _S k_1 \quad k \quad _{S^c} k_1) + \hat{b}_{\text{lasso}}(w) \quad \hat{b}_{\text{lasso}}(w_?);
\end{aligned}$$

where the second inequality follows from $\frac{" > x}{N} \quad \frac{k x > "k_1}{N} k \quad k_1$ and our assumption that $2\frac{k x > "k_1}{N} \quad _N$, and the last inequality is due to $k \quad k_1 = k \quad _S k_1 + k \quad _{S^c} k_1$. Therefore, we have

$$\begin{aligned}
k \quad k_1 = k \quad _S k_1 + k \quad _{S^c} k_1 \quad 4k \quad _S k_1 + \frac{2}{N} \quad \hat{b}_{\text{lasso}}(w) \quad \hat{b}_{\text{lasso}}(w_?) \\
4\sqrt{\bar{s}} k \quad k_2 + \frac{2}{N} \quad \hat{b}_{\text{lasso}}(w) \quad \hat{b}_{\text{lasso}}(w_?) \ ;
\end{aligned}$$

where the last inequality follows from $k \quad _S k_1 \quad \sqrt{\bar{s}} k \quad _S k_2 \quad \sqrt{\bar{s}} k \quad k_2$. This completes the proof of our main inequality. As for the corollary, we only need to use the definition that $\hat{b}_{\text{lasso}}(w_{\text{lasso}})$ $\hat{b}_{\text{lasso}}(w_?)$. □

Proposition H.1 (Gap to parameter estimation error)**.** *Suppose that Assumption C holds. Then for all* $w \ 2 \ \mathrm{R}^d$*,*

$$k w \quad w_?k_2^2 \quad C \ \frac{s \ \frac{2}{N}}{2} + \quad ^{-1}\text{gap}^2 + \text{gap} \ ;$$

where we write $\text{gap} := \hat{b}_{\text{lasso}}(w) \quad \hat{b}_{\text{lasso}}(w_{\text{lasso}})$*, and* $C = 120$ *is a universal constant. In particular, we have* $k w_{\text{lasso}} \quad w_?k_2^2 \quad 10 \frac{s \log d}{N}$.

Proof. We follow the notation in the proof of Lemma H.1. By (39), we have

$$0 \quad \frac{1}{2N}\|X \quad \|_2^2 \quad \frac{N}{2}(3\| \quad _S\|_1 \quad \| \quad _{S^c}\|_1) + \underline{b}_{\text{lasso}}(w) \quad \underline{b}_{\text{lasso}}(w_?);$$

and hence $\| \quad \|_1 \quad 4\sqrt{s}\| \quad \|_2 + \frac{2\text{gap}}{N}$ due to $\underline{b}_{\text{lasso}}(w) \quad \underline{b}_{\text{lasso}}(w_?) \quad \text{gap}$. On the other hand, by the RSC condition (38), it holds that

$$\frac{\|X \quad \|_2^2}{N} \quad \| \quad \|_2^2 \quad \frac{\log d}{N}\| \quad \|_1^2:$$

Therefore, we have

$$\begin{aligned}
\| \quad \|_2^2 \quad & 3 \quad _N\sqrt{s}\| \quad \|_2 + \frac{\log d}{N}\| \quad \|_1^2 + 2\text{gap} \\
& 3 \quad _N\sqrt{s}\| \quad \|_2 + \frac{\log d}{N} \quad 4\sqrt{s}\| \quad \|_2 + \frac{2\text{gap}}{N} \quad ^2 + 2\text{gap} \\
& \frac{5s \quad _N^2}{} + \frac{}{6}\| \quad \|_2^2 + \frac{20s\log d}{_N^2 N}\| \quad \|_2^2 + \frac{20\log d}{N}\text{gap}^2 + 2\text{gap};
\end{aligned}$$

where the last inequality uses AM-GM inequality and Cauchy inequality. Notice that $\frac{20s\log d}{N} \quad \frac{2}{3}$, we now derive that

$$\| \quad \|_2^2 \quad \frac{30s \quad _N^2}{_2} + \frac{120\log d}{_N^2 N}\text{gap}^2 + 12\text{gap}:$$

Plugging in $_N = \sqrt{\frac{\log d}{N}}$ completes the proof. The corollary follows immediately by letting $w = w_{\text{lasso}}$ in above proof (hence $\text{gap} = 0$). □

Lemma H.2 (Growth)**.** *It holds that*

$$\frac{1}{2N}\|X(w \quad w_{\text{lasso}})\|_2^2 \quad \underline{b}_{\text{lasso}}(w) \quad \underline{b}_{\text{lasso}}(w_{\text{lasso}}); \qquad 8w:$$

Proof. For simplicity we denote $w_{\text{lasso}} := w_{\text{lasso}}$. By the first order optimality condition, it holds that

$$0 \; 2\frac{1}{N}X^\top(Xw_{\text{lasso}} \quad y) + \partial R(w_{\text{lasso}});$$

where we write $R(w) := \quad _N\|w\|_1$. Then by the convexity of R, we have

$$\begin{aligned}
R(w) \quad R(w_{\text{lasso}}) \quad & \langle \partial R(w_{\text{lasso}}); w \quad w_{\text{lasso}}\rangle = \quad \frac{1}{N}X^\top(Xw_{\text{lasso}} \quad y); w \quad w_{\text{lasso}} \\
= \quad & \frac{1}{N}\langle Xw_{\text{lasso}} \quad y; (Xw \quad y) \quad (Xw_{\text{lasso}} \quad y)\rangle \\
= \quad & \frac{1}{2N}\|Xw \quad y\|_2^2 + \frac{1}{2N}\|Xw_{\text{lasso}} \quad y\|_2^2 + \frac{1}{2N}\|X(w \quad w_{\text{lasso}})\|_2^2:
\end{aligned}$$

Rearranging completes the proof. □

H.4 Proof of Theorem H.1

For the simplicity of presentation, we write $w_{\text{lasso}} = w_{\text{lasso}}$ and we denote $\text{gap}^t := \underline{b}_{\text{lasso}}(w^t) \quad \underline{b}_{\text{lasso}}(w_{\text{lasso}})$.

By Lemma H.1, we have $\|w^t \quad w_?\|_1 \quad 4\sqrt{s}\|w^t \quad w_?\|_2 + \frac{2\text{gap}^t}{N}$, which implies

$$\|w^t \quad w_{\text{lasso}} \quad _1 \quad \|w^t \quad w_? \quad _1 + \|w_{\text{lasso}} \quad w_?\|_1 \quad 4\sqrt{s} \quad \|w^t \quad w_{\text{lasso}} \quad _2 + 8\sqrt{s}\|w_{\text{lasso}} \quad w_?\|_2 + \frac{2\text{gap}^t}{N}:$$

We denote $_N = {}^{2\log d}_{N}$. Using the assumption that X is $(\ ; \ )$-RSC, we obtain that

$$\frac{1}{N} X(w^t \quad w_{lasso})_{2}^{2} \quad w^t \quad w_{lasso}{}_{2}^{2} \quad _N \ w^t \quad w_{lasso}{}_{1}^{2}$$
$$w^t \quad w_{lasso}{}_{2}^{2} \quad _N \ 20s \ w^t \quad w_{lasso}{}_{2}^{2} + 640s\, k w_{lasso} \quad w_? k_2^2 + \frac{40}{{}^2_N}(gap^t)^2 \ :$$

Thus, as long as $N \quad \frac{30 \ ^2 s\log d}{}$, we have

$$\frac{}{3} \ w^t \quad w_{lasso}{}_{2}^{2} \quad \frac{1}{N} X(w^t \quad w_{lasso})_{2}^{2} + 640s \ _N k w_{lasso} \quad w_? k_2^2 + \frac{40 \ _N}{{}^2_N}(gap^t)^2$$
$$2gap^t + 40 \ ^{-1}(gap^t)^2 + 640s \ _N k w_{lasso} \quad w_? k_2^2 ;$$

where the last inequality follows from Lemma H.2 and the definition of $_N ; \ _N$.

We define $"_{stat} := 640s \ _N k w_{lasso} \quad w_? k_2^2$, $T_0 := 10 \quad ^{-1} k w_{lasso} k_2^2$. By Proposition B.3(3), it holds that for $t \quad T_0$,

$$gap^t \quad \frac{}{2t} k w_{lasso} k_2^2 \quad \frac{}{2T_0} k w_{lasso} k_2^2 = \frac{}{20} :$$

Then for all $t \quad T_0 \quad 1$, we have (the second below uses Proposition B.3(2))

$$\frac{}{3} \ w^{t+1} \quad w_{lasso}{}_{2}^{2} \quad 4gap^{t+1} + "_{stat} \quad 2 \quad w^t \quad w_{lasso}{}_{2}^{2} \quad w^{t+1} \quad w_{lasso}{}_{2}^{2} + "_{stat} ;$$
$$) \quad w^{t+1} \quad w_{lasso}{}_{2}^{2} \quad \frac{3"_{stat}}{} \quad 1 + \frac{}{6} \quad ^{-1} \quad w^t \quad w_{lasso}{}_{2}^{2} \quad \frac{3"_{stat}}{} \ :$$

Therefore, for $t \quad T_0 \quad 1$,

$$w^t \quad w_{lasso}{}_{2}^{2} \quad \exp \quad \frac{}{12}(t \quad dT_0 e + 1) \quad w^{\lceil T_0 \rceil - 1} \quad w_{lasso}{}_{2}^{2} + \frac{3"_{stat}}{}$$
$$\exp \quad \frac{}{8}(t \quad T_0) \quad k w_{lasso} k_2^2 + \frac{3"_{stat}}{} ;$$

where the last inequality follows from Proposition B.3(2). Further, by Proposition B.3(3), we have

$$gap^{t+k} \quad \frac{}{2k} \ w^t \quad w_{lasso}{}_{2}^{2} \quad \frac{}{2k} \ \exp \quad \frac{}{8}(t \quad T_0) \quad k w_{lasso} k_2^2 + \frac{3"_{stat}}{} \ ; \quad 8t \quad T_0 \quad 1; k \quad 0:$$

Hence, we can conclude that $gap^T \quad "$ for all T such that

$$T \quad 10 \quad ^{-1} k w_{lasso} k_2^2 + 8 \ \log \quad \frac{k w_{lasso} k_2^2}{"} \quad ! \quad + \frac{3 \ "_{stat}}{"} + 1:$$

Now, by Proposition H.1, it holds that $k w_{lasso} \quad w_? k_2^2 \quad 10 \frac{}{2} \frac{s\log d}{N}$, and hence

$$k w_{lasso} k_2^2 \quad 2 k w_? k_2^2 + 2 k w_{lasso} \quad w_? k_2^2 \quad 2(B_w^?)^2 + \frac{20 \quad s\log d}{{}^2 N} :$$

Plugging in our definition of

$$_N = \frac{\log d}{N} ; \qquad "_{stat} := 400s \ _N k w_{lasso} \quad w_? k_2^2 ; \qquad ! _N = -\frac{s\log d}{N} \quad 1$$

completes the proof. □

H.5 Proof of Theorem 8

In this section, we present the proof of Theorem 8 based on Theorem H.2. We begin by recalling the following RSC property of a Gaussian random matrix [87, Theorem 7.16], a classical result in the high-dimensional statistics literature.

Proposition H.2 (RSC for Gaussian random design)**.** *Suppose that* $X = [x_1; \quad ; x_N]^\top \in \mathbb{R}^{N\times d}$ *is a random matrix with each row* x_i *being i.i.d. samples from* $\mathsf{N}(0; \quad)$*. Then there are universal constants* $c_1 = \frac{1}{8}; c_2 = 50$ *such that with probability at least* $1 - \frac{e^{-N/32}}{1-e^{-N/32}}$*,*

$$\frac{\|Xw\|_2^2}{N} \quad c_1\|w\|^2 \quad c_2 \ (\ )\frac{\log d}{N}\|w\|_1^2; \qquad \forall w \in \mathbb{R}^d; \tag{40}$$

where $(\ ) = \max_{i\in[d]} \ _{ii}$ *is the maximum of diagonal entries of* $\ $*.*

Fix a parameter $_1$ (which we will specify in proof) and a large universal constant C_0. Let us set

$$= c_1 = \ (1); \qquad = 8(1 + (d/N)); \qquad = c_2 = \ (1);$$
$$B_x = C_0\sqrt{\frac{}{d\log(N/ _1)}}; \qquad B_y = C_0(B_w^2 + \ )\sqrt{\frac{}{\log(N/ _1)}}.$$

Similar to the proof of Corollary 6 (Appendix F.4), we consider the following good events (where $\varepsilon = Xw_\star \quad y$)

$$E_w = \left\{\frac{\max(X^\top X/N)}{} \text{ and } X \text{ is } (\ ; \ )\text{-RSC}\right\};$$
$$E_r = \left\{\frac{\|X^\top \varepsilon\|_\infty}{} \ 4\sqrt{\frac{}{N\log(4d/)}}\right\};$$
$$E_b = \{\forall i \in [N],\ \|x_i\|_2 \quad B_x,\ |y_i| \quad B_y\};$$
$$E_{b;N+1} = \{\|x_{N+1}\|_2 \quad B_x,\ |y_{N+1}| \quad B_y\};$$

and we define $E := E_w \cap E_r \cap E_b \cap E_{b;N+1}$.

Furthermore, we choose $\ > 0$ that correspond to the choice $_N = 8\sqrt{\frac{\log(4d/)}{N}}$, and we also assume $N \quad \frac{32c_2}{c_1}\, s\log d$. Then, Assumption C holds on the event E.

Therefore, we can apply Theorem H.2 with $\varepsilon = \ !_N$, which implies that there exists a L-layer transformer $\ $ such that its prediction $\hat{y}_{N+1} := \mathrm{read}_y(\mathrm{TF}^0(H))$, so that under the good event E we have $\hat{y}_{N+1} = \mathrm{clip}_{B_y}(\langle x_{N+1}, \hat{w}\rangle)$, where

$$L_{\text{lasso}}(\hat{w}) \quad L_{\text{lasso}}(w_{\text{lasso}}) \quad \ !_N.$$

In the following, we show that $\ $ is indeed the desired transformer (similarly to the proof in Appendix F.4). Consider the conditional prediction error

$$\mathbb{E}\ (\hat{y}_{N+1} \quad y_{N+1})^2 \ D\ = \mathbb{E}\ 1\{E\}(\hat{y}_{N+1} \quad y_{N+1})^2 \ D\ + \mathbb{E}\ 1\{E^c\}(\hat{y}_{N+1} \quad y_{N+1})^2 \ D\ ;$$

and we analyze these two parts separately under the good event $E_0 := E_w \cap E_r \cap E_b$ of D.

Part I. We first note that

$$\mathbb{E}\ 1\{E\}(\hat{y}_{N+1} \quad y_{N+1})^2 \ D\ = \mathbb{E}\left[1\{E\}(\mathrm{clip}_{B_y}(\langle x_{N+1}, \hat{w}\rangle) \quad y_{N+1})^2 \ D\right]$$
$$\mathbb{E}\ 1\{E\}(\langle x_{N+1}, \hat{w}\rangle \quad y_{N+1})^2 \ D\ ;$$

where the inequality is because $y_{N+1} \in [\ B_y; B_y]$ under the good event E. Notice that by our construction, under the good event E, $\hat{w} = \hat{w}(D)$ depends only on the dataset D (because it is the $(L \quad 1)$-th iterate of PGD on (ICLasso) problem). Applying Proposition H.1 to $\hat{w}(D)$ and using the definition of $!_N$ and our choice of $_N$, we obtain that (under E_0)

$$\|\hat{w}(D) \quad w_\star\|_2^2 \quad C\left(\frac{s\ ^2_N}{2} + \ !_N^2 + \ !_N\right) = O\left(\frac{^2 s\log(d/)}{N}\right).$$

Therefore, under E_0,

$$\begin{aligned}
\mathbb{E}\ 1\{E\}(\langle x_{N+1}, \hat{w}\rangle \quad y_{N+1})^2 \ D\ &= \mathbb{E}\ 1\{E\}(\langle x_{N+1}, \hat{w}(D)\rangle \quad y_{N+1})^2 \ D \\
&\quad \mathbb{E}\ (\langle x_{N+1}, \hat{w}(D)\rangle \quad y_{N+1})^2 \ D \\
&= \mathbb{E}\ (\langle x_{N+1}, \hat{w}(D)\rangle \quad \langle x_{N+1}, w_\star\rangle)^2 \ D\ + \ ^2 \\
&= \|\hat{w}(D) \quad w_\star\|_2^2 + \ ^2 \\
&= \ ^2\left(1 + O\left(\frac{s\log(d/)}{N}\right)\right).
\end{aligned}$$

Part II. Notice that under good event E_0, the bad event E^c holds if and only if $E^c_{b;N+1}$ holds, and hence

$$\mathsf{E}\ \mathbf{1}\{E^c\}(\hat{y}_{N+1} - y_{N+1})^2\ D\ = \mathsf{E}\left[\frac{\mathbf{1}\{E^c_{b;N+1}\}(\hat{y}_{N+1} - y_{N+1})^2\ D}{\mathsf{P}(E^c_{b;N+1})\mathsf{E}[(\hat{y}_{N+1} - y_{N+1})^4]}\right].$$

With a large enough constant C_0, we clearly have $\mathsf{P}(E^c_{b;N+1})$ $(\ _1/N)^{10}$. Further, a simple calculation yields

$$\mathsf{E}(\hat{y}_{N+1} - y_{N+1})^4 \quad 8\mathsf{E}(\hat{y}^4_{N+1} + y^4_{N+1}) \quad 8B_y^4 + 8\mathsf{E}y^4_{N+1} \quad 16B_y^4;$$

where the last inequality is because the marginal distribution of y_{N+1} is simply $\mathsf{N}(0;\ ^2 + k\mathsf{w}_?k_2^2)$. Combining these yields

$$\mathsf{E}\ \mathbf{1}\{E^c\}(\hat{y}_{N+1} - y_{N+1})^2\ D \quad O\left(\frac{^5_1 B_y^2}{N^5}\right) \quad O\left(\frac{^5_1((B_w^?)^2 + \ ^2)\log(1/\ _1)}{N^4}\right):$$

Therefore, choosing $\ _1 = \min\{\ ; \frac{}{B_w^?}\}$ is enough for our purpose, and under such choice of $\ _1$,

$$\mathsf{E}\ \mathbf{1}\{E^c\}(\hat{y}_{N+1} - y_{N+1})^2\ D \quad O\left(\frac{^2}{N^4}\right):$$

Conclusion. Combining the inequalities above, we can conclude that under E_0,

$$\mathsf{E}\ (\hat{y}_{N+1} - y_{N+1})^2\ D \quad ^2\left(1 + O\left(\frac{s\log(d/\ )}{N}\right)\right):$$

It remains to show that $\mathsf{P}(E_0)$ 1 . By Proposition H.2, Lemma B.2 and Lemma B.4, we have

$$\mathsf{P}(E_w) \quad 3\exp(\ N/32); \qquad \mathsf{P}(E_r) \quad \frac{}{2}; \qquad \mathsf{P}(E_b) \quad \frac{}{4}:$$

Therefore, as long as N $32\log(12/\ )$, we have $\mathsf{P}(E_0)$ 1 . This completes the proof. □

We also remark that in the construction above,

$$R = O\left((B_w^? + \ )\sqrt{d}\log(N\ (1 + B_w^?/\ ))\right);$$

which would be useful for bounding $\|\|$ $\|\|$.

I Proofs for Section 4

I.1 Proof of Proposition 10

We begin by restating Proposition 10 into the following version, which contains additional size bounds on .

Theorem I.1 (Full statement of Proposition 10)**.** *Suppose that for*

$$\hat{}_{\mathsf{val}}(f) := \frac{1}{|D_{\mathsf{val}}|}\sum_{(\mathbf{x}_i;y_i)\in\mathcal{D}_{\mathsf{val}}} `(f(\mathbf{x}_i);y_i);$$

$`(\ ;\ )$ *is* $(\ =3;R;M;C)$*-approximable by sum of relus (Definition D.1). Then there exists a 3-layer transformer* TF *with*

$$\max_{`\in[3]} M^{(`)} \quad (M + 3)K; \quad \max_{`\in[3]} D^{(`)} \quad K^2 + K + 1; \quad \|\|\ \|\| \quad \frac{2NKC}{|D_{\mathsf{val}}|} + 3\ ^{-1} + 7KR:$$

that maps

$$\mathsf{h}_i = [\ ;f_1(\mathbf{x}_i); \quad ;f_K(\mathbf{x}_i);0_{K+1};1;t_i] \quad ! \quad \mathsf{h}'_i = [\ ;\hat{}(\mathbf{x}_i);1;t_i];\ i\ 2\ [N + 1];$$

where the predictor $\hat{}:\mathsf{R}^d\ !\ \mathsf{R}$ *is a convex combination of* $\{f_k : \hat{}_{\mathsf{val}}(f_k)\quad \min_{k_?\in[K]}\hat{}_{\mathsf{val}}(f_{k_?}) + \ \}$*. As a corollary, for any convex risk* $L:(\mathsf{R}^d\ !\ \mathsf{R})\ !\ \mathsf{R}$*,* $\hat{}$ *satisfies*

$$L(\hat{}) \quad \min_{k_?\in[K]} L(f_{k_?}) + \max_{k\in[K]}\hat{}_{\mathsf{val}}(f_k) \quad L(f_k) + \ :$$

To prove Theorem I.1, we first state and prove the following two propositions.

