How Transformers Utilize Multi-Head Attention in In-Context Learning? A Case Study on Sparse Linear Regression

Anonymous Authors[1]

Abstract

In this study, we investigate how a trained multi-head transformer performs in-context learning on sparse linear regression. We experimentally discover distinct patterns in multi-head utilization across layers: multiple heads are essential in the first layer, while subsequent layers predominantly utilize a single head. We propose that the first layer preprocesses input data, while later layers execute simple optimization steps on the preprocessed data. Theoretically, we prove such a preprocess-then-optimize algorithm can outperform naive gradient descent and ridge regression, corroborated by experiments. Our findings provide insights into the benefits of multi-head attention and the intricate mechanisms within trained transformers.

1. Introduction

Transformers(Vaswani et al., 2023) have shown remarkable performance in natural language processing (Ouyang et al., 2022; Achiam et al., 2023; Brown et al., 2020; Radford et al., 2019) and other domains (Dosovitskiy et al., 2020; Peebles & Xie, 2023), exhibiting capabilities like in-context learning(Brown et al., 2020; Xie et al., 2021). While numerous studies have explored transformers' expressive power(Kajitsuka & Sato, 2023; Takakura & Suzuki, 2023) and ability to emulate algorithms(Guo et al., 2023; Bai et al., 2023; Li et al.; Chen & Zou, 2024), understanding their inner workings remains a challenge, especially the roles of different attention layers and heads.

This work investigates how transformers utilize multi-head attention across layers for in-context learning on sparse linear regression tasks. Empirically, we observed a distinct pattern: while the first attention layer utilized all heads evenly, subsequent layers predominantly relied on a single head. This suggests different working mechanisms for the first and later layers. Based on these findings, we propose that transformers employ a preprocess-then-optimize algorithm: the first layer preprocesses input data using multiple heads, then subsequent layers perform iterative optimization (e.g., gradient descent) on the preprocessed data using a single head. We theoretically demonstrate that such an algorithm can be implemented by a modestly-sized transformer and achieve lower excess risk than traditional methods like gradient descent and ridge regression without preprocessing.

Our main contributions are:

- We empirically revealing the distinct head utilization patterns across layers.
- Building upon our empirical findings, we proposed a possible working mechanism for multi-head transformers.
- We further validated our proposed mechanism by theoretical analysis.
- We conducted additional experiments to further validate our theoretical framework.

2. Preliminaries

Sparse Linear Regression. We consider sparse linear models where $(\mathbf{x}, y) \sim \mathsf{P} = \mathsf{P}^{\mathsf{lin}}_{\mathbf{w}^\star}$ is sampled as $\mathbf{x} \sim \mathsf{N}(\mathbf{0}, \mathbf{\Sigma})$, $y = \langle \mathbf{w}^\star, \mathbf{x} \rangle + \mathsf{N}(0, \sigma^2)$, where the $\mathbf{\Sigma}$ is a diagonal matrix and ground truth $\mathbf{w}^\star \in \mathbb{R}^d$ satisfies $\|\mathbf{w}^\star\|_0 \leq s$. Then, we define the population risk of a parameter $\mathbf{w}$ as:

$$L(\mathbf{w}) := \mathbb{E}_{(\mathbf{x},y)\sim\mathsf{P}}\big[(\langle \mathbf{x}, \mathbf{w} \rangle - y)^2\big].$$

Moreover, we are interested in the excess risk:

$$\mathcal{E}(\mathbf{w}) := L(\mathbf{w}) - \min_{\mathbf{w}} L(\mathbf{w}).$$

Linear Attention-only Transformers To perform an intractable theoretical investigation on the role of multi-head in the attention layer, we make simplifications on the transformer model by considering linear attention-only transformers. These simplifications are widely adopted in many

[1] Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

recent works to study the behavior of transformer models (von Oswald et al., 2023; Zhang et al., 2023; Mahankali et al., 2023; Ahn et al.; Wu et al., 2023). In particular, the i-th layer TF_i performs the following update on the input sequence (or hidden state) $\mathbf{H}^{(i-1)}$ as follows:

$$\begin{aligned} \mathbf{H}^{(i)} &= \mathsf{TF}_i(\mathbf{H}^{(i-1)}) \\ &= \mathbf{W}_1\big(\mathbf{H}^{(i-1)} + \mathsf{Concat}[\{\mathbf{V}_i\mathbf{M}\mathbf{K}_i^\top\mathbf{Q}_i\}_{i=1}^h]\big), \\ \mathbf{M} &:= \begin{pmatrix} \mathbf{I}_n & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix} \in \mathbb{R}^{m\times m}, \end{aligned} \tag{2.1}$$

where $\{\mathbf{W}_{V_i}, \mathbf{W}_{K_i}, \mathbf{W}_{Q_i} \in \mathbb{R}^{\frac{d_{\mathsf{hid}}}{h}\times d_{\mathsf{hid}}}\}_{i=1}^h$ and $\mathbf{W}_1 \in \mathbb{R}^{d_{\mathsf{hid}}\times d_{\mathsf{hid}}}$ are learnable parameters. Besides, the mask matrix $\mathbf{M}$ is included in the attention to constrain the model focus the first n in-context examples rather than the subsequent $m-n$ queries (Ahn et al.; Mahankali et al., 2023). To adapt the transformer for solving sparse linear regression problems, we introduce additional linear layers $\mathbf{W}_E \in \mathbb{R}^{(d+1)\times d_{\mathsf{hid}}}$ and $\mathbf{W}_O \in \mathbb{R}^{d_{\mathsf{hid}}\times 1}$ for input embedding and output projection, respectively. Mathematically, let $\mathbf{E}$ denotes the input sequences with n in-context example followed by q queries,

$$\mathbf{E} = \begin{pmatrix} \mathbf{x}_1 & \mathbf{x}_2 & \cdots & \mathbf{x}_n & \mathbf{x}_{n+1} & \cdots & \mathbf{x}_{n+q} \\ y_1 & y_2 & \cdots & y_n & 0 & \cdots & 0 \end{pmatrix}. \tag{2.2}$$

Then model processes the input sequence $\mathbf{E}$, resulting in the output $\widehat{\mathbf{y}} \in \mathbb{R}^{1\times(n+q)}$:

$$\widehat{\mathbf{y}} = \mathbf{W}_O \circ \mathsf{TF}_L \circ \cdots \circ \mathsf{TF}_1 \circ \mathbf{W}_E(\mathbf{E}),$$

here, L is the layer number of the transformer, and $\widehat{y}_{i+n}$ is the prediction value for the query $\mathbf{x}_{i+n}$. During training, we set $q > 1$ for efficiency, and for inference and theoretical analysis, we set $q = 1$ and define the in-context learning excess risk $\mathcal{E}_{\mathsf{ICL}}$ as:

$$\mathcal{E}_{\mathsf{ICL}} := \mathbb{E}_{(\mathbf{x},y)\sim\mathsf{P}}(\widehat{y}_{n+1} - y_{n+1})^2 - \sigma^2.$$

3. Experimental Insights into Multi-head Attention for In-context Learning

To understand the hidden mechanism behind the trained transformer, we design a series of experiments, utilizing techniques like probing (Alain & Bengio, 2016) and pruning(Li et al., 2017) to help us gain initial insights into how the trained transformer utilizes multi-head attention.

ICL with Varying Heads: This experiment investigates the performance of transformers in solving in-context sparse linear regression with different numbers of attention heads. An example can be found in Figure 1b, where we display the excess risk for different models when using different numbers of in-context examples. We can observe that given few-shot in-context examples, transformers can outperform OLS and ridge. Moreover, we can also clearly observe the benefit of using multiple heads, which leads to lower excess risk when increasing the number of heads. This **highlights the importance of multi-head attention in transformer to perform in-context learning**.

Heads Assessment: This experiment evaluates the importance of each attention head by masking individual heads and measuring the change in risk. An example can be found in Figure 1c. We find that in the first layer, no head distinctly outweighs the others, while in the subsequent layers, there always exists a head that exhibits higher importance than others. This gives us insight that **in the first attention layer, all heads appear to be significant, while in the subsequent layers, only one head appears to be significant**.

Pruning and Probing: Based on the previous findings, this experiment prunes the trained model by retaining all heads in the first layer and only keeping the most important head in subsequent layers. The pruned model is then fine-tuned. Linear probes are used to evaluate the prediction performance of different layers. An example can be found in Figure 1d, it shows that the pruned model performs similarly to the original model, but differently from a single-head transformer. Noting that the main difference between them is the number of heads in the first layer (subsequent layers have the same structure), it can be deduced that **the working mechanisms of the multi-head transformer may be different for the first and subsequent layers**.

4. Potential Mechanism Behind Trained transformer

Based on the experimental insights from Section 3, we found that while all heads in the first layer are crucial, only one head plays a significant role in subsequent layers. Additionally, probing and pruning results suggest different working mechanisms for the first and subsequent layers. To this end, we hypothesize that the multi-layer transformer implements a preprocess-then-optimize approach for in-context learning, where the first layer preprocesses the in-context examples, and subsequent layers implement iterative optimization algorithms on the preprocessed data.

4.1. Preprocessing on In-context Examples

As the multihead attention is designed to facilitate to model to capture features from different representation subspaces (Vaswani et al., 2023), we abstract the algorithm implementation by the first layer of the transformers as a preprocessing procedure. In general, for the sparse linear regression, a

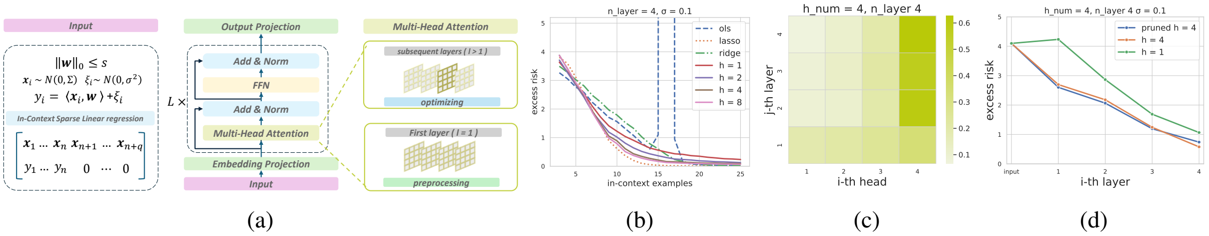


Figure 1: Experimental insights into multi-head attention for in-context learning. *(a)*: Overview of the experiments, including task, data, architecture, and our insights. *(b)*: ICL with Varying Heads. *(c)*: Heads Assessment. *(d)*: Pruning and Probing.

possible data preprocessing method is to perform reweighting of the data features by emphasizing the features that correspond to the nonzero entries of the ground truth $\mathbf{w}^*$ and disregard the remaining features. In the idealized case, if we know the nonzero support of $\mathbf{w}^*$, we can trivially zero out the date features of $\mathbf{x}$ on the complement of the nonzero support, as a data preprocessing procedure, and perform projected gradient descent to obtain the optimal solution. Although the nonzero support of $\mathbf{w}^*$ is intractable to the learner, we can estimate each dimension w_i^* by $\frac{1}{n}\sum_{i=1}^n x_{ij}y_i$, as $\mathbb{E}[x_i y] = \mathbb{E}[\sum_{i=1}^d w_i^* x_i \cdot x_i] + \mathbb{E}[\xi x_i] = w_i^* \mathbb{E}[x_i^2]$, resulting in Alg. 1.

Algorithm 1 Data preprocessing for in-context examples

```
1: Input : Sequence with $\{(\mathbf{x}_i, y_i)\}_{i=1}^n, \{(\mathbf{x}_i, 0)\}_{i=n+1}^{n+q}$ as in-context examples/queries.
2: for $k = 1, \dots, n$ do
3:     Compute $\widetilde{\mathbf{x}}_k$ by $\widetilde{\mathbf{x}}_k = \widehat{\mathbf{R}}\mathbf{x}_k$,
       where $\widehat{\mathbf{R}} = \mathrm{diag}\{\widehat{r}_1, \widehat{r}_2, \dots, \widehat{r}_d\}$, where $\widehat{r}_j$ is given by
           $\widehat{r}_j = \frac{1}{n}\sum_{i=1}^n x_{ij}y_i.$    (4.1)
4: end for
5: Output : Sequence with the preprocessed in-context examples/queries $\{(\widetilde{\mathbf{x}}_i, y_i)\}_{i=1}^n, \{(\widetilde{\mathbf{x}}_i, 0)\}_{i=n+1}^{n+q}$ .
```

4.2. Optimizing Over Preprocessed In-Context Examples

Based on the experimental results, we observe that the subsequent layers of transformers dominantly rely on one single head, suggesting their different but potentially simpler behavior compared to the first one. Motivated by a series of recent work (von Oswald et al., 2023; Cheng et al., 2023; Zhang et al., 2023) that reveal the connection between gradient descent steps and multi-layer single-head transformer in the in-context learning tasks, we conjecture that the subsequent layers also implement iterative optimization algorithms, e.g., gradient descent algorithm, on the (preprocessed) in-context examples.

We further prove that these two procedures (preprocessing then optimizing) can be implemented by linear attention-only transformers in Propositions C.1 and C.2 (presented in Appendix). More details about our preprocess-then-optimize algorithm can be found in Appendix C.

5. Excess Risk of the Preprocess-then-optimize Algorithm

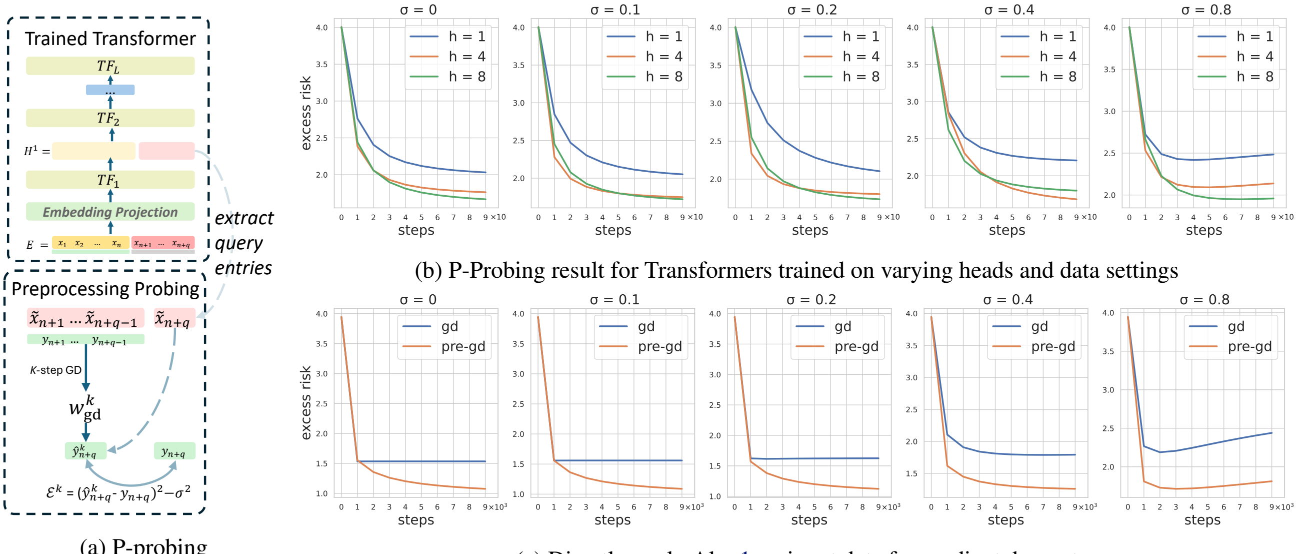
In this section, we will develop the theory to demonstrate the improved performance of the preprocess-then-optimize algorithm compared to the gradient descent algorithm on the raw inputs. Appendix E provides a more detailed analysis.

We first denote $\widetilde{\mathbf{w}}_{\mathsf{gd}}^t$ as the estimator obtained by t-step GD on $\{(\widetilde{\mathbf{x}}_i, y_i)\}_{i=1}^n$, which can be viewed as the solution generated by the $t+1$-layer transformer based on our discussion in Section 4, and $\mathbf{w}_{\mathsf{gd}}^t$ as the estimator obtained by t-step GD on $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$. Before presenting our main theorem, we first need to redefine the excess risk of GD on $\{(\widetilde{\mathbf{x}}_i, y_i)\}_{i=1}^n$. Note that in our algorithm, the learned predictor takes the form $\mathbf{x} \to \langle \widehat{\mathbf{R}}\mathbf{x}, \widetilde{\mathbf{w}}_{\mathsf{gd}}^t \rangle$. Consequently, the population risk of a parameter $\widetilde{\mathbf{w}}_{\mathsf{gd}}^t$ is naturally defined as $\widetilde{L}(\widetilde{\mathbf{w}}_{\mathsf{gd}}^t) := \frac{1}{2} \cdot \mathbb{E}_{(\mathbf{x},y)\sim \mathsf{P}}\left[(\langle \widehat{\mathbf{R}}\mathbf{x}, \widetilde{\mathbf{w}}_{\mathsf{gd}}^t \rangle - y)^2\right]$, and the excess risk is then defined as $\mathcal{E}(\mathbf{w}) := \widetilde{L}(\mathbf{w}) - \min_{\mathbf{w}} \widetilde{L}(\mathbf{w})$ [1].

To make a formal comparison between preprocess-then-optimize and baselines, we consider the example where $x_i \overset{\text{i.i.d.}}{\sim} \mathsf{N}(\mathbf{0}, \mathbf{I})$, based on which we can get the upper bound for our algorithm and the lower bound for OLS, ridge regression, and finite-step GD.

Theorem 5.1. *Suppose $\mathcal{S}$ with $|\mathcal{S}| = s$ is selected such that each element is chosen with equal probability from the set $\{1, 2, \dots, d\}$ and $w_i^\star \sim \mathsf{U}\{-1/\sqrt{s}, 1/\sqrt{s}\}$ has a restricted uniform prior for $i \in \mathcal{S}$, $\|\mathbf{w}^\star\|_2 \simeq \Theta(1)$ and $n \gtrsim t^2 s^3 d^{2/3}$.*

[1] Here for the ease to presentation and comparison, we slightly abuse the notation of $\mathcal{E}(\mathbf{w})$ by extending it to $\widetilde{\mathbf{w}}_{\mathsf{gd}}^t$, although $\mathcal{E}(\mathbf{w})$ is originally defined for the estimator for the raw feature vector $\mathbf{x}$.



(a) P-probing

(b) P-Probing result for Transformers trained on varying heads and data settings

(c) Directly apply Alg. 1 on input data for gradient descent

Figure 2: Supporting experiments for our preprocess-then-optimize algorithm and theoretical analysis

Then there exists a choice of η and t such that

$$\mathcal{E}\left(\widetilde{\mathbf{w}}_{\text{gd}}^t\right) \lesssim \sigma^2 \log^2\left(ns/\sigma^2\right)\log^2\left(d/\delta\right)\cdot\left(\frac{s}{n}+\frac{ds^2}{n^2}\right),$$

with probability at least $1-\delta$. Besides, let $\widehat{\mathbf{w}}_\lambda$ be the ridge regression estimator with regularized parameter λ, and $\mathbf{w}_{\text{ols}}$ be the OLS estimator, it holds that

$$\mathbb{E}_{\mathbf{w}^\star}\left[\mathcal{E}(\mathbf{w})\right] \gtrsim \begin{cases} \frac{\sigma^2 d}{n} & n \gtrsim d + \log\left(1/\delta\right) \\ 1-\frac{n}{d}+\frac{\sigma^2 n}{d} & d \gtrsim n + \log\left(1/\delta\right), \end{cases}$$

with probability at least $1-\delta$, where $\mathbf{w} \in \{\widehat{\mathbf{w}}_\lambda, \mathbf{w}_{\text{ols}}, \mathbf{w}_{\text{gd}}^t\}$.

It can be seen that for a wide range of under-parameterized and over-parameterized cases, $\widetilde{\mathbf{w}}_{\text{gd}}^t$ has a smaller excess risk than ridge regression, standard gradient descent, and OLS, when the sparsity s satisfies $s = o\min\{d,n\}$. This justifies the effectiveness of the preprocess-then-optimize algorithm for dealing with the sparse regression problem. Moreover, it is well known that Lasso can achieve $\widetilde{O}(s/n)$ excess risk bound in the setting of Theorem 5.1. Then we can conclude that the proprocess-then-optimize algorithm can be comparable to Lasso up to logarithmic factors when $d \lesssim n$, while becomes worse when $d \gtrsim n$.

6. Experiments

In Section 3, we conduct several experiments, and based on the observations, we propose that a trained transformer can apply a preprocess-then-optimize algorithm. While the second part (gradient descent over context) is supported by extensive theoretical analysis and experimental evidence (von Oswald et al., 2023; Cheng et al., 2023; Zhang et al., 2023; Ahn et al.), here we develop a technique called preprocessing probing (P-probing) on the trained transformer to support the first part of our algorithm, where we try to extract the preprocessed component $\{\widetilde{\mathbf{x}}_i\}_{i=n+1}^{n+q}$ from the first layer of transformer as in, as illustrated in Figure 2a. We also directly apply Alg. 1 on the in-context examples and then check the excess risk for multiple-step gradient descent to verify the effectiveness of our algorithm and theoretical analysis. Experimental details can be found in Appendix A.

