# Decoupled Descent: Exact Test Error Tracking Via Approximate Message Passing

**Max Lovig**
Statistics and Data Science
Yale University
New Haven, CT 06511
max.lovig@yale.edu

## Abstract

In high dimensional model training, full-batch gradient descent (and its variants) suffers due to progressively stronger biasing towards the exact realization of training data; this drives the systematic "generalization gap", where the train error becomes an unreliable proxy for test error. Existing approaches either argue this gap is benign through complex analysis or sacrifice data to a validation set. In contrast, we introduce decoupled descent (DD), a novel theory-based training algorithm that satisfies a train-test identity—enforcing the train error to asymptotically track the test error for stylized Gaussian mixture models, even when the underlying loss is non-convex. Within this specific regime, leveraging approximate message passing theory, DD iteratively cancels the biases due to data reuse, rigorously demonstrating the feasibility of zero-cost validation and $100\%$ data utilization. Moreover, DD is governed by a low-dimensional state evolution recursion, rendering the dynamics of the algorithm transparent and tractable. We validate DD on XOR classification, yielding superior performance compared to GD; additionally, we implement noisy MNIST and non-linear probing of CIFAR-10, demonstrating that even when our stylized assumptions are relaxed, DD narrows the generalization gap compared to GD.

## 1 Introduction

Consider training data $(X, y) = (x_i, y_i)_{i\in[n]} \in \mathbb{R}^{n\times d}$ where $x_i \in \mathbb{R}^d$ and $y_i \in \mathbb{R}$ are assumed to be i.i.d. $(x_i, y_i) \sim \mathbb{P}_{x,y}$. We model this data using a parametric function $\mathcal{M}_\beta : \mathbb{R}^d \to \mathbb{R}$ with parameters $\beta \in \mathbb{R}^p$ under the loss function $\mathcal{L}(\mathcal{M}_\beta(x), y) : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$. We learn $\beta$ by descending the objective

$$L(\beta) = \frac{1}{n}\sum_{i=1}^{n} \mathcal{L}(\mathcal{M}_\beta(x_i), y_i), \tag{1.1}$$

using some variant of gradient descent (GD). The desired goal, however, is not to directly minimize (1.1) but to minimize the *test error*, i.e. with $(\check{x}, \check{y}) \sim \mathbb{P}_{x,y}$, minimize

$$\check{L}(\beta) = \mathbb{E}_{\check{x},\check{y}}\left[\mathcal{L}(\mathcal{M}_\beta(\check{x}), \check{y})\right]. \tag{1.2}$$

In the classical regime of $n \to \infty$ with $d, p$ fixed, $L(\beta)$ is consistent for $\check{L}(\beta)$. Unfortunately, in the high-dimensional regime of $n, d \to \infty$ with $n \asymp d \asymp p$, the convergence $L(\beta) \to \check{L}(\beta)$ fails.

**Example 1.1.** *Consider high-dimensional linear regression, for $n$ i.i.d. samples $x_i \sim \mathcal{N}(0, \mathrm{Id}_d/d)$ with labels $y_i = 0$. We fit $\mathcal{M}_\beta(x) = x^\top\beta$ for $\beta \in \mathbb{R}^d$ under MSE loss $\mathcal{L}(\hat{y}, y) = (\hat{y} - y)^2 = (x^\top\beta)^2$. Initializing $\beta_1$ independent of $(X, y)$ and setting $M = X^\top X$, the GD update $\beta_2 = (\mathrm{Id} - \eta X^\top X)\beta_1$*

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

*has train* (1.1) *and test* (1.2) *errors,*

$$L(\beta_2) = \beta_1^\top(\mathrm{Id} - \eta M)M(\mathrm{Id} - \eta M)\beta_1/n, \quad \check{L}(\beta_2) = \beta_1^\top(\mathrm{Id} - \eta M)(\mathrm{Id} - \eta M)\beta_1/d.$$

*If $n \to \infty$ with $d$ bounded then $M/n \to \mathrm{Id}_d/d$ almost surely so $L(\beta_2) \to \check{L}(\beta_2)$; however, when $n, d \to \infty$ with $n \asymp d$, $M$ no longer concentrates to the identity and thus $L(\beta_2) - \check{L}(\beta_2) \not\to 0$.*

Consequently, in high-dimensional asymptotic regimes, a decrease in the training objective for GD may not represent a decrease in test error, termed the generalization gap [1]. Moreover, this gap is not necessarily fixed by stochastic variants [21]. Previous work on GD must either justify that this mismatch is benign or sacrifice a portion of data used for training to estimate the test error performance. In this work, we construct a class of algorithms termed decoupled descent (DD). **For a set of stylized high-dimensional problems DD satisfies a train-test identity—any mismatch between** (1.1) **and** (1.2) **is forbidden**. Consequently, this method allows for validation without sacrificing any training data.

**Theorem 1.2** (Informal version of Theorem 3.8)**.** *For $n$* i.i.d. *draws $(x_i)_{i\in[n]}$ from a $d$-dimensional Gaussian mixture with finitely many modes and responses $(y_i)_{i\in[n]}$, any DD iterate $\beta_t$ satisfies*

$$\textbf{Training Error}(\beta_t) - \textbf{Test Error}(\beta_t) \to 0, \quad \textit{almost surely as } n, d \to \infty \textit{ with } n/d \to \alpha.$$

As a benefit of Theorem 1.2, DD enables **zero-cost validation**, meaning the training iterate with the lowest train error also has the lowest test error asymptotically. By eliminating the need for the validation set, users achieve 100% data utilization. Moreover, techniques such as hyperparameter tuning and early stopping [18] on the train error immediately translate performance improvements to the population-level test error. Although we do not uniformly prove that DD outperforms GD in terms of test error, empirically DD does improve or match the performance of GD after proper tuning.

DD originates from a carefully designed approximate message passing (AMP) algorithm [9, 13, 19]. Thus, DD admits a tractable low-dimensional law for the train and test error. This permits a simple physical interpretation for DD's dynamics and gives rigorous asymptotic guarantees on the behavior of the test error, enabling a principled way to implement descent algorithms. While modern practice uses stochastic gradients, eliminating the full-batch generalization gap is a fundamental first step.

To conclude this section, we outline our contributions below.

- We introduce a family of training algorithms, termed *decoupled descent*, for a set of stylized learning problems with Gaussian mixture data, parametric models and general loss functions, covering a wide range of supervised learning tasks.
- In contrast to GD, where the training error is a biased proxy that worsens over time, we show that parameters trained by any DD algorithm enforce the train error to asymptotically equal the test error at each iterate.
- We derive a sequence of low-dimensional recursions that track the algorithm's dynamical trajectory. This provides insight into how DD algorithms explicitly control the test error.
- We illustrate the effectiveness of DD using various applications:
  - **XOR Classification:** Validates DD under stylized model assumptions and demonstrates better test error performance compared to GD.
  - **Noisy MNIST (0 vs. 8):** Demonstrates that DD is robust to different noise distributions and that the train-test identity is consistent across varying ratios of parameter count to data dimension.
  - **CIFAR-10 Probing:** Applies DD to (possibly whitened) ResNet-18 embeddings. Although the train-test identity is not exact, DD narrows the generalization gap compared to GD despite the lack of Gaussian structure.

We provide an overview of the notation for this paper in Section A.1.

**Related Literature:** AMP was previously applied to analyze the stationary points of training error minimization for generalized linear models [13] and describe the performance of M-estimation [8] in high-dimensional models. In contrast to designing AMP algorithms which converge to known estimators, DD is a novel method of parameter optimization. Additionally, when $n \asymp d$ with Gaussian data, dynamical mean field theory [28] is a related technique to AMP which can analyze the macroscopic trajectory of GD. This analysis isolates a "response" term involving the correlation between model parameters at different training times. DD purposefully removes this response term,

simplifying the analysis of the optimization trajectory (see Remark 3.9 for more details). Finally, although DD's method of 100% data utilization in multi-pass training is novel, Theorem 1.2 is immediate in the online learning setting where each data point is utilized only once in training.

## 2 The Train-Test Identity And Its Algorithmic Consequences

### 2.1 Correcting Full-batch GD

Let the $i$-th row (or element) of $X \in \mathbb{R}^{n\times d}$ and $y \in \mathbb{R}^n$ have $(x_i, y_i) \sim \mathbb{P}_{x,y}$ where $\mathbb{P}_x$ is a Gaussian mixture with modes $\mu_1, \dots, \mu_J$; conditioned on mode $j$, we have $y_i \sim \mathbb{P}_j$ independent of the Gaussian realization (see (3.1) for more details). Consider parameter $\beta = (\theta, a)$ where $\theta \in \mathbb{R}^{d\times L}$ and $a \in \mathbb{R}^{L'}$ (with $L, L'$ bounded) alongside a parametric function $\mathcal{M}_{\theta,a}(x) = \mathcal{M}_a(x^\top\theta)$ with $\mathcal{M}_a : \mathbb{R}^L \to \mathbb{R}$. We use the following shorthand.

**Definition 2.1.** Given a parametric model $\mathcal{M}_{\theta,a}(x) = \mathcal{M}_a(x^\top\theta) = \mathcal{M}_a(h)$ and loss function $\mathcal{L}$, let $\Psi(h, y, a) = \mathcal{L}(\mathcal{M}_a(h), y)$ and further define its derivatives, $\nabla_h\Psi(h, y, a) : \mathbb{R}^L \times \mathbb{R} \times \mathbb{R}^{L'} \to \mathbb{R}^L$, $\nabla_a\Psi(h, y, a) : \mathbb{R}^L \times \mathbb{R} \times \mathbb{R}^{L'} \to \mathbb{R}^{L'}$, and Hessian $\nabla_h^2\Psi(h, y, a) : \mathbb{R}^L \times \mathbb{R} \times \mathbb{R}^{L'} \to \mathbb{R}^{L\times L}$. Moreover, when $h \in \mathbb{R}^{n\times L}$ and $y \in \mathbb{R}^n$, each of the above functions is its corresponding row-wise application, for example $\Psi(h, y, a)_i = \Psi(h_i, y_i, a)$.

We then run full-batch GD, depending on learning rate parameters $\eta, \gamma > 0$, by the iteration

$$h_t = X\theta_t,\ \hat{h}_t = \nabla_h\Psi\Big(h_t, y, a_t\Big),\ \theta_{t+1} = \theta_t - \eta X^\top\hat{h}_t,\ a_{t+1} = a_t - \frac{\gamma}{n}\sum_{i=1}^n \nabla_a\Psi(h_{t,i}, y_i, a_t). \quad (2.1)$$

As explained in the introduction, this algorithm produces iterates $\theta_t, a_t$ where the training error does not track the test error. We correct this behavior using the following *pure decoupled descent* iteration,

$$h_t = X\theta_t + \eta\sum_{s=1}^{t-1}\hat{h}_s, \quad \hat{h}_t = \nabla_h\Psi\Big(h_t, y, a_t\Big),$$
$$\theta_{t+1} = \theta_t - \eta\Big(X^\top\hat{h}_t - \alpha\theta_t\Big(\frac{1}{n}\sum_{i=1}^n\nabla_h^2\Psi\Big(h_{t,i}, y_i, a_t\Big)\Big)^\top\Big), \quad a_{t+1} = a_t - \frac{\gamma}{n}\sum_{i=1}^n\nabla_a\Psi\Big(h_{t,i}, y_i, a_t\Big). \quad (2.2)$$

DD adds the correction terms $\eta\sum_{s=1}^{t-1}\hat{h}_s$, $\eta\alpha\left(\frac{1}{n}\sum_{i=1}^n\nabla_h^2\Psi\Big(h_{t,i}, y_i, a_t\Big)\right)$ applied to the pre-activation $X\theta_t$ and gradient of $\theta_t$, respectively. These terms cancel correlations due to reuse of data matrix $X$, avoiding the train-test disconnect that GD suffered from in the introduction. In Appendix A.2, we analyze both the first two GD and DD iterations in Example 1.1 using random matrix theory tools.

### 2.2 The Train-Test Identity

Next, we codify the notion of train and test error equality from Theorem 1.2.

**Definition 2.2.** Let $\mathcal{A}_n(X, y) \mapsto (h_1, \dots, h_T, \theta_1, \dots, \theta_T, a_1, \dots, a_T)$ be a sequence of algorithms over $n \in \mathbb{N}$ where, for each $t \in [T]$, $(h_t, \theta_t, a_t) \in \mathbb{R}^{n\times L} \times \mathbb{R}^{d\times L} \times \mathbb{R}^{L'}$. Here $\mathcal{A}_n = \mathcal{A}$ represents training a parametric model and $h_t$ is the pre-activations of our $n$ samples input into $\mathcal{M}_{a_t}$, depending on the parameters $a_t$. Moreover, $\theta_t$ is high dimensional parameter applied to new test examples after $t$ training steps. The sequence of algorithm $\mathcal{A}_n$ satisfies the *train-test identity* if the following holds: Let $(X, y)$ have i.i.d. rows $(x_i, y_i) \sim \mathbb{P}_{x,y}$ specified in Section 2.1, $(\check{x}, \check{y}) \sim \mathbb{P}_{x,y}$ and $\mathcal{L}$ be a loss function. For all $t \in [T]$, the following limit holds almost surely with respect to the data $(X, y)$,

$$\lim_{n,d\to\infty}\frac{1}{n}\sum_{i=1}^n\mathcal{L}(\mathcal{M}_{a_t}(h_{t,i}), y_i) - \mathbb{E}_{\check{x},\check{y}}[\mathcal{L}(\mathcal{M}_{a_t}(\check{x}^\top\theta_t), \check{y})] = 0. \quad (2.3)$$

Notably, the train-test identity does not require the loss to decrease. Assuming the train-test identity holds, Appendix A.3 provides a method to monotonically decrease the test error.

*Remark* 2.3. Instead of the almost sure convergence of Definition 2.2, related conditions have postulated a similar identity in expectation, under the term optimism [34], or control on (2.3) for some Renyi or $f$-divergences [10]. Related is the notion of algorithmic stability [17], which can control the train-test gap for stochastic gradient methods independent of an asymptotic limit.

*Remark* 2.4 (Train-test identity at initialization). For $t = 1$ with $\theta_1$ and $a_1$ independent of the data, both DD and GD satisfy (2.3). Indeed, conditioning on $\beta_1 = (\theta_1, a_1)$, let $L_{i,d}(\beta_1) = \mathcal{L}(\mathcal{M}_{a_1}(x_i^\top \theta_1), y_i)$. We see that $L_{1,d}, \dots, L_{n,d}$ are i.i.d. bounded random variables with respect to the randomness of $x_i, y_i$. Thus, by the strong law of large numbers for triangular arrays (say if $\mathcal{L}$ is bounded), we have almost surely over $(X, y)$ that, $\lim_{n,d\to\infty} \left| \frac{1}{n} \sum_{i=1}^n L_{i,d}(\beta_1) - \mathbb{E}\left[L_{1,d}(\beta_1)\right] \right| = 0$. This implies the train-test identity as the law of $(x_1, y_1)$ is equal the law of a new test point.

### 2.3 Natural Algorithmic Consequences

Algorithms satisfying the train-test identity (for example pure DD) can implement a "zero-cost validation" phenomenon. Standard practice saves an $\epsilon$ proportion of the data to estimate the test error with the average of the trained model's loss on the unseen validation set. This introduces a frustrating tradeoff where one wants $\epsilon$ to be small to maximize the data involved in training but not too small where the estimator for the test error becomes unreliable. Thus, a zero-cost validation method, i.e. when all data points are used directly in training, is desirable. This suggests the train-test identity as an algorithmic principle for training parameters; some benefits of this principle are given below.

- **Early Stopping (Online):** Denoting $E_t = \frac{1}{n} \sum_{i=1}^n \mathcal{L}(\mathcal{M}_{a_t}(h_{t,i}), y_i)$, let $D_t = E_{t+1} - E_t$ and $\tilde{D}_t = \log(E_{t+1}) - \log(E_t)$. Stopping when $D_t \geq -\epsilon$ or $\tilde{D}_t \geq \log(1-\epsilon)$ provides a certificate that the subsequent update will not improve the test error by the specified threshold. In particular, $\epsilon = 0$ guarantees the test error is non-increasing during training.
- **Early Stopping (Offline):** Save a subsequence of iterates $\mathcal{T} \subseteq [T]$ and select the specific parameters $(\theta_{t^*}, a_{t^*})$ that minimize the train loss $E_t$ across the saved time steps.
- **Initializations/Hyperparameter Tuning:** Conduct many parallel runs with varying initializations or hyperparameters (e.g. $\eta, \gamma$) and select the configuration that minimizes the train error, and by the train-test identity, minimizes the test error.
- **Architecture Search:** Optimize model complexity (dimension $L$ or architecture $\mathcal{M}_a$) by selecting the run with the lowest train error; this equivalently identifies the optimal test error architecture. We expect an online method for architecture selection is also possible.

Holistically, DD trades monotonic descent on the training error for exact generalization tracking, transforming hyperparameter tuning into an explicit feature of the optimization algorithm.

## 3 Decoupled Descent

**Data Model and Parametric Function** For bounded $J \in \mathbb{N}$, signal vectors $\mu_1, \dots, \mu_J$, response laws $\mathbb{P}_1, \dots, \mathbb{P}_J$, and class probabilities $p_1, \dots, p_J$, consider the Gaussian mixture classification model,

$$(x_i, y_i) \sim \left( \mathcal{N}\left( \frac{\mu_j}{d}, \frac{\mathrm{Id}_d}{d} \right) \otimes \mathbb{P}_j \right) \text{ with probability } p_j. \tag{3.1}$$

We then define the matrix-vector pair $X \in \mathbb{R}^{n\times d}$, $y \in \mathbb{R}^n$ where the $i$-th row of $X$ and $y$ are i.i.d. draws $x_i$ and $y_i$. We consider the limit $n, d \to \infty$ with $n/d \to \alpha$. The data matrix $X$ is commonly seen in mean field analysis of Gaussian mixture models [28]. Specific statistical problems using data (3.1) are provided in Example A.1. This model is motivated by generating structure in latent classification tasks with measurement error, instead of (say) by a teacher function.

*Remark* 3.1. This work requires that $J$ is bounded, we expect this can be relaxed to a $J \to \infty$ limit after the limit $n, d \to \infty$. This naturally follows by designing finer and finer discrete distribution approximations to a low-dimensional data-generating process. Such a low-dimensional process is seen in Example A.1 (3) where we could instead consider signal vectors $\mu \sim cv$ with $v \in \mathbb{R}^d$ and $c$ is endowed some prior distribution supported on a compact interval. Although the finite mixture assumption on the signal is stylized, it establishes a rigorous framework for isolating and correcting data-reuse bias in high-dimensional dynamics.

For parameters $\theta \in \mathbb{R}^{d\times L}$ and $a \in \mathbb{R}^{L'}$, with $L, L'$ bounded, define $\mathcal{M}_{\theta,a}(x) : \mathbb{R}^d \to \mathbb{R}$ with

$\mathcal{M}_{\theta,a}(x) = \mathcal{M}_a(x^\top\theta)$, where $\mathcal{M}_a : \mathbb{R}^L \to \mathbb{R}$ uses parameter $a$. To evaluate the performance of this model, we consider a (possibly non-convex) loss function $\mathcal{L}(\hat{y}, y) : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$. Specific parametric model and loss pairs following this format are given in Example A.2. We restrict to finite sized subsequent layers with $L, L' = O(1)$ so the high-dimensional interaction of the data $X$ and model parameters are isolated in $\theta$.

**Training Protocol** Now, we introduce the general family of DD algorithms.

**Definition 3.2.** Let functions $g : \mathbb{R}^L \times \mathbb{R} \times \mathbb{R}^{L'} \to \mathbb{R}^L$, $f : \mathbb{R}^L \times \mathbb{R} \times \mathbb{R}^{L'} \to \mathbb{R}^{L'}$ and hyperparameters $\eta_0, \eta_1, \gamma_0, \gamma_1$ parameterize *decoupled descent*. When $h \in \mathbb{R}^{n \times L}$ and $y \in \mathbb{R}^n$, we denote $g(h, y, a)_i = g(h_i, y_i, a)$ for each $i \in [n]$ as the row wise application of $g$; consider the iteration,

$$\begin{aligned} h_t &= X\theta_t + \eta_1 \sum_{s=1}^{t-1} \eta_0^{(t-1)-s} \hat{h}_s, \quad \hat{h}_t = g(h_t, y, a_t) \\ \tilde{\theta}_t &= X^\top \hat{h}_t - \alpha\theta_t \left( \frac{1}{n} \sum_{i=1}^{n} \nabla_h g(h_{t,i}, y_i, a_t) \right)^\top \\ \theta_{t+1} &= \eta_0\theta_t - \eta_1\tilde{\theta}_t, \quad a_{t+1} = \gamma_0 a_t - \gamma_1 \frac{1}{n} \sum_{i=1}^{n} f(h_{t,i}, y_i, a_t). \end{aligned} \tag{3.2}$$

To provide a streamlined analysis of DD algorithms, we make strong assumptions on the above setting. We describe them informally below, see Appendix A.5 for the formal statement.

**Assumption 3.3** (Informal Version Of Assumption A.4)**.** Assume that:

1. The rows of $(X, y)$ are i.i.d. from distribution (3.1) where $n, d \to \infty$ with $n/d \to \alpha \in (0, \infty)$. Moreover, we assume that $d^{-1}\mu_j^\top \mu_k \to \chi_{j,k}$ almost surely as $n, d \to \infty$.
2. Initializations $\theta_1, a_1$ are data-independent with bounded limiting norms and signal alignment.
3. Uniformly over $y$, the functions $f, g$ (and $\nabla_h f, \nabla_h g$) are Lipschitz and bounded in $h$ and $a$.
4. $\Psi$ is bounded and $(C^4, C^2) \cap \text{Lip}$ for $(h, a)$ with bounded expected derivatives under distribution (3.1).

*Remark* 3.4 (Universality and more general activation functions). As DD is a designed AMP algorithm, we conjecture our results are robust to relaxations of Assumption 3.3. Indeed, AMP is known to exhibit *universality* and we empirically demonstrate this robustness in Section 5, for example:

1. Previous works allow one to replace the Gaussian noise in (3.1) with mean zero, variance $1/d$, independent (but not identical) sub-Gaussian noise [35, 6, 4]. We expect our results to hold under this change in noise.
2. Previous works relax the bounded and Lipschitz requirement for the activation/loss functions $f, g$ and $\Psi$ [26, 31, 7], although truncating a desired function and considering a bounded Lipschitz extension can give a suitable analysis. See Remark A.5 for an example extension.

### 3.1 Main Result: The State Evolution Of Decoupled Descent

We begin by providing an asymptotic representation of the test error.

**Definition 3.5.** Given $m_{j,\theta} \in \mathbb{R}^L$ for $j \in [J]$, $\Omega_\theta \in \mathbb{R}^{L \times L}$ and $\bar{a} \in \mathbb{R}^{L'}$, define

$$\mathbf{test}(m_{1,\theta}, \ldots, m_{J,\theta}, \Omega_\theta, \bar{a}) = \sum_{j=1}^{J} p_j \mathbb{E}_{\substack{\check{Z}_\theta \sim \mathcal{N}(0, \Omega_\theta) \\ \check{Y}_j \sim \mathbb{P}_j}} [\mathcal{L}(\mathcal{M}_{\bar{a}}(m_{j,\theta} + \check{Z}_\theta), \check{Y}_j)].$$

It is easy to see that for trained parameters $\theta, a$ with almost sure limits $d^{-1}\mu_j^\top\theta \to m_{j,\theta}$, $d^{-1}\theta^\top\theta \to \Omega_\theta$, $a \to \bar{a}$, we have that $\lim_{d\to\infty} \mathbb{E}_{\check{x},\check{y}}[\mathcal{L}(\mathcal{M}_{\theta,a}(\check{x}), \check{y})] = \mathbf{test}(m_{1,\theta}, \ldots, m_{J,\theta}, \Omega_\theta, \bar{a})$ almost surely. This is proven in Appendix A.6.

We now present the main technical lemma of this work, the state evolution of decoupled descent. For simplicity, we defer the full low-dimensional system of recursive equations describing the state evolution parameters to Appendix A.7.

**Lemma 3.6.** *If Assumption 3.3 holds and $\phi : \mathbb{R}^L \times \mathbb{R} \times \mathbb{R}^{L'} \to \mathbb{R}$, $f, g$ are suitably regular (see Assumption A.9), then almost surely for $(h_t, a_t, \theta_t)_{t\in[T]}$ from (3.2),*

$$\lim_{n\to\infty} \frac{1}{n}\sum_{i=1}^{n} \phi(h_{t,i}, y_i, a_t) = \sum_{j=1}^{J} p_j \mathbb{E}[\phi(m_{j,t} + G^t, Y_j, \bar{a}_t)], \quad G^t \sim \mathcal{N}(0, \Omega_t[t,t]), \quad Y_j \sim \mathbb{P}_j,$$

*where $\Omega_t, \Sigma_t, \Xi_t, m_{j,t}, \bar{a}_t$ are given by the following almost sure limits as $n, d \to \infty$ with $n/d \to \alpha$,*

$$\frac{1}{d}\mu_j^\top \theta_t \to m_{j,t}, \quad \frac{1}{d}\theta_t^\top \theta_t \to \Omega_t[t,t], \quad \frac{1}{d}\theta_t^\top \tilde{\theta}_t \to \Xi_t[t,t], \quad \frac{1}{d}\tilde{\theta}_t^\top \tilde{\theta}_t \to \alpha\Sigma_t[t,t], \quad a_t \to \bar{a}_t.$$

*Remark* 3.7 (Finite Sample Guarantees). Again, basing our descent algorithm in approximate message passing has the benefits of appealing to prior AMP literature, specifically one can endow the above lemma with finite sample guarantees using works [32, 24, 31, 3].

The proof of Lemma 3.6 follows by a change of variables to traditional AMP algorithms and is deferred to Appendix A.8. As a consequence, we have the following theorem that confirms DD satisfies the train-test identity.

