Deep Equilibrium Models are Almost Equivalent to Not-so-deep Explicit Models for High-dimensional Gaussian Mixtures

Zenan Ling [1] **Longbo Li** [1 2] **Zhanbo Feng** [3] **Yixuan Zhang** [4] **Feng Zhou** [5] **Robert C. Qiu** [1 2] **Zhenyu Liao** [1]

Abstract

Deep equilibrium models (DEQs), as typical implicit neural networks, have demonstrated remarkable success on various tasks. There is, however, a lack of theoretical understanding of the connections and differences between implicit DEQs and explicit neural network models. In this paper, leveraging recent advances in random matrix theory (RMT), we perform an in-depth analysis of the conjugate kernel (CK) and neural tangent kernel (NTK) matrices for implicit DEQs, when the input data are drawn from a high-dimensional Gaussian mixture. We prove that, in this setting, the spectral behavior of these Implicit-CKs and NTKs depend on the DEQ activation function and initial weight variances, *but only via a system of four nonlinear equations*. As a direct consequence of this theoretical result, we demonstrate that a shallow explicit network can be carefully designed to produce the same CK or NTK as a given DEQ. Despite derived here for Gaussian mixture data, empirical results show that the proposed theory and design principles also apply to popular real-world datasets.

1. Introduction

Recently, a novel approach in neural network (NN) design has gained prominence in the form of Implicit Neural Networks (Bai et al., 2019; El Ghaoui et al., 2021). As typical implicit NNs, deep equilibrium models (DEQs) introduce a paradigm shift by resembling an infinite-depth weight-shared model with input-injection. In contrast to traditional explicit NNs such as multi-layer perceptrons, recurrent neural networks, and residual networks, DEQs derive features by directly solving for fixed points. These fixed points represent equilibrium states in the NN's computation, bypassing conventional layer-by-layer forward propagation.

DEQs have demonstrated remarkable performance across a variety of applications, including computer vision (Bai et al., 2020; Xie et al., 2022), natural language processing (Bai et al., 2019), neural rendering (Huang et al., 2021), and solving inverse problems (Gilton et al., 2021). Despite the empirical success achieved by DEQs, our theoretical understanding of these implicit models is still limited. As a telling example, it remains unclear whether the training and/or generalization properties of implicit DEQs can be connected to those of explicit NN models. Bai et al. (2019) show that any deep explicit NN can be reformulated as an implicit DEQ with carefully-designed weight re-parameterization. Nonetheless, questions such as

- *whether general DEQs have advantages over explicit networks*, or
- *whether an equivalent explicit NN exists for a given implicit DEQ*,

remain largely open. Novel insights into these questions are strongly desired, since implicit DEQs incur significantly higher computational costs than explicit NNs during training and inference, as a consequence of their reliance on iterative solutions to fixed points (Micaelli et al., 2023; Fung et al., 2022; Ramzi et al., 2021; Bai et al., 2021).

In this paper, we provide **affirmative** answers to the two open questions above, by considering input data following a Gaussian mixture model; refer to Remark 3.9 in Section 3 for a precise statement. Building upon recent advances in random matrix theory (RMT), we investigate the high-dimensional behavior of DEQs by focusing on their conjugate kernel (CK) and neural tangent kernel (NTK) matrices. These matrices offers an analytical assessment of the convergence and generalization properties for both implicit

[1]School of Electronic Information and Communications, Huazhong University of Science and Technology, Wuhan, China. [2]Ezhou Industrial Technology Research Institute, Huazhong University of Science and Technology, Wuhan, China. [3]Department of CSE, Shanghai Jiao Tong University, Shanghai, China. [4]China-Austria Belt and Road Joint Laboratory on AI and AM, Hangzhou Dianzi University, Hangzhou, China. [5]Center for Applied Statistics and School of Statistics, Renmin University of China, Beijing, China. Correspondence to: Zhenyu Liao <zhenyu_liao@hust.edu.cn>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024.

and explicit NNs, when the networks are wide, see Jacot et al. (2018) and Remark 2.5 below for a detailed discussion. For input data drawn a K-class Gaussian mixture model (GMM), we show, in the high-dimensional regime where the data dimension p and their size n are both large and comparable, that the Implicit-CKs and NTKs of DEQs can be evaluated via more accessible random matrix models that *only* depend on the variance parameter and the activation function via a system of *four* equations. Possibly more surprisingly, these high-dimensional "proxies" of Implicit-CKs and NTKs have consistent forms with those of explicit NNs recently established in Ali et al. (2022); Gu et al. (2022).

Inspired by this observation, we establish the high-dimensional equivalence (in the sense of the CK and/or NTK) between implicit DEQs and explicit models, by matching their determining equations derived above. In particular, our results reveal that a *shallow* explicit NN with carefully designed activations is destined to exhibit *identical* CK or NTK eigenspectral behavior as a given implicit DEQ, the depth of the latter being essentially *infinite*. This implies, at least for GMM data, that an equivalent *shallow* explicit NN (with the same amount of memory) can be designed, so as to avoid the significant computational overhead of implicit DEQs. Despite our theoretical results are derived for GMM data, we observe an unexpected close match between our theory and the empirical results on real-world datasets.

1.1. Our Contributions

Our contributions are summarized as follows.

(1) We provide, by considering high-dimensional GMM data, in Theorems 3.3 and 3.4, precise characterizations of CK and NTK matrices of implicit DEQs; we particularly show that the Implicit-CKs and NTKs *only* depend on the DEQ variance parameter and activation function via a system of *four* nonlinear equations.

(2) We present, in Section 3.2, a comprehensive methodology for crafting "equivalent" shallow NNs that emulate a given implicit DEQ. This involves determining the explicit NN activations through the system of equations derived in Theorems 3.3 and 3.4. We further illustrate the versatility of this framework in Examples 3.10 and 3.11, showcasing its applications to widely-used ReLU and Tanh DEQs.

(3) We provide empirical evidence on GMM and real-world datasets such as MNIST, Fashion-MNIST, and CIFAR-10. Our numerical results demonstrate that the carefully-designed explicit NNs exhibit performance on par with the given implicit DEQs. This parity in performance is observed across both GMM and diverse realistic datasets, affirming the broad applicability and effectiveness of the proposed framework.

1.2. Related Works

Here, we provide a brief review of related previous efforts.

Neural tangent kernels. Neural Tangent Kernel (NTK), initially proposed by Jacot et al. (2018), examines the behavior of wide and deep NNs when trained using gradient descent with small steps. Originally developed for fully-connected NNs, the NTK framework has since then been expanded to convolutional (Arora et al., 2019), graph (Du et al., 2019), and recurrent (Alemohammad et al., 2020) settings. See also Remark 2.5 below for the use of NTK in the analysis of DNNs.

Over-parameterized DEQs. Feng & Kolter (2020) extend previous NTK studies to implicit DEQs and derive the exact expressions of the CK and NTK of ReLU DEQs. Agarwala & Schoenholz (2022) investigate the NTK of DEQs under different random initializations. These studies particularly asserts that (i) the Implicit-NTKs of DEQs are equivalent to the corresponding weight-untied models in the infinitely wide regime and (ii) implicit DEQs have non-degenerate NTKs even in the infinite depth limit. These observations align with our findings. The connections between Implicit-CKs/NTKs and Explicit-CKs/NTKs, however, remain unexplored. Here we perform a fine-grained analysis on the Implicit-CKs and NTKs of DEQs and establish their equivalence to explicit NN model. Also, while training dynamics (and global convergence) of over-parameterized DEQs have been investigated in previous works (Gao et al., 2022; Gao & Gao, 2022; Ling et al., 2023; Truong, 2023) in the NTK regime, it remain unclear how these DEQ training dynamics distinguishes from those of explicit models.

Random matrix theory and NNs. Random matrix theory (RMT) has emerged as a versatile and potent tool for evaluating the behavior of large-scale systems characterized by a substantial "degree of freedom." Its application has been increasingly embraced in the realm of NN analysis, spanning shallow (Pennington & Worah, 2017; Liao & Couillet, 2018b;a) and deep (Benigni & Péché, 2019; Fan & Wang, 2020; Pastur, 2022; Pastur & Slavin, 2023) models, homogeneous (*e.g.,* standard normal) (Pennington & Worah, 2017; Mei & Montanari, 2022) and mixture-type datasets (Liao & Couillet, 2018b; Ali et al., 2022; Gu et al., 2022). From a technical perspective, the most relevant papers are Ali et al. (2022) and Gu et al. (2022), in which the eigenspectra of CK and NTK were evaluated, for explicit single-hidden-layer NN in Ali et al. (2022) and explicit deep NNs with multiple (but finite) layers in Gu et al. (2022). Here, we extend previous analysis to implicit DEQs with an effectively *infinite* number of layers, and establish an equivalence between implicit and explicit NN models.

2. Preliminaries

Notations. We use $\mathcal{N}(0, \boldsymbol{I})$ to denote standard multivariate Gaussian distribution. For a vector $\boldsymbol{v}$, $\|\boldsymbol{v}\|$ is the Euclidean norm of $\boldsymbol{v}$. For a matrix $\boldsymbol{A}$, we use $\boldsymbol{A}_{ij}$ to denote its (i, j)-th entry, $\|\boldsymbol{A}\|_F$ to denote its Frobenius norm, and $\|\boldsymbol{A}\|$ to denote its spectral norm. We use $\odot$ to denote the Hadamard product between matrices of the same size. We let $\mathcal{O}(\cdot)$, $\Theta(\cdot)$ and $\Omega(\cdot)$ denote standard Big-O, Big-Theta, and Big-Omega notations, respectively.

In this paper, we focus on the DEQ model introduced in Bai et al. (2019), defined as follows.

Definition 2.1 (Deep equilibrium model, DEQ)**.** Let $\boldsymbol{X} = [\boldsymbol{x}_1, \cdots, \boldsymbol{x}_n] \in \mathbb{R}^{p\times n}$ denote the input data, consider a vanilla DEQ with output $f(\boldsymbol{x}_i)$ given by

$$f(\boldsymbol{x}_i) = \boldsymbol{a}^\top \boldsymbol{z}_i^*, \tag{1}$$

where $\boldsymbol{a} \in \mathbb{R}^m$ and $\boldsymbol{z}_i^{(*)} \triangleq \lim_{l\to\infty} \boldsymbol{z}_i^{(l)} \in \mathbb{R}^m$ with

$$\boldsymbol{z}_i^{(l)} = \frac{1}{\sqrt{m}}\phi\left(\sigma_a \boldsymbol{A}\boldsymbol{z}_i^{(l-1)} + \sigma_b \boldsymbol{B}\boldsymbol{x}_i\right) \in \mathbb{R}^m, \text{ for } l \geq 1, \tag{2}$$

for some appropriate initialization $\boldsymbol{z}_i^{(0)}$. Here, $\boldsymbol{A} \in \mathbb{R}^{m\times m}$ and $\boldsymbol{B} \in \mathbb{R}^{m\times p}$ are the DEQ weight parameters, $\sigma_a, \sigma_b \in \mathbb{R}$ are constants, and ϕ is an element-wise activation. Note that $\boldsymbol{z}_i^*$ can also be determined as the equilibrium point of

$$\boldsymbol{z}_i^* = \frac{1}{\sqrt{m}}\phi\left(\sigma_a \boldsymbol{A}\boldsymbol{z}_i^* + \sigma_b \boldsymbol{B}\boldsymbol{x}_i\right). \tag{3}$$

We position ourselves under the following assumptions on the weights and activation functions of the DEQ.

Assumption 2.2 (Weights initialization)**.** For the DEQ model in Definition 2.1, the weight parameters $\boldsymbol{a} \in \mathbb{R}^m$, $\boldsymbol{A} \in \mathbb{R}^{m\times m}$ and $\boldsymbol{B} \in \mathbb{R}^{m\times p}$ are initialized as independent random vector or matrices having *i.i.d.* entries of zero mean, unit variance, and finite fourth-order moment.

Assumption 2.3 (Activation function)**.** For the DEQ model in Definition 2.1, the activation function ϕ is L_1-Lipschitz, and at least four-times weakly differentiable with respect to standard normal measure, *i.e.,* $\max_{k\in\{0,1,2,3,4\}} |\mathbb{E}[\phi^{(k)}(\xi)]| < \infty$ for $\xi \sim \mathcal{N}(0, 1)$.

Here, we consider the weak differentiability of a function, which generalizes the notation of derivative for non-differentiable (but integrable) functions. Specifically, using the Gaussian integration by parts formula, one has $\mathbb{E}[\phi'(\xi)] = \mathbb{E}[\xi\phi(\xi)]$ for $\xi \sim \mathcal{N}(0, 1)$, as long as the right-hand side expectation exists. It can be checked that Assumption 2.3 holds for commonly-used smooth, *e.g.*, Tanh, and piecewise linear activations, *e.g.*, ReLU and Leaky ReLU.

For the stability of training and inference of DEQs, it is of crucial significance to guarantee the existence and uniqueness of the equilibrium point in Eq. (3) (Winston & Kolter, 2020; El Ghaoui et al., 2021). To that end, we introduce the following assumption on the variance parameter σ_a.

Assumption 2.4 (Variance parameter)**.** For the DEQ model in Definition 2.1, the variance parameter σ_a in Eq. (3) satisfies $\sigma_a^2 < 1/(4L_1^2)$, with L_1 the Lipschitz constant of the activation function ϕ as demanded in Assumption 2.3.

It follows from Assumption 2.2 and standard singular value bounds of random matrices (Vershynin, 2018) that $\|\boldsymbol{A}\| \leq 2\sqrt{m}$ with high probability. Then, by noting that $\phi(\cdot)$ is L_1-Lipschitz, one has, under Assumption 2.4, that the transformation in Eq. (2) is a *contractive* mapping. This thus ensures the existence of the unique fixed point $\boldsymbol{z}^*$.

We are interested in the conjugate kernel and the neural tangent kernel (Implicit-CK and Implicit-NTK, for short) of the implicit DEQ in Definition 2.1.

Remark 2.5 (On CKs and NTKs). Conjugate kernels (CKs) and neural tangent kernels (NTKs) are closely related kernels useful in the analysis of NNs (Fan & Wang, 2020). During gradient descent training, the network parameters change and the NTK also evolves over time. It has been shown in Jacot et al. (2018) and follow-up works that for sufficiently wide DNNs trained on gradient descent with small learning rate: (i) the NTK is approximately constant after initialization; and (ii) running gradient descent to update the network parameters is *equivalent* to kernel gradient descent with the NTK. This duality allows one to assess the training dynamics, generalization, and predictions of wide DNNs as *closed-form* expressions involving NTK eigenvalues and eigenvectors, see Bartlett et al. (2021, Section 6).

For Implicit-CKs and NTKs, we recall the following result.

Proposition 2.6 (Implicit-CKs and NTKs of DEQ, (Feng & Kolter, 2020; Gao et al., 2023))**.** *Under Assumptions 2.2-2.4, the Implicit-CK of the DEQ model in Definition 2.1 takes the following form:*

$$\boldsymbol{G}^* = \lim_{l\to\infty} \boldsymbol{G}^{(l)}, \tag{4}$$

where the (i, j)-th entry of $\boldsymbol{G}^{(l)}$ is defined recursively as[1] *$\boldsymbol{G}_{ij}^{(l)} = \mathbb{E}[(\boldsymbol{z}_i^{(l)})^\top \boldsymbol{z}_j^{(l)}]$, i.e., $\boldsymbol{G}_{ij}^{(0)} = (\boldsymbol{z}_i^{(0)})^\top \boldsymbol{z}_j^{(0)}$ and*

$$\boldsymbol{G}_{ij}^{(l)} = \mathbb{E}_{(\mathrm{u}_l, \mathrm{v}_l)}[\phi(\mathrm{u}_l)\phi(\mathrm{v}_l)], \tag{5}$$

with $(\mathrm{u}_l, \mathrm{v}_l) \sim \mathcal{N}\left(0, \begin{bmatrix} \boldsymbol{\Lambda}_{ii}^{(l)} & \boldsymbol{\Lambda}_{ij}^{(l)} \\ \boldsymbol{\Lambda}_{ji}^{(l)} & \boldsymbol{\Lambda}_{jj}^{(l)} \end{bmatrix}\right)$ and $\boldsymbol{\Lambda}_{ij}^{(l)} = \sigma_a^2 \boldsymbol{G}_{ij}^{(l-1)} + \sigma_b^2 \boldsymbol{x}_i^\top \boldsymbol{x}_j$, for $l \geq 1$. The corresponding Implicit-NTK takes the form $\boldsymbol{K}^* = \lim_{l\to\infty} \boldsymbol{K}^{(l)}$ whose the (i, j)-th entry is defined as

$$\boldsymbol{K}_{ij}^{(l)} = \sum_{h=1}^{l+1} \left(\boldsymbol{G}_{ij}^{(h-1)} \prod_{h'=h}^{l+1} \dot{\boldsymbol{G}}_{ij}^{(h')}\right), \tag{6}$$

[1] Note that the expectation is conditioned on the input data, and is taken with respect to the random weights.

with $\dot{\boldsymbol{G}}_{ij}^{(l)} = \sigma_a^2 \mathbb{E}_{(\mathrm{u}_l,\mathrm{v}_l)\sim\mathcal{N}(0,\boldsymbol{\Lambda}_{ij}^{(l)})}[\phi'(\mathrm{u}_l)\phi'(\mathrm{v}_l)]$ *so that*

$$\boldsymbol{K}_{ij}^* \equiv \boldsymbol{G}_{ij}^*/(1-\dot{\boldsymbol{G}}_{ij}^*). \tag{7}$$

The existence and uniqueness of the DEQ Implicit-CK and NTK expressions given in Proposition 2.6 are guaranteed under Assumptions 2.2-2.4, see Gao et al. (2023) for a detailed proof using a Gaussian process argument.

For the purpose of our theoretical analysis, we consider input data drawn from the following high-dimensional Gaussian mixture model.

Assumption 2.7 (High-dimensional Gaussian mixture model, GMM)**.** Consider n data vectors $\boldsymbol{x}_1, \cdots, \boldsymbol{x}_n \in \mathbb{R}^p$ independently drawn from one of the K-class Gaussian mixtures $\mathcal{C}_1, \cdots, \mathcal{C}_K$, *i.e.*, for $\boldsymbol{x}_i \in \mathcal{C}_a$, we have

$$\sqrt{p}\boldsymbol{x}_i \sim \mathcal{N}(\boldsymbol{\mu}_a, \boldsymbol{C}_a), \quad a \in \{1, \cdots, K\}. \tag{8}$$

We assume, for n, p both large that (i) $p = \Theta(n)$ and n_a the cardinality of class $\mathcal{C}_a$ satisfies $n_a = \Theta(n)$; (ii) $\|\boldsymbol{\mu}_a\| = \mathcal{O}(1)$; (iii) for $\boldsymbol{C}^\circ \equiv \sum_{a=1}^K \frac{n_a}{n}\boldsymbol{C}_a$ and $\boldsymbol{C}_a^\circ \equiv \boldsymbol{C}_a - \boldsymbol{C}^\circ$, we have $\|\boldsymbol{C}_a\| = \mathcal{O}(1)$, $\operatorname{tr}\boldsymbol{C}_a^\circ = \mathcal{O}(\sqrt{p})$ and $\operatorname{tr}(\boldsymbol{C}_a\boldsymbol{C}_b) = \mathcal{O}(p)$ for $a, b \in \{1, \cdots, K\}$.

