A Dynamical Model of Neural Scaling Laws

Blake Bordelon [1 2] **Alexander Atanasov** [3 2] **Cengiz Pehlevan** [1 2]

Abstract

On a variety of tasks, the performance of neural networks predictably improves with training time, dataset size and model size across many orders of magnitude. This phenomenon is known as a neural scaling law. Of fundamental importance is the compute-optimal scaling law, which reports the performance as a function of units of compute when choosing model sizes optimally. We analyze a random feature model trained with gradient descent as a solvable model of network training and generalization. This reproduces many observations about neural scaling laws. First, our model makes a prediction about why the scaling of performance with training time and with model size have different power law exponents. Consequently, the theory predicts an asymmetric compute-optimal scaling rule where the number of training steps are increased faster than model parameters, consistent with recent empirical observations. Second, it has been observed that early in training, networks converge to their infinite-width dynamics at a rate $1/\textit{width}$ but at late time exhibit a rate $\textit{width}^{-c}$, where c depends on the structure of the architecture and task. We show that our model exhibits this behavior. Lastly, our theory shows how the gap between training and test loss can gradually build up over time due to repeated reuse of data.

1. Introduction

Large scale language and vision models have been shown to achieve better performance as the number of parameters and number of training steps are increased. Moreover, the scaling of various loss metrics (such as cross entropy or MSE test loss) has been empirically observed to exhibit remarkably regular, often power law behavior across several orders of magnitude (Hestness et al., 2017; Kaplan et al., 2020). These findings are termed "neural scaling laws".

Neural scaling laws play a central role in modern deep learning practice, and have substantial implications for the optimal trade-off between model size and training time (Hoffmann et al., 2022), as well as architecture selection (Alabdulmohsin et al., 2023). Understanding the origin of such scaling laws, as well as their exponents, has the potential to offer insight into better architectures, the design of better datasets (Sorscher et al., 2022), and the failure modes and limitations of deep learning systems. Yet, many questions about neural scaling laws remain open.

In this paper, we introduce and analyze a solvable model which captures many important aspects of neural scaling laws. In particular, we are interested in understanding the following empirically observed phenomena:

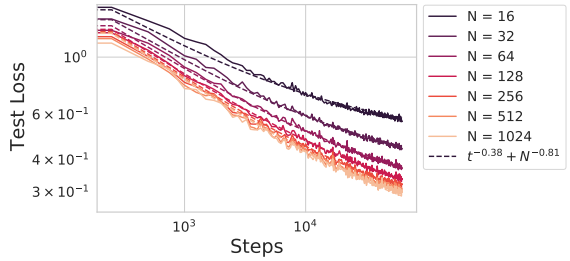
Test Loss Scales as a Power-law in Training Time and Model Size and Compute. In many domains of deep learning, the test loss of a model with N trainable parameters trained for t iterations has been found to scale as $\mathcal{L}(t, N) \approx \mathcal{L}_0 + a_t t^{-r_t} + a_N N^{-r_N}$ (Kaplan et al., 2020; Hoffmann et al., 2022). These scaling law exponents r_t, r_N generally depend on the dataset and architecture. We demonstrate scaling laws on simple vision and language tasks in Figure 1. The compute is proportional to the number of steps of gradient descent times the model size $C \propto Nt$. Setting N and t optimally gives that test loss scales as a power law in C. This is the *compute optimal scaling law*.

Compute-Optimal Training Time and Model Size Scaling Exponents Are Different. A discrepancy in exponents r_t and r_N is usually observed to some degree depdending on the data distribution and architecture Hoffmann et al. (2022); Bachmann et al. (2024). The gap between exponents would lead to asymmetric compute-optimal scaling of parameters. For compute budget C, model size should scale $N \propto C^{c_1}$ and training time $t \propto C^{c_2}$ with $c_2 > c_1$. This difference in exponents led to a change in the scaling rule for large language models, generating large performance gains.

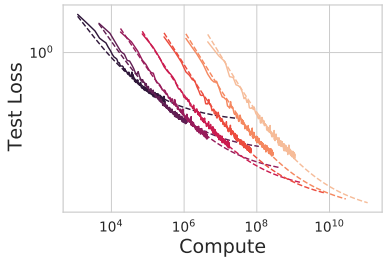
Larger Models Train Faster. Provided feature learning is held constant across model scales (i.e. adopting mean-field or μP scaling), wider networks tend to train faster (Yang et al., 2021) (Figure 1). If training proceeds in an online/one-pass setting where datapoints are not repeated,

[1]SEAS, Harvard University [2]Kempner Institute, Harvard University [3]Department of Physics, Harvard University. Correspondence to: Cengiz Pehlevan <cpehlevan@seas.harvard.edu>.

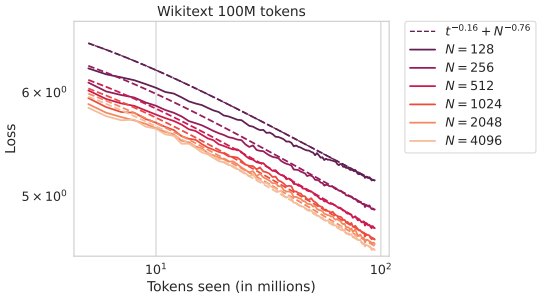
Proceedings of the 41^{st} International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024.



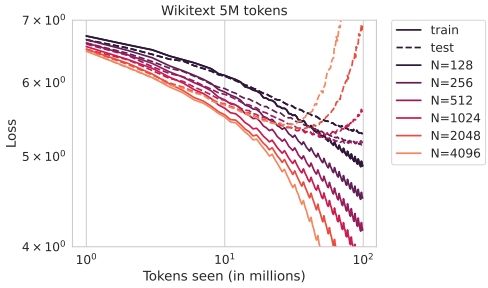
(a) One-Pass CIFAR-5M Test Dynamics



(b) CIFAR-5M Compute Scaling



(c) Wikitext-100M Train and Test



(d) Wikitext-5M Train and Test

Figure 1. Train and test losses (cross-entropy) as a function of training time t and width N. For models trained online, we do not make a distinction between training and test error because each new batch is drawn fresh and would have the same loss in expectation as an independent test set. (a) The test loss of a residual CNN on CIFAR-5M is well described by a fit of the form $\mathcal{L} \sim t^{-\alpha} + N^{-\beta}$ in the online training regime. (b) The compute optimal strategy requires scaling up both model size and training time simultaneously. (c) Transformer training on wikitext with 100M tokens before data-repetition. Model performance is monotonic in width N. (d) Wikitext with 5M subsampled tokens. Larger width N is not always better as wider models can overfit.

then the wider models will also obtain lower test loss at an equal number of iterations. This observation has been found to hold both in overparameterized and underparameterized regimes (Bordelon & Pehlevan, 2023; Vyas et al., 2023).

Models Accumulate Finite-Dataset and Finite-Width Corrections. Early training can be well described by the learning curves for stochastic gradient descent without reuse of samples (termed the online/ideal limiting dynamics), however over time the effect of reusing data accumulates and leads to worse test performance (Nakkiran et al., 2021b; Mignacco et al., 2020; Ghosh et al., 2022; Muennighoff et al., 2023). Similarly the gaps in model performance across various model sizes also grow with training time (Yang et al., 2021; Vyas et al., 2023). Figure 1 (d) shows overfitting and reversal of "wider is better" phenomenon due to data reuse.

Scaling Exponents are Task-Dependent at Late Training Time, but not at Early Time. Prior works (Dyer & Gur-Ari, 2020; Atanasov et al., 2023; Roberts et al., 2022; Bordelon & Pehlevan, 2023) predict early-time finite-width loss corrections that go as $1/\textit{width}$ near the infinite width limit in either lazy or feature-learning regimes. Bahri et al. (2021) et al provide experiments demonstrating the $1/\textit{width}$ convergence. However, finite-width models trained for a long time exhibit non-trivial exponents with respect to model width (Kaplan et al., 2020; Vyas et al., 2023). See Figure 1 for examples of nontrivial scalings at late time on CIFAR-5M and Wikitext.

Ensembling is Not the Same as Going Wider. Near the limit of infinite width, finite models can be thought of as noisy approximations of the infinite-width model with noise that can be eliminated through ensembling (Dyer & Gur-Ari, 2020; Geiger et al., 2020; Atanasov et al., 2023). However recent experiments (Vyas et al., 2023) indicate that ensembling is not enough to match performance of larger models.

These phenomena are not unique to deep networks, but can be observed in linear models, or linearized neural networks operating in the lazy/kernel regime. Though this regime does not capture feature learning, it has benefit of analytical tractability. In this paper, we focus on such linearized models to attempt to gain insight into the dynamics of training.

To attempt to explain these phenomena, we develop a mathematically tractable model of neural scaling laws which allows one to simultaneously vary time, model size, and dataset size. Our contributions are as follows:

1. We analyze the learning dynamics of a structured and randomly projected linear model trained with gradient flow, discrete time SGD, and momentum. In an asymptotic limit of the model, we obtain a dynamical mean field theory (DMFT) description of the learning curve in terms of correlation functions, which measure the cross-time correlation of training and test errors, and response functions which measure sensitivity of the dynamics to small perturbations.
2. We solve for the response functions exactly in Fourier domain. This solution reveals faster training for larger models. The low frequency range of these functions allow us to extract the long time limit of the loss.
3. We show that the model and data corrections to the dynamics accumulate over time. At early time, each of these corrections has a universal scaling, consistent with prior works (Bahri et al., 2021).
4. For power-law structured features we show that the model exhibits power law scaling of test loss with time, model size and dataset size. While the data and model exponents are the same, the time and model exponents are different in general. We show that this gives rise to an asymmetric compute optimal scaling strategy where training time increases faster than model size.
5. Our theory explains why ensembling is not compute optimal as it gives less benefit to performance than increase in model size.
6. We observe in Section 5.1 that feature learning networks can obtain better power law scalings, leading to a better compute optimal frontier. We empirically study this phenomenon in Appendix L.

1.1. Related Works

The learning curves for linear models with structured (non-isotropic) covariates, including infinite-width kernel regression, have been computed using tools from statistical physics and random matrix theory (Bordelon et al., 2020; Spigler et al., 2020; Canatar et al., 2021; Simon et al., 2021; Bahri et al., 2021; Hastie et al., 2022). Mei & Montanari (2022) analyzed a linear model with random projections of isotropic covariates. There, they study the limiting effects of width and dataset size, and observe model-wise and sample-wise double descent. In Adlam & Pennington (2020a) a related model is used to study the finite-width neural tangent kernel (NTK) (Jacot et al., 2018) of a given network. Further, d'Ascoli et al. (2020) and Adlam & Pennington (2020b) extend this analysis to understand the different sources of variance in the predictions of random feature models and the effect of ensembling and bagging on the test loss. Other works have extended this to models where an additional untrained projection is applied to the structured covariates (Loureiro et al., 2021; 2022; Zavatone-Veth et al., 2022; Atanasov et al., 2023; Maloney et al., 2022; Zavatone-Veth & Pehlevan, 2023; Ruben & Pehlevan, 2023; Simon et al., 2023). Within this literature, which considered fully trained models, the works of (Bordelon et al., 2020; Spigler et al., 2020) derived power-law decay rates for power-law features which were termed resolution limited by Bahri et al. (2021) and recovered by Maloney et al. (2022).

However, we also study the dependence on training time. The $t \to \infty$ limit of our DMFT equations recovers the final losses computed in these prior works. While these prior works find that the scaling exponents for model-size and dataset-size are the same, we find that the test loss scales with a different exponent with training time, leading to a different (model and task dependent) compute optimal scaling strategy.

DMFT methods have been used to analyze the test loss dynamics for general linear and spiked tensor models trained with high-dimensional random data (Mannelli et al., 2019; Mignacco et al., 2020; Mignacco & Urbani, 2022) and deep networks dynamics with random initialization (Bordelon & Pehlevan, 2022b; Bordelon et al., 2023). High dimensional limits of SGD have been analyzed with Volterra integral equations in the offline case (Paquette et al., 2021) or with recursive matrix equations in the online case (Varre et al., 2021; Bordelon & Pehlevan, 2022a). Random matrix approaches have also been used to study test loss dynamics in linear regression with isotropic covariates by (Advani et al., 2020) and for random feature models in (Bodin & Macris, 2021). In this work, we consider averaging over both the disorder in the sampled dataset and the random projection of the features simultaneously using DMFT.

Other models and hypotheses for scaling laws instead rely on a discrete collection of subtasks or skills which are learned as compute grows (Caballero et al., 2022; Arora & Goyal, 2023; Michaud et al., 2023). Our theory instead focuses on spectral components of a data distribution.

2. Setup of the Model

We consider a "teacher-student" setting, where data sampled from a generative teacher model is used to train a student random feature model. The teacher and student models mismatch in a particular way that will be described below. This mismatch is the key ingredient that leads to most of the phenomena that we will discuss.

Teacher Model. Take $\boldsymbol{x} \in \mathbb{R}^D$ to be drawn from a distribution $p(\boldsymbol{x})$ with a target function $y(\boldsymbol{x})$ expressible in terms of base features $\boldsymbol{\psi}(\boldsymbol{x}) \in \mathbb{R}^M$ up to noise:

$$y(\boldsymbol{x}) = \frac{1}{\sqrt{M}} \boldsymbol{w}^\star \cdot \boldsymbol{\psi}(\boldsymbol{x}) + \sigma \epsilon(\boldsymbol{x}). \tag{1}$$

Here $\psi(\boldsymbol{x})$ play the role of the infinite-width NTK eigenfunctions, which form a complete basis for square-integrable functions $L^2[p]$. The $\epsilon(\boldsymbol{x})$ function describes a component of y with which is uncorrelated with $\psi(\boldsymbol{x})$. We work in the eigenbasis of features as in (Bordelon et al., 2020), so the covariance given by:

$$\langle \psi_k(\boldsymbol{x})\psi_\ell(\boldsymbol{x})\rangle_{\boldsymbol{x}\sim p(\boldsymbol{x})} = \delta_{k\ell}\lambda_k. \tag{2}$$

The power law structure in the λ_k and $\boldsymbol{w}^*$ entries will lead to power law scalings for the test loss and related quantities.

Student Model. Our student model is motivated by a scenario where a randomly initialized finite-width network is trained in the linearized or lazy regime (Chizat et al., 2019; Jacot et al., 2018). Such training can be described through learning linear combinations of the finite-width NTK features. These features will span a lower-dimensional subspace of the space of square-integrable functions, and relate to infinite-width NTK features in a complicated way.

To model this key aspect, the student model uses a projection of the $\boldsymbol{\psi}(\boldsymbol{x})$ features, $\boldsymbol{A}\boldsymbol{\psi}(\boldsymbol{x})$ where $\boldsymbol{A} \in \mathbb{R}^{N\times M}$. These projected features represent the *finite-width* (i.e. empirical) NTK eigenfunctions. This is motivated by the fact that finite width kernel's features can be linearly expanded in the basis of infinite-width features, because infinite-width kernel eigenfunctions are complete.

Our learned function then has the form:

$$f(\boldsymbol{x}) = \frac{1}{\sqrt{N}}\boldsymbol{w}\cdot \boldsymbol{A}\boldsymbol{\psi}(\boldsymbol{x}). \tag{3}$$

Here, we will interpret N as the model size with the $N\to\infty$ limit recovering original kernel. Similar models were studied in (Maloney et al., 2022; Atanasov et al., 2023).

We will focus on the setting where the elements of $\boldsymbol{A}$ are drawn iid from a distribution of mean zero and variance one. See Appendix B for details on the technical assumptions. The motivations for this choice are (1) tractability and (2) it satisfies the constraint that as $N\to\infty$ the student's kernel approaches the infinite-width kernel $\boldsymbol{\psi}$. In more realistic settings, such as when projecting the eigenfunctions of an infinite-width NTK to a finite-width NTK, the form of the $\boldsymbol{A}$ matrix is generally not known.

Training. The model is trained on a random dataset $\mathcal{D} = \{\boldsymbol{x}_\mu, y_\mu\}_{\mu=1}^P$ of size P with gradient flow on MSE loss

$$\frac{\partial}{\partial t}\boldsymbol{w}(t) = \frac{\sqrt{M}}{P\sqrt{N}}\sum_{\mu=1}^P (y(\boldsymbol{x}_\mu) - f(\boldsymbol{x}_\mu, t))\boldsymbol{A}\boldsymbol{\psi}(\boldsymbol{x}_\mu). \tag{4}$$

We explore extensions (momentum, discrete time, one-pass SGD in Appendix K). We track the test and train loss

$$\begin{aligned}\mathcal{L}(t) &= \mathbb{E}_{\boldsymbol{x}}[(f(\boldsymbol{x},t) - y(\boldsymbol{x}))^2],\\ \hat{\mathcal{L}}(t) &= \frac{1}{P}\sum_{\mu=1}^P (f_\mu(t) - y_\mu)^2.\end{aligned} \tag{5}$$

In small size systems, these losses depend on the precise realization of the data $\mathcal{D}$ and matrix $\boldsymbol{A}$. These two quantities can be viewed as the *disorder*. For large systems, these losses approach a well-defined limit independent of the specific realization of $\mathcal{D}, \boldsymbol{A}$. We will use this fact in the next section when analyzing the model.

3. DMFT for Scaling Laws

We next describe a theoretical approach for characterizing the learning curves for this model. The full details of this approach is detailed in Appendices A, B.

We derive a mean field theory for M, N, P large. We analyze both the (1) proportional regime where $N/M = \nu, P/M = \alpha$ and $M, N, P \to \infty$, and (2) non-proportional regime where $M\to\infty$ first and $N, P \gg 1$. The theories derived in these limits are structurally identical (App. G). [1]

Let $\boldsymbol{\Psi} \in \mathbb{R}^{P\times M}$ with $\boldsymbol{\Psi}_k^\mu = \psi_k(\boldsymbol{x}^\mu)$. Also define $\boldsymbol{\Lambda}_{ij} = \lambda_i\delta_{ij}$.The discrepancy between the target weights and the model's effective weights is

$$\boldsymbol{v}^0 \equiv \boldsymbol{w}^\star - \frac{1}{\sqrt{N}}\boldsymbol{A}^\top \boldsymbol{w}(t). \tag{6}$$

The test loss is then given by $\mathcal{L}(t) = \frac{1}{M}\sum_k \lambda_k v_k^0(t)^2$. The $\boldsymbol{v}^0$ vector has the following dynamics:

$$\frac{d}{dt}\boldsymbol{v}^0(t) = -\left(\frac{1}{N}\boldsymbol{A}^\top\boldsymbol{A}\right)\left(\frac{1}{P}\boldsymbol{\Psi}^\top\boldsymbol{\Psi}\right)\boldsymbol{v}^0(t). \tag{7}$$

Already, we can see that generalization can be limited if $\boldsymbol{A}^\top\boldsymbol{A}$ or $\boldsymbol{\Psi}^\top\boldsymbol{\Psi}$ are low rank as the dynamics will be frozen in the nullspace of $\left(\frac{1}{N}\boldsymbol{A}^\top\boldsymbol{A}\right)\left(\frac{1}{P}\boldsymbol{\Psi}^\top\boldsymbol{\Psi}\right)$. Using DMFT, we characterize this limit by tracking $\boldsymbol{v}^0$ together with the following random vectors:

$$\begin{aligned}\boldsymbol{v}^1(t) &= \frac{1}{\sqrt{M}}\boldsymbol{\Psi}\boldsymbol{v}^0(t), \quad \boldsymbol{v}^2(t) = \frac{1}{\alpha\sqrt{M}}\boldsymbol{\Psi}^\top\boldsymbol{v}^1(t),\\ \boldsymbol{v}^3(t) &= \frac{1}{\sqrt{M}}\boldsymbol{A}\boldsymbol{v}^2(t), \quad \boldsymbol{v}^4(t) = \frac{1}{\nu\sqrt{M}}\boldsymbol{A}^\top\boldsymbol{v}^3(t).\end{aligned} \tag{8}$$

The key summary statistics (also called *order parameters*)

[1] While the proportional limit is exact, the finite size N, P theory will also contain fluctuations across realizations of disorder. When relevant, we show these in experiments by plotting standard deviations over draws of data and projection matrices $\boldsymbol{A}$. This variance decays as $\mathcal{O}(1/P + 1/N)$.

are the correlation functions:

$$C_0(t,s) = \frac{1}{M}\boldsymbol{v}^0(t)^\top \boldsymbol{\Lambda}\boldsymbol{v}^0(s),\ C_1(t,s) = \frac{1}{P}\boldsymbol{v}^1(t)\cdot\boldsymbol{v}^1(s),$$
$$C_2(t,s) = \frac{1}{M}\boldsymbol{v}^2(t)\cdot\boldsymbol{v}^2(s),\ C_3(t,s) = \frac{1}{N}\boldsymbol{v}^3(t)\cdot\boldsymbol{v}^3(s),$$

as well as the response functions:

$$R_1(t,s) = \frac{1}{P}\mathrm{Tr}\left[\frac{\delta \boldsymbol{v}^1(t)}{\delta \boldsymbol{v}^1(s)}\right], R_{2,4}(t,s) = \frac{1}{M}\mathrm{Tr}\left[\frac{\delta \boldsymbol{v}^2(t)}{\delta \boldsymbol{v}^4(s)}\right],$$
$$R_3(t,s) = \frac{1}{N}\mathrm{Tr}\left[\frac{\delta \boldsymbol{v}^3(t)}{\delta \boldsymbol{v}^3(s)}\right], R_{0,2}(t,s) = \frac{1}{M}\mathrm{Tr}\left[\boldsymbol{\Lambda}\frac{\delta \boldsymbol{v}^0(t)}{\delta \boldsymbol{v}^2(s)}\right].$$

Here $\frac{\delta \boldsymbol{v}^i(t)}{\delta \boldsymbol{v}^j(s)}$ is the response of $\boldsymbol{v}^i(t)$ to a kick in the dynamics of $\boldsymbol{v}^j$ at time s. See appendix B.2.1 for details.

The test loss $\mathcal{L}$ and train loss $\hat{\mathcal{L}}$ are related to the time-time diagonal of $C_0(t,s)$ and $C_1(t,s)$ respectively

$$\mathcal{L}(t) = C_0(t,t) + \sigma^2\ ,\ \hat{\mathcal{L}}(t) = C_1(t,t). \tag{9}$$

These collective quantities concentrate over random draws of the disorder (Sompolinsky & Zippelius, 1981). We show that these correlation and response functions satisfy a closed set of integro-differential equations which depend on α, ν which we provide in the Appendices A.2.

Further, we show in Appendix A.3 that the response functions possess a *time-translation invariance* property $R(t,s) = R(t-s)$. This enables exact analysis in the Fourier domain $R(\tau) = \int \frac{d\omega}{2\pi} e^{i\omega\tau}\mathcal{R}(\omega)$. These response functions can then be used to solve for the correlation functions $\{C_0, C_1, C_2, C_3\}$.