**Theorem 3.8.** *Consider the state evolution variables in Lemma 3.6 and* **test** *from Definition 3.5. If Assumption 3.3 holds and $g, f$ are suitably regular (Assumption A.9), the following holds almost surely for the iterates $(h_t, a_t, \theta_t)_{t\in[T]}$ from (3.2) and $(\check{x}, \check{y})$ are drawn from distribution (3.1),*

$$\lim_{n\to\infty} \frac{1}{n}\sum_{i=1}^{n} \mathcal{L}(\mathcal{M}_{a_t}(h_{t,i}), y_i) = \mathbf{test}(m_{1,t}, \ldots, m_{J,t}, \Omega_t[t,t], \bar{a}_t),$$

*as a consequence, the train-test identity holds for DD, i.e. almost surely,*

$$\lim_{n,d\to\infty} \frac{1}{n}\sum_{i=1}^{n} \mathcal{L}(\mathcal{M}_{a_t}(h_{t,i}), y_i) - \lim_{d\to\infty} \mathbb{E}_{\check{x},\check{y}}[\mathcal{L}(\mathcal{M}_{a_t}(\check{x}^\top \theta_t), \check{y})] = 0.$$

The proof of this statement is deferred to Appendix A.9. We sketch the proof of Theorem 3.8 below.

**Proof Sketch**: We rewrite DD, via a change of variables, as a non-separable matrix-valued AMP algorithm [5, 26]. This establishes that the coordinate-wise average of $\phi$ applied to $h_t$ is governed by a low-dimensional set of recursive equations (Appendix A.7). Choosing $\phi = \Psi$, this average (with respect to iterate $\theta_t$) only depends on the almost sure limits of $d^{-1}\mu_j^\top \theta_t$ and $d^{-1}\theta_t^\top \theta_t$. These moments evolve identically as if we trained on the data $X^{(t)} = S + Z^{(t)}$ at time $t$ where $Z^{(t)}$ is a fresh Gaussian matrix and $S$ has rows independently assigned $\mu_j/d$ with probability $p_j$. Due to this behavior, the training error must be the test error since DD essentially evaluates on a fresh independent data set at each update.

*Remark* 3.9. A dynamical mean field theory (DMFT) analysis of overfitting in full-batch GD is given in [29, Equations (C.7)-(C.10)] and [16, Algorithm 1]. This description (including asymptotic tracking or estimation of the test error) contains complex "response function" terms represented by the integration (or summation) of a two-time correlation matrix which accounts for the reuse of the data matrix. By implementing DD with AMP, however, our algorithm is self correcting in the sense that the response function is zero, admitting a simpler analysis than previous DMFT methods. This simplicity also allows for better physical interpretation and insight into algorithmic design.

## 4 Algorithm Design

The choice of $g, f$ in algorithm (3.2) controls the state evolution variables, and in turn, the asymptotic test error $\mathbf{test}_t = \mathbf{test}(m_{1,t}, \ldots, m_{J,t}, \Omega_T[t,t], \bar{a}_t)$ by Theorem 3.8. This means that given $g, f$ we can calculate $\mathbf{test}_t$ exactly by simulations. Thus, given candidate function classes $\mathcal{G}$ and $\mathcal{F}$, with a budgeted run-time $T$, the optimal $g^*, f^*$ is found by running $|\mathcal{F}||\mathcal{G}|$ total simulations and selecting the run that minimized $\min_{t\in[T]} \mathbf{test}_t$. Unfortunately, to run these simulations, we require exact knowledge of the data generating processes (3.1). Therefore, it is desirable to select $g, f$ which produce low test error independent of the data generating process.

### 4.1 Pure Decoupled Descent

Using (3.2), we recover pure DD from Section 2 by selecting $g = \nabla_h \Psi$, $f = \nabla_a \Psi$, $\eta_0 = \gamma_0 = 1$ and $\eta_1 = \gamma_1 = \eta$. See (2.2) for the exact iteration and Appendix A.10 for the corresponding state evolution.

Traditionally in the analysis of GD, we consider a Taylor expansion on the train error when the learning rate $\eta \to 0$. Because pure DD satisfies the train-test identity, in contrast to GD, we can directly expand the asymptotic test error. With $G^t$ defined by the state evolution in Appendix A.10 and $Y_j \sim \mathbb{P}_j$ from (3.1), let

$$\mathbf{G}_t = \Big(\mathbb{E}[\nabla_a \Psi(m_{j,t} + G^t, Y_j, \bar{a}_t)]\Big)_{j\in[J]}, \qquad \mathbf{U}_t = \Big(\mathbb{E}[\nabla_h \Psi(m_{j,t} + G^t, Y_j, \bar{a}_t)]\Big)_{j\in[J]},$$

and denote $\boldsymbol{\chi} = (\chi_{j,k})_{j,k\in[J]}$ and $\mathbf{p} = (p_j)_{j\in[J]}$ such that $(\mathrm{diag}(\mathbf{p})\boldsymbol{\chi}\mathrm{diag}(\mathbf{p}))_{j,k} = p_j \chi_{j,k} p_k$.

**Theorem 4.1.** *Let pure DD satisfy the conditions of Assumption 3.3 and Assumption A.9 for sufficiently small $\eta$ up to bounded time $T \in \mathbb{N}$ independent of $n$. As $\eta \to 0$, $\mathbf{test}_{t+1}$ satisfies the following Taylor expansion with state evolution parameters corresponding to the state evolution recursion (A.18),*

$$\mathbf{test}_{t+1} = \mathbf{test}_t - \eta \|\mathbf{G}_t \mathbf{p}\|_2^2 - \eta\alpha \|\mathbf{U}_t (\mathrm{diag}(\mathbf{p})\boldsymbol{\chi}\mathrm{diag}(\mathbf{p}))^{1/2}\|_{\mathrm{F}}^2 \tag{4.1}$$

$$- \frac{\eta}{2} \sum_{j=1}^{J} p_j \Big\langle \mathbb{E}[\nabla_h^2 \Psi(m_{j,t} + G^t, Y_j, \bar{a}_t)], \Xi_t[t,t] + \Xi_t[t,t]^\top \Big\rangle + \epsilon_t, \tag{4.2}$$

*where $\sup_{t\in[T]} |\epsilon_t| \le C\eta^2$ as $\eta \to 0$ (where $C$ depends on constants in Assumption A.9).*

Notice the terms in (4.1) are non-positive, meaning that they directly lead to a decrease in the test error. The remaining term in (4.2), however, may be positive. In general, the presence of this term can lead pure DD to under-perform compared to GD, the train test identity makes this effect transparent and possible to correct. Moreover, we describe methods to bound the impact of this term in Remark A.20. The proof of Theorem 4.1 is deferred to Appendix A.11 alongside a general Taylor expansion for generic choices of $g$, $f$ and learning rate parameters. To give insight on the global convergence properties of DD, we describe fixed points of DD algorithms and compare them to critical points of the test error in Appendix A.12.

## 5 Applications

Below we present three applications of decoupled descent, a fourth application on the signal-less regression problem from the introduction is deferred to Appendix B.1. Experiments were conducted on a M4 Mac mini (2024, 10-core CPU, 10-core GPU) with 16GB of unified memory.

### 5.1 Improved Training On The XOR Model

We consider a high-dimensional variant of the XOR problem ($J = 4$, $p_j = 1/4$) with $n$ i.i.d. data points with the following signal vectors dependent on fixed $v \in \mathbb{R}^{d/2}$, $\mu_1 = [v, v], \mu_2 = [-v, -v], \mu_3 = [-v, v], \mu_4 = [v, -v]$, where $y_j = 0$ for $j \in \{1, 2\}$ and $y_j = 1$ for $j \in \{3, 4\}$.

We fit model $\mathcal{M}_{a,\theta^1,\theta^2}(x) = \sigma(a(x^\top\theta^1)(x^\top\theta^2))$, where $\theta^1 \in \mathbb{R}^d$, $\theta^2 \in \mathbb{R}^d$, $a \in \mathbb{R}$ and $\sigma(x) = \frac{1}{1+e^{-x}}$, under cross entropy loss (i.e. $\mathcal{L}(\hat{y}, y) = -y\log(\hat{y}) - (1-y)\log(1-\hat{y})$). These parameters are initialized at $\theta_1^1 \sim \mathcal{N}(0, \mathrm{Id}_d)$, $\theta_1^2 \sim \mathcal{N}(0, \mathrm{Id}_d)$ and $a_1 \sim \mathcal{N}(0, 1)$ independently. Both GD and DD iterations for this problem are given in Appendix B.2. Under a suitable truncation of cross-entropy loss and the model $\mathcal{M}$ from Remark 3.4, this setup exactly fits Assumption 3.3.

Figure 1 plots the train and test errors for GD and DD when each $v_i = \lambda$ with $\lambda \in \{1, 4, 8\}$ as a signal to noise ratio (SNR) parameter. In all cases, DD maintains the train-test identity. In low SNR regimes ($\lambda \in \{1, 4\}$), DD outperforms GD and maintains the train-test identity while GD rapidly overfits; for high SNR regimes, DD matches GD's performance because, under the $\lambda \to \infty$ limit, the signal dominates the noise and the effect of memorization is negligible.

We also provide an example of zero-cost validation hyperparameter tuning from Section 2. We fix the

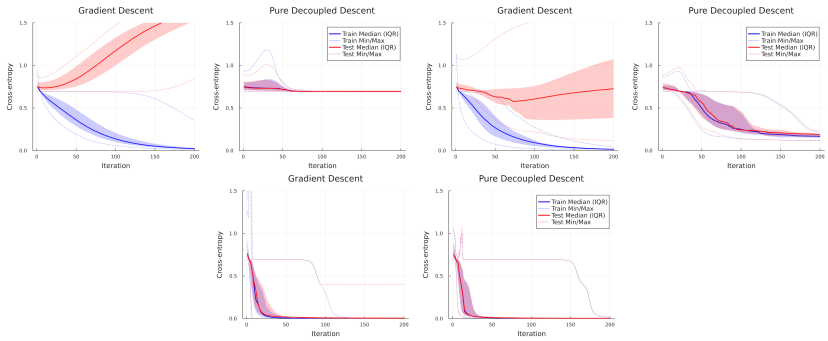


Figure 1: Summary statistics for 20 XOR runs ($n = d = 1000$): GD (left) vs. DD (right) with $\eta = 0.05$ and SNR $\lambda = 1$ (upper left), $\lambda = 4$ (upper right) and $\lambda = 8$ (bottom). Blue/red denote train/test error; solid lines are medians, shaded areas are interquartile ranges, and dotted lines show min/max. **Low SNR ($\lambda = 1$):** GD overfits (low train/high test error); DD stabilizes both near $\log(2)$, reflecting the non-informative regime. **Medium SNR ($\lambda = 4$):** DD maintains train-test parity, outperforming GD's overfit solution. **High SNR ($\lambda = 8$):** Signal dominates noise; both algorithms achieve similar performance.

value of $a_t = 1$ in $\mathcal{M}_a$ and train $\theta^1$ and $\theta^2$ with a damped variant of pure DD given in Appendix B.2; this training model incorporates varying $\eta_0$ and represents weight regularization for the parameters. In Figure 2, we run 50 shared replications of the XOR data and plot the train-test error for each choice of $\eta_0 \in \{1, 0.9, 0.8\}$ (recalling $\eta_0 = 1$ is pure DD). We can see that our zero-cost validation method is successful, each run demonstrates the train-test identity, and we observe that some weight regularization is beneficial in this case.

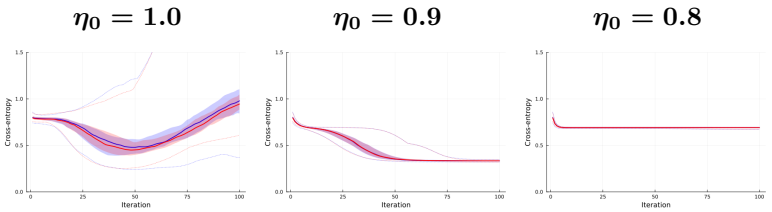


Figure 2: Summary statistics for 50 XOR runs ($n = d = 1000$) of damped DD (defined in Appendix B.2): $\eta = 0.05$, fixed $a_t = 1$, $\lambda = 4$ and $\eta_0 = 1$ (left), $\eta_0 = 0.9$ (middle) and $\eta_0 = 0.8$ (right). Blue/red colors and line type are equivalent to Figure 1. **No Regularization**: $\eta_0 = 1$ (pure DD) gives an initial decrease in the loss but suffers from inflated values of $\Omega_t[t, t]$ for later iterations (note, the early stopping technique of Section 2 would be helpful in this case). **Mild Regularization**: $\eta_0 = 0.9$ finds a good balance in weight regularization which controls variance inflation while achieving good test error and training stability. **High Regularization**: $\eta_0 = 0.8$ regularizes too strongly and leads to an estimator that is too conservative, leading to a larger train-test error.

### 5.2 Varying Layer Widths For Classifying Zeros And Eights For MNIST Data

We apply DD to a simple MNIST binary classification test (i.e. zeros vs. eights). For the train data, we sample $n = 800$ digits (grayscale images of dimension $d = 784$) from all zeros and eights in the MNIST data set, rescale by SNR $\lambda/d$, and inject mean zero, variance $1/d$, discrete noise,

$$\delta_{-\sqrt{2}/\sqrt{d}} \text{ wp } 1/4, \quad \delta_{\sqrt{2}/\sqrt{d}} \text{ wp } 1/4, \quad \delta_0 \text{ wp } 1/2, \tag{5.1}$$

(instead of the standard Gaussian noise from distribution (3.1)). The test set is all zero and eight digits in MNIST with identical processing; this experiment verifies if the train-test identity of decoupled descent demonstrates universality phenomena similar to AMP algorithms under discrete non-Gaussian noise and is robust when the data is not immediately represented by $J$ mixture modes.

Additionally, to see if the train-test identity for decoupled descent degrades as $L$ increases in size, we train a two-layer network (hidden widths $L \in \{3, 9, 27\}$, $\tanh$ activation, sigmoid output) using

cross entropy loss and learning rate $\eta = 1$. We directly implemented this training procedure in `Pytorch` using a special MLP class (see Appendix B.4 for details) and each parameter is initialized as a standard normal. Figure 3 plots the train-test error for these three models, we see the train-test identity approximately holds with a slight separation as the size of the hidden layer increases, the discrete (instead of Gaussian) noise and a possibly more complex signal distribution.

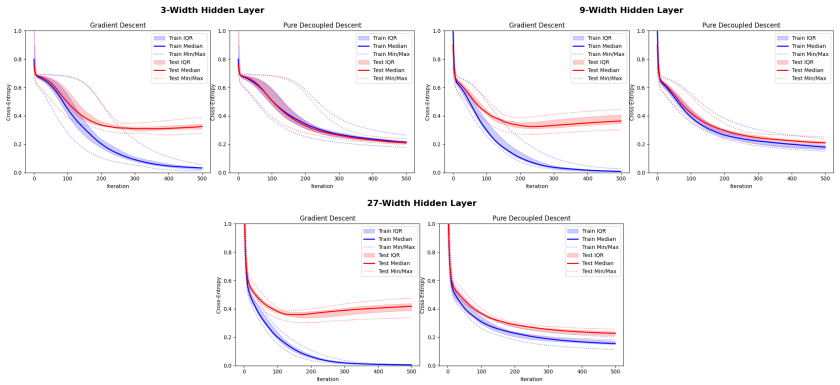


Figure 3: MNIST zeros vs. eights train/test errors ($d = 784, n = 800, \lambda = 30$) over 20 replications: GD (left) vs. DD (right) with discrete noise from distribution (5.1). Blue/red denote train/test error; solid lines are medians, shaded areas are IQRs, dotted lines are min/max. We run a two layer network with hidden layers ($L \in \{3, 9, 27\}$). The train-test error identity continues to hold as $L$ grows and when the Gaussian noise is replaced with this discrete counterpart.

*Remark* 5.1 (On computational complexity). Recall implementing pure DD required two correction terms. Term $\eta \sum_{s=1}^{t-1} \hat{h}_s$ adds $O(n)$ storage but no computational overhead, trivial compared to the size of the data. The Hessian term $\eta \cdot \alpha \left( \frac{1}{n} \sum_{i=1}^{n} \nabla_h^2 \Psi(h_{t,i}, y_i, a_t) \right)$ requires a per-iteration Jacobian. This Hessian overhead is a $d$ independent constant factor—the necessary mechanism for DD to maintain the train-test identity. This constant overhead does lengthen training time, but in applications where collecting 15-20% of additional validation is considerably expensive the additional computation time is a reasonable tradeoff. It is also worth noting that using standard auto-differentiation allows the process of deriving the Hessian automatically, avoiding the process of deriving the correction term on a case by case basis. This sacrifices efficiency for generality, methods using the Hutchinson trace trick to estimate the Onsager correction [5] may give speedups. For completeness, we provide some comparative run-times between GD and pure DD for the $L = 9$ case in Appendix B.3.

### 5.3 Training A MLP Head On CIFAR-10 ResNet Embeddings For Classifying Cats And Dogs

We finally consider a CIFAR-10 classification task. The goal of this task is to classify images as either pictures of cats (CIFAR-10 class 3) or dogs (CIFAR-10 class 5). We note that the DD algorithms in this section were also natively implemented using `Pytorch`'s `torch.autograd` function with standard normal initialization (the CIFAR-10 code is similar to Appendix B.4).

The data matrix for this application is wildly different from Assumption A.4. Instead of noisy images, we use pre-trained ResNet-18 embeddings of dimension $d = 512$. We sample $n = 800$ random embedded vectors from the full pool of embedded cat and dog vectors from CIFAR-10 for both the training and testing set independently. Then, we train a two layer MLP classification head with hidden layer width five and $\tanh$ activation using both GD and pure DD.

The motivation for using DD on these vectors is from Gaussian equivalence theory [15] which postulates one can analyze the intermediate layers of a trained network with Gaussian surrogates. We consider increasingly stronger "whitening" over our training and testing data. In all cases, we standardize the embedding vectors by subtracting their coordinate-wise mean and dividing by the coordinate-wise standard deviation (with respect to only the training distribution). A detailed explanation of the whitening procedures are deferred to Appendix B.5, informally we have (1) *Vanilla*, rescaling features by $1/\sqrt{d}$; (2) *ZCA (Train)*, applying whitening computed solely on training data [20]; and (3) *Joint ZCA*, whitening across the combined train-test pool to prevent covariance drift. These procedures should be interpreted as increasingly stronger processing steps to make the

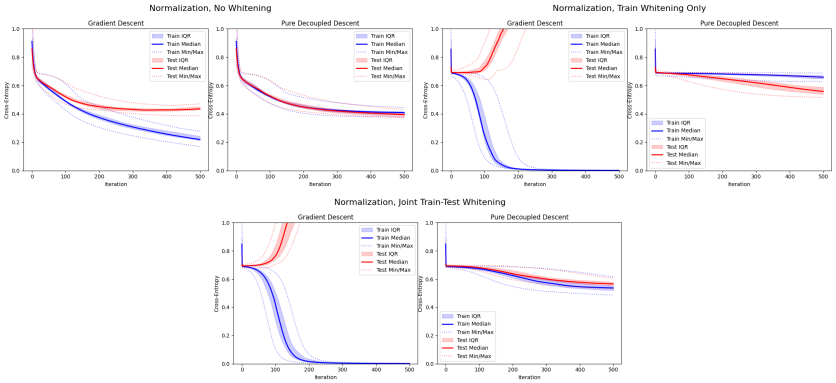


Figure 4: Train/test errors for CIFAR-10 (cats vs. dogs). GD (left) vs. DD (right). Blue/red denote train/test error; solid lines are medians, shaded areas cover the inter-quartile range, dotted lines show min/max, and title is the method of whitening (see Section 5.3). DD reduces overfitting effects compared to GD for training a classification head to ResNet-18 embeddings.

embedded vectors of the train and test set seem more Gaussian. This experiment stress tests how deterministic the data matrix can be for the train-test identity of DD to approximately apply.

We can see the training and test curves for both GD and pure DD in Figure 4. DD substantially reduced the generalization gap for varying degrees depending on the whitening method even when **we have not injected random noise in the data set** and therefore are unlikely to satisfy Assumption 3.3 completely, representing a boundary of our theory where the current Onsager is not guaranteed to completely remove the effect of overfitting. This suggests a more general method to apply DD to models with non-independent noise structures.

## 6 Discussion And Conclusion

To conclude, we presented a novel machine learning training algorithm and demonstrated it solves the train-test error disconnect present in full-batch GD. We additionally provided some initial guidance on how to design such algorithms and gave empirical validation of their success.

**Limitations** While DD guarantees an exact asymptotic train-test identity, the practical deployment faces several constraints. The proofs assume Gaussianity on the data $X$, although some relaxation is expected (Remark 3.4 (1)). Real-world data, however, contains structured correlations that do not satisfy our assumptions. Our results are asymptotic and would benefit from specified finite sample rates (Remark 3.7). We require full-batch training which rules out stochastic methods like SGD; this additionally introduces DD memory and computational overhead due to the correction terms which can result in $\approx L\times$ longer step times (Appendix B.3). We have only considered models with a wide first layer and finite subsequent layers; over-parameterization of later layers may break the train-test identity, although we do not see this in simulations (Section 5.2). Finally, we do not implement a mechanism in DD to represent signal drift, a common second source of overfitting.

**Broader Impacts** When Assumption 3.3 approximately holds, DD offers three benefits to practitioners: (1) Our zero-validation principle maximizes data utilization, vital in domains where data is scarce or expensive to collect. (2) The number of training runs can be limited by embedding hyper-parameter tuning into training dynamics. (3) Exact test error tracking gives a honest assessment of generalization, preventing the deployment of overfitted models.

**Future Directions** We present three future directions. **Orthogonally Invariant Noise**: Can DD extend to data with dependent elements [38, 11, 25] with symmetry properties? It remains open whether the standard Onsager term's partial success on CIFAR-10 is a coincidence or a general property of embedding vectors. **Stochastic DD**: To reduce the full-batch computational overhead, can recent DMFT derivations for SGD [12, 30] inform a mini-batch DD iteration, greatly expanding its practical applicability. **Large-Width Networks**: DD currently requires finite-width subsequent layers, adapting deep AMP/DMFT frameworks [37, 14, 36] to DD with multiple $d$-scaling layers would increase the applicability of these results.

## References


[1] Madhu S. Advani, Andrew M. Saxe, and Haim Sompolinsky. High-dimensional dynamics of generalization error in neural networks. *Neural Networks*, 132:428–446, 2020. DOI: https://doi.org/10.1016/j.neunet.2020.08.022.

[2] Z. D. Bai and Y. Q. Yin. Limit of the smallest eigenvalue of a large dimensional sample covariance matrix. *The Annals of Probability*, 21(3), July 1993. ISSN 0091-1798. DOI: https://doi.org/10.1214/aop/1176989118. URL http://dx.doi.org/10.1214/aop/1176989118.

[3] Zhigang Bao, Qiyang Han, and Xiaocong Xu. A leave-one-out approach to approximate message passing. *The Annals of Applied Probability*, 35(4), August 2025. ISSN 1050-5164. DOI: https://doi.org/10.1214/25-aap2186. URL http://dx.doi.org/10.1214/25-aap2186.

[4] Mohsen Bayati, Marc Lelarge, and Andrea Montanari. Universality in polytope phase transitions and message passing algorithms. *The Annals of Applied Probability*, 25(2), apr 2015. DOI: https://doi.org/10.1214/14-aap1010.

[5] Raphaël Berthier, Andrea Montanari, and Phan-Minh Nguyen. State evolution for approximate message passing with non-separable functions. *Information and Inference: A Journal of the IMA*, 9 (1):33–79, January 2019. ISSN 2049-8772. DOI: https://doi.org/10.1093/imaiai/iay021. URL http://dx.doi.org/10.1093/imaiai/iay021.

[6] Wei-Kuo Chen and Wai-Kit Lam. Universality of approximate message passing algorithms. *Electronic Journal of Probability*, 26, January 2021. ISSN 1083-6489. DOI: https://doi.org/10.1214/21-ejp604. URL http://dx.doi.org/10.1214/21-ejp604.

[7] Yatin Dandi, David Gamarnik, Francisco Pernice, and Lenka Zdeborová. Sequential dynamics in ising spin glasses, 2025.

[8] David Donoho and Andrea Montanari. High dimensional robust m-estimation: Asymptotic variance via approximate message passing. *Probability Theory and Related Fields*, 166(3–4):935–969, 2016. DOI: https://doi.org/10.1007/s00440-015-0675-z.

[9] David L. Donoho, Arian Maleki, and Andrea Montanari. Message passing algorithms for compressed sensing: I. motivation and construction. In *IEEE Information Theory Workshop 2010 (ITW 2010)*, page 1–5. IEEE, January 2010. DOI: https://doi.org/10.1109/itwksps.2010.5503193. URL http://dx.doi.org/10.1109/itwksps.2010.5503193.

[10] Amedeo Roberto Esposito, Michael Gastpar, and Ibrahim Issa. Generalization error bounds via rényi-, $f$-divergences and maximal leakage. *IEEE Transactions on Information Theory*, 67(8):4986–5004, 2021. DOI: https://doi.org/10.1109/TIT.2021.3087265.

[11] Zhou Fan. Approximate message passing algorithms for rotationally invariant matrices. *The Annals of Statistics*, 50(1), February 2022. ISSN 0090-5364. DOI: https://doi.org/10.1214/21-aos2101. URL http://dx.doi.org/10.1214/21-aos2101.

[12] Zhou Fan and Leda Wang. High-dimensional learning dynamics of multi-pass stochastic gradient descent in multi-index models, 2026.

[13] Oliver Y. Feng, Ramji Venkataramanan, Cynthia Rush, and Richard J. Samworth. *A Unifying Tutorial on Approximate Message Passing*. Now Publishers, 2022. ISBN 9781638280057. DOI: https://doi.org/10.1561/9781638280057. URL http://dx.doi.org/10.1561/9781638280057.

[14] Cédric Gerbelot and Raphaël Berthier. Graph-based approximate message passing iterations. *Information and Inference: A Journal of the IMA*, 12(4):2562–2628, September 2023. ISSN 2049-8772. DOI: https://doi.org/10.1093/imaiai/iaad020. URL http://dx.doi.org/10.1093/imaiai/iaad020.