Remark 2.8 (On GMM in Assumption 2.7). The normalization by $\sqrt{p}$ of the GMM in Eq. (8) is commonly used in the literature of high-dimensional statistics and over-parameterized DNNs and ensures that the data vectors have bounded Euclidean norms $\|\boldsymbol{x}_i\| = \mathcal{O}(1)$ with high probability for n, p large. The assumptions on the scaling of the means and covariances in Assumption 2.7, despite being technical at first sight, ensure the GMM classification in Eq. (8) remains non-trivial for n, p large, and have been extensively studied in the literature for various ML methods ranging from LDA, spectral clustering, SVM, to shallow and deep neural networks, see for example (Couillet & Benaych-Georges, 2016; Louart et al., 2018; Dobriban & Wager, 2018; Liao & Couillet, 2019; Elkhalil et al., 2020; Couillet & Liao, 2022; Gu et al., 2022) as well as (Blum et al., 2020, Section 2). On the other hand, it is known that GMM is a universal approximator, in that given a data distribution, there exists a GMM (with possibly a large number of components) that can approximate that distribution to an arbitrary error, see for example (Goodfellow et al., 2016). In the high-dimensional regime under study where the data dimension p and sample size n are both large and comparable, theoretical and empirical evidences have been provided to support the modeling of realistic image data using high-dimensional GMM, see (Seddik et al., 2020).

3. Main Results

In this section, we present in Section 3.1 our main technical results on the high-dimensional characterization of CK and NTK matrices of implicit DEQs, in Theorems 3.3 and 3.4, respectively. We then show in Section 3.2 that the proposed theoretical analysis allows to construct, for a given implicit DEQ model, an equivalent and shallow *explicit* NN model that shares the same CK eigenspectra as the implicit DEQ.

3.1. High-dimensional Characterization of Implicit-CK and NTK Matrices

Let us start by introducing some notations that will be used in the remainder of this paper. For GMM data in Assumption 2.7, denote

$$\boldsymbol{J} \equiv [\boldsymbol{j}_1, \cdots, \boldsymbol{j}_K] \in \mathbb{R}^{n\times K}, \quad \boldsymbol{j}_a \in \mathbb{R}^n,$$

with $[\boldsymbol{j}_a]_i = 1_{\boldsymbol{x}_i\in\mathcal{C}_a}$ of class $\mathcal{C}_a$, $a \in \{1, \ldots, K\}$ (note that the rows of $\boldsymbol{J}$ are standard one-hot-encoded label vectors in $\mathbb{R}^K$). We define the second-order data fluctuation vector as

$$\boldsymbol{\psi} \equiv \{\|\boldsymbol{x}_i - \mathbb{E}[\boldsymbol{x}_i]\|^2 - \mathbb{E}[\|\boldsymbol{x}_i - \mathbb{E}[\boldsymbol{x}_i]\|^2]\}_{i=1}^n \in \mathbb{R}^n,$$

and use

$$\boldsymbol{T} = \{\operatorname{tr}\boldsymbol{C}_a\boldsymbol{C}_b/p\}_{a,b=1}^K \in \mathbb{R}^{K\times K}, \; \boldsymbol{t} = \{\operatorname{tr}\boldsymbol{C}_a^\circ/\sqrt{p}\} \in \mathbb{R}^K,$$

to denote the GMM second-order statistics. Also, let

$$\tau_0 \equiv \sqrt{\operatorname{tr}\boldsymbol{C}^\circ/p}, \tag{9}$$

for $\boldsymbol{C}^\circ \equiv \sum_{a=1}^K \frac{n_a}{n}\boldsymbol{C}_a$ as in Assumption 2.7, and τ_* be the fixed point to the following equation

$$\tau_* = \sqrt{\sigma_a^2\mathbb{E}\left[\phi^2(\tau_*\xi)\right] + \sigma_b^2\tau_0^2}, \quad \xi \sim \mathcal{N}(0,1), \tag{10}$$

the existence and uniqueness of which is ensured under Assumption 3.1 as follows.

Assumption 3.1. For the DEQ model in Definition 2.1, the variance parameter σ_a satisfies $\sigma_a^2 < 2/(\mathbb{E}[(\phi^2(\tau\xi))''])$ for $\tau > 0$ and $\xi \sim \mathcal{N}(0,1)$.

Remark 3.2 (Existence and uniqueness of τ_*). It can be checked that for any given $\tau_0 > 0$ and variance parameter σ_a satisfying Assumption 3.1, the right-hand side of Eq. (10) constitutes a *contractive mapping*. This ensures the existence of a unique fixed point τ_* in Eq. (10). See Lemma A.2 in Appendix A for a detailed proof of this fact.

With these notations, we are ready to present our first result on the high-dimensional characterization of CK matrices for implicit DEQs, the proof of which is given in Appendix B.

Theorem 3.3 (High-dimensional approximation of Implicit-CKs)**.** *For the DEQ model in Definition 2.1 with GMM input as in Assumption 2.7, let Assumptions 2.2 and 2.3 hold, and let the activation ϕ be centered such that $\mathbb{E}[\phi(\tau_*\xi)] = 0$ for $\xi \sim \mathcal{N}(0,1)$ and τ_* in Eq. (10). Further assume that the variance parameter σ_a satisfies both Assumptions 2.4 and 3.1. Then, the Implicit-CK matrix $\boldsymbol{G}^*$ defined*

in Eq. (4) of Proposition 2.6 can be well approximated, in a spectral norm sense with $\|\boldsymbol{G}^* - \overline{\boldsymbol{G}}\| = \mathcal{O}(n^{-1/2})$*, by a random matrix* $\overline{\boldsymbol{G}}$ explicitly *given by*

$$\overline{\boldsymbol{G}} \equiv \alpha_{*,1}\boldsymbol{X}^\top\boldsymbol{X} + \boldsymbol{V}\boldsymbol{C}_*\boldsymbol{V}^\top + (\gamma_*^2 - \tau_0^2\alpha_{*,1})\boldsymbol{I}_n, \quad (11)$$

where $\boldsymbol{V} = \big[\boldsymbol{J}/\sqrt{p},\ \boldsymbol{\psi}\big] \in \mathbb{R}^{n\times(K+1)}$,

$$\boldsymbol{C}_* = \begin{bmatrix} \alpha_{*,2}\boldsymbol{t}\boldsymbol{t}^\top + \alpha_{*,3}\boldsymbol{T} & \alpha_{*,2}\boldsymbol{t} \\ \alpha_{*,2}\boldsymbol{t}^\top & \alpha_{*,2} \end{bmatrix} \in \mathbb{R}^{(K+1)\times(K+1)},$$

and non-negative scalars $\gamma_*, \alpha_{*,1}, \alpha_{*,2}, \alpha_{*,3} \geq 0$ *are defined, for* $\xi \sim \mathcal{N}(0,1)$*, as*

$$\begin{aligned} \gamma_* &= \sqrt{\mathbb{E}[\phi^2(\tau_*\xi)]},\ \alpha_{*,1} = \frac{\sigma_b^2\mathbb{E}[\phi'(\tau_*\xi)]^2}{1-\sigma_a^2\mathbb{E}[\phi'(\tau_*\xi)]^2}, \\ \alpha_{*,2} &= \frac{\mathbb{E}[\phi''(\tau_*\xi)]^2}{4(1-\sigma_a^2\mathbb{E}[\phi'(\tau_*\xi)]^2)}\alpha_{*,4}^2, \\ \alpha_{*,3} &= \frac{\mathbb{E}[\phi''(\tau_*\xi)]^2}{2(1-\sigma_a^2\mathbb{E}[\phi'(\tau_*\xi)]^2)}(\sigma_a^2\alpha_{*,1}+\sigma_b^2)^2 \end{aligned} \quad (12)$$

with $\alpha_{*,4} = (1 - \frac{\sigma_a^2}{2}\mathbb{E}[(\phi^2(\tau_*\xi))''])^{-1}\sigma_b^2$.

Theorem 3.3 reveals the surprising fact that, for high-dimensional GMM input in Assumption 2.7, the Implicit-CK $\boldsymbol{G}^*$, despite its mathematically involved form (as the fixed point of the recursion) in Eq. (4), is close to a much simpler random matrix $\overline{\boldsymbol{G}}$. This "equivalent" CK matrix $\overline{\boldsymbol{G}}$,

(1) depends, as expected, on the input GMM data ($\boldsymbol{X}$), their class structure ($\boldsymbol{J}$) and higher-order statistics ($\boldsymbol{t}$ and $\boldsymbol{T}$), but in a rather *explicit* fashion; and

(2) is *independent* of the distribution of the (randomly initialized) weight matrices $\boldsymbol{A}$ and $\boldsymbol{B}$; and

(3) depends on σ_a^2, σ_b^2, and the activation ϕ *only* via four scalars $\alpha_{*,1}, \alpha_{*,2}, \alpha_{*,3}, \gamma_*$ *explicitly* given in Eq. (12).

A similar result can be derived for the Implicit-NTK matrices and is given as follows, proven in Appendix C.

Theorem 3.4 (High-dimensional approximation of Implicit-NTKs)**.** *Under the same settings and notations of Theorem 3.3, we have, that the Implicit-NTK matrix* $\boldsymbol{K}^*$ *defined in Eq. (7) of Proposition 2.6 can be well approximated, in a spectral norm sense with* $\|\boldsymbol{K}^* - \overline{\boldsymbol{K}}\| = \mathcal{O}(n^{-1/2})$*, by a random matrix* $\overline{\boldsymbol{K}}$ explicitly *given by*

$$\overline{\boldsymbol{K}} \equiv \beta_{*,1}\boldsymbol{X}^\top\boldsymbol{X} + \boldsymbol{V}\boldsymbol{D}_*\boldsymbol{V}^\top + (\kappa_*^2 - \tau_0^2\beta_{*,1})\boldsymbol{I}_n, \quad (13)$$

where $\boldsymbol{V} \in \mathbb{R}^{n\times(K+1)}$ *is as defined in Theorem 3.3, and*

$$\boldsymbol{D}_* = \begin{bmatrix} \beta_{*,2}\boldsymbol{t}\boldsymbol{t}^\top + \beta_{*,3}\boldsymbol{T} & \beta_{*,2}\boldsymbol{t} \\ \beta_{*,2}\boldsymbol{t}^\top & \beta_{*,2} \end{bmatrix} \in \mathbb{R}^{(K+1)\times(K+1)},$$

as well as non-negative scalars $\kappa_*, \beta_{*,1}, \beta_{*,2}, \beta_{*,3} \geq 0$*,*

$$\begin{aligned} \kappa_* &= \frac{\tau_*}{\sqrt{1-\sigma_a^2\mathbb{E}[\phi'(\tau_*\xi)^2]}},\ \beta_{*,1} = \frac{\alpha_{*,1}}{1-\sigma_a^2\mathbb{E}[\phi'(\tau_*\xi)]^2}, \\ \beta_{*,2} &= \frac{\alpha_{*,2}}{1-\sigma_a^2\mathbb{E}[\phi'(\tau_*\xi)]^2}, \\ \beta_{*,3} &= \frac{\alpha_{*,3} + \beta_{*,1}(\sigma_a^2\mathbb{E}[\phi''(\tau_*\xi)]^2 + \sigma_b^2)\alpha_{*,1}}{1-\sigma_a^2\mathbb{E}[\phi'(\tau_*\xi)]^2}, \end{aligned}$$

for $\xi \sim \mathcal{N}(0,1)$*, with* $\alpha_{*,1}, \alpha_{*,2}, \alpha_{*,3}$ *as defined in Eq. (12).*

Theorem 3.4 tells us that the NTK matrices of implicit DEQs take a similar form as their CKs, and (approximately for n, p large) depend on σ_a, σ_b and the activation via the key parameters $\beta_{*,1}, \beta_{*,2}, \beta_{*,3}$ and κ_*.

Remark 3.5 (On centered activation). Given any activation function $\tilde{\phi}(\cdot)$ that satisfies Assumption 2.3, a centered activation ϕ can be obtained by simplify subtracting a constant as $\phi(x) = \tilde{\phi}(x) - \mathbb{E}[\tilde{\phi}(\tau_* x)]$ with $\tau_* = \sqrt{\sigma_a^2\mathbb{E}[(\tilde{\phi}(\tau_*\xi) - \mathbb{E}[\tilde{\phi}(\tau_*\xi)])^2] + \sigma_b^2\tau_0^2}$.

3.2. High-dimensional Equivalence between DEQs and Shallow Explicit Networks

Implicit DEQs are known, per Definition 2.1, to be formally equivalent to *infinitely* deep *explicit* NN models (Bai et al., 2020; Xie et al., 2022). In the sequel, we show how our theoretical analyses in Theorems 3.3 and 3.4 provide a general recipe to construct *shallow explicit* NN models that are "equivalent" to a given *implicit* DEQ model, in the sense that the CK and/or NTK matrices of the two networks are close in spectral norm. We first review previous results on explicit NN models in Section 3.2.1, and present, in Section 3.2.2, guidelines to construct a shallow explicit NN equivalent to a given DEQ.

3.2.1. A BRIEF REVIEW OF EXPLICIT CKS AND NTKS

We consider the following L-layer *explicit* NN model.

Definition 3.6 (Fully-connected explicit NN model)**.** Let $\boldsymbol{X} = [\boldsymbol{x}_1, \cdots, \boldsymbol{x}_n] \in \mathbb{R}^{p\times n}$ denote the input data, consider an L-layer fully-connected explicit NN model with output given by $\boldsymbol{a}^\top\boldsymbol{x}_i^{(L)}$ for $\boldsymbol{a} \in \mathbb{R}^{m_L}$, $\boldsymbol{x}_i^{(0)} = \boldsymbol{x}_i$ and

$$\boldsymbol{x}_i^{(l)} = \frac{1}{\sqrt{m_l}}\sigma_l(\boldsymbol{W}_l\boldsymbol{x}_i^{(l-1)}), \quad \text{for } l = 1, \cdots, L, \quad (14)$$

where $\boldsymbol{W}_l \in \mathbb{R}^{m_l\times m_{l-1}}$ are weight matrices and $\sigma_l : \mathbb{R} \to \mathbb{R}$ are element-wise activation functions.

As in Assumptions 2.2 and 2.3 for implicit DEQs, we also assume that the weights $\boldsymbol{W}_l$s in Definition 3.6 have *i.i.d.* entries of zero mean, unit variance, and finite fourth-order moment; and the activations σ_l are four-times weakly differentiable with respect to standard Gaussian measure.

For fully-connected explicit NNs in Definition 3.6, we recall the following result on their Explicit-CKs and NTKs.

Proposition 3.7 (Explicit-CKs and NTKs, (Jacot et al., 2018; Fan & Wang, 2020))**.** *For a fully-connected L-layer NN model in Definition 3.6, its Explicit-CK matrix $\boldsymbol{\Sigma}^{(l)}$ at layer $l \in \{1, \cdots, L\}$ is defined as*

$$\boldsymbol{\Sigma}_{ij}^{(l)} = \mathbb{E}_{\mathrm{u},\mathrm{v}}[\sigma_l(\mathrm{u}_l)\sigma_l(\mathrm{v}_l)], \quad \boldsymbol{\Sigma}^{(0)} = \boldsymbol{X}^\top \boldsymbol{X}, \tag{15}$$

with $(\mathrm{u}_l, \mathrm{v}_l) \sim \mathcal{N}\Big(0, \begin{bmatrix} \boldsymbol{\Sigma}_{ii}^{(l-1)} & \boldsymbol{\Sigma}_{ij}^{(l-1)} \\ \boldsymbol{\Sigma}_{ji}^{(l-1)} & \boldsymbol{\Sigma}_{jj}^{(l-1)} \end{bmatrix}\Big)$. *And the Explicit-NTK matrix $\boldsymbol{\Theta}^{(l)}$ at layer l is defined as*

$$\boldsymbol{\Theta}^{(l)} = \boldsymbol{\Sigma}^{(l)} + \boldsymbol{\Theta}^{(l-1)} \odot \dot{\boldsymbol{\Sigma}}^{(l)}, \quad \boldsymbol{\Theta}^{(0)} = \boldsymbol{X}^\top \boldsymbol{X}, \tag{16}$$

with $\dot{\boldsymbol{\Sigma}}_{ij}^{(l)} = \mathbb{E}_{\mathrm{u}_l,\mathrm{v}_l}[\sigma_l'(\mathrm{u}_l)\sigma_l'(\mathrm{v}_l)]$.

The Explicit-CK and NTK matrices in Proposition 3.7 for the fully-connected explicit NN model in Definition 3.6 have been recently studied in Gu et al. (2022).

Theorem 3.8 (High-dimensional approximation of Explicit-CKs, (Gu et al., 2022, Theorem 1))**.** *For fully-connected NN model in Definition 3.6 with GMM input in Assumption 2.7, let $\widetilde{\tau}_0 = \tau_0$ as defined in Eq. (9) and $\widetilde{\tau}_1, \cdots \widetilde{\tau}_L \geq 0$ be a sequence of non-negative scalars satisfying $\widetilde{\tau}_l = \sqrt{\mathbb{E}[\sigma_l^2(\widetilde{\tau}_{l-1}\xi)]}$, for $\xi \sim \mathcal{N}(0,1)$ and $l \in \{1, \cdots, L\}$. Further assume that the activation functions $\sigma_l(\cdot)$ are centered such that $\mathbb{E}[\sigma_l(\widetilde{\tau}_{l-1}\xi)] = 0$. Then, for the Explicit-CK matrix $\boldsymbol{\Sigma}^{(l)}$ defined in Eq. (15) of Proposition 3.7, it holds that $\|\boldsymbol{\Sigma}^{(l)} - \overline{\boldsymbol{\Sigma}}^{(l)}\| = \mathcal{O}\big(n^{-1/2}\big)$ for a random matrix $\overline{\boldsymbol{\Sigma}}^{(l)}$* explicitly *given by*

$$\overline{\boldsymbol{\Sigma}}^{(l)} = \widetilde{\alpha}_{l,1}\boldsymbol{X}^\top \boldsymbol{X} + \boldsymbol{V}\widetilde{\boldsymbol{C}}_l \boldsymbol{V}^\top + (\widetilde{\tau}_l^2 - \tau_0^2 \widetilde{\alpha}_{l,1})\boldsymbol{I}_n, \tag{17}$$

with $\boldsymbol{V} \in \mathbb{R}^{n\times(K+1)}$ as defined in Theorem 3.3,

$$\widetilde{\boldsymbol{C}}_l = \begin{bmatrix} \widetilde{\alpha}_{l,2}\boldsymbol{t}\boldsymbol{t}^\top + \widetilde{\alpha}_{l,3}\boldsymbol{T} & \widetilde{\alpha}_{l,2}\boldsymbol{t} \\ \widetilde{\alpha}_{l,2}\boldsymbol{t}^\top & \widetilde{\alpha}_{l,2} \end{bmatrix} \in \mathbb{R}^{(K+1)\times(K+1)},$$

and non-negative scalars $\widetilde{\alpha}_{l,1}, \widetilde{\alpha}_{l,2}, \widetilde{\alpha}_{l,3}$ defined recursively as $\widetilde{\alpha}_{0,1} = \widetilde{\alpha}_{0,4} = 1$, $\widetilde{\alpha}_{0,2} = \widetilde{\alpha}_{0,3} = 0$, and

$$\widetilde{\alpha}_{l,1} = \mathbb{E}[\sigma_l'(\widetilde{\tau}_{l-1}\xi)]^2 \widetilde{\alpha}_{l-1,1},$$

$$\widetilde{\alpha}_{l,2} = \mathbb{E}[\sigma_l'(\widetilde{\tau}_{l-1}\xi)]^2 \widetilde{\alpha}_{l-1,2} + \frac{1}{4}\mathbb{E}[\sigma_l''(\widetilde{\tau}_{l-1}\xi)]^2 \widetilde{\alpha}_{l-1,4}^2,$$

$$\widetilde{\alpha}_{l,3} = \mathbb{E}[\sigma_l'(\widetilde{\tau}_{l-1}\xi)]^2 \widetilde{\alpha}_{l-1,3} + \frac{1}{2}\mathbb{E}[\sigma_l''(\widetilde{\tau}_{l-1}\xi)]^2 \widetilde{\alpha}_{l-1,1}^2,$$

with $\widetilde{\alpha}_{l,4} = \mathbb{E}[(\sigma_l^2(\widetilde{\tau}_{l-1}\xi))'']\widetilde{\alpha}_{l-1,4}$, for $\xi \sim \mathcal{N}(0,1)$.

