"Lossless" Compression of Deep Neural Networks: A High-dimensional Neural Tangent Kernel Approach

Anonymous Author(s)
Affiliation
Address
email

Abstract

Modern deep neural networks (DNNs) are extremely powerful; however, this comes at the price of increased depth and having more parameters per layer, making their training and inference more computationally challenging. In an attempt to address this key limitation, efforts have been devoted to the compression (e.g., sparsification and/or quantization) of these large-scale machine learning models, so that they can be deployed on low-power IoT devices. In this paper, building upon recent research advances in neural tangent kernel (NTK) and random matrix theory, we provide a novel compression approach to wide and fully-connected *deep* neural nets. Specifically, we demonstrate that in the high-dimensional regime where the number of data points n and their dimension p are both large, and under a Gaussian mixture model for the data, there exists *asymptotic spectral equivalence* between the NTK matrices for a large family of DNN models. This theoretical result enables "lossless" compression of a given DNN to be performed, in the sense that the compressed network yields asymptotically the same NTK as the original (dense and unquantized) network, with its weights and activations taking values *only* in $\{0, \pm 1\}$ up to a scaling. Experiments on both synthetic and real-world data are conducted to support the numerical advantages of the proposed method.

1 Introduction

Modern deep neural networks (DNNs) are becoming increasingly over-parameterized, having more parameters than required to fit the also increasingly large, complex, and high-dimensional data. While the list of successful applications of these large-scale machine learning (ML) models is rapidly growing, the energy consumption of these models is also increasing, making them more challenging to deploy on close-to-user and low-power devices. To address this issue, compression techniques have been proposed that prune, sparsify, and/or quantize DNN models [13, 22], thereby yielding DNNs of a much smaller size that can still achieve satisfactory performance on a given ML task. As an illustrative example, it has been recently demonstrated that at least 90% of the weights in popular DNN models such as VGG19 and ResNet32 can be removed with virtually no performance loss [52].

Despite the remarkable progress achieved by various DNN model compression techniques, due to the nonlinear and highly non-convex nature of DNNs, our theoretical understanding of these large-scale ML models, as well as of their compression schemes, is progressing at a more modest pace. For example, it is unclear how much a given DNN model can be compressed *without* severe performance degradation; perhaps more importantly, on the degree to which such a *trade-off between performance and complexity* depends on the ML problem and the data also remains unknown.

In this respect, neural tangent kernels (NTKs) [27], provide a powerful tool for use in assessing the convergence and generalization properties of very wide (sometimes unrealistically so) DNNs by

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

studying their corresponding NTK eigenspectra, which are *solely* dependent on the input data, the network activation function, and the random weights distribution.[1]

In this paper, building upon recent advances in random matrix theory (RMT) and high-dimensional statistics, we demonstrate that for data $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathbb{R}^p$ drawn from a K-class Gaussian mixture model (GMM), in a high-dimensional and non-trivial classification regime where the input data dimension p and their size n are both large and comparable, the *eigenspectra* of both the NTK and the closely related conjugate kernel (CK) matrices at *any* layer $\ell \in \{1, \ldots, L\}$ are *independent* of the distribution of the i.i.d. entries of the (random) weight matrix $\mathbf{W}_\ell$, provided that they are normalized to have zero mean and unit variance, and *only* depend on the activation function $\sigma_\ell(\cdot)$ via *four* scalar parameters. In a sense, we establish, at least for GMM data, the asymptotic *spectral equivalence* between the NTK matrices of the corresponding network layers, and consequently of the whole network, for a large family of DNN models with possibly very different weights and activations, given that they have normalized weight entries and share the same few activation parameters.

Since the convergence and generalization properties of ultra-wide DNNs depend only on the eigenspectra (i.e., eigenvalue-eigenvector pairs; see also Remark 2 below) of the corresponding NTK matrices [27, 17], in the sense that, e.g., the time evolutions (when trained with gradient descent using a sufficiently small step size) of both the residual error and the in-sample prediction of the (sufficiently wide) network can be expressed as *explicit* functions of the NTK eigenvalues and eigenvectors [17, 1, 25]. We further exploit the above theoretical results to propose a novel NTK-based approach that allows for the "lossless" compression of a given *fully-connected* DNN model, by designing a sparse and quantized DNN that (i) has asymptotically the *same* NTK eigenspectra as the original "dense and full precision" network, and (ii) has both weights and activations taking values in the set $\{-1, 0, +1\}$ before scaling, and can thus be stored and computed much more efficiently.

Despite being derived here for Gaussian mixture data, an unexpected close match is observed between our theory and the empirical results on real-world datasets, suggesting possibly wider applicability for the proposed NTK-based Lossless Compression (NTK-LC) approach. Looking forward, we expect that our analysis will open the door to improved analysis based on RMT and high-dimensional statistics, which will demystify the seemingly striking empirical observations in modern DNNs.

1.1 Our contributions

Our main results can be summarized as follows:

1. We provide, in Theorems 1 and 2 respectively, for GMM data and in the high-dimensional regime (Assumption 1), *precise* spectral characterizations of the CK and NTK matrices of fully-connected networks; furthermore, we show that the CK and NTK eigenspectra do *not* depend on the distribution of i.i.d. weights, and *depend* solely on the activation function via a few scalar parameters.
2. In Corollary 1 and Algorithm 1, we exploit these results to propose a novel DNN compression scheme, named NTK-based Lossless Compression (NTK-LC), with *sparsified and ternarized* weights and activations, which nevertheless do not affect (the spectral behavior of) the NTK matrices.
3. In Section 4, we present empirical evidence on (not so) wide DNNs trained with both synthetic Gaussian and popular real-world data such as MNIST [30] and CIFAR10 [29], and show a factor of 10^3 less memory is needed with the proposed NTK-LC approach, with virtually no performance loss.

1.2 Related work

Neural network model compression. The study of NN compression dates back to early 1990 [31], at which point, in the absence of the (possibly more than) sufficient computational power that we have today, compression techniques allowed neural networks to be empirically evaluated on computers with limited computational and/or storage resources [48]. Alongside the rapid growth of increasingly powerful computing devices, the development of more efficient NN architectures and training/inference protocols, and the need to implement NNs on mobile and low-power devices, (D)NN model compression has become an active research topic and many elegant and efficient compression approaches have been proposed over the years [21, 24, 26, 22]. However, due to the

[1] In the remainder of this article, what we refer to as the "NTK matrix" is essentially the limiting (nonrandom) NTK matrix to which the random NTK converges under the infinite-wide limit.

nonlinear and highly non-convex nature of DNNs, our theoretical understanding of these large-scale ML models, as well as of (e.g., the fundamental "performance and complexity" trade-off of) compressed DNNs, is somewhat limited [22].

Neural tangent kernel. Neural tangent kernel (NTK) theory recently proposed in [27], by considering the limit of infinitely wide DNNs, characterizes the convergence and generalization properties of very wide DNNs when trained using gradient descent with small steps. Initially proposed for fully-connected nets, the NTK framework has been subsequently extended to convolutional [4], graph [15], and recurrent [2] settings. The NTK theory, while having the advantage of being mathematically more tractable (via, e.g., the characterization of the associated reproducing kernel Hilbert space [7]), seems to diverge from the regime on which modern (and not so wide) DNNs operate, see [9, 37, 17].

Random matrix theory and neural networks. Random matrix theory (RMT), a powerful and flexible tool for assessing the behavior of large-scale systems with a large "degree of freedom," is gaining popularity in the field of NN analysis [43, 44], in both shallow [47, 35, 36] and deep [6, 17, 46] settings, and considers both homogeneous (e.g., standard normal) [47, 45] and mixture-type data [35, 3]. From a technical perspective, the most relevant paper is [3] , in which the authors proposed a RMT-inspired NN compression scheme, albeit only in the single-hidden-layer setting. This paper extends the analysis in [3] to multi-layer fully-connected DNNs by focusing on the associated NTK matrices, and proposes a novel sparsification and quantization scheme for fully-connected DNNs (which is in spirit similar to, although formally different from, that proposed in [3]).