To understand the convergence of the learned function f along each eigenfunction of the kernel, we introduce the transfer function[2] for mode k, $H_k(t) \equiv \frac{\partial}{\partial w_k^\star}\left\langle v_k^0(t)\right\rangle$. Our key result is that the Fourier transform of H_k can be simply expressed in terms of the Fourier transforms of R_1, R_3:

$$\mathcal{H}_k(\omega) = \frac{1}{i\omega + \lambda_k \mathcal{R}_1(\omega)\mathcal{R}_3(\omega)}. \tag{10}$$

These functions satisfy the self-consistent equations:

$$\mathcal{R}_1(\omega) = 1 - \frac{1}{P}\sum_k \frac{\lambda_k \mathcal{R}_1(\omega)\mathcal{R}_3(\omega)}{i\omega + \lambda_k \mathcal{R}_1(\omega)\mathcal{R}_3(\omega)},$$
$$\mathcal{R}_3(\omega) = 1 - \frac{1}{N}\sum_k \frac{\lambda_k \mathcal{R}_1(\omega)\mathcal{R}_3(\omega)}{i\omega + \lambda_k \mathcal{R}_1(\omega)\mathcal{R}_3(\omega)}. \tag{11}$$

From these solved response functions $\mathcal{R}_1, \mathcal{R}_3$, we can compute local solutions to the correlation functions' two-variable Fourier transform $\mathcal{C}(\omega,\omega')$ which are independent equations for each pair of ω, ω'. Information about the early dynamics can be extracted from high frequencies $\omega \gg 1$ while information about the late-time limit of the system can be extracted from $\omega, \omega' \to 0$ (App. C, D). For example, for the final test loss,

$$\lim_{t\to\infty} \mathcal{L}(t,\alpha,\nu) = \lim_{\omega,\omega'\to 0} (i\omega)(i\omega')\,\mathcal{C}_0(\omega,\omega'). \tag{12}$$

The full temporal trajectory can be obtained with an inverse Fourier transform of $\mathcal{C}_0(\omega,\omega')$. See Appendix A.4.

4. Results

Our results hold for any λ_k and $w_k^\star$ and we provide some simple analytically solvable examples in Appendix I. However, based on empirical observations of NTK spectral decompositions on realistic datasets (Bordelon & Pehlevan, 2022a; Spigler et al., 2020; Bordelon & Pehlevan, 2022a; Bahri et al., 2021; Maloney et al., 2022), here, we focus on the case of power law features. In this setting, eigenvalues and target coefficients decay as a power law in the index k

$$(w_k^\star)^2\lambda_k \sim k^{-a}\ ,\ \lambda_k \sim k^{-b}. \tag{13}$$

We will refer to a as the *task-power* exponent and b as the *spectral decay* exponent[3]. See Figure 7 (a)-(b) for an example with a Residual CNN on CIFAR-5M.

Test loss power laws. For power law features, the test loss will generally be bottlenecked by either training time t (steps of gradient descent), the size of the training set P, or the size of the model N. We can derive bottleneck scalings from our exact expressions for $\mathcal{L}(t,P,N)$ (Appendix J)[4]:

$$\mathcal{L}(t,P,N) \approx \begin{cases} t^{-(a-1)/b}\ ,\ P,N\to\infty\ , \text{ (Time)} \\ P^{-\min\{a-1,2b\}}\ ,\ t,N\to\infty\ , \text{ (Data)} \\ N^{-\min\{a-1,2b\}}\ ,\ t,P\to\infty\ , \text{ (Model)} \end{cases} \tag{14}$$

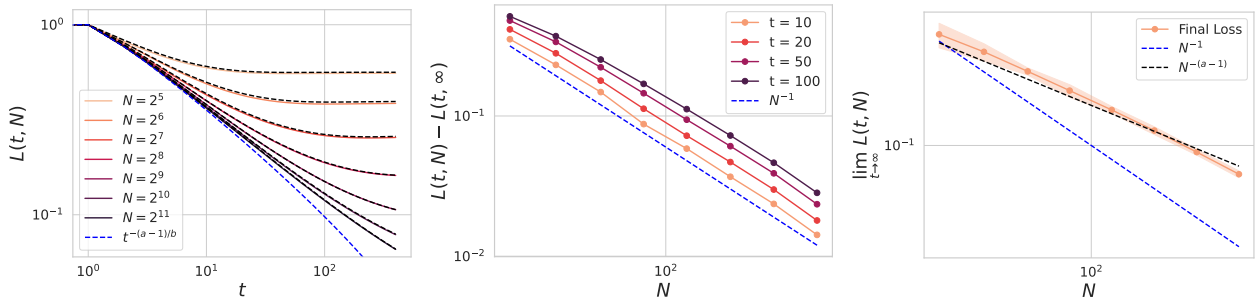
A consequence of this is an *asymmetry in exponent* between the model and data bottlenecks compared to the time bottleneck. We verify this asymmetry in Figure 2.

Bottlenecks as Rank-Constraints All three of the bottleneck scalings arise due to *rank constraints* in the effective dynamics. Heuristically, finite training time or the subsampling of data/features leads to an approximate projection of the target function onto the top $k_\star(t,P,N)$ eigenspace of the infinite-width kernel. The components of the target

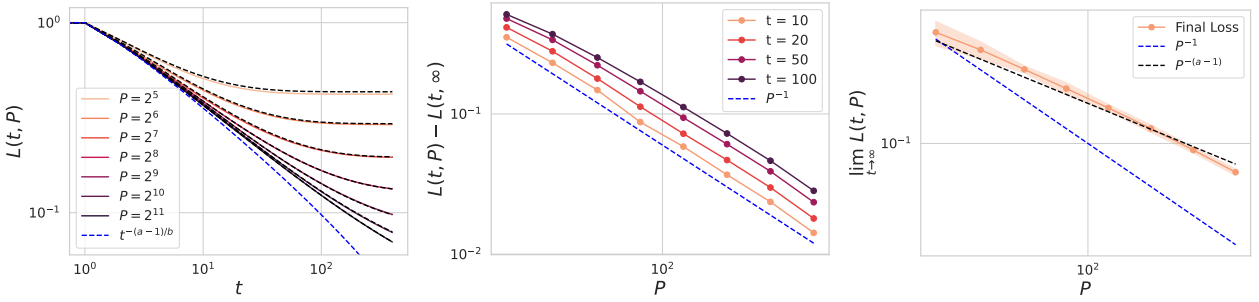
[2] There are dynamical analogues of the mode errors in (Bordelon et al., 2020; Canatar et al., 2021) or learnabilities in (Simon et al., 2021).

[3] These power-law decay rates are also known as source and capacity conditions in the kernel literature (Caponnetto & Vito, 2005; Cui et al., 2021)

[4] The alternative scaling exponents $\mathcal{L} \sim N^{-2b}, P^{-2b}$ occur for very easy tasks which satisfy $a > 2b+1$, but this condition is rarely satisfied in natural data (Appendix J).



(a) $P = 1000$ Test Loss Dynamics (b) Early Time Model Convergence (c) Late Time Model Bottleneck



(d) $N = 1000$ Test Loss Dynamics (e) Early Time Data Convergence (f) Late Time Data Bottleneck

Figure 2. Verification of the various bottleneck scalings for power-law features with $a = 1.5$ and $b = 1.25$. Dashed black lines are DMFT solutions while colors are simulations with standard deviation highlighted. (a) The loss dynamics at large α will be bottlenecked by either time or finite ν. (b) Early in training, the loss converges to its limit as N^{-1} (App. D). (c) At long times, the model's asymptotic loss scales as $N^{-(a-1)}$ (App. C). (d)-(f) The same results but for N and P switched. The model exhibits $1/P$ corrections and early time and power law data bottleneck scalings at late time.

function in the null-space of this projection are not learned. This leads to an approximate test loss of the form

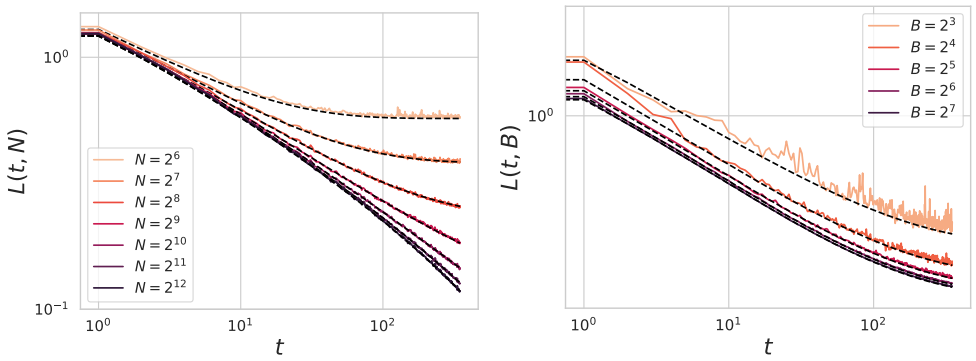
$$\mathcal{L} \approx \sum_{k>k_\star} (w_k^\star)^2 \lambda_k \approx k_\star^{-(a-1)}. \tag{15}$$

For model and data bottlenecks we have that $k_\star \propto N$ and $k_\star \propto P$ respectively (App. J). On the other hand, $k_\star$ for the time bottleneck also depends on the structure of the features through the exponent b. This is because the k-th eigenfeature is learned at a timescale $\tau_k \sim k^b$. At time t, we have learned the first $k_\star \approx t^{1/b}$ modes and the variance in the remaining modes gives $\sim t^{-(a-1)/b}$. In the limit of $t \to \infty$ our data and model bottleneck scalings agree with the resolution and variance-limited scalings studied in (Bahri et al., 2021) as well as prior works on kernels and random feature models (Bordelon et al., 2020; Maloney et al., 2022).

Connection to Online Learning with SGD Many modern deep learning models are trained in an online learning setting where each step of training uses a fresh batch of data to estimate the gradient of the population loss and batches are not reused over multiple steps. Our theoretical methods can also handle this regime. At each step t a fresh minibatch of B examples is used to estimate the gradient. In discrete time with learning rate η this leads to the following DMFT description of $v_k^0(t)$

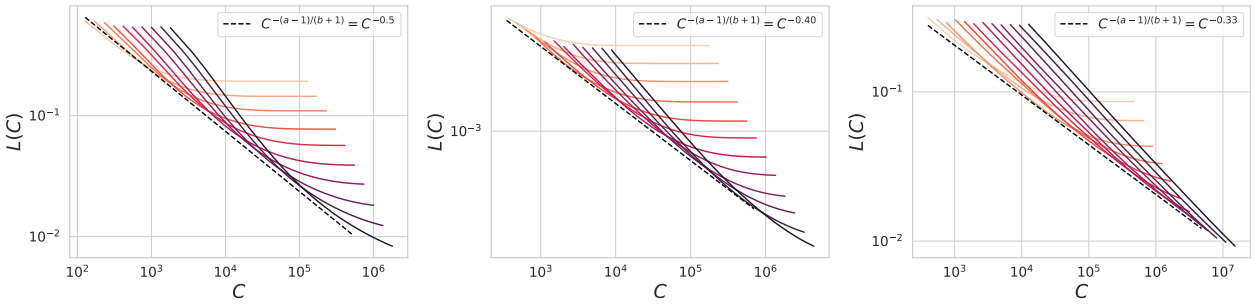
$$v_k^0(t+1) = v_k^0(t) - \eta u_k^4(t) - \eta \sum_{s \le t} R_3(t,s)[u_k^2(s) + \lambda_k v_k^0(s)] \tag{16}$$

where $u_k^4(t), u_k^2(t)$ are zero-mean Gaussian variables with known covariance (see Appendix K.3). The response function $R_3(t,s)$ satisfies a discrete time analog of Equation (11). The most important observation about this regime is that there is no longer a data bottleneck regime. Rather, the bias component of the test error can only be limited by either training time or model size. The finite batch B introduces SGD noise which introduces an additional variance component to the test loss. We illustrate these results in Figure 3. The $N \to \infty$ limit recovers the results of Bordelon & Pehlevan (2022a) which study online SGD without averaging over a random projection. The continuous time limit of the above expressions obtained from evaluating the theory for small η exactly matches the $P \to \infty$ limit of our gradient flow theory presented in the previous section. We



(a) SGD with batchsize $B = 32$ (b) SGD with $N = 256$ and varying batchsize B

Figure 3. Our DMFT can also capture online SGD learning including the effect of batch size fluctuations on the loss and the finite N bottleneck. (a) Power law features trained with SGD and a fixed random projection still generates asymptotes which depend on N. (b) The batchsize B impacts the loss through additional variance in the dynamics but does not lead to an asymptotic plateau.



(a) $(a, b) = (2, 1)$, $L(C) \sim C^{-0.5}$ (b) $(a, b) = (2, 1.5)$, $L(C) \sim C^{-0.4}$ (c) $(a, b) = (2, 2)$, $L(C) \sim C^{-0.33}$

Figure 4. Compute optimal scaling in our model is determined by tradeoff of time and model-size bottlenecks. Solid colored lines are simulations with power law features and in dashed black is the theoretical prediction of compute optimal scaling. Each color represents varying model sizes with $N \in [2^5, 2^{10}]$. The Pareto frontier is defined as the minimum value of L at each compute C over all possible choices of model size N. Although the final losses do not depend on the spectral decay rate b but only on the task-power exponent a, the compute optimal scaling depends does depend on b.

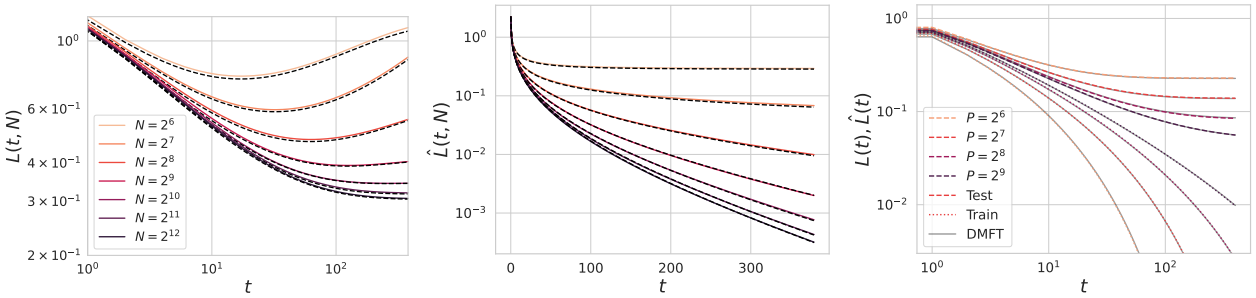
will therefore use this limiting behavior to analyze compute optimal tradeoffs of model size and training time.

Asymmetric Compute Optimal Scaling Strategy We now consider the regime where the total amount of data does not limit performance, but rather training is bottlenecked by time or model size. This could arise in the offline model with very large P or in one-pass SGD with sufficiently small learning rate or sufficiently large batch size (App. K.3). In either case, time and model size should scale differently with compute budget $C = Nt$ and $m = \min\{a - 1, 2b\}$

$$t \sim C^{\frac{bm}{a-1+bm}} \ , \ N \sim C^{\frac{a-1}{a-1+bm}} \ , \implies \mathcal{L}^\star(C) \sim C^{-\frac{(a-1)m}{a-1+bm}} . \tag{17}$$

For the regime of interest where $m = \min\{a - 1, 2b\} = a - 1$ this gives $\mathcal{L}^\star(C) \sim C^{-\frac{a-1}{1+b}}$. We obtain the above scaling by approximating the loss as a sum of the three terms in equation (14) and a constant as in (Hoffmann et al., 2022), see Appendix N. This analysis suggests that for features that have rapid decay in their eigenspectrum, it is preferable to allocate greater resources toward training time rather than model size as the compute budget increases. This is consistent with the small asymmetries observed in (Hoffmann et al., 2022) for language models and the larger asymmetries in MLPs on vision from (Bachmann et al., 2024). In the limit as $b \to 1$, the time and parameter count should be scaled linearly together. We verify this scaling rule and its b-dependence in Figure 4.

Wider is Better Requires Sufficient Data Larger models are not always better in terms of test loss for all time t, as we showed in Figure 1 (c), especially if the dataset is limited. In Figure 5, we illustrate that larger N can improve convergence to a data-bottlenecked loss for power law features. However, the loss may still be non-monotonic in training



(a) Test Loss ($P = 128$) (b) Train Loss ($P = 128$) (c) Train-Test Gaps $N = 512$

Figure 5. In a data limited regime, wider networks train faster but cannot indefinitely improve generalization by making N larger. (a) Test loss for power-law features with $a = 1.5$ and $b = 1.25$ with $P = 128$ and varying N. In this regime, there are diminishing returns to making the model size larger. (b) For $N < P$, the model is underparameterized and cannot achieve zero train loss. For $N > P$, the train loss will eventually decay at exponential rate which depends on N, despite the test loss saturating. (c) The train and test losses gradually separate at a rate which depends on P.

time, motivating regularization or early stopping.

Gradual Buildup of Overfitting Effects The exact gap between train and test losses can exactly be expressed in terms of the DMFT order parameters:

$$\mathcal{L}(t) - \hat{\mathcal{L}}(t) = -\frac{2}{P}\int_0^t dt'\, R_{0,2}(t,t')C_1(t,t') + \frac{1}{P^2}\int_0^t\int_0^t dt'ds' R_{0,2}(t,t')R_{0,2}(t,s')C_1(t',s'). \tag{18}$$

We derive this relation in the Appendix E. At early time this gap goes as $\mathcal{O}(1/P)$ (App. D, E). At late time, however, this picks up a nontrivial task-dependent scaling with P as we show in Figure 2 (e)-(f) and App. C. In Figure 5 (c) we show this gradual accumulation of finite data on the test-train loss gap. For larger datasets P it takes longer training time to begin overfitting (App. E).

Ensembling is Not Always Compute Optimal Ensembling a set of models means averaging their predictions over the same datasets but with different intitialization seeds. This reduces test loss by reducing the variance of the model output f due to initialization. This improvement can be predicted from an extension of our DMFT (App. H). Analogously, bagging over B datasets reduces variance due to sampling of data.

One might imagine that ensembling many finite sized models would allow one to approach the performance of an infinite sized model ($N \to \infty$). If this were possible, the compute optimal strategy could involve a tradeoff between ensemble count and model size. However, recent experiments show that there is a limited benefit from ensembling on large datasets when compared to increasing model size (Vyas et al., 2023). We illustrate this in Figure 6 (a). Our theory can explain these observations as it predicts the effect of ensembling E times on the learning dynamics as we show in App. H. The main reason to prefer increasing N rather than increasing E is that larger N has lower *bias* in the dynamics, whereas ensembling only reduces variance. The bias of the model $\mathcal{B}$ has the form

$$\mathcal{B}(t, N, P) = \sum_k \lambda_k (w_k^\star)^2 H_k(t, N, P)^2, \tag{19}$$

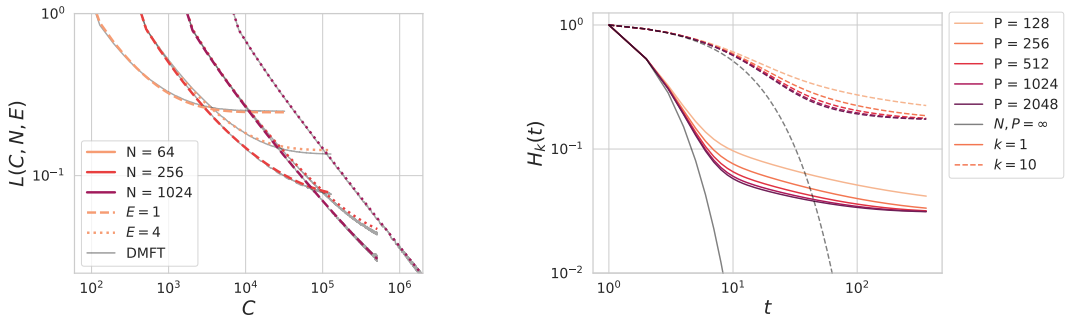
which depend on transfer function H_k that we illustrate for power-law features in Figure 6 (b). Since $H_k(t)$ depend on N, P, we see that ensembling/bagging cannot recover the learning curve of the $N, P \to \infty$ system since the bias is limited by finite N, P.

5. Tests on Realistic Networks

We now move beyond synthetic power-law datasets and consider realistic image datasets and architectures. We take the CIFAR-5M dataset introduced in (Nakkiran et al., 2021a) and consider the task of classfiying animate vs inanimate objects. We plot the spectra of the finite-width NTK at initialization across different widths for a Wide ResNet (Zagoruyko & Komodakis, 2016) in Figure 7 a). Here the width parameter corresponds to the number of channels in the hidden layers. Following (Canatar et al., 2021), we define $C(k)$ as the fraction of the task captured by the top k kernel eigenmodes:

$$C(k) \equiv \frac{\sum_{i \le k} \lambda_i (w_i^*)^2}{\sum_i \lambda_i (w_i^*)^2}. \tag{20}$$

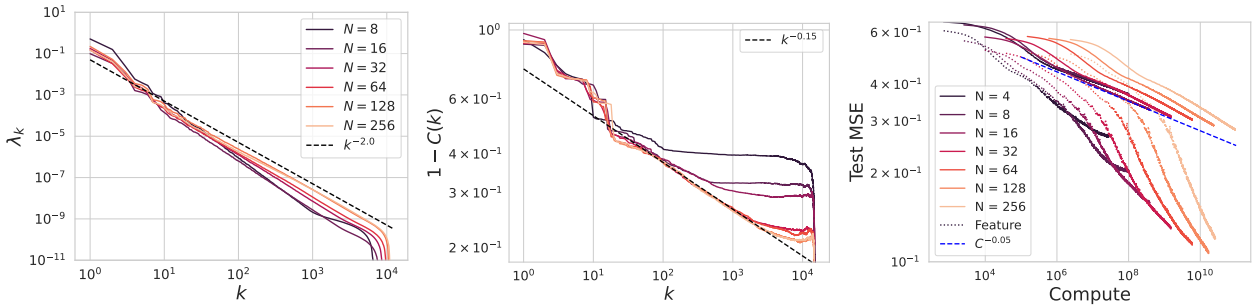
Then $1 - C(k)$ is the portion of the task left unexplained. We plot this for the initial NTKs across widths in Figure 7 b). We extract the spectral decay exponent b and the the task power exponent a from these two curves. Together, these give the learning scaling laws of the linearized neural



(a) E vs N allocation of compute

(b) Transfer Functions $H_k(t)$ for $N = 128$

Figure 6. Ensembling E models of size N improves performance by reducing initialization variance by a factor of E (see Appendix H.2) (a) However, at fixed compute $C = NEt$, increasing the model size N is preferable, since the bias is also reduced. (b) The transfer functions $H_k(t)$ computed from the DMFT determine the error as $E \to \infty$ depend on N, P and saturate in performance at long times, while the $N, P \to \infty$ curves decay exponentially.