In the following, we establish, by combining Theorems 3.3 and 3.8, *explicit* connections between the CK matrices of implicit DEQs and fully-connected explicit NNs. Exploiting this connection, we further provide a recipe to construct an explicit network "equivalent" to any given DEQ, with approximately the same CK. Results for NTK can be similarly obtained by combining our Theorem 3.4 with Gu et al. (2022, Theorem 2) and is omitted here.

3.2.2. Designing Equivalent Explicit NNs via CK matching

Comparing Theorem 3.3 to Theorem 3.8, we see that the high-dimensional approximation $\overline{\boldsymbol{G}}$ of the Implicit-CK in Eq. (11) takes a consistent form with that ($\overline{\boldsymbol{\Sigma}}^{(l)}$) of the Explicit-CK in Eq. (17), with coefficients α_*s and $\widetilde{\alpha}$s determined by the corresponding activation ϕ and σ_l, respectively. Inspired by this observation, our idea is to design activations of an L-layer explicit NN such that its Explicit-CK $\boldsymbol{\Sigma}^{(L)}$ shares the same coefficients as the Implicit-CK $\boldsymbol{G}^*$ of a given implicit DEQ of interest. Specifically, for a given implicit DEQ as in Definition 2.1,

(1) we first compute the four key parameters $\alpha_{*,1}, \alpha_{*,2}, \alpha_{*,3}$ and γ_* of the implicit DEQ according to Eq. (12) of Theorem 3.3;

(2) we then select activations σ_l with undetermined parameters for the L-layer explicit NN in Definition 3.6, and use Theorem 3.8 to represent $\widetilde{\alpha}_{L,1}, \widetilde{\alpha}_{L,1}, \widetilde{\alpha}_{L,1}, \widetilde{\tau}_L$ as functions of the activation parameters;

(3) we determine the activations σ_l of the explicit NN by solving the following set of equations,

$$\widetilde{\tau}_L = \gamma_*, \quad \widetilde{\alpha}_{L,i} = \alpha_{*,i},\ i \in \{1,2,3\}. \tag{18}$$

This gives the desired fully-connected explicit NN model that shares the same CK as the given DEQ.

It remains to determine the depth of the equivalent explicit NN model. Note, by comparing Theorem 3.8 to Theorem 3.3, that for a given implicit DEQ, it is *not always possible* to determine a single-hidden-layer explicit NN having the same CK. This is discussed in the following remark.

Remark 3.9 (Implicit- versus Explicit-CK). It follows from Theorem 3.8 that, for the single-hidden-layer explicit NN (with $L = 1$ in Definition 3.6), one *must* have $\widetilde{\alpha}_{1,2} = \frac{1}{2}\widetilde{\alpha}_{1,3}$, *regardless* of the choice of activation. On the contrast, $\alpha_{*,2} = \frac{1}{2}\alpha_{*,3}$ does *not* necessarily hold for the Implicit-CK of *all* DEQs. As such, for a given DEQ,

- if $\alpha_{*,2} = \frac{1}{2}\alpha_{*,3}$, then a single-hidden-layer explicit NN suffices to match the given DEQ;
- otherwise if $\alpha_{*,2} \neq \frac{1}{2}\alpha_{*,3}$, then an explicit NN with (at least) two hidden layers is required.

As a consequence of Remark 3.9, we discuss, in the following, the two instances of commonly used implicit DEQ with ReLU and Tanh activations, and illustrate how to construct equivalent shallow explicit NNs in both cases. The detailed expressions and proofs are given in Appendix D.

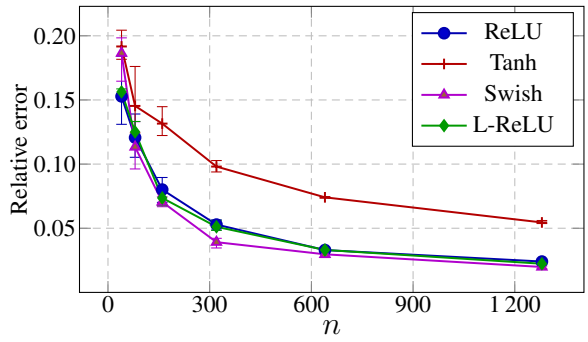

Figure 1. Evolution of relative spectral norm error $\|\boldsymbol{G}^* - \overline{\boldsymbol{G}}\|/\|\boldsymbol{G}^*\|$ *w.r.t.* sample size n, for DEQs in Definition 2.1 with different activations and $\sigma_a^2 = 0.2$, on two-class GMM, $p/n = 0.8$, $\boldsymbol{\mu}_a = [\mathbf{0}_{8(a-1)}; 8; \mathbf{0}_{p-8a+7}]$, and $\boldsymbol{C}_a = (1 + 8(a-1)/\sqrt{p})\boldsymbol{I}_p, a \in \{1, 2\}$.

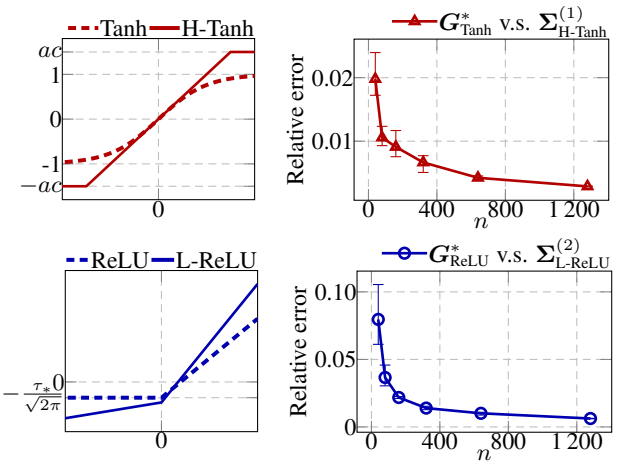

Figure 2. **Left:** Visualization of activations of DEQs (**dashed**) and those of equivalent explicit NNs (**solid**). **Right:** Evolution of relative spectral norm errors $\|\boldsymbol{G}^*_{\text{Tanh}} - \boldsymbol{\Sigma}^{(1)}_{\text{H-Tanh}}\|/\|\boldsymbol{G}^*_{\text{Tanh}}\|$ and $\|\boldsymbol{G}^*_{\text{ReLU}} - \boldsymbol{\Sigma}^{(2)}_{\text{L-ReLU}}\|/\|\boldsymbol{G}^*_{\text{ReLU}}\|$ *w.r.t.* sample size n on GMM as in Figure 1 for Example 3.10 (**red**) and Example 3.11 (**blue**), respectively.

Example 3.10 (DEQ with Tanh activation)**.** For a given implicit DEQ (denoted Tanh-DEQ) in Definition 2.1 with Tanh activation, *i.e.*, $\phi(x) = \text{Tanh}(x)$, a *single-hidden-layer* equivalent explicit NN (denoted H-Tanh-ENN) as in Definition 3.6, with Hard-Tanh-type activation:

$$\sigma_{\text{H-Tanh}}(x) \equiv ax \cdot 1_{-c \le x \le c} + ac \cdot (1_{x \ge c} - 1_{x \le -c}), \quad (19)$$

with undetermined parameters $a > 0, c \ge 0$, can be constructed so that their CKs, denoted as $\boldsymbol{G}^*_{\text{Tanh}}$ and $\boldsymbol{\Sigma}^{(1)}_{\text{H-Tanh}}$, satisfy $\|\boldsymbol{G}^*_{\text{Tanh}} - \boldsymbol{\Sigma}^{(1)}_{\text{H-Tanh}}\| = \mathcal{O}(n^{-1/2})$, by solving a system of nonlinear equations induced from Eq. (18).

Example 3.11 (DEQ with ReLU activation)**.** For a given implicit DEQ (denoted ReLU-DEQ) as in Definition 2.1 with centered ReLU activation, *i.e.*, $\phi(x) = \text{ReLU}(x) - \tau_*/\sqrt{2\pi}$, a *two-hidden-layer* equivalent explicit NN (denoted L-ReLU-ENN) with Leaky-ReLU-type activation:

$$\sigma^{(l)}_{\text{L-ReLU}}(x) \equiv \max(a_l x, b_l x) - \frac{a_l - b_l}{\sqrt{2\pi}}\widetilde{\tau}_l, \; l = 1, 2, \quad (20)$$

with undetermined parameters $a_l \ge b_l \ge 0$, can be constructed so that their CKs, denoted as $\boldsymbol{G}^*_{\text{ReLU}}$ and $\boldsymbol{\Sigma}^{(2)}_{\text{L-ReLU}}$, satisfy $\|\boldsymbol{G}^*_{\text{ReLU}} - \boldsymbol{\Sigma}^{(2)}_{\text{L-ReLU}}\| = \mathcal{O}(n^{-1/2})$, by solving a system of polynomial equations induced from Eq. (18).

4. Experiments

In this section, we provide numerical experiments to validate our theoretical results. We consider both Gaussian mixture data and samples drawn from commonly used real-world datasets such as MNIST (LeCun et al., 1998), Fashion-MNIST (Xiao et al., 2017), and CIFAR-10 (Krizhevsky, 2009). The experiments are conducted with a repetition of five trials, and we report both the average performance and accompanying error bars. Due to space limitation, we refer the readers to Appendix E for additional experiments. The code to reproduce the results in this section is available at `https://github.com/StephenLi24/INN_eqvi_ENN`.

High-dimensional approximations of Implicit-CKs and NTKs. Figure 1 compares the difference between Implicit-CKs $\boldsymbol{G}^*$ and their high-dimensional approximations $\overline{\boldsymbol{G}}$ given in Theorem 3.3, on binary Gaussian mixture data, for DEQs as Definition 2.1 with four commonly-used activations: ReLU, Tanh, Swish, and Leaky-ReLU (L-ReLU). The computation of $\overline{\boldsymbol{G}}$ follows from its definition in Theorem 3.3. For the Implicit-CK $\boldsymbol{G}^*$, we take an estimation approach similar to that in Gao et al. (2023): each element $\boldsymbol{G}^*_{ij}$ is estimated as $(\boldsymbol{z}^*_i)^\top \boldsymbol{z}^*_j$ using a high-dimensional DEQ defined in Eq. (3) with $m = 2^{12}$ and $\boldsymbol{z}^*_i$ estimated through a large number l of fixed-point iterations defined in Eq. (2). See Gao et al. (2023) for a convergence analysis of this estimation (to $\boldsymbol{G}^*$) *w.r.t.* the width m and the number l of fixed-point iterations. We refer the interested readers to Cho & Saul (2009); Tsuchida et al. (2018); Novak et al. (2019) for fast and efficient estimation/computation of CKs and NTKs.

We observe from Figure 1 that, for different activations, as n, p increase, the relative errors consistently and significantly decrease, as in line with our Theorem 3.3. The experimental observations regarding NTKs and Theorem 3.4 are similar and are placed in Appendix E.1. Possibly surprisingly, the high-dimensional approximations of Implicit-CKs and Implicit-NTKs, despite derived here for GMM in Theo-

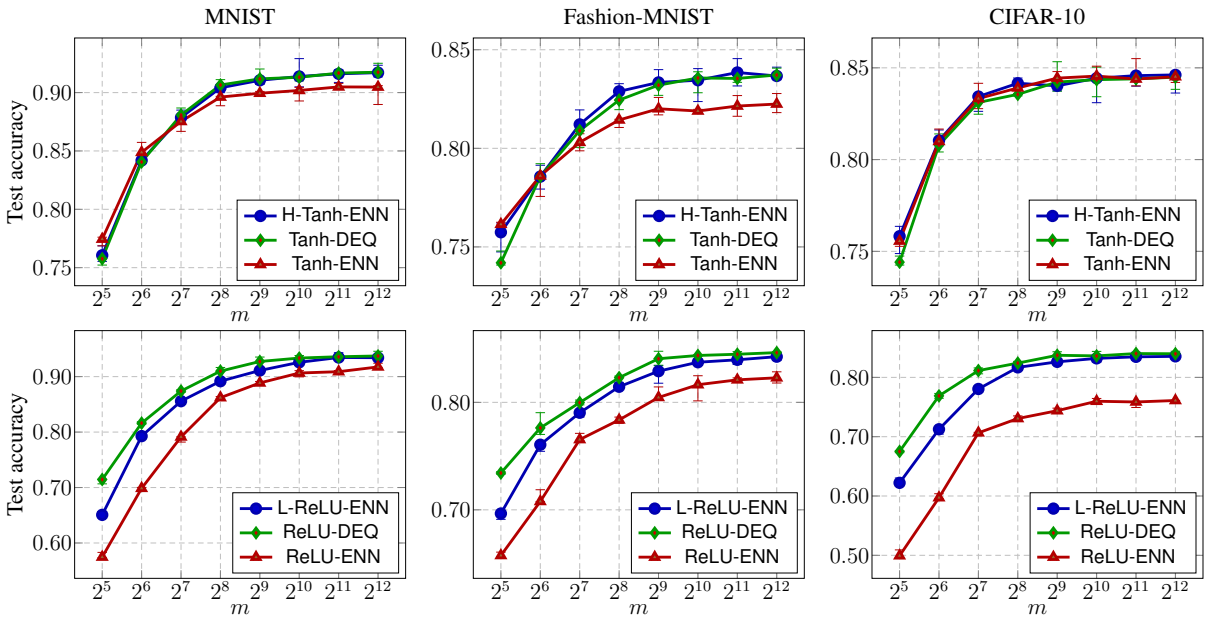

Figure 3. Classification accuracies of implicit DEQs and explicit models trained with SGD. **Top:** Evolution of classification accuracies *w.r.t.* the width m of Tanh-DEQ (**green**), the corresponding equivalent explicit H-Tanh-ENN (**blue**), and Tanh-ENN (**red**). **Bottom:** Evolution of classification accuracies *w.r.t.* the width m of ReLU-DEQ (**green**), the corresponding equivalent explicit L-ReLU-ENN (**blue**), and ReLU-ENN (**red**). For MNIST (**left**) and Fashion-MNIST datasets (**middle**), raw data are taken as the network input; for CIFAR-10 dataset (**right**) , flattened output of the 16th convolutional layer of VGG-19 are used.

rems 3.3 and 3.4, exhibits unexpected similar behavior on realistic MNIST data, see Appendix E.2 for detailed results.

Equivalent Explicit-CKs and NTKs. In Figure 2, we testify the results in Examples 3.10 and 3.11 by constructing shallow *explicit* networks with Hard Tanh-type (H-Tanh-ENN) and Leaky ReLU-type (L-ReLU-ENN) activation equivalent to implicit DEQs with Tanh (Tanh-DEQ) and ReLU (ReLU-DEQ) activation, respectively. We see that, while the two types of NN models are different in that (i) DEQs are implicitly defined while ENNs are explicitly defined, and (ii) ENNs use different activations from DEQs, their CK matrices are close in spectral norm, as long as the activation of ENNs are carefully chosen according to our Examples 3.10 and 3.11. This observation is again consistent on synthetic GMM, *and* possibly surprisingly, realistic MNIST data. We conjecture that this is due to a high-dimensional *universal* phenomenon and that our results hold more generally beyond GMM for, say, data drawn from the family of concentrated random vectors (Ledoux, 2005). We refer the interested readers to Couillet & Liao (2022, Chapter 8) for more discussions on this point.

Test performance of explicit NNs on realistic data. To explore the extent of the proposed high-dimensional equivalence between implicit DEQs and shallow explicit NN models across various realistic datasets, we conduct a comprehensive comparison of the classification accuracies using both implicit and explicit models. The results of this comparison, depicted in Figure 3, provide insights into the performance of DEQs against carefully (or not) designed explicit NNs. Following Examples 3.10 and 3.11, we construct a single-hidden-layer H-Tanh-ENN and a two-hidden-layer L-ReLU-ENN to match Tanh-DEQ and ReLU-DEQ, respectively. The undetermined parameters a, c and a_l, b_l of the activations H-Tanh-ENN and L-ReLU-ENN are determined by solving the system of equations induced from Eq. (18). For comparison, we also construct a single-hidden-layer explicit NN with Tanh activation (denoted Tanh-ENN) and a two-hidden-layer explicit NN with ReLU activation (denoted ReLU-ENN). Models are trained using SGD optimizer with learning rates of 10^{-1} for MNIST and Fashion-MNIST, and 10^{-2} for CIFAR-10. The batch size is set to 128 with a maximum training epoch of 100. To ensure a fair comparison, the hidden layer of explicit NNs share the *same* width, $m \in 2^{5-12}$, as the implicit DEQs. As m increases, the performance of L-ReLU-ENN closely matches that of ReLU-DEQ, while a noticeable performance gap exists between ReLU-ENN and ReLU-DEQ. A similar result is observed in the case of H-Tanh-ENN and Tanh-DEQ. These trends are in line with the theoretical guaranteed offered by our analysis, that focuses on CKs and NTKs and formally holds in the $m \to \infty$ limit. Experiments are also conducted using the Adam optimizer, where similar trends

can be observed. Please refer to Appendix E.4 for these results. Moreover, we observe the remarkable advantage of ENN over DEQs on the time costs of inference and training, see Appendix E.5 for detailed results. This observation substantiates our theory and underscores the practical advantages of our approach by, *e.g.,* enabling the design of memory-efficient explicit NNs that achieve the performance of implicit DEQs without the computational overhead associated with fixed-point iterations.

5. Conclusion

In this paper, we investigate the connections and differences between implicit DEQs and explicit NNs. We employ RMT to analyze the eigenspectra of the NTKs and CKs of implicit DEQs. For high-dimensional Gaussian mixture data, we establish high-dimensional approximations for the NTK and CK of implicit DEQs. Notably, we reveal that the eigenspectra of the NTK and CK of implicit DEQs are determined solely by the variance parameter and the activation function. Based on this observation, we establish the equivalence between implicit DEQs and explicit NNs in high dimensions. We propose a method for designing activation functions for explicit neural networks to match the spectral behavior of the CK (or NTK) of implicit DEQs. Results on GMM data and real-world data demonstrate that shallow explicit NNs using our theoretically designed activation functions achieve comparable performance to implicit DEQs, with significantly reduced computational overhead.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none of which we feel must be specifically highlighted here.