Based on Figure 2b, we can observe that compared to the transformer with single-head attention ($h = 1$), the query entries extracted from the transformer with multiple heads ($h = 4, 8$) preserve better convergence performance and can dive into a lower risk. This aligns well with our experiment result in Figure 2c, where compared to $\texttt{gd}$, the data preprocessed by Alg. 1 preserves better convergence performance and can dive into a lower risk space, supporting the existence of the preprocessing procedure in the trained transformer. Moreover, Figure 2c also aligns well with our theoretical analysis, where our algorithm can outperform ridge regression and OLS.

7. Conclusions

In this paper, we investigate a sparse linear regression problem and explore how a trained transformer leverages multi-head attention for in-context learning. Based on the experiment and theoretical characterizations, we show that transformer may implement the preprocess-then-optimize algorithm, by using multiple heads in the first layer and one head in the subsequent layer. Numerical experiments support our findings.

References

Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F. L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Ahn, K., Cheng, X., Daneshmand, H., and Sra, S. Transformers learn to implement preconditioned gradient descent for in-context learning.

Alain, G. and Bengio, Y. Understanding intermediate layers using linear classifier probes. *arXiv preprint arXiv:1610.01644*, 2016.

Allen-Zhu, Z. and Li, Y. Physics of language models: Part 3.2, knowledge manipulation. *arXiv preprint arXiv:2309.14402*, 2023.

Bai, Y., Chen, F., Wang, H., Xiong, C., and Mei, S. Transformers as Statisticians: Provable In-Context Learning with In-Context Algorithm Selection, July 2023.

Bartlett, P. L., Long, P. M., Lugosi, G., and Tsigler, A. Benign overfitting in linear regression. *Proceedings of the National Academy of Sciences*, 117(48):30063–30070, 2020.

Bills, S., Cammarata, N., Mossing, D., Tillman, H., Gao, L., Goh, G., Sutskever, I., Leike, J., Wu, J., and Saunders, W. Language models can explain neurons in language models. *URL https://openaipublic. blob. core. windows. net/neuron-explainer/paper/index. html.(Date accessed: 14.05. 2023)*, 2023.

Bricken, T., Templeton, A., Batson, J., Chen, B., Jermyn, A., Conerly, T., Turner, N., Anil, C., Denison, C., Askell, A., et al. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*, pp. 2, 2023.

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

Chen, X. and Zou, D. What can transformer learn with varying depth? case studies on sequence learning tasks. *arXiv preprint arXiv:2404.01601*, 2024.

Cheng, X., Chen, Y., and Sra, S. Transformers implement functional gradient descent to learn non-linear functions in context. *arXiv preprint arXiv:2312.06528*, 2023.

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

Friedman, D., Wettig, A., and Chen, D. Learning Transformer Programs.

Garg, S., Tsipras, D., Liang, P., and Valiant, G. What Can Transformers Learn In-Context? A Case Study of Simple Function Classes, August 2023.

Guo, T., Hu, W., Mei, S., Wang, H., Xiong, C., Savarese, S., and Bai, Y. How Do Transformers Learn In-Context Beyond Simple Functions? A Case Study on Learning with Representations, October 2023.

Huang, Y., Cheng, Y., and Liang, Y. In-context convergence of transformers. *arXiv preprint arXiv:2310.05249*, 2023.

Kajitsuka, T. and Sato, I. Are Transformers with One Layer Self-Attention Using Low-Rank Weight Matrices Universal Approximators?, July 2023.

Li, H., Kadav, A., Durdanovic, I., Samet, H., and Graf, H. P. Pruning filters for efficient convnets. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=rJqFGTslg.

Li, Y., Ildiz, M. E., Papailiopoulos, D., and Oymak, S. Transformers as Algorithms: Generalization and Stability in In-context Learning.

Li, Y., Li, Y., and Risteski, A. How do transformers learn topic structure: Towards a mechanistic understanding. In *International Conference on Machine Learning*, pp. 19689–19729. PMLR, 2023.

Lindner, D., Kramár, J., Farquhar, S., Rahtz, M., McGrath, T., and Mikulik, V. Tracr: Compiled Transformers as a Laboratory for Interpretability, November 2023.

Mahankali, A., Hashimoto, T. B., and Ma, T. One Step of Gradient Descent is Provably the Optimal In-Context Learner with One Layer of Linear Self-Attention, July 2023.

Nichani, E., Damian, A., and Lee, J. D. How transformers learn causal structure with gradient descent. *arXiv preprint arXiv:2402.14735*, 2024.

Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.

Pandit, O. and Hou, Y. Probing for bridging inference in transformer language models. *arXiv preprint arXiv:2104.09400*, 2021.

Peebles, W. and Xie, S. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4195–4205, 2023.

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

Takakura, S. and Suzuki, T. Approximation and Estimation Ability of Transformers for Sequence-to-Sequence Functions with Infinite Dimensional Input, May 2023.

Tarzanagh, D. A., Li, Y., Thrampoulidis, C., and Oymak, S. Transformers as Support Vector Machines, September 2023.

Tian, Y., Wang, Y., Chen, B., and Du, S. Scan and Snap: Understanding Training Dynamics and Token Composition in 1-layer Transformer, July 2023.

Tsigler, A. and Bartlett, P. L. Benign overfitting in ridge regression. *Journal of Machine Learning Research*, 24 (123):1–76, 2023.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. Attention Is All You Need, August 2023.

Vershynin, R. High-dimensional probability. *University of California, Irvine*, 10:11, 2020.

von Oswald, J., Niklasson, E., Randazzo, E., Sacramento, J., Mordvintsev, A., Zhmoginov, A., and Vladymyrov, M. Transformers learn in-context by gradient descent, May 2023.

Weiss, G., Goldberg, Y., and Yahav, E. Thinking Like Transformers, July 2021.

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., et al. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*, 2019.

Wu, J., Zou, D., Chen, Z., Braverman, V., Gu, Q., and Bartlett, P. How many pretraining tasks are needed for in-context learning of linear regression? In *The Twelfth International Conference on Learning Representations*, 2023.

Wu, Z., Chen, Y., Kao, B., and Liu, Q. Perturbed masking: Parameter-free probing for analyzing and interpreting bert. *arXiv preprint arXiv:2004.14786*, 2020.

Xie, S. M., Raghunathan, A., Liang, P., and Ma, T. An explanation of in-context learning as implicit bayesian inference. In *International Conference on Learning Representations*, 2021.

Zhang, R., Frei, S., and Bartlett, P. L. Trained transformers learn linear models in-context. *arXiv preprint arXiv:2306.09927*, 2023.

Zhou, H., Bradley, A., Littwin, E., Razin, N., Saremi, O., Susskind, J. M., Bengio, S., and Nakkiran, P. What algorithms can transformers learn? a study in length generalization. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=AsIuHnmHX.

Zhu, Z. A. and Li, Y. Physics of language models: Part 3.1, knowledge storage and extraction. *arXiv preprint arXiv:2309.14316*, 2023.

Zou, D., Wu, J., Braverman, V., Gu, Q., Foster, D. P., and Kakade, S. The benefits of implicit regularization from sgd in least squares problems. *Advances in neural information processing systems*, 34:5456–5468, 2021.

Zou, D., Wu, J., Braverman, V., Gu, Q., and Kakade, S. Risk bounds of multi-pass sgd for least squares in the interpolation regime. *Advances in Neural Information Processing Systems*, 35:12909–12920, 2022.

A. Additional Details for Sections 3 and 6

Architecture and Optimization We conduct extensive experiments on encoder-only transformers with $d_{\mathsf{hid}} = 256$, varying the number of heads $h \in \{1, 2, 4, 8\}$, layers $l \in \{3, 4, 5, 6\}$, and noise levels $\sigma \in \{0, 0.1, 0.2, 0.4, 0.8\}$. For the input sequence, we sample $\mathbf{x} \sim \mathsf{N}(\mathbf{0}, \mathbf{I})$. For $\mathbf{w}$, we first sample $\mathbf{w} \sim \mathsf{N}(\mathbf{0}, \mathbf{I}) \in \mathbb{R}^{16}$, and randomly choose $s = 4$ entries, setting the other elements to zero. Note that We don't apply positional encodings in our setting, as no positional information is needed in our input setting. To further support our preprocessing-then-optimize algorithm, we also try a decoder-only architecture(Figure 9) and train models with $s = d = 16$ (Figure 10) as a comparison in Appendix I. During training, we set $n = 12$ and $q = 4$, with a batch size of 64. We utilize the Adam optimizer with a learning rate $\gamma = 10^{-4}$ for 320000 updates. Each experiment takes about two hours on a single NVIDIA GeForce RTX 4090 GPU. We fix the random seed such that each model is trained and evaluated with the same training and evaluation dataset. We use HuggingFace (Wolf et al., 2019) library to implement our models.

ICL with Varying Heads We compare the model's performance with ridge regression, OLS, and lasso. For ridge regression and lasso, we tune $\lambda, \alpha \in \left\{1, 10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}\right\}$ respectively for the lowest risk, as in (Garg et al., 2023).

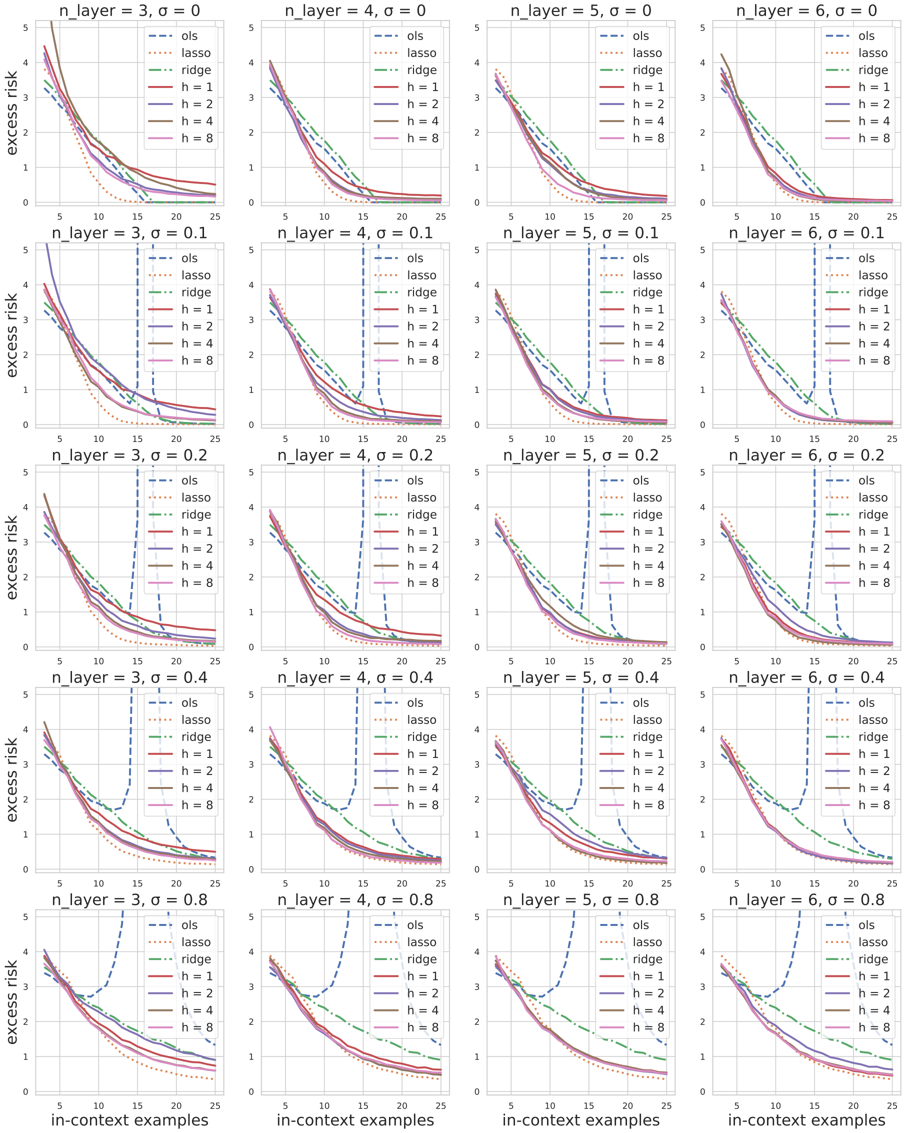


Figure 3: ICL with varying heads, layers and noise levels

From Figure 3, we can find that in most cases, transformers with single head ($h = 1$) exhibits higher risk compared to models with multiple heads ($h = 4, 8$). Note that in thesame subplot, models with different numbers of heads have the same number of parameters. This experiment highlights the importance of multi-head attention for transformers in in-context learning.

Heads Assessment Based on Eq.(2.1), we know that the j-th head at the i-th layer corresponds to the subspace of the intermediate output from $(j-1) \cdot d_{\mathsf{hid}}/h$ to $j \cdot d_{\mathsf{hid}}/h - 1$. To assess the importance of each attention head, we can mask the particular head by zeroing out the corresponding output entries, while keeping other dimensions unchanged. Then, let (i, j) be the layer and head indices, we evaluate the risk change before and after head masking, denoted by $\Delta\mathcal{E}_{\mathsf{ICL}(i,j)}$. Then we normalize the risk changes in the same layer to evaluate their relative importance:

$$\mathcal{W}_{i,j} = \frac{\Delta\mathcal{E}_{\mathsf{ICL}(i,j)}}{\sum_{k=1}^{h} \Delta\mathcal{E}_{\mathsf{ICL}(i,k)}}. \tag{A.1}$$

Here, we set $n = 10$ and $q = 1$, with an evaluation data size of 8192. For a model with h heads and l layers, we train $|\boldsymbol{\sigma}|$ models under different noise levels. We first compute the $\mathcal{W}^{h,l,\sigma}$ under different noise levels σ, then sort each row in $\mathcal{W}^{h,l,\sigma}$, and add them together as $\mathcal{W}^{h,l}_{\mathsf{avg}} = \frac{1}{|\boldsymbol{\sigma}|}\sum_{\sigma\in\boldsymbol{\sigma}} \mathcal{W}^{h,l,\sigma}$, resulting in the final weight for each head. An example can be found in Fig 1c. In Fig 4, we present more results for different h and l, and we also present the heat map for the decode-only transformers in Figure 9.

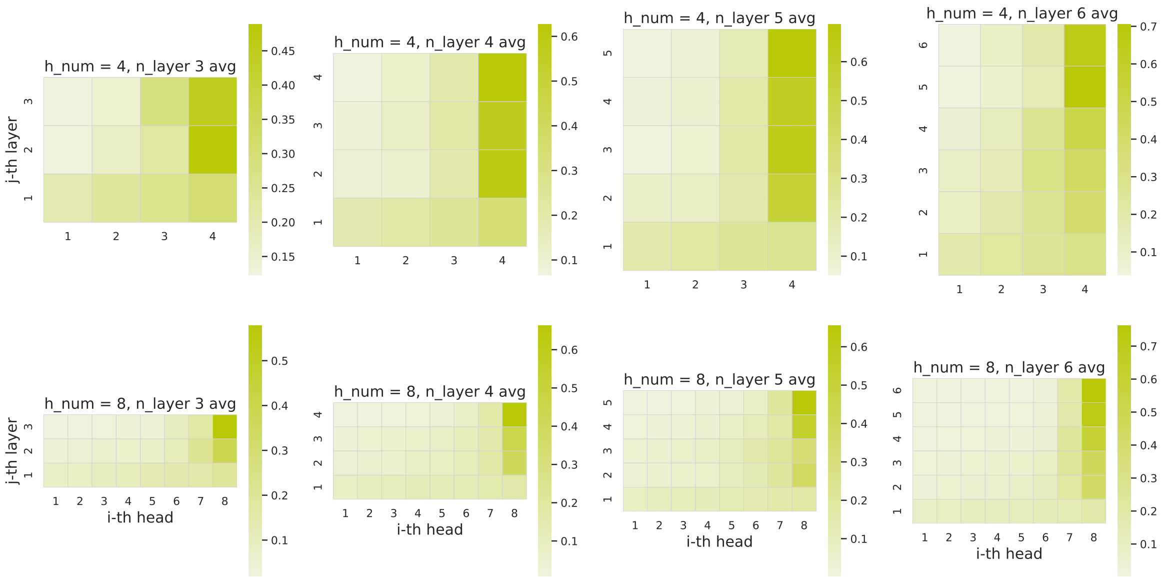


Figure 4: Head Assessment with varying heads, layers

From Fig 4, we can find that in most settings, each head contributes almost equally, while in the subsequent layers, there always exists a head that has a much larger weight than the others. This indicates that in trained transformers for in-context learning, in the first attention layer, all heads appear to be significant, while in the subsequent layers, only one head appears to be significant.

Pruning and Probing Here, we also set $n = 10$ and $q = 1$, with an evaluation data size of 8192. To further support our finding from the Head Assessment, we first prune the model based on our computed head weight $\mathcal{W}^{h,l}_{\mathsf{avg}}$, where we keep all heads in the first layer, whereas we only keep the head with the highest score weight and mask the others. We then train the pruned model with the same method as before for 60000 steps. In Fig 5, 6, 7, 8, we provide the Pruning and Probing results for different numbers of heads $h \in \{4, 8\}$ and noise levels $\sigma \in \{0, 0.1, 0.2, 0.4, 0.8\}$. It can be found that in almost all cases, the pruned model exhibits almost the same performance in each layer, while being largely different from the single-layer transformer. This further supports the results in the Heads Assessment and indicates that the working mechanisms of the multi-head transformer may be different for the first and subsequent layers.

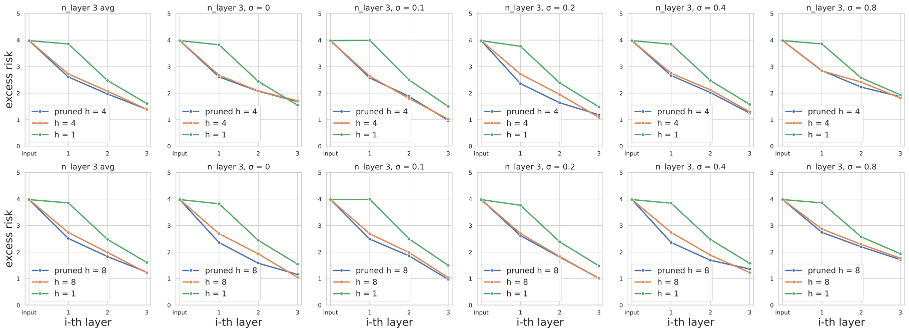


Figure 5: Pruning and Probing, 3 layers

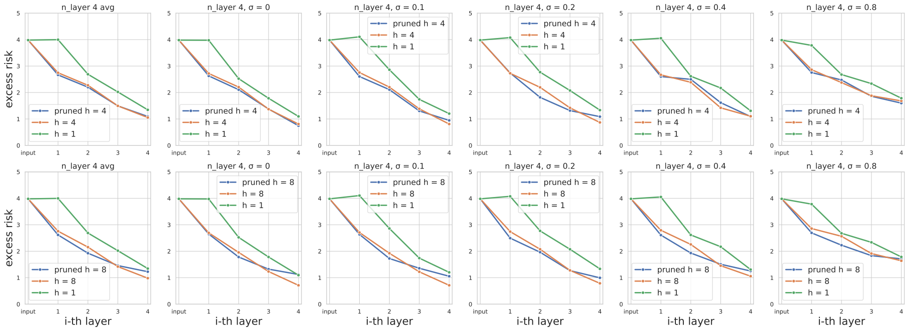


Figure 6: Pruning and Probing, 4 layers

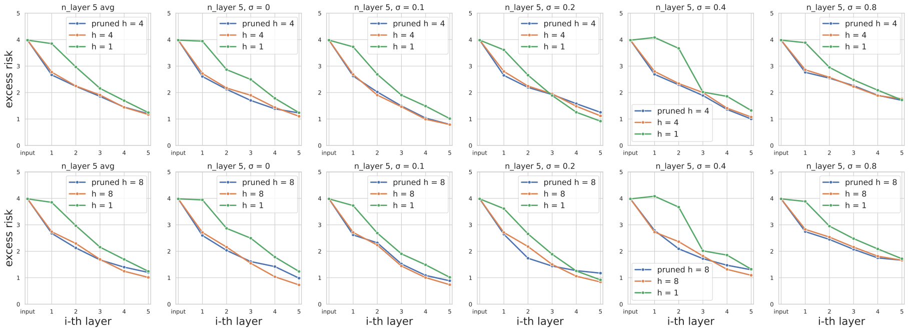


Figure 7: Pruning and Probing, 5 layers

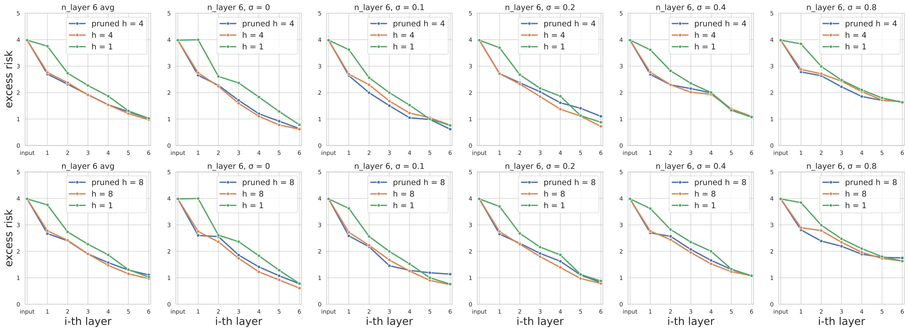


Figure 8: Pruning and Probing, 6 layers

P-probing: To verify the existence of a preprocessing procedure in the trained transformer, we develop a "preprocessing probing" (P-probing) technique on the trained transformers, as illustrated in Figure 2a. For a trained transformer, we first set the input sequence as in Eq.(2.2), where the first n examples $\{\mathbf{x}\}_{i=1}^{n}$ have the corresponding labels $\{y\}_{i=1}^{n}$, and the following q query entries only have $\{\mathbf{x}_i\}_{i=n+1}^{n+q}$ in the sequence. Then, we extract the last q vectors in the output hidden state $\mathbf{H}^1$ from the first layer of the transformer and treat these data as processed query entries. Next, we conduct gradient descent on the first $q-1$ query entries with their corresponding y, computing the excess risk on the last query. Additional experimental details can be found in Appendix A. We adapt this technique based on the intuition that, according to our theoretical analysis, we can extract the preprocessed entry $\{\widetilde{\mathbf{x}}_i\}_{i=n+1}^{n+q}$ from $\mathbf{H}^1$, besides, the excess risk computed by the preprocessed data has a better upper bound guarantee compared to raw data without preprocessing under the same number of gradient descent steps, so if the trained transformer utilize multihead attention for preprocess, compared with single head attention, the queries entries extract from $\mathbf{H}^1$ by multihead attention can have better gradient descent performance compared with single head attention. Here, we also set $n = 117$ and $q = 11$, with an evaluation data size of 1024. We choose $n \gg q$ such that the model can handle more queries ($q = 11$) than those in the training ($q = 4$) process.