(a) NTK Spectra for Varying Widths

(b) Task-Power Decay

(c) Lazy vs Feature Compute Scaling

Figure 7. Our theory predicts time and compute scalings for linearized networks on realistic datasets. (a) The initial NTK spectra and (b) task-power distributions for ResNets of width N on CIFAR-5M are well described by powerlaws $\lambda_k \sim k^{-2.0}$ and $k^{-0.15}$ for large k. (c) The predicted compute optimal scaling of for the ResNet obeys $\mathcal{L}_\star(C) \sim C^{-0.05}$. However, for networks outside of the kernel regime (dashed lines), feature learning can substantially alter the observed scaling laws and improve the loss curves as a function of compute.

network model on this dataset. We plot the compute optimal scaling laws of these linearized models in Figure 7 c). We also plot the predicted scaling law $C^{-(a-1)/(1+b)}$ in blue and find excellent agreement.

5.1. The Role of Feature Learning

We also compare these scalings to those of the compute optimal learning curves for feature-learning networks. We train several networks with different widths and initialization seeds for 64 epochs through the dataset. We observe substantially different compute-optimal scaling exponents in the dotted curves of Figure 7 c). This means that although our random feature model does capture the correct linearized scaling trends, which have all of the qualities observed in realistic scaling laws, more is needed to capture the acceleration of scaling induced by feature learning. Further analyses of the after-kernels of feature learning networks are performed in Appendix L. We see that the kernels continue to evolve substantially throughout training. This indicates that a full explanation of the compute optimal scaling exponents will require something resembling a mechanistic theory of kernel evolution (Long, 2021; Fort et al., 2020; Atanasov et al., 2022; Bordelon & Pehlevan, 2022b).

6. Conclusion

We have presented a model that recovers a wide variety of phenomena observed in more realistic deep learning settings. Our theory includes not just model size and dataset size as parameters but also explicitly treats the temporal dynamics of training. We observe different scaling exponents for performance in terms of model size and number of time steps. Future work to incorporate kernel evolution into this model could further shed insight into the improved scaling laws in the feature-learning regime. Overall, our results provide a theoretical interpretation of compute-optimal scaling as a competition between the training dynamics of the infinite width/infinite data limit and finite model-size bottleneck.

Impact Statement

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

Acknowledgements

We are grateful to Yasaman Bahri, Stefano Mannelli, Francesca Mignacco, Jascha Sohl-Dickstein, and Nikhil Vyas for useful conversations. We thank Clarissa Lauditi and Jacob Zavatone-Veth for comments on the manuscript.

B.B. is supported by a Google PhD Fellowship. A.A. is supported by a Fannie and John Hertz Fellowship. C.P. is supported by NSF grant DMS-2134157, NSF CAREER Award IIS-2239780, and a Sloan Research Fellowship. This work has been made possible in part by a gift from the Chan Zuckerberg Initiative Foundation to establish the Kempner Institute for the Study of Natural and Artificial Intelligence.

References

Adlam, B. and Pennington, J. The neural tangent kernel in high dimensions: Triple descent and a multi-scale theory of generalization. In *International Conference on Machine Learning*, pp. 74–84. PMLR, 2020a.

Adlam, B. and Pennington, J. Understanding double descent requires a fine-grained bias-variance decomposition. *Advances in neural information processing systems*, 33: 11022–11032, 2020b.

Advani, M. S., Saxe, A. M., and Sompolinsky, H. High-dimensional dynamics of generalization error in neural networks. *Neural Networks*, 132:428–446, 2020.

Alabdulmohsin, I., Zhai, X., Kolesnikov, A., and Beyer, L. Getting vit in shape: Scaling laws for compute-optimal model design. *arXiv preprint arXiv:2305.13035*, 2023.

Arora, S. and Goyal, A. A theory for emergence of complex skills in language models. *arXiv preprint arXiv:2307.15936*, 2023.

Atanasov, A., Bordelon, B., and Pehlevan, C. Neural networks as kernel learners: The silent alignment effect. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=1NvflqAdoom.

Atanasov, A., Bordelon, B., Sainathan, S., and Pehlevan, C. The onset of variance-limited behavior for networks in the lazy and rich regimes. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=JLINxPOVTh7.

Bachmann, G., Anagnostidis, S., and Hofmann, T. Scaling mlps: A tale of inductive bias. *Advances in Neural Information Processing Systems*, 36, 2024.

Bahri, Y., Dyer, E., Kaplan, J., Lee, J., and Sharma, U. Explaining neural scaling laws. *arXiv preprint arXiv:2102.06701*, 2021.

Bodin, A. and Macris, N. Model, sample, and epoch-wise descents: exact solution of gradient flow in the random feature model. *Advances in Neural Information Processing Systems*, 34:21605–21617, 2021.

Bordelon, B. and Pehlevan, C. Learning curves for SGD on structured features. In *International Conference on Learning Representations*, 2022a. URL https://openreview.net/forum?id=WPI2vbkAl3Q.

Bordelon, B. and Pehlevan, C. Self-consistent dynamical field theory of kernel evolution in wide neural networks. *arXiv preprint arXiv:2205.09653*, 2022b.

Bordelon, B. and Pehlevan, C. Dynamics of finite width kernel and prediction fluctuations in mean field neural networks. *arXiv preprint arXiv:2304.03408*, 2023.

Bordelon, B., Canatar, A., and Pehlevan, C. Spectrum dependent learning curves in kernel regression and wide neural networks. In *International Conference on Machine Learning*, pp. 1024–1034. PMLR, 2020.

Bordelon, B., Noci, L., Li, M. B., Hanin, B., and Pehlevan, C. Depthwise hyperparameter transfer in residual networks: Dynamics and scaling limit, 2023.

Caballero, E., Gupta, K., Rish, I., and Krueger, D. Broken neural scaling laws. *arXiv preprint arXiv:2210.14891*, 2022.

Canatar, A., Bordelon, B., and Pehlevan, C. Spectral bias and task-model alignment explain generalization in kernel regression and infinitely wide neural networks. *Nature communications*, 12(1):2914, 2021.

Caponnetto, A. and Vito, E. D. Fast rates for regularized least-squares algorithm. 2005.

Cheng, C. and Montanari, A. Dimension free ridge regression. *arXiv preprint arXiv:2210.08571*, 2022.

Chizat, L., Oyallon, E., and Bach, F. On lazy training in differentiable programming. *Advances in neural information processing systems*, 32, 2019.

Cortes, C., Mohri, M., and Rostamizadeh, A. Algorithms for learning kernels based on centered alignment. *The Journal of Machine Learning Research*, 13(1):795–828, 2012.

Crisanti, A. and Sompolinsky, H. Path integral approach to random neural networks. *Physical Review E*, 98(6): 062120, 2018.

Cui, H., Loureiro, B., Krzakala, F., and Zdeborová, L. Generalization error rates in kernel regression: The crossover from the noiseless to noisy regime. *Advances in Neural Information Processing Systems*, 34:10131–10143, 2021.

Dyer, E. and Gur-Ari, G. Asymptotics of wide networks from feynman diagrams. In *International Conference on Learning Representations*, 2020.

d'Ascoli, S., Refinetti, M., Biroli, G., and Krzakala, F. Double trouble in double descent: Bias and variance (s) in the lazy regime. In *International Conference on Machine Learning*, pp. 2280–2290. PMLR, 2020.

Fort, S., Dziugaite, G. K., Paul, M., Kharaghani, S., Roy, D. M., and Ganguli, S. Deep learning versus kernel learning: an empirical study of loss landscape geometry and the time evolution of the neural tangent kernel. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

Geiger, M., Jacot, A., Spigler, S., Gabriel, F., Sagun, L., d'Ascoli, S., Biroli, G., Hongler, C., and Wyart, M. Scaling description of generalization with number of parameters in deep learning. *Journal of Statistical Mechanics: Theory and Experiment*, 2020(2):023401, 2020.

Gerbelot, C., Troiani, E., Mignacco, F., Krzakala, F., and Zdeborova, L. Rigorous dynamical mean field theory for stochastic gradient descent methods. *arXiv preprint arXiv:2210.06591*, 2022.

Ghosh, N., Mei, S., and Yu, B. The three stages of learning dynamics in high-dimensional kernel methods. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=EQmAP4F859.

Hastie, T., Montanari, A., Rosset, S., and Tibshirani, R. J. Surprises in high-dimensional ridgeless least squares interpolation. *Annals of statistics*, 50(2):949, 2022.

Helias, M. and Dahmen, D. *Statistical field theory for neural networks*, volume 970. Springer, 2020.

Hestness, J., Narang, S., Ardalani, N., Diamos, G., Jun, H., Kianinejad, H., Patwary, M. M. A., Yang, Y., and Zhou, Y. Deep learning scaling is predictable, empirically. *arXiv preprint arXiv:1712.00409*, 2017.

Hoffmann, J., Borgeaud, S., Mensch, A., Buchatskaya, E., Cai, T., Rutherford, E., Casas, D. d. L., Hendricks, L. A., Welbl, J., Clark, A., et al. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*, 2022.

Jacot, A., Gabriel, F., and Hongler, C. Neural tangent kernel: Convergence and generalization in neural networks. *Advances in neural information processing systems*, 31, 2018.

Kaplan, J., McCandlish, S., Henighan, T., Brown, T. B., Chess, B., Child, R., Gray, S., Radford, A., Wu, J., and Amodei, D. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

Long, P. M. Properties of the after kernel. *arXiv preprint arXiv:2105.10585*, 2021.

Loureiro, B., Gerbelot, C., Cui, H., Goldt, S., Krzakala, F., Mezard, M., and Zdeborová, L. Learning curves of generic features maps for realistic datasets with a teacher-student model. *Advances in Neural Information Processing Systems*, 34:18137–18151, 2021.

Loureiro, B., Gerbelot, C., Refinetti, M., Sicuro, G., and Krzakala, F. Fluctuations, bias, variance & ensemble of learners: Exact asymptotics for convex losses in high-dimension. In *International Conference on Machine Learning*, pp. 14283–14314. PMLR, 2022.

Maloney, A., Roberts, D. A., and Sully, J. A solvable model of neural scaling laws. *arXiv preprint arXiv:2210.16859*, 2022.

Mannelli, S. S., Krzakala, F., Urbani, P., and Zdeborova, L. Passed & spurious: Descent algorithms and local minima in spiked matrix-tensor models. In *international conference on machine learning*, pp. 4333–4342. PMLR, 2019.

Martin, P. C., Siggia, E., and Rose, H. Statistical dynamics of classical systems. *Physical Review A*, 8(1):423, 1973.

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

Michaud, E. J., Liu, Z., Girit, U., and Tegmark, M. The quantization model of neural scaling. *arXiv preprint arXiv:2303.13506*, 2023.

Mignacco, F. and Urbani, P. The effective noise of stochastic gradient descent. *Journal of Statistical Mechanics: Theory and Experiment*, 2022(8):083405, 2022.

Mignacco, F., Krzakala, F., Urbani, P., and Zdeborová, L. Dynamical mean-field theory for stochastic gradient descent in gaussian mixture classification. *Advances in Neural Information Processing Systems*, 33:9540–9550, 2020.

Muennighoff, N., Rush, A. M., Barak, B., Scao, T. L., Piktus, A., Tazi, N., Pyysalo, S., Wolf, T., and Raffel, C. Scaling data-constrained language models. *arXiv preprint arXiv:2305.16264*, 2023.

Nakkiran, P., Neyshabur, B., and Sedghi, H. The deep bootstrap framework: Good online learners are good offline generalizers. In *International Conference on Learning Representations*, 2021a.

Nakkiran, P., Neyshabur, B., and Sedghi, H. The deep bootstrap framework: Good online learners are good offline generalizers. In *International Conference on Learning Representations*, 2021b. URL https://openreview.net/forum?id=guetrIHLFGI.

Paquette, C., Lee, K., Pedregosa, F., and Paquette, E. Sgd in the large: Average-case analysis, asymptotics, and stepsize criticality. In *Conference on Learning Theory*, pp. 3548–3626. PMLR, 2021.

Roberts, D. A., Yaida, S., and Hanin, B. *The principles of deep learning theory*. Cambridge University Press Cambridge, MA, USA, 2022.

Ruben, B. S. and Pehlevan, C. Learning curves for noisy heterogeneous feature-subsampled ridge ensembles. *ArXiv*, 2023.

Simon, J. B., Dickens, M., Karkada, D., and DeWeese, M. R. The eigenlearning framework: A conservation law perspective on kernel regression and wide neural networks. *arXiv preprint arXiv:2110.03922*, 2021.

Simon, J. B., Karkada, D., Ghosh, N., and Belkin, M. More is better in modern machine learning: when infinite overparameterization is optimal and overfitting is obligatory. *arXiv preprint arXiv:2311.14646*, 2023.

Sompolinsky, H. and Zippelius, A. Dynamic theory of the spin-glass phase. *Physical Review Letters*, 47(5):359, 1981.

Sorscher, B., Geirhos, R., Shekhar, S., Ganguli, S., and Morcos, A. Beyond neural scaling laws: beating power law scaling via data pruning. *Advances in Neural Information Processing Systems*, 35:19523–19536, 2022.

Spigler, S., Geiger, M., and Wyart, M. Asymptotic learning curves of kernel methods: empirical data versus teacher–student paradigm. *Journal of Statistical Mechanics: Theory and Experiment*, 2020(12):124001, 2020.

Varre, A. V., Pillaud-Vivien, L., and Flammarion, N. Last iterate convergence of sgd for least-squares in the interpolation regime. *Advances in Neural Information Processing Systems*, 34:21581–21591, 2021.

Vyas, N., Bansal, Y., and Nakkiran, P. Limitations of the ntk for understanding generalization in deep learning. *arXiv preprint arXiv:2206.10012*, 2022.

Vyas, N., Atanasov, A., Bordelon, B., Morwani, D., Sainathan, S., and Pehlevan, C. Feature-learning networks are consistent across widths at realistic scales. *arXiv preprint arXiv:2305.18411*, 2023.

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

Zagoruyko, S. and Komodakis, N. Wide residual networks. *arXiv preprint arXiv:1605.07146*, 2016.

Zavatone-Veth, J. A. and Pehlevan, C. Learning curves for deep structured gaussian feature models, 2023.

Zavatone-Veth, J. A., Tong, W. L., and Pehlevan, C. Contrasting random and learned features in deep bayesian linear regression. *Physical Review E*, 105(6):064118, 2022.

A. Derivation of Dynamical Model of Scaling Laws

We investigate the simplest possible model which can exhibit task-dependent time, model size and finite data bottlenecks. We therefore choose to study a linear model with projected features

$$f(\boldsymbol{x}) = \frac{1}{\sqrt{N}} \boldsymbol{w}^\top \left(\frac{1}{\sqrt{M}} \boldsymbol{A} \boldsymbol{\psi}(\boldsymbol{x}) \right) \ , \ y(\boldsymbol{x}) = \frac{1}{\sqrt{M}} \boldsymbol{w}_\star \cdot \boldsymbol{\psi}(\boldsymbol{x}). \tag{21}$$

The weights $\boldsymbol{w}$ are updated with gradient descent on a random training dataset which has (possibly) noise corrupted target values $y_\mu = y(\boldsymbol{x}_\mu) + \sigma \epsilon_\mu$. This leads to the following gradient flow dynamics

$$\frac{\partial}{\partial t} \boldsymbol{w}(t) = \frac{\sqrt{M}}{P\sqrt{N}} \sum_{\mu=1}^{P} (y_\mu - f_\mu) \boldsymbol{A} \boldsymbol{\psi}_\mu = \frac{1}{\sqrt{N}} \boldsymbol{A} \left(\frac{1}{P} \sum_{\mu=1}^{P} \boldsymbol{\psi}_\mu \left[\boldsymbol{\psi}_\mu^\top \left(\boldsymbol{w}_\star - \frac{1}{\sqrt{N}} \boldsymbol{A}^\top \boldsymbol{w} \right) + \sqrt{M} \sigma \epsilon_\mu \right] \right). \tag{22}$$

We introduce the variable $\boldsymbol{v}^0 = \boldsymbol{w}_\star - \frac{1}{\sqrt{N}} \boldsymbol{A}^\top \boldsymbol{w}$ to represent the residual error of the learned weight vector. This residual error has the following dynamics:

$$\partial_t \boldsymbol{v}^0(t) = - \left(\frac{1}{N} \boldsymbol{A}^\top \boldsymbol{A} \right) \left[\left(\frac{1}{P} \boldsymbol{\Psi}^\top \boldsymbol{\Psi} \right) \boldsymbol{v}^0(t) + \frac{\sigma}{\alpha \sqrt{M}} \boldsymbol{\Psi}^\top \boldsymbol{\epsilon} \right]. \tag{23}$$

The entries of each matrix are treated as random with $\Psi_k^\mu \sim \mathcal{N}(0, \lambda_k)$ and $A_{jk} \sim \mathcal{N}(0, 1)$. To study the dynamical evolution of the test error $\mathcal{L}(t) = \frac{1}{M} \boldsymbol{v}^0(t)^\top \boldsymbol{\Lambda} \boldsymbol{v}^0(t) + \sigma^2$, we introduce the sequence of vectors

$$\begin{aligned} \boldsymbol{v}^1(t) &= \frac{1}{\sqrt{M}} \boldsymbol{\Psi} \boldsymbol{v}^0(t) + \sigma \boldsymbol{\epsilon} \ , \ \boldsymbol{v}^2(t) = \frac{1}{\alpha \sqrt{M}} \boldsymbol{\Psi}^\top \boldsymbol{v}^1(t) \\ \boldsymbol{v}^3(t) &= \frac{1}{\sqrt{M}} \boldsymbol{A} \boldsymbol{v}^2(t) \ , \ \boldsymbol{v}^4(t) = \frac{1}{\nu \sqrt{M}} \boldsymbol{A}^\top \boldsymbol{v}^3(t). \end{aligned} \tag{24}$$

The train and test losses can be computed from the $\boldsymbol{v}^0$ and $\boldsymbol{v}^1$ fields

$$\hat{\mathcal{L}}(t) = \frac{1}{P} \sum_{\mu=1}^{P} v_\mu^1(t)^2 \ , \ \mathcal{L}(t) = \frac{1}{M} \sum_{k=1}^{M} \lambda_k v_k^0(t)^2 + \sigma^2. \tag{25}$$

In the next section, we derive a statistical description of the dynamics in an appropriate asymptotic limit using dynamical mean field theory methods.

A.1. DMFT Equations for the Asymptotic Limit

Standard field theoretic arguments such as the cavity or path integral methods can be used to compute the effective statistical description of the dynamics in the limit of large M, N, P with fixed ratios $\alpha = P/M$ and $\nu = \frac{N}{M}$ (see Appendix B). This computation gives us the following statistical description of the dynamics.

$$\begin{aligned} v^1(t) &= u^1(t) + \frac{1}{\alpha} \int ds R_{0,2}(t,s) v_1(s) + \sigma \epsilon \ , \ u^1(t) \sim \mathcal{GP}(0, C_0) \ , \ \epsilon \sim \mathcal{N}(0,1), \\ v_k^2(t) &= u_k^2(t) + \lambda_k \int ds R_1(t,s) v_k^0(s) \ , \ u_k^2(t) \sim \mathcal{GP}\left(0, \frac{1}{\alpha} \lambda_k C_1\right), \\ v^3(t) &= u^3(t) + \frac{1}{\nu} \int ds R_{2,4}(t,s) v^3(s) \ , \ u^3(t) \sim \mathcal{GP}\left(0, C_2\right), \\ v_k^4(t) &= u_k^4(t) + \int ds R_3(t,s) v_k^2(s) \ , \ u_k^4(t) \sim \mathcal{GP}\left(0, \frac{1}{\nu} C_3\right), \\ \partial_t v_k^0(t) &= -v_k^4(t). \end{aligned} \tag{26}$$

The correlation and response functions obey

$$C_0(t,s) = \frac{1}{M}\sum_k \lambda_k \left\langle v_k^0(t) v_k^0(s) \right\rangle \ , \ C_1(t,s) = \left\langle v^1(t) v^1(s) \right\rangle \ , \ C_2(t,s) = \frac{1}{M}\sum_{k=1}^M \left\langle v_k^2(t) v_k^2(s) \right\rangle$$
$$R_{0,2}(t,s) = \frac{1}{M}\sum_k \lambda_k \left\langle \frac{\delta v_k^0(t)}{\delta u_k^2(s)} \right\rangle \ , \ R_{2,4}(t,s) = \frac{1}{M}\sum_k \left\langle \frac{\delta v_k^2(t)}{\delta u_k^4(s)} \right\rangle$$
$$R_1(t,s) = \left\langle \frac{\delta v^1(t)}{\delta u^1(s)} \right\rangle \ , \ R_3(t,s) = \left\langle \frac{\delta v^3(t)}{\delta u^3(s)} \right\rangle$$

These equations are exact in the joint proportional limit for any value of α, ν.

A.2. Closing the Equations for the Order Parameters

Though we expressed the dynamics in terms of random fields, we stress in this section that all of the dynamics for the correlation and response functions close in terms of integro-differential equations. To shorten the expression, we will provide the expression for $\beta = 0$, but momentum can easily be added back by making the substitution $\partial_t \to \partial_t + \beta \partial_t^2$.

First, our closed integral equations for the response functions are

$$\begin{aligned}
R_{0,2,k}(t,s) &= -\int dt' \Theta(t-t') R_3(t',s) - \lambda_k \int dt' dt'' dt''' \Theta(t-t') R_3(t',t'') R_1(t'',t''') R_{0,2,k}(t''',s) \\
R_1(t,s) &= \delta(t-s) + \frac{1}{\alpha}\int dt' R_{0,2}(t,t') R_1(t',s) \\
R_{2,4,k}(t,s) &= -\lambda_k \int dt' dt'' R_1(t,t') \Theta(t'-t'') - \lambda_k \int dt' dt'' dt''' R_1(t,t') \Theta(t'-t'') R_3(t'',t''') R_{2,4,k}(t''',s) \\
R_3(t,s) &= \delta(t-s) + \frac{1}{\nu}\int dt' R_{2,4}(t,t') R_3(t',s) \\
R_{0,2}(t,s) &= \frac{1}{M}\sum_k \lambda_k R_{0,2,k}(t,s) \ , \ R_{2,4}(t,s) = \frac{1}{M}\sum_k R_{2,4,k}(t,s)
\end{aligned} \tag{27}$$

We note that these equations imply causality in all of the response functions since $R(t,s) = 0$ for $t < s$. Once these equations are solved for the response functions, we can determine the correlation functions, which satisfy

$$\begin{aligned}
\partial_{ts}^2 C_{0,k}(t,s) =& -\lambda_k \int dt' dt'' R_3(t,t') R_1(t',t'') \partial_s C_{0,k}(t'',s) \\
& -\lambda_k \int ds' R_3(s,s') R_1(s',s'') \partial_t C_{0,k}(t,s'') \\
& +\lambda_k^2 \int dt' dt'' ds' ds'' R_3(t,t') R_1(t',t'') R_3(s,s') R_1(s',s'') C_{0,k}(t'',s'') \\
& -(w_k^\star)^2 \delta(t)\delta(s) - \frac{1}{\nu} C_3(t,s) - \frac{1}{\alpha}\int dt' ds' R_3(t,t') R_3(s,s') C_1(t',s') \\
C_1(t,s) =& \int dt' R_1(t,t') R_1(s,s') C_0(t',s') \\
C_{2,k}(t,s) =& -\lambda_k \int dt' dt'' R_1(t,t') R_3(t',t'') C_{2,k}(t'',s) - \lambda_k \int ds' ds'' R_1(s,s') R_3(s',s'') C_2(t,s'') \\
& +\lambda_k^2 \int dt' dt'' ds' ds'' R_1(t,t') R_3(t',t'') R_1(s,s') R_3(s',s'') C_{2,k}(t'',s'') \\
C_3(t,s) =& \int dt' ds' R_3(t,t') R_3(s,s') C_2(t',s')
\end{aligned} \tag{28}$$

Solving these closed equations provide the complete statistical characterization of the limit. The test and train losses are given by the time-time diagonal of $C_0(t,t), C_1(t,t)$.