[15] Sebastian Goldt, Bruno Loureiro, Galen Reeves, Florent Krzakala, Marc Mezard, and Lenka Zdeborova. The gaussian equivalence of generative models for learning with shallow neural networks. In *Mathematical and Scientific Machine Learning*, 2020. URL https://api.semanticscholar.org/CorpusID:235165686.

[16] Qiyang Han and Masaaki Imaizumi. Precise gradient descent training dynamics for finite-width multi-layer neural networks, 2025. URL https://arxiv.org/abs/2505.04898.

[17] Moritz Hardt, Benjamin Recht, and Yoram Singer. Train faster, generalize better: Stability of stochastic gradient descent. In *Proceedings of the 33rd International Conference on Machine Learning (ICML)*, volume 48 of *Proceedings of Machine Learning Research*, pages 1225–1234, 2016.

[18] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer Series in Statistics. Springer, 2 edition, 2009. DOI: https://doi.org/10.1007/978-0-387-84858-7.

[19] A. Javanmard and A. Montanari. State evolution for general approximate message passing algorithms, with applications to spatial coupling. *Information and Inference*, 2(2):115–144, October 2013. ISSN 2049-8772. DOI: https://doi.org/10.1093/imaiai/iat004. URL http://dx.doi.org/10.1093/imaiai/iat004.

[20] Agnan Kessy, Alex Lewin, and Korbinian Strimmer. Optimal whitening and decorrelation. *The American Statistician*, 72(4):309–314, January 2018. ISSN 1537-2731. DOI: https://doi.org/10.1080/00031305.2016.1277159. URL http://dx.doi.org/10.1080/00031305.2016.1277159.

[21] Tomer Koren, Roi Livni, Yishay Mansour, and Uri Sherman. Benign underfitting of stochastic gradient descent. In *Advances in Neural Information Processing Systems*, volume 35, pages 19728–19740, 2022.

[22] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.

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

[24] Gen Li and Yuting Wei. A non-asymptotic framework for approximate message passing in spiked models, 2023.

[25] Songbin Liu and Junjie Ma. Unifying amp algorithms for rotationally-invariant models, 2024.

[26] Max Lovig, Tianhao Wang, and Zhou Fan. On universality of non-separable approximate message passing algorithms, 2025.

[27] Vladimir A. Marchenko and Leonid A. Pastur. Distribution of eigenvalues for some sets of random matrices. *Matematicheskii Sbornik*, 114(4):507–536, 1967. DOI: https://doi.org/10.1070/SM1967v001n04ABEH001994. English translation in Math. USSR-Sbornik 1(4): 457–483, 1967.

[28] Francesca Mignacco, Florent Krzakala, Pierfrancesco Urbani, and Lenka Zdeborová. Dynamical mean-field theory for stochastic gradient descent in gaussian mixture classification. *Journal of Statistical Mechanics: Theory and Experiment*, 2021(12):124008, December 2021. ISSN 1742-5468. DOI: https://doi.org/10.1088/1742-5468/ac3a80. URL http://dx.doi.org/10.1088/1742-5468/ac3a80.

[29] Andrea Montanari and Pierfrancesco Urbani. Dynamical decoupling of generalization and overfitting in large two-layer networks, 2025.

[30] Sota Nishiyama and Masaaki Imaizumi. High-dimensional limit of stochastic gradient flow via dynamical mean-field theory, 2026.

[31] Galen Reeves. Dimension-free bounds for generalized first-order methods via gaussian coupling, 2025.

[32] Cynthia Rush and Ramji Venkataramanan. Finite sample analysis of approximate message passing algorithms. *IEEE Transactions on Information Theory*, 64(11):7264–7286, nov 2018. DOI: https://doi.org/10.1109/tit.2018.2816681.

[33] Sophia Sklaviadis, Thomas Moellenhoff, Andre F. T. Martins, Mario A. T. Figueiredo, and Mohammad Emtiyaz Khan. A stein identity for q-gaussians with bounded support, 2026.

[34] Ryan J. Tibshirani and Saharon Rosset. Excess optimism: How biased is the apparent error of an estimator tuned by SURE? *Journal of the American Statistical Association*, 114(526):697–712, 2019. DOI: https://doi.org/10.1080/01621459.2018.1448825.

[35] Tianhao Wang, Xinyi Zhong, and Zhou Fan. Universality of approximate message passing algorithms and tensor networks. *The Annals of Applied Probability*, 34(4), August 2024. ISSN 1050-5164. DOI: https://doi.org/10.1214/24-aap2056. URL http://dx.doi.org/10.1214/24-aap2056.

[36] Yizhou Xu, TianQi Hou, ShanSuo Liang, and Marco Mondelli. Approximate message passing for multi-layer estimation in rotationally invariant models. In *2023 IEEE Information Theory Workshop (ITW)*, page 294–298. IEEE, April 2023. DOI: https://doi.org/10.1109/itw55543.2023.10160238. URL http://dx.doi.org/10.1109/itw55543.2023.10160238.

[37] Greg Yang and Edward J. Hu. Tensor programs iv: Feature learning in infinite-width neural networks. In *International Conference on Machine Learning (ICML)*, pages 11727–11737. PMLR, 2021.

[38] Xinyi Zhong, Tianhao Wang, and Zhou Fan. Approximate message passing for orthogonally invariant ensembles: multivariate non-linearities and spectral initialization. *Information and Inference: A Journal of the IMA*, 13(3), July 2024. ISSN 2049-8772. DOI: https://doi.org/10.1093/imaiai/iaae024. URL http://dx.doi.org/10.1093/imaiai/iaae024.

# A Deferred Theory

## A.1 Notation

Let $[n] = \{1, \dots, n\}$; $\mathbb{R}$ and $\mathbb{N}$ denote the real and natural numbers, with $(\cdot)^k$ as the $k$-fold Cartesian product. For $M \in \mathbb{R}^{n \times L}$, $M_i \in \mathbb{R}^L$ denotes the $i$-th row. For a block matrix $M$, $M[r, s]$ is its $(r, s)$-th block. $\mathrm{vec}(\cdot)$ and $\mathrm{mat}(\cdot)$ represent standard vectorization and matricization. $\nabla^k_{(\cdot)}$ is the $k$-fold partial derivative. Norms $\|\cdot\|_2, \|\cdot\|_\infty, \|\cdot\|_F, \|\cdot\|_{\mathrm{op}}$ are the Euclidean, infinity, Frobenius, and operator norms. $\mathbb{1}\{\cdot\}$ is the indicator function and $\delta_x$ is the Dirac distribution at $x \in \mathbb{R}$. $\otimes, \odot, \langle \cdot, \cdot \rangle$ denote Kronecker, Hadamard, and inner products (where $\otimes$ additionally means the independent product of two probability measures). $e_B \in \mathbb{R}^n$ is $(e_B)_i = \mathbb{1}\{i \in B\}$. $O, o, \Theta$ represent standard asymptotic notation with $n$-dependent growth.

## A.2 The First Two Iterations of DD and GD for Example 1.1

Recall Example 1.1, where we consider $n$ i.i.d. feature vectors $x_i \sim \mathcal{N}\left(0, \frac{1}{d} I_d\right)$ with $y_i = 0$ for all $i \in [n]$. We then train a linear model $\mathcal{M}(x) = x^\top \beta$ under MSE loss $\Psi(h, y) = h^2$. Then,

$$\nabla_h \Psi(h, y) = 2h, \quad \nabla^2_h \Psi(h, y) = 1.$$

Assume that the initialization $\beta_1$ is independent of the data with $\frac{1}{d}\|\beta_1\|^2 \to 1$ almost surely for concreteness; the general case follows analogously. Denoting $M = X^\top X \in \mathbb{R}^{d \times d}$, under the proportional asymptotic limit $n, d \to \infty$ with $n/d \to \alpha \in (0, \infty)$, the spectral distribution of $M$ converges to the Marchenko-Pastur law $\rho_{\mathrm{MP}}$ [27], whose first three moments are:

$$m_1 = \int \lambda \, d\rho_{\mathrm{MP}}(\lambda) = \alpha, \quad m_2 = \int \lambda^2 \, d\rho_{\mathrm{MP}}(\lambda) = \alpha^2 + \alpha, \quad m_3 = \int \lambda^3 \, d\rho_{\mathrm{MP}}(\lambda) = \alpha^3 + 3\alpha^2 + \alpha$$

We then have that the $t = 2$ update of (2.2) with $\theta_i = \beta_i$ (for $i \in \{1, 2\}$) and rescaling $\eta = \eta'/2$ is

$$\beta_2 = \beta_1 - \eta' \left(X^\top X \beta_1 - \alpha \beta_1\right) = \left((1 + \eta'\alpha) I_d - \eta' X^\top X\right) \beta_1 = \left((1 + \eta'\alpha) I_d - \eta' M\right) \beta_1,$$

and the pre-activation is

$$h_2 = X\beta_2 + \eta' X \beta_1 = X \left((1 + \eta' + \eta'\alpha) I_d - \eta' X^\top X\right) \beta_1 = X \left((1 + \eta' + \eta'\alpha) I_d - \eta' M\right) \beta_1.$$

Under this update, the test error of $\beta_2$ can be written as:

$$\begin{aligned}
\check{L}(\beta_2) &= \frac{1}{d}\|\beta_2\|_2^2 = \frac{1}{d} \beta_1^\top \left((1 + \eta'\alpha) I_d - \eta' M\right)^\top \left((1 + \eta'\alpha) I_d - \eta' M\right) \beta_1 \\
&= (1 + 2\eta'\alpha + \eta'^2 \alpha^2) \left(\frac{\beta_1^\top \beta_1}{d}\right) - (2\eta' + 2\eta'^2 \alpha) \left(\frac{\beta_1^\top M \beta_1}{d}\right) + \eta'^2 \left(\frac{\beta_1^\top M^2 \beta_1}{d}\right).
\end{aligned}$$

As $d \to \infty$, we have that $\frac{1}{d} \beta_1^\top M^k \beta_1 = m_k$, and therefore, almost surely:

$$\frac{1}{d}\|\beta_2\|_2^2 \to (1 + 2\eta'\alpha + \eta'^2 \alpha^2) - (2\eta' + 2\eta'^2 \alpha)\alpha + \eta'^2(\alpha^2 + \alpha) = 1 + \eta'^2 \alpha. \tag{A.1}$$

Evaluating the training error with $\frac{1}{n}\|h_2\|_2^2$, denoting $\xi = 1 + \eta' + \eta'\alpha$, we have that,

$$\begin{aligned}
\frac{1}{n}\|h_2\|_2^2 &= \frac{1}{n} \beta_1^\top \left(\xi I_d - \eta' M\right)^\top M \left(\xi I_d - \eta' M\right) \beta_1 = \frac{1}{n} \beta_1^\top \left(\xi^2 M - 2\eta'\xi M^2 + \eta'^2 M^3\right) \beta_1 \\
&= \frac{1}{\alpha} \xi^2 \frac{1}{d} \beta_1^\top M \beta_1 - \frac{2}{\alpha} \eta' \xi \frac{1}{d} \beta_1^\top M^2 \beta_1 + \frac{1}{\alpha} \eta'^2 \frac{1}{d} \beta_1^\top M^3 \beta_1.
\end{aligned}$$

And therefore, using the moments of the Marchenko-Pastur law, we have almost surely,

$$\frac{1}{n}\|h_2\|_2^2 \to \xi^2 - 2\eta'\xi(\alpha + 1) + \eta'^2(\alpha^2 + 3\alpha + 1) = 1 + \eta'^2 \alpha,$$

which is identical to the test error in (A.1). This is in contrast to gradient descent, where the $t = 2$ update of (2.1) (with $\eta = \eta'/2$) is $\beta_2 = (I_d - \eta' M)\beta_1$ Evaluating the analogous asymptotic limits then gives the train and test error through the Marchenko-Pastur law of gradient descent where,

$$\check{L}(\beta_2) \to 1 - 2\eta'\alpha + \eta'^2(\alpha^2 + \alpha)$$

$$L(\beta_2) \to 1 - 2\eta'(\alpha + 1) + \eta'^2(\alpha^2 + 3\alpha + 1)$$

which are not equal—resulting in the generalization gap.

### A.3 A Method Of Descent Under The Train-Test Identity

Suppose an algorithm $\mathcal{A}$ satisfies the train-test identity and generates the iterate $(\theta_t, h_t, a_t)$. In addition, there exists a number of updates $(\theta_{t+1}^{(k)}, h_{t+1}^{(k)}, a_{t+1}^{(k)})_{k\in[K]}$ generated by $K$ different algorithms $(\mathcal{A}^{(k)})_{k\in[K]}$, each maintaining the train-test identity. Each algorithm $\mathcal{A}^{(k)}$ produces an update with training error

$$\frac{1}{n}\sum_{i=1}^{n}\mathcal{L}(\mathcal{M}_{a_{t+1}^{(k)}}(h_{t+1,i}^{(k)}), y_i). \tag{A.2}$$

And, by the train-test identity, (A.2) is an asymptotically consistent estimator for the test error of update $(\theta_{t+1}^{(k)}, h_{t+1}^{(k)}, a_{t+1}^{(k)})$, i.e. $\mathbb{E}_{\check{x},\check{y}}[\mathcal{L}(\mathcal{M}_{a_{t+1}^{(k)}}(\check{x}^\top\theta_{t+1}^{(k)}), \check{y})]$. Therefore, conceptually reducing the test error at step $t+1$ across multiple possible algorithmic steps follows by choosing the update that minimizes (A.2). If no algorithm exists then we terminate the optimization protocol. Other algorithms which do not satisfy the train-test identity can fundamentally never offer such a guarantee due to the train-test disconnect from the introduction.

### A.4 Deferred Examples

**Example A.1.** *Many common learning problems can be realized by our data model.*

1. *The signal-less regression model from the introduction is given with $J = 1$, $\mu_j = 0$ and $\mathbb{P}_j = \delta_0$ (i.e. a Dirac measure at zero).*
2. *A simple classification problem with flipped class noise has $J = 2$, $\mu_1 = v$ and $\mu_2 = -v$, and for some $\epsilon \in (0, 1/2)$) we set $\mathbb{P}_1 = (1-\epsilon)\delta_{-1} + \epsilon\delta_1$ and $\mathbb{P}_2 = (1-\epsilon)\delta_1 + \epsilon\delta_{-1}$ with class probabilities $p_1 = p_2 = 1/2$.*
3. *A discrete-valued regression problem has $J$ being some large constant, say 100. We let $\mu_j = jv$ with $v \in \mathbb{R}^d$. Then let $\mathbb{P}_j$ be the convolution of $\delta_{c_1 j + c_2}$ and $\mathbb{P}_{\text{noise}}$ where $c_1, c_2 \in \mathbb{R}$ and $\mathbb{P}_{\text{noise}}$ is the response's noise distribution, finally we can let $p_1, \ldots, p_J$ be some arbitrary prior over data examples.*
4. *An XOR classification problem has $J = 4$, $\mu_1 = [v, v], \mu_2 = -[v, v], \mu_3 = [-v, v], \mu_4 = [v, -v]$ with $v \in \mathbb{R}^{d/2}$, $\mathbb{P}_j = \delta_0$ for $j \leq 2$ and $\mathbb{P}_j = \delta_1$ otherwise. Let each $p_j = 1/4$.*

**Example A.2.** *Many architectures utilized in practice can be used under our parametric models. Each example below is trained using either MSE loss $\frac{1}{2}(\hat{y} - y)^2$ or MAD loss $|\hat{y} - y|$ for simplicity.*

1. *A simple linear regression model is given by $\mathcal{M}_{\theta,a}(x) = x^\top\theta$.*
2. *A generalized linear model is given by $\mathcal{M}_{\theta,a}(x) = \sigma(x^\top\theta)$ for a link function $\sigma : \mathbb{R} \to \mathbb{R}$.*
3. *Let $L = L'$, and let $\sigma, \phi : \mathbb{R} \to \mathbb{R}$ be applied element-wise, a two-layer network is given by $\mathcal{M}_{\theta,a}(x) = \phi(\sigma(x^\top\theta)a)$. Deeper networks of finite width are given by setting $a = (\text{vec}(W_1), \ldots, \text{vec}(W_{k-1}), w_k)$ where $W_i \in \mathbb{R}^{L\times L}$ for $i \in [k-1]$, $w_k \in \mathbb{R}^L$ and selecting $\mathcal{M}_{\theta,a}(x) = \phi(\sigma(\cdots\sigma(\sigma(x^\top\theta)W_1)\cdots W_{k-1})w_k)$.*
4. *Corresponding to Example A.1 (4), we could consider a model for the XOR problem with $L = 2, \theta = (\theta_1, \theta_2), L' = 1$ and selecting $\mathcal{M}_{\theta,a}(x) = a(x^\top\theta_1)(x^\top\theta_2)$.*

*Remark* A.3 (Optimal Bayes Estimation Under Distribution (3.1)). If the practitioner has access to the underlying mixture probabilities $(p_j)_{j\in[J]}$, response laws $(\mathbb{P}_j)_{j\in[J]}$ and mode vectors $(\mu_j)$, then under squared loss there exists a closed form model which is Bayes optimal. Let $\bar{y}_j = \mathbb{E}_{\mathbb{P}_j}[y]$ for $(x, y)$ drawn from distribution (3.1), the optimal estimator $\mathcal{M}^*$ takes the form:

$$\mathcal{M}^*(x) = \mathbb{E}[y \mid x] = \sum_{j=1}^{J}\bar{y}_j\mathbb{P}(y \sim \mathbb{P}_j \mid x).$$

Recalling distribution (3.1), using Bayes' rule one can calculate that

$$\mathbb{P}(y \sim \mathbb{P}_j \mid x) = \frac{\exp(x^\top\mu_j + \log(p_j) - \|\mu_j\|_2^2/(2d))}{\sum_{k=1}\exp(x^\top\mu_k + \log(p_k) - \|\mu_k\|_2^2/(2d))}.$$

Thus, with $b_j = \ln p_j - \frac{\|\mu_j\|_2^2}{2d}$, $M = [\mu_1, \ldots, \mu_J] \in \mathbb{R}^{d\times J}$, and $\bar{y} = (\bar{y}_1, \ldots, \bar{y}_j)^\top \in \mathbb{R}^J$, we can equivalently write,

$$\mathcal{M}^*(x) = \bar{y}^\top\text{softmax}\left(M^\top x + b\right).$$

We clarify that a strength of our Decoupled Descent algorithm is that the train-test identity holds even if the model $\mathcal{M}$ differs from this Bayes-optimal $\mathcal{M}^*$.

### A.5 Main Assumption

**Assumption A.4.** The following hold when both $n, d \to \infty$, with $n/d \to \alpha \in (0, \infty)$ fixed:

1. [Data Composition] In distribution (3.1), for each $j, k \in [J]$, $\mathbb{P}_j$ has bounded moments of all orders, $p_j \in [0, 1]$ and $\lim_{d\to\infty} d^{-1}\mu_j^\top \mu_k$ exists. Further, each row of data $(X, y) \in \mathbb{R}^{n\times d} \times \mathbb{R}^n$ is drawn i.i.d. from distribution (3.1).
2. [Initialization Limits] Initialization $\theta_1$ is independent of the data $(X, y)$ and $\lim_{d\to\infty} d^{-1}\theta_1^\top\theta_1 = \bar{\theta}^2$ almost surely. For each $j \in [J]$, $\lim_{d\to\infty} d^{-1}\theta_1^\top\mu_j = m_{j,1}$ almost surely. The norm of the initialization $\|a_1\|_2$ is uniformly bounded.
3. [DD Lipschitzness] Uniformly over $y \in \mathbb{R}$, the functions $f(h, y, a)$, $g(h, y, a)$ and their derivatives $\nabla_h f(h, y, a)$, $\nabla_h g(h, y, a)$ are Lipschitz and bounded in $h$ and $a$.
4. [Model And Loss Smoothness] Let $\Psi(h, y, a)$ from Definition 2.1, uniformly over $y \in \mathbb{R}$:
   (a) $\Psi$ is bounded, Lipschitz, four times continuously differentiable with respect to $h$ and twice continuously differentiable with respect to $a$.
   (b) Let $\omega \in \mathbb{R}^{L\times L}$ be a covariance matrix with bounded operator norm, vectors $m \in \mathbb{R}^L$ and $a \in \mathbb{R}^{L'}$ with $\|m\|_2$, $\|a\|_2$ bounded, and let $h = m + \omega^{1/2}G$ where $G \sim \mathcal{N}(0, \mathrm{Id}_L)$. For each class $j \in [J]$ with $Y_j \sim \mathbb{P}_j$, there exists a bounded constant $C > 0$ where,
   $$\max\left(\|\mathbb{E}[\nabla_a^2\Psi(h, Y_j, a)]\|_{\mathrm{op}}, \|\mathbb{E}[\nabla_h\nabla_a\Psi(h, Y_j, a)]\|_{\mathrm{op}}, \|\mathbb{E}[\nabla_h^2\Psi(h, Y_j, a)]\|_{\mathrm{op}}\right) \le C$$
   $$\max\left(\|\mathbb{E}[\nabla_h^2\nabla_a\Psi(h, Y_j, a)]\|_{\mathrm{F}}, \|\mathbb{E}[\nabla_h^3\Psi(h, Y_j, a)]\|_{\mathrm{F}}, \|\mathbb{E}[\nabla_h^4\Psi(h, Y_j, a)]\|_{\mathrm{F}}\right) \le C$$

*Remark* A.5. As an example of a possible bounded Lipschitz extension from Remark 3.4 when $\mathcal{L}$ is square loss, $\cup_{j\in[J]}\mathrm{Supp}(\mathbb{P}_j) \in [-C, C]$ and model $\mathcal{M}_a(h) \in [-C, C]$, we can define

$$\Psi(h, y) = \frac{1}{2}\big(\tau(y) - \tau(\mathcal{M}_a(h))\big)^2,$$

where

$$\tau(u) = \int_0^u \psi\left(\frac{t}{C}\right) dt, \qquad \psi(s) = \begin{cases} 1, & |s| \le 1 \\ \exp\left(-\frac{(|s|-1)^2}{1-(|s|-1)^2}\right), & 1 < |s| < 2 \\ 0, & |s| \ge 2 \end{cases}.$$

We can see this definition of $\Psi$ is bounded, Lipschitz and sufficiently smooth while being exactly equal to square loss on the bounded domain of $y$ and $\mathcal{M}(h)$.

### A.6 The Asymptotic Test Error

We prove the following proposition.

**Proposition A.6.** *If Assumption A.4 holds and* $\lim_{d\to\infty} \mu_j^\top\theta/d = m_{j,\theta}$, $\lim_{d\to\infty} \theta^\top\theta/d = \Omega_\theta$, $\lim_{n,d\to\infty} a = \bar{a}$ *almost surely for trained parameters* $(\theta, a)$*, then there exists a deterministic function* **test** *depending only on the aforementioned limits such that,*

$$\lim_{d\to\infty} \mathbb{E}_{\check{x},\check{y}}[\mathcal{L}(\mathcal{M}_{\theta,a}(\check{x}), \check{y})] = \mathbf{test}(m_{1,\theta}, \dots, m_{J,\theta}, \Omega_\theta, \bar{a}).$$

*Proof.* Consider the test error from (1.2) with $\beta = (\theta, a)$ denoting $\check{L}(\theta, a) = \mathbb{E}_{\check{x},\check{y}}[\mathcal{L}(\mathcal{M}_{\theta,a}(\check{x}), \check{y})]$ where $(\check{x}, \check{y})$ are drawn from distribution (3.1), we write

$$\lim_{d\to\infty} \check{L}(\theta, a) = \lim_{d\to\infty} \sum_{j=1}^J p_j \mathbb{E}[\mathcal{L}(\mathcal{M}_a(\mu_j^\top\theta/d + \check{Z}_\theta), \check{Y}_j)],$$

where $\check{Z}_\theta \sim \mathcal{N}(0, \theta^\top\theta/d)$ and $\check{Y}_j \sim \mathbb{P}_j$. Under the assumption that $\mu_j^\top\theta/d$, $\theta^\top\theta/d$ and $a$ have almost sure limits $m_{j,\theta}$ (for each $j \in [J]$), $\Omega_\theta$ and $\bar{a}$ respectively, we have that

$$\lim_{d\to\infty} \check{L}(\theta, a) = \sum_{j=1}^J p_j \mathbb{E}_{\substack{\check{Z}_\theta\sim\mathcal{N}(0,\Omega_\theta) \\ \check{Y}_j\sim\mathbb{P}_j}}[\mathcal{L}(\mathcal{M}_{\bar{a}}(m_{j,\theta} + \check{Z}_\theta), \check{Y}_j)], \tag{A.3}$$

by dominated convergence as Assumption 3.3 assumes $\Psi(h, y, a) = \mathcal{L}(\mathcal{M}_a(h), y_j)$ is bounded. Noting the right hand side of (A.3) is from Definition 3.5 gives the proof. ∎

### A.7 Full Definition Of State Evolution

As we see momentarily, DD can be equivalently written as an AMP algorithm, thus off-the-shelf AMP results immediately provide a distributional characterization of the pre-activations $h_1, \dots, h_t \in \mathbb{R}^{n \times L}$ and parameters $\theta_1, \dots, \theta_t \in \mathbb{R}^{d \times L}$, $a_1, \dots, a_t \in \mathbb{R}^{L'}$. This description is given by a set of low-dimensional recursive equations termed *state evolution*. Informally, we derive a recursion of variables $\Omega_t, \Sigma_t, \Xi_t, m_{1,t}, \dots, m_{J,t}, \bar{a}_t$ which represent the almost sure limits of (self-)overlaps between $\theta_t, \mu_j$ and $\tilde{\theta}_t$ alongside the limiting value of $a_t$.

**Definition A.7.** Let $(\mathbb{P}_j)_{j\in[J]}, (\chi_{j,k})_{j,k\in[J]}, (p_j)_{j\in[J]}$ be from Assumption A.4 (1), $(m_{j,1})_{j\in[J]}, \bar{\theta}^2$ be the limits from Assumption A.4 (2) and let $g, f$ be the functions from Assumption A.4 (3).