Verifying the benefit of preprocessing: To further support the effectiveness of our algorithm, we directly apply Alg. 1 on the input data $\{\mathbf{x}_i, y_i\}_{i=1}^{n+1}$, and then implement gradient descent on the example entries $\{\mathbf{x}_i, y_i\}_{i=1}^{n}$ and compute the excess risk with the last query $\{\widehat{\mathbf{x}}_{n+1}, y_{n+1}\}$, we refer this procedure as `pre-gd`. We compare `pre-gd` with the excess risk obtained by directly applying gradient descent without preprocessing (referred to as `gd`). For all experiments (both P-probing and this), we set $\mathbf{w}_{\mathsf{gd}}^{0} = \mathbf{0}$ and tune the learning rate η for each model by choosing from $[1, 10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}, 10^{-6}]$ with the lowest average excess risk.

B. Theoretical Analysis of the Preprocess-then-optimize Algorthm

B.1. Notations

For two functions $f(x) \geq 0$ and $g(x) \geq 0$ defined on the positive real numbers ($x > 0$), we write $f(x) \lesssim g(x)$ if there exists two constants $c, x_0 > 0$ such that $\forall x \geq x_0$, $f(x) \leq c \cdot g(x)$; we write $f(x) \gtrsim g(x)$ if $g(x) \lesssim f(x)$; we write $f(x) \backsimeq g(x)$ if $f(x) \lesssim g(x)$ and $g(x) \lesssim f(x)$. If $f(x) \lesssim g(x)$, we can write $f(x)$ as $O(g(x))$. We can also write write $f(x)$ as $\widetilde{O}(g(x))$ if there exists a constant $k > 0$ such that $f(x) \lesssim g(x) \log^k(x)$.

B.2. Theoretical results

We first provide the upper bound of the excess risk for $\mathcal{E}(\widetilde{\mathbf{w}}_{\mathsf{gd}}^{t})$ and $\mathcal{E}(\mathbf{w}_{\mathsf{gd}}^{t})$ respectively.

Theorem B.1. *Denote $\mathcal{S} := \{i : w_i^\star \neq 0\}$ and $\mathbf{R} = \mathsf{diag}\{r_1, \ldots, r_d\}$, where $r_j = \sum_{i=1}^{d} w_i^\star \Sigma_{ij}$. Suppose that there exist a $\beta > 0$ such that $\min_{i \in \mathcal{S}} |r_i| \geq \beta$, $\|\mathbf{R}\|_2, \|\boldsymbol{\Sigma}\|_2, \|\mathbf{w}^\star\|_2 \simeq O(1)$ and $n \gtrsim 1/\beta^2 \cdot t^2 s \cdot \big(\mathrm{Tr}^{2/3}(\boldsymbol{\Sigma}) + \mathrm{Tr}(\mathbf{R}\boldsymbol{\Sigma}\mathbf{R})\big) \cdot \mathrm{poly}(\log(d/\delta))$. Then set $\eta \lesssim 1/\|\mathbf{R}\boldsymbol{\Sigma}\mathbf{R}\|_2$ and $\eta t \simeq \frac{1}{\beta} \cdot \big(\frac{\sigma^2 \mathrm{Tr}(\mathbf{R}\boldsymbol{\Sigma}\mathbf{R}) \log(d/\delta)}{n} + \frac{\sigma^2 s \mathrm{Tr}(\boldsymbol{\Sigma}) \log^2(d/\delta)}{n^2}\big)^{-1/2}$, it holds that*

$$\mathcal{E}\big(\widetilde{\mathbf{w}}_{\mathsf{gd}}^{t}\big) \lesssim \frac{\log t}{\beta} \sqrt{\frac{\sigma^2 \mathrm{Tr}(\mathbf{R}\boldsymbol{\Sigma}\mathbf{R}) \log(d/\delta)}{n} + \frac{\sigma^2 s \mathrm{Tr}(\boldsymbol{\Sigma}) \log^2(d/\delta)}{n^2}},$$

with probability at least $1 - \delta$.

Theorem E.1 provides an upper bound on the excess risk achieved by the preprocess-then-optimize algorithm, where we tuned learning rate η to balance the bias and variance error. Then, it can be seen that the risk bound is valid if $\mathrm{Tr}(\mathbf{R}\boldsymbol{\Sigma}\mathbf{R})/n \to 0$ and $\mathrm{Tr}(\boldsymbol{\Sigma})s/n^2 \to 0$ when $n \to \infty$. This can be readily satisfied if we have $\|\mathbf{w}^*\|_2$ and $\mathrm{Tr}(\boldsymbol{\Sigma})$ be bounded by some reasonable quantities that are independent of the sample size n, which are the common assumptions made in many prior works (Zou et al., 2022; 2021; Bartlett et al., 2020). Besides, it can be also seen that the excess risk bound explicitly depends on the sparsity parameter s and lower sparsity implies better performance. This implies the ability of the proposed preprocess-then-optimize for discovering and leveraging the nice sparse structure of the ground truth.

As a comparison, the following theorem states the excess risk bound for the standard gradient descents on the raw features. To make a fair comparison, we consider using the same number of steps but allow the step size to be tuned separately.

Theorem B.2. *Suppose that $\|\mathbf{\Sigma}\|, \|\mathbf{w}^\star\|_2 \simeq O(1)$ and $n \gtrsim t^2(\mathrm{Tr}(\mathbf{\Sigma}) + \log(1/\delta))$. When $\eta \lesssim 1/\|\mathbf{\Sigma}\|_2$ and $\eta t \simeq \left(\frac{\sigma^2 \mathrm{Tr}(\mathbf{\Sigma}) \log(d/\delta)}{n}\right)^{-1/2}$, it holds that*

$$\mathcal{E}\big(\mathbf{w}_{\mathsf{gd}}^t\big) \lesssim \log t \cdot \sqrt{\frac{\sigma^2 \mathrm{Tr}(\mathbf{\Sigma}) \log(d/\delta)}{n}},$$

with probability at least $1-\delta$.

We are now able to make a rough comparison between the excess risk bounds in Theorems E.1 and E.2. Then, it is clear that $\mathcal{E}(\widetilde{\mathbf{w}}_{\mathsf{gd}}^t) \lesssim \mathcal{E}(\mathbf{w}_{\mathsf{gd}}^t)$ requires $\mathrm{Tr}(\mathbf{R\Sigma R})/\beta^2 \lesssim \mathrm{Tr}(\mathbf{\Sigma})$ and $s/(n^2\beta^2) \leq 1/n$. Specifically, we can consider the case that $\mathbf{\Sigma}$ to be a diagonal matrix, assume $w_i^\star \sim \mathsf{U}\{-1/\sqrt{s}, 1/\sqrt{s}\}$ has a restricted uniform prior for $i \in \mathcal{S}$ and $\min_{i\in\mathcal{S}} \mathbf{\Sigma}_{ii} \geq 1/\kappa$ for some constant $\kappa > 1$, we can get $\beta \geq \sqrt{1/(s\kappa^2)}$, thus $\mathrm{Tr}(\mathbf{R\Sigma R})/\beta^2 \leq \kappa^2 \sum_{i: w_i^\star \neq 0} \mathbf{\Sigma}_{ii}$ and $s/(n^2\beta^2) \leq \kappa^2 s^2/n^2$. Note that $|\mathcal{S}| = s \ll d$, then if the covariance matrix $\mathbf{\Sigma}$ has a flat eigenspectrum such that $\sum_{i\in\mathcal{S}} \mathbf{\Sigma}_{ii} \ll \sum_{i\in[d]} \mathbf{\Sigma}_{ii} = \mathrm{Tr}(\mathbf{\Sigma})$, we have $\mathrm{Tr}(\mathbf{R\Sigma R})/\beta^2 \leq \mathrm{Tr}(\mathbf{\Sigma})$ and $s/(n^2\beta^2) \leq \kappa^2 s^2/n$ if $s = o\big(\min\{d, \sqrt{n}\}\big)$. This suggests that the preprocess-then-optimization algorithm can outperform the standard gradient descent for solving a sparse linear regression problem with $s = o\big(\min\{d, \sqrt{n}\}\big)$.

To make a more rigorous comparison, we next consider the example where $x_i \overset{\text{i.i.d.}}{\sim} \mathsf{N}(\mathbf{0}, \mathbf{I})$, based on which we can get the upper bound for our algorithm and the lower bound for OLS, ridge regression, and finite-step GD.

Theorem B.3 (Theorem 5.1, restated)**.** *Suppose $\mathcal{S}$ with $|\mathcal{S}| = s$ is selected such that each element is chosen with equal probability from the set $\{1, 2, \ldots, d\}$ and $w_i^\star \sim \mathsf{U}\{-1/\sqrt{s}, 1/\sqrt{s}\}$ has a restricted uniform prior for $i \in \mathcal{S}$, $\|\mathbf{w}^\star\|_2 \simeq \Theta(1)$ and $n \gtrsim t^2 s^3 d^{2/3}$. Then there exists a choice of η and t such that*

$$\mathcal{E}\big(\widetilde{\mathbf{w}}_{\mathsf{gd}}^t\big) \lesssim \sigma^2 \log^2\big(ns/\sigma^2\big) \log^2(d/\delta) \cdot \left(\frac{s}{n} + \frac{ds^2}{n^2}\right),$$

with probability at least $1-\delta$. Besides, let $\widehat{\mathbf{w}}_\lambda$ be the ridge regression estimator with regularized parameter λ, and $\mathbf{w}_{\mathrm{ols}}$ be the OLS estimator, it holds that

$$\mathbb{E}_{\mathbf{w}^\star}[\mathcal{E}(\mathbf{w})] \gtrsim \begin{cases} \frac{\sigma^2 d}{n} & n \gtrsim d + \log(1/\delta) \\ 1 - \frac{n}{d} + \frac{\sigma^2 n}{d} & d \gtrsim n + \log(1/\delta), \end{cases}$$

with probability at least $1-\delta$, where $\mathbf{w} \in \{\widehat{\mathbf{w}}_\lambda, \mathbf{w}_{\mathrm{ols}}, \mathbf{w}_{\mathsf{gd}}^t\}$.

C. Additional Details for Section 4

C.1. Details and Explanations of Preprocessing-then-Optimizing Algorithm

We note that (Guo et al., 2023) adapts a similar two-phase idea to explain how transformer learning specific functions in context, in their constructed transformers, the first few layers utilize MLPs to compute an appropriate representation for each entry, while the subsequent layers utilize the attention module to implement gradient descent over the context. We highlight that our algorithm mainly focus on utilizing multihead attention, and it aligns well with the our experimental observation and intuition. The details of our algorithm are as follows:

Preprocessing on In-context Examples We summarize this procedure in Alg. 1, we highlight that the preprocessing procedure aligns well with the structure of a multi-head attention layer with linear attention, which motivates our theoretical construction of the desired transformer. In particular, each head of the attention layer can be conceptualized as executing specific operations on a distinct subset of data entries. Then, the linear query-key calculation, represented as $(\mathbf{W}_{K_i}\mathbf{H})^\top \mathbf{W}_{Q_i}\mathbf{H}$, where $\mathbf{H} = \mathbf{E}$ denotes the input sequence embedding matrix, effectively estimates correlations between the i-th subset of data entries and the corresponding label y_i. Here, $\mathbf{W}_{K_i}$ and $\mathbf{W}_{Q_i}$ selectively extract entries from the i-th subset of features and the label, respectively, akin to an ”entries selection” process. Furthermore, when combined with the value calculation $\mathbf{W}_{V_i}\mathbf{H}$, each head of the attention layer conducts correlation calculations for the i-th subset of features and subsequently employs them to reweight the original features within the same subset. Consequently, by stacking the outputs of multiple heads, all data features can be reweighted accordingly, which matches the design of the proposed proprocessing procedure in Alg. 1. We formally prove this in the following theorem.

Proposition C.1 (Single-layer multi-head transformer implements Alg. 1)**.** *There exists a single-layer transformer function* TF_1*, with* d *heads and* $d_{\mathsf{hid}} = 3d$ *hidden dimension, together with an input embedding layer with weight* $\mathbf{W}_E \in \mathbb{R}^{d_{\mathsf{hid}} \times d}$*, that can implement Alg. 1. Let* $\mathbf{E}$ *be the input sequence defined in Eq.(2.2) and* $\widetilde{\mathbf{x}}_i = \widehat{\mathbf{R}}\mathbf{x}$ *be the preprocessed features defined in Alg. 1, it holds that*

$$\mathbf{H}^{(1)} := \mathsf{TF}_1 \circ \mathbf{W}_E(\mathbf{E}) = \begin{pmatrix} \widetilde{\mathbf{x}}_1 & \widetilde{\mathbf{x}}_2 & \cdots & \widetilde{\mathbf{x}}_n & \widetilde{\mathbf{x}}_{n+1} & \cdots & \widetilde{\mathbf{x}}_{n+q} \\ y_1 & y_2 & \cdots & y_n & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \end{pmatrix}, \tag{C.1}$$

where $\cdots$ *in third row implies arbitrary values.*

Optimizing Over Preprocessed In-Context Examples To maintain clarity in our construction and explanation, in each layer, we use a linear projection $\mathbf{W}_1^{(i)}$ to rearrange the dimensions of the sequence processed by the multi-head attention, resulting in the hidden state $\mathbf{H}^{(i)}$ of each layer. We refer to the first d rows of the input data as $\mathbf{x}$, and the $(d+1)$-th row as the corresponding y. For example, in Eq.(C.1), we take the first d rows, together with the $(d+1)$-th row, as the input data entry $\{\widetilde{\mathbf{x}}_i, y_i\}_{i=1}^{n+1}$. Then, the following proposition shows that the subsequent layers of transformer can implement multi-step gradient descent on the preprocessed in-context examples $\{(\widetilde{\mathbf{x}}_i, y_i)\}_{i=1,\ldots,n}$.

Proposition C.2 (Subsequent single-head transformer implements multi-step GD)**.** *There exists a transformer with* k *layers,* 1 *head,* $d_{\mathsf{hid}} = 3d$*, let* $\widehat{y}_{n+1}^{\ell}$ *be the prediction representation of the* ℓ*-th layer, then it holds that* $\widehat{y}_{(n+1)}^{\ell} = \langle \mathbf{w}_{\mathsf{gd}}^{\ell}, \widetilde{\mathbf{x}}_{n+1} \rangle$*, where* $\widetilde{\mathbf{x}}_{n+1} = \widehat{\mathbf{R}}\mathbf{x}_{n+1}$ *denotes the preprocessed data feature,* $\mathbf{w}_{\mathsf{gd}}^{\ell}$ *is defined as* $\mathbf{w}_{\mathsf{gd}}^{0} = 0$ *and as follows for* $\ell = 0, \ldots, k-1$*:*

$$\mathbf{w}_{\mathsf{gd}}^{\ell+1} = \mathbf{w}_{\mathsf{gd}}^{\ell} - \eta \nabla \widetilde{L}(\mathbf{w}_{\mathsf{gd}}^{\ell}), \quad \textit{where} \quad \widetilde{L}(\mathbf{w}) = \frac{1}{2n} \sum_{i=1}^{n} (y_i - \langle \mathbf{w}, \widetilde{\mathbf{x}}_i \rangle)^2. \tag{C.2}$$

C.2. Proof for Proposition C.1

Proposition C.3 (Restate of Proposition C.1)**.** *There exists a transformer with* 1 *layers,* $h = d$ *heads,* $d_{\mathsf{hid}} = 3d$ *and the input projection* $\mathbf{W}_E \in \mathbb{R}^{(d+1) \times d_{\mathsf{hid}}}$ *such that with the input sequence* $\mathbf{E}$ *set as Equation 2.2 the first attention layer can implement Algorithm 1 so that each of the enhanced data* $\{\widehat{r}_i \mathbf{x}_{i,j}\}_{i \in [d]}$ *can be found in the output representation* $\mathbf{H}^{(1)}$*:*

$$\mathbf{H}^{(1)} = \mathsf{TF}_1 \circ \mathbf{W}_E(\mathbf{E}) = \begin{pmatrix} \widetilde{\mathbf{x}}_1 & \widetilde{\mathbf{x}}_2 & \cdots & \widetilde{\mathbf{x}}_n & \widetilde{\mathbf{x}}_{n+1} & \cdots & \widetilde{\mathbf{x}}_{n+q} \\ y_1 & y_2 & \cdots & y_n & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots\ddots & \vdots & \end{pmatrix}.$$

Proof. Here we first explain the key steps of our constructed transformer: the model first rearrange the input entries with a input projection to divide the input data into d subspace $\mathbf{W}_E$, each subspace includes an entry of $\mathbf{x}$ and the corresponding y (step C.4), then use h parameters $\{\mathbf{W}_{V_i}, \mathbf{W}_{K_i}, \mathbf{W}_{Q_i}\}_{i=1}^{h}$ to calculate h queries, keys and values (step C.5), and compute the attention output for each head and concatenate them together (step C.6), finally use a projection matrix $\mathbf{W}_1$ rearrange the result, resulting the target output (step C.7):

$$\mathbf{E} = \begin{pmatrix} \mathbf{x}_1 & \mathbf{x}_2 & \cdots & \mathbf{x}_n & \mathbf{x}_{n+1} & \cdots & \mathbf{x}_{n+q} \\ y_1 & y_2 & \cdots & y_n & 0 & \cdots & 0 \end{pmatrix} \tag{C.3}$$

$$\xrightarrow[\mathbf{W}_E \in \mathbb{R}^{(d+1)\times d_{\mathsf{hid}}}]{\text{input projection}} \quad \mathbf{H} = \begin{pmatrix} \mathbf{x}_{1,1} & \mathbf{x}_{2,1} & \cdots & \mathbf{x}_{n,1} & \mathbf{x}_{(n+1),1} & \cdots & \mathbf{x}_{(n+q),1} \\ y_1 & y_2 & \cdots & y_n & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \end{pmatrix} \tag{C.4}$$

$$\xrightarrow[\mathbf{W}_{V_i}, \mathbf{W}_{K_i}, \mathbf{W}_{Q_i} \in \mathbb{R}^{3\times d_{\mathsf{hid}}}]{\text{compute } \mathbf{Q}_i, \mathbf{K}_i, \mathbf{V}_i} \quad \mathbf{K}_i = \frac{1}{n} \begin{pmatrix} 0 & \cdots & 0 & 0 & \cdots \\ 0 & \cdots & 0 & 0 & \cdots \\ y_1 & \cdots & y_n & 0 & \cdots \end{pmatrix}; \mathbf{Q}_i, \mathbf{V}_i = \begin{pmatrix} 0 & \cdots & 0 & 0 & \cdots \\ 0 & \cdots & 0 & 0 & \cdots \\ \mathbf{x}_{1,i} & \cdots & \mathbf{x}_{n,i} & \mathbf{x}_{(n+1),i} & \cdots \end{pmatrix} \tag{C.5}$$

$$\xrightarrow[\mathbf{H} + \mathsf{Concat}\{\mathbf{V}_i \mathbf{M} \mathbf{K}_i^\top \mathbf{Q}_i\}]{\mathsf{Attn}(\mathbf{W}_E(\mathbf{E}))} \begin{pmatrix} \mathbf{x}_{1,1} & \mathbf{x}_{2,1} & \cdots & \mathbf{x}_{n,1} & \mathbf{x}_{(n+1),1} & \cdots & \mathbf{x}_{(n+q),1} \\ y_1 & y_2 & \cdots & y_n & 0 & 0 & 0 \\ \widetilde{\mathbf{x}}_{1,1} & \widetilde{\mathbf{x}}_{2,1} & \cdots & \widetilde{\mathbf{x}}_{n,1} & \widetilde{\mathbf{x}}_{(n+1),1} & \cdots & \widetilde{\mathbf{x}}_{(n+q),1} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \end{pmatrix} \tag{C.6}$$

$$\xrightarrow[\mathbf{W}_1 \in \mathbb{R}^{d_{\mathsf{hid}} \times d_{\mathsf{hid}}}]{\mathbf{H}^{(1)} = \mathsf{TF}_1 \circ \mathbf{W}_E(\mathbf{E})} \begin{pmatrix} \widetilde{\mathbf{x}}_1 & \widetilde{\mathbf{x}}_2 & \cdots & \widetilde{\mathbf{x}}_n & \widetilde{\mathbf{x}}_{n+1} & \cdots & \widetilde{\mathbf{x}}_{n+q} \\ y_1 & y_2 & \cdots & y_n & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \end{pmatrix} \tag{C.7}$$