A.3. Time-translation Invariant (TTI) Solution to Response Functions

From the structure of the above equations, the response functions are time-translation invariant (TTI) since they are only functionals of TTI $\delta(t-s)$ Dirac-Delta function and $\Theta(t-s)$ Heaviside step-function. As a consequence, we write each of our response functions in terms of their Fourier transforms

$$R(t,s) = R(t-s) = \int_{-\infty}^{\infty} \frac{d\omega}{2\pi}\, e^{i\omega(t-s)}\mathcal{R}(\omega). \tag{29}$$

Using the fact that

$$\delta(\tau) = \int \frac{d\omega}{2\pi} e^{i\omega\tau}\ ,\ \Theta(\tau) = \lim_{\epsilon\to 0^+} e^{-\epsilon\tau}\Theta(\tau) = \lim_{\epsilon\to 0^+}\int \frac{d\omega}{2\pi}\frac{e^{i\omega\tau}}{\epsilon + i\omega} \tag{30}$$

We will keep track of the regulator ϵ and consider $\epsilon \to 0^+$ at the end of the computation. The resulting DMFT equations for the response functions have the following form in Fourier space

$$\begin{aligned}
\mathcal{R}_1(\omega) &= 1 + \frac{1}{\alpha}\mathcal{R}_{2,4}(\omega)\mathcal{R}_1(\omega)\\
\mathcal{R}_3(\omega) &= 1 + \frac{1}{\nu}\mathcal{R}_{2,4}(\omega)\mathcal{R}_3(\omega)\\
\mathcal{R}_{0,2}(\omega) &= -\frac{1}{M}\sum_k \frac{\lambda_k}{\epsilon + i\omega + \lambda_k\mathcal{R}_1(\omega)\mathcal{R}_3(\omega)}\mathcal{R}_3(\omega)\\
\mathcal{R}_{2,4}(\omega) &= -\frac{1}{M}\sum_k \frac{\lambda_k}{\epsilon + i\omega + \lambda_k\mathcal{R}_1(\omega)\mathcal{R}_3(\omega)}\mathcal{R}_1(\omega)
\end{aligned} \tag{31}$$

where $\epsilon \to 0$ will be taken after. Combining these equations, we arrive at the simple set of coupled equations

$$\begin{aligned}
\mathcal{R}_1(\omega) &= 1 - \frac{1}{P}\sum_k \frac{\lambda_k\mathcal{R}_3(\omega)\mathcal{R}_1(\omega)}{\epsilon + i\omega + \lambda_k\mathcal{R}_1(\omega)\mathcal{R}_3(\omega)}\\
\mathcal{R}_3(\omega) &= 1 - \frac{1}{N}\sum_k \frac{\lambda_k\mathcal{R}_1(\omega)\mathcal{R}_3(\omega)}{\epsilon + i\omega + \lambda_k\mathcal{R}_1(\omega)\mathcal{R}_3(\omega)}
\end{aligned} \tag{32}$$

After solving these equations for all ω, we can invert the dynamics of $v_k^0(t)$ to obtain its Fourier transform

$$\begin{aligned}
v_k^0(\omega) &= \mathcal{H}_k(\omega)\left[w_k^\star - u_k^4(\omega) - \lambda_k\mathcal{R}_3(\omega)u_k^2(\omega)\right]\\
\mathcal{H}_k(\omega) &\equiv \frac{1}{\epsilon + i\omega + \lambda_k\mathcal{R}_1(\omega)\mathcal{R}_3(\omega)}
\end{aligned} \tag{33}$$

where we defined the transfer functions $\mathcal{H}_k(\omega)$. From this equation, we can compute $v_k^0(t)$ through inverse Fourier-transformation and then compute the correlation function to calculate the test error. An interesting observation is that the response functions $\mathcal{R}_1(\omega), \mathcal{R}_3(\omega)$ alter the pole structure in the transfer function, generating ν, α dependent timescales of convergence.

A.4. Fourier Representations for Correlation Functions

While the response functions are TTI, the correlation functions transparently are not (if the time-time diagonal $C_0(t,t)$ did not evolve, then the loss $\mathcal{L}(t)$ wouldn't change!). We therefore define the need to define the double Fourier transform $\mathcal{C}(\omega,\omega')$ for each correlation function $C(t,s)$

$$\mathcal{C}(\omega,\omega') = \int dtds\, e^{-i\omega t - i\omega' s}C(t,s)\ ,\ C(t,s) = \int \frac{d\omega}{2\pi}\frac{d\omega'}{2\pi}e^{i\omega t + i\omega' s}\mathcal{C}(\omega,\omega') \tag{34}$$

Assuming that all response functions and transfer functions $\mathcal{H}_k$ have been solved for, the correlation functions satisfy the closed set of linear equations.

$$\begin{aligned}
\mathcal{C}_0(\omega,\omega') &= \frac{1}{M}\sum_k \lambda_k \mathcal{H}_k(\omega)\mathcal{H}_k(\omega')\left[(w_k^\star)^2 + \frac{1}{\nu}\mathcal{C}_3(\omega,\omega') + \frac{1}{\alpha}\lambda_k \mathcal{R}_3(\omega)\mathcal{R}_3(\omega')\mathcal{C}_1(\omega,\omega')\right] \\
\mathcal{C}_1(\omega,\omega') &= \mathcal{R}_1(\omega)\mathcal{R}_1(\omega')\mathcal{C}_0(\omega,\omega') \\
\mathcal{C}_2(\omega,\omega') &= \frac{1}{M}\sum_k \lambda_k \mathcal{H}_k(\omega)\mathcal{H}_k(\omega')\left[\frac{1}{\alpha}(i\omega)(i\omega')\mathcal{C}_1(\omega,\omega') + \lambda_k \mathcal{R}_1(\omega)\mathcal{R}_1(\omega')\left((w_k^\star)^2 + \frac{1}{\nu}\mathcal{C}_3(\omega,\omega')\right)\right] \\
\mathcal{C}_3(\omega,\omega') &= \mathcal{R}_3(\omega)\mathcal{R}_3(\omega')\mathcal{C}_2(\omega,\omega')
\end{aligned} \tag{35}$$

These equations can be efficiently solved for all pairs of ω, ω' after the response functions have been identified. Then one can take an inverse Fourier transform in both indices.

B. Field Theoretic Derivation of DMFT Equations

In this section, we derive the field theoretic description of our model. We will derive this using both the Martin-Siggia-Rose (MSR) path integral method (Martin et al., 1973) and the dynamical cavity method. For a recent review of these topics in the context of neural networks, see (Helias & Dahmen, 2020).

B.1. Statistical Assumptions for DMFT

The DMFT that we derive in the next few sections requires some assumptions on the structure of $\boldsymbol{A}$ and $\boldsymbol{\Psi}$. To carry out the classic MSR path integral computation, we assume that the entries of both matrices are Gaussian with mean zero and covariance

$$\langle A_{ij} A_{kl} \rangle = \delta_{ik}\delta_{jl} \ , \ \langle \Psi_{\mu k} \Psi_{\nu l} \rangle = \delta_{\mu\nu}\delta_{kl}\lambda_k. \tag{36}$$

These are sufficient conditions for the DMFT description to hold and we will take them as our primary assumptions. However, we note that these restrictions are not strictly necessary and can be relaxed. In general, a more flexible cavity derivation in Appendix B.3 shows that independent entries from any well behaved distribution which admits a central limit theorem for sums of independent draws would also have the same DMFT description of the proportional limit. Prior works on DMFT of M-estimators with random data have demonstrated universality for any data matrix $\boldsymbol{\Psi}$ with a covariance that has bounded spectral norm (Gerbelot et al., 2022).

B.2. Path Integral Derivation

With the MSR formalism, we evaluate the moment generating functional for the field $\boldsymbol{v}^0(t)$:

$$Z[\{\boldsymbol{j}(t)\}] = \left\langle \int \mathcal{D}\boldsymbol{v}^0(t)\, \delta\left(\dot{\boldsymbol{v}}^0(t) + \frac{1}{NP}\boldsymbol{A}^\top \boldsymbol{A}\boldsymbol{\Psi}^\top \boldsymbol{\Psi}\boldsymbol{v}^0(t)\right) \exp\left(\int dt\, \boldsymbol{j}(t)\cdot \boldsymbol{v}^0(t)\right)\right\rangle_{\boldsymbol{A},\boldsymbol{\Psi}}. \tag{37}$$

Note that at zero source, we have the important identity that

$$Z[0] = 1. \tag{38}$$

We insert a Dirac delta functions to enforce the definitions of each of the fields $\{\boldsymbol{v}^1, \boldsymbol{v}^2, \boldsymbol{v}^3, \boldsymbol{v}^4\}$ as in equation 8.

$$\begin{aligned}
Z[\{\boldsymbol{j}(t)\}] = \int \mathcal{D}[\boldsymbol{v}^0, \dots, \boldsymbol{v}^4, \hat{\boldsymbol{v}}^1 \dots \hat{\boldsymbol{v}}^4]\, \delta(\dot{\boldsymbol{v}}^0 + \boldsymbol{v}^4) \exp\left(\int dt\, \boldsymbol{j}(t)\cdot \boldsymbol{v}^0(t)\right) \\
\times \left\langle \exp\left[i \int dt \left[\hat{\boldsymbol{v}}_1(t)\cdot\left(\boldsymbol{v}^1(t) - \frac{1}{\sqrt{M}}\boldsymbol{\Psi}\boldsymbol{v}^0(t)\right) + \hat{\boldsymbol{v}}_2(t)\cdot\left(\boldsymbol{v}^2(t) - \frac{1}{\alpha\sqrt{M}}\boldsymbol{\Psi}^\top \boldsymbol{v}^1(t)\right)\right]\right]\right\rangle_{\boldsymbol{\Psi}} \\
\times \left\langle \exp\left[i \int dt \left[\hat{\boldsymbol{v}}_3(t)\cdot\left(\boldsymbol{v}^3(t) - \frac{1}{\sqrt{M}}\boldsymbol{A}\boldsymbol{v}^2(t)\right) + \hat{\boldsymbol{v}}_4(t)\cdot\left(\boldsymbol{v}^4(t) - \frac{1}{\nu\sqrt{M}}\boldsymbol{A}^\top \boldsymbol{v}^3(t)\right)\right]\right]\right\rangle_{\boldsymbol{A}}.
\end{aligned} \tag{39}$$

At this stage we can add sources $\tilde{\boldsymbol{j}}$ for each $\hat{\boldsymbol{v}}_i$ variable, yielding a $Z[\boldsymbol{j}(t), \tilde{\boldsymbol{j}}(t)]$. Interpreting each source as modification of the respective evolution equation, we see that this modified moment-generating function remains equal to unity at any value of $\tilde{\boldsymbol{j}}$, $Z[0, \tilde{\boldsymbol{j}}(t)] = 1$. As a consequence, all correlation functions consisting only of $\hat{\boldsymbol{v}}^i$ variables vanish. See (Crisanti & Sompolinsky, 2018) for further details and a worked example.

We now average over the sources of disorder. We assume that the entries of $\boldsymbol{A}$ are i.i.d. with mean zero and variance 1. In the proportional limit, we can replace the entries of $\boldsymbol{A}$ as a draw from a Gaussian $\mathcal{N}(0, 1)$ by appealing to Gaussian equivalence. We furhther justify this in the cavity derivation in the next section. This allows us to evaluate the averages over the matrix $\boldsymbol{A}$.

$$\begin{aligned}
&\left\langle \exp\left(-\frac{i}{\sqrt{M}} \mathrm{Tr} \boldsymbol{A}^\top \int dt \left[\hat{\boldsymbol{v}}^3(t) \boldsymbol{v}^2(t)^\top + \nu^{-1} \boldsymbol{v}^3(t) \hat{\boldsymbol{v}}^4(t)^\top \right] \right) \right\rangle_{\boldsymbol{A}} \\
&= \exp\left(-\frac{1}{2} \int dt ds [\hat{\boldsymbol{v}}^3(t) \cdot \hat{\boldsymbol{v}}^3(s) \underbrace{\frac{1}{M} \boldsymbol{v}^2(t) \cdot \boldsymbol{v}^2(s)}_{C_2(t,s)} + \nu^{-1} \hat{\boldsymbol{v}}^4(t) \cdot \hat{\boldsymbol{v}}^4(s) \underbrace{\frac{1}{N} \boldsymbol{v}^3(t) \cdot \boldsymbol{v}^3(s)}_{C_3(t,s)}] \right) \\
&\quad \times \exp\left(- \int dt ds \underbrace{\frac{1}{N} \hat{\boldsymbol{v}}^3(t) \cdot \boldsymbol{v}^3(s)}_{i R_3(s,t)} \boldsymbol{v}^2(t) \cdot \hat{\boldsymbol{v}}^4(s) \right).
\end{aligned} \tag{40}$$

Similarly, we can calculate the averages over the data, which enters via the design matrices $\boldsymbol{\Psi}$. Again in this proportional limit we can invoke Gaussian equivalence on $\boldsymbol{\Psi}$ to have it take the form $\boldsymbol{\Psi} \sim \boldsymbol{\Phi} \boldsymbol{\Lambda}^{1/2}$ where $\boldsymbol{\Phi}$ has entries drawn from a unit normal. Taking the average then gives us

$$\begin{aligned}
&\left\langle \exp\left(-\frac{i}{\sqrt{M}} \mathrm{Tr}\, \boldsymbol{\Psi}^\top \int dt \left[\hat{\boldsymbol{v}}^1(t) \boldsymbol{v}^0(t)^\top + \alpha^{-1} \boldsymbol{v}^1(t) \hat{\boldsymbol{v}}^2(t)^\top \right] \right) \right\rangle_{\boldsymbol{\Psi}} \\
&= \exp\left(-\frac{1}{2} \int dt ds [\hat{\boldsymbol{v}}^1(t) \cdot \hat{\boldsymbol{v}}^1(s) \underbrace{\frac{1}{M} \boldsymbol{v}^0(t) \cdot \boldsymbol{\Lambda} \boldsymbol{v}^0(s)}_{C_0(t,s)} + \alpha^{-1} \hat{\boldsymbol{v}}^2(t) \cdot \boldsymbol{\Lambda} \hat{\boldsymbol{v}}^2(s) \underbrace{\frac{1}{P} \boldsymbol{v}^1(t) \cdot \boldsymbol{v}^1(s)}_{C_1(t,s)}] \right), \\
&\quad \times \exp\left(- \int dt ds \underbrace{\frac{1}{P} \hat{\boldsymbol{v}}^1(t) \cdot \boldsymbol{v}^1(s)}_{i R_1(s,t)} \boldsymbol{v}^0(t) \cdot \boldsymbol{\Lambda} \cdot \hat{\boldsymbol{v}}^2(s) \right).
\end{aligned} \tag{41}$$

We now insert delta functions for following bracketed terms: C_0, C_1, C_2, C_3 and R_1, R_3 using the following identity (e.g. for C_0 at times s, t):

$$1 = \int \frac{dC_0(s,t) d\hat{C}_0(s,t)}{4\pi i M^{-1}} \exp\left[\frac{1}{2} M \int dt ds\, \hat{C}_0(t,s) \left(C_0(t,s) - \frac{1}{M} \boldsymbol{v}^0(t) \cdot \boldsymbol{\Lambda} \boldsymbol{v}^0(s) \right) \right]. \tag{42}$$

Here the $\hat{C}_i, \hat{R}_i$ integrals are taken over the imaginary axis. This yields a moment generating function (here we'll take $\boldsymbol{j} = 0$):

$$Z = \int \mathcal{D}[C_1, \hat{C}_1, \dots] \exp\left[M S[C_0, C_1, C_2, C_3, R_1, R_3, \hat{C}_0, \hat{C}_1, \hat{C}_2, \hat{C}_3, \hat{R}_1, \hat{R}_3] \right]. \tag{43}$$

The constraint that $Z = 1$ means that $S = 0$ at the saddle point. S here is given by:

$$\begin{aligned}
S[\dots] &= \frac{1}{2} \int dt ds \Big[\hat{C}_0(t,s) C_0(t,s) + \alpha \hat{C}_1(t,s) C_1(t,s) + \hat{C}_2(t,s) C_2(t,s) + \nu \hat{C}_3(t,s) C_3(t,s) \Big] \\
&\quad + \int dt ds \Big[- R_1(t,s) \hat{R}_1(s,t) - R_3(t,s) \hat{R}_3(s,t) \Big] \\
&\quad + \alpha \log \mathcal{Z}_1 + \nu \log \mathcal{Z}_3 + \frac{1}{M} \sum_k \log \mathcal{Z}_{0,2,4;k}.
\end{aligned} \tag{44}$$

We have chosen to take $\hat{R}_i(s,t)$ to have a different sign and s,t ordering convention than the $\hat{C}_i$ to simplify our notation later on. We have also used that Equations (40), (41) factorize over their respective indices, so each $\mathcal{Z}$ is a partition function over a single index. The individual $\mathcal{Z}_i$ are given by:

$$\begin{aligned}\mathcal{Z}_1 = \int \mathcal{D}[v^1,\hat{v}^1] \exp\left[i \int dtds \left(\delta(t-s) - \alpha^{-1}\hat{R}_1(s,t)\right) v^1(t)\hat{v}^1(s)\right] \\ \times \exp\left[-\frac{1}{2}\int dtds(\hat{v}^1(t)\hat{v}^1(s)C_0(t,s) + v^1(t)v^1(s)\hat{C}_1(t,s))\right],\end{aligned} \tag{45}$$

$$\begin{aligned}\mathcal{Z}_3 = \int \mathcal{D}[v^3,\hat{v}^3] \exp\left[i \int dtds \left(\delta(t-s) - \nu^{-1}\hat{R}_3(s,t)\right) v^3(t)\hat{v}^3(s)\right] \\ \times \exp\left[-\frac{1}{2}\int dtds(\hat{v}^3(t)\hat{v}^3(s)C_2(t,s) + v^3(t)v^3(s)\hat{C}_3(t,s))\right],\end{aligned} \tag{46}$$

$$\begin{aligned}\mathcal{Z}_{0,2,4;k} = \int \mathcal{D}[v^{0,2,4},\hat{v}^{0,2,4}] \exp\left[-\frac{1}{2}\int dtds\left(\alpha^{-1}\lambda_k \hat{v}_k^2(t)\hat{v}_k^2(s)C_1(t,s) + \nu^{-1}\hat{v}_k^4(t)\hat{v}_k^4(s)C_3(t,s)\right)\right] \\ \times \exp\left[-\frac{1}{2}\int dtds\left(\lambda_k v_k^0(t)v_k^0(s)\hat{C}_0(t,s) + v_k^2(t)v_k^2(s)\hat{C}_2(t,s) + v_k^4(t)v_k^4(s)\hat{C}_4(t,s)\right)\right] \\ \times \exp\left[-i\int dtds\left(R_3(t,s)v_k^2(s)\hat{v}_k^4(t) + \lambda_k R_1(t,s)v_k^0(s)\hat{v}_k^2(t)\right)\right].\end{aligned} \tag{47}$$

In the large M limit we evaluate this integral via saddle point. The saddle point equations give:

$$\begin{aligned}C_0(t,s) &= \frac{1}{M}\sum_k \lambda_k \left\langle v_k^0(t)v_k^0(s)\right\rangle \\ C_i(t,s) &= \langle v^\ell(t)v^\ell(s)\rangle, \quad \ell = \{1,2,3,4\} \\ \\ R_1(t,s) &= -i\langle v^1(t)\hat{v}^1(s)\rangle \\ R_3(t,s) &= -i\langle v^3(t)\hat{v}^3(s)\rangle \\ \hat{R}^1(t,s) &= -i\frac{1}{M}\sum_k \lambda_k \hat{v}_k^2(t)v_k^0(s) \equiv R_{0,2}(t,s) \\ \hat{R}^3(t,s) &= -i\frac{1}{M}\sum_k \lambda_k \hat{v}_k^4(t)v_k^2(s) \equiv R_{2,4}(t,s).\end{aligned} \tag{48}$$

Here $\langle\cdot\rangle$ denotes an average taken with respect to the statistical ensemble given by the corresponding partition function $\mathcal{Z}_i$. Lastly, the saddle point equations for the $\hat{C}_i(t,s)$ variables are all quadratic functions of the variables $\{\hat{v}^0,\hat{v}^1,\hat{v}^2,\hat{v}^3\}$ which vanish under the average defined by $\mathcal{Z}$ (Helias & Dahmen, 2020). Following the discussion below Equation 39, we take $\hat{C}_i(t,s) = 0$, which will enforce $\langle \hat{v}_i(t)\hat{v}_i(s)\rangle = 0$ and lead to the correct dynamical equations.

