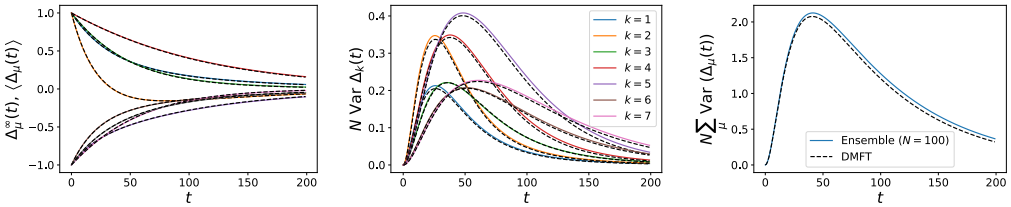
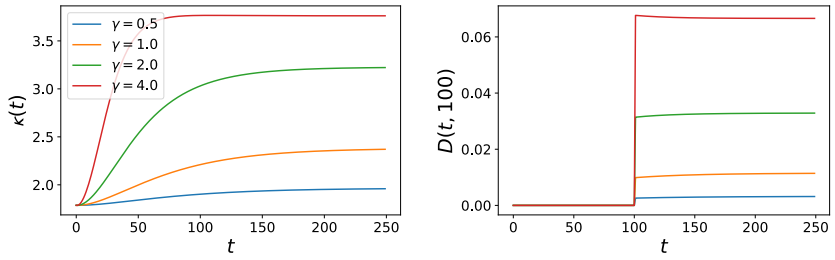
Appendix

A Additional Figures



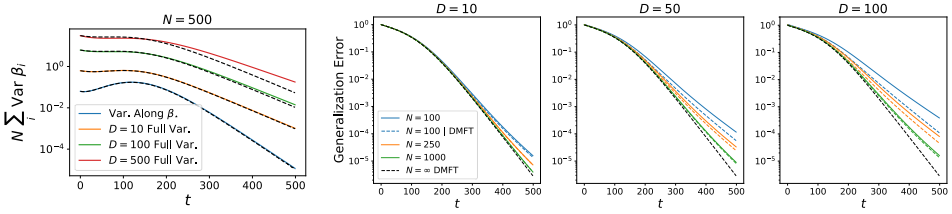
(a) Average Prediction Errors (b) Top Eigenmodes Variance (c) Total Train Variance

Figure A.1: We show the accuracy of the lazy-limit ODE in equation (where) comapared to a two-layer finite width $N = 100$ ReLU network trained with $\gamma = 0.05$ on $P = 10$ random training data points. (a) The average dynamics over an ensemble of $E = 500$ networks (solid colors) compared to the infinite width predictions (dashed black). (b) The predicted finite size variance for each eigenmode of the error $\Delta_k(t) = \boldsymbol{\Delta}(t) \cdot \boldsymbol{\phi}_k$. These are not ordered simply by magnitude of eigenvalues or the target projections $y_k = \boldsymbol{y} \cdot \boldsymbol{\phi}_k$, but rather depend on all eigenvalue gaps $\lambda_k - \lambda_\ell$ for $k \neq \ell$ and also the $\kappa_{k\ell nm}$ tensor. (c) The total variance for all training points $N \sum_\mu \text{Var}\Delta_\mu(t) = N \sum_k \text{Var}\Delta_k(t)$ is also well predicted by the DMFT propagator equations.



(a) Fourth Moment $\kappa(t, t)$ (b) Kernel sensitivity $D(t, 100)$

Figure A.2: The κ and D functions for varying γ in Figure 2. (a) The uncoupled kernel variance $\kappa(t, t)$ increases monotonically with γ. This reveals that the dynamical filtering of κ is what is responsible for the late time decrease in variance during feature learning. (b) The tensor $D(t, s)$ measures sensitivity of kernel at time t to perturbation in Δ at time s. The $D(t, s)$ entries also increase with γ. This suggests that the reduction in variance of the training error and the kernel are not due to reduction in κ, but rather a dynamical filtering effect due to scale growth in K_∞ and rapid reduction in Δ_∞.



(a) Online function variance (b) Online leading correction breaks down for large D

Figure A.3: Online learning follows identical finite size effects to offline training in two layer linear networks. (a) The variance of $\boldsymbol{\beta}(t)$ in online learning vs input dimension D. (b) The predicted leading correction to the generalization error is accurate for $N \gg D$ but breaks down as D becomes comparable to N.

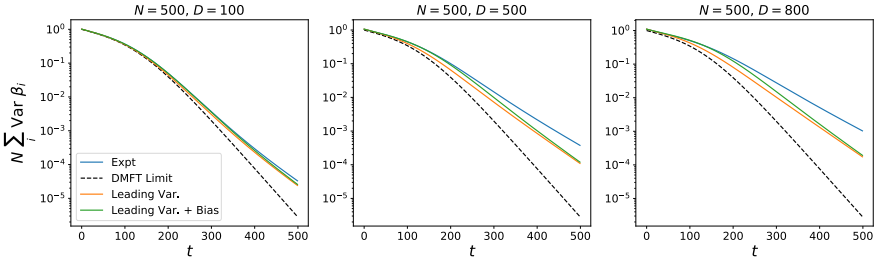


Figure A.4: A comparison of the bias and variance corrections in the toy model of Figure 3. At small D/N (or P/N for offline training) the leading variance and the leading variance and leading bias both track the experiment. Both the bias and the variance contribute positively towards the total generalization error since the variance correction alone (orange) exceeds the DMFT limiting error (dashed) and the variance and bias correction together (green) exceed variance alone (orange). However, for large D/N (or P/N) the leading order picture fails to describe the finite width experiment, indicating significant variance possibly at higher order scales (like $D^2/N^2, D^3/N^3$, ...).

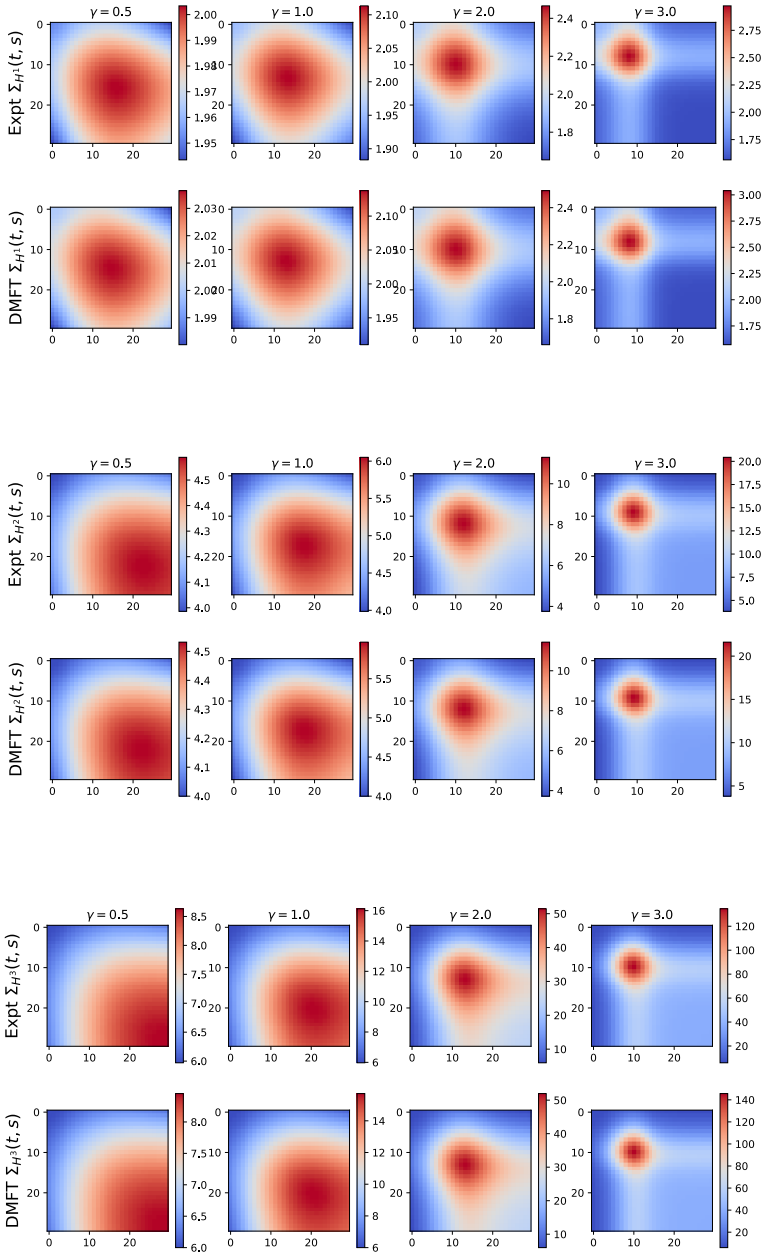
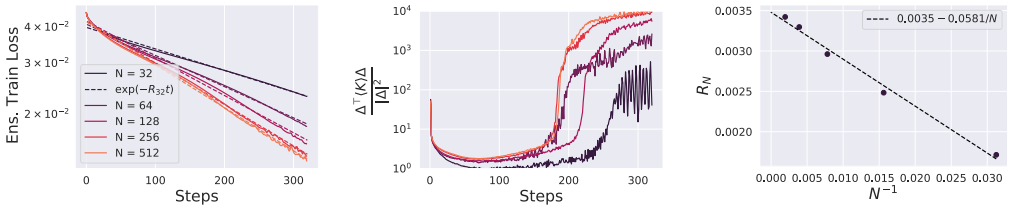
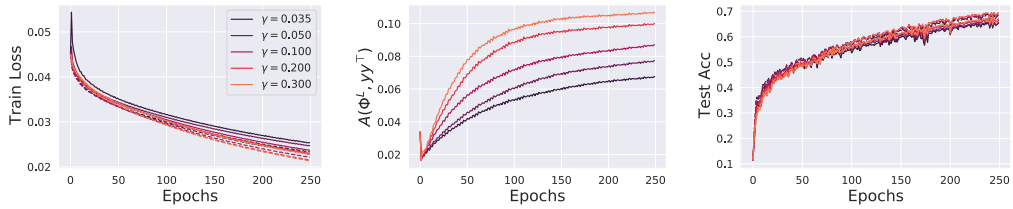


Figure A.5: The covariance of kernel entries across pairs of time points $\Sigma_{H^\ell}(t,s) = N\ \mathrm{Cov}(H^\ell(t,t), H^\ell(s,s))$ for depth 4 linear network trained on whitened data. The variance becomes increasingly localized in time as feature learning γ increases.

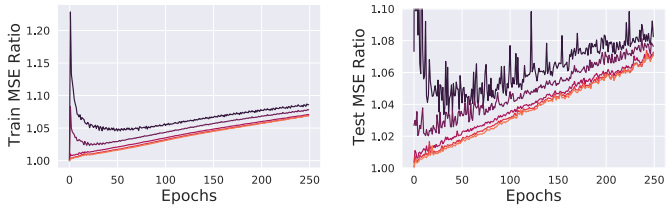


(a) Ens. Averaged Train Loss (b) Raleigh Quotient $\frac{\boldsymbol{\Delta}^\top \boldsymbol{K} \boldsymbol{\Delta}}{|\boldsymbol{\Delta}|^2}$ (c) Rates as Function of N

Figure A.6: The ensemble averaged train loss for the same depth 6 CNN trained on a random subsample of $P = 64$ CIFAR-10 points. Training is full batch gradient descent with $\gamma = 0.05$. (a) The ensemble train accuracy for this subset of CIFAR-10 is well modeled as an exponential in time $\mathcal{L}(t) \propto \exp(-R_N t)$ with a rate R_N that depends on width. (b) The projection of the errors Δ on the average NTK $\langle K \rangle$ (which is related to the rate of decay of the training loss, see Appendix F) reveals that wider networks are more aligned with their instantaneous error signals. (c) The rates R_N are indeed a linear functions of N^{-1}, with $R_N = R_\infty + \frac{R^1}{N}$, consistent with the average NTK $\langle K \rangle$ receiving a N^{-1} correction. Using ensembling to find a scaling law like that above can thus allow one extrapolate the training rate of infinite width mean field models.



(a) Training MSE (b) Kernel-Task Alignment (c) Test Acc.



(d) Single vs Ensembled Ratio (e) Single vs Ensembled Ratio

Figure A.7: Width $N = 64$ depth 6 CNNs trained on the full CIFAR-10 with MSE. An ensemble of size $E = 10$ randomly initialized networks are trained. (a) Training MSE for varying γ. (b) Final layer kernel-task alignment does strongly depend on γ, despite similar train dynamics. (c) Top-1 classification test accuracy is only slightly different across γ. A small benefit from ensembling is visible late in training. (c) Initialization variance (measured by the ratio of single model to ensembled MSE) for training and test losses. Richer networks have lower variance throughout training. (b) Networks have distinct kernel dynamics when trained with different γ as evidenced by the alignment (cosine similarity) between the final layer feature kernel Φ^L and the target test labels $\boldsymbol{y}$.

B CIFAR-10 Experimental Details

We trained the following depth 6 CNN architecture in the mean field parameterization using FLAX [58] on a single GPU. The bias parameters were zero in each hidden Conv layer, but were used for the readout weights. The networks were trained with MSE loss on centered 10 dimensional targets $\boldsymbol{y}_\mu \in \mathbb{R}^{10}$ for $\mu \in [P]$. Each convolution was followed by an average pooling operation. To obtain mean field behavior, NTK parameterization with a modified final layer is used [7, 9].

```
from flax import linen as nn
import jax.numpy as jnp

class CNN(nn.Module):

  width: int

  def setup(self):
    kif = nn.initializers.normal(stddev = 1.0) # O_N(1) entries
    self.conv1 = nn.Conv(features = self.width, kernel_init = kif,
    use_bias = False, kernel_size = (3,3))
    self.conv2 = nn.Conv(features = self.width, kernel_init = kif,
    use_bias = False, kernel_size = (3,3))
    self.conv3 = nn.Conv(features = self.width, kernel_init = kif,
    use_bias = False, kernel_size = (3,3))
    self.conv4 = nn.Conv(features = self.width, kernel_init  = kif,
    use_bias = False, kernel_size = (3,3))
    self.conv5 = nn.Conv(features = self.width, kernel_init  = kif,
    use_bias = False, kernel_size = (3,3))
    self.readout = nn.Dense(features = 10, use_bias = True,
    kernel_init = kif)
    return

  def __call__(self, x, train = True):
    N = self.width
    D = 3
    x = self.conv1(x) / jnp.sqrt(D * 9)
    x = jnp.sqrt(2.0) * nn.relu(x)
    x = nn.avg_pool(x, window_shape=(2,2), strides = (2,2)) # 32 x 32
    -> 16 x 16
    x = self.conv2(x) / jnp.sqrt(N*9) # explicit N^{-1/2}
    x = jnp.sqrt(2.0) * nn.relu(x)
    x = nn.avg_pool(x, window_shape=(2,2), strides = (2,2)) # 16 x 16
    -> 8 x 8
    x = self.conv3(x)/jnp.sqrt(N*9)
    x = jnp.sqrt(2.0) * nn.relu(x)
    x = nn.avg_pool(x, window_shape=(2,2), strides = (2,2)) # 8 x 8 ->
     4 x 4
    x = self.conv4(x) / jnp.sqrt(N*9)
    x = jnp.sqrt(2.0) * nn.relu(x)
    x = nn.avg_pool(x, window_shape=(2,2), strides = (2,2)) #  4 x 4
    -> 2 x 2
    x = self.conv5(x) / jnp.sqrt(N*9)
    x = jnp.sqrt(2.0) * nn.relu(x)
    x = nn.avg_pool(x, window_shape=(2,2), strides = (2,2)) #  2 x 2
    -> 1 x 1
    x = x.reshape((x.shape[0], -1))  # flatten
    x = self.readout(x) / N # for mean field scaling
    return x
```

All models were trained with standard SGD with a batch size of 256. Each element in the ensemble of E networks is trained on identical batches presented in identical order. For the Figure 6 experiments, the raw learning rate is scaled as $\eta = 10N\sqrt{\gamma}$ with $\gamma = 0.2$ (note that mean field theory requires scaling the raw learning rate linearly with N since the raw NTK is $\mathcal{O}(N^{-1})$ [9]). For Figure A.7, the learning rate is $\eta = 5N\sqrt{\gamma}$. We find that choosing $\eta \propto \sqrt{\gamma}$ gives approximately conserved training

times across γ (though distinct representation dynamics). The Figure A.6 shows the dynamics of fitting $P = 64$ training points with full batch gradient descent and $\gamma = 0.1$.

C Cumulant Expansion of Observables

We are interested in a principled power series expansion (in $1/N$) of any observable average $\langle O(\boldsymbol{q})\rangle$ that depends on DMFT order parameters $\boldsymbol{q}$. At any width N the observable average takes the form

$$\langle O(\boldsymbol{q})\rangle_N = \frac{\int d\boldsymbol{q} \exp\left(NS(\boldsymbol{q})\right) O(\boldsymbol{q})}{\int d\boldsymbol{q} \exp\left(NS(\boldsymbol{q})\right)} \tag{11}$$

As discussed in the main text, the $N \to \infty$ limit gives $\langle O(\boldsymbol{q})\rangle_N \sim O(\boldsymbol{q}_\infty)$ where $\frac{\partial S}{\partial \boldsymbol{q}}|_{\boldsymbol{q}_\infty} = 0$ by a steepest descent argument [55]. We assume that S's Hessian is negative semidefinite so that $\boldsymbol{\Sigma} \equiv -\left[\nabla^2 S(\boldsymbol{q})|_{\boldsymbol{q}_\infty}\right]^{-1} \succeq 0$ and Taylor expand $S(\boldsymbol{q})$ around the saddle point $\boldsymbol{q}_\infty$ giving $S(\boldsymbol{q}) = S(\boldsymbol{q}_\infty) + \frac{1}{2}(\boldsymbol{q}-\boldsymbol{q}_\infty)^\top \nabla^2 S(\boldsymbol{q})(\boldsymbol{q}-\boldsymbol{q}_\infty) + V(\boldsymbol{q}-\boldsymbol{q}_\infty)$. We note that the remainder function V contains only cubic and higher powers of $\boldsymbol{q}-\boldsymbol{q}_\infty \equiv \boldsymbol{\delta}/\sqrt{N}$. The variable $\boldsymbol{\delta}$ will be order $\mathcal{O}(1)$. This will allow us to verify that additional terms are suppressed in powers of $1/N$. Expanding both the numerator and denominator's integrands in powers of V, we find

$$\begin{aligned}
\langle O(\boldsymbol{q})\rangle_N =& \frac{\int d\boldsymbol{q} \exp\left(-\frac{N}{2}(\boldsymbol{q}-\boldsymbol{q}_\infty)^\top \boldsymbol{\Sigma}^{-1}(\boldsymbol{q}-\boldsymbol{q}_\infty) + NV(\boldsymbol{q}-\boldsymbol{q}_\infty)\right) O(\boldsymbol{q})}{\int d\boldsymbol{q} \exp\left(-\frac{N}{2}(\boldsymbol{q}-\boldsymbol{q}_\infty)^\top \boldsymbol{\Sigma}^{-1}(\boldsymbol{q}-\boldsymbol{q}_\infty) + NV(\boldsymbol{q}-\boldsymbol{q}_\infty)\right)} \\
=& \frac{\int d\boldsymbol{\delta} \exp\left(-\frac{1}{2}\boldsymbol{\delta}^\top \boldsymbol{\Sigma}^{-1}\boldsymbol{\delta}\right)\left(1 + NV + \frac{N^2}{2}V^2 + ...\right) O(\boldsymbol{q}_\infty + N^{-1/2}\boldsymbol{\delta})}{\int d\boldsymbol{\delta} \exp\left(-\frac{1}{2}\boldsymbol{\delta}^\top \boldsymbol{\Sigma}^{-1}\boldsymbol{\delta}\right)\left(1 + NV + \frac{N^2}{2}V^2 + ...\right)} \\
=& \frac{\langle O\rangle_\infty + N\langle VO\rangle_\infty + \frac{N^2}{2!}\langle V^2 O\rangle_\infty + \frac{N^3}{3!}\langle V^3 O\rangle_\infty + ...}{1 + N\langle V\rangle_\infty + \frac{N^2}{2!}\langle V^2\rangle_\infty + \frac{N^3}{3!}\langle V^3\rangle_\infty + ...} \\
=& \langle O\rangle_\infty \frac{1 + N\langle VO\rangle_\infty / \langle O\rangle_\infty + \frac{N^2}{2!}\langle V^2 O\rangle_\infty / \langle O\rangle_\infty + \frac{N^3}{3!}\langle V^3 O\rangle_\infty / \langle O\rangle_\infty + ...}{1 + N\langle V\rangle_\infty + \frac{N^2}{2!}\langle V^2\rangle_\infty + \frac{N^3}{3!}\langle V^3\rangle_\infty + ...}
\end{aligned} \tag{12}$$

where $\langle\rangle_\infty$ represents an average over the Gaussian fluctuation $\mathcal{N}\left(\boldsymbol{q}_\infty, -\frac{1}{N}\left[\nabla^2_{\boldsymbol{q}} S(\boldsymbol{q}_\infty)\right]^{-1}\right)$. We see that the series in the denominator contains terms of the form $\frac{N^k}{k!}\langle V^k\rangle_\infty$ while the numerator depends on terms of the form $\frac{N^k}{k!}\langle V^k O\rangle_\infty / \langle O\rangle_\infty$. In either of these power series, the k-th term can contribute at most

$$\frac{N^k \langle V^k O\rangle_\infty}{\langle O\rangle_\infty}\ ,\ N^k\langle V^k\rangle_\infty \sim \begin{cases} \mathcal{O}(N^{-(k+1)/2}) & k \text{ odd} \\ \mathcal{O}(N^{-k/2}) & k \text{ even} \end{cases} \tag{13}$$

since V contributes only cubic and higher terms. Thus each term in the numerator and denominator's series contains increasing powers of $1/N$. Concretely, each of the two series have terms of order $\{N^0, N^{-1}, N^{-1}, N^{-2}, N^{-2}, ...\}$. Thus any quantity of the form $\frac{\langle O\rangle}{\langle O\rangle_\infty}$ admits a ratio of power series in powers of $1/N$. One could truncate each of the series in the numerator and denominator to a desired order in N. Alternatively, the denominator could be expanded giving a single series (the cumulant expansion [56]). The first few terms in the cumulant expansion have the form

$$\begin{aligned}
\langle O\rangle_N =& \langle O\rangle_\infty + N\left[\langle OV\rangle_\infty - \langle O\rangle_\infty \langle V\rangle_\infty\right] \\
&+ \frac{N^2}{2}\left[\langle V^2 O\rangle_\infty - 2\langle VO\rangle_\infty \langle V\rangle_\infty + 2\langle V\rangle^2_\infty \langle O\rangle_\infty - \langle V^2\rangle_\infty \langle O\rangle_\infty\right] + ...
\end{aligned} \tag{14}$$

In this work, we mainly are interested in the leading order correction to $\langle O\rangle$ which can always be obtained with the truncation after the terms linear in V for any observable O.