. The detailed parameters and calculation process for each step are as follows:

- we set $\mathbf{W}_E \in \mathbb{R}^{(d+1)\times d_{\mathsf{hid}}}$ to rearrange the entries:

$$\mathbf{W}_E = \begin{pmatrix} \mathbb{1}[1] & \mathbb{1}[d+1] & \mathbf{0} & \mathbb{1}[2] & \mathbb{1}[d+1] & \mathbf{0} & \cdots & \mathbb{1}[d] & \mathbb{1}[d+1] & \mathbf{0} \end{pmatrix}^\top,$$

where $\mathbb{1}[k]$ is an $1 \times d_{\mathsf{hid}}$ vector with 1 at i-th entry and 0 elsewhere, such that

$$\mathbf{H} = \mathbf{W}_E \mathbf{E} = \begin{pmatrix} \mathbf{x}_{1,1} & \mathbf{x}_{2,1} & \cdots & \mathbf{x}_{n,1} & \mathbf{x}_{(n+1),1} & \cdots & \mathbf{x}_{(n+q),1} \\ y_1 & y_2 & \cdots & y_n & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & 0 & \cdots & 0 \\ \mathbf{x}_{1,2} & \mathbf{x}_{2,2} & \cdots & \mathbf{x}_{n,2} & \mathbf{x}_{(n+1),2} & \cdots & \mathbf{x}_{(n+q),2} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \end{pmatrix}.$$

- we set $\mathbf{W}_{V_i}, \mathbf{W}_{K_i}, \mathbf{W}_{Q_i} \in \mathbb{R}^{3\times d_{\mathsf{hid}}}$ for values, keys and queries:

$$\mathbf{W}_{K_i} = \frac{1}{n} \begin{pmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbb{1}[3i-1] \end{pmatrix}; \quad \mathbf{W}_{V_i}, \mathbf{W}_{Q_i} = \begin{pmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbb{1}[3i-2] \end{pmatrix},$$

such that the i-th head extract i-th entry of $\mathbf{x}$ and corresponding y

$$\mathbf{K}_i = \frac{1}{n} \begin{pmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbb{1}[3i-1] \end{pmatrix} \begin{pmatrix} \mathbf{x}_{1,1} & \mathbf{x}_{2,1} & \cdots & \mathbf{x}_{n,1} & \mathbf{x}_{(n+1),1} & \cdots \\ y_1 & y_2 & \cdots & y_n & 0 & \cdots \\ 0 & 0 & \cdots & 0 & 0 & \cdots \\ \mathbf{x}_{1,2} & \mathbf{x}_{2,2} & \cdots & \mathbf{x}_{n,2} & \mathbf{x}_{(n+1),2} & \cdots \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots \end{pmatrix} = \frac{1}{n} \begin{pmatrix} 0 & \cdots & 0 & 0 & \cdots \\ 0 & \cdots & 0 & 0 & \cdots \\ y_1 & \cdots & y_n & 0 & \cdots \end{pmatrix},$$

$$\mathbf{Q}_i, \mathbf{V}_i = \begin{pmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbb{1}[3i-2] \end{pmatrix} \begin{pmatrix} \mathbf{x}_{1,1} & \mathbf{x}_{2,1} & \cdots & \mathbf{x}_{n,1} & \mathbf{x}_{(n+1),1} & \cdots \\ y_1 & y_2 & \cdots & y_n & 0 & \cdots \\ 0 & 0 & \cdots & 0 & 0 & \cdots \\ \mathbf{x}_{1,2} & \mathbf{x}_{2,2} & \cdots & \mathbf{x}_{n,2} & \mathbf{x}_{(n+1),2} & \cdots \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots \end{pmatrix} = \begin{pmatrix} 0 & \cdots & 0 & 0 & \cdots \\ 0 & \cdots & 0 & 0 & \cdots \\ \mathbf{x}_{1,i} & \cdots & \mathbf{x}_{n,i} & \mathbf{x}_{(n+1),i} & \cdots \end{pmatrix},$$

$$\mathbf{V}_i\mathbf{M}\mathbf{K}_i^\top\mathbf{Q}_i = \begin{pmatrix} 0 & \cdots & 0 & 0 & \cdots & 0\\ 0 & \cdots & 0 & 0 & \cdots & 0\\ \mathbf{x}_{1,i} & \cdots & \mathbf{x}_{n,i} & \mathbf{x}_{(n+1),i} & \cdots & \mathbf{x}_{(n+q),i}\end{pmatrix}\begin{pmatrix}\mathbf{I}_n & \mathbf{0}\\ \mathbf{0} & \mathbf{0}\end{pmatrix}.$$

$$\begin{pmatrix} 0 & \cdots & 0 & 0 & \cdots & 0\\ 0 & \cdots & 0 & 0 & \cdots & 0\\ y_1 & \cdots & y_n & 0 & \cdots & 0\end{pmatrix}^\top\begin{pmatrix} 0 & \cdots & 0 & 0 & \cdots & 0\\ 0 & \cdots & 0 & 0 & \cdots & 0\\ \mathbf{x}_{1,i} & \cdots & \mathbf{x}_{n,i} & \mathbf{x}_{(n+1),i} & \cdots & \mathbf{x}_{(n+q),i}\end{pmatrix}$$

$$= \begin{pmatrix} 0 & \cdots & 0 & 0 & \cdots & 0\\ 0 & \cdots & 0 & 0 & \cdots & 0\\ \widetilde{\mathbf{x}}_{1,i} & \cdots & \widetilde{\mathbf{x}}_{n,i} & \widetilde{\mathbf{x}}_{(n+1),i} & \cdots & \widetilde{\mathbf{x}}_{(n+q),i}\end{pmatrix}.$$

- Then concatenate the output of each head $\{\mathbf{V}_i\mathbf{M}\mathbf{K}_i^\top\mathbf{Q}_i\}_{i=1}^h$ together with residue:

$$\mathbf{H} + \mathsf{Concat}[\{\mathbf{V}_i\mathbf{M}\mathbf{K}_i^\top\mathbf{Q}_i\}_{i=1}^h] = \begin{pmatrix} \mathbf{x}_{1,1} & \mathbf{x}_{2,1} & \cdots & \mathbf{x}_{n,1} & \mathbf{x}_{(n+1),1} & \cdots & \mathbf{x}_{(n+q),1}\\ y_1 & y_2 & \cdots & y_n & 0 & \cdots & 0\\ \widetilde{\mathbf{x}}_{1,1} & \widetilde{\mathbf{x}}_{2,1} & \cdots & \widetilde{\mathbf{x}}_{n,1} & \widetilde{\mathbf{x}}_{(n+1),1} & \cdots & \widetilde{\mathbf{x}}_{(n+q),1}\\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \end{pmatrix}. \tag{C.8}$$

- Finally, $\mathbf{W}_1$ is applied to rearrange the entries:

$$\mathbf{W}_1 = \begin{pmatrix}\mathbb{1}[3] & \cdots & \mathbb{1}[3d] & \mathbb{1}[2] & \cdots\end{pmatrix}^\top,$$

  where the first $\cdots$ implies the omitted $d-2$ vectors $\{\mathbb{1}[3i] | i = 2, 3, \ldots, (d-1)\}$, the second $\cdots$ implies arbitrary values, then resulting the final output:

$$\mathbf{H}^{(1)} = \mathbf{W}_1\big[\mathbf{H} + \mathsf{Concat}[\{\mathbf{V}_i\mathbf{M}\mathbf{K}_i^\top\mathbf{Q}_i\}_{i=1}^h]\big] = \begin{pmatrix} \widetilde{\mathbf{x}}_1 & \widetilde{\mathbf{x}}_2 & \cdots & \widetilde{\mathbf{x}}_n & \widetilde{\mathbf{x}}_{n+1} & \cdots & \widetilde{\mathbf{x}}_{n+q}\\ y_1 & y_2 & \cdots & y_n & 0 & \cdots & 0\\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \end{pmatrix}.$$

in this way we construct a transformer that can apply Alg. 1 so that each of the enhanced data $\{\widehat{r}_i\mathbf{x}_{i,j}\}_{i\in[d]}$ can be found in the output representation $\mathbf{H}^{(1)}$. □

C.3. Proof for Proposition C.2

Proposition C.4 (Restate of Proposition C.2)**.** *There exists a transformer with k layers, 1 head, $d_{\mathsf{hid}} = 3d$, let $\{(\widetilde{\mathbf{x}}_i, \widehat{y}_{(i)}^\ell)\}_{i=1}^{n+1}$ be the ℓ-th layer input data entry, then it holds that $\widehat{y}_{(n+1)}^\ell = \langle \mathbf{w}_{\mathsf{gd}}^\ell, \widetilde{\mathbf{x}}_{n+1}\rangle$, where $\mathbf{w}_{\mathsf{gd}}$ is defined as $\mathbf{w}_{\mathsf{gd}}^0 = 0$ and as follows for $\ell = 0, ..., k-1$:*

$$\mathbf{w}_{\mathsf{gd}}^{\ell+1} = \mathbf{w}_{\mathsf{gd}}^{\ell} - \eta\nabla\widetilde{L}(\mathbf{w}_{\mathsf{gd}}^\ell), \quad \textit{where} \quad \widetilde{L}(\mathbf{w}) = \frac{1}{2n}\sum_{i=1}^n (y_i - \langle \mathbf{w}, \widetilde{\mathbf{x}}_i\rangle)^2.$$

Proof. Here we directly provide the parameters $\mathbf{W}_V^\ell, \mathbf{W}_K^\ell, \mathbf{W}_Q^\ell \in \mathbb{R}^{d_{\mathsf{hid}}\times d_{\mathsf{hid}}}$ and $\mathbf{W}_1^\ell \in \mathbb{R}^{d_{\mathsf{hid}}\times d_{\mathsf{hid}}}$ for each layer TF_ℓ,

$$\mathbf{W}_V^\ell = -\frac{\eta}{n}\begin{pmatrix}\mathbf{0} & \mathbf{0}\\ \mathbf{0} & 1\end{pmatrix}; \quad \mathbf{W}_K^\ell, \mathbf{W}_Q^\ell = \begin{pmatrix}\mathbf{I}_{d\times d} & \mathbf{0}\\ \mathbf{0} & 0\end{pmatrix}; \quad \mathbf{W}_1^\ell = \mathbf{I}_{d_{\mathsf{hid}}\times d_{\mathsf{hid}}} \tag{C.9}$$

As we set $\mathbf{W}_1^\ell$ as the identity matrix, we can ignore it and then apply Lemma 1 in (Ahn et al.). By replacing $(\mathbf{W}_K^{\ell}{}^\top\mathbf{W}_Q^\ell)$ as $\mathbf{Q}_i$ and $\mathbf{W}_V^\ell$ with $\mathbf{P}_i$, then it holds that $\widehat{y}_{(n+1)}^\ell = \langle \mathbf{w}_{\mathsf{gd}}^\ell, \widetilde{\mathbf{x}}_{n+1}\rangle$, where $\mathbf{w}_{\mathsf{gd}}$ is defined as $\mathbf{w}_{\mathsf{gd}}^0 = 0$ and as follows for $\ell = 0, ..., k-1$:

$$\mathbf{w}_{\mathsf{gd}}^{\ell+1} = \mathbf{w}_{\mathsf{gd}}^{\ell} - \eta\nabla\widetilde{L}(\mathbf{w}_{\mathsf{gd}}^\ell), \quad \textit{where} \quad \widetilde{L}(\mathbf{w}) = \frac{1}{2n}\sum_{i=1}^n (y_i - \langle \mathbf{w}, \widetilde{\mathbf{x}}_i\rangle)^2.$$

□

D. Additional Related Work

In addition to works towards understanding the expressive power of transformers that we introduced before, there is also a body of research on the mechanism interpretation and the training dynamics of transformers:

Mechanism interpretation of trained transformers To understand the mechanisms in trained transformers, researchers have developed various techniques, including interpreting transformers into programming languages (Friedman et al.; Lindner et al., 2023; Weiss et al., 2021; Zhou et al., 2024), probing the behavior of individual layers (Pandit & Hou, 2021; Wu et al., 2020; Bricken et al., 2023; Allen-Zhu & Li, 2023; Zhu & Li, 2023), and incorporating transformers with other large language models to interpret individual neurons (Bills et al., 2023). While these techniques provide high-level insights into transformer mechanism understanding, providing a clear algorithms behind the trained transformers is still very challenging.

Training dynamics of transformers In parallel, a body of work has also investigated how transformers learn these algorithms, i.e., the training dynamics of transformers. (Tarzanagh et al., 2023) shows an equivalence between a single attention layer and a support vector machine. (Zhang et al., 2023; Huang et al., 2023) analyze the training dynamics of a single-head attention layer for in-context linear regression, where (Zhang et al., 2023) demonstrates that it can converge to implement one-step gradient over in-context examples. Additionally, (Tian et al., 2023; Li et al., 2023) study the convergence of transformers on sequences of discrete tokens. These works provide valuable insights towards the theoretical understanding of the training dynamics of transformers, which offer potential future extension aspects for our work.

E. Theoretical Analysis of the Preprocess-then-optimize Algorthm

E.1. Notations

For two functions $f(x) \geq 0$ and $g(x) \geq 0$ defined on the positive real numbers ($x > 0$), we write $f(x) \lesssim g(x)$ if there exists two constants $c, x_0 > 0$ such that $\forall x \geq x_0$, $f(x) \leq c \cdot g(x)$; we write $f(x) \gtrsim g(x)$ if $g(x) \lesssim f(x)$; we write $f(x) \eqsim g(x)$ if $f(x) \lesssim g(x)$ and $g(x) \lesssim f(x)$. If $f(x) \lesssim g(x)$, we can write $f(x)$ as $O(g(x))$. We can also write write $f(x)$ as $\widetilde{O}(g(x))$ if there exists a constant $k > 0$ such that $f(x) \lesssim g(x) \log^k(x)$.

E.2. Theoretical results

We first provide the upper bound of the excess risk for $\mathcal{E}(\widetilde{\mathbf{w}}_{\mathsf{gd}}^t)$ and $\mathcal{E}(\mathbf{w}_{\mathsf{gd}}^t)$ respectively.

Theorem E.1. *Denote $\mathcal{S} := \{i : w_i^\star \neq 0\}$ and $\mathbf{R} = \mathsf{diag}\{r_1, \ldots, r_d\}$, where $r_j = \sum_{i=1}^d w_i^\star \Sigma_{ij}$. Suppose that there exist a $\beta > 0$ such that $\min_{i\in\mathcal{S}} |r_i| \geq \beta$, $\|\mathbf{R}\|_2, \|\mathbf{\Sigma}\|_2, \|\mathbf{w}^\star\|_2 \simeq O(1)$ and $n \gtrsim 1/\beta^2 \cdot t^2 s \cdot \big(\mathrm{Tr}^{2/3}(\mathbf{\Sigma}) + \mathrm{Tr}(\mathbf{R\Sigma R})\big) \cdot \mathrm{poly}(\log(d/\delta))$. Then set $\eta \lesssim 1/\|\mathbf{R\Sigma R}\|_2$ and $\eta t \simeq \frac{1}{\beta} \cdot \big(\frac{\sigma^2 \mathrm{Tr}(\mathbf{R\Sigma R}) \log(d/\delta)}{n} + \frac{\sigma^2 s \mathrm{Tr}(\mathbf{\Sigma}) \log^2(d/\delta)}{n^2}\big)^{-1/2}$, it holds that*

$$\mathcal{E}\big(\widetilde{\mathbf{w}}_{\mathsf{gd}}^t\big) \lesssim \frac{\log t}{\beta} \sqrt{\frac{\sigma^2 \mathrm{Tr}(\mathbf{R\Sigma R}) \log(d/\delta)}{n} + \frac{\sigma^2 s \mathrm{Tr}(\mathbf{\Sigma}) \log^2(d/\delta)}{n^2}},$$

with probability at least $1 - \delta$.

Theorem E.1 provides an upper bound on the excess risk achieved by the preprocess-then-optimize algorithm, where we tuned learning rate η to balance the bias and variance error. Then, it can be seen that the risk bound is valid if $\mathrm{Tr}(\mathbf{R\Sigma R})/n \to 0$ and $\mathrm{Tr}(\mathbf{\Sigma})s/n^2 \to 0$ when $n \to \infty$. This can be readily satisfied if we have $\|\mathbf{w}^*\|_2$ and $\mathrm{Tr}(\mathbf{\Sigma})$ be bounded by some reasonable quantities that are independent of the sample size n, which are the common assumptions made in many prior works (Zou et al., 2022; 2021; Bartlett et al., 2020). Besides, it can be also seen that the excess risk bound explicitly depends on the sparsity parameter s and lower sparsity implies better performance. This implies the ability of the proposed preprocess-then-optimize for discovering and leveraging the nice sparse structure of the ground truth.

As a comparison, the following theorem states the excess risk bound for the standard gradient descents on the raw features. To make a fair comparison, we consider using the same number of steps but allow the step size to be tuned separately.

Theorem E.2. *Suppose that $\|\mathbf{\Sigma}\|, \|\mathbf{w}^\star\|_2 \simeq O(1)$ and $n \gtrsim t^2(\mathrm{Tr}(\mathbf{\Sigma}) + \log(1/\delta))$. When $\eta \lesssim 1/\|\mathbf{\Sigma}\|_2$ and $\eta t \simeq$*

$\left(\frac{\sigma^2 \mathrm{Tr}(\mathbf{\Sigma}) \log(d/\delta)}{n}\right)^{-1/2}$*, it holds that*

$$\mathcal{E}\big(\mathbf{w}_{\mathsf{gd}}^t\big) \lesssim \log t \cdot \sqrt{\frac{\sigma^2 \mathrm{Tr}(\mathbf{\Sigma}) \log(d/\delta)}{n}},$$

with probability at least $1-\delta$*.*

We are now able to make a rough comparison between the excess risk bounds in Theorems E.1 and E.2. Then, it is clear that $\mathcal{E}(\widetilde{\mathbf{w}}_{\mathsf{gd}}^t) \lesssim \mathcal{E}(\mathbf{w}_{\mathsf{gd}}^t)$ requires $\mathrm{Tr}(\mathbf{R\Sigma R})/\beta^2 \lesssim \mathrm{Tr}(\mathbf{\Sigma})$ and $s/(n^2\beta^2) \le 1/n$. Specifically, we can consider the case that $\mathbf{\Sigma}$ to be a diagonal matrix, assume $w_i^\star \sim \mathsf{U}\{-1/\sqrt{s}, 1/\sqrt{s}\}$ has a restricted uniform prior for $i \in \mathcal{S}$ and $\min_{i\in\mathcal{S}} \mathbf{\Sigma}_{ii} \ge 1/\kappa$ for some constant $\kappa > 1$, we can get $\beta \ge \sqrt{1/(s\kappa^2)}$, thus $\mathrm{Tr}(\mathbf{R\Sigma R})/\beta^2 \le \kappa^2 \sum_{i:w_i^\star \neq 0} \mathbf{\Sigma}_{ii}$ and $s/(n^2\beta^2) \le \kappa^2 s^2/n^2$. Note that $|\mathcal{S}| = s \ll d$, then if the covariance matrix $\mathbf{\Sigma}$ has a flat eigenspectrum such that $\sum_{i\in\mathcal{S}} \mathbf{\Sigma}_{ii} \ll \sum_{i\in[d]} \mathbf{\Sigma}_{ii} = \mathrm{Tr}(\mathbf{\Sigma})$, we have $\mathrm{Tr}(\mathbf{R\Sigma R})/\beta^2 \le \mathrm{Tr}(\mathbf{\Sigma})$ and $s/(n^2\beta^2) \le \kappa^2 s^2/n$ if $s = o\big(\min\{d, \sqrt{n}\}\big)$. This suggests that the preprocess-then-optimization algorithm can outperform the standard gradient descent for solving a sparse linear regression problem with $s = o\big(\min\{d, \sqrt{n}\}\big)$.

To make a more rigorous comparison, we next consider the example where $x_i \overset{\text{i.i.d.}}{\sim} \mathsf{N}(\mathbf{0}, \mathbf{I})$, based on which we can get the upper bound for our algorithm and the lower bound for OLS, ridge regression, and finite-step GD.