To evaluate the remaining, we can integrate out the $\hat{v}^i$ variables. First let us look at $\mathcal{Z}_1$. Using the Hubbard-Stratonovich trick we can write the action in terms linear in $\hat{v}^1$. This gives

$$\begin{aligned}\mathcal{Z}_1 = \int \mathcal{D}[v^1,\hat{v}^1,u^1] \exp\left[i\int dtds\, \hat{v}^1(t)\left[\delta(t-s)(v^1(s)-u(s)) - \alpha^{-1}\hat{R}_1(t,s)v^1(s)\right]\right] \\ \times \exp\left[-\frac{1}{2}\int dtds\, u(t)u(s)C_0^{-1}(t,s) + v^1(t)v^1(s)\hat{C}_1(t,s))\right]\end{aligned} \tag{49}$$

We now replace $\hat{C}_1$ by its saddle point value of 0 and $\hat{R}_1$ by $R_{0,2}$. Integrating over $\hat{v}$ gives a delta function:

$$v^1(t) = u^1(t) + \frac{1}{\alpha}\int dsR_{0,2}(t,s)v_1(s), \quad u^1(t) \sim \mathcal{GP}(0,C_0). \tag{50}$$

Analogously for v^3 we get

$$v^3(t) = u^3(t) + \frac{1}{\nu}\int ds R_{2,4}(t,s)v^3(s), \quad u^3(t) \sim \mathcal{GP}(0, C_2) \tag{51}$$

For $\mathcal{Z}_{0,2,4;k}$ after replacing $\hat{C}_0, \hat{C}_2, \hat{C}_4$ with their saddle point values we get:

$$\begin{aligned}\mathcal{Z}_{0,2,4;k} = \exp\left[-\frac{1}{2}\int dtds\left(\alpha^{-1}\lambda_k \hat{v}_k^2(t)\hat{v}_k^2(s)C_1(t,s) + \nu^{-1}\hat{v}_k^4(t)\hat{v}_k^4(s)C_3(t,s)\right)\right] \\ \times \exp\left[-\int dtds\left(iR_3(t,s)v_k^2(s)\hat{v}_k^4(t) + i\lambda_k R_1(t,s)v_k^0(s)\hat{v}_k^2(t)\right)\right]\end{aligned} \tag{52}$$

Using the same Hubbard-Stratonovich trick on $\hat{v}_k^2$ gives:

$$v_k^2(t) = u_k^2(t) + \lambda_k \int ds R_1(t,s)v_k^0(s), \quad u_k^2(t) \sim \mathcal{GP}\left(0, \frac{1}{\alpha}\lambda_k C_1\right). \tag{53}$$

On $\hat{v}_k^4$ we similarly get:

$$v_k^4(t) = u_k^4(t) + \int ds R_3(t,s)\cdot v_k^2(s), \quad u_k^4(t) \sim \mathcal{GP}\left(0, \frac{1}{\nu}C_3\right) \tag{54}$$

Lastly, the equations of motion for v_k^0 in terms of v_k^4 are known:

$$\partial_t v_k^0(t) = -v_k^4(t). \tag{55}$$

One can easily add momentum by replacing $\partial_t v_k^0(t)$ with $(\beta\partial_t^2 + \partial_t)v_k^0(t)$ without changing anything else about the derivation.

B.2.1. Interpretation of the Response Functions

Following (Crisanti & Sompolinsky, 2018; Helias & Dahmen, 2020), we can understand the $\langle \hat{v}^a(t)v^b(s)\rangle$ correlators by adding in the single-site moment generating function (e.g. Equation (49)) a source $\tilde{j}^b(s)$ that couples to $\hat{v}^b$ at time s. As in the discussion below equation 39, differentiating $\langle v^a(t)\rangle$ with respect to that source corresponds to its response to a kick in the dynamics of v^b at time s. We denote this by:

$$R_{i,j}(t,s) = \left\langle \frac{\delta v^i(t)}{\delta v^j(s)} \right\rangle. \tag{56}$$

B.3. Cavity Derivation

The cavity derivation relies on Taylor expanding the dynamics upon the addition of a new sample or feature. We will work through each cavity step one at a time by considering the influence of a single new base feature, new sample, and new projected feature. In each step, the goal is to compute the marginal statistics of the added variables. This requires tracking the linear response to all other variables in the system.

Adding a Base Feature Upon addition of a base feature with eigenvalue λ_0 so that there are $M+1$ instead of M features $\{v_k^0, v_k^2, v_k^4\}$ for $k \in \{0, 1, ..., M\}$, we have a perturbation to both $v_\mu^1(t)$ and $v_n^3(t)$. Denote the perturbed versions of the dynamics upon addition of the $M+1$st feature as $\tilde{v}_\mu^1(t)$ and $\tilde{v}_n^3(t)$. At large M we can use linear-response theory to relate the dynamics at $M+1$ features to the dynamics of the original M feature system

$$\begin{aligned}\tilde{v}_\mu^1(t) &\sim v_\mu^1(t) + \frac{1}{\sqrt{M}}\sum_{\nu=1}^{P}\int ds \frac{\partial v_\mu^1(t)}{\partial v_\nu^1(s)}\psi_0^\nu v_0^0(s) \\ \tilde{v}_n^3(t) &\sim v_\mu^1(t) + \frac{1}{\sqrt{M}}\sum_{m=1}^{N}\int ds \frac{\partial v_n^3(t)}{\partial v_m^3(s)}A_{m0}\, v_0^2(s)\end{aligned} \tag{57}$$

The next order corrections have a subleading influence on the dynamics. Now, inserting these perturbed dynamics into the dynamics for the new $(M+1)$st set of variables $\{v_0^2(t), v_0^4(t)\}$. For $v_0^2(t)$, we have

$$v_0^2(t) \sim \frac{1}{\alpha\sqrt{M}} \sum_{\mu=1}^{P} \psi_0^\mu v_\mu^1(t) + \frac{1}{\alpha M} \sum_{\mu,\nu=1}^{P} \int ds\, \psi_0^\mu \frac{\partial v_\mu^1(t)}{\partial v_\nu^1(s)} \psi_0^\nu v_0^0(s) \tag{58}$$

There are now two key steps in simplifying the above expression in the proportional limit:

1. By the fact that the $v_\mu^1(t)$ dynamics are statistically independent of the new feature ψ_0^μ, we can invoke a central limit theorem for the first term which is mean zero and variance $\mathcal{O}(1)$.
2. Similarly, we can invoke a law of large numbers for the second term, which has $\mathcal{O}(1)$ mean and variance on the order of $\mathcal{O}(M^{-1})$. Therefore in the asymptotic limit it can be safely approximated by its mean.

We note in passing that neither of these steps require the ψ_0^μ variables to be Gaussian. Thus we obtain the following asymptotic statistical description of the $v_0^2(t)$ random variable

$$\begin{aligned} v_0^2(t) &\sim u_0^2(t) + \int ds R_1(t,s) v_0^0(s) \\ u_0^2(t) &\sim \mathcal{GP}(0, \alpha^{-1}\lambda_0 C_1)\ ,\ R_1(t,s) \equiv \frac{1}{P}\sum_{\mu=1}^{P} \left\langle \frac{\partial v_\mu^1(t)}{\partial v_\mu^1(s)} \right\rangle . \end{aligned} \tag{59}$$

Following an identical argument for $v_0^4(t)$ we have

$$\begin{aligned} v_0^4(t) &\sim \frac{1}{\nu\sqrt{M}} \sum_{n=1}^{M} A_{n0} v_n^3(t) + \frac{1}{\nu M} \sum_{nm} \int ds\, A_{n0} \frac{\partial v_n^3(t)}{\partial v_m^3(s)} A_{m0} v_0^2(s) \\ &\sim u_0^3(t) + \int ds R_3(t,s) v_0^2(s) \\ u_0^3(t) &\sim \mathcal{GP}(0, \nu^{-1} C_3)\ ,\ R_3(t,s) = \frac{1}{N}\sum_{n=1}^{N} \left\langle \frac{\partial v_n^3(t)}{\partial v_n^3(s)} \right\rangle \end{aligned} \tag{60}$$

Adding a Sample Next, we can consider the influence of adding a new data point. We will aim to characterize a $P+1$ data point system in terms of the dynamics when P points are present. Upon the addition of a new data point $\boldsymbol{\psi}^0$ the field $v_k^0(t)$ will be perturbed to $\tilde{v}_k^0(t)$. Again invoking linear response theory, we can expand the perturbed value around the P-sample dynamics

$$\tilde{v}_k^0(t) \sim v_k^0(t) + \frac{1}{\alpha\sqrt{M}} \sum_{\ell=1}^{M} \int ds\, \frac{\partial v_k^0(t)}{\partial v_\ell^2(s)} \psi_\ell^0 v_0^1(s) \tag{61}$$

Now, computing the dynamics of the new random variable $v_0^1(t)$

$$\begin{aligned} v_0^1(t) &\sim \frac{1}{\sqrt{M}} \sum_{k=1}^{M} \psi_k^0 v_k^0(t) + \frac{1}{\alpha M} \int ds \sum_{k\ell} \psi_k^0\, \frac{\partial v_k^0(t)}{\partial v_\ell^2(s)}\, \psi_\ell^0\, v_0^1(s) \\ &\sim u_0^1(t) + \frac{1}{\alpha} \int ds R_{0,2}(t,s) v_0^1(s) \\ u_0^1(t) &\sim \mathcal{GP}\left(0, \frac{1}{M}\sum_k \lambda_k C_k^0\right)\ ,\ R_{0,2}(t,s) = \frac{1}{M}\sum_{k=1}^{M} \lambda_k \left\langle \frac{\partial v_k^0(t)}{\partial v_k^2(s)} \right\rangle \end{aligned} \tag{62}$$

Adding a Projected Feature Now, we finally consider the effect of introducing a single new projected feature so that instead of N we now have $N+1$ projected features. This causes a perturbation to $\{v_k^2(t)\}$ which we

$$\tilde{v}_k^2(t) \sim v_k^2(t) + \frac{1}{\nu\sqrt{M}} \sum_{\ell=1}^{M} \int ds\, \frac{\partial v_k^2(t)}{\partial v_\ell^4(s)} A_{0\ell}\, v_0^3(s) \tag{63}$$

Now, we compute the dynamics for the added variable $v_0^3(t)$

$$\begin{aligned} v_0^3(t) &\sim \frac{1}{\sqrt{M}} \sum_{k=1}^{M} A_{0k} v_k^2(t) + \frac{1}{\nu M} \sum_{k\ell} \int ds A_{0k} \frac{\partial v_k^2(t)}{\partial v_\ell^4(s)} A_{0\ell}\, v_0^3(s) \\ &\sim u_0^3(t) + \frac{1}{\nu} \int ds R_{2,4}(t,s) v_0^3(s) \\ u_0^3(t) &\sim \mathcal{GP}(0, C_2)\ ,\ R_{2,4}(t,s) = \frac{1}{M} \sum_{k=1}^{M} \left\langle \frac{\partial v_k^2(t)}{\partial v_k^4(s)} \right\rangle \end{aligned} \tag{64}$$

Putting it all together Now, using the information gained in the previous sections, we can combine all of the dynamics for each field into a closed set of stochastic processes. This recovers the DMFT equations of Appendix A.2.

C. Final Losses (the $t \to \infty$ Limit of DMFT)

In this section we work out exact expressions for the large time limit of DMFT. By comparing with prior computations of the mean-field statics of this problem computed in (Atanasov et al., 2023; Zavatone-Veth & Pehlevan, 2023; Ruben & Pehlevan, 2023; Maloney et al., 2022; Simon et al., 2021), we show that the large time and large M limits commute, specifically that $\lim_{M,N,P\to\infty} \lim_{t\to\infty} \mathcal{L}(M,N,P,t) = \lim_{t\to\infty} \lim_{M,N,P\to\infty} \mathcal{L}(M,N,P,t)$. We invoke the final value theorem and use the response functions as before.

Final Value Theorem We note that for functions which vanish at $t = -\infty$, that

$$\lim_{\omega\to 0} i\omega\, \mathcal{H}(\omega) = -\lim_{\omega\to 0} \int_{-\infty}^{\infty} d\tau \left[\frac{\partial}{\partial \tau} e^{-i\omega\tau}\right] H(\tau) = \lim_{\omega\to 0} \int_{-\infty}^{\infty} d\tau \left[\frac{\partial}{\partial \tau} H(\tau)\right] e^{-i\omega\tau} = \lim_{\tau\to\infty} H(\tau) \tag{65}$$

where we invoked integration by parts and used the assumption that $\lim_{\tau\to-\infty} H(\tau) = 0$, a condition that is satisfied for the correlation and response functions in our theory. We can therefore use the identity $\lim_{\tau\to\infty} H(\tau) = \lim_{\omega\to 0} i\omega\mathcal{H}(\omega)$ to extract the final values of our order parameters.

$$\lim_{\tau\to\infty} H_k(\tau) = \lim_{\omega\to 0} i\omega \mathcal{H}_k(\omega) = \lim_{\omega\to 0} \frac{1}{1 + \lambda_k (i\omega)^{-1} \mathcal{R}_1(\omega)\mathcal{R}_3(\omega)}. \tag{66}$$

We also need to invoke a similar relationship for the final values of the correlation functions

$$\begin{aligned} \lim_{t,s\to\infty} C(t,s) &= \lim_{\omega,\omega'\to 0} (i\omega)(i\omega')\, \mathcal{C}(\omega,\omega') \\ \mathcal{C}(\omega,\omega') &= \int dt \int ds\, e^{-i(\omega t + \omega' s)} C(t,s) \end{aligned} \tag{67}$$

where $\mathcal{C}$ is the two-variable Fourier transform. The final value of the test loss is $\lim_{t\to\infty} \mathcal{L}(t) = \lim_{t,s\to\infty} C_0(t,s)$.

C.1. General Case (Finite ν, α)

Before working out the solution to the response functions, we note that the following condition is always satisfied

$$\nu(1 - \mathcal{R}_3(\omega)) = \alpha(1 - \mathcal{R}_1(\omega)). \tag{68}$$

For $\nu = \alpha$, this equation implies that $\mathcal{R}_1 = \mathcal{R}_3$. For $\nu \neq \alpha$, we can have either $\mathcal{R}_1 \to 0$ or $\mathcal{R}_3 \to 0$ but not both. We consider each of these cases below.

Over-parameterized Case $\nu > \alpha$: In this case, the response function $\mathcal{R}_1 \sim \mathcal{O}(i\omega)$ as $\omega \to 0$ and $\mathcal{R}_3 \sim 1 - \frac{\alpha}{\nu}$ as $\omega \to 0$. We thus define

$$r \equiv \lim_{\omega\to 0} (i\omega)^{-1} \mathcal{R}_1(\omega)\mathcal{R}_3(\omega) \tag{69}$$

Using the equation which defines $\mathcal{R}_1$, we find that the variable r satisfies the following relationship at $\omega \to 0$

$$\alpha = \frac{1}{M} \sum_k \frac{\lambda_k r}{1 + \lambda_k r} \tag{70}$$

After solving this implicit equation, we can find the limiting value of $i\omega \mathcal{H}_k(\omega)$ as

$$H_k^\infty = \lim_{\tau \to \infty} H_k(\tau) = \lim_{\omega \to 0} i\omega \mathcal{H}_k(\omega) = \frac{1}{1 + \lambda_k r} \tag{71}$$

Next, we can work out the scaling of the correlation functions in the limit of low frequency. We define the following limiting quantities based on a scaling analysis performed on our correlation functions for small ω

$$\begin{aligned}
C_0^\infty &\equiv \lim_{t,s \to \infty} C_0(t,s) = \lim_{\omega,\omega' \to 0} (i\omega)(i\omega') \mathcal{C}_0(\omega, \omega') \\
C_1^\infty &\equiv \int_0^\infty \int_0^\infty dt' ds' C_1(t', s') = \lim_{\omega,\omega' \to 0} \mathcal{C}_1(\omega, \omega') \\
C_2^\infty &\equiv \int_0^\infty \int_0^\infty dt' ds' C_2(t', s') = \lim_{\omega,\omega' \to 0} \mathcal{C}_2(\omega, \omega') \\
C_3^\infty &\equiv \int_0^\infty \int_0^\infty dt' ds' C_3(t', s') = \lim_{\omega,\omega' \to 0} \mathcal{C}_3(\omega, \omega')
\end{aligned} \tag{72}$$

These limiting quantities satisfy the closed set of linear equations

$$\begin{aligned}
C_0^\infty &= \frac{1}{M} \sum_k \lambda_k (H_k^\infty)^2 \left[(w_k^\star)^2 + \frac{1}{\nu} C_3^\infty + \frac{1}{\alpha} \lambda_k \left(1 - \frac{\alpha}{\nu}\right)^2 C_1^\infty \right] \\
C_1^\infty &= \frac{r^2}{(1 - \frac{\alpha}{\nu})^2} C_0^\infty \\
C_2^\infty &= \frac{1}{\alpha M} \sum_k \lambda_k (H_k^\infty)^2 C_1^\infty + \frac{r^2}{(1 - \frac{\alpha}{\nu})^2} \frac{1}{M} \sum_k \lambda_k^2 (H_k^\infty)^2 [(w_k^\star)^2 + \nu^{-1} C_3^\infty] \\
C_3^\infty &= \left(1 - \frac{\alpha}{\nu}\right)^2 C_2^\infty
\end{aligned} \tag{73}$$

These equations can be solved for $\{C_0^\infty, C_1^\infty, C_2^\infty, C_3^\infty\}$. Simplifying the expressions to a two-variable system, we find

$$\begin{aligned}
C_0^\infty &= \frac{1}{M} \sum_k \lambda_k (H_k^\infty)^2 \left[(w_k^\star)^2 + \frac{1}{\nu} \left(1 - \frac{\alpha}{\nu}\right)^2 C_2^\infty + \frac{1}{\alpha} \lambda_k r^2 C_0^\infty \right] \\
C_2^\infty &= \frac{r^2}{\alpha (1 - \alpha/\nu)^2 M} \sum_k \lambda_k (H_k^\infty)^2 C_0^\infty + \frac{r^2}{(1 - \frac{\alpha}{\nu})^2} \frac{1}{M} \sum_k \lambda_k^2 (H_k^\infty)^2 \left[(w_k^\star)^2 + \frac{1}{\nu} \left(1 - \frac{\alpha}{\nu}\right)^2 C_2^\infty \right]
\end{aligned}$$

This expression recovers the ridgeless limit of the replica results of (Atanasov et al., 2023; Zavatone-Veth & Pehlevan, 2023) and the random matrix analysis of (Simon et al., 2023).

Under-parameterized Case $\nu < \alpha$: Following the same procedure, we note that for $\nu < \alpha$ that $\mathcal{R}_3 \sim \mathcal{O}(i\omega)$ and $\mathcal{R}_1 \sim 1 - \frac{\nu}{\alpha}$. We thus find the following equation for $r = \lim_{\omega \to \infty} (i\omega)^{-1} \mathcal{R}_1(\omega) \mathcal{R}_3(\omega)$.

$$\nu = \frac{1}{M} \sum_k \frac{\lambda_k r}{\lambda_k r + 1} \tag{74}$$

where as before $H_k^\infty = \frac{1}{1 + \lambda_k r}$. The analogous scaling argument for small ω gives us the following set of well-defined limiting quantities

$$\begin{aligned}
C_0^\infty &\equiv \lim_{t,s \to \infty} C_0(t,s) = \lim_{\omega,\omega' \to 0} (i\omega)(i\omega') \mathcal{C}(\omega, \omega') \\
C_1^\infty &\equiv \lim_{t,s \to \infty} C_1(t,s) = \lim_{\omega,\omega' \to 0} (i\omega)(i\omega') \mathcal{C}_1(\omega, \omega') \\
C_2^\infty &\equiv \lim_{t,s \to \infty} C_2(t,s) = \lim_{\omega,\omega' \to 0} (i\omega)(i\omega') \mathcal{C}_2(\omega, \omega') \\
C_3^\infty &\equiv \int_0^\infty \int_0^\infty dt ds \, C_3(t,s) = \lim_{\omega,\omega' \to 0} \mathcal{C}_3(\omega, \omega').
\end{aligned} \tag{75}$$

where these limiting correlation values satisfy

$$\begin{aligned}
C_0^\infty &= \frac{1}{M}\sum_k \lambda_k (H_k^\infty)^2 \left[(w_k^\star)^2 + \frac{1}{\nu} C_3^\infty + \frac{1}{\alpha}\lambda_k \frac{r^2}{(1-\frac{\nu}{\alpha})^2} C_1^\infty\right] \\
C_1^\infty &= \left(1-\frac{\nu}{\alpha}\right)^2 C_0^\infty \\
C_2^\infty &= \frac{1}{\alpha M}\sum_k \lambda_k (H_k^\infty)^2 C_1^\infty + \frac{1}{M}\left(1-\frac{\nu}{\alpha}\right)^2 \sum_k \lambda_k^2 (H_k^\infty)^2 \left[(w_k^\star)^2 + \frac{1}{\nu} C_3^\infty\right] \\
C_3^\infty &= \frac{r^2}{(1-\frac{\nu}{\alpha})^2} C_2^\infty .
\end{aligned} \tag{76}$$

This is again a closed linear system of equations for the variables $\{C_0^\infty, C_1^\infty, C_2^\infty, C_3^\infty\}$. In the next section, we recover the result for kernel regression where $\nu \to \infty$ and the learning curve for infinite data $\alpha \to \infty$ with respect to model size ν.

C.2. Learning Curves for Kernel Regression $\nu, t \to \infty$

In the $t \to \infty$ and $\nu \to \infty$ limit we recover the learning curve for kernel regression with eigenvalues λ_k. To match the notation of (Canatar et al., 2021), we define

$$\lim_{\omega \to 0} (i\omega)^{-1} \mathcal{R}_1(\omega) \equiv \alpha \kappa^{-1} \tag{77}$$

which generates the following self-consistent equation for κ

$$1 = \frac{1}{M}\sum_k \frac{\lambda_k}{\lambda_k \alpha + \kappa}. \tag{78}$$

Plugging this into the expression for the loss, we find

$$\begin{aligned}
(i\omega)(i\omega')\mathcal{C}_0(\omega,\omega') &\sim \frac{1}{M}\sum_k \lambda_k \frac{\kappa^2}{(\kappa+\lambda_k\alpha)^2}[(w_k^\star)^2 + \alpha^{-1}\lambda_k \mathcal{C}_1(\omega,\omega')] \\
\mathcal{C}_1(\omega,\omega') &= (i\omega)(i\omega')\alpha^2\kappa^{-2}\mathcal{C}_0(\omega,\omega')
\end{aligned} \tag{79}$$

Letting $C_\infty \equiv \lim_{\omega,\omega'\to 0}(i\omega)(i\omega')\mathcal{C}(s,s')$, we have

$$\begin{aligned}
C_\infty &= \frac{1}{M}\sum_k \lambda_k \frac{\kappa^2}{(\kappa+\lambda_k\alpha)^2}(w_k^\star)^2 + C_\infty \frac{\alpha}{M}\sum_k \frac{\lambda_k^2}{(\lambda_k\alpha+\kappa)^2} \\
&= \frac{1}{1-\gamma}\sum_k \lambda_k (w_k^\star)^2 \frac{\kappa^2}{(\kappa+\lambda_k\alpha)^2}\,, \quad \gamma = \frac{\alpha}{M}\sum_k \frac{\lambda_k^2}{(\lambda_k\alpha+\kappa)^2}
\end{aligned} \tag{80}$$

The variable κ decreases from $\left[\frac{1}{M}\sum_k \lambda_k, 0\right]$ as $\alpha \in [0,1]$. For $\alpha > 1$ we have $\kappa = 0$. The quantity $\frac{1}{1-\gamma}$ comes from overfitting due to variance from the randomly sampled dataset.