C.1 Square Deviation from DMFT

We will now analyze the fluctuation statistics of our order parameters around the saddle point $\left\langle (\boldsymbol{q}-\boldsymbol{q}_\infty)(\boldsymbol{q}-\boldsymbol{q}_\infty)^\top \right\rangle_N$ which has the form

$$\left\langle (\boldsymbol{q}-\boldsymbol{q}_\infty)(\boldsymbol{q}-\boldsymbol{q}_\infty)^\top \right\rangle_N = \frac{\left\langle (\boldsymbol{q}-\boldsymbol{q}_\infty)(\boldsymbol{q}-\boldsymbol{q}_\infty)^\top \right\rangle_\infty + N \left\langle V\,(\boldsymbol{q}-\boldsymbol{q}_\infty)(\boldsymbol{q}-\boldsymbol{q}_\infty)^\top \right\rangle_\infty + \ldots}{1 + N\left\langle V \right\rangle_\infty + \ldots}$$
$$= \left[\frac{\frac{1}{N}\boldsymbol{\Sigma} + \mathcal{O}(N^{-2})}{1+\mathcal{O}(N^{-1})} \right] \sim \frac{1}{N}\boldsymbol{\Sigma} + \mathcal{O}(N^{-2}), \tag{15}$$

as stated in the main text and verified empirically in Figure 2 (a). The reason that the terms in the numerator involving V can be no larger than $\mathcal{O}(N^{-2})$ comes from vanishing of odd moments for $\boldsymbol{q}-\boldsymbol{q}_\infty$ in the unperturbed distribution. Thus the leading expression for $\left\langle (\boldsymbol{q}-\boldsymbol{q}_\infty)(\boldsymbol{q}-\boldsymbol{q}_\infty)^\top \right\rangle$ only depends on $\boldsymbol{\Sigma}$ and not on V.

C.2 Mean Deviation from DMFT

Although the square displacement from DMFT only depended on Σ and not on V, we note that the *average order parameter displacement* $\langle \boldsymbol{q}-\boldsymbol{q}_\infty \rangle$ does receive a $\mathcal{O}(1/N)$ correction that depends on the perturbed potential V

$$\langle \boldsymbol{q}-\boldsymbol{q}_\infty \rangle_N = \frac{\langle \boldsymbol{q}-\boldsymbol{q}_\infty \rangle_\infty + N\langle (\boldsymbol{q}-\boldsymbol{q}_\infty)V \rangle_\infty + \frac{N^2}{2}\left\langle (\boldsymbol{q}-\boldsymbol{q}_\infty)V^2 \right\rangle_\infty + \ldots}{1 + N\langle V\rangle_\infty + \frac{N^2}{2}\langle V^2\rangle_\infty + \ldots}$$
$$\sim \frac{\boldsymbol{\Sigma}\left\langle \frac{\partial V}{\partial \boldsymbol{q}} \right\rangle_\infty + \mathcal{O}(N^{-2})}{1+\mathcal{O}(N^{-1})} \sim \boldsymbol{\Sigma}\left\langle \frac{\partial V}{\partial \boldsymbol{q}} \right\rangle_\infty + \mathcal{O}(N^{-2}). \tag{16}$$

where in the last line we used Stein's lemma (Gaussian integration by parts) for the Gaussian distribution over $\boldsymbol{q}$. Note that $\left\langle \frac{\partial V}{\partial \boldsymbol{q}} \right\rangle_\infty \sim \mathcal{O}\left(\frac{1}{N}\right)$ since the derivative of the cubic term in V gives a quadratic function of $\boldsymbol{q}-\boldsymbol{q}_\infty$, whose average must be $\mathcal{O}(N^{-1})$. In this work, we focus primarily on the structure of the propagator, but outline a general recipe for getting the leading mean correction in Appendix F and G.2.

C.3 Covariance of Order Parameters

Lastly, we combine the previous two observations to reason about the scaling of the order parameter covariance over initializations. We note that the leading covariance of the order parameters over random initializations is also given by the propagator: $\text{Cov}(\boldsymbol{q}) \sim \frac{1}{N}\boldsymbol{\Sigma} + \mathcal{O}(N^{-1})$, since

$$\begin{aligned}\text{Cov}(\boldsymbol{q}) &= \left\langle (\boldsymbol{q}-\langle \boldsymbol{q}\rangle_N)(\boldsymbol{q}-\langle \boldsymbol{q}\rangle_N)^\top \right\rangle_N \\ &= \left\langle (\boldsymbol{q}-\boldsymbol{q}_\infty)(\boldsymbol{q}-\boldsymbol{q}_\infty)^\top \right\rangle_N - \left\langle (\boldsymbol{q}_\infty-\langle \boldsymbol{q}\rangle_N)(\boldsymbol{q}_\infty-\langle \boldsymbol{q}\rangle_N)^\top \right\rangle_N \\ &\sim \frac{1}{N}\boldsymbol{\Sigma} + \mathcal{O}(N^{-2})\end{aligned} \tag{17}$$

due to the arguments above which showed that $\left\langle (\boldsymbol{q}-\boldsymbol{q}_\infty)(\boldsymbol{q}-\boldsymbol{q}_\infty)^\top \right\rangle \sim \frac{1}{N}\boldsymbol{\Sigma} + \mathcal{O}(N^{-2})$ and that $\boldsymbol{q}_\infty - \langle \boldsymbol{q}\rangle_N \sim \mathcal{O}(N^{-1})$. Therefore, in the leading order picture, it is safe to associate $\boldsymbol{\Sigma}$ with the covariance of order parameters over random initializations of the network weights.

D Propagator Structure for the full DMFT Action

In this section, we examine the propagator structure for the full DMFT action. This action is modified from other prior works [9, 46] to include the evolution of the network prediction errors $\Delta(t)$. Those prior works noted that Δ and the NTK K are deterministic functions of deterministic order parameters $\{\Phi^\ell, G^\ell\}$ in the $N\to\infty$ limit so those authors did not explicitly include Δ or K in the action. At finite width N, including Δ, K in the action is crucial as the fluctuation in prediction errors Δ has significant consequences for dynamical fluctuations of kernels through the preactivation and

pre-gradient fields. In this section, we will mainly focus on gradient flow, but we describe large step size in Appendix L.

$$
\begin{aligned}
S =& \sum_{\ell\mu\nu} \int dtds \left[\hat{\Phi}^{\ell}_{\mu\nu}(t,s)\Phi^{\ell}_{\mu\nu}(t,s) + \hat{G}^{\ell}_{\mu\nu}(t,s)G^{\ell}_{\mu\nu}(t,s) - \gamma^2 A^{\ell}_{\nu\mu}(s,t)B^{\ell}_{\mu\nu}(t,s) \right] \\
&+ \sum_{\mu} \int dt \hat{\Delta}_{\mu}(t) \left[\Delta_{\mu}(t) - y_{\mu} + \sum_{\nu} \int ds \Theta(t-s) K_{\mu\nu}(s) \Delta_{\nu}(s) \right] \\
&+ \sum_{\mu\nu} \int dt \hat{K}_{\mu\nu}(t) \left[K_{\mu\nu}(t) - \sum_{\ell} G^{\ell+1}_{\mu\nu}(t) \Phi^{\ell}_{\mu\nu}(t) \right] \\
&+ \sum_{\ell} \ln \mathcal{Z}_{\ell}[\boldsymbol{\Delta}, \hat{\boldsymbol{\Phi}}^{\ell}, \hat{\boldsymbol{G}}^{\ell}, \boldsymbol{\Phi}^{\ell-1}, \boldsymbol{G}^{\ell+1}, \boldsymbol{A}^{\ell-1}, \boldsymbol{B}^{\ell}]
\end{aligned} \tag{18}
$$

where the single site moment generating functionals $\mathcal{Z}_{\ell}$ have the form

$$
\begin{aligned}
\mathcal{Z}_{\ell} =& \mathbb{E}_{\{h^{\ell}_{\mu}(t), z^{\ell}_{\mu}(t)\}} \exp\left(- \sum_{\mu\nu} \int dtds \left[\phi(h^{\ell}_{\mu}(t)) \phi(h^{\ell}_{\nu}(s)) \hat{\Phi}^{\ell}_{\mu\nu}(t,s) + g^{\ell}_{\mu}(t) g^{\ell}_{\nu}(s) \hat{G}^{\ell}_{\mu\nu}(t,s) \right] \right) \\
h^{\ell}_{\mu}(t) =& u^{\ell}_{\mu}(t) + \gamma \int_0^t ds \sum_{\nu} \left[\Phi^{\ell-1}_{\mu\nu}(t,s) \Delta_{\nu}(s) + A^{\ell-1}_{\mu\nu}(t,s) \right] g^{\ell}_{\nu}(s) \ , \ \{u^{\ell}_{\mu}(t)\} \sim \mathcal{GP}(0, \boldsymbol{\Phi}^{\ell-1}) \\
z^{\ell}_{\mu}(t) =& r^{\ell}_{\mu}(t) + \gamma \int_0^t ds \sum_{\nu} \left[G^{\ell+1}_{\mu\nu}(t,s) \Delta_{\nu}(s) + B^{\ell}_{\mu\nu}(t,s) \right] \phi(h^{\ell}_{\nu}(s)) \ , \ \{r^{\ell}_{\mu}(t)\} \sim \mathcal{GP}(0, \boldsymbol{G}^{\ell+1})
\end{aligned} \tag{19}
$$

with $g^{\ell}_{\mu}(t) = \dot{\phi}(h^{\ell}_{\mu}(t)) z^{\ell}_{\mu}(t)$. The saddle point equations give the infinite width evolution of our order parameters.

$$
\begin{aligned}
\frac{\partial S}{\partial \hat{\Phi}^{\ell}_{\mu\nu}(t,s)} &= \Phi^{\ell}_{\mu\nu}(t,s) - \left\langle \phi(h^{\ell}_{\mu}(t)) \phi(h^{\ell}_{\nu}(s)) \right\rangle = 0 \\
\frac{\partial S}{\partial \hat{G}^{\ell}_{\mu\nu}(t,s)} &= G^{\ell}_{\mu\nu}(t,s) - \left\langle g^{\ell}_{\mu}(t) g^{\ell}_{\nu}(s) \right\rangle = 0 \\
\frac{\partial S}{\partial A^{\ell}_{\nu\mu}(s,t)} &= -\gamma^2 B^{\ell}_{\mu\nu}(t,s) + \gamma \left\langle \frac{\partial \phi(h^{\ell}_{\mu}(t))}{\partial r^{\ell}_{\nu}(s)} \right\rangle = 0 \\
\frac{\partial S}{\partial B^{\ell}_{\nu\mu}(s,t)} &= -\gamma^2 A^{\ell}_{\mu\nu}(t,s) + \gamma \left\langle \frac{\partial g^{\ell}_{\mu}(t)}{\partial u^{\ell}_{\nu}(s)} \right\rangle = 0 \\
\frac{\partial S}{\partial \hat{K}_{\mu\nu}(t)} &= K_{\mu\nu}(t) - \sum_{\ell} G^{\ell+1}_{\mu\nu}(t,t) \Phi^{\ell}_{\mu\nu}(t,t) = 0 \\
\frac{\partial S}{\partial \hat{\Delta}_{\mu}(t)} &= \Delta_{\mu}(t) - y_{\mu} + \int_0^t ds \sum_{\nu} K_{\mu\nu}(s) \Delta_{\nu}(s) = 0
\end{aligned} \tag{20}
$$

These equations exactly recover the mean field description obtained [9]. Note that $\langle \rangle$ for field averages is an average defined by $\mathcal{Z}_{\ell}$ and is distinct from the types averages $\langle \rangle$, $\langle \rangle_{\infty}$ we have been considering over the order parameters $\boldsymbol{q}$. The complementary set of equations for the primal variables, such as $\frac{\partial S}{\partial \Phi^{\ell}_{\mu\nu}(t,s)} = 0$, give that $\hat{K} = \hat{\Delta} = \hat{\Phi} = \hat{G} = 0$ at the saddle point. We now set out to compute the Hessan $\nabla^2_{\boldsymbol{q}} S$. To simplify the set of expressions, we will only explicitly write out the nonvanishing

blocks. We will start with second derivatives involving only pairs of dual variables $\{\hat{\Phi}, \hat{G}, A, B\}$

$$
\begin{aligned}
\frac{\partial^2 S}{\partial \hat{\Phi}^\ell_{\mu\nu}(t,s)\partial \hat{\Phi}^\ell_{\alpha\beta}(t',s')} &= \left\langle \phi(h^\ell_\mu(t))\phi(h^\ell_\nu(s))\phi(h^\ell_\alpha(t'))\phi(h^\ell_\beta(s'))\right\rangle - \Phi^\ell_{\mu\nu}(t,s)\Phi^\ell_{\alpha\beta}(t',s') \\
&\equiv \kappa^{\Phi^\ell}_{\mu\nu\alpha\beta}(t,s,t',s') \\
\frac{\partial^2 S}{\partial \hat{G}^\ell_{\mu\nu}(t,s)\partial \hat{G}^\ell_{\alpha\beta}(t',s')} &= \left\langle g^\ell_\mu(t) g^\ell_\nu(s) g^\ell_\alpha(t') g^\ell_\beta(s')\right\rangle - G^\ell_{\mu\nu}(t,s)G^\ell_{\alpha\beta}(t',s') \\
&\equiv \kappa^{G^\ell}_{\mu\nu\alpha\beta}(t,s,t',s') \\
\frac{\partial^2 S}{\partial \hat{\Phi}^\ell_{\mu\nu}(t,s)\partial \hat{G}^\ell_{\alpha\beta}(t',s')} &= \left\langle \phi(h^\ell_\mu(t))\phi(h^\ell_\nu(s)) g^\ell_\alpha(t') g^\ell_\beta(s')\right\rangle - \Phi^\ell_{\mu\nu}(t,s)G^\ell_{\alpha\beta}(t',s') \\
&\equiv \kappa^{\Phi^\ell G^\ell}_{\mu\nu\alpha\beta}(t,s,t',s') \\
\frac{\partial^2 S}{\partial \hat{\Phi}^\ell_{\mu\nu}(t,s)\partial A^{\ell-1}_{\beta\alpha}(s',t')} &= -\gamma \left\langle \frac{\partial \phi(h^\ell_\mu(t))}{\partial u^\ell_\beta(s')}\phi(h^\ell_\nu(s)) g^\ell_\alpha(t')\right\rangle \\
&\quad - \gamma \left\langle \phi(h^\ell_\mu(t))\frac{\partial \phi(h^\ell_\nu(t))}{\partial u^\ell_\beta(s')} g^\ell_\alpha(t')\right\rangle \\
&\quad - \gamma \left\langle \phi(h^\ell_\mu(t))\phi(h^\ell_\nu(s))\frac{\partial g^\ell_\alpha(t')}{\partial u^\ell_\beta(s')}\right\rangle - \gamma^2 \Phi^\ell_{\mu\nu}(t,s)B^{\ell-1}_{\alpha\beta}(t',s') \\
&\equiv -\gamma \kappa^{\Phi^\ell B^{\ell-1}}_{\mu\nu\alpha\beta}(t,s) \\
\frac{\partial^2 S}{\partial \hat{\Phi}^\ell_{\mu\nu}(t,s)\partial B^\ell_{\beta\alpha}(s',t')} &= -\gamma \left\langle \frac{\partial \phi(h^\ell_\mu(t))}{\partial r^\ell_\beta(s')}\phi(h^\ell_\nu(s))\phi(h^\ell_\alpha(t'))\right\rangle \\
&\quad - \gamma \left\langle \phi(h^\ell_\mu(t))\frac{\partial \phi(h^\ell_\nu(t))}{\partial r^\ell_\beta(s')}\phi(h^\ell_\alpha(t'))\right\rangle \\
&\quad - \gamma \left\langle \phi(h^\ell_\mu(t))\phi(h^\ell_\nu(s))\frac{\partial \phi(h^\ell_\alpha(t'))}{\partial r^\ell_\beta(s')}\right\rangle - \gamma^2 \Phi^\ell_{\mu\nu}(t,s)A^\ell_{\alpha\beta}(t',s') \\
&\equiv -\gamma \kappa^{\Phi^\ell A^\ell}_{\mu\nu\alpha\beta}(t,s) \\
\frac{\partial^2 S}{\partial \hat{G}^\ell_{\mu\nu}(t,s)\partial A^{\ell-1}_{\beta\alpha}(s',t')} &= -\gamma \left\langle \frac{\partial g^\ell_\mu(t)}{\partial u^\ell_\beta(s')} g^\ell_\nu(s) g^\ell_\alpha(t')\right\rangle - \gamma \left\langle g^\ell_\mu(t)\frac{\partial g^\ell_\nu(t)}{\partial u^\ell_\beta(s')} g^\ell_\alpha(t')\right\rangle \\
&\quad - \gamma \left\langle g^\ell_\mu(t) g^\ell_\nu(s)\frac{\partial g^\ell_\alpha(t')}{\partial u^\ell_\beta(s')}\right\rangle - \gamma^2 G^\ell_{\mu\nu}(t,s)B^{\ell-1}_{\alpha\beta}(t',s') \\
&\equiv -\gamma \kappa^{G^\ell B^{\ell-1}}_{\mu\nu\alpha\beta}(t,s) \\
\frac{\partial^2 S}{\partial \hat{G}^\ell_{\mu\nu}(t,s)\partial B^\ell_{\beta\alpha}(s',t')} &= -\gamma \left\langle \frac{\partial g^\ell_\mu(t)}{\partial r^\ell_\beta(s')} g^\ell_\nu(s)\phi(h^\ell_\alpha(t'))\right\rangle - \gamma \left\langle g^\ell_\mu(t)\frac{\partial g^\ell_\nu(t)}{\partial r^\ell_\beta(s')}\phi(h^\ell_\alpha(t'))\right\rangle \\
&= -\gamma \left\langle g^\ell_\mu(t) g^\ell_\nu(s)\frac{\partial \phi(h^\ell_\alpha(t'))}{\partial r^\ell_\beta(s')}\right\rangle - \gamma^2 G^\ell_{\mu\nu}(t,s)A^\ell_{\alpha\beta}(t',s') \\
&\equiv -\gamma \kappa^{G^\ell A^\ell}_{\mu\nu\alpha\beta}(t,s) \\
\frac{\partial^2 S}{\partial A^\ell_{\mu\nu}(t,s)\partial B^\ell_{\beta\alpha}(s',t')} &= -\gamma^2 \delta_{\mu\alpha}\delta_{\nu\beta}\delta(t-t')\delta(s-s') \\
\frac{\partial^2 S}{\partial A^{\ell-1}_{\nu\mu}(s,t)\partial B^\ell_{\beta\alpha}(s',t')} &= \gamma^2 \left\langle \frac{\partial^2}{\partial u^\ell_\nu(s)\partial r^\ell_\beta(s')}\left[g^\ell_\mu(t)\phi(h^\ell_\alpha(t'))\right]\right\rangle - \gamma^4 B^{\ell-1}_{\mu\nu}(t,s)A^\ell_{\alpha\beta}(t',s') \\
&\equiv \kappa^{B^{\ell-1}A^\ell}_{\mu\nu\alpha\beta}(t,s,t',s') \qquad (21)
\end{aligned}
$$