1.3 Notations and organization of the paper

We denote scalars by lowercase letters, vectors by bold lowercase, and matrices by bold uppercase. We denote the transpose operator by $(\cdot)^\mathsf{T}$, and use $\|\cdot\|$ to denote the Euclidean norm for vectors and the spectral/operator norm for matrices. For a random variable z, $\mathbb{E}[z]$ denotes the expectation of z. We use $\mathbf{1}_p$ and $\mathbf{I}_p$ to represent an all-ones vector of dimension p and the identity matrix of size $p \times p$.

The remainder of this article is structured as follows. In Section 2, we present the DNN model under study, together with our working assumptions. Section 3 contains our main technical results on the eigenspectra of the conjugate kernel $\mathbf{K}_{\mathrm{CK}}$ and NTK matrix $\mathbf{K}_{\mathrm{NTK}}$, along with an account of how they apply to the compression of fully-connected deep neural nets with the proposed NTK-based Lossless Compression (NTK-LC) approach. Empirical evidence is provided in Section 4 to demonstrate the significant computation and storage savings, with virtually no performance degradation, that can be obtained using NTK-LC. Conclusion and future perspectives are placed in Section 5.

2 Preliminaries

Let $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathbb{R}^p$ be n data vectors independently drawn from one of the K-class Gaussian mixtures $\mathcal{C}_1, \ldots, \mathcal{C}_K$, and let $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n] \in \mathbb{R}^{p \times n}$, with class $\mathcal{C}_a$ having cardinality n_a; that is,

$$\mathbf{x}_i \in \mathcal{C}_a \Leftrightarrow \mathbf{x}_i \sim \mathcal{N}(\boldsymbol{\mu}_a/\sqrt{p}, \mathbf{C}_a/p), \tag{1}$$

for the mean vector $\boldsymbol{\mu}_a \in \mathbb{R}^p$ and covariance matrix $\mathbf{C}_a \in \mathbb{R}^{p \times p}$ associated with class $\mathcal{C}_a$.

In the high-dimensional scenario where n, p are both large and comparable, we position ourselves in the following non-trivial classification setting, so that the K-class classification above is neither trivially easy nor impossible; see also [12] and [8, Section 2].

Assumption 1 (High-dimensional asymptotics)**.** *As $n \to \infty$, we have, for $a \in \{1, \ldots, K\}$ that (i) $p/n \to c \in (0, \infty)$ and $n_a/n \to c_a \in [0, 1)$; (ii) $\|\boldsymbol{\mu}_a\| = O(1)$; (iii) for $\mathbf{C}^\circ \equiv \sum_{a=1}^K \frac{n_a}{n} \mathbf{C}_a$ and $\mathbf{C}_a^\circ \equiv \mathbf{C}_a - \mathbf{C}^\circ$, we have $\|\mathbf{C}_a\| = O(1)$, $\operatorname{tr} \mathbf{C}_a^\circ = O(\sqrt{p})$ and $\operatorname{tr}(\mathbf{C}_a \mathbf{C}_b) = O(p)$ for $a, b \in \{1, \ldots, K\}$; and (iv) $\tau_0 \equiv \sqrt{\operatorname{tr} \mathbf{C}^\circ / p}$ converges in $(0, \infty)$.*

We consider using a *fully-connected* neural network model of depth L for the classification of the above K-class Gaussian mixture. Such a network can be parameterized by a sequence of weight matrices $\mathbf{W}_1 \in \mathbb{R}^{d_1 \times d_0}, \ldots, \mathbf{W}_L \in \mathbb{R}^{d_L \times d_{L-1}}$ (with $d_0 = p$), and nonlinear activation functions $\sigma_1, \ldots, \sigma_L$ that apply entry-wise, so that the network output is given by the following:

$$f(\mathbf{x}) = \frac{1}{\sqrt{d_L}} \mathbf{w}^\mathsf{T} \sigma_L \left(\frac{1}{\sqrt{d_{L-1}}} \mathbf{W}_L \sigma_{L-1} \left(\cdots \frac{1}{\sqrt{d_2}} \sigma_2 \left(\frac{1}{\sqrt{d_1}} \mathbf{W}_2 \sigma_1 (\mathbf{W}_1 \mathbf{x}) \right) \right) \right), \tag{2}$$

for an input data vector $\mathbf{x} \in \mathbb{R}^p$ and output vector $\mathbf{w} \in \mathbb{R}^{d_L}$. We denote $\mathbf{\Sigma}_\ell \in \mathbb{R}^{d_\ell \times n}$ the representations of the data matrix $\mathbf{X} \in \mathbb{R}^{p \times n}$ at layer $\ell \in \{1, \ldots, L\}$ defined as

$$\mathbf{\Sigma}_\ell = \frac{1}{\sqrt{d_\ell}} \sigma_\ell \left(\frac{1}{\sqrt{d_{\ell-1}}} \mathbf{W}_\ell \sigma_{\ell-1} \left(\ldots \frac{1}{\sqrt{d_2}} \sigma_2 \left(\frac{1}{\sqrt{d_1}} \mathbf{W}_2 \sigma_1 (\mathbf{W}_1 \mathbf{X}) \right) \right) \right). \tag{3}$$

The normalization by $\sqrt{d_\ell}$ follows from the NTK literature and ensures the consistent asymptotic behavior of the network in the high-dimensional setting in Assumption 1 and 2; see also [27, 7, 17].

The training and generalization performance of the neural network model defined in (2) are closely related to two types of kernel matrices: the Conjugate Kernel (CK) matrix and Neural Tangent Kernel (NTK) matrix, defined respectively for $\ell \in \{1, \ldots, L\}$ as follows:

$$\mathbf{K}_{\mathrm{CK},\ell} = \mathbb{E}[\mathbf{\Sigma}_\ell^\mathsf{T} \mathbf{\Sigma}_\ell] \in \mathbb{R}^{n \times n}, \tag{4}$$

with expectation taken with respect to the weights and $\mathbf{\Sigma}_\ell \in \mathbb{R}^{d_\ell \times n}$ the data representation at the output of layer ℓ defined in (3). In particular, CKs are known to satisfy the following recursive relation [27, 7]

$$[\mathbf{K}_{\mathrm{CK},\ell}]_{ij} = \mathbb{E}_{u,v}[\sigma_\ell(u)\sigma_\ell(v)], \text{ with } u, v \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} [\mathbf{K}_{\mathrm{CK},\ell-1}]_{ii} & [\mathbf{K}_{\mathrm{CK},\ell-1}]_{ij} \\ [\mathbf{K}_{\mathrm{CK},\ell-1}]_{ij} & [\mathbf{K}_{\mathrm{CK},\ell-1}]_{jj} \end{bmatrix} \right), \tag{5}$$

while for the NTK matrix $\mathbf{K}_{\mathrm{NTK},\ell} \in \mathbb{R}^{n \times n}$ of layer ℓ, we have:

$$\mathbf{K}_{\mathrm{NTK},\ell} = \mathbf{K}_{\mathrm{CK},\ell} + \mathbf{K}_{\mathrm{NTK},\ell-1} \circ \mathbf{K}'_{\mathrm{CK},\ell}, \quad \mathbf{K}_{\mathrm{NTK},0} = \mathbf{K}_{\mathrm{CK},0} = \mathbf{X}^\mathsf{T}\mathbf{X}, \tag{6}$$

where '$\mathbf{A} \circ \mathbf{B}$' denotes the Hadamard product between two matrices $\mathbf{A}, \mathbf{B}$ of the same size, and $\mathbf{K}'_{\mathrm{CK},\ell}$ denotes the CK matrix with nonlinear function σ'_ℓ instead of σ_ℓ as for $\mathbf{K}_{\mathrm{CK},\ell}$ defined in (5); that is, $[\mathbf{K}'_{\mathrm{CK},\ell}]_{ij} = \mathbb{E}_{u,v}[\sigma'_\ell(u)\sigma'_\ell(v)]$. Note in particular that for a given DNN model, the corresponding CK and NTK matrices depend *only* on the network structure (i.e., the number of layers and the activation function in each layer), the *distribution* of the random (initializations of the) weights to be integrated over (e.g., in the expectation in equation (4)), and the input data.

