Critical feature learning in deep neural networks

Kirsten Fischer [*1 2] **Javed Lindner** [*1 3 4] **David Dahmen** [1] **Zohar Ringel** [5] **Michael Krämer** [4] **Moritz Helias** [1 3]

Abstract

A key property of neural networks driving their success is their ability to learn features from data. Understanding feature learning from a theoretical viewpoint is an emerging field with many open questions. In this work we capture finite-width effects with a systematic theory of network kernels in deep non-linear neural networks. We show that the Bayesian prior of the network can be written in closed form as a superposition of Gaussian processes, whose kernels are distributed with a variance that depends inversely on the network width N. A large deviation approach, which is exact in the proportional limit for the number of data points $P = \alpha N \to \infty$, yields a pair of forward-backward equations for the maximum a posteriori kernels in all layers at once. We study their solutions perturbatively to demonstrate how the backward propagation across layers aligns kernels with the target. An alternative field-theoretic formulation shows that kernel adaptation of the Bayesian posterior at finite-width results from fluctuations in the prior: larger fluctuations correspond to a more flexible network prior and thus enable stronger adaptation to data. We thus find a bridge between the classical edge-of-chaos NNGP theory and feature learning, exposing an intricate interplay between criticality, response functions, and feature scale.

1. Introduction

A central quest of the theory of deep learning is to understand the inductive bias of network architectures, which is their ability to find solutions that generalize well despite networks being highly overparametrized. The regime of lazy learning (Chizat et al., 2019), in which the network width $N \to \infty$ tends to infinity while the number of training data points P stays constant, is well understood in terms of the neural network Gaussian process (NNGP) (Lee et al., 2018) and the neural tangent kernel (NTK) (Jacot et al., 2018). The NNGP is, however, identical to training the readout weights only (Lee et al., 2019; Yang, 2019). The NNGP kernel follows from the central limit theorem applied to random networks, neglecting any adaptation to the data. While the NTK describes the evolution of weights in all layers, it applies to the case of small learning rates, effectively linearizing the mapping between weights and outputs around the point of initialization. Consequently, weights change only negligibly compared to initialization.

At finite network width or when keeping the ratio $\alpha = P/N$ constant and taking the limit $N \to \infty$, the intermediate network layers adapt to data; they learn "features". Feature learning typically outperforms networks in the lazy regime (Novak et al., 2019; Lee et al., 2020; Geiger et al., 2020; Petrini et al., 2022) and is also required to understand transfer learning, the central mechanism that enables modern foundation models (Bommasani et al., 2022).

We here derive a theory of data-adaptive kernels in deep nonlinear networks trained in a Bayesian manner. We show that the prior for the network outputs f can be written as a superposition of Gaussian processes $f \sim \int \mathcal{N}(0, C)\, p(C)\, dC$. Feature learning may be understood as a reweighing of different components $\mathcal{N}(0, C)$ within this prior ensemble according to the evidence $p(Y|C) = \mathcal{N}(Y|0, C)$ of the training labels Y. As a result, the posterior is dominated by those Gaussian components $\mathcal{N}(0, C)$ that have a high evidence. A wide distribution $p(C)$ leads to a rich prior (see Fig. 1) and thereby enables strong adaptation to the training data. This view allows us to connect feature learning to the notion of criticality: these are points in hyperparameter space where the distribution $p(C)$ becomes particularly wide because the network is at the verge of transitioning between two qualitatively different regimes.

[*]Equal contribution [1] Institute for Advanced Simulation (IAS-6), Computational and Systems Neuroscience, Jülich Research Centre, Jülich, Germany [2]RWTH Aachen University, Aachen, Germany [3]Department of Physics, RWTH Aachen University, Aachen, Germany [4]Institute for Theoretical Particle Physics and Cosmology, RWTH Aachen University, Aachen, Germany [5]The Racah Institute of Physics, The Hebrew University of Jerusalem, Jerusalem, Israel. Correspondence to: Kirsten Fischer <ki.fischer@fz-juelich.de>, Javed Linder <javed.lindner@rwth-aachen.de>.

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

The main contributions of this work are:

- an exact decomposition of the network prior into a superposition of Gaussian processes, whose covariances are distributed with width of $\mathcal{O}(N^{-1})$;
- exact expressions for the Bayesian maximum a posteriori kernels in the proportional limit $N,\ P \to \infty$ with $P/N = \alpha$ that follow from a large deviation approach, yielding a set of forward-backward self-consistent kernel propagation equations;
- demonstration that a perturbative evaluation of the forward-backward propagation of kernels captures feature learning in trained networks;
- the discovery of a tight link between fluctuations near a critical point and the ability of the network to show feature learning, uncovering the driving mechanism behind feature learning as a tradeoff between criticality and feature learning scale of the network output.

2. Related works

Previous work has investigated deep networks within the Gaussian process limit for infinite width $N \to \infty$ (Schoenholz et al., 2017; Lee et al., 2018). (Schoenholz et al., 2017) found optimal backpropagation of signals and gradients when initializing networks at the critical point, the transition to chaos (Molgedey et al., 1992). Our work goes beyond the Gaussian process limit by studying the joint limit $N \to \infty$, $P \to \infty$ with $P/N = \alpha$ fixed. This limit has been investigated with tools from statistical mechanics in deep linear networks (Li & Sompolinsky, 2021), where kernels adapt to data by only changing their overall scale compared to the NNGP limit. A rigorous non-asymptotic solution for deep linear networks in terms of Meijer-G functions (Hanin & Zlokapa, 2023) has shown that the posterior of infinitely deep linear networks with data-agnostic priors is the same as that of shallow networks with evidence-maximizing data-dependent priors. For a teacher-student setting, (Zavatone-Veth et al., 2022) show that in deep linear networks feature learning corrections to the generalization error result from perturbation corrections only at quadratic order or higher. For deep kernel machines, (Yang et al., 2023) find a similar trade-off between network prior and data term as we do; in contrast to our work they study a different limit with P fixed and train on N copies of the data. Their main results can be obtained from ours in the special case of deep linear networks (see C); most importantly for non-linear networks they require the use of normalizing flows to capture non-Gaussian effects while our work provides a mechanistic understanding of such effects.

Previous theoretical work on non-linear networks of finite width $N < \infty$ has employed three different approximation techniques. First, a perturbative approach that computes corrections where the non-linear terms constitute the expansion parameter (Halverson et al., 2021). Second, a perturbative approach based on the Edgeworth expansion that uses the strength of the non-Gaussian cumulants as an expansion parameter. These corrections are computed either in the framework of gradient-based training (Dyer & Gur-Ari, 2020; Huang & Yau, 2020; Aitken & Gur-Ari, 2020; Roberts et al., 2022; Bordelon & Pehlevan, 2023) or Bayesian inference (Yaida, 2020; Antognini, 2019; Naveh et al., 2021; Cohen et al., 2021; Roberts et al., 2022). (Zavatone-Veth et al., 2021) derive a general form of finite-width corrections, resulting from the linear readout layer and the quadratic loss function. Third, non-perturbative Bayesian approaches (Naveh & Ringel, 2021; Seroussi et al., 2023; Pacelli et al., 2023; Cui et al., 2023), that derive self-consistency equations either by saddle-point integration or by variational methods to obtain the Bayesian posterior. (Cui et al., 2023) exploits the Nishimori conditions that hold for Bayes-optimal inference, where student and teacher have the same architecture and the student uses the teacher's weight distribution as a prior; the latter is assumed Gaussian i.i.d., which allows them to use the Gaussian equivalence principle (Goldt et al., 2020) to obtain closed-form solutions. Our work is most closely related to these non-perturbative Bayesian approaches. The qualitative difference is that we describe the trade-off between the data term and the network prior in a large deviation approach that is exact in the proportional limit and we do not require particular assumptions on the data statistics. Our alternative field-theoretical view connects this approach to finite-size fluctuations, by which we discover a link between feature learning corrections and criticality in deep networks.

3. Feature learning theory of Bayesian network posterior

We consider a fully-connected, deep, feed-forward network

$$\begin{aligned} h_\alpha^{(0)} &= W^{(0)} x_\alpha + b^{(0)}\,, \\ h_\alpha^{(l)} &= W^{(l)} \phi\left(h_\alpha^{(l-1)}\right) + b^{(l)} \quad l = 1, \dots, L, \\ f_\alpha &= h_\alpha^{(L)}, \end{aligned} \tag{1}$$

with data indices $\alpha \in \{1, \dots, P\}$, where P denotes the number of training samples. We have inputs $x_\alpha \in \mathbb{R}^D$, hidden states $h_\alpha^{(l)} \in \mathbb{R}^N$, and network output $f_\alpha \in \mathbb{R}$. To ease notation, we assume identical width N for all layers. We derive the theoretical framework for arbitrary activation functions $\phi : \mathbb{R} \mapsto \mathbb{R}$, but consider $\phi(x) = \mathrm{erf}(x)$ for quantitative results in subsequent sections. Further we assume Gaussian i.i.d. priors for all weights $W^{(0)} \in \mathbb{R}^{N\times D}$, $W^{(l)} \in \mathbb{R}^{N\times N}$, $W^{(L)} \in \mathbb{R}^{1\times N}$ and biases $b^{(l)} \in \mathbb{R}^N$, $b^{(L)} \in \mathbb{R}$ so that

$W_{ij}^{(0)} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, g_0/D)$, $W_{ij}^{(l)} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, g_l/N)$ for $i, j = 1, \dots, N$ and $l = 1, \dots, L-1$, $W_i^{(L)} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, g_L/N)$ and $b_i^{(l)} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, g_b)$ for $i = 1, \dots, N$ and $l = 0, \dots, L$. We study the Bayesian posterior distribution conditioned on a training data set consisting of inputs $X = (x_\alpha)_{\alpha=1,\dots,P}$ and corresponding labels $Y = (y_\alpha)_{\alpha=1,\dots,P}$ as in (Naveh et al., 2020; Li & Sompolinsky, 2021; Segadlo et al., 2022). This can alternatively be seen as training the network with stochastic Langevin dynamics (see Appendix G).

3.1. Network prior as superposition of Gaussians

Assuming sample-wise i.i.d. Gaussian regularization noise of variance κ, the network prior $p(Y|X) = \int \prod_{\alpha=1}^{P} \mathcal{N}(y_\alpha|f_\alpha, \kappa)\, p(f|X)\, df$ with network outputs $f = (f_\alpha)_{\alpha=1,\dots,P}$ follows from the network mapping (1) by enforcing the network architecture through Dirac distributions, taking the expectation over all parameters $\Theta = \{W^{(l)}, b^{(l)}\}_l$, and introducing auxiliary variables $C_{\alpha\beta}^{(l)} := g_l/N\, \phi_\alpha^{(l-1)} \cdot \phi_\beta^{(l-1)\mathsf{T}} + g_b$ with the shorthand $\phi_{\alpha i}^{(l)} = \phi\left(h_{\alpha i}^{(l)}\right)$, similar to (Segadlo et al., 2022) (see Appendix A)

$$p(Y|X) = \int \mathcal{D}C\, \mathcal{N}\left(Y|0, C^{(L)} + \kappa\mathbb{I}\right)\, p(C), \tag{2}$$

$$p(C) = \int \mathcal{D}\tilde{C}\, \exp\left(-\operatorname{tr}\tilde{C}^{\mathsf{T}}C + \mathcal{W}(\tilde{C}|C)\right), \tag{3}$$

where $\tilde{C}^{(l)}$ is the conjugate kernel to $C^{(l)}$ and $\operatorname{tr}\tilde{C}^{\mathsf{T}}C = \sum_{\alpha\beta l} \tilde{C}_{\alpha\beta}^{(l)} C_{\alpha\beta}^{(l)}$. This expression shows that the network output is a superposition of centered Gaussian processes $\mathcal{N}\left(0, C^{(L)} + \kappa\mathbb{I}\right)$. Its covariance depends on $C^{(L)}$ that itself is distributed as $p(C^{(L)}) = \int dC^{(1\le l<L)}\, p(C)$, where the joint distribution $p(C) = p(C^{(L)}|C^{(L-1)}) \cdots p(C^{(1)}|C^{(0)})$ of all $C^{(1\le l\le L)}$ decomposes into a chain of conditionals. The distribution $p(C^{(L)})$ is given by its cumulant generating function

$$\begin{aligned}&\mathcal{W}(\tilde{C}|C) \\ &= N \sum_{l=0}^{L-1} \ln\left\langle \exp\left(\frac{g_{l+1}}{N}\phi^{(l)\mathsf{T}}\tilde{C}^{(l+1)}\phi^{(l)}\right)\right\rangle_{\mathcal{N}(0,C^{(l)})} \\ &\quad + \tilde{C}g_b + \tilde{C}^{(0)\mathsf{T}}C^{(0)},\end{aligned} \tag{4}$$

where $\phi^{\mathsf{T}}\tilde{C}\phi = \sum_{\alpha\beta} \phi_\alpha \tilde{C}_{\alpha\beta}\phi_\beta$. We write here and in the following $\langle \dots \rangle_{\mathcal{N}(0,C^{(l)})} \equiv \langle \dots \rangle_{h^{(l)}\sim\mathcal{N}(0,C^{(l)})}$ for the Gaussian expectation value of the activations $h^{(l)}$ with regard to a centered Gaussian measure with covariance matrix $C^{(l)} \in \mathbb{R}^{P\times P}$ and denote as

$$C^{(0)} = \frac{g_0}{D} XX^{\mathsf{T}} + g_b \tag{5}$$

the Gaussian kernel after the readin layer. The network prior (2), written as a superposition of Gaussians, is an exact result. We next determine the maximum a posteriori (MAP) estimate for the $C^{(l)}$.

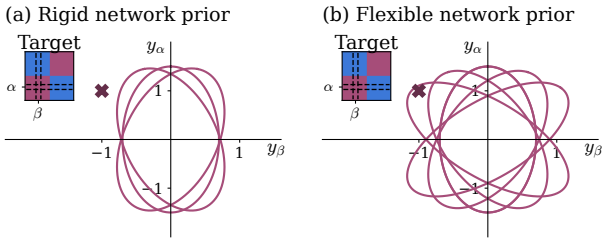

Figure 1. Larger kernel fluctuations enable stronger feature learning. The network prior is a superposition of Gaussians given by $f \sim \int \mathcal{N}(0, C)\, p(C)\, dC$ (pink ellipses). Depending on the network hyperparameters, the distribution of kernels is more concentrated (a) or wider (b), corresponding to smaller or larger kernel fluctuations. The target kernel is given by YY^{T} (inset); the target value for the indicated example samples α, β (dashed lines in inset) from different classes lies at $(+1, -1)$ (red cross). In the Bayesian posterior Gaussian components are reweighed according to the data. Larger fluctuations in (b) allow stronger adaptation to data, leading to richer feature learning.

3.2. Large deviation approach for the maximum a posteriori kernel

The cumulant-generating function (4) has what is known as a scaling form (Touchette, 2009) $\mathcal{W}(\tilde{C}) = N\,\lambda(\tilde{C}/N)$ with an N-independent function λ; thus its k-th cumulant scales with $1/N^{k-1}$ so that C concentrates as $N \to \infty$ around its mean. So while the kernel of the input layer $C^{(0)}$ is deterministic, all subsequent auxiliary variables $C^{(1\le l\le L)}$ are fluctuating quantities with a variance of order $\mathcal{W}'' \sim \mathcal{O}(N^{-1})$. The scaling form at large N implies that we may approximate the integral over $\tilde{C}^{(l)}$ in (3) for $1 \le l \le L$ by the Gärtner-Ellis theorem to obtain a large deviation principle (l.d.p.)

$$\begin{aligned}&-\ln p(C^{(l+1)}|C^{(l)}) \\ &\overset{\text{l.d.p.}}{\simeq} \sup_{\tilde{C}^{(l+1)}} \operatorname{tr}\tilde{C}^{(l+1)\mathsf{T}}C^{(l+1)} - \mathcal{W}(\tilde{C}^{(l+1)}|C^{(l)}) \\ &=: \Gamma(C^{(l+1)}|C^{(l)}),\end{aligned} \tag{6}$$

expressed in terms of the rate function Γ (Touchette, 2009). To provide more intuition for the rate functions Γ, we show in Appendix C that for linear networks the rate function reduces to the Kullback-Leibler divergence between the Gaussian distributions of the two adjacent layers' activations. Thus, the prior has the tendency to keep the distributions in adjacent layers close to one another. The joint probability

$p(C)$ in (3) then decomposes as

$$\ln p(C) = \ln p(C^{(L)}|C^{(L-1)}) \cdots p(C^{(1)}|C^{(0)}) \overset{\text{l.d.p.}}{\simeq} -\sum_{l=1}^{L} \Gamma(C^{(l)}|C^{(l-1)}) =: -\Gamma(C).$$

The supremum condition in (6) amounts to

$$C^{(l+1)}_{\alpha\beta} \equiv \frac{\partial \mathcal{W}}{\partial \tilde{C}^{(l+1)}_{\alpha\beta}} = g_{l+1} \left\langle \phi^{(l)}_\alpha \phi^{(l)}_\beta \right\rangle_{\mathcal{P}^{(l)}} + g_b, \tag{7}$$

$$\left\langle \dots \right\rangle_{\mathcal{P}^{(l)}} \propto \left\langle \dots \exp\left(\frac{g_l}{N} \phi^{(l)\mathsf{T}} \tilde{C}^{(l+1)} \phi^{(l)} \right) \right\rangle_{\mathcal{N}(0, C^{(l)})}, \tag{8}$$

where we defined the non-Gaussian measure $\langle \dots \rangle_{\mathcal{P}^{(l)}} \equiv \langle \dots \rangle_{h^{(l)} \sim \mathcal{P}(\tilde{C}^{(l+1)}, C^{(l)})}$ and its proportionality constant is given by the proper normalization (for details see A.3).

We condition on the training data, enforcing the labels $\{y_\alpha\}_\alpha$, to obtain the posterior distribution for C as $p(C|Y) \propto p(Y, C) \equiv \mathcal{N}\left(Y|0, C^{(L)} + \kappa \mathbb{I}\right) p(C)$, where we read off the latter form of the joint density of Y and C from (2). We are interested in the maximum a posteriori estimate for C, which is given by the stationary points of

$$\mathcal{S}(C) \coloneqq \ln p(C|Y) \overset{\text{l.d.p.}}{\simeq} \mathcal{S}_{\mathrm{D}}(C^{(L)}) - \Gamma(C) + \circ, \tag{9}$$

$$\mathcal{S}_{\mathrm{D}}(C^{(L)}) \coloneqq -\frac{1}{2} Y^{\mathsf{T}} (C^{(L)} + \kappa \mathbb{I})^{-1} Y - \frac{1}{2} \ln \det (C^{(L)} + \kappa \mathbb{I}),$$

where we dropped terms $\circ$ that are independent of C and approximated $p(C)$ by its rate function (6). The exponent $\mathcal{S}(C)$ has two terms: The log likelihood of the training labels $\mathcal{S}_{\mathrm{D}}(C^{(L)}) \sim \mathcal{O}(P)$ and the rate function $-\Gamma(C)$ which arises from the network prior. It is easy to see from (4) that its Legendre transform Γ scales with $\mathcal{O}(N)$.

The stationary point $\partial S(C)/\partial C^{(L)} \overset{!}{=} 0$ of (9) with regard to $C^{(L)}$ therefore arises from a trade-off between the network prior term in the form of Γ and the data term $\mathcal{S}_{\mathrm{D}}$. In the last layer this yields

$$\tilde{C}^{(L)} = \frac{1}{2} (C^{(L)} + \kappa \mathbb{I})^{-1} Y Y^{\mathsf{T}} (C^{(L)} + \kappa \mathbb{I})^{-1} - \frac{1}{2} (C^{(L)} + \kappa \mathbb{I})^{-1}, \tag{10}$$

which expresses the value of $\tilde{C}^{(L)}$ in the final layer in terms of the value of $C^{(L)}$ and the training labels Y. We further show in Appendix A.5 that the conjugate kernel $\tilde{C}^{(L)}$ can be expressed in terms of the second moment of the discrepancies between target and the network output and its trace measures the training loss. Using Price's theorem (see Appendix F) and the fundamental property of the Legendre transform in (6), stationarity $\partial S(C)/\partial C^{(l)} \overset{!}{=} 0$ yields for intermediate network layers $1 \leq l < L$

$$\begin{aligned} \tilde{C}^{(l)}_{\alpha\beta} &= -\frac{\partial \Gamma(C^{(l+1)}|C^{(l)})}{\partial C^{(l)}_{\alpha\beta}} \overset{\text{Legendre}}{=} \frac{\partial \mathcal{W}(\tilde{C}^{(l+1)}|C^{(l)})}{\partial C^{(l)}_{\alpha\beta}} \\ &\overset{\text{Price's theorem}}{=} g_{l+1}\, \tilde{C}^{(l+1)}_{\alpha\beta} \left\langle \left(\phi^{(l)}_\alpha\right)' \left(\phi^{(l)}_\beta\right)' \right\rangle_{\mathcal{P}^{(l)}} \\ &\quad + \delta_{\alpha\beta}\, g_{l+1}\, \tilde{C}^{(l+1)}_{\alpha\alpha} \left\langle \left(\phi^{(l)}_\alpha\right)'' \phi^{(l)}_\alpha \right\rangle_{\mathcal{P}^{(l)}} + \mathcal{O}(N^{-1}), \end{aligned} \tag{11}$$

where we do not spell out terms $\propto \mathcal{O}(N^{-1})$ (the form of which is given in Appendix A). This equation thus gives $\tilde{C}^{(l)}$ in terms of $\tilde{C}^{(l+1)}$ and $C^{(l)}$. The conjugate kernels $\tilde{C}^{(l)}$ propagate information about the relation between inputs and outputs backwards across layers. By (10), these are driven by the difference of two terms: The conjugate kernel of the output layer $\tilde{C}^{(L)}$ measures the mismatch between output kernel $C^{(L)}$ and target kernel YY^{T} and can be interpreted as an error signal. In the following we will see that this error signal on the level of the kernel is backpropagated by the backward response function and exhibits an exponential decay over layers (similar to the response studied in (Schoenholz et al., 2017)), indicating how information backpropagates within the network.