Define the following state evolution parameters $m_{j,t} \in \mathbb{R}^L$ for $j \in [J], t \in [T]$, block matrices $\Sigma_t, \Omega_t \in (\mathbb{R}^{L\times L})^{(t+1)\times(t+1)}, \Xi_t \in (\mathbb{R}^{L\times L})^{t\times t}$ and $\bar{a}_t \in \mathbb{R}^{L'}$ for $t \in [T]$ recursively as follows. Recall the indexing $[r, s]$ denotes the $(r, s)$-th $L \times L$ block of a given matrix, we have

$$\Sigma_t[r,s] = \sum_{j=1}^{J} p_j \mathbb{E}[g(G^r + m_{j,r}, Y_j, \bar{a}_r) g(G^s + m_{j,s}, Y_j, \bar{a}_s)^\top]$$

$$\ell_{j,t} = p_j \mathbb{E}[g(G^t + m_{j,t}, Y_j, \bar{a}_t)]$$

$$\Xi_t[r+1,s] = \eta_0 \Xi_{t-1}[r,s] - \eta_1 \Big( \alpha \Sigma_t[r,s] + \alpha^2 \sum_{j,k=1}^{J} \chi_{j,k} \ell_{j,r} \ell_{k,s}^\top \Big), \qquad r \le t-1$$

$$\Omega_{t+1}[r+1,s+1] = \eta_0^2 \Omega_t[r,s] - \eta_0\eta_1 \left( \Xi_t[r,s] + \Xi_t[s,r]^\top \right) + \eta_1^2 \Big( \alpha \Sigma_t[r,s] + \alpha^2 \sum_{j,k=1}^{J} \chi_{j,k} \ell_{j,r} \ell_{k,s}^\top \Big)$$

$$m_{j,t+1} = \eta_0 m_{j,t} - \eta_1 \alpha \sum_{k=1}^{J} \chi_{j,k} \ell_{k,t}$$

$$\bar{a}_{t+1} = \gamma_0 \bar{a}_t - \gamma_1 \sum_{j=1}^{J} p_j \mathbb{E}[f(G^t + m_{j,t}, Y_j, \bar{a}_t)]$$

with indices $r, s \in [t]$ and $j \in [J]$. The expectations are taken over $Y_j \sim \mathbb{P}_j$, the random vectors $G^t \sim \mathcal{N}(0, \Omega_t[t,t])$, and the pairs $(G^r, G^s) \sim \mathcal{N}\left(0, \left[\begin{smallmatrix} \Omega_t[r,r] & \Omega_t[r,s] \\ \Omega_t[s,r] & \Omega_t[s,s] \end{smallmatrix}\right]\right)$. The system is initialized with $\bar{a}_1 = a_1$, $\Omega_t[1,1] = \Omega_1[1,1] = \bar{\theta}^2$, $\Xi_t[1,s] = \alpha \sum_{j=1}^{J} m_{j,1} \ell_{j,s}^\top$, and $\Omega_{t+1}[1, s+1] = \eta_0 \Omega_t[1,s] - \eta_1 \Xi_t[1,s] = \Omega_{t+1}[s+1,1]^\top$.

*Remark* A.8. As a special case of Definition A.7, we have the recursion,

$$\Omega_{t+1}[t+1,t+1] = \eta_0^2 \Omega_t[t,t] - \eta_1\eta_0 (\Xi_t[t,t] + \Xi_t[t,t]^\top) + \eta_1^2 \left( \alpha \Sigma_t[t,t] + \alpha^2 \sum_{j,k=1}^{J} \chi_{j,k} \ell_{j,t} \ell_{k,t}^\top \right).$$

### A.8 Approximate Message Passing And Proving Lemma 3.6

For Lemma 3.6 and Theorem 3.8, we require the following assumption.

**Assumption A.9.** Uniformly over $y \in \mathbb{R}$, function $\phi(h, y, a) : \mathbb{R}^L \times \mathbb{R} \times \mathbb{R}^{L'} \to \mathbb{R}$ is Lipschitz in $h$ and $a$. Moreover, there exists a bounded constant $C > 0$, independent of $t$, where the matrices from Definition A.7 satisfy $\max_{t\in[T]} \max(\|\Sigma_t^{-1}\|_{\text{op}}, \|\Omega_t^{-1}\|_{\text{op}}, \|\Sigma_T[t,t]\|_{\text{op}}, \|\Omega_T[t,t]\|_{\text{op}}) \le C$.

#### A.8.1 Relating Back To AMP

Consider the original DD algorithm dependent on the activation functions $g : \mathbb{R}^L \times \mathbb{R} \times \mathbb{R}^{L'} \to \mathbb{R}^L$ and $f : \mathbb{R}^L \times \mathbb{R} \times \mathbb{R}^{L'} \to \mathbb{R}^{L'}$ and hyperparameters $\eta_0, \eta_1, \gamma_0, \gamma_1$ from Equation (3.2), we repeat

the algorithm below for convenience,

$$\begin{aligned} h_t &= X\theta_t + \eta_1 \sum_{s=1}^{t-1} \eta_0^{(t-1)-s} \hat{h}_s \\ \hat{h}_t &= g(h_t, y, a_t) \\ \tilde{\theta}_t &= X^\top \hat{h}_t - \alpha\theta_t \left( \frac{1}{n} \sum_{i=1}^{n} \nabla_h g(h_{t,i}, y_i, a_t) \right)^\top \\ \theta_{t+1} &= \eta_0 \theta_t - \eta_1 \tilde{\theta}_t \\ a_{t+1} &= \gamma_0 a_t - \gamma_1 \frac{1}{n} \sum_{i=1}^{n} f(h_{t,i}, y_i, a_t). \end{aligned} \tag{A.4}$$

Note, the above algorithm is invariant to a permutation of the rows of $X$, thus we fix a representation with $X = \frac{1}{d} S + Z = \frac{1}{d} \sum_{j=1}^{J} e_{B_j} \mu_j^\top + Z$ where $B_1, \dots, B_J$ partition $[n]$ and $Z$ is an element-wise independent Gaussian matrix with $Z_{ij} \sim \mathcal{N}(0, 1/d)$. Moreover, by Assumption A.4 (1), we have the almost sure limits $\lim_{n\to\infty} |B_j|/n = p_j$.

Therefore, we can equivalently write (A.4) as,

$$\begin{aligned} h_t &= \sum_{j=1}^{J} e_{B_j} \frac{\mu_j^\top \theta_t}{d} + Z\theta_t + \eta_1 \sum_{s=1}^{t-1} \eta_0^{(t-1)-s} \hat{h}_s \\ \hat{h}_t &= g(h_t, y, a_t) \\ \tilde{\theta}_t &= \sum_{j=1}^{J} \mu_j \frac{n}{d} \frac{e_{B_j}^\top \hat{h}_t}{n} + Z^\top \hat{h}_t - \alpha\theta_t \left( \frac{1}{n} \sum_{i=1}^{n} \nabla_h g(h_{t,i}, y_i, a_t) \right)^\top \\ \theta_{t+1} &= \eta_0 \theta_t - \eta_1 \tilde{\theta}_t \\ a_{t+1} &= \gamma_0 a_t - \gamma_1 \frac{1}{n} \sum_{i=1}^{n} f(h_{t,i}, y_i, a_t). \end{aligned}$$

**We prove Lemma 3.6 in three steps.**

(1) We replace $\frac{\mu_j^\top \theta_t}{d}$, $\frac{e_{B_j}^\top \hat{h}_t}{d}$ and $a_t$ by pre-specified vectors $m_{j,t}^\top, \ell_{j,t}^\top \in \mathbb{R}^{1\times L}$ and $\bar{a}_t \in \mathbb{R}^{L'}$ respectively. Moreover, with $\tilde{Z} \in \mathbb{R}^{n\times d}$, we couple $Z = \alpha^{1/2}\tilde{Z}$ and replace $Z$ by $\alpha^{1/2}\tilde{Z}$ noting that $\tilde{Z}$ has independent $\mathcal{N}(0, 1/n)$ entries.

We then analyze the following *frozen* algorithm, recalling that $\alpha = n/d$,

$$\begin{aligned} h_t^{\text{frozen}} &= \sum_{j=1}^{J} e_{B_j} m_{j,t}^\top + \alpha^{1/2} \tilde{Z} \theta_t^{\text{frozen}} + \eta_1 \sum_{s=1}^{t-1} \eta_0^{(t-1)-s} \hat{h}_s^{\text{frozen}} \\ \hat{h}_t^{\text{frozen}} &= g(h_t^{\text{frozen}}, y, \bar{a}_t) \\ \tilde{\theta}_t^{\text{frozen}} &= \alpha \sum_{j=1}^{J} \mu_j \ell_{j,t}^\top + \alpha^{1/2} \tilde{Z}^\top \hat{h}_t^{\text{frozen}} - \alpha\theta_t^{\text{frozen}} \left( \frac{1}{n} \sum_{i=1}^{n} \nabla_h g(h_{t,i}^{\text{frozen}}, y_i, \bar{a}_t) \right)^\top \\ \theta_{t+1}^{\text{frozen}} &= \eta_0 \theta_t^{\text{frozen}} - \eta_1 \tilde{\theta}_t^{\text{frozen}}. \end{aligned} \tag{A.5}$$

The state evolution for this algorithm is given in Appendix A.8.2.

(2) We prove that if $m_{j,t}^\top$, $\ell_{j,t}^\top$ and $\bar{a}_t$ are the almost sure limits of $\frac{\mu_j^\top \theta_t}{d}$, $\frac{e_{B_j}^\top \hat{h}_t}{d}$ and $a_t$ respectively, then for a suitable class of test functions, say represented by $\phi : \mathbb{R}^{n\times(L\cdot T)} \times \mathbb{R}^n \times \mathbb{R}^{L'\cdot T} \to \mathbb{R}^n$ and $\varphi : \mathbb{R}^{d\times(L\cdot T)} \times \mathbb{R}^{L'\cdot T} \to \mathbb{R}^n$, we have that both,

$$\lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} \phi(h_1, \dots, h_T, y, a_1, \dots, a_T)_i - \frac{1}{n} \sum_{i=1}^{n} \phi(h_1^{\text{frozen}}, \dots, h_T^{\text{frozen}}, y, \bar{a}_1, \dots, \bar{a}_T)_i = 0$$

and

$$\lim_{n\to\infty} \frac{1}{d}\sum_{i=1}^{d} \varphi(\tilde{\theta}_1,\dots,\tilde{\theta}_T,a_1,\dots,a_T)_i - \frac{1}{d}\sum_{i=1}^{d} \varphi(\tilde{\theta}_1^{\text{frozen}},\dots,\tilde{\theta}_T^{\text{frozen}},\bar{a}_1,\dots,\bar{a}_T)_i = 0,$$

almost surely. This is the content of Appendix A.8.3.

(3) We derive the desired almost sure limits $m_{j,t}$, $\ell_{j,t}$ and $\bar{a}_t$ in Appendix A.8.4. Combining these three steps proves Lemma 3.6.

### A.8.2 A Re-parameterized State Evolution

For this subsection, we drop the superscript *frozen* for notational convenience. We continue with our analysis of (A.5), consider the change of variables

$$\begin{aligned} z_t &= \alpha^{-1/2}\left(h_t - \sum_{j=1}^{J} e_{B_j} m_{j,t}^\top\right), \quad h_t = \alpha^{1/2} z_t + \sum_{j=1}^{J} e_{B_j} m_{j,t}^\top \\ w_t &= \alpha^{-1/2}\left(\tilde{\theta}_t - \alpha\sum_{j=1}^{J} \mu_j \ell_{j,t}^\top\right), \quad \tilde{\theta}_t = \alpha^{1/2} w_t + \alpha\sum_{j=1}^{J} \mu_j \ell_{j,t}^\top. \end{aligned} \tag{A.6}$$

Then, to analyze algorithm (A.5), it suffices to analyze the following algorithm and undo the above change of variables,

$$\begin{aligned} z_t &= \tilde{Z}\theta_t + \alpha^{-1/2}\eta_1 \sum_{s=1}^{t-1} \eta_0^{(t-1)-s}\hat{h}_s \\ \hat{h}_t &= g\left(\alpha^{1/2} z_t + \sum_{j=1}^{J} e_{B_j} m_{j,t}^\top, y, \bar{a}_t\right) \\ w_t &= \tilde{Z}^\top \hat{h}_t - \alpha^{1/2}\theta_t \left(\frac{1}{n}\sum_{i=1}^{n} \nabla_h g\left(\left(\alpha^{1/2} w_t + \alpha\sum_{j=1}^{J}\mu_j \ell_{j,t}^\top\right)_i, y_i, \bar{a}_t\right)\right)^\top \\ \theta_{t+1} &= \eta_0^t \theta_1 - \eta_1 \sum_{s=1}^{t} \eta_0^{t-s}\left(\alpha^{1/2} w_s + \alpha\sum_{j=1}^{J}\mu_j \ell_{j,s}^\top\right), \end{aligned} \tag{A.7}$$

where the final equality comes from unrolling the recursion of $\theta_{t+1} = \eta_0\theta_t - \eta_1\tilde{\theta}_t$ and changing variables. Then, we denote

$$\begin{aligned} G(z_t) &= g\left(\alpha^{1/2} z_t + \sum_{j=1}^{J} e_{B_j} m_{j,t}^\top, y, \bar{a}_t\right) \\ F_t(w_1,\dots,w_t) &= \eta_0^t\theta_1 - \eta_1\sum_{s=1}^{t}\eta_0^{t-s}\left(\alpha^{1/2} w_s + \alpha\sum_{j=1}^{J}\mu_j\ell_{j,s}^\top\right) \end{aligned} \tag{A.8}$$

Now, treating $y, \mu_1,\dots,\mu_J, e_{B_1},\dots,e_{B_j}, \theta_1$ as fixed vectors, we can equivalently write (A.7) as,

$$\begin{aligned} z_t &= \tilde{Z}\theta_t - \sum_{s=1}^{t} \hat{h}_s \mathbf{B}_{t,s}^\top \\ \hat{h}_t &= G(z_t) \\ w_t &= \tilde{Z}^\top \hat{h}_t - \theta_t \mathbf{C}_t^\top \\ \theta_{t+1} &= F_t(w_t), \end{aligned} \tag{A.9}$$

where, denoting the derivative with respect to a specified variable $(\cdot)$ as $\partial_{(\cdot)}$,

$$\begin{aligned}
\mathbf{B}_{t,s} &= \frac{1}{n}\sum_{i=1}^{d} \partial_{w_{s,i}} F_{t-1}(w_{1,i},\dots,w_{t-1,i}) = \frac{1}{d}\alpha^{-1}\sum_{i=1}^{d} \partial_{w_{s,i}} F_{t-1}(w_{1,i},\dots,w_{t-1,i}) \\
&= -\eta_1 \eta_0^{(t-1)-s}\alpha^{-1/2}\mathrm{Id} \\
\mathbf{C}_t &= \frac{1}{n}\sum_{i=1}^{n} \partial_{z_{t,i}} G(z_{t,i}) = \alpha^{1/2} n^{-1}\sum_{i=1}^{n} \nabla_h g(\alpha^{1/2} z_{t,i} + m_{j(i),t}^\top, y_i, \bar{a}_t)^\top,
\end{aligned}$$

with $j(i)$ mapping a coordinate $i \in [n]$ to its corresponding block index of $B_1,\dots,B_J$. Note, we can immediately check that $G$ is Lipschitz as $g$ is Lipschitz in $z_t$ (see Assumption A.4 (3)) where,

$$\begin{aligned}
\|G(z_t) - G(\tilde{z}_t)\|_{\mathrm{F}} &= \left\| g\left(\alpha^{1/2} z_t + \sum_{j=1}^{J} e_{B_j} m_{j,t}^\top, y, \bar{a}_t\right) - g\left(\alpha^{1/2}\tilde{z}_t + \sum_{j=1}^{J} e_{B_j} m_{j,t}^\top, y, \bar{a}_t\right)\right\|_{\mathrm{F}} \\
&\le L\|\alpha^{1/2}(z_t - \tilde{z}_t)\|_{\mathrm{F}} \\
&\le (L\alpha^{1/2})\|z_t - \tilde{z}_t\|_{\mathrm{F}},
\end{aligned}$$

and $F_t$ is clearly Lipschitz in $w_1,\dots,w_t$ as it is a linear function of $w_1,\dots,w_t$ with bounded coefficients.

We can then see that algorithm (A.9) is a full-history matrix-valued extension of the non-separable AMP theory from [5] (or see [26] for a full-history vector-valued AMP algorithm). Using the standard construction of reformatting the vector amp iterates (temporarily overloading the definition of the iterates $z_t, w_t$) $(z_1, w_1),\dots,(z_{t\cdot L}, w_{t\cdot L}) \in \mathbb{R}^{2\times n}$ into matrix amp iterates $z_t^{\mathrm{mat}}$ and $w_t^{\mathrm{mat}}$ with

$$z_t^{\mathrm{mat}} = [z_{(t-1)+1},\dots,z_{t\cdot L-1}], \quad w_t^{\mathrm{mat}} = [w_{(t-1)+1},\dots,w_{t\cdot L-1}],$$

we can conclude that (A.9) uses the correct Onsager correction terms and is a valid rectangular AMP algorithm. We leave further details of matrix valued AMP iterates to the survey [13, Section 6.7] and [19]. We then have the following low dimensional prescription of the iterates $(z_t, w_t)_{t\in[T]}$ from (A.9).

**Definition A.10.** Given the functions $G, F_t$ from (A.8) and deterministic vector $\theta_1$, with initialization $\frac{\alpha^{-1}}{d}\theta_1^\top\theta_1 = \Omega_1^z$, we recursively define the sequence of matrices $\Sigma_1^w,\dots,\Sigma_T^w,\Omega_2^z,\dots,\Omega_{T+1}^z$ with

$$\begin{aligned}
\Sigma_t^w[r,s] &= \frac{1}{n}\mathbb{E}[G(Z_r)^\top G(Z_s)], \quad r,s\in[t] \\
(Z_r, Z_s) &\sim \mathcal{N}\left(0, \begin{bmatrix}\Omega_t^z[r,r] & \Omega_t^z[r,s] \\ \Omega_t^z[s,r] & \Omega_t^z[s,s]\end{bmatrix} \otimes \mathrm{Id}_n\right) \\
\Omega_{t+1}^z[r+1,s+1] &= \frac{1}{n}\mathbb{E}[F_r(W_1,\dots,W_r)^\top F_s(W_1,\dots,W_s)], \quad r,s\in[t] \\
&= \frac{\alpha^{-1}}{d}\mathbb{E}[F_r(W_1,\dots,W_r)^\top F_s(W_1,\dots,W_s)] \\
\Omega_{t+1}^z[1,s+1] &= \frac{\alpha^{-1}}{d}\theta_1^\top\mathbb{E}[F_s(W_1,\dots,W_s)] = \Omega_{t+1}^z[s+1,1]^\top, \quad s\in[t] \\
\Omega_{t+1}^z[1,1] &= \Omega_1^z \\
(W_1,\dots,W_t) &\sim \mathcal{N}(0, \Sigma_t^w \otimes \mathrm{Id}_d).
\end{aligned}$$

**Lemma A.11** ([26], Theorem 3.3 (extended to matrix-valued AMP algorithms))**.** *Let $\phi_1,\phi_2 : \mathbb{R}^{n\times(L\cdot T)} \to \mathbb{R}^n, \varphi_1,\varphi_2 : \mathbb{R}^{d\times(L\cdot T)} \to \mathbb{R}^d$ each be uniformly Lipschitz function in $n, d$. Let $(z_t, w_t)$ be the iteration (A.9), and denote $(Z_1,\dots,Z_T) \sim \mathcal{N}(0, \Omega_T^z \otimes \mathrm{Id}_n)$, $(W_1,\dots,W_T) \sim \mathcal{N}(0, \Sigma_T^w \otimes \mathrm{Id}_d)$ from Definition A.10. If $\max_{t\in[T]}\max(\|\Sigma_t^w\|_{\mathrm{op}}, \|(\Sigma_t^w)^{-1}\|_{\mathrm{op}}, \|\Omega_t^z\|_{\mathrm{op}}, \|(\Omega_t^z)^{-1}\|_{\mathrm{op}})$ is bounded almost surely, then the following limits hold almost surely,*

$$\lim_{n\to\infty}\frac{1}{n}\sum_{i=1}^{n}\phi(z_1,\dots,z_T)_i - \lim_{n\to\infty}\frac{1}{n}\sum_{i=1}^{n}\mathbb{E}[\phi(Z_1,\dots,Z_T)_i] = 0$$

$$\lim_{d\to\infty}\frac{1}{d}\sum_{i=1}^{d}\varphi(w_1,\dots,w_T)_i-\lim_{d\to\infty}\frac{1}{d}\sum_{i=1}^{d}\mathbb{E}[\varphi(W_1,\dots,W_T)_i]=0,$$

*where* $\phi(\cdot)_i=\phi_1(\cdot)_i\phi_2(\cdot)_i$ *and* $\varphi(\cdot)_i=\varphi_1(\cdot)_i\varphi_2(\cdot)_i$.

**Definition A.12.** Let Assumption A.4 hold with $(\mathbb{P}_j)_{j\in[J]}$ from Assumption A.4 (1), $\bar{\theta}^2$ from Assumption A.4 (2), and $g$ be from Assumption A.4 (3).

Given initialization $\check{\Omega}_1=\bar{\theta}^2$, sequences $(m_{j,t})_{j\in[J],t\in[T]}$, $(\ell_{j,t})_{j\in[J],t\in[T]}$ and $(\bar{a}_t)_{t\in[T]}$, we recursively define the sequence of matrices $\check{\Sigma}_1,\dots,\check{\Sigma}_T,\check{\Omega}_2,\dots,\check{\Omega}_{T+1}$ with

$$\check{\Sigma}_t[r,s]=\sum_{j=1}^{J}p_j\mathbb{E}[g(\check{Z}_r+m_{j,r},Y_j,\bar{a}_r)g(\check{Z}_s+m_{j,s},Y_j,\bar{a}_s)^\top],\quad r,s\in[t]$$

$$(\check{Z}_r,\check{Z}_s)\sim\mathcal{N}\left(0,\begin{bmatrix}\check{\Omega}_t[r,r] & \check{\Omega}_t[r,s]\\ \check{\Omega}_t[s,r] & \check{\Omega}_t[s,s]\end{bmatrix}\right)$$

$$\check{\Xi}_t[r+1,s]=\eta_0\check{\Xi}_{t-1}[r,s]-\eta_1\Big(\alpha\check{\Sigma}_t[r,s]+\alpha^2\sum_{j,k=1}^{J}\chi_{j,k}\ell_{j,r}\ell_{k,s}^\top\Big),\quad r\in[t-1],s\in[t]$$

$$\check{\Xi}_t[1,s]=\alpha\sum_{j=1}^{J}m_{j,1}\ell_{j,s}^\top,\quad s\in[t]$$

$$\check{\Omega}_{t+1}[r+1,s+1]=\eta_0^2\check{\Omega}_t[r,s]-\eta_0\eta_1\left(\check{\Xi}_t[r,s]+\check{\Xi}_t[s,r]^\top\right)+\eta_1^2\Big(\alpha\check{\Sigma}_t[r,s]+\alpha^2\sum_{j,k=1}^{J}\chi_{j,k}\ell_{j,r}\ell_{k,s}^\top\Big)$$

$$r,s\in[t]$$

$$\check{\Omega}_{t+1}[1,s+1]=\eta_0\check{\Omega}_t[1,s]-\eta_1\check{\Xi}_t[1,s]=\check{\Omega}_{t+1}[s+1,1]^\top,\quad s\in[t]$$

$$\check{\Omega}_{t+1}[1,1]=\bar{\theta}^2,$$

**Corollary A.13.** *Let* $(h_t,\hat{h}_t,\tilde{\theta}_t,\theta_t)$ *be the iteration from* (A.5) *and let Assumption A.4 hold. The following limits hold almost surely, for* $j\in[J],r,s,t\in[T]$,

$$\lim_{d\to\infty}\left|\frac{1}{d}\mu_j^\top\theta_t-\frac{1}{d}\mu_j^\top\mathbb{E}[(\alpha^{1/2}\check{W}_t+\alpha\sum_{j=1}^{J}\mu_j\ell_{j,t}^\top)]\right|=0$$

$$\lim_{d\to\infty}\frac{1}{d}\theta_r^\top\theta_s=\check{\Omega}_T[r,s]$$

$$\lim_{d\to\infty}\frac{1}{d}\theta_r^\top\tilde{\theta}_s=\check{\Xi}[r,s]$$

$$\lim_{n\to\infty}\frac{1}{n}\hat{h}_r^\top\hat{h}_s=\check{\Sigma}_T[r,s],$$

*where* $(\check{W}_1,\dots,\check{W}_T)\sim\mathcal{N}(0,\check{\Sigma}_T\otimes\mathrm{Id}_d)$ *from Definition A.12 and* $Y_j\sim\mathbb{P}_j$.