Theorem E.3 (Theorem 5.1, restated)**.** *Suppose* $\mathcal{S}$ *with* $|\mathcal{S}| = s$ *is selected such that each element is chosen with equal probability from the set* $\{1, 2, \ldots, d\}$ *and* $w_i^\star \sim \mathsf{U}\{-1/\sqrt{s}, 1/\sqrt{s}\}$ *has a restricted uniform prior for* $i \in \mathcal{S}$, $\|\mathbf{w}^\star\|_2 \simeq \Theta(1)$ *and* $n \gtrsim t^2 s^3 d^{2/3}$*. Then there exists a choice of* η *and* t *such that*

$$\mathcal{E}\big(\widetilde{\mathbf{w}}_{\mathsf{gd}}^t\big) \lesssim \sigma^2 \log^2\left(ns/\sigma^2\right) \log^2(d/\delta) \cdot \left(\frac{s}{n} + \frac{ds^2}{n^2}\right),$$

with probability at least $1-\delta$*. Besides, let* $\widehat{\mathbf{w}}_\lambda$ *be the ridge regression estimator with regularized parameter* λ*, and* $\mathbf{w}_{\mathrm{ols}}$ *be the OLS estimator, it holds that*

$$\mathbb{E}_{\mathbf{w}^\star}[\mathcal{E}(\mathbf{w})] \gtrsim \begin{cases} \frac{\sigma^2 d}{n} & n \gtrsim d + \log(1/\delta) \\ 1 - \frac{n}{d} + \frac{\sigma^2 n}{d} & d \gtrsim n + \log(1/\delta), \end{cases}$$

with probability at least $1-\delta$*, where* $\mathbf{w} \in \{\widehat{\mathbf{w}}_\lambda, \mathbf{w}_{\mathrm{ols}}, \mathbf{w}_{\mathsf{gd}}^t\}$*.*

F. Proof of Theorem E.1

To simplify the notations, we use $\widehat{\mathbf{w}}_t$ to denote $\widetilde{\mathbf{w}}_{\mathsf{gd}}^t$. We first prove that with a high probability, there exists a $\overline{\mathbf{R}} \in \mathbb{R}^{d\times d}$ such that $\overline{\mathbf{R}}\widehat{\mathbf{R}} = \widehat{\mathbf{R}}\overline{\mathbf{R}} = \mathbf{I}_s$, where $\mathbf{I}_s = \mathsf{diag}\{a_1, \ldots, a_d\}$ with $a_j = 1_{\{j\in\mathcal{S}\}}$.

Lemma F.1. *Denote* $\mathbf{R} = \mathsf{diag}\{r_1, \ldots, r_d\}$*, where* $r_j = \sum_{i=1}^d w_i^\star \Sigma_{ij}$*. Suppose* $n \ge \mathcal{O}(\log(d/\delta))$*, then for any* $\delta \in (0,1)$ *with probability at least* $1-\delta$*, we have*

$$\|\widehat{\mathbf{R}} - \mathbf{R}\|_2 \lesssim K \cdot \sqrt{\frac{s \log(d/\delta)}{n}},$$

where $K := C\big(\max_i \Sigma_{ii} + \sigma^2\big)$*, where* C *is an absolute constant.*

Lemma F.2. *Define the event* $\mathcal{E}_R$ *by* $\mathcal{E}_R = \big\{|\widehat{r}|_i \ge \frac{1}{2}|r_i|,\ \forall i \in \mathcal{S}\big\}$*. Suppose that* $n \gtrsim s\log(d/\delta)/\beta^2$*, then* $\mathbb{P}(\mathcal{E}_1) \ge 1-\delta$*.*

We define $\overline{\mathbf{R}}$ by $\overline{\mathbf{R}} = \mathsf{diag}\{\overline{r}_1, \ldots, \overline{r}_d\}$, where $\overline{r}_j$ is given by

$$\overline{r}_j = \begin{cases} 0 & j \notin \mathcal{S}, \\ 1/\widehat{r}_j & j \in \mathcal{S}. \end{cases}$$

It is easy to see $\overline{\mathbf{R}}\widehat{\mathbf{R}} = \widehat{\mathbf{R}}\overline{\mathbf{R}} = \mathbf{I}_s$. On the event $\mathcal{E}_1$, we have that $\big\|\overline{\mathbf{R}}\big\| \lesssim 1/\beta$. Hereafter, we condition on $\mathcal{E}_1$.

F.1. Bias-variance Decomposition

Let $\widetilde{\mathbf{X}} = \mathbf{X}\widehat{\mathbf{R}}$ with $\widetilde{\mathbf{x}}_i = \widehat{\mathbf{R}}\mathbf{x}_i$. For $\widehat{\mathbf{w}}_t$, we have

$$\begin{aligned}
\widehat{\mathbf{w}}_{t+1} - \overline{\mathbf{R}}\mathbf{w}^\star &= \widehat{\mathbf{w}}_t - \overline{\mathbf{R}}\mathbf{w}^\star - \eta \cdot \frac{1}{n}\sum_{i=1}^n \widetilde{\mathbf{x}}_i\big(\widetilde{\mathbf{x}}_i^\top \widehat{\mathbf{w}}_t - y_i\big) \\
&= \widehat{\mathbf{w}}_t - \overline{\mathbf{R}}\mathbf{w}^\star - \eta \cdot \frac{1}{n}\sum_{i=1}^n \widetilde{\mathbf{x}}_i\big(\widetilde{\mathbf{x}}_i^\top \widehat{\mathbf{w}}_t - \widetilde{\mathbf{x}}_i^\top \overline{\mathbf{R}}\mathbf{w}^\star + \epsilon\big) \\
&= \Big(\mathbf{I} - \eta\widehat{\boldsymbol{\Sigma}}\Big)\big(\widehat{\mathbf{w}}_t - \overline{\mathbf{R}}\mathbf{w}^\star\big) + \eta \cdot \frac{1}{n}\widetilde{\mathbf{X}}^\top \epsilon.
\end{aligned}$$

Hence, we have

$$\widehat{\mathbf{w}}_t = \bigg(\mathbf{I} - \Big(\mathbf{I} - \eta\widehat{\boldsymbol{\Sigma}}\Big)^t\bigg)\overline{\mathbf{R}}\mathbf{w}^\star + \frac{1}{n}\sum_{i=1}^t \Big(\mathbf{I} - \eta\widehat{\boldsymbol{\Sigma}}\Big)^{i-1}\widetilde{\mathbf{X}}^\top \epsilon. \tag{F.1}$$

We can decompose the risk $L(\widehat{\mathbf{w}}_t)$ by

$$\begin{aligned}
\mathcal{E}(\widehat{\mathbf{w}}_t) &= \mathbb{E}_{(\mathbf{x},y)\sim \mathsf{P}}\bigg[\Big(\langle \widehat{\mathbf{R}}\mathbf{x}, \widehat{\mathbf{w}}_t\rangle - \langle \widehat{\mathbf{R}}\mathbf{x}, \overline{\mathbf{R}}\mathbf{w}^\star\rangle - \epsilon\Big)^2\bigg] - \sigma^2 && \text{(F.2)} \\
&= \mathbb{E}_{(\mathbf{x},y)\sim \mathsf{P}}\bigg[\Big(\langle \widehat{\mathbf{R}}\mathbf{x}, \widehat{\mathbf{w}}_t\rangle - \langle \widehat{\mathbf{R}}\mathbf{x}, \overline{\mathbf{R}}\mathbf{w}^\star\rangle\Big)^2\bigg] \\
&= \Big\|\boldsymbol{\Sigma}^{1/2}\widehat{\mathbf{R}}\big(\widehat{\mathbf{w}}_t - \overline{\mathbf{R}}\mathbf{w}^\star\big)\Big\|_2^2 \\
&= \left\|\boldsymbol{\Sigma}^{1/2}\widehat{\mathbf{R}}\left(-\Big(\mathbf{I} - \eta\widehat{\boldsymbol{\Sigma}}\Big)^t\overline{\mathbf{R}}\mathbf{w}^\star + \eta\cdot\frac{1}{n}\sum_{i=1}^t\Big(\mathbf{I} - \eta\widehat{\boldsymbol{\Sigma}}\Big)^{i-1}\widetilde{\mathbf{X}}^\top\epsilon\right)\right\|_2^2 \\
&= \underbrace{\Big\|\boldsymbol{\Sigma}^{1/2}\widehat{\mathbf{R}}\Big(\mathbf{I} - \eta\widehat{\boldsymbol{\Sigma}}\Big)^t\overline{\mathbf{R}}\mathbf{w}^\star\Big\|_2^2}_{\text{Bias}} + \underbrace{\eta^2\left\|\boldsymbol{\Sigma}^{1/2}\widehat{\mathbf{R}}\left(\frac{1}{n}\sum_{i=1}^t\Big(\mathbf{I} - \eta\widehat{\boldsymbol{\Sigma}}\Big)^{i-1}\widetilde{\mathbf{X}}^\top\epsilon\right)\right\|_2^2}_{\text{Variance}}. && \text{(F.3)}
\end{aligned}$$

Next, we present some lemmas.

Lemma F.3 (Theorem 9 in Bartlett et al. (2020))**.** *There is an absolute constant c such that for any $\delta \in (0,1)$ with probability at least $1-\delta$,*

$$\|\widehat{\boldsymbol{\Sigma}} - \boldsymbol{\Sigma}\|_2 \le c\|\boldsymbol{\Sigma}\|_2 \cdot \max\left\{\sqrt{\frac{r(\boldsymbol{\Sigma})}{n}}, \frac{r(\boldsymbol{\Sigma})}{n}, \sqrt{\frac{\log(1/\delta)}{n}}, \frac{\log(1/\delta)}{n}\right\},$$

where $r(\boldsymbol{\Sigma}) = \mathrm{Tr}(\boldsymbol{\Sigma})/\lambda_1$.

Lemma F.4. *With probability at least $1-\delta$, we have*

$$\|\widehat{\mathbf{R}}\widehat{\boldsymbol{\Sigma}}\widehat{\mathbf{R}} - \mathbf{R}\boldsymbol{\Sigma}\mathbf{R}\|_2 \lesssim \sqrt{s}\cdot \mathrm{poly}(\log(d/\delta))\cdot\left(\sqrt{\frac{r(\mathbf{R}\boldsymbol{\Sigma}\mathbf{R})}{n}} + \frac{\sqrt{r(\boldsymbol{\Sigma})} + r(\mathbf{R}\boldsymbol{\Sigma}\mathbf{R})}{n} + \frac{r(\boldsymbol{\Sigma})}{n^{3/2}}\right).$$

As a result, when $n \gtrsim st^2\big(r^{2/3}(\boldsymbol{\Sigma}) + r(\mathbf{R}\boldsymbol{\Sigma}\mathbf{R})\big)\cdot\mathrm{poly}(\log(d/\delta))$, with probability at least $1-\delta$, we have

$$\|\widehat{\mathbf{R}}\widehat{\boldsymbol{\Sigma}}\widehat{\mathbf{R}} - \mathbf{R}\boldsymbol{\Sigma}\mathbf{R}\|_2 \le 1/t.$$

We define the event $\mathcal{E}_2$ as follows:

$$\mathcal{E}_2 := \Big\{\|\mathbf{R}\boldsymbol{\Sigma}\mathbf{R}\|_2 \lesssim \widetilde{\alpha}(n,\delta) \le 1/t\Big\},$$

where

$$\widetilde{\alpha}(n,\delta)=\sqrt{s}\cdot\operatorname{poly}(\log(d/\delta))\cdot\left(\sqrt{\frac{r(\mathbf{R}\boldsymbol{\Sigma}\mathbf{R})}{n}}+\frac{\sqrt{r(\boldsymbol{\Sigma})}+r(\mathbf{R}\boldsymbol{\Sigma}\mathbf{R})}{n}+\frac{r(\boldsymbol{\Sigma})}{n^{3/2}}\right).$$

By Lemma F.4, $\mathbb{P}(\mathcal{E}_2)\geq 1-\delta$. Hereafter, we condition on $\mathcal{E}_1\cap\mathcal{E}_2$.

F.2. Bounding the Bias

On $\mathcal{E}_1\cap\mathcal{E}_2$, we have

$$\begin{aligned}\text{Bias}&=\left\|\boldsymbol{\Sigma}^{1/2}\widehat{\mathbf{R}}\left(\mathbf{I}-\eta\widehat{\boldsymbol{\Sigma}}\right)^t\overline{\mathbf{R}}\mathbf{w}^\star\right\|_2^2\\&=\mathbf{w}^{\star\top}\overline{\mathbf{R}}\left(\mathbf{I}-\eta\widehat{\boldsymbol{\Sigma}}\right)^t\widehat{\mathbf{R}}\boldsymbol{\Sigma}\widehat{\mathbf{R}}\left(\mathbf{I}-\eta\widehat{\boldsymbol{\Sigma}}\right)^t\overline{\mathbf{R}}\mathbf{w}^\star\\&=\underbrace{\mathbf{w}^{\star\top}\overline{\mathbf{R}}\left(\mathbf{I}-\eta\widehat{\boldsymbol{\Sigma}}\right)^t\widehat{\mathbf{R}}\left(\boldsymbol{\Sigma}-\widehat{\boldsymbol{\Sigma}}\right)\widehat{\mathbf{R}}\left(\mathbf{I}-\eta\widehat{\boldsymbol{\Sigma}}\right)^t\overline{\mathbf{R}}\mathbf{w}^\star}_{\text{I}}+\underbrace{\mathbf{w}^{\star\top}\overline{\mathbf{R}}\left(\mathbf{I}-\eta\widehat{\boldsymbol{\Sigma}}\right)^t\widehat{\mathbf{R}}\widehat{\boldsymbol{\Sigma}}\widehat{\mathbf{R}}\left(\mathbf{I}-\eta\widehat{\boldsymbol{\Sigma}}\right)^t\overline{\mathbf{R}}\mathbf{w}^\star}_{\text{II}}..\end{aligned}\tag{F.4}$$

Lemma F.5. *On $\mathcal{E}_1\cap\mathcal{E}_2$, we have*

$$\text{I}\lesssim\frac{1}{t\beta^2}$$

and

$$\text{II}\lesssim\frac{1}{\eta t\beta^2}.$$

hold with probability at least $1-\delta$.

By Lemma F.5, we obtain that with probability at least $1-\delta$,

$$\text{Bias}\lesssim\text{I}+\text{II}\leq\frac{1}{t\beta^2}+\frac{1}{\eta t\beta^2}\lesssim\frac{1}{\eta t\beta^2}\tag{F.5}$$

where the last inequality is by $\eta\lesssim 1/\|\boldsymbol{\Sigma}\|\lesssim 1$.

F.3. Bounding the Variance

$$\begin{aligned}\text{Variance}&=\eta^2\left\|\boldsymbol{\Sigma}^{1/2}\widehat{\mathbf{R}}\left(\frac{1}{n}\sum_{i=1}^t\left(\mathbf{I}-\eta\widehat{\boldsymbol{\Sigma}}\right)^{i-1}\widetilde{\mathbf{X}}^\top\epsilon\right)\right\|_2^2\\&=\frac{\eta^2}{n^2}\epsilon^\top\mathbf{X}\widehat{\mathbf{R}}\sum_{i=1}^t\left(\mathbf{I}-\eta\widehat{\boldsymbol{\Sigma}}\right)^{i-1}\widehat{\mathbf{R}}\boldsymbol{\Sigma}\widehat{\mathbf{R}}\sum_{i=1}^t\left(\mathbf{I}-\eta\widehat{\boldsymbol{\Sigma}}\right)^{i-1}\widehat{\mathbf{R}}\mathbf{X}^\top\epsilon\\&=\underbrace{\frac{\eta^2}{n^2}\epsilon^\top\mathbf{X}\widehat{\mathbf{R}}\sum_{i=1}^t\left(\mathbf{I}-\eta\widehat{\boldsymbol{\Sigma}}\right)^{i-1}\widehat{\mathbf{R}}\left(\boldsymbol{\Sigma}-\widehat{\boldsymbol{\Sigma}}\right)\widehat{\mathbf{R}}\sum_{i=1}^t\left(\mathbf{I}-\eta\widehat{\boldsymbol{\Sigma}}\right)^{i-1}\widehat{\mathbf{R}}\mathbf{X}^\top\epsilon}_{\text{I}}\\&\quad+\underbrace{\frac{\eta^2}{n^2}\epsilon^\top\mathbf{X}\widehat{\mathbf{R}}\sum_{i=1}^t\left(\mathbf{I}-\eta\widehat{\boldsymbol{\Sigma}}\right)^{i-1}\widehat{\mathbf{R}}\widehat{\boldsymbol{\Sigma}}\widehat{\mathbf{R}}\sum_{i=1}^t\left(\mathbf{I}-\eta\widehat{\boldsymbol{\Sigma}}\right)^{i-1}\widehat{\mathbf{R}}\mathbf{X}^\top\epsilon\,.}_{\text{II}}\end{aligned}\tag{F.6}$$

Lemma F.6. *On $\mathcal{E}_1 \cap \mathcal{E}_2$, with probability at least $1-\delta$, we have*

$$\mathrm{I} \lesssim \frac{\eta^2 t}{n^2} \cdot \left\| \widehat{\mathbf{R}} \mathbf{X}^\top \epsilon \right\|_2^2$$

and

$$\mathrm{II} \lesssim \frac{\eta t \log t}{n^2} \cdot \left\| \widehat{\mathbf{R}} \mathbf{X}^\top \epsilon \right\|_2^2.$$

By applying Lemma F.6 to Eq.(F.6), we obtain that

$$\text{Variance} = \mathrm{I} + \mathrm{II} \lesssim \frac{\eta^2 t}{n^2} \cdot \left\| \widehat{\mathbf{R}} \mathbf{X}^\top \epsilon \right\|_2^2 + \frac{\eta t \log t}{n^2} \cdot \left\| \widehat{\mathbf{R}} \mathbf{X}^\top \epsilon \right\|_2^2 \lesssim \frac{\eta t \log t}{n^2} \cdot \left\| \widehat{\mathbf{R}} \mathbf{X}^\top \epsilon \right\|_2^2. \tag{F.7}$$

Lemma F.7. *with probability at least $1-\delta$, we have*

$$\left\| \frac{1}{n} \cdot \widehat{\mathbf{R}} \mathbf{X}^\top \epsilon \right\|_2^2 \lesssim \frac{\sigma^2 \mathrm{Tr}(\mathbf{R} \boldsymbol{\Sigma} \mathbf{R}) \log(d/\delta)}{n} + \frac{\sigma^2 s \mathrm{Tr}(\boldsymbol{\Sigma}) \log^2(d/\delta)}{n^2}$$

By applying Lemma F.7 to Eq.(F.7), we obtain that

$$\text{Variance} \lesssim \eta t \log t \cdot \left(\frac{\sigma^2 \mathrm{Tr}(\mathbf{R} \boldsymbol{\Sigma} \mathbf{R}) \log(d/\delta)}{n} + \frac{\sigma^2 s \mathrm{Tr}(\boldsymbol{\Sigma}) \log^2(d/\delta)}{n^2} \right). \tag{F.8}$$

F.4. Final Bound

Combining Eq.(F.5) and Eq.(F.8), we obtain that

$$\begin{aligned} \mathcal{E}(\widehat{\mathbf{w}}_t) &\le \frac{1}{\eta t \beta^2} + \eta t \log t \cdot \left(\frac{\sigma^2 \mathrm{Tr}(\mathbf{R} \boldsymbol{\Sigma} \mathbf{R}) \log(d/\delta)}{n} + \frac{\sigma^2 s \mathrm{Tr}(\boldsymbol{\Sigma}) \log^2(d/\delta)}{n^2} \right) \\ &\lesssim \frac{\log t}{\beta} \sqrt{\frac{\sigma^2 \mathrm{Tr}(\mathbf{R} \boldsymbol{\Sigma} \mathbf{R}) \log(d/\delta)}{n} + \frac{\sigma^2 s \mathrm{Tr}(\boldsymbol{\Sigma}) \log^2(d/\delta)}{n^2}}, \end{aligned}$$

when $\eta t \simeq \frac{1}{\beta} \cdot \left(\frac{\sigma^2 \mathrm{Tr}(\mathbf{R} \boldsymbol{\Sigma} \mathbf{R}) \log(d/\delta)}{n} + \frac{\sigma^2 s \mathrm{Tr}(\boldsymbol{\Sigma}) \log^2(d/\delta)}{n^2} \right)^{-1/2}$.

