Neural Spectrum Alignment: Empirical Study

Dmitry Kopitkov[1] and
Vadim Indelman[2]

[1] Technion Autonomous Systems Program (TASP), Technion - Israel Institute of Technology, Haifa 32000, Israel
[2] Department of Aerospace Engineering, Technion - Israel Institute of Technology, Haifa 32000, Israel
{dimkak,vadim.indelman}@technion.ac.il

Appendix

This document provides supplementary material to the main paper [8]. Therefore, it should not be considered a self-contained document, but instead regarded as its appendix.

Appendices A-F present derivations and proofs required by the main paper. Further, in Appendix G the technical details of Fourier Transform calculation are described.

In Appendices H-N we place more information about the spectrum of FC-NN-L6-W256 - trained fully-connected NN with 6 layers, each containing 256 neurons. Each shown vector $\bar{f}_t$ was interpolated from training points $\{X^i\}_{i=1}^N$ to entire $[0,1]^2$ via a linear interpolation.

Further, in Appendix O we present spectrum information for FC-NN-L4-W256, in Appendix P - for FC-NN-L2-W256, and in Appendix Q - for FC-NN-L2-W66000. Observe that FC-NN-L6-W256 and FC-NN-L2-W66000 have roughly the same number of parameters, hence these two models demonstrate the depth effect on the expressiveness of *gradient similarity* kernel $g_t(\cdot,\cdot)$.

Next, in Appendix R the spectrum of FC-NN-L6-W256-shortcuts is presented where we additionally add residual (skip) connections between different NN layers [5]. This architecture is also compared with FC-NN-L6-W256 and FC-NN-L4-W256. Further, in Appendix S we evaluate FC-NN-L6-W256 with Swish activation function [10] and Adam optimization [7]. These experiments show that NN spectrum alignment phenomena is not limited only to simple NNs with Leaky-Rely and GD optimization, but actually is ubiquitous for various NN architectures and optimizers,

In Appendix T experiments are performed for noise contrastive estimation (NCE) approach [11,4]. The applied NN architecture is FC-NN-L6-W256, and optimization is performed via stochastic GD (SGD) over training dataset of size 10^5. The spectrum of NN and its alignment towards the target function is observed again, similarly to L2 regression experiments.

Lastly, in Appendices U-W experiments are performed for MNIST, CIFAR100 and UCI Buzz datasets, applying L2 loss and GD/SGD on FC network. These

experiments provide the insights on NN alignment for high-dimensional real-world data. Here again, the alignment towards the target function is very distinct, similarly to Mona Lisa experiments in the main paper.

A Relation between spectrums of $g_t(X, X')$ and its Gramian G_t

Consider N dataset points $\boldsymbol{\mathcal{X}} = \{X^i \in \mathbb{R}^d\}_{i=1}^N$ sampled from an arbitrary probability density function (pdf) $P(X)$. Further, consider a kernel $g_t(X, X')$ and the corresponding Gramian G_t defined on $\boldsymbol{\mathcal{X}}$, with $G_t(i,j) = g_t(X^i, X^j)$. Eigenvalues $\{\tilde{\lambda}_k\}_k$, sorted in decreasing order, and eigenvectors $\{\tilde{v}_k(\cdot)\}_k$ of $g_t(\cdot,\cdot)$ w.r.t. $P(X)$ are defined as solutions of:

$$\tilde{\lambda}_k \cdot \tilde{v}_k(X) = \int g_t(X, X') \cdot \tilde{v}_k(X') \cdot P(X) dX'. \tag{A1}$$

The integral in Eq. (A1) can be approximated via a sampled approximation:

$$\int g_t(X, X') \cdot \tilde{v}_k(X') \cdot P(X) dX' \approx \frac{1}{N} \sum_{i=1}^{N} g_t(X, X^i) \cdot \tilde{v}_k(X^i), \tag{A2}$$

with the LHS of the above expression converging to the RHS as $N \to \infty$ due to the law of large numbers.

Further, denote by $\bar{v}_k$ a $N \times 1$ vector whose i-th entry is $\tilde{v}_k(X^i)$. Combining Eq. (A1) and Eq. (A2), $\bar{v}_k$ can be written as:

$$\tilde{\lambda}_k \cdot \bar{v}_k = \frac{1}{N} G_t \cdot \bar{v}_k, \tag{A3}$$

where we can see $\bar{v}_k$ to be eigenvector of G_t. Therefore, eigenvectors $\{\bar{v}_k\}_k$ of G_t can be considered as unbiased estimations of eigenfunctions $\{\tilde{v}_k(\cdot)\}_k$ at points in $\boldsymbol{\mathcal{X}}$. Note that the above sampled approximations are expected to be less accurate for larger indexes k since the corresponding $\tilde{v}_k(\cdot)$ will contain more high-frequency oscillations.

Furthermore, from Eq. (A3) it is clear that each $\bar{v}_k$ is associated with the eigenvalue $\lambda_k = N \cdot \tilde{\lambda}_k$ of G_t. Hence, eigenvalues $\{\lambda_k\}_k$ of G_t can be considered as unbiased estimations of eigenfunctions $\{\tilde{\lambda}_k\}_k$, up to a multiplier N.

Likewise, $\tilde{v}_k(X)$ at an arbitrary point X can be estimated in a similar way, by combining Eq. (A1) and Eq. (A2):

$$\tilde{\lambda}_k \cdot \tilde{v}_k(X) \approx \frac{1}{N} \sum_{i=1}^{N} g_t(X, X^i) \cdot \tilde{v}_k(X^i) \quad \Longrightarrow \quad \lambda_k \cdot \tilde{v}_k(X) \approx g_t(X, \boldsymbol{\mathcal{X}}) \cdot \bar{v}_k, \tag{A4}$$

where $g_t(X, \boldsymbol{\mathcal{X}})$ is a row vector with $g_t(X, \boldsymbol{\mathcal{X}})_{(i)} = g_t(X, X^i)$. The above approximation is used in the Appendix E to derive NN dynamics at testing points.

B Relation between FIM and Hessian of the Loss

Hessian of a typical loss in Eq. (1) can be written as:

$$H_t \triangleq \frac{\partial^2 L(\theta_t, D)}{\partial \theta^2} = \frac{1}{N} A_t D_t A_t^T + \frac{1}{N} \sum_{i=1}^{N} \ell' \left[Y^i, f_{\theta_t}(X^i) \right] \cdot \mathcal{H}_t(X^i), \tag{A5}$$

where A_t is Jacobian matrix defined in Section 3, D_t is a diagonal matrix with $D_t(i,i) = \frac{\partial^2 \ell \left[Y^i, f_{\theta_t}(X^i) \right]}{\partial f_\theta^2}$ and $\mathcal{H}_t(X) \triangleq \frac{\partial^2 f_{\theta_t}(X)}{\partial \theta^2}$ is the model Hessian.

Further, in case of L2 loss we will have $D_t = I$ and

$$H_t = \frac{1}{N} F_t + \frac{1}{N} \sum_{i=1}^{N} \ell' \left[Y^i, f_{\theta_t}(X^i) \right] \cdot \mathcal{H}_t(X^i). \tag{A6}$$

Finally, considering final stages of the optimization, the residual $\ell' \left[Y^i, f_{\theta_t}(X^i) \right] = f_{\theta_t}(X^i) - Y^i$ is approximately zero and hence the second term of Eq. (A6) RHS can be neglected. Therefore, for L2 loss we will have $H_t \approx \frac{1}{N} F_t$.

Beyond L2 loss, a connection between FIM and the loss Hessian was also observed for the cross-entropy loss in [3]. Authors empirically observed that the loss gradient $\nabla_\theta L(\theta_t, D)$ converges very fast into a tiny subspace spanned by a few *top* eigenvectors of H_t. This suggests that *top* eigenvectors of H_t and F_t are tightly aligned and are spanning the same subspace of $\mathbb{R}^{|\theta|}$ also for cross-entropy case, as follows. Denote A_t's SVD as triplets $\{\sqrt{\lambda_i^t}, \bar{\omega}_i^t, \bar{v}_i^t\}_{i=1}^{N'}$ of ordered singular values, left and right singular vectors respectively, where N' is a number of non-zero singular values. Then, $\nabla_\theta L(\theta_t, D)$ can be written as:

$$\nabla_\theta L(\theta_t, D) = \frac{1}{N} A_t \cdot \bar{m}_t = \frac{1}{N} \left[\sum_{i=1}^{N'} \sqrt{\lambda_i^t} \cdot \bar{\omega}_i^t \cdot (\bar{v}_i^t)^T \right] \cdot \bar{m}_t = \\ = \frac{1}{N} \sum_{i=1}^{N'} \sqrt{\lambda_i^t} < \bar{v}_i^t, \bar{m}_t > \bar{\omega}_i^t. \tag{A7}$$

Due to typical extremely fast decay of λ_i^t w.r.t. i, described along this paper, $\nabla_\theta L(\theta_t, D)$ in the above expression can be roughly seen as a linear combination of only $\{\bar{\omega}_i^t\}$ associated with several *top* $\{\lambda_i^t\}$. Noting that these are also the *top* eigenvectors of F_t, we see that $\nabla_\theta L(\theta_t, D)$ is located in top-spectrum of F_t. Further, taking into account the empirical observation from [3], we can conclude from above that *top* eigenvectors of F_t and H_t are tightly aligned.

C Movement of θ along FIM Eigenvector causes Movement of NN Output along Gramian Eigenvector

To understand the relation between FIM F_t and Gramian G_t more intuitively, here we show their dual connection in terms of how the movement along FIM

eigenvector $\bar{\omega}_i^t$ in θ-space affects the movement in the function space. Specifically, consider $\bar{f}_t$ to be a vector of NN outputs at training points at optimization time t, similarly to the formulation in Section 2. Further, consider a movement of the model in θ-space from current θ_t to a new location $\theta_{t'} = \theta_t + \sqrt{\lambda_i^t} \cdot \bar{\omega}_i^t$ in direction $\bar{\omega}_i^t$ where $\sqrt{\lambda_i^t}$ is used as a step size. Then the $\bar{f}_{t'}$ at the new location can be approximated via first-order Taylor as:

$$\bar{f}_{t'} = \bar{f}_t + \sqrt{\lambda_i^t} \cdot A_t^T \cdot \bar{\omega}_i^t, \tag{A8}$$

where A_t is Jacobian matrix defined in Section 3. Moreover, considering the singular value decomposition (SVD) of A_t, we can see that $\bar{f}_{t'} - \bar{f}_t = \lambda_i^t \cdot \bar{v}_i^t$. That is, walking in the direction $\bar{\omega}_i^t$ in θ-space changes NN outputs only along $\bar{v}_i^t$, according to first-order dynamics.

D Dynamics of L2 Loss for a Fixed Gramian, at Training Points

Consider Eq. (3) with a fixed Gramian G whose eigenvalues and eigenvectors are $\{\lambda_i\}_{i=1}^N$ and $\{\bar{v}_i\}_{i=1}^N$ respectively. Define N' to be a number of non-zero eigenvalues. Likewise, consider the residual vector $\bar{m}_t = \bar{f}_t - \bar{y}$ whose first-order dynamics can be written as:

$$\begin{aligned} d\bar{m}_t \triangleq \bar{m}_{t+1} - \bar{m}_t = \bar{f}_{t+1} - \bar{f}_t = d\bar{f}_t = -\frac{\delta}{N} \cdot G \cdot \bar{m}_t \quad \Longrightarrow \\ \Longrightarrow \quad \bar{m}_{t+1} = \left[I - \frac{\delta}{N} \cdot G \right] \cdot \bar{m}_t \quad \Longrightarrow \\ \Longrightarrow \quad \bar{m}_t = \sum_{i=1}^{N'} \left[1 - \frac{\delta}{N} \lambda_i \right]^t < \bar{v}_i, \bar{m}_0 > \bar{v}_i + \bar{m}_0^z, \end{aligned} \tag{A9}$$

where $\bar{m}_0^z$ is a projection of $\bar{m}_0$ to null-space of G, with $G \cdot \bar{m}_0^z = \bar{0}$.