*Moreover, if* $\bar{\phi}(h,y,a,j):\mathbb{R}^L\times\mathbb{R}\times\mathbb{R}^{L'}\times[J]\to\mathbb{R}$ *is a Lipschitz function in* $h$ *uniformly over* $y,a,j$, *then the following limit also holds almost surely,*

$$\lim_{n,d\to\infty}\frac{1}{n}\sum_{i=1}^{n}\bar{\phi}(h_{t,i},y_i,\bar{a}_t,j(i))=\sum_{j\in[J]}p_j\mathbb{E}[\bar{\phi}(m_{j,t}+\check{Z}_t,Y_j,\bar{a}_t,j)],$$

*where* $j(i)$ *maps coordinate* $i$ *to its block* $B_j$ *with* $i\in B_j$ *and* $(\check{Z}_1,\dots,\check{Z}_T)\sim\mathcal{N}(0,\check{\Omega}_T)$ *from Definition A.12.*

*Proof.* First, we prove the following inductive claim, for all $t\in[T]$, there exists almost sure limits $\tilde{\Omega}_t^z,\tilde{\Sigma}_t^w$ of $\Omega_t^z,\Sigma_t^w$ as $d\to\infty$ which are recursively defined by the following system initialized at

$\tilde{\Omega}_1^z = \alpha^{-1}\bar{\theta}^2$ (with $\bar{\theta}^2$ from Assumption A.4) where $Y_j \sim \mathbb{P}_j$,

$$\begin{aligned}
\tilde{\Sigma}_t^w[r,s] &= \sum_{j=1}^J p_j \mathbb{E}[g(\alpha^{1/2}\tilde{Z}_r + m_{j,r}, Y_j, \bar{a}_r) g(\alpha^{1/2}\tilde{Z}_s + m_{j,s}, Y_j, \bar{a}_s)^\top], \quad r,s \in [t] \\
(\tilde{Z}_r, \tilde{Z}_s) &\sim \mathcal{N}\left(0, \begin{bmatrix} \tilde{\Omega}_t^z[r,r] & \tilde{\Omega}_t^z[r,s] \\ \tilde{\Omega}_t^z[s,r] & \tilde{\Omega}_t^z[s,s] \end{bmatrix}\right) \\
\tilde{\Xi}_t[r+1,s] &= \eta_0 \tilde{\Xi}_{t-1}[r,s] - \eta_1 \Big(\alpha \tilde{\Sigma}_t^w[r,s] + \alpha^2 \sum_{j,k=1}^J \chi_{j,k} \ell_{j,r} \ell_{k,s}^\top\Big), \quad r,s \in [t] \\
\tilde{\Xi}_t[1,s] &= \alpha \sum_{j=1}^J m_{j,1} \ell_{j,s}^\top, \quad s \in [t] \\
\tilde{\Omega}_{t+1}^z[r+1,s+1] &= \tilde{\Omega}_t^z[r,s] - \alpha^{-1}\tilde{\Xi}_t[r,s] - \alpha^{-1}\tilde{\Xi}_t[s,r]^\top + \alpha^{-1}\Big(\alpha\tilde{\Sigma}_t^w[r,s] + \alpha^2 \sum_{j,k=1}^J \chi_{j,k}\ell_{j,r}\ell_{k,s}^\top\Big), \\
&\qquad r,s \in [t] \\
\tilde{\Omega}_{t+1}^z[1,s+1] &= \eta_0 \tilde{\Omega}_t^z[1,s] - \alpha^{-1}\eta_1 \tilde{\Xi}_t[1,s] = \tilde{\Omega}_{t+1}[s+1,1]^\top, \quad s \in [t] \\
\tilde{\Omega}_{t+1}^z[1,1] &= \alpha^{-1}\bar{\theta}^2.
\end{aligned} \tag{A.10}$$

We proceed by induction. For each $t \in [T]$, we first prove the almost sure limit of $\Omega_t^z$ and then the almost sure limit of $\Sigma_t^w$, deriving the limits of intermediate variables in Definition A.10 between these steps. The base case follows immediately as $\lim_{d\to\infty} \Omega_1^z = \lim_{d\to\infty} \alpha^{-1}\frac{\theta_1^\top \theta_1}{d} = \alpha^{-1}\bar{\theta}^2$ by Assumption A.4. Then, with $Z_1 \sim \mathcal{N}(0, \Omega_1^z \otimes \mathrm{Id}_n)$, we write that

$$\begin{aligned}
\Sigma_1^w &= \frac{1}{n}\mathbb{E}\left[g\left(\alpha^{1/2}Z_1 + \sum_{j=1}^J e_{B_j} m_{j,1}^\top, y, \bar{a}_1\right)^\top g\left(\alpha^{1/2}Z_1 + \sum_{j=1}^J e_{B_j} m_{j,1}^\top, y, \bar{a}_1\right)\right] \\
&= \sum_{j=1}^J \frac{|B_j|}{n}\frac{1}{|B_j|}\sum_{i\in B_j} \mathbb{E}[g(\alpha^{1/2}Z_{1,i}^\top + m_{j(i),1}, y_i, \bar{a}_1) g(\alpha^{1/2}Z_{1,i}^\top + m_{j(i),1}, y, \bar{a}_1)^\top].
\end{aligned}$$

Then, applying the strong law of large numbers on the set of coordinates $i \in B_j$ for the random variables $y_i \sim \mathbb{P}_j$, the continuous mapping theorem for the almost sure limit $\lim_{n\to\infty} |B_j|/n = p_j$, and the continuous mapping theorem applied to the variance $Z_{1,i}$ (i.e. $\Omega_1^z$ with almost sure limit $\tilde{\Omega}_1^z$) using that $g$ is a continuous function, gives that the almost sure limit of $\Sigma_1^w$ is

$$\tilde{\Sigma}_1^w = \sum_{j=1}^J p_j \mathbb{E}[g(\alpha^{1/2}\tilde{Z}_1 + m_{j,1}, Y_j, \bar{a}_1) g(\alpha^{1/2}\tilde{Z}_1 + m_{j,1}, Y_j, \bar{a}_1)^\top],$$

where $\tilde{Z}_1 \sim \mathcal{N}(0, \tilde{\Omega}_1^z)$.

Now, we assume the inductive claim holds up to time $t-1$, i.e. the matrices $\tilde{\Omega}_{t-1}^z$ and $\tilde{\Sigma}_{t-1}^w$ are the almost sure limits of the matrices $\Omega_{t-1}^z$ and $\Sigma_{t-1}^w$ respectively.

Recall the choice of functions $G, F_t$ from (A.8), we can then rewrite $\Omega_t^z$, with $r,s \in [t-1]$, as

$$\begin{aligned}
\Omega_t^z[r+1,s+1] &= \frac{\alpha^{-1}}{d}\mathbb{E}\left[\left(\eta_0^r\theta_1 - \eta_1\sum_{p=1}^r \eta_0^{r-p}\left(\alpha^{1/2}W_p + \alpha\sum_{j=1}^J \mu_j \ell_{j,p}^\top\right)\right)^\top \left(\eta_0^s\theta_1 - \eta_1\sum_{q=1}^s \eta_0^{s-q}\left(\alpha^{1/2}W_q + \alpha\sum_{k=1}^J \mu_k \ell_{k,q}^\top\right)\right)\right] \\
&= \frac{\alpha^{-1}}{d}\mathbb{E}\Bigg[\Bigg\{\eta_0 \underbrace{\left(\eta_0^{r-1}\theta_1 - \eta_1\sum_{p=1}^{r-1}\eta_0^{(r-1)-p}\left(\alpha^{1/2}W_p + \alpha\sum_{j=1}^J \mu_j\ell_{j,p}^\top\right)\right)}_{\mathcal{A}_{r-1}} - \eta_1 \underbrace{\left(\alpha^{1/2}W_r + \alpha\sum_{j=1}^J \mu_j \ell_{j,r}^\top\right)}_{\mathcal{B}_r}\Bigg\}^\top
\end{aligned}$$

$$\times \left\{ \eta_0 \underbrace{\left( \eta_0^{s-1}\theta_1 - \eta_1 \sum_{q=1}^{s-1} \eta_0^{(s-1)-q} \left( \alpha^{1/2} W_q + \alpha \sum_{k=1}^{J} \mu_k \ell_{k,q}^\top \right) \right)}_{\mathcal{A}_{s-1}} - \eta_1 \underbrace{\left( \alpha^{1/2} W_s + \alpha \sum_{k=1}^{J} \mu_k \ell_{k,s}^\top \right)}_{\mathcal{B}_s} \right\} \Bigg]$$

$$= \alpha^{-1} \frac{1}{d} \left( \mathbb{E}[\eta_0^2 \mathcal{A}_{r-1}^\top \mathcal{A}_{s-1} - \eta_0 \eta_1 (\mathcal{A}_{r-1}^\top \mathcal{B}_s + \mathcal{B}_r^\top \mathcal{A}_{s-1}) + \eta_1^2 \mathcal{B}_r^\top \mathcal{B}_s] \right)$$

where $(W_1, \ldots, W_{t-1}) \sim \mathcal{N}(0, \Sigma_{t-1}^w \otimes \mathrm{Id}_d)$. Moreover, we have the following simplifications,

$$\frac{\alpha^{-1}}{d} \mathbb{E}[\mathcal{A}_{r-1}^\top \mathcal{A}_{s-1}] = \Omega_{t-1}^z[r,s], \qquad \text{(As } \mathcal{A}_{r-1} \text{ is } F_{r-1}(W_1, \ldots, W_r) \text{ and similarly for } \mathcal{A}_{s-1})$$

$$\frac{1}{d} \mathbb{E}[\mathcal{A}_{r-1}^\top \mathcal{B}_s] = \frac{1}{d} \mathbb{E} \left[ \left( \eta_0^{r-1}\theta_1 - \eta_1 \sum_{p=1}^{r-1} \eta_0^{(r-1)-p} \mathcal{B}_p \right)^\top \mathcal{B}_s \right],$$

$$\frac{1}{d} \mathbb{E}[\mathcal{B}_r^\top \mathcal{A}_{s-1}] = \frac{1}{d} \mathbb{E} \left[ \mathcal{B}_r^\top \left( \eta_0^{s-1}\theta_1 - \eta_1 \sum_{q=1}^{s-1} \eta_0^{(s-1)-q} \mathcal{B}_q \right) \right],$$

$$\frac{1}{d} \mathbb{E}[\mathcal{B}_r^\top \mathcal{B}_s] = \frac{1}{d} \Bigg( \alpha \mathbb{E}[W_r^\top W_s] + \alpha^{3/2} \sum_{k=1}^{J} \mathbb{E}[W_r]^\top \mu_k \ell_{k,r}^\top + \alpha^{3/2} \sum_{j=1}^{J} \ell_{j,r} \mu_j^\top \mathbb{E}[W_s] + \alpha^2 \sum_{j,k=1}^{J} \ell_{j,r} \mu_j^\top \mu_k \ell_{k,s}^\top \Bigg)$$

$$= \alpha \frac{\mathbb{E}[W_r^\top W_s]}{d} + \alpha^2 \sum_{j,k=1}^{J} \ell_{j,r} \frac{\mu_j^\top \mu_k}{d} \ell_{k,s}^\top$$

$$= \alpha \Sigma_{t-1}^w[r,s] + \alpha^2 \sum_{j,k=1}^{J} \ell_{j,r} \frac{\mu_j^\top \mu_k}{d} \ell_{k,s}^\top$$

as $\mathbb{E}[W_r] = 0$. This in turn implies that,

$$\Omega_t^z[r+1, s+1] = \eta_0^2 \Omega_{t-1}^z[r,s] - \alpha^{-1} \eta_0 \eta_1 (\Xi_{t-1}^z[r,s] + \Xi_{t-1}^z[s,r]^\top) + \alpha^{-1} \eta_1^2 \left( \alpha \Sigma_{t-1}^w[r,s] + \alpha^2 \sum_{j,k=1}^{J} \ell_{j,r} \frac{\mu_j^\top \mu_k}{d} \ell_{k,s}^\top \right), \tag{A.11}$$

where, with initialization $\Xi_t^z[1,s] = \frac{1}{d} \theta_1^\top \mathbb{E}[\mathcal{B}_s]$ and $r \in [t-2], s \in [t-1]$, we recursively define,

$$\Xi_{t-1}^z[r+1,s] = \frac{1}{d} \mathbb{E} \left[ \left( \eta_0 \theta_1 - \eta_1 \sum_{p=1}^{r} \eta_{\theta,0}^{r-p} \mathcal{B}_p \right)^\top \mathcal{B}_s \right]$$

$$= \frac{1}{d} \mathbb{E} \left[ \left( \eta_0 \left( \eta_0 \theta_1 - \eta_1 \sum_{p=1}^{r-1} \eta_{\theta,0}^{(r-1)-p} \mathcal{B}_p \right) - \eta_1 \mathcal{B}_r \right)^\top \mathcal{B}_s \right]$$

$$= \eta_0 \Xi_{t-2}^z[r,s] - \eta_1 \frac{1}{d} \mathbb{E} \left[ \mathcal{B}_r^\top \mathcal{B}_s \right]$$

$$= \eta_0 \Xi_{t-2}^z[r,s] - \eta_1 \left( \alpha \Sigma_{t-1}^w[r,s] + \alpha^2 \sum_{j,k=1}^{J} \ell_{j,r} \frac{\mu_j^\top \mu_k}{d} \ell_{k,s}^\top \right).$$

By the inductive hypothesis, we have the covariance matrix $\Sigma_{t-1}^w$ has the almost sure limit $\tilde{\Sigma}_{t-1}^w$ and the covariance matrix $\Omega_{t-1}^z$ has the almost sure limit $\tilde{\Omega}_{t-1}^z$. Thus, under equation (A.11), we apply the strong law of large numbers on the coordinates $i \in B_j$ for the independent random variables $y_i \sim \mathbb{P}_j$, the continuous mapping theorem for the almost sure limit $\lim_{n\to\infty} |B_j|/n = p_j$, limit $\lim_{d\to\infty} \frac{\mu_j^\top \mu_k}{d} = \chi_{j,k}$ from Assumption A.4 (1) and limit $\lim_{d\to\infty} \frac{\theta_1^\top \mu_j}{d} = m_{j,1}$ from Assumption A.4 (2) to give that $\tilde{\Omega}_t^z$ has the following almost sure limit, with $r, s \in [t-1]$,

$$\tilde{\Omega}_t^z[r+1, s+1] = \eta_0^2 \tilde{\Omega}_{t-1}^z[r,s] - \alpha^{-1} \eta_0 \eta_1 (\tilde{\Xi}_{t-1}[r,s] + \tilde{\Xi}_{t-1}[s,r]^\top)$$

$$+ \alpha^{-1}\eta_1^2 \left( \alpha \tilde{\Sigma}^w_{t-1}[r,s] + \alpha^2 \sum_{j,k=1}^{J} \chi_{j,k} \ell_{j,r} \ell_{k,s}^\top \right),$$

where we recursively define, with $r \in [t-2]$ and $s \in [t-1]$,

$$\tilde{\Xi}_{t-1}[r+1,s] = \eta_0 \tilde{\Xi}_{t-2}[r,s] - \eta_1 \left( \alpha \tilde{\Sigma}^w_{t-1}[r,s] + \alpha^2 \sum_{j,k=1}^{J} \chi_{j,k} \ell_{j,r} \ell_{k,s}^\top \right),$$

with $\tilde{\Xi}_{t-1}[1,s] = \alpha \sum_{j=1}^{J} m_{j,1} \ell_{j,s}^\top$.

Moreover, we can write

$$\begin{aligned}
\Omega^z_t[1,s+1] = \frac{\alpha^{-1}}{d} \theta_1^\top \mathbb{E} \Bigg[ \eta_0 \left( \eta_0^{s-1} \theta_1 - \eta_1 \sum_{q=1}^{s-1} \eta_0^{(s-1)-q} \left( \alpha^{1/2} W_q + \alpha \sum_{k=1}^{J} \mu_k \ell_{k,q}^\top \right) \right) \\
- \eta_1 \left( \alpha^{1/2} W_s + \alpha \sum_{k=1}^{J} \mu_k \ell_{k,s}^\top \right) \Bigg] \\
= \eta_0 \Omega^z_{t-1}[1,s] - \alpha^{-1} \eta_1 \Xi^z_{t-1}[1,s]
\end{aligned}$$

as $\mathbb{E}[W_q] = 0$, which can be argued identically as above to have the almost sure limit,

$$\tilde{\Omega}^z_t[1,s+1] = \eta_0 \tilde{\Omega}^z_{t-1}[1,s] - \alpha^{-1} \eta_1 \tilde{\Xi}_{t-1}[1,s].$$

As a final step, we can use the aforementioned argument on the convergence of $\Omega^z_1$ to give that $\Omega^z_t[1,1]$ has the almost sure limit $\tilde{\Omega}^z_t[1,1] = \tilde{\Omega}^z_1[1,1]$.

Then, we can write $\Sigma^w_t$ as,

$$\begin{aligned}
\Sigma^w_t[r,s] &= \frac{1}{n} \mathbb{E} \left[ g \left( \alpha^{1/2} Z_r + \sum_{j=1}^{J} e_{B_j} m_{j,r}^\top, y, \bar{a}_r \right) g \left( \alpha^{1/2} Z_s + \sum_{j=1}^{J} e_{B_j} m_{j,s}^\top, y, \bar{a}_s \right) \right] \\
&= \sum_{j=1}^{J} \frac{|B_j|}{n} \frac{1}{|B_j|} \sum_{i \in B_j} \mathbb{E}[g(\alpha^{1/2} Z_{r,i} + m_{j(i),r}, y_i, \bar{a}_r)^\top g(\alpha^{1/2} Z_{s,i} + m_{j(i),s}, y_i, \bar{a}_s)^\top]
\end{aligned}$$

where $(Z_1, \ldots, Z_t) \sim \mathcal{N}(0, \Omega^z_t \otimes \mathrm{Id}_n)$.

An application of the strong law of large numbers over the block $B_j$ where each $y_i \sim \mathbb{P}_j$ independently and the continuous mapping theorem on the covariance of $Z_{r,i}$ and $Z_{s,i}$, using that $g$ is a continuous function, then gives that $\Sigma^w_t$ has the almost sure limit,

$$\tilde{\Sigma}^w_t[r,s] = \sum_{j=1}^{J} p_j \mathbb{E}[g(\alpha^{1/2} \tilde{Z}_r + m_{j,r}, Y_j, \bar{a}_r) g(\alpha^{1/2} \tilde{Z}_s + m_{j,s}, Y_j, \bar{a}_s)^\top],$$

where $(\tilde{Z}_r, \tilde{Z}_s) \sim \mathcal{N}\left(0, \begin{bmatrix} \tilde{\Omega}^z_t[r,r] & \tilde{\Omega}^z_t[r,s] \\ \tilde{\Omega}^z_t[s,r] & \tilde{\Omega}^z_t[s,s] \end{bmatrix}\right)$. This proves the inductive claim.

Recalling system (A.10), we can relate this system back to Definition A.12 by considering the change of variables $\check{Z}_t = \alpha^{1/2} \tilde{Z}_t$, $\check{\Omega}_{t+1} = \alpha \tilde{\Omega}^z_{t+1}$, $\check{\Xi}_t = \tilde{\Xi}_t$ and $\check{\Sigma}_t = \tilde{\Sigma}^w_t$. Recall the reparameterization (A.6), where, for any two functions $\phi, \varphi$ given in Lemma A.11, we can write

$$\frac{1}{n} \sum_{i=1}^{n} \phi(h_1, \ldots, h_T)_i = \frac{1}{n} \sum_{i=1}^{n} \phi \left( \alpha^{1/2} z_1 + \sum_{j=1}^{J} e_{B_j} m_{j,1}^\top, \ldots, \alpha^{1/2} z_T + \sum_{j=1}^{J} e_{B_j} m_{j,T}^\top \right)_i,$$

or

$$\frac{1}{d} \sum_{i=1}^{d} \varphi(\tilde{\theta}_1, \ldots, \tilde{\theta}_T) = \frac{1}{d} \sum_{i=1}^{d} \varphi \left( \alpha^{1/2} w_1 + \alpha \sum_{j=1}^{J} \mu_j \ell_{j,1}^\top, \ldots, \alpha^{1/2} w_T + \alpha \sum_{j=1}^{J} \mu_j \ell_{j,T}^\top \right)_i.$$

Notice, if both $\phi_1, \phi_2$ and $\varphi_1, \varphi_2$ are $L$-Lipschitz in their first argument then the following functions are $L\alpha^{1/2}$ Lipschitz with respect to either $z_1, \dots, z_T$ or $w_1, \dots, w_T$, for both $i \in \{1, 2\}$,

$$(z_1, \dots, z_T) \mapsto \phi_i \left( \alpha^{1/2} z_1 + \sum_{j=1}^{J} e_{B_j} m_{j,1}^\top, \dots, \alpha^{1/2} z_T + \sum_{j=1}^{J} e_{B_j} m_{j,T}^\top \right)$$
$$(w_1, \dots, w_T) \mapsto \varphi_i \left( \alpha^{1/2} w_1 + \alpha \sum_{j=1}^{J} \mu_j \ell_{j,1}^\top, \dots, \alpha^{1/2} w_T + \alpha \sum_{j=1}^{J} \mu_j \ell_{j,T}^\top \right).$$

Therefore, by Lemma A.11 and the aforementioned change of variables between system (A.10) and Definition A.12, we have almost surely that

$$\lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} \phi(h_1, \dots, h_T)_i - \lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} \phi \left( \check{Z}_1 + \sum_{j=1}^{J} e_{B_j} m_{j,1}^\top, \dots, \check{Z}_T + \sum_{j=1}^{J} e_{B_j} m_{j,T}^\top \right)_i = 0, \tag{A.12}$$

or

$$\lim_{d\to\infty} \frac{1}{d} \sum_{i=1}^{d} \varphi(\tilde{\theta}_1, \dots, \tilde{\theta}_T)_i - \lim_{d\to\infty} \frac{1}{d} \sum_{i=1}^{d} \varphi \left( \alpha^{1/2} \check{W}_1 + \alpha \sum_{j=1}^{J} \mu_j \ell_{j,1}^\top, \dots, \alpha^{1/2} W_T + \alpha \sum_{j=1}^{J} \mu_j \ell_{j,T}^\top \right)_i = 0, \tag{A.13}$$

where $(\check{Z}_1, \dots, \check{Z}_T) \sim \mathcal{N}(0, \check{\Omega}_T \otimes \mathrm{Id}_n)$ and $(\check{W}_1, \dots, \check{W}_T) \sim \mathcal{N}(0, \check{\Sigma}_T \otimes \mathrm{Id}_d)$.

Each of the limits in the statement of the lemma follow immediately from the following choice of $\phi$ and $\varphi$, using equations (A.12) or (A.13) and using the identicality of the row distribution of $\check{Z}_1, \dots, \check{Z}_T$ and $\check{W}_1, \dots, \check{W}_T$ (recalling that $\varphi(\cdot)_i = \varphi_1(\cdot)_i \varphi_2(\cdot)_i$ and similarly for $\phi$):

$$\begin{aligned}
&\frac{1}{d} \mu_j^\top \tilde{\theta}_t, \quad \varphi(\tilde{\theta}_1, \dots, \tilde{\theta}_T)_i = \mu_{j,i} \tilde{\theta}_{t,i} \\
&\frac{1}{d} \theta_r^\top \theta_s, \quad \varphi(\tilde{\theta}_1, \dots, \tilde{\theta}_T)_i = \left( \eta_0^r \theta_1 - \eta_1 \sum_{p=1}^{r} \eta_0^{r-p} \tilde{\theta}_{p,i} \right) \left( \eta_0^s \theta_1 - \eta_1 \sum_{q=1}^{s} \eta_0^{s-q} \tilde{\theta}_{q,i} \right) \\
&\frac{1}{d} \theta_r^\top \tilde{\theta}_s, \quad \varphi(\tilde{\theta}_1, \dots, \tilde{\theta}_T)_i = \left( \eta_0^r \theta_1 - \eta_1 \sum_{p=1}^{r} \eta_0^{r-p} \tilde{\theta}_{p,i} \right) \tilde{\theta}_s \\
&\frac{1}{n} \hat{h}_r^\top \hat{h}_s, \quad \phi(h_1, \dots, h_T)_i = g(h_{r,i}, y_i, \bar{a}_r) g(h_{s,i}, y_i, \bar{a}_s)
\end{aligned} \tag{A.14}$$

and, for the first three limits, invoking the law of large numbers on row-wise independent laws of $\check{W}_1, \dots, \check{W}_t, \check{Z}_1, \dots, \check{Z}_t$ to relate back to Definition A.12. Note, (A.14), and more generally the statement on $\frac{1}{n} \sum_{i=1}^{n} \bar{\phi}(h_{t,i}, y_i, \bar{a}_t, j) = \sum_{j=1}^{J} \frac{|B_j|}{n} \sum_{i \in B_j} \bar{\phi}(h_t, y, \bar{a}_t, j)_i$ (where $\bar{\phi}$ is applied row-wise) in the statement of this corollary follows by selecting $(\phi_1)_i = \bar{\phi}(\cdot, y_i, \bar{a}_t, j)$ (or $g(\cdot, y_i, \bar{a}_t)$) for each $i \in [n]$ and $\phi_2$ being the all ones vector (or $g(\cdot, y_i, \bar{a}_t)$) and using equation (A.12). Then an application of the continuous mapping theorem for the almost sure limit $\lim_{n\to\infty} |B_j|/n$, the identicality of the row distribution of $\check{Z}_1, \dots, \check{Z}_T$, alongside the strong law of large numbers applied to the block $B_j$ where each $y_i \sim \mathbb{P}_j$ independently proves the claim. ■

We conclude this section by observing that Definition A.12, with the exception of the updates on the frozen state evolution parameters, is equivalent Definition A.7.

### A.8.3 Convergence of Lower-order Recursions Of $m_{j,t}, \ell_{j,t}$ And $\bar{a}_t$

Let $\tilde{m}_{j,t}^\top = \frac{\mu_j^\top \theta_t}{d}$ and $\tilde{\ell}_{j,t}^\top = \frac{e_{B_j}^\top \hat{h}_t}{d}$, consider the following algorithm equivalent to (A.4),

$$\begin{aligned}
h_t &= \sum_{j=1}^{J} e_{B_j} \tilde{m}_{j,t}^\top + Z\theta_t + \eta_1 \sum_{s=1}^{t-1} \eta_0^{(t-1)-s} \hat{h}_s \\
\hat{h}_t &= g(h_t, y, a_t) \\
\tilde{\theta}_t &= \sum_{j=1}^{J} \alpha \mu_j \tilde{\ell}_{j,t}^\top + Z^\top \hat{h}_t - \alpha \theta_t \left( \frac{1}{n} \sum_{i=1}^{n} \nabla_h g(h_{t,i}, y_i, a_t) \right)^\top \\
\theta_{t+1} &= \eta_0 \theta_t - \eta_1 \tilde{\theta}_t \\
a_{t+1} &= \gamma_0 a_t - \gamma_1 \frac{1}{n} \sum_{i=1}^{n} f(h_{t,i}, y_i, a_t).
\end{aligned}$$

We prove the following.