It has been shown in a series of previous efforts [27, 17, 25] that for very (and sometimes unrealistically) wide DNNs trained using gradient descent with a small step size, the time evolution of the residual errors and in-sample predictions of a given DNN are *explicit* functionals of the corresponding $\mathbf{K}_{\mathrm{NTK}}$ involving its eigenvalues and eigenvectors. In this respect, the NTK theory provides, via the eigenspectral behavior of $\mathbf{K}_{\mathrm{NTK}}$, precise characterizations of the convergence and generalization properties of DNNs [27, 7], by focusing on the impact of the network structure (e.g., the number of layers and the choice of activation functions), the input data, and the weight initialization schemes.

In this paper, we focus on fully-connected nets under the following assumption regarding the weights.

Assumption 2 (On random weights)**.** *The weight matrices $\mathbf{W}_1 \in \mathbb{R}^{d_1 \times p}, \ldots, \mathbf{W}_L \in \mathbb{R}^{d_L \times d_{L-1}}$ are independent and have i.i.d. entries of zero mean, unit variance, and finite fourth-order moment.*

Assumption 2, together with the $\sqrt{d_\ell}$ normalization, is compatible with *fully-connected* DNNs in (2), which are admittedly less interesting, from a practical perspective, compared to their convolutional counterparts. The proposed framework is envisioned to be extendable to a convolutional [4, 7] and more involved setting (e.g., graph NNs [15]) by considering (e.g., Toeplitz-type) structures on $\mathbf{W}$s.

Unlike most existing NTK literature [27, 7, 16], we do not assume the Gaussianity of the entries of $\mathbf{W}_\ell$s, but only that they are i.i.d. and "normalized" to have zero mean and unit variance. As it turns out, this assumption together with a (Lyapunov-type) central limit theorem argument, is sufficient to establish most existing results on the convergence and generalization of DNNs; see for example [33].

Assumption 3 (On activation functions)**.** *The activation functions $\sigma_1(\cdot), \ldots, \sigma_L(\cdot)$ in each layer are at least four-times differentiable with respect to standard normal measure, in the sense that $\max_{k \in \{0,1,2,3,4\}} \{|\mathbb{E}[\sigma_\ell^{(k)}(\xi)]|\}$ is finite for $\xi \sim \mathcal{N}(0,1)$ and $\ell \in \{1, \ldots, L\}$.*

Using the Gaussian integration by parts formula, one has $\mathbb{E}[\sigma'(\xi)] = \mathbb{E}[\xi\sigma(\xi)]$ for $\xi \sim \mathcal{N}(0,1)$, as long as the right side expectation exists. As a result, even for non-differentiable functions, it suffices to have $|\sigma_\ell|$ upper-bounded by some (high-degree) polynomial function for Assumption 3 to hold.

With these preliminaries, we can move on to present our main technical results on the spectral behavior of the CK and NTK matrices for a large family of fully-connected DNNs.

3 Main results

For a fully-connected DNN defined in (2), our first result is on the eigenspectral behavior of the corresponding CK matrices $\mathbf{K}_{\rm CK}$ defined in (4). More specifically, we show for Gaussian mixture data in (1) and in the high-dimensional setting of Assumption 1, that the $\mathbf{K}_{{\rm CK},\ell}$ of layer $\ell \in \{1, \ldots, L\}$ is asymptotically *spectrally equivalent* to another random matrix $\tilde{\mathbf{K}}_{{\rm CK},\ell}$, in the sense that their spectral norm difference $\|\mathbf{K}_{{\rm CK},\ell} - \tilde{\mathbf{K}}_{{\rm CK},\ell}\|$ vanishes as $n, p \to \infty$. This result is stated as follows, the proof of which is based on an induction on ℓ and is given in Section A.1 of the appendix.

Theorem 1 (Asymptotic equivalents for CK matrices)**.** *Let Assumptions 1–3 hold, and let $\tau_0, \tau_1, \ldots, \tau_L \geq 0$ be a sequence of non-negative numbers satisfying the following recursion:*

$$\tau_\ell = \sqrt{\mathbb{E}[\sigma_\ell^2(\tau_{\ell-1}\xi)]}, \quad \xi \sim \mathcal{N}(0,1), \quad \ell \in \{1, \ldots, L\}. \tag{7}$$

Further assume that the activation functions $\sigma_\ell(\cdot)$s are "centered," such that $\mathbb{E}[\sigma_\ell(\tau_{\ell-1}\xi)] = 0$. Then, for the CK matrix $\mathbf{K}_{{\rm CK},\ell}$ of layer $\ell \in \{1, \ldots, L\}$ defined in (4)*, as $n, p \to \infty$, one has that:*

$$\|\mathbf{K}_{{\rm CK},\ell} - \tilde{\mathbf{K}}_{{\rm CK},\ell}\| \to 0, \quad \tilde{\mathbf{K}}_{{\rm CK},\ell} \equiv \alpha_{\ell,1}\mathbf{X}^\mathsf{T}\mathbf{X} + \mathbf{V}\mathbf{A}_\ell\mathbf{V}^\mathsf{T} + (\tau_\ell^2 - \tau_0^2\alpha_{\ell,1} - \tau_0^4\alpha_{\ell,3})\mathbf{I}_n, \tag{8}$$

almost surely, with

$$\mathbf{V} = [\mathbf{J}/\sqrt{p},\ \boldsymbol{\psi}] \in \mathbb{R}^{n\times(K+1)}, \quad \mathbf{A}_\ell = \begin{bmatrix} \alpha_{\ell,2}\mathbf{t}\mathbf{t}^\mathsf{T} + \alpha_{\ell,3}\mathbf{T} & \alpha_{\ell,2}\mathbf{t} \\ \alpha_{\ell,2}\mathbf{t}^\mathsf{T} & \alpha_{\ell,2} \end{bmatrix} \in \mathbb{R}^{(K+1)\times(K+1)}, \tag{9}$$

for class label vectors $\mathbf{J} = [\mathbf{j}_1, \ldots, \mathbf{j}_K] \in \mathbb{R}^{n\times K}$, "second-order" data fluctuation vector $\boldsymbol{\psi} = \{\|\mathbf{x}_i - \boldsymbol{\mu}_a/\sqrt{p}\|^2 - \mathbb{E}[\|\mathbf{x}_i - \boldsymbol{\mu}_a/\sqrt{p}\|^2]\}_{i=1}^n \in \mathbb{R}^n$, second-order discriminative statistics $\mathbf{t} = \{\operatorname{tr}\mathbf{C}_a^\circ/\sqrt{p}\}_{a=1}^K \in \mathbb{R}^K$ and $\mathbf{T} = \{\operatorname{tr}\mathbf{C}_a\mathbf{C}_b/p\}_{a,b=1}^K \in \mathbb{R}^{K\times K}$ of the Gaussian mixture in (1)*, as well as non-negative scalars $\alpha_{\ell,1}, \alpha_{\ell,2}, \alpha_{\ell,3} \geq 0$ satisfying*

$$\alpha_{\ell,1} = \mathbb{E}[\sigma_\ell'(\tau_{\ell-1}\xi)]^2\alpha_{\ell-1,1}, \quad \alpha_{\ell,2} = \mathbb{E}[\sigma_\ell'(\tau_{\ell-1}\xi)]^2\alpha_{\ell-1,2} + \frac{1}{4}\mathbb{E}[\sigma_\ell''(\tau_{\ell-1}\xi)]^2\alpha_{\ell-1,4}^2, \tag{10}$$

$$\alpha_{\ell,3} = \mathbb{E}[\sigma_\ell'(\tau_{\ell-1}\xi)]^2\alpha_{\ell-1,3} + \frac{1}{2}\mathbb{E}[\sigma_\ell''(\tau_{\ell-1}\xi)]^2\alpha_{\ell-1,1}^2. \tag{11}$$

with $\alpha_{\ell,4} = \alpha_{\ell-1,4}\mathbb{E}\left[(\sigma_\ell'(\tau_{\ell-1}\xi))^2 + \sigma_\ell(\tau_{\ell-1}\xi)\sigma_\ell''(\tau_{\ell-1}\xi)\right]$ for $\xi \sim \mathcal{N}(0,1)$.