F.5. Proof for Appendix F

Proof of Lemma F.1. Since $y_i = \sum_{j=1}^d w_j^\star x_{ij} + \epsilon_i$, then we have

$$\widehat{r}_i = \frac{1}{n} \sum_{j=1}^n x_{ji} y_j = \frac{1}{n} \sum_{j=1}^n x_{ji} \cdot \left(\sum_{k=1}^d w_k^\star x_{jk} + \epsilon_j \right) = \sum_{k=1}^d \frac{w_k^\star}{n} \sum_{j=1}^n x_{jk} x_{ji} + \frac{1}{n} \sum_{j=1}^n x_{ji} \epsilon_j. \tag{F.9}$$

Since $x_{ji} \sim \mathsf{N}(0, \Sigma_{ii})$ for any i, j, by Lemma 2.7.7 in Vershynin (2020), there exists an absolute constant C such that $x_{jk} x_{ji}$ is a sub-exponential random variable with

$$\|x_{jk} x_{ji}\|_{\Psi_1} \le C \sqrt{\Sigma_{kk} \Sigma_{ii}} \le K,$$

where $\| \cdot \|_{\Psi_1}$ denotes the sub-exponential norm and the last inequality comes from the definition of K. By applying Bernstein's inequality (**?**)Theorem 2.8.1]vershynin2020high, we have

$$\begin{aligned} \left| \frac{1}{n} \sum_{j=1}^n x_{jk} x_{ji} - \mathbb{E}[x_{1k} x_{1i}] \right| &= \left| \frac{1}{n} \sum_{j=1}^n x_{jk} x_{ji} - \Sigma_{ki} \right| \\ &\le K \cdot \max \left\{ \sqrt{\frac{\log(d/\delta)}{n}}, \frac{\log(d/\delta)}{n} \right\} \\ &= K \cdot \sqrt{\frac{\log(d/\delta)}{n}}, \end{aligned} \tag{F.10}$$

where the last equality due to $n \geq \mathcal{O}(\log(d/\delta))$. We also note that $x_{ji}\epsilon_j$ is a sub-exponential random variable with $\|x_{ji}\epsilon_j\|_{\Psi_1} \leq K$. Hence, we also have

$$\left|\frac{1}{n}\sum_{j=1} x_{ji}\epsilon_j\right| \lesssim K \cdot \sqrt{\frac{\log(d/\delta)}{n}}. \tag{F.11}$$

Combining Eq.(F.9), Eq.(F.10) and Eq.(F.11), we have

$$|\widehat{r}_i - r_i| \lesssim K \cdot \sqrt{\frac{\log(d/\delta)}{n}} \sum_{k=1}^{d} |w_k^\star| + K \cdot \sqrt{\frac{\log(d/\delta)}{n}} = (\|w^\star\|_1 + 1)K \cdot \sqrt{\frac{\log(d/\delta)}{n}}.$$

By definition of $\widehat{\mathbf{R}}$ and $\mathbf{R}$, we obtain

$$\begin{aligned}\|\widehat{\mathbf{R}} - \mathbf{R}\|_2 &= \max_i |\widehat{r}_i - r_i| \leq K(\|w^\star\|_1 + 1) \cdot \sqrt{\frac{\log(d/\delta)}{n}}\\ &\leq K\left(\sqrt{s\|\mathbf{w}^\star\|_2^2} + 1\right) \cdot \sqrt{\frac{\log(d/\delta)}{n}} \lesssim K \cdot \sqrt{\frac{s\log(d/\delta)}{n}},\end{aligned}$$

which completes the proof. □

Proof of Lemma F.2. By Lemma F.1, for any $j \in \mathcal{S}$, with probability at least $1-\delta$, we have

$$|r_i - \widehat{r}_j| \lesssim \sqrt{\frac{s\log(d/\delta)}{n}} \lesssim \beta/2 \leq |r_j|/2, \tag{F.12}$$

where the last inequality is due to the definition of β. □

Proof of Lemma F.4. We can decompose $\|\widehat{\mathbf{R}}\widehat{\mathbf{\Sigma}}\widehat{\mathbf{R}} - \mathbf{R}\mathbf{\Sigma}\mathbf{R}\|_2$ as follows:

$$\begin{aligned}\|\widehat{\mathbf{R}}\widehat{\mathbf{\Sigma}}\widehat{\mathbf{R}} - \mathbf{R}\mathbf{\Sigma}\mathbf{R}\|_2 &= \|\widehat{\mathbf{R}}\widehat{\mathbf{\Sigma}}\widehat{\mathbf{R}} - \mathbf{R}\widehat{\mathbf{\Sigma}}\widehat{\mathbf{R}} + \mathbf{R}\widehat{\mathbf{\Sigma}}\widehat{\mathbf{R}} - \mathbf{R}\mathbf{\Sigma}\widehat{\mathbf{R}} + \mathbf{R}\mathbf{\Sigma}\widehat{\mathbf{R}} - \mathbf{R}\mathbf{\Sigma}\mathbf{R}\|_2\\ &\leq \underbrace{\|\widehat{\mathbf{R}}\widehat{\mathbf{\Sigma}}\widehat{\mathbf{R}} - \mathbf{R}\widehat{\mathbf{\Sigma}}\widehat{\mathbf{R}}\|_2}_{\text{I}} + \underbrace{\|\mathbf{R}\widehat{\mathbf{\Sigma}}\widehat{\mathbf{R}} - \mathbf{R}\mathbf{\Sigma}\widehat{\mathbf{R}}\|_2}_{\text{II}} + \underbrace{\|\mathbf{R}\mathbf{\Sigma}\widehat{\mathbf{R}} - \mathbf{R}\mathbf{\Sigma}\mathbf{R}\|_2}_{\text{III}}.\end{aligned} \tag{F.13}$$

Next, we proof the bound for I, II and III separately.

For term I,

$$\begin{aligned}\text{I} &= \|\widehat{\mathbf{R}}\widehat{\mathbf{\Sigma}}\widehat{\mathbf{R}} - \mathbf{R}\widehat{\mathbf{\Sigma}}\widehat{\mathbf{R}}\|_2 = \|\left(\widehat{\mathbf{R}} - \mathbf{R}\right)\widehat{\mathbf{\Sigma}}\widehat{\mathbf{R}}\|_2\\ &\leq \|\widehat{\mathbf{R}} - \mathbf{R}\|_2 \cdot \|\widehat{\mathbf{\Sigma}}\|_2 \cdot \|\widehat{\mathbf{R}}\|_2\\ &\leq \|\widehat{\mathbf{R}} - \mathbf{R}\|_2 \cdot \left(\|\mathbf{\Sigma}\|_2 + \|\widehat{\mathbf{\Sigma}} - \mathbf{\Sigma}\|_2\right) \cdot \left(\|\mathbf{R}\|_2 + \|\mathbf{R} - \widehat{\mathbf{R}}\|_2\right),\end{aligned} \tag{F.14}$$

where the last line is due to triangle inequality. By Lemma F.3, with probability at least $1-\delta/3$, we have

$$\begin{aligned}\|\widehat{\mathbf{\Sigma}} - \mathbf{\Sigma}\|_2 &\lesssim \|\mathbf{\Sigma}\|_2 \cdot \max\left\{\sqrt{\frac{r(\mathbf{\Sigma})}{n}}, \frac{r(\mathbf{\Sigma})}{n}, \sqrt{\frac{\log(1/\delta)}{n}}, \frac{\log(1/\delta)}{n}\right\}\\ &\lesssim \|\mathbf{\Sigma}\|_2 \cdot \max\left\{\sqrt{\frac{r(\mathbf{\Sigma}) + \log(1/\delta)}{n}}, \frac{r(\mathbf{\Sigma}) + \log(1/\delta)}{n}\right\}.\end{aligned} \tag{F.15}$$

By Lemma F.1, we obtain that

$$\|\widehat{\mathbf{R}} - \mathbf{R}\|_2 \leq K \cdot \sqrt{\frac{s\log(d/\delta)}{n}} \lesssim 1 \tag{F.16}$$

holds with probability at least $1-\delta/3$, where the last inequality is valid since $n \gtrsim K^2 s\|\mathbf{R}\|_2^2 \log(d/\delta)$. Combing Eq.(F.14), Eq.(F.15) and Eq.(F.16), we have

$$\begin{aligned} \mathrm{I} &\lesssim K\|\mathbf{\Sigma}\|_2\sqrt{\frac{s\log(d/\delta)}{n}} \cdot \left(1+\max\left\{\sqrt{\frac{r(\mathbf{\Sigma})+\log(1/\delta)}{n}}, \frac{r(\mathbf{\Sigma})+\log(1/\delta)}{n}\right\}\right) \\ &\leq K\|\mathbf{\Sigma}\|_2\sqrt{s\frac{\log(d/\delta)}{n}} \cdot \left(1+\sqrt{\frac{r(\mathbf{\Sigma})+\log(1/\delta)}{n}}+\frac{r(\mathbf{\Sigma})+\log(1/\delta)}{n}\right). \end{aligned} \tag{F.17}$$

For term II, we can decompose II as follows:

$$\|\mathbf{R}\left(\widehat{\mathbf{\Sigma}}-\mathbf{\Sigma}\right)\widehat{\mathbf{R}}\|_2 \leq \underbrace{\|\mathbf{R}\left(\widehat{\mathbf{\Sigma}}-\mathbf{\Sigma}\right)\mathbf{R}\|_2}_{\mathrm{II.a}} + \underbrace{\|\mathbf{R}\left(\widehat{\mathbf{\Sigma}}-\mathbf{\Sigma}\right)\left(\widehat{\mathbf{R}}-\mathbf{R}\right)\|_2}_{\mathrm{II.b}}.$$

For term II.a, by using Lemma F.3, we have with probability at least $1-\delta/3$,

$$\begin{aligned} \mathrm{II.a} &\lesssim \|\mathbf{R\Sigma R}\|_2 \cdot \max\left\{\sqrt{\frac{r(\mathbf{R\Sigma R})}{n}}, \frac{r(\mathbf{R\Sigma R})}{n}, \sqrt{\frac{\log(1/\delta)}{n}}, \frac{\log(1/\delta)}{n}\right\} \\ &\lesssim \|\mathbf{R\Sigma R}\|_2 \cdot \max\left\{\sqrt{\frac{r(\mathbf{R\Sigma R})+\log(1/\delta)}{n}}, \frac{r(\mathbf{R\Sigma R})+\log(1/\delta)}{n}\right\} \\ &\leq \|\mathbf{R\Sigma R}\|_2 \cdot \left(\sqrt{\frac{r(\mathbf{R\Sigma R})+\log(1/\delta)}{n}}+\frac{r(\mathbf{R\Sigma R})+\log(1/\delta)}{n}\right) \end{aligned} \tag{F.18}$$

Similar to the proof for bounding I, we can obtain that

$$\mathrm{II.b} \lesssim K\|\mathbf{\Sigma}\|_2\sqrt{\frac{s\log(d/\delta)}{n}} \cdot \left(1+\sqrt{\frac{r(\mathbf{\Sigma})+\log(1/\delta)}{n}}+\frac{r(\mathbf{\Sigma})+\log(1/\delta)}{n}\right). \tag{F.19}$$

For term III, we have

$$\mathrm{III} = \|\mathbf{R\Sigma}\left(\widehat{\mathbf{R}}-\mathbf{R}\right)\|_2 \leq |\mathbf{R}\|_2\|\mathbf{\Sigma}\|_2 K(\|\mathbf{w}^\star\|_1+1) \cdot \sqrt{\frac{s\log(d/\delta)}{n}}, \tag{F.20}$$

where the last inequality is by Eq.(F.16).

Combining Eq.(F.17), Eq.(F.18), Eq.(F.19) and Eq.(F.20) and taking the union bound, we obtain that with probability at least $1-\delta$,

$$
\begin{aligned}
&\|\widehat{\mathbf{R}}\widehat{\mathbf{\Sigma}}\widehat{\mathbf{R}}-\mathbf{R}\mathbf{\Sigma}\mathbf{R}\|_2 \le \mathrm{I}+\mathrm{II}+\mathrm{III}\\
&\lesssim K\|\mathbf{\Sigma}\|_2(\|\mathbf{w}^\star\|_1+1)\sqrt{\frac{\log(d/\delta)}{n}}\cdot\left(1+\sqrt{\frac{r(\mathbf{\Sigma})+\log(1/\delta)}{n}}+\frac{r(\mathbf{\Sigma})+\log(1/\delta)}{n}\right)\\
&+\|\mathbf{R}\mathbf{\Sigma}\mathbf{R}\|_2\cdot\left(\sqrt{\frac{r(\mathbf{R}\mathbf{\Sigma}\mathbf{R})+\log(1/\delta)}{n}}+\frac{r(\mathbf{R}\mathbf{\Sigma}\mathbf{R})+\log(1/\delta)}{n}\right)\\
&+\|\mathbf{R}\|_2\|\mathbf{\Sigma}\|_2K(\|\mathbf{w}^\star\|_1+1)\cdot\sqrt{\frac{\log(d/\delta)}{n}}\\
&\le (K\|\mathbf{\Sigma}\|_2(\|\mathbf{w}^\star\|_1+1)+\|\mathbf{R}\mathbf{\Sigma}\mathbf{R}\|_2+\|\mathbf{R}\|_2\|\mathbf{\Sigma}\|_2K(\|\mathbf{w}^\star\|_1+1))\\
&\cdot\left(\sqrt{\frac{\log(d/\delta)}{n}}\cdot\left(2+\sqrt{\frac{r(\mathbf{\Sigma})+\log(1/\delta)}{n}}+\frac{r(\mathbf{\Sigma})+\log(1/\delta)}{n}\right)\right.\\
&\left.+\sqrt{\frac{r(\mathbf{R}\mathbf{\Sigma}\mathbf{R})+\log(1/\delta)}{n}}+\frac{r(\mathbf{R}\mathbf{\Sigma}\mathbf{R})+\log(1/\delta)}{n}\right)\\
&\lesssim \widetilde{C}_{\mathrm{cov}}\cdot\left(\sqrt{\frac{r(\mathbf{R}\mathbf{\Sigma}\mathbf{R})+\log(1/\delta)}{n}}+\frac{\sqrt{r(\mathbf{\Sigma})\log(d/\delta)}+r(\mathbf{R}\mathbf{\Sigma}\mathbf{R})+\log(d/\delta)}{n}\right.\\
&\left.+\frac{r(\mathbf{\Sigma})\sqrt{\log(d/\delta)}+\log^{3/2}(d/\delta)}{n^{3/2}}\right)\\
&\lesssim \widetilde{C}_{\mathrm{cov}}\cdot\mathrm{poly}(\log(d/\delta))\cdot\left(\sqrt{\frac{r(\mathbf{R}\mathbf{\Sigma}\mathbf{R})}{n}}+\frac{\sqrt{r(\mathbf{\Sigma})}+r(\mathbf{R}\mathbf{\Sigma}\mathbf{R})}{n}+\frac{r(\mathbf{\Sigma})}{n^{3/2}}\right),
\end{aligned}
$$

where the second last inequality is by $aa'+bb'+cc' \le (a+b+c)(a'+b'+c')$ for $a,a',b,b',c,c' \ge 0$. Here $\widetilde{C}_{\mathrm{cov}} = K\|\mathbf{\Sigma}\|_2(\|\mathbf{w}^\star\|_1+1)+\|\mathbf{R}\mathbf{\Sigma}\mathbf{R}\|_2+\|\mathbf{R}\|_2\|\mathbf{\Sigma}\|_2K(\|\mathbf{w}^\star\|_1+1)\lesssim\sqrt{s}$. □

Proof of Lemma F.5. By the triangle inequality, we have

$$
\begin{aligned}
&\left\|\widehat{\mathbf{R}}\left(\mathbf{\Sigma}-\widehat{\mathbf{\Sigma}}\right)\widehat{\mathbf{R}}\right\|_2\\
&=\left\|\mathbf{R}\left(\mathbf{\Sigma}-\widehat{\mathbf{\Sigma}}\right)\mathbf{R}+\mathbf{R}\left(\mathbf{\Sigma}-\widehat{\mathbf{\Sigma}}\right)\left(\widehat{\mathbf{R}}-\mathbf{R}\right)+\left(\widehat{\mathbf{R}}-\mathbf{R}\right)\left(\mathbf{\Sigma}-\widehat{\mathbf{\Sigma}}\right)\mathbf{R}+\left(\widehat{\mathbf{R}}-\mathbf{R}\right)\left(\mathbf{\Sigma}-\widehat{\mathbf{\Sigma}}\right)\left(\widehat{\mathbf{R}}-\mathbf{R}\right)\right\|_2\\
&\le\left\|\mathbf{R}\left(\mathbf{\Sigma}-\widehat{\mathbf{\Sigma}}\right)\mathbf{R}\right\|_2+\left\|\mathbf{R}\left(\mathbf{\Sigma}-\widehat{\mathbf{\Sigma}}\right)\left(\widehat{\mathbf{R}}-\mathbf{R}\right)\right\|_2+\left\|\left(\widehat{\mathbf{R}}-\mathbf{R}\right)\left(\mathbf{\Sigma}-\widehat{\mathbf{\Sigma}}\right)\mathbf{R}\right\|_2+\left\|\left(\widehat{\mathbf{R}}-\mathbf{R}\right)\left(\mathbf{\Sigma}-\widehat{\mathbf{\Sigma}}\right)\left(\widehat{\mathbf{R}}-\mathbf{R}\right)\right\|_2.
\end{aligned}
$$

Following the proof of Lemma F.4, we can prove that with probability at least $1-\delta$,

$$
\left\|\widehat{\mathbf{R}}\left(\mathbf{\Sigma}-\widehat{\mathbf{\Sigma}}\right)\widehat{\mathbf{R}}\right\|_2\lesssim\widetilde{\alpha}(n,\delta)\le 1/t, \tag{F.21}
$$

where the last inequality is by $\mathcal{E}_2$. By Eq.(F.21), we have

$$
\widehat{\mathbf{R}}\left(\mathbf{\Sigma}-\widehat{\mathbf{\Sigma}}\right)\widehat{\mathbf{R}}\preceq 1/t\cdot\mathbf{I}.
$$

Hence, we obtain that

$$
\begin{aligned}
\mathrm{I}&\lesssim\mathbf{w}^{\star\top}\overline{\mathbf{R}}\left(\mathbf{I}-\eta\widehat{\mathbf{\Sigma}}\right)^t\cdot 1/t\cdot\mathbf{I}\cdot\left(\mathbf{I}-\eta\widehat{\mathbf{\Sigma}}\right)^t\overline{\mathbf{R}}\mathbf{w}^\star\\
&=\frac{1}{t}\mathbf{w}^{\star\top}\overline{\mathbf{R}}\left(\mathbf{I}-\eta\widehat{\mathbf{\Sigma}}\right)^{2t}\overline{\mathbf{R}}\mathbf{w}^\star\\
&\le\frac{1}{t}\mathbf{w}^{\star\top}\overline{\mathbf{R}}\overline{\mathbf{R}}\mathbf{w}^\star && \left(\text{by }\left(\mathbf{I}-\eta\widehat{\mathbf{\Sigma}}\right)^{2t}\preceq\mathbf{I}\right)\\
&\le\frac{1}{t}\|\mathbf{w}^\star\|_2^2, && \text{(F.22)}
\end{aligned}
$$

where the last line by $\overline{\mathbf{R}} \preceq \frac{2}{\beta} \cdot \mathbf{I}$. For the term II, we have

$$\begin{aligned}
\mathrm{II} &= \mathbf{w}^{\star\top}\overline{\mathbf{R}}\Big(\mathbf{I} - \eta\widehat{\boldsymbol{\Sigma}}\Big)^t\widehat{\mathbf{R}}\boldsymbol{\Sigma}\widehat{\mathbf{R}}\Big(\mathbf{I} - \eta\widehat{\boldsymbol{\Sigma}}\Big)^t\overline{\mathbf{R}}\mathbf{w}^\star \\
&\lesssim \frac{1}{\eta t}\mathbf{w}^{\star\top}\overline{\mathbf{R}}\overline{\mathbf{R}}\mathbf{w}^\star \\
&\frac{1}{\eta t\beta^2}\|\mathbf{w}^\star\|_2^2 \le \frac{1}{\eta t\beta^2},
\end{aligned} \tag{F.23}$$

where the second last line is by the fact that $x(1-x)^k \le 1/(k+1)$ for all $x \in [0,1]$ and all $k > 0$. □