Next, we consider the second derivatives involving only primal variables $\{\Phi^\ell, G^\ell, K, \Delta\}$ which all vanish

$$
\begin{aligned}
&\frac{\partial^2 S}{\partial \Phi^\ell_{\mu\nu}(t,s)\partial \Phi^{\ell'}_{\alpha\beta}(t',s')} = 0 \\
&\frac{\partial^2 S}{\partial G^\ell_{\mu\nu}(t,s)\partial G^{\ell'}_{\alpha\beta}(t',s')} = 0 \\
&\frac{\partial^2 S}{\partial \Phi^\ell_{\mu\nu}(t,s)\partial G^{\ell'}_{\alpha\beta}(t',s')} = 0 \\
&\frac{\partial^2 S}{\partial \Phi^\ell_{\mu\nu}(t,s)\partial K_{\alpha\beta}(s')} = 0 \\
&\frac{\partial^2 S}{\partial G^\ell_{\mu\nu}(t,s)\partial K_{\alpha\beta}(s')} = 0 \\
&\frac{\partial^2 S}{\partial \Phi^\ell_{\mu\nu}(t,s)\partial \Delta_\alpha(s')} = 0 \\
&\frac{\partial^2 S}{\partial G^\ell_{\mu\nu}(t,s)\partial \Delta_\alpha(s')} = 0 \\
&\frac{\partial^2 S}{\partial K_{\mu\nu}(t)\partial K_{\alpha\beta}(s)} = 0 \\
&\frac{\partial^2 S}{\partial K_{\mu\nu}(t)\partial \Delta_\alpha(s)} = 0 \\
&\frac{\partial^2 S}{\partial \Delta_\mu(t)\partial \Delta_\alpha(s)} = 0
\end{aligned}
\tag{22}
$$

Now we consider all derivatives which involve one of the dual variables $\{\hat{\Phi}^\ell, \hat{G}^\ell, A^\ell, B^\ell\}$ and the primal variable Δ

$$
\begin{aligned}
&\frac{\partial^2 S}{\partial \hat{\Phi}^\ell_{\mu\nu}(t,s)\partial \Delta_\alpha(t')} = -\left\langle \frac{\partial}{\partial \Delta_\alpha(t')}[\phi(h^\ell_\mu(t))\phi(h^\ell_\nu(s))] \right\rangle \equiv -D^{\Phi^\ell \Delta}_{\mu\nu\alpha}(t,s,t') \\
&\frac{\partial^2 S}{\partial \hat{G}^\ell_{\mu\nu}(t,s)\partial \Delta_\alpha(t')} = -\left\langle \frac{\partial}{\partial \Delta_\alpha(t')}[g^\ell_\mu(t) g^\ell_\nu(s)] \right\rangle \equiv -D^{G^\ell \Delta}_{\mu\nu\alpha}(t,s,t') \\
&\frac{\partial^2 S}{\partial A^{\ell-1}_{\nu\mu}(s,t)\partial \Delta_\alpha(t')} = \gamma \left\langle \frac{\partial}{\partial \Delta_\alpha(t')\partial u^\ell_\nu(s)} g^\ell_\mu(t) \right\rangle \equiv \gamma D^{B^{\ell-1},\Delta}_{\mu\nu\alpha}(t,s,t') \\
&\frac{\partial^2 S}{\partial B^\ell_{\nu\mu}(s,t)\partial \Delta_\alpha(t')} = \gamma \left\langle \frac{\partial}{\partial \Delta_\alpha(t')\partial r^\ell_\nu(s)} \phi(h^\ell_\mu(t)) \right\rangle \equiv \gamma D^{A^\ell \Delta}_{\mu\nu\alpha}(t,s,t')
\end{aligned}
$$

Now, we consider the second derivatives involving one derivative on a dual variable $\{\hat{\Phi}^\ell, \hat{G}^\ell, A, B\}$ and one of the primal variables $\{\Phi^\ell, G^\ell\}$.

$$\begin{aligned}
\frac{\partial^2 S}{\partial \hat{\Phi}^\ell_{\mu\nu}(t,s)\partial \Phi^{\ell'}_{\alpha\beta}(t',s')} &= \delta_{\ell,\ell'}\delta_{\mu\nu}\delta(t-t')\delta(s-s') \\
&\quad - \delta_{\ell-1,\ell'}\frac{\partial}{\partial \Phi^{\ell-1}_{\alpha\beta}(t',s')}\left\langle \phi(h^\ell_\mu(t))\phi(h^\ell_\nu(s))\right\rangle \\
&\equiv \delta_{\ell,\ell'}\delta_{\mu\nu}\delta(t-t')\delta(s-s') - \delta_{\ell-1,\ell'}D^{\Phi^\ell\Phi^{\ell-1}}_{\mu\nu\alpha\beta}(t,s,t',s') \\
\frac{\partial^2 S}{\partial \hat{G}^\ell_{\mu\nu}(t,s)\partial G^{\ell'}_{\alpha\beta}(t',s')} &= \delta_{\ell,\ell'}\delta_{\mu\nu}\delta(t-t')\delta(s-s') - \delta_{\ell+1,\ell'}\frac{\partial}{\partial G^{\ell+1}_{\alpha\beta}(t',s')}\left\langle g^\ell_\mu(t) g^\ell_\nu(s)\right\rangle \\
&\equiv \delta_{\ell,\ell'}\delta_{\mu\nu}\delta(t-t')\delta(s-s') - \delta_{\ell-1,\ell'}D^{G^\ell G^{\ell+1}}_{\mu\nu\alpha\beta}(t,s,t',s') \\
\frac{\partial^2 S}{\partial \hat{\Phi}^\ell_{\mu\nu}(t,s)\partial G^{\ell+1}_{\alpha\beta}(t',s')} &= -\frac{\partial}{\partial G^{\ell+1}_{\alpha\beta}(t',s')}\left\langle \phi(h^\ell_\mu(t))\phi(h^\ell_\nu(s))\right\rangle \equiv -D^{\Phi^\ell,G^{\ell+1}}_{\mu\nu\alpha\beta}(t,s,t',s') \\
\frac{\partial^2 S}{\partial \hat{G}^\ell_{\mu\nu}(t,s)\partial \Phi^{\ell-1}_{\alpha\beta}(t',s')} &= -\frac{\partial}{\partial \Phi^{\ell-1}_{\alpha\beta}(t',s')}\left\langle g^\ell_\mu(t) g^\ell_\nu(s)\right\rangle \equiv -D^{G^\ell,\Phi^{\ell-1}}_{\mu\nu\alpha\beta}(t,s,t',s') \\
\frac{\partial^2 S}{\partial A^{\ell-1}_{\nu\mu}(s,t)\partial \Phi^{\ell-1}_{\alpha\beta}(t',s')} &= \gamma\frac{\partial}{\partial \Phi^{\ell-1}_{\alpha\beta}(t',s')}\left\langle \frac{\partial g^\ell_\mu(t)}{\partial r^\ell_\nu(s)}\right\rangle \equiv \gamma D^{B^{\ell-1},\Phi^{\ell-1}}_{\mu\nu\alpha\beta}(t,s,t',s') \\
\frac{\partial^2 S}{\partial B^{\ell}_{\nu\mu}(s,t)\partial \Phi^{\ell-1}_{\alpha\beta}(t',s')} &= \gamma\frac{\partial}{\partial \Phi^{\ell-1}_{\alpha\beta}(t',s')}\left\langle \frac{\partial \phi(h^\ell_\mu(t))}{\partial u^\ell_\nu(s)}\right\rangle \equiv \gamma D^{A^\ell,\Phi^{\ell-1}}_{\mu\nu\alpha\beta}(t,s,t',s') \\
\frac{\partial^2 S}{\partial A^{\ell-1}_{\nu\mu}(s,t)\partial G^{\ell+1}_{\alpha\beta}(t',s')} &= \gamma\frac{\partial}{\partial G^{\ell+1}_{\alpha\beta}(t',s')}\left\langle \frac{\partial g^\ell_\mu(t)}{\partial r^\ell_\nu(s)}\right\rangle \equiv \gamma D^{B^{\ell-1},G^{\ell+1}}_{\mu\nu\alpha\beta}(t,s,t',s') \\
\frac{\partial^2 S}{\partial B^{\ell}_{\nu\mu}(s,t)\partial G^{\ell+1}_{\alpha\beta}(t',s')} &= \gamma\frac{\partial}{\partial G^{\ell+1}_{\alpha\beta}(t',s')}\left\langle \frac{\partial \phi(h^\ell_\mu(t))}{\partial u^\ell_\nu(s)}\right\rangle \equiv \gamma D^{A^\ell,G^{\ell+1}}_{\mu\nu\alpha\beta}(t,s,t',s')
\end{aligned} \tag{23}$$

We note that terms such as $\frac{\partial}{\partial \Phi^{\ell-1}_{\alpha\beta}(t',s')}\left\langle \phi(h^\ell_\mu(t))\phi(h^\ell_\nu(s))\right\rangle$ can be further decomposed since the average over the $\{u^\ell_\mu(t)\} \sim \mathcal{GP}(0, \mathbf{\Phi}^{\ell-1})$ and h^ℓ's explicit dynamics both depend on $\Phi^{\ell-1}$

$$\begin{aligned}
\frac{\partial}{\partial \Phi^{\ell-1}_{\alpha\beta}(t',s')}\left\langle \phi(h^\ell_\mu(t))\phi(h^\ell_\nu(s))\right\rangle &= \frac{1}{2}\left\langle \frac{\partial^2}{\partial u^\ell_\alpha(t')\partial u^\ell_\beta(s')}\phi(h^\ell_\mu(t))\phi(h^\ell_\nu(s))\right\rangle \\
&\quad + \left\langle \frac{\partial}{\partial \Phi^{\ell-1}_{\alpha\beta}(t',s')}\phi(h^\ell_\mu(t))\phi(h^\ell_\nu(s))\right\rangle
\end{aligned} \tag{24}$$

where the first term comes from differentiating the Gaussian probability density for u^ℓ (e.g. Price's theorem) and the second term is an explicit derivative of the preactivation fields with u^ℓ treated as constant. Next we consider the nonvanishing terms which involve $\{\hat{\Delta}, \hat{K}, \Delta, K\}$ which give

$$\begin{aligned}
\frac{\partial^2 S}{\partial \hat{\Delta}_\mu(t)\partial \Delta_\alpha(s)} &= \delta_{\mu\alpha}\delta(t-s) + \Theta(t-s)K_{\mu\alpha}(s) \\
\frac{\partial^2}{\partial \hat{\Delta}_\mu(t)\partial K_{\alpha\beta}(s)} &= \delta_{\mu\alpha}\Theta(t-s)\Delta_\beta(s) \\
\frac{\partial^2 S}{\partial \hat{K}_{\mu\nu}(t)\partial K_{\alpha\beta}(t')} &= \delta_{\mu\alpha}\delta_{\nu\beta}\delta(t-t') \\
\frac{\partial^2 S}{\partial \hat{K}_{\mu\nu}(t)\partial \Phi^{\ell}_{\alpha\beta}(t',s')} &= \delta_{\mu\alpha}\delta_{\nu\beta}G^{\ell+1}_{\alpha\beta}(t',s')\delta(t-t')\delta(t-s') \\
\frac{\partial^2 S}{\partial \hat{K}_{\mu\nu}(t)\partial G^{\ell}_{\alpha\beta}(t',s')} &= \delta_{\mu\alpha}\delta_{\nu\beta}\Phi^{\ell-1}_{\alpha\beta}(t',s')\delta(t-t')\delta(t-s')
\end{aligned} \tag{25}$$

This enumerates all possible non-vanishing terms in the Hessian. We can now construct a block matrix of these Hessians by partitioning our order parameters $\boldsymbol{q} = [\boldsymbol{q}_1, \boldsymbol{q}_2]^\top$ where

$$\boldsymbol{q}_1 = \text{Vec}\{\Phi^\ell_{\mu\nu}(t,s), G^\ell_{\mu\nu}(t,s), K_{\mu\nu}(t), \Delta_\mu(t), \hat{\Phi}^\ell_{\mu\nu}(t,s), \hat{G}^\ell_{\mu\nu}(t,s), \hat{K}_{\mu\nu}(t), \hat{\Delta}_\mu(t)\} \tag{26}$$

$$\boldsymbol{q}_2 = \text{Vec}\{A^\ell_{\mu\nu}(t,s), B^\ell_{\mu\nu}(t,s)\}. \tag{27}$$

This choice will become apparent shortly.

$$\nabla^2_{\boldsymbol{q}} S = \begin{bmatrix} \nabla^2_{\boldsymbol{q}_1} S & \nabla^2_{\boldsymbol{q}_1 \boldsymbol{q}_2} S \\ \nabla^2_{\boldsymbol{q}_2 \boldsymbol{q}_1} S & \nabla^2_{\boldsymbol{q}_2} S \end{bmatrix} \tag{28}$$

To calculate the full propagator $\boldsymbol{\Sigma} = -\left[\nabla^2_{\boldsymbol{q}} S\right]^{-1}$, we will assume invertibility of the upper block $\boldsymbol{\Sigma}^0 = -\left[\nabla^2_{\boldsymbol{q}_1} S\right]^{-1}$ and use this in the Schur complement

$$\begin{aligned}
\boldsymbol{\Sigma} &= -\left[\nabla^2_{\boldsymbol{q}} S\right]^{-1} = \begin{bmatrix} \boldsymbol{\Sigma}_{11} & \boldsymbol{\Sigma}_{12} \\ \boldsymbol{\Sigma}_{21} & \boldsymbol{\Sigma}_{22} \end{bmatrix} \\
\boldsymbol{\Sigma}_{11} &= \boldsymbol{\Sigma}^0 - \boldsymbol{\Sigma}^0 \left[\nabla^2_{\boldsymbol{q}_1 \boldsymbol{q}_2} S\right] \left(\nabla^2_{\boldsymbol{q}_2} S + (\nabla^2_{\boldsymbol{q}_2 \boldsymbol{q}_1} S) \boldsymbol{\Sigma}^0 (\nabla^2_{\boldsymbol{q}_1 \boldsymbol{q}_2} S)\right)^{-1} \left[\nabla^2_{\boldsymbol{q}_2 \boldsymbol{q}_1} S\right] \boldsymbol{\Sigma}^0 \\
\boldsymbol{\Sigma}_{12} &= \boldsymbol{\Sigma}_{21}^\top = -\boldsymbol{\Sigma}^0 \left[\nabla^2_{\boldsymbol{q}_1 \boldsymbol{q}_2} S\right] \left(\nabla^2_{\boldsymbol{q}_2} S + (\nabla^2_{\boldsymbol{q}_2 \boldsymbol{q}_1} S) \boldsymbol{\Sigma}^0 (\nabla^2_{\boldsymbol{q}_1 \boldsymbol{q}_2} S)\right)^{-1} \\
\boldsymbol{\Sigma}_{22} &= -\left(\nabla^2_{\boldsymbol{q}_2} S + (\nabla^2_{\boldsymbol{q}_2 \boldsymbol{q}_1} S) \boldsymbol{\Sigma}^0 (\nabla^2_{\boldsymbol{q}_1 \boldsymbol{q}_2} S)\right)^{-1}
\end{aligned} \tag{29}$$

We now need to solve for $\boldsymbol{\Sigma}^0 = -\left[\nabla^2_{\boldsymbol{q}_1} S\right]^{-1}$. To perform this inverse, we again partition $\boldsymbol{q}_1$ into two sets of order parameters $\boldsymbol{q}_1 = [\boldsymbol{q}_1^1, \boldsymbol{q}_1^2]$ where $\boldsymbol{q}_1^1 = \text{Vec}\{\Phi^\ell_{\mu\nu}(t,s), G^\ell_{\mu\nu}(t,s), K_{\mu\nu}(t), \Delta_\mu(t)\}$ and $\boldsymbol{q}_1^2 = \text{Vec}\{\hat{\Phi}^\ell_{\mu\nu}(t,s), \hat{G}^\ell_{\mu\nu}(t,s), \hat{K}_{\mu\nu}(t), \hat{\Delta}_\mu(t)\}$

$$\nabla^2_{\boldsymbol{q}_1} S = \begin{bmatrix} \boldsymbol{0} & \boldsymbol{U}^\top \\ \boldsymbol{U} & \boldsymbol{\kappa} \end{bmatrix}, \ \boldsymbol{\kappa} \equiv \nabla^2_{\boldsymbol{q}_1^2} S \, , \ \boldsymbol{U} \equiv \nabla^2_{\boldsymbol{q}_1^2 \boldsymbol{q}_1^1} S \tag{30}$$

We seek a physically sensible inverse where the variance of $\boldsymbol{q}_1^2$ is vanishing [51, 53]. This leads to the following sub-propagator $\boldsymbol{\Sigma}^0$

$$\boldsymbol{\Sigma}^0 = -[\nabla^2_{\boldsymbol{q}_1} S]^{-1} = \begin{bmatrix} \boldsymbol{U}^{-1} \boldsymbol{\kappa} [\boldsymbol{U}^{-1}]^\top & -\boldsymbol{U}^{-1} \\ -[\boldsymbol{U}^\top]^{-1} & \boldsymbol{0} \end{bmatrix} \tag{31}$$

Thus given $\boldsymbol{\kappa}, \boldsymbol{U}$, we can solve for $\boldsymbol{\Sigma}^0$ and ultimately for the full propagator $\boldsymbol{\Sigma}$. The relevant entries in $\boldsymbol{\kappa}$ and $\boldsymbol{U}$ are given by those second derivatives calculated above. We note that each of the field derivatives needed for $\boldsymbol{U}$ can be computed implicitly from the field dynamics. For example, for the $\Delta_\mu(t)$ derivatives we have

$$\begin{aligned}
\frac{\partial}{\partial \Delta_{\nu'}(t')} h^\ell_\mu(t) &= \gamma \Theta(t - t') \Phi^{\ell-1}_{\mu\nu'}(t, t') g^\ell_\nu(t') \\
&\quad + \gamma \int_0^t ds \sum_\nu \left[A^{\ell-1}_{\mu\nu}(t,s) + \Phi^{\ell-1}_{\mu\nu}(t,s) \Delta_\nu(s)\right] \frac{\partial g^\ell_\nu(s)}{\partial \Delta_{\nu'}(t')} \\
\frac{\partial}{\partial \Delta_{\nu}(t')} z^\ell_\mu(t) &= \gamma \Theta(t - t') G^{\ell+1}_{\mu\nu}(t, t') \phi(h^\ell_\nu(t')) \\
&\quad + \gamma \int_0^t ds \sum_\nu \left[B^\ell_{\mu\nu}(t,s) + G^{\ell+1}_{\mu\nu}(t,s) \Delta_\nu(s)\right] \frac{\partial \phi(h^\ell_\nu(s))}{\partial \Delta_{\nu'}(t')}
\end{aligned} \tag{32}$$

These can then be used in the averages such as $\left\langle \frac{\partial}{\partial \Delta_{\nu'}(t')} \phi(h^\ell_\mu(t)) \phi(h^\ell_\nu(s)) \right\rangle$. Similarly, we can compute terms such as $\frac{\partial h^\ell_\mu(t)}{\partial \Phi^\ell_{\alpha\beta(t',s')}}$ through the following closed equations

$$\begin{aligned}
\frac{\partial h^\ell_\mu(t)}{\partial \Phi^{\ell-1}_{\alpha\beta}(t', s')} &= \gamma \delta(t - t') \delta_{\mu\alpha} \Theta(t - s') \Delta_\beta(s') \\
&\quad + \gamma \int_0^t ds \sum_\nu \left[A^{\ell-1}_{\mu\nu}(t,s) + \Delta_\nu(s) \Phi^{\ell-1}_{\mu\nu}(t,s)\right] \frac{\partial g^\ell_\nu(s)}{\partial \Phi^\ell_{\alpha\beta}(t', s')} \\
\frac{\partial z^\ell_\mu(t)}{\partial \Phi^{\ell-1}_{\alpha\beta}(t', s')} &= \gamma \int_0^t ds \sum_\nu \left[B^\ell_{\mu\nu}(t,s) + \Delta_\nu(s) G^{\ell+1}_{\mu\nu}(t,s)\right] \frac{\partial \phi(h^\ell_\nu(s))}{\partial \Phi^{\ell-1}_{\alpha\beta}(t', s')}
\end{aligned} \tag{33}$$

These terms can then be used to compute quantities like D^{Φ^ℓ}.