Proposition I.1 (Evaluation layer)**.** *There exists a 1-layer transformer* TF *with* MK *heads and* $\|\cdot\|_{\mathrm{op}} \le 3R + 2NKC/|\mathcal{D}_{\mathrm{val}}|$ *such that for all* H *such that* $\max_i |y'_i| \le R, \max_{i,k} |f_k(x_i)| \le R$, TF *maps*

$$h_i = [x_i; y'_i; \ldots; f_1(x_i); \ldots; f_K(x_i); 0_{K+1}; 1; t_i]$$
$$\to h'_i = [x_i; y'_i; \ldots; f_1(x_i); \ldots; f_K(x_i); \widehat{\mathcal{E}}_{\mathrm{val}}(f_1); \ldots; \widehat{\mathcal{E}}_{\mathrm{val}}(f_K); 0; 1; t_i], \quad i \in [N+1];$$

where $\widehat{\mathcal{E}}_{\mathrm{val}}(\cdot)$ *is a functional such that* $\max_k |\widehat{\mathcal{E}}_{\mathrm{val}}(f_k) - \mathcal{E}_{\mathrm{val}}(f_k)| \le \varepsilon$.

Proof of Proposition I.1. As ℓ is (ε, R, M, C)-approximable by sum of relus, there exists a function $g : \mathbb{R}^2 \to \mathbb{R}$ of form

$$g(s,t) = \sum_{m=1}^{M} c_m \sigma(a_m s + b_m t + d_m) \quad \text{with} \quad \sum_{m=1}^{M} |c_m| \le C, \ |a_m| + |b_m| + |d_m| \le 1, \ \forall m \in [M],$$

such that $\sup_{(s,t)\in[-R,R]^2} |g(s,t) - \ell(s,t)| \le \varepsilon$. We define

$$\widehat{\mathcal{E}}_{\mathrm{val}}(f) := \frac{1}{|\mathcal{D}_{\mathrm{val}}|} \sum_{(x_i,y_i)\in\mathcal{D}_{\mathrm{val}}} g(f(x_i), y_i),$$

Next, for every $m \in [M]$ and $k \in [K]$, we define matrices $Q_{m,k}, K_{m,k}, V_{m,k} \in \mathbb{R}^{D\times D}$ such that for all $i, j \in [N+1]$,

$$Q_{m,k} h_i = \begin{bmatrix} a_m \\ b_m \\ d_m \\ 2 \\ 0 \end{bmatrix}; \quad K_{m,k} h_j = \begin{bmatrix} f_k(x_j) \\ y_j \\ 1 \\ R(1+t_j) \\ 0 \end{bmatrix}; \quad V_{m,k} h_j = \frac{(N+1)c_m}{|\mathcal{D}_{\mathrm{val}}|} e_{D-(K-k)-3}$$

where $e_s \in \mathbb{R}^D$ is the vector with s-th entry being 1 and others being 0. As the input has structure $h_i = [x_i; y'_i; \ldots; f_1(x_i); \ldots; f_K(x_i); 0_{K+1}; 1; t_i]$, these matrices indeed exist, and further it is straightforward to check that they have norm bounds

$$\max_{m\in[M];k\in[K]} \|Q_{m,k}\|_{\mathrm{op}} \le 3; \quad \max_{m\in[M];k\in[K]} \|K_{m,k}\|_{\mathrm{op}} \le 2 + R; \quad \sum_{m\in[M];k\in[K]} \|V_{m,k}\|_{\mathrm{op}} \le \frac{K(N+1)C}{|\mathcal{D}_{\mathrm{val}}|}.$$

Now, for every $i, j \in [N+1]$, we have

$$\begin{aligned}(\langle Q_{m,k} h_i, K_{m,k} h_j\rangle) &= (a_m f_k(x_j) + b_m y_j + d_m \ge 2R(1+t_j)) \\ &= a_m w^\top x_j + b_m y_j + d_m \ge 1\{t_j = 1\},\end{aligned}$$

where the last equality follows from the bound $|a_m f_k(x_j) + b_m y_j + d_m| \le R(|a_m| + |b_m|) + |d_m| \le 2R$, so that the above relu equals 0 if $t_j = 0$. Therefore, for each $i \in [N+1]$ and $k \in [K]$,

$$\begin{aligned}&\sum_{m=1}^{M} (\langle Q_{m,k} h_i, K_{m,k} h_j\rangle) V_{m,k} h_j \\ &= \sum_{m=1}^{M} c_m \left(a_m w^\top x_j + b_m y_j + d_m \ge \frac{(N+1)}{|\mathcal{D}_{\mathrm{val}}|} 1\{t_j = 1\}\right) e_{D-(K-k)-3} \\ &= g(f_k(x_j), y_j) \frac{(N+1)}{|\mathcal{D}_{\mathrm{val}}|} 1\{t_j = 1\} e_{D-(K-k)-3}.\end{aligned}$$

Thus letting the attention layer $= f(V_{m,k}, Q_{m,k}, K_{m,k})_{(m,k)\in[M]\times[K]}$, we have

$$\widehat{h}_i = [\mathrm{Attn}(H)]_i = h_i + \frac{1}{N+1} \sum_{j=1}^{N+1} \sum_{m,k} (\langle Q_{m,k} h_i, K_{m,k} h_j\rangle) V_{m,k} h_j$$

$$= \mathsf{h}_i + \frac{1}{|D_{\text{val}}|} \sum_{j=1}^{N+1} \sum_{k=1}^{K} g(f_k(\mathsf{x}_j); y_j) \; \mathbb{1}\{t_j = \ell\} \mathsf{e}_{D-(K-k)-3}$$

$$= \mathsf{h}_i + \sum_{k=1}^{K} \left(\frac{1}{|D_{\text{val}}|} \sum_{(\mathbf{x}_j, y_j) \in \mathcal{D}_{\text{val}}} g(f_k(\mathsf{x}_j); y_j) \right) \mathsf{e}_{D-(K-k)-3}$$

$$= \mathsf{h}_i + \sum_{k=1}^{K} \mathsf{err}_{\text{val}}(f_k) \; \mathsf{e}_{D-(K-k)-3}$$

$$= [\mathsf{x}_i; y'_i; \; ; f_1(\mathsf{x}_i); \quad ; f_K(\mathsf{x}_i); 0_{K+1}; 1; t_i] + [0_{D-K-3}; \mathsf{err}_{\text{val}}(f_1); \quad ; \mathsf{err}_{\text{val}}(f_K); 0; 0; 0]$$

$$= [\mathsf{x}_i; y'_i; \; ; f_1(\mathsf{x}_i); \quad ; f_K(\mathsf{x}_i); \mathsf{err}_{\text{val}}(f_1); \quad ; \mathsf{err}_{\text{val}}(f_K); 0; 1; t_i]; \qquad i \in [N+1].$$

This is the desired result. □

Proposition I.2 (Selection layer)**.** *There exists a 3-layer transformer* TF *with*

$$\max_{\ell \in [3]} M^{(\ell)} \quad 2K + 2; \quad \max_{\ell \in [3]} D^{(\ell)} \quad K^2 + K + 1; \quad \|\|\|\| \quad ^{-1} + 3KR + 2.$$

such that TF *maps*

$$\mathsf{h}_i = [\; ; f_1(\mathsf{x}_i); \quad ; f_K(\mathsf{x}_i); \mathsf{L}_1; \quad ; \mathsf{L}_K; 0; 1; t_i]$$

$$\to \quad \mathsf{h}'_i = [\; ; f_1(\mathsf{x}_i); \quad ; f_K(\mathsf{x}_i); \; ; \quad ; \; ; \hat{p}(\mathsf{x}_i); 1; t_i]; \qquad i \in [N+1];$$

where $\hat{p} = \sum_{k=1}^{K} {}_k f_k$ *is an aggregated predictor, where the weights* ${}_1; \quad ; {}_K \quad 0$ *are functions only on* $\mathsf{L}_1; \quad ; \mathsf{L}_K$ *such that*

$$\sum_{k=1}^{K} {}_k = 1; \qquad {}_k > 0 \text{ only if } \mathsf{L}_k \quad \min_{k' \in [K]} \mathsf{L}_{k'} + \; :$$

Proof of Proposition I.2. We construct a which is a composition of 2 MLP layers followed by an attention layer $(\mathsf{MLP}^{(1)}_{\text{mlp}}, \mathsf{MLP}^{(2)}_{\text{mlp}}, \mathsf{ATTN}^{(3)}_{\text{attn}})$.

Step 1: construction of $\mathsf{MLP}^{(1)}_{\text{mlp}}$. We consider matrix $\mathsf{W}_1^{(1)}$ that maps

$$\mathsf{h} = [\; _{D-K-3}; \mathsf{L}_1; \quad ; \mathsf{L}_K; \; ; \; ; \;]$$

$$\to \quad \mathsf{W}_1^{(1)} \mathsf{h} = [\mathsf{L}_1 \quad \mathsf{L}_2; \quad ; \mathsf{L}_1 \quad \mathsf{L}_K; \quad ; \mathsf{L}_K \quad \mathsf{L}_{K-1}; \mathsf{L}_1; \quad \mathsf{L}_1; \quad ; \mathsf{L}_K; \quad \mathsf{L}_K];$$

i.e. $\mathsf{W}_1^{(1)} \mathsf{h}$ is a $K^2 + K$ dimensional vector so that its entry contains $\{\mathsf{L}_k \quad \mathsf{L}_l\}_{k,l \in [K]}$ and $\{\mathsf{L}_k; \quad \mathsf{L}_k\}_{k \in [K]}$. Clearly, such $\mathsf{W}_1^{(1)}$ exists and can be chosen so that $\|\mathsf{W}_1^{(1)}\|_{\text{op}} \quad 2K$. We then consider a matrix $\mathsf{W}_2^{(1)}$ that maps

$$(\mathsf{W}_1^{(1)} \mathsf{h}) \to \mathsf{W}_2^{(1)} \; (\mathsf{W}_1^{(1)} \mathsf{h}) = [0_{D-K-3}; c_1 \quad \mathsf{L}_1; \quad ; c_K \quad \mathsf{L}_K; 0_3] \in \mathbb{R}^D;$$

where $c_k = c_k(\mathsf{L}) := \sum_{l \neq k} (\mathsf{L}_k \quad \mathsf{L}_l)$. Notice that

$$c_k \quad \mathsf{L}_k = \quad (\mathsf{L}_k) + \quad (\quad \mathsf{L}_k) + \sum_{l \neq k} (\mathsf{L}_k \quad \mathsf{L}_l);$$

and hence such $\mathsf{W}_2^{(1)}$ exists and can be chosen so that $\|\mathsf{W}_2^{(1)}\|_{\text{op}} \quad K + 1$. We set $\mathsf{MLP}^{(1)}_{\text{mlp}} = (\mathsf{W}_1^{(1)}; \mathsf{W}_2^{(1)})$, then $\mathsf{MLP}^{(1)}_{\text{mlp}}$ maps h_i to

$$\mathsf{h}_i^{(1)} = [\; ; f_1(\mathsf{x}_i); \quad ; f_K(\mathsf{x}_i); c_1; \quad ; c_K; 0; 1; t_i].$$

The basic property of $\{c_k\}_{k \in [K]}$ is that, if c_k , then $\mathsf{L}_k \quad \min_{k' \in [K]} \mathsf{L}_{k'} + \; .$

Step 2: construction of $\mathsf{MLP}^{(2)}_{\text{mlp}}$. We consider matrix $\mathsf{W}_1^{(2)}$ that maps

$$\mathsf{h} = [\; _{D-K-3}; c_1; \quad ; c_K; \; ; 1; \;]$$

$$\forall\ W_1^{(2)} h = [1 \quad {}^{-1}c_1; c_1; \quad c_1; \quad ; 1 \quad {}^{-1}c_K; c_K; \quad c_K] \in \mathbb{R}^{3K};$$

and $W_1^{(2)}$ can be chosen so that $\|W_1^{(2)}\|_{\mathrm{op}} \quad K + 1 + {}^{-1}$. We then consider a matrix $W_2^{(2)}$ that maps

$$(W_1^{(2)} h) \mapsto W_2^{(2)} (W_1^{(1)} h) = [0_{D-K-3}; (1 \quad {}^{-1}c_1) \quad c_1; \quad ; (1 \quad {}^{-1}c_K) \quad c_K; 0_3] \in \mathbb{R}^D;$$

which exists and can be chosen so that $\|W_2^{(1)}\|_{\mathrm{op}} \quad 2$. We set $\theta^{(2)}_{\mathtt{mlp}} = (W_1^{(2)}, W_2^{(2)})$, then $\mathrm{MLP}_{\theta^{(2)}_{\mathtt{mlp}}}$ maps $h_i^{(1)}$ to

$$h_i^{(2)} = [\ ; f_1(x_i); \quad ; f_K(x_i); u_1; \quad ; u_K; 0; 1; t_i];$$

where $u_k = (1 \quad {}^{-1}c_k) \forall k \in [K]$. Clearly, $u_k \in [0, 1]$, and $u_k > 0$ if and only if c_k .

Step 3: construction of $\theta^{(3)}_{\mathtt{attn}}$. We define

$$\alpha_1 = 1 \quad (1 \quad u_1); \qquad \alpha_k = (1 \quad u_1 \quad \cdots \quad u_{k-1}) \quad (1 \quad u_1 \quad \cdots \quad u_k) \forall k \quad 2.$$

Clearly, $\alpha_k \quad 0$, and $\sum_k \alpha_k = 1$. Further,

$$\alpha_k > 0 \Rightarrow u_k > 0 \Rightarrow c_k \quad \Rightarrow L_k \quad \min_{k' \in [K]} L_{k'} + .$$

Therefore, it remains to construct $\theta^{(3)}_{\mathtt{attn}}$ that implements $\hat{f} = \sum_{k=1}^{K} \alpha_k f_k$ based on $[h_i^{(2)}]_i$. Notice that

$$\hat{f}(x_i) = (1) \quad f_1(x_i) + \sum_{k=1}^{K-1} (1 \quad u_1 \quad \cdots \quad u_{k-1}) \quad (f_k(x_i) \quad f_{k-1}(x_i)) \\ (1 \quad u_1 \quad \cdots \quad u_K) \quad f_K(x_i); \tag{41}$$

and hence we construct $\theta^{(3)}_{\mathtt{attn}}$ as follows: for every $k \in [K + 1]$ and $w \in \{0, 1\}$, we define matrices $Q_{k;w}, K_{k;w}, V_{k;w} \in \mathbb{R}^{D \times D}$ such that for all $k \in [K + 1]$

$$Q_{k;0} h_i^{(2)} = \begin{pmatrix} (f_k(x_i) + R) \ 1_k \\ 0 \end{pmatrix}; \quad Q_{k;1} h_i^{(2)} = \begin{pmatrix} (f_{k-1}(x_i) + R) \ 1_k \\ 0 \end{pmatrix};$$

$$K_{k;0} h_j^{(2)} = K_{k;1} h_j^{(2)} = \begin{bmatrix} 1 \\ u_1 \\ \vdots \\ u_{k-1} \\ 0 \end{bmatrix}; \qquad V_{k;0} h_j^{(2)} = e_{D-2} = V_{k;1} h_j^{(2)};$$

for all $i, j \in [N + 1]$, where we understand $f_0 = f_{K+1} = 0$ and 1_k is the k-dimensional vector with all entries being 1. By the structure of $h_i^{(2)}$, these matrices indeed exist, and further it is straightforward to check that they have norm bounds

$$\max_{k \in [K+1]; w \in \{0,1\}} \|Q_{k;w}\|_{\mathrm{op}} \quad KR; \qquad \max_{k \in [K+1]; w \in \{0,1\}} \|K_k\|_{\mathrm{op}} \quad 1; \qquad \sum_{k \in [K+1]; w \in \{0,1\}} \|V_{k;w}\|_{\mathrm{op}} \quad 2K + 2.$$

Now, for every $i, j \in [N + 1]$, $k \in [K + 1]$, $w \in \{0, 1\}$, we have

$$\left\langle Q_{k;w} h_i^{(2)}, K_{k;w} h_j^{(2)} \right\rangle = ((1 \quad u_1 \quad \cdots \quad u_{k-1})(f_{k-w}(x_i) + R)) \\ = (1 \quad u_1 \quad \cdots \quad u_{k-1}) \quad (f_{k-w}(x_i) + R);$$

where the last equality follows from $f_k(x_i) + R \quad 0 \forall k \in [K]$. Therefore,

$$\sum_{k \in [K+1]; w \in \{0,1\}} \left\langle Q_{m;k} h_i^{(2)}, K_{m;k} h_j^{(2)} \right\rangle V_{m;k} h_j^{(2)} \\ = \sum_{k=1}^{K} \Big[(1 \quad u_1 \quad \cdots \quad u_{k-1}) \quad (f_k(x_i) + R) \quad (1 \quad u_1 \quad \cdots \quad u_{k-1}) \quad (f_{k-1}(x_i) + R) \Big] e_{D-2}$$

$= \hat{p}(x_i)\ e_{D-2};$

where the last equality is due to (41). Thus letting the attention layer $\overset{(3)}{\text{attn}} = f(V_{k;w}; Q_{k;w}; K_{k;w})g_{(k;w)\in[K+1]\times\{0;1\}}$, we have

$$h_i^{(3)} = \left[\text{Attn}\ (H^{(2)})\right]_i = h_i + \frac{1}{N+1}\sum_{j=1}^{N+1}\sum_{k;w} D\, Q_{k;w}h_i^{(2)}; K_{k;w}h_j^{(2)}E\, V_{k;w}h_j^{(2)}$$

$$= h_i^{(2)} + \hat{p}(x_i)\ e_{D-2}$$

$$= [\ ; f_1(x_i);\ \ ; f_K(x_i); u_1;\ \ ; u_K; \hat{p}(x_i); 1; t_i].$$

This is the desired result. □

Now, we are ready to prove Theorem I.1.

Proof of Theorem I.1 As $\ell(\ ;\ )$ is $(\ =3; R; M; C)$-approximable by sum of relus, we can invoke Proposition I.1 to show that there exists a single attention layer $\overset{(1)}{\text{attn}}$ so that $\text{Attn}_{\overset{(1)}{\text{attn}}}$ maps

$$h_i \ \ !\ \ h_i' = [x_i; y_i';\ ; f_1(x_i);\ \ ; f_K(x_i); \hat{e}_{\text{val}}(f_1);\ \ ; \hat{e}_{\text{val}}(f_K); 0; 1; t_i];\qquad i \ 2\ [N+1];$$

for any input $H = [h_i]_i$ of the form described in Theorem I.1, and $\hat{e}_{\text{val}}(\ )$ is a functional such that $\max_k \hat{e}_{\text{val}}(f_k)\ \ \hat{b}_{\text{val}}(f_k)\ \ =3$.

Next, by the proof of Proposition I.2, there exists $(\overset{(1)}{\text{mlp}}; \overset{(2)}{\text{mlp}}; \overset{(3)}{\text{attn}})$ that maps

$$h_i' \ \ !\ \ h_i^{(3)} = \left[\ x_i; y_i';\ ; f_1(x_i);\ \ ; f_K(x_i);\ ;\ \sum_{k=1}^{K}{}_k f_k(x_i); 1; t_i\ \right];\qquad i \ 2\ [N+1];$$

where $\ = (\ _1;\ \ ;\ _K)\ 2\ ([K])$ and $_k > 0$ only when $\hat{e}_{\text{val}}(f_k)\ \ \min_{k?}\hat{e}_{\text{val}}(f_{k?}) + \ =3$. Using the fact that $\max_k j\hat{e}_{\text{val}}(f_k)\ \ \hat{b}_{\text{val}}(f_k)j\ \ =3$, we deduce that $\ $ is supported on $fk : \hat{b}_{\text{val}}(f_k)\ \ \min_{k_?\in[K]}\hat{b}_{\text{val}}(f_{k_?}) + \ g$.