Further, noting that:

$$\sum_{j=0}^{t-1} \bar{m}_j = \sum_{i=1}^{N'} \frac{1 - \left[1 - \frac{\delta}{N} \lambda_i\right]^t}{\frac{\delta}{N} \lambda_i} < \bar{v}_i, \bar{m}_0 > \bar{v}_i + t \bar{m}_0^z, \tag{A10}$$

the $\bar{f}_t$ can be then rewritten as:

$$\begin{aligned} \bar{f}_t = \bar{f}_0 + \sum_{j=0}^{t-1} d\bar{f}_j = \bar{f}_0 - \frac{\delta}{N} G \cdot \sum_{j=0}^{t-1} \bar{m}_j = \\ = \bar{f}_0 - \frac{\delta}{N} G \cdot \sum_{i=1}^{N'} \frac{1 - \left[1 - \frac{\delta}{N} \lambda_i\right]^t}{\frac{\delta}{N} \lambda_i} < \bar{v}_i, \bar{m}_0 > \bar{v}_i = \\ = \bar{f}_0 - \sum_{i=1}^{N'} \left[1 - \left[1 - \frac{\delta}{N} \lambda_i \right]^t \right] < \bar{v}_i, \bar{m}_0 > \bar{v}_i. \end{aligned} \tag{A11}$$

E Dynamics of L2 Loss for a Fixed Gramian, at Testing Points

From Eq. (2) we can also derive dynamics of NN output at an arbitrary testing point X':

$$df_{\theta_t}(X') = f_{\theta_{t+1}}(X') - f_{\theta_t}(X') = -\frac{\delta}{N} g(X', \boldsymbol{\mathcal{X}}) \cdot \bar{m}_t, \tag{A12}$$

where $g(X', \boldsymbol{\mathcal{X}}) \triangleq \nabla_\theta f_{\theta_t}(X')^T \cdot A_t$ is a row vector with $g(X', \boldsymbol{\mathcal{X}})_{(j)} = g(X', X^j)$. Moreover, similarly to Eq. (A11) we get:

$$\begin{aligned} f_{\theta_t}(X') &= f_{\theta_0}(X') + \sum_{j=0}^{t-1} df_{\theta_j}(X') = f_{\theta_0}(X') - \frac{\delta}{N} g(X', \boldsymbol{\mathcal{X}}) \cdot \sum_{j=0}^{t-1} \bar{m}_j = \\ &= f_{\theta_0}(X') - \frac{\delta}{N} g(X', \boldsymbol{\mathcal{X}}) \cdot \left[\sum_{i=1}^{N'} \frac{1 - \left[1 - \frac{\delta}{N}\lambda_i\right]^t}{\frac{\delta}{N}\lambda_i} < \bar{v}_i, \bar{m}_0 > \bar{v}_i + t\bar{m}_0^z \right]. \end{aligned} \tag{A13}$$

In case G is invertible (i.e. $\lambda_{min} > 0$), the above expression can also be written as $f_{\theta_t}(X') = f_{\theta_0}(X') - g(X', \boldsymbol{\mathcal{X}}) \cdot G^{-1} \cdot \left[I - \left[I - \frac{\delta}{N} \cdot G \right]^t \right] \cdot \bar{m}_0$; a very similar expression was previously derived in [9]. Likewise, considering the stability condition $\delta < \frac{2N}{\lambda_{max}}$, which is required for a proper optimization convergence $\lim_{t\to\infty} \left[1 - \frac{\delta}{N}\lambda_i\right]^t = 0$, at time $t = \infty$ we will have $f_{\theta_\infty}(X') = f_{\theta_0}(X') - g(X', \boldsymbol{\mathcal{X}}) \cdot G^{-1} \cdot \bar{m}_0$.

Furthermore, for a singular G Eq. (A13) can be simplified via two methods, using a gradient at X' or eigenfunctions of the kernel $g(\cdot, \cdot)$.

Simplification via Gradient Observe that for $G = A_t^T \cdot A_t$ to be time-invariant it is necessary for gradients $\{\nabla_\theta f_{\theta_t}(X^i)\}_{i=1}^N$ at training points either to be constant along the optimization or rotating together via some time-variant rotation matrix R_t, $\nabla_\theta f_{\theta_t}(X^i) = R_t \cdot \nabla_\theta f_{\theta_0}(X^i)$ and $A_t = R_t \cdot A_0$. Such rotational behavior will lead to the required time-independence of $G = A_0^T \cdot R_t^T \cdot R_t \cdot A_0 = A_0^T \cdot A_0$. Similarly, for $g(X', \boldsymbol{\mathcal{X}})$ to be time-invariant the gradient $\nabla_\theta f_{\theta_t}(X')$ at the testing point must rotate with the same rotation R_t, $\nabla_\theta f_{\theta_t}(X') = R_t \cdot \nabla_\theta f_{\theta_0}(X')$.

Assuming the above gradient rotation, the row vector $g(X', \boldsymbol{\mathcal{X}})$ can be written as:

$$g(X', \boldsymbol{\mathcal{X}}) = \nabla_\theta f_{\theta_t}(X')^T \cdot A_t = \nabla_\theta f_{\theta_0}(X')^T \cdot R_t^T \cdot R_t \cdot A_0 = \nabla_\theta f_{\theta_0}(X')^T \cdot A_0. \tag{A14}$$

Next, consider A_0's SVD as triplets $\{\sqrt{\lambda_i}, \bar{\omega}_i, \bar{v}_i\}_{i=1}^{N'}$ of ordered singular values, left and right singular vectors respectively, and denote $\nabla_\theta f_{\theta_0}(X') = \sum_{i=1}^{N'} a_i \cdot \sqrt{\lambda_i} \cdot \bar{\omega}_i$ for $a_i \triangleq \frac{<\bar{\omega}_i, \nabla_\theta f_{\theta_0}(X')>}{\sqrt{\lambda_i}}$. Using SVD properties of A_0, we get an identity $g(X', \boldsymbol{\mathcal{X}}) = \sum_{i=1}^{N'} a_i \cdot \lambda_i \cdot \bar{v}_i^T$, and we can rewrite $f_{\theta_t}(X')$ from Eq. (A13) as (note

that $\bar{m}_0^z$ is reduced since it is orthogonal to $\{\bar{v}_i : \lambda_i \neq 0\}$):

$$f_{\theta_t}(X') = f_{\theta_0}(X') - \sum_{i=1}^{N'} \left[1 - \left[1 - \frac{\delta}{N}\lambda_i\right]^t\right] a_i < \bar{v}_i, \bar{m}_0 >= \\ = f_{\theta_0}(X') - \sum_{i=1}^{N'} \left[1 - \left[1 - \frac{\delta}{N}\lambda_i\right]^t\right] \frac{1}{\sqrt{\lambda_i}} < \bar{v}_i, \bar{m}_0 >< \bar{\omega}_i, \nabla_\theta f_{\theta_0}(X') > . \tag{A15}$$

Likewise, under the stability condition $\delta < \frac{2N}{\lambda_{max}}$, $f_{\theta_t}(X')$ at time $t = \infty$ can be expressed as:

$$f_{\theta_\infty}(X') = f_{\theta_0}(X') - \sum_{i=1}^{N'} \frac{1}{\sqrt{\lambda_i}} < \bar{v}_i, \bar{m}_0 >< \bar{\omega}_i, \nabla_\theta f_{\theta_0}(X') > . \tag{A16}$$

Simplification via Kernel Eigenfunctions According to Eq. (A4), a product $g(X', \boldsymbol{\mathcal{X}}) \cdot \bar{v}_i$ can be approximated by $\lambda_i \cdot \tilde{v}_i(X')$, with $\tilde{v}_i(\cdot)$ being an eigenfunction of $g(\cdot, \cdot)$. Using this approximation, Eq. (A13) is reduced to:

$$f_{\theta_t}(X') \approx f_{\theta_0}(X') - \frac{\delta}{N} \cdot \left[\sum_{i=1}^{N'} \frac{1 - \left[1 - \frac{\delta}{N}\lambda_i\right]^t}{\frac{\delta}{N}} < \bar{v}_i, \bar{m}_0 > \tilde{v}_i(X') + \right. \\ \left. + t \cdot \sum_{i:\lambda_i=0} \lambda_i < \bar{v}_i, \bar{m}_0 > \tilde{v}_i(X')\right] = f_{\theta_0}(X') - \sum_{i=1}^{N'} \left[1 - \left[1 - \frac{\delta}{N}\lambda_i\right]^t\right] < \bar{v}_i, \bar{m}_0 > \tilde{v}_i(X'), \tag{A17}$$

which at time $t = \infty$ will converge to:

$$f_{\theta_\infty}(X') = f_{\theta_0}(X') - \sum_{i=1}^{N'} < \bar{v}_i, \bar{m}_0 > \tilde{v}_i(X'). \tag{A18}$$

Intuition Eq. (A15) and Eq. (A17) describe first-order dynamics of NN output at a testing point. The intuition behind these expressions can be summarized as following. First, for standard NN initialization $f_{\theta_0}(X')$ is typically very close to be zero and can be neglected, leading to $\bar{m}_0 \approx -\bar{y}$. Like in Eq. (A11), the inner-product term $< \bar{v}_i, \bar{m}_0 >$, independent of testing point X', defines which part of the signal contained in $\bar{m}_0$ is learned along each spectral direction. In general, $\left[1 - \frac{\delta}{N}\lambda_i\right]^t$ converges faster for large eigenvalues. Also, due to large λ_i being typically associated with $\bar{v}_i$ that contains a low-frequency signal, this leads to fast learning of low-frequency information and slow (sometimes infinitely slow) learning of high-frequency information. Further, the inner-product term $< \bar{\omega}_i, \nabla_\theta f_{\theta_0}(X') >$ in Eq. (A15) or the eigenfunction $\tilde{v}_i(X')$ in Eq. (A17), that

are functions of X', determine amount of information along i-th spectral direction that is transferred into $f_{\theta_t}(X')$, basically describing the generalization behind Eq. (3) for a fixed Gramian G. Note that the convergence rate of $f_{\theta_t}(X')$ towards $f_{\theta_\infty}(X')$ is governed by how close terms $1-\frac{\delta}{N}\lambda_i$ in Eq. (A15) and Eq. (A17) are to zero, similarly to the convergence rate of a system in Eq. (A11). Hence, we expect f_{θ_t} to converge to its final state at both training and testing points with a similar speed.

F First-order Change of G_t

Here we describe the first-order Taylor approximation of a change in G_t between sequential iterations of GD optimization. We theorize that the thorough analysis of below expressions will lead to the mathematical explanation required to understand evolution of G_t as also to better understanding of NN dynamics.

First, change of the Jacobian A_t, defined in Section 3, can be described as:

$$dA_t \triangleq A_{t+1} - A_t \approx -\frac{\delta}{N} \cdot W_t, \tag{A19}$$

where W_t is $|\theta| \times N$ matrix with i-th column being $\mathcal{H}_t(X^i) \cdot A_t \cdot \bar{m}_t$, with $\mathcal{H}_t(X) \triangleq \frac{\partial^2 f_{\theta_t}(X)}{\partial \theta^2}$ being the model Hessian.

Hence, the change between $G_{t+1} = A_{t+1}^T \cdot A_{t+1}$ and $G_t = A_t^T \cdot A_t$ can be written as:

$$dG_t \triangleq G_{t+1} - G_t \approx -\frac{\delta}{N} \cdot \left[A_t^T \cdot W_t + W_t^T \cdot A_t\right] + \frac{\delta^2}{N^2} \cdot W_t^T \cdot W_t. \tag{A20}$$

The last term can be neglected due to $\frac{\delta^2}{N^2}$ being significantly smaller than $\frac{\delta}{N}$, which leads to:

$$dG_t \triangleq G_{t+1} - G_t \approx -\frac{\delta}{N} \cdot \left[Q_t + Q_t^T\right], \tag{A21}$$

where Q_t is $N \times N$ matrix whose i-th column is $A_t^T \cdot \mathcal{H}_t(X^i) \cdot A_t \cdot \bar{m}_t$.