E Solving for the Propagator

Below we provide a pseudocode algorithm to solve for the propagator elements.

Algorithm 1: Propagator Solver

Data: $\boldsymbol{K}^x, \boldsymbol{y}$, Initial Guesses $\{\boldsymbol{\Phi}^\ell, \boldsymbol{G}^\ell\}_{\ell=1}^{L}$, $\{\boldsymbol{A}^\ell, \boldsymbol{B}^\ell\}_{\ell=1}^{L-1}$, Sample count $\mathcal{S}$, Update Speed β
Result: Propagator Matrix $\boldsymbol{\Sigma}$

Solve DMFT equations with Algorithm 2 for order parameters $f_\mu(t), \Phi^\ell_{\mu\alpha}(t,s), \ldots$;
Draw $\mathcal{S}$ samples $\{u^\ell_{\mu,n}(t)\}_{n=1}^{\mathcal{S}} \sim \mathcal{GP}(0, \boldsymbol{\Phi}^{\ell-1})$, $\{r^\ell_{\mu,n}(t)\}_{n=1}^{\mathcal{S}} \sim \mathcal{GP}(0, \boldsymbol{G}^{\ell+1})$;
Integrate dynamics for each sample to get $\{h^\ell_{\mu,n}(t), z^\ell_{\mu,n}(t)\}_{n=1}^{\mathcal{S}}$;
Estimate κ functions with Monte-Carlo integration, for instance
$\kappa^{\Phi^\ell}_{\mu\nu\alpha\beta}(t,s,t',s') = \frac{1}{\mathcal{S}} \sum_{n\in[\mathcal{S}]} \phi(h^\ell_{\mu,n}(t))\phi(h^\ell_{\nu,n}(s))\phi(h^\ell_{\alpha,n}(t'))\phi(h^\ell_{\beta,n}(s')) - \Phi^\ell_{\mu\nu}(t,s)\Phi^\ell_{\alpha\beta}(t',s')$;
For each sample, compute field sensitivities to error signals, such as $\frac{\partial h^\ell_{\mu,n}(t)}{\partial \Delta_\nu(s)}$, and kernels $\frac{\partial h^\ell_{\mu,n}(t)}{\partial \Phi^\ell_{\alpha\beta}(t',s')}$ implicitly using equations (32) (33) ;
Use these sensitivities to compute the necessary D tensors such as $D^{\Phi^\ell \Delta}_{\mu\nu\alpha} = \frac{1}{\mathcal{S}} \sum_{n\in[\mathcal{S}]} \frac{\partial}{\partial \Delta_\alpha(t')} \left[\phi(h^\ell_{\mu,n}(t))\phi(h^\ell_{\nu,n}(s))\right]$;
Invert $\boldsymbol{U}$ matrix and compute $\boldsymbol{\Sigma}_0$ in equation (31);
Compute the Schur-complement in equation (29) to handle the response functions ;

The above propagator solver builds on the solution to the DMFT equations which is provided below.

Algorithm 2: Alternating Monte-Carlo Solution to Saddle Point Equations

Data: $\boldsymbol{K}^x, \boldsymbol{y}$, Initial Guesses $\{\boldsymbol{\Phi}^\ell, \boldsymbol{G}^\ell\}_{\ell=1}^{L}$, $\{\boldsymbol{A}^\ell, \boldsymbol{B}^\ell\}_{\ell=1}^{L-1}$, Sample count $\mathcal{S}$, Update Speed β
Result: Final Kernels $\{\boldsymbol{\Phi}^\ell, \boldsymbol{G}^\ell\}_{\ell=1}^{L}$, $\{\boldsymbol{A}^\ell, \boldsymbol{B}^\ell\}_{\ell=1}^{L-1}$, Network predictions through training $f_\mu(t)$

$\boldsymbol{\Phi}^0 = \boldsymbol{K}^x \otimes \boldsymbol{1}\boldsymbol{1}^\top$, $\boldsymbol{G}^{L+1} = \boldsymbol{1}\boldsymbol{1}^\top$;
while *Kernels Not Converged* **do**
  From $\{\boldsymbol{\Phi}^\ell, \boldsymbol{G}^\ell\}$ compute $\boldsymbol{K}^{NTK}(t,t)$ and solve $\frac{d}{dt} f_\mu(t) = \sum_\alpha \Delta_\alpha(t) K^{NTK}_{\mu\alpha}(t,t)$;
  $\ell = 1$;
  while $\ell < L+1$ **do**
    Draw $\mathcal{S}$ samples $\{u^\ell_{\mu,n}(t)\}_{n=1}^{\mathcal{S}} \sim \mathcal{GP}(0, \boldsymbol{\Phi}^{\ell-1})$, $\{r^\ell_{\mu,n}(t)\}_{n=1}^{\mathcal{S}} \sim \mathcal{GP}(0, \boldsymbol{G}^{\ell+1})$;
    Integrate dynamics for each sample to get $\{h^\ell_{\mu,n}(t), z^\ell_{\mu,n}(t)\}_{n=1}^{\mathcal{S}}$;
    Compute new $\boldsymbol{\Phi}^\ell, \boldsymbol{G}^\ell$ estimates:
    $\tilde{\Phi}^\ell_{\mu\alpha}(t,s) = \frac{1}{\mathcal{S}} \sum_{n\in[\mathcal{S}]} \phi(h^\ell_{\mu,n}(t))\phi(h^\ell_{\alpha,n}(s))$, $\tilde{G}^\ell_{\mu\alpha}(t,s) = \frac{1}{\mathcal{S}} \sum_{n\in[\mathcal{S}]} g^\ell_{\mu,n}(t) g^\ell_{\alpha,n}(s)$;
    Solve for Jacobians on each sample $\frac{\partial \phi(\boldsymbol{h}^\ell_n)}{\partial \boldsymbol{r}^{\ell\top}_n}$, $\frac{\partial \boldsymbol{g}^\ell_n}{\partial \boldsymbol{u}^{\ell\top}_n}$;
    Compute new $\boldsymbol{A}^\ell, \boldsymbol{B}^{\ell-1}$ estimates:
    $\tilde{\boldsymbol{A}}^\ell = \frac{1}{\mathcal{S}} \sum_{n\in[\mathcal{S}]} \frac{\partial \phi(\boldsymbol{h}^\ell_n)}{\partial \boldsymbol{r}^{\ell\top}_n}$, $\tilde{\boldsymbol{B}}^{\ell-1} = \frac{1}{\mathcal{S}} \sum_{n\in[\mathcal{S}]} \frac{\partial \boldsymbol{g}^\ell_n}{\partial \boldsymbol{u}^{\ell\top}_n}$;
    $\ell \leftarrow \ell + 1$;
  end
  $\ell = 1$;
  while $\ell < L+1$ **do**
    Update feature kernels: $\boldsymbol{\Phi}^\ell \leftarrow (1-\beta)\boldsymbol{\Phi}^\ell + \beta\tilde{\boldsymbol{\Phi}}^\ell$, $\boldsymbol{G}^\ell \leftarrow (1-\beta)\boldsymbol{G}^\ell + \beta\tilde{\boldsymbol{G}}^\ell$;
    if $\ell < L$ **then**
      Update $\boldsymbol{A}^\ell \leftarrow (1-\beta)\boldsymbol{A}^\ell + \beta\tilde{\boldsymbol{A}}^\ell, \boldsymbol{B}^\ell \leftarrow (1-\beta)\boldsymbol{B}^\ell + \beta\tilde{\boldsymbol{B}}^\ell$
    end
    $\ell \leftarrow \ell + 1$
  end
end
return $\{\boldsymbol{\Phi}^\ell, \boldsymbol{G}^\ell\}_{\ell=1}^{L}, \{\boldsymbol{A}^\ell, \boldsymbol{B}^\ell\}_{\ell=1}^{L-1}, \{f_\mu(t)\}_{\mu=1}^{P}$

F Leading Correction to the Mean Order Parameters

In this section we use the propagator structure derived in the last section to reason about the leading finite size correction to $\langle \boldsymbol{q} \rangle$ at width N. Letting the indices i, j, k, n enumerate all entries of the order parameters in $\boldsymbol{q}$ (technically this is a sum over samples and an integral over time for gradient flow), we find the leading Pade Approximant for the mean has the form (App C)

$$\begin{aligned}
\langle q_i - q_i^\infty \rangle_N &= \frac{N \langle (q_i - q_i^\infty) V \rangle_\infty + \frac{N^2}{2} \langle (q_i - q_i^\infty) V^2 \rangle_\infty \ldots}{1 + N \langle V \rangle_\infty + \frac{N^2}{2} \langle V^2 \rangle_\infty + \ldots} \\
&\sim \frac{1}{3! N} \sum_{jkl} \frac{\partial^3 S}{\partial q_j \partial q_k \partial q_l} \langle \delta_i \delta_j \delta_k \delta_l \rangle_\infty + \mathcal{O}(N^{-2}). \qquad (34) \\
&= \frac{1}{2N} \sum_{jkl} \frac{\partial^3 S}{\partial q_j \partial q_k \partial q_l} \Sigma_{ij} \Sigma_{kl} + \mathcal{O}(N^{-2}) \qquad (35)
\end{aligned}$$

where $\delta_j = \sqrt{N}(q_j - q_j^\infty)$ and the derivatives are computed at the saddle point. In the last line, we utilized Wick's theorem and the permutation symmetry of the third derivative $\frac{\partial^3 S}{\partial q_i \partial q_j \partial q_k}$ to evaluate the four point averages in terms of the propagator Σ_{ij}, which was provided in the preceding section D. In practice computing even the full set of second derivatives for the DMFT action to get Σ is quite challenging. Despite the challenge of computing the mean order parameter correction, these corrections are relevant in practice and crucially distinguish the training timescales of deep networks at different widths as we show in Figures 6 and A.6.

F.1 Correction to Mean Predictions and Full MSE Correction

Supposing that we solved for the propagator $\boldsymbol{\Sigma}$, using the formalism in the preceeding section, we can compute the $\mathcal{O}(N^{-1})$ correction to the average network prediction error due to finite size. We let $\langle \boldsymbol{\Delta}(t) \rangle$ represent the average of errors over an ensemble of width N networks.

$$\begin{aligned}
\frac{d}{dt} \langle \Delta_\mu(t) \rangle &= -\sum_\nu \langle K_{\mu\nu}(t) \Delta_\nu(t) \rangle \\
&= -\sum_\nu \langle K_{\mu\nu}(t) \rangle \langle \Delta_\nu(t) \rangle - \sum_\nu \text{Cov}\left(K_{\mu\nu}(t), \Delta_\nu(t) \right) \\
&\sim -\sum_\nu \langle K_{\mu\nu}(t) \rangle \langle \Delta_\nu(t) \rangle - \frac{1}{N} \sum_\nu \Sigma^{K\Delta}_{\mu\nu\nu}(t,t) + \mathcal{O}(N^{-2}) \qquad (36)
\end{aligned}$$

where $\Sigma^{K\Delta}_{\mu\nu\nu}(t,t)$ is the leading covariance (propagator element) between the kernel $K_{\mu\nu}(t)$ and prediction error $\Delta_\nu(t)$. We see that the average kernel $\langle K_{\mu\nu}(t) \rangle$ (which depends on the finite width N) plays an important role in characterizing the timescales of the average prediction dynamics. Once this equation is solved for $\langle \Delta_\mu(t) \rangle$, the square loss at width N and time t has the form

$$\sum_\mu \langle \Delta_\mu(t)^2 \rangle \sim \left(1 - \frac{2}{N}\right) \sum_\mu \Delta_\mu^\infty(t)^2 + \frac{2}{N} \sum_\mu \langle \Delta_\mu(t) \rangle_\infty \Delta_\mu^\infty(t) + \frac{1}{N} \sum_\mu \Sigma^\Delta_{\mu\mu}(t,t) + \mathcal{O}(N^{-2}) \qquad (37)$$

We will now comment on the structure of the cross term in this above solution. First, if $\langle \boldsymbol{K} \rangle \succeq \boldsymbol{K}^\infty$ and $\Sigma^{K\Delta}$ is negligible then the average errors at finite width will decay more rapidly than the infinite width model. However, we suspect that in general, $\langle \boldsymbol{K} \rangle - \boldsymbol{K}^\infty$ contains many negative eigenvalues since signal propagation at finite width tends to reduce the scale of feature kernels [14]. We suspect that this is the cause of the slower dynamics of ensembled predictors for narrower networks in Figure 6 and Figure A.6. Additionally, the term involving $\Sigma^{K\Delta}$ will generically increase the cross term since the dynamics of Δ cause its fluctuations to become anti-correlated with the fluctuations in K. In general, it is challenging to make strong definitive statements about the relative scale of these competing effects on the cross term. However, we can say more about this solution in the lazy limit, where we find that the cross term will generically be positive, leading to larger MSE (Appendix G.2).

F.2 Perturbation Theory in Rates rather than Predictions

In experiments on deep CNNs trained on CIFAR-10 in 6 and A.6, we find that the loss curves for the ensemble averaged predictors are effectively time rescaled by a function of network width. In this section, we argue that a proper way to account for this is to compute a perturbation expansion in the *exponent* which defines the rate of decay of the training errors. To illustrate the point, we first consider the case of a single training example before describing larger datasets. In this case, we consider the change of variables $\Delta(t) = e^{-r(t)}y$. We now treat r as an order parameter of the theory with dynamics

$$\frac{d}{dt}r(t) = K(t) \tag{38}$$

Note that this equation is now a linear relation between two order parameters $(r(t), K(t))$, whereas the relation was previously quadratic. In the lazy limit, if $K \to K - \epsilon$ then $r \to r - \epsilon t$, giving an effective rescaling of training time by $1 - \frac{\epsilon}{K}$.

For multiple training examples, we introduce the notion of a transition matrix $\boldsymbol{T}(t) \in \mathbb{R}^{P\times P}$ which has dynamics

$$\frac{d}{dt}\boldsymbol{T}(t) = -\boldsymbol{K}(t)\boldsymbol{T}(t)\ ,\ \boldsymbol{T}(0) = \mathbf{I}. \tag{39}$$

The solution to the training prediction errors can be obtained at any time t by multiplying the initial condition $\boldsymbol{\Delta}(0) = \boldsymbol{y}$ with the transition matrix $\boldsymbol{\Delta}(t) = \boldsymbol{T}(t)\boldsymbol{y}$, where $\boldsymbol{y}$ are the training targets. In this case, the relevant *rate matrix*, which would be an alternative order parameter is

$$\boldsymbol{R}(t) = -\log \boldsymbol{T}(t) \tag{40}$$

where log is the matrix logarithm function. Note that in general $\boldsymbol{T}(t)$ admits a Peano-Baker series solution [59–61]. In the special case where $\boldsymbol{K}(t)$ commutes with $\bar{\boldsymbol{K}}(t) = \frac{1}{t}\int_0^t ds\boldsymbol{K}(s)$, we obtain the following simplified formula for the rate matrix $\boldsymbol{R}$

$$\boldsymbol{R}(t) = \int_0^t ds\ \boldsymbol{K}(s) \tag{41}$$

The benefit of this representation is the elimination of coupled order parameter dynamics which are quadratic in fluctuations (in $\boldsymbol{\Delta}$ and $\boldsymbol{K}$) into a linear dynamical relation between order parameters $\boldsymbol{R}$ and $\boldsymbol{K}$. An expansion in $\boldsymbol{R}$ will thus give better predictions at long times t than a direct expansion in $\boldsymbol{\Delta}$. In the lazy $\gamma \to 0$ limit, the constancy of $\boldsymbol{K}(t) = \boldsymbol{K}$ gives the further simplification $\boldsymbol{R} = \boldsymbol{K}t$. Working with this representation, we have the following finite width expression for the training loss

$$\begin{aligned}\left\langle |\boldsymbol{\Delta}(t)|^2 \right\rangle &= \boldsymbol{y}^\top \left\langle \exp\left(-2\boldsymbol{R}(t)\right)\right\rangle \boldsymbol{y} \\ &\sim \boldsymbol{y}^\top \exp\left(-2\left(\boldsymbol{R}_\infty(t) + \frac{1}{N}\boldsymbol{R}^1(t)\right)\right)\boldsymbol{y} \\ &+ \frac{1}{2}\sum_{\mu\nu\alpha\beta}\Sigma^R_{\mu\nu\alpha\beta}(t,t)\frac{\partial^2}{\partial R_{\mu\nu}\partial R_{\alpha\beta}}\boldsymbol{y}^\top \exp\left(-2\boldsymbol{R}\right)\boldsymbol{y}|_{\boldsymbol{R}=\boldsymbol{R}_\infty(t)+\frac{1}{N}\boldsymbol{R}^1(t)} + \mathcal{O}(N^{-2})\end{aligned} \tag{42}$$

where $\langle \boldsymbol{R}\rangle \sim \boldsymbol{R}_\infty + \frac{1}{N}\boldsymbol{R}^1 + \mathcal{O}(N^{-2})$ is the leading correction to the mean $\boldsymbol{R}$. In this representation, it is clear that finite width can alter the timescale of the dynamics through a correction to the mean of $\boldsymbol{R}$, as well as contribute an additive correction from fluctuations. This justifies the study perturbation analysis of rates R_N as a function of $1/N$ in Figures 6 and A.6.