Therefore, $\ = (\overset{(1)}{\text{attn}}; \overset{(1)}{\text{mlp}}; \overset{(2)}{\text{mlp}}; \overset{(3)}{\text{attn}})$ is the desired transformer, with

$$\max_{\ell\in[3]} M^{(\ell)}\ \ (M+3)K;\quad \max_{\ell\in[3]} D^{(\ell)}\ \ K^2+K+1;$$

and

$$\left\| \right\| \ \ \max\ \ 3R + \frac{2NKC}{jD_{\text{val}}j} + 3K + 1; K + 3 + \ ^{-1}; KR + 2K + 2$$

$$7KR + \frac{2NKC}{jD_{\text{val}}j} + \ ^{-1}.$$

This completes the proof. □

I.2 Proof of Theorem 11

We first restate Theorem 11 into the following version which provides additional size bounds for $\ $. For the simplicity of presentation, throughout this subsection and Appendix J, we denote $I_t = fi : (x_i; y_i)\ 2\ D_{\text{train}}g$, $I_v = fi : (x_i; y_i)\ 2\ D_{\text{val}}g$, $X_{\text{train}} = [x_i]_{i\in\mathcal{I}_t}$ to be the input matrix corresponding to the training split only, and $N_{\text{train}} = jD_{\text{train}}j$, $N_{\text{val}} = jD_{\text{val}}j$.

Theorem I.2. *For any sequence of regularizations $f\ _k g_{k\in[K]}$, $0\ \ $ with $\ := \max_k \frac{+\ _k}{+\ _k}$, $B_w > 0$, $\ > 0$, and $" < B_w/2$, suppose in input format (3) we have $D\ \ (Kd)$. Then there exists an L-layer transformer $\mathsf{TF}\ $ with*

$$L = d2\ \log(B_w/(2"))e + 4;\quad \max_{\ell\in[L]} M^{(\ell)}\ \ 3K+1;\quad \max_{\ell\in[L]} D^{(\ell)}\ \ K^2+K+1;$$

$$\|\theta\| \le O\left(KR + (\alpha + \beta)^{-1} + \frac{N}{N_{\text{val}}} + \gamma^{-1}\right); \qquad R := \max\{B_x B_w, B_y, 1\},$$

such that the following holds. On any input data (D, x_{N+1}) *such that the problem (ICRidge) is well-conditioned and has a bounded solution:*

$$\lambda_{\min}(X_{\text{train}}^\top X_{\text{train}}/N_{\text{train}}) \ge \alpha \quad \lambda_{\max}(X_{\text{train}}^\top X_{\text{train}}/N_{\text{train}}) \le \beta \quad ; \quad \max_{k\in[K]} \|w_{\text{ridge}}^{\lambda_k}(D_{\text{train}})\|_2 \le B_w/2, \tag{42}$$

TF^0 *approximately implements ridge selection: its prediction*

$$\hat{y}_{N+1} = \text{read}_y(\mathsf{TF}^0(H)) = \langle \hat{w}, x_{N+1}\rangle, \qquad \hat{w} = \sum_{k=1}^{K} \alpha_k \hat{w}_k$$

satisfies the following.

1. *For each* $k \in [K]$, $\hat{w}_k = \hat{w}_k(D_{\text{train}})$ *approximates the ridge estimator* $w_{\text{ridge}}^{\lambda_k}(D_{\text{train}})$, *i.e.* $\|\hat{w}_k - w_{\text{ridge}}^{\lambda_k}(D_{\text{train}})\|_2 \le \varepsilon$.
2. $\alpha = (\alpha_1, \ldots, \alpha_K) \in \Delta([K])$ *so that*
$$\alpha_k > 0 \text{ only if } \hat{L}_{\text{val}}(\hat{w}_k) \le \min_{k'\in[K]} \hat{L}_{\text{val}}(\hat{w}_{k'}) + \varepsilon.$$

In particular, if we set $\varepsilon' = 2(B_x B_w + B_y)B_x\varepsilon + \varepsilon$, then it holds that[8]

$$\text{dist}\left(\hat{w}, \text{conv}\{\hat{w}_{\text{ridge};\text{train}}^{\lambda_k} : \hat{L}_{\text{val}}(\hat{w}_{\text{ridge};\text{train}}^{\lambda_k}) \le \min_{k'\in[K]} \hat{L}_{\text{val}}(\hat{w}_{\text{ridge};\text{train}}^{\lambda_{k'}}) + \varepsilon'\}\right) \le \varepsilon'';$$

where we denote $\hat{w}_{\text{ridge};\text{train}}^{\lambda_k} := w_{\text{ridge}}^{\lambda_k}(D_{\text{train}})$.

To prove Theorem I.2, we first show that, for the squared validation loss, there exists a 3-layer transformer that performs predictor selection based on the *exactly* evaluated $\hat{L}_{\text{val}}(f_k)$ for each $k \in [K]$. (Proof in Appendix I.2.1.)

Theorem I.3 (Square-loss version of Theorem I.1)**.** *Consider the squared validation loss*

$$\hat{L}_{\text{val}}(f) := \frac{1}{2|D_{\text{val}}|} \sum_{(x_i,y_i)\in\mathcal{D}_{\text{val}}} (f(x_i) - y_i)^2.$$

Then there exists a 3-layer transformer TF *with*

$$\max_{\ell\in[3]} M^{(\ell)} \le 2K + 2; \quad \max_{\ell\in[3]} D^{(\ell)} \le K^2 + K + 1; \quad \|\theta\| \le 7KR + \frac{2N}{|D_{\text{val}}|} + \gamma^{-1};$$

such that for any input H *that takes form*

$$h_i = [x_i; y_i'; \ldots; f_1(x_i); \ldots; f_K(x_i); 0_K; \ldots; 1; t_i];$$

where TF *outputs* $h_{N+1} = [x_{N+1}; \hat{p}(x_{N+1}); \ldots; 1; 0]$, *where the predictor* $\hat{p} : \mathbb{R}^d \to \mathbb{R}$ *is a convex combination of* $\{f_k : \hat{L}_{\text{val}}(f_k) \le \min_{k'\in[K]} \hat{L}_{\text{val}}(f_{k'}) + \varepsilon\}$. *As a corollary, for any convex risk* $L : (\mathbb{R}^d \to \mathbb{R}) \to \mathbb{R}$, $\hat{p}$ *satisfies*

$$L(\hat{p}) \le \min_{k'\in[K]} L(f_{k'}) + \max_{k\in[K]} \hat{L}_{\text{val}}(f_k) - L(f_k) + \varepsilon.$$

Proof of Theorem I.2 First, by the proof[9] of Theorem 4 and Proposition B.6, for each $k \in [K]$, there exists a $T = L + 3$ layer transformer $\mathsf{TF}^{(1:T)}$ such that $\mathsf{TF}^{(1:T)}$ maps

$$h_i \to h_i^{(T)} = [x_i; y_i'; \ldots; \langle \hat{w}_1, x_i\rangle; \ldots; \langle \hat{w}_K, x_i\rangle; 0_K; 1; t_i];$$

[8] This is because $\hat{L}_{\text{val}}(w)$ is $(B_x B_w + B_y)B_x$-Lipschitz w.r.t. $w \in B_2(B_w)$.

[9] Technically, an adapted version where the underlying ICGD mechanism operates on the training split (with $t_i = 1$) with size N_{train} instead of on all N training examples, which only changes $\|\theta\|$ by at most a constant factor, and does not change the number of layers and heads.

so that if (42) holds, we have $\|\widehat{w} - w_{\text{ridge}}^{k}\|_2 \le \varepsilon$ and $\widehat{w} \in B_2(B_w)$.

Next, by Theorem I.3, there exists a 3-layer transformer $\mathrm{TF}^{(T+1:T+3)}$ that outputs

$$h_{N+1}^{(T+3)} = [x_{N+1}; h\widehat{w}; x_{N+1}\rangle; \ldots; 1; t_i];$$

where $\widehat{w} = \sum_{k=1}^{K} \alpha_k \widehat{w}_k$, $\alpha = (\alpha_1, \ldots, \alpha_K) \in \Delta([K])$ so that

$$\alpha_k > 0 \text{ only if } \widehat{\ell}_{\text{val}}(\widehat{w}_k) \le \min_{k' \in [K]} \widehat{\ell}_{\text{val}}(\widehat{w}_{k'}) + \varepsilon.$$

This is the desired result. □

I.2.1 Proof of Theorem I.3

Similar to the proof of Proposition 10, Theorem I.3 is a direct corollary by combining Proposition I.3 with Proposition I.2.

Proposition I.3 (Evaluation layer for the squared loss)**.** *There exists an attention layer* TF *with* $2K$ *heads and* $\|\cdot\|_{\infty} \le 3R + 2NK/|D_{\text{val}}|$ *such that* TF *maps*

$$h_i = [\ldots; f_1(x_i); \ldots; f_K(x_i); 0_K; \ldots; 1; t_i]$$
$$\to h'_i = [\ldots; f_1(x_i); \ldots; f_K(x_i); \widehat{\ell}_{\text{val}}(f_1); \ldots; \widehat{\ell}_{\text{val}}(f_K); \ldots; 1; t_i]; \quad i \in [N+1].$$

Proof of Proposition I.3. For every $k \in [K]$, we define matrices $Q_{m;k}, K_{m;k}, V_{m;k} \in \mathbb{R}^{D \times D}$ such that for all $i, j \in [N+1]$,

$$Q_{k;0} h_i = \begin{bmatrix} 1 \\ 2 \\ 0 \end{bmatrix}; \quad Q_{k;1} h_i = \begin{bmatrix} 1 \\ 2 \\ 0 \end{bmatrix}; \quad K_{k;0} h_j = K_{k;1} h_j = \begin{bmatrix} f_k(x_j) \\ y_j \\ R(1+t_j) \end{bmatrix};$$

$$V_{k;0} h_j = V_{k;1} h_j = \frac{(N+1)}{2|D_{\text{val}}|} (f_k(x_j) - y_j) e_{D-(K-k)-3}.$$

As the input has structure $h_i = [x_i; y'_i; \ldots; f_1(x_i); \ldots; f_K(x_i); 0_{K+1}; 1; t_i]$, these matrices indeed exist, and further it is straightforward to check that they have norm bounds

$$\max_{k \in [K]; w \in \{0;1\}} \|Q_{k;w}\|_{\text{op}} \le 3; \quad \max_{k \in [K]; w \in \{0;1\}} \|K_{k;w}\|_{\text{op}} \le 1 + R; \quad \sum_{k \in [K]; w \in \{0;1\}} \|V_{k;w}\|_{\text{op}} \le \frac{K(N+1)}{|D_{\text{val}}|}.$$

Now, for every $i, j \in [N+1]$, we have

$$\sum_{w \in \{0;1\}} (\langle Q_{k;w} h_i, K_{k;w} h_j \rangle) V_{k;w} h_j$$
$$= [-(f_k(x_j) - y_j - 2R(1+t_j)) \quad (y_j - f_k(x_j) - 2R(1+t_j))] \frac{(N+1)}{2|D_{\text{val}}|} (f_k(x_j) - y_j) e_{D-(K-k)-3}$$
$$= 1\{t_j = 1\} [-(f_k(x_j) - y_j) \quad (y_j - f_k(x_j))] \frac{(N+1)}{2|D_{\text{val}}|} (f_k(x_j) - y_j) e_{D-(K-k)-3}$$
$$= 1\{t_j = 1\} \frac{(N+1)}{2|D_{\text{val}}|} (f_k(x_j) - y_j)^2 e_{D-(K-k)-3};$$

where the second equality follows from the bound $|f_k(x_j) - y_j| \le 2R$, so that the relus equals 0 if $t_j = 0$. Thus letting the attention layer $\mathrm{Attn} = f(V_{k;w}, Q_{k;w}, K_{k;w})_{(k;w) \in [K] \times \{0;1\}}$, we have

$$\mathrm{Attn}_i = [\mathrm{Attn}(H)]_i = h_i + \frac{1}{N+1} \sum_{j=1}^{N+1} \sum_{k;w} (\langle Q_{k;w} h_i, K_{k;w} h_j \rangle) V_{k;w} h_j$$

$$= \mathsf{h}_i + \frac{1}{2jD_{\text{val}}j} \sum_{j=1}^{N+1} \sum_{k=1}^{K} (f_k(\mathsf{x}_j) \quad y_j)^2 \quad 1ft_j = \quad 1ge_{D-(K-k)-3}$$

$$= \mathsf{h}_i + \sum_{k=1}^{K} @\frac{1}{2jD_{\text{val}}j} \sum_{(\mathbf{x}_j;y_j)\in\mathcal{D}_{\text{val}}} 1 \quad (f_k(\mathsf{x}_j) \quad y_j)^2 A e_{D-(K-k)-3}$$

$$= \mathsf{h}_i + \sum_{k=1}^{K} b_{\text{val}}(f_k) \quad e_{D-(K-k)-3}$$

$$= [\mathsf{x}_i; y'_i; \ ; f_1(\mathsf{x}_i); \quad ; f_K(\mathsf{x}_i); 0_{K+1}; 1; t_i] + [0_{D-K-3}; b_{\text{val}}(f_1); \quad ; b_{\text{val}}(f_K); 0; 0; 0]$$

$$= [\mathsf{x}_i; y'_i; \ ; f_1(\mathsf{x}_i); \quad ; f_K(\mathsf{x}_i); b_{\text{val}}(f_1); \quad ; b_{\text{val}}(f_K); 0; 1; t_i]; \qquad i \ 2 \ [N+1]:$$

This is the desired result. □

I.3 Proofs for Section 4.2

I.3.1 Proof of Lemma 13

It is straightforward to check that the binary type check $\quad : \mathsf{R} \ ! \ \mathsf{R}$ can be expressed as a linear combination of 6 relu's (recalling $\quad ( ) = \mathsf{ReLU}( )$):

$$(y) = \quad \frac{y+"}{"} \quad 2 \quad \frac{y}{"} \quad + \quad \frac{y \quad "}{"} \quad + \quad \frac{y \quad (1 \quad ")}{"} \quad 2 \quad \frac{y \quad 1}{"} \quad + \quad \frac{y \quad (1+")}{"}$$

$$=: \sum_{m=1}^{6} a_m \quad (b_m y + c_m);$$

with $\sum_m ja_m j \ = \ 8="$, $\max_m \max fjb_m j; jc_m jg \quad 2$. We can thus construct an attention layer $= f(\mathsf{Q}_m; \mathsf{K}_m; \mathsf{V}_m)g_{m=1}^{6}$ with 6 heads such that

$$\mathsf{Q}_m \mathsf{h}_i = [b_m; c_m; 0_{D-2}]; \quad \mathsf{K}_m \mathsf{h}_j = [y_j; 1; 0_{D-2}]; \quad \mathsf{V}_m \mathsf{h}_j = \quad \frac{N+1}{N} a_m \quad t_j; 0_{D-1} \ ;$$

which gives that for every $i \ 2 \ [N+1]$,

$$\sum_{m=1}^{6} \frac{1}{N+1} \sum_{j\in[N+1]} (h\mathsf{Q}_m \mathsf{h}_i; \mathsf{K}_m \mathsf{h}_j i)[\mathsf{V}_m \mathsf{h}_j]_1$$

$$= \sum_{m=1}^{6} \frac{1}{N} \sum_{j=1}^{N} (b_m y_j + c_m) a_m = \frac{1}{N} \sum_{j=1}^{N} (y_j) = \quad {}^{\text{binary}}(D):$$

Further, we have $\quad 18=" = O(1=")$. This is the desired result. □

By composing the above attention layer with one additional layer (with 2 heads) that implement the following function

$$(2(t \quad 1=2)) \quad (2(t \quad 1));$$

on the output ${}^{\text{binary}}(D)$, we directly obtain the following corollary.

Corollary I.1 (Thresholded binary test)**.** *There exists a two-layer attention-only transformer with* $\max_{`\in[2]} M^{(`)} \quad 6$ *and* $\quad O(1=")$ *that exactly implements the thresholded binary test*

$${}^{\text{binary}}_{\text{thres}}(D) := \begin{cases} 1; & \text{if } {}^{\text{binary}}(D) \quad 1; \\ 0; & \text{if } {}^{\text{binary}}(D) \quad \frac{1}{2}; \\ \text{linear interpolation}; & \text{o.w.} \end{cases} \tag{43}$$

at every token $i \ 2 \ [N+1]$, *where we recall the definition of* ${}^{\text{binary}}$ *in Lemma 13.*

I.3.2 Formal statement and proof of Proposition 14

We say a distribution P_y on R is $(C; \varepsilon_0)$-not-concentrated around $\{0; 1\}$ if

$$P_y([\ \ \varepsilon; \varepsilon] \cup [1\ \ \varepsilon; 1+\varepsilon]) \ \ C\varepsilon$$

for all $\varepsilon \in (0; \varepsilon_0]$. A sufficient condition is that the density p_y is upper bounded by C within $[\ \ \varepsilon_0; \varepsilon_0] \cup [1\ \ \varepsilon_0; 1+\varepsilon_0]$.

Throughout this section, let $\sigma_{\log}(t) := (1+e^{-t})^{-1}$ denote the sigmoid activation, and let $w_{\log}$ denote the solution to the in-context logistic regression problem, i.e. (ICGLM) with $g(\cdot) = \sigma_{\log}(\cdot)$.

Proposition I.4 (Adaptive regression or classification; Formal version of Proposition 14)**.** *For any $B_w > 0$, $\varepsilon \ \ B_x B_w$=10, $0 <$ with $:= \ = $, and any $(C; \varepsilon_0)$, there exists a L-layer attention-only transformer with*

$$L \ \ O\left(\log \frac{B_x B_w}{\varepsilon}\right); \ \max_{\ell \in [L]} M^{(\ell)} \ \ O\left(1 + \frac{B_x^4}{2}\right)^{\varepsilon^{-2}}; \ \|w\| \|w\| \ \ O\ R + \frac{1}{-} + \frac{1}{\varepsilon}$$

(with $R := \max\{B_x B_w; B_y; 1\}$, and ε depending only on $(C; \varepsilon_0)$) such that the following holds. Suppose the input format is (3) with dimension $D \ \ 3d+4$.

On any classification *instance $(D; x_{N+1})$ (such that $\{y_i\}_{i\in[N]} \ \ \{0; 1\}$) that is well-conditioned for logistic regression in the sense of (35), it outputs $\hat{y}_{N+1}$ that ε-approximates the prediction of in-context logistic regression:*

$$|\hat{y}_{N+1} \ \ \sigma_{\log}(\langle x_{N+1}; w_{\log}\rangle)| \ \ \varepsilon.$$

On the contrary, for regression problems*, i.e. any in-context distribution P whose marginal P_y is $(C; \varepsilon_0)$-not-concentrated around $\{0; 1\}$, with probability at least $1 \ \ \exp(\ \ cN)$ over D (where $c > 0$ depends only on $(C; \varepsilon_0)$), $\hat{y}_{N+1}$ ε-approximates the prediction of in-context least squares if the data is well-conditioned:*

$$|\hat{y}_{N+1} \ \ \langle x_{N+1}; w_{\mathrm{LS}}\rangle| \ \ \varepsilon \quad \textit{whenever } D \textit{ satisfies (5) with } \ = 0;$$

where w_{LS} denotes the in-context least squares estimator, i.e. (ICRidge) with $= 0$.