3.3. Forward-backward kernel propagation in the proportional limit

The main result of the previous section is the pair of equations (7) and (11)

$$C^{(l+1)} \overset{(7)}{=} F(C^{(l)}, \tilde{C}^{(l+1)}), \tag{12}$$

$$\tilde{C}^{(l)} \overset{(11)}{=} G(C^{(l)}, \tilde{C}^{(l+1)})\, \tilde{C}^{(l+1)}, \tag{13}$$

with initial and final conditions, respectively, given by (5) and (10), rewritten as

$$\tilde{C}^{(L)} = \frac{1}{2} (C^{(L)} + \kappa \mathbb{I})^{-1} (Y Y^{\mathsf{T}} - C^{(L)} - \kappa \mathbb{I}) (C^{(L)} + \kappa \mathbb{I})^{-1}. \tag{14}$$

This set of equations (including the term $\mathcal{O}(N^{-1})$ in (11)) is exact in the proportional limit $P = \alpha N \to \infty$; this is so because the rate function (6) approximates $-\ln p(C)$ correct up to additive constants, so that the stationary points correctly determine the mode of the posterior for $C^{(l)}$.

The first equation (12) maps the MAP kernel $C^{(l)} \mapsto C^{(l+1)}$ forward through the network. This mapping in the l-th layer depends on $\tilde{C}^{(l+1)}$. This result is similar to the NNGP limit (Neal, 1996; Williams, 1996; Lee et al., 2017): We in fact recover the latter in the case of a fixed number of training samples P and an infinitely wide network with $N \to \infty$ from the stationary point of (9) which is then approximated as $\mathcal{S}(C) \simeq -\Gamma(C)$. In this limit it follows

from the equation of state $\partial\Gamma(C)/\partial C^{(l)}_{\alpha\beta} = \tilde{C}^{(l)}_{\alpha\beta}$ that $\tilde{C} \equiv 0$ vanishes and the measure (8) becomes the Gaussian measure with covariance $C^{(l)}$. In consequence (7) reduces to the NNGP $C^{(l+1)}_{\alpha\beta} = g_{l+1} \left\langle \phi^{(l)}_\alpha \phi^{(l)}_\beta \right\rangle_{\mathcal{N}(0,C^{(l)})} + g_b$. Among others, (Yang & Hu, 2020) show that the NNGP limit fails to capture feature learning which appears in neural networks in the rich regime. Furthermore, we show in Appendix E that the NTK is contained in our framework as a special case that assumes a linear dependence of the output on all layer's weights.

The here presented theoretical framework captures feature learning in settings where the log-likelihood of the data S_D is not negligible compared to Γ in (9); so either in the limit $N \to \infty$ when the number of data samples scales linearly $P = \alpha N$, or when N, P are both large but finite. In the latter case, feature learning results from the leading-order fluctuation corrections in N^{-1}, as we show in the Appendix B. In both cases, the maximum a posteriori for C balances the maximization of the likelihood of the data S_D and the maximization of the log probability $-\Gamma(C)$ from the prior, leading to the equation (13), which propagates $\tilde{C}^{(l+1)} \to \tilde{C}^{(l)}$ backwards and in addition depends on $C^{(l)}$. In particular, we will see that the data term in (9) leads to the correction of the output kernel $C^{(L)}$ towards the target kernel $YY^\intercal$ in (10) and (13). Such an alignment means that the output of the network more closely reproduces the outputs given by the training data. Such a term is absent both in the NNGP and the NTK, both of which only depend on the training data inputs x, and are hence unable to form relationships for the input-label pairs (x, y). In Appendix D, we show for deep linear networks that to leading order the correction terms add a rank one contribution $YY^\intercal$ to the kernel.

Instead of taking expectations over the standard Gaussian measure with covariance $C^{(l)}$ as in the NNGP, the forward propagation (12) here employs a non-Gaussian probability measure (8) that involves the activation function ϕ, the kernels $C^{(l)}$, and the conjugate kernel $\tilde{C}^{(l+1)}$.

Finally, the value for $\tilde{C}^{(L)}$ given by (14) allows for an intuitive interpretation: $C^{(L)} + \kappa\mathbb{I} = YY^\intercal$ implies $\tilde{C}^{(L)} = 0$ and, subsequently by the linear dependence on $\tilde{C}^{(l+1)}$ in (13), that all vanish, $\tilde{C}^{(1\leq l\leq L)} = 0$. Hence at this point $\tilde{C}$ does not drive further adaptation towards the target, as the output kernel is already perfectly aligned to the desired target.

3.4. Perturbative, leading-order solution of the forward-backward equations

The presented approach does not depend on the choice of the activation function ϕ. For general activation functions ϕ, however, the exact expressions for the feature learning limit are hardly tractable due to the non-Gaussian expectation value with regard to the measure (8). The non-Gaussianity in the measure (8) comes in the form of $\frac{g_l}{N}\phi^{(l)}_\alpha \tilde{C}^{(l+1)}_{\alpha\beta}\phi^{(l)}_\beta$ in the exponent, so the magnitude of the entries are diminished by N^{-1} compared to those of $-\frac{1}{2}h^{(l)}_\alpha [C^{(l)}]_{\alpha\beta} h^{(l)}_\alpha$ from the Gaussian part of the measure. So expanding in N^{-1}, which amounts to expanding to linear order in $\tilde{C}$, we may replace the forward propagation (7) by

$$C^{(l+1)}_{\alpha\beta} = g_{l+1} \left\langle \phi^{(l)}_\alpha \phi^{(l)}_\beta \right\rangle_{\mathcal{N}(0,C^{(l)})} + g_b + \frac{g^2_{l+1}}{N} \sum_{\gamma,\delta} V^{(l)}_{\alpha\beta,\gamma\delta} \tilde{C}^{(l+1)}_{\gamma\delta} + \mathcal{O}\left(N^{-2}\right), \tag{15}$$

$$V^{(l)}_{\alpha\beta,\gamma\delta} := \left\langle \phi^{(l)}_\alpha \phi^{(l)}_\beta \phi^{(l)}_\gamma \phi^{(l)}_\delta \right\rangle_{\mathcal{N}(0,C^{(l)})} - \left\langle \phi^{(l)}_\alpha \phi^{(l)}_\beta \right\rangle_{\mathcal{N}(0,C^{(l)})} \left\langle \phi^{(l)}_\gamma \phi^{(l)}_\delta \right\rangle_{\mathcal{N}(0,C^{(l)})}, \tag{16}$$

where all expectation values are Gaussian $\langle \ldots \rangle_{\mathcal{N}(0,C^{(l)})}$. Likewise, at the same order of approximation, we may replace in (11) $\langle \ldots \rangle_{\mathcal{P}^{(l)}}$ by $\langle \ldots \rangle_{\mathcal{N}(0,C^{(l)})}$, because corrections come with at least one factor N^{-1}. While the two-point integrals in (11), (15) and (16) have closed-form analytical solutions for certain non-linearities such as $\phi = \mathrm{erf}(x)$, the four-point integral in (16) is evaluated numerically (for details see Appendix H). The kernels $C^{(l+1)}$ thus receive a correction from the backpropagated error signal in the form of $\tilde{C}^{(l+1)}$. The correction of $C^{(l+1)}_{\alpha\beta}$ results not only from the kernel element itself $\tilde{C}^{(l+1)}_{\alpha\beta}$, but also depends on its interaction with all other data samples via the four-point interaction term $\sum_{\gamma,\delta} V^{(l)}_{\alpha\beta,\gamma\delta} \tilde{C}^{(l+1)}_{\gamma\delta}$.

We solve the self-consistency equations for both kernels $C^{(l)}$ and conjugate kernels $\tilde{C}^{(l)}$ iteratively. Details on a numerically stable implementation are given in Appendix H. All code is available under (10.5281/zenodo.11205498).

Experiments

We compare the obtained analytical results for the output kernel $C^{(L)}$ conditioned on the training data to the numerical implementation of sampling the kernel $C^{(L)}_{\text{emp}}$ from the posterior distribution using Langevin stochastic gradient descent (see Appendix G). As a measure we use the centered kernel alignments (CKA, see Appendix I) of both the analytical kernel $C^{(L)}$ and the Langevin sampled kernels $C^{(L)}_{\text{emp}}$ with the target kernel $YY^\top$ respectively. Since our framework does not presuppose any assumptions on the data, we study two different tasks: XOR and binary classification on MNIST digits; the numerical results match our theoretical expectations consistently in both cases.

XOR (Refinetti et al., 2021) show that random feature models, which are known to correspond to the NNGP (Mei & Montanari, 2022), are unable to solve the non-linearly

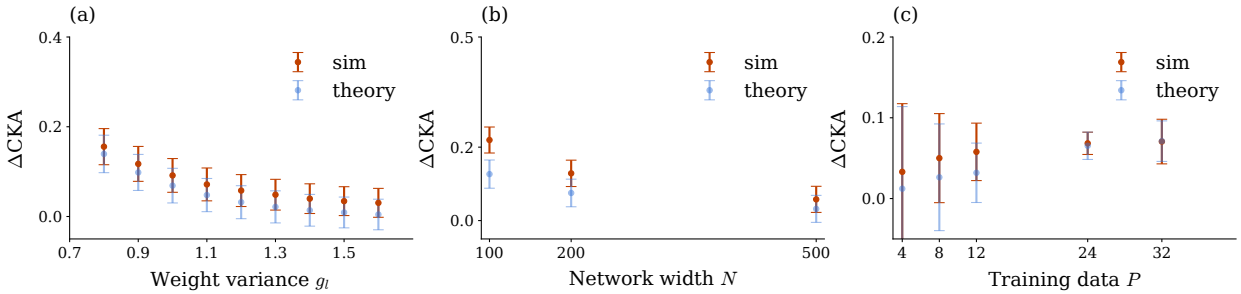

Figure 2. Comparison between theory and simulation for the XOR task. The difference $\Delta\text{CKA} = \text{CKA}(C^{(l)}, YY^\mathsf{T}) - \text{CKA}(C^{\text{NNGP}}, YY^\mathsf{T})$ measures kernel adaptation relative to the naive NNGP kernels. The ΔCKA of the data-dependent kernels (blue: theory; red: empirical) increases significantly (a) for smaller weight variance, (b) for more narrow networks, and (c) for larger amounts of training data. Parameters: XOR task with $\sigma^2 = 0.4, D = 100, L = 3$, (a) $N = 500$, $P = 12$, (b) $P = 12$, $g_l = 1.2$, (c) $N = 500$, $g_l = 1.2$. Results are averaged over 10 training data sets and error bars indicate standard deviation.

separable task XOR optimally. We study the XOR task in a setting where neural networks exhibit feature learning compared to random feature models (Refinetti et al., 2021). The feature-corrected kernels that we obtain from our theory have a larger CKA than the NNGP (see Fig. 2), indicating that finite-width effects lead to kernel corrections in the direction of the target kernel. Note that the CKA is by construction invariant to a global rescaling of the kernel and instead captures the kernel structure. Thus, the difference between NNGP and empirical kernels is further numerical evidence that the kernels acquire structure beyond a global rescaling, in contrast to deep linear networks (Li & Sompolinsky, 2021) and opposed to approximate results employing Gaussian equivalence theory (Pacelli et al., 2023; Baglioni et al., 2024). We observe a stronger kernel alignment for smaller weight variance g_l (see Fig. 2(a)). As expected, the feature-corrected kernels approach the NNGP limit for $\alpha = P/N \to 0$ when keeping P fixed in Fig. 2(b). Deviations for small N in Fig. 2(b) and in Fig. 2(c) for increasing P at fixed N result from the perturbative treatment of $\tilde{C}$ in the numerical solution of the self-consistency equations, which is strictly valid only for $\alpha = P/N \ll 1$.

MNIST We study a binary classification task on MNIST (LeCun et al., 1998) between digits 0 and 3. The feature-corrected kernels obtained from theory show increased kernel alignment with the target kernel YY^T compared to the NNGP, matching the behavior in neural networks trained by Langevin dynamics (see Fig. 3).

4. Interplay between criticality and output scale

To understand the driving forces behind kernel adaptation to data as presented in the previous section, we study the self-consistency equations (11) for the network kernels in detail. For the presented feature learning theory, we reveal a link to fluctuations, the response function and the scales within the network.

4.1. Fluctuations lead to feature learning

For the network prior in (2), we have defined auxiliary variables $C^{(l)}_{\alpha\beta} = g_l/N\, \phi^{(l-1)}_\alpha \cdot \phi^{(l-1)\mathsf{T}}_\beta + g_b$. For infinitely-wide networks $N \to \infty$ these quantities concentrate, the scalar product over neuron indices becomes an expectation value and we obtain the NNGP kernel given by $C^{(l),\text{NNGP}}_{\alpha\beta} = g_l \langle \phi^{(l-1)}_\alpha \phi^{(l-1)}_\beta \rangle_{\mathcal{N}(0,C^{(l-1)})} + g_b$. For large but finite network width $N < \infty$, the realizations of the auxiliary variables measured from a particular network realization $\Theta = \{W^{(l)}, b^{(l)}\}_l$ fluctuate around the NNGP kernel

$$C^{(l)}_{\alpha\beta} = C^{(l),\text{NNGP}}_{\alpha\beta} + \delta C^{(l)}_{\alpha\beta}.$$

We now show that corrections to the NNGP result derived from the perturbative approach (15) can alternatively be understood as fluctuation corrections in a field-theoretic formulation (see Appendix B). We rewrite (2) and (3) as

$$p(Y|X) = \int \mathcal{D}C \int \mathcal{D}\tilde{C}\, \exp\left(\mathcal{S}(C,\tilde{C}) + \mathcal{S}_\text{D}(C^{(L)}|Y)\right),$$
$$\mathcal{S}_\text{D}(C^{(L)}|Y) = \ln \mathcal{N}(Y|0, C^{(L)} + \kappa\mathbb{I}). \tag{17}$$

We perform a Laplace approximation of $\exp\left(\mathcal{S}(C,\tilde{C})\right)$ around its saddle point $C^{(l),*} = C^{(l),\text{NNGP}}$, $\tilde{C}^{(l),*} = 0$:

$$p(Y|X) \tag{18}$$
$$\simeq \int \mathcal{D}\delta C \int \mathcal{D}\delta\tilde{C}\, \exp\Big(\frac{1}{2}(\delta C, \delta\tilde{C})^\mathsf{T} \mathcal{S}^{(2)} (\delta C, \delta\tilde{C}) + \mathcal{S}_\text{D}(C^{(L)}_* + \delta C^{(L)}|Y)\Big),$$

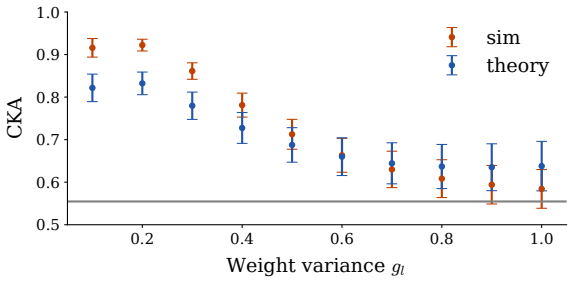

Figure 3. Comparison between theory and simulation for MNIST. The strength of kernel adaptation measured as $\mathrm{CKA}(C^{(l)}, YY^\mathsf{T})$ shows a maximum that is consistent for theory (blue) and simulation (red). Kernel adaptation increases significantly relative to the NNGP (gray). Parameters: MNIST task with $L = 2$, $N = 2000$. Results are averaged over 10 training data sets and error bars indicate plus minus one standard deviation.

where we write $\delta C = C - C^*$, $\delta\tilde{C} = \tilde{C} - \tilde{C}^*$ and we denote the Hessian of $\mathcal{S}(C, \tilde{C})$ as $S^{(2)} = \partial^2 S / \partial(C, \tilde{C})$, whose negative inverse yields the second cumulant of $(C, \tilde{C})$

$$\left[-S^{(2)}\right]^{-1} = \begin{pmatrix} \langle \delta C\, \delta C \rangle & \langle \delta C\, \delta\tilde{C} \rangle \\ \langle \delta\tilde{C}\, \delta C \rangle & 0 \end{pmatrix}.$$

Computing the saddle point of δC in (18) by taking the effect of $\partial \mathcal{S}_\mathrm{D} / \partial \delta C^{(L)}$ into account, we get

$$\delta C^{(l)}_{\alpha\beta} = g_l \sum_{\gamma\delta} \frac{\partial \langle \phi^{(l-1)}_\alpha \phi^{(l-1)}_\beta \rangle_{\mathcal{N}(0, C^{(l-1)})}}{\partial C^{(l-1)}_{\gamma\delta}} \delta C^{(l-1)}_{\gamma\delta} + g_l^2 \sum_{\gamma\delta} V^{(l-1)}_{\alpha\beta,\gamma\delta}\, \delta\tilde{C}^{(l)}_{\gamma\delta}. \tag{19}$$

The first term in (19) results from linearly correcting the NNGP expression by the shift in δC

$$g_l \left\langle \phi^{(l-1)}_\alpha \phi^{(l-1)}_\beta \right\rangle_{\mathcal{N}(0, C^{(l-1)} + \delta C^{(l-1)})} = C^{(l),\mathrm{NNGP}}_{\alpha\beta} + \sum_{\gamma\delta} \frac{\partial C^{(l),\mathrm{NNGP}}_{\alpha\beta}}{\partial C^{(l-1)}_{\gamma\delta}} \delta C^{(l-1)}_{\gamma\delta} + \mathcal{O}(\delta C)^2.$$

The second term in (19) corresponds to the corrections in (15). By identifying $g_l^2 V^{(l-1)}_{\alpha\beta,\gamma\delta}$ given by (16) with the covariance of the auxiliary variables, we see how the feature learning corrections in the self-consistency equations result from fluctuation corrections. Thus, larger fluctuations lead to stronger feature learning. Fluctuations become especially larger close to critical points that mark phase transitions between qualitatively different states in neural networks.

4.2. Feature learning corrections close to criticality

We study the relation of the self-consistency equations to criticality in neural networks (Schoenholz et al., 2017). The self-consistency equations for the conjugate kernels $\tilde{C}^{(l)}$ are given by the iterative expression (11). Ultimately, the network kernels $C^{(l-1)}$ receive a correction from the conjugate kernel $\tilde{C}^{(l)}$. For $\alpha \neq \beta$ it can be explicitly written to linear order as

$$\tilde{C}^{(l)}_{\alpha\beta} = \tilde{C}^{(L)}_{\alpha\beta}\, \chi^{(l),\leftarrow}_{\alpha\beta}, \tag{20}$$

$$\chi^{(l),\leftarrow}_{\alpha\beta} := \prod_{s=l}^{L-1} g_{s+1} \left\langle \left(\phi^{(s)}_\alpha\right)' \left(\phi^{(s)}_\beta\right)' \right\rangle_{h^{(s)} \sim \mathcal{N}(0, C^{(s)})}, \tag{21}$$

where $\tilde{C}^{(L)}$ is given by (10). As discussed in the previous section, the term $\tilde{C}^{(L)}$ is related to the mismatch between the output kernel $C^{(L)}$ and the target kernel given by YY^T. This error signal gets backpropagated from layer to layer by the multiplicative terms in (21). We identify $\chi^{(l),\leftarrow}$ as the gradient response function. It is related to the forward response function (Schoenholz et al., 2017) that measures how perturbations in the input kernel affect network kernels in later layers $\chi^{l,\rightarrow}_{\alpha\beta} = \partial C^{(l)}_{\alpha\beta} / \partial C^{(0)}_{\alpha\beta}\big|_{C_\mathrm{NNGP}}$. Both response functions appear naturally in a field-theoretic description of neural networks by considering Gaussian fluctuations of the kernels as a first-order correction at finite width (Segadlo et al., 2022; Fischer et al., 2023). The main difference between the forward and gradient response function is that the signal perturbation leading to responses in the network arises in different layers and thus propagates in opposite directions: forward response propagates from input to output, gradient response propagates from output to input.