Recently, similar expressions were reported by [1] (specifically, see Eq. (100-102)) and by [6].

G Computation Details of Fourier Transform

Here we provide more details on how Fourier Transform was calculated in our experiments. Consider a function $\varphi(X)$ and N dataset points $\boldsymbol{\mathcal{X}} = \{X^k \in \mathbb{R}^d\}_{k=1}^N$ sampled from an arbitrary pdf $P(X)$. Further, consider a $N \times 1$ vector $\bar{\varphi}$ with entries $\bar{\varphi}(k) = \varphi(X^k)$. Given $\bar{\varphi}$, we compute Fourier Transform $\hat{\varphi}(\xi)$ of a function $\varphi(X)$ at $\xi \in \mathbb{R}^d$ as following:

$$\hat{\varphi}(\xi) = \int \varphi(X) \cdot \exp\left[-2\pi i \cdot < \xi, X >\right] \cdot P(X) dX \approx$$

$$\approx \frac{1}{N} \sum_{k=1}^{N} \varphi(X^k) \cdot \exp\left[-2\pi i \cdot < \xi, X^k >\right] = \frac{1}{N} \bar{\varphi}^T \bar{\varepsilon}, \tag{A22}$$

where $\bar{\varepsilon}$ is a $N \times 1$ vector with entries $\bar{\varepsilon}(k) = \exp\left[-2\pi i \cdot < \xi, X^k >\right]$. Note that the above definition of Fourier Transform w.r.t. pdf $P(X)$ is identical to the common formulation without a term $P(X)$ inside, since in our experiments data distribution is $P(X) = 1$ (see "Setup" in Section 6).

In all our experiments we compute $\hat{\varphi}(\xi)$ for ξ taking values in $[-40, 40]^2$. Further, we present a frequency component $|\hat{\varphi}(\xi)|$ as an image.

To perform the above computation, we require sampled values $\bar{\varphi}$ of the analyzed function $\varphi(X)$. In case this function is the eigenfunction of *gradient similarity* kernel, the eigenvector of G_t approximates this eigenfunction' values at the training points, as is shown in the Appendix A. Hence, in this case the eigenvector of G_t serves as a vector $\bar{\varphi}$ in Eq. (A22). Likewise, the above calculation using the residual vector $\bar{m}_t$ can be considered as a Fourier Transform of a function $r(X) \triangleq f_{\theta_t}(X) - y(X)$.

H NN Spectrum Preservation

Here, we examine how stable are eigenvectors of G_t along t. For this we explore the relative energy of G_{600000}'s eigenvectors, final eigenvectors of the optimization, within spectrum of G_{20000}. Note that we compare spectrums at $t = 600000$ and $t = 20000$ to skip first several thousands of iterations since during this bootstrap period the change of G_t is highly notable.

In Figure 1 we depict $E_{20000}(\bar{v}_i^{600000}, k)$ as a function of k, for various $\{\bar{v}_i^{600000}\}$. As observed, 10 first *top* eigenvectors of G_{600000} are also located in the *top* spectrum of G_{20000} - the function $E_{20000}(\bar{v}_i^{600000}, k)$ is almost 1 for even relatively small k. Hence, the *top* Gramian spectrum was preserved, roughly, along the performed optimization. Further, eigenvectors of smaller eigenvalues (i.e. with higher indexes i) are significantly less stable, with large amount of their energy widely spread inside *bottom* eigenvectors of G_{20000}. Moreover, we can see a clear trend that with higher i the associated eigenvector is less preserved.

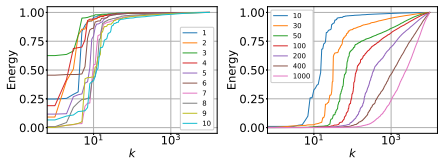


Fig. 1: For different i, relative energy of $\bar{v}_i^{600000}$ in spectrum of G_{20000}, $E_{20000}(\bar{v}_i^{600000}, k)$, as a function of k, with horizontal axes being log-scaled. As seen, 10 first *top* eigenvectors at final time $t = 600000$ are located also in the *top* spectrum of G_{20000}, hence the *top* Gramian spectrum was preserved along the optimization. Yet, *bottom* eigenvectors are significantly less stable.

I NN Spectrum at time $t = 0$

Here we show the NN state before the optimization - both its output $\bar{f}_0$ and Gramian G_0.

Fig. 2: Interpolation of $\bar{f}_0$.

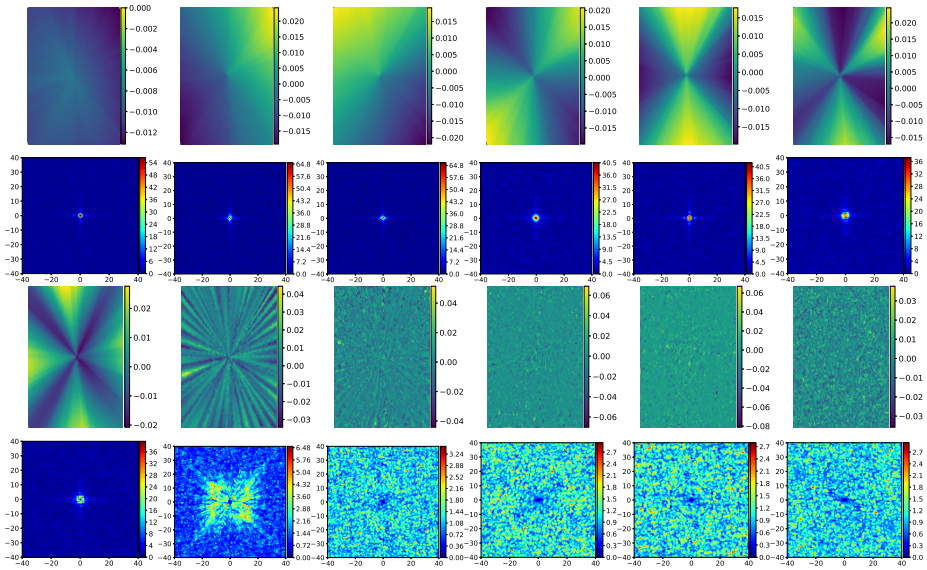


Fig. 3: First two rows: from left-to-right, 6 first eigenvectors of G_0 and their Fourier Transforms. Last two rows: from left-to-right, 10-th, 100-th, 500-th, 1000-th, 2000-th and 4000-th eigenvectors of G_0 and their Fourier Transforms.

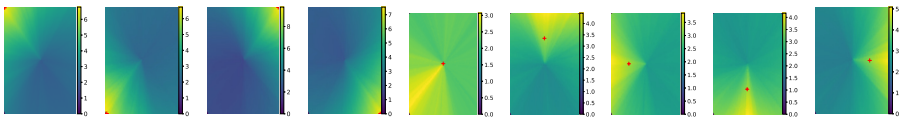


Fig. 4: *Gradient similarity* kernel $g_t(X, \cdot)$ for various points within the input domain $[0, 1]^2$ at $t = 0$. In each plot we draw $g_t(X, \cdot)$ between specific X, marked by red "+", and rest of the points. As observed, the kernel $g_t(X, X')$ has a local behavior, with its outputs being higher for nearby points X and X'.

J NN Spectrum at time $t = 1$

Here we show the NN state after a single optimization iteration - both its output $\bar{f}_1$ and Gramian G_1.

Fig. 5: Interpolation of $\bar{f}_1$.

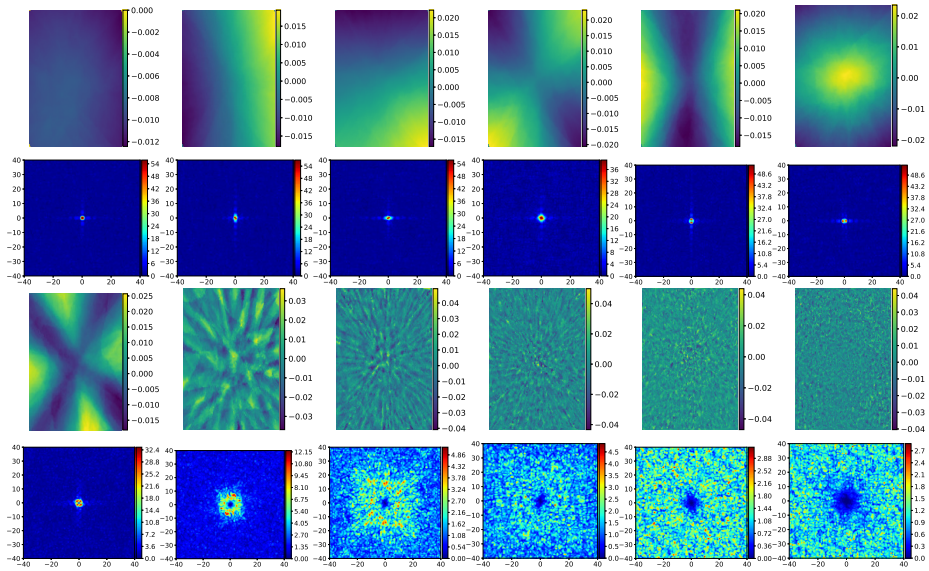


Fig. 6: First two rows: from left-to-right, 6 first eigenvectors of G_1 and their Fourier Transforms. Last two rows: from left-to-right, 10-th, 100-th, 500-th, 1000-th, 2000-th and 4000-th eigenvectors of G_1 and their Fourier Transforms.

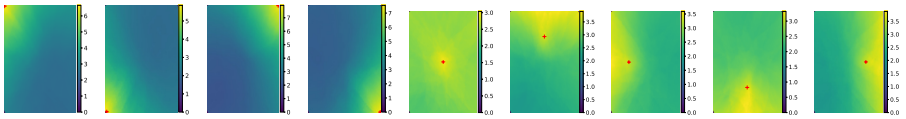


Fig. 7: *Gradient similarity* kernel $g_t(X, \cdot)$ for various points within the input domain $[0, 1]^2$ at $t = 1$. In each plot we draw $g_t(X, \cdot)$ between specific X, marked by red "+", and rest of the points. As observed, the kernel $g_t(X, X')$ has a local behavior, with its outputs being higher for nearby points X and X'.

K NN Spectrum at time $t = 100$

Fig. 8: Interpolation of $\bar{f}_{100}$.

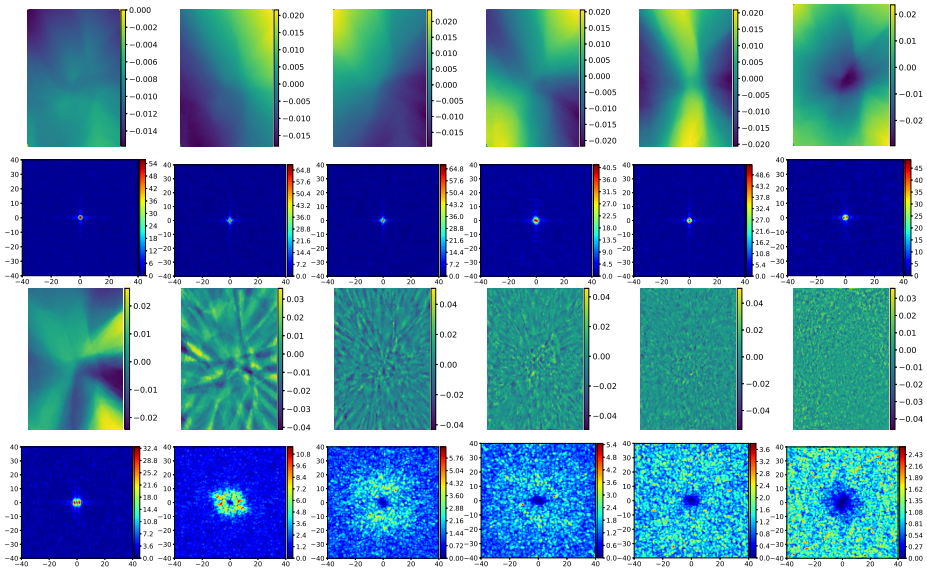


Fig. 9: First two rows: from left-to-right, 6 first eigenvectors of G_{100} and their Fourier Transforms. Last two rows: from left-to-right, 10-th, 100-th, 500-th, 1000-th, 2000-th and 4000-th eigenvectors of G_{100} and their Fourier Transforms.