Proof. The result follows by combining the binary test in Corollary I.1 with Theorem 4 and Theorem G.1. By those results, there exists three attention-only transformers LS; $\log$; bin, with (below $L_g; C_g = (1)$ for $g = \sigma_{\log}(\cdot)$)

$$L_{\mathrm{LS}} \ \ O\left(\log \frac{B_x B_w}{\varepsilon}\right); \ \max_{\ell \in [L_{\mathrm{LS}}]} M_{\mathrm{LS}}^{(\ell)} \ \ 3; \ \|w_{\mathrm{LS}}\| \ \ O\ R + \frac{1}{-}\ !\ !;$$

$$L_{\log} \ \ O\left(\log \frac{L_g B_x B_w}{\varepsilon}\right); \ \max_{\ell \in [L_{\log}]} M_{\log}^{(\ell)} \ \ O\ C_g^2\ 1 + \frac{L_g^2 B_x^4}{2}\Big)^{\varepsilon^{-2}}; \ \|w_{\log}\| \ \ O\ R + \frac{C_g}{-};$$

$$L_{\mathrm{bin}} = 2; \ \max_{\ell \in [2]} M_{\mathrm{bin}}^{(\ell)} \ \ 6; \ \|w_{\mathrm{bin}}\| \ \ O(1/\varepsilon);$$

that outputs prediction $\hat{y}_{N+1}^{\mathrm{LS}}$, $\hat{y}_{N+1}^{\log}$ (at the $(N+1)$-th token) and $\mathrm{thres}_{\mathrm{binary}}(D)$ (at every token) respectively, which satisfy

$$|\hat{y}_{N+1}^{\log} \ \ \sigma_{\log}(\langle x_{N+1}; w_{\log}\rangle)| \ \ \varepsilon;$$
$$|\hat{y}_{N+1}^{\mathrm{LS}} \ \ \langle x_{N+1}; w_{\mathrm{LS}}\rangle| \ \ \varepsilon.$$

when the corresponding well-conditionednesses are satisfied. In particular, we can make $w_{\log}$ well-defined on non-binary data, by multiplying $\mathrm{thres}_{\mathrm{binary}}(D)$ onto the x_i's (which can be implemented by slightly modifying $\log$ without changing the order of the number of layers, heads, and norms) so that $w_{\log} = 0$ on any data where $\mathrm{thres}_{\mathrm{binary}}(D) = 0$.

By joining LS and $\log$ using Proposition B.6, concatenating with bin before, and concatenating with one additional attention layer with 2 heads after to implement

$$\mathrm{thres}_{\mathrm{binary}}(D)\hat{y}_{N+1}^{\log} + \ 1 \quad \mathrm{thres}_{\mathrm{binary}}(D) \ \hat{y}_{N+1}^{\mathrm{LS}}; \tag{44}$$

we obtain a single transformer with

$$L = O\left(\log \frac{B_x B_w}{\varepsilon}\right), \quad \max_{\ell \in [L]} M^{(\ell)} = O\left(1 + \frac{B_x^4}{2}\right)^{\varepsilon^{-2}}, \quad \|W_{\mathrm{LS}}\| = O\left(R + \frac{1}{\gamma} + \frac{1}{\varepsilon}\right),$$

which outputs (44) as its prediction (at the location for $\hat{y}_{N+1}$).

It remains to show that (44) reduces to either one of $\hat{y}^{\log}_{N+1}$ or $\hat{y}^{\mathrm{LS}}_{N+1}$. When the data are binary ($y_i \in \{0,1\}$), we have $\mathrm{thres}^{\mathrm{binary}}(D) = 1$ and $\mathrm{thres}^{\mathrm{binary}}(D) = 1$, in which case (44) becomes exactly $\hat{y}^{\log}_{N+1}$. By contrast, when data is sampled from a distribution that is (C, ε_0)-not-concentrated around $\{0,1\}$, we have for any fixed $\varepsilon \le \varepsilon_0 \wedge \frac{1}{4C}$ that, letting $B_\varepsilon := [0, \varepsilon] \cup [1-\varepsilon, 1+\varepsilon]$ and $p_\varepsilon := P_y(B_\varepsilon) \le C\varepsilon \le \frac{1}{4}$, by Hoeffding's inequality,

$$P\left(\mathrm{thres}^{\mathrm{binary}}(D) \neq 0\right) = P\left(\mathrm{thres}^{\mathrm{binary}}(D) \ge \frac{1}{2}\right) = P\left(\frac{1}{N}\sum_{i=1}^{N} 1\{y_j \in B_\varepsilon\} \ge \frac{1}{2}\right)$$
$$\le \exp\left(-c(1/2 - p_\varepsilon)^2 N\right) \le \exp(-c'N),$$

where $c' > 0$ is an absolute constant. On the event $\mathrm{thres}^{\mathrm{binary}}(D) = 0$ (which happens with probability at least $1 - \exp(-c'N)$), (44) becomes exactly $\hat{y}^{\mathrm{LS}}_{N+1}$. This finishes the proof. □

I.4 Linear correlation test and application

In this section, we give another instantiation of the pre-ICL testing mechanism by showing that the transformer can implement a *linear correlation test* that tests whether the correlation vector $\mathbb{E}[xy]$ has a large norm. We then use this test to construct a transformer to perform "confident linear regression", i.e. output a prediction from linear regression only when the signal-to-noise ratio is high.

For any fixed parameters $\gamma_{\min}, B_w^? > 0$, consider the linear correlation test over data D defined as

$$\mathrm{lin}(D) := \frac{1}{\gamma_{\min}^2 (B_w^?)^2/2} \left[\|\hat{\beta}\|_2^2 - (\gamma_{\min} B_w^?/4)^2 - \|\hat{\beta}\|_2^2 - (3\gamma_{\min} B_w^?/4)^2\right]$$
$$= \begin{cases} 0, & \|\hat{\beta}\|_2^2 \le (\gamma_{\min} B_w^?/4)^2, \\ 1, & \|\hat{\beta}\|_2^2 \ge (3\gamma_{\min} B_w^?/4)^2, \\ \text{linear interpolation}, & \text{o.w.}, \end{cases} \tag{45}$$

where $\hat{\beta} = T(D) := \frac{1}{N}\sum_{i=1}^{N} x_i y_i$.

Recall that $\mathrm{ReLU}(\cdot) = \max(\cdot, 0)$ above denotes the relu activation.

We show that lin can be exactly implemented by a 3-layer transformer.

Lemma I.1 (Expressing lin by transformer)**.** *There exists a 3-layer attention-only transformer* TF *with at most* 2 *heads per layer and* $\|\mathrm{TF}\| \le O(1 + \gamma_{\min}^{-2}(B_w^?)^2)$ *such that on input sequence* H *of the form (3) with* $D \ge 2d + 4$*, the transformer* exactly *implements* lin*: it outputs* $\hat{H}$ *such that* $\hat{h}_i = [x_i, y_i t_i, \ldots, \mathrm{lin}(D), 1]$ *for all* $i \in [N+1]$.

Proof. We begin by noting the following basic facts:

- Identity function can be implemented exactly by two ReLUs: $t = \mathrm{ReLU}(t) - \mathrm{ReLU}(-t)$.
- Squared ℓ_2 norm can be implemented exactly by a single attention head (assuming every input h_i contains the same vector g): $\|g\|_2^2 = \mathrm{ReLU}(\langle g, g\rangle)$.

We construct the transformer as follows.

Layer 1: Use 2 heads to implement $\hat{\beta} = \frac{1}{N}\sum_{i=1}^{N} x_i y_i$, where $V^{(1)}_{\{1,2\}} h_j = [x_j; 0_{D-d}]$, $Q^{(1)}_{\{1,2\}} h_i = [\frac{N+1}{N}; 0_{D-1}]$, and $K^{(1)}_{\{1,2\}} h_j = [y_j t_j; 0_{D-1}] = [y_j 1\{j < N+1\}; 0_{D-1}]$ (where we recall $t_j = 1\{j < N+1\}$ and note that $y_j t_j$ corresponds exactly to the location for y_j in H, cf. (3)).

By manipulating the output dimension in $V^{(1)}$, write the result $\|x\|_2^2$ into blank memory space with dimension d at every token $i \in [N+1]$.

Layer 2: Use a single head to compute $\|x\|_2^2$: $Q_1^{(2)} h_i^{(1)} = [\|x\|_2^2; 0_{D-d}]$, $K_1^{(2)} h_j^{(1)} = [\|x\|_2^2; 0_{D-d}]$, and $V_1^{(2)} h_j^{(1)} = [1; 0_{D-1}]$. By manipulating the output dimension in $V^{(2)}$, write the result $\|x\|_2^2$ into blank memory space with dimension 1 at every token $i \in [N+1]$. After layer 2, we have $h_i^{(3)} = [x_i; y_i t_i; \ldots; \|x\|_2^2; \ldots; 1]$.

Layer 3: Use 2 heads to implement two ReLU functions with bias: $\|x\|_2^2 \mapsto \frac{1}{B-A}(\mathrm{ReLU}(\|x\|_2^2 - A) - \mathrm{ReLU}(\|x\|_2^2 - B))$. The two query (or key) matrices contain values A and B. In our problem we take

$$A = (\min B_w^\star / 4)^2, \quad B = (3 \min B_w^\star / 4)^2,$$

so that the above ReLU function implements $\mathrm{lin}(D)$ exactly. Write the result into a blank memory space with dimension 1. We finish the proof by noting that $\|h\| \le O(1 + \lambda_{\min}^2 (B_w^\star)^2)$. □

Statistical guarantee for lin We consider the following well-posedness assumption for the linear correlation test lin. Note that, similar as Assumption A, the assumption does not require the data to be generated from any true linear model, but rather only requires some properties about the best linear fit $w_P^\star$, as well as sub-Gaussianity conditions.

Assumption D (Well-posedness for linear correlation test)**.** *We say a distribution P on $\mathbb{R}^d \times \mathbb{R}$ is well-posed for linear independence tests, if $(x, y) \sim P$ satisfies*

(1) $\|x\|_2 \le B_x$ and $|y| \le B_y$ almost surely;

(2) The covariance $\Sigma_P := \mathbb{E}_P[xx^\top]$ satisfies $\lambda_{\min} I_d \preceq \Sigma_P \preceq \lambda_{\max} I_d$, with $0 < \lambda_{\min} \le \lambda_{\max}$, and $\kappa := \lambda_{\max} / \lambda_{\min}$.

(3) The whitened vector $\Sigma_P^{-1/2} x$ is K^2-sub-Gaussian for some $K \ge 1$.

(4) The best linear predictor $w_P^\star := \mathbb{E}_P[xx^\top]^{-1} \mathbb{E}_P[xy]$ satisfies $\|w_P^\star\|_2 \le \overline{B_w^\star}$.

(5) The label y is σ^2-sub-Gaussian.

(6) The residual $z := y - \langle x, w_P^\star \rangle$ is σ^2-sub-Gaussian with probability one (over x).

The following results states that lin achieves high power as long as the sample size is high enough, and the signal $\|w_P^\star\|_2$ is either sufficiently high or sufficiently low.

Proposition I.5 (Power of linear correlation test)**.** *Suppose distribution P satisfies Assumption D with parameters $\lambda_{\min}, \lambda_{\max}, \overline{B_w^\star}$. Then, for the linear correlation test lin with parameters $(\lambda_{\min}, B_w^\star)$ with $B_w^\star \le \overline{B_w^\star}$ and any $N \ge C \max\{K^4, \frac{\kappa K^2 \sigma^2}{(B_w^\star)^2 \lambda_{\min}^2}\} d$, we have*

1. *If $\|w_P^\star\|_2 \ge B_w^\star$, then with probability at least $1 - \exp(-d)$ over D, we have $\mathrm{lin}(D) = 1$.*
2. *If $\|w_P^\star\|_2 \le \frac{\lambda_{\min}}{10 \lambda_{\max}} B_w^\star$, then with probability at least $1 - \exp(-d)$ over D, we have $\mathrm{lin}(D) = 0$.*

Proof. For any P satisfying Assumption D, note that $\mathbb{E}[xz] = \mathbb{E}[x(y - \langle w_P^\star, x \rangle)] = 0$ by construction. Therefore, by standard sub-Gaussian and sub-exponential concentration combined with union bound, the following events hold simultaneously with probability at least $1 - \exp(-d)$:

$$0.9 \Sigma_P \preceq \hat{\Sigma} = \frac{1}{N} \sum_{i=1}^N x_i x_i^\top \preceq 1.1 \Sigma_P \text{ as } N \ge C(dK^4) \text{ by (27)};$$

$$\left\| \frac{1}{N} \sum_{i=1}^N x_i z_i \right\|_{\Sigma_P^{-1}}^2 \le C \lambda_{\max} K^2 \sigma^2 \left(\frac{d}{N} + \frac{d}{N} \right)$$

$$\le C \lambda_{\max} K^2 \sigma^2 \frac{d}{N} \le \frac{\lambda_{\min} B_w^\star}{8}, \text{ as } N \ge C \frac{\lambda_{\max} d K^2 \sigma^2}{(B_w^\star)^2 \lambda_{\min}^2}.$$

On the above event, we have

$$\left\| \frac{1}{N}\sum_{i=1}^{N} x_i(\langle x_i, w_P^\star\rangle + z_i) \right\|_2 = \left\| w_P^\star + \frac{1}{N}\sum_{i=1}^{N} x_i z_i \right\|_2 .$$

Therefore, in case 1, we have

$$\left\| \frac{1}{N}\sum_{i=1}^{N} x_i z_i \right\|_2 \le 0.9 \min \|w_P^\star\|_2 \le \frac{\min B_w^\star}{8} \le \frac{3\min B_w^\star}{4}.$$

In case 2, we have

$$\left\| w_P^\star + \frac{1}{N}\sum_{i=1}^{N} x_i z_i \right\|_2 \ge \max \frac{\min B_w^\star}{10\max} + \frac{\min B_w^\star}{8} \ge \frac{\min B_w^\star}{4}.$$

The proof is finished by recalling the definition of Ψ^{lin} in (45), so that $\Psi^{\mathrm{lin}}(D) = 1$ if $\|\hat{w}\|_2 \ge 3\min B_w^\star/4$, and $\Psi^{\mathrm{lin}}(D) = 0$ if $\|\hat{w}\|_2 \le \min B_w^\star/4$. □

Application: Confident linear regression By directly composing the linear correlation test in Lemma I.1 with the transformer construction in Corollary 5 (using an argument similar as the proof of Proposition I.4), and using the power of the linear correlation test Proposition I.5, we immediately obtain the following result, which outputs a prediction from (approximately) least squares if $\hat{b} := \Psi^{\mathrm{lin}}(D) = 1$, and abstains from predicting if $\hat{b} = 0$. This can be viewed as a form of "confident linear regression", where the model predicts only if it thinks the linear signal is strong enough.

Proposition I.6 (Confident linear regression)**.** *For any $B_w > 0$, $0 < B_w^\star \le \overline{B_w^\star}$, $0 < \min \le \max$, $\varepsilon \le B_x B_w/10$, $0 < \le$ with $:= =$, there exists a L-layer attention-only transformer with*

$$L \le O\left(\log \frac{B_x B_w}{\varepsilon}\right); \quad \max_{\ell\in[L]} M^{(\ell)} \le O(1); \quad \|\cdot\| \le O\left(R + \frac{1}{-} + {}^{2}_{\min}(B_w^\star)^2\right)$$

(with $R := \max\{B_x B_w; B_y; 1\}$) such that the following holds. Let $N \ge \max\{K^4; \frac{\max K^2\ {}^2}{(B_w^\star)^2\ {}^2_{\min}}\}\ d$. Suppose the input format is (3) with dimension $D \ge 2d + 4$. Let ICL instance $(D; x_{N+1})$ be drawn from any distribution $\mathbb{P}$ satisfying Assumption D. Then the transformer outputs a 2-dimensional prediction (within the test token $\hat{\ }$)

$$(\hat{y}_{N+1}; \hat{b}) \in \mathbb{R} \quad \{0; 1\}$$

such that the following holds:

1. *If $\|w_P^\star\|_2 \le B_w^\star$, then with probability at least 1 over D, we have $|\hat{y}_{N+1} - \langle \hat{w}_{\mathrm{LS}}; x_{N+1}\rangle| \le \varepsilon$, and $\hat{b} = 1$ if D is in addition well-conditioned for least squares (in the sense of (5) with $= 0$).*
2. *If $\|w_P^\star\|_2 \le \frac{\min}{10\max} B_w^\star$, then with probability at least 1 over D, we have $\hat{y}_{N+1} = 0$ and $\hat{b} = 0$.*

J Proof of Theorem 12: Noisy linear model with mixed noise levels

For each fixed $k \in [K]$, we consider the following data generating model $\mathbb{P}_k$, where we first sample $\mathbb{P} = \mathbb{P}_{w_?;\ k}$ from $w_? \sim N(0; I_d/d)$, and then sample data $\{(x_i; y_i)\}_{i\in[N+1]} \overset{\text{iid}}{\sim} \mathbb{P}_{w_?;\ k}$ as

$$\mathbb{P}_{w_?;\ k}: x_i \sim N(0; I_d); \quad y_i = \langle x_i; w_?\rangle + \varepsilon_i; \quad \varepsilon_i \sim N(0; {}^{2}_{k}).$$

Also, recall that the Bayes optimal estimator on $\mathbb{P}_k$ is given by $\hat{y}^{\mathrm{Bayes}}_{N+1} = \left\langle w^{\ k}_{\mathrm{ridge}}(D); x_{N+1}\right\rangle$ with $_{\mathrm{ridge}}\ _k = {}^{2}_{k}d/N$, and the Bayes risk on $\mathbb{P}_k$ is given by

$$\mathrm{BayesRisk}_k := \inf_{\mathcal{A}} \mathbb{E}_k\ \tfrac{1}{2}(\mathcal{A}(D)(x_{N+1}) - y_{N+1})^2 = \mathbb{E}_k\left[\tfrac{1}{2}\left(\hat{y}^{\mathrm{Bayes}}_{N+1} - y_{N+1}\right)^2\right].$$

Recall that in Section 4.1.1, we consider a mixture law P that generates data from P_k with k . It is clear that we have (pushing $\inf_{\mathcal{A}}$ into $\mathsf{E}_{k\sim}$ does not increase the value) we have

$$\text{BayesRisk} \quad \mathsf{E}_{k\sim}[\text{BayesRisk}_k];$$

i.e., the Bayes risk can only be greater if we consider a mixture of models. In other words, if a transformer can achieve near-Bayes ICL on each meta-task P_k, then it can perform near-Bayes ICL on any meta-task which is a mixture of P_k with k . Therefore, to prove Theorem 12, it suffices to show the following (strengthened) result.

Theorem J.1 (Formal version of Theorem 12)**.** *Suppose that* $N \quad 0{:}1d$ *and we write* $_{\max} = \max_k f_k; 1g;$ $_{\min} = \min_k f_k; 1g.$ *Suppose in input format (3) we have* $D \quad (Kd)$*. Then there exists a transformer with*

$$L \quad O\left(\tfrac{-2}{\min}\log(N=\ _{\min})\right); \qquad \max_{`\in[L]} M^{(`)} \quad O(K); \qquad \max_{`\in[L]} D^{(`)} \quad O(K^2);$$

$$\text{jjwjj} \quad O(\ _{\max} Kd\log(N));$$

such that for any $k \ 2\ [K]$*, it holds that*

$$\mathsf{E}_k\left[\frac{1}{2}(y_{N+1} \quad \text{by}_{N+1})^2\right] \quad \text{BayesRisk}_k + \mathcal{O}\left(\frac{\max^{2=3}}{\min^{2=3}}\frac{\log K}{N}\right)^{1=3}$$

if we choose $N_{\text{val}} := jD_{\text{val}}j \quad N^{2=3}[\log K]^{1=3}$.

The core of the proof of Theorem J.1 is to show that any estimator $\hat{\mathbf{w}}$ that achieves small validation loss $\hat{L}_{\text{val}}$ must achieve small population loss.

Throughout the rest of this section, recall that we define $N_{\text{train}} = jD_{\text{train}}j; N_{\text{val}} = jD_{\text{val}}j,\ I_{\text{train}} = fi : (x_i; y_i)\ 2\ D_{\text{train}}g,\ I_{\text{val}} = fi : (x_i; y_i)\ 2\ D_{\text{val}}g$, and $X_{\text{train}} = [x_i]_{i\in\mathcal{I}_{\text{train}}}$.

J.1 Proof of Theorem J.1

Fix parameters $; "; \ > 0$ and a large universal constant C_0. Let us set

$$= \max\left\{0; 1=2 \quad \sqrt{d=N_{\text{train}}}\right\}^2; \qquad = 25;$$

$$B_w^? = 1 + C_0\sqrt{\frac{\log(N)}{d}}; \qquad B_w = C_0(B_w^? + \ _{\max}=\ _{\min});$$
$$B_x = C_0\sqrt{d\log(N)}; \qquad B_y = C_0(B_w^? + \ _{\max})\sqrt{\log(N)};$$

Then, we define good events similarly to the proof of Corollary 6 (Appendix F.4):

$$\begin{aligned}
E \ &= fk\mathbf{w}_?k_2 \quad B_w^?; k"k_2 \quad 2\ _{\max}\sqrt{N}g; \\
E_w \ &= f\ _{\min}(X_{\text{train}}^\top X_{\text{train}}=N_{\text{train}}) \quad _{\max}(X_{\text{train}}^\top X_{\text{train}}=N_{\text{train}}) \quad g; \\
E_{b;\text{train}} \ &= f8(x_i; y_i)\ 2\ D_{\text{train}}; kx_ik_2 \quad B_x; jy_ij \quad B_y g; \\
E_{b;\text{val}} \ &= f8(x_i; y_i)\ 2\ D_{\text{val}}; kx_ik_2 \quad B_x; jy_ij \quad B_y g; \\
E_{b;N+1} \ &= fkx_{N+1}k_2 \quad B_x; jy_{N+1}j \quad B_y g:
\end{aligned}$$

For the good event $E := E \ \setminus E_w \setminus E_{b;\text{train}} \setminus E_{b;\text{test}} \setminus E_{b;N+1}$, we can show that $\mathsf{P}(E^c) \quad O\ N^{-10}$. Further, by the proof of Lemma F.1 (see e.g. (34)), we know that $\max_{k\in[K]} \mathbf{w}_{\text{ridge}}^k(D_{\text{train}})\ _2$ $B_w=2$ holds under the good event E.