For a particular set of network hyperparameters, both response functions exhibit long-range correlation across layers (see Fig. 4(a)-(b)). This hyperparameter manifold separates an ordered and a chaotic phase for which the network signal for different inputs either strongly correlates or decorrelates. Therefore, this is referred to as the critical point. Close to criticality the signal can propagate to large depths and thus network trainability in deep feed-forward neural networks is improved. This is known as edge-of-chaos initialization (Poole et al., 2016; Schoenholz et al., 2017) and closely related to the idea of dynamical isometry (Saxe et al., 2014; Pennington et al., 2017; Burkholz & Dubatovka, 2019).

Due to the gradient response function appearing in the self-consistency equation (11) for feature learning corrections, these fluctuation corrections also propagate furthest close to criticality (see Fig. 4(d)). However, the error signal $\tilde{C}^{(L)}$ (10) itself depends non-linearly on both weight variance g_l and bias variance g_b, becoming largest for small weight variance g_l (see Fig. 4(d)). To get a qualitative idea, we study the interplay of these two effects at the NNGP, which corre-

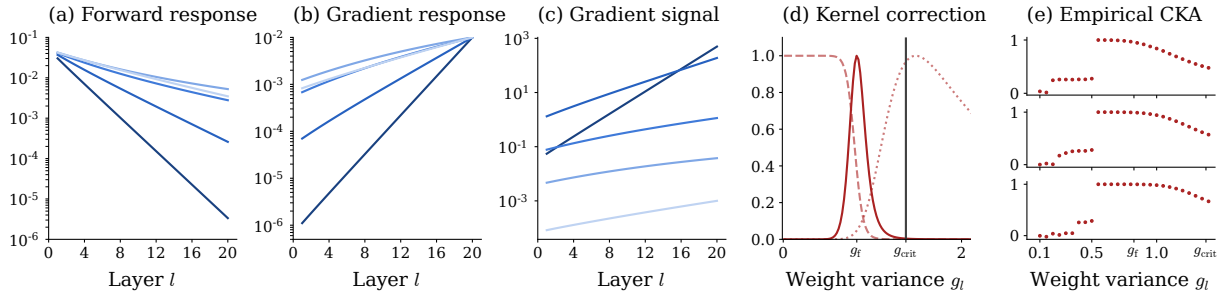

Figure 4. Finite-size effects close to criticality in feature learning. (a)-(b) Forward response $\chi^{l,\rightarrow}$ and gradient response $\chi^{l,\leftarrow}$ measure relative signal propagation across network layers. The signal propagates furthest close to criticality (g_l increases from dark to light blue). (c) Backpropagated conjugate kernel $\tilde{C}^{(l)}$ across network layers for varying weight variance g_l (increasing from dark to light). The kernel mismatch $\tilde{C}^{(L)}$ in the output also depends on the weight variance g_l, so that the curves of $\tilde{C}^{(l)}$ intersect at different depths. Larger $\tilde{C}^{(1)}$ in the first layer leads to stronger feature learning corrections in (15). (d) The kernel correction term $\tilde{C}^{(0)}$ in the readin layer (solid line, slice for $l = 1$ in (c)) is composed of the gradient response $\chi^{1,\leftarrow}$ (dotted line, slice for $l = 1$ in (b)) and the error signal $\tilde{C}^{(L)}$ (dashed line). Thus, strongest feature learning corrections occur for a weight variance g_f shifted away from the critical point (vertical line) to smaller values. (e) CKA for trained networks between $C^{(l)}_{\text{emp}}$ and $YY^\intercal$ ($l = 10,\ 15,\ 20$ from top to bottom). Other parameters: XOR task with $\sigma^2 = 0.4$, $g_l \in \{0.6,\ 0.825,\ 1.1,\ g_{\text{crit}} \approx 1.38,\ 2.2\}$, $g_b = 0.05$, $L = 20$, $N = 500$, $\kappa = 10^{-3}$, $P = 12$.

sponds to the initial iteration step of the full self-consistent solution. The total gradient signal consisting of the error signal $\tilde{C}^{(L)}$ and the gradient response $\chi^{(l),\leftarrow}$ depends on the network depth (see Fig. 4(c)). The product of the two terms in (20) leads to a peak of $\tilde{C}^{(1)}$ in the first layer at a weight variance $g_l \simeq g_f$, which is well below the critical value g_{crit} (see Fig. 4(d)). Numerical evidence in Fig. 4(e) confirms that indeed kernel adaptation in fully trained networks tends to increase up to around $g \simeq g_f$ when approached from above. Below g_f adaptation suddenly drops, as the network enters the regime of vanishing gradients. While criticality is known to be not the only relevant criteria for network training (Bukva et al., 2023), we are able to explicitly point out the interaction of criticality with other factors like output scale.

4.3. Downscaling of network output enhances feature learning

Now that we have seen how feature learning corrections result from the interplay between response function and error signal of the output kernel, we can ask how we can promote feature learning in deep neural networks. While the response function depends on the behavior across layers, the error signal depends solely on the output layer. Reducing only the weight variance of the output layer g_L shrinks the scale of the output kernel $C^{(L)}$ relative to the target kernel $YY^\intercal$, thereby directly increasing feature learning corrections in (11).

Previous works (Geiger et al., 2020; 2021; Yang & Hu, 2020; Yang et al., 2021; Bordelon & Pehlevan, 2023) studied how the scaling of the output layer affects the transition between lazy and feature learning. We here consider the case where the output weight variance is reduced by a factor γ_0 that is not extensive in the number of hidden units so that $g_L \mapsto g_L/\gamma_0$. To understand the effect of such a feature scale γ_0 on the self-consistency equations (11), we derive the dependence of feature learning corrections on the output kernel $C^{(L)}_{\alpha\beta} \propto g_L/\gamma_0$ as $\tilde{C}^{(L)}_{\alpha\beta} \overset{(10)\text{ for }\kappa=0}{\propto} \gamma_0^2 + \mathcal{O}(\gamma_0)$. From (21) follows $\chi^{(L),\leftarrow} \propto g_L/\gamma_0$, so that the fluctuation corrections resulting from the conjugate kernel in the input layer $\tilde{C}^{(0)}_{\alpha\beta}$ increase linearly with the feature scale

$$\tilde{C}^{(0)}_{\alpha\beta} = \chi^{(L),\leftarrow}\tilde{C}^{(L)}_{\alpha\beta} \propto \gamma_0 + \mathcal{O}_{\gamma_0}(1), \tag{22}$$

leading to a stronger adaptation of the network to given training data. From (22) follows that gradually increasing the feature scale γ_0 consistently increases feature learning in all network layers Fig. 5(a). The intuition for this effect is that the reduced scale of the output kernel $C^{(L)}$ causes the network kernels $C^{(l)}$ to expand into the direction of the target kernel $YY^\intercal$. While the interplay between criticality and weight variance g_l for $l < L$ from the previous subsection stays the same, increasing the feature scale overall increases feature learning for any weight variance g_l Fig. 5(b).

5. Discussion

We here present a new theoretical framework that describes how network kernels adapt non-linearly to training data,

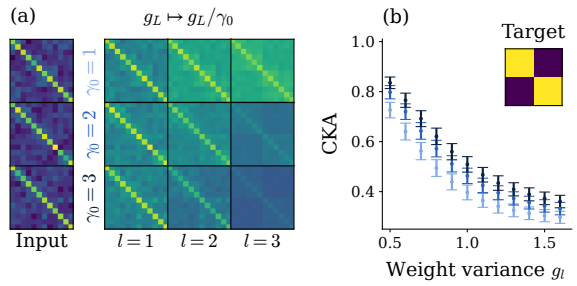

Figure 5. Increase of feature scale leads to stronger kernel adaption. (a) Network kernel $C^{(l)}$ across layers $l = 1,\ 2,\ 3$ for different values of feature scale γ_0. Despite only rescaling the output layer, feature learning is enhanced in all network layers. Other parameters: XOR task with $\sigma^2 = 0.4$, $P = 12$, $N = 200$, $L = 3$, $g_l = g = 0.5$. (b) CKA between output kernel $C^{(L)}$ and target kernel $YY^\intercal$ for different levels of feature scaling ($\gamma_0 = 1,\ 2,\ 3$ from dark to light blue). Larger feature scale consistently leads to stronger kernel adaptation. Results are averaged over 10 training data sets and error bars indicate standard deviation. Other parameters: XOR task with $\sigma^2 = 0.4$, $P = 12$, $L = 3$.

thereby learning features of a given task. We present two complimentary approaches: For the proportional limit $P = \alpha N \to \infty$, we employ a large deviation principle which yields a pair of forward-backward propagation equations for the maximum a posteriori kernel and its conjugate kernel that need to be solved self-consistently. We here explore a perturbative solution in N^{-1}, which can be regarded as the limit where $P/N = \alpha \ll 1$, which yet shows reasonable agreement with fully trained networks. In particular, it correctly predicts the adaptation of the kernel to the target in deep non-linear networks, in contrast to the kernel scaling theory for deep linear networks (Li & Sompolinsky, 2021) or shallow non-linear networks in a Gaussian equivalence setting (Baglioni et al., 2024). In the limit $P = \text{const.}, N \to \infty$, the solutions of our theory converge to the NNGP as they should. A complimentary view obtains qualitatively identical equations for finite-width networks by deriving kernels from fluctuating auxiliary variables. This view shows that the network prior comprises a plethora of kernels in the form of a superposition of Gaussians, a result that holds exactly. When conditioning on the training data in a Bayesian manner, the Gaussian components and their associated kernels get reweighed in the network posterior, yielding a data-dependent maximum a posteriori kernel. The kernel fluctuations allow the network to sample from many different kernels, making it more adaptive to data.

In addition to obtaining data-dependent posterior network kernels, the presented theory allows us to understand driving forces behind feature learning: We observe an interplay between the response function of the network and the error signal that is being propagated backwards through the network by the response function. While being close to criticality allows the signal to propagate to deep layers (Schoenholz et al., 2017), the error signal itself depends differently on network hyperparameters. In consequence, kernel adaptation is strongest slightly away from criticality. Finally, we see how downscaling the network output by feature scaling increases the error signal, thereby promoting feature learning in the network.

Limitations For the self-consistency equations to be tractable for non-linear networks, we approximate them to linear order in the conjugate kernels. This assumes small corrections relative to the NNGP limit, more specifically $\alpha = P/N \ll 1$. For linear networks, this additional approximation is not required. By iterating from wider networks to more narrow networks, we are able to determine kernel corrections for different network widths. Nevertheless, the here presented approach is strictly valid only for large N, since we use a large deviation principle, and for non-linear networks is limited to small amounts of training data relative to the network size $\alpha = P/N \ll 1$.

Outlook While the here presented results focus on kernels, we aim to extend the theoretical framework to study the predictor statistics in the future. This requires computing non-Gaussian corrections from the posterior of kernels and determining the interaction between test samples with training samples. The theoretical framework can be straightforwardly extended to other network architectures such as RNNs, CNNs, and ResNets, using the respective network priors (Segadlo et al., 2022; Garriga-Alonso et al., 2019; Fischer et al., 2023). Investigating the differences in kernel adaptation for these network architectures is an interesting question for future work. To study the effect of noise in input data on feature learning (Lindner et al., 2023), we plan to include fluctuations of the input kernel in the theoretical framework. Furthermore, the theoretical framework can be extended to study feature learning in other network architectures such as transformers, for which the NNGP is already known (Hron et al., 2020). We believe that the presented theoretical framework constitutes a versatile tool for studying different aspects of data-dependent kernels and feature learning.

Impact Statement

Engineering of novel technologies in AI continues to supersede our theoretical understanding of it. Understanding what mechanisms drive kernel adaptation in neural networks is highly relevant for task-sensitive hyperparameter optimization (HPO) as it enables informed decisions about network

width, network depth, network initialization etc. In this work, we draw a novel link between kernel adaptation and criticality, showing that maximal kernel adaptation happens at prior weight variances that are significantly different from those predicted by criticality only.

Acknowledgements

We thank Claudia Merger, Itay Lavie, Inbar Seroussi, and Noa Rubin for helpful discussions. This work was partly supported by the German Federal Ministry for Education and Research (BMBF Grant 01IS19077A to Jülich and BMBF Grant 01IS19077B to Aachen) and funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) - 368482240/GRK2416 , and the Helmholtz Association Initiative and Networking Fund under project number SO-092 (Advanced Computing Architectures, ACA). This work was supported by a fellowship of the German Academic Exchange Service (DAAD). Open access publication funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – 491111487. MK would like to thank the Institute for Advanced Simulation (IAS-6) at Forschungszentrum Jülich and its directors Markus Diesmann and Sonja Grün for their hospitality during regular visits.

References

10.5281/zenodo.11205498. URL https://doi.org/10.5281/zenodo.11205498. 3.4

Aitken, K. and Gur-Ari, G. On the asymptotics of wide networks with polynomial activations, 2020. URL https://arxiv.org/abs/2006.06687. 2

Antognini, J. M. Finite size corrections for neural network gaussian processes. 2019. 2

Baglioni, P., Pacelli, R., Aiudi, R., Di Renzo, F., Vezzani, A., Burioni, R., and Rotondo, P. Predictive power of a bayesian effective action for fully-connected one hidden layer neural networks in the proportional limit. (arXiv:2401.11004), January 2024. URL http://arxiv.org/abs/2401.11004. arXiv:2401.11004 [cond-mat]. 3.4, 5

Bommasani, R., Hudson, D. A., Adeli, E., Altman, R., Arora, S., von Arx, S., Bernstein, M. S., Bohg, J., Bosselut, A., Brunskill, E., Brynjolfsson, E., Buch, S., Card, D., Castellon, R., Chatterji, N., Chen, A., Creel, K., Davis, J. Q., Demszky, D., Donahue, C., Doumbouya, M., Durmus, E., Ermon, S., Etchemendy, J., Ethayarajh, K., Fei-Fei, L., Finn, C., Gale, T., Gillespie, L., Goel, K., Goodman, N., Grossman, S., Guha, N., Hashimoto, T., Henderson, P., Hewitt, J., Ho, D. E., Hong, J., Hsu, K., Huang, J., Icard, T., Jain, S., Jurafsky, D., Kalluri, P., Karamcheti, S., Keeling, G., Khani, F., Khattab, O., Koh, P. W., Krass, M., Krishna, R., Kuditipudi, R., Kumar, A., Ladhak, F., Lee, M., Lee, T., Leskovec, J., Levent, I., Li, X. L., Li, X., Ma, T., Malik, A., Manning, C. D., Mirchandani, S., Mitchell, E., Munyikwa, Z., Nair, S., Narayan, A., Narayanan, D., Newman, B., Nie, A., Niebles, J. C., Nilforoshan, H., Nyarko, J., Ogut, G., Orr, L., Papadimitriou, I., Park, J. S., Piech, C., Portelance, E., Potts, C., Raghunathan, A., Reich, R., Ren, H., Rong, F., Roohani, Y., Ruiz, C., Ryan, J., Ré, C., Sadigh, D., Sagawa, S., Santhanam, K., Shih, A., Srinivasan, K., Tamkin, A., Taori, R., Thomas, A. W., Tramér, F., Wang, R. E., Wang, W., Wu, B., Wu, J., Wu, Y., Xie, S. M., Yasunaga, M., You, J., Zaharia, M., Zhang, M., Zhang, T., Zhang, X., Zhang, Y., Zheng, L., Zhou, K., and Liang, P. On the opportunities and risks of foundation models. (arXiv:2108.07258), July 2022. URL http://arxiv.org/abs/2108.07258. arXiv:2108.07258 [cs]. 1

Bordelon, B. and Pehlevan, C. Self-consistent dynamical field theory of kernel evolution in wide neural networks*. *Journal of Statistical Mechanics: Theory and Experiment*, 2023(11):114009, nov 2023. doi: 10.1088/1742-5468/ad01b0. URL https://dx.doi.org/10.1088/1742-5468/ad01b0. 2, 4.3

Bukva, A., de Gier, J., Grosvenor, K. T., Jefferson, R., Schalm, K., and Schwander, E. Criticality versus uniformity in deep neural networks. (arXiv:2304.04784), April 2023. URL http://arxiv.org/abs/2304.04784. arXiv:2304.04784 [cs, stat]. 4.2

Burkholz, R. and Dubatovka, A. Initialization of relus for dynamical isometry. In *Advances in Neural Information Processing Systems*, volume 32, 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/d9731321ef4e063ebbee79298fa36f56-Paper.pdf. 4.2

Canatar, A. and Pehlevan, C. A kernel analysis of feature learning in deep neural networks. In *2022 58th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pp. 1–8, 2022. doi: 10.1109/Allerton49937.2022.9929375. I

Chizat, L., Oyallon, E., and Bach, F. On lazy training in differentiable programming. In *Advances in neural information processing systems*, volume 32, 2019. URL https://openreview.net/pdf?id=rkgxDVSlLB. 1

Cohen, O., Malka, O., and Ringel, Z. Learning curves for overparametrized deep neural networks: A field theory perspective. 3:023034, 2021. doi: 10.1103/PhysRevResearch.3.023034. 2

Cortes, C., Mohri, M., and Rostamizadeh, A. Algorithms for learning kernels based on centered alignment. *Journal of Machine Learning Research*, 13(28): 795–828, 2012. URL http://jmlr.org/papers/v13/cortes12a.html. I

Cui, H., Krzakala, F., and Zdeborova, L. Bayes-optimal learning of deep random networks of extensive-width. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 6468–6521. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/cui23b.html. 2

Dyer, E. and Gur-Ari, G. Asymptotics of wide networks from feynman diagrams. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=S1gFvANKDS. 2

Fischer, K., Dahmen, D., and Helias, M. Optimal signal propagation in resnets through residual scaling. (arXiv:2305.07715), May 2023. URL http://arxiv.org/abs/2305.07715. arXiv:2305.07715 [cond-mat, stat]. 4.2, 5

Garriga-Alonso, A., Rasmussen, C. E., and Aitchison, L. Deep convolutional networks as shallow gaussian processes. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=Bklfsi0cKm. 5

Geiger, M., Spigler, S., Jacot, A., and Wyart, M. Disentangling feature and lazy training in deep neural networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2020(11):113301, November 2020. ISSN 1742-5468. doi: 10.1088/1742-5468/abc4de. 1, 4.3

Geiger, M., Petrini, L., and Wyart, M. Landscape and training regimes in deep learning. 924:1–18, 2021. doi: 10.1016/j.physrep.2021.04.001. 4.3

Goldt, S., Reeves, G., Mézard, M., Krzakala, F., and Zdeborová, L. The Gaussian equivalence of generative models for learning with two-layer neural networks. 2020. URL https://arxiv.org/abs/2006.14709v1. 2

Halverson, J., Maiti, A., and Stoner, K. Neural networks and quantum field theory. *Machine Learning: Science and Technology*, 2(3):035002, apr 2021. doi: 10.1088/2632-2153/abeca3. URL https://doi.org/10.1088/2632-2153/abeca3. 2

Hanin, B. and Zlokapa, A. Bayesian interpolation with deep linear networks. 120:e2301345120, June 2023. ISSN 0027-8424, 1091-6490. doi: 10.1073/pnas.2301345120. 2

Hron, J., Bahri, Y., Sohl-Dickstein, J., and Novak, R. Infinite attention: NNGP and NTK for deep attention networks. In III, H. D. and Singh, A. (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 4376–4386. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/hron20a.html. 5

Huang, J. and Yau, H.-T. Dynamics of deep neural networks and neural tangent hierarchy. In *International Conference on Machine Learning*, pp. 4542–4551. PMLR, 2020. 2

Jacot, A., Gabriel, F., and Hongler, C. Neural tangent kernel: Convergence and generalization in neural networks. In *Advances in Neural Information Processing Systems 31*, pp. 8580–8589, 2018. URL https://proceedings.neurips.cc/paper/2018/file/5a4be1fa34e62bb8a6ec6b91d2462f5a-Paper.pdf. 1, E, E, E

Krogh, A. and Hertz, J. A simple weight decay can improve generalization. In Moody, J., Hanson, S., and Lippmann, R. (eds.), *Advances in Neural Information Processing Systems*, volume 4. Morgan-Kaufmann, 1991. URL https://proceedings.neurips.cc/paper_files/paper/1991/file/8eefcfdf5990e441f0fb6f3fad709e21-Paper.pdf. G

LeCun, Y., Cortes, C., and Burges, C. J. The mnist database of handwritten digits, 1998. 3.4

Lee, J., Bahri, Y., Novak, R., Schoenholz, S. S., Pennington, J., and Sohl-Dickstein, J. Deep neural networks as gaussian processes. pp. 1711.00165, 2017. 3.3

Lee, J., Sohl-Dickstein, J., Pennington, J., Novak, R., Schoenholz, S., and Bahri, Y. Deep neural networks as gaussian processes. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=B1EA-M-0Z. 1, 2

Lee, J., Xiao, L., Schoenholz, S., Bahri, Y., Novak, R., Sohl-Dickstein, J., and Pennington, J. Wide neural networks of any depth evolve as linear models under gradient descent. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/0d1a9651497a38d8b1c3871c84528bd4-Paper.pdf. 1, E, E, E