G Variance in the Lazy Limit

We can simplify the propagator equations in the lazy $\gamma \to 0$ limit. To demonstrate how to use our formalism, we go through the complete process of inverting the Hessian, however, for this case, this procedure is a bit cumbersome. A simplified derivation for the lazy limit can be found below in section G.1 which relies only on linearizing the dynamics around the infinite width solution. In the

$\gamma \to 0$ limit, all of the D tensors vanish and the κ tensors are constant in time. Thus, it suffices to analyze the kernels restricted to $t = 0$ and study the evolution of the prediction variance $\mathbf{\Delta}(t)$.

$$
\begin{aligned}
S &= \int dt \sum_{\mu} \hat{\Delta}_\mu(t) \left(\Delta_\mu(t) - y_\mu + \int ds \sum_{\nu} \Theta(t-s) K_{\mu\nu} \Delta_\nu(s) \right) \\
&+ \sum_{\ell} \sum_{\mu\nu} \left[\hat{\Phi}^\ell_{\mu\nu} \Phi^\ell_{\mu\nu} + G^\ell_{\mu\nu} \hat{G}^\ell_{\mu\nu} \right] + \sum_{\mu\nu} \hat{K}_{\mu\nu} \left[K_{\mu\nu} - \sum_\ell G^{\ell+1}_{\mu\nu} \Phi^\ell_{\mu\nu} \right] + \sum_\ell \ln \mathcal{Z}_\ell \\
\mathcal{Z}_\ell &= \mathbb{E}_{\{u^\ell_\mu\}, \{r^\ell_\mu\}} \exp \left(- \sum_{\mu\nu} \hat{\Phi}^\ell_{\mu\nu} \phi(u^\ell_\mu) \phi(u^\ell_\nu) - \sum_{\mu\nu} \hat{G}^\ell_{\mu\nu} g^\ell_\mu g^\ell_\nu \right) \ , \ g^\ell_\mu = r^\ell_\mu \dot{\phi}(u^\ell_\mu)
\end{aligned} \tag{43}
$$

where $\{u^\ell_\mu\} \sim \mathcal{N}(0, \mathbf{\Phi}^{\ell-1}), \{r^\ell_\mu\} \sim \mathcal{N}(0, \boldsymbol{G}^{\ell+1})$. Taking two derivatives with respect to $\{\hat{\Phi}^\ell, \hat{G}^\ell\}$ give terms of the form

$$
\begin{aligned}
\kappa^{\Phi^\ell}_{\mu\nu\alpha\beta} &= \left\langle \phi(u^\ell_\mu) \phi(u^\ell_\nu) \phi(u^\ell_\alpha) \phi(u^\ell_\beta) \right\rangle - \Phi^\ell_{\mu\nu} \Phi^\ell_{\alpha\beta} \\
\kappa^{G^\ell}_{\mu\nu\alpha\beta} &= \left\langle g^\ell_\mu g^\ell_\nu g^\ell_\alpha g^\ell_\beta \right\rangle - G^\ell_{\mu\nu} G^\ell_{\alpha\beta} \\
\kappa^{\Phi^\ell, G^\ell}_{\mu\nu\alpha\beta} &= \left\langle \phi(u^\ell_\mu) \phi(u^\ell_\nu) g^\ell_\alpha g^\ell_\beta \right\rangle - \Phi^\ell_{\mu\nu} G^\ell_{\alpha\beta}
\end{aligned} \tag{44}
$$

Given these we also have the relevant non-vanishing sensitivity tensors

$$
\begin{aligned}
D^{\Phi^{\ell+1}\Phi^\ell}_{\mu\nu\alpha\beta} &= \frac{\partial^2}{\partial \Phi^\ell_{\alpha\beta}} \left\langle \phi(u^{\ell+1}_\mu) \phi(u^{\ell+1}_\nu) \right\rangle \ , \ D^{G^\ell G^{\ell+1}}_{\mu\nu\alpha\beta} = \frac{\partial}{\partial G^{\ell+1}_{\alpha\beta}} \left\langle g^\ell_\mu g^\ell_\nu \right\rangle \\
D^{G^\ell \Phi^{\ell-1}}_{\mu\nu\alpha\beta} &= \frac{\partial}{\partial \Phi^{\ell-1}_{\alpha\beta}} \left\langle g^\ell_\mu g^\ell_\nu \right\rangle
\end{aligned} \tag{45}
$$

$$
\begin{aligned}
D^{K\Phi^\ell}_{\mu\nu\alpha\beta} &= \delta_{\mu\alpha} \delta_{\nu\beta} G^{\ell+1}_{\mu\nu} \ , \ D^{KG^\ell}_{\mu\nu\alpha\beta} = \delta_{\mu\alpha} \delta_{\nu\beta} \Phi^{\ell-1}_{\mu\nu} \\
D^{\Delta K}_{\mu\alpha\beta}(t) &= \int ds \Theta(t-s) \delta_{\mu\alpha} \Delta_\beta(s)
\end{aligned} \tag{46}
$$

As before we let $\boldsymbol{q}_1 = \text{Vec}\{\Delta_\mu(t), \Phi^\ell_{\mu\nu}, G^\ell_{\mu\nu}, K_{\mu\nu}\}$ and $\boldsymbol{q}_2 = \text{Vec}\{\hat{\Delta}_\mu(t), \hat{\Phi}^\ell_{\mu\nu}, \hat{G}^\ell_{\mu\nu}, \hat{K}_{\mu\nu}\}$. The propagator has the form

$$
\boldsymbol{U} \equiv \nabla^2_{\boldsymbol{q}_2 \boldsymbol{q}_1} S = \begin{bmatrix} \mathbf{I} + \mathbf{\Theta}_K & \mathbf{0} & \mathbf{0} & \boldsymbol{D}^{\Delta K} \\ \mathbf{0} & \mathbf{I} - \boldsymbol{D}^{\Phi,\Phi} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -\boldsymbol{D}^{G\Phi} & \mathbf{I} - \boldsymbol{D}^{GG} & \mathbf{0} \\ \mathbf{0} & -\boldsymbol{D}^{K\Phi} & -\boldsymbol{D}^{KG} & \mathbf{I} \end{bmatrix} \ , \ \nabla^2_{\boldsymbol{q}_2 \boldsymbol{q}_2} S = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\kappa}^{\Phi,\Phi} & \boldsymbol{\kappa}^{\Phi G} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\kappa}^{G\Phi} & \boldsymbol{\kappa}^{GG} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix} \tag{47}
$$

The propagator of interest is $\mathbf{\Sigma}_{\boldsymbol{q}_1} = \boldsymbol{U}^{-1} \left[\nabla^2_{\boldsymbol{q}_2 \boldsymbol{q}_2} S \right] \boldsymbol{U}^{-1\top}$. We can exploit the block structure of $\boldsymbol{U}$ to find an inverse

$$
\boldsymbol{U}^{-1} = \begin{bmatrix} \boldsymbol{U}^{-1}_{\Delta\Delta} & \boldsymbol{U}^{-1}_{\Delta\Phi} & \boldsymbol{U}^{-1}_{\Delta G} & \boldsymbol{U}^{-1}_{\Delta K} \\ \mathbf{0} & \boldsymbol{U}^{-1}_{\Phi\Phi} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{U}^{-1}_{G\Phi} & \boldsymbol{U}^{-1}_{GG} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{U}^{-1}_{K\Phi} & \boldsymbol{U}^{-1}_{KG} & \mathbf{I} \end{bmatrix} \tag{48}
$$

where each sub-block can be computed with the Schur-complement formula. Altogether, we multiply through to get the propagator

$$
\begin{aligned}
\mathbf{\Sigma} = &\begin{bmatrix} \mathbf{0} & \boldsymbol{U}^{-1}_{\Delta\Phi} \boldsymbol{\kappa}^{\Phi\Phi} + \boldsymbol{U}^{-1}_{\Delta G} \boldsymbol{\kappa}^{G\Phi} & \boldsymbol{U}^{-1}_{\Delta\Phi} \boldsymbol{\kappa}^{\Phi G} + \boldsymbol{U}^{-1}_{\Delta G} \boldsymbol{\kappa}^{GG} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{U}^{-1}_{\Phi\Phi} \boldsymbol{\kappa}^{\Phi\Phi} & \boldsymbol{U}^{-1}_{\Phi\Phi} \boldsymbol{\kappa}^{\Phi G} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{U}^{-1}_{G\Phi} \boldsymbol{\kappa}^{\Phi\Phi} + \boldsymbol{U}^{-1}_{GG} \boldsymbol{\kappa}^{G\Phi} & \boldsymbol{U}^{-1}_{G\Phi} \boldsymbol{\kappa}^{\Phi G} + \boldsymbol{U}^{-1}_{GG} \boldsymbol{\kappa}^{GG} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{U}^{-1}_{K\Phi} \boldsymbol{\kappa}^{\Phi\Phi} + \boldsymbol{U}^{-1}_{KG} \boldsymbol{\kappa}^{G\Phi} & \boldsymbol{U}^{-1}_{K\Phi} \boldsymbol{\kappa}^{\Phi G} + \boldsymbol{U}^{-1}_{KG} \boldsymbol{\kappa}^{GG} & \mathbf{0} \end{bmatrix} \\
&\times \begin{bmatrix} \boldsymbol{U}^{-1}_{\Delta\Delta} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ [\boldsymbol{U}^{-1}_{\Delta\Phi}]^\top & \boldsymbol{U}^{-1}_{\Phi\Phi} & [\boldsymbol{U}^{-1}_{G\Phi}]^\top & [\boldsymbol{U}^{-1}_{K\Phi}]^\top \\ [\boldsymbol{U}^{-1}_{\Delta G}]^\top & \mathbf{0} & \boldsymbol{U}^{-1}_{GG} & [\boldsymbol{U}^{-1}_{KG}]^\top \\ [\boldsymbol{U}^{-1}_{\Delta K}]^\top & \mathbf{0} & \mathbf{0} & \mathbf{I} \end{bmatrix}
\end{aligned} \tag{49}
$$

Two of these blocks corresponding to K, Δ are especially important for characterizing the fluctuations of network predictions. The covariance structure for K has the form

$$\boldsymbol{\Sigma}_K = \boldsymbol{U}_{K\Phi}^{-1}\boldsymbol{\kappa}^{\Phi\Phi}[\boldsymbol{U}_{K\Phi}^{-1}]^\top + \boldsymbol{U}_{KG}^{-1}\boldsymbol{\kappa}^{G\Phi}[\boldsymbol{U}_{K\Phi}^{-1}]^\top + \boldsymbol{U}_{K\Phi}^{-1}\boldsymbol{\kappa}^{\Phi G}[\boldsymbol{U}_{KG}^{-1}]^\top + \boldsymbol{U}_{KG}^{-1}\boldsymbol{\kappa}^{GG}[\boldsymbol{U}_{KG}^{-1}]^\top \quad (50)$$

Next we use the fact that $\boldsymbol{U}_{\Delta\Phi}^{-1} = \boldsymbol{U}_{\Delta K}^{-1}\boldsymbol{U}_{K\Phi}^{-1}$ and that $\boldsymbol{U}_{\Delta G}^{-1} = \boldsymbol{U}_{\Delta K}^{-1}\boldsymbol{U}_{KG}^{-1}$, which follows from the block structure of $\boldsymbol{U}$. Consequently we arrive at the identity

$$\begin{aligned}\boldsymbol{\Sigma}_\Delta &= \boldsymbol{U}_{\Delta\Phi}^{-1}\boldsymbol{\kappa}^{\Phi\Phi}[\boldsymbol{U}_{\Delta\Phi}^{-1}]^{-1} + \boldsymbol{U}_{\Delta G}^{-1}\boldsymbol{\kappa}^{G\Phi}[\boldsymbol{U}_{\Delta\Phi}^{-1}]^{-1} + \boldsymbol{U}_{\Delta G}^{-1}\boldsymbol{\kappa}^{G\Phi}[\boldsymbol{U}_{\Delta\Phi}^{-1}]^{-1} + \boldsymbol{U}_{\Delta G}^{-1}\boldsymbol{\kappa}^{GG}[\boldsymbol{U}_{\Delta G}^{-1}]^{-1} \\ &= \boldsymbol{U}_{\Delta K}^{-1}\boldsymbol{\Sigma}_K[\boldsymbol{U}_{K\Delta}^{-1}]^\top. \end{aligned} \quad (51)$$

Lastly, we note that, by the Schur-complement formula that $\boldsymbol{U}_{\Delta K}^{-1} = -\left(\mathbf{I} + \boldsymbol{\Theta}_K\right)^{-1}\boldsymbol{D}^{\Delta K}$. Thus, writing $\left(\mathbf{I} + \boldsymbol{\Theta}_K\right)\boldsymbol{\Sigma}_\Delta\left(\mathbf{I} + \boldsymbol{\Theta}_K\right)^\top = \boldsymbol{D}^{\Delta K}\boldsymbol{\Sigma}_K[\boldsymbol{D}^{\Delta K}]^\top$ as an integral equation, we find

$$\begin{aligned}&\Sigma^\Delta_{\mu\nu}(t,s) + \int_0^t dt' \sum_\alpha K_{\mu\alpha}\Sigma^\Delta_{\alpha\nu}(t',s) + \int_0^s ds' \sum_\beta K_{\nu\beta}\Sigma^\Delta_{\mu\beta}(t,s') \\ &+ \int_0^t dt' \int_0^s ds' \sum_{\alpha\beta} K_{\mu\alpha}K_{\nu\beta}\Sigma^\Delta_{\alpha\beta}(t',s') = \sum_{\alpha\beta}\int_0^t \Delta_\alpha(t')\int_0^s ds'\Delta_\beta(s')\Sigma^K_{\mu\alpha,\nu\beta}\end{aligned} \quad (52)$$

Differentiation with respect to t and s gives a simple differential equation

$$\begin{aligned}&\frac{\partial^2}{\partial t\partial s}\Sigma^\Delta_{\mu\nu}(t,s) + \sum_\alpha K_{\mu\alpha}\frac{\partial}{\partial s}\Sigma^\Delta_{\alpha\nu}(t,s) + \sum_\beta K_{\nu\beta}\frac{\partial}{\partial t}\Sigma^\Delta_{\mu\beta}(t,s) \\ &+ \sum_{\alpha\beta} K_{\mu\alpha}K_{\nu\beta}\Sigma^\Delta_{\alpha\beta}(t,s) = \sum_{\alpha\beta}\Delta_\alpha(t)\Delta_\beta(s)\Sigma^K_{\mu\alpha,\nu\beta}\end{aligned} \quad (53)$$

Let $\{\boldsymbol{\psi}_k\}$ be the eigenvectors of the kernel matrix $\boldsymbol{K}$. Projecting these dynamics on the eigenspace $\Sigma_{k\ell}(t,s) = \boldsymbol{\psi}_k^\top\boldsymbol{\Sigma}(t,s)\boldsymbol{\psi}_\ell$ recovers the equation in the main text

$$\left(\frac{\partial}{\partial t} + \lambda_k\right)\left(\frac{\partial}{\partial s} + \lambda_\ell\right)\Sigma_{k\ell}(t,s) = \sum_{k'\ell'}\Delta_{k'}(t)\Delta_{\ell'}(s)\Sigma^K_{kk'\ell\ell'} \quad (54)$$

Replacing $\Sigma^K = \kappa$ recovers the equation (7) in the main text.

G.1 Perturbed Linear System

In this section, we provide a simpler derivation of the lazy limit training error variance dynamics. In this case, we merely perturb the dynamics around its infinite width value $\boldsymbol{\Delta}(t) = \boldsymbol{\Delta}_\infty(t) + \boldsymbol{\epsilon}^\Delta(t)$ and $\boldsymbol{K} = \boldsymbol{K}_\infty + \boldsymbol{\epsilon}^K$, and keep terms only linear in these perturbations. The perturbation $\boldsymbol{\epsilon}^K$ is fixed in time and the dynamics of $\boldsymbol{\epsilon}^\Delta(t)$ are

$$\frac{d}{dt}\boldsymbol{\epsilon}^\Delta(t) = -\boldsymbol{K}_\infty\boldsymbol{\epsilon}^\Delta(t) - \boldsymbol{\epsilon}^K\boldsymbol{\Delta}_\infty(t) \quad (55)$$

Projecting this equation on the eigenspace of $\boldsymbol{K}_\infty$ gives

$$\frac{d}{dt}\epsilon^\Delta_k(t) = -\lambda_k\epsilon_k(t) - \sum_{k'}\epsilon^K_{kk'}\Delta^\infty_{k'}(t) \quad (56)$$

This immediately recovers the final result of the last section

$$\begin{aligned}&N\left(\frac{\partial}{\partial t} + \lambda_k\right)\left(\frac{\partial}{\partial t} + \lambda_k\right)\left\langle\epsilon^\Delta_k(t)\epsilon^\Delta_\ell(s)\right\rangle = \left(\frac{\partial}{\partial t} + \lambda_k\right)\left(\frac{\partial}{\partial t} + \lambda_k\right)\Sigma^\Delta_{k\ell}(t,s) \\ &= \sum_{k'\ell'}\Sigma^K_{kk'\ell\ell'}\Delta^\infty_{k'}(t)\Delta^\infty_{\ell'}(s)\end{aligned} \quad (57)$$

Qualitatively, the process of computing this linear correction (in ϵ^K) to the dynamics of $\boldsymbol{\Delta}$ is identical to the argument utilized in prior work on perturbative feature learning corrections [11]. In that context, the perturbation is caused by small amounts of feature learning, rather than initialization fluctuations.

G.2 Mean Prediction Error Correction in the Lazy Limit

Using a similar heuristic as in the preceeding section, we now consider the correction to the mean predictor $\langle \Delta_\mu(t) \rangle$ in the lazy limit. Taylor expanding $\langle \boldsymbol{\Delta}(t) \rangle$ in powers of $1/N$, we find

$$\begin{aligned}
\frac{d}{dt}\langle \boldsymbol{\Delta}(t) \rangle &= \frac{d}{dt}\boldsymbol{\Delta}^\infty(t) + \frac{1}{N}\frac{d}{dt}\boldsymbol{\Delta}^1(t) + ... \\
&= -\langle (\boldsymbol{K} - \boldsymbol{K}^\infty + \boldsymbol{K}^\infty)(\boldsymbol{\Delta} - \boldsymbol{\Delta}^\infty + \boldsymbol{\Delta}^\infty) \rangle \\
&= -\boldsymbol{K}^\infty \boldsymbol{\Delta}^\infty - \boldsymbol{K}^\infty \langle \boldsymbol{\Delta} - \boldsymbol{\Delta}^\infty \rangle \\
&- \langle (\boldsymbol{K} - \boldsymbol{K}^\infty) \rangle \boldsymbol{\Delta}^\infty - \langle (\boldsymbol{K} - \boldsymbol{K}^\infty)(\boldsymbol{\Delta} - \boldsymbol{\Delta}^\infty) \rangle \\
&\sim -\boldsymbol{K}^\infty \boldsymbol{\Delta}^\infty - \frac{1}{N}\boldsymbol{K}^\infty \boldsymbol{\Delta}^1 - \frac{1}{N}\boldsymbol{K}^1 \boldsymbol{\Delta}^\infty - \frac{1}{N}\left\langle \boldsymbol{\epsilon}^K \boldsymbol{\epsilon}^\Delta \right\rangle_\infty + \mathcal{O}(N^{-2})
\end{aligned} \tag{58}$$

From the previous section we have that

$$\frac{d}{dt}\boldsymbol{\epsilon}^\Delta = -\boldsymbol{K}^\infty \boldsymbol{\epsilon}^\Delta - \boldsymbol{\epsilon}^K \boldsymbol{\Delta}^\infty \implies \boldsymbol{\epsilon}^\Delta(t) = -\int_0^t ds \exp\left(-\boldsymbol{K}^\infty (t-s)\right) \boldsymbol{\epsilon}^K \exp\left(-\boldsymbol{K}^\infty s\right) \boldsymbol{y} \tag{59}$$

Projecting these dynamics onto the eigenspace of the kernel gives

$$\epsilon_k^\Delta(t) = -\sum_\ell \epsilon_{k\ell}^K \frac{e^{-\lambda_\ell t} - e^{-\lambda_k t}}{\lambda_k - \lambda_\ell} y_\ell \tag{60}$$

where $\ell = k$ should be seen as the limit where $\lambda_k \to \lambda_\ell$ of the above. Thus we find that the leading mean correction to the error solves the following differential equation

$$\begin{aligned}
\left(\frac{d}{dt} + \lambda_k\right)\Delta_k^1(t) &= -\sum_\ell K_{k\ell}^1 y_\ell e^{-\lambda_\ell t} + \sum_{\ell\ell'} \Sigma_{k\ell\ell'}^K \frac{e^{-\lambda_{\ell'} t} - e^{-\lambda_\ell t}}{\lambda_\ell - \lambda_{\ell'}} y_{\ell'}. \\
&= \sum_\ell y_\ell e^{-\lambda_\ell t}\left[-K_{k\ell}^1 + \Sigma_{k\ell\ell\ell}^K t\right] + \sum_{\ell \neq \ell'} \Sigma_{k\ell\ell'}^K \frac{e^{-\lambda_{\ell'} t} - e^{-\lambda_\ell t}}{\lambda_\ell - \lambda_{\ell'}} y_{\ell'}
\end{aligned} \tag{61}$$

We see that at late sufficiently large t, that the terms involving Σ^K will dominate. We can gain more intuition by considering the special case of a single training data point where the mean error correction has the form

$$\begin{aligned}
&\left(\frac{d}{dt} + \lambda\right)\Delta^1(t) = y e^{-\lambda t}\left[-K^1 + t\Sigma^K\right] \implies \Delta^1(t) = y\left[-tK^1 + \frac{1}{2}t^2\Sigma^K\right] e^{-\lambda t} \\
&\implies \langle \Delta(t)^2 \rangle \sim \Delta^\infty(t)^2 + \frac{1}{N}\left[2y^2 t e^{-2\lambda t}\left[-K^1 + \frac{1}{2}t\Sigma^K\right] + \Sigma^\Delta(t,t)\right] + \mathcal{O}(N^{-2}) \\
&\sim \Delta^\infty(t)^2 + \frac{2}{N}y^2 t e^{-2\lambda t}\left[-K^1 + \Sigma^K t\right] + \mathcal{O}(N^{-2})
\end{aligned} \tag{62}$$

While the term involving Σ^K is positive for all t, K^1 could be positive or negative for a given architecture. If K^1 is positive, then MSE is initially improved at early times but after $t > \frac{K^1}{\Sigma^K}$ the MSE is worse than the infinite width. On the other hand, if K^1 is negative (as we suspect is typically the case), then the MSE will strictly decrease with network width for any time t.

