TRANSFORMERS HANDLE ENDOGENEITY IN IN-CONTEXT LINEAR REGRESSION

Anonymous authors
Paper under double-blind review

ABSTRACT

We explore the capability of transformers to address endogeneity in in-context linear regression. Our main finding is that transformers inherently possess a mechanism to handle endogeneity effectively using instrumental variables (IV). First, we demonstrate that the transformer architecture can emulate a gradient-based bi-level optimization procedure that converges to the widely used two-stage least squares (2SLS) solution at an exponential rate. Next, we propose an in-context pretraining scheme and provide theoretical guarantees showing that the global minimizer of the pre-training loss achieves a small excess loss. Our extensive experiments validate these theoretical findings, showing that the trained transformer provides more robust and reliable in-context predictions and coefficient estimates than the 2SLS method, in the presence of endogeneity.

1 INTRODUCTION

The transformer architecture (Vaswani et al., 2017) has demonstrated remarkable in-context learning (ICL) capabilities across various domains, such as natural language processing (Devlin et al., 2019; Radford et al., 2019; Brown et al., 2020), computer vision (Dosovitskiy et al., 2021; Carion et al., 2020), and reinforcement learning (Lee et al., 2022; Parisotto et al., 2019). Self-attention mechanism, a core component of transformers, allows these models to capture long-range dependencies in data, which is critical for success in these tasks. Despite their impressive performance, the theoretical understanding of transformers remains limited, leaving important questions unanswered about their true capabilities and the underlying mechanisms driving their exceptional results.

Recent efforts to understand transformers' ICL capabilities have focused on the performance in fundamental statistical tasks. Focusing on simple function classes, Garg et al. (2022) highlighted that transformers, when trained on sufficiently large and diverse data from a specific function class, can generalize across most functions of that class without task-specific fine-tuning. Building on this, subsequent work by Bai et al. (2024) established that attention layers enable transformers to perform gradient descent, implementing algorithms like linear regression, logistic regression, and LASSO; see also Akyürek et al. (2023); Von Oswald et al. (2023); Li et al. (2023); Fu et al. (2023); Ahn et al. (2024).The learning dynamics of transformer trained via gradient descent for in-context learning linear function classes was analyzed in Huang et al. (2024). Furthermore Zhang et al. (2024a;b) showed that *trained* transformers' ICL abilities for linear regression tasks are theoretically robust under distributional shifts and characterized the corresponding sample complexities.

Existing works on analyzing the ICL ability of transformers for linear regression tasks, however, ignore *endogeneity* and have mainly focused on the *exogenous* setup where the additive noise is uncorrelated with the explanatory variables. Ignoring *endogeneity* in linear regression leads to biased and inconsistent estimates, resulting from issues like omitted variable bias, simultaneity, and measurement error, which can distort causal inferences and lead to incorrect policy conclusions (Hausman, 2001; Wooldridge, 2015; Angrist & Pischke, 2009; Greene, 2018). Instrumental variable (IV) regression is a widely adopted method to handle endogeneity by utilizing instruments that are correlated with the endogenous variables but uncorrelated with the error term (Angrist & Krueger, 2001). A naturally intriguing question that therefore arises is:

Can transformers leverage instrumental variables and provide reliable predictions and coefficient estimates, in the presence of endogeneity?

In this work, we aim to answer this question and offer new insights on in-context linear regression tasks. Our key contributions include:

- We demonstrate that looped transformers can address endogeneity in linear regression by leveraging instrumental variables. Specifically, we show that transformers can implement two-stage least squares (2SLS) regression through a bi-level gradient descent procedure, where each iteration is executed by a two-layer transformer block. Moreover, the convergence rate to the 2SLS estimator is exponential with respect to the number of blocks.
- We propose an ICL training scheme for transformers to efficiently handle endogeneity. Under this scheme, we show that the global minimizer of the in-context pre-training loss achieves a small excess loss compared to the global optimal expected loss.
- We evaluate the performance of the trained transformer model through extensive experiments, finding that it not only matches the performance of the 2SLS estimator on standard IV tasks but also generalizes effectively to more complex scenarios, including the challenging cases of weak instruments, non-linear IV, and underdetermined IV problems.
- As part of our analysis, we derive the first non-asymptotic bound for the 2SLS estimator under random design, providing valuable insights for future theoretical work.

1.1 Related works

In-context Learning. Initial works by Garg et al. (2022) and Bai et al. (2024) adopted the standard multi-layer transformer architecture to conduct the experiments. Later, Giannou et al. (2023) and Yang et al. (2023) showed that a looped architecture reduces the required depth of transformers and exhibits better efficiency in learning algorithms. Gao et al. (2024) illustrated that the looped transformer architecture with extra pre-processing and post-processing layers can achieve higher expressive power than a standard transformer with the same number of parameters. Apart from works concerning the implementability of first-order gradient descent algorithms by transformers, other works have also examined higher-order and non-parametric optimization methods. Specifically, Giannou et al. (2024) showed that transformers can emulate Newton's method for logistic regression. Cheng et al. (2024) showed that transformers can implement functional gradient descent and hence enable them to learn non-linear functions in-context. Relationship between in-context learning and Bayesian inference is also studied in Ye et al. (2024); Falck et al. (2024).

Nichani et al. (2024) illustrated how the transformers can learn the causal structure by encoding the latent causal graph in the first attention layer. Goel & Bartlett (2024) explored the representational power of transformer for learning linear dynamical systems. Makkuva et al. (2024a;b); Rajaraman et al. (2024); Edelman et al. (2024) considered ICL Markov chains with transformers, including both landscape and training dynamics analyses. To the best of our knowledge, we are not aware of prior works on handling endogeniety with transformers.

Instrumental Variable Regression. IV regression has been widely studied in econometrics (Angrist & Krueger, 2001; Angrist & Pischke, 2009). Recent works in machine learning explored the optimization based approaches for the IV regression problem. Singh et al. (2019) proposed the kernel IV regression to model non-linear relationship between variables. Muandet et al. (2020) proposed that a non-linear IV regression problem can be formulated as a convex-concave saddle point problem. Della Vecchia & Basu (2023); Chen et al. (2024); Peixoto et al. (2024) proposed a stochastic optimization algorithm for IV regression.

Notation: Throughout this paper, unless otherwise specified, lower-case letters denote random variables or samples, while upper-case letters represent datasets (collections of samples). Bolded letters indicate vectors or matrices, whereas unbolded letters indicate scalars. The notation $\boldsymbol{X}_{:,i}$ refers to the i-th column, and $\boldsymbol{X}_{i,:}$ refers to the i-th row of matrix $\boldsymbol{X}$. By default, $\|\cdot\|$ denotes the Euclidean norm for a vector, or the spectral norm for a matrix.

2 Endogeneity and Instrumental Variable Regression

Suppose we are interested in estimating the relationship between response variable $y \in \mathbb{R}$ and predictor variable $\boldsymbol{x} \in \mathbb{R}^p$ with endogeneity. Given instruments $\boldsymbol{z} \in \mathbb{R}^q$, we consider the model

$$y = \boldsymbol{\beta}^\top \boldsymbol{x} + \epsilon_1, \quad \text{and} \quad \boldsymbol{x} = \boldsymbol{\Theta}^\top \boldsymbol{z} + \boldsymbol{\epsilon_2}, \tag{1}$$

where $\boldsymbol{\beta} \in \mathbb{R}^p$, and $\boldsymbol{\Theta} \in \mathbb{R}^{q \times p}$ are the true model parameters, $\epsilon_1 \in \mathbb{R}$ and $\boldsymbol{\epsilon}_2 \in \mathbb{R}^p$ are (centered) random noise terms with variance σ_1^2 and covariance matrix $\boldsymbol{\Sigma}_2$, respectively. Further, $\boldsymbol{\epsilon}_2$ is an unobserved noise correlated with ϵ_1, leading to the correlation between $\boldsymbol{x}$ and ϵ_1, which introduces confounding in the model between $\boldsymbol{x}$ and y. Under this setting, the standard ordinary least squares (OLS) estimator is a biased and inconsistent estimator of $\boldsymbol{\beta}$ (see Wooldridge (2015), Chapter 9). To address this issue, instrumental variable (IV) regression is a widely used method to provide a consistent estimate for $\boldsymbol{\beta}$.

Definition 2.1 (2SLS estimator)**.** IV regression is a regression model to provide consistent estimate on the causal effect $\boldsymbol{\beta}$ for the endogeneity problem (1), by utilizing the instrument $\boldsymbol{z}$. Given observational values $(\boldsymbol{Z}, \boldsymbol{X}, \boldsymbol{Y}) = \{(\boldsymbol{z}_i, \boldsymbol{x}_i, y_i)\}_{i=1}^n$, the standard approach to estimate the IV regression model is 2SLS; see, for example, Wooldridge (2015), Chapter 15.

i. *First stage*: Regress $\boldsymbol{X}$ on $\boldsymbol{Z}$ to obtain $\hat{\boldsymbol{\Theta}}$

$$\hat{\boldsymbol{\Theta}} = (\boldsymbol{Z}^\top \boldsymbol{Z})^{-1} \boldsymbol{Z}^\top \boldsymbol{X}.$$

ii. *Second stage*: Regress $\boldsymbol{Y}$ on $\boldsymbol{Z}\hat{\boldsymbol{\Theta}}$ to obtain:

$$\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}} = (\hat{\boldsymbol{\Theta}}^\top \boldsymbol{Z}^\top \boldsymbol{Z} \hat{\boldsymbol{\Theta}})^{-1} \hat{\boldsymbol{\Theta}}^\top \boldsymbol{Z}^\top \boldsymbol{Y}. \tag{2}$$

We introduce the standard assumptions required to show the convergence rate of the above estimator.

Assumption 1 (Instrumental variable)**.** A random variable $\boldsymbol{z} \in \mathbb{R}^q$ is a valid IV, if it satisfies the following conditions:

i. Fully identification: $q \geq p$ (without loss of generality, we assume data $\boldsymbol{Z}, \boldsymbol{X}$ are full rank).
ii. Correlated to $\boldsymbol{x}$: $\mathsf{Corr}(\boldsymbol{z}, \boldsymbol{x}) \neq \boldsymbol{0}$.
iii. Conditional uncorrelated to y: $\mathsf{Corr}(\boldsymbol{z}, \epsilon_1) = 0$.

In particular, condition (i) above ensures the existence of unique solution for $\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}$. We refer to Stock & Watson (2011, Chapter 12) for additional elaborate discussions on the above conditions. To derive non-asymptotic convergence rates, we further assume the following regularity conditions.

Assumption 2 (Regularity conditions)**.** We assume the following conditions hold:

i. Bounded parameters: $\|\boldsymbol{\beta}\| \leq B_\beta$, $\|\boldsymbol{\Theta}\| \leq B_\Theta$.

ii. Population condition: $\|\boldsymbol{z}\| \leq B_z$, $\|\boldsymbol{x}\| \leq B_x$, $|\epsilon_1| \leq B_{\epsilon_1}$, almost surely.

iii. Sample condition: There exists constants $\lambda_z, \lambda_{\tilde{x}}, \delta_{z\epsilon_2} > 0$, such that the following event holds with probability at least $1 - \xi$:

$$\mathcal{A} = \left\{ \lambda_{\min}\left(\frac{\boldsymbol{Z}^\top \boldsymbol{Z}}{n}\right) \geq \lambda_z, \lambda_{\min}\left(\frac{\boldsymbol{X}^\top \boldsymbol{P}_Z \boldsymbol{X}}{n}\right) \geq \lambda_{\tilde{x}}, \left\| \frac{\boldsymbol{Z}^\top \boldsymbol{\mathcal{E}}_2}{n} \right\| \leq \delta_{z\epsilon_2} \right\}, \tag{3}$$

where $\lambda_{\min}(\cdot)$ denotes the smallest eigenvalue of a matrix, $\boldsymbol{P}_Z := \boldsymbol{Z}(\boldsymbol{Z}^\top \boldsymbol{Z})^{-1} \boldsymbol{Z}^\top$ denotes the projection matrix.

iv. Without loss of generality, we assume Assumption 1 holds under event $\mathcal{A}$, and $\mathbb{E}[\epsilon_1^2 | \mathcal{A}] = \mathcal{O}(\sigma_1^2)$.

The almost sure boundedness condition in (ii) is required to invoke matrix Bernstein inequalities (Tropp, 2015) in the analysis. We anticipate that this condition may be relaxed to sub-Gaussian or moment conditions by using more sophisticated matrix concentration results. We emphasize that condition (iii) is a mild condition in general, given some common assumptions on the population distributions. See Appendix C.1 for related analysis and empirical justifications. Now we establish the consistency of the 2SLS estimator under random design.

Theorem 2.1 (Non-asymptotic error bound of 2SLS estimator)**.** *Given Assumptions 1 and 2, for any $\varepsilon > 0$, we have:*

$$\mathbb{P}\big(\|\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}} - \boldsymbol{\beta}\| \geq \varepsilon \mid \mathcal{A}\big) \leq (q+1) \exp\left(- \frac{\frac{\lambda_{\tilde{x}}^2 n^2 \varepsilon^2}{2}}{(B_\Theta + \frac{\delta_{z\epsilon_2}}{\lambda_z})^2 \nu(\boldsymbol{\Omega}_{z\epsilon_1} | \mathcal{A}) + \lambda_{\tilde{x}} (B_\Theta + \frac{\delta_{z\epsilon_2}}{\lambda_z}) B_z B_{\epsilon_1} \frac{n\varepsilon}{3}} \right),$$

where $\boldsymbol{\Omega}_{z\epsilon_1} := \sum_{i=1}^n \boldsymbol{z}_i \epsilon_{1,i}$*, and* $\nu(\cdot)$ *is the matrix variance statistic:*

$$\nu(\boldsymbol{\Omega}) := \max\{\left\|\mathbb{E}(\boldsymbol{\Omega}\boldsymbol{\Omega}^\top)\right\|, \left\|\mathbb{E}(\boldsymbol{\Omega}^\top\boldsymbol{\Omega})\right\|\}.$$

Furthermore, consider clipping operation $\mathsf{clip}_{B_\beta}(\hat{\boldsymbol{\beta}}) := \begin{cases} \hat{\boldsymbol{\beta}} & \text{if } \|\hat{\boldsymbol{\beta}}\| \le B_\beta \\ \frac{B_\beta}{\|\hat{\boldsymbol{\beta}}\|}\hat{\boldsymbol{\beta}} & \text{if } \|\hat{\boldsymbol{\beta}}\| > B_\beta \end{cases}$*, then the expectation of the estimation error is bounded by:*

$$\mathbb{E}\left[\|\mathsf{clip}_{B_\beta}(\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}) - \boldsymbol{\beta}\|^2\right] \le \mathcal{O}\left(\frac{C_{z,x}^2 q \sigma_1^2}{n} + \xi B_\beta^2\right), \tag{4}$$

where $C_{z,x} := \frac{(B_\Theta + \frac{\delta_{z\epsilon_2}}{\lambda_z})B_z}{\lambda_{\tilde{x}}}$ *is a constant depending on the population distributions of* $\boldsymbol{z}$ *and* $\boldsymbol{x}$*.*

We note that although the consistency of the 2SLS estimator is a standard result in econometrics, most existing works focus on the asymptotic properties of the estimator. Theorem 2.1 provides the first non-asymptotic bound for estimation error $\|\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}} - \boldsymbol{\beta}\|^2$, under random design. The detailed proof is provided in Appendix A.1.

3 Transformers Handle Endogeniety

3.1 Transformer Architecture

Denote the input matrix as $\boldsymbol{H} = [\boldsymbol{h}_1, \ldots, \boldsymbol{h}_n] \in \mathbb{R}^{D\times n}$, where each column corresponds to one sample vector.

Definition 3.1 (Attention layer)**.** A self-attention layer with M heads is denoted as $\mathsf{ATTN}_{\boldsymbol{\theta}}(\cdot)$, with parameters $\boldsymbol{\theta} = \{(\boldsymbol{Q}_m, \boldsymbol{K}_m, \boldsymbol{V}_m)\}_{m\in[M]} \subseteq \mathbb{R}^{D\times D}$. Given input $\boldsymbol{H}$,

$$\tilde{\boldsymbol{H}} = \mathsf{ATTN}_{\boldsymbol{\theta}}(\boldsymbol{H}) := \boldsymbol{H} + \frac{1}{n}\sum_{m=1}^M (\boldsymbol{V}_m\boldsymbol{H}) \times \sigma((\boldsymbol{Q}_m\boldsymbol{H})^\top(\boldsymbol{K}_m\boldsymbol{H})) \in \mathbb{R}^{D\times n}, \tag{5}$$

or element-wise:

$$\tilde{\boldsymbol{h}}_i = [\mathsf{ATTN}_{\boldsymbol{\theta}}(\boldsymbol{H})]_i := \boldsymbol{h}_i + \sum_{m=1}^M \frac{1}{n}\sum_{j=1}^n \sigma(\langle \boldsymbol{Q}_m\boldsymbol{h}_i, \boldsymbol{K}_m\boldsymbol{h}_j\rangle)\cdot \boldsymbol{V}_m\boldsymbol{h}_j \in \mathbb{R}^D, \tag{6}$$

where $\sigma(\cdot)$ is the ReLU function.

Definition 3.2 (MLP layer)**.** An MLP layer is denoted as $\mathsf{MLP}_{\boldsymbol{\theta}}(\cdot)$, with parameters $\boldsymbol{\theta} = (\boldsymbol{W}_1, \boldsymbol{W}_2) \in \mathbb{R}^{D'\times D\times D\times D'}$. Given input $\boldsymbol{H}$,

$$\tilde{\boldsymbol{H}} = \mathsf{MLP}_{\boldsymbol{\theta}}(\boldsymbol{H}) := \boldsymbol{H} + \boldsymbol{W}_2\sigma(\boldsymbol{W}_1\boldsymbol{H}),$$

or element-wise:

$$\tilde{\boldsymbol{h}}_i = [\mathsf{MLP}_{\boldsymbol{\theta}}(\boldsymbol{H})]_i := \boldsymbol{h}_i + \boldsymbol{W}_2\sigma(\boldsymbol{W}_1\boldsymbol{h}_i).$$

Definition 3.3 (Transformer)**.** An L-layer transformer is denoted as $\mathsf{TF}_{\boldsymbol{\theta}}(\cdot)$, with parameters $\boldsymbol{\theta} = (\boldsymbol{\theta}_{\mathsf{ATTN}}^{(1:L)}, \boldsymbol{\theta}_{\mathsf{MLP}}^{(1:L)})$. Given input $\boldsymbol{H} = \boldsymbol{H}^{(0)}$,

$$\boldsymbol{H}^{(l)} = \mathsf{MLP}_{\boldsymbol{\theta}_{\mathsf{MLP}}^{(l)}}(\mathsf{ATTN}_{\boldsymbol{\theta}_{\mathsf{ATTN}}^{(l)}}(\boldsymbol{H}^{(l-1)})), \quad l = 1, \ldots, L.$$

The output of this transformer is the final layer output: $\tilde{\boldsymbol{H}} := \boldsymbol{H}^{(L)} = \mathsf{TF}_{\boldsymbol{\theta}}(\boldsymbol{H}^{(0)})$.

Definition 3.4 (Looped transformer)**.** An $\bar{L}$-looped transformer is a special transformer architecture, denoted as $\mathsf{LTF}_{\bar{\boldsymbol{\theta}},\bar{L}}(\cdot)$, with parameters $\bar{\boldsymbol{\theta}} = (\bar{\boldsymbol{\theta}}_{\mathsf{ATTN}}^{(1:L_0)}, \bar{\boldsymbol{\theta}}_{\mathsf{MLP}}^{(1:L_0)})$. Given input $\boldsymbol{H} = \boldsymbol{H}^{(0)}$,

$$\boldsymbol{H}^{(l)} = \mathsf{TF}_{\bar{\boldsymbol{\theta}}}(\boldsymbol{H}^{(l-1)}), \quad l = 1, \ldots, \bar{L}.$$

The output of this looped transformer is the final loop output: $\tilde{\boldsymbol{H}} := \boldsymbol{H}^{(\bar{L})} = \mathsf{LTF}_{\bar{\boldsymbol{\theta}},\bar{L}}(\boldsymbol{H}^{(0)})$.

Previous works (e.g., Bai et al. (2024), Zhang et al. (2024a)) have shown that transformers can perform in-context linear regression by emulating gradient descent (GD) with in-context pretraining. However, these studies have two key limitations. First, their analysis is based on single-level optimization algorithms, which is insufficient to demonstrate that transformers can efficiently learn more complex algorithms like 2SLS (Definition 2.1). Second, most ICL-related research focuses on the predictive performance of transformers, paying little attention to their ability to provide accurate coefficient estimates. We extend the current ICL framework by showing that transformers can implement a bi-level GD procedure (see Section 3.2) with looped transformer architecture (Definition 3.4), allowing them to efficiently emulate 2SLS and provide coefficient estimates that are at least as accurate as 2SLS in the presence of endogeneity (as in Equation (1)).

3.2 Gradient descent based IV regression

We first introduce a gradient-based bi-level optimization procedure to obtain the 2SLS estimator in Equation (2). Given the dataset $(\boldsymbol{Z}, \boldsymbol{X}, \boldsymbol{Y}) = \{(\boldsymbol{z}_i, \boldsymbol{x}_i, y_i)\}_{i=1}^n$, the objective funtion of IV regression can be formulated as the following bi-level optimization problem:

$$\min_{\boldsymbol{\beta}} \quad \mathcal{L}(\boldsymbol{\beta}) = \frac{1}{n}\sum_{i=1}^n (y_i - \boldsymbol{z}_i^\top \hat{\boldsymbol{\Theta}}\boldsymbol{\beta})^2, \quad \text{where} \quad \hat{\boldsymbol{\Theta}} := \arg\min_{\boldsymbol{\Theta}} \quad \frac{1}{n}\sum_{j=1}^n (\boldsymbol{x}_j - \boldsymbol{z}_j^\top \boldsymbol{\Theta})^2. \tag{7}$$

Consider the following gradient updates with learning rates α, η:

$$\boldsymbol{\Theta}^{(t+1)} = \boldsymbol{\Theta}^{(t)} - \eta \boldsymbol{Z}^\top(\boldsymbol{Z}\boldsymbol{\Theta}^{(t)} - \boldsymbol{X}), \tag{8a}$$

$$\boldsymbol{\beta}^{(t+1)} = \boldsymbol{\beta}^{(t)} - \alpha \boldsymbol{\Theta}^{(t)\top}\boldsymbol{Z}^\top(\boldsymbol{Z}\boldsymbol{\Theta}^{(t)}\boldsymbol{\beta}^{(t)} - \boldsymbol{Y}). \tag{8b}$$

Note that the GD-2SLS updates in Equation (8) are designed to solve Equation (7). We now show that regardless the convergence of $\boldsymbol{\Theta}^{(t)}$, the GD estimator $\boldsymbol{\beta}^{(t)}$ will always converge to the 2SLS estimator in Equation (2) with exponential rate.

Theorem 3.1 (Implementing 2SLS with gradient-based method)**.** *Given training data* $(\boldsymbol{Z}, \boldsymbol{X}, \boldsymbol{Y}) = \{(\boldsymbol{z}_i, \boldsymbol{x}_i, y_i)\}_{i=1}^n$. *Suppose the learning rates* α, η *satisfy the following conditions:*

$$0 < \alpha < \frac{2}{\sigma_{\max}^2(\boldsymbol{Z}\hat{\boldsymbol{\Theta}})} \quad \textit{and} \quad 0 < \eta < \frac{2}{\sigma_{\max}^2(\boldsymbol{Z})},$$

where $\sigma_{\max}(\cdot)$ *denotes the largest singular value of a matrix. Then, the GD updates in Equation (8) converge to the 2SLS estimator at an exponential rate:*

$$\|\boldsymbol{\beta}^{(t)} - \hat{\boldsymbol{\beta}}_{\textit{2SLS}}\| \le \mathcal{O}\left(\Lambda^t\right),$$

where, with $\rho(\cdot)$ *denoting the spectral radius of the matrix,*

$$\Lambda := \max\{\gamma(\alpha), \kappa(\eta)\}, \qquad \gamma(\alpha) := \rho(\boldsymbol{I} - \alpha\hat{\boldsymbol{\Theta}}^\top\boldsymbol{Z}^\top\boldsymbol{Z}\hat{\boldsymbol{\Theta}}), \quad \kappa(\eta) := \rho(\boldsymbol{I} - \eta\boldsymbol{Z}^\top\boldsymbol{Z}). \tag{9}$$

To the best of our knowledge, Theorem 3.1 provides the first theoretical result demonstrating that 2SLS can be efficiently implemented using a gradient-based method, with an exponential convergence rate. We provide the proof in Appendix B.1 and present simulation results in Appendix C.2 to examine the convergence behavior of the optimization process.