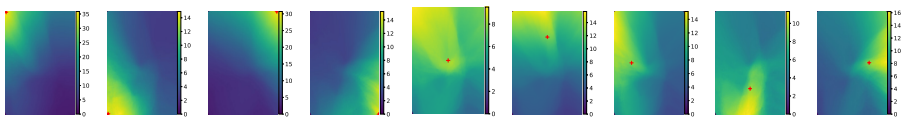


Fig. 10: *Gradient similarity* kernel $g_t(X, \cdot)$ for various points within the input domain $[0, 1]^2$ at $t = 100$. In each plot we draw $g_t(X, \cdot)$ between specific X, marked by red "+", and rest of the points. As observed, the kernel $g_t(X, X')$ has a local behavior, with its outputs being higher for nearby points X and X'.

L NN Spectrum at time $t = 5000$

Fig. 11: Interpolation of $\bar{f}_{5000}$.

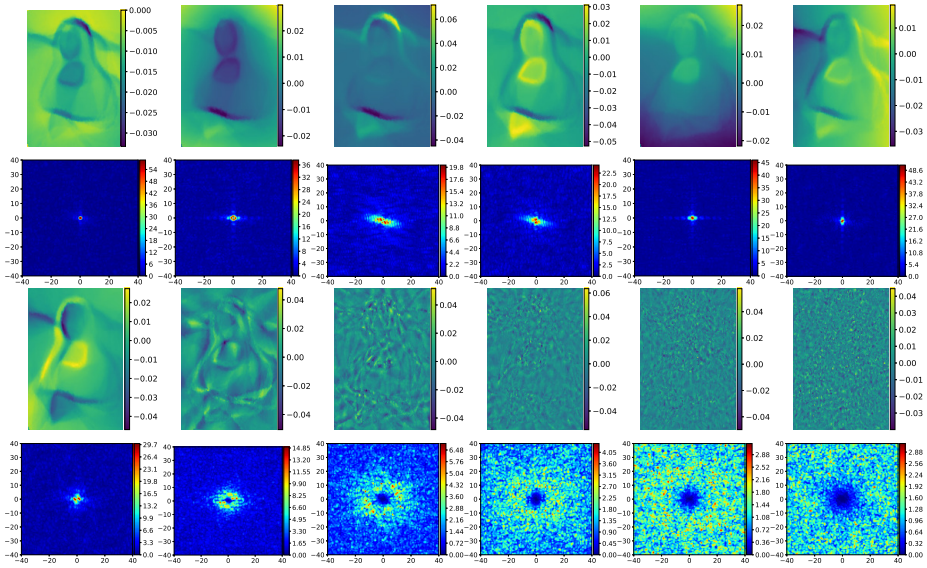


Fig. 12: First two rows: from left-to-right, 6 first eigenvectors of G_{5000} and their Fourier Transforms. Last two rows: from left-to-right, 10-th, 100-th, 500-th, 1000-th, 2000-th and 4000-th eigenvectors of G_{5000} and their Fourier Transforms.

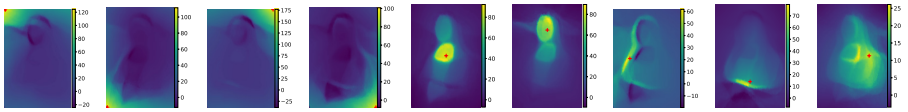


Fig. 13: *Gradient similarity* kernel $g_t(X, \cdot)$ for various points within the input domain $[0, 1]^2$ at $t = 5000$. In each plot we draw $g_t(X, \cdot)$ between specific X, marked by red "+", and rest of the points. As observed, the kernel $g_t(X, X')$ has a local behavior, with its outputs being higher for nearby points X and X'.

M NN Spectrum at time $t = 50000$

Fig. 14: Interpolation of $\bar{f}_{50000}$.

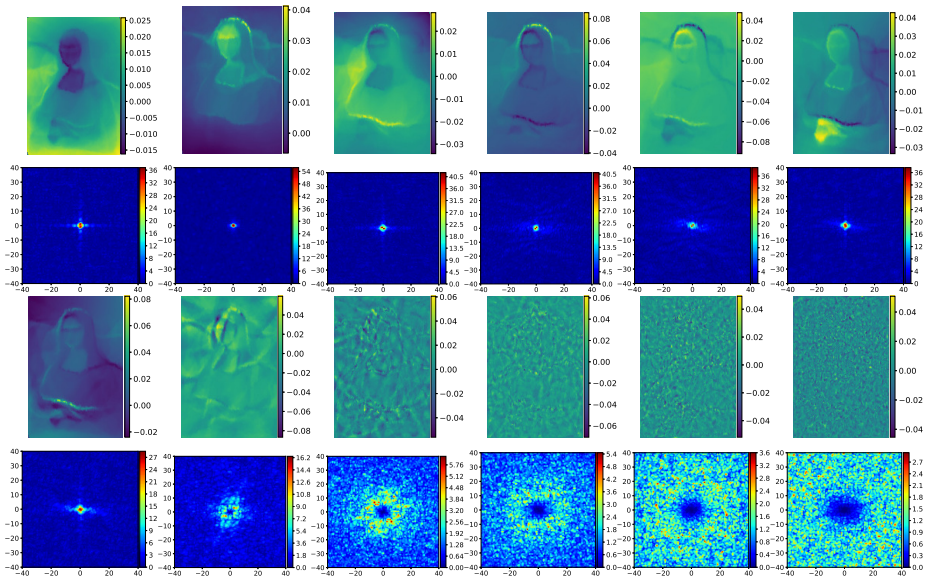


Fig. 15: First two rows: from left-to-right, 6 first eigenvectors of G_{50000} and their Fourier Transforms. Last two rows: from left-to-right, 10-th, 100-th, 500-th, 1000-th, 2000-th and 4000-th eigenvectors of G_{50000} and their Fourier Transforms.

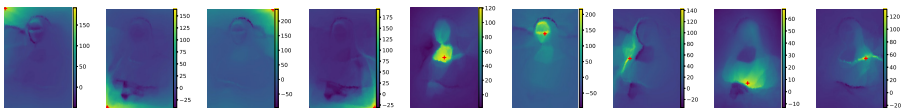


Fig. 16: *Gradient similarity* kernel $g_t(X, \cdot)$ for various points within the input domain $[0, 1]^2$ at $t = 50000$. In each plot we draw $g_t(X, \cdot)$ between specific X, marked by red "+", and rest of the points. As observed, the kernel $g_t(X, X')$ has a local behavior, with its outputs being higher for nearby points X and X'.

N NN Spectrum at time $t = 600000$

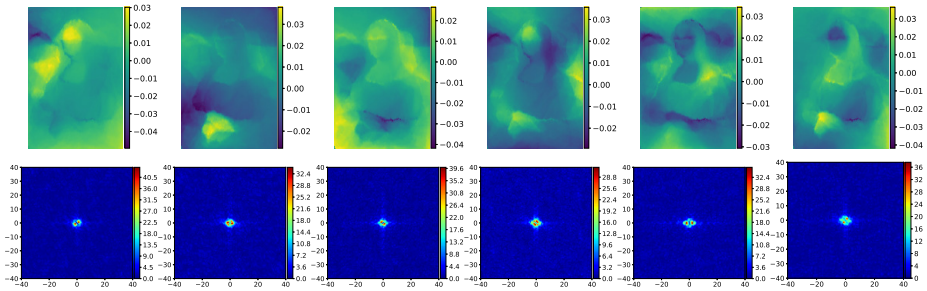


Fig. 17: From left-to-right, 7-th, 8-th, 9-th, 11-th, 12-th and 13-th eigenvector of Gramian G_t at $t = 600000$, and their Fourier Transforms.

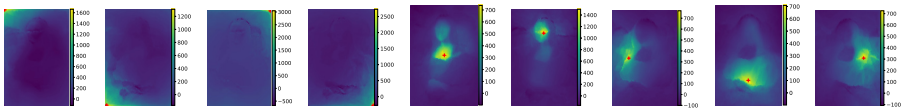


Fig. 18: *Gradient similarity* kernel $g_t(X, \cdot)$ for various points within the input domain $[0, 1]^2$ at $t = 600000$. In each plot we draw $g_t(X, \cdot)$ between specific X, marked by red "+", and rest of the points. As observed, the kernel $g_t(X, X')$ has a local behavior, with its outputs being higher for nearby points X and X'.

O 4-Layer 256-Wide NN Spectrum at time $t = 600000$

Fig. 19: Interpolation of $\bar{f}_{600000}$.

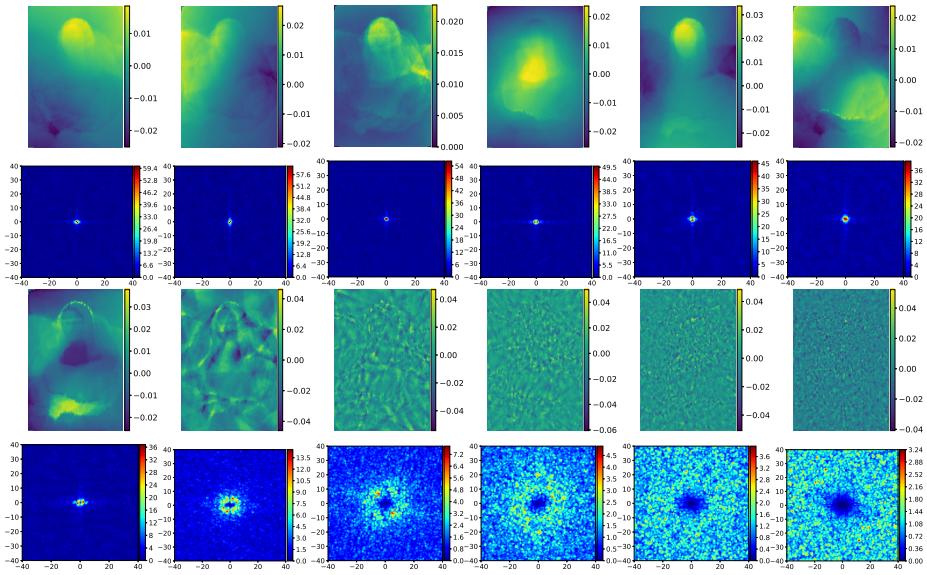


Fig. 20: First two rows: from left-to-right, 6 first eigenvectors of G_{600000} and their Fourier Transforms. Last two rows: from left-to-right, 10-th, 100-th, 500-th, 1000-th, 2000-th and 4000-th eigenvectors of G_{600000} and their Fourier Transforms.

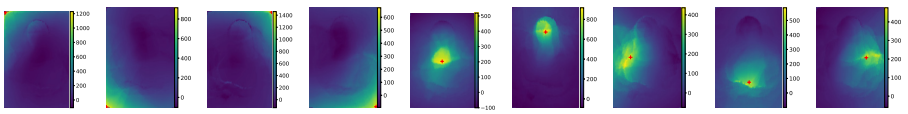


Fig. 21: *Gradient similarity* kernel $g_t(X, \cdot)$ for various points within the input domain $[0, 1]^2$ at $t = 600000$. In each plot we draw $g_t(X, \cdot)$ between specific X, marked by red "+", and rest of the points. As observed, the kernel $g_t(X, X')$ has a local behavior, with its outputs being higher for nearby points X and X'.