A few remarks on Theorem 1 are in order. The first remark is on the condition $\mathbb{E}[\sigma_\ell(\tau_{\ell-1}\xi)] = 0$.

Remark 1 (On activation centering)**.** *The condition $\mathbb{E}[\sigma_\ell(\tau_{\ell-1}\xi)] = 0$, seemingly restrictive at first sight, in fact only subtracts an* identical constant *from all entries of the data representation $\boldsymbol{\Sigma}_\ell$ at layer ℓ as defined in* (3)*, and should* not *restrict the expressive power of the network, nor its performance on downstream ML tasks. For a given DNN model of interest, it suffices to "center" the output of each layer by subtracting a constant to satisfy Assumption 3, and to further apply our Theorem 1.*

Theorem 1 unveils the (possibly surprising) fact that, for the high-dimensional and non-trivial Gaussian mixture classification of (1), the spectral behavior of $\tilde{\mathbf{K}}_{{\rm CK},\ell}$ (and thus that of the CK matrix $\mathbf{K}_{{\rm CK},\ell}$) is (i) *independent* of the distribution of the (entries of the) weights $\mathbf{W}_\ell$ when they are "normalized" to have zero mean and unit variance, as demanded in Assumption 2, and (ii) depends on the activation function σ_ℓ *only* via four[2] scalar parameters $\alpha_{\ell,1}, \alpha_{\ell,2}, \alpha_{\ell,3}$ and τ_ℓ: such universal results have been previously observed in random matrix theory and high-dimensional statistics literature (see for example [11, 5, 55]) and indicate the wide applicability of our theoretical result.

On closer inspection of Theorem 1, we further observe that:

(i) for a given DNN, Theorem 1 characterizes, via the form of $\tilde{\mathbf{K}}_{{\rm CK},\ell}$ in (8) and the recursions in (10) and (11), how the linear (via $\alpha_{\ell,1}$, which is multiplied by $\mathbf{X}^\mathsf{T}\mathbf{X}$) and nonlinear (via $\alpha_{\ell,2}$ and $\alpha_{\ell,3}$ in $\mathbf{A}_\ell$, which respectively weight the second-order data statistics $\mathbf{t}$ and $\mathbf{T}$) data features "propagate," in a layer-by-layer fashion, as ℓ increases, as *quantitatively* measured by the corresponding α_ℓs; and

[2] It is worth noting that the parameter τ_ℓ appears in the CK eigenspectrum *only* by shifting all its eigenvalues (by τ_ℓ^2), thereby acting as an (implicit) ridge-type regularization in large-scale DNN models, see also [28, 14, 38].

(ii) for two DNNs with the same number of layers, but possibly different weights and activations, given the same input data $\mathbf{X}$ (so that the two nets have the same $\mathbf{K}_{\mathrm{CK},0}$), if they have asymptotically equivalent CK matrices $\mathbf{K}_{\mathrm{CK},\ell-1}$ at layer $\ell-1$ with the *same* $\alpha_{\ell-1,1}$, $\alpha_{\ell-1,2}$ and $\alpha_{\ell-1,3}$, then having the *same* $\mathbb{E}[\sigma_\ell'(\tau_{\ell-1}\xi)]^2$, $\mathbb{E}[\sigma_\ell''(\tau_{\ell-1}\xi)]^2$, and $\mathbb{E}[(\sigma_\ell^2(\tau_{\ell-1}\xi))'']$ (which *only* depends on the activation σ_ℓ of layer ℓ and $\tau_{\ell-1}$) will suffice for the two nets to have equivalent $\mathbf{K}_{\mathrm{CK},\ell}$ at layer ℓ.

It follows from the above item (ii) that for a given DNN of depth L, it is possible to design a novel DNN model that "matches" the original one – in the sense that both models will have asymptotically equivalent CK matrices *at each layer*, by using the following layer-by-layer matching strategy: Starting from the same $\mathbf{K}_{\mathrm{CK},0} = \mathbf{X}^\mathsf{T}\mathbf{X}$, one chooses the weights $\mathbf{W}_1$ of the novel DNN according to Assumption 2, and then select the first-layer activation σ_1 in such a way that the novel net has the same parameters $\alpha_{1,1}$, $\alpha_{1,2}$ and $\alpha_{1,3}$ as the original one, so that the first-layer CK matrices $\mathbf{K}_{\mathrm{CK},1}$ of the two nets are *spectrally* matched as per Theorem 1; one then proceeds similarly to match the second, the third, etc., and eventually the Lth layer of the two nets. As we shall see below, this layer-by-layer matching strategy facilitates the "lossless" compression of a given DNN.

Using the relation in (6), a similar result (as in Theorem 1 for CKs) can be established for NTK matrices, as shown in the following theorem. The proof is provided in Section A.2 of the appendix.

Theorem 2 (Asymptotic equivalent for NTK matrices)**.** *Let Assumptions 1–3 hold, let $\sigma_\ell(\cdot)$s be centered so that $\mathbb{E}[\sigma_\ell(\tau_{\ell-1}\xi)] = 0$ for τ_ℓs defined in* (7)*, and let $\kappa_1, \ldots, \kappa_L \geq 0$ be a sequence of non-negative numbers satisfying the following recursion:*

$$\kappa_\ell = \sqrt{\tau_\ell^2 + \kappa_{\ell-1}^2 \mathbb{E}[(\sigma_{\ell-1}'(\tau_{\ell-2}\xi))^2]}, \quad \xi \sim \mathcal{N}(0,1), \quad \ell \in \{1, \ldots, L\}. \tag{12}$$

Then, for the NTK matrix $\mathbf{K}_{\mathrm{NTK},\ell}$ of layer ℓ defined in (6)*, as $n, p \to \infty$ one has that*

$$\|\mathbf{K}_{\mathrm{NTK},\ell} - \tilde{\mathbf{K}}_{\mathrm{NTK},\ell}\| \to 0, \quad \tilde{\mathbf{K}}_{\mathrm{NTK},\ell} \equiv \beta_{\ell,1}\mathbf{X}^\mathsf{T}\mathbf{X} + \mathbf{V}\mathbf{B}_\ell\mathbf{V}^\mathsf{T} + (\kappa_\ell^2 - \tau_0^2\beta_{\ell,1} - \tau_0^4\beta_{\ell,3})\mathbf{I}_n, \tag{13}$$

almost surely, with $\mathbf{V} \in \mathbb{R}^{n\times(K+1)}$, $\mathbf{t} \in \mathbb{R}^K$, $\mathbf{T} \in \mathbb{R}^{K\times K}$ as defined in Theorem 1, and

$$\mathbf{B}_\ell \equiv \begin{bmatrix} \beta_{\ell,2}\mathbf{t}\mathbf{t}^\mathsf{T} + \beta_{\ell,3}\mathbf{T} & \beta_{\ell,2}\mathbf{t} \\ \beta_{\ell,2}\mathbf{t}^\mathsf{T} & \beta_{\ell,2} \end{bmatrix}, \tag{14}$$

as well as the non-negative constants $\beta_{\ell,1}, \beta_{\ell,2}, \beta_{\ell,3}, \beta_{\ell,4} \geq 0$, such that

$$\beta_{\ell,1} = \alpha_{\ell,1}, \quad \beta_{\ell,2} = \alpha_{\ell,2}, \quad \beta_{\ell,3} = \alpha_{\ell,3} + \beta_{\ell-1,1}\beta_{\ell,4}, \tag{15}$$

and $\beta_{\ell,4}$ satisfies the recursion $\beta_{\ell,4} = \mathbb{E}[\sigma_\ell''(\tau_{\ell-1}\xi)]^2\beta_{\ell-1,4}$.