Acknowledgements

Z. Liao is partially supported by the National Natural Science Foundation of China (via NSFC-62206101) and the Fundamental Research Funds for the Central Universities of China (2021XXJS110). Z. Liao and Z. Ling are partially supported by the Guangdong Provincial Key Laboratory of Mathematical Foundations for Artificial Intelligence (2023B1212010001). R. C. Qiu and Z. Liao would like to acknowledge the National Natural Science Foundation of China (via NSFC-12141107), the Interdisciplinary Research Program of HUST (2023JCYJ012), the Key Research and Development Program of Guangxi (GuiKe-AB21196034). F. Zhou was supported by the National Natural Science Foundation of China Project (via NSFC-62106121) and the MOE Project of Key Research Institute of Humanities and Social Sciences (22JJD110001).

References

Agarwala, A. and Schoenholz, S. S. Deep equilibrium networks are sensitive to initialization statistics. In *International Conference on Machine Learning*, pp. 136–160. PMLR, 2022.

Alemohammad, S., Wang, Z., Balestriero, R., and Baraniuk, R. The recurrent neural tangent kernel. *arXiv preprint arXiv:2006.10246*, 2020.

Ali, H. T., Liao, Z., and Couillet, R. Random matrices in service of ml footprint: ternary random features with no performance loss. *ICLR*, 2022.

Arora, S., Du, S. S., Hu, W., Li, Z., Salakhutdinov, R. R., and Wang, R. On exact computation with an infinitely wide neural net. In *Advances in Neural Information Processing Systems*, pp. 8141–8150, 2019.

Bai, S., Kolter, J. Z., and Koltun, V. Deep equilibrium models. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

Bai, S., Koltun, V., and Kolter, J. Z. Multiscale deep equilibrium models. *Advances in Neural Information Processing Systems*, 2020.

Bai, S., Koltun, V., and Kolter, J. Z. Neural deep equilibrium solvers. In *International Conference on Learning Representations*, 2021.

Bartlett, P. L., Montanari, A., and Rakhlin, A. Deep learning: A statistical viewpoint. *Acta Numerica*, 30:87–201, May 2021. ISSN 0962-4929, 1474-0508. doi: 10.1017/S0962492921000027.

Benigni, L. and Péché, S. Eigenvalue distribution of nonlinear models of random matrices. *arXiv preprint arXiv:1904.03090*, 2019.

Blum, A., Hopcroft, J., and Kannan, R. *Foundations of Data Science*. Cambridge University Press, 2020. ISBN 978-1-108-48506-7. doi: 10.1017/9781108755528.

Cho, Y. and Saul, L. Kernel methods for deep learning. *Advances in neural information processing systems*, 22, 2009.

Couillet, R. and Benaych-Georges, F. Kernel spectral clustering of large dimensional data. *Electronic Journal of Statistics*, 10(1):1393–1454, 2016. ISSN 1935-7524. doi: 10.1214/16-ejs1144.

Couillet, R. and Liao, Z. *Random Matrix Methods for Machine Learning*. Cambridge University Press, 2022. ISBN 9781009186742.

Dobriban, E. and Wager, S. High-dimensional asymptotics of prediction: Ridge regression and classification. *The Annals of Statistics*, 46(1):247–279, 2018. ISSN 0090-5364. doi: 10.1214/17-aos1549.

Du, S. S., Hou, K., Salakhutdinov, R. R., Poczos, B., Wang, R., and Xu, K. Graph neural tangent kernel: Fusing graph neural networks with graph kernels. *Advances in neural information processing systems*, 32, 2019.

El Ghaoui, L., Gu, F., Travacca, B., Askari, A., and Tsai, A. Implicit deep learning. *SIAM Journal on Mathematics of Data Science*, 3(3):930–958, 2021.

Elkhalil, K., Kammoun, A., Couillet, R., Al-Naffouri, T. Y., and Alouini, M.-S. A large dimensional study of regularized discriminant analysis. *IEEE Transactions on Signal Processing*, 68:2464–2479, 2020. ISSN 1053-587X. doi: 10.1109/tsp.2020.2984160.

Fan, Z. and Wang, Z. Spectra of the conjugate kernel and neural tangent kernel for linear-width neural networks. *Advances in neural information processing systems*, 33: 7710–7721, 2020.

Feng, Z. and Kolter, J. Z. On the neural tangent kernel of equilibrium models. *arxiv*, 2020.

Fung, S. W., Heaton, H., Li, Q., McKenzie, D., Osher, S., and Yin, W. Jfb: Jacobian-free backpropagation for implicit networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 6648–6656, 2022.

Gao, T. and Gao, H. Gradient descent optimizes infinite-depth relu implicit networks with linear widths. *arxiv*, 2022.

Gao, T., Liu, H., Liu, J., Rajan, H., and Gao, H. A global convergence theory for deep relu implicit networks via over-parameterization. *ICLR*, 2022.

Gao, T., Huo, X., Liu, H., and Gao, H. Wide neural networks as gaussian processes: Lessons from deep equilibrium models. *NeurIPS*, 2023.

Gilton, D., Ongie, G., and Willett, R. Deep equilibrium architectures for inverse problems in imaging. *arXiv preprint arXiv:2102.07944*, 2021.

Goodfellow, I., Bengio, Y., and Courville, A. Deep learning mit press (2016). In *Conference on information and communication systems (ICICS)*, pp. 151–156, 2016.

Gu, L., Du, Y., Yuan, Z., Xie, D., Pu, S., Qiu, R., and Liao, Z. " lossless" compression of deep neural networks: A high-dimensional neural tangent kernel approach. *Advances in Neural Information Processing Systems*, 35:3774–3787, 2022.

Huang, Z., Bai, S., and Kolter, J. Z. $(\text{Implicit})^2$: Implicit layers for implicit representations. In *Advances in Neural Information Processing Systems*, volume 34, pp. 9639–9650, 2021.

Jacot, A., Gabriel, F., and Hongler, C. Neural tangent kernel: convergence and generalization in neural networks. In *Advances in Neural Information Processing Systems*, pp. 8580–8589, 2018.

Krizhevsky, A. Learning multiple layers of features from tiny images. pp. 32–33, 2009. URL https://www.cs.toronto.edu/~kriz/learning-features-2009-TR.pdf.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Ledoux, M. *The Concentration of Measure Phenomenon*. Mathematical Surveys and Monographs. American Mathematical Soc., 2005. ISBN 9780821837924. doi: 10.1090/surv/089.

Liao, Z. and Couillet, R. The dynamics of learning: A random matrix approach. In *International Conference on Machine Learning*, pp. 3072–3081. PMLR, 2018a.

Liao, Z. and Couillet, R. On the spectrum of random features maps of high dimensional data. In *International Conference on Machine Learning*, pp. 3063–3071. PMLR, 2018b.

Liao, Z. and Couillet, R. A large dimensional analysis of least squares support vector machines. *IEEE Transactions on Signal Processing*, 67(4):1065–1074, 2019. ISSN 1053-587X. doi: 10.1109/tsp.2018.2889954.

Ling, Z., Xie, X., Wang, Q., Zhang, Z., and Lin, Z. Global convergence of over-parameterized deep equilibrium models. In *International Conference on Artificial Intelligence and Statistics*, pp. 767–787. PMLR, 2023.

Louart, C. and Couillet, R. Concentration of Measure and Large Random Matrices with an application to Sample Covariance Matrices. *arXiv*, 2018. URL https://arxiv.org/pdf/1805.08295.

Louart, C., Liao, Z., and Couillet, R. A random matrix approach to neural networks. *Annals of Applied Probability*, 28(2):1190–1248, 2018. ISSN 1050-5164. doi: 10.1214/17-aap1328.

Mei, S. and Montanari, A. The generalization error of random features regression: Precise asymptotics and the double descent curve. *Communications on Pure and Applied Mathematics*, 75(4):667–766, 2022.

Micaelli, P., Vahdat, A., Yin, H., Kautz, J., and Molchanov, P. Recurrence without recurrence: Stable video landmark detection with deep equilibrium models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 22814–22825, 2023.

Novak, R., Xiao, L., Hron, J., Lee, J., Alemi, A. A., Sohl-Dickstein, J., and Schoenholz, S. S. Neural tangents: Fast and easy infinite neural networks in python. *arXiv preprint arXiv:1912.02803*, 2019.

Pastur, L. Eigenvalue distribution of large random matrices arising in deep neural networks: Orthogonal case. *Journal of Mathematical Physics*, 63(6), 2022.

Pastur, L. and Slavin, V. On random matrices arising in deep neural networks: General iid case. *Random Matrices: Theory and Applications*, 12(01):2250046, 2023.

Pennington, J. and Worah, P. Nonlinear random matrix theory for deep learning. *Advances in neural information processing systems*, 30, 2017.

Ramzi, Z., Mannel, F., Bai, S., Starck, J.-L., Ciuciu, P., and Moreau, T. Shine: Sharing the inverse estimate from the forward pass for bi-level optimization and implicit models. *arXiv preprint arXiv:2106.00553*, 2021.

Seddik, M. E. A., Louart, C., Tamaazousti, M., and Couillet, R. Random matrix theory proves that deep learning representations of gan-data behave as gaussian mixtures. In *International Conference on Machine Learning*, pp. 8573–8582. PMLR, 2020.

Truong, L. V. Global convergence rate of deep equilibrium models with general activations. *arXiv preprint arXiv:2302.05797*, 2023.

Tsuchida, R., Roosta, F., and Gallagher, M. Invariance of weight distributions in rectified mlps. In *International Conference on Machine Learning*, pp. 4995–5004. PMLR, 2018.

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

Winston, E. and Kolter, J. Z. Monotone operator equilibrium networks. In *Advances in Neural Information Processing Systems*, 2020.

Xiao, H., Rasul, K., and Vollgraf, R. Fashion-MNIST: A novel image dataset for benchmarking machine learning algorithms. *arXiv*, 2017.

Xie, X., Wang, Q., Ling, Z., Li, X., Liu, G., and Lin, Z. Optimization induced equilibrium networks: An explicit optimization perspective for understanding equilibrium models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

Supplementary Material

Deep Equilibrium Models are Almost Equivalent to Not-so-deep Explicit Models for High-dimensional Gaussian Mixtures

A. Preliminaries

We are interested in the associated conjugate kernel and the neural tangent kernel (Implicit-CK and Implicit-NTK, for short) of implicit neural networks defined in Eq. (3). According to the results in (Feng & Kolter, 2020, Theorem 2), the corresponding Implicit-CK takes the following form

$$\boldsymbol{G}^* = \lim_{l\to\infty} \boldsymbol{G}^{(l)}, \tag{21}$$

where the (i,j)-th entry of $\boldsymbol{G}^{(l)}$ is defined recursively as $\boldsymbol{G}_{ij}^{(0)} = 0$ and[2]

$$\boldsymbol{G}_{ij}^{(l)} = \mathbb{E}[\phi(\mathrm{u}^{(l)})\phi(\mathrm{v}^{(l)})], \text{ with } (\mathrm{u}^{(l)}, \mathrm{v}^{(l)}) \sim \mathcal{N}\left(0, \begin{bmatrix} \boldsymbol{\Lambda}_{ii}^{(l)} & \boldsymbol{\Lambda}_{ij}^{(l)} \\ \boldsymbol{\Lambda}_{ji}^{(l)} & \boldsymbol{\Lambda}_{jj}^{(l)} \end{bmatrix}\right), \quad l \geq 1, \tag{22}$$

where $\boldsymbol{\Lambda}_{ij}^{(1)} = \boldsymbol{x}_i^\top \boldsymbol{x}_j$, and $\boldsymbol{\Lambda}_{ij}^{(l)} = \sigma_a^2 \boldsymbol{G}_{ij}^{(l-1)} + \sigma_b^2 \boldsymbol{x}_i^\top \boldsymbol{x}_j$, *i.e.*, $\boldsymbol{\Lambda}^{(l)} = \sigma_a^2 \boldsymbol{G}^{(l-1)} + \sigma_b^2 \boldsymbol{X}^\top \boldsymbol{X}$. The Implicit-NTK is defined as $\boldsymbol{K}^* = \lim_{l\to\infty} \boldsymbol{K}^{(l)}$ whose the (i,j)-th entry is defined as

$$\boldsymbol{K}_{ij}^{(l)} = \sum_{h=1}^{l+1} \left(\boldsymbol{G}_{ij}^{(h-1)} \prod_{h'=h}^{l+1} \dot{\boldsymbol{G}}_{ij}^{(h')} \right), \tag{23}$$

where $\dot{\boldsymbol{G}}_{ij}^{(l)} = \sigma_a^2 \mathbb{E}_{(\mathrm{u}^{(l)}, \mathrm{v}^{(l)})}[\phi'(\mathrm{u}^{(l)})\phi'(\mathrm{v}^{(l)})]$. The limit of Implicit-NTK is

$$\boldsymbol{K}_{ij}^* \equiv \frac{\boldsymbol{G}_{ij}^*}{1 - \dot{\boldsymbol{G}}_{ij}^*}. \tag{24}$$

We consider n data vectors $\boldsymbol{x}_1, \cdots, \boldsymbol{x}_n \in \mathbb{R}^p$ independently drawn from one of the K-class Gaussian mixture $\mathcal{C}_1, \cdots, \mathcal{C}_K$ and denote $\boldsymbol{X} = [\boldsymbol{x}_1, \cdots, \boldsymbol{x}_n] \in \mathbb{R}^{p\times n}$, with class $\mathcal{C}_a$ having cardinality n_a, *i.e.*, for $\boldsymbol{x}_i \in \mathcal{C}_a$, we have

$$\boldsymbol{x}_i \sim \mathcal{N}(\boldsymbol{\mu}_a/\sqrt{p}, \boldsymbol{C}_a/p)$$

Assumption A.1. We assume that, as $n \to \infty$, we have, for $a \in \{1, \cdots, K\}$ that,

- $p/n \to c \in (0, \infty)$ and $n_a/n \to c_a \in (0, 1)$; and
- $\|\boldsymbol{\mu}_a\| = \mathcal{O}(1)$; and
- for $\boldsymbol{C}^\circ \equiv \sum_{a=1}^K \frac{n_a}{n} \boldsymbol{C}_a$ and $\boldsymbol{C}_a^\circ \equiv \boldsymbol{C}_a - \boldsymbol{C}^\circ$, we have $\|\boldsymbol{C}_a\| = \mathcal{O}(1)$, $\operatorname{tr} \boldsymbol{C}_a^\circ = \mathcal{O}\left(p^{\frac{1}{2}}\right)$ and $\operatorname{tr}(\boldsymbol{C}_a \boldsymbol{C}_b) = \mathcal{O}(p)$ for $a, b \in \{1, \cdots, K\}$; and
- $\tau_0 = \sqrt{\operatorname{tr} \boldsymbol{C}^\circ / p}$ converges in $(0, \infty)$.

Some quantities. We first introduce the following notations. For $\boldsymbol{x}_i, \boldsymbol{x}_j \in \mathbb{R}^p$ with $i \neq j$, let

$$\boldsymbol{x}_i = \boldsymbol{\mu}_i/\sqrt{p} + \boldsymbol{\varepsilon}_i/\sqrt{p}, \quad \boldsymbol{x}_j = \boldsymbol{\mu}_j/\sqrt{p} + \boldsymbol{\varepsilon}_j/\sqrt{p},$$

[2]Note that the expectation is conditioned on the input data, and is taken with respect to the random weights.

so that $\boldsymbol{\varepsilon}_i \sim \mathcal{N}(0, \boldsymbol{C}_i)$, $\boldsymbol{\varepsilon}_j \sim \mathcal{N}(0, \boldsymbol{C}_j)$, and

$$\begin{aligned}
\boldsymbol{x}_i^\top \boldsymbol{x}_j &= \underbrace{\frac{1}{p}\boldsymbol{\varepsilon}_i^\top \boldsymbol{\varepsilon}_j}_{\mathcal{O}(p^{-1/2})} + \underbrace{\frac{1}{p}\boldsymbol{\mu}_i^\top \boldsymbol{\mu}_j + \frac{1}{p}(\boldsymbol{\mu}_i^\top \boldsymbol{\varepsilon}_j + \boldsymbol{\mu}_j^\top \boldsymbol{\varepsilon}_i)}_{\mathcal{O}(p^{-1})}, \\
\psi_i &= \frac{1}{p}\|\boldsymbol{\varepsilon}_i\|^2 - \frac{1}{p}\operatorname{tr}\boldsymbol{C}_i = \mathcal{O}\Big(p^{-1/2}\Big), \quad s_i \equiv \|\mu_i\|^2/p + 2\boldsymbol{\mu}_i^\top \boldsymbol{\varepsilon}_i/p = \mathcal{O}\big(p^{-1}\big), \\
t_i &= \frac{1}{p}\operatorname{tr}\boldsymbol{C}_i^\circ = \mathcal{O}\Big(p^{-1/2}\Big), \quad \tau_0 = \sqrt{\frac{1}{p}\boldsymbol{C}^\circ} = \mathcal{O}(1), \\
\chi_i &= \underbrace{t_i + \psi_i}_{\mathcal{O}(p^{-1/2})} + \underbrace{s_i}_{\mathcal{O}(p^{-1})} = \|\boldsymbol{x}_i\|^2 - \tau_0^2.
\end{aligned}$$

It can be checked that

$$\begin{aligned}
\|\boldsymbol{x}_i\|^2 &= \frac{1}{p}(\boldsymbol{\mu}_i + \boldsymbol{\varepsilon}_i)^\top(\boldsymbol{\mu}_i + \boldsymbol{\varepsilon}_i) = \frac{1}{p}\|\boldsymbol{\mu}_i\|^2 + \frac{2}{p}\boldsymbol{\mu}_i^\top \boldsymbol{\varepsilon}_i + \frac{1}{p}\boldsymbol{\varepsilon}_i^\top \boldsymbol{\varepsilon}_i \\
&= \underbrace{\frac{1}{p}\|\boldsymbol{\mu}_i\|^2 + \frac{2}{p}\boldsymbol{\mu}_i^\top \boldsymbol{\varepsilon}_i}_{\equiv s_i = \mathcal{O}(p^{-1})} + \underbrace{\frac{1}{p}\operatorname{tr}\boldsymbol{C}^\circ}_{\equiv \tau_0^2 = \mathcal{O}(1)} + \underbrace{\frac{1}{p}\operatorname{tr}\boldsymbol{C}_i^\circ}_{\equiv t_i = \mathcal{O}\left(p^{-1/2}\right)} + \underbrace{\psi_i}_{\mathcal{O}\left(p^{-1/2}\right)}
\end{aligned}$$

By Taylor-expanding $\sqrt{\|\boldsymbol{x}_i\|^2}$ around τ_0^2, we have

$$\|\boldsymbol{x}_i\| = \tau_0 + \frac{1}{2\tau_0}(\|\boldsymbol{\mu}_i\|^2/p + 2\boldsymbol{\mu}_i^\top \boldsymbol{\varepsilon}_i/p + t_i + \psi_i) - \frac{1}{8\tau_0^3}(t_i + \psi_i)^2 + \mathcal{O}\Big(p^{-3/2}\Big). \tag{25}$$

Additionally, we denote S_{ij} terms of the form

$$S_{ij} \equiv S_{ij}(\gamma_1, \gamma_2) = \frac{1}{p}\boldsymbol{\varepsilon}_i^\top \boldsymbol{\varepsilon}_j(\gamma_1(t_i + \psi_i) + \gamma_2(t_j + \psi_j)),$$

for random or deterministic scalars $\gamma_1, \gamma_2 = \mathcal{O}(1)$ (with high probability when being random). Note that $S_{ij} = \mathcal{O}\big(p^{-1}\big)$ and more importantly, it leads to, in matrix form, a matrix of spectral norm order $\mathcal{O}\big(p^{-1}\big)$ (Couillet & Benaych-Georges, 2016).