For the ridge $_k = \frac{d\ ^2_k}{N_{\text{train}}}$ and parameters $(\ ;\ ;\ ; ")$, we consider the transformer constructed in Theorem I.2, with a clipped prediction $\hat{y}_{N+1} = \text{read}_y(\text{TF}\ (H))$.

In the following, we upper bound the quantity $\mathsf{E}_k(\hat{y}_{N+1} \quad y_{N+1})^2$ for any fixed k. Similar to the proof of Corollary 6 (Appendix F.4), we decompose

$$\mathsf{E}_k(\hat{y}_{N+1} \quad y_{N+1})^2 = \mathsf{E}_k\ 1fEg(\hat{y}_{N+1} \quad y_{N+1})^2 \ + \mathsf{E}_k\ 1fE^cg(\hat{y}_{N+1} \quad y_{N+1})^2;$$

and we analyze these two parts separately.

Part I. Recall that by our construction, when E holds, we have $\hat{y}_{N+1} = \mathrm{clip}_{B_y}(\langle \hat{\mathbf{w}}, \mathbf{x}_{N+1}\rangle)$ and the statements of Theorem I.2 hold for $\hat{\mathbf{w}}$. Thus, we have

$$\mathbb{E}_k\left[\mathbf{1}\{E\}(\hat{y}_{N+1} - y_{N+1})^2\right] = \mathbb{E}_k\left[\mathbf{1}\{E\}(\mathrm{clip}_{B_y}(\langle \mathbf{x}_{N+1}, \hat{\mathbf{w}}\rangle) - y_{N+1})^2\right] \le \mathbb{E}_k\left[\mathbf{1}\{E\}(\langle \mathbf{x}_{N+1}, \hat{\mathbf{w}}\rangle - y_{N+1})^2\right].$$

Let us consider the following risk functional

$$L_{\mathrm{val};\mathbf{w}_\star}(\mathbf{w}) = \mathbb{E}_{(\mathbf{x},y)\sim P_{\mathbf{w}_\star;\sigma_k}}\left[\tfrac{1}{2}(\langle \mathbf{w}, \mathbf{x}\rangle - y)^2\right] = \tfrac{1}{2}\|\mathbf{w} - \mathbf{w}_\star\|_2^2 + \sigma_k^2.$$

Then, under the good event $E_0 := E \cap E_w \cap E_{b;\mathrm{train}} \cap E_{b;\mathrm{test}}$ of $(\mathbf{w}_\star, D)$,

$$\begin{aligned}\mathbb{E}_k\left[\mathbf{1}\{E\}(\langle \mathbf{x}_{N+1}, \hat{\mathbf{w}}\rangle - y_{N+1})^2 \,\middle|\, \mathbf{w}_\star, D\right] &= \mathbb{E}_k\left[\mathbf{1}\{E\}(\langle \mathbf{x}_{N+1}, \hat{\mathbf{w}}(D)\rangle - y_{N+1})^2 \,\middle|\, \mathbf{w}_\star, D\right] \\ &\le \mathbb{E}_k\left[(\langle \mathbf{x}_{N+1}, \hat{\mathbf{w}}(D)\rangle - y_{N+1})^2 \,\middle|\, \mathbf{w}_\star, D\right] \\ &= \mathbb{E}_{(\mathbf{x},y)\sim P_{\mathbf{w}_\star;\sigma_k}}(\langle \mathbf{x}_{N+1}, \hat{\mathbf{w}}(D)\rangle - y_{N+1})^2 \\ &= L_{\mathrm{val};\mathbf{w}_\star}(\hat{\mathbf{w}}(D)).\end{aligned}$$

By our construction, under the good event E_0, we have

$$L_{\mathrm{val};\mathbf{w}_\star}(\hat{\mathbf{w}}(D)) \le L_{\mathrm{val};\mathbf{w}_\star}(\hat{\mathbf{w}}_k(D_{\mathrm{train}})) + \max_{l\in[K]}\left|\hat{L}_{\mathrm{val}}(\hat{\mathbf{w}}_l(D_{\mathrm{train}})) - L_{\mathrm{val};\mathbf{w}_\star}(\hat{\mathbf{w}}_l(D_{\mathrm{train}}))\right| + \varepsilon,$$

where $\|\hat{\mathbf{w}}_l(D_{\mathrm{train}}) - \mathbf{w}^l_{\mathrm{ridge}}(D_{\mathrm{train}})\|_2 \le \varepsilon''$ for each $l \in [K]$. Clearly,

$$\begin{aligned}&2\mathbb{E}_k[\mathbf{1}\{E_0\}L_{\mathrm{val};\mathbf{w}_\star}(\hat{\mathbf{w}}_k(D_{\mathrm{train}}))] = \mathbb{E}_k\left[\mathbf{1}\{E_0\}\,\|\hat{\mathbf{w}}_k(D_{\mathrm{train}}) - \mathbf{w}_\star\|_2^2 + \sigma_k^2\right] \\ &\le \mathbb{E}_k\left[\mathbf{1}\{E_0\}\,\|\mathbf{w}^k_{\mathrm{ridge}}(D_{\mathrm{train}}) - \mathbf{w}_\star\|_2^2 + 2\varepsilon''\|\mathbf{w}^k_{\mathrm{ridge}}(D_{\mathrm{train}}) - \mathbf{w}_\star\|_2 + \varepsilon''^2\right] + \sigma_k^2 \\ &\le \mathbb{E}_k\left[\|\mathbf{w}^k_{\mathrm{ridge}}(D_{\mathrm{train}}) - \mathbf{w}_\star\|_2^2 + 2\varepsilon''\|\mathbf{w}^k_{\mathrm{ridge}}(D_{\mathrm{train}}) - \mathbf{w}_\star\|_2 + \varepsilon''^2\right] + \sigma_k^2 \\ &\le 2\mathrm{Risk}_{k;\mathrm{train}} + 2\varepsilon''\sqrt{2\mathrm{Risk}_{k;\mathrm{train}}} + \varepsilon''^2,\end{aligned}$$

where we denote $2\mathrm{Risk}_{k;\mathrm{train}} = \mathbb{E}_k\|\mathbf{w}^k_{\mathrm{ridge}}(D_{\mathrm{train}}) - \mathbf{w}_\star\|_2^2 + \sigma_k^2$, and we also note that $\mathrm{Risk}_{k;\mathrm{train}} \ge 1 + \sigma_k^2$ by definition. By Lemma J.1, we have

$$\mathrm{Risk}_{k;\mathrm{train}} \le \mathrm{BayesRisk}_k + O\left((\sigma_k^2 + 1)\frac{N_{\mathrm{val}}}{N}\right).$$

We next deal with the term $\varepsilon''_{\mathrm{val}} := \max_{l\in[K]}\left|\hat{L}_{\mathrm{val}}(\hat{\mathbf{w}}_l(D_{\mathrm{train}})) - L_{\mathrm{val};\mathbf{w}_\star}(\hat{\mathbf{w}}_l(D_{\mathrm{train}}))\right|$. Note that for the good event $E_{\mathrm{train}} := E \cap E_w \cap E_{b;\mathrm{train}}$ of $(\mathbf{w}_\star, D_{\mathrm{train}})$, we have

$$\mathbb{E}_k[\mathbf{1}\{E_0\}\varepsilon''_{\mathrm{val}}] \le \mathbb{E}_k[\mathbf{1}\{E_{\mathrm{train}}\}\varepsilon''_{\mathrm{val}}] \le \mathbb{E}_{\mathbf{w}_\star, D_{\mathrm{train}}\sim P_k}[\mathbf{1}\{E_{\mathrm{train}}\}\,\mathbb{E}_{\mathcal{D}_{\mathrm{val}}}[\varepsilon''_{\mathrm{val}} \mid \mathbf{w}_\star, D_{\mathrm{train}}]].$$

Thus, Lemma J.2 yields

$$\mathbb{E}_k[\mathbf{1}\{E_0\}\varepsilon''_{\mathrm{val}}] \le O\left(B_w^2\left[\sqrt{\frac{\log K}{N_{\mathrm{val}}}} + \frac{\log K}{N_{\mathrm{val}}}\right]\right).$$

Therefore, we can conclude that

$$\mathbb{E}_k\left[\mathbf{1}\{E\}(\hat{y}_{N+1} - y_{N+1})^2\right] \le 2\mathrm{BayesRisk}_k + O\left(\varepsilon''_{\max} + \varepsilon''^2 + \frac{\sigma_{\max}^2 N_{\mathrm{val}}}{N} + B_w^2\left[\sqrt{\frac{\log K}{N_{\mathrm{val}}}} + \frac{B_w^2\log K}{N_{\mathrm{val}}}\right]\right).$$

Therefore, we can choose $(\varepsilon'', N_{\mathrm{val}})$ so that $N_{\mathrm{val}} \le N/2$ as

$$N_{\mathrm{val}} = \max\left(\frac{B_w^2}{\sigma_{\max}^2}N\right)^{2/3}\log^{1/3}(K), \log K\Big); \qquad \varepsilon'' = \frac{\sigma_{\max}}{N}.$$

It is worth noting that such choice of N_{val} is feasible as long as $N \gtrsim \frac{B_w^4}{\sigma_{\max}^4}\log K$. Under such choice, we obtain

$$\frac{1}{2}\mathbb{E}_k\left[\mathbf{1}\{E\}(\hat{y}_{N+1} - y_{N+1})^2\right] \le \mathrm{BayesRisk}_k + O\left(\sigma_{\max}^{4/3}B_w^{2/3}\left[\frac{\log K}{N}\right]^{1/3}\right).$$

Part II. Similar to the proof of Corollary 6, we have

$$\mathsf{E}\,\mathbb{1}\{E^c\}(\hat{y}_{N+1} - y_{N+1})^2 \quad O\left(\frac{B_y^2}{N^5}\right) \quad O\left(\frac{\max^2}{N^4}\right) :$$

Conclusion. Combining the both cases, we obtain

$$\mathsf{E}_k\,\tfrac{1}{2}(y_{N+1} - \hat{y}_{N+1})^2 \quad \mathrm{BayesRisk}_k + O\left(\max^{4/3} B_w^{2/3}\,\frac{\log K}{N}\right)^{1/3}$$

$$\mathrm{BayesRisk}_k + O\left(\frac{\max^2}{\min^{2/3}}\,\frac{\log K}{N}\right)^{1/3} + \max^{4/3}\frac{\log^{2/3}(N)\log^{1/3}(K)}{d^{2/3}N^{1/3}}$$

$$\mathrm{BayesRisk}_k + O\left(\frac{\max^2}{\min^{2/3}}\,\frac{\log K}{N}\right)^{1/3};$$

where we plug in our choice of B_y. The bounds on $M^{(\cdot)}; D^{(\cdot)}$ and $\|\cdot\|$ follows immediately from Theorem I.2. This completes the proof. □

J.2 Derivation of the exact Bayes predictor

Let $(D; x_{N+1}; y_{N+1})$ be $(N+1)$ observations from the data generating model considered in Section 4.1.1. On observing $(D; x_{N+1})$, the Bayes predictor of y_{N+1} is given by its posterior mean:

$$\mathsf{E}\,[y_{N+1}|D; x_{N+1}] = \mathsf{E}\,[\langle x_{N+1}, w_\star\rangle + \varepsilon_{N+1}|D; x_{N+1}] = \langle x_{N+1}, \mathsf{E}\,[w_\star|D]\rangle :$$

It thus remains to derive $\mathsf{E}\,[w_\star|D]$. Recall that our data generating model is given by k , By Bayes' rule, we have

$$\mathsf{E}\,[w_\star|D] = \sum_{k' \in [K]} \mathsf{P}\,(k = k'|D)\;\mathsf{E}\,[w_\star|D; k = k']. \tag{46}$$

On $k = k'$, the data is generated from the noisy linear model $w_\star \sim N(0; I_d/d)$, and $y = Xw_\star + \varepsilon$ where $\varepsilon_i \overset{\text{iid}}{\sim} N(0; \sigma_{k'}^2)$. It is a standard result that $\mathsf{E}\,[w_\star|D; k = k']$ is given by the ridge estimator

$$\mathsf{E}\,[w_\star|D; k = k'] = \underbrace{\left(X^\top X + d\,\sigma_{k'}^2\right)^{-1}}_{\hat{\Sigma}_{k'}^{-1}} X^\top y =: \hat{w}_{k'}$$

$$= \left(\frac{X^\top X}{N} + \frac{d\,\sigma_{k'}^2}{N}\right)^{-1}\frac{X^\top y}{N}.$$

(Note that the sample covariance within $\hat{\Sigma}_{k'}$ is not normalized by N, which is not to be confused with remaining parts within the paper.) Therefore, the posterior mean (46) is exactly a weighted combination of K ridge regression estimators, each with regularization $d\,\sigma_k^2/N$.

It remains to derive the mixing weights $\mathsf{P}\,(k = k'|D)$ for all $k' \in [K]$. By Bayes' rule, we have

$$\mathsf{P}\,(k = k'|D) \propto_{k'} \mathsf{P}\,(k = k') \int_{w_\star} p(w_\star)\; p_{k';w_\star}(D|w_\star)\,dw_\star$$

$$\propto_{k'} \int_w \frac{1}{(2\pi/d)^{d/2}(2\pi\sigma_{k'}^2)^{N/2}} \exp\left(-\frac{d\|w\|_2^2}{2} - \frac{\|Xw - y\|_2^2}{2\sigma_{k'}^2}\right) dw$$

$$\propto_{k'} \int_w \frac{1}{(2\pi\sigma_{k'}^2)^{N/2}} \exp\left(-\frac{1}{2}w^\top\left(\frac{X^\top X}{\sigma_{k'}^2} + dI_d\right)w + \left\langle w, \frac{X^\top y}{\sigma_{k'}^2}\right\rangle - \frac{\|y\|_2^2}{2\sigma_{k'}^2}\right) dw$$

$$\propto_{k'} \int_w \frac{1}{(2\pi\sigma_{k'}^2)^{N/2}} \exp\left(-\frac{1}{2\sigma_{k'}^2}(w - \hat{w}_{k'})^\top \hat{\Sigma}_{k'}(w - \hat{w}_{k'}) - \frac{1}{2\sigma_{k'}^2}\|y\|_2^2 - y^\top X\hat{\Sigma}_{k'}^{-1}X^\top y\right) dw$$

$$\propto_{k'} \frac{\det(\hat{\Sigma}_{k'}/\sigma_{k'}^2)^{-1/2}}{\sigma_{k'}^N} \exp\left(-\frac{1}{2\sigma_{k'}^2}\|y\|_2^2 - y^\top X\hat{\Sigma}_{k'}^{-1}X^\top y\right)$$

$$/ \quad {}_{k^0} \quad \frac{1}{\binom{N-d}{k^0}\det(X^\top X + d\ {}^2_{k^0} I_d)^{1=2}} \exp \quad \frac{1}{2\ {}^2_{k^0}} \quad ky k_2^2 \quad hy; X \mathbf{w}_{k^0} i \quad :$$

Note that such mixing weights involve the determinant of the matrix $b_{k^0} = X^\top X + d\ {}^2_{k^0} I_d$, which depends on the data X in a non-trivial fashion; Any transformer has to approximate these weights if their mechanism is to directly approximate the exact Bayesian predictor (46).

J.3 Useful lemmas

Lemma J.1. *For* $2\text{Risk}_{k;\text{train}} = \mathsf{E}_k \ \mathbf{w}^k_{\text{ridge}}(D_{\text{train}}) \quad \mathbf{w}_? \ {}^2_2 + \ {}^2_k$*, there exists universal constant* C *such that*

$$\text{Risk}_{k;\text{train}} \quad \text{BayesRisk}_k + C(\ {}^2_k + 1)\frac{N_{\text{val}}}{N}:$$

Proof. Recall that under P_k, we have

$$\mathbf{w}_? \quad \mathsf{N}(0; I_d/d); \qquad y_i = hx_i; \mathbf{w}_? i + "_i; \qquad "_i \quad \mathsf{N}(0; \ {}^2):$$

We denote $y_t = [y_i]_{i\in\mathcal{I}_{\text{train}}}$, then by definition $\mathbf{w}^k_{\text{ridge}}(D_{\text{train}}) = (X^\top_{\text{train}} X_{\text{train}} + d\ {}^2_k)^{-1} X_{\text{train}} y_t$ (with ${}_k = d\ {}^2_k/N_{\text{train}}$). Thus, a simple calculation yields

$$2\text{Risk}_{k;\text{train}} = \mathsf{E}_k \ \mathbf{w}^k_{\text{ridge}}(D_{\text{train}}) \quad \mathbf{w}_? \ {}^2_2 + \ {}^2_k = \ {}^2_k \text{Etr}\ (X^\top_{\text{train}} X_{\text{train}} + d\ {}^2_k)^{-1} \ + \ {}^2_k;$$

and analogously, $2\text{BayesRisk}_k = \ {}^2_k \text{Etr}\ (X^\top X + d\ {}^2_k I_d)^{-1} \ + \ {}^2_k$. Therefore,

$$2\text{Risk}_{k;\text{train}} \quad 2\text{BayesRisk}_k = \ {}^2_k \text{Etr}\ (X^\top_{\text{train}} X_{\text{train}} + d\ {}^2_k I_d)^{-1} \quad {}^2_k \text{Etr}\ (X^\top X + d\ {}^2_k I_d)^{-1}$$
$$ {}^2_k N_{\text{val}} \mathsf{E}_k[\ {}_{\min}(\ )^{-1}];$$

where in the above inequality we denote $\quad := X^\top_{\text{train}} X_{\text{train}} + d\ {}^2_k I_d$ and use the following fact:

$$\text{tr}\ \ {}^{-1} \quad \text{tr}\ (\ + X_v^\top X_v)^{-1} = \text{tr}\ \ {}^{-1=2}(I_d \quad (I_d + \ {}^{-1=2} X_v^\top X_v \ {}^{-1=2})^{-1}) \ {}^{-1=2}$$
$$= \text{tr}\ \ {}^{-1=2}(I_d + \ {}^{-1=2} X_v^\top X_v \ {}^{-1=2})^{-1} \ {}^{-1=2} X_v^\top X_v \ {}^{-1}$$
$$= \overset{D}{(I_d + \ {}^{-1=2} X_v^\top X_v \ {}^{-1=2})^{-1} \ {}^{-1=2} X_v^\top X_v \ {}^{-1=2}; \ {}^{-1}}^{E}$$
$$\text{rank}(\ {}^{-1=2} X_v^\top X_v \ {}^{-1=2}) \ {}_{\max}(\ {}^{-1}) \quad N_{\text{val}} \ {}_{\min}(\ )^{-1}:$$

Case 1. We first suppose that $N_{\text{train}} \quad 16d$. Then by definition $\quad d\ {}^2_k I_d$, and hence

$$ {}^2_k N_{\text{val}} \mathsf{E}_k[\ {}_{\min}(\ )^{-1}] \quad \frac{ {}^2_k N_{\text{val}}}{d\ {}^2_k} \quad \frac{16 N_{\text{val}}}{N_{\text{train}}} \quad \frac{32 N_{\text{val}}}{N}:$$

Case 2. When $N_{\text{train}} \quad 9d$, then we consider the event $E_t := f \ {}_{\min}(X^\top_{\text{train}} X_{\text{train}}/N_{\text{train}}) \quad \frac{1}{16} g$. By Lemma B.2 we have $\mathsf{P}(E_t^c) \quad \exp(\ N_{\text{train}}/8)$. Therefore,

$$ {}^2_k N_{\text{val}} \mathsf{E}_k[\ {}_{\min}(\ )^{-1}] = \ {}^2_k N_{\text{val}} \mathsf{E}_k[1 f E_t g \ {}_{\min}(\ )^{-1}] + \ {}^2_k N_{\text{val}} \mathsf{E}_k[1 f E_t^c g \ {}_{\min}(\ )^{-1}]$$
$$\frac{16\ {}^2_k N_{\text{val}}}{N_{\text{train}}} \ \mathsf{P}(E_t) + \frac{N_{\text{val}}}{d} \ \mathsf{P}(E_t^c)$$
$$\frac{32\ {}^2_k N_{\text{val}}}{N} + \frac{N_{\text{val}}}{d} \ \exp(\ N/16) = O \quad \frac{(\ {}^2_k + 1) N_{\text{val}}}{N} \quad :$$

Combining these two cases finishes the proof. □

Lemma J.2. *Condition on the event* E_{train}*, we have*

$$\mathsf{E}_{\mathcal{D}_{\text{val}} \sim \mathsf{P}_k | \mathbf{w}_?; \mathcal{D}_{\text{train}}} \ \max_{l\in[K]} \ b_{\text{val}}(\mathbf{w}_l) \quad L_{\text{val};\mathbf{w}_?}(\mathbf{w}_l) \qquad C B_w^2 \, 4 \frac{\log(2K)}{N_{\text{val}}} + \sqrt{\frac{\log(2K)}{N_{\text{val}}}} 5;$$

where we denote $\mathbf{w}_l = \mathbf{w}_l(D_{\text{train}})$.