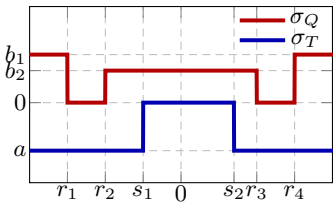
Roughly speaking, Theorem 2 shows that a similar eigenspectral behavior established in Theorem 1 for CK matrices also holds for NTK matrices if we change the associated coefficients α_ℓs to β_ℓs. The remarks after Theorem 1 thus remain valid, at least in spirit, for NTK matrices.

Remark 2 (On spectral norm characterization)**.** *Note that the characterizations in Theorem 1 and 2 for CK and NTK matrices are provided in a spectral norm sense. It then follows from Weyl's inequality [23] and the Davis–Kahan theorem [59] that the difference between the eigenvalues (e.g., when listed in a decreasing order) and the associated eigenvectors (when the eigenvalues under study are "isolated") of $\mathbf{K}_{\mathrm{NTK}}$ and $\tilde{\mathbf{K}}_{\mathrm{NTK}}$ vanish asymptotically as $n, p \to \infty$. As such, the spectral norm guarantees in Theorem 1 and 2 provide more tractable access to the convergence and generalization properties of wide DNNs, at least for GMM data, via the spectral study of $\tilde{\mathbf{K}}_{\mathrm{NTK}}$ [27, 17].*

Despite being derived here for the Gaussian mixture model in (1), we conjecture that the results in Theorem 1 and 2 hold beyond the Gaussian setting and can be extended, for example, to the family of concentrated random vectors [49, 32]. As previously discussed after Assumption 2 for the distribution of $\mathbf{W}$, universality commonly arises in random matrix theory and high-dimensional statistics [11, 5, 55]; we refer interested readers to Remark 4 in Appendix A for further discussions.

From a technical perspective, the results in Theorem 1 and 2 extend the single-hidden-layer CK analysis in [3, 35] to both CK and NTK matrices of fully-connected DNNs with an arbitrary number of layers. In particular, taking $\ell = 1$ in Theorem 1, one obtains [3, Theorem 1] as a special case.[3]

[3] Note that in [3, Theorem 1], the authors do *not* assume $\mathbb{E}[\sigma(\tau_0\xi)] = 0$ as in our Theorem 1, but instead "center" the CK matrices by pre- and post-multiplying $\mathbf{K}_{\mathrm{CK}}$ with $\mathbf{P} = \mathbf{I}_n - \mathbf{1}_n\mathbf{1}_n^\mathsf{T}/n$. This can be shown equivalent to taking $\mathbb{E}[\sigma(\tau_0\xi)] = 0$ in the single-hidden-layer setting; see Appendix A.3 for more discussions.



$$\mathbb{E}[\sigma_T(\tau\xi)] = \frac{a}{2}\left(\mathrm{erf}\left(\frac{s_1}{\sqrt{2}\tau}\right) - \mathrm{erf}\left(\frac{s_2}{\sqrt{2}\tau}\right)\right) + a$$

$$\mathbb{E}[\sigma_Q(\tau\xi)] = \frac{b_1}{2}\left(\mathrm{erf}\left(\frac{r_1}{\sqrt{2}\tau}\right) - \mathrm{erf}\left(\frac{r_4}{\sqrt{2}\tau}\right)\right) + b_1 + \frac{b_2}{2}\left(\mathrm{erf}\left(\frac{r_3}{\sqrt{2}\tau}\right) - \mathrm{erf}\left(\frac{r_2}{\sqrt{2}\tau}\right)\right)$$

Figure 1: Visual representations of activations σ_T and σ_Q defined in (17) **(left)** and the detailed expressions of $\mathbb{E}[\sigma_T(\tau\xi)]$ and $\mathbb{E}[\sigma_Q(\tau\xi)]$**(right)**, with specifically $r_1 - r_2 = r_3 - r_4$ here.

Remark 3 (On the equivalent CK and NTK matrices for two neural nets)**.** *It follows from Theorem 1 that for a given DNN model, it suffices to "match" the coefficients $\alpha_{\ell,1}, \alpha_{\ell,2}$ and $\alpha_{\ell,3}$ in a layer-by-layer manner to design a novel DNN with asymptotically equivalent CK. In doing so, one can choose the activation σ_ℓ at layer ℓ to satisfy a system of equations involving $\mathbb{E}[\sigma'_\ell(\tau_{\ell-1}\xi)]^2$, $\mathbb{E}[\sigma''_\ell(\tau_{\ell-1}\xi)]^2$, and $\mathbb{E}\left[(\sigma'_\ell(\tau_{\ell-1}\xi))^2 + \sigma_\ell(\tau_{\ell-1}\xi)\sigma''_\ell(\tau_{\ell-1}\xi)\right]$. Interestingly, it can then be checked from Theorem 2 that, by performing the above matching, if one has equivalent NTK matrices $\mathbf{K}_{\mathrm{NTK},\ell-1}$ at layer $\ell - 1$ for the two nets under consideration, then the corresponding $\mathbf{K}_{\mathrm{NTK},\ell}$ at layer ℓ are also "matched" in a spectral sense, since the (additional) parameter $\mathbb{E}[\sigma''_\ell(\tau_{\ell-1}\xi)]^2$ in the recursion of $\beta_{\ell,4}$ in Theorem 2 has already been matched. Also, similarly to Footnote 2 for CK, matching (or nor) the parameter κ_ℓ only affects the level of regularization by "shifting" all eigenvalues of $\mathbf{K}_{\mathrm{NTK}}$.*

In the following corollary, we present a concrete example of how to apply the results in Theorem 1 and 2 in the design of a novel computationally and storage efficient DNN, that shares the same CK and NTK eigenspectra with any given fully-connected neural net having centered activation.

Corollary 1 (Sparse and quantized DNNs)**.** *For a given fully-connected DNN (referred to as* DNN1*) of depth L with centered activation such that $\mathbb{E}[\sigma_\ell(\tau_{\ell-1}\xi)] = 0$ for $\xi \sim \mathcal{N}(0,1)$, one is able to construct (again in a layer-by-layer manner) a sparse and quantized "equivalent" DNN model, of depth L and referred to as* DNN2*, such that the two nets have asymptotically the same eigenspectra for their CK (and thus NTK, as per Remark 3) matrices, by using the following ternary weights:*

$$[\mathbf{W}]_{ij} = 0 \text{ with proba } \varepsilon \in [0,1), \quad [\mathbf{W}]_{ij} = \pm(1-\varepsilon)^{-1/2} \text{ each with proba } 1/2 - \varepsilon/2, \tag{16}$$

as well as quantized activations (as visually displayed in Figure 1):

$$\sigma_T(t) = a \cdot (1_{t<s_1} + 1_{t>s_2}), \quad \sigma_Q(t) = b_1 \cdot (1_{t<r_1} + 1_{t>r_4}) + b_2 \cdot 1_{r_2 \le t \le r_3}. \tag{17}$$

We refer readers to Section A.4 in the appendix for the proof and discussions of Corollary 1, as well as the detailed expressions of $\mathbb{E}[\sigma'(\tau\xi)]$, $\mathbb{E}[\sigma''(\tau\xi)]$, and $\mathbb{E}[(\sigma^2(\tau\xi))'']$, of direct algorithmic use for both σ_T and σ_Q as functions of a, s_1, s_2 and $b_1, b_2, r_1, r_2, r_3, r_4$. The proposed NTK-LC compression approach based on Corollary 1, is described in detail below.

Note that, the "sign" of activations does not really matter in the design of computationally efficient DNNs described in Algorithm 1, in the sense that the key parameters $\alpha_{\ell,1}, \alpha_{\ell,2}$ and $\alpha_{\ell,3}$ for CKs, as well as $\beta_{\ell,1}, \beta_{\ell,2}$ and $\beta_{\ell,3}$ for NTKs, remain unchanged when $-\sigma_\ell(t)$ is used instead of $\sigma_\ell(t)$.