H Two Layer Equations and Time/Time Diagonal

In this section, we analyze two layer networks in greater detail. Unlike the deep network case, two layer networks can be analyzed on the time-time diagonal: ie the dynamics only depend on $\Phi(t,t)$ and $G(t,t)$ rather than on all possible off-diagonal pairs of time points. Further, there are no response functions A^ℓ, B^ℓ which complicate the recipe for calculating the propagator (Appendix D).

H.1 A Single Training Point

For a two layer network trained on a single training point with norm constraint $|\boldsymbol{x}|^2 = D$, we have the following DMFT action

$$S[\{K(t), \hat{K}(t), \Delta(t), \hat{\Delta}(t)\}] \tag{63}$$
$$= \int dt \left[K(t)\hat{K}(t) + \hat{\Delta}(t) \left(\Delta(t) - y + \int ds\, \Theta(t-s)\Delta(s)K(s) \right) \right]$$
$$+ \ln \mathcal{Z}[\hat{K}, f] \ , \ \mathcal{Z} = \mathbb{E}_{h,g} \exp\left(- \int dt \hat{K}(t)[\phi(h(t))^2 + g(t)^2] \right).$$

The saddle point equations are

$$\begin{aligned}
\frac{\partial S}{\partial \hat{K}(t)} &= K(t) - \left\langle [\phi(h(t))^2 + g(t)^2] \right\rangle = 0 \\
\frac{\partial S}{\partial \hat{\Delta}(t)} &= \Delta(t) - y + \int ds\, \Theta(t-s)\Delta(s)K(s) = 0 \\
\frac{\partial S}{\partial K(s)} &= \hat{K}(s) + \Delta(s) \int dt\, \hat{\Delta}(t)\Theta(t-s) = 0 \\
\frac{\partial S}{\partial \Delta(s)} &= \hat{\Delta}(s) + K(s) \int dt\, \hat{\Delta}(t)\Theta(t-s) = 0
\end{aligned} \tag{64}$$

From these equations, we can compute the entries in the Hessian of the DMFT action S. Letting $\boldsymbol{q}(t) = \begin{bmatrix} \Delta(t) \\ K(t) \end{bmatrix}$ and $\hat{\boldsymbol{q}}(t) = \begin{bmatrix} \hat{\Delta}(t) \\ \hat{K}(t) \end{bmatrix}$

$$\begin{aligned}
\frac{\partial^2 S}{\partial \boldsymbol{q}(t) \partial \boldsymbol{q}(s)^\top} &= \mathbf{0} \\
\frac{\partial^2 S}{\partial \hat{\boldsymbol{q}}(t) \partial \boldsymbol{q}(s)^\top} &= \begin{bmatrix} \delta(t-s) + \Theta(t-s)K(s) & \Theta(t-s)\Delta(s) \\ -\left\langle \frac{\partial}{\partial \Delta(s)} (\phi(h(t))^2 + g(t)^2) \right\rangle & \delta(t-s) \end{bmatrix} \\
\frac{\partial^2 S}{\partial \hat{\boldsymbol{q}}(t) \partial \hat{\boldsymbol{q}}(s)^\top} &= \begin{bmatrix} 0 & 0 \\ 0 & \kappa(t,s) \end{bmatrix}
\end{aligned} \tag{65}$$

where $\kappa(t,s) = \left\langle (\phi(h(t))^2 + g(t)^2)(\phi(h(s))^2 + g(s)^2) \right\rangle - K(t)K(s)$ is the NTK's fourth cumulant. We now vectorize our order parameters over time $\boldsymbol{q} = \text{Vec}\{\boldsymbol{q}(t)\}_{t \in \mathbb{R}_+}$ and $\hat{\boldsymbol{q}} = \text{Vec}\{\hat{\boldsymbol{q}}(t)\}_{t \in \mathbb{R}_+}$ and express the full Hessian

$$\nabla^2 S = \begin{bmatrix} \mathbf{0} & \frac{\partial^2 S}{\partial \boldsymbol{q} \partial \hat{\boldsymbol{q}}^\top} \\ \frac{\partial^2 S}{\partial \hat{\boldsymbol{q}} \partial \boldsymbol{q}^\top} & \frac{\partial^2 S}{\partial \hat{\boldsymbol{q}} \partial \hat{\boldsymbol{q}}^\top} \end{bmatrix} \implies -[\nabla^2 S]^{-1} = \begin{bmatrix} (\frac{\partial^2 S}{\partial \hat{\boldsymbol{q}} \partial \boldsymbol{q}^\top})^{-1} \frac{\partial^2 S}{\partial \hat{\boldsymbol{q}} \partial \hat{\boldsymbol{q}}} (\frac{\partial^2 S}{\partial \boldsymbol{q} \partial \hat{\boldsymbol{q}}^\top})^{-1} & -(\frac{\partial^2 S}{\partial \hat{\boldsymbol{q}} \partial \boldsymbol{q}^\top})^{-1} \\ -(\frac{\partial^2 S}{\partial \boldsymbol{q} \partial \hat{\boldsymbol{q}}^\top})^{-1} & \mathbf{0} \end{bmatrix} \tag{66}$$

The covariance matrix of interest (for $\boldsymbol{q}(t)$) is thus

$$\boldsymbol{\Sigma}_{\boldsymbol{q}} = \begin{bmatrix} \mathbf{I} + \boldsymbol{\Theta}_K & \boldsymbol{\Theta}_\Delta \\ -\boldsymbol{D} & \mathbf{I} \end{bmatrix}^{-1} \begin{bmatrix} 0 & 0 \\ 0 & \boldsymbol{\kappa} \end{bmatrix} \begin{bmatrix} \mathbf{I} + \boldsymbol{\Theta}_K & \boldsymbol{\Theta}_\Delta \\ -\boldsymbol{D} & \mathbf{I} \end{bmatrix}^{-1\top} . \tag{67}$$

where $[\boldsymbol{\Theta}_K](t,s) = \Theta(t-s)K(s)$ and $[\boldsymbol{\Theta}_\Delta](t,s) = \Theta(t-s)\Delta(s)$. The above equations allow one to use the infinite width DMFT dynamics for $K(t), \Delta(t)$ to compute the finite size fluctuation dynamics of the kernel K and the error signal Δ.

H.1.1 Computing Field Sensitivities

In this section, we compute $D(t,s)$ by solving for the sensitivity of order parameters. We start with the DMFT field equations

$$h(t) = u + \gamma \int_0^t ds \Delta(s) g(s) \ , \ z(t) = r + \gamma \int_0^t ds \Delta(s) \phi(h(t)). \tag{68}$$

Now, differentiating both sides with respect to $\Delta(s')$ gives

$$\frac{\partial h(t)}{\partial \Delta(s')} = \gamma \Theta(t-s') g(s') + \gamma \int_0^t ds \Delta(s) \frac{\partial g(s)}{\partial \Delta(s')}$$
$$\frac{\partial z(t)}{\partial \Delta(s')} = \gamma \Theta(t-s') \phi(h(s')) + \gamma \int_0^t ds \Delta(s) \frac{\partial \phi(h(s))}{\partial \Delta(s')}. \tag{69}$$

We can compute D Monte carlo by iteratively solving the above equations for each sampled trajectory $\{h(t), z(t)\}$ [62, 46]. Averaging the necessary fields over the Monte-carlo samples will give us the final expressions for $D(t,s)$.

$$D(t,s) = \left\langle \frac{\partial}{\partial \Delta(s)} (\phi(h(t))^2 + g(t)^2) \right\rangle \tag{70}$$

Similarly, the uncoupled kernel variance $\kappa(t,s)$ can be evaluated via Monte-carlo sampling for nonlinear networks.

H.2 Test Point Fluctuation Dynamics

We now are in a position to calculate the test/train kernel and test prediction fluctuations. To do this systematically, we augment S with the test point prediction $f_\star$ and field $h_\star$ and introduce the kernel $K_\star(t) = \langle \phi(h(t))\phi(h_\star(t)) + g(t) g_\star(t) \rangle$. The test prediction $f_\star$ and field $h_\star$ have dynamics

$$h_\star(t) = u_\star + \gamma \int_0^t ds \Delta(s) \dot{\phi}(h_\star(s)) z(s) K_\star^x \ , \ \langle u_\star u \rangle = K_\star^x$$
$$\frac{\partial}{\partial t} f_\star(t) = K_\star(t) \Delta(t) \ , \ K_\star(t) = \langle \phi(h(t))\phi(h_\star(t)) + g(t) g_\star(t) \rangle \tag{71}$$

The augmented action for this DMFT has the form

$$\begin{aligned} S = &\int dt \ \hat{f}_\star(t) \left(f_\star(t) - \int ds \Theta(t-s) \Delta(s) K_\star(s) \right) + \int dt \ \hat{K}_\star(t) K_\star(t) \\ &+ \int dt \ \hat{\Delta}(t) \left(\Delta(t) - y + \int ds \ \Theta(t-s) \Delta(s) K(s) \right) + \int dt \ \hat{K}(t) K(t) \\ &+ \ln \mathbb{E} \exp \left(- \int \hat{K}(t) (\phi(h(t))^2 + g(t)^2) - \int \hat{K}_\star(t) (\phi(h(t))\phi(h_\star(t)) + g(t) g_\star(t)) \right) \end{aligned} \tag{72}$$

We let $\boldsymbol{q}(t) = [\Delta(t), f_\star(t), K(t), K_\star(t)]^\top$

$$\nabla^2_{\hat{\boldsymbol{q}}\hat{\boldsymbol{q}}} S[\boldsymbol{q}, \hat{\boldsymbol{q}}] = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \boldsymbol{\kappa} & \boldsymbol{\kappa}_\star^\top \\ 0 & 0 & \boldsymbol{\kappa}_\star & \boldsymbol{\kappa}_{\star\star} \end{bmatrix} \ , \ \nabla^2_{\hat{\boldsymbol{q}},\boldsymbol{q}} S[\boldsymbol{q}, \hat{\boldsymbol{q}}] = \begin{bmatrix} \mathbf{I} + \boldsymbol{\Theta}_K & 0 & \boldsymbol{\Theta}_\Delta & 0 \\ -\boldsymbol{\Theta}_{K_\star} & \mathbf{I} & 0 & -\boldsymbol{\Theta}_\Delta \\ -\boldsymbol{D} & 0 & \mathbf{I} & 0 \\ -\boldsymbol{D}_\star & 0 & 0 & \mathbf{I} \end{bmatrix}$$
$$D(t,s) = \left\langle \frac{\partial}{\partial \Delta(s)} (\phi(h(t))^2 + g(t)^2) \right\rangle \tag{73}$$
$$D_\star(t,s) = \left\langle \frac{\partial}{\partial \Delta(s)} (\phi(h(t))\phi(h_\star(t)) + g(t) g_\star(t)) \right\rangle \tag{74}$$

Our total covariance matrix / propagator is thus

$$\boldsymbol{\Sigma} = \begin{bmatrix} \mathbf{I} + \boldsymbol{\Theta}_K & 0 & \boldsymbol{\Theta}_\Delta & 0 \\ -\boldsymbol{\Theta}_{K_\star} & \mathbf{I} & 0 & -\boldsymbol{\Theta}_\Delta \\ -\boldsymbol{D} & 0 & \mathbf{I} & 0 \\ -\boldsymbol{D}_\star & 0 & 0 & \mathbf{I} \end{bmatrix}^{-1} \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \boldsymbol{\kappa} & \boldsymbol{\kappa}_\star^\top \\ 0 & 0 & \boldsymbol{\kappa}_\star & \boldsymbol{\kappa}_{\star\star} \end{bmatrix} \begin{bmatrix} \mathbf{I} + \boldsymbol{\Theta}_K & 0 & \boldsymbol{\Theta}_\Delta & 0 \\ -\boldsymbol{\Theta}_{K_\star} & \mathbf{I} & 0 & -\boldsymbol{\Theta}_\Delta \\ -\boldsymbol{D} & 0 & \mathbf{I} & 0 \\ -\boldsymbol{D}_\star & 0 & 0 & \mathbf{I} \end{bmatrix}^{-1\top} \tag{75}$$

This is the equation provided in the main text Equation (8).

H.3 Two Layer Linear Network Closed Form

For a linear network on a single data point, we can compute $D(t,s)$ and $\kappa(t,s)$ analytically. We start from the field equations

$$\frac{dh(t)}{dt} = \gamma \Delta(t) z(t) \ , \ \frac{dz(t)}{dt} = \gamma \Delta(t) h(t) \tag{76}$$

We can make a change of variables $v_+(t) = \frac{1}{\sqrt{2}}(h(t) + z(t))$ and $v_-(t) = \frac{1}{\sqrt{2}}(h(t) - z(t))$. We note that $v_+(0) = \frac{1}{\sqrt{2}}(u + r)$ and $v_-(0) = \frac{1}{\sqrt{2}}(u - r)$ are independent Gaussians. These functions $v_+(t), v_-(t)$ satisfy dynamics

$$\begin{aligned}
&\frac{dv_+}{dt} = \gamma \Delta(t) v_+(t) \ , \ \frac{dv_-(t)}{dt} = -\gamma \Delta(t) v_-(t) \\
&\quad \implies v_+(t) = \exp\left(\gamma \int_0^t ds \Delta(s) \right) v_+(0) \implies \frac{\partial v_+(t)}{\partial \Delta(s)} = \gamma v_+(t) \Theta(t-s) \\
&\quad \implies v_-(t) = \exp\left(-\gamma \int_0^t ds \Delta(s) \right) v_-(0) \implies \frac{\partial v_+(t)}{\partial \Delta(s)} = -\gamma v_-(t) \Theta(t-s)
\end{aligned} \tag{77}$$

Now, we use the fact that $v_+(0) = \frac{1}{\sqrt{2}}(u + r)$ and $v_-(0) = \frac{1}{\sqrt{2}}(u - r)$ are independent standard normal random variables to compute $K(t) = \left\langle h(t)^2 + z(t)^2 \right\rangle = \left\langle v_+(t)^2 + v_-(t)^2 \right\rangle$

$$\begin{aligned}
D(t,s) &= \frac{\partial}{\partial \Delta(s)} \left\langle h(t)^2 + z(t)^2 \right\rangle = 2\gamma \left[\left\langle v_+(t)^2 \right\rangle - \left\langle v_-(t)^2 \right\rangle \right] \Theta(t-s) \\
&= 2\gamma \left[\exp\left(2\gamma \int_0^t ds \Delta(s) \right) - \exp\left(-2\gamma \int_0^t ds \Delta(s) \right) \right] \Theta(t-s)
\end{aligned} \tag{78}$$

This operator is causal ($D(t,s) = 0$ for $s > t$) as expected and vanishes as $t \to 0$. If we take $\gamma \to 0$, we have $D(t,s) \to 0$ which agrees with our reasoning that fields h, z only depend on Δ in the feature learning regime. Since all fields are Gaussian in the linear network case, we can use Wick's theorem to obtain the exact uncoupled kernel variance in the two layer case.

$$\begin{aligned}
\kappa(t,s) &= \left\langle (h(t)^2 + z(t)^2)(h(s)^2 + z(s)^2) \right\rangle - K(t)K(s) \\
&= 2 \left\langle h(t)h(s) \right\rangle^2 + 2 \left\langle h(t)z(s) \right\rangle^2 + 2 \left\langle z(t)h(s) \right\rangle^2 + 2 \left\langle z(t)z(s) \right\rangle^2 \\
&= \left\langle v_+(t)v_+(s) + v_-(t)v_-(s) \right\rangle^2 + \left\langle v_+(t)v_+(s) - v_-(t)v_-(s) \right\rangle^2
\end{aligned} \tag{79}$$

The $v_\pm(t)$ functions are those given above. Using the fact that $\left\langle v_+(0)^2 \right\rangle = \left\langle v_-(0)^2 \right\rangle = 1$ allows us to easily compute the single site average above.