Moreover, we recursively define τ_l as

$$\tau_l = \sqrt{\sigma_a^2 \mathbb{E}[\phi^2(\tau_{l-1}\xi)] + \sigma_b^2 \tau_0^2}, \tag{26}$$

for $l = 1, 2, \cdots$. The following lemma shows that the unique fixed point of Eq. (26) exits under Assumption 3.1.

Lemma A.2. *Let Assumption 3.1 hold. As $l \to \infty$, τ_l converges to a fixed point τ^* such that*

$$\lim_{l\to\infty} \tau_l \equiv \tau_* = \sqrt{\sigma_a^2 \mathbb{E}[\phi^2(\tau_*\xi)] + \sigma_b^2 \tau_0^2}.$$

Proof. Let $t = \tau_{l-1}^2$. By taking the derivative with respect to t on the RHS of Eq. (10), we have

$$\begin{aligned}
&\frac{\partial}{\partial t}\left(\sigma_a^2 \mathbb{E}\left[f(\tau_{l-1}\xi)\right] + \sigma_b^2 \tau_0^2\right) \\
=&\sigma_a^2 \frac{\partial}{\partial t}\mathbb{E}\left[f(\sqrt{t}\cdot\xi)\right] \\
=&\sigma_a^2 \frac{\partial}{\partial t}\left(\int \frac{1}{\sqrt{2\pi}} f(\sqrt{t}\cdot x) e^{-\frac{x^2}{2}} dx\right) \\
=&\sigma_a^2 \frac{1}{\sqrt{2\pi}} \int f'(\sqrt{t}\cdot x)\frac{x}{2\sqrt{t}} e^{-\frac{x^2}{2}} dx \\
=&\frac{\sigma_a^2}{2}\cdot \mathbb{E}[f''(\tau_{l-1}\xi)], \quad \text{by the Gaussian integration by parts formula,}
\end{aligned}$$

which implies that the RHS of Eq. (10) is a *contractive mapping* if

$$\sigma_a^2 < \frac{2}{L_2}.$$

As a result, under Assumption 3.1, the unique fixed point τ_* exists. □

The quantity τ_* will play a crucial role in our proof.

B. Proof of Theorem 3.3

With loss of generality, we assume that $\boldsymbol{G}^{(0)} = \mathbb{E}[\phi^2(\tau_*\xi)] \cdot \boldsymbol{I}_n$, *i.e.*, $\boldsymbol{G}_{ii}^{(0)} = \mathbb{E}[\phi^2(\tau_*\xi)]$ and $\boldsymbol{G}_{ij}^{(0)} = 0$ for $i \neq j$.

We prove Theorem 3.3 by performing induction on the hypothesis that $\|\boldsymbol{G}^{(l-1)} - \widetilde{\boldsymbol{G}}^{(l-1)}\| \to 0$ holds at layer $l-1$ with

$$\widetilde{\boldsymbol{G}}^{(l-1)} \equiv \alpha_{l-1,1}\boldsymbol{X}^\top\boldsymbol{X} + \boldsymbol{V}\boldsymbol{C}^{(l-1)}\boldsymbol{V}^\top + (\mathbb{E}[\phi^2(\tau_*\xi)] - \tau_0^2\alpha_{l-1,1})\boldsymbol{I}_n, \tag{27}$$

for $\boldsymbol{C}^{(l-1)} = \begin{bmatrix} \alpha_{l-1,2}\boldsymbol{t}\boldsymbol{t}^\top + \alpha_{l-1,3}\boldsymbol{T} & \alpha_{l-1,2}\boldsymbol{t} \\ \alpha_{l-1,2}\boldsymbol{t}^\top & \alpha_{l-1,2} \end{bmatrix}$, and work on $\boldsymbol{G}^{(l)}$ at layer l.

Note that $\boldsymbol{\Lambda}_{ij}^{(l)} = \sigma_a^2\boldsymbol{G}_{ij}^{(l-1)} + \sigma_b^2\boldsymbol{x}_i^\top\boldsymbol{x}_j$, *i.e.*, $\boldsymbol{\Lambda}^{(l)} = \sigma_a^2\boldsymbol{G}^{(l-1)} + \sigma_b^2\boldsymbol{X}^\top\boldsymbol{X}$. Thus, it holds that $\|\boldsymbol{\Lambda}^{(l)} - \widetilde{\boldsymbol{\Lambda}}^{(l)}\| \to 0$ for

$$\widetilde{\boldsymbol{\Lambda}}^{(l)} \equiv \lambda_{l,1}\boldsymbol{X}^\top\boldsymbol{X} + \boldsymbol{V}\boldsymbol{C}_{\boldsymbol{\Lambda}}^{(l)}\boldsymbol{V}^\top + (\tau_*^2 - \tau_0^2\lambda_{l,1})\boldsymbol{I}_n,$$

for $\boldsymbol{C}_{\boldsymbol{\Lambda}}^{(l)} = \begin{bmatrix} \lambda_{l,2}\boldsymbol{t}\boldsymbol{t}^\top + \lambda_{l,3}\boldsymbol{T} & \lambda_{l,2}\boldsymbol{t} \\ \lambda_{l,2}\boldsymbol{t}^\top & \lambda_{l,2} \end{bmatrix}$ where $\lambda_{l,1} = \sigma_a^2\alpha_{l-1,1} + \sigma_b^2$, $\lambda_{l,2} = \sigma_a^2\alpha_{l-1,2}$, and $\lambda_{l,2} = \sigma_a^2\alpha_{l-1,3}$.

The following lemma plays an important role in our proof.

Lemma B.1 ((Gu et al., 2022)). *Let Assumptions 2.3-A.1 hold. Given a matrix $\boldsymbol{\Lambda} \in \mathbb{R}^{n\times n}$ such that*

$$\begin{aligned}
\boldsymbol{\Lambda}_{ii} &= \tau^2 + \lambda_4\chi_i + \lambda_5(t_i + \psi_i)^2 + \mathcal{O}\Big(p^{-3/2}\Big) \\
\boldsymbol{\Lambda}_{ij} &= \lambda_1\boldsymbol{x}_i^\top\boldsymbol{x}_j + \lambda_2(t_i + \psi_i)(t_j + \psi_j) + \lambda_3\left(\frac{1}{p}\boldsymbol{\varepsilon}_i^\top\boldsymbol{\varepsilon}_j\right)^2 + S_{ij} + \mathcal{O}\Big(p^{-3/2}\Big),
\end{aligned}$$

where λ_k, $k = 1, \cdots, 5$, and τ are arbitrary constants, it holds that

$$\begin{aligned}
&\mathbb{E}\left[\phi\left(\sqrt{\boldsymbol{\Lambda}_{ii}}\cdot\xi_i\right) \times \phi\left(\frac{\boldsymbol{\Lambda}_{ij}}{\sqrt{\boldsymbol{\Lambda}_{ii}}}\cdot\xi_i + \sqrt{\boldsymbol{\Lambda}_{jj} - \frac{(\boldsymbol{\Lambda}_{ij})^2}{\boldsymbol{\Lambda}_{ii}}}\cdot\xi_j\right)\right] \\
=&\mathbb{E}[\phi'(\tau\xi)]^2 + \mathbb{E}[\phi''(\tau\xi)]^2 \cdot \lambda_1\boldsymbol{x}_i^\top\boldsymbol{x}_j \\
&+ \mathbb{E}[\phi'(\tau\xi)]\mathbb{E}[\phi'''(\tau\xi)] \cdot \frac{\lambda_4}{2}(\chi_i + \chi_j) + \mathcal{O}\big(p^{-1}\big),
\end{aligned}$$

for independent ξ_i, ξ_j and $\xi \sim \mathcal{N}(0,1)$. Moreover, if the activation function $\phi(\cdot)$ is "centered", such that $\mathbb{E}[\phi(\tau\xi)] = 0$, it holds that

$$\begin{aligned}
&\mathbb{E}\left[\phi\left(\sqrt{\boldsymbol{\Lambda}_{ii}}\cdot\xi_i\right) \times \phi\left(\frac{\boldsymbol{\Lambda}_{ij}}{\sqrt{\boldsymbol{\Lambda}_{ii}}}\cdot\xi_i + \sqrt{\boldsymbol{\Lambda}_{jj} - \frac{(\boldsymbol{\Lambda}_{ij})^2}{\boldsymbol{\Lambda}_{ii}}}\cdot\xi_j\right)\right] \\
=&\mathbb{E}[\phi'(\tau\xi)]^2\boldsymbol{\Lambda}_{ij} + \frac{\lambda_1^2}{2}\mathbb{E}[\phi''(\tau\xi)]^2\left(\frac{1}{p}\boldsymbol{\varepsilon}_i^\top\boldsymbol{\varepsilon}_j\right)^2 + \frac{\lambda_4^2}{4}\mathbb{E}[\phi''(\tau\xi)]^2(t_i + \psi_i)(t_j + \psi_j) \\
&+ S_{ij} + \mathcal{O}\Big(p^{-3/2}\Big).
\end{aligned}$$

On the diagonal. By induction hypothesis on the layer l, we have

$$\mathbf{\Lambda}_{ii}^{(l)} = \tau_*^2 + \lambda_{l,4}\chi_i + \lambda_{l,5}(t_i+\psi_i)^2 + \mathcal{O}\Big(p^{-3/2}\Big). \tag{28}$$

For $l=1$, $\mathbf{\Lambda}^{(1)} = \sigma_a^2 \boldsymbol{G}_{ii}^{(0)} + \sigma_b^2\|\boldsymbol{x}_i\|^2 = \sigma_a^2\mathbb{E}[\phi^2(\tau^*\xi)] + \sigma_b^2\|\boldsymbol{x}_i\|^2 = \tau_*^2$, and the hypothesis holds with $\lambda_{1,4}=\lambda_{1,5}=0$.

For $l>1$, by Eq. (4), we have

$$\mathbf{\Lambda}_{ii}^{(l+1)} = \sigma_a^2\mathbb{E}\left[\phi\left(\sqrt{\mathbf{\Lambda}_{ii}^{(l)}}\cdot\xi\right)^2\right] + (1-\sigma_a^2)\|\boldsymbol{x}_i\|^2,$$

for $\xi\sim\mathcal{N}(0,1)$.

By Taylor-expanding, one gets

$$\sqrt{\mathbf{\Lambda}_{ii}^{(l)}} = \tau_* + \frac{1}{2\tau_*}\lambda_{l,4}\chi_i + \frac{4\tau_*^2\lambda_{l,5}-\lambda_{l,4}^2}{8\tau_*^3}(t_i+\psi_i)^2 + \mathcal{O}\Big(p^{-3/2}\Big).$$

For simplicity, we denote the shortcut $f(\cdot)=\phi^2(\cdot)$. By Talor-expanding and Eq. (25), one gets

$$\begin{aligned}
\mathbf{\Lambda}_{ii}^{(l+1)} =& \sigma_a^2\mathbb{E}\left[\phi\left(\sqrt{\mathbf{\Lambda}_{ii}^{(l)}}\cdot\xi\right)^2\right] + \sigma_b^2\|\boldsymbol{x}_i\|^2 = \sigma_a^2\mathbb{E}\left[f\left(\sqrt{\mathbf{\Lambda}_{ii}^{(l)}}\cdot\xi\right)\right] + \sigma_b^2\|\boldsymbol{x}_i\|^2\\
=& \sigma_a^2\mathbb{E}\left[f(\tau_*\xi) + f'(\tau_*\xi)\xi\left(\frac{1}{2\tau_*}\lambda_{l,4}\chi_i + \frac{4\tau_*^2\lambda_{l,5}-\lambda_{l,4}^2}{8\tau_*^3}(t_i+\psi_i)^2\right)\right]\\
&+ \sigma_a^2\mathbb{E}\left[\frac{1}{2}f''(\tau_*\xi)\xi^2\right]\frac{\lambda_{l,4}^2}{4\tau_*}(t_i+\psi_i)^2 + \sigma_b^2(\tau_0^2+\chi_i) + \mathcal{O}\Big(p^{-3/2}\Big)\\
=& \sigma_a^2\mathbb{E}\left[f(\tau_*\xi)\right] + \sigma_b^2\tau_0^2 + \left(\sigma_a^2\frac{\lambda_{l,4}}{2}\mathbb{E}[f''(\tau_*\xi)] + \sigma_b^2\right)\chi_i\\
&+ \sigma_a^2\frac{4\lambda_{l,5}\mathbb{E}[f''(\tau_*\xi)] + \lambda_{l,4}^2\mathbb{E}[f''''(\tau_*\xi)]}{8}(t_i+\psi_i)^2 + \mathcal{O}\Big(p^{-3/2}\Big),
\end{aligned} \tag{29}$$

where we use the facts that

$$\mathbb{E}[f'(\tau_*\xi)] = \tau_*\mathbb{E}[f''(\tau_*\xi)], \quad \mathbb{E}[f''''(\tau_*\xi)(\xi^2-1)] = \tau_*^2\mathbb{E}[f''''(\tau_*\xi)],$$

for $\xi\sim\mathcal{N}(0,1)$, as a result of the Gaussian integration by parts formula.

Thus , we prove that $\mathbf{\Lambda}_{ii}^{(l+1)} = \tau_*^2 + \lambda_{l+1,4}\chi_i + \lambda_{l+1,5}(t_i+\psi_i)^2 + \mathcal{O}\big(p^{-3/2}\big)$, where

$$\begin{aligned}
\lambda_{l+1,4} &= \frac{\sigma_a^2}{2}\mathbb{E}[f''(\tau_*\xi)]\lambda_{l,4} + \sigma_b^2,\\
\lambda_{l+1,5} &= \frac{\sigma_a^2}{2}\mathbb{E}[f''(\tau_*\xi)]\lambda_{l,5} + \frac{\sigma_a^2}{8}\mathbb{E}[f''''(\tau_*\xi)]\lambda_{l,4}^2.
\end{aligned} \tag{30}$$

By Lemma A.2, under Assumption 3.1, it holds that $\frac{\sigma_a^2}{2}\mathbb{E}[f''(\tau_{l-1}\xi)] < 1$, which implies that, as $l\to\infty$, the iterations in Eq. (30) converge. Let $l\to\infty$, we obtain that

$$\begin{aligned}
\lambda_{*,4} &\equiv \lim_{l\to\infty}\lambda_{l,4} = \left(1-\frac{\sigma_a^2}{2}\mathbb{E}[f''(\tau_*\xi)]\right)^{-1}\sigma_b^2\\
\lambda_{*,5} &\equiv \lim_{l\to\infty}\lambda_{l,5} = \frac{\sigma_a^2}{8}\left(1-\frac{\sigma_a^2}{2}\mathbb{E}[f''(\tau_*\xi)]\right)^{-1}\mathbb{E}[f''''(\tau^*\xi)]\lambda_{*,4}^2.
\end{aligned} \tag{31}$$

Off the diagonal. For $i \neq j$, by induction hypothesis on the layer $l-1$, we have

$$\boldsymbol{\Lambda}_{ij}^{(l)} = \lambda_{l,1} A_{ij} + \lambda_{l,2}(t_i + \psi_i)(t_j + \psi_j) + \lambda_{l,3}\left(\frac{1}{p}\boldsymbol{\varepsilon}_i^\top \boldsymbol{\varepsilon}_j\right)^2 + S_{ij} + \mathcal{O}\Big(p^{-3/2}\Big).$$

Using the Gram-Schmidt orthogonalization for standard Gaussian random variable, we write

$$\begin{aligned}
\boldsymbol{\Lambda}_{ii}^{(l+1)} &= \sigma_a^2 \mathbb{E}\left[\phi^2\left(\sqrt{\boldsymbol{\Lambda}_{ii}^{(l)}} \cdot \xi_i\right)\right] + \sigma_b^2 \|\boldsymbol{x}_i\|^2, \\
\boldsymbol{\Lambda}_{ij}^{(l+1)} &= \sigma_a^2 \mathbb{E}\left[\phi\left(\sqrt{\boldsymbol{\Lambda}_{ii}^{(l)}} \cdot \xi_i\right) \times \phi\left(\frac{\boldsymbol{\Lambda}_{ij}^{(l)}}{\sqrt{\boldsymbol{\Lambda}_{ii}^{(l)}}} \cdot \xi_i + \sqrt{\boldsymbol{\Lambda}_{jj}^{(l)} - \frac{\left(\boldsymbol{\Lambda}_{ij}^{(l)}\right)^2}{\boldsymbol{\Lambda}_{ii}^{(l)}}} \cdot \xi_j\right)\right] + \sigma_b^2 \boldsymbol{x}_i^\top \boldsymbol{x}_j.
\end{aligned}$$

Using Lemma B.1, we have

$$\begin{aligned}
\boldsymbol{\Lambda}_{ij}^{(l+1)} =& \sigma_a^2 \mathbb{E}\left[\phi\left(\sqrt{\boldsymbol{\Lambda}_{ii}^{(l)}} \cdot \xi_i\right) \times \phi\left(\frac{\boldsymbol{\Lambda}_{ij}^{(l)}}{\sqrt{\boldsymbol{\Lambda}_{ii}^{(l)}}} \cdot \xi_i + \sqrt{\boldsymbol{\Lambda}_{jj}^{(l)} - \frac{\left(\boldsymbol{\Lambda}_{ij}^{(l)}\right)^2}{\boldsymbol{\Lambda}_{ii}^{(l)}}} \cdot \xi_j\right)\right] + \sigma_b^2 \boldsymbol{x}_i^\top \boldsymbol{x}_j \\
=& \sigma_a^2 \mathbb{E}[\phi'(\tau_* \xi)]^2 \boldsymbol{\Lambda}_{ij}^{(l)} \\
&+ \sigma_a^2 \left(\frac{\lambda_{l,1}}{2}\mathbb{E}[\phi''(\tau_*\xi)]^2 \left(\frac{1}{p}\boldsymbol{\varepsilon}_i^\top \boldsymbol{\varepsilon}_j\right)^2 + \frac{\lambda_{l,4}^2}{4}\mathbb{E}[\phi''(\tau_*\xi)]^2 (t_i + \psi_i)(t_j + \psi_j)\right) \\
&+ S_{ij} + \sigma_b^2 \boldsymbol{x}_i^\top \boldsymbol{x}_j + \mathcal{O}\Big(p^{-3/2}\Big) \\
=& \sigma_a^2 \mathbb{E}[\phi'(\tau_* \xi)]^2 \left(\lambda_{l,1} \boldsymbol{x}_i^\top \boldsymbol{x}_j + \lambda_{l,2}(t_i + \psi_i)(t_j + \psi_j) + \lambda_{l,3}\left(\frac{1}{p}\boldsymbol{\varepsilon}_i^\top \boldsymbol{\varepsilon}_j\right)^2\right) \\
&+ \sigma_a^2 \left(\frac{\lambda_{l,1}^2}{2}\mathbb{E}[\phi''(\tau_*\xi)]^2 \left(\frac{1}{p}\boldsymbol{\varepsilon}_i^\top \boldsymbol{\varepsilon}_j\right)^2 + \frac{\lambda_{l,4}^2}{4}\mathbb{E}[\phi''(\tau_*\xi)]^2 (t_i + \psi_i)(t_j + \psi_j)\right) \\
&+ S_{ij} + \sigma_b^2 \boldsymbol{x}_i^\top \boldsymbol{x}_j + \mathcal{O}\Big(p^{-3/2}\Big).
\end{aligned}$$