P 2-Layer 256-Wide NN Spectrum at time $t = 600000$

Fig. 22: Interpolation of $\bar{f}_{600000}$.

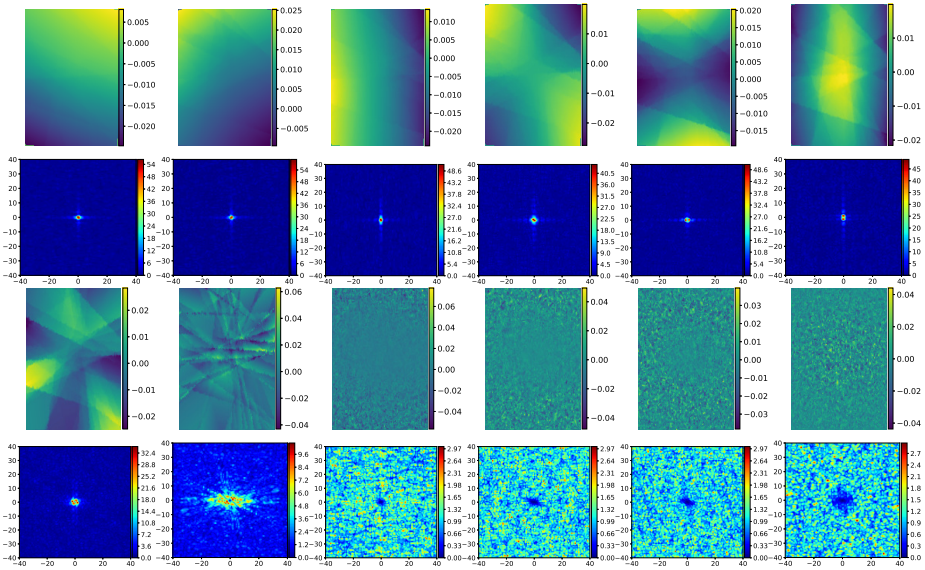


Fig. 23: First two rows: from left-to-right, 6 first eigenvectors of G_{600000} and their Fourier Transforms. Last two rows: from left-to-right, 10-th, 100-th, 500-th, 1000-th, 2000-th and 4000-th eigenvectors of G_{600000} and their Fourier Transforms.

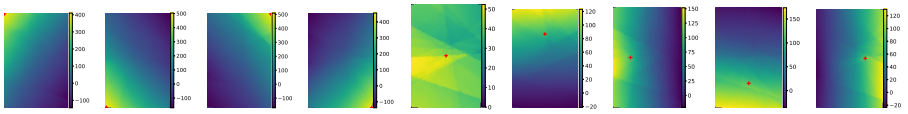


Fig. 24: *Gradient similarity* kernel $g_t(X, \cdot)$ for various points within the input domain $[0, 1]^2$ at $t = 600000$. In each plot we draw $g_t(X, \cdot)$ between specific X, marked by red "+", and rest of the points. As observed, the kernel $g_t(X, X')$ has a local behavior, with its outputs being higher for nearby points X and X'.

Q 2-Layer 66000-Wide NN Spectrum at time $t = 600000$

Fig. 25: Interpolation of $\bar{f}_{600000}$.

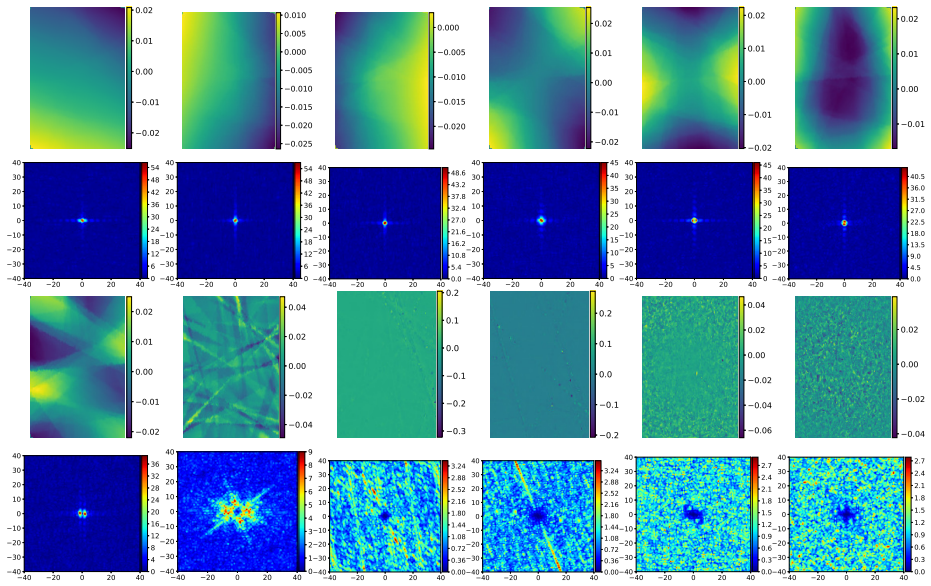


Fig. 26: First two rows: from left-to-right, 6 first eigenvectors of G_{600000} and their Fourier Transforms. Last two rows: from left-to-right, 10-th, 100-th, 500-th, 1000-th, 2000-th and 4000-th eigenvectors of G_{600000} and their Fourier Transforms.

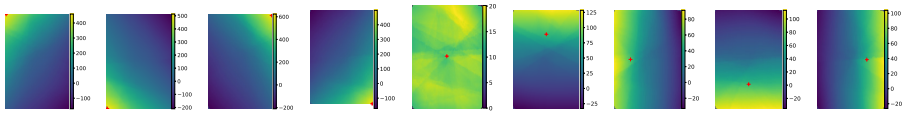


Fig. 27: *Gradient similarity* kernel $g_t(X, \cdot)$ for various points within the input domain $[0, 1]^2$ at $t = 600000$. In each plot we draw $g_t(X, \cdot)$ between specific X, marked by red "+", and rest of the points. As observed, the kernel $g_t(X, X')$ has a local behavior, with its outputs being higher for nearby points X and X'.

R 6-Layer 256-Wide NN with shortcuts, Spectrum at time $t = 600000$

Fig. 28: Interpolation of $\bar{f}_{600000}$.

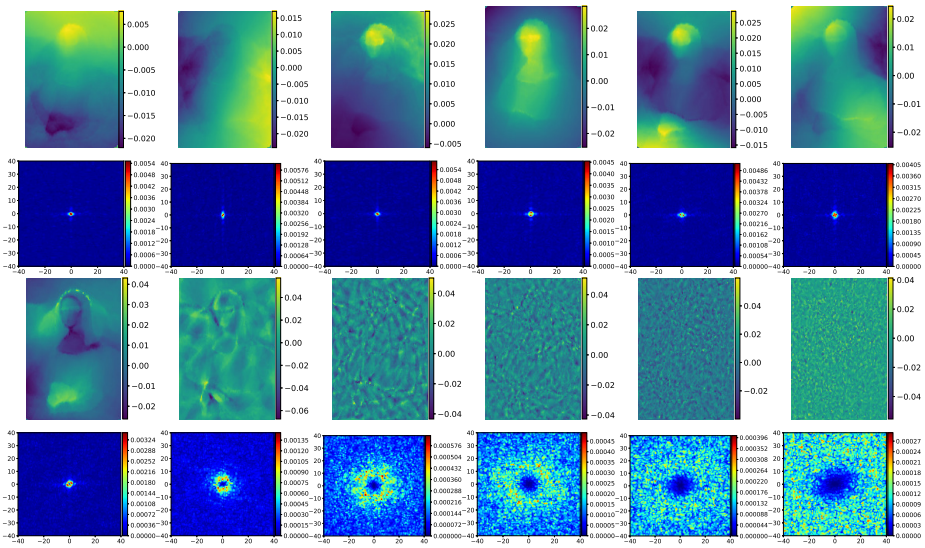


Fig. 29: First two rows: from left-to-right, 6 first eigenvectors of G_{600000} and their Fourier Transforms. Last two rows: from left-to-right, 10-th, 100-th, 500-th, 1000-th, 2000-th and 4000-th eigenvectors of G_{600000} and their Fourier Transforms.

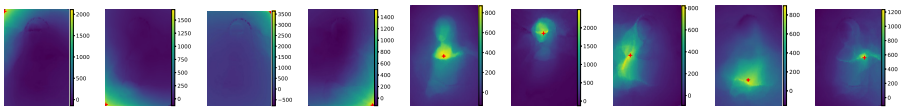


Fig. 30: *Gradient similarity* kernel $g_t(X, \cdot)$ for various points within the input domain $[0, 1]^2$ at $t = 600000$. In each plot we draw $g_t(X, \cdot)$ between specific X, marked by red "+", and rest of the points. As observed, the kernel $g_t(X, X')$ has a local behavior, with its outputs being higher for nearby points X and X'.

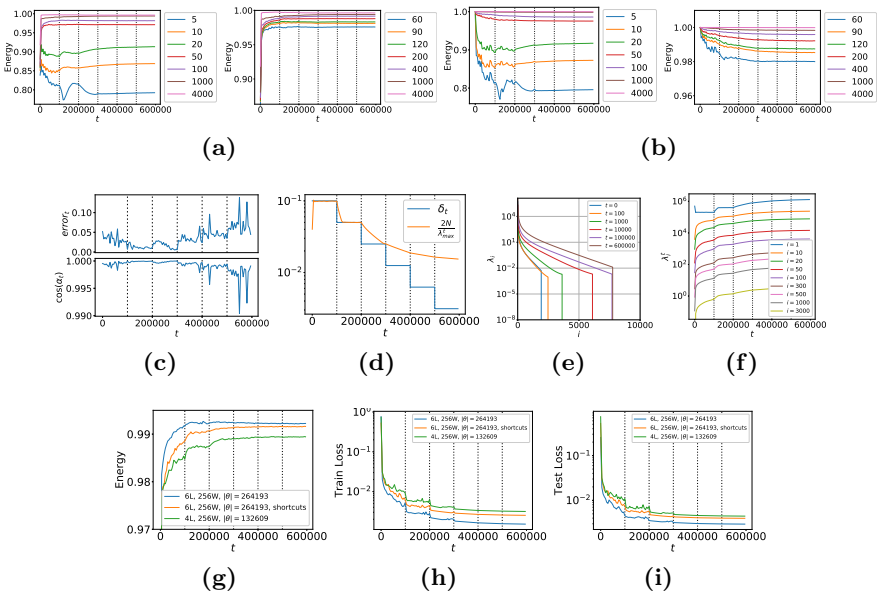


Fig. 31: (a) For different k, relative energy of the label vector $\bar{y}$ in *top* k eigenvectors of G_t, $E_t(\bar{y}, k)$, along the optimization time t. As observed, the alignment between $\bar{y}$ and *top* eigenvectors is growing along the optimization. Dashed vertical lines depict time t at which learning rate δ was decayed. (b) Relative energy of NN output, $E_t(\bar{f}_t, k)$. 99% of $\bar{f}_t$'s energy is contained within *top* 200 eigenvectors for all t. (c) Accuracy of first order NN dynamics, similar to Figure 1d in the main paper. Depicted is $error_t = \frac{\left\|d\tilde{f}_t - d\bar{f}_t\right\|}{\left\|d\tilde{f}_t\right\|}$, where $d\bar{f}_t = -\frac{\delta_t}{N} \cdot G_t \cdot \bar{m}_t$ is the first-order approximation of a real differential $d\tilde{f}_t \triangleq \bar{f}_{t+1} - \bar{f}_t$; $\cos(\alpha_t)$ is cosine of an angle between $d\tilde{f}_t$ and $d\bar{f}_t$. As observed, first-order approximation is more accurate for NN with shortcuts, especially at the training initial stage. (d) Learning rate δ_t and its upper stability boundary $\frac{2N}{\lambda^t_{max}}$ along the optimization. We empirically observe that the boundary chases the learning rate, with a relation $\lambda^t_{max} \propto \frac{1}{\delta_t}$. (e) Eigenvalues $\{\lambda^t_i\}_{i=1}^N$ for different t. Eigenvalues $\lambda_i < \lambda_{max} \cdot 10^{-8}$ are cut out. (f) Individual eigenvalues along t. As observed, eigenvalues monotonically grow along t, with growing boost at times of the learning rate drop. (g) For various NNs, relative energy of the label vector $\bar{y}$ in *top* 400 eigenvectors of G_t, $E_t(\bar{y}, 400)$, along the optimization time t; as seen the alignment of 6-layer NN with shortcuts is higher than of 4-layer NN and is lower than of 6-layer NN. (h) Training loss and (i) testing loss of these models. Similarly to the alignment, losses of 6-layer NN with shortcuts are smaller than of 4-layer NN and are bigger than of 6-layer NN. Hence, we can again observe that the alignment correlates with the optimization and generalization performances.