3.3 Transformers Can Efficiently Implement GD-2SLS

The looped transformer architecture (Definition 3.4), as proposed by Giannou et al. (2023), introduces an efficient approach to learn iterative algorithms by cascading the same transformer block for multiple times. With the GD updates in Equation (8), we will show that there exists a looped transformer architecuture that can efficiently learn the 2SLS estimator. We emphasize here that although we can implement 2SLS by sequentially attaching two separate GD iterates (each handling OLS for one stage), the overall convergence depends heavily on the convergence of the first stage estimate $\hat{\boldsymbol{\Theta}}$. Hence, significantly more number of layers are needed to ensure convergence. In addition, the advantage of looped transformer architecture cannot be fully exploited with this approach.

Theorem 3.2 (Implement a step of GD-2SLS with a transformer block)**.** *Suppose the embedded input matrix takes the form:*

$$\boldsymbol{H}^{(2l)} = \begin{bmatrix} \boldsymbol{z}_1 & \cdots & \boldsymbol{z}_n & \boldsymbol{z}_{n+1} \\ \boldsymbol{x}_1 & \cdots & \boldsymbol{x}_n & \boldsymbol{x}_{n+1} \\ y_1 & \cdots & y_n & t \\ \boldsymbol{\Theta}^{(l)}_{:,1} & \cdots & \boldsymbol{\Theta}^{(l)}_{:,1} & \boldsymbol{\Theta}^{(l)}_{:,1} \\ \vdots & \vdots & \vdots & \vdots \\ \boldsymbol{\Theta}^{(l)}_{:,p} & \cdots & \boldsymbol{\Theta}^{(l)}_{:,p} & \boldsymbol{\Theta}^{(l)}_{:,p} \\ \boldsymbol{\beta}^{(l)} & \cdots & \boldsymbol{\beta}^{(l)} & \boldsymbol{\beta}^{(l)} \\ \hat{\boldsymbol{x}}^{(l)}_1 & \cdots & \hat{\boldsymbol{x}}^{(l)}_n & \hat{\boldsymbol{x}}^{(l)}_{n+1} \\ 1 & \cdots & 1 & 1 \\ 1 & \cdots & 1 & 0 \end{bmatrix} \in \mathbb{R}^{D\times(n+1)}. \tag{10}$$

Given $\boldsymbol{H}^{(2l)}$, there exists a double-layer attention-only transformer block with parameters $\boldsymbol{\theta} = \boldsymbol{\theta}^{(2l+1:2l+2)}_{\mathsf{ATTN}} = \{(\boldsymbol{Q}^{(2l+1:2l+2)}_m, \boldsymbol{K}^{(2l+1:2l+2)}_m, \boldsymbol{V}^{(2l+1:2l+2)}_m)\}_{m\in[M^{(2l+1:2l+2)}]} \subset \mathbb{R}^{D\times D}$, where the number of heads $M^{(2l+1)} = 2p$, $M^{(2l+2)} = 2(p+1)$ and embedding dimension $D = qp+3p+q+3$, that implements a 2SLS gradient update in Equation (8) with any given learning rates α, η:

$$\boldsymbol{H}^{2(l+1)} = \mathsf{TF}_{\boldsymbol{\theta}^{(2l+1:2l+2)}_{\mathsf{ATTN}}}(\boldsymbol{H}^{(2l)}) = \begin{bmatrix} \boldsymbol{z}_1 & \cdots & \boldsymbol{z}_n & \boldsymbol{z}_{n+1} \\ \boldsymbol{x}_1 & \cdots & \boldsymbol{x}_n & \boldsymbol{x}_{n+1} \\ y_1 & \cdots & y_n & 0 \\ \boldsymbol{\Theta}^{(l+1)}_{:,1} & \cdots & \boldsymbol{\Theta}^{(l+1)}_{:,1} & \boldsymbol{\Theta}^{(l+1)}_{:,1} \\ \vdots & \vdots & \vdots & \vdots \\ \boldsymbol{\Theta}^{(l+1)}_{:,p} & \cdots & \boldsymbol{\Theta}^{(l+1)}_{:,p} & \boldsymbol{\Theta}^{(l+1)}_{:,p} \\ \boldsymbol{\beta}^{(l+1)} & \cdots & \boldsymbol{\beta}^{(l+1)} & \boldsymbol{\beta}^{(l+1)} \\ \hat{\boldsymbol{x}}^{(l+1)}_1 & \cdots & \hat{\boldsymbol{x}}^{(l+1)}_n & \hat{\boldsymbol{x}}^{(l+1)}_{n+1} \\ 1 & \cdots & 1 & 1 \\ 1 & \cdots & 1 & 0 \end{bmatrix} \in \mathbb{R}^{D\times(n+1)}.$$

Our existence proof specifies an attention structure such that one layer updates only the first-stage estimate $\hat{\boldsymbol{x}}^{(l)}_i$ for all samples, followed by another layer to update the parameters $\boldsymbol{\Theta}^{(l)}$ and $\boldsymbol{\beta}^{(l)}$. Furthermore, as noted in the proof of Theorem 3.2 (ref. Appendix B.2), regardless of the initial values of $\boldsymbol{\Theta}^{(l)}, \boldsymbol{\beta}^{(l)}$ and $\hat{\boldsymbol{x}}^{(l)}$, the structures of the transformer blocks remain the same. This allows us to exploit the looped transformer architecture to significantly reduce the number of parameters and improve learning efficiency (Yang et al., 2023). By cascading the transformer block $\bar{L}$ times, with Theorem 3.1, one can show that transformers are able to mimic the 2SLS estimator with exponential convergence rate, as described in the following corollary.

Corollary 3.1 (Implementing GD-2SLS with looped transformer)**.** *For any $0 < \varepsilon < 1$, given learning rates α, η, and $\Lambda \in (0, 1)$, as defined in Equation (9), there exists a transformer formulated as $\mathsf{TF}_{\boldsymbol{\theta}}(\cdot) := \mathsf{TF}_{\boldsymbol{\theta}'}(\mathsf{LTF}_{\bar{\boldsymbol{\theta}},\bar{L}}(\cdot))$, which consists of an $\bar{L}$-looped transformer $\mathsf{LTF}_{\bar{\boldsymbol{\theta}},\bar{L}}$ with $\bar{\boldsymbol{\theta}} = \bar{\boldsymbol{\theta}}^{(1:2)}_{\mathsf{ATTN}} = \{(\bar{\boldsymbol{Q}}^{(1:2)}_m, \bar{\boldsymbol{K}}^{(1:2)}_m, \bar{\boldsymbol{V}}^{(1:2)}_m)\}_{m\in[\bar{M}^{(1:2)}]} \subset \mathbb{R}^{D\times D}$, $\bar{L} = \lceil \mathcal{O}(\log_\Lambda(\varepsilon)) \rceil$, and a final attention layer[1] $\boldsymbol{\theta}' = \boldsymbol{\theta}'_{\mathsf{ATTN}} = \{(\boldsymbol{Q}'_m, \boldsymbol{K}'_m, \boldsymbol{V}'_m)\}_{m\in[M']} \subset \mathbb{R}^{D\times D}$, where $\bar{M}^{(1)} = 2p$, $\bar{M}^{(2)} = 2(p+1)$, $M' = 2$, such that given embedded input $\boldsymbol{H}^{(0)}$ taking the format in Equation (10), the model output satisfies:*

$$|\mathsf{read}_y(\mathsf{TF}_{\boldsymbol{\theta}}(\boldsymbol{H}^{(0)})) - \hat{\boldsymbol{\beta}}^\top_{\mathsf{2SLS}}\boldsymbol{x}_{n+1}| \le B_x\varepsilon,$$

where $\mathsf{read}_y(\cdot)$ is a function that reads the prediction $\hat{y}_{n+1}$ from the output of the transformer.

We emphasize here that our construction differs from the implementation of Bai et al. (2024, Theorem 4) for OLS in the following aspects:

[1]This layer updates the prediction $\hat{y}_{n+1} := \boldsymbol{\beta}^{(\bar{L})\top}\boldsymbol{x}_{n+1}$, which can be constructed with 2 attention heads using the same architecture as Bai et al. (2024, Theorem 13)

i. We apply the square loss as defined in Equation (7) to learn the 2SLS estimator, which simplifies the loss function's sum-of-ReLU representation.

ii. The dimension of the input embedding is $D = qp + 3p + q + 3$, where the extra dimensions store the vectorized parameters $\boldsymbol{\Theta}^{(l)}, \boldsymbol{\beta}^{(l)}$, and the first stage estimate $\hat{\boldsymbol{x}}^{(l)}$.

iii. We use a two-layer attention-only transformer block $\bar{\boldsymbol{\theta}}$ to implement a 2SLS GD update Equation (8), with the first layer to update the current first-stage estimate $\hat{\boldsymbol{x}}^{(l)}$, and the second layer to update the parameters $\boldsymbol{\Theta}^{(l)}$ and $\boldsymbol{\beta}^{(l)}$.

iv. For each transformer block, in the first layer, we equip 2 heads to update each dimension of $\hat{\boldsymbol{x}}_i^{(l)} \in \mathbb{R}^p$ for all samples. In the second layer, we equip 2 heads to update each column of $\boldsymbol{\Theta}^{(l)} \in \mathbb{R}^{q \times p}$ and $\boldsymbol{\beta}^{(l)} \in \mathbb{R}^p$.

3.4 Pretraining and Excess Loss Bound

With slightly abuse of notations, we denote the (formulated) training prompt as:

$$\boldsymbol{H}_k = \begin{bmatrix} \boldsymbol{z}_{1,k} & \cdots & \boldsymbol{z}_{n,k} & \boldsymbol{z}_{n+1,k} \\ \boldsymbol{x}_{1,k} & \cdots & \boldsymbol{x}_{n,k} & \boldsymbol{x}_{n+1,k} \\ y_{1,k} & \cdots & y_{n,k} & 0 \end{bmatrix} \in \mathbb{R}^{(p+q+1)\times(n+1)}, \quad k = 1, \ldots, N.$$

Note that we denote each training prompt by the subscript $k = 1, \ldots, N$, where N is the total number of prompts. Each training prompt consists of n labeled training samples $\{(\boldsymbol{z}_i, \boldsymbol{x}_i, y_i)\}_{i=1}^n$, and one unlabeled query sample $(\boldsymbol{z}_{n+1}, \boldsymbol{x}_{n+1})$. Our goal is to predict y_{n+1} given the context provided by the prompt.

We introduce the following ICL data generating scheme such that endogeneity occurs in the training samples, but does not extend to the query sample. Each training prompt is generated by the in-context distribution $\boldsymbol{\mathcal{P}}$, described by Algorithm 1.

Algorithm 1 In-Context Distribution $\boldsymbol{\mathcal{P}}$

1: **Parameters:** Sample size n, clipping thresholds B_z, B_x, B_y. Task parameters $\boldsymbol{\Theta}, \boldsymbol{\beta}, \boldsymbol{\Phi}, \boldsymbol{\phi}$, $\boldsymbol{\Sigma}_z, \boldsymbol{\Sigma}_u, \boldsymbol{\Sigma}_\omega, \sigma_\epsilon$ from meta distribution $\boldsymbol{\pi}$.
2: **Output:** Training samples $\{(\boldsymbol{z}_i, \boldsymbol{x}_i, y_i)\}_{i=1}^n$, query sample $(\boldsymbol{z}_{n+1}, \boldsymbol{x}_{n+1}, \boldsymbol{y}_{n+1})$.
3: **for** $i = 1, \ldots, n$ **do**
4: **Generate:** $\boldsymbol{z}_i \sim \mathcal{N}(0, \boldsymbol{\Sigma}_z)$, $\boldsymbol{u}_i \sim \mathcal{N}(0, \boldsymbol{\Sigma}_u)$, $\boldsymbol{\omega}_i \sim \mathcal{N}(0, \boldsymbol{\Sigma}_\omega)$, $\epsilon_i \sim \mathcal{N}(0, \sigma_\epsilon^2)$.
5: **Compute:** $\boldsymbol{x}_i = \boldsymbol{\Theta}^\top \boldsymbol{z}_i + \boldsymbol{\Phi}^\top \boldsymbol{u}_i + \boldsymbol{\omega}_i$.
6: **Compute:** $y_i = \boldsymbol{\beta}^\top \boldsymbol{x}_i + \boldsymbol{\phi}^\top \boldsymbol{u}_i + \epsilon_i$.
7: **end for**
8: **Generate:** $\boldsymbol{z}_{n+1} \sim \mathcal{N}(0, \boldsymbol{\Sigma}_z)$, $\boldsymbol{\omega}_{n+1} \sim \mathcal{N}(0, \boldsymbol{\Sigma}_\omega)$, $\epsilon_{n+1} \sim \mathcal{N}(0, \sigma_\epsilon^2)$.
9: **Compute:** $\boldsymbol{x}_{n+1} = \boldsymbol{\Theta}^\top \boldsymbol{z}_{n+1} + \boldsymbol{\omega}_{n+1}$.
10: **Compute:** $y_{n+1} = \boldsymbol{\beta}^\top \boldsymbol{x}_{n+1} + \epsilon_{n+1}$.
11: **Clip:** $\boldsymbol{z}_i = \mathsf{clip}_{B_z}(\boldsymbol{z}_i)$, $\boldsymbol{x}_i = \mathsf{clip}_{B_x}(\boldsymbol{x}_i)$, $y_i = \mathsf{clip}_{B_y}(y_i)$ for $i = 1, \ldots, n+1$.

In Algorithm 1, $\boldsymbol{u} \in \mathbb{R}^p$ is the source of endogenous error, $\boldsymbol{w} \in \mathbb{R}^p, \epsilon \in \mathbb{R}$ are the exogenous errors. Note that we have $\epsilon_{1,i} = \boldsymbol{\phi}^\top \boldsymbol{u}_i + \epsilon_i$ and $\boldsymbol{\epsilon}_{2,i} = \boldsymbol{\Phi}^\top \boldsymbol{u}_i + \boldsymbol{\omega}_i$, corresponding to the notations in Equation (1). $\boldsymbol{\Theta} \in \mathbb{R}^{q \times p}, \boldsymbol{\beta} \in \mathbb{R}^p, \boldsymbol{\Phi} \in \mathbb{R}^{p \times p}, \boldsymbol{\phi} \in \mathbb{R}^p, \boldsymbol{\Sigma}_z \in \mathbb{R}^{q \times q}, \boldsymbol{\Sigma}_u \in \mathbb{R}^{p \times p}, \boldsymbol{\Sigma}_\omega \in \mathbb{R}^{p \times p}, \sigma_\epsilon \in \mathbb{R}$ are task-specific parameters following meta distribution $\boldsymbol{\pi}$. $\mathsf{clip}_B(\cdot)$ is a clipping operator to bound the norm of input within radius B. We say that the in-context samples $\{(\boldsymbol{z}_i, \boldsymbol{x}_i, y_i)\}_{i=1}^{n+1}$ are drawn from the in-context distribution $\boldsymbol{\mathcal{P}}$, and $\boldsymbol{\mathcal{P}} \sim \boldsymbol{\pi}$ if the task parameters $(\boldsymbol{\Theta}, \boldsymbol{\beta}, \boldsymbol{\Phi}, \boldsymbol{\phi}, \boldsymbol{\Sigma}_z, \boldsymbol{\Sigma}_u, \boldsymbol{\Sigma}_\omega, \sigma_\epsilon)$ are sampled from $\boldsymbol{\pi}$. One can check that Assumption 1 and Assumption 2(ii)(iv) are directly satisfied with the data generated from the in-context distribution $\boldsymbol{\mathcal{P}}$.

Following the theoretical framework of (Bai et al., 2024), we define the population ICL loss[2]:

$$L_{\mathsf{ICL}}(\boldsymbol{\theta}) = \mathbb{E}_\pi \mathbb{E}_{\mathcal{P}}[y_{n+1} - \mathsf{clip}_{B_y}(\mathsf{read}_y(\mathsf{TF}_{\boldsymbol{\theta}}^R(\boldsymbol{H}^{(0)})))]^2, \tag{11}$$

[2]All the clipping operations are only for analytical purpose. In practice, the behavior of the trained transformer is consistent even without the clipping bounds.

where $\boldsymbol{H}^{(0)}$ is the embedded input as defined in Equation (10), $\mathsf{TF}_{\boldsymbol{\theta}}^{R}$ is the transformer model with parameter $\boldsymbol{\theta}$ and clipping operation $\mathsf{clip}_R(\cdot)$ applied to each layer output. For simplicity, we denote $\widetilde{\mathsf{TF}}_\theta(\boldsymbol{H}) := \mathsf{clip}_{B_y}(\mathsf{read}_y(\mathsf{TF}_{\boldsymbol{\theta}}^{R}(\boldsymbol{H}^{(0)})))$.

The transformer is trained to minimize the in-context loss in Equation (11) with the following empirical loss:

$$\hat{L}_{\mathsf{ICL}}(\boldsymbol{\theta}) = \frac{1}{N}\sum_{k=1}^{N}(y_{n+1,k} - \widetilde{\mathsf{TF}}_{\boldsymbol{\theta}}(\boldsymbol{H}_k))^2. \tag{12}$$

We consider the following constrained optimization problem:

$$\begin{aligned} \hat{\boldsymbol{\theta}} &:= \underset{\boldsymbol{\theta}\in\boldsymbol{\vartheta}_{L,M,D',B_\theta}}{\arg\min} \ \hat{L}_{\mathsf{ICL}}(\boldsymbol{\theta}), \\ \boldsymbol{\vartheta}_{L,M,D',B_\theta} &:= \{\boldsymbol{\theta} = (\boldsymbol{\theta}_{\mathsf{Attn}}^{(1:L)}, \boldsymbol{\theta}_{\mathsf{MLP}}^{(1:L)}) : \max_{l\in[L]} M^{(l)} \le M, \ \max_{l\in[L]} D^{(l)} \le D', \ \|\boldsymbol{\theta}\| \le B_\theta\}, \end{aligned} \tag{13}$$

where $\|\boldsymbol{\theta}\| := \max_{l\in[L]}\{\max_{m\in[M]}\{\|\boldsymbol{Q}_m^{(l)}\|, \|\boldsymbol{K}_m^{(l)}\|\} + \sum_{m=1}^{M}\|\boldsymbol{V}_m^{(l)}\| + \|\boldsymbol{W}_1^{(l)}\| + \|\boldsymbol{W}_2^{(l)}\|\}$.

We now establish excess loss bound for the trained transformer model.

Theorem 3.3 (Excess loss bound for in-context pretrained transformer)**.** *Suppose Condition (i)(iii) in Assumption 2 holds and the meta distribution $\boldsymbol{\pi}$ satisfies the following conditions:*

$$\mathbb{E}_\pi\left[\boldsymbol{\phi}^\top\boldsymbol{\Sigma}_u\boldsymbol{\phi} + \sigma_\epsilon^2\right] \le \tilde{\sigma}^2 \text{ and } \mathbb{E}_\pi\left[\sigma_\epsilon^2\right] \le \tilde{\sigma}_\epsilon^2. \tag{14}$$

Let the in-context distribution $\boldsymbol{\mathcal{P}} \sim \boldsymbol{\pi}$ such that the samples $(\boldsymbol{z}_i, \boldsymbol{x}_i, y_i)_{i=1}^{n+1}$ are drawn independently from $\boldsymbol{\mathcal{P}}$ (ref. Algorithm 1). With training prompts $\boldsymbol{H}_k, k = 1, \ldots, N$, under ICL loss in Equation (11), the trained transformer in Equation (13) with $L = 2\bar{L} + 1, M = 2(p+1), D = qp + 3p + q + 3, D' = 0$ (attention-only) achieves the following excess loss with probability at least $1 - \zeta$:

$$\begin{aligned} L_{\mathsf{ICL}}(\hat{\boldsymbol{\theta}}) - \mathbb{E}_\pi\mathbb{E}_{\boldsymbol{\mathcal{P}}}\left[(y_{n+1} - \langle\boldsymbol{\beta}, \boldsymbol{x}_{n+1}\rangle)^2\right] \le \mathcal{O}\Bigg(&(\Lambda^\star)^{\bar{L}}\Big(B_x^2\sqrt{\frac{C_{z,x}^2 q\tilde{\sigma}^2}{n} + \xi B_\beta^2} + B_x\tilde{\sigma}_\epsilon\Big) \\ &+ B_x^2\Big(\frac{C_{z,x}^2 q\tilde{\sigma}^2}{n} + \xi B_\beta^2 + \mu_{\Lambda,2}^\star\Big) + B_y^2\sqrt{\frac{L^2 M D^2 \log(2 + \max\{B_\theta, R, B_y\}) + \log(1/\zeta)}{N}}\Bigg), \end{aligned}$$

where $\Lambda^\star := \min_{\alpha,\eta}\mathbb{E}_\pi\mathbb{E}_{\boldsymbol{\mathcal{P}}}[\Lambda|\boldsymbol{H}, \alpha, \eta] < 1$, and $\mu_{\Lambda,2}^\star := \mathbb{E}_\pi\mathbb{E}_{\boldsymbol{\mathcal{P}}}[\Lambda^{2\bar{L}}|\boldsymbol{H}, \alpha^\star, \eta^\star]$ is close to 0.

In practical training, the number of prompts N is usually large enough such that the last term of the above bound is negligible. Thus, given a meta distribution $\boldsymbol{\pi}$, the excess loss is dominated by two factors: (i) number of attention layers, and (ii) number of in-context samples. The proof of Theorem 3.3 is provided in Appendix B.3.

3.5 Extracting the regression coefficients

The primary goal of IV regression is to estimate the causal effect, i.e. the coefficient $\boldsymbol{\beta}$ under the stated endogeneity in Equation (1). For 2SLS, the estimated causal effect is given by the coefficients of the endogenous variable in the second stage regression (2). For transformer models, we propose a straightforward method to extract these estimated coefficients by differentiating the output with respect to each dimension of the endogenous variable. The specific approach is summarized in Algorithm 2. We observe that the choice of Δ within a reasonable range does not significantly affect the estimation of the coefficients. In practice, usually a slightly larger Δ (for example $\Delta = 5$) can lead to a more stable estimation, which is possibly due to the elemination of rounding errors during computation.

4 Experiments

4.1 Experiment Setup

We conduct a simulation study to evaluate the performance of the ICL-pretrained transformer model in handling endogeneity. We set the maximum input sample size to 51 ($n = 50$ training samples and

Algorithm 2 Extracting the regression coefficients

1: **Input:** Trained transformer model $\mathsf{TF}_{\hat{\boldsymbol{\theta}}}$, input matrix $\boldsymbol{H}$, perturbation Δ.
2: **Output:** Estimated coefficient $\hat{\boldsymbol{\beta}}$.
3: **Procedure:**
4: Compute the output of the transformer model: $\hat{\boldsymbol{Y}} = \widetilde{\mathsf{TF}}_{\hat{\boldsymbol{\theta}}}(\boldsymbol{H})$.
5: **for** each dimension $k = 1, \ldots, p$ **do**
6: Copy $\boldsymbol{H}_{\Delta(k)} = \boldsymbol{H}$. Set the k-th dimension of $\boldsymbol{x}_{n+1}$ to be $(\boldsymbol{x}_{n+1})_k + \Delta$ for $\boldsymbol{H}_{\Delta(k)}$.
7: Compute the new output value: $\hat{\boldsymbol{Y}}_{\Delta(k)} = \widetilde{\mathsf{TF}}_{\hat{\boldsymbol{\theta}}}(\boldsymbol{H}_{\Delta(k)})$.
8: Compute the estimated coefficient: $\hat{\beta}_k = \frac{\hat{\boldsymbol{Y}}_{\Delta(k)} - \hat{\boldsymbol{Y}}}{\Delta}$.
9: **end for**

one query sample), the dimension of endogenous variable $p = 5$, and the dimension of instrument $q = 10$. The training prompts are generated using Algorithm 1, with task parameters $\boldsymbol{\Theta}, \boldsymbol{\beta}, \boldsymbol{\Phi}, \boldsymbol{\phi}$ sampled from standard Gaussian distribution, and the covariance matrices $\boldsymbol{\Sigma}_z, \boldsymbol{\Sigma}_u, \boldsymbol{\Sigma}_\omega$ set to be identity matrices. The noise level σ_ϵ is set to 1. We ignore all the clipping bounds in the experiment ($B_\beta, B_\Theta, B_z, B_x, B_y, B_\theta, R$ set to infinity).

The backbone of the transformer block is initialized using GPT-2 settings, with 12 attention heads ($M = 12$), 80-dimensional embedding space ($D = 80$) and 2 layers ($L_0 = 2$), following the theoretical guidelines in Theorem 3.2. We employ the looped transformer architecture, consisting of 10 identical cascading transformer blocks. The transformer model is trained under the ICL loss (12) with a batch size of $N = 64$, over a total of 300,000 training steps.

We evaluate the trained transformer model on test prompts that are not included during training. As benchmarks, we compare the transformer's performance against the 2SLS and the OLS estimators, which are obtained by directly fitting the training samples $\{(\boldsymbol{z}_i, \boldsymbol{x}_i, y_i)\}_{i=1}^n$ within the text prompts. In contrast, the same trained transformer model is used without any parameter adjustments for each task. We compare the performance of these models from two aspects: the in-context prediction error (ICPE) on the query sample y_{n+1}, and the mean squared error (MSE) on the coefficient $\boldsymbol{\beta}$.