**Lemma A.14.** *Assume that almost surely the following limits exist,* $\lim_{d\to\infty} \tilde{m}_{j,t} = m_{j,t} \in \mathbb{R}^L$, $\lim_{d\to\infty} \tilde{\ell}_{j,t} = \ell_{j,t} \in \mathbb{R}^L$ *and* $\lim_{d\to\infty} a_t = \bar{a}_t$. *If* $\phi_1, \phi_2 : \mathbb{R}^{n\times(L\cdot T)} \times \mathbb{R}^n \times \mathbb{R}^{L'\cdot T} \to \mathbb{R}^n$, $\varphi_1, \varphi_2 : \mathbb{R}^{d\times(L\cdot T)} \times \mathbb{R}^{L'\cdot T} \to \mathbb{R}^n$ *are Lipschitz in their first argument (and third argument for* $\phi_1, \phi_2$*, both uniformly over the second argument* $y \in \mathbb{R}^n$*), then we have that almost surely,*

$$\lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} \phi(h_1, \ldots, h_T, y, a_1, \ldots, a_T)_i - \frac{1}{n} \sum_{i=1}^{n} \phi(h_1^{\text{frozen}}, \ldots, h_T^{\text{frozen}}, y, \bar{a}_1, \ldots, \bar{a}_T)_i = 0$$

*and*

$$\lim_{n\to\infty} \frac{1}{d} \sum_{i=1}^{d} \varphi(\tilde{\theta}_1, \ldots, \tilde{\theta}_T, a_1, \ldots, a_T)_i - \frac{1}{n} \sum_{i=1}^{n} \phi(\tilde{\theta}_1^{\text{frozen}}, \ldots, \tilde{\theta}_T^{\text{frozen}}, \bar{a}_1, \ldots, \bar{a}_T)_i = 0,$$

*where* $\phi(\cdot)_i = \phi_1(\cdot)_i \phi_2(\cdot)_i$ *and* $\varphi(\cdot)_i = \varphi_1(\cdot)_i \varphi_2(\cdot)_i$.

*Proof.* We proceed by induction on $t \in [T]$ to prove the iterates of the original and frozen algorithm have a vanishing difference in two-norm almost surely as $n, d \to \infty$ with $n/d \to \alpha$. Specifically, we prove the following limits hold almost surely,

$$\lim_{n\to\infty} \frac{1}{n} \|h_t - h_t^{\text{frozen}}\|_2^2 = 0, \quad \lim_{n\to\infty} \frac{1}{n} \|\hat{h}_t - \hat{h}_t^{\text{frozen}}\|_2^2 = 0, \tag{A.15}$$

$$\lim_{d\to\infty} \frac{1}{d} \|\tilde{\theta}_t - \tilde{\theta}_t^{\text{frozen}}\|_2^2 = 0, \quad \lim_{d\to\infty} \frac{1}{d} \|\theta_{t+1} - \theta_{t+1}^{\text{frozen}}\|_2^2 = 0. \tag{A.16}$$

**Base Case ($t = 1$):** By initialization, $\theta_1 = \theta_1^{\text{frozen}}$. Evaluating the pre-activations gives,

$$\begin{aligned}
h_1 - h_1^{\text{frozen}} &= \left( \sum_{j=1}^{J} e_{B_j} \tilde{m}_{j,1}^\top + Z\theta_1 \right) - \left( \sum_{j=1}^{J} e_{B_j} m_{j,1}^\top + Z\theta_1^{\text{frozen}} \right) \\
&= \sum_{j=1}^{J} e_{B_j} (\tilde{m}_{j,1} - m_{j,1})^\top.
\end{aligned}$$

Since $\lim_{d\to\infty} \tilde{m}_{j,1} = m_{j,1}$ almost surely, and that $\lim_{n\to\infty} \|e_{B_j}\|_2^2/n = p_j \in [0, 1]$ from Assumption A.4 (1) (with $J$ being $n$-independent), we have $\lim_{n\to\infty} \frac{1}{n} \|h_1 - h_1^{\text{frozen}}\|_2^2 = 0$ almost surely. Thus, as the function $g$ is Lipschitz with respect to input $h_1, a_1$ (for some bounded constant $C > 0$), we have that

$$\frac{1}{n} \|\hat{h}_1 - \hat{h}_1^{\text{frozen}}\|_2^2 = \frac{1}{n} \|g(h_1, y, a_1) - g(h_1^{\text{frozen}}, y, \bar{a}_1)\|_2^2$$

$$\leq C\left(\frac{1}{n}\|h_1 - h_1^{\text{frozen}}\|_2^2 + \|a_1 - \bar{a}_1\|_2^2\right).$$

Therefore, as an immediate consequence of $\lim_{n\to\infty}\frac{1}{n}\|h_1 - h_1^{\text{frozen}}\|_2^2 = 0$ and $\lim_{d\to\infty} a_1 = \bar{a}_1$, we have $\lim_{n\to\infty}\frac{1}{n}\|\hat{h}_1 - \hat{h}_1^{\text{frozen}}\|_2^2 = 0$ almost surely.

**Inductive Step:** Assume equations (A.15) and (A.16) hold for all $s \leq t$. We first consider the update for $\tilde{\theta}_t$:

$$\begin{aligned}\tilde{\theta}_t - \tilde{\theta}_t^{\text{frozen}} = \alpha\sum_{j=1}^{J}\mu_j(\tilde{\ell}_{j,t} - \ell_{j,t})^\top + Z^\top(\hat{h}_t - \hat{h}_t^{\text{frozen}}) \\ - \alpha\left(\langle\nabla_h g(h_t, y, a_t)\rangle\,\theta_t - \left\langle\nabla_h g(h_t^{\text{frozen}}, y, \bar{a}_t)\right\rangle\theta_t^{\text{frozen}}\right),\end{aligned}$$

where $\langle\cdot\rangle$ is a shorthand for the empirical average over $n$ coordinates. We bound the two-norm of each of these three terms separately:

1. **Signal Term:** Since $\lim_{d\to\infty}\tilde{\ell}_{j,t} = \ell_{j,t}$ and $\frac{1}{d}\|\mu_j\|_2^2 = O(1)$ by Assumption A.4 (1), we immediately have $\lim_{d\to\infty}\frac{1}{d}\|\alpha\sum_{j=1}^{J}\mu_j(\tilde{\ell}_{j,t} - \ell_{j,t})^\top\|_2^2 = 0$ almost surely.
2. **Noise Term:** Since $Z \in \mathbb{R}^{n\times d}$ is a matrix with independent $\mathcal{N}(0, 1/d)$ entries, the operator norm $\|Z\|_{\text{op}}$ is almost surely bounded [2] as $n, d \to \infty$. Thus, almost surely,

$$\lim_{d\to\infty}\frac{1}{d}\|Z^\top(\hat{h}_t - \hat{h}_t^{\text{frozen}})\|_2^2 \leq \lim_{d\to\infty}\frac{\|Z\|_{\text{op}}^2}{d}\|\hat{h}_t - \hat{h}_t^{\text{frozen}}\|_2^2 = \lim_{d\to\infty}\alpha\|Z\|_{\text{op}}^2\frac{1}{n}\|\hat{h}_t - \hat{h}_t^{\text{frozen}}\|_2^2 = 0,$$

   where the final equality applied the inductive hypothesis on $\hat{h}_t$.
3. **Onsager Term:** Adding and subtracting $\langle\nabla_h g(h_t^{\text{frozen}}, y, \bar{a}_t)\rangle\theta_t$, we have by the triangle inequality and Cauchy-Schwarz,

$$\begin{aligned}&\frac{\alpha^2}{d}\left\|\langle\nabla_h g(h_t, y, a_t)\rangle\theta_t - \langle\nabla_h g(h_t^{\text{frozen}}, y, \bar{a}_t)\rangle\theta_t^{\text{frozen}}\right\|_2^2 \\ &\leq 2\alpha^2\left\|\langle\nabla_h g(h_t, y, a_t)\rangle - \langle\nabla_h g(h_t^{\text{frozen}}, y, \bar{a}_t)\rangle\right\|_{\text{op}}^2\frac{\|\theta_t\|_2^2}{d} \\ &\quad + 2\alpha^2\left\|\langle\nabla_h g(h_t^{\text{frozen}}, y, \bar{a}_t)\rangle\right\|_{\text{op}}^2\frac{\|\theta_t - \theta_t^{\text{frozen}}\|_2^2}{d}.\end{aligned}$$

   By Assumption A.4 (3), $\nabla_h g$ is Lipschitz in both $h$ and $a$. Thus, the operator norm difference $\left\|\langle\nabla_h g(h_t, y, a_t)\rangle - \langle\nabla_h g(h_t^{\text{frozen}}, y, \bar{a}_t)\rangle\right\|_{\text{op}}^2$ vanishes almost surely by the inductive hypothesis on $h_t$ and the almost sure convergence of $\lim_{n\to\infty} a_t = \bar{a}_t$. Since $\frac{1}{d}\|\theta_t\|_2^2 = O(1)$ almost surely (by the inductive hypothesis and Corollary A.13), the first term vanishes. The second term vanishes because $\frac{1}{d}\|\theta_t - \theta_t^{\text{frozen}}\|_2^2 \to 0$ by the inductive hypothesis, and the empirical average of $\nabla_h g$ is bounded.

Therefore, $\frac{1}{d}\|\tilde{\theta}_t - \tilde{\theta}_t^{\text{frozen}}\|_2^2 \to 0$.

By the linear update $\theta_{t+1} = \eta_0\theta_t - \eta_1\tilde{\theta}_t$, we trivially obtain $\frac{1}{d}\|\theta_{t+1} - \theta_{t+1}^{\text{frozen}}\|_2^2 \to 0$.

Similarly, evaluating the update for $h_{t+1}$,

$$\begin{aligned}h_{t+1} - h_{t+1}^{\text{frozen}} = \sum_{j=1}^{J} e_{B_j}(\tilde{m}_{j,t+1} - m_{j,t+1})^\top + Z(\theta_{t+1} - \theta_{t+1}^{\text{frozen}}) \\ + \eta_1\sum_{s=1}^{t}\eta_0^{t-s}(\hat{h}_s - \hat{h}_s^{\text{frozen}}).\end{aligned}$$

Applying the assumed limits for $\tilde{m}_{j,t+1}$, the bounded operator norm of $Z$, and the inductive hypothesis on the past $\hat{h}_s$ iterates, we find $\lim_{n\to\infty}\frac{1}{n}\|h_{t+1} - h_{t+1}^{\text{frozen}}\|_2^2 = 0$ almost surely.

**Convergence of Test Functions:** We now prove that almost sure convergence of the iterates implies the almost sure convergence of the test functions. Consider $\phi(\cdot)_i = \phi_1(\cdot)_i\phi_2(\cdot)_i$ where $\phi_1, \phi_2 : \mathbb{R}^{n\times(L\cdot T)} \times \mathbb{R}^n \times \mathbb{R}^{L'\cdot T} \to \mathbb{R}^n$ are Lipschitz functions satisfying the statement of

the Lemma. For notational simplicity, we abbreviate $\phi(h_1, \ldots, h_T, y, a_1, \ldots, a_T)$ as $\phi(h)$ and . $\phi(h_1^{\text{frozen}}, \ldots, h_T^{\text{frozen}}, y, \bar{a}_1, \ldots, \bar{a}_T)$ as $\phi(h^{\text{frozen}})$.

We then have that,

$$\begin{aligned}\frac{1}{n}\left|\sum_{i=1}^{n}(\phi(h)_i - \phi(h^{\text{frozen}})_i)\right| &= \frac{1}{n}\left|\phi_1(h)^\top\phi_2(h) - \phi_1(h^{\text{frozen}})^\top\phi_2(h^{\text{frozen}})\right| \\ &\leq \frac{1}{n}\left(\left|\phi_1(h)^\top\phi_2(h) - \phi_1(h)^\top\phi_2(h^{\text{frozen}})\right| + \left|\phi_1(h)^\top\phi_2(h^{\text{frozen}}) - \phi_1(h^{\text{frozen}})^\top\phi_2(h^{\text{frozen}})\right|\right)\end{aligned} \tag{A.17}$$

By the Cauchy-Schwarz inequality, the first term is bounded by,

$$\left(\frac{1}{n}\|\phi_1(h)\|_2^2\right)^{1/2}\left(\frac{1}{n}\|\phi_2(h) - \phi_2(h^{\text{frozen}})\|_2^2\right)^{1/2}.$$

Because $\phi_1$ is Lipschitz (for some constant $C > 0$) in $h, a$ we have that

$$\begin{aligned}\frac{1}{n}\|\phi_1(h)\|_2^2 &= \frac{1}{n}\|\phi_1(h) - \phi_1(h^{\text{frozen}}) + \phi_1(h^{\text{frozen}})\|_2^2 \\ &\leq \frac{2}{n}\|\phi_1(h) - \phi_1(h^{\text{frozen}})\|_2^2 + \frac{2}{n}\|\phi_1(h^{\text{frozen}})\|_2^2 \\ &\leq 2C\left(\frac{1}{n}\|h - h^{\text{frozen}}\|_F^2 + \sum_{t=1}^{T}\|a_t - \bar{a}_t\|_2^2\right) + \frac{2}{n}\|\phi_1(h^{\text{frozen}})\|_2^2\end{aligned}$$

and using the inductive hypothesis, the assumed almost sure convergence of $a_t$ and Corollary A.13 on $\frac{2}{n}\|\phi_1(h^{\text{frozen}})\|_2^2$ implies that it is almost surely bounded by a constant. Moreover, using that $\phi_2$ is Lipschitz (for some bounded Lipschitz constant $C > 0$) in both argument $h_1, \ldots, h_T$ and $a_1, \ldots, a_T$, we have that

$$\begin{aligned}\frac{1}{n}\|\phi_2(h) - \phi_2(h^{\text{frozen}})\|_2^2 &= \frac{1}{n}\|\phi_2(h, y, a_1, \ldots, a_T) - \phi_2(h^{\text{frozen}}, y, \bar{a}_1, \ldots, \bar{a}_T)\|_2^2 \\ &\leq C\left(\frac{1}{n}\|h - h^{\text{frozen}}\|_F^2 + \sum_{t=1}^{T}\|a_t - \bar{a}_t\|_2^2\right).\end{aligned}$$

As we have previously shown that $(\frac{1}{n}\|h - h^{\text{frozen}}\|_F^2)$ converges to 0 almost surely, then the first term of (A.17) converges to zero almost surely. Applying the symmetric argument to the second term in (A.17) gives the desired almost sure limit for $\phi$.

An identical argument gives the exact same result for $\varphi(\tilde{\theta}_1, \ldots, \tilde{\theta}_T, a_1, \ldots, a_T)$. This completes the proof. ■

### A.8.4 The Almost Sure Limits $m_{j,t}$, $\ell_{j,t}$ And $\bar{a}_t$

To avoid confusion with the function $\phi$ used in the appendix, we denote the test function in Lemma 3.6 as $\bar{\phi}$.

Combining Corollary A.13 and Lemma A.14, the proof of Lemma 3.6 follows after observing that Assumption A.9 can allow $\bar{\phi}$ to be utilized in both Corollary A.13 and Lemma A.14 with the identification of $\phi_1 = \bar{\phi}$ (applied row-wise) and $\phi_2$ being the all ones vector. As Definition A.10 and Definition A.7 are identical up to the frozen state evolution parameters, and have proven we can replace $\tilde{m}_{j,t}, \tilde{\ell}_{j,t}$ and $a_t$ by the frozen sequence of their respective almost sure limits, we are done once we identify the recursions for the almost sure limits of $\tilde{m}_{j,t}, \tilde{\ell}_{j,t}$ and $a_t$.

For the case of $\tilde{m}_{j,t}$, we have that,

$$\tilde{m}_{j,t+1} = \frac{\theta_{t+1}^\top\mu_j}{d} = \eta_0\tilde{m}_{j,t} - \eta_1\frac{\tilde{\theta}_t^\top\mu_j}{d},$$

recursively plugging in the almost sure limit of $\frac{1}{d}\mu_j^\top\tilde{\theta}_t$ from Corollary A.13 and that $\mathbb{E}[\check{W}_t] = 0$, we

have that for each $s \in [T]$,

$$\lim_{d\to\infty} \frac{\tilde{\theta}_s^\top \mu_j}{d} = \lim_{d\to\infty} \alpha \sum_{k=1}^{J} \frac{\mu_k^\top \mu_j}{d} \ell_{k,s} = \alpha \sum_{k=1}^{J} \chi_{j,k} \ell_{k,s},$$

by the assumed limit of $\frac{\mu_j^\top \mu_k}{d} = \frac{\mu_k^\top \mu_j}{d}$ in Assumption A.4 (1). Thus, we have that almost surely that $\lim_{d\to\infty} \tilde{m}_{j,t} = m_{j,t}$ defined by the recursion

$$m_{j,t+1} = \eta_0 m_{j,t} - \eta_1 \sum_{k=1}^{J} \alpha \chi_{j,k} \ell_{k,t}.$$

Similarly, using that $\tilde{\ell}_{j,t} = \frac{e_{B_j}^\top \hat{h}_t}{d} = \frac{|B_j|}{n} \frac{1}{|B_j|} \sum_{i\in B_j} \hat{h}_{t,i} = \frac{|B_j|}{n} \frac{1}{|B_j|} \sum_{i\in B_j} g(h_{t,i}, y_i, a_t)$, we have the almost sure limit,

$$\ell_{j,t} = p_j \mathbb{E}[g(Z_t + m_{j,t}, Y_j, \bar{a}_t)],$$

where $Z_t \sim \mathcal{N}(0, \check{\Omega}_t[t,t])$.

As a final note, using that $a_{t+1} = \gamma_0 a_t - \gamma_1 \sum_{j=1}^{J} \frac{|B_j|}{n} \frac{1}{|B_j|} \sum_{i\in B_j} f(h_{t,i}, y_i, a_t)$, and thus by Corollary A.13 with $\phi_1 = f$ and $\phi_2$ being the all ones vector (recalling from Assumption A.4 (3) that $f$ is Lipschitz in its first argument), we have the following almost sure limit,

$$\bar{a}_{t+1} = \gamma_0 \bar{a}_t - \gamma_1 \sum_{j=1}^{J} p_j \mathbb{E}[f(G^t + m_{j,t}, Y_j, \bar{a}_t)],$$

noting that $\bar{a}_1 = a_1$ and we assumed in Assumption A.4 (2) that $\|a_1\|_2$ is bounded and that $f$ is bounded from Assumption A.4 (3), meaning that each limit has $\|\bar{a}_t\|_2$ bounded for $t \in [T]$.

Combining these almost sure limits with the state evolution recursion from Definition A.12, we obtain the state evolution in Definition A.7 and thus conclude Lemma 3.6.

### A.9 The Proof Of Theorem 3.8

*Proof Of Theorem 3.8.* Under Assumption A.4, we have that Assumption A.9 holds for the choice of $\phi(h, y, a) = \mathcal{L}(\mathcal{M}_a(h), y)$. Invoking Lemma 3.6 gives the following almost surely,

$$\lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} \mathcal{L}(\mathcal{M}_{a_t}(h_{t,i}), y_i) = \sum_{j=1}^{J} p_j \mathbb{E}[\mathcal{L}(\mathcal{M}_{\bar{a}_t}(m_{j,t}+G^t), Y_j)], \quad G^t \sim \mathcal{N}(0, \Omega_t[t,t]), \quad Y_j \sim \mathbb{P}_j,$$

it is then immediate that $\lim_{n\to\infty} \frac{1}{n} \sum_{i=1}^{n} \mathcal{L}(\mathcal{M}_{a_t}(h_{t,i}), y_i) = \mathbf{test}(m_{1,t}, \ldots, m_{J,t}, \Omega_t[t,t], \bar{a}_t)$ from Definition 3.5, invoking Proposition A.6 concludes the proof. ■

### A.10 The State Evolution Of Pure DD

Using Definition A.7, we can specify the state evolution for this algorithm. We abbreviate $\nabla_h \Psi_{r,j} = \nabla_h \Psi(m_{j,r} + G^r, Y_j, \bar{a}_r)$ where $G^1, \ldots, G^t \sim \mathcal{N}(0, \Omega_t)$, $Y_j \sim \mathbb{P}_j$ and define $V_{r,s} = \alpha \Sigma_t[r,s] + \alpha^2 \sum_{j,k=1}^{J} \chi_{j,k} p_j p_k \mathbb{E}[\nabla_h \Psi_{r,j}] \mathbb{E}[\nabla_h \Psi_{s,k}]^\top$. With identical initializations from Definition A.7 and indices $r, s \in [t]$, we recursively define,

$$\begin{aligned}
\Sigma_t[r,s] &= \sum_{j=1}^{J} p_j \mathbb{E}[\nabla_h \Psi_{r,j} \nabla_h \Psi_{s,j}^\top] \\
\Xi_t[r+1,s] &= \Xi_{t-1}[r,s] - \eta V_{r,s} \\
\Omega_{t+1}[r+1,s+1] &= \Omega_t[r,s] - \eta(\Xi_t[r,s] + \Xi_t[s,r]^\top) + \eta^2 V_{r,s} \\
m_{j,t+1} &= m_{j,t} - \eta\alpha \sum_{k=1}^{J} \chi_{j,k} p_k \mathbb{E}[\nabla_h \Psi_{t,k}] \\
\bar{a}_{t+1} &= \bar{a}_t - \eta \sum_{j=1}^{J} p_j \mathbb{E}[\nabla_a \Psi(m_{j,t} + G^t, Y_j, \bar{a}_t)].
\end{aligned} \tag{A.18}$$

### A.11 Taylor Expansion Of (Pure) DD

**Lemma A.15.** *Let $\mu \in \mathbb{R}^L$, $\Sigma \in \mathbb{R}^{L\times L}$ and $c \in \mathbb{R}$. Then for $X \sim \mathcal{N}(c\mu, \Sigma)$, the following identities hold,*

$$\begin{aligned}
\nabla_\mu \mathbb{E}[f(X)] &= c\mathbb{E}[\nabla_x f(x)|_{x=X}] \\
\nabla_\Sigma \mathbb{E}[f(X)] &= \frac{1}{2}\mathbb{E}[\nabla_x^2 f(x)|_{x=X}] \\
\nabla_\mu^2 \mathbb{E}[f(X)] &= c^2\mathbb{E}[\nabla_x^2 f(x)|_{x=X}] \\
\nabla_\Sigma^2 \mathbb{E}[f(X)] &= \frac{1}{4}\mathbb{E}[\nabla_x^4 f(x)|_{x=X}] \\
\nabla_\mu \nabla_\Sigma \mathbb{E}[f(X)] &= \frac{c}{2}\mathbb{E}[\nabla_x^3 f(x)|_{x=X}].
\end{aligned}$$

*Proof.* See [33, Equation (2)]. ■

**Lemma A.16.** *Suppose Assumption A.4 holds. Let $\mathcal{T}$ be the space of all $\theta = (\bar{a}, m_1, \ldots, m_J, \omega)$ where $\|\bar{a}\|_2 \le C$, $\|m_j\|_2 \le C$, and $\|\omega^{-1}\|_{\text{op}} \le C$, $\|\omega\|_{\text{op}} \le C$ for some constant $C > 0$. If $G \sim \mathcal{N}(0, \text{Id}_L)$ and $Y_j \sim \mathbb{P}_j$, then the expectations of the derivatives of $\Psi(m_j + \omega^{1/2}G, Y_j, \bar{a})$ with respect to $\bar{a}$, $m_j$, and $\omega$ exist, and the expectation operator can be interchanged with the derivative operator.*

*Proof.* To interchange the expectation and derivative, it suffices to show that the partial derivatives of $\Psi(m_j + \omega^{1/2}G, Y_j, \bar{a})$ with respect to $\bar{a}$, $m_j$, and $\omega$ are uniformly bounded by an integrable function for all $\theta \in \mathcal{T}$.

Let $x = m_j + \omega^{1/2}G$. By Assumption A.4 (4) (a), $\Psi$ is Lipschitz with respect to $h$ and $a$. Therefore, the gradients $\nabla_h \Psi$ and $\nabla_a \Psi$ exist almost everywhere and are uniformly bounded by constant $L > 0$. Thus, each of the following derivatives satisfy

$$\begin{aligned}
\|\nabla_{\bar{a}} \Psi\|_2 &= \|\nabla_a \Psi\|_2 \le L \\
\|\nabla_{m_j} \Psi\|_2 &= \|\nabla_h \Psi\|_2 \le L \\
\|\nabla_\omega \Psi\|_{\text{F}} &\le \|\nabla_h \Psi\|_2 \|\nabla_\omega(\omega^{1/2}G)\|_{\text{op}} \le L \cdot C' \|G\|_2,
\end{aligned}$$

where $C' > 0$ is a constant bounding the derivative of the matrix square root uniformly over the domain $\mathcal{T}$; such a constant exists by the assumption that $\max(\|\omega^{-1}\|_{\text{op}}, \|\omega\|_{\text{op}}) \le C$.

We construct the dominating function $H(G) = \max(L, L \cdot C' \|G\|_2)$. Because $G \sim \mathcal{N}(0, \text{Id}_L)$, the second moment $\|G\|_2$ is bounded. Thus, $H(G)$ is integrable and bounds the parameter derivatives independently of $\theta \in \mathcal{T}$, allowing the interchange of expectation and differentiation. ■

**Lemma A.17.** *Let Assumption A.4 (4) hold for a bounded constant $C > 0$, the operator norm of the block Hessian matrix $\nabla^2_{a,m,\text{vec}(\omega)} \mathbb{E}[\Psi(m + \omega^{1/2}G, Y_j, a)]$ is bounded by a constant $C' > 0$ depending only on $C$.*

*Proof.* We can expand the matrix $\nabla^2_{a,m,\text{vec}(\omega)} \mathbb{E}[\Psi]$ into the following $3 \times 3$ block matrix,

$$\begin{bmatrix}
\nabla_a^2 \mathbb{E}[\Psi(m+\omega^{1/2}G, y, a)] & \nabla_a \nabla_m \mathbb{E}[\Psi(m+\omega^{1/2}G, y, a)] & \nabla_a \nabla_{\text{vec}(\omega)} \mathbb{E}[\Psi(m+\omega^{1/2}G, y, a)] \\
\nabla_m \nabla_a \mathbb{E}[\Psi(m+\omega^{1/2}G, y, a)] & \nabla_m^2 \mathbb{E}[\Psi(m+\omega^{1/2}G, y, a)] & \nabla_m \nabla_{\text{vec}(\omega)} \mathbb{E}[\Psi(m+\omega^{1/2}G, y, a)] \\
\nabla_{\text{vec}(\omega)} \nabla_a \mathbb{E}[\Psi(m+\omega^{1/2}G, y, a)] & \nabla_{\text{vec}(\omega)} \nabla_m \mathbb{E}[\Psi(m+\omega^{1/2}G, y, a)] & \nabla^2_{\text{vec}(\omega)} \mathbb{E}[\Psi(m+\omega^{1/2}G, y, a)]
\end{bmatrix}.$$

Applying the derivative identities from Lemma A.15, each block in the above matrix becomes one of $\mathbb{E}[\nabla_a^2 \Psi]$, $\mathbb{E}[\nabla_h \nabla_a \Psi]$, $\mathbb{E}[\nabla_h^2 \Psi]$, $\frac{1}{2}\mathbb{E}[\nabla_h^2 \nabla_a \Psi]$, $\frac{1}{2}\mathbb{E}[\nabla_h^3 \Psi]$, and $\frac{1}{4}\mathbb{E}[\nabla_h^4 \Psi]$, where we have suppressed the input for notational simplicity.