Before embarking on the detailed numerical experiments in Section 4, we would like to bring the readers' attention to the recent line of works [34, 52, 56, 53, 18] showing that for wide and deep NN models, very efficient sparse sub-networks can be found that almost match the performance of the original dense nets *with little or even no training*, for instance by uniformly pruning the network weights [52]. To develop a theoretical grasp of these (extremely counterintuitive) empirical successes, a few attempts have been made, for example, to carefully prune the network weights to retain the same (limiting) NTK [39], or to show that randomly pruned sparse nets have the same (limiting) NTK as the original net up to a scaling factor [58]. Instead, our work, by considering the *statistical structure* of the input data, leverages tools from RMT to "compress" both the weights and activations (per Theorem 2 and Corollary 1) without affecting the NTK eigenstructure.

Algorithm 1 NTK-based "lossless" compression (NTK-LC)

Input: Input data $\mathbf{x}_1, \ldots, \mathbf{x}_n$, sparsity level $\varepsilon \in [0, 1)$, and DNN1 with $\sigma_1, \ldots, \sigma_L$ activations
Output: Sparse and quantized DNN2 model with weights $\mathbf{W}_\ell$s and activations $\tilde{\sigma}_\ell$s.

Estimate τ_0 from input data as $\tau_0 = \sqrt{\frac{1}{n}\sum_{i=1}^n \|\mathbf{x}_i\|^2}$. Set $\tau = \tau_0$, $\tilde{\tau} = \tau_0$.

for $\ell = 1, \ldots, L-1$ **do**

  Compute $\mathbb{E}[\sigma'_\ell(\tau\xi)]^2$, $\mathbb{E}[\sigma''_\ell(\tau\xi)]^2$, and $\mathbb{E}[(\sigma^2_\ell(\tau\xi))'']$, and use them to solve for the coefficients a, s_1, s_2 in the novel activation $\tilde{\sigma}_\ell$, with $\tilde{\tau}$, Corollary 1, and expressions in Appendix A.4.

  Set $\tau = \sqrt{\mathbb{E}[\sigma^2_\ell(\tau\xi)]}$, $\tilde{\tau} = \sqrt{\mathbb{E}[\sigma^2_\ell(\tilde{\tau}\xi)]}$.

end for

For layer $\ell = L$, compute $\sqrt{\mathbb{E}[\sigma^2_L(\tau\xi)]}$, $\mathbb{E}[\sigma'_L(\tau\xi)]^2$, $\mathbb{E}[\sigma''_L(\tau\xi)]^2$, and $\mathbb{E}[(\sigma^2_L(\tau\xi))'']$, and use them to solve for the coefficients $b_1, b_2, r_1, r_2, r_3, r_4$ in the novel activation $\tilde{\sigma}_L$, with $\tilde{\tau}$, Corollary 1 and detailed expressions in Appendix A.4.

Draw independently the i.i.d. entries of $\mathbf{W}_1, \ldots, \mathbf{W}_L$ according to (16) with sparsity level ε.

return DNN2 model with weights $\mathbf{W}_\ell$ and activations $\tilde{\sigma}_\ell$, $\ell = 1, \ldots, L$.

4 Numerical experiments

In this section, we provide numerical experiments to (i) validate the asymptotic characterizations in Theorem 1 and 2, on both synthetic GMM and real-world data (such as MNIST and CIFAR10) of (in fact not so) large sizes and dimensions; and to (ii) show how these results can be used to sparsify and quantize fully-connected DNNs, leading to huge savings in computational and storage resources (up to a factor of 10^3 in memory and a level of sparsity $\varepsilon - 90\%$) without significant performance degradation. We refer readers to Section B in the appendix for further experiments and discussions.

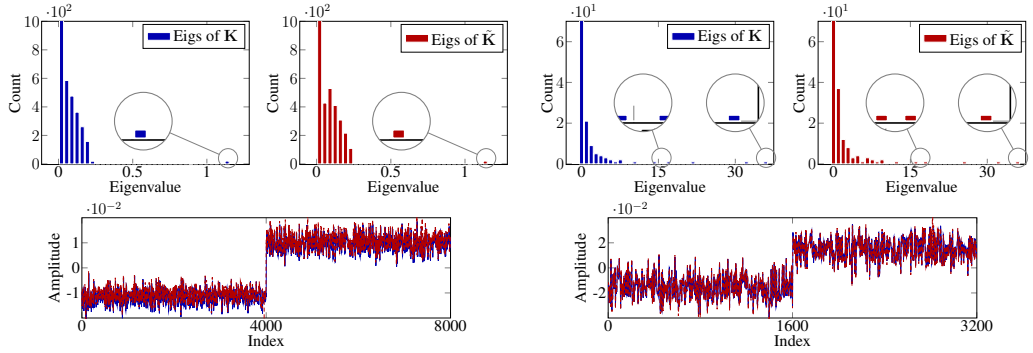


Figure 2: Eigenvalue histograms (**top**) and dominant eigenvectors (**bottom**) of last-layer CK matrices $\mathbf{K}_{\text{CK}}$ (**blue**) defined in (4) (with the expectation estimated from 1000 independent realizations of $\mathbf{W}$s) and the asymptotic equivalent $\tilde{\mathbf{K}}_{\text{CK}}$ (**red**) matrices. (**Left**) Gaussian $\mathbf{W}$ on two-class GMM data, with $p = 2\,000$, $n = 8\,000$, $\boldsymbol{\mu}_a = [\mathbf{0}_{8(a-1)};\ 8;\ \mathbf{0}_{p-8a+7}]$, $\mathbf{C}_a = (1 + 8(a-1)/\sqrt{p})\mathbf{I}_p$, $a \in \{1, 2\}$ using [ReLU, ReLU, ReLU] activations, $\|\mathbf{K}_{\text{CK}} - \tilde{\mathbf{K}}_{\text{CK}}\| = 0.15$; and (**right**) symmetric Bernoulli $\mathbf{W}$ on MNIST data (number 6 versus 8) [30], with $p = 784$, $n = 3\,200$, using [poly, ReLU, ReLU] activations, $\|\mathbf{K}_{\text{CK}} - \tilde{\mathbf{K}}_{\text{CK}}\| = 6.86$. $\mathbf{x}_1, \ldots, \mathbf{x}_{n/2} \in \mathcal{C}_1$ and $\mathbf{x}_{n/2+1}, \ldots, \mathbf{x}_n \in \mathcal{C}_2$ in both cases.

Figure 2 compares the eigenvalues and dominant eigenvectors of the CK matrices $\mathbf{K}_{\text{CK}}$ defined in (4) versus those of their asymptotic approximations $\tilde{\mathbf{K}}_{\text{CK}}$ given in Theorem 1, in the case of fully-connected DNNs having three hidden layers (of width $d_1 = 2\,000, d_2 = 2\,000, d_3 = 1\,000$). For different types of activations ($\text{poly}(t) = 0.2t^2 + t$ and $\text{ReLU}(t) = \max(t, 0)$), different weight distributions (Gaussian and symmetric Bernoulli), and on synthetic GMM as well as MNIST data, we consistantly observe a close match between the eigenvalues and the dominant eigenvectors of $\mathbf{K}_{\text{CK}}$ and $\tilde{\mathbf{K}}_{\text{CK}}$, as a consequence of the spectral norm convergence in Theorem 1 (and Remark 2), suggesting a possibly wider applicability of the proposed results beyond GMM data.[4]

[4]Smallest eigenvalues of $\mathbf{K}_{\text{CK}}, \tilde{\mathbf{K}}_{\text{CK}}$ are removed from the top displays in Figure 2 for better visualization.