Lee, J., Schoenholz, S., Pennington, J., Adlam, B., Xiao, L., Novak, R., and Sohl-Dickstein, J. Finite

versus infinite neural networks: an empirical study. volume 33, pp. 15156–15172. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/ad086f59924fffe0773f8d0ca22ea712-Paper.pdf. 1

Li, Q. and Sompolinsky, H. Statistical Mechanics of Deep Linear Neural Networks: The Backpropagating Kernel Renormalization. 11(3):031059, 2021. doi: 10.1103/PhysRevX.11.031059. 2, 3, 3.4, 5, C

Lindner, J., Dahmen, D., Krämer, M., and Helias, M. A theory of data variability in neural network bayesian inference. (arXiv:2307.16695), November 2023. URL http://arxiv.org/abs/2307.16695. arXiv:2307.16695 [cond-mat, stat]. 5

Mei, S. and Montanari, A. The generalization error of random features regression: Precise asymptotics and the double descent curve. *Communications on Pure and Applied Mathematics*, 75(4): 667–766, 2022. doi: https://doi.org/10.1002/cpa.22008. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/cpa.22008. 3.4

Molgedey, L., Schuchhardt, J., and Schuster, H. Suppressing chaos in neural networks by noise. 69(26):3717, 1992. doi: 10.1103/PhysRevLett.69.3717. 2

Naveh, G. and Ringel, Z. A self consistent theory of gaussian processes captures feature learning effects in finite CNNs. 2021. URL https://openreview.net/forum?id=vBYwwBxVcsE. 2

Naveh, G., Ben-David, O., Sompolinsky, H., and Ringel, Z. Predicting the outputs of finite networks trained with noisy gradients. 2020. 3

Naveh, G., Ben David, O., Sompolinsky, H., and Ringel, Z. Predicting the outputs of finite deep neural networks trained with noisy gradients. 104: 064301, Dec 2021. doi: 10.1103/PhysRevE.104.064301. URL https://link.aps.org/doi/10.1103/PhysRevE.104.064301. 2, G

Neal, R. M. *Bayesian Learning for Neural Networks*. Springer New York, 1996. doi: 10.1007/978-1-4612-0745-0. URL https://doi.org/10.1007/978-1-4612-0745-0. 3.3

Novak, R., Xiao, L., Bahri, Y., Lee, J., Yang, G., Abolafia, D. A., Pennington, J., and Sohl-dickstein, J. Bayesian deep convolutional networks with many channels are gaussian processes. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=B1g30j0qF7. 1

Pacelli, R., Ariosto, S., Pastore, M., Ginelli, F., Gherardi, M., and Rotondo, P. A statistical mechanics framework for bayesian deep neural networks beyond the infinite-width limit. *Nature Machine Intelligence*, 5(12):1497–1507, December 2023. ISSN 2522-5839. doi: 10.1038/s42256-023-00767-6. 2, 3.4

Papoulis, A. and Pillai, S. U. *Probability, Random Variables, and Stochastic Processes*. McGraw-Hill, Boston, 4th edition, 2002. F

Pennington, J., Schoenholz, S. S., and Ganguli, S. Resurrecting the sigmoid in deep learning through dynamical isometry: theory and practice. 2017. URL https://arxiv.org/abs/1711.04735. 4.2

Petrini, L., Cagnetta, F., Vanden-Eijnden, E., and Wyart, M. Learning sparse features can lead to overfitting in neural networks. (arXiv:2206.12314), October 2022. URL http://arxiv.org/abs/2206.12314. arXiv:2206.12314 [cs, stat]. 1

Poole, B., Lahiri, S., Raghu, M., Sohl-Dickstein, J., and Ganguli, S. Exponential expressivity in deep neural networks through transient chaos. In *Advances in Neural Information Processing Systems 29*. 2016. URL https://proceedings.neurips.cc/paper/2016/file/148510031349642de5ca0c544f31b2ef-Paper.pdf. 4.2

Price, R. A useful theorem for nonlinear devices having gaussian inputs. 4(2):69–72, 1958. F

Refinetti, M., Goldt, S., Krzakala, F., and Zdeborova, L. Classifying high-dimensional gaussian mixtures: Where kernel methods fail and neural networks succeed. In Meila, M. and Zhang, T. (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8936–8947. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/refinetti21b.html. 3.4

Risken, H. *The Fokker-Planck Equation*. Springer Verlag Berlin Heidelberg, 1996. doi: 10.1007/978-3-642-61544-3_4. URL https://doi.org/10.1007/978-3-642-61544-3_4. G

Roberts, D. A., Yaida, S., and Hanin, B. *The Principles of Deep Learning Theory*. Cambridge University Press, May 2022. doi: 10.1017/9781009023405. URL https://doi.org/10.1017/9781009023405. 2

Saxe, A., Mcclelland, J., and Ganguli, S. Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. In *International Conference on Learning Represenatations*, 2014. 4.2

Schoenholz, S. S., Gilmer, J., Ganguli, S., and Sohl-Dickstein, J. Deep information propagation. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=H1W1UN9gg. 2, 3.2, 4.2, 4.2, 5, H.1

Segadlo, K., Epping, B., van Meegen, A., Dahmen, D., Krämer, M., and Helias, M. Unified field theoretical approach to deep and recurrent neuronal networks. 2022 (10):103401, 2022. 3, 3.1, 4.2, 5

Seroussi, I., Naveh, G., and Ringel, Z. Separation of scales and a thermodynamic description of feature learning in some cnns. *Nature Communications*, 14(1): 908, February 2023. ISSN 2041-1723. doi: 10.1038/s41467-023-36361-y. 2

Touchette, H. The large deviation approach to statistical mechanics. 478:1–69, 2009. 3.2, 3.2, A.2

Williams, C. Computing with infinite networks. volume 9. MIT Press, 1996. URL https://proceedings.neurips.cc/paper/1996/file/ae5e3ce40e0404a45ecacaaf05e5f735-Paper.pdf. 3.3

Yaida, S. Non-Gaussian processes and neural networks at finite widths. In Lu, J. and Ward, R. (eds.), *Proceedings of The First Mathematical and Scientific Machine Learning Conference*, volume 107 of *Proceedings of Machine Learning Research*, pp. 165–192, Princeton University, Princeton, NJ, USA, 20–24 Jul 2020. PMLR. URL http://proceedings.mlr.press/v107/yaida20a.html. 2

Yang, A. X., Robeyns, M., Milsom, E., Anson, B., Schoots, N., and Aitchison, L. A theory of representation learning gives a deep generalisation of kernel methods. In Krause, A., Brunskill, E., Cho, K., Engelhardt, B., Sabato, S., and Scarlett, J. (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 39380–39415. PMLR, Jul 2023. URL https://proceedings.mlr.press/v202/yang23k.html. 2, C, C, C

Yang, G. Wide feedforward or recurrent neural networks of any architecture are gaussian processes. volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper/2019/file/5e69fda38cda2060819766569fd93aa5-Paper.pdf. 1

Yang, G. and Hu, E. J. Feature Learning in Infinite-Width Neural Networks. 2020. URL https://arxiv.org/abs/2011.14522. 3.3, 4.3

Yang, G., Hu, E. J., Babuschkin, I., Sidor, S., Liu, X., Farhi, D., Ryder, N., Pachocki, J., Chen, W., and Gao, J. Tuning large neural networks via zero-shot hyperparameter transfer. In Beygelzimer, A., Dauphin, Y., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems*, 2021. URL https://openreview.net/forum?id=Bx6qKuBM2AD. 4.3

Zavatone-Veth, J. A., Canatar, A., Ruben, B., and Pehlevan, C. Asymptotics of representation learning in finite bayesian neural networks. 2021. URL https://openreview.net/forum?id=1oRFmD0Fl-5. 2

Zavatone-Veth, J. A., Tong, W. L., and Pehlevan, C. Contrasting random and learned features in deep bayesian linear regression. 105:064118, Jun 2022. doi: 10.1103/PhysRevE.105.064118. 2, C

Appendix

A. Detailed derivation of feature learning theory

In this appendix we present all details of the calculations leading to the self-consistency equations in Section 3. We start from the network architecture

$$h_\alpha^{(0)} = W^{(0)} x_\alpha + b^{(0)}, \tag{23}$$

$$h_\alpha^{(l)} = W^{(l)} \phi\left(h_\alpha^{(l-1)}\right) + b^{(l)} \quad l = 1, \dots, L, \tag{24}$$

$$f_\alpha = h_\alpha^{(L)}, \tag{25}$$

with Gaussian i.i.d. priors $W_{ij}^{(0)} \overset{\text{i.i.d.}}{\sim} \mathcal{N}\left(0, g_0/D\right)$, $W_{ij}^{(l)} \overset{\text{i.i.d.}}{\sim} \mathcal{N}\left(0, g_l/N\right)$, $W_i^{(L)} \overset{\text{i.i.d.}}{\sim} \mathcal{N}\left(0, g_L/N\right)$, $b_i^{(l)} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, g_b)$ on the network weights and biases. We assume that the network width N is the same across all layers $l = 0, \dots, L$. We condition on the set of training data consisting of inputs $X = (x_\alpha)_{\alpha=1,\dots,P}$ and corresponding labels $Y = (y_\alpha)_{\alpha=1,\dots,P}$.

A.1. Network prior

With the assumption of sample-wise Gaussian regularization noise κ, the prior reads

$$p(Y|X) = \int \prod_{\alpha=1}^{P} \mathcal{N}(y_\alpha | f_\alpha, \kappa)\, p(f|X)\, df. \tag{26}$$

We obtain $p(f|X)$ by enforcing the network architecture (25) using Dirac Delta distributions

$$p(f|X) = \int \mathcal{D}\{W^{(l)}, b^{(l)}\} \prod_{\alpha=1}^{P} \delta\left(-f_\alpha + W^{(L)} \phi\left(h_\alpha^{(L-1)}\right) + b^{(L)}\right) \tag{27}$$

$$\times \prod_{l=1}^{L-1} \delta\Big(-h_\alpha^{(l)} + W^{(l)}\, \phi\left(h_\alpha^{(l-1)}\right) + b^{(l)}\Big) \tag{28}$$

$$\times\, \delta\left(-h_\alpha^{(0)} + W^{(0)} x_\alpha + b^{(0)}\right), \tag{29}$$

where we use the shorthand $\mathcal{D}\{W^{(l)}, b^{(l)}\}$ to indicate the Gaussian measures given by the prior distributions on the weights and biases. In order to perform the averages, we express the Dirac Delta distributions using their Fourier transform $\delta(x) = 1/(2\pi i) \int_{-i\infty}^{i\infty} \exp(x\tilde{x})\, d\tilde{x}$, yielding

$$\delta\left(-h_{\alpha k}^{(l)} + \sum_{j=1}^{N} W_{kj}^{(l)} \phi\left(h_{\alpha j}^{(l-1)}\right) + b_k^{(l)}\right) = \int_{-i\infty}^{i\infty} \frac{d\tilde{h}_{\alpha k}^{(l)}}{2\pi i} \exp\left(-h_{\alpha k}^{(l)} \tilde{h}_{\alpha k}^{(l)} + \tilde{h}_{\alpha k}^{(l)} \sum_{j=1}^{N} W_{kj}^{(l)} \phi\left(h_{\alpha j}^{(l-1)}\right) + \tilde{h}_{\alpha k}^{(l)} b_k^{(l)}\right), \tag{30}$$

$$\delta\left(-f_\alpha + \sum_{j=1}^{N} W_j^{(L)} \phi\left(h_{\alpha j}^{(L-1)}\right) + b_k^{(L)}\right) = \int_{-i\infty}^{i\infty} \frac{d\tilde{f}_\alpha}{2\pi i} \exp\left(-f_\alpha \tilde{f}_\alpha + \tilde{f}_\alpha \sum_{j=1}^{N} W_j^{(L)} \phi\left(h_{\alpha j}^{(L-1)}\right) + \tilde{f}_\alpha b^{(L)}\right). \tag{31}$$

By doing the Fourier transform, we introduce conjugate variables $\tilde{f}$ for the network output f and $\tilde{h}^{(l)}$ for the layer activations $h^{(l)}$. Next we perform the averages over network parameters $\Theta = \{W^{(l)}, b^{(l)}\}_l$ which are i.i.d. Gaussian random variables. In doing so, we identify the moment generating function (MGF) of these variables; for a Gaussian it computes to

$\langle \exp(j\,x)\rangle_{x\sim\mathcal{N}(0,C)} = \exp\left(C/2\,j^2\right)$. Thus we get for the final layer of the network

$$\begin{aligned}
&\prod_\alpha \left\langle \exp\left(\tilde{f}_\alpha \sum_{j=1}^N W_j^{(L)}\phi_{\alpha j}^{(L-1)} + \tilde{f}_\alpha b^{(L)}\right)\right\rangle_{W^{(L)},b^{(L)}} \\
&= \left\langle \exp\left(\sum_{\alpha,j}\tilde{f}_\alpha W_j^{(L)}\phi_{\alpha j}^{(L-1)}\right)\right\rangle_{W^{(L)}} \left\langle \exp\left(\sum_\alpha \tilde{f}_\alpha b^{(L)}\right)\right\rangle_{b^{(L)}}, \qquad (32)\\
&\overset{\text{MGF}}{=} \exp\left(\frac{g_L}{2N}\sum_{\alpha\beta}\tilde{f}_\alpha\left[\sum_{j=1}^N \phi_{\alpha j}^{(L-1)}\phi_{\beta j}^{(L-1)}\right]\tilde{f}_\beta\right)\exp\left(\frac{g_b}{2}\sum_{\alpha\beta}\tilde{f}_\alpha\tilde{f}_\beta\right), \qquad (33)
\end{aligned}$$

with the shorthand $\phi_{\alpha j}^{(l-1)} = \phi\left(h_{\alpha j}^{(l-1)}\right)$. All subsequent layers yield a similar structure

$$\begin{aligned}
&\prod_\alpha \left\langle \exp\left(\tilde{h}_\alpha^{(l)} \sum_{j=1}^N W_j^{(l)}\phi_{\alpha j}^{(l-1)} + \tilde{h}_\alpha^{(l)} b^{(l)}\right)\right\rangle_{W^{(l)},b^{(l)}} \\
&= \left\langle \exp\left(\sum_{\alpha,j}\tilde{h}_\alpha^{(l)} W_j^{(l)}\phi_{\alpha j}^{(l-1)}\right)\right\rangle_{W^{(l)}} \left\langle \exp\left(\sum_\alpha \tilde{h}_\alpha^{(l)} b^{(l)}\right)\right\rangle_{b^{(l)}}, \qquad (34)\\
&\overset{\text{MGF}}{=} \exp\left(\frac{g_l}{2N}\sum_{\alpha\beta}\tilde{h}_\alpha^{(l)}\left[\phi^{(l-1)}\phi^{(l-1)\mathsf{T}}\right]_{\alpha\beta}\tilde{h}_\beta^{(l)}\right)\exp\left(\frac{g_b}{2}\sum_{\alpha\beta}\tilde{h}_\alpha^{(l)}\tilde{h}_\beta^{(l)}\right), \qquad (35)
\end{aligned}$$

where we introduced $\left[\phi^{(l-1)}\phi^{(l-1)\mathsf{T}}\right]_{\alpha\beta} := \sum_j \phi_{\alpha j}^{(l-1)}\phi_{\beta j}^{(l-1)}$. The exception is the input layer, which contains the input overlap matrix $XX^\mathsf{T} = \sum_{j=1}^D x_{\alpha j}x_{\beta j}$

$$\prod_\alpha \left\langle \exp\left(\sum_{i,j}\tilde{h}_{\alpha i}^{(0)} W_{ij}^{(0)} x_{\alpha j} + \tilde{h}_{\alpha i}^{(0)} b_i^{(0)}\right)\right\rangle_{W^{(0)},b^{(0)}} = \exp\left(\frac{g_0}{2D}\sum_{\alpha\beta i}\tilde{h}_{\alpha i}^{(0)}\left[XX^\mathsf{T}\right]_{\alpha\beta}\tilde{h}_{\beta i}^{(0)}\right)\exp\left(\frac{g_b}{2}\sum_{\alpha\beta i}\tilde{h}_{\alpha i}^{(0)}\tilde{h}_{\beta i}^{(0)}\right). \qquad (36)$$

From this we can see that introducing the auxiliary variables

$$C_{\alpha\beta}^{(0)} = \frac{g_0}{D}\left(XX^\mathsf{T}\right)_{\alpha\beta} + g_b, \qquad (37)$$

$$C_{\alpha\beta}^{(l)} = \frac{g_l}{N}\left[\phi^{(l-1)}\phi^{(l-1)\mathsf{T}}\right]_{\alpha\beta} + g_b \quad l = 1,\dots,L, \qquad (38)$$

is beneficial as we can show that

$$\int \mathcal{D}\tilde{h}_i^{(l)} \exp\left(-\tilde{h}_{\alpha i}^{(l)} h_{\alpha i}^{(l)} + \frac{1}{2}\sum_{\alpha\beta}\tilde{h}_{\alpha i}^{(l)} C_{\alpha\beta}^{(l)}\tilde{h}_{\beta i}^{(l)}\right) = \mathcal{N}\left(h_i^{(l)}|0, C_{\alpha\beta}^{(l)}\right) \quad 0 \le l < L,$$

by the Fourier representation of the Gaussian. Likewise one obtains $\mathcal{N}(f|0, C_{\alpha\beta}^{(L)})$. The form above shows that the $h_{\alpha i}^{(l)}$ are independent across neuron index i. Note that the input kernel $C^{(0)}$ is static for fixed input data sets X, whereas all subsequent auxiliary variables $C^{(l)}$ include network activations $\phi_{\alpha j}^{(L-1)}$ and are hence fluctuating. We now enforce the structure of the fluctuating auxiliary variables using Dirac Delta distributions

$$\delta\left(-C_{\alpha\beta}^{(l)} + \frac{g_l}{N}\left[\phi^{(l-1)}\phi^{(l-1)\mathsf{T}}\right]_{\alpha\beta} + g_b\right) = \int_{-i\infty}^{i\infty}\frac{d\tilde{C}_{\alpha\beta}^{(l)}}{2\pi i}\exp\left(-\tilde{C}_{\alpha\beta}^{(l)}C_{\alpha\beta}^{(l)} + \tilde{C}_{\alpha\beta}^{(l)}\frac{g_l}{N}\left[\phi^{(l-1)}\phi^{(l-1)\mathsf{T}}\right]_{\alpha\beta} + \tilde{C}_{\alpha\beta}^{(l)}g_b\right). \qquad (39)$$

Combining all those expressions in (27) yields $p(f|X)$, which depends on X only through $C^{(0)}$ given by (37)

$$p(f|X) \equiv p(f|C^{(0)}) \tag{40}$$

$$= \int \mathcal{D}\{\tilde{C}, C\} \mathcal{N}\left(f|0, C^{(L)}_{\alpha\beta}\right) \left\langle \exp(\mathcal{S}(C, \tilde{C})) \right\rangle_h, \tag{41}$$

$$\mathcal{S}(C, \tilde{C}) = -\sum_{l=1}^{L} \tilde{C}^{(l)}_{\alpha\beta} C^{(l)}_{\alpha\beta} + \tilde{C}^{(l)}_{\alpha\beta} \big(\frac{g_l}{N} \left[\phi^{(l-1)} \phi^{(l-1)\mathsf{T}}\right]_{\alpha\beta} + g_b \big),$$

where the average $\langle \dots \rangle_h$ indicates the averaging over the Gaussian distributed hidden states $\mathcal{N}\left(h^{(l)}_i|0, C^{(l)}_{\alpha\beta}\right)$ and repeated indices α, β are summed over. As these distributions are independent for each neuron index j, averaging the third line reduces to

$$\left\langle \exp\left(\tilde{C}^{(l)}_{\alpha\beta} \frac{g_l}{N} \sum_{j=1}^{N} \phi^{(l-1)}_{\alpha j} \phi^{(l-1)}_{\beta j} \right) \right\rangle_{\{\mathcal{N}\left(h^{(l-1)}_j|0, C^{(l-1)}_{\alpha\beta}\right)\}_j} \overset{h^{(l-1)}_j \text{ i.i.d. in } j}{=} \left\langle \exp\left(\frac{g_l}{N} \tilde{C}^{(l)}_{\alpha\beta} \phi^{(l-1)}_{\alpha} \phi^{(l-1)}_{\beta} \right) \right\rangle^{N}_{\mathcal{N}\left(h^{(l-1)}|0, C^{(l-1)}_{\alpha\beta}\right)}. \tag{42}$$