4.2 Results

We first investigate the performance of the trained transformer model over endogeneity tasks with varying training sample sizes from 20 to 50. The results are shown in Figure 1a. Under endogeneity, our transformer model achieves similar performance to that of the 2SLS estimator, with only small gaps in ICPE and MSE, both outperforming the OLS estimator.

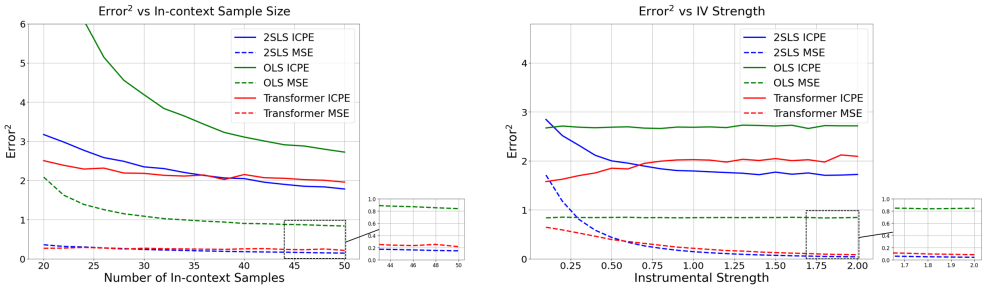
Next, we examine the performance of the trained transformer model in handling varying levels of IV strength. The strength of an instrument is measured by the correlation between the IV and the endogenous variable. To vary the IV strength, we generate prompts with $\boldsymbol{z}_i$ and $\boldsymbol{x}_i$ following different correlation levels. Specifically, in Algorithm 1, we adjust the IV strength by multiplying $\boldsymbol{\Theta}$ by a factor $r \in (0, 2)$ when generating test prompts. The results are shown in Figure 1b.

Interestingly, the trained transformer model outperforms the 2SLS estimator in handling weaker IVs (when IV strength < 0.5). This suggests that, beyond merely mimicking 2SLS, the ICL training process may equip the transformer model with a more advanced mechanism for handling endogeneity with weak IVs than the 2SLS estimator. At the same time, when the IV is strong, the transformer model maintains performance comparable to that of the 2SLS estimator.

This finding motivates us to further examine the performance of the trained transformer model in non-standard endogeneity tasks. We consider two scenarios: (a) the IV has a quadratic effect on the endogenous variable, i.e. $\boldsymbol{x}_{i,k} = \boldsymbol{\Theta}_k^\top \boldsymbol{z}_{i,k}^2 + \mathsf{error}_{i,k}$ in Algorithm 1, and (b) the dimension of IV is not sufficient to identify the endogenous variable[3], where we set $q = 3$ (by zeroing out the remaining dimensions of $\boldsymbol{z}$ in test prompts) and $p = 5$.

We evaluate the same trained transformer model as before, with results presented in Figure 2a and Figure 2b, respectively. Once again, the trained transformer model consistently outperforms both

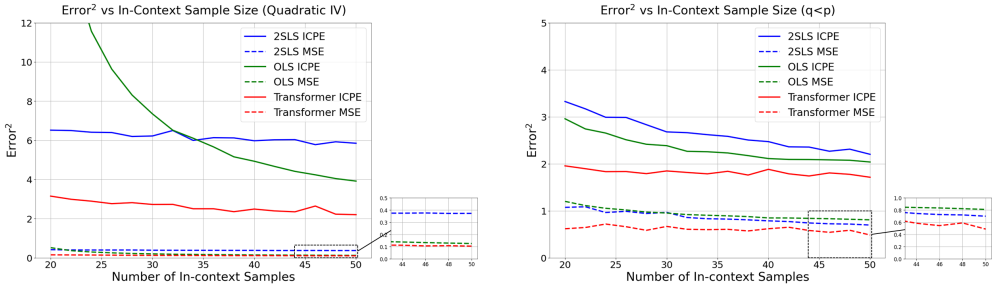
[3]For 2SLS estimate, the actual computation uses pseudoinverse to handle rank deficiency.



(a) (b)

Figure 1: The ICL performance of the trained transformer model in endogeneity tasks. We compare in-context prediction error (ICPE) and coefficient MSE versus (a) the number of in-context samples; (b) the IV strength. The curves are averaged over 500 simulations.

2SLS and OLS estimators in handling these non-standard endogeneity tasks. All these results suggest that the trained transformer can be generalized effectively to a broader range of endogeneity tasks while still providing reliable in-context predictions and coefficient estimates. To further illustrate this capability, we also examine other cases including multicollinearity, complex non-linear IV, and varying endogeneity strengths, see Appendix C.4,C.5,C.6 . We suspect that, in our pretraining scheme, although the 2SLS estimator already achieves small excess loss, a gap remains between the 2SLS estimator and the optimal predictor that the transformer model successfully bridges. Finally, we conclude that through ICL training, the transformer model performs at least as well as 2SLS and appears to be a promising tool for handling endogeneity in difficult scenarios.



(a) (b)

Figure 2: The ICL performance of the trained transformer model in non-standard endogeneity tasks: (a) The IV has quadratic effect on the endogenous variable; (b) The dimension of IV is not sufficient to identify the endogenous variable. The curves are averaged over 500 simulations.

5 Conclusion

This paper presents a novel perspective on the transformer model in its ability to handle endogeneity in in-context linear regression. We have theoretically shown that the transformer model exists an intrinsic structure that enables it to learn the 2SLS algorithm through an efficient GD procedure. We have further provided a theoretical guarantee that the trained transformer model can achieve a small excess loss over the optimal loss, under our proposed ICL training scheme. Our simulation study demonstrates that the trained transformer model can achieve comparable performance to the 2SLS estimator in handling standard endogeneity tasks. Furthermore, our investigation illustrates that it exhibits significantly better performances in handling complex scenarios such as weak instruments, non-linear IV, and underdetermined IV problems, compared to the 2SLS estimator. These results suggest that the ICL pre-trained transformer model is a promising tool for making reliable in-context predictions and coefficient estimates under endogeneity, especially when dealing with non-standard IV problems.

REFERENCES

Kwangjun Ahn, Xiang Cheng, Hadi Daneshmand, and Suvrit Sra. Transformers learn to implement preconditioned gradient descent for in-context learning. *Advances in Neural Information Processing Systems*, 36, 2024.

Ekin Akyürek, Dale Schuurmans, Jacob Andreas, Tengyu Ma, and Denny Zhou. What learning algorithm is in-context learning? investigations with linear models. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=0g0X4H8yN4I.

Joshua D. Angrist and William N. Evans. Children and their parents' labor supply: Evidence from exogenous variation in family size. *The American Economic Review*, 88(3):450–477, 1998. ISSN 00028282. URL http://www.jstor.org/stable/116844.

Joshua D. Angrist and Alan B. Krueger. Instrumental variables and the search for identification: From supply and demand to natural experiments. *Journal of Economic Perspectives*, 15(4):69–85, December 2001. doi: 10.1257/jep.15.4.69. URL https://www.aeaweb.org/articles?id=10.1257/jep.15.4.69.

Joshua D Angrist and Jörn-Steffen Pischke. *Mostly harmless econometrics: An empiricist's companion*. Princeton university press, 2009.

Yu Bai, Fan Chen, Huan Wang, Caiming Xiong, and Song Mei. Transformers as statisticians: Provable in-context learning with in-context algorithm selection. *Advances in neural information processing systems*, 36, 2024.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. *Advances in Neural Information Processing Systems*, 33, 2020.

Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm (eds.), *Computer Vision – ECCV 2020*, pp. 213–229, Cham, 2020. Springer International Publishing. ISBN 978-3-030-58452-8.

Xuxing Chen, Abhishek Roy, Yifan Hu, and Krishnakumar Balasubramanian. Stochastic optimization algorithms for instrumental variable regression with streaming data, 2024. URL https://arxiv.org/abs/2405.19463.

Xiang Cheng, Yuxin Chen, and Suvrit Sra. Transformers implement functional gradient descent to learn non-linear functions in context. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 8002–8037. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/cheng24a.html.

Riccardo Della Vecchia and Debabrota Basu. Stochastic online instrumental variable regression: Regrets for endogeneity and bandit feedback. *arXiv e-prints*, pp. arXiv–2302, 2023.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *North American Chapter of the Association for Computational Linguistics*, 2019. URL https://api.semanticscholar.org/CorpusID:52967399.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=YicbFdNTTy.

Benjamin L Edelman, Ezra Edelman, Surbhi Goel, Eran Malach, and Nikolaos Tsilivis. The evolution of statistical induction heads: In-context learning markov chains. *arXiv preprint arXiv:2402.11004*, 2024.

Fabian Falck, Ziyu Wang, and Chris Holmes. Is in-context learning in large language models bayesian? a martingale perspective. *arXiv preprint arXiv:2406.00793*, 2024.

Deqing Fu, Tian-Qi Chen, Robin Jia, and Vatsal Sharan. Transformers learn higher-order optimization methods for in-context learning: A study with linear models. *arXiv preprint arXiv:2310.17086*, 2023.

Yihang Gao, Chuanyang Zheng, Enze Xie, Han Shi, Tianyang Hu, Yu Li, Michael K. Ng, Zhenguo Li, and Zhaoqiang Liu. On the expressive power of a variant of the looped transformer, 2024. URL `https://arxiv.org/abs/2402.13572`.

Shivam Garg, Dimitris Tsipras, Percy S Liang, and Gregory Valiant. What can transformers learn in-context? a case study of simple function classes. *Advances in Neural Information Processing Systems*, 35:30583–30598, 2022.

Angeliki Giannou, Shashank Rajput, Jy-Yong Sohn, Kangwook Lee, Jason D. Lee, and Dimitris Papailiopoulos. Looped transformers as programmable computers. In *Proceedings of the 40th International Conference on Machine Learning*, pp. 11398–11442, 2023.

Angeliki Giannou, Liu Yang, Tianhao Wang, Dimitris Papailiopoulos, and Jason D. Lee. How well can transformers emulate in-context newton's method?, 2024. URL `https://arxiv.org/abs/2403.03183`.

Gautam Goel and Peter Bartlett. Can a transformer represent a kalman filter? In *6th Annual Learning for Dynamics & Control Conference*, pp. 1502–1512. PMLR, 2024.

W.H. Greene. *Econometric Analysis*. Econometric Analysis. Pearson, 2018. ISBN 9780134461366. URL `https://books.google.com/books?id=xGZRvgAACAAJ`.

Jerry Hausman. Mismeasured variables in econometric analysis: problems from the right and problems from the left. *Journal of Economic perspectives*, 15(4):57–67, 2001.

Yu Huang, Yuan Cheng, and Yingbin Liang. In-context convergence of transformers. In *International Conference on Machine Learning*. PMLR, 2024.

Yanhao Jin, Krishnakumar Balasubramanian, and Debashis Paul. Meta-learning with generalized ridge regression: High-dimensional asymptotics, optimality and hyper-covariance estimation, 2024.

Kuang-Huei Lee, Ofir Nachum, Mengjiao (Sherry) Yang, Lisa Lee, Daniel Freeman, Sergio Guadarrama, Ian Fischer, Winnie Xu, Eric Jang, Henryk Michalewski, and Igor Mordatch. Multi-game decision transformers. In *Advances in Neural Information Processing Systems*, 2022.

Yingcong Li, Muhammed Emrullah Ildiz, Dimitris Papailiopoulos, and Samet Oymak. Transformers as algorithms: Generalization and stability in in-context learning. In *International Conference on Machine Learning*, pp. 19565–19594. PMLR, 2023.

Ashok Vardhan Makkuva, Marco Bondaschi, Chanakya Ekbote, Adway Girish, Alliot Nagle, Hyeji Kim, and Michael Gastpar. Local to global: Learning dynamics and effect of initialization for transformers. In *ICML 2024 Workshop on Theoretical Foundations of Foundation Models*, 2024a. URL `https://openreview.net/forum?id=OYoCJPwbfC`.

Ashok Vardhan Makkuva, Marco Bondaschi, Alliot Nagle, Adway Girish, Hyeji Kim, Martin Jaggi, and Michael Gastpar. Attention with markov: A curious case of single-layer transformers. In *ICML 2024 Workshop on Mechanistic Interpretability*, 2024b. URL `https://openreview.net/forum?id=xi6lie0SUr`.

Krikamol Muandet, Arash Mehrjou, Si Kai Lee, and Anant Raj. Dual instrumental variable regression. *Advances in Neural Information Processing Systems*, 2020.

Eshaan Nichani, Alex Damian, and Jason D. Lee. How transformers learn causal structure with gradient descent, 2024. URL https://arxiv.org/abs/2402.14735.

Emilio Parisotto, H. Francis Song, Jack W. Rae, Razvan Pascanu, Caglar Gulcehre, Siddhant M. Jayakumar, Max Jaderberg, Raphael Lopez Kaufman, Aidan Clark, Seb Noury, Matthew M. Botvinick, Nicolas Heess, and Raia Hadsell. Stabilizing transformers for reinforcement learning, 2019.

Caio Peixoto, Yuri Saporito, and Yuri Fonseca. Nonparametric instrumental variable regression through stochastic approximate gradients. *arXiv preprint arXiv:2402.05639*, 2024.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019. URL https://api.semanticscholar.org/CorpusID:160025533.

Nived Rajaraman, Marco Bondaschi, Kannan Ramchandran, Michael Gastpar, and Ashok Vardhan Makkuva. Transformers on markov data: Constant depth suffices. *arXiv preprint arXiv:2407.17686*, 2024.

Mark Rudelson and Roman Vershynin. Non-asymptotic theory of random matrices: extreme singular values, 2010. URL https://arxiv.org/abs/1003.2990.

Rahul Singh, Maneesh Sahani, and Arthur Gretton. Kernel instrumental variable regression. *Advances in Neural Information Processing Systems*, 2019.

J.H. Stock and M.W. Watson. *Introduction to Econometrics*. Addison-Wesley series in economics. Addison-Wesley, 2011. ISBN 9780138009007. URL https://books.google.com/books?id=prLxRQAACAAJ.

Joel A. Tropp. An introduction to matrix concentration inequalities, 2015. URL https://arxiv.org/abs/1501.01571.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

Johannes Von Oswald, Eyvind Niklasson, Ettore Randazzo, João Sacramento, Alexander Mordvintsev, Andrey Zhmoginov, and Max Vladymyrov. Transformers learn in-context by gradient descent. In *International Conference on Machine Learning*, pp. 35151–35174. PMLR, 2023.

Charles F. Westoff and Robert Parke. *Demographic and social aspects of population growth*. Commission on Population Growth and the American Future, 1972.

J.M. Wooldridge. *Introductory Econometrics: A Modern Approach*. Cengage Learning, 2015. ISBN 9781473754393. URL https://books.google.com/books?id=HveHAQAACAAJ.

Liu Yang, Kangwook Lee, Robert Nowak, and Dimitris Papailiopoulos. Looped transformers are better at learning learning algorithms. *Workshop on Efficient Systems for Foundation Models @ ICML2023*, 2023.

Naimeng Ye, Hanming Yang, Andrew Siah, and Hongseok Namkoong. Pre-training and in-context learning is bayesian inference a la de finetti. *ICLR 2024 Workshop on Mathematical and Empirical Understanding of Foundation Models*, 2024.

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

Ruiqi Zhang, Jingfeng Wu, and Peter L Bartlett. In-context learning of a linear transformer block: benefits of the mlp component and one-step gd initialization. *arXiv preprint arXiv:2402.14951*, 2024b.

A PROOFS FOR SECTION 2

A.1 PROOF OF THEOREM 2.1

We first introduce the following lemmas that are used in the proof of Theorem 2.1.

Lemma A.1 (Bernstein Inequality, from Theorem 6.1.1 in Tropp (2015))**.** Let $\boldsymbol{S}_1, \ldots, \boldsymbol{S}_n$ be independent, centered random matrices with common dimension $d_1 \times d_2$, and assume that each one is almost surely bounded:

$$\mathbb{E}[\boldsymbol{S}_i] = \boldsymbol{0}, \mathbb{P}(\|\boldsymbol{S}_i\| \leq b) = 1, \quad \forall i = 1, \ldots, n.$$

With the sum:

$$\boldsymbol{\Omega} = \sum_{i=1}^{n} \boldsymbol{S}_i,$$

and the matrix variance statistic of the sum:

$$\nu(\boldsymbol{\Omega}) := \max\{\left\|\mathbb{E}(\boldsymbol{\Omega}\boldsymbol{\Omega}^\top)\right\|, \left\|\mathbb{E}(\boldsymbol{\Omega}^\top\boldsymbol{\Omega})\right\|\},$$

then the following inequality holds:

$$\mathbb{P}\{\|\boldsymbol{\Omega}\| \geq \varepsilon\} \leq (d_1 + d_2) \cdot \exp\left(\frac{-\varepsilon^2/2}{\nu(\boldsymbol{\Omega}) + b\varepsilon/3}\right) \text{ for any } \varepsilon \geq 0.$$

Lemma A.2 (Inverse Convergence, adapted from Lemma 2.1 in Jin et al. (2024))**.** Suppose we have a random invertible matrix $\boldsymbol{\Omega}$ and invertible matrix sequence $\{\hat{\boldsymbol{\Omega}}^{(n)}\}$ such that $\hat{\boldsymbol{\Omega}}^{(n)} \xrightarrow{\mathsf{p}} \boldsymbol{\Omega}$. If there exists a constant $\tilde{\lambda} > 0$ such that $\sigma_{\min}(\hat{\boldsymbol{\Omega}}) \geq \tilde{\lambda}$ almost surely, then it holds that:

$$(\hat{\boldsymbol{\Omega}}^{(n)})^{-1} \xrightarrow{\mathsf{p}} \boldsymbol{\Omega}^{-1}.$$

Further, given convergence rate

$$\mathbb{P}\left\{\left\|\hat{\boldsymbol{\Omega}}^{(n)} - \boldsymbol{\Omega}\right\| \geq \varepsilon\right\} \leq \xi(n, \varepsilon),$$

then:

$$\mathbb{P}\left\{\left\|(\hat{\boldsymbol{\Omega}}^{(n)})^{-1} - \boldsymbol{\Omega}^{-1}\right\| \geq \varepsilon\right\} \leq \xi(n, \tilde{\lambda}^2\varepsilon).$$

Proof. We have the following decomposition:

$$(\hat{\boldsymbol{\Omega}}^{(n)})^{-1} - \boldsymbol{\Omega}^{-1} = (\hat{\boldsymbol{\Omega}}^{(n)})^{-1}(\boldsymbol{\Omega} - \hat{\boldsymbol{\Omega}}^{(n)})\boldsymbol{\Omega}^{-1}.$$

It follows that:

$$\begin{aligned}\left\|(\hat{\boldsymbol{\Omega}}^{(n)})^{-1} - \boldsymbol{\Omega}^{-1}\right\| &\leq \left\|(\hat{\boldsymbol{\Omega}}^{(n)})^{-1}\right\| \left\|\boldsymbol{\Omega} - \hat{\boldsymbol{\Omega}}^{(n)}\right\| \|\boldsymbol{\Omega}^{-1}\| \\ &\leq \frac{1}{\tilde{\lambda}^2} \left\|\boldsymbol{\Omega} - \hat{\boldsymbol{\Omega}}^{(n)}\right\|.\end{aligned}$$

Then

$$\begin{aligned}\mathbb{P}\left\{\left\|(\hat{\boldsymbol{\Omega}}^{(n)})^{-1} - \boldsymbol{\Omega}^{-1}\right\| \geq \varepsilon\right\} &\leq \mathbb{P}\left\{\frac{1}{\tilde{\lambda}^2}\left\|\boldsymbol{\Omega} - \hat{\boldsymbol{\Omega}}^{(n)}\right\| \geq \varepsilon\right\} \\ &\leq \xi(n, \tilde{\lambda}^2\varepsilon).\end{aligned}$$

□

Lemma A.3 (Product Convergence)**.** Let $\{\hat{\boldsymbol{\Omega}}_1^{(n)}\}, \{\hat{\boldsymbol{\Omega}}_2^{(n)}\}, \ldots, \{\hat{\boldsymbol{\Omega}}_K^{(n)}\}$ be K sequences of matrices such that $\hat{\boldsymbol{\Omega}}_1^{(n)} \xrightarrow{\mathsf{p}} \boldsymbol{\Omega}_1, \hat{\boldsymbol{\Omega}}_2^{(n)} \xrightarrow{\mathsf{p}} \boldsymbol{\Omega}_2, \ldots, \hat{\boldsymbol{\Omega}}_K^{(n)} \xrightarrow{\mathsf{p}} \boldsymbol{\Omega}_K$, where each $\|\hat{\boldsymbol{\Omega}}_k^{(n)}\|$ is almost surely bounded for every $k = 1, \ldots, K$. If the dimensions match, then it holds that:

$$\hat{\boldsymbol{\Omega}}_1^{(n)}\hat{\boldsymbol{\Omega}}_2^{(n)} \cdots \hat{\boldsymbol{\Omega}}_K^{(n)} \xrightarrow{\mathsf{p}} \boldsymbol{\Omega}_1\boldsymbol{\Omega}_2 \cdots \boldsymbol{\Omega}_K.$$

Further, given convergence rates:

$$\begin{aligned}
\mathbb{P}\left\{\left\|\hat{\mathbf{\Omega}}_1^{(n)} - \mathbf{\Omega}_1\right\| \geq \varepsilon\right\} &\leq \xi_1(n, \varepsilon), \\
\mathbb{P}\left\{\left\|\hat{\mathbf{\Omega}}_2^{(n)} - \mathbf{\Omega}_2\right\| \geq \varepsilon\right\} &\leq \xi_2(n, \varepsilon), \\
&\vdots \\
\mathbb{P}\left\{\left\|\hat{\mathbf{\Omega}}_K^{(n)} - \mathbf{\Omega}_K\right\| \geq \varepsilon\right\} &\leq \xi_K(n, \varepsilon),
\end{aligned}$$

then it holds that:

$$\mathbb{P}\left\{\left\|\hat{\mathbf{\Omega}}_1^{(n)}\hat{\mathbf{\Omega}}_2^{(n)}\cdots\hat{\mathbf{\Omega}}_K^{(n)} - \mathbf{\Omega}_1\mathbf{\Omega}_2\cdots\mathbf{\Omega}_K\right\| \geq \varepsilon\right\} \leq \sum_{i=1}^{K} \xi_i\left(n, \frac{\varepsilon}{K\prod_{k\neq i}^{K} M_k}\right), \tag{15}$$

where M_k is an upper bound such that $\|\hat{\mathbf{\Omega}}_k^{(n)}\| \leq M_k$ almost surely, $\forall k = 1, \ldots, K$.