Proof. We only need to work with a fixed pair of $(w_\star, D_{\text{train}})$ such that E_{train} holds. Hence, in the following we only consider the randomness of D_{val} conditional on such a $(w_\star, D_{\text{train}})$.

Recall that for any w,

$$\hat{L}_{\text{val}}(w) = \frac{1}{2|D_{\text{val}}|} \sum_{(x_i, y_i) \in \mathcal{D}_{\text{val}}} (\langle x_i, w \rangle - y_i)^2,$$

and we have $\mathbb{E}_{\mathcal{D}_v}[\hat{L}_{\text{val}}(w)] = L_{\text{val};w_\star}(w)$. For each $i \in I_{\text{val}}$,

$$y_i - \langle x_i, \hat{w}_l \rangle = \varepsilon_i - \langle x_i, w_\star - \hat{w}_l \rangle \sim \mathrm{SG}(\sigma_k^2 + k\|w_\star - \hat{w}_l\|^2).$$

Note that under E_{train}, we have $\hat{w}_l \in B_2(B_w)$ for all $l \in [K]$, and hence $\sigma_k^2 + k\|w_\star - \hat{w}_l\|^2 \le 5B_w^2$. We then have $(y_i - \langle x_i, \hat{w}_l \rangle)^2$'s are (conditional) i.i.d random variables in $\mathrm{SE}(cB_w^4)$. Then, by Bernstein's inequality, we have

$$\mathbb{P}_{\mathcal{D}_{\text{val}}} \left[\hat{L}_{\text{val}}(\hat{w}_l) - L_{\text{val};w_\star}(\hat{w}_l) \ge t \right] \le 2\exp\left(-cN_{\text{val}} \min\left\{ \frac{t^2}{B_w^4}, \frac{t}{B_w^2} \right\} \right),$$

where c is a universal constant. Applying the union bound, we obtain

$$\mathbb{P}_{\mathcal{D}_{\text{val}}} \left[\max_{l \in [K]} \hat{L}_{\text{val}}(\hat{w}_l) - L_{\text{val};w_\star}(\hat{w}_l) \ge t \right] \le K\exp\left(-cN_{\text{val}} \min\left\{ \frac{t^2}{B_w^4}, \frac{t}{B_w^2} \right\} \right).$$

Taking integration completes the proof. □

J.4 Generalized linear models with adaptive link function selection

Suppose that $(g_k : \mathbb{R} \to \mathbb{R})_{k \in [K]}$ is a set of link functions such that g_k is non-decreasing and C^2-smooth for each $k \in [K]$. We consider the input format we introduce in Section 4.1 with $|D_{\text{train}}| = \lceil N/2 \rceil$, $|D_{\text{val}}| = \lfloor N/2 \rfloor$.

Theorem J.2 (GLMs with adaptive link function selection)**.** *For any fixed set of parameters defined in Assumption B, as long as $N \ge O(d)$, there exists a transformer with $L \le O(\log(N))$ layers, input dimension $D = \Theta(dK)$ and $\max_{\ell \in [L]} M^{(\ell)} \le \Theta(d^3 N)$, such that the following holds.*

For any $k^\star \in [K]$ and any distribution $\mathbb{P}$ that is a generalized linear model of the link function $g_{k^\star}$ and some parameter , if Assumption B holds for each pair $(\mathbb{P}, g_k)$, then

$$\mathbb{E}_{(\mathcal{D}, x_{N+1}, y_{N+1}) \sim \mathbb{P}} (\hat{y}_{N+1} - y_{N+1})^2 \le \mathbb{E}_{(x,y) \sim \mathbb{P}} (g_{k^\star}(\langle x, \rangle) - y)^2 + O\left(\frac{d}{N} + \sqrt{\frac{\log(K)}{N}} \right);$$

or equivalently, $\mathbb{E}_{(\mathcal{D}, x_{N+1}) \sim \mathbb{P}}[(\hat{y}_{N+1} - \mathbb{E}[y_{N+1} | x_{N+1}])^2] \le O\left(d/N + \sqrt{\log(K)/N} \right)$.

Proof. For each $k \in [K]$, we consider optimizing the following training loss:

$$w_{\mathrm{GLM}}^{(k)} := \arg\min_{w} \hat{L}_{\text{train}}^{(k)}(w) := \frac{1}{N_{\text{train}}} \sum_{(x_i, y_i) \in \mathcal{D}_{\text{train}}} \ell_k(\langle x_i, w \rangle; y_i),$$

where $\ell_k(t; y) := -yt + \int_0^t g_k(s)ds$ is the convex (integral) loss associated with g_k (as in Section 3.1).

Also, for each predictor $f : \mathbb{R}^d \to \mathbb{R}$, we consider the squared validation loss $\hat{L}_{\text{val}}$:

$$\hat{L}_{\text{val}}(f) := \frac{1}{2N_{\text{val}}} \sum_{(x_i, y_i) \in \mathcal{D}_{\text{val}}} (f(x_i) - y_i)^2.$$

Fix a large universal constant C_0. Let us set

$$= \quad _g \quad _x/8; \qquad = 8L_g K_x;$$

$$B_x = C_0 K_x \sqrt{d \log(N)}; \qquad B_y = C_0 K_y \sqrt{\log(N)};$$

Then, we define good events similarly to the proof of Corollary 6 (Appendix F.4):

$$E_w = \left\{ \forall k \in [K],\ \forall w \in B_2(B_w),\ \min(r^2 \hat{p}^{(k)}_{\text{train}}(w)) \quad \max(r^2 \hat{p}^{(k)}_{\text{train}}(w)) \right\};$$
$$E_r = \left\{ \forall k \in [K],\ \|w^{(k)}_{\text{GLM}}\|_2 \quad B_w/2 \right\};$$
$$E_{b;\text{train}} = \{\forall (x_i, y_i) \in D_{\text{train}}, \|x_i\|_2 \quad B_x, |y_i| \quad B_y\};$$
$$E_{b;\text{val}} = \{\forall (x_i, y_i) \in D_{\text{val}}, \|x_i\|_2 \quad B_x, |y_i| \quad B_y\};$$
$$E_{b;N+1} = \{\|x_{N+1}\|_2 \quad B_x, |y_{N+1}| \quad B_y\}.$$

Similar to the proof of Theorem G.2 (Appendix G.2), we know the good event $E := E_w \cap E_r \cap E_{b;\text{train}} \cap E_{b;\text{test}} \cap E_{b;N+1}$ holds with high probability: $\mathbb{P}(E^c) \quad O(N^{-10})$.

Similar to the proof of Theorem I.2, we can show that there exists a transformer with prediction $\hat{y}_{N+1} = \text{read}_y(\text{TF}(H))$ (clipped by B_y), such that (for any $\mathbb{P}$) the following holds under E:

(a) For each $k \in [K]$, $f_k = A_k(D_{\text{train}})$ is a predictor such that $f_k(x_i) \quad g_k(\langle x_i, w^{(k)}_{\text{GLM}} \rangle)$ " for all $i \in [N+1]$ (where " is chosen as in Appendix G.2).

(b) $\hat{y}_{N+1} = \text{clip}_{B_y}(\hat{f}(x_{N+1}))$, where $\hat{f} = A_{\text{TF}}(D)$ is an aggregated predictor given by $\hat{f} = \sum_k \hat{q}_k f_k$, such that $(\hat{q}_k)$ is a distribution supported on $k \in [K]$ such that $\hat{L}_{\text{val}}(f_k) \quad \min_{k' \in [K]} \hat{L}_{\text{val}}(f_{k'}) + \ $.

Similar to the proof of Theorem G.2, for $E_0 := E_w \cap E_r \cap E_{b;\text{train}} \cap E_{b;\text{test}}$, we have

$$\mathbb{E}_{(\mathcal{D}, x_{N+1}, y_{N+1}) \sim \mathbb{P}} (\hat{y}_{N+1} \quad y_{N+1})^2 \quad \mathbb{E}_{\mathcal{D} \sim \mathbb{P}} \left[1\{E_0\} L_{\text{val}}(\hat{f}) \right] + O\left(\frac{B_y^2}{N^5} \right);$$

where we denote $L_{\text{val}}(f) := \mathbb{E}_{(x,y) \sim \mathbb{P}} \left[1\{\|x\|_2 \quad B_x\} (f(x) \quad y)^2 \right]$ for each predictor f. By the definition of $\hat{f}$, we then have (under E_0)

$$L_{\text{val}}(\hat{f}) \quad L_{\text{val}}(f_{k^\star}) + \max_l \hat{L}_{\text{val}}(f_l) \quad L_{\text{val}}(f_l) + \ .$$

For the first term, repeating the argument in the proof of Theorem G.2 directly yields that for $E_{\text{train}} := E_w \cap E_r \cap E_{b;\text{train}}$,

$$\mathbb{E}_{\mathcal{D}_{\text{train}} \sim \mathbb{P}} [1\{E_{\text{train}}\} L_{\text{val}}(f_{k^\star})] \quad \mathbb{E}_{(x,y) \sim \mathbb{P}} (g_{k^\star}(\langle x, \ \rangle) \quad y)^2 + O(d/N_{\text{train}}).$$

For the second term, similar to Lemma J.2, we can show that conditional on D_{train} such that E_{train} holds, it holds

$$\mathbb{E}_{\mathcal{D}_{\text{val}} \sim \mathbb{P} | \mathcal{D}_{\text{train}}} \left[1\{E_0\} \max_l \hat{L}_{\text{val}}(f_l) \quad L_{\text{val}}(f_l) \right] \quad O\left(K_y^2 \left(\sqrt{\frac{\log K}{N_{\text{val}}}} + \frac{\log K}{N_{\text{val}}} \right) \right).$$

Combining these inequalities and suitably choosing complete the proof. □

K Analysis of pretraining

Thus far, we have established the existence of transformers for performing various ICL tasks with good in-context statistical performance. We now analyze the sample complexity of pretraining these transformers from a finite number of training ICL instances.

K.1 Generalization guarantee for pretraining

Setup At pretraining time, each training ICL instance has form $Z := (H, y_{N+1})$, where $H := H(D, x_{N+1}) \in \mathbb{R}^{D \times (N+1)}$ denote the input sequence formatted as in (3). We consider the square loss between the in-context prediction and the ground truth label:

$$\ell_{\text{icl}}(\ ; Z) := \frac{1}{2} \left(y_{N+1} \quad \underbrace{\text{clip}_{B_y} \text{read}_y(\text{TF}^R(H))}_{\text{read}_y} \right)^2.$$

Above, $\mathrm{clip}_{B_y}(t) := \max\{\min\{t, B_y\}, -B_y\}$ is the standard clipping operator onto $[-B_y, B_y]$, and TF^R the transformer architecture as in Definition 3 with clipping operators after each layer: let $H^{(0)} = \mathrm{clip}_R(H)$,

$$H^{(\ell)} = \mathrm{clip}_R\left(\mathrm{MLP}^{(\ell)}_{\mathrm{mlp}}\left(\mathrm{Attn}^{(\ell)}_{\mathrm{attn}}(H^{(\ell-1)})\right)\right) \quad \text{for all } \ell \in [L], \quad \mathrm{clip}_R(H) := [\mathrm{Proj}_{\|h\|_2 \le R}(h_i)]_i.$$

The clipping operator is used to control the Lipschitz constant of TF with respect to θ, and we typically choose a sufficiently large clipping radius R so that it does not modify the behavior of the transformer on any input sequence of our concern.

We draw ICL instances $Z := (H, y_{N+1}) = (D, (x_{N+1}, y_{N+1}))$ from a (meta-)distribution denoted as P, which first sample an in-context data distribution P, then sample iid examples $(x_i, y_i)_{i=1}^{N+1} \overset{\text{iid}}{\sim} \mathsf{P}^{\otimes(N+1)}$ and form $D = \{(x_i, y_i)\}_{i\in[N]}$. Our pretraining loss is the average ICL loss on n pretraining instances $Z^{(1:n)} \overset{\text{iid}}{\sim} \mathsf{P}$, and we consider the corresponding test ICL loss on a new test instance:

$$\hat{L}_{\mathrm{icl}}(\theta) := \frac{1}{n}\sum_{j=1}^{n} \ell_{\mathrm{icl}}(\theta; Z^j), \quad L_{\mathrm{icl}}(\theta) := \mathbb{E}_{\mathsf{P}\sim Q, Z\sim\mathsf{P}^{(N+1)}}[\ell_{\mathrm{icl}}(\theta; Z)].$$

Our pretraining algorithm is to solve a standard constrained empirical risk minimization (ERM) problem over transformers with L layers, M heads, and norm bound B (recall the definition of the $|||\cdot|||$ norm in (2)):

$$\hat{\theta} := \operatorname*{arg\,min}_{\theta \in \Theta_{L,M,D',B}} \hat{L}_{\mathrm{icl}}(\theta);$$

$$\Theta_{L,M,D',B} := \left\{\theta = (\theta^{(1:L)}_{\mathrm{attn}}, \theta^{(1:L)}_{\mathrm{mlp}}) : \max_{\ell\in[L]} M^{(\ell)} \le M, \max_{\ell\in[L]} D^{(\ell)} \le D', |||\theta||| \le B\right\}. \tag{TF-ERM}$$

Generalization guarantee By standard uniform concentration analysis via chaining arguments (Proposition B.4; see also [87, Chapter 5] for similar arguments), we have the following excess loss guarantee for (TF-ERM). The proof can be found in Appendix L.2.

Theorem K.1 (Generalization for pretraining)**.** *With probability at least* $1-\delta$ *(over the pretraining instances* $\{Z^j\}_{j\in[n]}$*), the solution* $\hat{\theta}$ *to (TF-ERM) satisfies*

$$L_{\mathrm{icl}}(\hat{\theta}) \le \inf_{\theta\in\Theta_{L,M,D',B}} L_{\mathrm{icl}}(\theta) + O\left(B_y^2 \sqrt{\frac{L^2(MD^2 + DD') + \log(1/\delta)}{n}}\right);$$

where $\iota = \log(2 + \max\{B, R, B_y\})$ *is a log factor.*

K.2 Examples of pretraining for in-context regression problems

In Theorem K.1, the comparator $\inf_{\theta\in\Theta_{L,M,D',B}} L_{\mathrm{icl}}(\theta)$ is simply the smallest expected ICL loss for ICL instances drawn from P, among all transformers within the norm ball $\Theta_{L,M,D',B}$. Using our constructions in Section 3 & 4, we show that this comparator loss is small on various (meta-)distribution P's, by which we obtain end-to-end guarantees for pretraining transformers with small ICL loss at test time. Here we showcase this argument on several representative regression problems.

Linear regression For any in-context data distribution P, let $w^\star_{\mathsf{P}} := \mathbb{E}_{\mathsf{P}}[xx^\top]^{-1}\mathbb{E}_{\mathsf{P}}[xy]$ denote the best linear predictor for P. We show that with mild choices of L, M, B, the learned transformer can perform in-context linear regression with near-optimal statistical power, in that on the sampled P and ICL instance $\{(x_i, y_i)\}_{i\in[N+1]} \overset{\text{iid}}{\sim} \mathsf{P}$, it competes with the best linear predictor $w^\star_{\mathsf{P}}$ *for this particular* P. The proof follows directly by on combining Corollary 5 with Theorem K.1, and can be found in Appendix L.3.

Theorem K.2 (Pretraining transformers for in-context linear regression)**.** *Suppose* P *is almost surely* well-posed for in-context linear regression *(Assumption A) with the canonical parameters. Then, for* $N \ge \Theta(d)$*, with probability at least* $1-\delta$ *(over the training instances* $Z^{(1:n)}$*), the solution*

$\hat{\theta}$ *of (TF-ERM) with* $L = O(\log(N/\epsilon))$ *layers,* $M = 3$ *heads,* $D' = 0$ *(attention-only), and* $B = O(\sqrt{d})$ *achieves small excess ICL risk over* $w_P^?$*:*

$$L_{\mathrm{icl}}(\hat{\theta}) \le E_{P\sim} E_{(x,y)\sim P} \frac{1}{2}(y - \langle w_P^?, x\rangle)^2 \le \tilde{O}\left(\sqrt{\frac{\sigma^2 d^2 + \log(1/\delta)}{n}} + \frac{d^2}{N}\right);$$

where $\tilde{O}(\cdot)$ *only hides polylogarithmic factors in* $\sigma, d, N, 1/\delta$.

To our best knowledge, Theorem K.2 offers the first end-to-end result for pretraining a transformer to perform in-context linear regression with explicit excess loss bounds. The $\tilde{O}(\sqrt{\sigma^2 d^2/n})$ term originates from the generalization of pretraining (Theorem K.1), where as the $\tilde{O}(d^2/N)$ term agrees with the standard fast rate for the excess loss of linear regression [38]. Further, as long as $n \gtrsim \tilde{O}(\sigma^2 N/d^2)$, the excess risk achieves the optimal rate $\tilde{O}(d^2/N)$ (up to log factors).

Additional examples By similar arguments as in the proof of Theorem K.2, we can directly turn most of our other expressivity results into results on the pretrained transformers. Here we present three such additional examples (proofs in Appendix L.4-L.6). The first example is for the sparse linear regression problem considered in Theorem 8.

Theorem K.3 (Pretraining transformers for in-context sparse linear regression)**.** *Suppose each* P *is almost surely an instance of the sparse linear model specified in Theorem 8 with parameters* $B_w^?$ *and* σ*. Suppose* $N \gtrsim \tilde{O}(s\log((d/\epsilon)N/\delta))$ *and let* $\epsilon := B_w^?/N$*.*

Then with probability at least $1-\delta$ *(over the training instances* $Z^{(1:n)}$*), the solution* $\hat{\theta}$ *of (TF-ERM) with* $L = \tilde{O}(\log(1/\epsilon)(1 + d/N))$ *layers,* $M = 2$ *heads,* $D' = 2d$*, and* $B = \tilde{O}(\mathrm{poly}(d, B_w^?, 1/\epsilon))$ *achieves small excess ICL risk:*

$$L_{\mathrm{icl}}(\hat{\theta}) \le \epsilon^2 + \tilde{O}\left(\sqrt{\frac{\sigma^4 d^2(1+d/N)^2 + \log(1/\delta)}{n}} + \epsilon^2\frac{s\log d}{N}\right);$$

where $\tilde{O}(\cdot)$ *only hides polylogarithmic factors in* $d, N, 1/\delta$.

Our next example is for the problem of noisy linear regression with mixed noise levels considered in Theorem 12 and Theorem J.1. There, the constructed transformer uses the post-ICL validation mechanism to perform ridge regression with an adaptive regulariation strength depending on the particular input sequence.