Overall, this yields

$$p(f|X) = \int \mathcal{D}\{\tilde{C}, C\} \mathcal{N}\left(f|0, C^{(L)}_{\alpha\beta}\right) \exp\big(-\sum_{l=1}^{L} \tilde{C}^{(l)}_{\alpha\beta} C^{(l)}_{\alpha\beta} + \mathcal{W}(\tilde{C}|C) \big), \tag{43}$$

$$\mathcal{W}(\tilde{C}|C) = \sum_{l=1}^{L} \sum_{\alpha\beta} \tilde{C}^{(l)}_{\alpha\beta} g_b + N \sum_{l=1}^{L} \ln \left\langle \exp\big(\frac{g_l}{N} \tilde{C}^{(l)}_{\alpha\beta} \phi^{(l-1)}_{\alpha} \phi^{(l-1)}_{\beta} \big) \right\rangle_{\mathcal{N}(0, C^{(l-1)})}, \tag{44}$$

$$C^{(0)}_{\alpha\beta} = \frac{g_0}{D} \left(X X^{\mathsf{T}}\right)_{\alpha\beta} + g_b. \tag{45}$$

As we are interested in the prior $p(Y|X)$ and assume Gaussian i.i.d. noise on the labels, the prior reads

$$p(Y|X) = \int \mathcal{D}\{\tilde{C}, C\} \prod_{\alpha} df_\alpha \, \mathcal{N}(y_\alpha|f_\alpha, \kappa) \mathcal{N}\left(f_\alpha|0, C^{(L)}_{\alpha\beta}\right) \exp(-\mathrm{tr}\, \tilde{C}^{\mathsf{T}} C + \mathcal{W}(\tilde{C}|C)).$$

Here we use the same shorthand as in the main text $\mathrm{tr}\, \tilde{C}^{\mathsf{T}} C = \sum_{\alpha\beta l} \tilde{C}^{(l)}_{\alpha\beta} C^{(l)}_{\alpha\beta}$. The integral over f_α has the form of a convolution of the normal distribution $\mathcal{N}(y_\alpha|f_\alpha, \kappa) \propto \exp\big(-(y_\alpha - f_\alpha)^2/(2\kappa) \big)$ with $\mathcal{N}\left(f_\alpha|0, C^{(L)}_{\alpha\beta}\right)$, which amounts to the summation of two random variables $\eta_\alpha \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \kappa)$ and $f_\alpha \sim \mathcal{N}\left(f_\alpha|0, C^{(L)}_{\alpha\beta}\right)$, so their variances add up to the variance $C^{(L)}_{\alpha\beta} + \kappa\mathbb{I}$. One therefore obtains the expression for the network prior in (2)

$$p(Y|X) = \int \mathcal{D}C \, \mathcal{N}\left(Y|0, C^{(L)} + \kappa\mathbb{I}\right) p(C), \tag{46}$$

$$p(C) \coloneqq \int \mathcal{D}\tilde{C} \, \exp\big(-\mathrm{tr}\, \tilde{C}^{\mathsf{T}} C + \mathcal{W}(\tilde{C}|C) \big). \tag{47}$$

A.2. Large deviation approach to network posterior

When writing the integrals for $p(C)$ given by (47) we see that the conditional probabilities $p\left(C^{(l)}|C^{(l-1)}\right)$ appear naturally as Fourier integrals over the conjugate variable $\tilde{C}^{(l)}$

$$p\left(C^{(l)}|C^{(l-1)}\right) = \int \mathcal{D}\tilde{C}^{(l)} \exp\left(-\mathrm{tr}\, \tilde{C}^{(l)\mathsf{T}} C^{(l)} + \mathcal{W}\left(\tilde{C}^{(l)}|C^{(l-1)}\right) \right), \tag{48}$$

$$\mathcal{W}\left(\tilde{C}^{(l)}|C^{(l-1)}\right) = \tilde{C}^{(l)} g_b + N \ln \left\langle \exp\big(\frac{g_l}{N} \phi^{(l-1)\mathsf{T}} \tilde{C}^{(l)} \phi^{(l-1)} \right\rangle_{h^{(l-1)} \sim \mathcal{N}(0, C^{(l-1)})} \quad 1 \le l < L, \tag{49}$$

where $C^{(0)}$ is given by (45). Due to the layerwise summations in (44), the full expressions $p(C)$ consists of the product of these conditional probabilities $p(C) = p\left(C^{(L)}|C^{(L-1)}\right)\cdots p\left(C^{(1)}|C^{(0)}\right)$. Computing the conditional probabilities $p\left(C^{(l)}|C^{(l-1)}\right)$ by the Fourier integral in (48) is intractable for general non-linearities. However, we can write $\mathcal{W}\left(\tilde{C}^{(l)}|C^{(l-1)}\right)$ in the form

$$\mathcal{W}\left(\tilde{C}^{(l)}|C^{(l-1)}\right) = N\,\lambda\left(\frac{\tilde{C}^{(l)}}{N}|C^{(l-1)}\right), \tag{50}$$

$$\lambda\left(K|C^{(l-1)}\right) \coloneqq K\,g_b + \ln\left\langle \exp\left(g_l\phi^{(l-1)\intercal}K\phi^{(l-1)}\right\rangle_{\mathcal{N}(0,C^{(l-1)})}. \tag{51}$$

We observe that $\mathcal{W}$ has the form of a scaled cumulant-generating function so that the limit $\lim_{N\to\infty} N^{-1}\mathcal{W}(N\,K|C^{(l-1)}) = \lambda(K|C^{(l-1)})$ exists trivially. This allows us to utilize the Gärtner-Ellis theorem to approximate $\ln p\left(C^{(l)}|C^{(l-1)}\right)$ for $N \gg 1$ (see, e.g., (Touchette, 2009), i.p. their Appendix C) by a large-deviation principle as

$$\begin{aligned} -\ln p\left(C^{(l)}|C^{(l-1)}\right) &\simeq \sup_{\tilde{C}^{(l)}} \mathrm{tr}\tilde{C}^{(l)\top}C^{(l)} - \mathcal{W}\left(\tilde{C}^{(l)}|C^{(l-1)}\right) \\ &=: \Gamma\left(C^{(l)}|C^{(l-1)}\right), \end{aligned} \tag{52}$$

with the rate function $\Gamma\left(C^{(l)}|C^{(l-1)}\right)$. We can hence approximate the full distribution $p(C)$ by

$$\ln p(C) = \ln\left(p\left(C^{(L)}|C^{(L-1)}\right)\cdots p\left(C^{(1)}|C^{(0)}\right)\right), \tag{53}$$

$$\simeq -\sum_{l=1}^{L}\Gamma\left(C^{(l)}|C^{(l-1)}\right) =: -\Gamma(C). \tag{54}$$

With the rate function we can express the prior $p(Y|X)$ as:

$$p(Y|X) \simeq \int \mathcal{D}C\mathcal{N}\left(Y|0, C^{(L)} + \kappa\mathbb{I}\right)\exp\left(-\Gamma(C)\right), \tag{55}$$

From the supremum condition (52) and by evaluating the integral

$$p(Y|X) \simeq \int \mathcal{D}C\exp\left(\mathcal{S}(C)\right), \tag{56}$$

$$\mathcal{S}(C) := \ln p(C|Y) \overset{\text{l.d.p.}}{\simeq} \mathcal{S}_{\mathrm{D}}(C^{(L)}) - \Gamma(C), \tag{57}$$

$$\mathcal{S}_{\mathrm{D}}(C^{(L)}) := -\frac{1}{2}Y^{\top}(C^{(L)} + \kappa\mathbb{I})^{-1}Y - \frac{1}{2}\ln\det(C^{(L)} + \kappa\mathbb{I}), \tag{58}$$

in a saddle-point approximation in $C^{(l)}$, we obtain the equations for the network kernels $C^{(l)}$ and the conjugate kernels $\tilde{C}^{(l)}$.

A.3. Maximum a posteriori network kernels $C^{(l)}$

The definition of $\Gamma(C^{(l)}|C^{(l-1)})$ (52) enforces the supremum in $\tilde{C}^{(l)}$. Hence we require stationarity in $\tilde{C}^{(l)}$

$$\frac{\partial}{\partial\tilde{C}^{(l)}}\left[\mathrm{tr}\tilde{C}^{(l)\top}C^{(l)} - \mathcal{W}\left(\tilde{C}^{(l)}|C^{(l-1)}\right)\right] \overset{!}{=} 0$$

to obtain the supremum as

$$C^{(l)}_{\alpha\beta} - \frac{\partial\mathcal{W}}{\partial\tilde{C}^{(l)}_{\alpha\beta}} = 0 \tag{59}$$

$$\to C^{(l)}_{\alpha\beta} = g_l\left\langle\phi^{(l-1)}_{\alpha}\phi^{(l-1)}_{\beta}\right\rangle_{\mathcal{P}^{(l-1)}} + g_b \tag{60}$$

where we used the form of $\mathcal{W}$ in (49) and introduced $\langle \ldots \rangle_{\mathcal{P}^{(l)}}$ to indicate averages over the measure $\mathcal{P}^{(l)}$, which is given by

$$\langle \ldots \rangle_{\mathcal{P}^{(l)}} = \frac{\left\langle \ldots \exp\left(\frac{g_l}{N}\phi^{(l)\mathsf{T}}\tilde{C}^{(l+1)}\phi^{(l)}\right)\right\rangle_{\mathcal{N}(0,C^{(l)})}}{\left\langle \exp\left(\frac{g_l}{N}\phi^{(l)\mathsf{T}}\tilde{C}^{(l+1)}\phi^{(l)}\right)\right\rangle_{\mathcal{N}(0,C^{(l)})}}. \tag{61}$$

This expression corresponds to 7 in the main text.

A.4. Self-consistent conjugate kernels $\tilde{C}^{(l)}$

By performing the saddle-point approximation of (56) in $C^{(l)}$, we obtain expressions for $\tilde{C}^{(l)}$. We first need two fundamental properties that follow from the Legendre transform in the definition of $\Gamma(C^{(l)}|C^{(l-1)})$: The first is the equation of state

$$\frac{\partial}{\partial C^{(l)}}\Gamma(C^{(l)}|C^{(l-1)}) = \tilde{C}^{(l)}, \tag{62}$$

which follows because the supremum condition yields

$$\begin{aligned}&\frac{\partial}{\partial C^{(l)}}\sup_{\tilde{C}^{(l)}} \mathrm{tr}\tilde{C}^{(l)\mathsf{T}}C^{(l)} - \mathcal{W}(\tilde{C}^{(l)}|C^{(l-1)})\\ =&\tilde{C}^{(l)} + \mathrm{tr}\frac{\partial\tilde{C}^{(l)\mathsf{T}}}{\partial C^{(l)}}C^{(l)} - \mathrm{tr}\underbrace{\frac{\partial\mathcal{W}(\tilde{C}^{(l)}|C^{(l-1)})^{\mathsf{T}}}{\partial\tilde{C}^{(l)}}}_{\equiv C^{(l)\mathsf{T}}}\frac{\partial\tilde{C}^{(l)}}{\partial C^{(l)}}\end{aligned}$$

so that the latter two terms cancel each other.

The second fundamental property of the Legendre transform applies to the derivative by $C^{(l-1)}$, which here plays the role of a parameter, for which holds

$$\frac{\partial}{\partial C^{(l-1)}}\Gamma(C^{(l)}|C^{(l-1)}) = -\frac{\partial\mathcal{W}(\tilde{C}^{(l)}|C^{(l-1)})}{\partial C^{(l-1)}}, \tag{63}$$

again, because of the supremum condition

$$\begin{aligned}&\frac{\partial}{\partial C^{(l-1)}}\sup_{\tilde{C}^{(l)}} \mathrm{tr}\tilde{C}^{(l)\mathsf{T}}C^{(l)} - \mathcal{W}(\tilde{C}^{(l)}|C^{(l-1)})\\ &= \mathrm{tr}\frac{\partial\tilde{C}^{(l)\mathsf{T}}}{\partial C^{(l-1)}}C^{(l)} - \mathrm{tr}\underbrace{\frac{\partial\mathcal{W}^{\mathsf{T}}}{\partial\tilde{C}^{(l)}}}_{\equiv C^{(l)}}\frac{\partial\tilde{C}^{(l)}}{\partial C^{(l-1)}} - \frac{\partial\mathcal{W}}{\partial C^{(l-1)}},\end{aligned}$$

the first two terms on the right hand side cancel. The stationary points of (57) for $1 \le l < L$ then follow with the explicit form of $\Gamma(C) = \sum_{l=1}^{L}\Gamma\left(C^{(l)}|C^{(l-1)}\right)$ from (54) as

$$\begin{aligned}0 \overset{!}{=} \frac{\partial\mathcal{S}(C)}{\partial C^{(l)}} &= \frac{\partial}{\partial C^{(l)}}\sum_{l=1}^{L}\Gamma\left(C^{(l)}|C^{(l-1)}\right)\\ &= \frac{\partial\Gamma(C^{(l)}|C^{(l-1)})}{\partial C^{(l)}} + \frac{\partial\Gamma(C^{(l+1)}|C^{(l)})}{\partial C^{(l)}}\\ &\overset{(62),(63)}{=} \tilde{C}^{(l)} - \frac{\partial\mathcal{W}(\tilde{C}^{(l+1)}|C^{(l)})}{\partial C^{(l)}}.\end{aligned} \tag{64}$$

Using the definition of $\mathcal{W}$, we compute

$$\frac{\partial}{\partial C^{(l)}_{\alpha\beta}}\mathcal{W}(\tilde{C}^{(l+1)}|C^{(l)}) = N\frac{\frac{\partial}{\partial C^{(l)}_{\alpha\beta}}\left\langle \exp\left(\frac{g_{l+1}}{N}\phi^{(l)\mathsf{T}}\tilde{C}^{(l+1)}\phi^{(l)}\right)\right\rangle_{\mathcal{N}(0,C^{(l)})}}{\left\langle \exp\left(\frac{g_{l+1}}{N}\phi^{(l)\mathsf{T}}\tilde{C}^{(l+1)}\phi^{(l)}\right)\right\rangle_{\mathcal{N}(0,C^{(l)})}} \tag{65}$$

by Price's theorem (see F): the derivative in the numerator is

$$\frac{\partial}{\partial C^{(l)}_{\alpha\beta}}\Big\langle \exp\Big(\frac{g_{l+1}}{N}\phi^{(l)\intercal}\tilde{C}^{(l+1)}\phi^{(l)}\Big)\Big\rangle_{h^{(l)}\sim\mathcal{N}(0,C^{(l)})} \overset{(108)}{=} \frac{1}{2}\Big\langle \frac{\partial}{\partial h^{(l)}_{\alpha}\partial h^{(l)}_{\beta}}\exp\left(\frac{g_{l+1}}{N}\sum_{\gamma\delta}\phi^{(l)}_{\gamma}\tilde{C}^{(l+1)}_{\gamma\delta}\phi^{(l)}_{\delta}\right)\Big\rangle_{h^{(l)}\sim\mathcal{N}(0,C^{(l)})}, \tag{66}$$

which yields with (64) for $\tilde{C}^{(l)}$ with $1\le l<L$

$$\begin{aligned}\tilde{C}^{(l)}_{\alpha\beta} = \frac{\partial\mathcal{W}(\tilde{C}^{(l+1)}|C^{(l)})}{\partial C^{(l)}_{\alpha\beta}} = g_{l+1}\left[\Big\langle \Big(\phi^{(l)}_{\alpha}\Big)'\Big(\phi^{(l)}_{\beta}\Big)'\Big\rangle_{\mathcal{P}^{(l)}}\tilde{C}^{(l+1)}_{\alpha\beta} + \delta_{\alpha\beta}\sum_{\gamma}\Big\langle \phi^{(l)}_{\gamma}\Big(\phi^{(l)}_{\alpha}\Big)''\Big\rangle_{\mathcal{P}^{(l)}}\tilde{C}^{(l+1)}_{\gamma\alpha}\right]\\ + 2\frac{g^2_{l+1}}{N}\sum_{\gamma,\delta}\tilde{C}^{(l+1)}_{\alpha\gamma}\tilde{C}^{(l+1)}_{\beta\delta}\Big\langle \Big(\phi^{(l)}_{\alpha}\Big)'\phi^{(l)}_{\gamma}\Big(\phi^{(l)}_{\beta}\Big)'\phi^{(l)}_{\delta}\Big\rangle_{\mathcal{P}^{(l)}},\end{aligned} \tag{67}$$

with measure $\mathcal{P}^{(l)}$ as defined in (61). In the main text in (11), we only keep terms $\propto\tilde{C}$ on the right hand sides of (67). In the case $l=L$, we additionally need to consider the data-term in (56), yielding

$$0 \overset{!}{=} \frac{\partial\mathcal{S}(C)}{\partial C^{(L)}_{\alpha\beta}} = \frac{\partial}{\partial C^{(L)}_{\alpha\beta}}\big(\mathcal{S}_{\mathrm{D}}(C^{(L)}) - \Gamma(C^{(L)}|C^{(L-1)})\big) \tag{68}$$

$$\overset{(62)}{=} \frac{1}{2}\Big((C^{(L)}+\kappa\mathbb{I})^{-1}YY^{\intercal}(C^{(L)}+\kappa\mathbb{I})^{-1} - (C^{(L)}+\kappa\mathbb{I})^{-1}\Big) - \tilde{C}^{(L)}, \tag{69}$$

which is the result (10) in the main text. To compute $\partial\mathcal{S}_{\mathrm{D}}(C^{(L)})/\partial C^{(L)}$ with (58) we used the matrix derivatives

$$\frac{\partial\ln\det(C)}{\partial C_{\alpha\beta}} = \big[C\big]^{-1}_{\alpha\beta}, \tag{70}$$

$$\frac{\partial\big[C\big]^{-1}_{\gamma\delta}}{\partial C_{\alpha\beta}} = -\big[C\big]^{-1}_{\gamma\alpha}\big[C\big]^{-1}_{\beta\delta}, \tag{71}$$

yielding

$$\tilde{C}^{(L)} = \frac{1}{2}(C^{(L)}+\kappa\mathbb{I})^{-1}(YY^{\intercal})(C^{(L)}+\kappa\mathbb{I})^{-1} - \frac{1}{2}(C^{(L)}+\kappa\mathbb{I})^{-1}. \tag{72}$$

A.5. Relation between conjugate kernel and discrepancy

To understand the meaning of the conjugate kernel $\tilde{C}$, we generalize the regularization $\kappa\mathbb{I}$ with a generic covariance matrix $K_{\alpha\beta}$ in (46)

$$p(Y|X,K) := \int\mathcal{D}C\int\mathcal{D}f\,\mathcal{N}(Y|f,K)\,\mathcal{N}(f|0,C^{(L)})\,p(C), \tag{73}$$

which shows that, given $C^{(L)}$, the statistics of Y is a convolution of two centered Gaussian distributions with covariances $C^{(L)}$ and K, respectively. In large deviation theory, this yields the action

$$S(C|K) = -\frac{1}{2}y_{\alpha}\big[C^{(L)}+K\big]^{-1}_{\alpha\beta}y_{\beta} - \frac{1}{2}\ln\det(C+K) - \Gamma(C). \tag{74}$$

Writing (73) explicitly

$$p(Y|X,K) = \frac{1}{(2\pi)^{\frac{M}{2}}(\det K)^{\frac{1}{2}}}\int\mathcal{D}C\int\mathcal{D}f\,\exp\big(-\frac{1}{2}(y_{\alpha}-f_{\alpha})\big[K^{-1}\big]_{\alpha\beta}(y_{\beta}-f_{\beta})\big)\,\mathcal{N}(f|0,C^{(L)})\,p(C),$$

we may use K^{-1} as a bi-linear source term so that we obtain within the MAP approximation for the kernels C, the second moment of the discrepancies as

$$
\begin{aligned}
-\frac{1}{2}\left\langle (y_\alpha - f_\alpha)\,(y_\beta - f_\beta)\right\rangle &= \frac{\partial}{\partial [K^{-1}]_{\alpha\beta}}\left(\ln p(Y|X,K) - \frac{1}{2}\det K^{-1}\right)\Big|_{K=\kappa\mathbb{I}} \\
&\overset{\text{MAP}}{\simeq} \frac{\partial}{\partial [K^{-1}]_{\alpha\beta}} S(C|K) + \underbrace{\frac{\partial S}{\partial C}}_{=0}\frac{\partial C}{\partial (K^{-1})_{\alpha\beta}} - \frac{1}{2}K\Big|_{K=\kappa\mathbb{I}} \\
&\overset{(74)}{=} \Big[-\frac{1}{2}K\left[C^{(L)}+K\right]^{-1} YY^\intercal \left[C^{(L)}+K\right]^{-1} K + \frac{1}{2}K\left(C^{(L)}+K\right)^{-1}K\Big]_{\alpha\beta} - \frac{1}{2}K\Big|_{K=\kappa\mathbb{I}} \\
&= \Big[-\frac{1}{2}\kappa^2\left[C^{(L)}+\kappa\mathbb{I}\right]^{-1}(YY^\intercal)\left[C^{(L)}+\kappa\mathbb{I}\right]^{-1} + \frac{1}{2}\kappa^2\left(C^*+\kappa\mathbb{I}\right)^{-1} - \frac{1}{2}\kappa\mathbb{I}\Big]_{\alpha\beta} \\
&\overset{(72)}{=} -\kappa^2\tilde{C}^*_{\alpha\beta} - \frac{1}{2}\kappa\delta_{\alpha\beta},
\end{aligned}
\tag{75}
$$

where we used that $\partial [K^{-1}]_{\gamma\delta}/\partial K_{\alpha\beta} = -K^{-1}_{\gamma\alpha}\,K^{-1}_{\beta\delta}$ and by symmetry $\partial K_{\gamma\delta}/\partial [K^{-1}]_{\alpha\beta} = -K_{\gamma\alpha}\,K_{\beta\delta}$. This may also be rewritten as

$$
\begin{aligned}
\left\langle \Delta_\alpha\,\Delta_\beta\right\rangle &= 2\kappa^2\tilde{C}^{(L)}_{\alpha\beta} + \kappa\delta_{\alpha\beta}, \\
\Delta_\alpha &= y_\alpha - f_\alpha,
\end{aligned}
\tag{76}
$$

so that the expected training error is

$$
\begin{aligned}
\left\langle \mathcal{L}\right\rangle :=& \frac{1}{2}\mathrm{tr}\left\langle \Delta\,\Delta\right\rangle \\
=& \kappa^2\,\mathrm{tr}\tilde{C}^{(L)} + \frac{1}{2}\kappa P.
\end{aligned}
\tag{77}
$$

Expressions (76) and (77) show that the conjugate kernel $\tilde{C}$ is, apart from the diagonal term, given by the expected squared discrepancies Δ between target and network output.