D. Early Time Dynamics (High-Frequency Range)

In this section, we explore the early time dynamical effects of this model. Similar to how the late time dynamical effects could be measured by examining the low frequency $\omega \ll 1$ part of the response and correlation functions, in this section, we analyze the high frequency components $\omega \gg 1$. We start by noting the following expansions valid near $\omega \to \infty$

$$\begin{aligned}
\mathcal{R}_1(\omega) &\sim 1 - \frac{1}{\alpha(i\omega)}\left[\frac{1}{M}\sum_k \lambda_k\right] + \mathcal{O}(\omega^{-2}) \\
\mathcal{R}_3(\omega) &\sim 1 - \frac{1}{\nu(i\omega)}\left[\frac{1}{M}\sum_k \lambda_k\right] + \mathcal{O}(\omega^{-2})
\end{aligned} \tag{81}$$

We let $c = \frac{1}{M}\sum_k \lambda_k$. These can be plugged into the transfer function for mode k

$$\mathcal{H}_k(\omega) \sim \frac{1}{i\omega + \lambda_k - c(\alpha^{-1}+\nu^{-1})(i\omega)^{-1}} \sim \frac{1}{i\omega+\lambda_k}\left[1 + \frac{c\lambda_k(\alpha^{-1}+\nu^{-1})}{i\omega(i\omega+\lambda_k)}\right] + \mathcal{O}\left(\omega^{-2}\right) \tag{82}$$

Performing an inverse Fourier transform, we find the following early time asymptotics

$$\begin{aligned} H_k(t) &\sim e^{-\lambda_k t} + c\lambda_k(\alpha^{-1}+\nu^{-1})\int \frac{d\omega}{2\pi}\frac{e^{i\omega t}}{i\omega(i\omega+\lambda_k)^2} \\ &= e^{-\lambda_k t} - c\lambda_k(\alpha^{-1}+\nu^{-1})\frac{\partial}{\partial\lambda_k}\int \frac{d\omega}{2\pi}\frac{e^{i\omega t}}{i\omega(i\omega+\lambda_k)} \\ &= e^{-\lambda_k t} - c\lambda_k(\alpha^{-1}+\nu^{-1})\frac{\partial}{\partial\lambda_k}\left[\frac{1}{\lambda_k} - \frac{1}{\lambda_k}e^{-\lambda_k t}\right] \\ &= e^{-\lambda_k t} + \frac{c(\alpha^{-1}+\nu^{-1})}{\lambda_k}\left[1 - e^{-\lambda_k t} - \lambda_k t\, e^{-\lambda_k t}\right] \end{aligned} \tag{83}$$

We see from this expression that the early time corrections always scale as $1/\alpha$ or $1/\nu$ and that these corrections build up over time. We also note that in this picture, $H_k(t)$ is minimized in the limit of large model and large data $\alpha, \nu \to \infty$ (limited data and limited model size strictly harm performance). A similar expansion can be performed for all of the correlation functions $\mathcal{C}(\omega, \omega')$ with $\omega, \omega' \gg 1$ which also give leading corrections which scale as $1/\alpha$ and $1/\nu$.

E. Buildup of Overfitting Effects

In this section, we derive a formula for the gap between test loss $\mathcal{L}(t)$ and train loss $\hat{\mathcal{L}}(t)$. We start from the following formula

$$v_1(t) = u_1(t) + \frac{1}{\alpha}\int_0^t ds R_{0,2}(t,s)v_1(s) \tag{84}$$

Moving the $v_1(t)$ term to the other side, and using the fact that $\langle u_1(t)u_1(s)\rangle = C_1(t,s)$, we find the following relationship between train and test loss

$$\begin{aligned} \mathcal{L}(t) = \langle u_1(t)u_1(t)\rangle &= \langle v_1(t)v_1(t)\rangle - \frac{2}{\alpha}\int_0^t dt' R_{0,2}(t,t')\langle v_1(t)v_1(t')\rangle \\ &+ \frac{1}{\alpha^2}\int_0^t dt'\int_0^t ds' R_{0,2}(t,t')R_{0,2}(t,s')\langle v_1(t')v_1(s')\rangle \\ &= \hat{\mathcal{L}}(t) - \frac{2}{\alpha}\int_0^t dt' R_{0,2}(t,t')C_1(t,t') + \frac{1}{\alpha^2}\int_0^t dt'\int_0^t ds' R_{0,2}(t,t')R_{0,2}(t,s')C_1(t',s'). \end{aligned} \tag{85}$$

To get a sense of these expressions at early and late timescales, we investigate the Fourier transforms at high $\omega \gg 1$ and low $\omega \ll 1$ frequencies respectively.

E.1. High Frequency Range / Early Time

The relationship between Fourier transforms at high frequencies $\omega \gg 1$ is

$$\mathcal{C}_0(\omega,\omega') = \frac{1}{\mathcal{R}_1(\omega)\mathcal{R}_1(\omega')}\mathcal{C}_1(\omega,\omega') \sim \mathcal{C}_1(\omega,\omega') + \frac{c}{\alpha(i\omega')}\mathcal{C}_1(\omega,\omega') + \frac{c}{\alpha(i\omega')}\mathcal{C}_1(\omega,\omega') + \mathcal{O}((i\omega)^{-2} + (i\omega')^{-2}) \tag{86}$$

where $c = \frac{1}{M}\sum_k \lambda_k$. Taking a Fourier transform back to real time gives us the following early time differential equation for the test-loss train loss gap

$$\partial_{ts}^2[C_0(t,s) - C_1(t,s)] = \partial_{ts}^2 C_1(t,s) + \frac{c}{\alpha}(\partial_t + \partial_s)C_1(t,s). \tag{87}$$

The above equation should hold for early times. We note that $C_0(t,t) - C_1(t,t) = \mathcal{L}(t) - \hat{\mathcal{L}}(t)$ exactly recovers the test-train gap.

E.2. Low Frequency Range/Late Time

At late time/low frequency, as we showed in Appendix C, the behavior of the C_1 correlation function depends on whether the model is over-parameterized or under-parameterized. In the overparameterized case, the asymptotic train loss is zero while the asymptotic test loss is nonzero. In the underparameterized case, we have a limiting value for both the test and train loss which can be computed from the expressions in Appendix C.

F. Timescale/Eigenvalue Density Interpretation

We can use an alternative interpretation of the Fourier transforms derived in previous sections to obtain the timescale density for the dynamics. Since this is a linear model defined by an effective matrix $\frac{d}{dt}\boldsymbol{v}^0 = -\left(\frac{1}{N}\boldsymbol{A}^\top \boldsymbol{A}\right)\left(\frac{1}{P}\boldsymbol{\Psi}^\top \boldsymbol{\Psi}\right)\boldsymbol{v}^0$, this is equivalent to computing the eigenvalue density. We start by expanding the transfer function for mode k in the basis of exponentials

$$H_k(t) = \int_0^\infty du\, \rho_k(u)\, e^{-ut}. \tag{88}$$

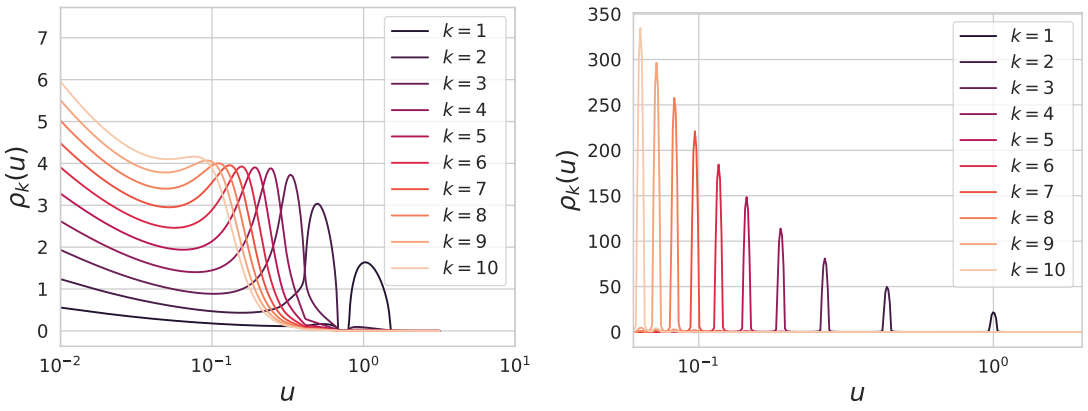
We allow for Dirac-delta masses at $u = 0$ which correspond to the constant (unlearnable) components. Next, we note that the Fourier transform has the form

$$\mathcal{H}_k(\omega) = \int_{-\infty}^\infty dt\, e^{-i\omega t} H_k(t) = \int_0^\infty du\, \rho_k(u) \int_{-\infty}^\infty dt\, e^{-(u+i\omega)t} = \int_0^\infty du\, \frac{\rho_k(u)}{i\omega + u}. \tag{89}$$

We can recover the density $\rho_k(s)$ by using the Sokhotski–Plemelj theorem $\frac{1}{\pi}\text{Im}\frac{1}{-i\epsilon+u-s} = \delta(u-s)$ which gives us

$$\rho_k(u) = \lim_{\epsilon\to 0} \frac{1}{\pi}\text{Im}\, \mathcal{H}_k(iu - \epsilon). \tag{90}$$

This allows us to interpret the spread of timescales from the random sampling of data and the random projection $\boldsymbol{A}$. In the limit of $\alpha, \nu \to \infty$ we have $\rho_k(s) = \delta(s - \lambda_k)$ but for finite α, ν the density spreads out. We visualize these densities for power law features in Figure 8.



(a) $N = 50, P = 100$ (b) $N = 10000, P = 10000$

Figure 8. Timescale (eigenvalue) densities for each transfer function $H_k(\tau)$ with power law features with $b = 1.2$. For limited N, P there is a significant spread of timescales for each mode. For $N, P \to \infty$ the density converges to a Dirac mass at $u = \lambda_k$.

F.1. Recovering the Marchenko-Pastur Law from DMFT Response Functions

To further illustrate the validity of this perspective, we show that it is possible to recover known random matrix theory results using this technique. To illustrate this, we study the case where $\lambda_k = 1$ and take $\nu \to \infty$. In this case, we have the coupled equations

$$\mathcal{H}(\omega) = \frac{1}{i\omega + \mathcal{R}_1(\omega)}\ ,\ \mathcal{R}_1(\omega) = 1 - \frac{1}{\alpha}\mathcal{R}_1(\omega)\mathcal{H}(\omega) \tag{91}$$

Combining these equations gives the single equation

$$\mathcal{H}(\omega) = \frac{1}{i\omega + \frac{\alpha}{\alpha + \mathcal{H}(\omega)}} \ , \implies i\omega\mathcal{H}(\omega)^2 + (\alpha i\omega + \alpha - 1)\mathcal{H}(\omega) - \alpha = 0$$
$$\mathcal{H}(\omega) = -\frac{1}{2i\omega}\left[(\alpha i\omega + \alpha - 1) + \sqrt{(\alpha i\omega + \alpha - 1)^2 + 4i\omega\alpha}\right] \tag{92}$$

Now, evaluating this expression at $i\omega = -s - i\epsilon$ gives

$$\mathcal{H}(is - \epsilon) = \frac{1}{2(s + i\epsilon)}\left[(-\alpha s - i\alpha\epsilon + \alpha - 1) + \sqrt{(-\alpha s - \alpha i\epsilon + \alpha - 1)^2 - 4(s + i\epsilon)\alpha}\right] \tag{93}$$

The radical has an imaginary solution in the $\epsilon \to 0$ limit provided that

$$s \in [s_-, s_+] \ , \ s_\pm = \left(1 \pm \frac{1}{\sqrt{\alpha}}\right)^2 \tag{94}$$

In this interval $[s_-, s_+]$, the density $\rho(s) = \lim_{\epsilon \to 0} \frac{1}{\pi}\text{Im}\mathcal{H}(is - \epsilon)$ has the form

$$\rho(s) = \frac{\alpha\sqrt{(s - s_-)(s - s_+)}}{2\pi s} \ , \ s \in [s_-, s_+] \tag{95}$$

which is precisely the bulk of the Marchenko-Pastur law.

G. Non-Proportional (Dimension-Free) Limit

We can imagine a situation where the original features are already infinite dimensional ($M \to \infty$ is taken first). This would correspond more naturally to the connection between infinite dimensional RKHS's induced by neural networks at infinite width (Bordelon et al., 2020; Canatar et al., 2021; Cheng & Montanari, 2022). Further, we will assume a trace class kernel $K(\boldsymbol{x}, \boldsymbol{x}') = \boldsymbol{\psi}(\boldsymbol{x}) \cdot \boldsymbol{\psi}(\boldsymbol{x}')$ for the base features $\boldsymbol{\psi}$ which diagonalizes over the data distribution $p(\boldsymbol{x})$ as

$$\int K(\boldsymbol{x}, \boldsymbol{x}')\phi_k(\boldsymbol{x}')p(\boldsymbol{x}')d\boldsymbol{x}' = \lambda_k\phi_k(\boldsymbol{x}) \quad , \quad \sum_{k=1}^{\infty}\lambda_k < \infty. \tag{96}$$

As before, we are concerned with the test and train losses

$$\mathcal{L}(t) = \sum_{k=1}^{\infty}\lambda_k v_k^0(t)^2 \ , \ \hat{\mathcal{L}}(t) = \frac{1}{P}\sum_{\mu=1}^{P} v_\mu^1(t)^2. \tag{97}$$

The appropriate scaling of our four fields of interest in this setting are

$$\boldsymbol{v}^1(t) = \boldsymbol{\Psi}\boldsymbol{v}^0(t) \ , \ \boldsymbol{v}^2(t) = \frac{1}{P}\boldsymbol{\Psi}^\top\boldsymbol{v}^1(t)$$
$$\boldsymbol{v}^3(t) = \boldsymbol{A}\boldsymbol{v}^2(t) \ , \ \boldsymbol{v}^4(t) = \frac{1}{N}\boldsymbol{A}^\top\boldsymbol{v}^3(t). \tag{98}$$

Following the cavity argument given in the previous section, we can approximate the the correlation and response functions as concentrating to arrive at the following field description of the training dynamics

$$\begin{aligned}
&\partial_t v_k^0(t) = -v_k^4(t) \\
&v^1(t) = u^1(t) + \frac{1}{P}\int ds\ R_{0,2}(t,s)v^1(s) \ , \ u^1(t) \sim \mathcal{GP}(0, C_0) \\
&v_k^2(t) = u_k^2(t) + \lambda_k \int ds\ R_1(t,s)v_k^0(s) \ , \ u_k^2(t) \sim \mathcal{GP}\left(0, \frac{1}{P}\lambda_k C^1\right) \\
&v^3(t) = u^3(t) + \frac{1}{N}\int ds\ R_{2,4}(t,s)v^3(s) \ , \ u^3(t) \sim \mathcal{GP}(0, C_2) \\
&v_k^4(t) = u_k^4(t) + \int ds\ R_3(t,s)v_k^2(s) \ , \ u_k^4(t) \sim \mathcal{N}\left(0, \frac{1}{N}C_3\right).
\end{aligned} \tag{99}$$

which are exactly the same equations as in the proportional limit except with the substitution $\nu \to N$ and $\alpha \to P$. The correlation and response functions have the form

$$\begin{aligned} C_0(t,s) &= \sum_{k=1}^{\infty} \lambda_k \left\langle v_k^0(t) v_k^0(s) \right\rangle \ , \ C_1(t,s) = \left\langle v^1(t) v^1(s) \right\rangle \\ C_2(t,s) &= \sum_{k=1}^{\infty} v_k^2(t) v_k^2(s) \ , \ C_3(t,s) = \left\langle v^3(t) v^3(s) \right\rangle \\ R_{0,2}(t,s) &= \sum_{k=1}^{\infty} \lambda_k \left\langle \frac{\delta v_k^0(t)}{\delta u_k^2(s)} \right\rangle \ , \ R_1(t,s) = \left\langle \frac{\delta v^1(t)}{\delta u^1(s)} \right\rangle \\ R_{2,4}(t,s) &= \sum_{k=1}^{\infty} \left\langle \frac{\delta v_k^2(t)}{\delta u_k^4(s)} \right\rangle \ , \ R_3(t,s) = \left\langle \frac{\delta v^3(t)}{\delta u^3(s)} \right\rangle \end{aligned} \tag{100}$$

which will all be $\mathcal{O}(1)$ under this scaling.

H. Effect of Ensembling and Bagging on Dynamics

H.1. What Does/Doesn't Concentrate in the DMFT Limit?

To help gain insight into bias and variance decompositions, we first provide a short primer on which entities concentrate over random draws of matrices $\boldsymbol{A}$ and $\boldsymbol{\Psi}$. For any distinct randomly sampled system, the following objects will always be the same in the asymptotic limit

1. The response functions $\{R_{0,2}(t,s), R_1(t,s), R_{2,4}(t,s), R_3(t,s)\}$
2. The correlation functions $\{C_i(t,s)\}_{i \in \{1,2,3,4\}}$.
3. The train and test loss dynamics

While the above quantities behave as concentrating or "self-averaging" random variables, many important quantities are not the same across different realizations of $\{\boldsymbol{A}, \boldsymbol{\Psi}\}$. For example,

1. The (random) entries of the vectors $\{\boldsymbol{v}^0(t), \boldsymbol{v}^1(t), \boldsymbol{v}^2(t), \boldsymbol{v}^3(t), \boldsymbol{v}^4(t)\}$.
2. The Gaussian sources $\{u^1(t), u^2(t), u^3(t), u^4(t)\}$ which appear in the large system size limit.

In particular, the first implies that the model outputs $f(\boldsymbol{x})$ will generally depend on random variations across datasets or model initializations. This means that we can consider drawing multiple realizations of, for example, projection matrices $\{\boldsymbol{A}_e\}_{e=1}^E$ and then training E separate models using each of them. Averaging these vectors gives us

$$\bar{\boldsymbol{v}}^0(t) = \frac{1}{E} \sum_{e=1}^{E} \boldsymbol{v}_e^0(t) \tag{101}$$

This operation will intuitively "average out" noise from the random projection matrices $\boldsymbol{A}_e$ and in the limit of infinite ensembling $E \to \infty$ will completely eliminate it.

H.2. Definition of Bias and Variance

We adopt the language of the fine-grained bias-variance decomposition in (Adlam & Pennington, 2020b). There, a given learned function generally depends on both the dataset $\mathcal{D}$ and initialization seed θ_0. We write this as $f_{\mathcal{D},\theta_0}$. The role of random initialization is played by the $\boldsymbol{A}$ matrix in our setting. For a given function, its variance over datasets and its *variance* over initializations are respectively given by

$$\mathrm{Var}_{\mathcal{D}} f \equiv \mathbb{E}_{\mathcal{D}} (f_{\mathcal{D},\theta_0} - \mathbb{E}_{\mathcal{D}}[f_{\mathcal{D},\theta_0}])^2 \tag{102}$$

$$\mathrm{Var}_{\theta_0} f \equiv \mathbb{E}_{\theta_0} (f_{\mathcal{D},\theta_0} - \mathbb{E}_{\theta_0}[f_{\mathcal{D},\theta_0}])^2 \tag{103}$$

Here $\mathbb{E}_{\mathcal{D}}[f_{\mathcal{D},\theta_0}]$ and $\mathbb{E}_{\theta_0}[f_{\mathcal{D},\theta_0}]$ can be viewed as *infinitely bagged* or *infinitely ensembled* predictors respectively. The *bias* of a function over datasets or initializations is given by the test error of $\mathbb{E}_{\mathcal{D}}[f_{\mathcal{D},\theta_0}]$, $\mathbb{E}_{\theta_0}[f_{\mathcal{D},\theta_0}]$ respectively. The irreducible bias is given by $\mathbb{E}_{\mathcal{D},\theta_0}[f_{\mathcal{D},\theta_0}]$.

H.3. Derivation

In this section, we consider the effect of ensembling over E random initial conditions and bagging over B random datasets. We let $\boldsymbol{v}^0_{e,b}(t)$ represent the weight discrepancy for model e on dataset b. Here e runs from 1 to E and b runs from 1 to B. The (e,b)th vector has dynamics:

$$\frac{d}{dt}\boldsymbol{v}^0_{e,b}(t) = -\left(\frac{1}{N}\boldsymbol{A}_e^\top \boldsymbol{A}_e\right)\left(\frac{1}{N}\boldsymbol{\Psi}_b^\top \boldsymbol{\Psi}_b\right)\boldsymbol{v}^0_{e,b}(t). \tag{104}$$

Ensembling and bagging would correspond to averaging these $\boldsymbol{v}^0$s over these EB systems

$$\bar{\boldsymbol{v}}^0(t) = \frac{1}{EB}\sum_{e=1}^{E}\sum_{b=1}^{B}\boldsymbol{v}^0_{e,b}(t). \tag{105}$$

The key vectors to track for this computation are

$$\begin{aligned}
\boldsymbol{v}^1_{e,b}(t) &= \frac{1}{\sqrt{M}}\boldsymbol{\Psi}_b\boldsymbol{v}^0_{e,b}(t) + \sigma\boldsymbol{\epsilon}_b\ ,\ \boldsymbol{v}^2_{e,b}(t) = \frac{1}{\alpha\sqrt{M}}\boldsymbol{\Psi}_b^\top\boldsymbol{v}^1_{e,b}(t)\\
\boldsymbol{v}^3_{e,b}(t) &= \frac{1}{\sqrt{M}}\boldsymbol{A}_e\boldsymbol{v}^2_{e,b}(t)\ ,\ \boldsymbol{v}^4_{e,b}(t) = \frac{1}{\nu\sqrt{M}}\boldsymbol{A}_e^\top\boldsymbol{v}^3_{e,b}(t).
\end{aligned} \tag{106}$$

We can further show that the $\boldsymbol{v}^0_{e,b}$ and $\boldsymbol{v}^0_{e',b'}$ have response functions that decouple across e, b. Intuitively, giving the dynamical system e, b a kick should not alter the trajectory of the separate e', b' dynamical system, even if they share disorder $\{\boldsymbol{\Psi}, \boldsymbol{A}\}$. The DMFT description of the proportional limit yields the following integral equations for the v fields:

$$\begin{aligned}
\partial_t v^0_{e,b,k}(t) &= -v^4_{e,b,k}(t)\\
v^1_{e,b}(t) &= u^1_{e,b}(t) + \frac{1}{\alpha}\int dsR_{0,2}(t,s)v^1_{e,b}(s)\\
v^2_{e,b,k}(t) &= u^2_{e,b,k}(t) + \lambda_k\int dsR_1(t,s)v^0_{e,b,k}(s)\\
v^3_{e,b}(t) &= u^3_{e,b}(t) + \frac{1}{\nu}\int dsR_{2,4}(t,s)v^1_{e,b}(s)\\
v^4_{e,b,k}(t) &= u^4_{e,b,k}(t) + \int dsR_3(t,s)v^2_{e,b,k}(s).
\end{aligned} \tag{107}$$

Here, the response functions R are to be computed within a single system. In what follows, we will use $\langle\cdot\rangle$ to denote averages over the disorder, and explicitly write out any averages over the ensemble members and datasets.