Assumption A.4 (4) guarantees the operator or Frobenius norm of each of these derivatives is bounded by $C$. Because the operator norm of any block matrix is bounded by the summed operator norms of its block matrices and the Frobenius norm strictly upper-bounds the operator norm, the operator norm of the entire block matrix is bounded by a finite linear combination of constant $C$, concluding the proof. ■

**Theorem A.18.** *Let $T \in \mathbb{N}$ finite. Consider algorithm (3.2); recall the state evolution variables (specifically $\Omega_t$, $m_{j,t}$ and $\bar{a}_t$) from Definition A.7 and $\mathbf{test}_t = \mathbf{test}(m_{1,t}, \dots, m_{J,t}, \Omega_t[t,t], \bar{a}_t)$ from Definition 3.5.*

*Let $\epsilon = \max(|1-\eta_0|, |1-\gamma_0|, \eta_1, \gamma_1)$. If Assumption A.4 and Assumption A.9 (with respect to $g, f$) hold for sufficiently small $\epsilon > 0$, then in the limit $\epsilon \to 0$, $\mathbf{test}_t$ satisfies the following expansion with $G^t \sim \mathcal{N}(0, \Omega_t[t,t])$,*

$$\begin{aligned}
\mathbf{test}_{t+1} = \mathbf{test}_t
&- (1-\gamma_0) \sum_{j=1}^{J} p_j \Big\langle \mathbb{E}[\nabla_a \Psi(m_{j,t} + G^t, Y_j, \bar{a}_t)], \bar{a}_t \Big\rangle && \text{(A.19)} \\
&- \gamma_1 \sum_{j,j'=1}^{J} p_j p_{j'} \Big\langle \mathbb{E}[\nabla_a \Psi(m_{j,t} + G^t, Y_j, \bar{a}_t)], \mathbb{E}[f(m_{j',t} + G^t, Y_{j'}, \bar{a}_t)] \Big\rangle && \text{(A.20)} \\
&- (1-\eta_0) \sum_{j=1}^{J} p_j \Big\langle \mathbb{E}[\nabla_h \Psi(m_{j,t} + G^t, Y_j, \bar{a}_t)], m_{j,t} \Big\rangle && \text{(A.21)} \\
&- \eta_1 \alpha \sum_{j,j'=1}^{J} p_j \chi_{j,j'} p_{j'} \Big\langle \mathbb{E}[\nabla_h \Psi(m_{j,t} + G^t, Y_j, \bar{a}_t)], \mathbb{E}[g(m_{j',t} + G^t, Y_{j'}, \bar{a}_t)] \Big\rangle && \text{(A.22)} \\
&- \eta_0(1-\eta_0) \sum_{j=1}^{J} p_j \Big\langle \mathbb{E}[\nabla_h^2 \Psi(m_{j,t} + G^t, Y_j, \bar{a}_t)], \Omega_t[t,t] \Big\rangle && \text{(A.23)} \\
&- \frac{1}{2} \eta_0 \eta_1 \sum_{j=1}^{J} p_j \Big\langle \mathbb{E}[\nabla_h^2 \Psi(m_{j,t} + G^t, Y_j, \bar{a}_t)], \Xi_t[t,t] + \Xi_t[t,t]^\top \Big\rangle && \text{(A.24)} \\
&+ \delta_t,
\end{aligned}$$

*where $\sup_{t \in [T]} |\delta_t| \leq C\epsilon^2$ as $\epsilon \to 0$.*

*Remark* A.19 (Physical Interpretation Of The First Order Expansion). The generalization error expansion from Theorem A.18 decomposes the local behavior of DD optimization into distinct physical terms. Connecting to works in spin glasses, one can interpret the order parameters $m_{j,t}$ and $\Omega_t[t,t]$ as the magnetization (with respect to external field vectors $\mu_1, \dots, \mu_J$) and self-overlap of the parameter $\theta_t$ respectively. In addition, each of terms (A.19)-(A.24) has the following physical interpretation:

- Equations (A.19) and (A.21) correspond to weight decay due to the introduction of a non-zero damping factor of $1-\gamma_0$ and $1-\eta_0$ respectively.
- Equation (A.20) enforces the representation of the model $\mathcal{M}_a$ to align the low-dimensional signal $a$ with the loss landscape through the use of function $f$.
- Equation (A.22) drives the signal acquisition of each of the vectors $\mu_1, \dots, \mu_J$ by $\theta$ using the function $g$. Notice, the presence of $\chi_{j,j'}$ and $\alpha$ incorporates the overlap structure and data aspect ratio natively into the path of the optimization trajectory.
- Equations (A.23) and (A.24) govern the variance inflation of the relative noise in the directions perpendicular to $\mu_1, \dots, \mu_J$. The first term represents the effect of damping. The second term, through unrolling the definition of $\Xi_t[t,t]$, can be seen as cumulative correlation of the current gradient direction with the direction of past gradient evaluations. Each of these effects is compared to the relative scale of the expected Hessian of the post-activations corresponding to the iterate $\theta_t$, eliciting the intuition that flat regions of the loss landscape suffer from little inflation while steep regions are subject to strong inflation.

*Proof Of Theorem A.18.* By Definition 3.5, the value of $\mathbf{test}_{t+1}$ can be written as,

$$\mathbf{test}_{t+1} = \sum_{j=1}^{J} p_j \mathbb{E}[\mathcal{L}(\mathcal{M}_{\bar{a}_{t+1}}(m_{j,t+1} + (\Omega_{t+1}[t+1,t+1])^{1/2} G), Y_j)],$$

where $G \sim \mathcal{N}(0, \mathrm{Id}_L)$ and $Y_j \sim \mathbb{P}_j$. For convenience, we abbreviate $\Omega_s[s,s] = \omega_s$ for all $s \in [t+1]$.

Using the recursive definitions of $m_{j,t+1}$, $\bar{a}_{t+1}$ from Definition A.7 and $\omega_{t+1}$ from Remark A.8, we have that

$$\begin{aligned}\mathbf{test}_{t+1} &= \sum_{j=1}^{J} p_j \mathbb{E}[\mathcal{L}(\mathcal{M}_{\bar{a}_t+\Delta\bar{a}}((m_{j,t}+\Delta m_j)+(\omega_t+\Delta\omega)^{1/2}G), Y_j)] \\ &= \sum_{j=1}^{J} p_j \mathbb{E}[\Psi((m_{j,t}+\Delta m_j)+(\omega_t+\Delta\omega)^{1/2}G, Y_j, \bar{a}_t+\Delta\bar{a})],\end{aligned}$$

where we define (recalling $\ell_{k,t}$ from Definition A.7)

$$\begin{aligned}\Delta\bar{a} &= -(1-\gamma_0)\bar{a}_t - \gamma_1 A, \\ \Delta m_j &= -(1-\eta_0)m_{j,t} - \eta_1 B_j, \\ \Delta\omega &= -(1-\eta_0^2)\omega_t - \eta_0\eta_1(\Xi_t[t,t]+\Xi_t[t,t]^\top) + \eta_1^2\left(\alpha\Sigma_t[t,t] + \alpha^2\sum_{k,k'=1}^{J}\chi_{k,k'}\ell_{k,t}\ell_{k',t}^\top\right) \\ A &= \sum_{j=1}^{J} p_j \mathbb{E}[f(m_{j,t}+\omega_t^{1/2}G, Y_j, \bar{a}_t)], \\ B_j &= \alpha\sum_{k=1}^{J}\chi_{j,k}p_k\mathbb{E}[g(m_{k,t}+\omega_t^{1/2}G, Y_k, \bar{a}_t)] \\ \ell_{k,t} &= p_k\mathbb{E}[g(m_{k,t}+\omega_t^{1/2}G, Y_k, \bar{a}_t)],\end{aligned}$$

with $(G^1,\dots,G^t)\sim\mathcal{N}(0,\Omega_t)$ implicit in the definition of $\Sigma_t$ and $\Xi_t$.

By Assumption A.4 (3), the functions $f, g$ are bounded, therefore each of $A, (B_j)_{j\in[J]}, (\ell_{j,t})_{j\in[J]}$ are element-wise bounded which further implies that $\|A\|_2, \|B_j\|_2, \|\ell_{j,t}\|_2$ are bounded (dependent on dimension $L$) for all $j\in[J]$.

Moreover, by Assumption A.9 the operator norm of the matrices $\omega_t = \Omega_t[t,t] = \Omega_T[t,t]$ and $\Sigma_t[t,t] = \Sigma_T[t,t]$ have bounded operator norm independent of $t$, and therefore the values of $\|\omega_t\|_F$, $\|\Sigma_t[t,t]\|_F$ are bounded independent of $t\in[T]$. Invoking Lemma 3.6, we have that $\lim_{d\to\infty}\frac{\theta_t^\top\tilde{\theta}_t}{d} = \Xi_t[t,t]$, then using that

$$\left\|\frac{\theta_t^\top\tilde{\theta}_t}{d}\right\|_F \le \left\|\frac{\theta_t}{\sqrt{d}}\right\|_F\left\|\frac{\tilde{\theta}_t}{\sqrt{d}}\right\|_F = \sqrt{\mathrm{Tr}\left(\frac{\theta_t^\top\theta_t}{d}\right)}\sqrt{\mathrm{Tr}\left(\frac{\tilde{\theta}_t^\top\tilde{\theta}_t}{d}\right)}$$

and the limiting values of $\lim_{d\to\infty}\frac{\theta_t^\top\theta_t}{d} = \omega_t$ and $\lim_{d\to\infty}\frac{\tilde{\theta}_t^\top\tilde{\theta}_t}{d} = \alpha\Sigma_t[t,t]$, we have that $\|\Xi_t[t,t]\|_F$ is also bounded independent of $t$. Therefore, each term of $\Delta\bar{a}$, $\Delta m_j$ (for each $j\in[J]$) and $\Delta\omega$ is bounded by $C\epsilon$ (for some constant $C>0$ when $\epsilon$ is sufficiently small) uniformly over $t\in[T]$.

Now, define

$$\mathcal{G}(\bar{a}, m_1,\dots,m_J,\omega) = \sum_{j=1}^{J} p_j\mathbb{E}[\Psi(m_j+\omega^{1/2}G, Y_j, \bar{a})].$$

By Taylor's theorem, there exists a point $\xi = (\xi_{\bar{a}}, \xi_{(m_j)_{j\in[J]}}, \xi_\omega)$ on the line connecting $(\bar{a}_t, (m_{j,t})_{j\in[J]}, \omega_t)$ to $(\bar{a}_{t+1}, (m_{j,t+1})_{j\in[J]}, \omega_{t+1})$ such that,

$$\begin{aligned}\mathcal{G}(\bar{a}_{t+1}, m_{1,t+1},\dots,m_{J,t+1},\omega_{t+1}) &= \mathcal{G}(\bar{a}_t, m_{1,t},\dots,m_{J,t},\omega_t) \\ &+ \nabla\mathcal{G}(\bar{a}_t, m_{1,t},\dots,m_{J,t},\mathrm{vec}(\omega_t))^\top\Big(\Delta\bar{a},\Delta m_1,\dots,\Delta m_J,\mathrm{vec}(\Delta\omega)\Big) \\ &+ \frac{1}{2}\Big(\Delta\bar{a},\Delta m_1,\dots,\Delta m_J,\mathrm{vec}(\Delta\omega)\Big)^\top\nabla^2\mathcal{G}(\xi) \\ &\times\Big(\Delta\bar{a},\Delta m_1,\dots,\Delta m_J,\mathrm{vec}(\Delta\omega)\Big),\end{aligned} \tag{A.25}$$

where both the gradient and Hessian are with respect to $(\bar{a}, m_1, \ldots, m_J, \mathrm{vec}(\omega))$.

Notice the matrix $\nabla^2 \mathcal{G}(\xi)$ is a finite linear combination of the Hessian matrices from Lemma A.17. Recall from Definition A.7 that each state evolution parameter is a derived a recursion of expectations over the functions $g, f$. Using the boundedness of these functions and the boundedness of the initialization $\|a_1\|_2 = \|\bar{a}_1\|_2$, we can therefore conclude that each of $\|\bar{a}_t\|_2$, $\|\bar{a}_{t+1}\|_2$, $\max_{j\in[J]} \|m_{j,t}\|_2$, $\max_{j\in[J]} \|m_{j,t+1}\|_2$, $\|\omega_t\|_{\mathrm{op}}$, $\|\omega_{t+1}\|_{\mathrm{op}}$ are bounded. Thus, the corresponding values for $\xi$ are also bounded due to being a linear interpolation of the above iterates.

Then, applying the triangle equality on the operator norm of $\nabla^2 \mathcal{G}(\xi)$ to each $j$-term and using Assumption A.4 (4) (b) to invoke Lemma A.17, we have that $\|\nabla^2 \mathcal{G}(\xi)\|_{\mathrm{op}} \le C$ for some constant $C > 0$. Moreover, as we previously concluded that $(\Delta\bar{a}, \Delta m_1, \ldots, \Delta m_J, \Delta\omega)$ is bounded by $C\epsilon$ for sufficiently small $\epsilon$ uniformly over $t$, each second order term in (A.25) is of order $\epsilon^2$ uniformly over $t \in [T]$.

It remains to compute the first order Taylor expansion. Using Lemma A.16 to interchange expectation and derivative, we can calculate,

$$\nabla_{\bar{a}} \mathcal{G}(\bar{a}, m_1, \ldots, m_J, \omega) = \sum_{j=1}^{J} p_j \mathbb{E}\Big[\nabla_{\bar{a}} \Psi(m_j + \omega^{1/2} G, Y_j, \bar{a})\Big], \tag{A.26}$$

$$\nabla_{m_j} \mathcal{G}(\bar{a}, m_1, \ldots, m_J, \omega) = p_j \mathbb{E}\Big[\nabla_h \Psi(m_j + \omega^{1/2} G, Y_j, \bar{a})\Big], \tag{A.27}$$

$$\nabla_{\omega} \mathcal{G}(\bar{a}, m_1, \ldots, m_J, \omega) = \frac{1}{2} \sum_{j=1}^{J} p_j \mathbb{E}\Big[\nabla_h^2 \Psi(m_j + \omega^{1/2} G, Y_j, \bar{a})\Big], \tag{A.28}$$

where the latter two derivatives follow from Lemma A.15. Using (A.26), the first order contribution from $\Delta\bar{a}$ gives,

$$\begin{aligned}\langle \nabla_{\bar{a}} \mathcal{G}, \Delta\bar{a} \rangle = &-(1-\gamma_0) \sum_{j=1}^{J} p_j \Big\langle \mathbb{E}[\nabla_a \Psi(m_{j,t} + G^t, Y_j, \bar{a}_t)], \bar{a}_t \Big\rangle \\ &- \gamma_1 \sum_{j,k=1}^{J} p_j p_k \Big\langle \mathbb{E}[\nabla_a \Psi(m_{j,t} + G^t, Y_j, \bar{a}_t)], \mathbb{E}[f(m_{k,t} + G^t, Y_k, \bar{a}_t)] \Big\rangle.\end{aligned}$$

This produces terms (A.19) and (A.20). Using (A.27), the first order contribution from $\Delta m_j$ gives,

$$\begin{aligned}\sum_{j=1}^{J} \langle \nabla_{m_j} \mathcal{G}, \Delta m_j \rangle = &-(1-\eta_0) \sum_{j=1}^{J} p_j \Big\langle \mathbb{E}[\nabla_h \Psi(m_{j,t} + G^t, Y_j, \bar{a}_t)], m_{j,t} \Big\rangle \\ &- \eta_1 \alpha \sum_{j,k=1}^{J} p_j \chi_{j,k} p_k \Big\langle \mathbb{E}[\nabla_h \Psi(m_{j,t} + G^t, Y_j, \bar{a}_t)], \mathbb{E}[g(m_{k,t} + G^t, Y_k, \bar{a}_t)] \Big\rangle.\end{aligned}$$

This produces terms (A.21) and (A.22). Using (A.28), The first order contribution from $\Delta\omega$ gives,

$$\begin{aligned}\langle \nabla_{\omega} \mathcal{G}, \Delta\omega \rangle = &-\frac{1}{2}(1-\eta_0^2) \sum_{j=1}^{J} p_j \Big\langle \mathbb{E}[\nabla_h^2 \Psi(m_{j,t} + G^t, Y_j, \bar{a}_t)], \omega_t \Big\rangle \\ &- \frac{1}{2}\eta_0\eta_1 \sum_{j=1}^{J} p_j \Big\langle \mathbb{E}[\nabla_h^2 \Psi(m_{j,t} + G^t, Y_j, \bar{a}_t)], \Xi_t[t,t] + \Xi_t[t,t]^\top \Big\rangle \\ &+ \frac{1}{2}\eta_1^2 \sum_{j=1}^{J} p_j \Big\langle \mathbb{E}[\nabla_h^2 \Psi(m_{j,t} + G^t, Y_j, \bar{a}_t)], \alpha\Sigma_t[t,t] + \alpha^2 \sum_{k,k'=1}^{J} \chi_{k,k'} \ell_{k,t} \ell_{k',t}^\top \Big\rangle \\ = &-\eta_0(1-\eta_0) \sum_{j=1}^{J} p_j \Big\langle \mathbb{E}[\nabla_h^2 \Psi(m_{j,t} + G^t, Y_j, \bar{a}_t)], \omega_t \Big\rangle \\ &- \frac{1}{2}\eta_0\eta_1 \sum_{j=1}^{J} p_j \Big\langle \mathbb{E}[\nabla_h^2 \Psi(m_{j,t} + G^t, Y_j, \bar{a}_t)], \Xi_t[t,t] + \Xi_t[t,t]^\top \Big\rangle\end{aligned}$$

$$+\frac{1}{2}\eta_1^2\sum_{j=1}^{J} p_j\Big\langle \mathbb{E}[\nabla_h^2\Psi(m_{j,t}+G^t,Y_j,\bar{a}_t)],\alpha\Sigma_t[t,t]+\alpha^2\sum_{k,k'=1}^{J}\chi_{k,k'}\ell_{k,t}\ell_{k',t}^\top\Big\rangle \quad \text{(A.29)}$$

$$-\frac{1}{2}(1-\eta_0)^2\sum_{j=1}^{J} p_j\Big\langle \mathbb{E}[\nabla_h^2\Psi(m_{j,t}+G^t,Y_j,\bar{a}_t)],\omega_t\Big\rangle \quad \text{(A.30)}$$

$$=-\eta_0(1-\eta_0)\sum_{j=1}^{J} p_j\Big\langle \mathbb{E}[\nabla_h^2\Psi(m_{j,t}+G^t,Y_j,\bar{a}_t)],\omega_t\Big\rangle$$

$$-\frac{1}{2}\eta_0\eta_1\sum_{j=1}^{J} p_j\Big\langle \mathbb{E}[\nabla_h^2\Psi(m_{j,t}+G^t,Y_j,\bar{a}_t)],\Xi_t[t,t]+\Xi_t[t,t]^\top\Big\rangle+\delta_t,$$

where $\sup_{t\in[T]}|\delta_t|\le C\epsilon^2$ for some constant $C>0$ for sufficiently small $\epsilon$. The final inequality in the above display follows by rewriting (A.30) using $(1-\eta_0^2)=(1-\eta_0)^2+2\eta_0(1-\eta_0)$, and using the inequality $\langle M_1,M_2\rangle\le\|M_1\|_F\|M_2\|_F$ to absorb term (A.29) using the $\eta_1^2$ prefactor. This second argument is allowed by the assumed boundedness of $\|\mathbb{E}[\nabla_h^2\Psi(m_{j,t}+G^t,Y_j,\bar{a}_t)]\|_F$ (Assumption A.4 (4) (b)), the boundedness of $\|\omega_t\|_F$, $\|\Sigma_t[t,t]\|_F$ independent of time $t\in[T]$ and recognizing that $\|\ell_{k,t}\ell_{k',t}^\top\|_F\le\|\ell_{k,t}\|_F\|\ell_{k',t}\|_F$ is bounded as $\ell_{k,t}$ is an expectation over $g$, a bounded function. This produces terms (A.23) and (A.24).

Combining all first order contributions and bounding the second order remainder by the $\epsilon^2$ order remainder $\delta_t$ (say by enlarging its bound by $2C\epsilon^2$) completes the proof. ■

*Proof of Theorem 4.1.* First, we invoke Theorem A.18 with the choice of $\eta_0,\gamma_0=1$, $\eta_1=\gamma_1=\eta$, $g=\nabla_h\Psi$, $f=\nabla_a\Psi$. This gives the Taylor expansion

$$\begin{aligned}\mathbf{test}_{t+1}=\mathbf{test}_t&\\ -\eta\sum_{j,j'=1}^{J} p_jp_{j'}\Big\langle \mathbb{E}[\nabla_a\Psi(m_{j,t}+G^t,Y_j,\bar{a}_t)],\mathbb{E}[\nabla_a\Psi(m_{j',t}+G^t,Y_{j'},\bar{a}_t)]\Big\rangle&\\ -\eta\alpha\sum_{j,j'=1}^{J} p_j\chi_{j,j'}p_{j'}\Big\langle \mathbb{E}[\nabla_h\Psi(m_{j,t}+G^t,Y_j,\bar{a}_t)],\mathbb{E}[\nabla_h\Psi(m_{j',t}+G^t,Y_{j'},\bar{a}_t)]\Big\rangle&\\ -\frac{\eta}{2}\sum_{j=1}^{J} p_j\Big\langle \mathbb{E}[\nabla_h^2\Psi(m_{j,t}+G^t,Y_j,\bar{a}_t)],\Xi_t[t,t]+\Xi_t[t,t]^\top\Big\rangle+\epsilon_t,&\end{aligned}$$

where $\sup_{t\in[T]}|\epsilon_t|\le C\eta^2$ for constant $C>0$ for sufficiently small $\eta>0$. Using the notation given in the statement of Theorem 4.1 and simplifying the above equation gives the proof. ■

Used as an important step in the proof of Theorem A.18, we present a high level observation that allows one control the norm of the matrix $\Xi_t[t,t]$ in the Taylor expansion of Theorem 4.1.

*Remark* A.20. Observe, the first term on line (4.2) need not be negative and may increase the test error. Moreover, unrolling $\Xi_t[t,t]$ from Definition A.7, this term is equal to,

$$\eta\sum_{j=1}^{J} p_j\Big\langle \mathbb{E}[\nabla_h^2\Psi(m_{j,t}+G^t,Y_j,\bar{a}_t)],\alpha\sum_{k=1}^{J} m_{k,1}\ell_{k,t}^\top-\eta\sum_{s=1}^{t-1}\Big(\alpha\Sigma_t[s,t]+\alpha^2\sum_{a,b=1}^{J}\chi_{a,b}\ell_{a,s}\ell_{b,t}^\top\Big)\Big\rangle.$$

It is not clear the above summation is bounded independent of time $t$. Thankfully, the assumption that $\Sigma_t[t,t]$ and $\Omega_t[t,t]$ have bounded operator norms (and thus bounded Frobenius norms) from Assumption A.9 allows for such control. Indeed, using the limits from Lemma 3.6 and

$$\left\|\frac{\theta_t^\top\tilde{\theta}_t}{d}\right\|_F\le\left\|\frac{\theta_t}{\sqrt{d}}\right\|_F\left\|\frac{\tilde{\theta}_t}{\sqrt{d}}\right\|_F,$$

we have that $\|\Xi_t[t,t]\|_F$, and thus the term in (4.2) by Assumption A.4 (4), is bounded in the $n,d$ limit.

### A.12 The Benefits of Damping For Test Error Critical Points

Further tuning hyperparameters in (3.2) can give favorable global behavior as well.