Proof. We begin by showing the case of $K = 2$. By the triangle inequality, we have:

$$\begin{aligned}
\left\|\hat{\mathbf{\Omega}}_1^{(n)}\hat{\mathbf{\Omega}}_2^{(n)} - \mathbf{\Omega}_1\mathbf{\Omega}_2\right\| &\leq \left\|\hat{\mathbf{\Omega}}_1^{(n)}\hat{\mathbf{\Omega}}_2^{(n)} - \mathbf{\Omega}_1\hat{\mathbf{\Omega}}_2^{(n)}\right\| + \left\|\mathbf{\Omega}_1\hat{\mathbf{\Omega}}_2^{(n)} - \mathbf{\Omega}_1\mathbf{\Omega}_2\right\| \\
&\leq \left\|\hat{\mathbf{\Omega}}_2^{(n)}\right\| \left\|\hat{\mathbf{\Omega}}_1^{(n)} - \mathbf{\Omega}_1\right\| + \left\|\hat{\mathbf{\Omega}}_2^{(n)} - \mathbf{\Omega}_2\right\| \|\mathbf{\Omega}_1\| \\
&\leq M_2 \left\|\hat{\mathbf{\Omega}}_1^{(n)} - \mathbf{\Omega}_1\right\| + M_1 \left\|\hat{\mathbf{\Omega}}_2^{(n)} - \mathbf{\Omega}_2\right\|.
\end{aligned}$$

Using the union bound, we have:

$$\begin{aligned}
&\mathbb{P}\left\{\left\|\hat{\mathbf{\Omega}}_1^{(n)}\hat{\mathbf{\Omega}}_2^{(n)} - \mathbf{\Omega}_1\mathbf{\Omega}_2\right\| \geq \varepsilon\right\} \\
\leq &\mathbb{P}\left\{M_2 \left\|\hat{\mathbf{\Omega}}_1^{(n)} - \mathbf{\Omega}_1\right\| + M_1 \left\|\hat{\mathbf{\Omega}}_2^{(n)} - \mathbf{\Omega}_2\right\| \geq \varepsilon\right\} \\
\leq &\mathbb{P}\left\{M_2 \left\|\hat{\mathbf{\Omega}}_1^{(n)} - \mathbf{\Omega}_1\right\| \geq \varepsilon/2\right\} + \mathbb{P}\left\{M_1 \left\|\hat{\mathbf{\Omega}}_2^{(n)} - \mathbf{\Omega}_2\right\| \geq \varepsilon/2\right\} \\
\leq &\xi_1\left(n, \frac{\varepsilon}{2M_2}\right) + \xi_2\left(n, \frac{\varepsilon}{2M_1}\right).
\end{aligned}$$

For any $K > 2$, suppose the statement (15) holds for $k = 2, \ldots, K-1$. Observe that:

$$\begin{aligned}
&\left\|\hat{\mathbf{\Omega}}_1^{(n)}\hat{\mathbf{\Omega}}_2^{(n)}\cdots\hat{\mathbf{\Omega}}_K^{(n)} - \mathbf{\Omega}_1\mathbf{\Omega}_2\cdots\mathbf{\Omega}_K\right\| \\
&\leq \left\|\hat{\mathbf{\Omega}}_1^{(n)}\hat{\mathbf{\Omega}}_2^{(n)}\cdots\hat{\mathbf{\Omega}}_K^{(n)} - \mathbf{\Omega}_1\mathbf{\Omega}_2\cdots\mathbf{\Omega}_{K-1}\hat{\mathbf{\Omega}}_K^{(n)}\right\| + \left\|\mathbf{\Omega}_1\mathbf{\Omega}_2\cdots\mathbf{\Omega}_{K-1}\hat{\mathbf{\Omega}}_K^{(n)} - \mathbf{\Omega}_1\mathbf{\Omega}_2\cdots\mathbf{\Omega}_K\right\| \\
&\leq M_K \left\|\hat{\mathbf{\Omega}}_1^{(n)}\hat{\mathbf{\Omega}}_2^{(n)}\cdots\hat{\mathbf{\Omega}}_{K-1}^{(n)} - \mathbf{\Omega}_1\mathbf{\Omega}_2\cdots\mathbf{\Omega}_{K-1}\right\| + \prod_{k=1}^{K-1} M_k \left\|\hat{\mathbf{\Omega}}_K^{(n)} - \mathbf{\Omega}_K\right\|.
\end{aligned} \tag{16}$$

Then it follows that:

$$
\begin{aligned}
&\mathbb{P}\left\{\left\|\hat{\boldsymbol{\Omega}}_1^{(n)}\hat{\boldsymbol{\Omega}}_2^{(n)}\cdots\hat{\boldsymbol{\Omega}}_K^{(n)}-\boldsymbol{\Omega}_1\boldsymbol{\Omega}_2\cdots\boldsymbol{\Omega}_K\right\|\geq\varepsilon\right\}\\
&\leq\mathbb{P}\left\{M_K\left\|\hat{\boldsymbol{\Omega}}_1^{(n)}\hat{\boldsymbol{\Omega}}_2^{(n)}\cdots\hat{\boldsymbol{\Omega}}_{K-1}^{(n)}-\boldsymbol{\Omega}_1\boldsymbol{\Omega}_2\cdots\boldsymbol{\Omega}_{K-1}\right\|+\prod_{k=1}^{K-1}M_k\left\|\hat{\boldsymbol{\Omega}}_K^{(n)}-\boldsymbol{\Omega}_K\right\|\geq\varepsilon\right\}\\
&\leq\mathbb{P}\left\{M_K\left\|\hat{\boldsymbol{\Omega}}_1^{(n)}\hat{\boldsymbol{\Omega}}_2^{(n)}\cdots\hat{\boldsymbol{\Omega}}_{K-1}^{(n)}-\boldsymbol{\Omega}_1\boldsymbol{\Omega}_2\cdots\boldsymbol{\Omega}_{K-1}\right\|\geq\frac{K-1}{K}\varepsilon\right\}\\
&\qquad+\mathbb{P}\left\{\prod_{k=1}^{K-1}M_k\left\|\hat{\boldsymbol{\Omega}}_K^{(n)}-\boldsymbol{\Omega}_K\right\|\geq\frac{1}{K}\varepsilon\right\}\\
&\leq\sum_{i=1}^{K-1}\xi_i\left(n,\frac{\varepsilon}{KM_K\prod_{k\neq i}^{K-1}M_k}\right)+\xi_K\left(n,\frac{\varepsilon}{K\prod_{k=1}^{K-1}M_k}\right)\\
&=\sum_{i=1}^{K}\xi_i\left(n,\frac{\varepsilon}{K\prod_{k\neq i}^{K}M_k}\right).
\end{aligned}
$$

Thus, by induction, the proof is complete. □

Remark A.1. In Lemma A.3, consider the special case where $\boldsymbol{\Omega}_1=\mathbf{0}$. Then the inequality (16) can be simplified as follows:

$$
\left\|\hat{\boldsymbol{\Omega}}_1^{(n)}\hat{\boldsymbol{\Omega}}_2^{(n)}\cdots\hat{\boldsymbol{\Omega}}_K^{(n)}-\mathbf{0}\right\|\leq\prod_{k=2}^{K}M_k\left\|\hat{\boldsymbol{\Omega}}_1^{(n)}\right\|.
$$

And we have the following simplified form:

$$
\begin{aligned}
\mathbb{P}\left\{\left\|\hat{\boldsymbol{\Omega}}_1^{(n)}\hat{\boldsymbol{\Omega}}_2^{(n)}\cdots\hat{\boldsymbol{\Omega}}_K^{(n)}-\mathbf{0}\right\|\geq\varepsilon\right\}&\leq\mathbb{P}\left\{\prod_{k=2}^{K}M_k\left\|\hat{\boldsymbol{\Omega}}_1^{(n)}\right\|\geq\varepsilon\right\}\\
&\leq\xi_1\left(n,\frac{\varepsilon}{\prod_{k=2}^{K}M_k}\right).
\end{aligned}
$$

Proof of Theorem 2.1. We denote the observational values $(\boldsymbol{Z},\boldsymbol{X},\boldsymbol{Y})=\{(\boldsymbol{z}_i,\boldsymbol{x}_i,y_i)\}_{i=1}^n$, and $\boldsymbol{\mathcal{E}}_1=\{\epsilon_{1,i}\}_{i=1}^n$. The 2SLS estimator is given by:

$$
\begin{aligned}
\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}&=\left(\hat{\boldsymbol{\Theta}}^\top\boldsymbol{Z}^\top\boldsymbol{Z}\hat{\boldsymbol{\Theta}}\right)^{-1}\hat{\boldsymbol{\Theta}}^\top\boldsymbol{Z}^\top\boldsymbol{Y}\\
&=\left[\left((\boldsymbol{Z}^\top\boldsymbol{Z})^{-1}\boldsymbol{Z}^\top\boldsymbol{X}\right)^\top\boldsymbol{Z}^\top\boldsymbol{Z}(\boldsymbol{Z}^\top\boldsymbol{Z})^{-1}\boldsymbol{Z}^\top\boldsymbol{X}\right]^{-1}\left((\boldsymbol{Z}^\top\boldsymbol{Z})^{-1}\boldsymbol{Z}^\top\boldsymbol{X}\right)^\top\boldsymbol{Z}^\top\boldsymbol{Y}\\
&=\left(\boldsymbol{X}^\top\boldsymbol{Z}(\boldsymbol{Z}^\top\boldsymbol{Z})^{-1}\boldsymbol{Z}^\top\boldsymbol{X}\right)^{-1}\boldsymbol{X}^\top\boldsymbol{Z}(\boldsymbol{Z}^\top\boldsymbol{Z})^{-1}\boldsymbol{Z}^\top\boldsymbol{Y}\\
&=\boldsymbol{\beta}+\left(\boldsymbol{X}^\top\boldsymbol{Z}(\boldsymbol{Z}^\top\boldsymbol{Z})^{-1}\boldsymbol{Z}^\top\boldsymbol{X}\right)^{-1}\boldsymbol{X}^\top\boldsymbol{Z}(\boldsymbol{Z}^\top\boldsymbol{Z})^{-1}\boldsymbol{Z}^\top\boldsymbol{\mathcal{E}}_1.
\end{aligned}
\tag{17}
$$

We first consider the case when event $\mathcal{A}$ is true. Let $\boldsymbol{Q}_{zz}:=\mathbb{E}[\boldsymbol{z}\boldsymbol{z}^\top|\mathcal{A}],\boldsymbol{Q}_{zx}:=\mathbb{E}[\boldsymbol{z}\boldsymbol{x}^\top|\mathcal{A}]$, $\bar{\boldsymbol{\Omega}}_{zz}:=\sum_{i=1}^n(\boldsymbol{z}_i\boldsymbol{z}_i^\top-\boldsymbol{Q}_{zz}),\bar{\boldsymbol{\Omega}}_{zx}:=\sum_{i=1}^n(\boldsymbol{z}_i\boldsymbol{x}_i^\top-\boldsymbol{Q}_{zx}),\boldsymbol{\Omega}_{z\epsilon_1}:=\sum_{i=1}^n\boldsymbol{z}_i\epsilon_{1,i}$.

Let $\bar{B}_{zz},\bar{B}_{zx},B_{zx},B_{z\epsilon_1}$ be some upper bounds such that $\left\|\boldsymbol{z}_i\boldsymbol{z}_i^\top-\boldsymbol{Q}_{zz}\right\|\leq\bar{B}_{zz},\left\|\boldsymbol{z}_i\boldsymbol{x}_i^\top-\boldsymbol{Q}_{zx}\right\|\leq\bar{B}_{zx},\left\|\boldsymbol{z}_i\boldsymbol{x}_i^\top\right\|\leq B_{zx},\|\boldsymbol{z}_i\epsilon_{1,i}\|\leq B_{z\epsilon_1}$ almost surely, for all $i=1,\ldots,n$. The existence of $\bar{B}_{zz},\bar{B}_{zx},B_{zx},B_{z\epsilon_1}$ is guaranteed under Assumption 2.

By Lemma A.1, we have:

$$\begin{aligned}\mathbb{P}\left\{\left\|\frac{\boldsymbol{Z}^\top\boldsymbol{Z}}{n}-\boldsymbol{Q}_{zz}\right\|\geq\varepsilon\middle|\mathcal{A}\right\}&=\mathbb{P}\left\{\left\|\frac{\sum_{i=1}^n\boldsymbol{z}_i\boldsymbol{z}_i^\top}{n}-\boldsymbol{Q}_{zz}\right\|\geq\varepsilon\middle|\mathcal{A}\right\}\\&=\mathbb{P}\left\{\left\|\sum_{i=1}^n(\boldsymbol{z}_i\boldsymbol{z}_i^\top-\boldsymbol{Q}_{zz})\right\|\geq n\varepsilon\middle|\mathcal{A}\right\}\\&\leq 2q\exp\left(-\frac{n^2\varepsilon^2/2}{\nu(\bar{\boldsymbol{\Omega}}_{zz}|\mathcal{A})+\bar{B}_{zz}n\varepsilon/3}\right).\end{aligned}\tag{18}$$

Similarly,

$$\begin{aligned}\mathbb{P}\left\{\left\|\frac{\boldsymbol{Z}^\top\boldsymbol{X}}{n}-\boldsymbol{Q}_{zx}\right\|\geq\varepsilon\middle|\mathcal{A}\right\}&=\mathbb{P}\left\{\left\|\frac{\sum_{i=1}^n\boldsymbol{z}_i\boldsymbol{x}_i^\top}{n}-\boldsymbol{Q}_{zx}\right\|\geq\varepsilon\middle|\mathcal{A}\right\}\\&=\mathbb{P}\left\{\left\|\sum_{i=1}^n(\boldsymbol{z}_i\boldsymbol{x}_i^\top-\boldsymbol{Q}_{zx})\right\|\geq n\varepsilon\middle|\mathcal{A}\right\}\\&\leq (p+q)\exp\left(-\frac{n^2\varepsilon^2/2}{\nu(\bar{\boldsymbol{\Omega}}_{zx}|\mathcal{A})+\bar{B}_{zx}n\varepsilon/3}\right).\end{aligned}\tag{19}$$

By Assumption 1(iii), the instrument $\boldsymbol{z}$ is uncorrelated with the error term ϵ_1, which implies $\mathbb{E}[\boldsymbol{z}\epsilon_1|\mathcal{A}]=\mathbf{0}$. Applying Lemma A.1 again, we have:

$$\begin{aligned}\mathbb{P}\left\{\left\|\frac{\boldsymbol{Z}^\top\boldsymbol{\mathcal{E}}_1}{n}\right\|\geq\varepsilon\middle|\mathcal{A}\right\}&=\mathbb{P}\left\{\left\|\frac{\sum_{i=1}^n\boldsymbol{z}_i\epsilon_{1,i}}{n}\right\|\geq\varepsilon\middle|\mathcal{A}\right\}\\&=\mathbb{P}\left\{\left\|\sum_{i=1}^n\boldsymbol{z}_i\epsilon_{1,i}\right\|\geq n\varepsilon\middle|\mathcal{A}\right\}\\&\leq (q+1)\exp\left(-\frac{n^2\varepsilon^2/2}{\nu(\boldsymbol{\Omega}_{z\epsilon_1}|\mathcal{A})+B_{z\epsilon_1}n\varepsilon/3}\right).\end{aligned}\tag{20}$$

With Lemma A.2 and (18), we have:

$$\begin{aligned}\mathbb{P}\left\{\left\|n(\boldsymbol{Z}^\top\boldsymbol{Z})^{-1}-\boldsymbol{Q}_{ZZ}^{-1}\right\|\geq\varepsilon\middle|\mathcal{A}\right\}&\leq 2q\exp\left(-\frac{n^2(\lambda_z^2\varepsilon)^2/2}{\nu(\bar{\boldsymbol{\Omega}}_{zz}|\mathcal{A})+\bar{B}_{zz}n(\lambda_z^2\varepsilon)/3}\right)\\&=2q\exp\left(-\frac{\lambda_z^4n^2\varepsilon^2/2}{\nu(\bar{\boldsymbol{\Omega}}_{zz}|\mathcal{A})+\lambda_z^2\bar{B}_{zz}n\varepsilon/3}\right).\end{aligned}\tag{21}$$

Note that under event $\mathcal{A}$,

$$\begin{aligned}\left\|\hat{\boldsymbol{\Theta}}\right\|&=\left\|\left(\boldsymbol{Z}^\top\boldsymbol{Z}\right)^{-1}\boldsymbol{Z}^\top\boldsymbol{X}\right\|\\&=\left\|\boldsymbol{\Theta}+\left(\boldsymbol{Z}^\top\boldsymbol{Z}\right)^{-1}\boldsymbol{Z}^\top\boldsymbol{\mathcal{E}}_2\right\|\\&\leq B_\Theta+\left\|\left(\boldsymbol{Z}^\top\boldsymbol{Z}\right)^{-1}\boldsymbol{Z}^\top\boldsymbol{\mathcal{E}}_2\right\|\\&\leq B_\Theta+\frac{\delta_{z\epsilon_2}}{\lambda_z}:=B_{\hat{\Theta}}.\end{aligned}\tag{22}$$

With Lemma A.3 (Remark A.1), combining (20)(22), we have:

$$\begin{aligned}\mathbb{P}\left\{\left\|\frac{1}{n}\boldsymbol{X}^\top\boldsymbol{Z}(\boldsymbol{Z}^\top\boldsymbol{Z})^{-1}\boldsymbol{Z}^\top\boldsymbol{\mathcal{E}}_1-\mathbf{0}\right\|\geq\varepsilon\middle|\mathcal{A}\right\}&\leq (q+1)\exp\left(-\frac{n^2(\frac{\varepsilon}{B_{\hat{\Theta}}})^2/2}{\nu(\boldsymbol{\Omega}_{z\epsilon_1}|\mathcal{A})+B_{z\epsilon_1}n(\frac{\varepsilon}{B_{\hat{\Theta}}})/3}\right)\\&=(q+1)\exp\left(-\frac{n^2\varepsilon^2/2}{B_{\hat{\Theta}}^2\nu(\boldsymbol{\Omega}_{z\epsilon_1}|\mathcal{A})+B_{\hat{\Theta}}B_{z\epsilon_1}n\varepsilon/3}\right).\end{aligned}\tag{23}$$

Additionally, with Lemma A.3, combining (19)(21), we have:

$$
\begin{aligned}
&\mathbb{P}\left\{\left\|\frac{1}{n}\boldsymbol{X}^\top\boldsymbol{Z}(\boldsymbol{Z}^\top\boldsymbol{Z})^{-1}\boldsymbol{Z}^\top\boldsymbol{X}-\boldsymbol{Q}_{zx}^\top\boldsymbol{Q}_{zz}^{-1}\boldsymbol{Q}_{zx}\right\|\geq\varepsilon\middle|\mathcal{A}\right\}\\
&\leq 2(p+q)\exp\left(-\frac{n^2(\frac{\lambda_z\varepsilon}{3\bar{B}_{zx}})^2/2}{\nu(\bar{\boldsymbol{\Omega}}_{zx}|\mathcal{A})+\bar{B}_{zx}n(\frac{\lambda_z\varepsilon}{3\bar{B}_{zx}})/3}\right)+2q\exp\left(-\frac{\lambda_z^4n^2(\frac{\varepsilon}{3\bar{B}_{zx}^2})^2/2}{\nu(\bar{\boldsymbol{\Omega}}_{zz}|\mathcal{A})+\lambda_z^2\bar{B}_{zz}n(\frac{\varepsilon}{3\bar{B}_{zx}^2})/3}\right)\\
&=2(p+q)\exp\left(-\frac{\lambda_z^2n^2\varepsilon^2/2}{9B_{zx}^2\nu(\bar{\boldsymbol{\Omega}}_{zx}|\mathcal{A})+\lambda_zB_{zx}\bar{B}_{zx}n\varepsilon}\right)+2q\exp\left(-\frac{\lambda_z^4n^2\varepsilon^2/2}{9B_{zx}^4\nu(\bar{\boldsymbol{\Omega}}_{zz}|\mathcal{A})+\lambda_z^2B_{zx}^2\bar{B}_{zz}n\varepsilon}\right).
\end{aligned}
$$

Applying Lemma A.2 again, we have:

$$
\begin{aligned}
&\mathbb{P}\left\{\left\|n\left(\boldsymbol{X}^\top\boldsymbol{Z}(\boldsymbol{Z}^\top\boldsymbol{Z})^{-1}\boldsymbol{Z}^\top\boldsymbol{X}\right)^{-1}-(\boldsymbol{Q}_{ZX}^\top\boldsymbol{Q}_{zz}^{-1}\boldsymbol{Q}_{zx})^{-1}\right\|\geq\varepsilon\middle|\mathcal{A}\right\}\\
&\leq 2(p+q)\exp\left(-\frac{\lambda_z^2n^2(\lambda_{\tilde{x}}^2\varepsilon)^2/2}{9B_{zx}^2\nu(\bar{\boldsymbol{\Omega}}_{zx}|\mathcal{A})+\lambda_zB_{zx}\bar{B}_{zx}n(\lambda_{\tilde{x}}^2\varepsilon)}\right)+2q\exp\left(-\frac{\lambda_z^4n^2(\lambda_{\tilde{x}}^2\varepsilon)^2/2}{9B_{zx}^4\nu(\bar{\boldsymbol{\Omega}}_{zz}|\mathcal{A})+\lambda_z^2B_{zx}^2\bar{B}_{zz}n(\lambda_{\tilde{x}}^2\varepsilon)}\right)\\
&=2(p+q)\exp\left(-\frac{\lambda_z^2\lambda_{\tilde{x}}^4n^2\varepsilon^2/2}{9B_{zx}^2\nu(\bar{\boldsymbol{\Omega}}_{zx}|\mathcal{A})+\lambda_z\lambda_{\tilde{x}}^2B_{zx}\bar{B}_{zx}n\varepsilon}\right)+2q\exp\left(-\frac{\lambda_z^4\lambda_{\tilde{x}}^4n^2\varepsilon^2/2}{9B_{zx}^4\nu(\bar{\boldsymbol{\Omega}}_{zz}|\mathcal{A})+\lambda_z^2\lambda_{\tilde{x}}^2B_{zx}^2\bar{B}_{zz}n\varepsilon}\right).
\end{aligned}
\tag{24}
$$

Therefore, we have shown that under event $\mathcal{A}$,

$$
n\left(\boldsymbol{X}^\top\boldsymbol{Z}(\boldsymbol{Z}^\top\boldsymbol{Z})^{-1}\boldsymbol{Z}^\top\boldsymbol{X}\right)^{-1}\xrightarrow{\mathsf{p}}(\boldsymbol{Q}_{ZX}^\top\boldsymbol{Q}_{zz}^{-1}\boldsymbol{Q}_{zx})^{-1}.
$$

From equation (17), combining (23) and (24) with Lemma A.3 (Remark A.1), we have:

$$
\begin{aligned}
&\mathbb{P}\left\{\left\|\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}-\boldsymbol{\beta}\right\|\geq\varepsilon\middle|\mathcal{A}\right\}\\
&=\mathbb{P}\left\{\left\|\left(\boldsymbol{X}^\top\boldsymbol{Z}(\boldsymbol{Z}^\top\boldsymbol{Z})^{-1}\boldsymbol{Z}^\top\boldsymbol{X}\right)^{-1}\boldsymbol{X}^\top\boldsymbol{Z}(\boldsymbol{Z}^\top\boldsymbol{Z})^{-1}\boldsymbol{Z}^\top\boldsymbol{\mathcal{E}}_1-\mathbf{0}\right\|\geq\varepsilon\middle|\mathcal{A}\right\}\\
&\leq(q+1)\exp\left(-\frac{n^2(\lambda_{\tilde{x}}\varepsilon)^2/2}{B_{\hat{\Theta}}^2\nu(\boldsymbol{\Omega}_{z\epsilon_1}|\mathcal{A})+B_{\hat{\Theta}}B_{z\epsilon_1}n(\lambda_{\tilde{x}}\varepsilon)/3}\right)\\
&=(q+1)\exp\left(-\frac{\lambda_{\tilde{x}}^2n^2\varepsilon^2/2}{B_{\hat{\Theta}}^2\nu(\boldsymbol{\Omega}_{z\epsilon_1}|\mathcal{A})+\lambda_{\tilde{x}}B_{\hat{\Theta}}B_{z\epsilon_1}n\varepsilon/3}\right).
\end{aligned}
$$

For the second part of the theorem, let $c:=\left(\frac{3B_{\hat{\Theta}}\nu(\boldsymbol{\Omega}_{z\epsilon_1}|\mathcal{A})}{\lambda_{\tilde{x}}B_{z\epsilon_1}n}\right)^2$, we have:

$$
\begin{aligned}
&\mathbb{E}\left[\|\mathsf{clip}_{B_\beta}(\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}})-\boldsymbol{\beta}\|^2\right]\\
&=\mathbb{E}\left[\|\mathsf{clip}_{B_\beta}(\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}})-\boldsymbol{\beta}\|^2\middle|\mathcal{A}\right]\mathbb{P}\left\{\mathcal{A}\right\}+\mathbb{E}\left[\|\mathsf{clip}_{B_\beta}(\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}})-\boldsymbol{\beta}\|^2\middle|\mathcal{A}^c\right]\mathbb{P}\left\{\mathcal{A}^c\right\}\\
&\leq\mathbb{E}\left[\|\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}-\boldsymbol{\beta}\|^2\middle|\mathcal{A}\right]\mathbb{P}\left\{\mathcal{A}\right\}+\mathbb{E}\left[\|\mathsf{clip}_{B_\beta}(\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}})-\boldsymbol{\beta}\|^2\middle|\mathcal{A}^c\right]\mathbb{P}\left\{\mathcal{A}^c\right\}\\
&\leq\mathbb{E}\left[\|\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}-\boldsymbol{\beta}\|^2\middle|\mathcal{A}\right]+\mathbb{E}\left[\|\mathsf{clip}_{B_\beta}(\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}})-\boldsymbol{\beta}\|^2\middle|\mathcal{A}^c\right]\cdot\xi,
\end{aligned}
\tag{25}
$$

where

$$
\mathbb{E}\left[\|\mathsf{clip}_{B_\beta}(\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}})-\boldsymbol{\beta}\|^2\middle|\mathcal{A}^c\right]\leq 4B_\beta^2,
\tag{26}
$$

and

$$
\begin{aligned}
&\mathbb{E}\left[\|\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}-\boldsymbol{\beta}\|^2\Big|\mathcal{A}\right] \\
&=\int_0^\infty \mathbb{P}\left\{\|\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}-\boldsymbol{\beta}\|^2\geq\varepsilon\Big|\mathcal{A}\right\}d\varepsilon \\
&=\int_0^\infty \mathbb{P}\left\{\|\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}-\boldsymbol{\beta}\|\geq\sqrt{\varepsilon}\Big|\mathcal{A}\right\}d\varepsilon \\
&\leq\int_0^\infty (q+1)\exp\left(-\frac{\lambda_{\tilde{x}}^2 n^2\varepsilon/2}{B_{\hat{\Theta}}^2\nu(\boldsymbol{\Omega}_{z\epsilon_1}|\mathcal{A})+\lambda_{\tilde{x}}B_{\hat{\Theta}}B_{z\epsilon_1}n\sqrt{\varepsilon}/3}\right)d\varepsilon \\
&\leq(q+1)\left[\int_0^c \exp\left(-\frac{\lambda_{\tilde{x}}^2 n^2\varepsilon/2}{2B_{\hat{\Theta}}^2\nu(\boldsymbol{\Omega}_{z\epsilon_1}|\mathcal{A})}\right)d\varepsilon+\int_c^\infty \exp\left(-\frac{\lambda_{\tilde{x}}n\sqrt{\varepsilon}/2}{2B_{\hat{\Theta}}B_{z\epsilon_1}/3}\right)d\varepsilon\right] \\
&=(q+1)\left[\frac{4B_{\hat{\Theta}}^2\nu(\boldsymbol{\Omega}_{z\epsilon_1}|\mathcal{A})}{\lambda_{\tilde{x}}^2 n^2}\left(1-\exp\left(-\frac{9\nu(\boldsymbol{\Omega}_{z\epsilon_1}|\mathcal{A})}{4B_{z\epsilon_1}^2}\right)\right)+\left(\frac{8B_{\hat{\Theta}}^2\nu(\boldsymbol{\Omega}_{z\epsilon_1}|\mathcal{A})}{\lambda_{\tilde{x}}^2 n^2}+\frac{32B_{\hat{\Theta}}^2B_{z\epsilon_1}^2}{9\lambda_{\tilde{x}}^2 n^2}\right)\exp\left(-\frac{9\nu(\boldsymbol{\Omega}_{z\epsilon_1}|\mathcal{A})}{4B_{z\epsilon_1}^2}\right)\right] \\
&\leq(q+1)\left[\frac{4B_{\hat{\Theta}}^2\nu(\boldsymbol{\Omega}_{z\epsilon_1}|\mathcal{A})}{\lambda_{\tilde{x}}^2 n^2}+\frac{8B_{\hat{\Theta}}^2\nu(\boldsymbol{\Omega}_{z\epsilon_1}|\mathcal{A})}{\lambda_{\tilde{x}}^2 n^2}+\frac{32B_{\hat{\Theta}}^2B_{z\epsilon_1}^2}{9\lambda_{\tilde{x}}^2 n^2}\right] \\
&=\frac{(q+1)B_{\hat{\Theta}}^2}{\lambda_{\tilde{x}}^2 n^2}\left[12\nu(\boldsymbol{\Omega}_{z\epsilon_1}|\mathcal{A})+\frac{32B_z^2B_{\epsilon_1}^2}{9}\right].
\end{aligned}
\tag{27}
$$