Proof of Lemma F.6. Similar to the proof of Lemma F.5, with probability at least $1-\delta$, we have $\widehat{\mathbf{R}}\Big(\boldsymbol{\Sigma} - \widehat{\boldsymbol{\Sigma}}\Big)\widehat{\mathbf{R}} \preceq \frac{1}{t} \cdot \mathbf{I}$. Then we have

$$\begin{aligned}
\mathrm{I} &= \frac{\eta^2}{n^2}\epsilon^\top\mathbf{X}\widehat{\mathbf{R}}\sum_{i=1}^t\Big(\mathbf{I} - \eta\widehat{\boldsymbol{\Sigma}}\Big)^{i-1}\widehat{\mathbf{R}}\Big(\boldsymbol{\Sigma} - \widehat{\boldsymbol{\Sigma}}\Big)\widehat{\mathbf{R}}\sum_{i=1}^t\Big(\mathbf{I} - \eta\widehat{\boldsymbol{\Sigma}}\Big)^{i-1}\widehat{\mathbf{R}}\mathbf{X}^\top\epsilon \\
&\lesssim \frac{\eta^2}{tn^2}\epsilon^\top\mathbf{X}\widehat{\mathbf{R}}\sum_{i=1}^t\Big(\mathbf{I} - \eta\widehat{\boldsymbol{\Sigma}}\Big)^{i-1}\sum_{i=1}^t\Big(\mathbf{I} - \eta\widehat{\boldsymbol{\Sigma}}\Big)^{i-1}\widehat{\mathbf{R}}\mathbf{X}^\top\epsilon \\
&\le \frac{\eta^2 t}{n^2}\epsilon^\top\mathbf{X}\widehat{\mathbf{R}}\widehat{\mathbf{R}}\mathbf{X}^\top\epsilon \\
&= \frac{\eta^2 t}{n^2} \cdot \Big\|\widehat{\mathbf{R}}\mathbf{X}^\top\epsilon\Big\|_2^2,
\end{aligned}$$

where the second last line is by $\sum_{i=1}^t\Big(\mathbf{I} - \eta\widehat{\boldsymbol{\Sigma}}\Big)^{i-1} \preceq t \cdot \mathbf{I}$. By the fact that $x(1-x)^k \le 1/(k+1)$ for all $x \in [0,1]$ and all $k > 0$, we have

$$\begin{aligned}
\mathrm{II} &= \frac{\eta^2}{n^2}\epsilon^\top\mathbf{X}\widehat{\mathbf{R}}\sum_{i=1}^t\Big(\mathbf{I} - \eta\widehat{\boldsymbol{\Sigma}}\Big)^{i-1}\widehat{\mathbf{R}}\widehat{\boldsymbol{\Sigma}}\widehat{\mathbf{R}}\sum_{i=1}^t\Big(\mathbf{I} - \eta\widehat{\boldsymbol{\Sigma}}\Big)^{i-1}\widehat{\mathbf{R}}\mathbf{X}^\top\epsilon \\
&= \frac{\eta}{n^2}\epsilon^\top\mathbf{X}\widehat{\mathbf{R}}\left(\sum_{i,j=1}^t\Big(\mathbf{I} - \eta\widehat{\boldsymbol{\Sigma}}\Big)^{i+j-2}\eta\widehat{\mathbf{R}}\widehat{\boldsymbol{\Sigma}}\right)\widehat{\mathbf{R}}\mathbf{X}^\top\epsilon \\
&\le \frac{\eta}{n^2} \cdot (\sum_{i,j=1}^t\frac{1}{i+j-1})\Big\|\widehat{\mathbf{R}}\mathbf{X}^\top\epsilon\Big\|_2^2 \\
&\le \frac{\eta t}{n^2} \cdot (\sum_{i=1}^t\frac{1}{i})\Big\|\widehat{\mathbf{R}}\mathbf{X}^\top\epsilon\Big\|_2^2 \\
&\lesssim \frac{\eta t\log t}{n^2} \cdot \Big\|\widehat{\mathbf{R}}\mathbf{X}^\top\epsilon\Big\|_2^2,
\end{aligned}$$

where the last inequality is by the fact that $\sum_{i=1}^t\frac{1}{i} \lesssim \log t$. □

Proof of Lemma F.7. First, we can decompose $\Big\|\frac{1}{n} \cdot \widehat{\mathbf{R}}\mathbf{X}^\top\epsilon\Big\|_2^2$ by

$$\Big\|\frac{1}{n} \cdot \widehat{\mathbf{R}}\mathbf{X}^\top\epsilon\Big\|_2^2 \lesssim \Big\|\frac{1}{n} \cdot \mathbf{R}\mathbf{X}^\top\epsilon\Big\|_2^2 + \Big\|\frac{1}{n} \cdot \Big(\widehat{\mathbf{R}} - \mathbf{R}\Big)\mathbf{X}^\top\epsilon\Big\|_2^2.$$

Let $\mathbf{z}_i = \mathbf{R}\mathbf{x}_i$, then $\mathbf{z}_i \sim \mathsf{N}(\mathbf{G})$, where $\mathbf{G} := \mathbf{R}\boldsymbol{\Sigma}\mathbf{R}$. For any i, j, by Lemma 2.7.7 in Vershynin (2020), there exists an absolute constant C such that $\epsilon_j z_{ji}$ is a sub-exponential random variable with

$$\|\epsilon_j z_{ji}\|_{\Psi_1} \le C\sigma\sqrt{G_{ii}}.$$

By applying Bernstein's inequality Vershynin (2020, Theorem 2.8.1), for any $1 \le i \le d$, we have that

$$\begin{aligned}\left|\frac{1}{n}\sum_{j=1}^{n}\epsilon_j z_{ji} - \mathbb{E}[\epsilon_1 z_{1i}]\right| &= \left|\frac{1}{n}\sum_{j=1}^{n}\epsilon_j z_{ji}\right| \\ &\lesssim \sigma\sqrt{G_{ii}}\cdot\max\left\{\sqrt{\frac{\log(d/\delta)}{n}}, \frac{\log(d/\delta)}{n}\right\} = \sigma\sqrt{G_{ii}}\cdot\sqrt{\frac{\log(d/\delta)}{n}} \end{aligned}\tag{F.24}$$

hold with probability $1-\frac{\delta}{3d}$, where the last inequality is due to $n \ge \mathcal{O}(\log(d/\delta))$. By taking the union bound, we obtain that

$$\left|\frac{1}{n}\sum_{j=1}^{n}\epsilon_j z_{ji}\right| \lesssim \sigma\sqrt{G_{ii}}\cdot\sqrt{\frac{\log(d/\delta)}{n}}$$

holds for any i, with probability $1-\frac{\delta}{3}$. Then we have

$$\mathrm{I} = \sum_{i=1}^{d}\left(\frac{1}{n}\sum_{j=1}^{n}\epsilon_j \mathbf{z}_{ji}\right)^2 \lesssim \sum_{i=1}^{d}\sigma^2 G_{ii}\cdot\frac{\log(d/\delta)}{n} = \frac{\sigma^2\mathrm{Tr}(\mathbf{R}\boldsymbol{\Sigma}\mathbf{R})\log(d/\delta)}{n}.$$

In the same way, we can prove that with probability at least $1-\delta/3$,

$$\left\|\frac{1}{n}\mathbf{X}^\top\epsilon\right\|_2^2 \lesssim \frac{\sigma^2\mathrm{Tr}(\Sigma)\log(d/\delta)}{n}.\tag{F.25}$$

By applying Lemma F.1, with probability at least $1-\delta/3$, we have

$$\left\|\widehat{\mathbf{R}}-\mathbf{R}\right\|_2^2 \lesssim \frac{s\log(d/\delta)}{n}.\tag{F.26}$$

By Eq.(F.25) and Eq.(F.26), with probability $1-2\delta/3$, we have

$$\left\|\frac{1}{n}\cdot\left(\widehat{\mathbf{R}}-\mathbf{R}\right)\mathbf{X}^\top\epsilon\right\|_2^2 \le \left\|\widehat{\mathbf{R}}-\mathbf{R}\right\|_2^2\left\|\frac{1}{n}\mathbf{X}^\top\epsilon\right\|_2^2 \lesssim \frac{\sigma^2 s\mathrm{Tr}(\boldsymbol{\Sigma})\log^2(d/\delta)}{n^2}.$$

By taking the union bound, we derive the desired result. □

G. Proof for Theorem E.2

To simplify the notations, we use $\mathbf{w}_t$ to denote $\mathbf{w}_{\mathsf{gd}}^t$.

Lemma G.1. *with probability at least $1-\delta$, we have*

$$\left\|\widehat{\boldsymbol{\Sigma}}-\boldsymbol{\Sigma}\right\| \lesssim \alpha(n,\delta),\tag{G.1}$$

where $\alpha(n,\delta) = \sqrt{\frac{\mathrm{Tr}(\boldsymbol{\Sigma})+\log(1/\delta)}{n}} + \frac{\mathrm{Tr}(\boldsymbol{\Sigma})+\log(1/\delta)}{n}$. As a result, when $n \gtrsim t^2(\mathrm{Tr}(\boldsymbol{\Sigma})+\log(1/\delta))$, with probability at least $1-\delta$,

$$\left\|\widehat{\boldsymbol{\Sigma}}-\boldsymbol{\Sigma}\right\| \lesssim 1/t.$$

Proof of Lemma G.1. By Lemma F.3, we have

$$\begin{aligned}\|\widehat{\boldsymbol{\Sigma}}-\boldsymbol{\Sigma}\|_2 &\le c\|\boldsymbol{\Sigma}\|_2\cdot\max\left\{\sqrt{\frac{r(\boldsymbol{\Sigma})}{n}}, \frac{r(\boldsymbol{\Sigma})}{n}, \sqrt{\frac{\log(1/\delta)}{n}}, \frac{\log(1/\delta)}{n}\right\} \\ &\lesssim \max\left\{\sqrt{\frac{r(\boldsymbol{\Sigma})+\log(1/\delta)}{n}}, \frac{r(\boldsymbol{\Sigma})+\log(1/\delta)}{n}\right\} \\ &\le \sqrt{\frac{r(\boldsymbol{\Sigma})+\log(1/\delta)}{n}} + \frac{r(\boldsymbol{\Sigma})+\log(1/\delta)}{n}\end{aligned}\tag{G.2}$$

holds with probability at least $1-\delta$, where the last line is by the inequality that $\max\{a,b\} \leq a+b$ for all $a,b \geq 0$. □

We define the event $\mathcal{E}$ as follows:

$$\mathcal{E} := \Big\{ \mathbf{R}\boldsymbol{\Sigma}\mathbf{R}\|_2 \lesssim \alpha(n,\delta) \leq 1/t \Big\}.$$

By Lemma G.1, $\mathbb{P}(\mathcal{E}) \geq 1-\delta$. Hereafter, we condition on $\mathcal{E}$.

Bias-variance Decomposition Similar to Eq.(F.1), we have

$$\mathbf{w}_t = \Big(\mathbf{I} - \Big(\mathbf{I} - \eta\widehat{\boldsymbol{\Sigma}}\Big)^t\Big)\mathbf{w}^\star + \frac{1}{n}\sum_{i=1}^{t}\Big(\mathbf{I} - \eta\widehat{\boldsymbol{\Sigma}}\Big)^{i-1}\mathbf{X}^\top\epsilon. \tag{G.3}$$

In the same way, we can decompose the risk $\mathcal{E}(\mathbf{w}_t)$ by

$$\mathcal{E}(\mathbf{w}_t) = \underbrace{\Big\|\boldsymbol{\Sigma}^{1/2}\Big(\mathbf{I} - \eta\widehat{\boldsymbol{\Sigma}}\Big)^t\mathbf{w}^\star\Big\|_2^2}_{\text{Bias}} + \underbrace{\eta^2\Big\|\boldsymbol{\Sigma}^{1/2}\Big(\frac{1}{n}\sum_{i=1}^{t}\Big(\mathbf{I} - \eta\widehat{\boldsymbol{\Sigma}}\Big)^{i-1}\mathbf{X}^\top\epsilon\Big)\Big\|_2^2}_{\text{Variance}}. \tag{G.4}$$

Bounding the Bias

$$\begin{aligned}
\text{Bias} &= \mathbf{w}^{\star\top}\Big(\mathbf{I} - \eta\widehat{\boldsymbol{\Sigma}}\Big)^t\boldsymbol{\Sigma}\Big(\mathbf{I} - \eta\widehat{\boldsymbol{\Sigma}}\Big)^t\mathbf{w}^\star \\
&= \underbrace{\mathbf{w}^{\star\top}\Big(\mathbf{I} - \eta\widehat{\boldsymbol{\Sigma}}\Big)^t\Big(\boldsymbol{\Sigma} - \widehat{\boldsymbol{\Sigma}}\Big)\Big(\mathbf{I} - \eta\widehat{\boldsymbol{\Sigma}}\Big)^t\mathbf{w}^\star}_{\text{I}} + \underbrace{\mathbf{w}^{\star\top}\Big(\mathbf{I} - \eta\widehat{\boldsymbol{\Sigma}}\Big)^t\widehat{\boldsymbol{\Sigma}}\Big(\mathbf{I} - \eta\widehat{\boldsymbol{\Sigma}}\Big)^t\mathbf{w}^\star}_{\text{II}}.
\end{aligned}$$

Similar to the proof of Lemma F.5, we have the following lemma.

Lemma G.2. *On $\mathcal{E}$, we have*

$$\text{I} \lesssim \frac{1}{t}$$

and

$$\text{II} \lesssim \frac{1}{\eta t}$$

hold with probability at least $1-\delta$.

As a result, the bound of the bias term is given by

$$\text{Bias} \leq \frac{1}{\eta t} + \frac{1}{t} \lesssim \frac{1}{\eta t}. \tag{G.5}$$

Bounding the Variance By using the same way of the proof for bounding the variance term of Theorem E.1, we have the following lemma.

Lemma G.3. *On $\mathcal{E}$, with probability at least $1-\delta$, we have that*

$$\text{Variance} \lesssim \eta t \log t \cdot \Big\|\frac{1}{n}\cdot\mathbf{X}^\top\epsilon\Big\|_2^2 \lesssim \eta t \log t \cdot \frac{\sigma^2\mathrm{Tr}(\Sigma)\log(d/\delta)}{n}. \tag{G.6}$$

Combining Eq.(G.5) and Eq.(G.6), we obtain that

$$\mathcal{E}(\mathbf{w}_t) \lesssim \frac{1}{\eta t} + \eta t \log t \cdot \frac{\sigma^2\mathrm{Tr}(\Sigma)\log(d/\delta)}{n} \lesssim \log t \cdot \sqrt{\frac{\sigma^2\mathrm{Tr}(\Sigma)\log(d/\delta)}{n}},$$

when $\eta t \simeq \Big(\frac{\sigma^2\mathrm{Tr}(\Sigma)\log(d/\delta)}{n}\Big)^{-1/2}$

H. Proof for Theorem 5.1

To simplify the notation, we use $\widehat{\mathbf{w}}_t$ to denote $\widetilde{\mathbf{w}}_{\mathsf{gd}}^t$ and $\mathbf{w}_t$ to denote $\mathbf{w}_{\mathsf{gd}}^t$.

H.1. Proof for the upper bound of the excess risk

When $\mathbf{\Sigma} = \mathbf{I}$, by Eq.(F.2), we have

$$\mathcal{E}(\widehat{\mathbf{w}}_t) = \underbrace{\left\|\widehat{\mathbf{R}}\left(\mathbf{I} - \eta\widehat{\mathbf{\Sigma}}\right)^t \overline{\mathbf{R}}\mathbf{w}^\star\right\|_2^2}_{\text{Bias}} + \underbrace{\eta^2\left\|\widehat{\mathbf{R}}\left(\frac{1}{n}\sum_{i=1}^t \left(\mathbf{I} - \eta\widehat{\mathbf{\Sigma}}\right)^{i-1}\widetilde{\mathbf{X}}^\top \epsilon\right)\right\|_2^2}_{\text{Variance}}.$$

Following the proof of Theorem E.1, it holds that

$$\text{Variance} \lesssim \eta t \log t \cdot \frac{\sigma^2 \log\left(d/\delta\right)}{n} + \frac{\sigma^2 s d \log^2\left(d/\delta\right)}{n^2}$$

with probability at least $1-\delta$, when $n \gtrsim t^2 s d^{2/3}$

Similar to the proof of Lemma F.2, we can prove that

$$\widehat{r}_i \geq \frac{r_i}{2} \ \ \forall i \in \mathcal{S}, \qquad\qquad \widehat{r}_i \lesssim 1 \ \ \forall i,$$

with probability at least $1-\delta$.

When $\mathbf{\Sigma} = \mathbf{I}$, by Lemma F.4, we have that

$$\left\|\widehat{\mathbf{R}}\widehat{\mathbf{\Sigma}}\widehat{\mathbf{R}} - \mathbf{R}\mathbf{\Sigma}\mathbf{R}\right\|_2 \lesssim \frac{\beta^2}{t}$$

holds with probability at least $1-\delta$, when $n \gtrsim \frac{t^2\|\mathbf{w}^\star\|_1^2 d^{2/3}}{\beta^4}$. As a result, $\mathbf{R}\mathbf{\Sigma}\mathbf{R} - \frac{\beta^2}{t}\cdot\mathbf{I} \preceq \widehat{\mathbf{R}}\widehat{\mathbf{\Sigma}}\widehat{\mathbf{R}}$. Hereafter, we condition on the above events. For the bias term, we have

$$\begin{aligned}
\left\|\widehat{\mathbf{R}}\left(\mathbf{I} - \eta\widehat{\mathbf{\Sigma}}\right)^t \overline{\mathbf{R}}\mathbf{w}^\star\right\|_2^2 &\leq \left\|\widehat{\mathbf{R}}\right\|_2^2 \cdot \left\|\left(\mathbf{I} - \eta\widehat{\mathbf{\Sigma}}\right)^t \overline{\mathbf{R}}\mathbf{w}^\star\right\|_2^2 \\
&\leq \mathbf{w}^{\star\top}\overline{\mathbf{R}}\left(\mathbf{I} - \eta\widehat{\mathbf{\Sigma}}\right)^{2t}\overline{\mathbf{R}}\mathbf{w}^\star \\
&\lesssim \mathbf{w}^{\star\top}\overline{\mathbf{R}}\left(\mathbf{I} - \eta\left(\mathbf{R}\mathbf{\Sigma}\mathbf{R} - \frac{\beta^2}{t}\cdot\mathbf{I}\right)\right)^{2t}\overline{\mathbf{R}}\mathbf{w}^\star \\
&= \sum_{i\in\mathcal{S}} (w_i^\star/\widehat{r}_i)^2 \cdot \left(1 - \eta\left((w_i^\star)^2 - \frac{\beta^2}{t}\right)\right)^{2t} \\
&\leq s \cdot \left(1 - \eta\beta^2/2\right)^{2t},
\end{aligned}$$

where the last line is by the definition of β. When $t \gtrsim \log\left(\frac{\sigma^2}{ns}\right)/\left(2\log\left(1-\eta\beta^2/2\right)\right)$, we have

$$\text{Bias} = \left\|\widehat{\mathbf{R}}\left(\mathbf{I} - \eta\widehat{\mathbf{\Sigma}}\right)^t \overline{\mathbf{R}}\mathbf{w}^\star\right\|_2^2 \leq \frac{\sigma^2}{n}. \tag{H.1}$$

When $\eta\beta^2/2 \leq 1/2$, there exist a $c > 0$, such that

$$\log\left(1-\eta\beta^2/2\right) \geq c\eta\beta^2/2.$$

Hence, the variance term is bounded by

$$\begin{aligned}
\text{Variance} &\lesssim \eta t \log t \cdot \left(\frac{\sigma^2 \log\left(d/\delta\right)}{n} + \frac{\sigma^2\|\mathbf{w}^\star\|_1^2 d \log^2\left(d/\delta\right)}{n^2}\right) \\
&\lesssim \frac{\sigma^2 \log^2\left(ns/\sigma^2\right)\log^2\left(d/\delta\right)}{\beta^2} \cdot \left(\frac{s}{n} + \frac{ds}{n^2}\right),
\end{aligned} \tag{H.2}$$

where the last line is by $\|\mathbf{w}^\star\|_1 \le s \cdot \|\mathbf{w}^\star\|_2^2 = s$ and $\eta t \lesssim \frac{\log\left(ns/\sigma^2\right)}{\beta^2}$. Combining Eq.(H.1) and Eq.(H.2), we have that

$$\mathcal{E}(\widehat{\mathbf{w}}_t) \lesssim \frac{\sigma^2}{n} + \frac{\sigma^2 \log^2\left(ns/\sigma^2\right)\log^2\left(d/\delta\right)}{\beta^2} \cdot \left(\frac{1}{n} + \frac{ds}{n^2}\right) \lesssim \frac{\sigma^2 \log^2\left(ns/\sigma^2\right)\log^2\left(d/\delta\right)}{\beta^2} \cdot \left(\frac{1}{n} + \frac{ds}{n^2}\right),$$

when $n \gtrsim \frac{t^2 s d^{2/3}}{\beta^4} \ge \frac{t^2 \|\mathbf{w}^\star\|_1^2 d^{2/3}}{\beta^4}$ and $t \gtrsim \frac{\log(ns)}{\eta\beta^2}$. When $w_i^\star \in \mathsf{U}\{-1/\sqrt{s}, 1/\sqrt{s}\}$, $\beta = 1/\sqrt{s}$. In this case, we have that

$$\mathcal{E}(\widehat{\mathbf{w}}_t) \lesssim \sigma^2 \log^2\left(ns/\sigma^2\right)\log^2\left(d/\delta\right) \cdot \left(\frac{s}{n} + \frac{ds^2}{n^2}\right),$$

when $n \gtrsim t^2 s^3 d^{2/3}$ and $t \gtrsim \frac{\log(ns)}{\eta s}$.

H.2. Lower bound for Ridge Regression

When $n \gtrsim d + \log(1/\delta)$, by Lemma F.3, we have that $\frac{1}{2}\cdot\mathbf{I} \preceq \widehat{\mathbf{\Sigma}} \preceq 2\cdot\mathbf{I}$ For the ridge estimator $\widehat{\mathbf{w}}_\lambda = \frac{1}{n}\cdot\left(\widehat{\mathbf{\Sigma}} + \lambda\cdot\mathbf{I}\right)^{-1}\mathbf{X}^\top\mathbf{y}$, we have

$$\begin{aligned}
\mathbb{E}_{\mathbf{w}^\star}[\mathcal{E}(\widehat{\mathbf{w}}_\lambda)] &= \left\|\left(\mathbf{I} - \left(\widehat{\mathbf{\Sigma}} + \lambda\mathbf{I}\right)^{-1}\widehat{\mathbf{\Sigma}}\right)\mathbf{w}^\star\right\|_2^2 + \left\|\frac{1}{n}\cdot\left(\widehat{\mathbf{\Sigma}} + \lambda\cdot\mathbf{I}\right)^{-1}\mathbf{X}^\top\epsilon\right\|_2^2 \\
&\ge \left\|\frac{1}{n}\cdot\left(\widehat{\mathbf{\Sigma}} + \lambda\cdot\mathbf{I}\right)^{-1}\mathbf{X}^\top\epsilon\right\|_2^2.
\end{aligned}$$

By Lemma F.3, when $\frac{1}{2}\cdot\mathbf{I} \preceq \widehat{\mathbf{\Sigma}} \preceq 2\cdot\mathbf{I}$, with probability at least $1-\delta$, we have

$$\begin{aligned}
\mathbb{E}_{\mathbf{w}^\star}[\mathcal{E}(\widehat{\mathbf{w}}_\lambda)] &\ge \left\|\frac{1}{n}\cdot\left(\widehat{\mathbf{\Sigma}} + \lambda\cdot\mathbf{I}\right)^{-1}\mathbf{X}^\top\epsilon\right\|_2^2 \\
&= \frac{1}{n^2}\cdot\epsilon^\top\mathbf{X}\left(\widehat{\mathbf{\Sigma}} + \lambda\mathbf{I}\right)^{-2}\mathbf{X}^\top\epsilon \\
&\ge \frac{1}{n^2(2+\lambda)^2}\cdot\epsilon^\top\mathbf{X}\mathbf{X}^\top\epsilon,
\end{aligned}$$

where the last line is due to the fact that $\widehat{\mathbf{\Sigma}} + \lambda\mathbf{I} \preceq (2+\lambda)\cdot\mathbf{I}$.