The Gaussian variables in the DMFT have the following covariance

$$\begin{aligned}
\left\langle u^1_{e,b}(t)u^1_{e',b'}(s)\right\rangle &= \delta_{b,b'}C^0_{e,e'}(t,s)\\
\left\langle u^2_{e,b,k}(t)u^2_{e',b',k}(s)\right\rangle &= \delta_{b,b'}\frac{\lambda_k}{\alpha}C_{1,e,e'}(t,s)\\
\left\langle u^3_{e,b}(t)u^3_{e',b'}(s)\right\rangle &= \delta_{e,e'}C_{2,b,b'}(t,s)\\
\left\langle u^4_{e,b,k}(t)u^4_{e',b',k}(s)\right\rangle &= \delta_{e,e'}\frac{1}{\nu}C_{3,b,b'}(t,s)
\end{aligned} \tag{108}$$

The covariances above $C_{0,e,e'}, C_{1,e,e'}, C_{2,b,b'}, C_{3,b,b'}$ allow for different ensemble or dataset index but not both. We will use C_0, C_1, C_2, C_3 etc to represent the correlation functions *within a single system*. For instance, $C^0_{e,e'}(t,s) = \frac{1}{M}\sum_k \lambda_k\left\langle v^0_{e,b}(t)v^0_{e',b}(s)\right\rangle$ while $C^0 = \frac{1}{M}\sum_k\lambda_k\left\langle v^0_{e,b}(t)v^0_{e,b}(s)\right\rangle$. The correlation function of interest is thus

$$\begin{aligned}
\mathcal{C}_{0,k,e,e'}(\omega,\omega') &= \mathcal{H}_k(\omega)\mathcal{H}_k(\omega')\left[(w^\star_k)^2 + \frac{1}{\nu}\delta_{e,e'}\mathcal{C}_3(\omega,\omega') + \frac{\lambda_k}{\alpha}\mathcal{C}_{1,e,e'}(\omega,\omega')\right]\\
\mathcal{C}_{1,e,e'}(\omega,\omega') &= \mathcal{R}_1(\omega)\mathcal{R}_1(\omega')\mathcal{C}^0_{e,e'}(\omega,\omega')\\
\mathcal{C}_{2,b,b',k}(\omega,\omega') &= (i\omega)(i\omega')\mathcal{H}_k(\omega)\mathcal{H}_k(\omega')\frac{\lambda_k}{\alpha}\delta_{b,b'}\mathcal{C}_1(\omega,\omega') + \lambda_k^2\mathcal{R}_1(\omega)\mathcal{R}_1(\omega')\left[(w^\star_k)^2 + \frac{1}{\nu}\mathcal{C}_{3,b,b'}(\omega,\omega')\right]\\
\mathcal{C}_{3,b,b'}(\omega,\omega') &= \mathcal{R}_3(\omega)\mathcal{R}_3(\omega)\mathcal{C}_{2,b,b'}(\omega,\omega')
\end{aligned} \tag{109}$$

We can combine the first two equations and the second two equations to identify the structure of the cross-ensemble and cross-dataset (across-system) correlations in terms of the marginal (within-system) correlation statistics

$$\begin{aligned}
\mathcal{C}_{0,e,e'}(\omega,\omega') &= \frac{1}{1-\gamma_0(\omega,\omega')}\frac{1}{M}\sum_k \lambda_k \mathcal{H}_k(\omega)\mathcal{H}_k(\omega')\left[(w_k^\star)^2 + \frac{1}{\nu}\delta_{e,e'}\mathcal{C}_3(\omega,\omega')\right] \\
\gamma_0(\omega,\omega') &= \frac{1}{\alpha M}\sum_k \lambda_k^2 \mathcal{H}_k(\omega)\mathcal{H}_k(\omega')\mathcal{R}_1(\omega)\mathcal{R}_1(\omega') \\
\mathcal{C}_{2,b,b'}(\omega,\omega') &= \frac{1}{1-\gamma_2(\omega,\omega')}\frac{1}{M}\sum_k \lambda_k \left[\mathcal{R}_1(\omega)\mathcal{R}_1(\omega')\lambda_k (w_k^\star)^2 + \frac{1}{\alpha}(i\omega)(i\omega')\mathcal{H}_k(\omega)\mathcal{H}_k(\omega')\delta_{b,b'}\mathcal{C}_1(\omega,\omega')\right] \\
\gamma_2(\omega,\omega') &= \frac{1}{\nu}\mathcal{R}_3(\omega)\mathcal{R}_3(\omega)\frac{1}{M}\sum_k \lambda_k^2
\end{aligned} \tag{110}$$

These equations give the necessary cross-ensemble and cross-dataset correlations. Now we can consider the effect of ensembling and bagging on the dynamics. To do so, consider the Fourier transform of the bagged-ensembled error $\bar{v}_k^0(t) = \frac{1}{EB}\sum_{eb} v_{k,e,b}^0(t)$, which has the Fourier transform

$$\bar{v}_k^0(\omega) = \mathcal{H}_k(\omega)\left[w_k^\star - \frac{1}{EB}\sum_{e,b}\left(u_{e,b,k}^4(\omega) + \mathcal{R}_3(\omega)u_{e,b,k}^2(\omega)\right)\right] \tag{111}$$

Computing the correlation function for this bagged-ensembled field random variable, we find

$$\begin{aligned}
\left\langle \bar{v}_k^0(\omega)\bar{v}_k^0(\omega')\right\rangle &= \mathcal{H}_k(\omega)\mathcal{H}_k(\omega')\left[(w_k^\star)^2 + \frac{1}{\nu E^2 B^2}\sum_{e,e',b,b'}\delta_{e,e'}\mathcal{C}_{3,b,b'}(\omega,\omega') + \frac{\lambda_k}{\alpha E^2 B^2}\sum_{ee'bb'}\delta_{b,b'}\mathcal{C}_{1,e,e'}(\omega,\omega')\right] \\
&= \mathcal{H}_k(\omega)\mathcal{H}_k(\omega')\left[(w_k^\star)^2 + \frac{1}{\nu E B^2}\sum_{b,b'}\mathcal{C}_{3,b,b'}(\omega,\omega') + \frac{\lambda_k}{\alpha E^2 B}\sum_{ee'}\mathcal{C}_{1,e,e'}(\omega,\omega')\right] \\
&= \mathcal{H}_k(\omega)\mathcal{H}_k(\omega')(w_k^\star)^2 \\
&+ \frac{1}{\nu E}\frac{\mathcal{H}_k(\omega)\mathcal{H}_k(\omega')\mathcal{R}_3(\omega)\mathcal{R}_3(\omega')\mathcal{R}_1(\omega)\mathcal{R}_1(\omega')}{1-\gamma_2(\omega,\omega')}\left[\frac{1}{M}\sum_\ell \lambda_\ell^2 (w_\ell^\star)^2\right] \\
&+ \frac{1}{\nu\alpha EB}\frac{\mathcal{H}_k(\omega)\mathcal{H}_k(\omega')\mathcal{C}_1(\omega,\omega')(i\omega)(i\omega')}{(1-\gamma_2(\omega,\omega'))}\frac{1}{M}\sum_\ell \lambda_\ell \mathcal{H}_\ell(\omega)\mathcal{H}_\ell(\omega') \\
&+ \frac{\lambda_k}{\alpha B}\frac{\mathcal{H}_k(\omega)\mathcal{H}_k(\omega')\mathcal{R}_1(\omega)\mathcal{R}_1(\omega')}{1-\gamma_0(\omega,\omega')}\frac{1}{M}\sum_\ell \lambda_\ell \mathcal{H}_\ell(\omega)\mathcal{H}_\ell(\omega')(w_\ell^\star)^2 \\
&+ \frac{\lambda_k}{\alpha\nu EB}\frac{\mathcal{H}_k(\omega)\mathcal{H}_k(\omega')\mathcal{C}_3(\omega,\omega')}{1-\gamma_0(\omega,\omega')}\frac{1}{M}\sum_\ell \lambda_\ell \mathcal{H}_\ell(\omega)\mathcal{H}_\ell(\omega')
\end{aligned} \tag{112}$$

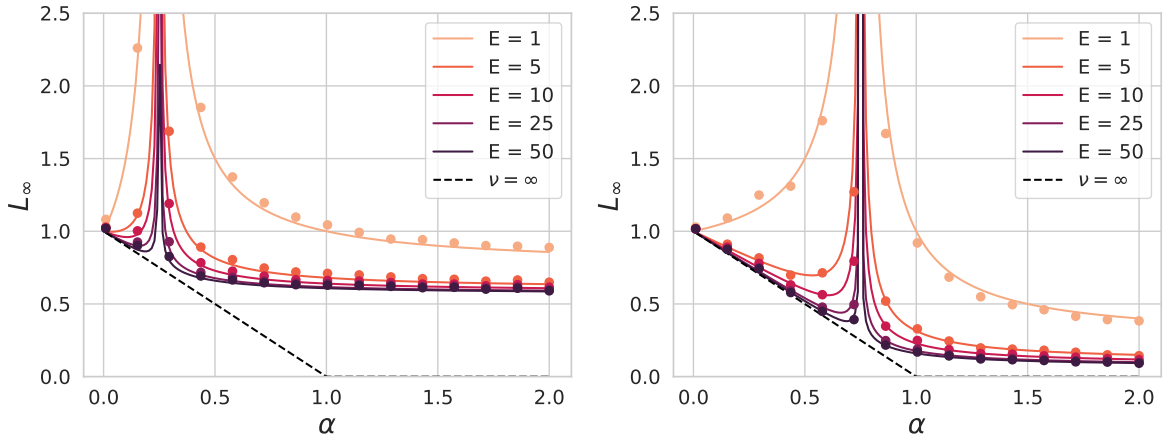
The first term is the irreducible bias for mode k which is the loss for mode k when the learned function is averaged over all possible datasets and all possible projections. We see that the second term scales as $\frac{1}{\nu E}$ which will persist even if $B\alpha \to \infty$. Similarly, there is a term that is order $\frac{1}{\alpha B}$ which will persist even if $\nu E \to \infty$. Lastly, there are two terms which depend on both B, E. This is similar to the variance that is explained by the interaction of the dataset and the random projection (Adlam & Pennington, 2020b). The test loss is then a Fourier transform of the above function

$$\bar{\mathcal{L}}(t) = \frac{1}{M}\sum_k \lambda_k \left\langle \bar{v}_k^0(t)^2\right\rangle. \tag{113}$$

If $E, B \to \infty$, then we obtain the stated *irreducible bias* of the main paper

$$\lim_{E,B\to\infty}\bar{\mathcal{L}}(t) = \frac{1}{M}\sum_k \lambda_k (w_k^\star)^2 H_k(t)^2. \tag{114}$$

This is the error of the mean output function over all possible datasets and random projections of a certain size.



(a) $\nu = 0.25$ (b) $\nu = 0.75$

Figure 9. The infinite time limit of the loss when ensembling with isotropic features $\lambda_k = 1$ recovers prior results on ensembling and double descent (d'Ascoli et al., 2020; Adlam & Pennington, 2020b). There is an overfitting peak (double descent) at $\alpha = \nu$. In the *overparameterized regime* where $\alpha < \nu$, the infinite ensembled model matches the performance of the $\nu \to \infty$ limit. This is because the bias is limited by dataset size rather than model size. In the *underparameterized regime* $\alpha > \nu$, the infinite ensembled model *does not* achieve the loss of the infinite model due to a bias limited by ν.

H.4. Ensembling is Not Always Compute Optimal

For a compute budget $C = NEt$, we find that ensembling does not provide as much benefit as increasing the size of the model. From the results in the last section, we note that ensembling reduces the variance. For this section, we consider the $P \to \infty$ limit. We let $\mathcal{B}(N, t)$ represent the bias and $\mathcal{V}(N, t)$ represent the variance within a single ensemble. The loss at fixed compute then takes the form

$$\mathcal{L}(\nu, C, t) = \mathcal{B}(\nu, t) + \frac{1}{\nu E}\mathcal{V}(\nu, t). \tag{115}$$

For any ν which satisfies the condition that

$$\frac{\partial}{\partial \nu}\mathcal{B}(\nu, t) \leq 0 \ , \ \frac{\partial}{\partial \nu}\mathcal{V}(\nu, t) \leq 0 \tag{116}$$

we have that ensembling is strictly dominated by increasing ν.

I. White Bandlimited Model

To gain intuition for the model, we can first analyze the case where $\lambda_k = 1$, which has a simpler DMFT description since each of the M features are statistically identical. We illustrate the dependence of the loss on model size ν and training time t for $\alpha < 1$ in Figure 10. We note that the loss can be non-monotonic in ν at late training times, but that monotonicity is maintained for optimal early stopping, similar to results on optimal regularization in linear models (Advani et al., 2020) and random feature models (Mei & Montanari, 2022; Simon et al., 2023).

I.1. Derivation

In the case of all $\lambda_k = 1$ we have the following definitions

$$\begin{aligned} \mathcal{R}_1(\omega) &= 1 - \frac{1}{\alpha}\frac{\mathcal{R}_1(\omega)\mathcal{R}_3(\omega)}{i\omega + \mathcal{R}_1(\omega)\mathcal{R}_3(\omega)} \\ \mathcal{R}_3(\omega) &= 1 - \frac{1}{\nu}\frac{\mathcal{R}_1(\omega)\mathcal{R}_3(\omega)}{i\omega + \mathcal{R}_1(\omega)\mathcal{R}_3(\omega)} \end{aligned} \tag{117}$$

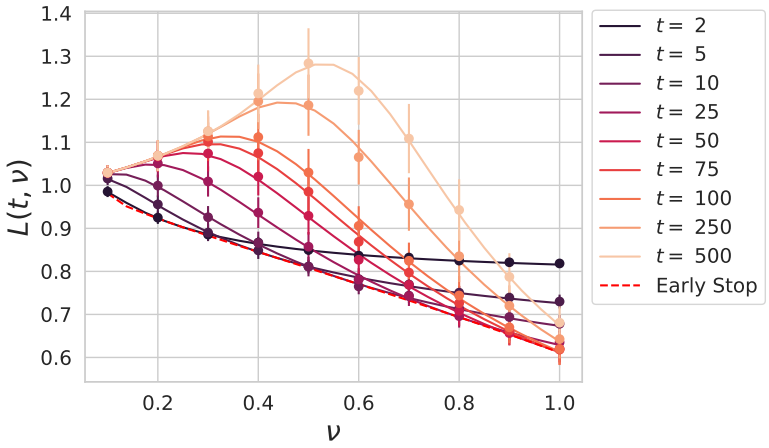


Figure 10. The white bandlimited model ($\lambda_k = 1$) with $\alpha = 0.8$ and varying model size ν with no explicit noise $\sigma = 0$ exhibits double descent at late time. Optimal early stopping, like optimal regularization, recovers monotonic scaling with ν.

Writing $\mathcal{R}_1 = 1 - \frac{\nu}{\alpha}(\mathcal{R}_3 - 1)$ allows us to solve for $\mathcal{R}_3$ exactly:

$$\mathcal{R}_3 \left(i\omega + \mathcal{R}_3(1 + \frac{\nu}{\alpha}(\mathcal{R}_3 - 1)) \right) = i\omega + \mathcal{R}_3(1 + \frac{\nu}{\alpha}(\mathcal{R}_3 - 1)) - \frac{1}{\nu}\left(\mathcal{R}_3(1 + \frac{\nu}{\alpha}(\mathcal{R}_3 - 1)) \right). \tag{118}$$

This is a cubic equation that can be solved for $\mathcal{R}_3$ as a function of ω. In the limit of $\alpha \to \infty$ this simplifies to:

$$\begin{aligned} \mathcal{R}_3 i\omega + \mathcal{R}_3^2 &= i\omega + \left(1 - \frac{1}{\nu}\right)\mathcal{R}_3 \\ \Rightarrow \mathcal{R}_3 &= \frac{1}{2}[(1 - \nu^{-1} - i\omega) + \sqrt{(1 - \nu^{-1} - i\omega)^2 + 4i\omega}]. \end{aligned} \tag{119}$$

I.2. Timescale Corrections in The Small ν Regime

By expanding the above in the limit of small ν we get that $\mathcal{R}_3$ goes as

$$\mathcal{R}_3 \sim \frac{i\omega}{\nu^{-1} - 1 + i\omega}\ ,\ \nu \to 0 \tag{120}$$

From this approximate response function, we find that the transfer function takes the form

$$\begin{aligned} H(\tau) &= \int \frac{d\omega}{2\pi} \frac{e^{i\omega\tau}}{i\omega + \frac{i\omega}{\nu^{-1} - 1 + i\omega}} = \int \frac{d\omega}{2\pi} \frac{(\nu^{-1} - 1 + i\omega)e^{i\omega\tau}}{i\omega\left[\nu^{-1} + i\omega\right]} \\ &= (1 - \nu) + \nu e^{-\tau/\nu}, \end{aligned} \tag{121}$$

where in the last line, we used the residue theorem. We note that in this perturbative approximation that this transfer function is always greater than the transfer function at $\nu \to \infty$ which is $e^{-\tau}$. Thus finite ν leads to higher bias in this regime. We define bias and variance precisely in Appendix H.2.

I.3. Timescale corrections in fully expressive regime $\nu > 1$

For $\nu \gg 1$, we can approximate $R_3(\omega) \sim 1 - \nu^{-1}(1 + i\omega)^{-1}$, we have

$$\begin{aligned} H(\tau) &\sim \int \frac{d\omega}{2\pi} \frac{e^{i\omega\tau}}{i\omega + 1 - \nu^{-1}(i\omega + 1)^{-1}} = \int \frac{d\omega}{2\pi} \frac{e^{i\omega\tau}(1 + i\omega)}{(i\omega + 1 - \nu^{-1/2})(i\omega + 1 + \nu^{-1/2})} \\ &= \frac{1}{2}e^{-\tau(1+\sqrt{\nu})} + \frac{1}{2}e^{-\tau(1-\sqrt{\nu})} = e^{-\tau}\cosh\left(\tau/\sqrt{\nu}\right) \end{aligned} \tag{122}$$

where we used the residue theorem after closing the contour in the upper half-plane. In Figure 11, we show that this perturbative approximation does capture a slowdown in the dynamics for large but finite ν.

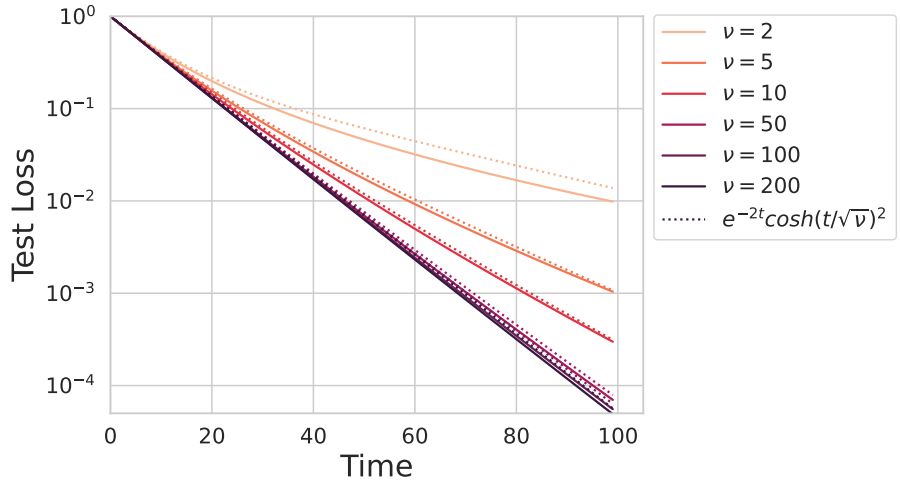


Figure 11. Slower timescales in the $\nu > 1$ regime for white bandlimited features.

J. Power-Law Bottleneck Scalings

In this section we calculate the scaling of the loss with the various limiting resources (time, model size, and data) when using power law features. Since the power-law features give a trace class kernel (*i.e.* $\sum_{k=1}^{\infty} \lambda_k < \infty$), we use the non-proportional limit formalism in Appendix G, which gives an expression for $\mathcal{L}(t, N, P)$ with M already considered infinite. While the resulting expressions are not a formal proportional thermodynamic limit and finite N, P corrections exist in the form of fluctuations from one random realization of the system to another. These corrections decay rapidly enough at finite N, P for this mean field theory to be accurate and descriptive in realistic systems (Bordelon et al., 2020; Simon et al., 2023; Cheng & Montanari, 2022). We plot this variability of random finite size experiments as highlighted standard deviations in the main text figures.

J.1. Time Bottleneck

The time bottleneck is defined as the limiting dynamics in the absence of any model or data finite size effects. To eliminate those effects, we simply study the $\alpha, \nu \to \infty$ limit

$$\mathcal{L}_{\infty}(t) = \lim_{P,N \to \infty} \mathcal{L}(t, P, N). \tag{123}$$

In this limit, the response functions simplify to $\mathcal{R}_1(\omega)\mathcal{R}_3(\omega) \to 1$ so that

$$\mathcal{H}_k(\omega) = \frac{1}{i\omega + \lambda_k} \implies H_k(\tau) = e^{-\lambda_k \tau}\Theta(\tau). \tag{124}$$

Further, in this limit, we have that $C_0(t, s) = \frac{1}{M}\sum_k \lambda_k H_k(t) H_k(s) (w_k^\star)^2$ since all the variance terms (which depend on ν^{-1}, α^{-1}) drop out. Thus we have the following loss at time t,

$$\mathcal{L}(t) = \sum_k \lambda_k (w_k^\star)^2 e^{-2\lambda_k t} \sim \int_1^\infty dk k^{-a} \exp\left(-2k^{-b}t\right) \sim t^{-(a-1)/b}. \tag{125}$$

where the final scaling with time can be obtained through either change of variables or steepest descent methods (Bordelon & Pehlevan, 2022a).

J.2. Model Bottleneck

In this section we take $\alpha, t \to \infty$. This leaves us with the following equation for $r \equiv \lim_{\omega\to 0}(i\omega)^{-1}\mathcal{R}_3(\omega)$.

$$N = \sum_k \frac{\lambda_k r}{\lambda_k r + 1} \approx \int_1^\infty \frac{dk}{k^b/r + 1} \approx r^{1/b} \implies r \approx N^b. \tag{126}$$

Now, the large time limit of the transfer functions $H_k(\tau)$ can be obtained from the final-value theorem

$$\lim_{t\to\infty} H_k(\tau) = \lim_{\omega\to 0} \frac{i\omega}{i\omega + \lambda_k r i\omega} = \frac{1}{1+\lambda_k r}. \tag{127}$$

Now, integrating over the eigenvalue density to get the total loss (and disregarding prefactors)

$$\begin{aligned}
\mathcal{L}(t) &\sim \int_1^\infty dk\, \frac{k^{-a}}{(1+k^{-b}r)^2} \\
&\approx \frac{1}{r^2}\int_1^N dk\, k^{2b-a} + \int_N^\infty k^{-a} \\
&= \frac{1}{2b+1-a}[N^{-(a-1)} - N^{-2b}] + \frac{1}{a-1}N^{-(a-1)} \\
&\sim N^{-\min\{a-1,2b\}}
\end{aligned} \tag{128}$$

For difficult tasks where $a - 1 < 2b$, we thus expect a powerlaw scaling of the form $\mathcal{L} \sim N^{-(a-1)}$ in this regime.