B. Relation between feature learning and fluctuation corrections

We here show that the shift of the saddle point of $C^{(l)}$ by conditioning on the training data can be regarded as accounting for fluctuation corrections for the auxiliary variables C and $\tilde{C}$ around the reference point, which is the NNGP. As opposed to the rigorous approach of the main text that is based on a large deviation principle, we here obtain the result from a perspective of field theory. To this end, we rewrite the network prior (2) as an integral over the pair of fields $(C,\tilde{C})$ as in (46)

$$
\begin{aligned}
p(Y|X) &= \int DC \int D\tilde{C}\,\mathcal{N}(Y|0,C^{(L)}+\kappa\mathbb{I})\,\exp\left(S(C,\tilde{C})\right), \\
\mathcal{S}(C,\tilde{C}) &:= -\mathrm{tr}\,\tilde{C}^\intercal C + \mathcal{W}(\tilde{C}|C),
\end{aligned}
\tag{78}
$$

with cumulant-generating function $\mathcal{W}$ given by (4). Adding the normal distribution in (78) as $\exp(\mathcal{S}_\mathrm{D})$ one has a joint measure for the pair of variables $(C,\tilde{C})$ which is, up to normalization, given by

$$
\begin{aligned}
(C,\tilde{C}) &\sim \exp\left(\mathcal{S}(C,\tilde{C}) + \mathcal{S}_\mathrm{D}(C^{(L)}|Y)\right) \\
\mathcal{S}_\mathrm{D}(C^{(L)}|Y) &:= \ln\mathcal{N}(Y|0,C^{(L)}+\kappa\mathbb{I}) \\
&\equiv -\frac{1}{2}Y^\intercal(C^{(L)}+\kappa\mathbb{I})^{-1}Y - \frac{1}{2}\ln\det(C^{(L)}+\kappa\mathbb{I}).
\end{aligned}
$$

Now we will expand $S(C,\tilde{C})$ around its saddle point, which, with regard to $\tilde{C}^{(l)}$ for $1\le l\le L$ yields

$$
\begin{aligned}
0 \overset{!}{=} \frac{\partial\mathcal{S}}{\partial\tilde{C}^{(l)}_{\alpha\beta}} &= -C^{(l)}_{\alpha\beta} + \frac{\partial\mathcal{W}}{\partial\tilde{C}^{(l)}_{\alpha\beta}} \\
&= -C^{(l)}_{\alpha\beta} + g_l\left\langle \phi^{(l-1)}_\alpha\phi^{(l-1)}_\beta\right\rangle_{\mathcal{P}^{(l-1)}} + g_b,
\end{aligned}
\tag{79}
$$

and the initial value $C^{(0)}$ given by (5) for $l = 0$. This result is of course identical to (7), as it corresponds to the supremum condition in (6). The second saddle point equation for $C^{(L)}$ yields

$$0 \stackrel{!}{=} \frac{\partial \mathcal{S}}{\partial C^{(L)}_{\alpha\beta}} = -\tilde{C}^{(L)}_{\alpha\beta}, \tag{80}$$

and for $1 \leq l < L$ the equation is given by (11) of the main text. Together this shows by induction that the stationary point is $\tilde{C}^{(1\leq l\leq L)}_* \equiv 0$; so the measure $\mathcal{P}^{(l-1)}$ appearing in (79) reduces to a Gaussian $\mathcal{N}(0, C^{(l-1)})$ and the propagation of $C^{(l)}$ over layers becomes the NNGP, whose solution we will denote as $C^{(l)}_*$ and $\tilde{C}^{(l)}_* \equiv 0$.

Computing the next-to-leading-order in N^{-1}, we need the Hessian of the action $\mathcal{S}$, evaluated at the NNGP saddle point, which is

$$\mathcal{S}^{(2)(l,m)}_{(\alpha\beta)(\gamma\delta)}\Big|_{C_*,\tilde{C}\equiv 0} = \begin{pmatrix} \frac{\partial^2 \mathcal{S}}{\partial C^{(l)}_{\alpha\beta}\partial C^{(m)}_{\gamma\delta}} & \frac{\partial^2 \mathcal{S}}{\partial C^{(l)}_{\alpha\beta}\partial \tilde{C}^{(m)}_{\gamma\delta}} \\ \frac{\partial^2 \mathcal{S}}{\partial \tilde{C}^{(l)}_{\alpha\beta}\partial C^{(m)}_{\gamma\delta}} & \frac{\partial^2 \mathcal{S}}{\partial \tilde{C}^{(l)}_{\alpha\beta}\partial \tilde{C}^{(m)}_{\gamma\delta}} \end{pmatrix} \tag{81}$$

$$= \begin{pmatrix} 0 & -\delta_{l,m}\,\delta_{(\alpha\beta),(\gamma\delta)} + \delta_{m-1,l}\, g_m \frac{\partial \langle \phi^{(m-1)}_\gamma \phi^{(m-1)}_\delta \rangle_{\mathcal{N}(0,C^{(l)}_*)}}{\partial C^{(l)}_{\alpha\beta}} \\ -\delta_{l,m}\,\delta_{(\alpha\beta),(\gamma\delta)} + \delta_{l-1,m}\, g_l \frac{\partial \langle \phi^{(l-1)}_\alpha \phi^{(l-1)}_\beta \rangle_{\mathcal{N}(0,C^{(l)}_*)}}{\partial C^{(m)}_{\gamma\delta}} & \delta_{l,m}\, g_l^2 \,\langle \phi^{(l-1)}_\alpha \phi^{(l-1)}_\beta, \phi^{(l-1)}_\gamma \phi^{(l-1)}_\delta \rangle_{c,\mathcal{N}(0,C^{(l)}_*)} \end{pmatrix},$$

where we used that $\mathcal{W}(0|C) \equiv 1 \quad \forall C$, so that its derivative in the upper left element vanishes and all expectation values $\langle \ldots \rangle_{\mathcal{N}(0,C^{(l)}_*)}$ are with regard to the Gaussian measure of the NNGP. Since the expectation values of products of $\phi^{(l)}$ only depend on the value of $C^{(l)}$ in the same layer, the non-trivial terms in the off-diagonal elements are proportional to Kronecker symbols $\delta_{m-1,l}$ (upper right) and $\delta_{l-1,m}$ (lower left). The lower right element contains the connected two-point correlation function $\langle f, g\rangle_c := \langle fg\rangle - \langle f\rangle\langle g\rangle$, coming from the second derivative by $\tilde{C}$; derivatives by $\tilde{C}^{(l)}$ and $\tilde{C}^{(m)}$ with $l \neq m$ vanish, because the cumulant-generating function (4) decomposes into a sum of cumulant-generating functions across layers, showing their statistical independence, so that connected correlations across layers vanish.

Within this expansion, the network prior (78) takes the form

$$p(Y|X) \simeq \int D\delta C \int D\delta\tilde{C}\, \exp\Big(\frac{1}{2}(\delta C, \delta\tilde{C})^{\mathsf{T}} \mathcal{S}^{(2)}\, (\delta C, \delta\tilde{C}) + \mathcal{S}_{\mathrm{D}}(C^{(L)}_* + \delta C^{(L)}|Y)\Big) \tag{82}$$

because for $\tilde{C}_* \equiv 0$ the zeroth order Taylor term $\mathcal{S}(C_*, 0) \equiv 0$ and also the linear term vanishes, because C_* has been chosen as the stationary point. We may now study the influence of $\mathcal{S}_{\mathrm{D}}$ on the saddle point of δC and $\delta\tilde{C}$. This term only affects the saddle point through its affect on $\delta C^{(L)}$, namely like a source term $\mathrm{tr} J^{\mathsf{T}}\, \delta C^{(L)}$ with

$$J_{\alpha\beta} := \frac{\partial \mathcal{S}_{\mathrm{D}}}{\partial C^{(L)}_{\alpha\beta}}. \tag{83}$$

So the saddle point equation for the shift $(\delta C, \delta\tilde{C})$ of the saddle points reads

$$\Big[\mathcal{S}^{(2)} \begin{pmatrix} \delta C \\ \delta\tilde{C} \end{pmatrix}\Big]^{(l)} + \begin{pmatrix} J \\ 0 \end{pmatrix} \delta_{l\,L} = 0, \tag{84}$$

Written explicitly with help of the Hessian (81), the first line of (84) therefore reads

$$-\delta\tilde{C}^{(l)}_{\alpha\beta} + g_{l+1} \sum_{\gamma\delta} \frac{\partial \langle \phi^{(l)}_\gamma \phi^{(l)}_\delta \rangle_{\mathcal{N}(0,C^{(l)})}}{\partial C^{(l)}_{\alpha\beta}}\, \delta\tilde{C}^{(l+1)}_{\gamma\delta} + \delta_{l,L}\, J_{\alpha\beta} = 0. \tag{85}$$

Employing Price's theorem (108) one has

$$\begin{aligned}
&\sum_{\gamma\delta} \frac{\partial \langle \phi_\gamma^{(l)} \phi_\delta^{(l)} \rangle_{\mathcal{N}(0,C^{(l)})}}{\partial C_{\alpha\beta}^{(l)}} \, \delta \tilde{C}_{\gamma\delta}^{(l+1)} \\
&= \frac{1}{2} \sum_{\gamma\delta} \Big\langle \frac{\partial}{\partial h_\alpha^{(l)}} \frac{\partial}{\partial h_\beta^{(l)}} \, \phi_\gamma^{(l)} \phi_\delta^{(l)} \Big\rangle_{\mathcal{N}(0,C^{(l)})} \, \delta \tilde{C}_{\gamma\delta}^{(l+1)} \\
&= \big\langle \big(\phi_\alpha^{(l)}\big)' \big(\phi_\beta^{(l)}\big)' \big\rangle_{\mathcal{N}(0,C^{(l)})} \, \delta \tilde{C}_{\alpha\beta}^{(l+1)} \\
&+ \delta_{\alpha\beta} \sum_{\gamma} \big\langle \big(\phi_\alpha^{(l)}\big)'' \big(\phi_\gamma^{(l)}\big) \big\rangle_{\mathcal{N}(0,C^{(l)})} \, \delta \tilde{C}_{\alpha\gamma}^{(l+1)},
\end{aligned}$$

where we used that $\tilde{C}$ and C are both symmetric. Inserted into (85), we obtain, to linear order in $\tilde{C}$, the same propagation equation as stated in (11).

The second line of (84) reads explicitly

$$-\delta C_{\alpha\beta}^{(l)} + g_l \sum_{\gamma\delta} \frac{\partial \langle \phi_\alpha^{(l-1)} \phi_\beta^{(l-1)} \rangle_{\mathcal{N}(0,C^{(l-1)})}}{\partial C_{\gamma\delta}^{(l-1)}} \, \delta C_{\gamma\delta}^{(l-1)} + g_l^2 \sum_{\gamma\delta} \langle \phi_\alpha^{(l-1)} \phi_\beta^{(l-1)}, \phi_\gamma^{(l-1)} \phi_\delta^{(l-1)} \rangle_{c,\mathcal{N}(0,C^{(l-1)})} \, \delta \tilde{C}_{\gamma\delta}^{(l)} = 0.$$

Rewritten, this reads

$$\delta C_{\alpha\beta}^{(l)} = g_l \sum_{\gamma\delta} \frac{\partial \langle \phi_\alpha^{(l-1)} \phi_\beta^{(l-1)} \rangle_{\mathcal{N}(0,C^{(l-1)})}}{\partial C_{\gamma\delta}^{(l-1)}} \, \delta C_{\gamma\delta}^{(l-1)} + g_l^2 \sum_{\gamma\delta} V_{\alpha\beta,\gamma\delta}^{(l-1)} \, \delta \tilde{C}_{\gamma\delta}^{(l)},$$

where we used the definition $V_{\alpha\beta,\gamma\delta}^{(l-1)} \equiv \langle \phi_\alpha^{(l-1)} \phi_\beta^{(l-1)}, \phi_\gamma^{(l-1)} \phi_\delta^{(l-1)} \rangle_{c,\mathcal{N}(0,C^{(l-1)})}$ from (15). The first term on the right hand side yields the linear correction to δC due to the shift δC in (79) and the second term is identical to the correction from $\delta \tilde{C}$ in (15). This shows that the self-consistency equations derived in the main text, up to linear order in δC, are identical to taking fluctuations of C up to Gaussian order into account. This allows us to link points where these fluctuations are large, critical points, to feature learning.

C. Deep linear networks

In this section we consider the special case of a deep linear network to make the connection to previous works (Li & Sompolinsky, 2021; Zavatone-Veth et al., 2022; Yang et al., 2023). This case allows us to obtain closed-form expressions for both, the forward iteration equation (7) and the backward equation (11); these expressions in particular do not require us to apply a perturbative treatment but only rest on the use of the large deviation principle.

We also show here that in the case of a linear network, our action (9), valid for general non-linear networks, reduces to the one by (Yang et al., 2023), their Eq. (1), derived for deep kernel machines. This new result has three main implications:

1. It shows that in the proportional limit $P = \alpha N$ considered here, deep linear networks reduce to deep kernel machines.
2. The here found iterative forward-backward equations may also be used to determine the MAP estimate for the kernels in deep linear neural networks and in deep kernel machines.
3. It allows us to provide the alternative view of kernel adaptation generated by kernel fluctuations, as outlined in (4), also for deep linear networks and deep kernel machines; this point is useful, because it shows how the corrections found in the proportional $P = \alpha N$, $N \to \infty$, apply to networks at finite size.

The derivation here will follow along the same steps as in the general non-linear case in Appendix A. We will here only point out the important differences. The setting of a deep linear neural network here is (25), only replacing the activation function by the identity $\phi = \mathrm{id}$. This change only affects the mapping for the hidden layers which are now

$$h_\alpha^{(l)} = W^{(l)} \, h_\alpha^{(l-1)} + b^{(l)} \quad l = 1, \dots, L.$$

Moreover, we use the same prior distributions for all parameters as in the general case. Following the same steps as in Section A.1, the network prior corresponding to (44) of the non-linear network takes the following form for the linear network

$$p(f|X) = \int \mathcal{D}\{\tilde{C}, C\}\, \mathcal{N}\left(f|0, C^{(L)}_{\alpha\beta}\right) \exp\big(-\sum_{l=1}^{L} \tilde{C}^{(l)}_{\alpha\beta} C^{(l)}_{\alpha\beta} + \mathcal{W}(\tilde{C}|C)\big), \tag{86}$$

$$\mathcal{W}(\tilde{C}|C) = \sum_{l=1}^{L} \sum_{\alpha\beta} \tilde{C}^{(l)}_{\alpha\beta} g_b + N \sum_{l=1}^{L} \ln \left\langle \exp\big(\frac{g_l}{N} \tilde{C}^{(l)}_{\alpha\beta} h^{(l-1)}_{\alpha} h^{(l-1)}_{\beta}\big)\right\rangle_{\mathcal{N}(0, C^{(l-1)})}, \tag{87}$$

$$C^{(0)}_{\alpha\beta} = \frac{g_0}{D} \left(X X^{\mathsf{T}}\right)_{\alpha\beta} + g_b. \tag{88}$$

In contrast to the non-linear case, here the expectation value in the definition of $\mathcal{W}(\tilde{C}|C)$ is a Gaussian integral with the closed-form solution

$$\mathcal{W}(\tilde{C}|C) = \sum_{l=1}^{L} \mathcal{W}(\tilde{C}^{(l)}|C^{(l-1)}) \tag{89}$$

$$\begin{aligned}
\mathcal{W}(\tilde{C}^{(l)}|C^{(l-1)}) :=& \sum_{\alpha\beta} \tilde{C}^{(l)}_{\alpha\beta} g_b + N \ln \left\langle \exp\big(\frac{g_l}{N} \tilde{C}^{(l)}_{\alpha\beta} h^{(l-1)}_{\alpha} h^{(l-1)}_{\beta}\big)\right\rangle_{\mathcal{N}(0, C^{(l-1)})} \\
=& \sum_{\alpha\beta} \tilde{C}^{(l)}_{\alpha\beta} g_b - \frac{N}{2} \ln\det\, \big([C^{(l-1)}]^{-1} - 2\frac{g_l}{N}\, \tilde{C}^{(l)}\big) - \frac{N}{2} \ln\det(C^{(l-1)}).
\end{aligned}$$

Since this cumulant-generating function has the scaling form so that the limit $\lambda(k) := \lim_{N\to\infty} \mathcal{W}(Nk)/N$ exists trivially, we may employ the Gärtner-Ellis theorem to approximate the conditional probabilities $p(C^{(l)}|C^{(l-1)})$ (cf. (48)) by a rate function Γ

$$\begin{aligned}
-\ln p\left(C^{(l)}|C^{(l-1)}\right) :=& -\int \mathcal{D}\tilde{C}^{(l)} \exp\left(-\mathrm{tr}\, \tilde{C}^{(l)\mathsf{T}} C^{(l)} + \mathcal{W}\left(\tilde{C}^{(l)}|C^{(l-1)}\right)\right) \\
\overset{\text{l.d.p.}}{\simeq}& \Gamma(C^{(l)}|C^{(l-1)}) \\
:=& \sup_{\tilde{C}^{(l)}} \mathrm{tr} \tilde{C}^{(l)\top} C^{(l)} - \mathcal{W}\left(\tilde{C}^{(l)}|C^{(l-1)}\right) \\
=& \frac{N}{2g_l} \mathrm{tr}\big([C^{(l-1)}]^{-1}(C^{(l)} - g_b)\big) - \frac{N}{2} \ln\det\, \big(C^{(l)} - g_b\big) + \frac{N}{2} \ln\det(C^{(l-1)}) + \text{const.},
\end{aligned} \tag{90}$$

where we dropped terms that are constant in the C and used that the supremum condition in the penultimate line evaluates to

$$\begin{aligned}
0 \overset{!}{=}& \frac{\partial}{\partial \tilde{C}^{(l)}_{\alpha\beta}} \Big(\mathrm{tr} \tilde{C}^{(l)\top} \tilde{C}^{(l)} - \mathcal{W}\left(\tilde{C}^{(l)}|C^{(l-1)}\right)\Big) \\
=& C^{(l)}_{\alpha\beta} - g_b - g_l \left([C^{(l-1)}]^{-1} - 2\frac{g_l}{N}\, \tilde{C}^{(l)}\right)^{-1}_{\alpha\beta}.
\end{aligned} \tag{91}$$

This yields the equation to propagate the kernels C forward through the network (corresponding to (7) and (60) in the non-linear case), which, for $\tilde{C} = 0$, again reduces to the NNGP result as expected. Solved for $\tilde{C}$ the latter yields

$$\tilde{C}^{(l)} = \frac{N}{2} \left([g_l C^{(l-1)}]^{-1} - [C^{(l)} - g_b]^{-1}\right),$$

which, inserted into the penultimate line of (90) yields the last line there.