S 6-Layer 256-Wide NN, Swish Activation Function, Adam optimization, Spectrum at time $t = 600000$

Here, we explore the alignment dynamics beyond Leaky-Relu activation function and GD optimization. The considered NN here is identical to the one from the main paper, with Swish activation function $\sigma(x) = x \cdot sigmoid(x)$ [10]. Applied optimization is Adam [7], with initial learning rate 1e-4, $\beta_1 = 0.75$, $\beta_2 = 0.999$ and $\epsilon =$ 1e-10. Importantly, note that first order NN dynamics via kernel $g_t(X, X') \triangleq \nabla_\theta f_{\theta_t}(X)^T \cdot \nabla_\theta f_{\theta_t}(X')$ were specifically derived for GD and hence are improper for Adam. Yet, below we still can observe the alignment between $g_t(X, X')$'s eigenvectors/eigenfunctions and the target function. We speculate that the reason for this is that the implicit first-order kernel of Adam is closely linked with $g_t(X, X')$. We leave investigation of Adam's kernel for future work.

(a) **(b)**

Fig. 32: (a) NN $f_\theta(X)$ at convergence. (b) Interpolation of $\bar{f}_{600000}$.

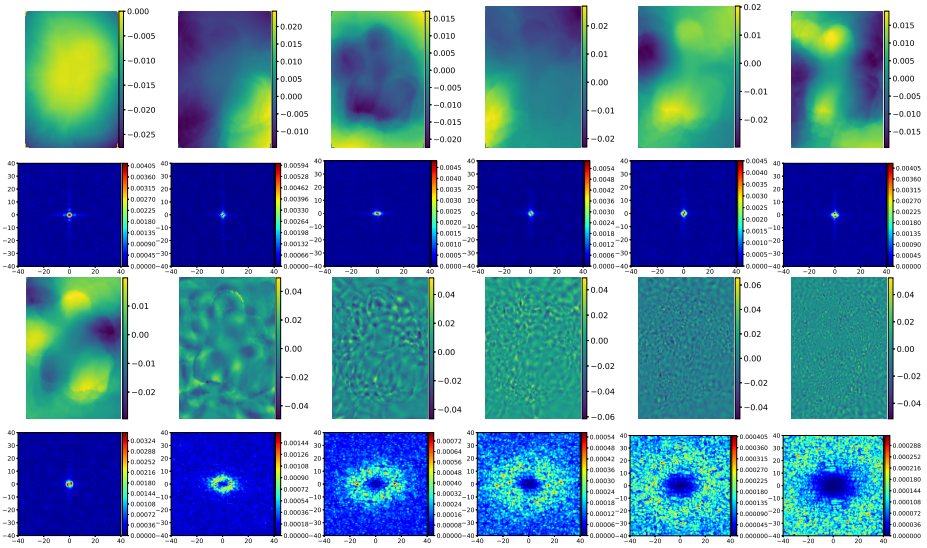


Fig. 33: First two rows: from left-to-right, 6 first eigenvectors of G_{600000} and their Fourier Transforms. Last two rows: from left-to-right, 10-th, 100-th, 500-th, 1000-th, 2000-th and 4000-th eigenvectors of G_{600000} and their Fourier Transforms.

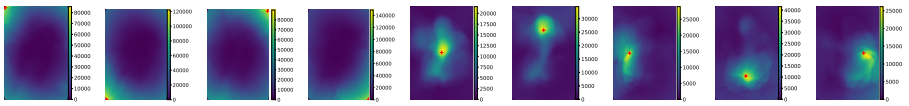


Fig. 34: *Gradient similarity* kernel $g_t(X, \cdot)$ for various points within the input domain $[0, 1]^2$ at $t = 600000$. In each plot we draw $g_t(X, \cdot)$ between specific X, marked by red "+", and rest of the points. As observed, the kernel $g_t(X, X')$ has a local behavior, with its outputs being higher for nearby points X and X'.

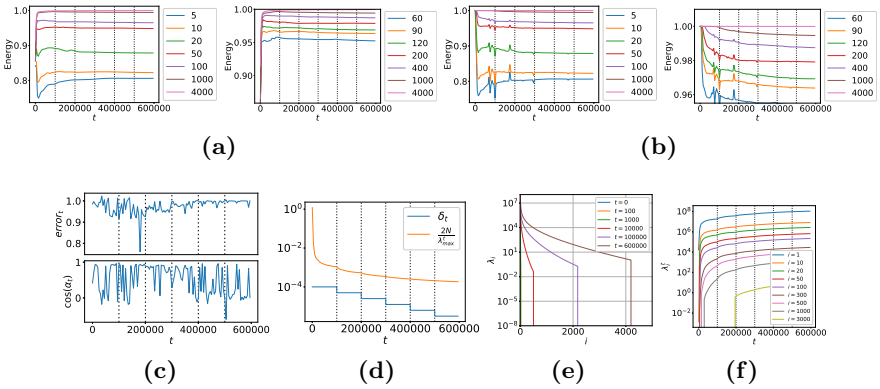


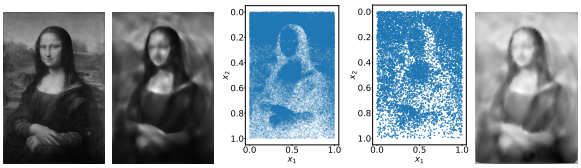
Fig. 35: (a) For different k, relative energy of the label vector $\bar{y}$ in *top* k eigenvectors of G_t, $E_t(\bar{y}, k)$, along the optimization time t. As observed, the alignment between $\bar{y}$ and *top* eigenvectors is growing along the optimization, especially during the first few thousands of iterations. Particularly, 98% of $\bar{y}$'s energy is contained within *top* 400 eigenvectors. Dashed vertical lines depict time t at which learning rate δ was decayed. (b) Relative energy of NN output, $E_t(\bar{f}_t, k)$. 98% of $\bar{f}_t$'s energy is contained within *top* 200 eigenvectors for all t. (c) Accuracy of first order NN dynamics, similar to Figure 1d in the main paper. Depicted is $error_t = \frac{\left\| d\tilde{f}_t - d\bar{f}_t \right\|}{\left\| d\tilde{f}_t \right\|}$, where $d\bar{f}_t = -\frac{\delta_t}{N} \cdot G_t \cdot \bar{m}_t$ is the first-order approximation of a real differential $d\tilde{f}_t \triangleq \bar{f}_{t+1} - \bar{f}_t$; $\cos(\alpha_t)$ is cosine of an angle between $d\tilde{f}_t$ and $d\bar{f}_t$. As expected, first-order approximation derived for GD is not valid anymore for Adam optimization. (d) Learning rate δ_t and $\frac{2N}{\lambda^t_{max}}$ along the optimization. We empirically observe that $\frac{2N}{\lambda^t_{max}}$ accelerates its decrease at learning rate drop, with a relation $\lambda^t_{max} \propto \frac{1}{\delta_t}$ similarly to GD case. (e) Eigenvalues $\{\lambda^t_i\}_{i=1}^N$ for different t. Eigenvalues $\lambda_i < \lambda_{max} \cdot 10^{-8}$ are cut out. (f) Individual eigenvalues along t. As observed, eigenvalues monotonically grow along t, with growing boost at times of the learning rate drop. Additionally, the decay of $\{\lambda^t_i\}_{i=1}^N$ w.r.t. i is much faster than for Leaky-Rely NN, suggesting that Swish NN is less expressive or that Xavier initialization [2] is not optimal for it.

T 6-Layer 256-Wide NN, Noise Contrastive Estimation loss, SGD optimization, Spectrum at time $t = 600000$

The objective of NCE approach [11,4] is to estimate probability density function (pdf) of the available dataset $\boldsymbol{\mathcal{X}} = \{X^i \in \mathbb{R}^d\}_{i=1}^N$ sampled from pdf $\mathbb{P}(X)$. The simplified version of its loss is defined as:

$$L(\theta, \boldsymbol{\mathcal{X}}, \{X_n^i \in \mathbb{R}^d\}_{i=1}^N) = \\ = \frac{1}{N}\sum_{i=1}^{N} \log \frac{\exp[f_\theta(X^i)] + \mathbb{P}_n(X^i)}{\exp[f_\theta(X^i)]} + \frac{1}{N}\sum_{i=1}^{N} \log \frac{\exp[f_\theta(X_n^i)] + \mathbb{P}_n(X_n^i)}{\mathbb{P}_n(X_n^i)}. \tag{A23}$$

Here $\mathbb{P}_n(X)$ denotes an arbitrary known "noise" pdf (e.g. Gaussian or Gaussian Mixture Model) whose support covers the support of $\mathbb{P}(X)$. Further, $\{X_n^i \in \mathbb{R}^d\}_{i=1}^N$ are N samples from $\mathbb{P}_n(X)$. At convergence $f_\theta(X)$ is unbiased estimator of $\log \mathbb{P}(X)$. See [11,4] for more information about NCE.



(a) **(b)** **(c)** **(d)** **(e)**

Fig. 36: (a) Mona Lisa target function for a pdf estimation task. (b) NN exp $f_\theta(X)$ at convergence. (c) 10^5 sampled training points. (d) 10^4 sampled testing points on which Gramian G_t was computed. (e) Interpolation of $\bar{f}_{600000}$.

Setup NCE experiment is configured similarly to the L2 regression demonstrated in the main paper. We specifically consider the target pdf $\mathbb{P}(X)$ with $X \in [0,1]^2 \subseteq \mathbb{R}^2$ which is proportional to Mona Lisa image in Figure 36a, up to a normalization constant. We approximate this pdf with Leaky-Relu FC network via NCE loss, using $N = 100000$ training points sampled from $\mathbb{P}(X)$ (see Figure 36c). $\mathbb{P}_n(X)$ is chosen to be Uniform distribution over $[0,1]^2$. Due to large training dataset we applied SGD with mini-batch size 1000 - at each iteration 1000 training points $\{X^i\}$ and 1000 "noisy" points $\{X_n^i\}$ are sampled from a priori prepared training dataset. Further, Eq. (A23) is optimized over sampled mini-batches. Learning rate δ starts at 0.003 and is decayed twice each 10^5 iterations. At convergence $f_\theta(X)$ gets very close to its target, see Figure 36b. To reduce runtime complexity, we calculated Gramian G_t, NN outputs $\bar{f}_t$ and their dynamics only for a set of 10000 testing points depicted in Figure 36c. Below the spectrum of $g_t(\cdot,\cdot)$ is visualized, showing the same trends as in case of L2 regression.