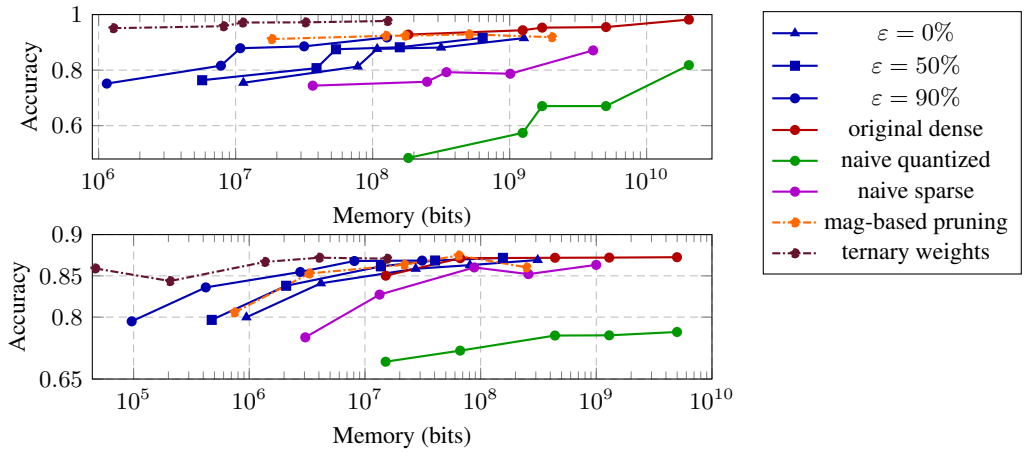


Figure 3: Test accuracy of classification on MNIST [30] (**top**) and CIFAR10 [29] (**bottom**) datasets. **Blue** curves represent the proposed NTK-LC approach with different levels of sparsity $\varepsilon \in \{0\%, 50\%, 90\%\}$, **purple** curves represent the heuristic sparsification approach by uniformly zeroing out 80% of the weights, **green** curves represent the heuristic quantization approach using the binary activation $\sigma(t) = 1_{t<-1} + 1_{t>1}$, **red** curves represent the original network, **orange** curves represent NTK-LC *without* activation quantization, with $\varepsilon = 90\%$ for MNIST (**top**) and $\varepsilon = 95\%$ for CIFAR10 (**bottom**), and **brown** curves represent magnitude-based pruning [19] with the same sparsity level ε as **orange**. Memory varies due to the **change of layer width** of the network.

Figure 3 depicts the test accuracies of (i) the original dense and unquantized network with three fully-connected layers, (ii) the proposed NTK-LC approach described in Corollary 1 and Algorithm 1, as well as its variant having ternarized weights but *dense and unquantized* activations, (iii) the popular magnitude-based pruning approach as proposed in [19], together with (iiii) two "heuristic" compression approaches: (iiii-i) sparsification by uniformly zeroing out 80% of the weights (we *cannot* do more, as the resultant performance is too poor to be compared with other curves) , and (iiii-ii) binarization using $\sigma(t) = 1_{t<-1} + 1_{t>1}$, for different choices of width per layer, and the ten-class classification problems of MNIST and CIFAR10. Specifically, Neural networks before and after compression both have three fully-connected layers, and the original network uses ReLU activation each layer. Classification is performed on a concatenated trainable fully-connected layer. Raw data are taken as network input for MNIST datasets; flattened output of the 16th convolutional layer of VGG19 [51] are taken as net input for CIFAR10 dataset.

In Figure 3, we observe that the proposed NTK-LC approach occupies (up to) a factor of 10^3 less memory, and produces significantly sparser networks (up to 90% of weights set to zero) with minimal performance loss, when compared to the original or the "heuristically" compressed nets. In addition, when compared with its ternary weights variant (with *unquantized* activations), the proposed *sparse and quantized* NTK-LC trades limited performance drop for 32 times less memory, and achieves higher accuracies than the popular magnitude-based pruning under the same memory budget. We also see that the sparsity level ε has limited impact on the classification accuracy, which is in line with our theory. These experimental results show that the proposed NTK-LC approach achieves a better *performance-complexity trade-off* than commonly used heuristic DNN compression methods.

5 Conclusion and perspectives

In this paper, built upon recent advances in random matrix theory and high-dimensional statistics, we provide *precise* characterizations of the eigenspectra of both conjugate kernel and neural tangent kernel matrices, for high-dimensional Gaussian mixture data and fully-connected multi-layer neural nets. These results further allows us to sparsify and quantize fully-connected deep nets, resulting in a factor of 10^3 less memory consumption with virtually no performance degradation.

References

[1] Ben Adlam and Jeffrey Pennington. The Neural Tangent Kernel in High Dimensions: Triple Descent and a Multi-Scale Theory of Generalization. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 74–84. PMLR, 2020.

[2] Sina Alemohammad, Zichao Wang, Randall Balestriero, and Richard Baraniuk. The Recurrent Neural Tangent Kernel. *arXiv*, 2020.

[3] Hafiz Tiomoko Ali, Zhenyu Liao, and Romain Couillet. Random matrices in service of ML footprint: ternary random features with no performance loss. In *International Conference on Learning Representations*, 2022.

[4] Sanjeev Arora, Simon S Du, Wei Hu, Zhiyuan Li, Russ R Salakhutdinov, and Ruosong Wang. On Exact Computation with an Infinitely Wide Neural Net. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

[5] Zhidong Bai and Jack W. Silverstein. *Spectral Analysis of Large Dimensional Random Matrices*, volume 20 of *Springer Series in Statistics*. Springer-Verlag New York, 2 edition, 2010.

[6] Lucas Benigni and Sandrine Péché. Eigenvalue distribution of nonlinear models of random matrices. *arXiv*, 2019.

[7] Alberto Bietti and Julien Mairal. On the Inductive Bias of Neural Tangent Kernels. In *Advances in Neural Information Processing Systems*, volume 32 of *NIPS'19*, pages 12893–12904. Curran Associates, Inc., 2019.

[8] Avrim Blum, John Hopcroft, and Ravindran Kannan. *Foundations of Data Science*. Cambridge University Press, 2020.

[9] Lénaïc Chizat, Edouard Oyallon, and Francis Bach. On Lazy Training in Differentiable Programming. In *Advances in Neural Information Processing Systems*, volume 32 of *NIPS'19*, pages 2937–2947. Curran Associates, Inc., 2019.

[10] Romain Couillet and Florent Benaych-Georges. Kernel spectral clustering of large dimensional data. *Electronic Journal of Statistics*, 10(1):1393–1454, 2016.

[11] Romain Couillet and Zhenyu Liao. *Random Matrix Methods for Machine Learning*. Cambridge University Press.

[12] Romain Couillet, Zhenyu Liao, and Xiaoyi Mai. Classification Asymptotics in the Random Matrix Regime. In *2018 26th European Signal Processing Conference (EUSIPCO)*, volume 00 of *EUSIPCO'18*, pages 1875–1879, 2018.

[13] By Lei Deng, Guoqi Li, Song Han, Luping Shi, and Yuan Xie. Model Compression and Hardware Acceleration for Neural Networks: A Comprehensive Survey. *Proceedings of the IEEE*, 108(4):485–532, 2020.

[14] Michal Derezinski, Feynman T Liang, and Michael W Mahoney. Exact expressions for double descent and implicit regularization via surrogate random design. In *Advances in Neural Information Processing Systems*, volume 33, pages 5152–5164. Curran Associates, Inc., 2020.

[15] Simon S Du, Kangcheng Hou, Russ R Salakhutdinov, Barnabas Poczos, Ruosong Wang, and Keyulu Xu. Graph Neural Tangent Kernel: Fusing Graph Neural Networks with Graph Kernels. 32.

[16] Zhou Fan and Andrea Montanari. The spectral norm of random inner-product kernel matrices. *Probability Theory and Related Fields*, 173(1-2):27–85, 2019.

[17] Zhou Fan and Zhichao Wang. Spectra of the Conjugate Kernel and Neural Tangent Kernel for linear-width neural networks. In *Advances in Neural Information Processing Systems*, volume 33, pages 7710–7721. Curran Associates, Inc., 2020.

[18] Jonathan Frankle, Gintare Karolina Dziugaite, Daniel Roy, and Michael Carbin. Pruning neural networks at initialization: Why are we missing the mark? In *International Conference on Learning Representations*, 2021.