Theorem K.4 (Pretraining transformers for in-context noisy linear regression with algorithm selection)**.** *Suppose* $\quad$ *is the data generating model (noisy linear model with mixed noise levels) considered in Theorem J.1, with* $\sigma_{\max} \le O(1)$*. Let* $N \ge d/\epsilon$*.*

Then, with probability at least $1-\delta$ *(over the training instances* $Z^{(1:n)}$*), the solution* $\hat{\theta}$ *of (TF-ERM) with input dimension* $D = \quad (dK)$*,* $L = O(\sigma_{\min}^{-2}\log(N/\epsilon\delta))$ *layers,* $M = O(K)$ *heads,* $D' = O(K^2)$*, and* $B = O(\mathrm{poly}(K, \sigma_{\min}^{-1}, d, N))$ *achieves small excess ICL risk:*

$$L_{\mathrm{icl}}(\hat{\theta}) \le \mathrm{BayesRisk} + \tilde{O}\left(\sqrt{\frac{\sigma_{\min}^{-4}K^3d^2 + \log(1/\delta)}{n}} + \frac{\sigma_{\max}^2}{\sigma_{\min}^{2=3}}\frac{\log K}{N}\right)^{1=3};$$

where $\tilde{O}(\cdot)$ *only hides polylogarithmic factors in* $d, N, K, 1/\epsilon, \sigma_{\min}$.

Our final example is for in-context logistic regression. For simplicity we consider the realizable case.

Theorem K.5 (Pretraining transformers for in-context logistic regression; square loss guarantee)**.** *Suppose for* P *,* P *is almost surely a realizable logistic model (i.e.* $P = P^{\log}$ *with* $k \le k_2$ $\quad B_w^?$ *as in Corollary G.1). Suppose that* $B_w^? = O(1)$ *and* $N \ge O(d)$*.*

Then, with probability at least $1-\delta$ *(over the training instances* $Z^{(1:n)}$*), the solution* $\hat{\theta}$ *of (TF-ERM) with* $L = O(\log(N))$ *layers,* $M = \tilde{O}(d^3N)$ *heads,* $D' = 0$*, and* $B = O(\mathrm{poly}(d, N))$ *achieves small excess ICL risk:*

$$L_{\mathrm{icl}}(\hat{\theta}) \le E_{P^{\log}\sim} E_{(x,y)\sim P^{\log}} \frac{1}{2}(y - \log(\langle h, x\rangle))^2 \le \tilde{O}\left(\sqrt{\frac{d^5N + \log(1/\delta)}{n}} + \frac{d}{N}\right);$$

where $\tilde{O}(\cdot)$ *only hides polylogarithmic factors in* d, N.

Remark on generality of transformer All results above are established by the expressivity results in Section 3 & 4 for transformers to implement various ICL procedures (such as least squares, Lasso, GLM, and ridge regression with in-context algorithm selection), combined with the generalization bound (Theorem K.1). However, the transformer itself was not specified to encode any actual structure about the problem at hand in any result above, other than having sufficiently large number of layers, number of heads, and weight norms, which illustrates the flexibility of the transformer architecture.

L Proofs for Section K

L.1 Lipschitzness of transformers

For any $p \geq [1, 1]$, let $\|H\|_{2,p} := (\sum_{i=1}^{N} \|h_i\|_2^p)^{1/p}$ denote the column-wise $(2,p)$-norm of H. For any radius $R > 0$, we denote $H_R := \{H : \|H\|_{2,\infty} \leq R\}$ be the ball of radius R under norm $\|\cdot\|_{2,\infty}$.

Lemma L.1. *For a single MLP layer* $\theta_{\mathtt{mlp}} = (W_1, W_2)$*, we introduce its norm (as in (2))*

$$\|\theta_{\mathtt{mlp}}\| = \|W_1\|_{\mathrm{op}} + \|W_2\|_{\mathrm{op}}.$$

For any fixed hidden dimension D', we consider

$$\Theta_{\mathtt{mlp};B} := \{\theta_{\mathtt{mlp}} : \|\theta_{\mathtt{mlp}}\| \leq B\}.$$

Then for $H \in H_R$, $\theta_{\mathtt{mlp}} \in \Theta_{\mathtt{mlp};B}$, the function $(\theta_{\mathtt{mlp}}, H) \mapsto \mathrm{MLP}_{\theta_{\mathtt{mlp}}}(H)$ is (BR)-Lipschitz w.r.t. $\theta_{\mathtt{mlp}}$ and $(1+B^2)$-Lipschitz w.r.t. H.

Proof. Recall that by our definition, for the parameter $\theta_{\mathtt{mlp}} = (W_1, W_2) \in \Theta_{\mathtt{mlp};B}$ and the input $H = [h_i] \in \mathbb{R}^{D\times N}$, the output $\mathrm{MLP}_{\theta_{\mathtt{mlp}}}(H) = H + W_2\,(W_1 H) = [h_i + W_2\,(W_1 h_i)]_i$. Therefore, for $\theta'_{\mathtt{mlp}} = (W_1', W_2') \in \Theta_{\mathtt{mlp};B}$, we have

$$\begin{aligned}
&\|\mathrm{MLP}_{\theta_{\mathtt{mlp}}}(H) - \mathrm{MLP}_{\theta'_{\mathtt{mlp}}}(H)\|_{2,\infty} \\
&= \max_i \|W_2\,(W_1 h_i) - W_2'\,(W_1' h_i)\|_2 \\
&= \max_i \|(W_2 - W_2')\,(W_1 h_i) + W_2'(\,(W_1 h_i) - (W_1' h_i))\|_2 \\
&\leq \max_i \|W_2 - W_2'\|_{\mathrm{op}}\,\|(W_1 h_i)\|_2 + \|W_2'\|_{\mathrm{op}}\,\|(W_1 h_i) - (W_1' h_i)\|_2 \\
&\leq \max_i \|W_2 - W_2'\|_{\mathrm{op}}\,\|W_1 h_i\|_2 + \|W_2'\|_{\mathrm{op}}\,\|W_1 h_i - W_1' h_i\|_2 \\
&\leq BR\,\|W_2 - W_2'\|_{\mathrm{op}} + BR\,\|W_1 - W_1'\|_{\mathrm{op}},
\end{aligned}$$

where the second inequality follows from the 1-Lipschitznees of $\cdot + [\,]_+$. Similarly, for $H' = [h_i'] \in \mathbb{R}^{D\times N}$,

$$\begin{aligned}
\|\mathrm{MLP}_{\theta_{\mathtt{mlp}}}(H) - \mathrm{MLP}_{\theta_{\mathtt{mlp}}}(H')\|_{2,\infty} &= \max_i \|h_i + W_1\,(W_2 h_i) - h_i' - W_1\,(W_2 h_i')\|_2 \\
&\leq \|H - H'\|_{2,\infty} + \max_i \|W_1(\,(W_2 h_i) - (W_2 h_i'))\|_2 \\
&\leq \|H - H'\|_{2,\infty} + \max_i B\,\|\,(W_2 h_i) - (W_2 h_i')\|_2 \\
&\leq \|H - H'\|_{2,\infty} + B^2\,\|H - H'\|_{2,\infty}.
\end{aligned}$$

□

Lemma L.2. *For a single attention layer* $\theta_{\mathtt{attn}} = \{(V_m, Q_m, K_m)\}_{m\in[M]} \in \mathbb{R}^{D\times D}$*, we introduce its norm (as in (2))*

$$|||\theta_{\mathtt{attn}}||| := \max_{m\in[M]} \{\|Q_m\|_{\mathrm{op}}, \|K_m\|_{\mathrm{op}}\} + \sum_{m=1}^{M} \|V_m\|_{\mathrm{op}}.$$

For any fixed dimension D, we consider

$$\Theta_{\mathtt{attn};B} := \{\theta_{\mathtt{attn}} : |||\theta_{\mathtt{attn}}||| \leq B\}.$$

Then for $H \in H_R$, $\theta_{\mathtt{attn}} \in \Theta_{\mathtt{attn};B}$, the function $(\theta_{\mathtt{attn}}, H) \mapsto \mathrm{Attn}_{\theta_{\mathtt{attn}}}(H)$ is (B^2R^3)-Lipschitz w.r.t. $\theta_{\mathtt{attn}}$ and $(1+B^3R^2)$-Lipschitz w.r.t. H.

Proof. Recall that by our definition, for the parameter $_{\text{attn}} = f(V_m; Q_m; K_m)g_{m\in[M]}$ 2 $_{\text{attn};B}$ and the input $H = [h_i]$ 2 $R^{D\times N}$, the output $\text{Attn}_{\text{attn}}(H) = [\hat{h}_i]$ is given by

$$\hat{h}_i = h_i + \sum_{m=1}^{M} \frac{1}{N} \sum_{j=1}^{N} (\langle Q_m h_i; K_m h_j\rangle) \quad V_m h_j:$$

Now, for $'_{\text{attn}} = f(V'_m; Q'_m; K'_m)g_{m\in[M]}$, we consider

$$\hat{h}'_i = \text{Attn}_{'_{\text{attn}}}(H)_i = h_i + \sum_{m=1}^{M} \frac{1}{N} \sum_{j=1}^{N} (\langle Q'_m h_i; K'_m h_j\rangle) \quad V'_m h_j; \qquad 8i \; 2 \; [N]:$$

Clearly $\text{Attn}_{\text{attn}}(H) \quad \text{Attn}_{'_{\text{attn}}}(H)_{2;\infty} = \max_i \quad \hat{h}_i \quad \hat{h}'_i \quad _2$. For any $i \; 2 \; [N]$, we have

$$\begin{aligned}
\hat{h}_i \quad \hat{h}'_i \quad _2 = & \sum_{m=1}^{M} \frac{1}{N} \sum_{j=1}^{N} [\ (\langle Q_m h_i; K_m h_j\rangle) V_m h_j \quad (\langle Q'_m h_i; K'_m h_j\rangle) V'_m h_j] \quad _2 \\
& \sum_{m=1}^{M} \frac{1}{N} \sum_{j=1}^{N} k \ (\langle Q_m h_i; K_m h_j\rangle) V_m \quad (\langle Q'_m h_i; K'_m h_j\rangle) V'_m k_{\text{op}} \, k h_j k_2 \\
& \sum_{m=1}^{M} \frac{1}{N} \sum_{j=1}^{N} k h_j k_2 \quad \Big((\langle Q_m h_i; K_m h_j\rangle) \quad k V_m \quad V'_m k_{\text{op}} \\
& \qquad + \quad (\langle Q_m h_i; K_m h_j\rangle) \quad (\langle Q'_m h_i; K_m h_j\rangle) \quad k V'_m k_{\text{op}} \\
& \qquad + \quad (\langle Q'_m h_i; K_m h_j\rangle) \quad (\langle Q'_m h_i; K'_m h_j\rangle) \quad k V'_m k_{\text{op}} \Big) \\
& \sum_{m=1}^{M} \frac{1}{N} \sum_{j=1}^{N} R \Big(B^2 R^2 \quad k V_m \quad V'_m k_{\text{op}} + k Q_m h_i \quad Q'_m h_i k_2 \quad k K_m h_j k_2 \quad k V'_m k_{\text{op}} \\
& \qquad + k Q'_m h_i k_2 \quad k K_m h_j \quad K'_m h_j k_2 \quad k V'_m k_{\text{op}} \Big) \\
& \sum_{m=1}^{M} R \Big(B^2 R^2 \, k V_m \quad V'_m k_{\text{op}} + B R^2 \, k Q_m \quad Q'_m k_{\text{op}} \quad k V'_m k_{\text{op}} + B R^2 \, k K_m \quad K'_m k_{\text{op}} \quad k V'_m k_{\text{op}} \Big) \\
& B^2 R^3 \sum_{m=1}^{M} k V_m \quad V'_m k_{\text{op}} + \max_m k Q_m \quad Q'_m k_{\text{op}} + \max_m k K_m \quad K'_m k_{\text{op}} \\
= & \ B^2 R^3 \| \quad _{\text{attn}} \quad '_{\text{attn}} \|;
\end{aligned}$$

where the second inequality uses the definition of operator norm, the third inequality follows from the triangle inequality, the forth inequality is because $k Q_m h_i k_2 \quad BR; k K_m h_j k_2 \quad BR$, and is 1-Lipschitz. This completes the proof of the Lipschitzness w.r.t. $_{\text{attn}}$.

Similarly, we consider $H' = [h'_i]$, and

$$\hat{h}'_i = \text{Attn}_{'_{\text{attn}}}(H)_i = h'_i + \sum_{m=1}^{M} \frac{1}{N} \sum_{j=1}^{N} \quad Q_m h'_i; K_m h'_j \quad V_m h'_j; \qquad 8i \; 2 \; [N]:$$

By definition, we can similarly bound

$$\begin{aligned}
& \hat{h}'_i \quad h'_i \quad \hat{h}_i \quad h_i \quad _2 \\
= & \sum_{m=1}^{M} \frac{1}{N} \sum_{j=1}^{N} (\langle Q_m h_i; K_m h_j\rangle) V_m h_j \quad Q_m h'_i; K_m h'_j \quad V_m h'_j \quad _2 \\
& \sum_{m=1}^{M} \frac{1}{N} \sum_{j=1}^{N} k V_m k_{\text{op}} \quad (\langle Q_m h_i; K_m h_j\rangle) h_j \quad Q_m h'_i; K_m h'_j \quad h'_j \quad _2
\end{aligned}$$

$$\sum_{m=1}^{M}\frac{1}{N}\sum_{j=1}^{N}\|V_m\|_{\mathrm{op}}\Big\|\Big(\|Q_m h_i\|\,\|K_m h_j\|\Big)\,h_j-\Big(\|Q_m h_i'\|\,\|K_m h_j\|\Big)\,h_j'\Big\|_2$$
$$+\Big(\|Q_m h_i'\|\,\|K_m h_j\|\Big)\,h_j'-\Big(\|Q_m h_i'\|\,\|K_m h_j'\|\Big)\,h_j'\Big\|_2$$
$$+\Big(\|Q_m h_i'\|\,\|K_m h_j\|\Big)\,h_j'-\Big(\|Q_m h_i'\|\,\|K_m h_j'\|\Big)\,h_j'\Big\|_2\Big\}$$
$$\sum_{m=1}^{M}\frac{1}{N}\sum_{j=1}^{N}\|V_m\|_{\mathrm{op}}\,3\|Q_m\|_{\mathrm{op}}\|K_m\|_{\mathrm{op}}R^2\,\|h_j-h_j'\|_2$$
$$R^2\|H-H'\|_{2,\infty}\,3\max_{m\in[M]}\|Q_m\|_{\mathrm{op}}\|K_m\|_{\mathrm{op}}\sum_{m=1}^{M}\|V_m\|_{\mathrm{op}}$$
$$B^3R^2\|H-H'\|_{2,\infty};$$

where the last inequality uses $\|\theta\|,\|\theta'\|\le B$ and the AM-GM inequality. This completes the proof of the Lipschitzness w.r.t. H. □

Corollary L.1. *For a fixed number of heads M and hidden dimension D', we consider*

$$\Theta_{\mathrm{TF},1,B}=\{\theta=(\theta_{\mathrm{attn}},\theta_{\mathrm{mlp}}):M\text{ heads, hidden dimension }D',\|\theta\|\le B\}.$$

Then for the function TF^R given by

$$\mathrm{TF}^R:(\theta,H)\mapsto\mathrm{clip}_R\big(\mathrm{MLP}_{\theta_{\mathrm{mlp}}}(\mathrm{Attn}_{\theta_{\mathrm{attn}}}(H))\big),\qquad\theta\in\Theta_{\mathrm{TF},1,B},H\in B_R,$$

TF^R is B'-Lipschitz w.r.t θ and L_H-Lipschitz w.r.t. H, where $B':=BR(1+BR^2+B^3R^2)$ and $B_H:=(1+B^2)(1+B^2R^3)$.

Proof. For any $\theta=(\theta_{\mathrm{attn}},\theta_{\mathrm{mlp}})$, $H\in B_R$, and $\theta'=(\theta'_{\mathrm{attn}},\theta'_{\mathrm{mlp}})$, we have

$$\|\mathrm{TF}_\theta(H)-\mathrm{TF}_{\theta'}(H)\|_{2,\infty}\le\big\|\mathrm{MLP}_{\theta_{\mathrm{mlp}}}(\mathrm{Attn}_{\theta_{\mathrm{attn}}}(H))-\mathrm{MLP}_{\theta_{\mathrm{mlp}}}(\mathrm{Attn}_{\theta'_{\mathrm{attn}}}(H))\big\|_{2,\infty}$$
$$+\big\|\mathrm{MLP}_{\theta_{\mathrm{mlp}}}(\mathrm{Attn}_{\theta'_{\mathrm{attn}}}(H))-\mathrm{MLP}_{\theta'_{\mathrm{mlp}}}(\mathrm{Attn}_{\theta'_{\mathrm{attn}}}(H))\big\|_{2,\infty}$$
$$(1+B^2)\|\mathrm{Attn}_{\theta_{\mathrm{attn}}}(H)-\mathrm{Attn}_{\theta'_{\mathrm{attn}}}(H)\|_{2,\infty}+B\bar{R}\|\theta_{\mathrm{mlp}}-\theta'_{\mathrm{mlp}}\|$$
$$(1+B^2)B^2R^3\|\theta_{\mathrm{attn}}-\theta'_{\mathrm{attn}}\|+B\bar{R}\|\theta_{\mathrm{mlp}}-\theta'_{\mathrm{mlp}}\|$$
$$B\|\theta-\theta'\|;$$

where the second inequality follows from Lemma L.2 and Lemma L.1 and the fact that $\|\mathrm{Attn}_{\theta_{\mathrm{attn}}}(H)\|_{2,\infty}\le\bar{R}:=R+B^3R^3$ for all $H\in B_R$.

Furthermore, for $H'\in B_R$, we have

$$\|\mathrm{TF}_\theta(H)-\mathrm{TF}_\theta(H')\|_{2,\infty}\le(1+B^2)\|\mathrm{Attn}_{\theta_{\mathrm{attn}}}(H)-\mathrm{Attn}_{\theta_{\mathrm{attn}}}(H')\|_{2,\infty}$$
$$(1+B^2)(1+B^3R^2)\|H-H'\|_{2,\infty};$$

which also follows from Lemma L.2 and Lemma L.1. □

Proposition L.1 (Lipschitzness of transformers)**.** *For a fixed number of heads M and hidden dimension D', we consider*

$$\Theta_{\mathrm{TF},L,B}=\Big\{\theta=(\theta_{\mathrm{attn}}^{(1:L)},\theta_{\mathrm{mlp}}^{(1:L)}):M^{(\ell)}=M;D^{(\ell)}=D';\|\theta\|\le B\Big\}.$$

Then the function TF^R is $(LB_H^{L-1}B')$-Lipschitz w.r.t $\theta\in\Theta_{\mathrm{TF},L,B}$ for any fixed H.

Proof. For $\theta=\theta^{(1:L)}\in\Theta_{\mathrm{TF},L,B}$, $\theta_e=\theta_e^{(1:L)}\in\Theta_{\mathrm{TF},L,B}$, we have

$$\|\mathrm{TF}^R(H)-\mathrm{TF}_e^R(H)\|_{2,\infty}$$

$$\sum_{\ell=1}^{L} \left\| \mathrm{TF}^{R}_{(\ell+1:L)}\left(\mathrm{TF}^{R}_{(\ell)}\left(\mathrm{TF}^{R}_{e(1:\ell-1)}(H)\right)\right) - \mathrm{TF}^{R}_{(\ell+1:L)}\left(\mathrm{TF}^{R}_{e(\ell)}\left(\mathrm{TF}^{R}_{e(1:\ell-1)}(H)\right)\right) \right\|_{2,\infty}$$
$$\le \sum_{\ell=1}^{L} B^{L-\ell} \left\| \mathrm{TF}^{R}_{(\ell)}\left(\mathrm{TF}^{R}_{e(1:\ell-1)}(H)\right) - \mathrm{TF}^{R}_{e(\ell)}\left(\mathrm{TF}^{R}_{e(1:\ell-1)}(H)\right) \right\|_{2,\infty}$$
$$\le \sum_{\ell=1}^{L} B_H^{L-\ell} B \left\| \mathrm{TF}^{(\ell)}(H) - \mathrm{TF}^{e(\ell)}(H) \right\| \le L B_H^{L-1} B \varepsilon,$$

where the second inequality follows from Corollary L.1, and the last inequality is because $B_H \ge 1$. □

L.2 Proof of Theorem K.1

In this section, we prove a slightly more general result by considering the general ICL loss

$$\ell_{\mathrm{icl}}(\mathrm{TF}; Z) := \ell(\mathrm{TF}(X), \mathrm{read}_y(\mathrm{TF}^R(H)), y_{N+1}).$$

We assume that the loss function ℓ satisfies $\sup_{|y|\le B_y^0} |\ell(y,y)| \le B_\ell^0$ and $\sup_{|y|\le B_y^0} |\partial_1 \ell(y,y)| \le B_\ell^1$. For the special case $\ell(s,t) = \frac{1}{2}(s-t)^2$, we can take $B_\ell^0 = 4B_y^2, B_\ell^1 = 2B_y$.