I Multiple Samples with Whitened Data

In this section, we analyze the role that sample number plays in dynamics in a simplified model of a two layer linear network trained on whitened data. Concretely, we assume that $\frac{\boldsymbol{x}_\mu \cdot \boldsymbol{x}_\nu}{D} = \delta_{\mu\nu}$. The field equations for preactivations $h_\mu(t)$ and pregradients $z(t)$ obey

$$\frac{d}{dt} h_\mu(t) = \gamma \Delta_\mu(t) z(t) \ , \ \frac{d}{dt} z(t) = \gamma \sum_{\mu=1}^P \Delta_\mu(t) h_\mu(t) \tag{80}$$

We will assume the targets have unit norm $|\boldsymbol{y}|^2 = 1$ and we define the projection of $\boldsymbol{\Delta}$ onto the target as $\Delta_y(t) = \boldsymbol{y} \cdot \boldsymbol{\Delta}(t)$. The other $P - 1$ orthogonal components are denoted $\boldsymbol{\Delta}_\perp(t)$ so that $\boldsymbol{\Delta} = \Delta_y(t) \boldsymbol{y} + \boldsymbol{\Delta}_\perp(t)$ with $\boldsymbol{\Delta}_\perp(t) \cdot \boldsymbol{y} = 0$. At infinite width, $\boldsymbol{\Delta}_\perp = 0$ and our field equations become

$$\frac{d}{dt} h_y(t) = \Delta_y(t) z(t) \ , \ \frac{d}{dt} z(t) = \Delta_y(t) h_y(t) \ , \ \boldsymbol{\Delta}_\perp(t) = 0 \ , \ h_\perp \sim \mathcal{N}(0,1) \tag{81}$$

However, at finite width N, the off-target predictions $\boldsymbol{\Delta}_\perp$ fluctuate over random initialization. To model all of the fluctuations simultaneously, we consider the following action

$$S = \gamma \int dt \sum_\mu \hat{\Delta}_\mu(t) (\Delta_\mu(t) - y_\mu) + \ln \mathbb{E} \exp\left(\int dt \sum_\mu \hat{\Delta}_\mu(t) z(t) h_\mu(t) \right) \tag{82}$$

which enforces the constraint that $\Delta_\mu(t) = y_\mu - \frac{1}{\gamma}\langle z(t) h_\mu(t) \rangle$ at infinite width. The Hessian over order parameters $\boldsymbol{q} = \text{Vec}\{\Delta_\mu(t), \hat{\Delta}_\mu(t)\}$ has the form

$$\nabla^2_{\boldsymbol{q}} S = \begin{bmatrix} \mathbf{0} & (\gamma \mathbf{I} + \boldsymbol{D})^\top \\ \gamma \mathbf{I} + \boldsymbol{D} & \boldsymbol{\kappa} \end{bmatrix} \ , \ D_{\mu\nu}(t,s) = \left\langle \frac{\partial}{\partial \Delta_\nu(s)} z(t) h_\mu(t) \right\rangle \tag{83}$$

We thus get the following covariance for predictions $\boldsymbol{\Sigma}_\Delta = (\gamma \mathbf{I} + \boldsymbol{D})^{-1} \boldsymbol{\kappa} \left[(\gamma \mathbf{I} + \boldsymbol{D})^{-1} \right]^\top$. We now compute the necessary components of the D tensor

$$\begin{aligned} \frac{\partial h_\mu(t)}{\partial \Delta_\nu(s)} &= \gamma \delta_{\mu\nu} \Theta(t-s) z(s) + \gamma \int_0^t dt' \Delta_\mu(t') \frac{\partial z(t')}{\partial \Delta_\nu(s)} \\ \frac{\partial z(t)}{\partial \Delta_\nu(s)} &= \gamma \Theta(t-s) h_\nu(s) + \gamma \int_0^t dt' \sum_\mu \Delta_\mu(t') \frac{\partial h_\mu(t')}{\partial \Delta_\nu(s)} \\ &= \gamma \Theta(t-s) h_\nu(s) + \gamma \int_0^t dt' \Delta_y(t') \frac{\partial h_y(t')}{\partial \Delta_\nu(s)} \end{aligned} \tag{84}$$

In the last line, we used the fact that these equations are to be evaluated at the mean field infinite width stochastic process where $\Delta_\perp(t) = 0$. To compute the sensitivity tensor D, we find the following equations for our correlators of interest:

$$\begin{aligned} \left\langle \frac{\partial h_\mu(t)}{\partial \Delta_\nu(s)} z(t) \right\rangle &= \delta_{\mu\nu} \gamma \Theta(t-s) \langle z(s) z(t) \rangle \ , \ \mu, \nu \neq y \\ \left\langle \frac{\partial z(t)}{\partial \Delta_\nu(s)} h_\mu(t) \right\rangle &= \gamma \Theta(t-s) \delta_{\mu\nu} \ , \ \mu, \nu \neq y \\ \left\langle \frac{\partial h_y(t)}{\partial \Delta_y(s)} z(t) \right\rangle &= \gamma \Theta(t-s) \langle z(s) z(t) \rangle + \gamma \int_0^t dt' \Delta_y(t') \left\langle \frac{\partial z(t')}{\partial \Delta_y(s)} z(t) \right\rangle \\ \left\langle \frac{\partial z(t)}{\partial \Delta_y(s)} z(t') \right\rangle &= \gamma \Theta(t-s) \langle h_y(s) z(t) \rangle + \gamma \int_0^t dt'' \Delta_y(t'') \left\langle \frac{\partial h_y(t'')}{\partial \Delta_y(s)} z(t') \right\rangle \end{aligned} \tag{85}$$

We therefore see that the components of D decouple over indices. In the $\boldsymbol{y}$ direction, we have the following equations

$$D_y(t,s) = \left\langle \frac{\partial h_y(t)}{\partial \Delta_y(s)} z(t) \right\rangle + \left\langle \frac{\partial z(t)}{\partial \Delta_y(s)} h_y(t) \right\rangle \tag{86}$$

where the correlators must be solved self-consistently. We will provide this solution in one moment, but first, we will look at the orthogonal directions. For the $P-1$ orthogonal directions, we obtain the explicit formula for D in each of these directions

$$\begin{aligned} D_\perp(t,s) &= \left\langle \frac{\partial h_\perp(t)}{\partial \Delta_\perp(s)} z(t) \right\rangle + \left\langle \frac{\partial z(t)}{\partial \Delta_\perp(s)} h_\perp(t) \right\rangle \\ &= \gamma \Theta(t-s) \langle z(t) z(s) \rangle + \gamma \Theta(t-s) \end{aligned} \tag{87}$$

Now, we return to D_y. To solve these equations we utilize the change of variables employed in the single sample case $v_+(t) = \frac{1}{\sqrt{2}}(h_y(t) + z(t)), v_-(t) = \frac{1}{\sqrt{2}}(h_y(t) - z(t))$ (see Appendix H.3). This orthogonal transformation decouples the dynamics

$$\frac{d}{dt} v_+(t) = \gamma \Delta_y(t) v_+(t) \ , \ \frac{d}{dt} v_-(t) = -\gamma \Delta_y(t) v_-(t) \tag{88}$$

As a consequence, the field derivatives close

$$\begin{aligned} \frac{\partial v_+(t)}{\partial \Delta_y(s)} &= \gamma \Theta(t-s) v_+(s) + \int_0^t dt' \Delta_y(t') \frac{\partial v_+(t')}{\partial \Delta_y(s)} \\ \frac{\partial v_-(t)}{\partial \Delta_y(s)} &= -\gamma \Theta(t-s) v_-(s) - \int_0^t dt' \Delta_y(t') \frac{\partial v_-(t')}{\partial \Delta_y(s)} \end{aligned} \tag{89}$$

The correlator of interest is

$$\langle h_y(t)z(t)\rangle = \frac{1}{2}\left\langle [v_+(t)+v_-(t)][v_+(t)-v_-(t)]\right\rangle = \frac{1}{2}\left\langle v_+(t)^2 - v_-(t)^2\right\rangle \tag{90}$$

So we get that

$$\begin{aligned} D_y(t,s) &= \frac{1}{2}\left\langle \frac{\partial}{\partial \Delta_y(s)}\left(v_+(t)^2 - v_-(t)^2\right)\right\rangle \\ &= \left\langle v_+(t)\frac{\partial v_+(t)}{\partial \Delta_y(s)}\right\rangle - \left\langle v_-(t)\frac{\partial v_-(t)}{\partial \Delta_y(s)}\right\rangle \end{aligned} \tag{91}$$

Similarly, we can derive the on-target and off-target uncoupled variances $\kappa_y(t,s)$ and $\kappa_\perp(t,s)$, which satisfy

$$\begin{aligned} \kappa_y(t,s) &= \left\langle v_+(t)v_+(s) + v_-(t)v_-(s)\right\rangle^2 + \left\langle v_+(t)v_+(s) - v_-(t)v_-(s)\right\rangle^2 \\ \kappa_\perp(t,s) &= \frac{1}{2}\left\langle v_+(t)v_+(s) + v_-(t)v_-(s)\right\rangle \end{aligned} \tag{92}$$

Using these functions, we arrive at the following variance for each of the P dimensions

$$\begin{aligned} \boldsymbol{\Sigma}_{\Delta_y} &= (\gamma \mathbf{I} + \boldsymbol{D}_y)^{-1}\boldsymbol{\kappa}_y(\gamma \mathbf{I} + \boldsymbol{D}_y)^{-1} \\ \boldsymbol{\Sigma}_{\Delta_\perp} &= (\gamma \mathbf{I} + \boldsymbol{D}_\perp)^{-1}\boldsymbol{\kappa}_\perp(\gamma \mathbf{I} + \boldsymbol{D}_\perp)^{-1} \end{aligned} \tag{93}$$

Using the fact that all $\Delta_\perp$ variables are independent and identically distributed under the leading order picture, the expected training loss has the form

$$\left\langle |\boldsymbol{\Delta}|^2\right\rangle \approx \Delta_y^\infty(t)^2 + \frac{2}{N}\Delta_y^1(t)\Delta_y^\infty(t) + \frac{1}{N}\Sigma_{\Delta_y}(t,t) + \frac{(P-1)}{N}\Sigma_{\Delta_\perp}(t,t) + \mathcal{O}(N^{-2}). \tag{94}$$

where $\left\langle \Delta_y - \Delta_y^\infty\right\rangle = \frac{1}{N}\Delta_y^1(t) + \mathcal{O}(N^{-2})$. We note that the bias correction if $\mathcal{O}(N^{-1})$ while the variance is $\mathcal{O}(P/N)$. We compare the above leading order theory with and without the bias correction in Appendix Figure A.4.

J Online Learning

Our technology for computing finite size effects can easily be translated to a setting where the neural network is trained in an online fashion, disregarding the effect of SGD noise. At each step, we compute the gradient over the full data distribution $p(\boldsymbol{x})$. Focusing on MSE loss, we study the following equation

$$\frac{d}{dt}\Delta(\boldsymbol{x},t) = -\mathbb{E}_{\boldsymbol{x}'\sim p(\boldsymbol{x}')}K(\boldsymbol{x},\boldsymbol{x}';t)\Delta(\boldsymbol{x}',t) \tag{95}$$

where $K(\boldsymbol{x},\boldsymbol{x}';t)$ is the dynamic NTK and $\Delta(\boldsymbol{x},t) = y(\boldsymbol{x}) - f(\boldsymbol{x},t)$ is the prediction error. In general the distribution involves integration over an uncountable set of possible inputs $\boldsymbol{x}$. To remedy this, we utilize a countable orthonormal basis of functions for the data distribution $\{\psi_k(\boldsymbol{x})\}_{k=1}^\infty$. For example, if $p(\boldsymbol{x})$ were the isotropic Gaussian density for $\mathcal{N}(0,\mathbf{I})$, then ψ_k could be Hermite polynomials. We expand Δ and K in this basis ψ_k, and arrive at the following differential equation

$$\frac{d}{dt}\Delta_k(t) = -\sum_\ell K_{k\ell}(t)\Delta_\ell(t) \tag{96}$$

By orthonormality, the average turned into a sum over all possible orthonormal functions $\{\psi_k\}$. We note that since K is evolving in time, there is not generally a fixed basis of functions that diagonalize K, resulting in the couplings across eigenmodes in Equation (96). Since, in online learning, there is no distinction between the training and test distribution, our error of interest is simply $\mathcal{L}(t) = \sum_k \Delta_k(t)^2$. To obtain the finite size corrections to this quantity, we compute the joint propagator for all variables $\{K_{k\ell}(t), \Delta_k(t)\}$. If we wanted to pursue a perturbation theory in rates (Appendix F.2), we could again define a transition matrix $\boldsymbol{T}$ and rate matrix $\boldsymbol{R}(t)$ as

$$\boldsymbol{R}(t) = -\log \boldsymbol{T}(t)\ ,\ \frac{d}{dt}T_{k\ell}(t) = -\sum_{k'} K_{kk'}(t)T_{k'\ell}(t)\ ,\ T_{k\ell}(0) = \delta_{k\ell} \tag{97}$$

We can then obtain $\boldsymbol{\Delta} = \exp(-\boldsymbol{R}(t))\boldsymbol{y}$, where $y_k = \mathbb{E}_{\boldsymbol{x}}\psi_k(\boldsymbol{x})y(\boldsymbol{x})$. Since $\boldsymbol{R}$ has a finite size mean correction and finite size fluctuations, so too does the error $\Delta_k(t)$ and the loss $\mathcal{L}$ (Appendix F.2).

J.1 Two Layer Networks

In the two layer case, instead of tracking kernels, we could instead deal with the distribution over read-in vectors $\boldsymbol{w} \in \mathbb{R}^D$ and readout scalars $a \in \mathbb{R}$ as in the original works on mean field networks [6, 63]. When training on the population risk equations for $\boldsymbol{x} \sim \mathcal{N}(0, \mathbf{I})$

$$\begin{aligned}\frac{d}{dt}\boldsymbol{w} &= a\mathbb{E}_{\boldsymbol{x}}\Delta(\boldsymbol{x})\dot{\phi}(\boldsymbol{w}\cdot\boldsymbol{x})\boldsymbol{x} = \mathbb{E}_{\boldsymbol{x}}\frac{\partial\Delta(\boldsymbol{x})}{\partial\boldsymbol{x}}\dot{\phi}(\boldsymbol{w}\cdot\boldsymbol{x}) + \mathbb{E}\Delta(\boldsymbol{x})\ddot{\phi}(\boldsymbol{w}\cdot\boldsymbol{x})\boldsymbol{w} \\ \frac{d}{dt}a &= \mathbb{E}_{\boldsymbol{x}}\Delta(\boldsymbol{x})\phi(\boldsymbol{w}\cdot\boldsymbol{x})\end{aligned} \tag{98}$$

The action has the form

$$S = \gamma\int dtd\boldsymbol{x}\hat{\Delta}(t,\boldsymbol{x})(\Delta(t,\boldsymbol{x}) - y(\boldsymbol{x})) + \ln\mathbb{E}_{a,\boldsymbol{w}}\exp\left(\int dtd\boldsymbol{x}\hat{\Delta}(t,\boldsymbol{x})a(t)\phi(\boldsymbol{w}(t)\cdot\boldsymbol{x})\right) \tag{99}$$

The Hessian over $\boldsymbol{q} = \{\Delta_\mu(t), \hat{\Delta}_\mu(t)\}$ is

$$\nabla^2 S = \begin{bmatrix} 0 & \mathbf{I} + \boldsymbol{D}_\Delta \\ \mathbf{I} + \boldsymbol{D}_\Delta & \boldsymbol{\kappa} \end{bmatrix}. \tag{100}$$

where $D_\Delta(t,\boldsymbol{x};s,\boldsymbol{x}') = \left\langle \frac{\partial}{\partial\Delta(s,\boldsymbol{x}')}a(t)\phi(\boldsymbol{w}(t)\cdot\boldsymbol{x})\right\rangle$ We can use the following implicit rule

$$\begin{aligned}\frac{\partial a(t)}{\partial\Delta(s,\boldsymbol{x})} &= \gamma\Theta(t-s)p(\boldsymbol{x})\phi(\boldsymbol{w}(s)\cdot\boldsymbol{x}) + \gamma\mathbb{E}_{\boldsymbol{x}'}\int_0^t dt'\Delta(t',\boldsymbol{x}')\dot{\phi}(\boldsymbol{w}\cdot\boldsymbol{x}')\boldsymbol{x}'\cdot\frac{\partial\boldsymbol{w}(t)}{\partial\Delta(s,\boldsymbol{x})} \\ \frac{\partial\boldsymbol{w}(t)}{\partial\Delta(s,\boldsymbol{x})} &= \gamma\Theta(t-s)p(\boldsymbol{x})a(s)\dot{\phi}(\boldsymbol{w}(s)\cdot\boldsymbol{x})\boldsymbol{x} \\ &\quad + \gamma\mathbb{E}_{\boldsymbol{x}'}\int_0^t dt'\Delta(t',\boldsymbol{x}')\left[\frac{\partial a(t')}{\partial\Delta(s,\boldsymbol{x})}\dot{\phi}(\boldsymbol{w}\cdot\boldsymbol{x}') + a(t')\ddot{\phi}(\boldsymbol{w}\cdot\boldsymbol{x}')\frac{\partial\boldsymbol{w}(t')}{\partial\Delta(s,\boldsymbol{x})}\cdot\boldsymbol{x}'\right]\end{aligned} \tag{101}$$

The above equations could be solved and then used to compute $D_\Delta(t,\boldsymbol{x};s,\boldsymbol{x}')$ which must then be inverted to get the observed prediction variance.

J.2 Linear Activations

Using the ideas in the preceding sections, we can make more progress in the case of a two layer linear network in the online learning setting. The key idea is to track the kernel and prediction error projections onto the space of linear functions. In this case we get the following DMFT over the order parameter $\boldsymbol{\beta}(t) = \frac{1}{N}\boldsymbol{W}^\top\boldsymbol{a} \in \mathbb{R}^D$.

$$\begin{aligned}\frac{d}{dt}a(t) &= \gamma(\boldsymbol{\beta}_\star - \boldsymbol{\beta}(t))\cdot\boldsymbol{w}(t) \\ \frac{d}{dt}\boldsymbol{w}(t) &= \gamma a(t)(\boldsymbol{\beta}_\star - \boldsymbol{\beta}(t)) \\ \boldsymbol{\beta}(t) &= \frac{1}{\gamma}\langle a(t)\boldsymbol{w}(t)\rangle\end{aligned} \tag{102}$$

At infinite width, we see that the dynamics can be reduced to tracking the projection of the weights $\boldsymbol{w}$ and $\boldsymbol{\beta}$ on the $\boldsymbol{\beta}_\star$ direction. The $D-1$ off-target dimensions vanish $\boldsymbol{\beta}_\perp(t) = 0$. At infinite width, we arrive at the alignment dynamics studied in prior work [61, 9]

$$\begin{aligned}\frac{d}{dt}\boldsymbol{\beta}(t) &= \boldsymbol{M}(t)(\boldsymbol{\beta}_\star - \boldsymbol{\beta}(t)) \\ \frac{d}{dt}\boldsymbol{M}(t) &= \gamma^2\boldsymbol{\beta}(t)(\boldsymbol{\beta}_\star - \boldsymbol{\beta}(t))^\top + \gamma^2\boldsymbol{\beta}(t)(\boldsymbol{\beta}_\star - \boldsymbol{\beta}(t))^\top \\ &\quad + 2\gamma^2(\boldsymbol{\beta}_\star - \boldsymbol{\beta}(t))\cdot\boldsymbol{\beta}(t)\mathbf{I}\end{aligned} \tag{103}$$

We note that $\boldsymbol{\beta}(t) = \beta(t)\boldsymbol{\beta}_\star$ and that $\boldsymbol{M}$ has only one special eigenvector $\boldsymbol{\beta}_\star$ with eigenvalue $m_\star(t)$. It thus suffices to track evolution in this single direction

$$\frac{d}{dt}\beta(t) = m_\star(t)(\beta_\star - \beta(t))\ ,\ \frac{d}{dt}m_\star(t) = 4\gamma^2\beta(t)(\beta_\star - \beta(t)) \tag{104}$$

We note that this equation is identical to the differential equation for a single training example in Appendix I. Here $\beta_\star - \beta(t)$ plays the role of $\Delta_y(t)$ and $m_\star(t)$ plays the role of the kernel $K_y(t)$. A key observation is the conservation law $4\gamma^2 \frac{d}{dt}\beta(t)^2 = \frac{d}{dt} m_\star(t)^2$, from which it follows that $m_\star(t)^2 - 4 = 4\gamma^2 \beta(t)$ [9]

$$\frac{d}{dt}\beta(t) = 2\sqrt{1 + \gamma^2 \beta(t)^2}(\beta_\star - \beta(t)) \tag{105}$$

This is identical to the differential equations for a single sample (producing prediction $f(t)$ and kernel $K(t)$) if the following substitutions are made

$$f(t) \leftrightarrow \beta(t) \ , \ K(t) \leftrightarrow m_\star(t) \tag{106}$$

We now proceed to compute finite size corrections starting from the action

$$S = \gamma \int dt \hat{\boldsymbol{\beta}}(t) \cdot \boldsymbol{\beta}(t) + \ln \mathbb{E} \exp\left(- \int dt \hat{\boldsymbol{\beta}}(t) \cdot \boldsymbol{w}(t) a(t) \right) \tag{107}$$

The necessary ingredients are

$$\begin{aligned} \boldsymbol{\kappa}(t,s) &= \left\langle a(t)a(s)\boldsymbol{w}(t)\boldsymbol{w}(s)^\top \right\rangle - \gamma^2 \boldsymbol{\beta}(t)\boldsymbol{\beta}(s) \\ &= \langle a(t)a(s) \rangle \left\langle \boldsymbol{w}(t)\boldsymbol{w}(s)^\top \right\rangle + \langle a(s)\boldsymbol{w}(t) \rangle \left\langle a(t)\boldsymbol{w}(s)^\top \right\rangle \in \mathbb{R}^{D \times D} \end{aligned} \tag{108}$$

Similarly we have to compute the sensitivity tensor

$$\boldsymbol{D}(t,s) = \left\langle \frac{\partial}{\partial \boldsymbol{\beta}(s)^\top} a(t)\boldsymbol{w}(t) \right\rangle \in \mathbb{R}^{D \times D} \tag{109}$$

We start from the dynamics

$$\frac{d}{dt}\boldsymbol{w}(t) = \gamma a(t)(\boldsymbol{\beta}_\star - \boldsymbol{\beta}(t)) \ , \ \frac{d}{dt} a(t) = \gamma(\boldsymbol{\beta}_\star - \boldsymbol{\beta}(t)) \cdot \boldsymbol{w}(t) \tag{110}$$