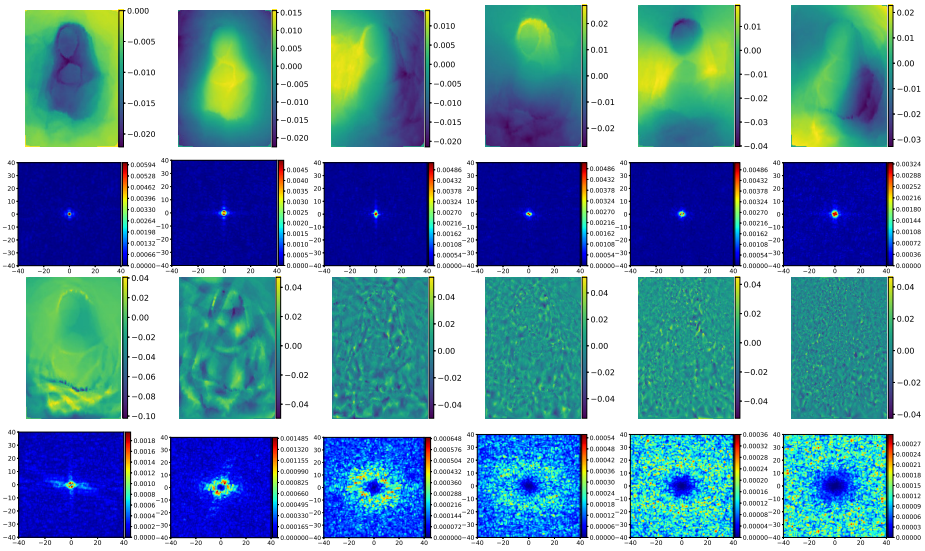


Fig. 37: First two rows: from left-to-right, 6 first eigenvectors of G_{600000} and their Fourier Transforms. Last two rows: from left-to-right, 10-th, 100-th, 500-th, 1000-th, 2000-th and 4000-th eigenvectors of G_{600000} and their Fourier Transforms.

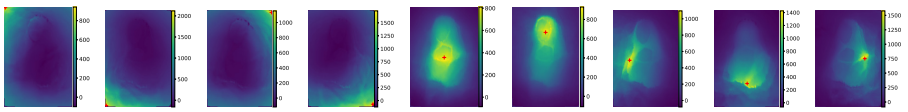


Fig. 38: *Gradient similarity* kernel $g_t(X, \cdot)$ for various points within the input domain $[0, 1]^2$ at $t = 600000$. In each plot we draw $g_t(X, \cdot)$ between specific X, marked by red "+", and rest of the points. As observed, the kernel $g_t(X, X')$ has a local behavior, with its outputs being higher for nearby points X and X'.

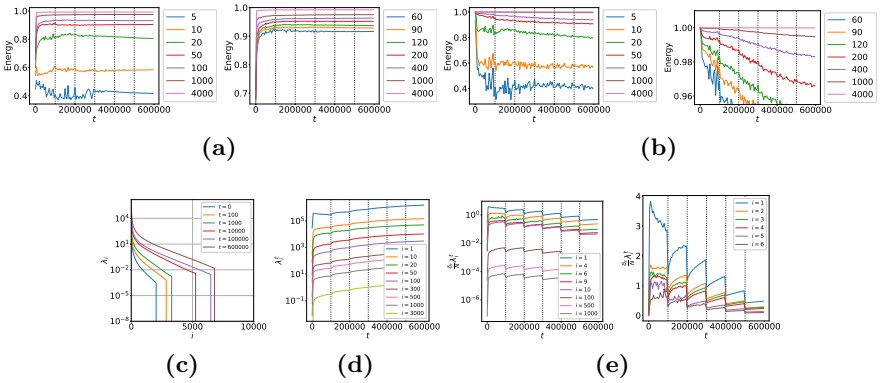


Fig. 39: (a) For different k, relative energy of the label vector $\bar{y}$ in *top* k eigenvectors of G_t, $E_t(\bar{y}, k)$, along the optimization time t. The i-th entry of $\bar{y}$ is computed as $\log \mathbb{P}(X^i)$, where X^i is i-th training point. As observed, the alignment between $\bar{y}$ and *top* eigenvectors is growing along the optimization. Dashed vertical lines depict time t at which learning rate δ was decayed. (b) Relative energy of NN output, $E_t(\bar{f}_t, k)$. 99% of $\bar{f}_t$'s energy is contained within *top* 400 eigenvectors for all t. (c) Eigenvalues $\{\lambda_i^t\}_{i=1}^N$ for different t. Eigenvalues $\lambda_i < \lambda_{max} \cdot 10^{-8}$ are cut out. (d) Individual eigenvalues along t. As observed, eigenvalues monotonically grow along t, with growing boost at times of the learning rate drop. (e) $\frac{\delta_t}{N}\lambda_i^t$ along time t, for various i. Similarly to regression, also in NCE case each $\frac{\delta_t}{N}\lambda_i^t$ is balancing, roughly, around the same value along the entire optimization.

U 6-Layer 512-Wide NN, L2 loss, MNIST, GD optimization, Spectrum at time $t = 600000$

Setup MNIST experiment is configured identically to Mona Lisa experiment demonstrated in the main paper. Chosen architecture is Leaky-Relu FC with 6 layers of 512 width each, with overall number of parameters $|\theta| = 1453057$. Training dataset consists of only 10000 images, 1000 images per class were chosen from the original training dataset. Dimension of each image is 784. Gradient descent with constant learning rate 10^{-5} is applied for optimization. The used loss is L2. Dynamics of $g_t(\cdot, \cdot)$ are visualized below, showing the same alignment trends as in case of L2 regression of 2D Mona Lisa dataset.

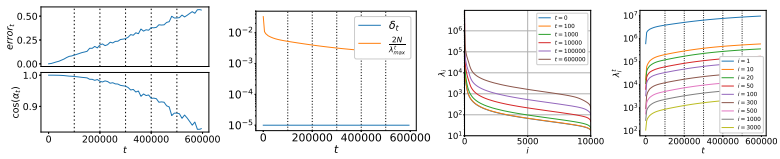


Fig. 40: (a) Accuracy of first order NN dynamics, similar to Figure 1d in the main paper. Depicted is $error_t = \frac{\left\|d\tilde{f}_t - d\bar{f}_t\right\|}{\left\|d\tilde{f}_t\right\|}$, where $d\bar{f}_t = -\frac{\delta_t}{N} \cdot G_t \cdot \bar{m}_t$ is the first-order approximation of a real differential $d\tilde{f}_t \triangleq \bar{f}_{t+1} - \bar{f}_t$; $\cos(\alpha_t)$ is cosine of an angle between $d\tilde{f}_t$ and $d\bar{f}_t$. As observed, first-order approximation is less accurate for high-dimensional data, and degrades with the optimization time t. (b) Learning rate δ_t and its upper stability boundary $\frac{2N}{\lambda^t_{max}}$ along the optimization. (c) Eigenvalues $\{\lambda^t_i\}_{i=1}^N$ for different t. (d) Individual eigenvalues along t. As observed, eigenvalues monotonically grow along t, and their overall magnitude is much higher than in 2D experiments.

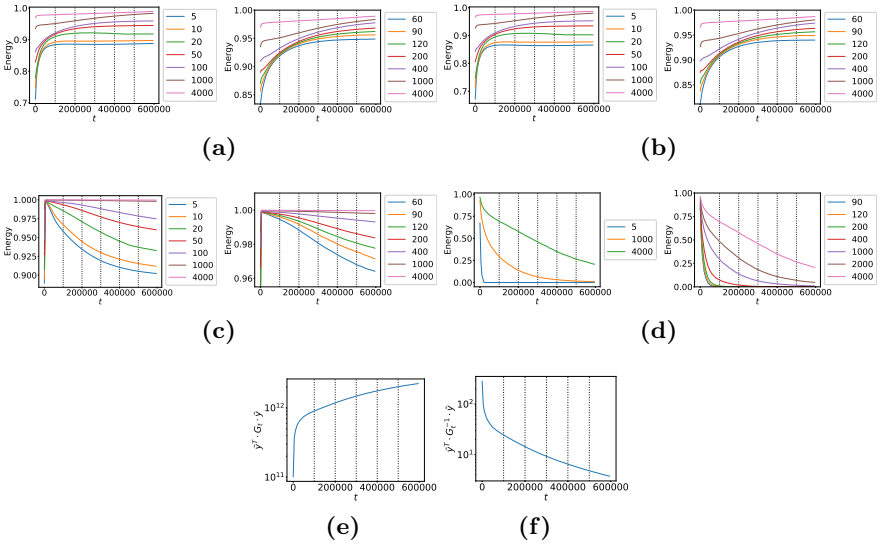
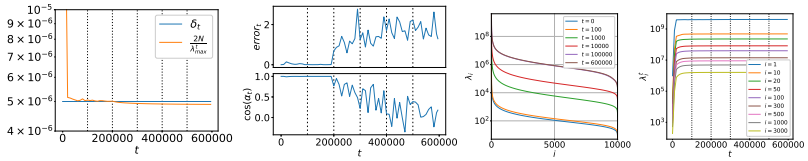


Fig. 41: (a) For different k, relative energy of the label vector $\bar{y}$ in *top* k eigenvectors of G_t, $E_t(\bar{y}, k)$, along the optimization time t. (b) Relative energy of the initial residual $\bar{m}_0 = \bar{f}_0 - \bar{y}$, $E_t(\bar{m}_0, k)$. (c) Relative energy of NN output, $E_t(\bar{f}_t, k)$. (d) Relative energy of the residual, $E_t(\bar{m}_t, k)$. As observed, the alignment between $\bar{y}$ and *top* eigenvectors is growing along the optimization. Similar alignment is also observed with $\bar{m}_0$ due to $\bar{f}_0$ being very close to zero in FC architecture. Further, 99% of $\bar{f}_t$'s energy is contained within *top* 200 eigenvectors for all t. (e) Multiplication $\bar{y}^T \cdot G_t \cdot \bar{y}$ and (f) multiplication $\bar{y}^T \cdot G_t^{-1} \cdot \bar{y}$, along time t. $\bar{y}^T \cdot G_t \cdot \bar{y}$ can be considered as an alignment measure between $\bar{y}$ and *top* eigenvectors, whereas $\bar{y}^T \cdot G_t^{-1} \cdot \bar{y}$ - alignment measure between $\bar{y}$ and *bottom* eigenvectors. Hence, similarly to (a), these alignment measures indicate that the *top* spectrum of G_t aligns towards $\bar{y}$.

V 6-Layer 512-Wide NN, L2 loss, CIFAR100, GD optimization, Spectrum at time $t = 600000$

Setup CIFAR100 experiment is configured identically to MNIST experiment above. Overall number of NN parameters is $|\theta| = 2624513$. Training dataset consists of only 10000 images, 100 images per class were chosen from the original training dataset. Gradient descent with constant learning rate $5 \cdot 10^{-6}$ is applied for optimization. Dynamics of $g_t(\cdot, \cdot)$ are visualized below, showing the same alignment trends as in case of L2 regression of 2D Mona Lisa dataset.



(a) **(b)** **(c)** **(d)**

Fig. 42: (a) Learning rate δ_t and its upper stability boundary $\frac{2N}{\lambda^t_{max}}$ along the optimization. As observed, after $t = 200000$ the stability condition is not satisfied, $\forall t > 200000 : \delta_t > \frac{2N}{\lambda^t_{max}}$. According to constant-Gramian dynamics (Section 4 in the main paper) the unsatisfied stability criteria must diverge the entire optimization, by creating numerical overflows. Yet, in practice the optimization algorithm continues, indicating that not all conclusions for constant-Gramian system are also correct for time-variant Gramian. Also note that $\frac{2N}{\lambda^t_{max}}$ reduces itself, so that the learning rate and its upper stability boundary are almost equal. This behavior was detected in all our experiments. (b) Accuracy of first order NN dynamics, similar to Figure 1d in the main paper. Depicted is $error_t = \frac{\left\| d\tilde{f}_t - d\bar{f}_t \right\|}{\left\| d\tilde{f}_t \right\|}$, where $d\bar{f}_t = -\frac{\delta_t}{N} \cdot G_t \cdot \bar{m}_t$ is the first-order approximation of a real differential $d\tilde{f}_t \triangleq \bar{f}_{t+1} - \bar{f}_t$; $\cos(\alpha_t)$ is cosine of an angle between $d\tilde{f}_t$ and $d\bar{f}_t$. As observed, the accuracy of first-order approximation degrades after $t = 200000$. Hence, this indicates that NN dynamics are far from being linear when the stability criteria is not fulfilled. (c) Eigenvalues $\{\lambda^t_i\}_{i=1}^N$ for different t. (d) Individual eigenvalues along t. As observed, eigenvalues monotonically grow along t, and their overall magnitude is much higher than in 2D experiments.