[19] Trevor Gale, Erich Elsen, and Sara Hooker. The state of sparsity in deep neural networks. *arXiv preprint arXiv:1902.09574*, 2019.

[20] J. Han and C. Moraga. The influence of the sigmoid function parameters on the speed of backpropagation learning. 1995.

[21] Song Han, Huizi Mao, and William J Dally. Deep Compression: Compressing Deep Neural Networks with Pruning, Trained Quantization and Huffman Coding. *arXiv*, 2015.

[22] Torsten Hoefler, Dan Alistarh, Tal Ben-Nun, Nikoli Dryden, and Alexandra Peste. Sparsity in deep learning: Pruning and growth for efficient inference and training in neural networks. *Journal of Machine Learning Research*, 22(241):1–124, 2021.

[23] Roger A. Horn and Charles R. Johnson. *Matrix Analysis*. Cambridge University Press, 2 edition, 2012.

[24] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. *arXiv*, 2017.

[25] Jiaoyang Huang and Horng-Tzer Yau. Dynamics of Deep Neural Networks and Neural Tangent Hierarchy. 119:4542–4551, 2020-13.

[26] Itay Hubara, Matthieu Courbariaux, Daniel Soudry, Ran El-Yaniv, and Yoshua Bengio. Binarized Neural Networks. In *Advances in Neural Information Processing Systems*, volume 29 of *NIPS'16*, pages 4107–4115. Curran Associates, Inc., 2016.

[27] Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural Tangent Kernel: Convergence and Generalization in Neural Networks. In *Advances in Neural Information Processing Systems*, volume 31 of *NIPS'18*, pages 8571–8580. Curran Associates, Inc., 2018.

[28] Arthur Jacot, Berfin Simsek, Francesco Spadaro, Clement Hongler, and Franck Gabriel. Implicit regularization of random feature models. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 4631–4640. PMLR, 13–18 Jul 2020.

[29] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[30] Yann LeCun, Leon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[31] Yann LeCun, John Denker, and Sara Solla. Optimal Brain Damage. In *Advances in Neural Information Processing Systems*, volume 2 of *NIPS'90*, pages 598–605. Morgan-Kaufmann, 1990.

[32] Michel Ledoux. *The Concentration of Measure Phenomenon*. Mathematical Surveys and Monographs. 2005.

[33] Jaehoon Lee, Jascha Sohl-dickstein, Jeffrey Pennington, Roman Novak, Sam Schoenholz, and Yasaman Bahri. Deep Neural Networks as Gaussian Processes. In *International Conference on Learning Representations*, 2018.

[34] Namhoon Lee, Thalaiyasingam Ajanthan, and Philip Torr. SNIP: Single-shot network pruning based on connection sensitivity. In *International Conference on Learning Representations*, 2018.

[35] Zhenyu Liao and Romain Couillet. On the Spectrum of Random Features Maps of High Dimensional Data. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 3063–3071, Stockholmsmässan, Stockholm Sweden, 2018. PMLR.

[36] Zhenyu Liao and Romain Couillet. The Dynamics of Learning: A Random Matrix Approach. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 3072–3081, Stockholmsmässan, Stockholm Sweden, 2018. PMLR.

[37] Chaoyue Liu, Libin Zhu, and Misha Belkin. On the linearity of large non-linear models: when and why the tangent kernel is constant. 33:15954–15964.

[38] Fanghui Liu, Zhenyu Liao, and Johan Suykens. Kernel regression in high dimensions: Refined analysis beyond double descent. In *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, volume 130 of *Proceedings of Machine Learning Research*, pages 649–657. PMLR, 13–15 Apr 2021.

[39] Tianlin Liu and Friedemann Zenke. Finding trainable sparse networks through neural tangent transfer. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 6336–6347. PMLR, 13–18 Jul 2020.

[40] Cosme Louart and Romain Couillet. Concentration of Measure and Large Random Matrices with an application to Sample Covariance Matrices. *arXiv*, 2018.

[41] Andrew L. Maas, Awni Y. Hannun, and Andrew Y. Ng. Rectifier Nonlinearities Improve Neural Network Acoustic Models. In *ICML Workshop on Deep Learning for Audio, Speech and Language Processing*, ICML Workshop, page 3, 2013.

[42] Vladimir A Marcenko and Leonid Andreevich Pastur. Distribution of eigenvalues for some sets of random matrices. *Mathematics of the USSR-Sbornik*, 1(4):457, 1967.

[43] Charles H. Martin and Michael W. Mahoney. Traditional and Heavy Tailed Self Regularization in Neural Network Models. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 4284–4293. PMLR, 2019.

[44] Charles H. Martin and Michael W. Mahoney. Heavy-Tailed Universality Predicts Trends in Test Accuracies for Very Large Pre-Trained Deep Neural Networks. In *Proceedings of the 2020 SIAM International Conference on Data Mining (SDM)*, SDM'20, pages 505–513. SIAM, 2020.

[45] Song Mei and Andrea Montanari. The generalization error of random features regression: Precise asymptotics and double descent curve. *arXiv*, 2019.

[46] Leonid Pastur. On Random Matrices Arising in Deep Neural Networks. Gaussian Case. *arXiv*, 2020.

[47] Jeffrey Pennington and Pratik Worah. Nonlinear random matrix theory for deep learning. In *Advances in Neural Information Processing Systems*, volume 30 of *NIPS'17*, pages 2637–2646. Curran Associates, Inc., 2017.

[48] Jürgen Schmidhuber. Deep learning in neural networks: An overview. *Neural Networks*, 61:85–117, 2015.

[49] Mohamed El Amine Seddik, Cosme Louart, Mohamed Tamaazousti, and Romain Couillet. Random Matrix Theory Proves that Deep Learning Representations of GAN-data Behave as Gaussian Mixtures. In *Proceedings of the 37th International Conference on Machine Learning*, Proceedings of Machine Learning Research, pages 8573–8582. PMLR, 2020.

[50] Mohamed El Amine Seddik, Mohamed Tamaazousti, and Romain Couillet. A Kernel Random Matrix-Based Approach for Sparse PCA. In *International Conference on Learning Representations*, ICLR'19, 2019.

[51] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*, ICLR'14, 2014.

[52] Jingtong Su, Yihang Chen, Tianle Cai, Tianhao Wu, Ruiqi Gao, Liwei Wang, and Jason D Lee. Sanity-Checking Pruning Methods: Random Tickets can Win the Jackpot. In *Advances in Neural Information Processing Systems*, volume 33, pages 20390–20401. Curran Associates, Inc.

[53] Hidenori Tanaka, Daniel Kunin, Daniel L Yamins, and Surya Ganguli. Pruning neural networks without any data by iteratively conserving synaptic flow. In *Advances in Neural Information Processing Systems*, volume 33, pages 6377–6389. Curran Associates, Inc., 2020.

[54] Terence Tao and Van Vu. Random matrices: the circular law. *Communications in Contemporary Mathematics*, 10(02):261–307, 2008.

[55] Roman Vershynin. *High-Dimensional Probability: An Introduction with Applications in Data Science*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, 2018.

[56] Chaoqi Wang, Guodong Zhang, and Roger Grosse. Picking winning tickets before training by preserving gradient flow. In *International Conference on Learning Representations*, 2020.

[57] Eugene P. Wigner. Characteristic Vectors of Bordered Matrices with Infinite Dimensions. *The Annals of Mathematics*, 62(3):548, 1955.

[58] Hongru Yang and Zhangyang Wang. On the Neural Tangent Kernel Analysis of Randomly Pruned Wide Neural Networks. *arXiv*, 2022.

[59] Y. Yu, T. Wang, and R. J. Samworth. A useful variant of the Davis–Kahan theorem for statisticians. *Biometrika*, 102(2):315–323, 2015.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes]
   (c) Did you discuss any potential negative societal impacts of your work? [N/A]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes]
   (b) Did you include complete proofs of all theoretical results? [Yes]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes]
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [No]

(d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [No]
   (c) Did you include any new assets either in the supplemental material or as a URL? [No]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [No]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]