Consequently, it holds that

$$\boldsymbol{\Lambda}_{ij}^{(l+1)} = \lambda_{l+1,1} \boldsymbol{x}_i^\top \boldsymbol{x}_j + \lambda_{l+1,2}(t_i + \psi_i)(t_j + \psi_j) + \lambda_{l+1,3}\left(\frac{1}{p}\boldsymbol{\varepsilon}_i^\top \boldsymbol{\varepsilon}_j\right)^2 + S_{ij} + \mathcal{O}\Big(p^{-3/2}\Big), \tag{32}$$

where

$$\begin{aligned}
\lambda_{l+1,1} &= \sigma_a^2 \mathbb{E}[\phi'(\tau_* \xi)]^2 \lambda_{l,1} + \sigma_b^2, \\
\lambda_{l+1,2} &= \sigma_a^2 \mathbb{E}[\phi'(\tau_* \xi)]^2 \lambda_{l,2} + \frac{\sigma_a^2}{4}\mathbb{E}[\phi''(\tau_*\xi)]^2 \lambda_{l,4}^2, \\
\lambda_{l+1,3} &= \sigma_a^2 \mathbb{E}[\phi'(\tau_* \xi)]^2 \lambda_{l,3} + \frac{\sigma_a^2}{2}\mathbb{E}[\phi''(\tau_*\xi)]^2 \lambda_{l,1}^2.
\end{aligned} \tag{33}$$

Assembling in matrix form. By using the fact that $\|\boldsymbol{M}\|_2 \leq n \max_{i,j} |\boldsymbol{M}_{ij}|$ for $\boldsymbol{M} \in \mathbb{R}^{n \times n}$ and $\{S_{ij}\}_{ij} = \mathcal{O}_{\|\cdot\|}(p^{-1/2})$ (Couillet & Benaych-Georges, 2016), and by noting the fact that $\boldsymbol{\Lambda}^{(l+1)} = \sigma_a^2 \boldsymbol{G}^{(l)} + \sigma_b^2 \boldsymbol{X}^\top \boldsymbol{X}$, *i.e*, $\lambda_{l,1} = \sigma_a^2 \alpha_{l-1,1} + \sigma_b^2$, $\lambda_{l,2} = \sigma_a^2 \alpha_{l-1,2}$, and $\lambda_{l,2} = \sigma_a^2 \alpha_{l-1,3}$, it holds that

$$\boldsymbol{G}^{(l)} = \alpha_{l,1} \boldsymbol{X}^\top \boldsymbol{X} + \boldsymbol{V} \boldsymbol{C}^{(l)} \boldsymbol{V}^\top + (\mathbb{E}[\phi^2(\tau_* \xi)] - \tau_0^2 \alpha_{l,1}) \boldsymbol{I}_n + \mathcal{O}_{\|\cdot\|}(p^{-\frac{1}{2}}) \tag{34}$$

where

$$\boldsymbol{V} = [\boldsymbol{J}/\sqrt{p}, \boldsymbol{\psi}], \quad \boldsymbol{C}^{(l)} = \begin{bmatrix} \alpha_{l,2} \boldsymbol{t}\boldsymbol{t}^\top + \alpha_{l,3} \boldsymbol{T} & \alpha_{l,2} \boldsymbol{t} \\ \alpha_{l,2} \boldsymbol{t}^\top & \alpha_{l,2} \end{bmatrix}, \tag{35}$$

with non-negative scalars $\alpha_{l,1}, \alpha_{l,2}, \alpha_{l,3}, \alpha_{l,4} \geq 0$ defined recursively as

$$\begin{aligned}
\alpha_{l,1} &= \sigma_a^2 \mathbb{E}[\phi'(\tau_* \xi)]^2 \alpha_{l-1,1} + \sigma_b^2 \mathbb{E}[\phi'(\tau_* \xi)]^2, \\
\alpha_{l,2} &= \sigma_a^2 \mathbb{E}[\phi'(\tau_* \xi)]^2 \alpha_{l-1,2} + \frac{1}{4} \mathbb{E}[\phi''(\tau_* \xi)]^2 \alpha_{l-1,4}^2, \\
\alpha_{l,3} &= \sigma_a^2 \mathbb{E}[\phi'(\tau_* \xi)]^2 \alpha_{l-1,3} + \frac{1}{2} \mathbb{E}[\phi''(\tau_* \xi)]^2 (\sigma_a^2 \alpha_{l-1,1} + \sigma_b^2)^2, \\
\alpha_{l,4} &= \frac{\sigma_a^2}{2} \mathbb{E}[(\phi^2(\tau_* \xi))''] \alpha_{l-1,4} + \sigma_b^2,
\end{aligned} \tag{36}$$

Note that it holds that $\sigma_a^2 \mathbb{E}[\phi'(\tau_* \xi)]^2 < 1$ and $\frac{1}{2} \sigma_a^2 \mathbb{E}[(\phi^2(\tau_* \xi))''] < 1$ under Assumptions 2.4 and 3.1. This means that, as $l \to \infty$, the iterations in Eq. (36) converge. Let $l \to \infty$, we obtain that

$$\boldsymbol{G}^* = \alpha_{*,1} \boldsymbol{X}^\top \boldsymbol{X} + \boldsymbol{V} \boldsymbol{C} \boldsymbol{V}^\top + (\mathbb{E}[\phi^2(\tau_* \xi)] - \tau_0^2 \alpha_{*,1}) \boldsymbol{I}_n + \mathcal{O}_{\|\cdot\|}(p^{-\frac{1}{2}}) \tag{37}$$

where

$$\boldsymbol{V} = [\boldsymbol{J}/\sqrt{p}, \boldsymbol{\psi}], \quad \boldsymbol{C} = \begin{bmatrix} \alpha_{*,2} \boldsymbol{t}\boldsymbol{t}^\top + \alpha_{*,3} \boldsymbol{T} & \alpha_{*,2} \boldsymbol{t} \\ \alpha_{*,2} \boldsymbol{t}^\top & \alpha_{*,2} \end{bmatrix}, \tag{38}$$

with non-negative scalars $\alpha_{*,1}, \alpha_{*,2}, \alpha_{*,3}, \alpha_{*,4} \geq 0$ defined as

$$\alpha_{*,1} = \frac{\sigma_b^2 \mathbb{E}[\phi'(\tau_* \xi)]^2}{1 - \sigma_a^2 \mathbb{E}[\phi'(\tau_* \xi)]^2}, \qquad \alpha_{*,2} = \frac{\alpha_{*,4}^2 \mathbb{E}[\phi''(\tau_* \xi)]^2}{4(1 - \sigma_a^2 \mathbb{E}[\phi'(\tau_* \xi)]^2)}, \tag{39}$$

$$\alpha_{*,3} = \frac{(\sigma_a^2 \alpha_{*,1} + \sigma_b^2)^2 \mathbb{E}[\phi''(\tau_* \xi)]^2}{2(1 - \sigma_a^2 \mathbb{E}[\phi'(\tau_* \xi)]^2)}, \qquad \alpha_{*,4} = \frac{\sigma_b^2}{1 - \frac{\sigma_a^2}{2} \mathbb{E}[(\phi^2(\tau_* \xi))'']}. \tag{40}$$

C. Proof of Theorem 3.4

C.1. The CK $\dot{\boldsymbol{G}}$

Before proving Theorem 3.4, one needs to deal with the CK $\dot{\boldsymbol{G}}$.

Recall that

$$\dot{\boldsymbol{G}}_{ij}^{(l)} = \sigma_a^2 \mathbb{E}_{(\mathrm{u}^{(l)}, \mathrm{v}^{(l)})}[\phi'(\mathrm{u}^{(l)}) \phi'(\mathrm{v}^{(l)})], \text{ with } (\mathrm{u}^{(l)}, \mathrm{v}^{(l)}) \sim \mathcal{N}\left(0, \begin{bmatrix} \boldsymbol{\Lambda}_{ii}^{(l)} & \boldsymbol{\Lambda}_{ij}^{(l)} \\ \boldsymbol{\Lambda}_{ji}^{(l)} & \boldsymbol{\Lambda}_{jj}^{(l)} \end{bmatrix}\right).$$

Using the Gram-Schmidt orthogoalizaiton procedure, we have

$$\begin{aligned}
\dot{\boldsymbol{G}}_{ii}^{(l)} &= \sigma_a^2 \mathbb{E}\left[\phi'\left(\sqrt{\boldsymbol{\Lambda}_{ii}^{(l)}} \cdot \xi_i\right)^2\right] \\
\dot{\boldsymbol{G}}_{ij}^{(l)} &= \sigma_a^2 \mathbb{E}\left[\phi'\left(\sqrt{\boldsymbol{\Lambda}_{ii}^{(l)}} \cdot \xi_i\right) \times \phi'\left(\frac{\boldsymbol{\Lambda}_{ij}^{(l)}}{\sqrt{\boldsymbol{\Lambda}_{ii}^{(l)}}} \cdot \xi_i + \sqrt{\boldsymbol{\Lambda}_{jj}^{(l)} - \frac{\left(\boldsymbol{\Lambda}_{ij}^{(l)}\right)^2}{\boldsymbol{\Lambda}_{ii}^{(l)}}} \cdot \xi_j\right)\right]
\end{aligned} \tag{41}$$

On the diagonal. First, recall that

$$\sqrt{\boldsymbol{\Lambda}_{ii}^{(l)}} = \tau_* + \frac{1}{2\tau_*} \lambda_{l,4} \chi_i + \frac{4\tau_*^2 \lambda_{l,5} - \lambda_{l,4}^2}{8\tau_*^3} (t_i + \psi_i)^2 + \mathcal{O}\Big(p^{-3/2}\Big).$$

Denote the shortcut $f(t) = (\phi'(t))^2$ for simplicity, using Taylor-expand again, we have

$$\begin{aligned}
\dot{\boldsymbol{G}}_{ii}^{(l)} =& \sigma_a^2 \mathbb{E}\left[\phi'\left(\sqrt{\boldsymbol{\Lambda}_{ii}^{(l)}}\cdot\xi\right)^2\right] = \sigma_a^2\mathbb{E}\left[f\left(\sqrt{\boldsymbol{\Lambda}_{ii}^{(l)}}\cdot\xi\right)\right] \\
=& \sigma_a^2\mathbb{E}\left[f(\tau_*\xi) + f'(\tau_*\xi)\xi\left(\frac{1}{2\tau_*}\lambda_{l,4}\chi_i + \frac{4\tau_*^2\lambda_{l,5} - \lambda_{l,4}^2}{8\tau_{l-1}^3}(t_i+\psi_i)^2\right)\right] \\
&+ \sigma_a^2\mathbb{E}\left[\frac{1}{2}f''(\tau_*\xi)\xi^2\right]\frac{\lambda_{l,4}^2}{4\tau_*^2}(t_i+\psi_i)^2 + \mathcal{O}\Big(p^{-3/2}\Big) \\
=& \sigma_a^2\mathbb{E}\left[f(\tau_*\xi)\right] + \left(\sigma_a^2\frac{\lambda_{l,4}}{2}\mathbb{E}[f''(\tau_*\xi)]\right)\chi_i \\
&+ \sigma_a^2\frac{4\lambda_{l,5}\mathbb{E}[f''(\tau_*\xi)] + \lambda_{l,4}^2\mathbb{E}[f''''(\tau_*\xi)]}{8} + \mathcal{O}\Big(p^{-3/2}\Big),
\end{aligned}$$

Thus, we conclude that

$$\dot{\boldsymbol{G}}_{ii}^{(l)} = \sigma_a^2\mathbb{E}\left[\phi'\left(\sqrt{\boldsymbol{\Lambda}_{ii}^{(l)}}\xi\right)^2\right] = \sigma_a^2\dot{\tau}_*^2 + \mathcal{O}\Big(p^{-1/2}\Big), \tag{42}$$

with the sequence $\dot{\tau}_*$ defined as follows

$$\dot{\tau}_* = \sqrt{\mathbb{E}\left[\phi'(\tau_*\xi)^2\right]}.$$

Off the diagonal. For $i \neq j$, by Lemma B.1, it holds that

$$\begin{aligned}
&\mathbb{E}\left[\phi'\left(\sqrt{\boldsymbol{\Lambda}_{ii}^{(l)}}\cdot\xi_i\right)\times\phi'\left(\frac{\boldsymbol{\Lambda}_{ij}^{(l)}}{\sqrt{\boldsymbol{\Lambda}_{ii}^{(l-1)}}}\cdot\xi_i + \sqrt{\boldsymbol{\Lambda}_{jj}^{(l)} - \frac{\left(\boldsymbol{\Lambda}_{ij}^{(l)}\right)^2}{\boldsymbol{\Lambda}_{ii}^{(l)}}}\cdot\xi_j\right)\right] \\
=& \mathbb{E}[\phi'(\tau_*\xi)]^2 + \mathbb{E}[\phi''(\tau_*\xi)]^2\cdot\lambda_{l,1}\boldsymbol{x}_i^\top\boldsymbol{x}_j \\
&+ E[\phi'(\tau_*\xi)]E[\phi'''(\tau_*\xi)]\cdot\frac{\lambda_{l,4}}{2}(\chi_i+\chi_j) + \mathcal{O}\big(p^{-1}\big)
\end{aligned}$$

Thus, we have

$$\begin{aligned}
\dot{\boldsymbol{G}}_{ij}^{(l)} &= \sigma_a^2\mathbb{E}\left[\phi'\left(\sqrt{\boldsymbol{\Lambda}_{ii}^{(l)}}\cdot\xi_i\right)\times\phi'\left(\frac{\boldsymbol{\Lambda}_{ij}^{(l)}}{\sqrt{\boldsymbol{\Lambda}_{ii}^{(l)}}}\cdot\xi_i + \sqrt{\boldsymbol{\Lambda}_{jj}^{(l)} - \frac{\left(\boldsymbol{\Lambda}_{ij}^{(l)}\right)^2}{\boldsymbol{\Lambda}_{ii}^{(l)}}}\cdot\xi_j\right)\right] \\
&= \sigma_a^2\mathbb{E}[\phi'(\tau_*\xi)]^2 + \sigma_a^2\mathbb{E}[\phi''(\tau_*\xi)]^2\cdot\lambda_{l,1}\boldsymbol{x}_i^\top\boldsymbol{x}_j \\
&+ \sigma_a^2\mathbb{E}[\phi'(\tau_*\xi)]\mathbb{E}[\phi'''(\tau_*\xi)]\cdot\frac{\lambda_{l,4}}{2}(\chi_i+\chi_j) + \mathcal{O}\big(p^{-1}\big) \\
&= \dot{\alpha}_{l,0} + \dot{\alpha}_{l,1}\boldsymbol{x}_i^\top\boldsymbol{x}_j + \dot{\alpha}_{l,2}(\chi_i+\chi_j) + \mathcal{O}\big(p^{-1}\big),
\end{aligned} \tag{43}$$

with

$$\begin{aligned}
\dot{\alpha}_{l,0} &= \sigma_a^2\mathbb{E}[\phi'(\tau_*\xi)]^2, \\
\dot{\alpha}_{l,1} &= \sigma_a^2\mathbb{E}[\phi''(\tau_*\xi)]^2\lambda_{l,1} = \sigma_a^2\mathbb{E}[\phi''(\tau_*\xi)]^2(\sigma_a^2\alpha_{l-1,1} + \sigma_b^2), \\
\dot{\alpha}_{l,2} &= \frac{\sigma_a^2}{2}\mathbb{E}[\phi'(\tau_*\xi)]\mathbb{E}[\phi'''(\tau_*\xi)]\lambda_{l,4} = \frac{\sigma_a^2}{2}\mathbb{E}[\phi'(\tau_*\xi)]\mathbb{E}[\phi'''(\tau_*\xi)]\alpha_{l,4}.
\end{aligned}$$

As $l \to \infty$, it holds that

$$\begin{aligned}
\dot{\alpha}_{*,0} &\equiv \lim_{l\to\infty} \dot{\alpha}_{l,0} = \sigma_a^2 \mathbb{E}[\phi'(\tau_* \xi)]^2, \\
\dot{\alpha}_{*,1} &\equiv \lim_{l\to\infty} \dot{\alpha}_{l,1} = \sigma_a^2 \mathbb{E}[\phi''(\tau_* \xi)]^2 (\sigma_a^2 \alpha_{*,1} + \sigma_b^2), \\
\dot{\alpha}_{*,2} &\equiv \lim_{l\to\infty} \dot{\alpha}_{l,2} = \frac{\sigma_a^2}{2} \mathbb{E}[\phi'(\tau_* \xi)] \mathbb{E}[\phi'''(\tau_* \xi)] \alpha_{*,4}.
\end{aligned}$$

C.2. Implicit NTKs

With the above results at hand, we now proceed to the proof of Theorem 3.4. We assume the induction hypothesis holds for $l-1$, that

$$\begin{aligned}
\boldsymbol{K}_{ii}^{(l-1)} &= \kappa_{l-1}^2 + \mathcal{O}\Big(p^{-1/2}\Big), \\
\boldsymbol{K}_{ij}^{(l-1)} &= \beta_{l-1,1} \boldsymbol{x}_i^\top \boldsymbol{x}_j + \beta_{l-1,2}(t_i + \psi_i)(t_i + \psi_i) + \beta_{l-1,3} \left(\frac{1}{p} \boldsymbol{\varepsilon}_i^\top \boldsymbol{\varepsilon}_j\right)^2 + S_{ij} + \mathcal{O}\Big(p^{-3/2}\Big).
\end{aligned}$$

On the diagonal. First, using the results of Eq. (29) and Eq. (42), we have

$$\boldsymbol{K}_{ii}^{(l)} = \boldsymbol{G}_{ii}^{(l)} + \boldsymbol{K}_{ii}^{(l-1)} \cdot \dot{\boldsymbol{G}}_{ii}^{(l)} = \mathbb{E}[\phi^2(\tau_* \xi)] + \sigma_a^2 \kappa_{l-1}^2 \mathbb{E}[\phi'(\tau_* \xi)^2] + \mathcal{O}\Big(p^{-1/2}\Big).$$