Note that we further have the following bound:

$$
\begin{aligned}
\nu(\boldsymbol{\Omega}_{z\epsilon_1}|\mathcal{A})&=\max\left\{\left\|\mathbb{E}\Big[\Big(\sum_{i=1}^n \boldsymbol{z}_i\epsilon_{1,i}\Big)^\top\Big(\sum_{j=1}^n \boldsymbol{z}_j\epsilon_{1,j}\Big)\Big|\mathcal{A}\Big]\right\|,\left\|\mathbb{E}\Big[\Big(\sum_{i=1}^n \boldsymbol{z}_i\epsilon_{1,i}\Big)\Big(\sum_{j=1}^n \boldsymbol{z}_j\epsilon_{1,j}\Big)^\top\Big|\mathcal{A}\Big]\right\|\right\} \\
&=\max\left\{\left\|\mathbb{E}\Big[\sum_{i=1}^n \epsilon_{1,i}^2\boldsymbol{z}_i^\top\boldsymbol{z}_i\Big|\mathcal{A}\Big]\right\|,\left\|\mathbb{E}\Big[\sum_{i=1}^n \epsilon_{1,i}^2\boldsymbol{z}_i\boldsymbol{z}_i^\top\Big|\mathcal{A}\Big]\right\|\right\} \\
&\leq\mathcal{O}(nB_z^2\sigma_1^2).
\end{aligned}
\tag{28}
$$

Plugging (26)(27)(28) into (25) completes the proof. □

B Proofs For Section 3

B.1 Proof of Theorem 3.1

Lemma B.1. Suppose $\{\boldsymbol{\Omega}^{(1)},\ldots,\boldsymbol{\Omega}^{(t)},\ldots\}$ is a $d\times d$-matrix sequence decaying with exponential rate r, i.e. for some constant $c>0$ and $0<r<1$,

$$\left\|\boldsymbol{\Omega}^{(t)}\right\|_F\leq cr^t.$$

Then for any $\varepsilon>0$, there exists a finite constant:

$$T_0=\left\lceil\log_r\frac{(1-r)(\varepsilon/d)}{c\,(1+(1-r)(\varepsilon/d))}\right\rceil,$$

such that

$$\left\|\prod_{t=T_0}^\infty(\boldsymbol{I}+\boldsymbol{\Omega}^{(t)})-\boldsymbol{I}\right\|_F<\varepsilon,$$

and hence

$$\left\|\prod_{t=T_0}^\infty(\boldsymbol{I}+\boldsymbol{\Omega}^{(t)})\right\|_F<\sqrt{d}+\varepsilon.$$

Proof. By definition,

$$\left\|\mathbf{\Omega}^{(k)}\right\|_F = \sqrt{\sum_{i,j=1}^{p} \mathbf{\Omega}_{ij}^{(k)2}} \leq cr^k,$$

which implies:

$$\left|\mathbf{\Omega}_{ij}^{(k)}\right| \leq cr^k, \quad \forall i,j,k.$$

Consider the product of any two matrices. By sub-multiplicativity,

$$\left\|\mathbf{\Omega}^{(k)}\mathbf{\Omega}^{(l)}\right\|_F \leq \left\|\mathbf{\Omega}^{(k)}\right\|_F \left\|\mathbf{\Omega}^{(l)}\right\|_F \leq c^2 r^{k+l},$$

which implies:

$$\left|\left[\mathbf{\Omega}^{(k)}\mathbf{\Omega}^{(l)}\right]_{ij}\right| \leq c^2 r^{k+l}, \quad \forall i,j,k,l.$$

Similarly, for the product of any number of matrices:

$$\left|\left[\mathbf{\Omega}^{(k_1)}\mathbf{\Omega}^{(k_2)}\cdots\mathbf{\Omega}^{(k_n)}\right]_{ij}\right| \leq c^n r^{k_1+k_2+\cdots+k_n}, \quad \forall i,j,k_1,\ldots,k_n.$$

Thus

$$\begin{aligned}
&\left\|\prod_{t=t_1}^{t_2}\left(\boldsymbol{I}+\mathbf{\Omega}^{(t)}\right)-\boldsymbol{I}\right\|_F \\
&= \left\|\left(\boldsymbol{I}+\mathbf{\Omega}^{(t_1)}\right)\left(\boldsymbol{I}+\mathbf{\Omega}^{(t_1+1)}\right)\cdots\left(\boldsymbol{I}+\mathbf{\Omega}^{(t_2)}\right)-\boldsymbol{I}\right\|_F \\
&= \left\|\sum_{t_1\leq k\leq t_2}\mathbf{\Omega}^{(k)} + \sum_{t_1\leq k<l\leq t_2}\mathbf{\Omega}^{(k)}\mathbf{\Omega}^{(l)} + \cdots + \mathbf{\Omega}^{(t_1)}\mathbf{\Omega}^{(t_1+1)}\cdots\mathbf{\Omega}^{(t_2)}\right\|_F \\
&\leq \left\|\sum_{t_1\leq k\leq t_2} cr^k\mathbf{1}\mathbf{1}^\top + \sum_{t_1\leq k<l\leq t_2} c^2r^{k+l}\mathbf{1}\mathbf{1}^\top + \cdots + c^{t2-t1+1}r^{t_1+\cdots+t_2}\mathbf{1}\mathbf{1}^\top\right\|_F.
\end{aligned} \tag{29}$$

Note that the last inequality can be checked by comparing matrix elements of both sides. For any $\varepsilon > 0$, we take $T_0 = \lceil \log_r \frac{(1-r)(\varepsilon/d)}{c(1+(1-r)(\varepsilon/d))} \rceil$. Consider $t_1 = T_0$ and $t_2 \to \infty$ in (29). For notation convenience, let

$$\mathbf{\Xi} := \sum_{T_0\leq k} cr^k\mathbf{1}\mathbf{1}^\top + \sum_{T_0\leq k<l} c^2r^{k+l}\mathbf{1}\mathbf{1}^\top + \sum_{T_0\leq k<l<m} c^3r^{k+l+m}\mathbf{1}\mathbf{1}^\top + \cdots.$$

Then

$$\begin{aligned}
\mathbf{\Xi}_{ij} &= \sum_{T_0\leq k} cr^k + \sum_{T_0\leq k<l} c^2r^{k+l} + \sum_{T_0\leq k<l<m} c^3r^{k+l+m} + \cdots \\
&< c\sum_{k\geq T_0} r^k + c^2r^{T_0}\sum_{k\geq T_0} r^k + c^3r^{2T_0}\sum_{k\geq T_0} r^k + \cdots \\
&= \frac{cr^{T_0}}{1-r} + \frac{c^2r^{2T_0}}{1-r} + \frac{c^3r^{3T_0}}{1-r} + \cdots \\
&= \frac{cr^{T_0}}{(1-r)(1-cr^{T_0})} \\
&\leq \frac{\varepsilon}{d}.
\end{aligned}$$

Thus

$$\left\| \prod_{t=T_0}^{\infty} (\boldsymbol{I} + \boldsymbol{\Omega}^{(t)}) - \boldsymbol{I} \right\|_F = \|\boldsymbol{\Xi}\|_F = \sqrt{\sum_{i,j=1}^{d} \boldsymbol{\Xi}_{ij}^2} \leq \varepsilon.$$

Hence completes the proof.

□

Proof of Theorem 3.1. In the following proof, we treat $\boldsymbol{Z}, \boldsymbol{X}, \boldsymbol{Y}$ as deterministic matrices.

We begin by checking the inner loop (8a):

$$\begin{aligned}
\boldsymbol{\Theta}^{(t)} - \hat{\boldsymbol{\Theta}} &= \boldsymbol{\Theta}^{(t-1)} - \hat{\boldsymbol{\Theta}} - \eta \boldsymbol{Z}^\top \left(\boldsymbol{Z}\boldsymbol{\Theta}^{(t-1)} - \boldsymbol{X} \right) \\
&= \left(\boldsymbol{I} - \eta \boldsymbol{Z}^\top \boldsymbol{Z} \right) \left(\boldsymbol{\Theta}^{(t-1)} - \hat{\boldsymbol{\Theta}} \right) + \eta \boldsymbol{Z}^\top \left(\boldsymbol{X} - \boldsymbol{Z}\hat{\boldsymbol{\Theta}} \right) \\
&= \left(\boldsymbol{I} - \eta \boldsymbol{Z}^\top \boldsymbol{Z} \right)^2 \left(\boldsymbol{\Theta}^{(t-2)} - \hat{\boldsymbol{\Theta}} \right) + \eta \boldsymbol{Z}^\top \left(\boldsymbol{X} - \boldsymbol{Z}\hat{\boldsymbol{\Theta}} \right) + \eta \left(\boldsymbol{I} - \eta \boldsymbol{Z}^\top \boldsymbol{Z} \right) \boldsymbol{Z}^\top \left(\boldsymbol{X} - \boldsymbol{Z}\hat{\boldsymbol{\Theta}} \right) \\
&\qquad \vdots \\
&= \left(\boldsymbol{I} - \eta \boldsymbol{Z}^\top \boldsymbol{Z} \right)^t \left(\boldsymbol{\Theta}^{(0)} - \hat{\boldsymbol{\Theta}} \right) + \sum_{i=0}^{t-1} \eta \left(\boldsymbol{I} - \eta \boldsymbol{Z}^\top \boldsymbol{Z} \right)^{t-1-i} \boldsymbol{Z}^\top \left(\boldsymbol{X} - \boldsymbol{Z}\hat{\boldsymbol{\Theta}} \right) \\
&= \left(\boldsymbol{I} - \eta \boldsymbol{Z}^\top \boldsymbol{Z} \right)^t \left(\boldsymbol{\Theta}^{(0)} - \hat{\boldsymbol{\Theta}} \right) + \eta \left[\boldsymbol{I} - (\boldsymbol{I} - \eta \boldsymbol{Z}^\top \boldsymbol{Z})^t \right] \left(\eta \boldsymbol{Z}^\top \boldsymbol{Z} \right)^{-1} \boldsymbol{Z}^\top \left(\boldsymbol{X} - \boldsymbol{Z}\hat{\boldsymbol{\Theta}} \right) \\
&= \left(\boldsymbol{I} - \eta \boldsymbol{Z}^\top \boldsymbol{Z} \right)^t \left(\boldsymbol{\Theta}^{(0)} - \hat{\boldsymbol{\Theta}} \right) + \left[\boldsymbol{I} - (\boldsymbol{I} - \eta \boldsymbol{Z}^\top \boldsymbol{Z})^t \right] \left[(\boldsymbol{Z}^\top \boldsymbol{Z})^{-1} \boldsymbol{Z}^\top \boldsymbol{X} - \hat{\boldsymbol{\Theta}} \right] \\
&= \left(\boldsymbol{I} - \eta \boldsymbol{Z}^\top \boldsymbol{Z} \right)^t \left(\boldsymbol{\Theta}^{(0)} - \hat{\boldsymbol{\Theta}} \right).
\end{aligned}$$

With learning rate $0 < \eta < \frac{2}{\sigma_{\max}^2(\boldsymbol{Z})}$, let $\kappa(\eta) := \rho\left(\boldsymbol{I} - \eta \boldsymbol{Z}^\top \boldsymbol{Z}\right)$, where $\rho(\cdot)$ denotes the spectral radius. Then it follows that $0 < \kappa(\eta) < 1$. We have:

$$\begin{aligned}
\left\| \boldsymbol{\Theta}^{(t)} - \hat{\boldsymbol{\Theta}} \right\| &= \left\| (\boldsymbol{I} - \eta \boldsymbol{Z}^\top \boldsymbol{Z})^t (\boldsymbol{\Theta}^{(0)} - \hat{\boldsymbol{\Theta}}) \right\| \\
&\leq \left\| (\boldsymbol{I} - \eta \boldsymbol{Z}^\top \boldsymbol{Z})^t \right\| \left\| \boldsymbol{\Theta}^{(0)} - \hat{\boldsymbol{\Theta}} \right\| \\
&\leq \kappa(\eta)^t \left\| \boldsymbol{\Theta}^{(0)} - \hat{\boldsymbol{\Theta}} \right\| \\
&= \mathcal{O}(\kappa(\eta)^t).
\end{aligned}$$

Thus $\{\boldsymbol{\Theta}^{(t)}\}$ converges to $\hat{\boldsymbol{\Theta}}$ exponentially with rate $\kappa(\eta)$.

For the outer loop (8b), we have:

$$\begin{aligned}
\boldsymbol{\beta}^{(t)} - \hat{\boldsymbol{\beta}}_{\mathsf{2SLS}} &= \boldsymbol{\beta}^{(t-1)} - \hat{\boldsymbol{\beta}}_{\mathsf{2SLS}} - \alpha \boldsymbol{\Theta}^{(t-1)\top} \boldsymbol{Z}^\top \left(\boldsymbol{Z}\boldsymbol{\Theta}^{(t-1)} \boldsymbol{\beta}^{(t-1)} - \boldsymbol{Y} \right) \\
&= \left(\boldsymbol{I} - \alpha \boldsymbol{\Theta}^{(t-1)\top} \boldsymbol{Z}^\top \boldsymbol{Z} \boldsymbol{\Theta}^{(t-1)} \right) \left(\boldsymbol{\beta}^{(t-1)} - \hat{\boldsymbol{\beta}}_{\mathsf{2SLS}} \right) + \alpha \boldsymbol{\Theta}^{(t-1)\top} \boldsymbol{Z}^\top \left(\boldsymbol{Y} - \boldsymbol{Z}\boldsymbol{\Theta}^{(t-1)} \hat{\boldsymbol{\beta}}_{\mathsf{2SLS}} \right) \\
&\qquad \vdots \\
&= \underbrace{\prod_{i=0}^{t-1} \left(\boldsymbol{I} - \alpha \boldsymbol{\Theta}^{(i)\top} \boldsymbol{Z}^\top \boldsymbol{Z} \boldsymbol{\Theta}^{(i)} \right) \left(\boldsymbol{\beta}^{(0)} - \hat{\boldsymbol{\beta}}_{\mathsf{2SLS}} \right)}_{\boldsymbol{\Delta}_1 \boldsymbol{\beta}^{(t)}} \\
&\quad + \underbrace{\sum_{i=0}^{t-1} \alpha \left[\prod_{j=i+1}^{t-1} \left(\boldsymbol{I} - \alpha \boldsymbol{\Theta}^{(j)\top} \boldsymbol{Z}^\top \boldsymbol{Z} \boldsymbol{\Theta}^{(j)} \right) \right] \boldsymbol{\Theta}^{(i)\top} \boldsymbol{Z}^\top \left(\boldsymbol{Y} - \boldsymbol{Z}\boldsymbol{\Theta}^{(i)} \hat{\boldsymbol{\beta}}_{\mathsf{2SLS}} \right)}_{\boldsymbol{\Delta}_2 \boldsymbol{\beta}^{(t)}}.
\end{aligned} \tag{30}$$

To simplify notations, let

$$\begin{aligned}
\boldsymbol{R}^{(t)} &:= \boldsymbol{\Theta}^{(t)} - \hat{\boldsymbol{\Theta}} = \left(\boldsymbol{I} - \eta \boldsymbol{Z}^\top \boldsymbol{Z}\right)^t \left(\boldsymbol{\Theta}^{(0)} - \hat{\boldsymbol{\Theta}}\right), \\
\boldsymbol{V}^{(t)} &:= \left(\boldsymbol{I} - \alpha \hat{\boldsymbol{\Theta}}^\top \boldsymbol{Z}^\top \boldsymbol{Z} \hat{\boldsymbol{\Theta}}\right)^t, \\
\boldsymbol{W}^{(t)} &:= \boldsymbol{R}^{(t)\top} \boldsymbol{Z}^\top \boldsymbol{Z} \hat{\boldsymbol{\Theta}} + \hat{\boldsymbol{\Theta}}^\top \boldsymbol{Z}^\top \boldsymbol{Z} \boldsymbol{R}^{(t)} + \boldsymbol{R}^{(t)\top} \boldsymbol{Z}^\top \boldsymbol{Z} \boldsymbol{R}^{(t)}.
\end{aligned}$$

With learning rates $0 < \alpha < \frac{2}{\sigma_{\max}^2(\boldsymbol{Z}\hat{\boldsymbol{\Theta}})}$, $0 < \eta < \frac{2}{\sigma_{\max}^2(\boldsymbol{Z})}$, let $\gamma(\alpha) := \rho\left(\boldsymbol{I} - \alpha \hat{\boldsymbol{\Theta}}^\top \boldsymbol{Z}^\top \boldsymbol{Z} \hat{\boldsymbol{\Theta}}\right)$. Then it follows that $0 < \gamma(\alpha) < 1$. We have:

$$\left\|\boldsymbol{R}^{(t)}\right\| \leq \kappa(\eta)^t \left\|\boldsymbol{\Theta}^{(0)} - \hat{\boldsymbol{\Theta}}\right\|, \tag{31}$$

$$\left\|\boldsymbol{V}^{(t)}\right\| \leq \gamma(\alpha)^t,$$

and

$$\begin{aligned}
\left\|\boldsymbol{W}^{(t)}\right\| &= \left\|\boldsymbol{R}^{(t)\top} \boldsymbol{Z}^\top \boldsymbol{Z} \hat{\boldsymbol{\Theta}} + \hat{\boldsymbol{\Theta}}^\top \boldsymbol{Z}^\top \boldsymbol{Z} \boldsymbol{R}^{(t)} + \boldsymbol{R}^{(t)\top} \boldsymbol{Z}^\top \boldsymbol{Z} \boldsymbol{R}^{(t)}\right\| \\
&\leq 2 \left\|\hat{\boldsymbol{\Theta}}^\top \boldsymbol{Z}^\top \boldsymbol{Z}\right\| \left\|\boldsymbol{R}^{(t)}\right\| + \left\|\boldsymbol{Z}^\top \boldsymbol{Z}\right\| \left\|\boldsymbol{R}^{(t)}\right\|^2 \\
&\leq 2\kappa(\eta)^t \left\|\hat{\boldsymbol{\Theta}} \boldsymbol{Z}^\top \boldsymbol{Z}\right\| \left\|\boldsymbol{\Theta}^{(0)} - \hat{\boldsymbol{\Theta}}\right\| + \kappa(\eta)^{2t} \left\|\boldsymbol{Z}^\top \boldsymbol{Z}\right\| \\
&\leq \kappa(\eta)^t \left(2 \left\|\hat{\boldsymbol{\Theta}} \boldsymbol{Z}^\top \boldsymbol{Z}\right\| \left\|\boldsymbol{\Theta}^{(0)} - \hat{\boldsymbol{\Theta}}\right\| + \left\|\boldsymbol{Z}^\top \boldsymbol{Z}\right\| \left\|\boldsymbol{\Theta}^{(0)} - \hat{\boldsymbol{\Theta}}\right\|^2\right) \\
&= \mathcal{O}(\kappa(\eta)^t).
\end{aligned}$$

Then from equation (30), we have:

$$\begin{aligned}
\boldsymbol{\Delta}_1 \boldsymbol{\beta}^{(t)} &= \prod_{i=0}^{t-1} \left(\boldsymbol{I} - \alpha \boldsymbol{\Theta}^{(i)\top} \boldsymbol{Z}^\top \boldsymbol{Z} \boldsymbol{\Theta}^{(i)}\right) \left(\boldsymbol{\beta}^{(0)} - \hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}\right) \\
&= \prod_{i=0}^{t-1} \left[\boldsymbol{I} - \alpha \hat{\boldsymbol{\Theta}}^\top \boldsymbol{Z}^\top \boldsymbol{Z} \hat{\boldsymbol{\Theta}} - \alpha \left(\boldsymbol{R}^{(i)\top} \boldsymbol{Z}^\top \boldsymbol{Z} \hat{\boldsymbol{\Theta}} + \hat{\boldsymbol{\Theta}}^\top \boldsymbol{Z}^\top \boldsymbol{Z} \boldsymbol{R}^{(i)} + \boldsymbol{R}^{(i)\top} \boldsymbol{Z}^\top \boldsymbol{Z} \boldsymbol{R}^{(i)}\right)\right] \left(\boldsymbol{\beta}^{(0)} - \hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}\right) \\
&= \prod_{i=0}^{t-1} \left[\boldsymbol{I} - \alpha \hat{\boldsymbol{\Theta}}^\top \boldsymbol{Z}^\top \boldsymbol{Z} \hat{\boldsymbol{\Theta}} - \alpha \boldsymbol{W}^{(i)}\right] \left(\boldsymbol{\beta}^{(0)} - \hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}\right) \\
&= \left(\boldsymbol{I} - \alpha \hat{\boldsymbol{\Theta}}^\top \boldsymbol{Z}^\top \boldsymbol{Z} \hat{\boldsymbol{\Theta}}\right)^t \prod_{i=0}^{t-1} \left[\boldsymbol{I} - \alpha \left(\boldsymbol{I} - \alpha \hat{\boldsymbol{\Theta}}^\top \boldsymbol{Z}^\top \boldsymbol{Z} \hat{\boldsymbol{\Theta}}\right)^{-1} \boldsymbol{W}^{(i)}\right] \left(\boldsymbol{\beta}^{(0)} - \hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}\right) \\
&= \boldsymbol{V}^{(t)} \prod_{i=0}^{t-1} \left[\boldsymbol{I} - \alpha \left(\boldsymbol{I} - \alpha \hat{\boldsymbol{\Theta}}^\top \boldsymbol{Z}^\top \boldsymbol{Z} \hat{\boldsymbol{\Theta}}\right)^{-1} \boldsymbol{W}^{(i)}\right] \left(\boldsymbol{\beta}^{(0)} - \hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}\right).
\end{aligned}$$

We denote $\boldsymbol{\Psi} := \alpha \left(\boldsymbol{I} - \alpha \hat{\boldsymbol{\Theta}}^\top \boldsymbol{Z}^\top \boldsymbol{Z} \hat{\boldsymbol{\Theta}}\right)^{-1}$. By Lemma B.1, we take $\varepsilon = 1$, c_0 be a constant such that $\left\|\boldsymbol{W}^{(t)}\right\|_F \leq c_0 \kappa(\eta)^t$, and $T_0 = \lceil \log_{\kappa(\eta)} \frac{(1-\kappa(\eta))}{\|\Psi\|_F c_0 (p + (1-\kappa(\eta)))} \rceil$.