Next, we have to calculate causal derivatives for fields

$$\begin{aligned} \frac{\partial}{\partial \boldsymbol{\beta}(s)^\top} \boldsymbol{w}(t) &= -\gamma \Theta(t-s) a(s) \mathbf{I} + \gamma \int_0^t dt' (\boldsymbol{\beta}_\star - \boldsymbol{\beta}(t')) \frac{\partial a(t')}{\partial \boldsymbol{\beta}(s)^\top} \\ \frac{\partial}{\partial \boldsymbol{\beta}(s)} a(t) &= -\gamma \Theta(t-s) \boldsymbol{w}(s) + \gamma \int_0^t dt' (\boldsymbol{\beta}_\star - \boldsymbol{\beta}(t')) \cdot \frac{\partial \boldsymbol{w}(t')}{\partial \boldsymbol{\beta}(s)} \end{aligned} \tag{111}$$

Following an identical argument as in I, we see that $\boldsymbol{D}$ has block diagonal structure with $D_{\beta_\star}(t,s)$ on the $\boldsymbol{\beta}_\star \boldsymbol{\beta}_\star^\top$ direction and $D_\perp(t,s)$ in any of the $D-1$ remaining directions

$$D_{\beta_\star}(t,s) = \left\langle \frac{\partial}{\partial \beta(s)} a(t) w_{\beta_\star}(t) \right\rangle \ , \ D_\perp(t,s) = \left\langle \frac{\partial}{\partial \beta_\perp(s)} a(t) w_\perp(t) \right\rangle \tag{112}$$

Similarly, $\boldsymbol{\kappa}(t,s)$ has a similar decomposition

$$\begin{aligned} \kappa_{\beta_\star}(t,s) &= \langle a(t)a(s) \rangle \langle w_{\beta_\star}(t) w_{\beta_\star}(s) \rangle + \langle a(s) w_{\beta_\star}(t) \rangle \langle a(t) w_{\beta_\star}(s) \rangle \\ \kappa_\perp(t,s) &= \langle a(t)a(s) \rangle \langle w_\perp(t) w_\perp(s) \rangle + \langle a(s) w_\perp(t) \rangle \langle a(t) w_\perp(s) \rangle \end{aligned} \tag{113}$$

The processes have the following equations at infinite width

$$\frac{d}{dt} w_{\beta_\star}(t) = \gamma a(t)(\beta_\star - \beta(t)) \ , \ \frac{d}{dt} a(t) = \gamma w_{\beta_\star}(t)(\beta_\star - \beta(t)) \ , \ \frac{d}{dt} w_\perp(t) = 0 \tag{114}$$

As a consequence we note that $\langle w_\perp(t) a(s) \rangle = 0$ so that $\kappa_\perp(t,s) = \langle a(t)a(s) \rangle$. Letting $v_+(t) = \frac{1}{\sqrt{2}}(w_{\beta_\star}(t) + a(t))$ and $v_-(t) = \frac{1}{\sqrt{2}}(w_{\beta_\star}(t) + a(t))$, we find the same decoupled stochastic processes as in Appendix H.3

$$\frac{d}{dt} v_+(t) = \gamma(\beta_\star - \beta(t)) v_+(t) \ , \ \frac{d}{dt} v_-(t) = -\gamma(\beta_\star - \beta(t)) v_-(t) \tag{115}$$

We can use these equations to perform the necessary averages for $\kappa_{\beta_\star}$ and $D_{\beta_\star}$. Lastly, we use

$$\frac{\partial}{\partial \beta_\perp(s)} w_\perp(t) = -\gamma \Theta(t-s) a(s) \tag{116}$$

to evaluate $D_\perp(t,s)$. The observed covariances are just

$$\boldsymbol{\Sigma}_{\beta_\star} = (\gamma \mathbf{I} - \boldsymbol{D}_{\beta_\star})^{-1} \boldsymbol{\kappa}_{\beta_\star} (\gamma \mathbf{I} - \boldsymbol{D}_{\beta_\star})^{-1\top} \ , \ \boldsymbol{\Sigma}_\perp = (\gamma \mathbf{I} - \boldsymbol{D}_\perp)^{-1} \boldsymbol{\kappa}_\perp (\gamma \mathbf{I} - \boldsymbol{D}_\perp)^{-1\top} \tag{117}$$

We note that these expressions are identical to those in Appendix I under the substitution $\beta_\star - \beta(t) \to \Delta(t)$ and $D \to P$. Thus the expected test risk is

$$\left\langle |\boldsymbol{\beta}(t) - \boldsymbol{\beta}_\star|^2 \right\rangle \sim (\beta(t) - \beta_\star)^2 + \frac{1}{N} \Sigma_{\beta_\star}(t,t) + \frac{(D-1)}{N} \Sigma_{\beta_\perp}(t,t) + \mathcal{O}(N^{-2}) \tag{118}$$

This recovers the variance we obtained in the multiple-sample whitened data case I.

J.3 Connections to Offline Learning in Linear Model

Remark 1 *The finite size variance of generalization error in an online learning setting with linear target function $y = \boldsymbol{\beta}^* \cdot \boldsymbol{x}$ has an identical form as the model described above. In this setting, we sample infinitely many fresh data points $\boldsymbol{x} \sim \mathcal{N}(0, \mathbf{I})$ at each step leading to the flow $\frac{d}{dt} \boldsymbol{w}_i(t) = \gamma a_i(t) \mathbb{E}_{\boldsymbol{x}} \Delta(\boldsymbol{x}) \boldsymbol{x}$ and $\frac{d}{dt} a_i(t) = \gamma \boldsymbol{w}_i(t) \cdot \mathbb{E}_{\boldsymbol{x}} \Delta(\boldsymbol{x}) \boldsymbol{x}$. The order parameter of interest in this setting is $\boldsymbol{\beta}(t) = \frac{1}{\gamma N} \sum_{i=1}^N \boldsymbol{w}_i(t) a_i(t)$. The precise correspondence between this setting and the offline setting is summarized in Table 1. We note that this argument could be extended to higher degree monomial activations as well, at the cost of tracking higher degree tensors (eg for quadratic activations $\boldsymbol{M} = \frac{1}{N} \sum_{i=1}^N a_i \boldsymbol{w}_i \boldsymbol{w}_i^\top \in \mathbb{R}^{D \times D}$ is sufficient).*

Setting	Order Params.	Target	Off-target Dims.	Loss	Variance	Infinite Quantity
Offline	$\boldsymbol{\Delta} = \boldsymbol{y} - \boldsymbol{f}$	$\boldsymbol{y}$	$P-1$	Train	$\mathcal{O}(\frac{P}{N})$	D
Online	$\boldsymbol{\beta}_\star - \boldsymbol{\beta}$	$\boldsymbol{\beta}_\star$	$D-1$	Test	$\mathcal{O}(\frac{D}{N})$	P

Table 1: Summary of the equivalence between the leading $1/N$ correction in the offline setting and the online setting for two layer linear networks. In the offline training setting, the order parameters are the errors $\boldsymbol{\Delta} = \boldsymbol{y} - \boldsymbol{f} \in \mathbb{R}^P$ while in the online case they are $\boldsymbol{\beta}_\star - \boldsymbol{\beta} \in \mathbb{R}^D$.

As in the offline case, in Fig. 3 (c) and (d) we see that the variance contribution to test loss $|\boldsymbol{\beta} - \boldsymbol{\beta}_\star|^2$ increases with input dimension D. We note that this perturbative effect to the loss dynamics is reminiscent of the deviations from mean field behavior studied in SGD [43, 44], though this present work concerns fluctuations driven by initialization variance rather than stochastic sampling of data. In Fig. 3 (e) we show that richer networks have lower variance at fixed N. Similarly, leading order theory for richer networks more accurately captures their dynamics as D/N increases (Fig. 3 (f)).

K Deep Linear Networks

For deep linear networks, the fields $h_\mu^\ell(t), g_\mu^\ell(t)$ are Gaussian and have the following self-consistent equations

$$\begin{aligned} h_\mu^\ell(t) &= u_\mu^\ell(t) + \gamma \int_0^t ds \sum_\nu \left[A_{\mu\nu}^{\ell-1}(t,s) + \Delta_\nu(s) H_{\mu\nu}^{\ell-1}(t,s) \right] g_\nu^\ell(s) \ , \ u_\mu^\ell(t) \sim \mathcal{GP}(0, \boldsymbol{H}^{\ell-1}) \\ g_\mu^\ell(t) &= r_\mu^\ell(t) + \gamma \int_0^t ds \sum_\nu \left[B_{\mu\nu}^\ell(t,s) + \Delta_\nu(s) G_{\mu\nu}^{\ell+1}(t,s) \right] h_\nu^\ell(s) \ , \ r_\mu^\ell(t) \sim \mathcal{GP}(0, \boldsymbol{G}^{\ell+1}). \end{aligned} \tag{119}$$

where $H_{\mu\nu}^\ell(t,s) = \left\langle h_\mu^\ell(t) h_\nu^\ell(s) \right\rangle$ and $G_{\mu\nu}^\ell(t,s) = \left\langle g_\mu^\ell(t) g_\nu^\ell(s) \right\rangle$ and $A_{\mu\nu}^\ell(t,s) = \left\langle \frac{\partial h_\mu^\ell(t)}{\partial r_\nu(s)} \right\rangle$ and $B_{\mu\nu}^\ell(t,s) = \left\langle \frac{\partial h_\mu^\ell(t)}{\partial r_\nu(s)} \right\rangle$ [9]. Therefore, we express the action as a differentiable function of the

order parameters by integrating over the Gaussian field distribution. For concreteness, we vectorize our fields over time and samples $\boldsymbol{h}^\ell = \text{Vec}\{h^\ell_\mu(t)\}_{\{\mu\in[P],t\in\mathbb{R}_+\}}$, $\boldsymbol{g}^\ell = \text{Vec}\{g^\ell_\mu(t)\}_{\{\mu\in[P],t\in\mathbb{R}_+\}}$ we consider the contribution of a single hidden layer.

$$\mathcal{Z}_\ell = \int d\hat{\boldsymbol{h}}^\ell d\hat{\boldsymbol{g}}^\ell d\boldsymbol{h}^\ell d\boldsymbol{g}^\ell \exp\left(-\frac{1}{2}\hat{\boldsymbol{h}}^\ell \boldsymbol{\Sigma}_u \hat{\boldsymbol{h}}^\ell + i\hat{\boldsymbol{h}}^\ell \cdot (\boldsymbol{h}^\ell - \boldsymbol{C}^\ell \boldsymbol{g}^\ell) - \frac{1}{2}\boldsymbol{h}^{\ell\top}\hat{\boldsymbol{H}}^\ell \boldsymbol{h}^\ell\right)$$
$$\exp\left(-\frac{1}{2}\hat{\boldsymbol{g}}^\ell \boldsymbol{\Sigma}^\ell_r \hat{\boldsymbol{g}}^\ell + i\hat{\boldsymbol{g}}^\ell \cdot (\boldsymbol{g}^\ell - \boldsymbol{D}^\ell \boldsymbol{h}^\ell) - \frac{1}{2}\boldsymbol{g}^{\ell\top}\hat{\boldsymbol{G}}^\ell \boldsymbol{g}^\ell\right)$$

where $C^\ell_{\mu\nu}(t,s) = \gamma\Theta(t-s)\left[A^{\ell-1}_{\mu\nu}(t,s) + H^{\ell-1}_{\mu\nu}(t,s)\Delta_\nu(s)\right]$ and $D^\ell_{\mu\nu}(t,s) = \gamma\Theta(t-s)\left[B^\ell_{\mu\nu}(t,s) + G^{\ell+1}_{\mu\nu}(t,s)\Delta_\nu(s)\right]$. Performing the joint Gaussian integrals over $(\boldsymbol{h}^\ell, \boldsymbol{g}^\ell, \hat{\boldsymbol{h}}^\ell, \hat{\boldsymbol{g}}^\ell)$ we find

$$\ln \mathcal{Z}_\ell = -\frac{1}{2}\ln\det\begin{bmatrix} -\hat{\boldsymbol{H}}^\ell & 0 & \mathbf{I} & -\boldsymbol{D}^{\ell\top} \\ 0 & -\hat{\boldsymbol{G}}^\ell & -\boldsymbol{C}^{\ell\top} & \mathbf{I} \\ \mathbf{I} & -\boldsymbol{C}^\ell & \boldsymbol{\Sigma}_u & 0 \\ -\boldsymbol{D}^\ell & \mathbf{I} & 0 & \boldsymbol{\Sigma}_r \end{bmatrix} \tag{120}$$

We can then automatically differentiate the DMFT action to get the propagator. For example, for a three layer linear network, the full DMFT action has the form

$$\begin{aligned} S = &\frac{1}{2}\text{Tr}\left[\hat{\boldsymbol{H}}^1\boldsymbol{H}^1 + \hat{\boldsymbol{H}}^2\boldsymbol{H}^2 + \hat{\boldsymbol{G}}^1\boldsymbol{G}^1 + \hat{\boldsymbol{G}}^2\boldsymbol{G}^2\right] - \gamma^2\text{Tr}\boldsymbol{A}\boldsymbol{B} \\ &-\frac{1}{2}\ln\det\begin{bmatrix} -\hat{\boldsymbol{H}}^1 & 0 & \mathbf{I} & -\boldsymbol{D}^{1\top} \\ 0 & -\hat{\boldsymbol{G}}^1 & -\boldsymbol{C}^{1\top} & \mathbf{I} \\ \mathbf{I} & -\boldsymbol{C}^1 & \mathbf{1}\mathbf{1}^\top & 0 \\ -\boldsymbol{D}^1 & \mathbf{I} & 0 & \boldsymbol{G}^2 \end{bmatrix} \\ &-\frac{1}{2}\ln\det\begin{bmatrix} -\hat{\boldsymbol{H}}^2 & 0 & \mathbf{I} & -\boldsymbol{D}^{2\top} \\ 0 & -\hat{\boldsymbol{G}}^2 & -\boldsymbol{C}^{2\top} & \mathbf{I} \\ \mathbf{I} & -\boldsymbol{C}^2 & \boldsymbol{H}^1 & 0 \\ -\boldsymbol{D}^2 & \mathbf{I} & 0 & \mathbf{1}\mathbf{1}^\top \end{bmatrix} \end{aligned} \tag{121}$$

where $\boldsymbol{C}^1 = \gamma\boldsymbol{\Theta}_\Delta$ and $\boldsymbol{C}^2 = \gamma\boldsymbol{\Theta}_\Delta \odot \boldsymbol{H}^1 + \gamma\boldsymbol{A}$ and $\boldsymbol{D}^1 = \gamma\boldsymbol{\Theta}_\Delta \odot \boldsymbol{G}^2 + \gamma\boldsymbol{B}$ and $\boldsymbol{D}^2 = \gamma\boldsymbol{\Theta}_\Delta$. This above example can be extended to deeper networks. The total size of the block matrices which we compute determinants over is $4PT \times 4PT$ for a dataset of size P trained for T steps.

L Discrete Time Dynamics and Edge of Stability Effects

Large step size effects can induce qualitatively different dynamics in neural network training. For instance, if the step size exceeds that required for linear stability with the initial kernel, the kernel can decrease in order to stabilize the dynamics [57]. Alternatively, during training the kernel may exhibit a "progressive sharpening" phase where its top eigenvalue grows before reaching a stability bound set by the learning rate [19]. It is therefore well motivated to study how dynamics in this regime alter finite size effects in neural networks. We will first solve a special model which was considered in prior work [57]: a two layer linear network trained on a single training point. We will then provide the full DMFT equations for the discrete time case and provide an outline for how one could obtain finite size effects in that picture.

L.1 Two Layer Linear Equations

In a two layer linear network, the DMFT equations are

$$\begin{aligned} h(t+1) = h(t) + \eta\gamma\Delta(t)z(t)\ ,\ z(t+1) = z(t) + \eta\gamma\Delta(t)h(t) \\ f(t) = \frac{1}{\gamma}\langle z(t)h(t)\rangle \end{aligned} \tag{122}$$

The NTK has the form $K(t) = \left\langle h(t)^2 + z(t)^2 \right\rangle$. We can easily show that the kernel and error have coupled dynamics

$$\begin{aligned} f(t+1) &= f(t) + \eta \left\langle h(t)^2 + z(t)^2 \right\rangle \Delta(t) + \eta^2 \gamma \Delta(t)^2 \left\langle h(t) z(t) \right\rangle \\ &= f(t) + \eta K(t) \Delta(t) + \eta^2 \gamma^2 \Delta(t)^2 f(t) \end{aligned} \tag{123}$$

$$\begin{aligned} K(t+1) &= K(t) + 4\eta\gamma \Delta(t) \left\langle h(t) z(t) \right\rangle + \eta^2 \gamma^2 \Delta(t)^2 \left\langle h(t)^2 + z(t)^2 \right\rangle \\ &= K(t) + 4\eta\gamma^2 \Delta(t) f(t) + \eta^2 \gamma^2 \Delta(t)^2 K(t) \end{aligned} \tag{124}$$

These equations define the infinite width evolution of $\Delta(t)$ and $K(t)$. Already at this level of analysis, we can reason about the evolution of $K(t)$. In the small η limit, we could disregard terms of order $\mathcal{O}(\eta^2)$ and arrive at the following gradient flow approximation for $K(t) \sim 2\sqrt{1 + \gamma^2 f(t)^2}$ [9]. This evolution will not reach the edge of stability provided that $\eta < \frac{1}{\sqrt{1+\gamma^2 y^2}}$. For large γ and $y = 1$, this leads to the constraint $\eta\gamma < 1$. However, if η exceeds this bound, the gradient flow approximation is no longer reasonable and the system reaches an edge of stability effect as shown in Figure 5.

To calculate the finite size effects, we need to compute κ and $D(t,s) = \frac{\partial}{\partial \Delta(s)} \left\langle h(t)^2 + z(t)^2 \right\rangle$. To evaluate these quantities we utilize the same change of variables employed in Appendix H.3. In discrete time, these decoupled equations are

$$v_+(t+1) = v_+(t) + \eta\gamma \Delta(t) v_+(t) \ , \ v_-(t+1) = v_-(t) - \eta\gamma \Delta(t) v_-(t). \tag{125}$$

Given $\Delta(t)$, these can be expressed as linear systems of equations. Now, we can easily compute the uncoupled kernel variance

$$\begin{aligned} \kappa(t,s) &= 2 \left\langle h(t) h(s) \right\rangle^2 + 2 \left\langle z(t) z(s) \right\rangle^2 + 2 \left\langle h(t) z(s) \right\rangle^2 + 2 \left\langle z(t) h(s) \right\rangle^2 \\ &= \left\langle v_+(t) v_+(s) + v_-(t) v_-(s) \right\rangle^2 + \left\langle v_+(t) v_+(s) - v_-(t) v_-(s) \right\rangle^2 . \end{aligned} \tag{126}$$

Similarly, we can calculate $D(t,s)$ by using the fact $\left\langle h(t)^2 + z(t)^2 \right\rangle = \left\langle v_+(t)^2 + v_-(t)^2 \right\rangle$

$$\begin{aligned} D(t,s) &= 2 \left\langle v_+(t) \frac{\partial v_+(t)}{\partial \Delta(s)} \right\rangle + 2 \left\langle v_-(t) \frac{\partial v_-(t)}{\partial \Delta(s)} \right\rangle \\ \frac{\partial v_+(t)}{\partial \Delta(s)} &= \gamma \Theta(t-s) v_+(s) + \sum_{t' < t} \Delta(t') \frac{\partial v_+(t')}{\partial \Delta(s)} \\ \frac{\partial v_-(t)}{\partial \Delta(s)} &= -\gamma \Theta(t-s) v_-(s) - \sum_{t' < t} \Delta(t') \frac{\partial v_-(t')}{\partial \Delta(s)} \end{aligned} \tag{127}$$

These can be directly solved as a linear system of equations.

M Computing Details

Experiments for Figures 2, 5 and A.1 were conducted on a Google Colab GPU with JAX. Experiments for Figures 4, A.5, 6 were performed on a NVIDIA SMX4-A100-80GB GPU. The total compute required for all Figures in the paper took around 4 hours.