We then consider

$$X := \frac{1}{n}\sum_{j=1}^{n} \ell_{\mathrm{icl}}(\mathrm{TF}; Z^j) - \mathbb{E}_Z[\ell_{\mathrm{icl}}(\mathrm{TF}; Z)],$$

where $Z^{(1:n)}$ are i.i.d copies of $Z \sim P$. It remains to apply Proposition B.4 to the random process $\{X_{\mathrm{TF}}\}$. We verify the preconditions:

(a) By [87, Example 5.8], it holds that $\log N(\varepsilon, B_{\|\cdot\|}(r), \|\cdot\|) \le L(3MD^2 + 2DD')\log(1 + 2r/\varepsilon)$, where $B_{\|\cdot\|}(r)$ is any ball of radius r under norm $\|\cdot\|$.

(b) $|\ell_{\mathrm{icl}}(\mathrm{TF}; Z)| \le B_\ell^0$ and hence B_ℓ^0-sub-Gaussian.

(c) $|\ell_{\mathrm{icl}}(\mathrm{TF}; Z) - \ell_{\mathrm{icl}}(\mathrm{TF}^e; Z)| \le B_\ell^1 \cdot (L B_H^{L-1} B) \cdot \varepsilon$, by Proposition L.1.

Therefore, we can apply the uniform concentration result in Proposition B.4 to obtain that, with probability at least $1-\delta$,

$$\sup_{\mathrm{TF}} |X_{\mathrm{TF}}| \le C B_\ell^0 \sqrt{\frac{L(MD^2 + DD') + \log(1/\delta)}{n}},$$

where $C = \log(2 + B \cdot L B_H^{L-1} B \cdot B_\ell^1/B_\ell^0) \le 20L \log(2 + \max\{B, R, B_\ell^1/B_\ell^0\})$. Recalling that

$$L_{\mathrm{icl}}(\mathrm{TF}) \le \inf_{\mathrm{TF}} L_{\mathrm{icl}}(\mathrm{TF}) + 2\sup_{\mathrm{TF}} |X_{\mathrm{TF}}|$$

completes the proof. □

L.3 Proof of Theorem K.2

By Corollary 5, there exists a transformer TF such that for every P satisfying Assumption A with canonical parameters (and thus in expectation over $P \sim \mathcal{P}$) and every $N \ge \Theta(d)$, it outputs prediction $\hat{y}_{N+1} = \mathrm{read}_y(\mathrm{TF}(H))$ such that

$$L_{\mathrm{icl}}(\mathrm{TF}) = \mathbb{E}_{P\sim\mathcal{P}, (\mathcal{D}, \mathbf{x}_{N+1}, y_{N+1}\sim P)} \frac{1}{2}(\hat{y}_{N+1} - y_{N+1})^2 \le \mathbb{E}_{P\sim\mathcal{P}}[L_P(w_P^?)] + O\left(\frac{d}{N}\right)^2,$$

where we recall that $L_P(w_P^?) := \frac{1}{2}\mathbb{E}_{(\mathbf{x},y)\sim P}(y - \langle w_P^?, \mathbf{x}\rangle)^2$. By inspecting the proof, the same result holds if we change TF to the clipped version TF^R if we choose $R^2 = O(B_x^2 + B_y^2 + B_w^2 + 1) = O(d + \cdot)$, so that on the good event $E_{\mathrm{cov}} \setminus E_w$ considered therein, all intermediate outputs within

TF has $k k_{2;\infty}$ R and thus the clipping does not modify the transformer output on $E_{\text{cov}} \setminus E_w$. Further, recall by (33) that has size bounds

$$L \quad O \quad \log \frac{N}{\quad} \; ; \quad \max_{`\in[L]} M^{(`)} \quad 3; \quad \text{jjjj} \quad O(\sqrt{d}):$$

We can thus apply Theorem K.1 to obtain that the solution $\hat{}$ to (TF-ERM) with the above choice of $(L; M; B)$ and $D' = 0$ (attention-only) satisfies the following with probability at least 1 :

$$L_{\text{icl}}(\hat{}) \quad {}^{0\in} \inf_{L;M;D^0;B} L_{\text{icl}}(\; ') + O\left(\sqrt{\frac{L^2 M D^2 \; + \log(1=\;)}{n}}\right)$$

$$L_{\text{icl}}(\;) + \tilde{O}\left(\sqrt{\frac{L^2 M D^2 + \log(1=\;)}{n}}\right) \quad \tilde{O}\left(\sqrt{\frac{{}^2 d^2 + \log(1=\;)}{n}} + \frac{d}{N}^2\right) :$$

Above, $= O(\log(1 + \max\{B_y; R; Bg)) = \tilde{O}(1)$. This finishes the proof. □

L.4 Proof of Theorem K.3

We invoke Theorem 8 (using the construction in Theorem 7 with a different choice of L) with the following parameters:

$$L = \tilde{O}(B_w^?)^2 = {}^2 \quad (1 + d=N) \; = \tilde{O}\left({}^2(1 + d=N)\right) ; \quad M = \quad (1); \quad D' = 2d;$$

$$B_x = \tilde{O}(\sqrt{d}); \quad B_y = \tilde{O}(B_w^? + \quad); \qquad = \quad {}^2\frac{1}{B_y^2 N}{}^2 ;$$

$$\text{jjjj} \quad B = O\left(R + (1 + \quad_N)^{-1} \quad O\left(R + \sqrt{\log d}\right) \quad \tilde{O}(\text{poly}(d; B_w^?; \quad))\right);$$

where $\tilde{O}(\;)$ hides polylogarithmic factors in $d; N; B_w^?; \quad$.

Then, Theorem 8 shows that there exists a transformer with L layers, $\max_{`\in[L]} M^{(`)} \quad M$ heads, D' hidden dimension for the MLP layers, and $\text{jjjj} \quad B$ such that, on almost surely every P , it returns a prediction $\hat{y}_{N+1}$ such that, on the good event E_0 considered therein (over D P) which satisfies $P(E_0) \quad 1$,

$$E_{(\mathbf{x}_{N+1};y_{N+1})\sim P}\left[(\hat{y}_{N+1} \quad y_{N+1})^2\right] \quad {}^2[1 + O(s \log(d=\;)=N)]:$$

By inspecting the proof, the same result holds if we change TF to the clipped version TF^R if we choose $R^2 = O(B_x^2 + B_y^2 + (B_w^?)^2 + 1) = O(d + (B_w^?)^2 + \quad ^2)$, so that on the good event E_0 considered therein, all intermediate outputs within TF has $k k_{2;\infty}$ R and thus the clipping does not modify the transformer output on the good event. On the bad event E_0^c, using the same argument as in the proof of Theorem 8, we have

$$E_{\mathcal{D};(\mathbf{x}_{N+1};y_{N+1})\sim P}\left[1_{E_0^c}(\hat{y}_{N+1} \quad y_{N+1})^2\right] \quad \sqrt{P_{\mathcal{D}}(E_0^c)} \quad 8E_{y_{N+1}\sim P}\left[B_y^4 + y_{N+1}^4\right]^{1=2} \quad \tilde{O}\left(\frac{}{N}^2\right) :$$

Combining the above two bounds and further taking expectation over P gives

$$L_{\text{icl}}(\;) = E_{P\sim \; ;(\mathcal{D};\mathbf{x}_{N+1};y_{N+1})\sim P}\left[\frac{1}{2}(\hat{y}_{N+1} \quad y_{N+1})^2\right] \quad {}^2 + \tilde{O}\left({}^2 s \log d=N\right) :$$

We can thus apply Theorem K.1 to obtain that the solution $\hat{}$ to (TF-ERM) with the above choice of $(L; M; B; D')$ satisfies the following with probability at least 1 :

$$L_{\text{icl}}(\hat{}) \quad {}^{0\in} \inf_{L;M;D^0;B} L_{\text{icl}}(\; ') + O\left(\sqrt{\frac{L^2(M D^2 + D D') \; + \log(1=\;)}{n}}\right)$$

$$L_{\text{icl}}(\;) + \tilde{O}\left(\sqrt{\frac{L^2(M D^2 + D D') + \log(1=\;)}{n}}\right)$$

$${}^2 + \tilde{O}\left(\sqrt{\frac{{}^4 d^2(1 + d=N)^2 + \log(1=\;)}{n}} + {}^2\frac{s \log d}{N}\right) :$$

Above, $= O(\log(1 + \max\{B_y; R; Bg)) = \tilde{O}(1)$. This finishes the proof. □

L.5 Proof of Theorem K.4

We invoke Theorem 12 and Theorem J.1, which shows that (recalling the input dimension $D = (Kd)$) there exists a transformer with the following size bounds:

$$L \quad O\left(\tfrac{-2}{\min}\log(N=\ _{\min})\right); \qquad \max_{`\in[L]} M^{(`)} \quad M = O(K); \qquad \max_{`\in[L]} D^{(`)} \quad D' = O(K^2);$$

$$\text{jjjj} \quad O(\ _{\max} Kd\log(N));$$

such that it outputs $\hat{y}_{N+1}$ that satisfies

$$\mathsf{E}\ \frac{1}{2}(y_{N+1} \quad \hat{y}_{N+1})^2 \qquad \text{BayesRisk} \ + \mathcal{O}\left(\frac{\frac{2}{\max}}{\min_{2=3}}\ \frac{\log K}{N}\right)^{1=3}!:$$

By inspecting the proof, the same result holds if we change TF to the clipped version TF^{R} if we choose $\mathsf{R}^2 = O(B_x^2 + B_y^2 + (B_w^?)^2 + 1) = O(d + \ ^2_{\max})$, so that on the good event considered therein, all intermediate outputs within TF has $k\,k_{2;\infty}$ R and thus the clipping does not modify the transformer output on the good event. Using this clipping radius, we obtain

$$L_{\text{icl}}(\ ) = \mathsf{E}_{\mathsf{P}\sim\ ;(\mathcal{D};\mathbf{x}_{N+1};y_{N+1})\sim\mathsf{P}}\ \frac{1}{2}(\hat{y}_{N+1} \quad y_{N+1})^2 \qquad \text{BayesRisk} \ + \mathcal{O}\left(\frac{\frac{2}{\max}}{\min_{2=3}}\ \frac{\log K}{N}\right)^{1=3}!:$$

We can thus apply Theorem K.1 to obtain that the solution $\hat{}$ to (TF-ERM) with the above choice of $(L; M; B; D')$ satisfies the following with probability at least 1 :

$$L_{\text{icl}}(\hat{}) \quad {}^{0}\in \inf_{L;M;D^0;B} L_{\text{icl}}(\ ') + O\left(\sqrt{\frac{L^2(MD^2 + DD') \ + \log(1=\ )}{n}}\right)!$$

$$L_{\text{icl}}(\ ) + \mathcal{O}\left(\sqrt{\frac{L^2(MD^2 + DD') + \log(1=\ )}{n}}\right)!$$

$$0\ \text{s}\ \frac{}{\text{BayesRisk} \ + \mathcal{O}\left(\frac{\tfrac{-4}{\min}K^3d^2 + \log(1=\ )}{n} + \frac{\frac{2}{\max}}{\min_{2=3}}\ \frac{\log K}{N}\right)^{1=3}}\ 1\ \text{A}:$$

Above, $= O(\log(1 + \max\{B_y; \mathsf{R}; B\})) = \mathcal{O}(1)$. This finishes the proof. □

L.6 Proof of Theorem K.5

The proof follows from similar arguments as of Theorem K.3 and Theorem K.4, where we plug in the size bounds (number of layers, heads, and weight norms) from Theorem G.2 and Corollary G.1. □

M Experimental details and additional studies

M.1 Additional details for Section 6

Architecture and optimization We train a 12-layer encoder-only transformer, where each layer consists of an attention layer as in Definition 1 with $M = 8$ heads, hidden dimension $D = 64$, and ReLU activation (normalized by the sequence length), as well as an MLP layer as in Definition 2 hidden dimension $D' = 64$. We add Layer Normalization [3] after each attention and MLP layer to help optimization, as in standard implementations [84]. We append linear read-in layer and linear read-out layer before and after the transformer respectively, both applying a same affine transform to all tokens in the sequence and are trainable. The read-in layer maps any input vector to a D-dimensional hidden state, and the read-out layer maps a D-dimensional hidden state to a 1-dimensional scalar.

Each training sequence corresponds to a single ICL instance with N in-context training examples $\{(\mathbf{x}_i; y_i)\}_{i=1}^N$ R^d R and test input $\mathbf{x}_{N+1}$ 2 R^d. The input to the transformer is formatted as

in (3) where each token has dimension $d + 1$ (no zero-padding). The transformer is trained by minimizing the following loss with fresh mini-batches:

$$L(\ ) = \mathsf{E}_{\mathsf{P}\sim\ ;(\mathbf{H};y_{N+1})\sim\mathsf{P}}[\grave{}_{\mathsf{P}}(\mathrm{read}_{\mathsf{y}}(\mathsf{TF}\ (\mathsf{H})); y_{N+1})]; \tag{47}$$

where the loss function $\grave{}_{\mathsf{P}} : \mathsf{R}^2\ !\ \mathsf{R}$ may depend on the training data distribution P in general; we use the square loss when P is regression data, and the logistic loss when P is classification data. We use the Adam optimizer with a fixed learning rate 10^{-4}, which we find works well for all our experiments. Throughout all our experiments except for the sparse linear regression experiment in Figure 3a, we train the model for 300K steps, where each step consists of a (fresh) minibatch with batch size 64 in the base mode, and K minibatches each with batch size 64 in the mixture mode.

For the sparse linear regression experiment, we find that minimizing the training objective (47) alone was not enough, e.g. for the learned transformer to achieve better loss than the least squares algorithm (which achieves much higher test loss than the Lasso; cf. Figure 3a). To help optimization, we augment (47) with another loss that encourages the second-to-last hidden states to recover the true (sparse) coefficient $\mathsf{w}_?$:

$$L_{\text{t-w}}(\ ) = \frac{1}{N_0}\sum_{j=1}^{N_0} \mathsf{E}_{\mathsf{P}=\mathsf{P}_{\mathsf{w}^?}\sim\ ;(\mathbf{H};y_{N+1})\sim\mathsf{P}} \left\| \left[\mathsf{TF}^{(1:L-1)}(\mathsf{H})\right]_{j;(D-d+1):D} - \mathsf{w}^?\right\|_2^2; \tag{48}$$

Specifically, the above loss encourages the first N_0 N tokens within the second-to-last layer to be close to $\mathsf{w}^?$. We choose $N_0 = 5$ (recall that the total number of tokens is $N = 10$ and sequence length is $N + 1 = 11$ for this experiment). We minimize the loss $L(\ ) + L_{\text{t-w}}(\ )$ with $= 0{:}1$ for 2M steps for this task.

Evaluation All evaluations are done on the trained transformer with 6400 test instances. We use the square loss for regression tasks, and the classification error (1 accuracy) between the true label $y_{N+1}\ 2\ f0; 1g$ and the predicted label $1\mathit{fy}_{N+1}\ 1{=}2g$. We report the means in all experiments, as well as their standard deviations (using one-std error bars) in Figure 2a, 2b, 5a, 5b. In Figure 2c, 3b, 3c 5c, all standard deviations are sufficiently small (not significantly exceeding the width of the markers), thus we did not show error bars in those plots.

Baseline algorithms We implement various baseline machine learning algorithms to compare with the learned transformers. A superset of the algorithms is shown in Figure 3a:

- `Least squares`, `Logistic regression`: Standard algorithms for linear regression and linear classification, respectively. Note that least squares is also a valid algorithm for classification.
- `Averaging`: The simple algorithm which computes the linear predictor $\hat{\mathsf{w}} = \frac{1}{N}\sum_{i=1}^{N} y_i\mathsf{x}_i$ and predicts $\mathit{y}_{N+1} = h\hat{\mathsf{w}}; \mathsf{x}_{N+1}i$;
- `3-NN`: 3-Nearest Neighbors.
- `Ridge`: Standard ridge regression as in (ICRidge). We specifically consider two 's (denoted as `lam_1` and `lam_2`): $_1;\ _2 = (0{:}005; 0{:}125)$. These are the Bayes-optimal regularization strengths for the noise levels $(\ _1;\ _2) = (0{:}1; 0{:}5)$ respectively under the noisy linear model (cf. Corollary 6), using the formula $^? = d\ ^2{=}N$, with $(d; N) = (20; 40)$.
- `Lasso`: Standard Lasso as in (ICLasso) with $2\ f1; 0{:}1; 0{:}01; 0{:}001g$.

In Figure 2c, the `ridge_analytical` curve plots the expected risk of ridge regression under the noisy linear model over 20 geometrically spaced values of 's in between $(\ _1;\ _2)$, using analytical formulae (with Monte Carlo simulations). The `Bayes_err_{1,2}` indicate the expected risks of $_1$ on task 1 (with noise $_1$) and $_2$ on task 2 (with noise $_2$), respectively.

M.2 Decoder-based architecture

ICL capabilities have also been demonstrated in the literature for decoder-based architectures [31, 2, 47]. There, the transformer can do in-context predictions at every token x_i using past tokens $f(\mathsf{x}_j; \mathsf{y}_j)g_{j\le i-1}$ as training examples. Here we show that such architectures is also able to perform in-context algorithm selection *at every token*; For results for this architecture on "base" ICL tasks (such as those considered in Figure 3a), we refer the readers to Garg et al. [31].

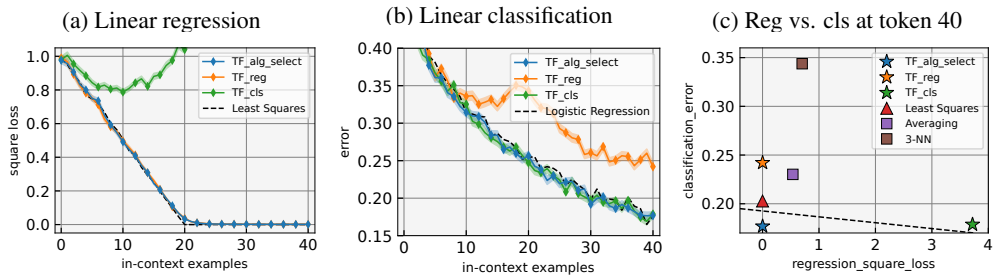

Figure 5: In-context algorithm selection abilities of transformers between linear regression and linear classification. *(a,b)* On these two tasks, a **single transformer** `TF_alg_select` **simultaneously approaches the performance of the strongest baseline algorithm** `Least Squares` on linear regression and `Logistic Regression` on linear classification. *(c)* At token 40 (using example $f0;:::;39g$ for training), `TF_alg_select` matches the performance of the best baseline algorithm for both tasks. *(a,b,c)* Note that transformers pretrained on a single task (`TF_reg`, `TF_cls`) perform near-optimally on their pretraining task but suboptimally on the other task.

Setup Our setup is the same as the two "mixture" modes (linear model + linear classification model, and noisy linear models with two different noise levels) as in Section 6, except that the architecture is GPT-2 following Garg et al. [31], and the input format is changed to (11) (so that the input sequence has $2N + 1$ tokens) without positional encodings. For every i 2 $[N + 1]$, we extract the prediction $\hat{y}_i$ using a linear read-out function applied on output token $2i$ 1, and the (learnable) linear read-out function is the same across all tokens, similar as in Appendix M.1. The rest of the setup (optimization, training, and evaluation) is the same as in Section 6 & M.1. Note that we also train on the objective (47) for all tokens averaged, instead of for the last test token as in Section 6.

Result Figure 2 shows the results for noisy linear models with two different noise levels, and Figure 5 shows the results for linear model + linear classification model. We observe that at every token, In both cases, `TF_alg_select` nearly matches the strongest baseline for both tasks simultaneously, whereas transformers trained on a single task perform suboptimally on the other task. Further, this phenomenon consistently shows up at every token. For example, in Figure 2a & 2b, `TF_alg_select` matches ridge regression with the optimal on all tokens i 2 $f1;:::;Ng$ (N = 40). In Figure 5a & 5b, `TF_alg_select` matches least squares on the regression task and logistic regression on the classification task on all tokens i 2 $[N]$. This demonstrates the in-context algorithm selection capabilities of standard decoder-based transformer architectures.

M.3 Computational resource

All our experiments are performed on 8 Nvidia Tesla A100 GPUs (40GB memory). The total GPU time is approximately 5 days (on 8 GPUs), with the largest individual training run taking about a single day on a single GPU.