At this point we are able to make the connection to deep kernel machines studied in (Yang et al., 2023). The action for the kernels C, corresponding to (9) in the non-linear case, in the case of the linear networks with (87) and (90) takes the form

$$\begin{aligned}
\mathcal{S}(C) :=& \ln p(C|Y) \overset{\text{l.d.p.}}{\simeq} \mathcal{S}_{\mathrm{D}}(C^{(L)}) - \Gamma(C) + \circ\,, \\
\Gamma(C) =& \sum_{l=1}^{L} \Gamma(C^{(l)}|C^{(l-1)}).
\end{aligned} \tag{92}$$

The specific form of the rate function $\Gamma(C^{(l)}|C^{(l-1)})$ in (90) is that of a Kullback-Leibler divergence between two pairs of centered Gaussian covariates with $\langle z_{\alpha i}^{(l-1)} z_{\beta j}^{(l-1)} \rangle = \delta_{ij} g_l C^{(l-1)}$ and $\langle z_{\alpha i}^{(l)} z_{\beta j}^{(l)} \rangle = \delta_{ij} \big(C^{(l)} - g_b\big)$, respectively, namely

$$\begin{aligned} &\mathrm{KL}(\mathcal{N}(0, C^{(l)} - g_b)||\mathcal{N}(0, g_l C^{(l-1)})) \\ &= -\big\langle \ln \mathcal{N}(z^{(l)}|0, g_l C^{(l-1)}) \big\rangle_{z^{(l)} \sim \mathcal{N}(0, C^{(l)} - g_b)} + \big\langle \ln \mathcal{N}(z^{(l)}|0, C^{(l)} - g_b) \big\rangle_{z^{(a)} \sim \mathcal{N}(0, C^{(a)} - g_b)} \\ &= \frac{N}{2g_l} \operatorname{tr} [C^{(l-1)}]^\intercal (C^{(l)} - g_b) + \frac{N}{2} \ln\det \big(C^{(l-1)}\big) - \frac{N}{2} \ln\det \big(C^{(l)} - g_b\big) + \text{const.}\,, \end{aligned} \tag{93}$$

where the factors N result from the $z_{\alpha i}$ being i.i.d. in $i = 1, \dots, N$. Apart from constant terms that we dropped, this is the same form as (90). In the case $g_b = 0$, the action (92) thus reduces to the main result by (Yang et al., 2023), their Eq. (1), when setting all relative layer width $\nu_\ell = 1$ as assumed here and using $K = \mathrm{id}$, valid for deep kernel machines. Our approach is thus consistent with theirs, if we study deep linear networks. Note that our general result, the action (9), is valid for deep non-linear networks.

As for the non-linear networks considered in the main text, we may derive the pair of forward-backward equations for the kernel adaptation in the linear case. The forward iteration (91) can be rewritten as

$$C^{(l)} = g_b + g_l\, C^{(l-1)} \big(\mathbb{I} - 2\frac{g_l}{N} \tilde{C}^{(l)} C^{(l-1)}\big)^{-1}. \tag{94}$$

The backward equation for $\tilde{C}$ arises from computing the MAP estimate of C from (92) as $\partial \mathcal{S}(C)/\partial C_{\alpha\beta}^{(l)} \stackrel{!}{=} 0$, which for $l = L$ yields (10) and for $1 \leq l < L$ evaluates to (cf. (64))

$$\begin{aligned} 0 &\stackrel{!}{=} \frac{\partial}{\partial C_{\alpha\beta}^{(l)}} \big(\Gamma(C^{(l)}|C^{(l-1)}) + \Gamma(C^{(l+1)}|C^{(l)})\big) \\ &= \tilde{C}_{\alpha\beta}^{(l)} - \frac{\partial}{C_{\alpha\beta}^{(l)}} \mathcal{W}(\tilde{C}^{(l+1)}|C^{(l)}), \end{aligned}$$

with the explicit form (89) written as $\mathcal{W}(\tilde{C}^{(l+1)}|C^{(l)}) = \sum_{\alpha\beta} \tilde{C}_{\alpha\beta}^{(l+1)} g_b - \frac{N}{2} \ln\det \big(\mathbb{I} - 2\frac{g_{l+1}}{N} \tilde{C}^{(l+1)} C^{(l)}\big)$ so that

$$\tilde{C}^{(l)} = g_{l+1}\, \tilde{C}^{(l+1)} \big(\mathbb{I} - 2\frac{g_{l+1}}{N} \tilde{C}^{(l+1)} C^{(l)}\big)^{-1}. \tag{95}$$

Note that the form of the forward equation (94) and the backward equation (95) show a symmetry such that $[g_l \tilde{C}^{(l)}]^{-1} \tilde{C}^{(l-1)} = [g_l C^{(l-1)}]^{-1} (C^{(l)} - g_b) = \big(\mathbb{I} - 2\frac{g_l}{N} \tilde{C}^{(l)} C^{(l-1)}\big)^{-1}$.

To test the behavior of linear networks, we use a linearly separable Ising task: Each pattern x_α in the Ising task is D-dimensional and $x_{\alpha i} \in \{\pm 1\}$. If the pattern belongs to class -1, each $x_{\alpha i}$ realizes $x_{\alpha i} = +1$ with a probability of $p_1 = 0.5 - \Delta p$ and the value $x_{\alpha i} = -1$ with $p_2 = 0.5 + \Delta p$. The value for each pattern element $x_{\alpha i}$ is drawn independently. If the pattern belongs to class $+1$, the probabilities for $x_{\alpha i} = 1$ and $x_{\alpha i} = -1$ are inverted. The task separability increases with larger Δp. In Fig. 6 we plot the mean squared error difference between the numerically sampled kernels and the feature-corrected kernels from theory, the NNGP kernels and the linear approximation in $\tilde{C}$ of the feature corrections for a linear single-hidden layer network as a function of the ratio $\alpha = P/N$ between the number of training samples P and network width N. The feature-corrected kernels from the full theory converge to the empirically measured kernels when increasing the network width N while keeping $\alpha = P/N$ fixed, showing that the large deviation result becomes more and more precise. The linear approximation in $\tilde{C}$ yields only a slightly higher error, justifying this approximation. The deviation between the NNGP and the empirical kernel is consistently higher, showing that feature corrections remain important in the proportional limit.

D. Adaptation towards the target in linear networks

To gain more intuition into the adaptation of the kernels towards the target, we may investigate a linear network. To this end consider the second-order cumulant expansion of $\mathcal{W}$ given by (89) as

$$\mathcal{W}(\tilde{C}^{(l)}|C^{(l-1)}) = \sum_{\alpha\beta} \tilde{C}_{\alpha\beta}^{(l)} g_b + g_l\, \tilde{C}_{\alpha\beta}^{(l)} C_{\alpha\beta}^{(l-1)} + \frac{g_l^2}{2N} \tilde{C}_{\alpha\beta}^{(l)} \big(C_{\alpha\gamma}^{(l-1)} C_{\beta\delta}^{(l-1)} + C_{\alpha\delta}^{(l-1)} C_{\beta\gamma}^{(l-1)}\big) \tilde{C}_{\gamma\delta}^{(l)} + \mathcal{O}(\tilde{C}^3), \tag{96}$$

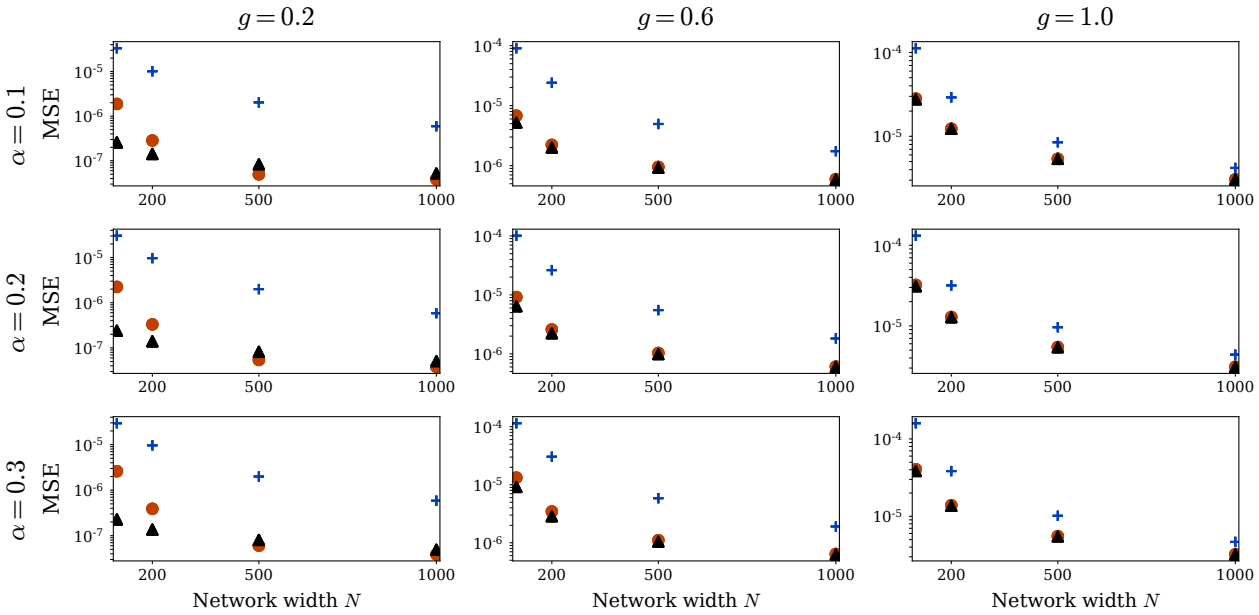

Figure 6. Dependence on the ratio between training samples and network width $\alpha = P/N$ for linear single-hidden layer network. The mean squared error difference $\mathrm{MSE}(C, C_{\mathrm{emp}}) = 1/D^2 \sum_{\alpha,\beta=1}^{D} (C_{\alpha\beta} - C_{\alpha\beta}^{\mathrm{emp}})^2$ measures kernel adaptation relative to the numerically sampled kernels. The MSE of the data-dependent kernels (blue: NNGP; red: linear approximation; black: full theory) shows that the feature-corrected kernels are consistently closer to the empirical kernel than the NNGP kernel. Parameters: Ising task $\Delta p = 0.2$, $D = 1000$, $L = 1$, $g_l = \{0.2, 0.6, 1.0\}$, $\kappa = 0.001$, $\sigma_b^2 = 0.05$ for $N = 100, 200, 500, 1000$. Results are averaged over 10 training data sets and error bars indicate standard deviation.

where summation over repeated indices on the right are implied and where we used that $\langle h_\alpha h_\beta, h_\gamma h_\delta \rangle^c = \langle h_\alpha h_\beta h_\gamma h_\delta \rangle - \langle h_\alpha h_\beta \rangle \langle h_\gamma h_\delta \rangle$, decomposed the fourth moment into cumulants by Wick's theorem and what remains are the pairings $\langle h_\alpha h_\gamma \rangle \langle h_\beta h_\delta \rangle + \langle h_\alpha h_\delta \rangle \langle h_\beta h_\gamma \rangle = C_{\alpha\gamma} C_{\beta\delta} + C_{\alpha\delta} C_{\beta\gamma}$. The stationarity condition (91) with (96) takes the form

$$C_{\alpha\beta}^{(l+1)} = g_b + g_{l+1} C_{\alpha\beta}^{(l)} + 2 \frac{g_{l+1}^2}{N} C_{\alpha\gamma}^{(l)} \tilde{C}_{\gamma\delta}^{(l+1)} C_{\delta\beta}^{(l)}.$$

Considering the case $g_b = 0$ in the following, the correction term comes with a factor N^{-1}, so we may approximate within the correction term $g_{l+1} C^{(l)} \simeq C^{(l+1)} + \mathcal{O}(N^{-1})$. The correction term in the last layer $l = L$ in this approximation becomes

$$C_{\alpha\beta}^{(L)} \simeq g_L C_{\alpha\beta}^{(L-1)} + \frac{2}{N} C_{\alpha\gamma}^{(L)} \tilde{C}_{\gamma\delta}^{(L)} C_{\delta\beta}^{(L)} + \mathcal{O}(N^{-1}).$$

Now assuming $\kappa = 0$ (no regularization, training without noise) and inserting the form of $\tilde{C}^{(L)} = \frac{1}{2}(C^{(L)})^{-1} Y Y^\intercal (C^{(L)})^{-1} - \frac{1}{2}(C^{(L)})^{-1}$ given by (10), we obtain the correction term

$$\begin{aligned} \frac{2}{N} C_{\alpha\gamma}^{(L)} \tilde{C}_{\gamma\delta}^{(L)} C_{\delta\beta}^{(L)} &= \frac{2}{N} C_{\alpha\gamma}^{(L)} \big[\frac{1}{2}(C^{(L)})^{-1} Y Y^\intercal (C^{(L)})^{-1} - \frac{1}{2}(C^{(L)})^{-1}\big]_{\gamma\delta} C_{\delta\beta}^{(L)} \\ &= \frac{1}{N}\big(Y Y^\intercal - C^{(L)}\big), \end{aligned}$$

so that we get the final result

$$C_{\alpha\beta}^{(L)} \simeq g_L C_{\alpha\beta}^{(L-1)} + \frac{1}{N}\big(Y Y^\intercal - C^{(L)}\big)_{\alpha\beta}. \tag{97}$$

This shows that correction term tends to push the kernel into the rank-one direction of the target $Y Y^\intercal$ in order to increase the log-likelihood of the data.

E. Relation to the Neural Tangent Kernel

This section links our work to the neural tangent kernel (NTK). This material only serves the purpose to recast the known results on the NTK (Jacot et al., 2018; Lee et al., 2019) into our setting and notation. We here consider the specific case of the NTK for the squared error loss function

$$\mathcal{L}(f;Y) = \frac{1}{2}\sum_{\alpha=1}^{P} \|f_\alpha - y_\alpha\|^2 \tag{98}$$

and the network architecture is given by

$$\begin{aligned} h_\alpha^{(0)} &= W^{(0)} x_\alpha, \\ h_\alpha^{(l)} &= W^{(l)} \phi\left(h_\alpha^{(l-1)}\right), \quad l = 1, \dots, L, \\ f_\alpha &= h_\alpha^{(L)} \in \mathbb{R}, \end{aligned} \tag{99}$$

where to simplify notation we set the biases to zero – an extension to non-zero biases is of course possible. To further simplify notation and to connect the calculations to the main results of the current work, we assume that the widths for all layers are the same and are denoted as N. In the NTK architecture the weights are scaled as $W^{(l)} = w^{(l)}/\sqrt{N}$ with $\mathcal{O}(1) \sim w^{(l)} \sim \mathcal{N}(0, g_l)$ at initialization. For simplicity, we also assume the same dimension for the data $x_\alpha \in \mathbb{R}^N$ here. The $w^{(l)}$ are considered the trainable parameters which implies that the gradient is multiplied by $1/\sqrt{N}$ and is hence reduced for large networks. This leads to the weights $w^{(l)}$ not departing strongly from their initialization. For equal layer widths the gradient scaling with $1/\sqrt{N}$ corresponds to using a rescaled loss function $\bar{\mathcal{L}} = \mathcal{L}/\sqrt{N}$, so the NTK studies the learning dynamics

$$\partial_t W = -\nabla_W \bar{\mathcal{L}}. \tag{100}$$

In the following we will only make the link to the NTK right after initialization, because it has been shown that on the limit $N \to \infty$ the NTK stays constant over training (Jacot et al., 2018). As shown in (Lee et al., 2019) the NTK at initialization is equivalent to linearizing the network outputs f_α around a set of initial weights. This assumption corresponds to assuming that the trained weights do not depart strongly from their initial point. We will show here that the framework of Bayesian inference we employ here reduces to the NTK at initialization under the additional assumption of such a linear dependence of the network output on the parameters of the network. This corresponds replacing the mapping between inputs $X \in \mathbb{R}^{(P+1)\times N}$ and outputs $f(X|W) \in \mathbb{R}^{P+1}$ implied by (99) by a linearized dependency on the parameters $W = \{w_{ij}^{(0)} .... w_{ij}^{(L)}\}$

$$\begin{aligned} f(X|W) &\simeq f(X|W_0) + \nabla f(X|W_0)\,\omega \\ &=: f_0 + \nabla f\,\omega, \\ \omega &= W - W_0, \end{aligned} \tag{101}$$

where $\mathbb{R}^{(P+1)} \ni \nabla f(X|W_0)\,\omega := \sum_{l,ij} \frac{\partial f(X|W)}{\partial w_{ij}^{(l)}}\big|_{W_0} \omega_{ij}^{(l)}$ and the ω denote the deviations of the weights from their initial values. Here $X \in \mathbb{R}^{(P+1)\times N}$ is the matrix of all $1 \le \alpha \le P+1$ data points, corresponding to P training points and one test point $\alpha = *$ and $\nabla f \in \mathbb{R}^{(P+1)\times L\,N^2+N}$ is the Jacobi matrix of the corresponding network outputs with regard to the $L\,N^2 + N$ weight parameters of the network (99).

In contrast to (100), the main part of our work considers training with a stochastic learning dynamics with weight decay (see G)

$$\begin{aligned} \partial_t W(t) &= -\gamma W(t) - \nabla \mathcal{L}(f(X, W(t)); Y) + \sqrt{2T}\zeta(t)\,, \\ \big\langle \zeta_i(t)\zeta_j(s)\big\rangle &= \delta_{ij}\delta(t-s). \end{aligned} \tag{102}$$

We recover NTK training (100) by first changing the time scale by a factor $\tau = \sqrt{N}$ as

$$\begin{aligned} \tau\,\partial_t W(t) &= -\gamma W(t) - \nabla \mathcal{L}(f(X, W(t)); Y) + \sqrt{2T\tau}\,\zeta(t)\,, \\ \big\langle \zeta_i(t)\zeta_j(s)\big\rangle &= \delta_{ij}\delta(t-s). \end{aligned} \tag{103}$$

Since the stationary distribution is invariant under a change of time scale, both dynamics (102) and (103) obey the same stationary distribution. Ultimately we need to set the temperature T and the weight decay γ to zero. We will take $\gamma = 0$ immediately, but leave T finite for the intermediate steps of the calculation and only take the limit in the end. In addition we will use the linearization (101) and therefore study the dynamics of the $\omega(t)$ following from (103) by dividing by τ as

$$\partial_t \omega(t) = -\nabla_{\omega(t)} \bar{\mathcal{L}}(f_0 + \nabla f\, \omega(t)\,;\, Y) + \sqrt{2T/\tau}\, \zeta(t).$$

The stationary distribution of $\omega(t)$ under this dynamics obeys

$$p_0(\omega|W_0) \propto \exp\big(-\frac{\tau}{T}\, \bar{\mathcal{L}}(f_0 + \nabla f\, \omega;\, Y)\big),$$

which for the quadratic loss function (98) is a Gaussian distribution in ω. The resulting joint distribution of network outputs f and labels Y is then Gaussian, too, because of the affine linear dependence of f on ω in (101). It corresponds to the network prior we compute in the main text (2) and here takes the form

$$\begin{aligned} p(Y, f|X, W_0) &\propto \int d\omega\, \exp\big(-\frac{\tau}{T}\, \bar{\mathcal{L}}(f;Y)\big)\, \delta(f - f_0 - \nabla f\, \omega)) \\ &= \int d\omega\, \exp\big(-\frac{1}{T}\, \mathcal{L}(f;Y)\big)\, \delta(f - f_0 - \nabla f\, \omega)). \end{aligned}$$

To investigate the statistics of the output conditioned on the training labels Y, it is easiest to introduce the cumulant-generating function for the conditional $p(f|Y, X, W_0) := p(Y, f|X, W_0)/\int df\, p(Y, f|X, W_0)$ with $j^\mathsf{T} f = \sum_{\alpha=1}^{P+1} j_\alpha f_\alpha$

$$\begin{aligned} \mathcal{W}(j|Y, X, W_0) &= \ln \frac{\int df\, p(Y, f|X, W_0)\, e^{j^\mathsf{T} f}}{\int df\, p(Y, f|X, W_0)} \qquad (104) \\ &= \ln \left\langle e^{j^\mathsf{T} f} \right\rangle_{f\sim p(Y,f|X,W_0)} + \text{const.} \\ &= \ln \int df \int d\omega\, \exp\big(j^\mathsf{T} f - \frac{1}{2T}\, \|Y - f\|_P^2\big)\, \delta\big(f - f_0 - \nabla f\, \omega\big) + \text{const.} \\ &= \ln \int d\omega\, \exp\big(j^\mathsf{T}(f_0 + \nabla f\, \omega) - \frac{1}{2T}\, \|Y - f_0 - \nabla f\, \omega\|_P^2\big) + \text{const.} \\ &= \ln \int d\omega\, \exp\big(j^\mathsf{T}(f_0 + \nabla f\, \omega) - \frac{1}{2T}\omega^\mathsf{T}\, \nabla f^\mathsf{T} \nabla f\, \omega + \frac{1}{T}\, (Y - f_0)^\mathsf{T}\, \nabla f\, \omega\big) + \text{const.} \\ &= \ln \int d\omega\, \exp\big(j^\mathsf{T} f_0 + \big(j^\mathsf{T} \nabla f + \frac{1}{T}\, (Y - f_0)^\mathsf{T} \nabla f\big)\, \omega - \frac{1}{2T}\omega^\mathsf{T}\, \nabla f^\mathsf{T} \nabla f\, \omega\big) + \text{const.} \\ &= j^\mathsf{T} f_0 + \frac{T}{2}\, \big(j^\mathsf{T} \nabla f + \frac{1}{T}\, (Y - f_0)^\mathsf{T}\, \nabla f\big)\, \big[\nabla f^\mathsf{T} \nabla f\big]^{-1} \big(j^\mathsf{T} \nabla f + \frac{1}{T}\, (Y - f_0)\, \nabla f\big)^\mathsf{T} + \text{const.}, \end{aligned}$$

where we performed the Gaussian integral over the $\omega \in \mathbb{R}^{(L\,N^2+N)}$ in the last step and dropped all constant terms (independent of j) along the way. Note that here the norm $\|Y - f\|_P^2$ is with regard to the P training points only, likewise all inner products following from it; the only inner products that involve the test point are those in $j^\mathsf{T} \nabla f$ and $j^\mathsf{T} f_0$. From the latter form we can read off that the statistics of the output is Gaussian, because we obtain a polynomial of degree two in j; the mean for the test point $\alpha = *$ is hence its linear coefficient

$$\mu_* = \frac{\partial}{\partial j_*} W\big|_{j=0} = f_{0,*} + \nabla f_*\, \big[\nabla f^\mathsf{T} \nabla f\big]^{-1}\, \nabla f^\mathsf{T} (Y - f_0), \qquad (105)$$

which is in particular independent of T, so that the limit $T \to 0$ exists. The variance is given by the term $\propto j^2$ which is linear in T and hence vanishes for $T \to 0$. To recover the NTK result in the known form, we may use that for any matrix X associativity holds, so that

$$(X^\mathsf{T}\, X)\, X^\mathsf{T} = X^\mathsf{T}\, (X\, X^\mathsf{T}),$$

from which follows by multiplying with $(X^\mathsf{T} X)^{-1}$ from left and by $(X X^\mathsf{T})^{-1}$ from right

$$X^\mathsf{T}\, (X\, X^\mathsf{T})^{-1} = (X^\mathsf{T}\, X)^{-1}\, X^\mathsf{T}.$$

We may use this to rewrite the mean of the predictor in (105) as

$$\mu_* = \frac{\partial}{\partial j_*} W\big|_{j=0} = f_{0,*} + \nabla f_* \nabla f^\mathsf{T} \left[\nabla f \nabla f^\mathsf{T}\right]^{-1} (Y - f_0), \tag{106}$$

where the NTK kernel

$$\Theta_{\alpha\beta} = [\nabla f \nabla f^\mathsf{T}]_{\alpha\beta} \equiv \sum_{l,ij} \frac{\partial f_\alpha}{\partial W_{ij}^{(l)}} \frac{\partial f_\beta}{\partial W_{ij}^{(l)}} \tag{107}$$

for $1 \leq \alpha, \beta \leq P+1$ appears; specifically, the matrix $\left[\nabla f \nabla f^\mathsf{T}\right]_{1\leq\alpha,\beta\leq P}$ is with regard to the P training points only (inherited from the norm $\| \dots \|_P^2$), while $[\nabla f_* \nabla f^\mathsf{T}]_{1\leq\beta\leq P}$ is a vector of dimension P, fixing the left index to the training point $\alpha = *$ (coming from the derivative by j_*). The expression (106) is the stationary point of the NTK predictor for the linearized network (cf. Eqs. (10)-(11) in (Lee et al., 2019)).