J.3. Data Bottleneck

In this section we take $\nu, t \to \infty$. This leaves us with the following equation for $r \equiv \lim_{\omega\to 0}(i\omega)^{-1}\mathcal{R}_1(\omega)$.

$$P = \sum_k \frac{\lambda_k r}{\lambda_k r + 1} \approx \int_1^\infty \frac{dk}{k^b/r + 1} \approx r^{1/b} \implies r \approx P^b. \tag{129}$$

Now, the large time limit of the transfer functions $H_k(\tau)$ can again be obtained from the final-value theorem

$$\lim_{t\to\infty} H_k(\tau) = \lim_{\omega\to 0} \frac{i\omega}{i\omega + \lambda_k r i\omega} = \frac{1}{1+\lambda_k r} \tag{130}$$

Now, integrating over the eigenvalue density to get the total loss gives

$$\begin{aligned}
\mathcal{L}(t) &\sim \int_1^\infty dk\, \frac{k^{-a}}{(1+k^{-b}r)^2} \\
&\approx \frac{1}{r^2}\int_1^P dk\, k^{2b-a} + \int_P^\infty k^{-a} \\
&= \frac{1}{2b+1-a}[P^{-(a-1)} - P^{-2b}] + \frac{1}{a-1}P^{-(a-1)} \\
&\sim P^{-\min\{a-1,2b\}}
\end{aligned} \tag{131}$$

For difficult tasks with $a < 2b + 1$, the loss will therefore scale as $P^{-(a-1)}$ in this data-bottleneck regime.

K. Optimization Extensions

K.1. Discrete Time

In this section, we point out that DMFT can also completely describe discrete time training as well. In this section we consider discrete time gradient descent with learning rate η

$$\begin{aligned}
\boldsymbol{v}^0(t+1) &= \boldsymbol{v}^0(t) - \eta \boldsymbol{v}^4(t) \\
\boldsymbol{v}^4(t) &= \frac{1}{\nu\sqrt{M}}\boldsymbol{A}^\top \boldsymbol{v}^3(t)\ ,\ \boldsymbol{v}^3(t) = \frac{1}{\sqrt{M}}\boldsymbol{A}\boldsymbol{v}^2(t) \\
\boldsymbol{v}^2(t) &= \frac{1}{\alpha\sqrt{M}}\boldsymbol{\Psi}^\top \boldsymbol{v}^1(t)\ ,\ \boldsymbol{v}^1(t) = \frac{1}{\sqrt{M}}\boldsymbol{\Psi}\boldsymbol{v}^0(t)
\end{aligned} \tag{132}$$

Following either the MSR or cavity derivation, we obtain an analgous set of limiting DMFT equations defined for integer times $t \in \mathbb{Z}$,

$$\begin{aligned} v_k^0(t+1) &= v_k^0(t) - \eta v_k^4(t) + \delta(t+1) w_k^\star \\ v^1(t) &= u^1(t) + \alpha^{-1} \sum_s R_{0,2}(t,s) v^1(s) \\ v_k^2(t) &= u_k^2(t) + \sum_s R_1(t,s) v_k^0(s) \\ v^3(t) &= u^3(t) + \sum_s R_{2,4}(t,s) v^3(s) \\ v_k^4(t) &= u_k^4(t) + \sum_s R_3(t,s) v_k^2(s) \end{aligned} \tag{133}$$

The delta function in this context is defined as

$$\delta(t+1) = \begin{cases} 1 & t = -1 \\ 0 & \text{else} \end{cases} \tag{134}$$

ensures that the initial condition $v_k^0(0) = w_k^\star$ is satisfied. These iteration equations can be closed for the response functions and correlation functions and solved over $T \times T$ matrices.

Alternatively, we can also solve this problem in an analogous frequency space. Analogous to the Fourier transform method, the equations in discrete time can be closed in terms of the Z-transform

$$v(z) = \sum_{t=-\infty}^{\infty} z^{-t} v(t) \tag{135}$$

Applying this transform gives us the following expression for the v_k^0 fields.

$$v_k^0(z) = \frac{z w_k^\star - \eta u_k^4(z) - \eta \mathcal{R}_3(z) u_k^2(z)}{z - 1 + \eta \lambda_k \mathcal{R}_1(z) \mathcal{R}_3(z)} \equiv \mathcal{H}_k(z) \left[z w_k^\star - \eta u_k^4(z) - \eta \mathcal{R}_3(z) u_k^2(z) \right] \tag{136}$$

Similar to the Fourier case, the final losses can be extracted as the $z \to 1$ limit of these objects.

K.2. Momentum

As mentioned in appendix B, it is straightforward to extend the DMFT treatment beyond just gradient descent dynamics to include a momentum term with momentum β.

We first consider this replacement in continuous time. This requires applying the following replacement:

$$\partial_t v_k^0(t) = -v_k^4(t) \to (\beta \partial_t^2 + \partial_t) v_k^0(t) = -v_k^4(t). \tag{137}$$

This slightly modifies the expressions for the response functions. For example, in Fourier space the response functions become:

$$\begin{aligned} \mathcal{R}_{0,2}(\omega) &= -\frac{1}{M} \sum_k \frac{\lambda_k}{\epsilon + i\omega + \beta(i\omega)^2 + \lambda_k \mathcal{R}_1(\omega) \mathcal{R}_3(\omega)} \mathcal{R}_3(\omega) \\ \mathcal{R}_{2,4}(\omega) &= -\frac{1}{M} \sum_k \frac{\lambda_k}{\epsilon + i\omega + \beta(i\omega)^2 + \lambda_k \mathcal{R}_1(\omega) \mathcal{R}_3(\omega)} \mathcal{R}_1(\omega). \end{aligned} \tag{138}$$

In discrete time, momentum updates can be expressed as

$$\begin{aligned} \boldsymbol{v}^0(t+1) &= \boldsymbol{v}^0(t) - \eta \boldsymbol{b}(t) \\ \boldsymbol{b}(t) &= \boldsymbol{v}^4(t) + \mu \boldsymbol{b}(t-1) \end{aligned} \tag{139}$$

where $\boldsymbol{b}(t)$ is the filtered version of the loss gradient (the $\boldsymbol{v}^4(t)$ field) with momentum coefficient μ and η is the learning rate. The dependence on the $\boldsymbol{b}(t)$ field can be eliminated by turning this into a second order difference equation

$$\boldsymbol{v}^0(t+1) - \boldsymbol{v}^0(t) - \mu\left(\boldsymbol{v}^0(t) - \boldsymbol{v}^0(t-1)\right) = -\eta \boldsymbol{v}^4(t). \tag{140}$$

Again, the final result can be expressed in terms of the Z-transformed transfer functions $\mathcal{H}_k(z)$ which have the form

$$\mathcal{H}_k(z) = \frac{1}{z - 1 - \mu + \mu z^{-1} + \eta \mathcal{R}_1(z)\mathcal{R}_3(z)}. \tag{141}$$

K.3. One Pass SGD

In this section we derive online SGD with projected features. At each step a random batch of B samples are collected (independent of previous samples), giving a matrix $\boldsymbol{\Psi}(t) \in \mathbb{R}^{B\times M}$ of sampled features. The update at step t is

$$\boldsymbol{v}^0(t+1) = \boldsymbol{v}^0(t) + \eta \left(\frac{1}{N}\boldsymbol{A}^\top \boldsymbol{A}\right)\left(\frac{1}{B}\boldsymbol{\Psi}(t)^\top \boldsymbol{\Psi}(t)\right)\boldsymbol{v}^0(t). \tag{142}$$

The DMFT limit gives the following statistical description of the fields, which decouple over time for the $v^1(t), v_k^2(t)$ but remain coupled across time for $v^3(t), v_k^4(t)$

$$\begin{aligned}
v^1(t) &= u^1(t)\ ,\ u^1(t) \sim \mathcal{N}(0, C_0(t,t)\delta(t-s)),\\
v_k^2(t) &= u_k^2(t) + \lambda_k v_k^0(t)\ ,\ u_k^2(t) \sim \mathcal{N}\left(0, \frac{1}{B}\lambda_k C_1(t,t)\delta(t-s)\right)\\
v^3(t) &= u^3(t) + \frac{1}{N}\sum_s R_{2,4}(t,s)v^3(s)\ ,\ u^3(t) \sim \mathcal{N}(0, C_2(t,s))\\
v_k^4(t) &= u_k^4(t) + \sum_s R_3(t,s)v_k^2(s)\ ,\ u_k^4(t) \sim \mathcal{N}\left(0, \frac{1}{N}C_3(t,s)\right)\\
v_k^0(t+1) &= v_k^0(t) - \eta v_k^4(t).
\end{aligned} \tag{143}$$

This system cannot exhibit overfitting effects as we have the statistical equivalence between the covariance of v^1 and the test loss:

$$\hat{\mathcal{L}}(t) = \left\langle v^1(t)^2 \right\rangle = \left\langle u^1(t)^2 \right\rangle = C_0(t,t) = \mathcal{L}(t) \tag{144}$$

We note that this is very different than the case where data is reused at every step, which led to a growing gap between train and test loss as we derive in Appendix E.

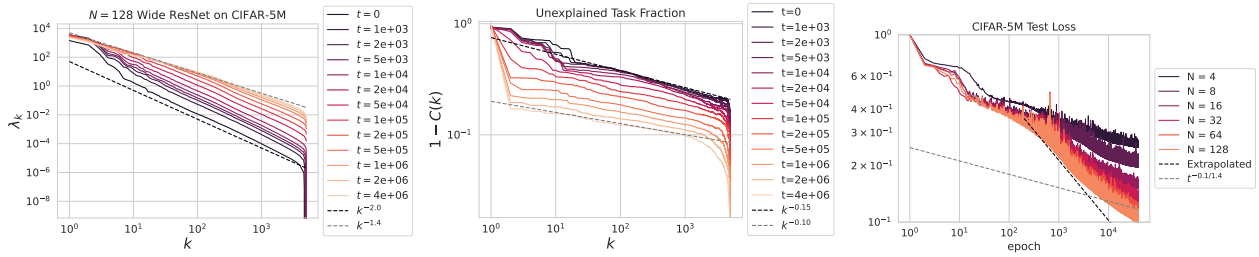
We visualize some example results for one-pass SGD with power law features in Figure 3. While we see that the same scaling laws with t and N hold, the dependence on batchsize B is much weaker: the model never reaches an asymptote that scales with B but rather experiences SGD noise that scales with η/B for learning rate η.

We summarize the key similarities and differences between the one-pass SGD and multi-pass batch GD settings

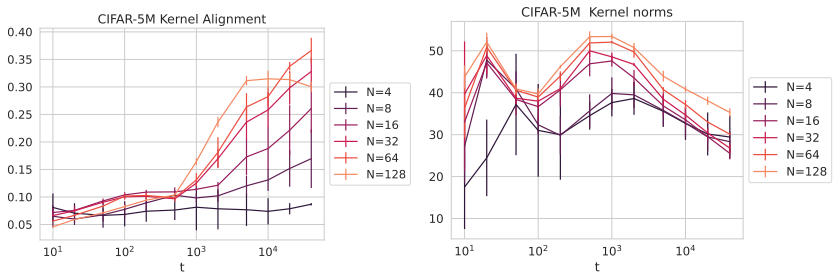
1. If the learning rate is small and a continuous time limit of the dynamics is taken, then the SGD dynamics will agree with the $P \to \infty$ limit of our full batch gradient flow theory. This is a setting where finite data and SGD noise are negligible.
2. If learning rate is non-negligible and batch size is finite, then SGD noise cannot be neglected and the SGD dynamics will be different than full pass GD. The SGD dynamics will be described by a discrete time DMFT given above.
3. In general, the multi-pass version of the theory can have a train loss and test loss gap while the SGD theory never has a gap between training and test loss.
4. The SGD test loss can be limited by t, N, but the effect of finite batch size is basically some additive variance in the model outputs. Finite dataset size in the full batch GD can lead to a bottleneck scaling law (like $L \sim P^{-(a-1)}$).

L. Kernel Analysis of Feature Learning Networks

In Section 5.1, we observed that feature learning networks can achieve better loss and compute-optimal scaling. In such settings, it may be useful to observe the *after kernel*, namely the NTK at the end of training. This object can often shed



(a) Final NTK Spectra for Varying Widths (b) Final NTK Task-Power Decay (c) Losses for Feature Learning Networks



(d) (e)

Figure 12. a) The observed power law spectrum on a held out test set of the after-kernel for a width $N = 128$ ResNet trained on CIFAR-5m. Early on in training, the spectrum flattens quite rapidly. At later times, the spectral decay remains relatively constant. b) The fraction of the task unexplained, as defined in Equation 20. Throughout training, the top eigenmode of the after-kernel explains more and more of the task. c) The test loss of the network. We see that the observed scaling of this quantity is faster than that predicted from analyzing the after-kernel. d) The kernel-target alignment of the after kernel improves throughout training time. The error bars here denote different ensemble members. Their relatively small size implies that the kernel trajectory is relatively deterministic over different initialization seeds. e) The norm of the after-kernel throughout training is relatively constant for this task.

insight into the structure of the learned network function (Atanasov et al., 2022; Long, 2021) and its generalization. In some cases, it has been observed that the final kernel stabilizes during the course of training (Fort et al., 2020), potentially allowing one to potentially deduce scaling laws from the spectrum and task-model alignment of this after-kernel, though other papers have observed contrary results (Vyas et al., 2022).

Motivated by this, we study the NTKs of the finite-width networks trained for 64 epochs with the animate-inanimate CIFAR-5m discrimination task. We observe in Figure 12 a) that the spectrum becomes flatter, with a decay exponent of close to 1.4 down from 2.0 for the initial kernel.

The fraction of the task power unexplained is also observed to have a lower exponent in Figure 12 b), however there is also the presence of a low rank spike indicative of the kernel aligning to this discrimination tasks.

From these scalings we can obtain the a and b exponents and get a prediction for the scaling of the test loss. We plot this in grey in Figure 12 c). The observed scaling (in black) is much better than that predicted by the after-kernel. This is an indication the the after kernel continues evolving in this task, improving the scaling exponent of the test loss.

The kernel-target alignment (Cortes et al., 2012), as measured by

$$A = \frac{\boldsymbol{y}^\top \boldsymbol{K} \boldsymbol{y}}{\boldsymbol{y}^\top \boldsymbol{y} |\boldsymbol{K}|_F}, \tag{145}$$

is plotted in 12 d). Here $\boldsymbol{y}$ is the target labels on a held-out test set, and $\boldsymbol{K}$ is the gram matrix of the after-kernel on this test set. We indeed observe a consistent increase in this quantity across time. This gives an indication that understanding the evolution of the after-kernel will be useful

M. Numerical Recipes

M.1. Iteration of DMFT Equations on Time $\times$ Time matrices

The simplest way to solve the DMFT equations is to iterate them from a reasonable initial condition (Mignacco et al., 2020; Bordelon & Pehlevan, 2022b). We solve in discrete time for $T \times T$ matrices $\{\boldsymbol{R}_{0,2}, \boldsymbol{R}_1, \boldsymbol{R}_{2,4}, \boldsymbol{R}_3, \boldsymbol{C}_0, \boldsymbol{C}_1, \boldsymbol{C}_2, \boldsymbol{C}_3\}$ which have entries $[\boldsymbol{R}]_{t,s} = R(t,s), [\boldsymbol{C}]_{t,s} = C(t,s)$, etc. We let $\boldsymbol{\Theta}(t,s) = \eta\Theta(t-s)$ where η is the learning rate.

1. Solve for the response functions by updating the closed equations as matrices by iterating the equations.

$$\begin{aligned}
\boldsymbol{R}_{0,2,k} &\leftarrow -\left[\boldsymbol{\Theta}^{-1} + \lambda_k \boldsymbol{R}_3 \boldsymbol{R}_1\right]^{-1} \boldsymbol{R}_3, \\
\boldsymbol{R}_{0,2} &\leftarrow \frac{1}{M}\sum_k \lambda_k \boldsymbol{R}_{0,2,k}, \\
\boldsymbol{R}_1 &\leftarrow \left[\boldsymbol{I} - \alpha^{-1}\boldsymbol{R}_{0,2}\right]^{-1}, \\
\boldsymbol{R}_{2,4,k} &\leftarrow -\lambda_k \left[\boldsymbol{I} + \lambda_k \boldsymbol{R}_1 \boldsymbol{\Theta} \boldsymbol{R}_3\right]^{-1} \boldsymbol{R}_1 \boldsymbol{\Theta}\ ,\ \boldsymbol{R}_{2,4} = \frac{1}{M}\sum_k \boldsymbol{R}_{2,4,k}, \\
\boldsymbol{R}_3 &\leftarrow \left[\boldsymbol{I} - \nu^{-1}\boldsymbol{R}_{2,4}\right]^{-1}.
\end{aligned} \tag{146}$$

2. Once these response functions have converged, we can iterate the equations for the correlation functions

$$\begin{aligned}
\boldsymbol{C}_{0,k} &\leftarrow \left[\boldsymbol{I} + \lambda_k \boldsymbol{\Theta}\boldsymbol{R}_3\boldsymbol{R}_1\right]^{-1}\left[(w_k^\star)^2 \mathbf{1}\mathbf{1}^\top + \boldsymbol{\Theta}\left(\nu^{-1}\boldsymbol{C}_3 + \frac{\lambda_k}{\alpha}\boldsymbol{R}_3\boldsymbol{C}_1\boldsymbol{R}_3^\top\right)\boldsymbol{\Theta}^\top\right]\left[\boldsymbol{I} + \lambda_k\boldsymbol{\Theta}\boldsymbol{R}_3\boldsymbol{R}_1\right]^{-1\top}, \\
\boldsymbol{C}_0 &\leftarrow \frac{1}{M}\sum_{k=1}^M \lambda_k \boldsymbol{C}_{0,k}, \\
\boldsymbol{C}_1 &\leftarrow \boldsymbol{R}_1\boldsymbol{C}_0\boldsymbol{R}_1^\top, \\
\boldsymbol{C}_{2,k} &\leftarrow \left[\boldsymbol{I} + \lambda_k\boldsymbol{R}_1\boldsymbol{\Theta}\boldsymbol{R}_3\right]^{-1}\left(\frac{\lambda_k}{\alpha}\boldsymbol{C}_1 + \boldsymbol{R}_1\left[(w_k^\star)^2\lambda_k^2\mathbf{1}\mathbf{1}^\top + \frac{\lambda_k^2}{\nu}\boldsymbol{\Theta}\boldsymbol{C}_3\boldsymbol{\Theta}^\top\right]\boldsymbol{R}_1^\top\right)\left[\boldsymbol{I} + \lambda_k\boldsymbol{R}_1\boldsymbol{\Theta}\boldsymbol{R}_3\right]^{-1\top}, \\
\boldsymbol{C}_2 &\leftarrow \frac{1}{M}\sum_k \boldsymbol{C}_{2,k}.
\end{aligned} \tag{147}$$

After iterating these equations, one has the discrete time solution to the DMFT order parameters and any other observable can then be calculated.

M.2. Fourier Transform Method

To accurately compute the Fourier transforms in the model/data bottleneck regime ($\alpha < 1$ or $\nu < 1$) we have that $\mathcal{R}_1(\omega)\mathcal{R}_3(\omega) \sim i\omega r$ as $\omega \to 0$ so we must resort to analyzing the principal part and the delta-function contribution to the integral. Construct a shifted and non-divergent version of the function $\mathcal{H}(\omega)$.

$$\begin{aligned}
\mathcal{H}(\omega) &= \tilde{\mathcal{H}}(\omega) + \frac{1}{\epsilon + i\omega(1+r)} \\
\tilde{\mathcal{H}}(\omega) &= \frac{1}{\epsilon + i\omega + \mathcal{R}_1(\omega)\mathcal{R}_3(\omega)} - \frac{1}{\epsilon + i\omega(1+r)} = \frac{i\omega r - \mathcal{R}_1(\omega)\mathcal{R}_3(\omega)}{(\epsilon + i\omega + \mathcal{R}_1(\omega)\mathcal{R}_3(\omega))(\epsilon + i\omega(1+r))},
\end{aligned} \tag{148}$$

where $r = \lim_{\omega\to 0}\frac{1}{i\omega}\mathcal{R}_1(\omega)\mathcal{R}_3(\omega)$. We see that rather than diverging like $\mathcal{H}(\omega)$, this function $\tilde{\mathcal{H}}(\omega)$ vanishes as $\omega \to 0$. We therefore numerically perform Fourier integral against $\tilde{H}(\omega)$ and then add the singular component which can be computed separately.

$$H(\tau) = \int \frac{d\omega}{2\pi} e^{i\omega\tau}\tilde{\mathcal{H}}(\omega) + \int \frac{d\omega}{2\pi}\frac{e^{i\omega\tau}}{\epsilon + i\omega(1+r)} = \int \frac{d\omega}{2\pi} e^{i\omega\tau}\tilde{\mathcal{H}}(\omega) + \frac{1}{1+r} \tag{149}$$

where we used the fact that

$$\frac{1}{\epsilon + i\omega(1+r)} = \frac{\pi}{1+r}\delta(\omega) - \frac{i}{1+r}\mathcal{P}(\omega^{-1})\ ,\ \epsilon \to 0 \tag{150}$$

The Dirac mass is trivial to integrate over giving $\frac{1}{2(1+r)}$. Lastly, we must perform an integral of the type

$$-\frac{i}{1+r}\mathcal{P}\int_{-\infty}^{\infty}\frac{d\omega}{2\pi}\frac{e^{i\omega\tau}}{\omega}=\frac{1}{\pi(1+r)}\int_{0}^{\infty}d\omega\frac{\sin(\omega\tau)}{\omega}=\frac{1}{2(1+r)} \tag{151}$$

Adding these two terms together, our transfer function has the form

$$H(\tau)=\frac{1}{1+r}+\int_{-\infty}^{\infty}\frac{d\omega}{2\pi}e^{i\omega\tau}\tilde{\mathcal{H}}(\omega). \tag{152}$$

The last integral can be performed numerically, giving a more stable result.

N. Compute Optimal Scaling from Sum of Power-Laws

We suppose that the loss scales as (neglecting irrelevant prefactors)

$$\mathcal{L}=t^{-r_t}+N^{-r_N}+P^{-r_P}+\mathcal{L}_\infty \tag{153}$$

Our goal is to minimize the above expression subject to the constraint that compute $C=Nt$ is fixed. Since C is fixed we can reduce this to a one-dimensional optimization problem

$$\min_N\left[C^{-r_t}N^{r_t}+N^{-r_N}\right] \tag{154}$$

The optimality condition $\partial_N L=0$ is

$$\begin{aligned}&r_tC^{-r_t}N^{r_t-1}-r_NN^{-r_N-1}=0\\&\implies N\propto C^{\frac{r_t}{r_t+r_N}}\implies t\propto C^{\frac{r_t}{r_t+r_N}}\end{aligned} \tag{155}$$

From this last expression one can evaluate the loss at the optimum

$$\mathcal{L}_\star(C)\propto C^{-\frac{r_tr_N}{r_t+r_N}}. \tag{156}$$