**Theorem A.21.** *Let $g = \nabla_h \Psi$ and $f = \nabla_a \Psi$ in Algorithm (3.2), with $\eta, \gamma \in (0,1)$, consider the one dimensional hyperparameter subspace over $c \in \mathbb{R}_+$,*

$$\eta_0 = 1 - \eta, \quad \eta_1 = c\eta, \quad \gamma_0 = 1, \quad \gamma_1 = \gamma.$$

*Any fixed point $(m_1^*, \dots, m_J^*, \Omega^*, \bar{a}^*)$ of Definition A.7, must satisfy the equations*

$$m_j^* = -c\alpha \left( \sum_{k=1}^{J} \chi_{j,k} p_k \mathbb{E}[g(m_k^* + G^*, Y_k, \bar{a}^*)] \right) \tag{A.31}$$

$$0 = -\sum_{j=1}^{J} p_j \mathbb{E}[f(m_j^* + G^*, Y_j, \bar{a}^*)], \tag{A.32}$$

*where $Y_j \sim \mathbb{P}_j$ and $G^* \sim \mathcal{N}(0, \Omega^*)$. Moreover, for any deterministic vector $v \in \mathbb{R}^d$ perpendicular to $(\mu_j)_{j \in [J]}$, any iterate $\theta$ produced by the above fixed point must satisfy $\lim_{d \to \infty} d^{-1} v^\top \theta = 0$.*

*Proof.* Assume the fixed point given in the statement of the theorem. Using the equations of $m_{j,t+1}$ and $\bar{a}_{t+1}$ for the given fixed point, we have that $(m_j^*)_{j \in [J]}$, $\Omega^*$, $\bar{a}^*$ satisfy the following system,

$$m_j^* = \eta_0 m_j^* - \eta_1 \alpha \sum_{k=1}^{J} \chi_{j,k} p_k \mathbb{E}[g(m_k^* + G^*, Y_k, \bar{a}^*)]$$

$$\bar{a}^* = \gamma_0 \bar{a}^* - \gamma_1 \sum_{j=1}^{J} p_j \mathbb{E}[f(m_j^* + G^*, Y_j, \bar{a}^*)],$$

where $Y_j \sim \mathbb{P}_j$ and $G^* \sim \mathcal{N}(0, \Omega^*)$. Rearranging these equations and simplifying gives the fixed point condition in the theorem. Note, by adding a dummy signal vector $\mu_{J+1} = v, p_{J+1} = 0$ where $v \perp \mu_j$ for $j \in [J]$ into distribution (3.1), we can also conclude the second statement in the theorem, i.e. that $m_{J+1}^* = 0$, as $\chi_{J+1,k} = 0$ for all $k \in [J]$ and $\eta_0 \in (0,1)$. ■

Let $(\theta^*, a^*)$ be a critical point for the test error, assume that $H^* = \sum_{j=1}^{J} p_j \mathbb{E}[\nabla_h^2 \Psi(\mu_j^\top \theta^*/d + Z^\top \theta^*, Y_j, a^*)]$ is invertible and isotropic when $Z \sim \mathcal{N}(0, \mathrm{Id}_d/d)$. Then, there exists a $c \in \mathbb{R}_+$ where (A.31) and (A.32) are equivalent to projected critical point conditions on $(\theta^*, a^*)$.

A test error critical point satisfies $\nabla_a \mathbb{E}_{\check{x},\check{y}} \left[ \mathcal{L}(\mathcal{M}_{a^*}(\check{x}^\top \theta^*), \check{y}) \right] = 0$ and $\nabla_\theta \mathbb{E}_{\check{x},\check{y}} \left[ \mathcal{L}(\mathcal{M}_{a^*}(\check{x}^\top \theta^*), \check{y}) \right] = 0$ with $(\check{x}, \check{y})$ drawn from distribution (3.1). Expanding (3.1) in terms of the mixture on $\mu_1, \dots, \mu_J$, interchanging expectation and derivative by Lemma A.16, we equivalently write this system as,

$$\sum_{j=1}^{J} p_j \mathbb{E}[\nabla_a \Psi(\mu_j^\top \theta^*/d + Z^\top \theta^*, Y_j, a^*)] = 0, \tag{A.33}$$

$$-\sum_{j=1}^{J} p_j \mu_j \mathbb{E}[\nabla_h \Psi(\mu_j^\top \theta^*/d + Z^\top \theta^*, Y_j, a^*)] = \theta^* H^*, \tag{A.34}$$

where $Z \sim \mathcal{N}(0, \mathrm{Id}_d/d)$. As $H^*$ is invertible, for any deterministic vector $v \in \mathbb{R}^d$, (A.34) implies that,

$$\frac{v^\top \theta^*}{d} = -\Big( \sum_{j=1}^{J} p_j \frac{v^\top \mu_j}{d} \mathbb{E}[\nabla_h \Psi(\mu_j^\top \theta^*/d + Z^\top \theta^*, Y_j, a^*)] \Big) (H^*)^{-1}. \tag{A.35}$$

There are two cases of interest: (1) If $v = \mu_j$ for some $j \in [J]$, assuming that $\lim_{d \to \infty} \mu_j^\top \theta^*/d = m_{j,\theta^*}$, $\lim_{d \to \infty} (\theta^*)^\top \theta^*/d = \Omega_{\theta^*}$, $\lim_{n,d \to \infty} a^* = \bar{a}_{\theta^*}$, and replacing $(H^*)^{-1} = (c^*)^{-1} \mathrm{Id}$ for

some constant $c^* > 0$ by the isotropic assumption, asymptotically solving (A.35) is equivalent to,

$$m_{j,\theta} = -c^* \sum_{k=1}^{J} \chi_{j,k} p_k \mathbb{E}[\nabla_h \Psi(m_{k,\theta^*} + G, Y_k, \bar{a}_{\theta^*}))], \quad G \sim \mathcal{N}(0, \Omega_{\theta^*}); \tag{A.36}$$

(2) If $v$ is perpendicular to the span of $(\mu_j)_{j\in[J]}$, then assuming that $\lim_{d\to\infty} \frac{v^\top \theta}{d} = m^\perp$ almost surely, we have the asymptotic equation

$$m^\perp = 0. \tag{A.37}$$

Next, under the same limits $a, \mu_j^\top \theta/d$ and $\theta^\top \theta/d$, (A.33) is asymptotically equivalent to,

$$\sum_{j=1}^{J} p_j \mathbb{E}[\nabla_{\bar{a}} \Psi(m_{j,\theta^*} + G, Y_j, \bar{a}_{\theta^*})] = 0, \quad G \sim \mathcal{N}(0, \Omega_{\theta^*}). \tag{A.38}$$

Comparing (A.36), (A.37) and (A.38) with $m_{j,\theta^*} = m_j^*$ for each $j \in [J]$, $\Omega_{\theta^*} = \Omega^*$ and $\bar{a}_{\theta^*} = \bar{a}^*$, and selecting $c = (c^*\alpha)^{-1}$, the statement of Theorem A.21 is identical to the set of equations (A.33) and (A.36).

*Remark* A.22 (Adaptive Hessian estimation). Note, in the case where $H^*$ is not isotropic, one naturally would incorporate the Hessian $H_t$ into a DD algorithm, the data-based estimate of

$$\hat{H}_t = \frac{1}{n} \sum_{i=1}^{n} \nabla_h^2 \Psi(h_{t,i}, y_i, a_t),$$

is already calculated for the correction term in both pure DD and is expected to be an asymptotically exact estimate of $H_t$. But, this requires a time-inhomogeneous choice of $g = g_t$; such an extension to our results is possible but is left to future work.

# B Deferred Applications

## B.1 Faithful Test Error Tracking In Signal-less Learning

Recall this problem from the introduction, consider $n$ i.i.d. data points $x_i \sim \mathcal{N}(0, \mathrm{Id}_d/d)$ and $y_i = 0$, generating the data $(X, y)$. Let $\mathcal{M}_\theta(x) = x^\top \theta$ with $\theta \in \mathbb{R}^d$ initialized at $\theta_1 \sim \mathcal{N}(0, \mathrm{Id}_d)$ and trained under loss $\mathcal{L}(\hat{y}, y) = \frac{1}{2}(\hat{y} - y)^2$ (rescaled for convenience), realized by distribution (3.1) with $J = 1$ and $\mu_1$ being the zero vector.

We run the following algorithms:

| **GD** | **(Damped) Pure DD** |
|---|---|
| $h_t = X\theta_t$ | $h_t = X\theta_t + \eta_1 \sum_{s=1}^{t-1} \eta_0^{(t-1)-s} h_s$ |
| $\tilde{\theta}_t = X^\top h_t$ | $\tilde{\theta}_t = X^\top h_t - \alpha\theta_t$ |
| $\theta_{t+1} = \theta_t - \eta\tilde{\theta}_t$ | $\theta_{t+1} = \eta_0\theta_t - \eta_1\tilde{\theta}_t$ |

Figure 5 plots the train and test errors for GD and pure DD ($\eta_0 = 1, \eta_1 = \eta$). We immediately see the train-test disconnect in GD is corrected in both DD algorithms. Note, for pure DD, the train-test error explodes. This is predicted from the variance inflation terms in $\Omega_t$ from Definition A.7 and occurs because this problem has no signal. Specifically, we can use Remark A.8 to conclude the variance recursion (and thus the test error recursion) takes form,

$$\Omega_{t+1}[t+1, t+1] = \eta_0^2 \Omega_t[t,t] - \eta_1\eta_0(\Xi_t[t,t] + \Xi_t[t,t]^\top) + \eta_1^2 \alpha \Omega_t[t,t],$$

which may or may not be a contraction depending on the choices of $\eta_0, \eta_1$.

This naturally leads into relying on the zero-cost validation feature of DD. We consider the following one dimensional subspace dependent on $\eta$,

$$\eta_0 = 1 - \eta, \qquad \eta_1 = c\eta. \tag{B.1}$$

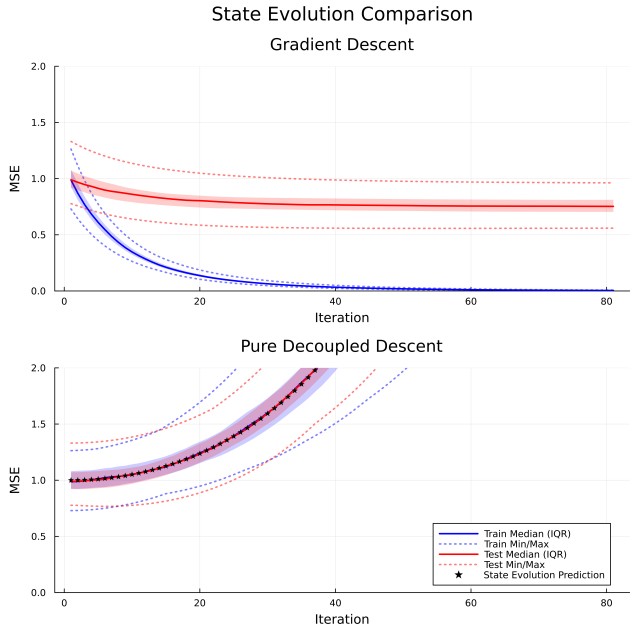


Figure 5: Summary stats for 100 signal-less regression runs ($n = 200, d = 800$): GD (top) vs. DD (bottom) with $\eta = 0.05$. Blue/red denote train/test error; solid lines are medians, shaded areas are interquartile ranges, and dotted lines show min/max. Notice, by design, the trajectories of the train and test error are identical for pure DD while they significantly diverge for GD.

We then sweep the value of $c$ over some specified range for a fixed data realization. The results of many individual replications of this experiment are given in Figure 6. Again, for each simulation the train-test identity holds perfectly as expected and the trajectory of the test error is tracked by the state evolution prediction. Moreover, since each of these replications used the same dataset, we can optimize over the hyperparameter $c$ by selecting whichever algorithm gets the optimal training (and thus test) error.

### B.2 The XOR Training Iterations

**GD**

$$
\begin{aligned}
(h_t^1, h_t^2) &= X(\theta_t^1, \theta_t^2) \\
\hat{y}_t &= \sigma(a_t(h_t^1 \odot h_t^2)) \\
\hat{h}_t^1 &= (\hat{y}_t - y) \odot (a_t h_t^2) \\
\hat{h}_t^2 &= (\hat{y}_t - y) \odot (a_t h_t^1) \\
\theta_{t+1}^1 &= \theta_t^1 - \eta X^\top \hat{h}_t^1 \\
\theta_{t+1}^2 &= \theta_t^2 - \eta X^\top \hat{h}_t^2 \\
a_{t+1} &= a_t - \eta(\hat{y}_t - y)^\top (h_t^1 \odot h_t^2)/n
\end{aligned}
$$

**Pure DD**

$$
\begin{aligned}
(h_t^1, h_t^2) &= X(\theta_t^1, \theta_t^2) + \eta \sum_{s=1}^{t-1} (\hat{h}_s^1, \hat{h}_s^2), \\
\hat{y}_t &= \sigma(a_t(h_t^1 \odot h_t^2)), \\
\hat{h}_t^1 &= (\hat{y}_t - y) \odot (a_t h_t^2) \\
\hat{h}_t^2 &= (\hat{y}_t - y) \odot (a_t h_t^1), \\
H_t &= \frac{1}{n} \sum_{i=1}^{n} \left( \hat{y}_{t,i}(1-\hat{y}_{t,i}) a_t^2 \begin{bmatrix} (h_{t,i}^2)^2 & h_{t,i}^1 h_{t,i}^2 \\ h_{t,i}^1 h_{t,i}^2 & (h_{t,i}^1)^2 \end{bmatrix} \right. \\
&\qquad \left. + (\hat{y}_{t,i} - y_i) \begin{bmatrix} 0 & a_t \\ a_t & 0 \end{bmatrix} \right) \\
\tilde{\theta}_t &= X^\top (\hat{h}_t^1, \hat{h}_t^2) - \alpha(\theta_t^1, \theta_t^2) H_t \\
(\theta_{t+1}^1, \theta_{t+1}^2) &= (\theta_t^1, \theta_t^2) - \eta \tilde{\theta}_t \\
a_{t+1} &= a_t - \eta(\hat{y}_t - y)^\top (h_t^1 \odot h_t^2)/n
\end{aligned}
$$

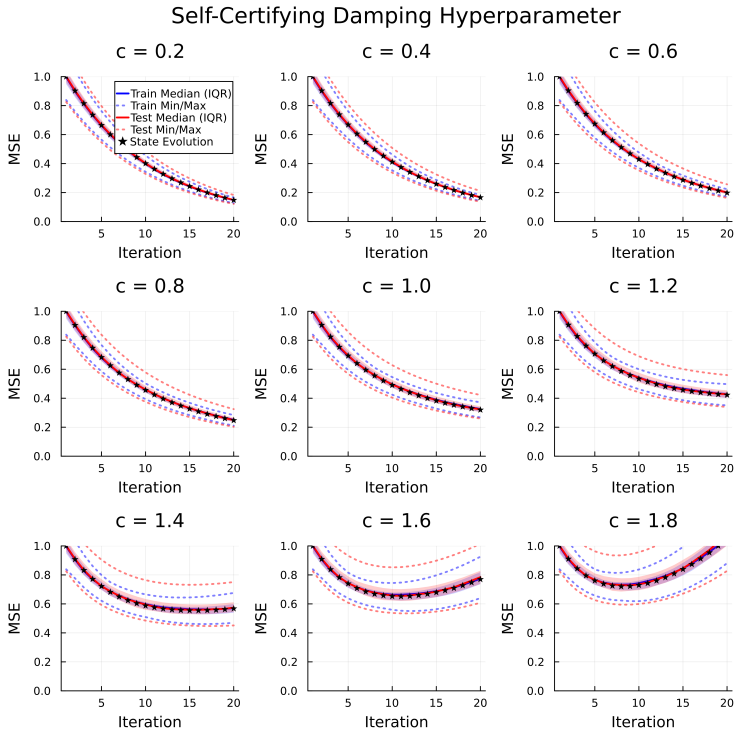


Figure 6: An example of hyperparameter tuning with DD over 100 individual replications, in each figure is the train and test error for damped pure DD algorithm in the signal-less regression problem with $n = 800, d = 800$, $\eta = 0.05$ and we sweep $c$ over the sub-space given in (B.1). Blue lines refer to train error and red lines refer to test error; the solid line is the median error, the shaded region is the range from the 25-th to 75-th quartiles and the dotted lines are the minimum and maximum error runs. This represents an application where multiple instances of DD are run on the same data set and, because the train-test identity, we get a zero-cost validation method to selecting good hyperparameters that aid in training.

**Damped Variant Of Pure DD (Fixed $a_t = 1$)**

$$
\begin{aligned}
(h_t^1, h_t^2) &= X(\theta_t^1, \theta_t^2) + \eta_1 \sum_{s=1}^{t-1} \eta_0^{(t-1)-s} (\hat{h}_s^1, \hat{h}_s^2), \\
\hat{y}_t &= \sigma(h_t^1 \odot h_t^2), \\
\hat{h}_t^1 &= (\hat{y}_t - y) \odot (h_t^2) \\
\hat{h}_t^2 &= (\hat{y}_t - y) \odot (h_t^1), \\
H_t &= \frac{1}{n} \sum_{i=1}^{n} \left( \hat{y}_{t,i}(1-\hat{y}_{t,i}) \begin{bmatrix} (h_{t,i}^2)^2 & h_{t,i}^1 h_{t,i}^2 \\ h_{t,i}^1 h_{t,i}^2 & (h_{t,i}^1)^2 \end{bmatrix} + (\hat{y}_{t,i} - y_i) \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \right) \\
\tilde{\theta}_t &= X^\top (\hat{h}_t^1, \hat{h}_t^2) - \alpha(\theta_t^1, \theta_t^2) H_t \\
(\theta_{t+1}^1, \theta_{t+1}^2) &= \eta_0 (\theta_t^1, \theta_t^2) - \eta_1 \tilde{\theta}_t
\end{aligned}
$$

### B.3 A Run-time Analysis For The MNIST Problem

Figure 7 tracks the "wall clock" time per epoch (i.e. a single full batch) for our width-nine hidden layer model at both the per epoch level and for total training time. Although our implementation of DD suffers from a $\approx L\times$ constant factor overhead, we believe the practical benefits of DD (including use of the full training set) may outweigh this computational penalty. A variant of DD mimicking SGD may further reduce this overhead (Section 6; future directions).

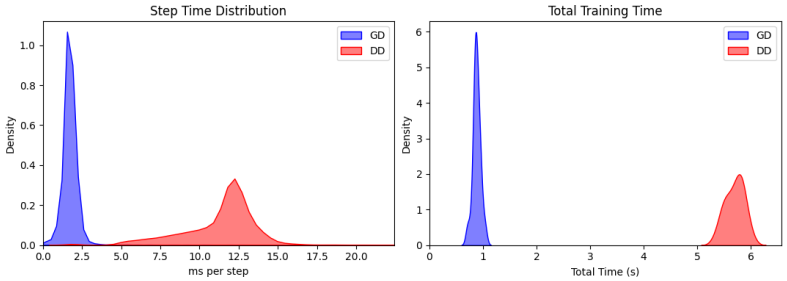


Figure 7: Empirical density estimates for runtime on a width-9 two-layer neural network (25 replications, 500 epochs). Left: Distribution of per-epoch clock time (ms) for GD vs. DD. Right: Total training time (s) per replication for the MNIST zeros/eights problem.

### B.4 Implementation of MNIST Problem With a Width-Nine Hidden Layer

Below is our implementation of the MNIST zeros/eights problem for a width-nine hidden layer model under multiple repeated simulations.

```
import time, math, torch, numpy as np
import torch.nn as nn, torch.nn.functional as F
from torchvision import datasets, transforms

def get_data(train):
    d = datasets.MNIST("./data", train=train, download=True, transform=transforms.ToTensor())
    m = (d.targets == 0) | (d.targets == 8)
    return d.data[m].float() / 255.0, (d.targets[m] == 8).float()

x_tr, y_tr = get_data(True)
x_test, y_test = get_data(False)

N, T, H, lr, lam, reps = 800, 500, 9, 1, 30, 20
device = "mps" if torch.backends.mps.is_available() else "cpu"
x_test, y_test = x_test.unsqueeze(1).to(device), y_test.to(device)
alpha = N / 784

def inject_noise(x, lam):
    D = x.shape[-1] * x.shape[-2]
    p = torch.rand_like(x)
    z = torch.where(p < 0.25, -math.sqrt(2/D), torch.where(p < 0.5, math.sqrt(2/D), 0.0))
    return (lam / D) * x + z.to(x.device)

class DecoupledMLP(nn.Module):
    def __init__(self):
        super().__init__()
        self.fc1, self.fc2 = nn.Linear(784, H, bias=False), nn.Linear(H, 1, bias=False)
    def forward(self, x, mem=0):
        z = self.fc1(x.view(x.size(0), -1)) + mem
        return self.fc2(torch.tanh(z)).squeeze(1), z

def init_mlp_weights(m):
    if hasattr(m, 'fc1') and m.fc1.weight is not None:
        nn.init.normal_(m.fc1.weight, mean=0.0, std=1.0)
    if hasattr(m, 'fc2') and m.fc2.weight is not None:
        nn.init.normal_(m.fc2.weight, mean=0.0, std=1.0)

dd_err = np.zeros((2, reps, T))
dd_time = []

for r in range(reps):
    idx = torch.randperm(len(x_tr))[:N]
    x_n, y_n = inject_noise(x_tr[idx].unsqueeze(1).to(device), lam), y_tr[idx].to(device)
    x_te_n = inject_noise(x_test, lam)

    m_dd, mem = DecoupledMLP().to(device), torch.zeros(N, H, device=device)
```

```
    init_mlp_weights(m_dd)
    t0 = time.time()
    for t in range(T):
        logits, z = m_dd(x_n, mem)
        loss = F.binary_cross_entropy_with_logits(logits, y_n)
        h_hat = torch.autograd.grad(loss, z, create_graph=True)[0]

        J_bar = torch.stack([torch.autograd.grad(h_hat[:, i].sum(), z,
            ↪ retain_graph=True)[0].mean(dim=0) for i in range(H)])

        with torch.no_grad():
            m_dd.fc2.weight -= lr * torch.autograd.grad(loss, m_dd.fc2.weight,
                ↪ retain_graph=True)[0]
            theta = m_dd.fc1.weight
            m_dd.fc1.weight.copy_(theta - lr * (h_hat.t() @ x_n.view(N, -1) - alpha * (J_bar
                ↪ @ theta)))
            mem += lr * h_hat.detach()

        dd_err[0, r, t] = loss.item()
        with torch.no_grad(): dd_err[1, r, t] =
            ↪ F.binary_cross_entropy_with_logits(m_dd(x_te_n)[0], y_test).item()
    dd_time.append(time.time() - t0)
```

### B.5 Whitening Technique For CIFAR-10 Embedding Vectors

We considered the following pre-processing steps, which approach the conditions of Assumption A.4 to varying degrees:

- *Vanilla (no whitening):* Features are rescaled by a $1/\sqrt{d}$ factor (assuming the embedding vectors have Gaussian entries). This leaves correlations between coordinates of the embedding vectors which our current DD algorithm does not account for.
- *ZCA (whitening) on the train data:* We compute the empirical covariance matrix $\Sigma_{\text{train}} = \frac{1}{N-1} X_{\text{train}}^T X_{\text{train}}$ exclusively on the training set and compute the eigen-decomposition $\Sigma_{\text{train}} = V \Lambda V^T$. We construct the Zero-phase Component Analysis (ZCA) [20] matrix $W_{\text{train}} = V(\Lambda)^{-1/2} V^T$, which is then applied to both the train and test data using $X' = X W_{\text{train}}$ and $\check{X}' = \check{X} W_{\text{train}}$. Finally, we apply the $1/\sqrt{d}$ normalization from the Vanilla setting.
- *Joint ZCA (whitening) on train and test data:* To prevent drift in the covariance between the train and test sets, we concatenate both data sets to form a joint covariance matrix $\Sigma_{\text{joint}}$. The resulting $W_{\text{joint}}$ is applied to all data with,

$$\begin{bmatrix} X' \\ \check{X}' \end{bmatrix} = \begin{bmatrix} X W_{\text{joint}} \\ \check{X} W_{\text{joint}} \end{bmatrix}.$$

  Finally, we apply the $1/\sqrt{d}$ normalization from the Vanilla setting. Note, this whitening variant is an idealized diagnostic for covariance alignment, not an implementable training procedure.

### B.6 Data Collection and Attribution

This work utilized both the **MNIST** and **CIFAR-10** datasets, standard benchmarks in the high-dimensional statistics and machine learning literature.

- **MNIST:** A collection of $70,000$ handwritten digit images [23]. The dataset is available under the **Creative Commons Attribution-Share Alike 3.0** license.
- **CIFAR-10:** Consists of $60,000$ $32 \times 32$ color images in 10 classes [22]. The dataset is used in accordance with the researchers' terms at the University of Toronto.

## NeurIPS Paper Checklist

1. **Claims**
   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?
   Answer: [Yes]
   Justification: Both the abstract and introduction mention the major assumption where the data must come from a mixture Gaussian Law. Each claim on avoiding overfitting and zero-cost validation is explained in the main text.
   Guidelines:
   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.
2. **Limitations**
   Question: Does the paper discuss the limitations of the work performed by the authors?
   Answer: [Yes]
   Justification: There is a limitations subsection in the conclusion which lists out many limitations of this method. More are dispersed through the main text including methods on possibly relaxing strong assumptions.
   Guidelines:
   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.
3. **Theory assumptions and proofs**
   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?
   Answer: [Yes]
   Justification: Many of the proofs and formal assumptions are delayed to the appendix but are present in the paper, informal theorems / assumptions also point to formal statements.
   Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The paper gives verbatim equations on how to implement the training protocol and gives pytorch code used to train the MNIST example (which can be further extrapolated to the CIFAR-10) example).

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No]

   Justification: We have included the implementation of the MNIST example in Appendix C but are not providing access to the full code.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.

- Please see the NeurIPS code and data submission guidelines (https://neurips.cc/public/guides/CodeSubmissionPolicy) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://neurips.cc/public/guides/CodeSubmissionPolicy) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**
   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?
   Answer: [Yes]
   Justification: Each application in this paper specifies the required hyperparameters and settings. Further details are given in the appendix for each application.
   Guidelines:
   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.
7. **Experiment statistical significance**
   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?
   Answer: [Yes]
   Justification: Although the paper does not perform experiments using statistical significance, we do include IQR on our simulations to gibe a notion of spread in our simulations.
   Guidelines:
   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
   - If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.
8. **Experiments compute resources**
   Question: For each experiment, does the paper provide sufficient information on the com-

puter resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?
Answer: [Yes]
Justification: The paper gives the machine used to run each experiment in the paper, they were identical. It also comments on the run-time for a specific example but the run-time is not neccessary to reproduce any of the experiments.
Guidelines:
- The answer [N/A] means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**
Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?
Answer: [Yes]
Justification: This paper does not violate the code of ethics.
Guidelines:
- The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**
Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?
Answer: [Yes]
Justification: The paper gives a brief broader impacts section in the conclusion.
Guidelines:
- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**
Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]
Justification: [N/A]
Guidelines:
- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**
Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?
Answer: [Yes]
Justification: In the final subsection of appendix A, the paper cites the licences and / or original works for both the MNIST data set and the CIFAR-10 dataset.
Guidelines:
- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, paperswithcode.com/datasets has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**
Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?
Answer: [N/A]
Justification: [N/A]
Guidelines:
- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**
Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?
Answer: [N/A]
Justification: [N/A]
Guidelines:
- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.

- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: [N/A]

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: Google Gemini was utilized for writing the code in the application section and the supplementary material.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.