The original work (Theorem 2 in Sec 4.2 of (Jacot et al., 2018)) has shown that in the limit $N \to \infty$ the NTK stays constant over training, which implies that the expressions obtained here remain valid throughout training.

This shows that the posterior we compute in our general framework is consistent with the NTK if we make the additional assumption that the dependence of the network output can be linearized with regard to the trained parameters; such an assumption is justified if the weights do not depart strongly from their initial values, as in the NTK setting in the $N \to \infty$ limit. Our general approach does not require this linearization. In addition, for the NTK we took vanishing weight decay – an extension to non-zero weight decay would be possible, though, still remaining with a Gaussian ω and hence f. The computation here also shows that we may use a non-zero T in the training dynamics as we do in the main text and would still obtain the same result (106) for the mean of the predictor, albeit with a non-zero variance that can be read off from (104).

The conceptually important difference between the NTK kernel Θ (107) and the kernels C we study is that the NTK kernel is agnostic to the training labels Y – the shape of the kernel only depends on the network architecture and the data points X, but not on the labels Y. Similar to the NNGP the NTK is hence unable to relate the inputs X and the labels Y and hence does not capture feature learning. Our work, in contrast, investigates how kernels are shaped by the joint statistics of X and Y – this is evident from the fact that the MAP estimate of the kernels results from an interplay of the likelihood of the labels $\mathcal{S}_D$ and the prior term in (9) and is shown by the increase of the CKA between $C^{(L)}$ and YY^T; for linear networks this increase is shown explicitly in (97).

F. Price's theorem

Consider an expectation value of $f : \mathbb{R}^N \to \mathbb{R}$ over centered jointly Gaussian distributed x_i with covariance C

$$\langle f(x) \rangle_{x \sim \mathcal{N}(0,C)}.$$

We assume that $f(x)$ grows slower than $e^{x_i^2}$ for large x_i. Rewriting the Gaussian $\mathcal{N}(0, C)$ in terms of its Fourier transform $\mathcal{N}(0, C) = \left\{ \prod_j \int_{-i\infty}^{i\infty} \frac{d\tilde{x}_j}{2\pi i} \right\} \exp\left(-x^\mathsf{T}\tilde{x} + \frac{1}{2}\tilde{x}^\mathsf{T} C \tilde{x} \right)$ one obtains

$$\langle f(x) \rangle_{x \sim \mathcal{N}(0,C)} = \prod_j \left\{ \int_{-\infty}^{\infty} dx_j \int_{-i\infty}^{i\infty} \frac{d\tilde{x}_j}{2\pi i} \right\} f(x) \exp\left(-x^\mathsf{T}\tilde{x} + \frac{1}{2}\tilde{x}^\mathsf{T} C \tilde{x} \right),$$

which yields the property

$$\frac{\partial}{\partial C_{kl}} \langle f(x) \rangle_{x \sim \mathcal{N}(0,C)} = \prod_j \left\{ \int_{-\infty}^{\infty} dx_j \int_{-i\infty}^{i\infty} \frac{d\tilde{x}_j}{2\pi i} \right\} f(x) \frac{1}{2} \tilde{x}_k \tilde{x}_l \exp\left(-x^\mathsf{T}\tilde{x} + \frac{1}{2}\tilde{x}^\mathsf{T} C \tilde{x} \right).$$

One notices that one may replace both occurrences of $\tilde{x}_i \to -\partial/\partial x_i$ under the integral: integrating by parts twice and using the assumption that f grows slower than $e^{x_i^2}$ for large x_i so that boundary terms vanish, one obtains

$$\begin{aligned} \frac{\partial}{\partial C_{kl}} \langle f(x) \rangle_{x \sim \mathcal{N}(0,C)} &= \prod_j \left\{ \int_{-\infty}^{\infty} dx_j \int_{-i\infty}^{i\infty} \frac{d\tilde{x}_j}{2\pi i} \right\} \frac{1}{2} \left\{ \frac{\partial}{\partial x_k} \frac{\partial}{\partial x_l} f(x) \right\} \exp\left(x^\mathsf{T}\tilde{x} + \frac{1}{2}\tilde{x}^\mathsf{T} C \tilde{x} \right) \\ &= \frac{1}{2} \left\langle f^{(2)}_{kl} \right\rangle_{x \sim \mathcal{N}(0,C)}, \end{aligned} \tag{108}$$

where $f^{(2)}_{kl}$ is the Hessian of f. This expression is known as Price's theorem (Price, 1958; Papoulis & Pillai, 2002). Note that sometimes the theorem is only stated for derivatives by $C_{k\neq l}$ only.

This theorem can be used to rewrite

$$\frac{\partial}{\partial C^{(l)}_{\alpha\beta}} \Big\langle \exp\big(\frac{g_{l+1}}{N}\phi^{(l)\intercal}\tilde{C}^{(l+1)}\phi^{(l)}\big)\Big\rangle_{h^{(l)}\sim\mathcal{N}(0,C^{(l)})} \tag{109}$$

to obtain an expression for $\frac{\partial}{\partial C^{(l)}_{\alpha\beta}}\mathcal{W}(\tilde{C}^{(l+1)}|C^{(l)})$ in (11) (see A.4).

G. Langevin stochastic gradient descent

To compare theoretical results to real networks we sample numerically from the posterior of networks that have been conditioned on the training data $X = (x_\alpha)_{\alpha=1,\dots,P}, Y = (y_\alpha)_{\alpha=1,\dots,P}$. We therefore train the network (1) using Langevin stochastic gradient descent (LSGD). According to (Naveh et al., 2021) evolving parameters Θ with the stochastic differential equation

$$\begin{aligned} \partial_t\Theta(t) &= -\gamma\Theta(t) - \nabla_\Theta\mathcal{L}(\Theta(t);Y) + \sqrt{2T}\zeta(t), \\ \big\langle\zeta_i(t)\zeta_j(s)\big\rangle &= \delta_{ij}\delta(t-s), \end{aligned} \tag{110}$$

with the squared error loss $\mathcal{L}(\Theta;Y) = \sum_{\alpha=1}^{P}(f_\alpha(\Theta)-y_\alpha)^2$ and $f_\alpha(\Theta)$ denoting the network output for sample α, leads to sampling from the equilibrium distribution for Θ for large times t which reads

$$\lim_{t\to\infty} p\left(\Theta(t)\right) \sim \exp\left(-\frac{\gamma}{2T}\|\Theta\|^2 - \frac{1}{T}\mathcal{L}(\Theta;Y)\right). \tag{111}$$

The equilibrium distribution may be derived from the Fokker-Planck equation (Risken, 1996) for the density of Θ. Conversely, this implies a density for the output

$$\begin{aligned} p(Y|X) &\propto \int d\Theta\, \exp\big(-\frac{\gamma}{2T}\|\Theta\|^2 - \frac{1}{T}\|f-Y\|^2\big) \\ &\propto \Big\langle \exp\big(-\frac{1}{T}\|f-Y\|^2\big)\Big\rangle_{\Theta_k\overset{\text{i.i.d.}}{\sim}\mathcal{N}(0,T/\gamma)} \\ &\propto \mathcal{N}(Y|f,T/2)\,\langle\delta\big[f-f(\Theta)\big]\rangle_{\Theta_k\overset{\text{i.i.d.}}{\sim}\mathcal{N}(0,T/\gamma)}, \end{aligned} \tag{112}$$

which, with $p(f|X) \equiv \langle\delta\big[f-f(\Theta)\big]\rangle_{\Theta_k\overset{\text{i.i.d.}}{\sim}\mathcal{N}(0,T/\gamma)}$, is identical to (2) if one identifies $\kappa = T/2$ with the regularization noise and $T/\gamma = g/N$ with the variance of the parameter Θ_k. To implement different variances in practice, one requires a different weight decay γ for each parameter.

The time discrete version of (110) is implemented as

$$\begin{aligned} \Theta_t &= \Theta_{t-1} - \eta\left(\gamma\Theta_{t-1} + \nabla_\Theta\mathcal{L}(\Theta_{t-1};Y)\right) + \sqrt{2T\eta}\,\zeta_t, \\ \langle\zeta_t\zeta_s\rangle &= \delta_{ts}, \end{aligned} \tag{113}$$

with standard normal ζ_t and finite time step η, which can also be interpreted as a learning rate. To accurately reflect the time evolution according to (110) the learning rate needs to be small. Hence Langevin stochastic gradient descent corresponds to full-batch gradient descent with the addition of i.i.d. distributed standard normal noise and weight decay regularization (Krogh & Hertz, 1991). The value for κ, which appears in the main text as the regularizer on the diagonal of the output kernel $C^{(L)}$, quantifies the tradeoff between the influence of the prior and the influence of the training data via the loss term. Choosing large κ corresponds to large $T = 2\kappa$ and hence a large noise in the LSGD, putting more emphasis on the parameter priors. In contrast, small regularization values κ favor the training data in the loss in the exponent. When using LSGD to sample from the equilibrium distribution, it needs to be ensured that the distribution is equilibrated and subsequent network samples drawn from the distribution are uncorrelated. For empirical results, we therefore sample networks after an initial warmup of 50.000 training steps in distances of 1.000 time steps.

H. Additional details of theory implementation

H.1. Setting weight variance of input layer

The response functions $\chi^{l,\to}$ describe the effect of a perturbation in the input kernel the kernel in layer l. It depends on the network kernels $C^{(k)}_{\alpha\alpha}$ of all layers before layer k. For simplicity, we set the weight variance of the input layer g_0 such that the diagonal elements of the network kernels $C^{(l)}_{\alpha\alpha}$ are already at their fixed point value for large depth (Schoenholz et al., 2017). Consequently, the convergence of the diagonal kernel elements does not influence the response functions and there remains only one relevant relaxation scale for the latter.

H.2. Gaussian integrals

We solve the self-consistency equations in (11) iteratively. This requires computing two-point and four-point Gaussian integrals. For $\phi = \text{erf}$, we obtain the following analytical expressions for the two-point integrals

$$\langle \phi(h_\alpha)\phi(h_\beta)\rangle_{h\sim\mathcal{N}(0,C)} = \begin{cases} \frac{4}{\pi}\arctan\left(\sqrt{1+4C_{\alpha\alpha}}\right) - 1 & \alpha = \beta, \\ \frac{2}{\pi}\arcsin\left(\frac{2C_{\alpha\beta}}{\sqrt{1+2C_{\alpha\alpha}}\sqrt{1+2C_{\beta\beta}}}\right) & \text{else,} \end{cases}$$

$$\langle \phi'(h_\alpha)\phi'(h_\beta)\rangle_{h\sim\mathcal{N}(0,C)} = \begin{cases} \frac{4}{\pi}\frac{1}{\sqrt{4C_{\alpha\alpha}+1}} & \alpha = \beta, \\ \frac{4}{\pi}\left(2\left(C_{\alpha\alpha}+C_{\beta\beta}\right)+1+4\left(C_{\alpha\alpha}C_{\beta\beta}-C^2_{\alpha\beta}\right)\right)^{-0.5} & \text{else,} \end{cases}$$

$$\langle \phi(h_\alpha)\phi''(h_\beta)\rangle_{h\sim\mathcal{N}(0,C)} = \begin{cases} -\frac{8}{\pi}\frac{C_{\alpha\alpha}}{(2C_{\alpha\alpha}+1)\sqrt{4C_{\alpha\alpha}+1}} & \alpha = \beta, \\ -\frac{8}{\pi}\frac{C_{\beta\alpha}}{(2C_{\alpha\alpha}+1)\sqrt{2(C_{\alpha\alpha}+C_{\beta\beta})+1+4\left(C_{\alpha\alpha}C_{\beta\beta}-C^2_{\alpha\beta}\right)}} & \text{else.} \end{cases}$$

We are not aware of an analytical solution for the appearing four-point integral $\left\langle \phi(h_\alpha)\phi(h_\beta)\phi(h_\gamma)\phi(h_\delta)\right\rangle_{h\sim\mathcal{N}(0,C)}$. Therefore, we determine this integral numerically using Monte-Carlo sampling with $n_{MC} = 10^5$ samples.

H.3. Annealing in network width

In Section 3 we derived self-consistency equations for the posterior kernels perturbatively up to linear order in the conjugate kernels $\tilde{C}^{(l)}$ (see (11)). We solve these equations iteratively: i) Initialize $C^{(0)}$ by (5) and set $\tilde{C} = 0$ initially. ii) Iterate (15) forward until $C^{(L)}$; in the first iteration this step still corresponds to the NNGP. iii) Determine $\tilde{C}^{(L)}$ in the final layer from (10). iv) Propagate $\tilde{C}$ backward with (11) (but using the Gaussian measure $\langle\ldots\rangle_{\mathcal{N}(0,C^{(l)})}$ instead of the non-Gaussian measure $\langle\ldots\rangle_{\mathcal{P}^{(l)}}$ throughout (11)). Then go back to step ii) with $\tilde{C} \neq 0$ and iterate until convergence. To improve the stability of these iterations, we use a damping parameter $\gamma = 0.5$ and replace per iteration i as

$$C^{(l),i} \mapsto (1-\gamma)C^{(l),i+1} + \gamma C^{(l),i},$$
$$\tilde{C}^{(l),i} \mapsto (1-\gamma)\tilde{C}^{(l),i+1} + \gamma \tilde{C}^{(l),i}.$$

When solving these equations iteratively, we use the NNGP kernel as the starting value. For wide networks and fixed training data $P/N \to 0$, corrections to the NNGP kernel become small and posterior kernels are well described by including corrections up to linear order. To obtain posterior kernels for arbitrary network widths, we use that corrections are small when determining the posterior kernels based on the posterior kernels of a slightly wider network: We start from very wide networks and compute corrections to the NNGP kernel. Then we use these corrected kernels as the starting point for slightly narrower networks and repeat until a certain network width (see pseudo code in 1).

In Fig. 7, we show the CKA between the output kernel $C^{(L)}$ and target kernel $YY^\intercal$ relative to the NNGP kernel for annealing in network width.

I. Centered kernel alignment

According to (Canatar & Pehlevan, 2022), the kernel alignment between two kernels $A,\ B \in \mathbb{R}^{P\times P}$ is measured by

$$\frac{\text{Tr}(A\,B)}{\sqrt{\text{Tr}(A\,A)\,\text{Tr}(B\,B)}}.$$

Algorithm 1 Width annealing of kernels

Input: data X, labels Y, network widths $\{N_i\}_i$
Compute NNGP kernel $C^{(l)}_{\text{NNGP}}$ from data X.
Set start values to NNGP kernel $C^{(l)}_{\text{init}} = C^{(l)}_{\text{NNGP}}$ and $\tilde{C}^{(l)}_{\text{init}} = 0$.
for N **in** $\{N_i\}_i$ **do**
  Compute corrected kernels $C^{(l)}_{\text{corr}} = f(C^{(l)}_{\text{init}}, \tilde{C}^{(l)}_{\text{init}}, Y, N)$ and conjugate kernels $\tilde{C}^{(l)}_{\text{corr}} = g(C^{(l)}_{\text{init}}, \tilde{C}^{(l)}_{\text{init}}, Y, N)$.
  Reset start values $C^{(l)}_{\text{init}} = C^{(l)}_{\text{corr}}$ and $\tilde{C}^{(l)}_{\text{init}} = \tilde{C}^{(l)}_{\text{corr}}$.
end for

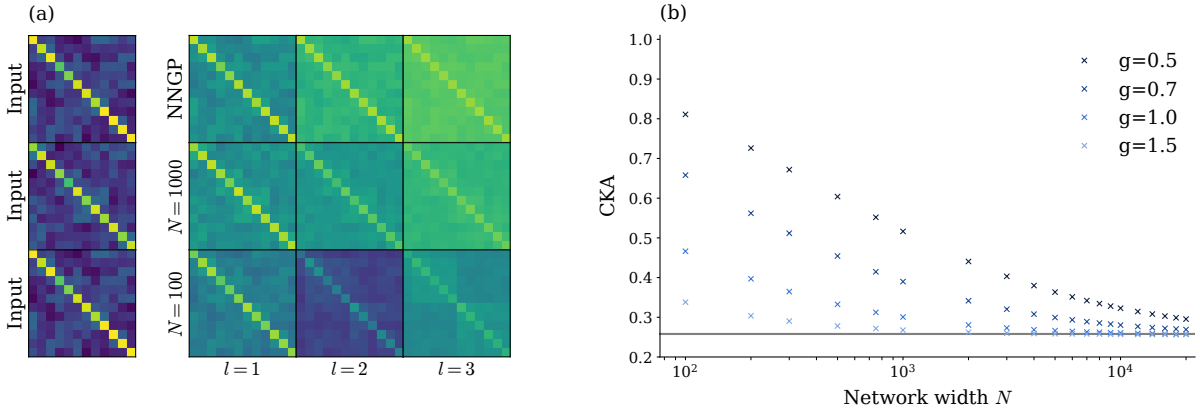

Figure 7. Annealing in network width to solve self-consistency equations. (a) Network kernel $C^{(l)}$ across layers $l = 1, 2, 3$ for different network width N and $g_l = g = 0.5$. More narrow networks show stronger adaptation to the target kernel $YY^\intercal$ across layers. (b) CKA between network kernels $C^{(L)}$ and target kernel $YY^\intercal$ for different network width. For wide networks, the CKA (blue markers) remains close to that of the NNGP (solid line). For more narrow networks, the corrections towards the target kernel and away from the NNGP kernel increase continuously. The correction strength depends on other network hyperparameters such as the weight variance g_l (increasing from dark to light). Other parameters: XOR task with $\sigma^2 = 0.4$, $g_l \in \{0.5, 0.7, 1.0, 1.5\}$, $g_b = 0.05$, $L = 3$, $\kappa = 10^{-3}$, $P = 12$.

This corresponds to the cosine similarity between the flattened kernels and is thus invariant under scaling the kernels by a scalar $A \mapsto aA$. To remove constant components in the eigendecomposition of the kernel (Cortes et al., 2012), we use the centered kernel alignment (CKA): the kernels are transformed as $A \mapsto HAH$ and $B \mapsto HBH$ with the centering matrix $H = \mathbb{I} - \frac{1}{P}11^\intercal$ where 1 is the matrix with all ones as elements. Throughout this work, we study the CKA between network kernels $C^{(l)}$ and the target kernel $YY^\intercal$.