Thus, it holds that

$$\boldsymbol{K}_{ii}^{(l)} = \kappa_l^2 + \mathcal{O}\Big(p^{-1/2}\Big),$$

with

$$\kappa_l^2 = \mathbb{E}\left[\phi^2(\tau_* \xi)\right] + \sigma_a^2 \mathbb{E}[\phi'(\tau_* \xi)^2] \cdot \kappa_{l-1}^2.$$

Under Assumption 2.4, it holds that $\sigma_a^2 \mathbb{E}\left[\phi'(\tau_* \xi)^2\right] < 1$. Thus, for $l \to \infty$, one gets that

$$\kappa_*^2 \equiv \lim_{l\to\infty} \kappa_l^2 = \left(1 - \sigma_a^2 \mathbb{E}\left[\phi'(\tau_* \xi)^2\right]\right)^{-1} \mathbb{E}\left[\phi^2(\tau_* \xi)\right]. \tag{44}$$

Off the diagonal. For $i \neq j$, using the results of Eq. (32) and Eq. (32), we get

$$\begin{aligned}
\boldsymbol{K}_{ij}^{(l)} =& \boldsymbol{G}_{ij}^{(l)} + \boldsymbol{K}_{ij}^{(l-1)} \dot{\boldsymbol{G}}_{ij}^{(l)} \\
=& \alpha_{l,1} \boldsymbol{x}_i^\top \boldsymbol{x}_j + \alpha_{l,2}(t_i + \psi_i)(t_j + \psi_j) + \alpha_{l,3} \left(\frac{1}{p} \boldsymbol{\varepsilon}_i^\top \boldsymbol{\varepsilon}_j\right)^2 \\
&+ \left(\beta_{l-1,1} \boldsymbol{x}_i^\top \boldsymbol{x}_j + \beta_{l-1,2}(t_i + \psi_i)(t_j + \psi_j) + \beta_{l-1,3} \left(\frac{1}{p} \boldsymbol{\varepsilon}_i^\top \boldsymbol{\varepsilon}_j\right)^2\right) \\
&\times \left(\dot{\alpha}_{l,0} + \dot{\alpha}_{l,1} \boldsymbol{x}_i^\top \boldsymbol{x}_j + \dot{\alpha}_{l,2} (\chi_i + \chi_j) + \mathcal{O}\big(p^{-1}\big)\right) + \mathcal{O}\Big(p^{-3/2}\Big) \\
=& (\alpha_{l,1} + \beta_{l-1,1} \cdot \dot{\alpha}_{l,0}) \boldsymbol{x}_i^\top \boldsymbol{x}_j + (\alpha_{l,2} + \beta_{l-1,2} \cdot \dot{\alpha}_{l,0})(t_i + \psi_i)(t_j + \psi_j) \\
&+ (\alpha_{l,3} + \beta_{l-1,3} \cdot \dot{\alpha}_{l,0} + \beta_{l-1,1} \cdot \dot{\alpha}_{l,1}) \left(\frac{1}{p} \boldsymbol{\varepsilon}_i^\top \boldsymbol{\varepsilon}_j\right)^2 + S_{ij} + \mathcal{O}\Big(p^{-3/2}\Big),
\end{aligned}$$

so that it holds that

$$\boldsymbol{K}_{ij}^{(l)} = \beta_{l,1} \boldsymbol{x}_i^\top \boldsymbol{x}_j + \beta_{l,2}(t_i + \psi_i)(t_i + \psi_i) + \beta_{l,3} \left(\frac{1}{p} \boldsymbol{\varepsilon}_i^\top \boldsymbol{\varepsilon}_j\right)^2 + S_{ij} + \mathcal{O}\Big(p^{-3/2}\Big).$$

with

$$\begin{aligned}\beta_{l,1} &= \alpha_{l,1} + \beta_{l-1,1}\dot{\alpha}_{l,0},\\ \beta_{l,2} &= \alpha_{l,2} + \beta_{l-1,2}\dot{\alpha}_{l,0},\\ \beta_{l,3} &= \alpha_{l,3} + \beta_{l-1,3}\dot{\alpha}_{l,0} + \beta_{l-1,1}\dot{\alpha}_{l,1}.\end{aligned}$$

As $l \to \infty$, it holds that $\lim_{l\to\infty} \tau_l = \tau_*$, $\lim_{l\to\infty} \alpha_{l,k} = \alpha_{*,k}$, and $\lim_{l\to\infty} \dot{\alpha}_{l,k} = \dot{\alpha}_{*,k}$, for $k = 1, 2, 3$. Therefore, for $l \to \infty$, one gets that

$$\boldsymbol{K}^*_{ij} = \beta_{*,1}\boldsymbol{x}_i^\top \boldsymbol{x}_j + \beta_{*,2}(t_i + \psi_i)(t_j + \psi_j) + \beta_{*,3}\left(\frac{1}{p}\boldsymbol{\varepsilon}_i^\top \boldsymbol{\varepsilon}_j\right)^2 + S_{ij} + \mathcal{O}\Big(p^{-3/2}\Big),$$

where

$$\begin{aligned}\beta_{*,1} &\equiv \lim_{l\to\infty} \beta_{l,1} = (1 - \dot{\alpha}_{*,0})^{-1}\alpha_{*,1},\\ \beta_{*,2} &\equiv \lim_{l\to\infty} \beta_{l,2} = (1 - \dot{\alpha}_{*,0})^{-1}\alpha_{*,2},\\ \beta_{*,3} &\equiv \lim_{l\to\infty} \beta_{l,3} = (1 - \dot{\alpha}_{*,0})^{-1}(\alpha_{*,3} + \beta_{*,1}\dot{\alpha}_{*,1}).\end{aligned}$$

Assembling in matrix form: Using the fact that $\|\boldsymbol{M}\|_2 \le n \max_{i,j} |\boldsymbol{M}_{ij}|$ for $\boldsymbol{M} \in \mathbb{R}^{n\times n}$ and $\{S_{ij}\}_{ij} = \mathcal{O}_{\|\cdot\|}(p^{-1/2})$ (Couillet & Benaych-Georges, 2016), it holds that

$$\boldsymbol{K}^* = \beta_{*,1}\boldsymbol{X}^\top\boldsymbol{X} + \boldsymbol{V}\boldsymbol{D}_*\boldsymbol{V}^\top + (\kappa_*^2 - \tau_0^2\beta_{*,1})\boldsymbol{I}_n + \mathcal{O}_{\|\cdot\|}(p^{-\frac{1}{2}}), \tag{45}$$

with

$$\boldsymbol{V} = [\boldsymbol{J}/\sqrt{p}, \boldsymbol{\psi}], \quad \boldsymbol{D}_* = \begin{bmatrix} \beta_{*,2}\boldsymbol{t}\boldsymbol{t}^\top + \beta_{*,3}\boldsymbol{T} & \beta_{*,2}\boldsymbol{t} \\ \beta_{*,2}\boldsymbol{t}^\top & \beta_{*,2}\end{bmatrix}, \tag{46}$$

and

$$\boldsymbol{T} = \left\{\frac{1}{p}\operatorname{tr}\boldsymbol{C}_a\boldsymbol{C}_b\right\}_{a,=1}^K, \quad \boldsymbol{t} = \left\{\frac{1}{\sqrt{p}}\operatorname{tr}\boldsymbol{C}_a^\circ\right\}. \tag{47}$$

D. Proof and discussions of Examples 3.10 and 3.11

Let $\widetilde{\tau}_0 = \tau_0$ as defined in Eq. (9) and $\widetilde{\tau}_1, \cdots \widetilde{\tau}_L \ge 0$ be a sequence of non-negative scalars satisfying $\widetilde{\tau}_l = \sqrt{\mathbb{E}[\sigma_l^2(\widetilde{\tau}_{l-1}\xi)]}$, for $\xi \sim \mathcal{N}(0, 1)$ and $l \in \{1, \cdots, L\}$. It follows from Theorem 3.8 that $\|\boldsymbol{\Sigma}^{(l)} - \overline{\boldsymbol{\Sigma}}^{(l)}\| = \mathcal{O}\big(n^{-1/2}\big)$, where

$$\overline{\boldsymbol{\Sigma}}^{(l)} = \widetilde{\alpha}_{l,1}\boldsymbol{X}^\top\boldsymbol{X} + \boldsymbol{V}\widetilde{\boldsymbol{C}}_l\boldsymbol{V}^\top + (\widetilde{\tau}_l^2 - \tau_0^2\widetilde{\alpha}_1)\boldsymbol{I}_n, \tag{48}$$

with $\boldsymbol{V} \in \mathbb{R}^{n\times(K+1)}$ as defined in Theorem 3.3,

$$\widetilde{\boldsymbol{C}}_l = \begin{bmatrix} \widetilde{\alpha}_{l,2}\boldsymbol{t}\boldsymbol{t}^\top + \widetilde{\alpha}_{l,3}\boldsymbol{T} & \widetilde{\alpha}_{l,2}\boldsymbol{t} \\ \widetilde{\alpha}_{l,2}\boldsymbol{t}^\top & \widetilde{\alpha}_{l,2}\end{bmatrix} \in \mathbb{R}^{(K+1)\times(K+1)}.$$

and non-negative scalars $\widetilde{\alpha}_{l,1}, \widetilde{\alpha}_{l,2}, \widetilde{\alpha}_{l,3}$ defined recursively as $\widetilde{\alpha}_{0,1} = \widetilde{\alpha}_{0,4} = 1$, $\widetilde{\alpha}_{0,2} = \widetilde{\alpha}_{0,3} = 0$, and

$$\begin{aligned}\widetilde{\alpha}_{l,1} &= \mathbb{E}[\sigma_l'(\widetilde{\tau}_{l-1}\xi)]^2\widetilde{\alpha}_{l-1,1},\\ \widetilde{\alpha}_{l,2} &= \mathbb{E}[\sigma_l'(\widetilde{\tau}_{l-1}\xi)]^2\widetilde{\alpha}_{l-1,2} + \frac{1}{4}\mathbb{E}[\sigma_l''(\widetilde{\tau}_{l-1}\xi)]^2\widetilde{\alpha}_{l-1,4}^2,\\ \widetilde{\alpha}_{l,3} &= \mathbb{E}[\sigma_l'(\widetilde{\tau}_{l-1}\xi)]^2\widetilde{\alpha}_{l-1,3} + \frac{1}{2}\mathbb{E}[\sigma_l''(\widetilde{\tau}_{l-1}\xi)]^2\widetilde{\alpha}_{l-1,1}^2.\end{aligned}$$

For a given Tanh-DEQ, we first compute the four key parameters $\alpha_{*,1}, \alpha_{*,2}, \alpha_{*,3}$ and γ_* of the implicit DEQ according to Eq. (12) of Theorem 3.3. For the single-hidden-layer Hard-tanh explicit NN, it can be easily checked that the corresponding CK matrix is determined by

$$\widetilde{\alpha}_{1,1} = \mathbb{E}[\sigma_l'(\tau_0\xi)]^2, \quad \widetilde{\alpha}_{1,2} = \widetilde{\alpha}_{1,3} = 0.$$

For H-Tanh-ENN with the activation $\sigma_{\text{H-Tanh}}(x) \equiv ax \cdot 1_{-c\le x\le c} + ac \cdot (1_{x\ge c} - 1_{x\le -c})$ with $a > 0$ and $c \ge 0$, $\widetilde{\alpha}_{1,1}, \widetilde{\alpha}_{1,2}, \widetilde{\alpha}_{1,3}, \widetilde{\tau}_1$ can be represented as functions of the activation parameters by following results

$$\mathbb{E}[(\sigma_{\text{H-Tanh}}(\tau_0\xi))^2] = \frac{1}{2}\left(c^2 + a^2 + (c^2 - a^2)e^{-2\tau_0^2} - c^2 e^{-\tau_0^2}\right), \quad \mathbb{E}[(\sigma_{\text{H-Tanh}}(\tau_0\xi))'] = ae^{-\tau_0^2/2},$$
$$\mathbb{E}[(\sigma_{\text{H-Tanh}}(\tau_0\xi))''] = -ce^{-\tau_0^2/2}, \quad \mathbb{E}[((\sigma_{\text{H-Tanh}}(\tau_0\xi))^2)''] = 2e^{-2\tau_0^2}\left(a^2 + c^2(e^{\tau_0^2} - 1)\right).$$

To match Tanh-DEQ, we determine the activations of H-Tanh-ENN by solving the system

$$\widetilde{\alpha}_{1,1} = \alpha_{*,1}, \quad \widetilde{\alpha}_{1,2} = \alpha_{*,2}, \quad \widetilde{\alpha}_{1,3} = \alpha_{*,3}, \quad \widetilde{\tau}_1 = \gamma_*.$$

For a given ReLU-DEQ, we first compute the four key parameters $\alpha_{*,1}, \alpha_{*,2}, \alpha_{*,3}$ and γ_* of the implicit DEQ according to Eq. (12) of Theorem 3.3. For a two-hidden-layer explicit NN, it can be easily checked that the corresponding CK matrix is determined by

$$\begin{aligned}
\widetilde{\alpha}_{2,1} &= \mathbb{E}[\sigma_2'(\widetilde{\tau}_1\xi)]^2\widetilde{\alpha}_{1,1} \\
\widetilde{\alpha}_{2,2} &= \mathbb{E}[\sigma_2'(\widetilde{\tau}_1\xi)]^2\widetilde{\alpha}_{1,2} + \frac{1}{4}\mathbb{E}[\sigma_2''(\widetilde{\tau}_1\xi)]^2\widetilde{\alpha}_{1,4}^2, \\
\widetilde{\alpha}_{2,3} &= \mathbb{E}[\sigma_2'(\widetilde{\tau}_1\xi)]^2\widetilde{\alpha}_{1,3} + \frac{1}{2}\mathbb{E}[\sigma_2''(\widetilde{\tau}_1\xi)]^2\widetilde{\alpha}_{1,1}^2,
\end{aligned}$$

and

$$\widetilde{\alpha}_{1,1} = \mathbb{E}[\sigma_1'(\widetilde{\tau}_0\xi)]^2, \quad \widetilde{\alpha}_{1,2} = \frac{1}{4}\mathbb{E}[\sigma_1''(\widetilde{\tau}_0\xi)]^2, \quad \widetilde{\alpha}_{1,3} = \frac{1}{2}\mathbb{E}[\sigma_1''(\widetilde{\tau}_0\xi)]^2, \quad \widetilde{\alpha}_{1,4} = \frac{1}{2}\mathbb{E}[(\sigma_1^2(\tau_0\xi))''].$$

For L-ReLU-ENN with the activation $\sigma^{(l)}_{\text{L-ReLU}}(x) \equiv \max(a_l x, b_l x) - \frac{a_l - b_l}{\sqrt{2\pi}}\tau_{l-1}$, $\widetilde{\alpha}_{2,1}, \widetilde{\alpha}_{2,2}, \widetilde{\alpha}_{2,3}, \widetilde{\tau}_2$ can be represented as functions of the activation parameters by following results

$$\mathbb{E}[(\sigma^{(l)}_{\text{L-ReLU}}(\tau_{l-1}\xi))^2] = \frac{(a_l^2 + b_l^2)(\pi - 1) + 2a_l b_l}{2\pi}\tau_{l-1}^2, \quad \mathbb{E}[(\sigma^{(l)}_{\text{L-ReLU}}(\tau_{l-1}\xi))'] = \frac{a_l + b_l}{2},$$
$$\mathbb{E}[(\sigma^{(l)}_{\text{L-ReLU}}(\tau_{l-1}\xi))'']^2 = \frac{a_l - b_l}{\sqrt{2}\pi\tau_{l-1}}, \quad \mathbb{E}[((\sigma^{(l)}_{\text{L-ReLU}}(\tau_{l-1}\xi))^2)''] = \frac{(a_l^2 + b_l^2)(\pi - 1) + 2a_l b_l}{\pi}.$$

To match ReLU-DEQ, we determine the activations of L-ReLU-ENN by solving the system

$$\widetilde{\alpha}_{2,1} = \alpha_{*,1}, \quad \widetilde{\alpha}_{2,2} = \alpha_{*,2}, \quad \widetilde{\alpha}_{2,3} = \alpha_{*,3}, \quad \widetilde{\tau}_2 = \gamma_*.$$

On the numerical determination of $\sigma^{(l)}_{\text{L-ReLU}}$ and $\sigma_{\text{H-Tanh}}$. The system of nonlinear equations mentioned above does not admit explicit solutions but can be efficiently solved using numerical methods, such as the least squares method (implemented through the optimize.minimize function in the SciPy library).

E. Additional Experimental Results

E.1. High dimensional equivalents of Implicit-NTKs

Figure 4 compares the difference between Implicit-NTKs $\boldsymbol{K}^*$ and their high-dimensional approximation $\overline{\boldsymbol{K}}$ given in Theorem 3.4, on two-class Gaussian mixture data, on DEQs as Definition 2.1 with four commonly used activations: ReLU, Tanh, Swish, and Leaky-ReLU (L-ReLU). We observe from Figure 4 that, for different activations, as n, p increase, the relative errors consistently and significantly decrease, as in line with our Theorem 3.4.

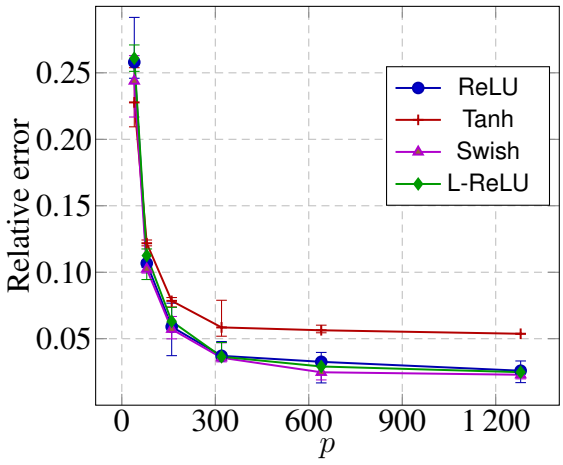

Figure 4. Evolution of relative spectral norm error $\|\boldsymbol{K}^* - \overline{\boldsymbol{K}}\|/\|\boldsymbol{K}^*\|$ *w.r.t.* sample size n, for DEQs in Definition 2.1 with different activations and $\sigma_a^2 = 0.2$, on two-class GMM, $p/n = 0.8$, $\boldsymbol{\mu}_a = [\mathbf{0}_{8(a-1)}; 8; \mathbf{0}_{p-8a+7}]$, and $\boldsymbol{C}_a = (1 + 8(a-1)/\sqrt{p})\boldsymbol{I}_p$, $a \in \{1, 2\}$. Implicit-NTK matrices $\boldsymbol{K}^*$ defined in Eq. (7) are taken with expectation estimated from DEQs with random $\boldsymbol{A}$ and $\boldsymbol{B}$ of width $m = 2^{12}$. The asymptotic equivalent matrices $\overline{\boldsymbol{K}}$ are obtained by Theorem 3.4.

E.2. Visualization results of the spectrum of Implicit-CKs and those Implicit-NTKs

In Figure 5 and Figure 6, we provide visualization results of the spectral densities of Implicit-CKs and Implicit-NTKs, respectively, along with their corresponding high-dimensional approximations. We observe from Figure 5 and Figure 6 that the proposed theoretical results, despite derived here for GMM data *and* in the limit of $n, p \to \infty$, provide extremely accurate prediction of the Implicit-CK eigenspectral behavior (i) for not-so-large n, p *and* (ii) possibly surprisingly, also on realistic MNIST data.