Then we have:

$$\left\|\prod_{i=T_0}^{t-1} \left(\boldsymbol{I} - \boldsymbol{\Psi} \boldsymbol{W}^{(i)}\right)\right\| \leq \left\|\prod_{i=T_0}^{t-1} \left(\boldsymbol{I} - \boldsymbol{\Psi} \boldsymbol{W}^{(i)}\right)\right\|_F < \sqrt{p} + 1. \tag{32}$$

Hence

$$
\begin{aligned}
\left\|\boldsymbol{\Delta}_1\boldsymbol{\beta}^{(t)}\right\| &= \left\|\boldsymbol{V}^{(t)}\prod_{i=0}^{t-1}(\boldsymbol{I}-\boldsymbol{\Psi}\boldsymbol{W}^{(i)})(\boldsymbol{\beta}^{(0)}-\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}})\right\| \\
&\leq \left\|\boldsymbol{V}^{(t)}\right\|\left\|\prod_{i=0}^{T_0-1}\left(\boldsymbol{I}-\boldsymbol{\Psi}\boldsymbol{W}^{(i)}\right)\right\|\left\|\prod_{i=T_0}^{t-1}(\boldsymbol{I}-\boldsymbol{\Psi}\boldsymbol{W}^{(i)})\right\|\left\|\boldsymbol{\beta}^{(0)}-\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}\right\| \\
&< \gamma(\alpha)^t\left\|\prod_{i=0}^{T_0-1}\left(\boldsymbol{I}-\boldsymbol{\Psi}\boldsymbol{W}^{(i)}\right)\right\|(\sqrt{p}+1)\left\|\boldsymbol{\beta}^{(0)}-\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}\right\| \\
&= \mathcal{O}(\gamma(\alpha)^t).
\end{aligned}
\tag{33}
$$

Next we consider $\boldsymbol{\Delta}_2\boldsymbol{\beta}^{(t)}$:

$$
\begin{aligned}
\boldsymbol{\Delta}_2\boldsymbol{\beta}^{(t)} &= \sum_{i=0}^{t-1}\alpha\left[\prod_{j=i+1}^{t-1}\left(\boldsymbol{I}-\alpha\boldsymbol{\Theta}^{(j)\top}\boldsymbol{Z}^\top\boldsymbol{Z}\boldsymbol{\Theta}^{(j)}\right)\right]\boldsymbol{\Theta}^{(i)\top}\boldsymbol{Z}^\top\left(\boldsymbol{Y}-\boldsymbol{Z}\boldsymbol{\Theta}^{(i)}\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}\right) \\
&= \sum_{i=0}^{t-1}\alpha\left[\prod_{j=i+1}^{t-1}\left(\boldsymbol{I}-\alpha\hat{\boldsymbol{\Theta}}^\top\boldsymbol{Z}^\top\boldsymbol{Z}\hat{\boldsymbol{\Theta}}-\alpha\boldsymbol{W}^{(j)}\right)\right]\left(\boldsymbol{R}^{(i)}+\hat{\boldsymbol{\Theta}}\right)^\top\boldsymbol{Z}^\top\left[\boldsymbol{Y}-\boldsymbol{Z}\left(\boldsymbol{R}^{(i)}+\hat{\boldsymbol{\Theta}}\right)\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}\right] \\
&= \sum_{i=0}^{t-1}\alpha\left(\boldsymbol{I}-\alpha\hat{\boldsymbol{\Theta}}^\top\boldsymbol{Z}^\top\boldsymbol{Z}\hat{\boldsymbol{\Theta}}\right)^{t-1-i}\prod_{j=i+1}^{t-1}\left[\boldsymbol{I}-\alpha\left(\boldsymbol{I}-\alpha\hat{\boldsymbol{\Theta}}^\top\boldsymbol{Z}^\top\boldsymbol{Z}\hat{\boldsymbol{\Theta}}\right)^{-1}\boldsymbol{W}^{(j)}\right] \\
&\qquad\cdot\left(\boldsymbol{R}^{(i)}+\hat{\boldsymbol{\Theta}}\right)^\top\boldsymbol{Z}^\top\left[\boldsymbol{Y}-\boldsymbol{Z}\left(\boldsymbol{R}^{(i)}+\hat{\boldsymbol{\Theta}}\right)\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}\right] \\
&= \sum_{i=0}^{t-1}\alpha\left[\boldsymbol{V}^{(t-1-i)}\prod_{j=i+1}^{t-1}\left(\boldsymbol{I}-\boldsymbol{\Psi}\boldsymbol{W}^{(j)}\right)\right]\left(\boldsymbol{R}^{(i)}+\hat{\boldsymbol{\Theta}}\right)^\top\boldsymbol{Z}^\top\left[\boldsymbol{Y}-\boldsymbol{Z}\left(\boldsymbol{R}^{(i)}+\hat{\boldsymbol{\Theta}}\right)\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}\right].
\end{aligned}
$$

For convenience, let $\boldsymbol{\Delta}_2\boldsymbol{\beta}^{(t)} := \boldsymbol{\Delta}_{21}\boldsymbol{\beta}^{(t)} + \boldsymbol{\Delta}_{22}\boldsymbol{\beta}^{(t)}$, where

$$
\begin{aligned}
\boldsymbol{\Delta}_{21}\boldsymbol{\beta}^{(t)} &:= \sum_{i=0}^{t-1}\alpha\left[\boldsymbol{V}^{(t-1-i)}\prod_{j=i+1}^{t-1}\left(\boldsymbol{I}-\boldsymbol{\Psi}\boldsymbol{W}^{(j)}\right)\right]\boldsymbol{R}^{(i)\top}\boldsymbol{Z}^\top\left[\boldsymbol{Y}-\boldsymbol{Z}\left(\boldsymbol{R}^{(i)}+\hat{\boldsymbol{\Theta}}\right)\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}\right], \\
\boldsymbol{\Delta}_{22}\boldsymbol{\beta}^{(t)} &:= \sum_{i=0}^{t-1}\alpha\left[\boldsymbol{V}^{(t-1-i)}\prod_{j=i+1}^{t-1}\left(\boldsymbol{I}-\boldsymbol{\Psi}\boldsymbol{W}^{(j)}\right)\right]\hat{\boldsymbol{\Theta}}^\top\boldsymbol{Z}^\top\left[\boldsymbol{Y}-\boldsymbol{Z}\left(\boldsymbol{R}^{(i)}+\hat{\boldsymbol{\Theta}}\right)\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}\right].
\end{aligned}
$$

Suppose $\tilde{M}_1, \tilde{M}_2$ are the upper bounds such that

$$
\left\|\boldsymbol{Z}^\top\left[\boldsymbol{Y}-\boldsymbol{Z}\left(\boldsymbol{R}^{(i)}+\hat{\boldsymbol{\Theta}}\right)\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}\right]\right\| \leq \tilde{M}_1, \quad \forall i=0,\ldots,t-1,
$$

$$
\left\|\prod_{j=i+1}^{t-1}\left(\boldsymbol{I}-\boldsymbol{\Psi}\boldsymbol{W}^{(j)}\right)\right\| \leq \tilde{M}_2, \quad \forall i=0,\ldots,t-1.
$$

We know such $\tilde{M}_1, \tilde{M}_2$ exist because of the bounds given by (31) and (32). Let $\tilde{M} = \tilde{M}_1\tilde{M}_2$. Then

$$\begin{aligned}
\left\|\boldsymbol{\Delta}_{21}\boldsymbol{\beta}^{(t)}\right\| &\leq \tilde{M}\left\|\sum_{i=0}^{t-1}\alpha\boldsymbol{V}^{(t-1-i)}\boldsymbol{R}^{(i)\top}\right\| \\
&\leq \tilde{M}\alpha\sum_{i=0}^{t-1}\left\|\boldsymbol{V}^{(t-1-i)}\right\|\left\|\boldsymbol{R}^{(i)}\right\| \\
&\leq \tilde{M}\alpha\left\|\boldsymbol{\Theta}^{(0)}-\hat{\boldsymbol{\Theta}}\right\|\sum_{i=0}^{t-1}\gamma(\alpha)^{t-1-i}\kappa(\eta)^{i} \\
&= \tilde{M}\alpha\left\|\boldsymbol{\Theta}^{(0)}-\hat{\boldsymbol{\Theta}}\right\|\sum_{i=0}^{t-1}\gamma(\alpha)^{t-1}\left(\frac{\kappa(\eta)}{\gamma(\alpha)}\right)^{i} \\
&= \mathcal{O}\left(\frac{\gamma(\alpha)^t-\kappa(\eta)^t}{\gamma(\alpha)-\kappa(\eta)}\right) \\
&\leq \mathcal{O}(\max\{\gamma(\alpha)^t,\kappa(\eta)^t\}),
\end{aligned}$$

and similarly,

$$\begin{aligned}
\left\|\boldsymbol{\Delta}_{22}\boldsymbol{\beta}^{(t)}\right\| &\leq \tilde{M}\left\|\sum_{i=0}^{t-1}\alpha\boldsymbol{V}^{(t-1-i)}\hat{\boldsymbol{\Theta}}^{T}\right\| \\
&\leq \tilde{M}\alpha\left\|\hat{\boldsymbol{\Theta}}\right\|\sum_{i=0}^{t-1}\left\|\boldsymbol{V}^{(t-1-i)}\right\| \\
&\leq \tilde{M}\alpha\left\|\hat{\boldsymbol{\Theta}}\right\|\sum_{i=0}^{t-1}\gamma(\alpha)^{t-1-i} \\
&= \mathcal{O}(\gamma(\alpha)^t).
\end{aligned}$$

Thus

$$\begin{aligned}
\left\|\boldsymbol{\Delta}_{2}\boldsymbol{\beta}^{(t)}\right\| &= \left\|\boldsymbol{\Delta}_{21}\boldsymbol{\beta}^{(t)}+\boldsymbol{\Delta}_{22}\boldsymbol{\beta}^{(t)}\right\| \\
&\leq \left\|\boldsymbol{\Delta}_{21}\boldsymbol{\beta}^{(t)}\right\|+\left\|\boldsymbol{\Delta}_{22}\boldsymbol{\beta}^{(t)}\right\| \\
&\leq \mathcal{O}(\max\{\gamma(\alpha)^t,\kappa(\eta)^t\}).
\end{aligned} \tag{34}$$

Therefore, plugging (33) and (34) into (30), we have:

$$\left\|\boldsymbol{\beta}^{(t)}-\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}\right\| \leq \mathcal{O}(\max\{\gamma(\alpha)^t,\kappa(\eta)^t\}),$$

Hence completes the proof. □

B.2 Proof of Theorem 3.2

Proof of Theorem 3.2. For ease of notations, we ignore l in the following proof. Consider the input matrix taking the form:

$$\boldsymbol{H}^{(0)} = \begin{bmatrix}
\boldsymbol{z}_1 & \cdots & \boldsymbol{z}_n & \boldsymbol{z}_{n+1} \\
\boldsymbol{x}_1 & \cdots & \boldsymbol{x}_n & \boldsymbol{x}_{n+1} \\
y_1 & \cdots & y_n & 0 \\
\boldsymbol{\Theta}^{(0)}_{:,1} & \cdots & \boldsymbol{\Theta}^{(0)}_{:,1} & \boldsymbol{\Theta}^{(0)}_{:,1} \\
\vdots & \vdots & \vdots & \vdots \\
\boldsymbol{\Theta}^{(0)}_{:,p} & \cdots & \boldsymbol{\Theta}^{(0)}_{:,p} & \boldsymbol{\Theta}^{(0)}_{:,p} \\
\boldsymbol{\beta}^{(0)} & \cdots & \boldsymbol{\beta}^{(0)} & \boldsymbol{\beta}^{(0)} \\
\hat{\boldsymbol{x}}^{(0)}_1 & \cdots & \hat{\boldsymbol{x}}^{(0)}_n & \hat{\boldsymbol{x}}^{(0)}_{n+1} \\
1 & \cdots & 1 & 1 \\
1 & \cdots & 1 & 0
\end{bmatrix} \in \mathbb{R}^{D\times(n+1)},$$

i.e., element-wise,

$$\boldsymbol{h}_i^{(0)} = \left(\boldsymbol{z}_i, \boldsymbol{x}_i, y_i t_i, \boldsymbol{\Theta}_{:,1}^{(0)}, \ldots, \boldsymbol{\Theta}_{:,p}^{(0)}, \boldsymbol{\beta}^{(0)}, \hat{\boldsymbol{x}}_i^{(0)}, 1, t_i\right)^\top, \quad i = 1, \ldots, n+1,$$

where $D = qp + 3p + q + 3$, $t_i := \mathbb{1}\{i \le n\}$ is the indicator for training sample. We can take any initialization for $\boldsymbol{\Theta}^{(0)}$, $\boldsymbol{\beta}^{(0)}$ and $\hat{\boldsymbol{x}}^{(0)}$. To avoid abuse of notations, we omit the superscript of those parameters to be updated in the following proof.

Recall the definitions (5) and (6). Our goal is to show that there exists a series of attention parameters $\boldsymbol{\theta}_{\mathsf{ATTN}}^{(1:2)} = \{(\boldsymbol{Q}_m^{(1:2)}, \boldsymbol{K}_m^{(1:2)}, \boldsymbol{V}_m^{(1:2)})\}_{m\in[M]} \subset \mathbb{R}^{D\times D}$ such that $\boldsymbol{\theta}_{\mathsf{ATTN}}^{(1:2)}$ updates $\boldsymbol{\Theta}, \boldsymbol{\beta}$ on the corresponding rows. i.e, if we denote $D_0 := q + p + 1$, the updates on row $D_0 + 1$ to row $D_0 + qp$ correspond to $\boldsymbol{\Theta}$, and the updates on row $D_0 + qp + 1$ to row $D_0 + qp + p$ correspond to $\boldsymbol{\beta}$.

1) In the first layer, the transformer updates the current first-stage estimate $\hat{\boldsymbol{x}}$.

For $m = 2k-1, k = 1, \ldots, p$, define $\boldsymbol{Q}_m^{(1)}, \boldsymbol{K}_m^{(1)}, \boldsymbol{V}_m^{(1)}$ such that:

$$\boldsymbol{Q}_m^{(1)}\boldsymbol{h}_i^{(0)} = \begin{bmatrix} z_{i1} \\ \vdots \\ z_{iq} \\ \hat{x}_{ik}^{(0)} \\ \mathbf{0} \end{bmatrix}, \boldsymbol{K}_m^{(1)}\boldsymbol{h}_j^{(0)} = \begin{bmatrix} \boldsymbol{\Theta}_{1k}^{(0)} \\ \vdots \\ \boldsymbol{\Theta}_{qk}^{(0)} \\ -1 \\ \mathbf{0} \end{bmatrix}, \boldsymbol{V}_m^{(1)}\boldsymbol{h}_j^{(0)} = \boldsymbol{e}_{D_0+qp+p+k}. \tag{35}$$

For $m = 2k, k = 1, \ldots, p$, define $\boldsymbol{Q}_m^{(1)}, \boldsymbol{K}_m^{(1)}, \boldsymbol{V}_m^{(1)}$ such that:

$$\boldsymbol{Q}_m^{(1)}\boldsymbol{h}_i^{(0)} = \begin{bmatrix} -z_{i1} \\ \vdots \\ -z_{iq} \\ \hat{x}_{ik}^{(0)} \\ \mathbf{0} \end{bmatrix}, \boldsymbol{K}_m^{(1)}\boldsymbol{h}_j^{(0)} = \begin{bmatrix} \boldsymbol{\Theta}_{1k}^{(0)} \\ \vdots \\ \boldsymbol{\Theta}_{qk}^{(0)} \\ 1 \\ \mathbf{0} \end{bmatrix}, \boldsymbol{V}_m^{(1)}\boldsymbol{h}_j^{(0)} = -\boldsymbol{e}_{D_0+qp+p+k}, \tag{36}$$

where $\boldsymbol{e}_j \in \mathbb{R}^D$ is the standard unit vector with only one 1 at the j-th coordinate. Note that the above are just linear transformations on $\boldsymbol{h}_i$ or $\boldsymbol{h}_j$, hence such matrices $\boldsymbol{Q}_m^{(1)}, \boldsymbol{K}_m^{(1)}, \boldsymbol{V}_m^{(1)}$ must exist. Then we have:

$$\begin{aligned}
\boldsymbol{h}_i^{(1)} &= \boldsymbol{h}_i^{(0)} + \sum_{m=1}^{2p} \frac{1}{n+1} \sum_{j=1}^{n+1} \sigma\left(\langle \boldsymbol{Q}_m^{(1)}\boldsymbol{h}_i^{(0)}, \boldsymbol{K}_m^{(1)}\boldsymbol{h}_j^{(0)}\rangle\right) \cdot \boldsymbol{V}_m^{(1)}\boldsymbol{h}_j^{(0)} \\
&= \boldsymbol{h}_i^{(0)} + \sum_{k=1}^{p} \frac{1}{n+1} \sum_{j=1}^{n+1} \left[\sigma\left(\sum_{l=1}^{q} z_{il}\boldsymbol{\Theta}_{lk}^{(0)} - \hat{x}_{ik}^{(0)}\right) - \sigma\left(-\sum_{l=1}^{q} z_{il}\boldsymbol{\Theta}_{lk}^{(0)} + \hat{x}_{ik}^{(0)}\right)\right] \cdot \boldsymbol{e}_{D_0+qp+p+k} \\
&= \boldsymbol{h}_i^{(0)} + \sum_{k=1}^{p} \left[\sum_{l=1}^{q} z_{il}\boldsymbol{\Theta}_{lk}^{(0)} - \hat{x}_{ik}^{(0)}\right] \boldsymbol{e}_{D_0+qp+p+k} \\
&= \boldsymbol{h}_i^{(0)} + \sum_{k=1}^{p} \left(\hat{x}_{ik}^{(1)} - \hat{x}_{ik}^{(0)}\right) \boldsymbol{e}_{D_0+qp+p+k}.
\end{aligned}$$

Thus this layer correctly updates the first-stage prediction values $\hat{\boldsymbol{x}}_1^{(1)}, \ldots, \hat{\boldsymbol{x}}_{n+1}^{(1)}$, where $\hat{\boldsymbol{x}}_i^{(1)} := [\boldsymbol{Z}\boldsymbol{\Theta}^{(0)}]_{i,:} = \sum_{l=1}^{q} z_{il}\boldsymbol{\Theta}_{l,:}^{(0)}$. We will have:

$$\boldsymbol{H}^{(1)} = \begin{bmatrix} \boldsymbol{z}_1 & \cdots & \boldsymbol{z}_n & \boldsymbol{z}_{n+1} \\ \boldsymbol{x}_1 & \cdots & \boldsymbol{x}_n & \boldsymbol{x}_{n+1} \\ y_1 & \cdots & y_n & 0 \\ \boldsymbol{\Theta}_{:,1}^{(0)} & \cdots & \boldsymbol{\Theta}_{:,1}^{(0)} & \boldsymbol{\Theta}_{:,1}^{(0)} \\ \vdots & \vdots & \vdots & \vdots \\ \boldsymbol{\Theta}_{:,p}^{(0)} & \cdots & \boldsymbol{\Theta}_{:,p}^{(0)} & \boldsymbol{\Theta}_{:,p}^{(0)} \\ \boldsymbol{\beta}^{(0)} & \cdots & \boldsymbol{\beta}^{(0)} & \boldsymbol{\beta}^{(0)} \\ \hat{\boldsymbol{x}}_1^{(1)} & \cdots & \hat{\boldsymbol{x}}_n^{(1)} & \hat{\boldsymbol{x}}_{n+1}^{(1)} \\ 1 & \cdots & 1 & 1 \\ 1 & \cdots & 1 & 0 \end{bmatrix}.$$

2) In the second layer, the transformer does the gradient updates on the parameters $\boldsymbol{\Theta}$ and $\boldsymbol{\beta}$.

For $m = 2k-1, k = 1, \ldots, p$, define $\boldsymbol{Q}_m^{(2)}, \boldsymbol{K}_m^{(2)}, \boldsymbol{V}_m^{(2)}$ such that:

$$\boldsymbol{Q}_m^{(2)}\boldsymbol{h}_i^{(1)} = \begin{bmatrix} \boldsymbol{\Theta}_{:,k}^{(0)} \\ -1 \\ -1 \\ \vdots \\ \boldsymbol{0} \end{bmatrix}, \boldsymbol{K}_m^{(2)}\boldsymbol{h}_j^{(1)} = \begin{bmatrix} \boldsymbol{z}_j \\ x_{jk}t_j \\ R(1-t_j) \\ \vdots \\ \boldsymbol{0} \end{bmatrix}, \boldsymbol{V}_m^{(2)}\boldsymbol{h}_j^{(1)} = -(n+1)\eta \sum_{l=1}^{q} z_{jl}\boldsymbol{e}_{D_0+(k-1)q+l}. \tag{37}$$

For $m = 2k, k = 1, \ldots, p$, define $\boldsymbol{Q}_m^{(2)}, \boldsymbol{K}_m^{(2)}, \boldsymbol{V}_m^{(2)}$ such that:

$$\boldsymbol{Q}_m^{(2)}\boldsymbol{h}_i^{(1)} = \begin{bmatrix} -\boldsymbol{\Theta}_{:,k}^{(0)} \\ 1 \\ -1 \\ \vdots \\ \boldsymbol{0} \end{bmatrix}, \boldsymbol{K}_m^{(2)}\boldsymbol{h}_j^{(1)} = \begin{bmatrix} \boldsymbol{z}_j \\ x_{jk}t_j \\ R(1-t_j) \\ \vdots \\ \boldsymbol{0} \end{bmatrix}, \boldsymbol{V}_m^{(2)}\boldsymbol{h}_j^{(1)} = (n+1)\eta \sum_{l=1}^{q} z_{jl}\boldsymbol{e}_{D_0+(k-1)q+l}, \tag{38}$$

where $R = \max_{\substack{i=1,\ldots,n+1 \\ t=0,1,\ldots}} \{||\boldsymbol{\Theta}^{(t)\top}\boldsymbol{z}_i||\}$. Then we have:

$$\begin{aligned} \sigma\left(\langle \boldsymbol{Q}_{2k-1}^{(2)}\boldsymbol{h}_i^{(1)}, \boldsymbol{K}_{2k-1}^{(2)}\boldsymbol{h}_j^{(1)}\rangle\right) &= \sigma\left(\boldsymbol{\Theta}_{:,k}^{(0)\top}\boldsymbol{z}_j - x_{jk}t_j - R(1-t_j)\right) \\ &= \sigma\left(\boldsymbol{\Theta}_{:,k}^{(0)\top}\boldsymbol{z}_j - x_{jk}\right)\mathbb{1}\{t_j = 1\} \\ &= \sigma\left(\boldsymbol{\Theta}_{:,k}^{(0)\top}\boldsymbol{z}_j - x_{jk}\right)t_j, \end{aligned}$$

and

$$\begin{aligned} \sigma\left(\langle \boldsymbol{Q}_{2k}^{(2)}\boldsymbol{h}_i^{(1)}, \boldsymbol{K}_{2k}^{(2)}\boldsymbol{h}_j^{(1)}\rangle\right) &= \sigma\left(-\boldsymbol{\Theta}_{:,k}^{(0)\top}\boldsymbol{z}_j + x_{jk}t_j - R(1-t_j)\right) \\ &= \sigma\left(-\boldsymbol{\Theta}_{:,k}^{(0)\top}\boldsymbol{z}_j + x_{jk}\right)\mathbb{1}\{t_j = 1\} \\ &= \sigma\left(-\boldsymbol{\Theta}_{:,k}^{(0)\top}\boldsymbol{z}_j + x_{jk}\right)t_j. \end{aligned}$$

So that

$$\begin{aligned} &\sum_{m=1}^{2p} \sigma\left(\langle \boldsymbol{Q}_m^{(2)}\boldsymbol{h}_i^{(1)}, \boldsymbol{K}_m^{(2)}\boldsymbol{h}_j^{(1)}\rangle\right)\boldsymbol{V}_m^{(2)}\boldsymbol{h}_j^{(1)} \\ &= -(n+1)t_j\eta \sum_{k=1}^{p}\left[\sigma\left(\boldsymbol{\Theta}_{:,k}^{(0)\top}\boldsymbol{z}_j - x_{jk}\right) - \sigma\left(-\boldsymbol{\Theta}_{:,k}^{(0)\top}\boldsymbol{z}_j + x_{jk}\right)\right] \cdot \sum_{l=1}^{q} z_{jl}\boldsymbol{e}_{D_0+(k-1)q+l} \\ &= -(n+1)t_j\eta \sum_{k=1}^{p}\sum_{l=1}^{q} z_{jl}\left(\boldsymbol{\Theta}_{:,k}^{(0)\top}\boldsymbol{z}_j - x_{jk}\right)\boldsymbol{e}_{D_0+(k-1)q+l}. \end{aligned}$$