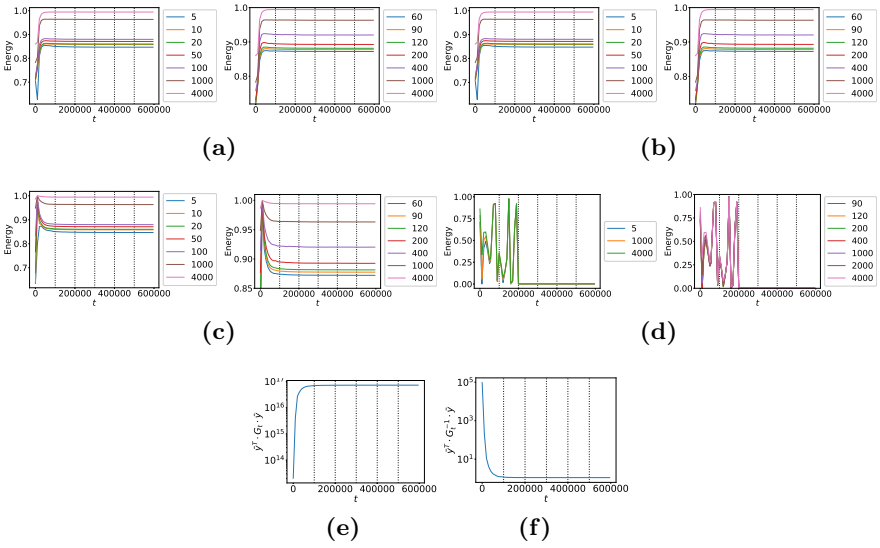


Fig. 43: (a) For different k, relative energy of the label vector $\bar{y}$ in *top* k eigenvectors of G_t, $E_t(\bar{y}, k)$, along the optimization time t. (b) Relative energy of the initial residual $\bar{m}_0 = \bar{f}_0 - \bar{y}$, $E_t(\bar{m}_0, k)$. (c) Relative energy of NN output, $E_t(\bar{f}_t, k)$. (d) Relative energy of the residual, $E_t(\bar{m}_t, k)$. As observed, the alignment between $\bar{y}$ and *top* eigenvectors is growing along the optimization. Similar alignment is also observed with $\bar{m}_0$ due to $\bar{f}_0$ being very close to zero in FC architecture. Further, 97% of $\bar{f}_t$'s energy is contained within *top* 1000 eigenvectors for all t. (e) Multiplication $\bar{y}^T \cdot G_t \cdot \bar{y}$ and (f) multiplication $\bar{y}^T \cdot G_t^{-1} \cdot \bar{y}$, along time t. $\bar{y}^T \cdot G_t \cdot \bar{y}$ can be considered as an alignment measure between $\bar{y}$ and *top* eigenvectors, whereas $\bar{y}^T \cdot G_t^{-1} \cdot \bar{y}$ - alignment measure between $\bar{y}$ and *bottom* eigenvectors. Hence, similarly to (a), these alignment measures indicate that the *top* spectrum of G_t aligns towards $\bar{y}$.

W 6-Layer 256-Wide NN, L2 loss, Social Buzz, SGD optimization, Spectrum at time $t = 600000$

Setup Here we perform experiment over one of UCI regression datasets, specifically Social Buzz. This experiment is configured identically to Mona Lisa experiment demonstrated in the main paper. Chosen architecture is Leaky-Relu FC with 6 layers of 256 width each, with overall number of parameters $|\theta| = 283393$. Training dataset consists of $5 \cdot 10^5$ data points. Dimension of each sample is 77. SGD with initial learning rate $5 \cdot 10^{-6}$ and batch size 10000 is applied for the optimization. The used loss is L2. This experiment uses real-world high-dimensional data, with huge training dataset, and applies very popular SGD algorithm. Hence, its setting is very close to DL usage in practice, and it can serve to further support all conclusions of the main paper. Dynamics of $g_t(\cdot, \cdot)$ are visualized below, showing the same alignment trends as in case of L2 regression of 2D Mona Lisa dataset. To reduce runtime complexity, we calculated Gramian G_t, NN outputs $\bar{f}_t$ and their dynamics only for a set of first 10000 training points.

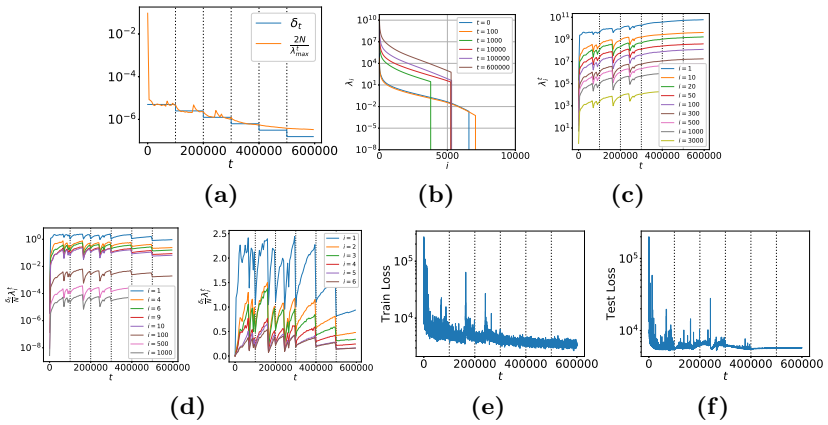


Fig. 44: (a) Learning rate δ_t and its upper stability boundary $\frac{2N}{\lambda^t_{max}}$ along the optimization, where N is batch size 10^4. Dashed vertical lines depict time t at which learning rate δ was decayed. We empirically observe that the boundary chases the learning rate, with a relation $\lambda^t_{max} \propto \frac{1}{\delta_t}$. This behavior was detected in all our experiments, typically for GD optimization. Here we see it to be true also for SGD. (b) Eigenvalues $\{\lambda^t_i\}_{i=1}^N$ for different t. Eigenvalues $\lambda_i < \lambda_{max} \cdot 10^{-8}$ are cut out. (c) Individual eigenvalues along t. As observed, eigenvalues monotonically grow along t, with growing boost at times of the learning rate drop. (d) $\frac{\delta_t}{N}\lambda^t_i$ along time t, for various i. Similarly to 2D Mona Lisa experiment, also here each $\frac{\delta_t}{N}\lambda^t_i$ is balancing, roughly, around the same value along the entire optimization. Hence, we see another evidence for existence of the balancing mechanism mentioned in [8]. This mechanism keeps product of learning rate and eigenvalue almost fixed during the entire optimization. (e) and (f) are training and testing losses respectively.

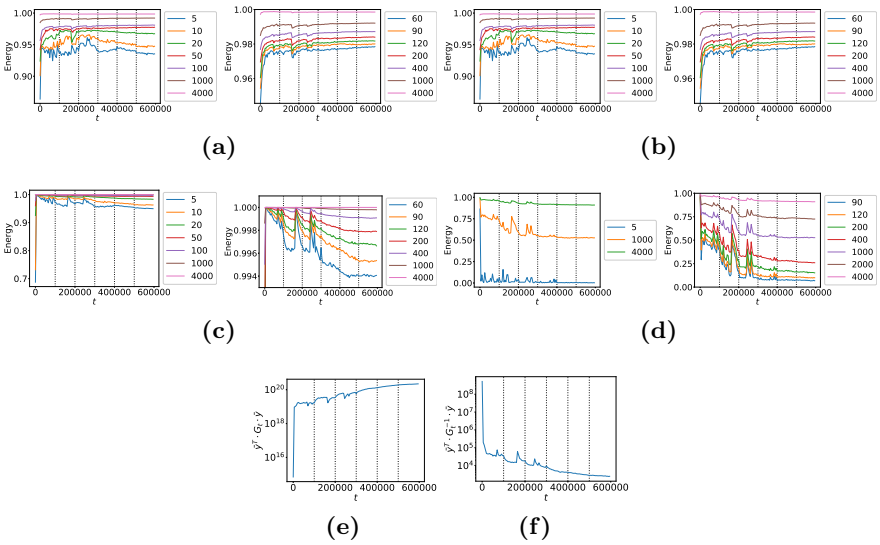


Fig. 45: (a) For different k, relative energy of the label vector $\bar{y}$ in *top* k eigenvectors of G_t, $E_t(\bar{y}, k)$, along the optimization time t. (b) Relative energy of the initial residual $\bar{m}_0 = \bar{f}_0 - \bar{y}$, $E_t(\bar{m}_0, k)$. (c) Relative energy of NN output, $E_t(\bar{f}_t, k)$. (d) Relative energy of the residual, $E_t(\bar{m}_t, k)$. As observed, the alignment between $\bar{y}$ and *top* eigenvectors is growing along the optimization. Similar alignment is also observed with $\bar{m}_0$ due to $\bar{f}_0$ being very close to zero in FC architecture. Further, 99% of $\bar{f}_t$'s energy is contained within *top* 60 eigenvectors for all t. (e) Multiplication $\bar{y}^T \cdot G_t \cdot \bar{y}$ and (f) multiplication $\bar{y}^T \cdot G_t^{-1} \cdot \bar{y}$, along time t. $\bar{y}^T \cdot G_t \cdot \bar{y}$ can be considered as an alignment measure between $\bar{y}$ and *top* eigenvectors, whereas $\bar{y}^T \cdot G_t^{-1} \cdot \bar{y}$ - alignment measure between $\bar{y}$ and *bottom* eigenvectors. Hence, similarly to (a), these alignment measures indicate that the *top* spectrum of G_t aligns towards $\bar{y}$.

References

1. Dyer, E., Gur-Ari, G.: Asymptotics of wide networks from feynman diagrams. arXiv preprint arXiv:1909.11304 (2019)
2. Glorot, X., Bengio, Y.: Understanding the difficulty of training deep feedforward neural networks. In: Proceedings of the thirteenth international conference on artificial intelligence and statistics. pp. 249–256 (2010)
3. Gur-Ari, G., Roberts, D.A., Dyer, E.: Gradient descent happens in a tiny subspace. arXiv preprint arXiv:1812.04754 (2018)
4. Gutmann, M., Hyvärinen, A.: Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In: Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics. pp. 297–304 (2010)
5. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: IEEE Conf. on Computer Vision and Pattern Recognition (CVPR). pp. 770–778 (2016)

6. Huang, J., Yau, H.T.: Dynamics of deep neural networks and neural tangent hierarchy. arXiv preprint arXiv:1909.08156 (2019)
7. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
8. Kopitkov, D., Indelman, V.: Neural spectrum alignment: Empirical study. In: International Conference on Artificial Neural Networks. Springer (2020)
9. Lee, J., Xiao, L., Schoenholz, S.S., Bahri, Y., Sohl-Dickstein, J., Pennington, J.: Wide neural networks of any depth evolve as linear models under gradient descent. arXiv preprint arXiv:1902.06720 (2019)
10. Ramachandran, P., Zoph, B., Le, Q.V.: Searching for activation functions. arXiv preprint arXiv:1710.05941 (2017)
11. Smith, N.A., Eisner, J.: Contrastive estimation: Training log-linear models on unlabeled data. In: Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics. pp. 354–362. Association for Computational Linguistics (2005)