Lemma H.1. *Given X such that $\frac{1}{2}\mathbf{I} \preceq \widehat{\mathbf{\Sigma}} \preceq 2\mathbf{I}$, it holds that*

$$\left\|\frac{1}{n}\mathbf{X}^\top\epsilon\right\|_2^2 \gtrsim \frac{\sigma^2 d}{n},$$

with probability at least $1-\delta$, when $n \ge \mathcal{O}(\log(1/\delta))$.

Proof of Lemma H.1. We consider the singular value decomposition of $\frac{1}{\sqrt{n}}\mathbf{X}^\top$: $\frac{1}{\sqrt{n}}\mathbf{X}^\top = \mathbf{U}\mathbf{\Lambda}\mathbf{V}^\top$, where $\mathbf{U} \in \mathbb{R}^{d\times d}$ is an orthogonal matrix, $\mathbf{\Lambda} \in \mathbb{R}^{d\times n}$ is a rectangular diagonal matrix with non-negative real numbers on the diagonal, $\mathbf{V} \in \mathbb{R}^{n\times n}$ is an orthogonal matrix. Let $\{\sigma_1, \ldots, \sigma_d\}$ be the singular values of $\frac{1}{\sqrt{n}}\mathbf{X}^\top$. Then we have

$$\begin{aligned}
\left\|\frac{1}{n}\mathbf{X}^\top\epsilon\right\|_2^2 &= \left\|\frac{1}{\sqrt{n}}\mathbf{U}\mathbf{\Lambda}\mathbf{V}^\top\epsilon\right\|_2^2 = \left\|\frac{1}{\sqrt{n}}\mathbf{\Lambda}\mathbf{V}^\top\epsilon\right\|_2^2 \\
&= \left\|\frac{1}{\sqrt{n}}\mathbf{\Lambda}\widetilde{\epsilon}\right\|_2^2 = \frac{1}{n}\sum_{i=1}^d \sigma_i^2\widetilde{\epsilon}_i^2,
\end{aligned}$$

where $\widetilde{\epsilon} = \mathbf{V}^\top \epsilon \sim \mathsf{N}(\mathbf{0}, \mathbf{I})$. By [Lemma 22], we have

$$\left| \left\| \frac{1}{n} \mathbf{X}^\top \epsilon \right\|_2^2 - \mathbb{E}\left[\left\| \frac{1}{n} \mathbf{X}^\top \epsilon \right\|_2^2 \right] \right| \lesssim \sigma^2 \max\left\{ \frac{\sqrt{\sum_{i=1}^d \sigma_i^4 \log(1/\delta)}}{n}, \frac{\max_i \sigma_i^2 \log(1/\delta)}{n} \right\}$$

$$\lesssim \sigma^2 \max\left\{ \frac{\sqrt{d \log(1/\delta)}}{n}, \frac{\log(1/\delta)}{n} \right\}, \tag{H.3}$$

where the last line is valid since $\{\sigma_1^2, \ldots, \sigma_d^2\}$ is the eigenvalues of $\widehat{\mathbf{\Sigma}} = \frac{1}{n}\mathbf{X}^\top\mathbf{X}$ and $\frac{1}{2}\mathbf{I} \preceq \widehat{\mathbf{\Sigma}} \preceq 2\mathbf{I}$. By Eq.(H.3), we obtain that

$$\begin{aligned} \left\| \frac{1}{n} \mathbf{X}^\top \epsilon \right\|_2^2 &\geq \mathbb{E}\left[\left\| \frac{1}{n} \mathbf{X}^\top \epsilon \right\|_2^2 \right] - \sigma^2 \max\left\{ \frac{\sqrt{d \log(1/\delta)}}{n}, \frac{\log(1/\delta)}{n} \right\} \\ &= \sigma^2 \sum_{i=1}^d \sigma_i^2 - \sigma^2 \max\left\{ \frac{\sqrt{d \log(1/\delta)}}{n}, \frac{\log(1/\delta)}{n} \right\} \\ &= \sigma^2 \frac{d}{n} - \sigma^2 \max\left\{ \frac{\sqrt{d \log(1/\delta)}}{n}, \frac{\log(1/\delta)}{n} \right\} && (\text{by } \tfrac{1}{2}\mathbf{I} \preceq \widehat{\mathbf{\Sigma}} \preceq 2\mathbf{I}) \\ &\lesssim \sigma^2 \frac{d}{n}, \end{aligned}$$

where the last line is due to $d \geq \mathcal{O}(\log(1/\delta))$. □

Next, we define the event $\mathcal{E}$ as follows:

$$\mathcal{E}_{\text{ridge}} := \left\{ \frac{1}{2}\mathbf{I} \preceq \widehat{\mathbf{\Sigma}} \preceq 2\mathbf{I}, \left\| \frac{1}{n} \mathbf{X}^\top \epsilon \right\|_2^2 \gtrsim \frac{\sigma^2 d}{n} \right\}.$$

By Lemma H.1, we have $\mathbb{P}(\mathcal{E}) \geq 1 - \delta$ when $n \geq \mathcal{O}(d) \geq \mathcal{O}(\log(1/\delta))$. On $\mathcal{E}_{\text{ridge}}$, we have

$$\mathbb{E}_{\mathbf{w}^\star}[\mathcal{E}(\widehat{\mathbf{w}}_\lambda)] \gtrsim \frac{\sigma^2 d}{(1+\lambda)^2 n}. \tag{H.4}$$

When $d \gtrsim n + \log(1/\delta)$, by Lemma F.3, with probability at least $1 - \delta$, we have that $\frac{d}{2} \cdot \mathbf{I} \preceq \mathbf{X}\mathbf{X}^\top \preceq 2d \cdot \mathbf{I}$. Hereafter, we condition on this event. By direct calculation, we can decompose the excess risk by

$$\mathbb{E}_{\mathbf{w}^\star}[\mathcal{E}(\widehat{\mathbf{w}}_\lambda)] = \mathbb{E}_{\mathbf{w}^\star} \left\| \left(\mathbf{I} - \left(\widehat{\mathbf{\Sigma}} + \lambda \mathbf{I} \right)^{-1} \widehat{\mathbf{\Sigma}} \right) \mathbf{w}^\star \right\|_2^2 + \left\| \frac{1}{n} \cdot \left(\widehat{\mathbf{\Sigma}} + \lambda \cdot \mathbf{I} \right)^{-1} \mathbf{X}^\top \epsilon \right\|_2^2.$$

For the first term, we have

$$\begin{aligned} \mathbb{E}_{\mathbf{w}^\star} \left\| \left(\mathbf{I} - \left(\widehat{\mathbf{\Sigma}} + \lambda \mathbf{I} \right)^{-1} \widehat{\mathbf{\Sigma}} \right) \mathbf{w}^\star \right\|_2^2 &= \mathbb{E}_{\mathbf{w}^\star} \left\| \left(\mathbf{I} - \mathbf{X}^\top \left(\mathbf{X}\mathbf{X}^\top + n\lambda \mathbf{I} \right)^{-1} \mathbf{X} \right) \mathbf{w}^\star \right\|_2^2 \\ &= (1 - \frac{n}{d}) \mathbb{E}_{\mathbf{w}^\star} \left[\|\mathbf{w}^\star\|_2^2 \right], && \text{(H.5)} \\ &= 1 - \frac{n}{d} && \text{(H.6)} \end{aligned}$$

where the last line is due to $\left(\mathbf{I} - \mathbf{X}^\top \left(\mathbf{X}\mathbf{X}^\top + n\lambda \mathbf{I} \right)^{-1} \mathbf{X} \right)$ is a $d - n$ space.

$$\begin{aligned} \left\| \frac{1}{n} \cdot \left(\widehat{\mathbf{\Sigma}} + \lambda \cdot \mathbf{I} \right)^{-1} \mathbf{X}^\top \epsilon \right\|_2^2 &= \epsilon^\top \mathbf{X}\mathbf{X}^\top \left(\mathbf{X}\mathbf{X}^\top + n\lambda \mathbf{I} \right)^{-2} \epsilon \\ &\geq \frac{dn}{2(2d + n\lambda)^2} \cdot \frac{1}{n} \sum_{i=1}^n \epsilon_i^2, && \text{(H.7)} \end{aligned}$$

where the first line is by $\left(\mathbf{X}^\top\mathbf{X}+n\lambda\mathbf{I}\right)^{-1}\mathbf{X}^\top = \mathbf{X}^\top\left(\mathbf{X}\mathbf{X}^\top+n\lambda\mathbf{I}\right)^{-1}$ and the last line is by $\frac{d}{2(d+n\lambda)^2}\cdot\mathbf{I}\preceq\mathbf{X}\mathbf{X}^\top\left(\mathbf{X}\mathbf{X}^\top+n\lambda\mathbf{I}\right)^{-2}$. By Tsigler & Bartlett (2023, Lemma 22), we obatain that

$$\left|\sum_{i=1}^n \epsilon_i^2 - n\sigma^2\right| \lesssim \sigma^2\sqrt{n\log(1/\delta)}+\sigma^2$$

holds with probability at least $1-\delta$. When $n\gtrsim\log(1/\delta)$, we have $\left|\sum_{i=1}^n\epsilon_i^2-n\sigma^2\right|\geq\frac{n\sigma^2}{2}$ holds with probability at least $1-\delta$. Taking the union bound, we obtain that

$$\mathbb{E}_{\mathbf{w}^\star}[\mathcal{E}(\widehat{\mathbf{w}}_\lambda)]\gtrsim 1-\frac{n}{d}+\sigma^2\cdot\frac{dn}{2(2d+n\lambda)^2}\gtrsim 1-\frac{n}{d}+\sigma^2\frac{n}{(1+\lambda)^2d}. \tag{H.8}$$

H.3. Lower Bound for Finite-Step GD

We first consider the case where $n\gtrsim d+\log(1/\delta)$. Define the event $\mathcal{E}_{\rm GD}$ by $\mathcal{E}_{\rm GD}=\left\{\frac{1}{2}\cdot\mathbf{I}\preceq\widehat{\boldsymbol{\Sigma}}\preceq 2\mathbf{I}\right\}$. By Lemma F.3, $\mathbb{P}(\mathcal{E}_{\rm GD})\geq 1-\delta$. By Eq.(G.4), we have

$$\begin{aligned}
\mathbb{E}_{\mathbf{w}^\star}[\mathcal{E}(\mathbf{w}_t)] &= \mathbb{E}_{\mathbf{w}^\star}\left\|\left(\mathbf{I}-\eta\widehat{\boldsymbol{\Sigma}}\right)^t\mathbf{w}^\star\right\|_2^2+\eta^2\left\|\left(\frac{1}{n}\sum_{i=1}^t\left(\mathbf{I}-\eta\widehat{\boldsymbol{\Sigma}}\right)^{i-1}\mathbf{X}^\top\epsilon\right)\right\|_2^2\\
&\geq \eta\left\|\left(\frac{1}{n}\sum_{i=1}^t\left(\mathbf{I}-\eta\widehat{\boldsymbol{\Sigma}}\right)^{i-1}\mathbf{X}^\top\epsilon\right)\right\|_2^2\\
&= \frac{\eta^2}{n^2}\cdot\left\|\left(\widehat{\boldsymbol{\Sigma}}\left(\mathbf{I}-\left(\mathbf{I}-\eta\widehat{\boldsymbol{\Sigma}}\right)^t\right)^{-1}\right)^{-1}\mathbf{X}^\top\epsilon\right\|_2^2\\
&\gtrsim \frac{\eta^2}{n^2}\cdot\left\|\left(\widehat{\boldsymbol{\Sigma}}+\frac{1}{\eta t}\cdot\mathbf{I}\right)^{-1}\mathbf{X}^\top\epsilon\right\|_2^2\\
&\gtrsim \sigma^2\frac{\eta^2 d}{\left(1+1/(\eta t)\right)^2 n},
\end{aligned}$$

where the second last line is by $\widehat{\boldsymbol{\Sigma}}\left(\mathbf{I}-\left(\mathbf{I}-\eta\widehat{\boldsymbol{\Sigma}}\right)^t\right)^{-1}\preceq\boldsymbol{\Sigma}+\frac{2}{t\eta}\cdot\mathbf{I}$ and the last line is by Eq.(H.4).

We then consider the case where $d\gtrsim n+\log(1/\delta)$. Define the event $\mathcal{E}'_{\rm GD}=\left\{\frac{d}{2}\cdot\mathbf{I}\preceq\mathbf{X}\mathbf{X}^\top\prec 2d\mathbf{I}\right\}$. By Lemma F.3, $\mathbb{P}(\mathcal{E}'_{\rm GD})\geq 1-\delta$. Following the proof of Zou et al. (2022, Theorem 4.3), we have

$$\begin{aligned}
\mathbb{E}_{\mathbf{w}^\star}[\mathcal{E}(\mathbf{w}_t)] &\geq \mathbb{E}_{\mathbf{w}^\star}\left\|\left(\mathbf{I}-\mathbf{X}^\top\left(\mathbf{X}\mathbf{X}^\top+\frac{n}{\eta t}\mathbf{I}\right)^{-1}\mathbf{X}\right)\right\|_2^2+\left\|\frac{1}{n}\mathbf{X}^\top\left(\mathbf{X}\mathbf{X}^\top+\frac{n}{\eta t}\mathbf{I}\right)^{-1}\epsilon\right\|_2^2\\
&= 1-\frac{n}{d}+\frac{\sigma^2 n}{\left(1+\frac{1}{\eta t}\right)^2 d},
\end{aligned}$$

where we use the results from Appendix H.2.

H.4. Lower bound of OLS

Let $\mathbf{w}_{\rm ols}$ be the OLS estimator. It is easy to see $\mathbf{w}_{\rm ols}=\mathbf{w}_0$. Hence, we have

$$\mathbb{E}_{\mathbf{w}^\star}[\mathcal{E}(\mathbf{w}_{\rm ols})]\gtrsim\begin{cases}\frac{\sigma^2 d}{n} & n\gtrsim d+\log(1/\delta)\\ 1-\frac{n}{d}+\frac{\sigma^2 n}{d} & d\gtrsim n+\log(1/\delta),\end{cases}$$

holds with probability at least $1-\delta$.

I. Additional Experiments

Here, we provide additional experiments on the decoder-only architecture and train models with $s = d = 16$.

Training Decoder-Only Transformer In this experiment, we adapt the same input setting and training objective as in (Garg et al., 2023). During training, we set $n = 24$ and $k = 8$ in Eq.(I.1) (where in y_i, we use zero padding to align with $\mathbf{x}_i$), $d_{\mathsf{hid}} = 256$. We choose $h = 8$ and $l \in \{4, 5, 6\}$.[2] We then conduct heads assessment experiments on the trained decoder-only transformers with 10 in-context examples, as in the previous settings. The result is shown in Figure 9. We can observe that the decoder-only transformer exhibits the similar weight distribution for each layer as the encoder-based models, indicating that our algorithm may extend to decoder-only based models.

$$\mathbf{E} = \begin{pmatrix} \mathbf{x}_1 & y_1 & \mathbf{x}_2 & y_2 & \dots & \mathbf{x}_n & y_n \end{pmatrix}, \quad L = \sum_{i=k}^{n} (\widehat{y}_i - y_i)^2. \tag{I.1}$$

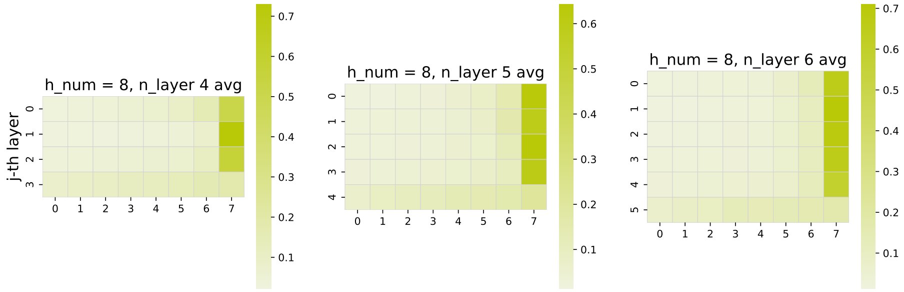


Figure 9: Heads Assessment for decoder-only transformers

Training Models with $s = d = 16$ Here, we adapt the encoder-only transformer and the same settings as introduced in A, but set $s = d = 16$. We observe that in these cases, there is no distinct performance difference between models with different numbers of heads. As shown in Figure 3, when we set $s = 4, d = 16$, transformers with more heads ($h = 4, 8$) always perform better than models with fewer heads ($h = 1, 2$). However, in Figure 10, such a difference is unclear, which aligns well with the theoretical analysis. When s is close to d, a clear better upper bound guarantee, as ensured in cases where $s \ll d$ may not hold.

[2] We also tried other settings with fewer heads or layers, but even with delicate hyperparameter tuning, decoder-only transformers with fewer heads or layers consistently failed to learn how to solve our sparse linear regression problem. A possible reason is that decoder-only transformers first need to learn the causal structure (Nichani et al., 2024) and then apply an optimization algorithm to the in-context entries, which is more challenging than our encoder-based settings.

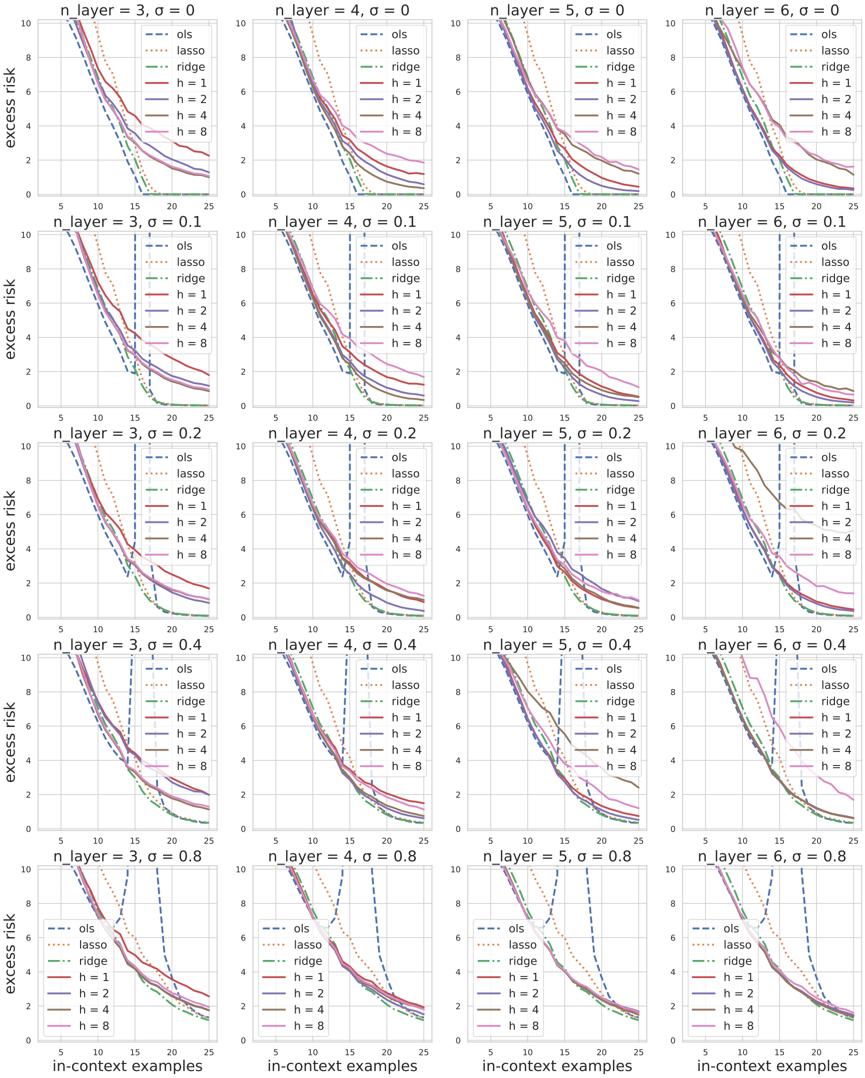


Figure 10: Train Models with $s = d = 16$