Similarly, for $m = 2p+1, 2p+2$, define $\boldsymbol{Q}_m^{(2)}, \boldsymbol{K}_m^{(2)}, \boldsymbol{V}_m^{(2)}$ such that:

$$\boldsymbol{Q}_{2p+1}^{(2)}\boldsymbol{h}_i^{(1)} = \begin{bmatrix} \boldsymbol{\beta}^{(0)} \\ -1 \\ -1 \\ \vdots \\ \mathbf{0} \end{bmatrix}, \boldsymbol{K}_{2p+1}^{(2)}\boldsymbol{h}_j^{(1)} = \begin{bmatrix} \hat{\boldsymbol{x}}_j^{(1)} \\ y_j t_j \\ R'(1-t_j) \\ \vdots \\ \mathbf{0} \end{bmatrix}, \boldsymbol{V}_{2p+1}\boldsymbol{h}_j^{(1)} = -(n+1)\alpha \sum_{l=1}^{p} \hat{x}_{jl}^{(1)} \boldsymbol{e}_{D_0+qp+l}, \tag{39}$$

$$\boldsymbol{Q}_{2p+2}^{(2)}\boldsymbol{h}_i^{(1)} = \begin{bmatrix} -\boldsymbol{\beta}^{(0)} \\ 1 \\ -1 \\ \vdots \\ \mathbf{0} \end{bmatrix}, \boldsymbol{K}_{2p+2}^{(2)}\boldsymbol{h}_j^{(1)} = \begin{bmatrix} \hat{\boldsymbol{x}}_j^{(1)} \\ y_j t_j \\ R'(1-t_j) \\ \vdots \\ \mathbf{0} \end{bmatrix}, \boldsymbol{V}_{2p+2}\boldsymbol{h}_j^{(1)} = (n+1)\alpha \sum_{l=1}^{p} \hat{x}_{jl}^{(1)} \boldsymbol{e}_{D_0+qp+l}, \tag{40}$$

where $R' = \max_{\substack{i=1,\ldots,n+1 \\ t=0,1,\ldots}} \{|\boldsymbol{\beta}^{(t)\top}\boldsymbol{x}_i|\}$. Then

$$\begin{aligned} \sigma\left(\langle \boldsymbol{Q}_{2p+1}^{(2)}\boldsymbol{h}_i^{(1)}, \boldsymbol{K}_{2p+1}^{(2)}\boldsymbol{h}_j^{(1)}\rangle\right) &= \sigma\left(\boldsymbol{\beta}^{(0)\top}\hat{\boldsymbol{x}}_j^{(1)} - y_j t_j - R'(1-t_j)\right) \\ &= \sigma\left(\boldsymbol{\beta}^{(0)\top}\hat{\boldsymbol{x}}_j^{(1)} - y_j\right)\mathbb{1}\{t_j = 1\} \\ &= \sigma\left(\boldsymbol{\beta}^{(0)\top}\hat{\boldsymbol{x}}_j^{(1)} - y_j\right) t_j, \end{aligned}$$

and

$$\begin{aligned} \sigma\left(\langle \boldsymbol{Q}_{2p+2}^{(2)}\boldsymbol{h}_i^{(1)}, \boldsymbol{K}_{2p+2}^{(2)}\boldsymbol{h}_j^{(1)}\rangle\right) &= \sigma\left(-\boldsymbol{\beta}^{(0)\top}\hat{\boldsymbol{x}}_j^{(1)} + y_j t_j - R'(1-t_j)\right) \\ &= \sigma\left(-\boldsymbol{\beta}^{(0)\top}\hat{\boldsymbol{x}}_j^{(1)} + y_j\right)\mathbb{1}\{t_j = 1\} \\ &= \sigma\left(-\boldsymbol{\beta}^{(0)\top}\hat{\boldsymbol{x}}_j^{(1)} + y_j\right) t_j. \end{aligned}$$

So that

$$\begin{aligned} &\sum_{m=2p+1}^{2p+2} \sigma\left(\langle \boldsymbol{Q}_m^{(2)}\boldsymbol{h}_i^{(1)}, \boldsymbol{K}_m^{(2)}\boldsymbol{h}_j^{(1)}\rangle\right) \boldsymbol{V}_m^{(2)}\boldsymbol{h}_j^{(1)} \\ &= -(n+1)t_j\alpha\left[\sigma\left(\boldsymbol{\beta}^{(0)\top}\hat{\boldsymbol{x}}_j^{(1)} - y_j\right) - \sigma\left(-\boldsymbol{\beta}^{(0)\top}\hat{\boldsymbol{x}}_j^{(1)} + y_j\right)\right] \cdot \sum_{l=1}^{p} \hat{x}_{jl}^{(1)} \boldsymbol{e}_{D_0+qp+l} \\ &= -(n+1)t_j\alpha \sum_{l=1}^{p} \hat{x}_{jl}^{(1)} \left(\boldsymbol{\beta}^{(0)\top}\hat{\boldsymbol{x}}_j^{(1)} - y_j\right) \boldsymbol{e}_{D_0+qp+l}. \end{aligned}$$

Thus the final output, for $i = 1, \ldots, n+1$:

$$
\begin{aligned}
\boldsymbol{h}_i^{(2)} &= \boldsymbol{h}_i^{(1)} + \sum_{m=1}^{2p+2} \frac{1}{n+1} \sum_{j=1}^{n} \sigma\left(\langle \boldsymbol{Q}_m^{(2)} \boldsymbol{h}_i^{(1)}, \boldsymbol{K}_m^{(2)} \boldsymbol{h}_j^{(1)} \rangle\right) \boldsymbol{V}_m^{(2)} \boldsymbol{h}_j^{(1)} \\
&= \begin{bmatrix} \boldsymbol{z}_i \\ \boldsymbol{x}_i \\ y_i t_i \\ \boldsymbol{\Theta}_{:,1}^{(0)} - \eta \sum_{j=1}^{n} \boldsymbol{z}_j \left(\boldsymbol{z}_j^\top \boldsymbol{\Theta}_{:,1}^{(0)} - x_{j1}\right) \\ \vdots \\ \boldsymbol{\Theta}_{:,p}^{(0)} - \eta \sum_{j=1}^{n} \boldsymbol{z}_j \left(\boldsymbol{z}_j^\top \boldsymbol{\Theta}_{:,p}^{(0)} - x_{jp}\right) \\ \boldsymbol{\beta}^{(0)} - \alpha \sum_{j=1}^{n} \hat{\boldsymbol{x}}_j^{(1)} \left(\hat{\boldsymbol{x}}_j^{(1)\top} \boldsymbol{\beta}^{(0)} - y_j\right) \\ \hat{\boldsymbol{x}}_i^{(1)} \\ 1 \\ t_i \end{bmatrix} \\
&= \begin{bmatrix} \boldsymbol{z}_i \\ \boldsymbol{x}_i \\ y_i t_i \\ \boldsymbol{\Theta}_{:,1}^{(0)} - \eta \boldsymbol{Z}^\top \left[\boldsymbol{Z}\boldsymbol{\Theta}^{(0)} - \boldsymbol{X}\right]_{:,1} \\ \vdots \\ \boldsymbol{\Theta}_{:,p}^{(0)} - \eta \boldsymbol{Z}^\top \left[\boldsymbol{Z}\boldsymbol{\Theta}^{(0)} - \boldsymbol{X}\right]_{:,p} \\ \boldsymbol{\beta}^{(0)} - \alpha \boldsymbol{\Theta}^{(0)\top} \boldsymbol{Z}^\top \left(\boldsymbol{Z}\boldsymbol{\Theta}^{(0)} \boldsymbol{\beta}^{(0)} - \boldsymbol{y}\right) \\ \hat{\boldsymbol{x}}_i^{(1)} \\ 1 \\ t_i \end{bmatrix}.
\end{aligned}
$$

This corresponds to a one-step 2SLS GD update of $\boldsymbol{\Theta}, \boldsymbol{\beta}$, according to (8). Therefore, the final output matrix is:

$$
\boldsymbol{H}^{(2)} = \begin{bmatrix}
\boldsymbol{z}_1 & \cdots & \boldsymbol{z}_n & \boldsymbol{z}_{n+1} \\
\boldsymbol{x}_1 & \cdots & \boldsymbol{x}_n & \boldsymbol{x}_{n+1} \\
y_1 & \cdots & y_n & 0 \\
\boldsymbol{\Theta}_{:,1}^{(1)} & \cdots & \boldsymbol{\Theta}_{:,1}^{(1)} & \boldsymbol{\Theta}_{:,1}^{(1)} \\
\vdots & \vdots & \vdots & \vdots \\
\boldsymbol{\Theta}_{:,p}^{(1)} & \cdots & \boldsymbol{\Theta}_{:,p}^{(1)} & \boldsymbol{\Theta}_{:,p}^{(1)} \\
\boldsymbol{\beta}^{(1)} & \cdots & \boldsymbol{\beta}^{(1)} & \boldsymbol{\beta}^{(1)} \\
\hat{\boldsymbol{x}}_1^{(1)} & \cdots & \hat{\boldsymbol{x}}_n^{(1)} & \hat{\boldsymbol{x}}_{n+1}^{(1)} \\
1 & \cdots & 1 & 1 \\
1 & \cdots & 1 & 0
\end{bmatrix}.
$$

Thus the proof is complete. We further note that in construction steps (35)(36)(37)(38)(39)(40), regardless of the initial values of $\boldsymbol{\Theta}^{(0)}, \boldsymbol{\beta}^{(0)}$, and $\hat{\boldsymbol{x}}^{(0)}$, the matrices $\boldsymbol{Q}_m^{(1:2)}, \boldsymbol{K}_m^{(1:2)}, \boldsymbol{V}_m^{(1:2)}$ do the same linear transformations on the input vectors. Therefore they are identical across different layers. □

B.3 Proof of Theorem 3.3

Lemma B.2 (Generalization of pretraining, from Theorem 20 in Bai et al. (2024))**.** Given optimization problm (13), with probability at least $1 - \zeta$, the solution $\hat{\boldsymbol{\theta}}$ satisfies:

$$
L_{\mathsf{ICL}}(\hat{\boldsymbol{\theta}}) \leq \inf_{\boldsymbol{\theta} \in \boldsymbol{\vartheta}} L_{\mathsf{ICL}}(\boldsymbol{\theta}) + \mathcal{O}\left(B_y^2 \sqrt{\frac{L^2(MD^2 + DD')\log(2 + \max\{B_\theta, R, B_y\}) + \log(1/\zeta)}{N}}\right).
$$

Proof of Theorem 3.3. We begin by showing the (clipped) 2SLS predictor achieves small excess loss under in-context distribution $\boldsymbol{\mathcal{P}}$:

$$
\begin{aligned}
&\mathbb{E}_{\mathcal{P}}\left[\left(\langle \mathsf{clip}_{B_\beta}(\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}), \boldsymbol{x}_{n+1}\rangle - y_{n+1}\right)^2\right] \\
&= \mathbb{E}_{\mathcal{P}}\left[\left(\langle \mathsf{clip}_{B_\beta}(\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}) - \boldsymbol{\beta}, \boldsymbol{x}_{n+1}\rangle + \langle \boldsymbol{\beta}, \boldsymbol{x}_{n+1}\rangle - y_{n+1}\right)^2\right] \\
&= \mathbb{E}_{\mathcal{P}}\left[\langle \mathsf{clip}_{B_\beta}(\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}) - \boldsymbol{\beta}, \boldsymbol{x}_{n+1}\rangle^2\right] + 2E_{\mathcal{P}}\left[\langle \mathsf{clip}_{B_\beta}(\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}) - \boldsymbol{\beta}, \boldsymbol{x}_{n+1}\rangle\big(\langle \boldsymbol{\beta}, \boldsymbol{x}_{n+1}\rangle - y_{n+1}\big)\right] \\
&\qquad + \mathbb{E}_{\mathcal{P}}\left[\big(\langle \boldsymbol{\beta}, \boldsymbol{x}_{n+1}\rangle - y_{n+1}\big)^2\right] \\
&= \underbrace{\mathbb{E}_{\mathcal{P}}\left[\langle \mathsf{clip}_{B_\beta}(\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}) - \boldsymbol{\beta}, \boldsymbol{x}_{n+1}\rangle^2\right]}_{\text{Excess Loss}} + \mathbb{E}_{\mathcal{P}}\left[\big(\langle \boldsymbol{\beta}, \boldsymbol{x}_{n+1}\rangle - y_{n+1}\big)^2\right],
\end{aligned}
$$

where $\mathbb{E}_{\mathcal{P}}[\langle \mathsf{clip}_{B_\beta}(\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}) - \boldsymbol{\beta}, \boldsymbol{x}_{n+1}\rangle\big(\langle \boldsymbol{\beta}, \boldsymbol{x}_{n+1}\rangle - y_{n+1}\big)] = 0$ follows from the independence between $\langle \mathsf{clip}_{B_\beta}(\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}) - \boldsymbol{\beta}, \boldsymbol{x}_{n+1}\rangle$ and $\big(\langle \boldsymbol{\beta}, \boldsymbol{x}_{n+1}\rangle - y_{n+1}\big)$ with $\mathbb{E}_{\mathcal{P}}\left[\langle \boldsymbol{\beta}, \boldsymbol{x}_{n+1}\rangle - y_{n+1}\right] = \mathbb{E}_{\mathcal{P}}[\epsilon_{n+1}] = 0$.

To bound the excess loss, we have

$$
\begin{aligned}
\mathbb{E}_{\mathcal{P}}\left[\langle \mathsf{clip}_{B_\beta}(\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}) - \boldsymbol{\beta}, \boldsymbol{x}_{n+1}\rangle^2\right] &= \mathbb{E}_{\mathcal{P}}\left[\left\|\boldsymbol{x}_{n+1}^\top\left(\mathsf{clip}_{B_\beta}(\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}) - \boldsymbol{\beta}\right)\right\|^2\right] \\
&\le \mathbb{E}_{\mathcal{P}}\left[\|\boldsymbol{x}_{n+1}\|^2\left\|\mathsf{clip}_{B_\beta}(\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}) - \boldsymbol{\beta}\right\|^2\right] \\
&= \mathbb{E}_{\mathcal{P}}\left[\|\boldsymbol{x}_{n+1}\|^2\right]\mathbb{E}_{\mathcal{P}}\left[\left\|\mathsf{clip}_{B_\beta}(\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}) - \boldsymbol{\beta}\right\|^2\right] \\
&\le \mathcal{O}\left(B_x^2\Big(\frac{C_{z,x}^2 q(\boldsymbol{\phi}^\top\boldsymbol{\Sigma}_u\boldsymbol{\phi} + \sigma_\epsilon^2)}{n} + \xi B_\beta^2\Big)\right),
\end{aligned}
\tag{41}
$$

where the last inequality follows from (4).

Next, for the ICL loss, we have

$$
\begin{aligned}
&L_{\mathsf{ICL}}(\boldsymbol{\theta}) \\
&= \mathbb{E}_\pi\mathbb{E}_{\mathcal{P}}\left[\left(\widetilde{\mathsf{TF}}_{\boldsymbol{\theta}}(\boldsymbol{H}) - y_{n+1}\right)^2\right] \\
&= \mathbb{E}_\pi\mathbb{E}_{\mathcal{P}}\left[\left(\widetilde{\mathsf{TF}}_{\boldsymbol{\theta}}(\boldsymbol{H}) - \langle \mathsf{clip}_{B_\beta}(\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}), \boldsymbol{x}_{n+1}\rangle + \langle \mathsf{clip}_{B_\beta}(\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}), \boldsymbol{x}_{n+1}\rangle - y_{n+1}\right)^2\right] \\
&= \mathbb{E}_\pi\Bigg\{\mathbb{E}_{\mathcal{P}}\Big[\left(\widetilde{\mathsf{TF}}_{\boldsymbol{\theta}}(\boldsymbol{H}) - \langle \mathsf{clip}_{B_\beta}(\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}), \boldsymbol{x}_{n+1}\rangle\right)^2\Big] + \mathbb{E}_{\mathcal{P}}\Big[\left(\langle \mathsf{clip}_{B_\beta}(\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}), \boldsymbol{x}_{n+1}\rangle - y_{n+1}\right)^2\Big] \\
&\qquad + 2\mathbb{E}_{\mathcal{P}}\Big[\left(\widetilde{\mathsf{TF}}_{\boldsymbol{\theta}}(\boldsymbol{H}) - \langle \mathsf{clip}_{B_\beta}(\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}), \boldsymbol{x}_{n+1}\rangle\right)\left(\langle \mathsf{clip}_{B_\beta}(\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}), \boldsymbol{x}_{n+1}\rangle - y_{n+1}\right)\Big]\Bigg\} \\
&\le \mathbb{E}_\pi\Bigg\{\mathbb{E}_{\mathcal{P}}\Big[\left(\widetilde{\mathsf{TF}}_{\boldsymbol{\theta}}(\boldsymbol{H}) - \langle \mathsf{clip}_{B_\beta}(\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}), \boldsymbol{x}_{n+1}\rangle\right)^2\Big] + \mathbb{E}_{\mathcal{P}}\Big[\langle \mathsf{clip}_{B_\beta}(\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}) - \boldsymbol{\beta}, \boldsymbol{x}_{n+1}\rangle^2\Big] + \mathbb{E}_{\mathcal{P}}\Big[\big(\langle \boldsymbol{\beta}, \boldsymbol{x}_{n+1}\rangle - y_{n+1}\big)^2\Big] \\
&\qquad + 2\mathbb{E}_{\mathcal{P}}\Big[\left|\widetilde{\mathsf{TF}}_{\boldsymbol{\theta}}(\boldsymbol{H}) - \langle \mathsf{clip}_{B_\beta}(\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}), \boldsymbol{x}_{n+1}\rangle\right|\Big]\mathbb{E}_{\mathcal{P}}\Big[\left|\langle \mathsf{clip}_{B_\beta}(\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}), \boldsymbol{x}_{n+1}\rangle - y_{n+1}\right|\Big]\Bigg\} \\
&\le \mathbb{E}_\pi\Bigg\{\mathbb{E}_{\mathcal{P}}\Big[\left(\widetilde{\mathsf{TF}}_{\boldsymbol{\theta}}(\boldsymbol{H}) - \langle \mathsf{clip}_{B_\beta}(\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}), \boldsymbol{x}_{n+1}\rangle\right)^2\Big] + \mathbb{E}_{\mathcal{P}}\Big[\langle \mathsf{clip}_{B_\beta}(\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}) - \boldsymbol{\beta}, \boldsymbol{x}_{n+1}\rangle^2\Big] + \mathbb{E}_{\mathcal{P}}\Big[\big(\langle \boldsymbol{\beta}, \boldsymbol{x}_{n+1}\rangle - y_{n+1}\big)^2\Big] \\
&\qquad + 2\mathbb{E}_{\mathcal{P}}\Big[\left|\widetilde{\mathsf{TF}}_{\boldsymbol{\theta}}(\boldsymbol{H}) - \langle \mathsf{clip}_{B_\beta}(\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}), \boldsymbol{x}_{n+1}\rangle\right|\Big]\left(\mathbb{E}_{\mathcal{P}}\Big[\left|\langle \mathsf{clip}_{B_\beta}(\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}) - \boldsymbol{\beta}, \boldsymbol{x}_{n+1}\rangle\right|\Big] + \mathbb{E}_{\mathcal{P}}\Big[\left|\langle \boldsymbol{\beta}, \boldsymbol{x}_{n+1}\rangle - y_{n+1}\right|\Big]\right)\Bigg\}.
\end{aligned}
\tag{42}
$$

From Corollary 3.1, we know that there exists a $L = 2\bar{L} + 1$-layer attention-only transformer model $\boldsymbol{\theta}$, with $M = 2(p+1)$ heads, and embedding dimension $D = qp + 3p + q + 3$, such that for any $\boldsymbol{H}$, given any learning rates α, η and Λ as defined in Equation (9), the following holds[4]:

$$\left|\widetilde{\mathsf{TF}}_{\boldsymbol{\theta}}(\boldsymbol{H}) - \langle \mathsf{clip}_{B_\beta}(\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}), \boldsymbol{x}_{n+1}\rangle\right| \leq \mathcal{O}\left(B_x \Lambda^{\bar{L}}\right).$$

Denote $\Lambda^\star := \min_{\alpha,\eta} \mathbb{E}_\pi \mathbb{E}_{\mathcal{P}}\left[\Lambda | \boldsymbol{H}, \alpha, \eta\right]$, then under $\alpha^\star, \eta^\star$, we have:

$$\mathbb{E}_\pi \mathbb{E}_{\mathcal{P}}\left[\left|\widetilde{\mathsf{TF}}_{\boldsymbol{\theta}}(\boldsymbol{H}) - \langle \mathsf{clip}_{B_\beta}(\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}), \boldsymbol{x}_{n+1}\rangle\right|\right] \leq \mathcal{O}\left(B_x (\Lambda^\star)^{\bar{L}}\right), \tag{43}$$

and

$$\mathbb{E}_\pi \mathbb{E}_{\mathcal{P}}\left[\left(\widetilde{\mathsf{TF}}_{\boldsymbol{\theta}}(\boldsymbol{H}) - \langle \mathsf{clip}_{B_\beta}(\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}), \boldsymbol{x}_{n+1}\rangle\right)^2\right] \leq \mathcal{O}\left(B_x^2 \mu^\star_{\Lambda,2}\right), \tag{44}$$

where $\mu^\star_{\Lambda,2} := \mathbb{E}_\pi \mathbb{E}_{\mathcal{P}}\left[\Lambda^{2\bar{L}} | \boldsymbol{H}, \alpha^\star, \eta^\star\right]$ is close to 0.

With conidition (14), from Cauchy-Schwarz inequality, we have:

$$\mathbb{E}_\pi \mathbb{E}_{\mathcal{P}}\left[\left|\langle \boldsymbol{\beta}, \boldsymbol{x}_{n+1}\rangle - y_{n+1}\right|\right] \leq \mathbb{E}_\pi\left[\sqrt{\mathbb{E}_{\mathcal{P}}\left(\epsilon_{n+1}^2\right)}\right] = \mathbb{E}_\pi[\sigma_\epsilon] \leq \tilde{\sigma}_\epsilon. \tag{45}$$

Also, from (41), we have:

$$\mathbb{E}_\pi \mathbb{E}_{\mathcal{P}}\left[\langle \mathsf{clip}_{B_\beta}(\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}) - \boldsymbol{\beta}, \boldsymbol{x}_{n+1}\rangle^2\right] \leq \mathcal{O}\left(B_x^2\Big(\frac{C_{z,x}^2 q\tilde{\sigma}^2}{n} + \xi B_\beta^2\Big)\right). \tag{46}$$

Further,

$$\begin{aligned}\mathbb{E}_\pi \mathbb{E}_{\mathcal{P}}\left[\left|\langle \mathsf{clip}_{B_\beta}(\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}) - \boldsymbol{\beta}, \boldsymbol{x}_{n+1}\rangle\right|\right] &\leq \sqrt{\mathbb{E}_\pi \mathbb{E}_{\mathcal{P}}\left[\langle \mathsf{clip}_{B_\beta}(\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}) - \boldsymbol{\beta}, \boldsymbol{x}_{n+1}\rangle^2\right]} \\ &\leq \mathcal{O}\left(B_x\sqrt{\frac{C_{z,x}^2 q\tilde{\sigma}^2}{n} + \xi B_\beta^2}\right).\end{aligned} \tag{47}$$

Finally, with (43)(44)(45)(46)(47), rearranging the terms in (42), we have:

$$\begin{aligned}&L_{\mathsf{ICL}}(\boldsymbol{\theta}) - \mathbb{E}_\pi \mathbb{E}_{\mathcal{P}}\left[(y_{n+1} - \langle \boldsymbol{\beta}, \boldsymbol{x}_{n+1}\rangle)^2\right] \\ &\leq \mathcal{O}\left(B_x^2\Big(\mu^\star_{\Lambda,2} + (\Lambda^\star)^{\bar{L}}\sqrt{\frac{C_{z,x}^2 q\tilde{\sigma}^2}{n} + \xi B_\beta^2} + \frac{C_{z,x}^2 q\tilde{\sigma}^2}{n} + \xi B_\beta^2\Big) + B_x (\Lambda^\star)^{\bar{L}}\tilde{\sigma}_\epsilon\right). \\ &\leq \mathcal{O}\left((\Lambda^\star)^{\bar{L}}\Big(B_x^2\sqrt{\frac{C_{z,x}^2 q\tilde{\sigma}^2}{n} + \xi B_\beta^2} + B_x\tilde{\sigma}_\epsilon\Big) + B_x^2\Big(\frac{C_{z,x}^2 q\tilde{\sigma}^2}{n} + \xi B_\beta^2 + \mu^\star_{\Lambda,2}\Big)\right).\end{aligned} \tag{48}$$

Thus combining Lemma B.2 with (48) completes the proof.

□

C Additional Experiments and Discussions

For all experiments in this section, to be consistent with our main experiment in Section 4, we generate $n = 50$ training samples with $p = 5, q = 10$, following the data generating process described in Algorithm 1. The task parameters $\boldsymbol{\Theta}, \boldsymbol{\beta}, \boldsymbol{\Phi}, \boldsymbol{\phi}$ are sampled from standard Gaussian distribution, the covariance matrices $\boldsymbol{\Sigma}_z, \boldsymbol{\Sigma}_u, \boldsymbol{\Sigma}_\omega$ are set to be identity matrices, and the noise level σ_ϵ is set to 1.

[4]The clipping bound on $\hat{\boldsymbol{\beta}}_{\mathsf{2SLS}}$ can be matched by adjusting the clipping threshold on the last layer of $\widetilde{\mathsf{TF}}_{\boldsymbol{\theta}}$.