E.3. Visualization results of the spectrum of Implicit-CKs and equivalent Explicit-CKs

In Figure 7, we compare the spectral densities of Implicit-CK matrices of given DEQs with those of Explicit-CK matrices of the corresponding "equivalent" shallow explicit NNs by following Examples 3.10 and 3.11. We observe that the CK matrices of ENNs are close to those of the corresponding DEQs. This observation is consistent on GMM data and realistic MNIST data. We conjecture that this is due to a high-dimensional *universal* phenomenon and that our results (on both CK and NTK matrices) hold more generally beyond the GMM setting, say, for data drawn from the family of concentrated random vectors (Ledoux, 2005; Louart & Couillet, 2018).

E.4. Adam Results

We present the classification results of implicit DEQs and explicit models trained with the Adam optimizer in Figure 8. Each model is trained with the Adam optimizer, using initial learning rates of 10^{-2} for MNIST and Fashion-MNIST, and 10^{-3} for CIFAR-10. The remaining experimental settings mirror those of the SGD experiment depicted in Figure 3. The results obtained with the Adam optimizer are similar to those achieved with SGD.

E.5. Time cost comparison

We compare the time costs of the inference and training of DEQs and the corresponding "equivalent" ENNs. As shown in Tables 1 and 2, the inference time cost of a DEQ is about $2 - 3\times$ that of an ENN with the same dimension. This is due to the fact that it takes numerous iterations for a DEQ to reach the iteration error threshold. Additionally, we observe that ENNs have a remarkable advantage over DEQs in terms of training speed.

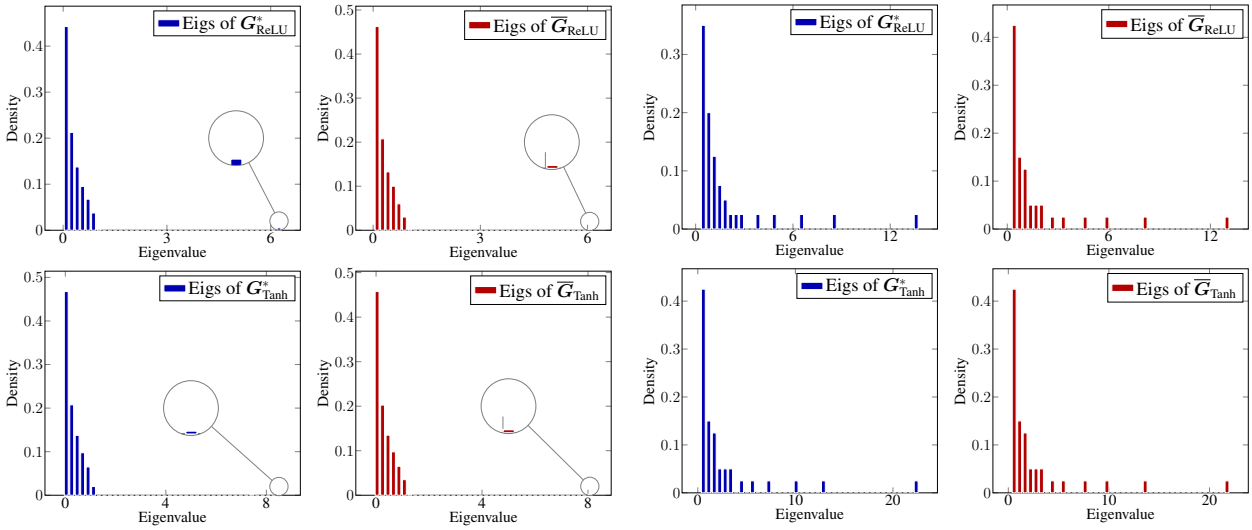

Figure 5. Eigenvalue density of Implicit-CK matrices (**blue**) $\boldsymbol{G}^*_{\text{ReLU}}$ of ReLU-DEQ (**top**) and $\boldsymbol{G}^*_{\text{Tanh}}$ of Tanh-DEQ (**bottom**) and the corresponding high dimensional approximation $\overline{\boldsymbol{G}}_{\text{ReLU}}$ and $\overline{\boldsymbol{G}}_{\text{Tanh}}$ (**red**) , on two-class GMM data (**left**) with $p = 1\,000$, $n = 800$, $\boldsymbol{\mu}_a = [\mathbf{0}_{8(a-1)}; 8; \mathbf{0}_{p-8a+7}]$, $\boldsymbol{C}_a = (1+8(a-1)/\sqrt{p})\boldsymbol{I}_p$, for $a \in \{1, 2\}$, here $\|\boldsymbol{G}^*_{\text{ReLU}} - \overline{\boldsymbol{G}}_{\text{ReLU}}\| \approx 0.26$ and $\|\boldsymbol{G}^*_{\text{Tanh}} - \overline{\boldsymbol{G}}_{\text{Tanh}}\| \approx 0.81$; and on two-class MNIST data (**right**) (number 6 versus number 8), with $p = 784$, $n = 3\,000$, for which $\|\boldsymbol{G}^*_{\text{ReLU}} - \overline{\boldsymbol{G}}_{\text{ReLU}}\| \approx 1.80$ and $\|\boldsymbol{G}^*_{\text{Tanh}} - \overline{\boldsymbol{G}}_{\text{Tanh}}\| \approx 2.02$. For the MNIST case, small eigenvalues close to zero are removed for better visualization.

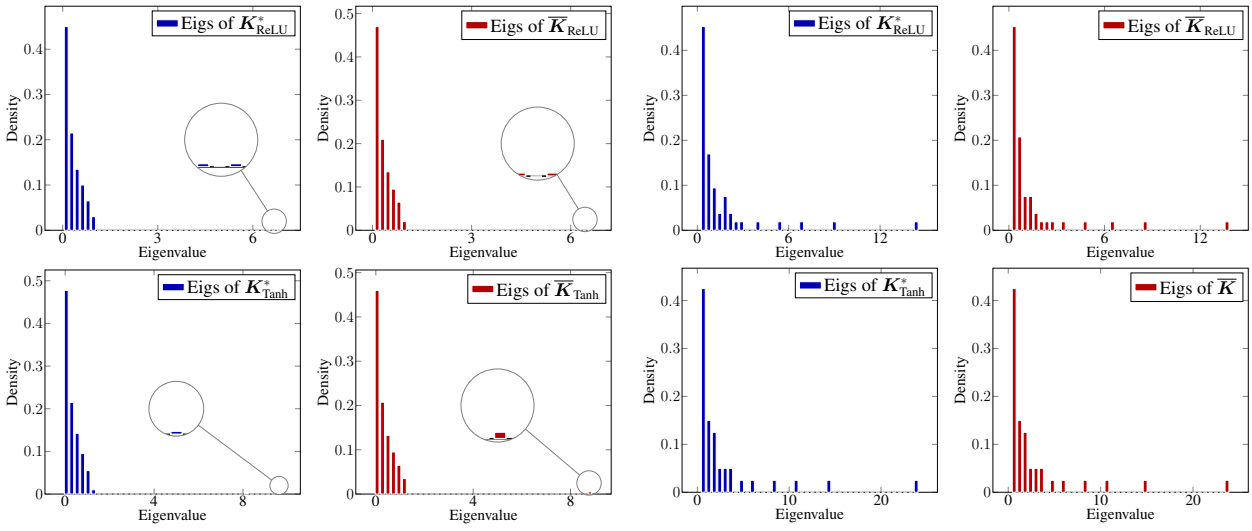

Figure 6. Eigenvalue density of Implicit-NTK matrices (**blue**) $\boldsymbol{K}^*_{\text{ReLU}}$ of ReLU-DEQ (**top**) and $\boldsymbol{K}^*_{\text{Tanh}}$ of Tanh-DEQ (**bottom**) and the corresponding high dimensional approximation $\overline{\boldsymbol{K}}_{\text{ReLU}}$ and $\overline{\boldsymbol{K}}_{\text{Tanh}}$ (**red**) , on two-class GMM data (**left**) with $p = 1\,000$, $n = 800$, $\boldsymbol{\mu}_a = [\mathbf{0}_{8(a-1)}; 8; \mathbf{0}_{p-8a+7}]$, $\boldsymbol{C}_a = (1+8(a-1)/\sqrt{p})\boldsymbol{I}_p$, for $a \in \{1, 2\}$, here $\|\boldsymbol{K}^*_{\text{ReLU}} - \overline{\boldsymbol{K}}_{\text{ReLU}}\| \approx 0.34$ and $\|\boldsymbol{K}^*_{\text{Tanh}} - \overline{\boldsymbol{K}}_{\text{Tanh}}\| \approx 0.76$; and on two-class MNIST data (**right**) (number 6 versus number 8), with $p = 784$, $n = 3\,000$, for which $\|\boldsymbol{K}^*_{\text{ReLU}} - \overline{\boldsymbol{K}}_{\text{ReLU}}\| \approx 3.93$ and $\|\boldsymbol{K}^*_{\text{Tanh}} - \overline{\boldsymbol{K}}_{\text{Tanh}}\| \approx 4.82$. For the MNIST case, small eigenvalues close to zero are removed for better visualization.

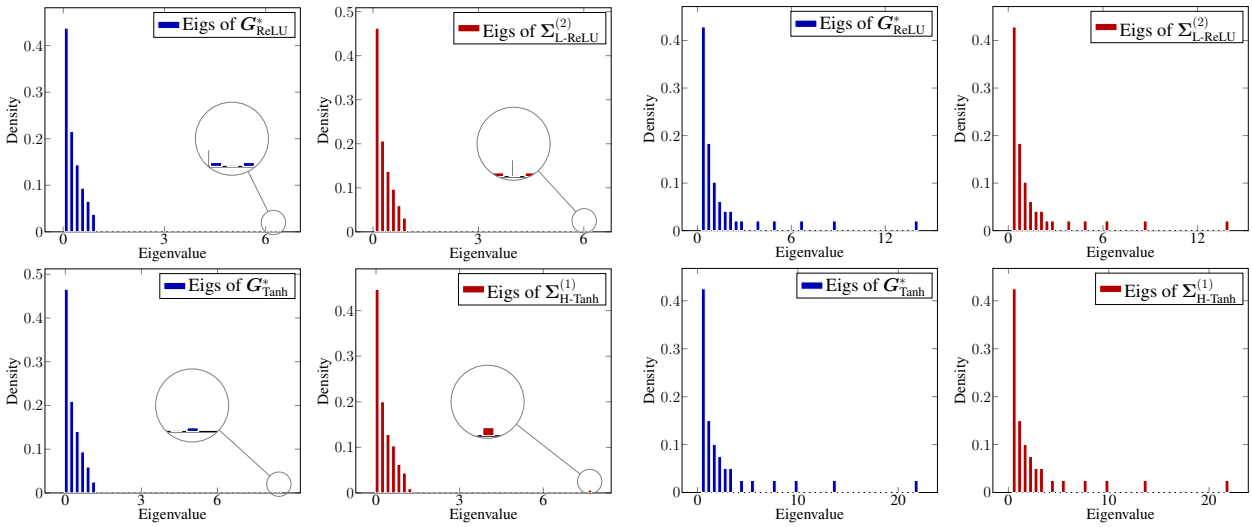

Figure 7. Eigenvalue density of Implicit-CK matrices (**blue**) of ReLU-DEQ (**top**) and Tanh-DEQ (**bottom**) and Explicit-CK matrices (**red**) of the corresponding "equivalent" L-ReLU-ENN and H-Tanh-ENN, on two-class GMM data (**left**) with $p = 1\,000$, $n = 800$, $\boldsymbol{\mu}_a = [\mathbf{0}_{8(a-1)}; 8; \mathbf{0}_{p-8a+7}]$, $\boldsymbol{C}_a = (1+8(a-1)/\sqrt{p})\boldsymbol{I}_p$, for $a \in \{1, 2\}$, here $\|\boldsymbol{G}^*_{\text{ReLU}} - \boldsymbol{\Sigma}^{(2)}_{\text{L-ReLU}}\| \approx 0.32$ and $\|\boldsymbol{G}^*_{\text{Tanh}} - \boldsymbol{\Sigma}^{(1)}_{\text{H-Tanh}}\| \approx 0.88$; and on two-class MNIST data (**right**) (number 6 versus number 8), with $p = 784$, $n = 3\,000$, for which $\|\boldsymbol{G}^*_{\text{ReLU}} - \boldsymbol{\Sigma}^{(2)}_{\text{L-ReLU}}\| \approx 2.34$ and $\|\boldsymbol{G}^*_{\text{Tanh}} - \boldsymbol{\Sigma}^{(1)}_{\text{H-Tanh}}\| \approx 4.79$. For the MNIST case, small eigenvalues close to zero are removed for better visualization.

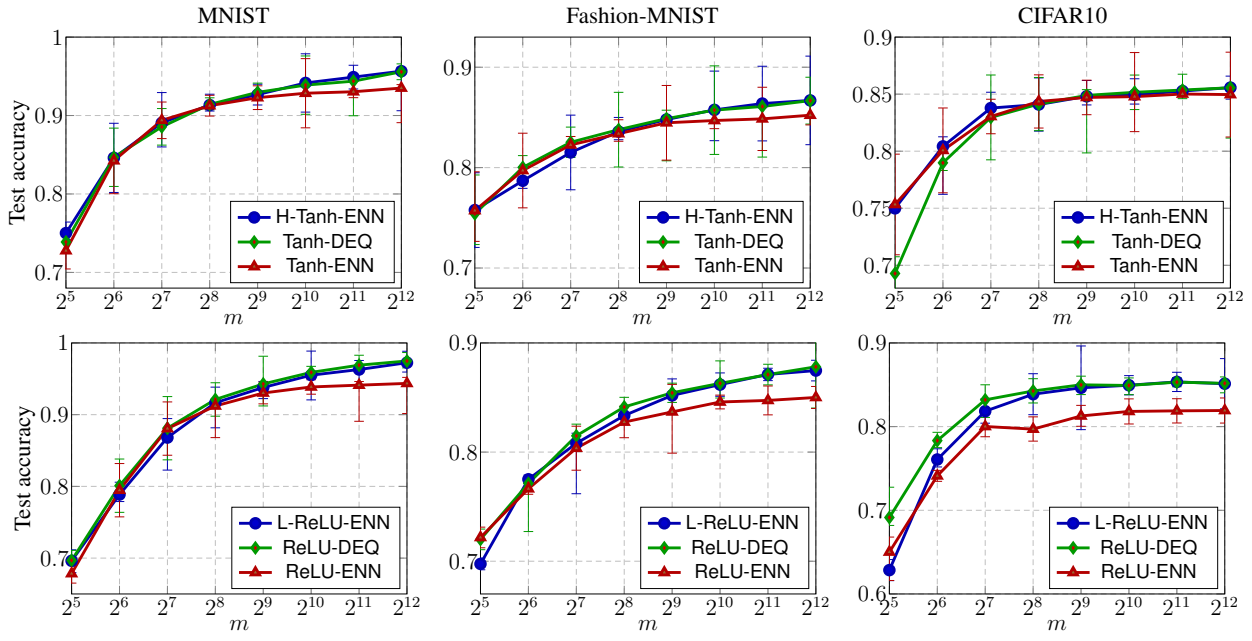

Figure 8. Classification accuracies of implicit DEQs and explicit models trained with Adam. **Top:** Evolution of classification accuracies *w.r.t.* the width m of Tanh-DEQ (**green**), the corresponding equivalent explicit H-Tanh-ENN (**blue**), and Tanh-ENN (**red**). **Bottom:** Evolution of classification accuracies *w.r.t.* the width m of ReLU-DEQ (**green**), the corresponding equivalent explicit L-ReLU-ENN (**blue**), and ReLU-ENN (**red**). For MNIST (**left**) and Fashion-MNIST datasets (**middle**), raw data are taken as the network input; for CIFAR-10 dataset (**right**) , flattened output of the 16th convolutional layer of VGG-19 are used.

Dimension		32	64	128	256	512	1024	2048	4096
MINIST	ReLU-DEQ	0.26	0.26	0.27	0.28	0.27	0.29	0.30	0.32
	L-ReLU-ENN	0.09	0.11	0.12	0.15	0.14	0.11	0.10	0.16
	Tanh-DEQ	0.24	0.24	0.27	0.26	0.28	0.26	0.28	0.29
	H-Tanh-ENN	0.12	0.12	0.11	0.11	0.14	0.11	0.10	0.10
Fashion MINIST	ReLU-DEQ	0.26	0.27	0.26	0.26	0.32	0.32	0.31	0.32
	L-ReLU-ENN	0.17	0.13	0.14	0.11	0.12	0.13	0.12	0.14
	Tanh-DEQ	0.22	0.24	0.25	0.26	0.28	0.27	0.28	0.30
	H-Tanh-ENN	0.13	0.13	0.12	0.12	0.11	0.12	0.10	0.12
CIFAR 10	ReLU-DEQ	0.23	0.22	0.23	0.24	0.27	0.28	0.28	0.31
	L-ReLU-ENN	0.09	0.08	0.09	0.09	0.09	0.09	0.10	0.10
	Tanh-DEQ	0.20	0.20	0.21	0.22	0.23	0.23	0.26	0.27
	H-Tanh-ENN	0.09	0.08	0.09	0.09	0.09	0.08	0.08	0.09

Table 1. Comparison of the inference time for a single input image between DEQs and explicit NNs across different datasets. The inference time is recorded on a machine with Intel(R) Xeon(R) Gold 6138 CPU @ 2.00GHz with a single 3090 GPU.

Dimension		32	64	128	256	512	1024	2048	4096
MINIST	ReLU-DEQ	68.76	72.72	82.00	95.52	97.38	93.96	110.04	189.54
	L-ReLU-ENN	3.045	2.67	3.52	3.75	4.12	3.65	3.09	3.97
	Tanh-DEQ	65.34	83.94	84.72	84.78	85.14	94.14	306.00	322.56
	H-Tanh-ENN	2.76	5.32	2.80	2.65	2.96	2.69	6.59	2.64
Fashion MINIST	ReLU-DEQ	58.46	81.72	72.78	73.38	141.72	156.42	169.74	246.42
	L-ReLU-ENN	3.53	3.17	3.07	2.70	2.84	3.06	2.71	3.11
	Tanh-DEQ	70.80	80.28	81.48	79.38	91.50	96.00	109.98	113.04
	H-Tanh-ENN	2.98	2.78	2.76	2.19	2.71	3.30	2.19	2.47
CIFAR 10	ReLU-DEQ	225.00	253.26	302.16	172.68	870.00	1167.60	1208.40	1204.20
	L-ReLU-ENN	2.47	3.21	2.28	3.05	4.65	5.34	4.71	4.53
	Tanh-DEQ	68.60	79.26	89.22	92.98	108.00	192.46	254.46	307.56
	H-Tanh-ENN	2.24	2.90	2.91	3.06	2.99	2.91	3.36	7.17

Table 2. Comparison of the training time for one epoch (batch size 128) between DEQs and explicit NNs across different datasets. The running time is recorded on a machine with Intel(R) Xeon(R) Gold 6138 CPU @ 2.00GHz with a single 3090 GPU.