C.1 Empirical Justification for Assumption 2

In this section, we aim to provide a better understanding of the values of λ_z, $\lambda_{\tilde{x}}$ and $\delta_{z\epsilon_2}$ required to satisfy condition (3).

We first investigate λ_z. According to Rudelson & Vershynin (2010), given that random matrix $\boldsymbol{Z}$ has independent sub-Gaussian entries, one can show that

$$\mathbb{P}\left(\lambda_{\min}\left(\frac{\boldsymbol{Z}^\top \boldsymbol{Z}}{n}\right) \leq \left(1-\sqrt{\frac{q}{n}}-\frac{n}{n-q}\varepsilon\right)^2\right) \leq \frac{C}{1-\sqrt{q/n}} \exp\left(-c' q \varepsilon^{3/2}\right).$$

This inequality implies that setting λ_z slightly lower than $(1-\sqrt{\frac{q}{n}})^2$ is sufficient to ensure a small tail probability.

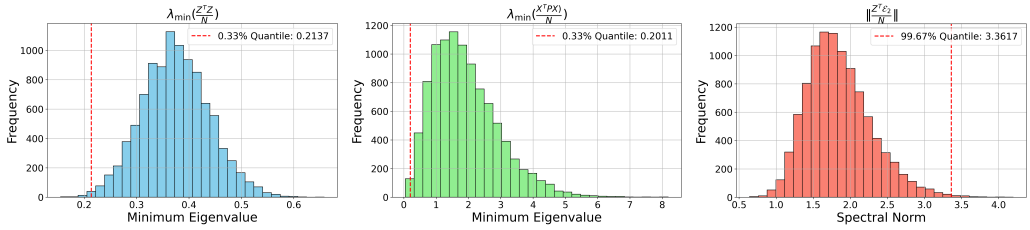
For $\delta_{z\epsilon_2}$, consider the case when $\boldsymbol{z}$ and $\boldsymbol{\epsilon}_2$ are uncorrelated (i.e, $\mathbb{E}[\boldsymbol{z}\boldsymbol{\epsilon}_2^\top] = \mathbf{0}$), and both $\boldsymbol{z}$ and $\boldsymbol{\epsilon}_2$ are almost surely bounded by B_z and B_{ϵ_2}, respectively. Applying the matrix Bernstein inequality (see Lemma A.1) directly yields:

$$\mathbb{P}\left(\left\|\frac{\boldsymbol{Z}^\top \boldsymbol{\mathcal{E}}_2}{n}\right\| \geq \varepsilon\right) \leq (q+p)\exp\left(-\frac{n^2\varepsilon^2/2}{\nu(\boldsymbol{\Omega}_{z\epsilon_2}) + B_z B_{\epsilon_2} n\varepsilon/3}\right)$$

where $\nu(\boldsymbol{\Omega}_{z\epsilon_2})$ increases on the order of $\mathcal{O}(n)$. Thus, the tail probability decreases sharply at a rate of $\mathcal{O}\left(\exp(-n)\right)$. For the case where $\mathbb{E}[\boldsymbol{z}\boldsymbol{\epsilon}_2^\top] \neq \mathbf{0}$, a similar conclusion can be reached by applying the Bernstein inequality on the cetered random matrix $\frac{\boldsymbol{Z}^\top \boldsymbol{\mathcal{E}}_2}{n} - \mathbb{E}[\boldsymbol{z}\boldsymbol{\epsilon}_2^\top]$.

To validate $\lambda_{\tilde{x}}$, we note that the entries of $\frac{\boldsymbol{X}^\top \boldsymbol{P}_Z \boldsymbol{X}}{n}$ are not independent, making it challenging to derive a non-asymptotic bound. However, it is known that $\lambda_{\min}\left(\frac{\boldsymbol{X}^\top \boldsymbol{X}}{n}\right)$ can be lower bounded with high probability. Since $\boldsymbol{z}$ and $\boldsymbol{x}$ are correlated and $\mathsf{rank}(\boldsymbol{Z}) > \mathsf{rank}(\boldsymbol{X})$, projecting $\boldsymbol{X}$ onto the column space of $\boldsymbol{Z}$ should preserve the eigenproperties of the sample covariance matrix (i.e. the minimal spread along any dimension should not decrease significantly towards 0).

To further verify our statements, we conduct 10000 experiments and record the values of $\lambda_{\min}(\frac{\boldsymbol{Z}^\top \boldsymbol{Z}}{n})$, $\lambda_{\min}(\frac{\boldsymbol{X}^\top \boldsymbol{P}_Z \boldsymbol{X}}{n})$, and $\|\frac{\boldsymbol{Z}^\top \boldsymbol{\mathcal{E}}_2}{n}\|$. The histograms shown in Figure 3 indicate that mild choices on $\lambda_z, \lambda_{\tilde{x}}$ and $\delta_{z\epsilon_2}$ are sufficient to satisfy the sample condition (3). Specifically, we have $\xi = 0.01, \lambda_z = 0.2137, \lambda_{\tilde{x}} = 0.2011$, and $\delta_{z\epsilon_2} = 3.3617$ as empirical guidelines.



(a) (b) (c)

Figure 3: Histograms of (a) $\lambda_{\min}(\frac{\boldsymbol{Z}^\top \boldsymbol{Z}}{n})$, (b) $\lambda_{\min}(\frac{\boldsymbol{X}^\top \boldsymbol{P}_Z \boldsymbol{X}}{n})$ and (c) $\|\frac{\boldsymbol{Z}^\top \boldsymbol{\mathcal{E}}_2}{n}\|$ over 10000 experiments.

C.2 Simulations Verifying Theorem 3.1

We use the GD-based 2SLS method (8) to estimate the causal effect $\boldsymbol{\beta}$. For the simulated data, we calculate the following metrics:

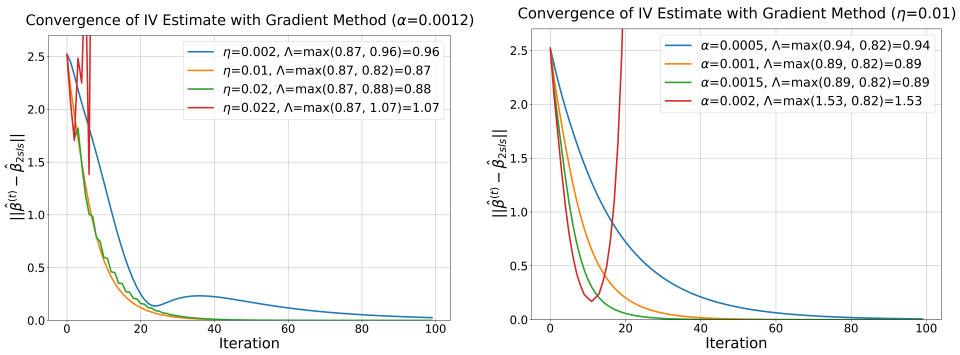
$$\frac{2}{\sigma_{\max}^2(\boldsymbol{Z}\hat{\boldsymbol{\Theta}})} = 0.0016, \frac{2}{\sigma_{\max}^2(\boldsymbol{Z})} = 0.0212.$$

By Theorem 3.1, the gradient descent converges when $\alpha \in (0, 0.0016)$ and $\eta \in (0, 0.0212)$. The overall convergence rate is determined by $\Lambda := \max\{\gamma(\alpha), \kappa(\eta)\}$, where

$$\gamma(\alpha) := \rho\left(\boldsymbol{I} - \alpha \hat{\boldsymbol{\Theta}}^\top \boldsymbol{Z}^\top \boldsymbol{Z} \hat{\boldsymbol{\Theta}}\right),$$

$$\kappa(\eta) := \rho\left(\boldsymbol{I} - \eta \boldsymbol{Z}^\top \boldsymbol{Z}\right).$$

We first set $\alpha = 0.0012$ and vary η. The corresponding convergence rates are determined by $\Lambda = \max(0.87, \kappa(\eta))$. Next, we set $\eta = 0.01$ and vary α. The corresponding convergence curves are determined by $\Lambda = \max(\gamma(\alpha), 0.82)$. We compare the estimates $\hat{\boldsymbol{\beta}}^{(t)}$ with the 2SLS estimate $\hat{\boldsymbol{\beta}}_{\text{2SLS}}$ as the iteration proceeds. The convergence results are shown in Figure 4.



(a) (b)

Figure 4: The convergence of the GD-based 2SLS method with (a) fixed $\alpha = 0.0012$ and varying η and (b) fixed $\eta = 0.01$ and varying α.

The results in Figure 4 are consistent with our theoretical analysis in Theorem 3.1. It is worth noting that in Figure 4a, when η is relatively large (or small), the convergence curves exhibit some suiggly patterns. This is due to the innerloop updates (8a) are converging faster (or slower) than the outer loop updates (8b). However, the overall convergence rate is still determined by Λ. This pattern doesn't appear in Figure 4b as we set η to be a moderate value, which ensures that the inner loop and outer loop converge synchronously.

Next, we show the bias of the GD estimator. For better convergence, we set $\alpha^\star = \frac{1}{\sigma^2_{\max}(\boldsymbol{Z}\hat{\boldsymbol{\Theta}})}$ and $\eta^\star = \frac{1}{\sigma^2_{\max}(\boldsymbol{Z})}$. We compare the biases of the GD estimator with $n = 50, 100, 150$ in-context samples. The results are shown in Figure 5.

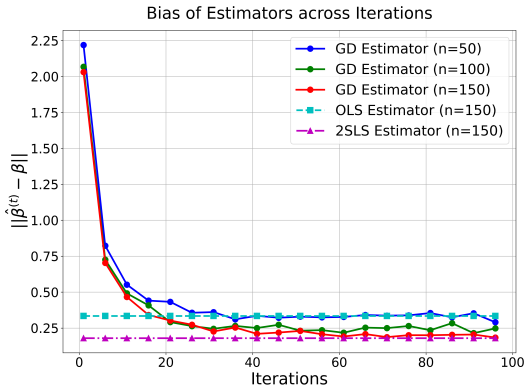


Figure 5: The convergence of the GD-based 2SLS method with $\alpha^\star = \frac{1}{\sigma^2_{\max}(\boldsymbol{Z}\hat{\boldsymbol{\Theta}})}$ and $\eta^\star = \frac{1}{\sigma^2_{\max}(\boldsymbol{Z})}$. The biases of 2SLS estimator and OLS estimator at $n = 150$ are plotted for comparison.

C.3 Discussions on 2SLS with ℓ_2-Regularization

In this section, we briefly discuss a generalization of our analysis to the case where the 2SLS estimator is regularized by the ℓ_2 penalty (Ridge 2SLS). For this case, the bi-level optimization problem Equation (7) is modified as follows:

$$\min_{\boldsymbol{\beta}} \quad \mathcal{L}(\boldsymbol{\beta}) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \boldsymbol{z}_i^\top \hat{\boldsymbol{\Theta}}\boldsymbol{\beta})^2 + \frac{1}{2}\lambda\|\boldsymbol{\beta}\|^2,$$

$$\text{where} \quad \hat{\boldsymbol{\Theta}} := \arg\min_{\boldsymbol{\Theta}} \quad \frac{1}{n}\sum_{j=1}^{n}(\boldsymbol{x}_j - \boldsymbol{z}_j^\top \boldsymbol{\Theta})^2 + \frac{1}{2}\tau\|\boldsymbol{\Theta}\|_F^2,$$

where $\lambda, \tau \geq 0$ are regularization parameters. To solve this problem, the GD updates in Equation (8) is modified as follows:

$$\boldsymbol{\Theta}^{(t+1)} = \boldsymbol{\Theta}^{(t)} - \eta\left[\boldsymbol{Z}^\top(\boldsymbol{Z}\boldsymbol{\Theta}^{(t)} - \boldsymbol{X}) + \tau\boldsymbol{\Theta}^{(t)}\right], \tag{49a}$$

$$\boldsymbol{\beta}^{(t+1)} = \boldsymbol{\beta}^{(t)} - \alpha\left[\boldsymbol{\Theta}^{(t)\top}\boldsymbol{Z}^\top(\boldsymbol{Z}\boldsymbol{\Theta}^{(t)}\boldsymbol{\beta}^{(t)} - \boldsymbol{Y}) + \lambda\boldsymbol{\beta}^{(t)}\right]. \tag{49b}$$

The only difference between Equation (8) and Equation (49) is the additional terms $\tau\boldsymbol{\Theta}^{(t)}$ in Equation (49a), and $\lambda\boldsymbol{\beta}^{(t)}$ in Equation (49b). The convergence analysis of the ℓ_2-regularized GD updates in Equation (49) can be conducted in a similar manner as in Theorem 3.1. The only difference is that the convergence rate Λ is now determined by the spectral radiuses of $\boldsymbol{I} - \eta(\boldsymbol{Z}^\top\boldsymbol{Z} + \tau\boldsymbol{I})$ and $\boldsymbol{I} - \alpha(\hat{\boldsymbol{\Theta}}^\top\boldsymbol{Z}^\top\boldsymbol{Z}\hat{\boldsymbol{\Theta}} + \lambda\boldsymbol{I})$, respectively.

With the same configuration as in Theorem 3.2 but adding $p+1$ attention heads in the second layer (i.e. the second layer needs $3p+3$ heads), we can show that transformers are able to implement the ℓ_2-regularized GD updates in Equation (49). The proof follows directly from Appendix B.2, with the construction of the new attention heads in the second layer as follows.

For $m = 2p+2+k, k = 1, \ldots, p$, define $\boldsymbol{Q}_m^{(2)}, \boldsymbol{K}_m^{(2)}, \boldsymbol{V}_m^{(2)}$ such that:

$$\boldsymbol{Q}_m^{(2)}\boldsymbol{h}_i^{(1)} = \begin{bmatrix}1\\0\end{bmatrix}, \boldsymbol{K}_m^{(2)}\boldsymbol{h}_i^{(1)} = \begin{bmatrix}1\\0\end{bmatrix}, \boldsymbol{V}_m^{(2)}\boldsymbol{h}_i^{(1)} = -\eta\tau\sum_{l=1}^{q}\Theta_{lk}^{(0)}\boldsymbol{e}_{D_0+(k-1)q+l}.$$

For $m = 3p+3$, define $\boldsymbol{Q}_{3p+3}^{(2)}, \boldsymbol{K}_{3p+3}^{(2)}, \boldsymbol{V}_{3p+3}^{(2)}$ such that:

$$\boldsymbol{Q}_{3p+3}^{(2)}\boldsymbol{h}_i^{(1)} = \begin{bmatrix}1\\0\end{bmatrix}, \boldsymbol{K}_{3p+3}^{(2)}\boldsymbol{h}_i^{(1)} = \begin{bmatrix}1\\0\end{bmatrix}, \boldsymbol{V}_{3p+3}^{(2)}\boldsymbol{h}_i^{(1)} = -\alpha\lambda\sum_{l=1}^{q}\beta_l^{(0)}\boldsymbol{e}_{D_0+qp+l}.$$

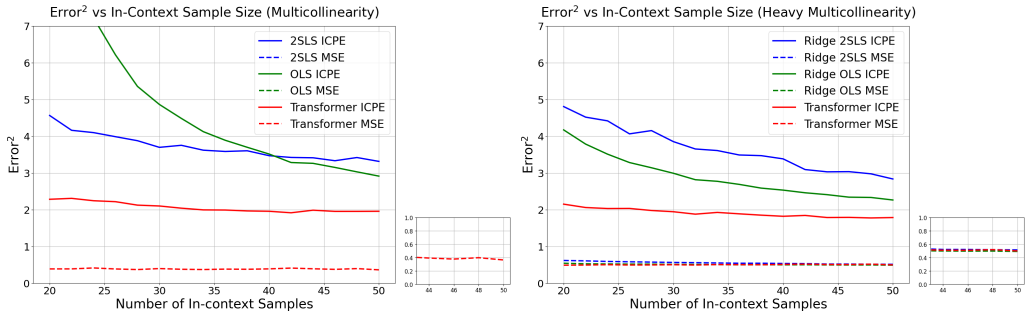
Then the remaining proof follows exactly the same as Appendix B.2. This result indicates that transformers are potentially capable of handling multicollinearity in IV regression problem. We conduct experiments to validate this and the results are shown in Appendix C.4.

C.4 Experiments on Multicollinearity IV Problem

As a supplement to Section 4.2, we examine the case where multicollinearity occurs in the IV regression problem. We generate the test prompts in the same way using Algorithm 1, but introduce multicollinearity in the endogenous variable $\boldsymbol{x}$ and instrument $\boldsymbol{z}$.

Specifically, we first generate 4 columns of $\boldsymbol{X}$, and 9 columns of $\boldsymbol{Z}$, and then set $\boldsymbol{X}_{:,5} \sim \mathcal{N}(2\boldsymbol{X}_{:,4}, 10^{-6}\boldsymbol{I}_n)$, and $\boldsymbol{Z}_{:,10} \sim \mathcal{N}(2\boldsymbol{Z}_{:,9}, 10^{-6}\boldsymbol{I}_n)$. The results are shown in Figure 6a. As shown in Figure 6a, both ordinary OLS and 2SLS estimators fail to estimate the coefficients, while the trained transformer model is still able to provide consistent predictions and coefficient estimates.

We further examine a more difficult case where heavy multicollinearity occurs. We first generate 3 column of $\boldsymbol{X}$, and 5 column of $\boldsymbol{Z}$, and then set $\boldsymbol{X}_{:,j} \sim \mathcal{N}(2\boldsymbol{X}_{:,j-2}, 10^{-6}\boldsymbol{I}_n)$ for $j = 4, 5$, and $\boldsymbol{Z}_{:,j} \sim \mathcal{N}(2\boldsymbol{Z}_{:,j-5}, 10^{-6}\boldsymbol{I}_n)$ for $j = 6, 7, 8, 9, 10$. For better comparisons, we now compare the



(a) (b)

Figure 6: The ICL performance of the trained transformer model in endogeneity tasks with multicollinearity. (a) 1 collinear column in $\boldsymbol{X}$, and 1 collinear column in $\boldsymbol{Z}$. Note that the coefficient MSEs for 2SLS and OLS are both out of range. (b) 2 collinear columns in $\boldsymbol{X}$, and 5 collinear columns in $\boldsymbol{Z}$. We compare the performance to the ℓ_2-regularized 2SLS and OLS estimators. The curves are averaged over 500 simulations.

performance of the trained transformer model to the ℓ_2-regularized 2SLS and OLS estimators (with all regularization parameters set to 1). The results are shown in Figure 6b. These results suggest that the trained transformer model is capable to handle multicollinearity in IV regression problems, even though it has not been specifically trained with multicollinearity tasks.

C.5 Experiments on Complex Non-Linear IV Problem

As a supplement to Section 4.2, we examine a more complex scenario where the instrument $\boldsymbol{z}$ has non-linear effect on the endogenous regressor $\boldsymbol{x}$. We consider the following data generating process:

$$y = \langle \boldsymbol{\beta}, \boldsymbol{x} \rangle + \epsilon_1, \quad \text{and} \quad \boldsymbol{x} = g(\boldsymbol{z}) + \boldsymbol{\epsilon_2},$$

where $g : \mathbb{R}^q \to \mathbb{R}^p$ is a two-layer fully connected neural network with ReLU activation function. Similar to Section 4, the test prompts are generated using Algorithm 1, with all task parameters and weights of neural network sampled from standard Gaussian distribution. The results are shown in Figure 7. From this figure, we can see that the trained transformer model still achieves optimal performance in this complex non-linear setting.

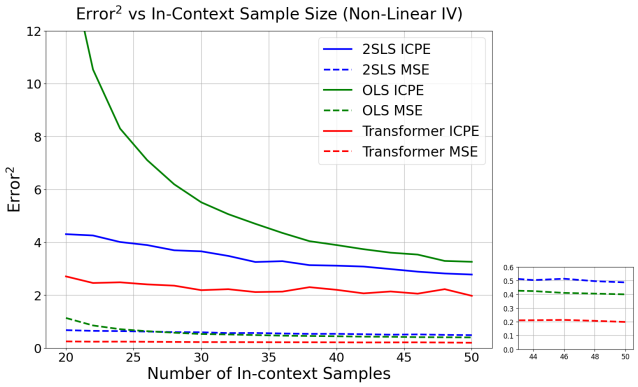


Figure 7: The ICL performance of the trained transformer model in complex non-linear endogeneity tasks where the IV has non-linear effect on the endogenous variable. The curves are averaged over 500 simulations.

C.6 Experiments on Varying Endogeneity Strength

As a supplement to Section 4.2, we examine the performance of the trained transformer model in standard IV tasks with varying endogeneity strengths. The strength of endogeneity is determined by the correlation between $\boldsymbol{x}$ and the endogenous error ϵ_1. To vary the endogeneity strength, in Algorithm 1, we multiply $\boldsymbol{u}$ by a factor $r \in (0, 2)$ when generating test prompts. The results are shown in Figure 8, which illustrates that the trained transformer model is comparable with the optimal 2SLS estimator in these standard IV tasks, regardless of the endogeneity level.

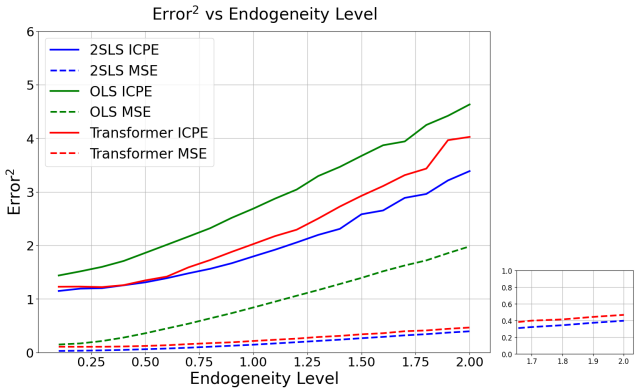


Figure 8: The ICL performance of the trained transformer model in tasks with varying endogeneity strengths. The curves are averaged over 500 simulations.

C.7 Experiments on Real-World Dataset

In this section we provide an example to illustrate how the pretrained transformer model can be applied to a real-world dataset. We use the dataset from the study of Angrist & Evans (1998). This study investigates the effect of childbearing on labor supply. For demonstration purpose, we consider a simplified setup. We focus on a subset of the dataset that contains 6421 samples from Alabama. The outcome variable y is mother's labor supply (number of working weeks in a year divided by 52), the endogenous variable x is the number of children (≥ 2), and the instrument z is an indicator variable of whether the first and second children are of the same sex[5]. For each run we randomly select 50 samples from the dataset, and make the boxplot of the estimated β over 500 runs[6]. As the ground truth effect β is unknown, we compare them to the OLS and 2SLS estimates over all samples. The results are shown in Figure 9.

The final estimate $\beta_{\text{GPT}} = -0.091$, which suggests that with each increase in the number of children, the mother's labor supply is expected to drop 9.1% (approximately 4.73 weeks per year). This result is closer to the 2SLS estimate $\beta_{\text{2SLS}} = -0.089$ than the OLS estimate $\beta_{\text{OLS}} = -0.023$. This example demonstrates the potential of the pretrained transformer model in handling real-world IV problems.

C.8 Experimental Detail

The training of the transformer in our experiment was conducted on a Windows 11 machine with the following specifications:

- GPU: NVIDIA GeForce RTX 4090
- CPU: Intel Core i9-14900KF

[5]Research found that parents of same-sex siblings are significantly more likely to go on to have an additional child (Westoff & Parke, 1972), while it is not directly correlated with mother's labor supply as mixture of sex of the first two children can be considered as randomly assigned.

[6]For large enough model that can fit in the entire dataset, this step can be ignored. As shown in the simulation study in Section 4, a single estimate is expected to perform at least as good as 2SLS estimator, given the same number of samples.

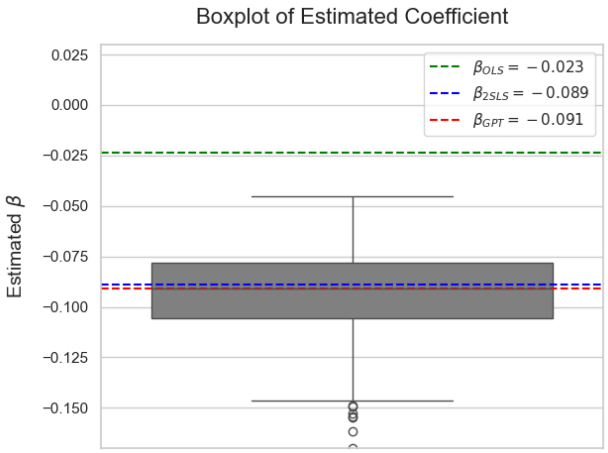


Figure 9: The boxplot of transformer's estimates over 500 runs on the labor supply dataset, comparing to the OLS and 2SLS estimates. β_{GPT} is taken to be the median of all runs. The gray box represents the interquartile range, where the middle 50% of the estimated values fall. The whiskers of the box indicate the spread of the estimates. Any points falling outside of the whisker can be considered as outliers.

- Memory: 32 GB DDR5, 5600MHz

The training process took around 10 hours.