Sliding Down the Stairs: How Correlated Latent Variables Accelerate Learning with Neural Networks

Lorenzo Bardone [1] **Sebastian Goldt** [1]

Abstract

Neural networks extract features from data using stochastic gradient descent (SGD). In particular, higher-order input cumulants (HOCs) are crucial for their performance. However, extracting information from the pth cumulant of d-dimensional inputs is computationally hard: the number of samples required to recover a single direction from an order-p tensor (tensor PCA) using online SGD grows as d^{p-1}, which is prohibitive for high-dimensional inputs. This result raises the question of how neural networks extract relevant directions from the HOCs of their inputs *efficiently*. Here, we show that correlations between latent variables along the directions encoded in different input cumulants speed up learning from higher-order correlations. We show this effect analytically by deriving nearly sharp thresholds for the number of samples required by a single neuron to weakly-recover these directions using online SGD from a random start in high dimensions. Our analytical results are confirmed in simulations of two-layer neural networks and unveil a new mechanism for hierarchical learning in neural networks.

1. Introduction

Neural networks excel at learning rich representations of their data, but which parts of a data set are actually important for them? From a statistical point of view, we can decompose the data distribution into cumulants, which capture correlations between groups of variables. The first cumulant is the mean, the second describes pair-wise correlations, and *higher-order cumulants* (HOCs) encode correlations between three or more variables. In image classification, HOCs are particularly important: on CIFAR10, removing HOCs from the training distribution incurs a drop in test accuracy of up to 65 % for DenseNets, ResNets, and Vision transformers (Refinetti et al., 2023).

While HOCs are important for the performance of neural networks, extracting information from HOCs with stochastic gradient descent (SGD) is computationally hard. Take the simple case of tensor PCA (Richard & Montanari, 2014), where one aims to recover a "spike" $u \in \mathbb{R}^d$ from an order-p tensor T, which could be the order-p cumulant of the inputs. Modelling the tensor as $T = \beta u^{\otimes p} + Z$, with signal-to-noise ratio (SNR) $\beta > 0$ and noise tensor Z with i.i.d. entries of mean zero and variance 1, Ben Arous et al. (2021) showed that online SGD with a single neuron requires a number of samples $n \gtrsim d^{p-1}$ to recover u. While smoothing the loss landscape can reduce to $n \gtrsim d^{p/2}$, as suggested by Correlational Statistical Query bounds (CSQ) (Damian et al., 2022; Abbé et al., 2023), a sample complexity that is exponential in p is too expensive for high-dimensional inputs like images. For supervised learning, Székely et al. (2023) showed that the number of samples required to strongly distinguish two classes of inputs $x \in \mathbb{R}^d$ whose distributions have the same mean and covariance, but different fourth- and higher-order cumulants, scales as $n \gtrsim d^2$ for the wide class of polynomial-time algorithms covered by the low-degree conjecture (Barak et al., 2019; Hopkins et al., 2017; Hopkins & Steurer, 2017; Hopkins, 2018). Their numerical experiments confirmed that two-layer neural networks indeed require quadratic sample complexity to learn this binary classification task.

The theoretical difficulty of learning from HOCs is in apparent contradiction to the importance of HOCs for the performance of neural networks, and the speed with which neural networks pick them up (Ingrosso & Goldt, 2022; Refinetti et al., 2023; Merger et al., 2023; Belrose et al., 2024), lead us to the following question:

How do neural networks extract information from higher-order input correlations efficiently*?*

In this paper, we show that neural networks can learn efficiently from higher-order cumulants by exploiting **correlations between the latent variables of type $u \cdot x$ corresponding to input cumulants of different orders**.

[1]International School of Advanced Studies, Trieste, Italy. Correspondence to: Lorenzo Bardone <lbardone@sissa.it>, Sebastian Goldt <sgoldt@sissa.it>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

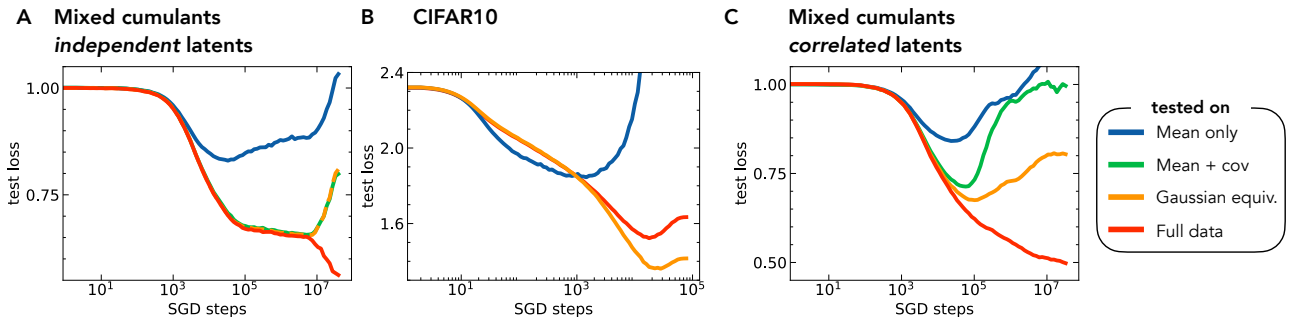

Figure 1. **Correlated latent variables speed up learning of neural networks. A** Test error of a two-layer neural network trained on the mixed cumulant model (MCM) of Equation (1) with signal-to-noise ratios $\beta_m = 1, \beta_u = 5, \beta_v = 10$. The MCM is a binary classification tasks where the inputs in the two classes have a different mean, a different covariance, and different higher-order cumulants (HOCs). We show the test error on the full data set (red) and on several "censored" data sets: a test set where only the mean of the inputs is different in each class (blue, $\beta_m = 1, \beta_u = \beta_v = 0$), a test set where mean and covariance are different (green, $\beta_m = 1, \beta_u = 5, \beta_v = 0$), and a Gaussian mixture that is fitted to the true data set (orange). The neural networks learn distributions of increasing complexity: initially, only the difference means matter, as the blue and red curves coincide; later, the network learns about differences at the level of the covariance, and finally at the level of higher-order cumulants. **B** Test loss of a two-layer neural network trained on CIFAR10 and evaluated on CIFAR10 (red), a Gaussian mixture with the means fitted to CIFAR10 (blue) and a Gaussian mixture with the means and covariance fitted on CIFAR10 (orange). **C** Same setup as in **A**, but here the latent variables corresponding to the covariance and the cumulants of the inputs are correlated, leading to a significant speed-up of learning from HOCs (the red and orange line separate after $\gtrsim 10^4$ steps, rather than $\gtrsim 10^6$ steps). *Parameters:* $\beta_m = 1, \beta_u = 5, \beta_v = 10, d = 128, m = 512$ hidden neurons, ReLU activation function. Full details in Appendix B.1.

2. Results and insights: an informal overview

2.1. The mixed cumulant model (MCM)

We illustrate how correlated latent variables speed up learning by introducing a simple model for the data, the mixed-cumulant model (MCM). The MCM is a binary discrimination task where the signals that differentiate the two classes are carried by different input cumulants. Specifically, we draw n data points $x^\mu = (x_i^\mu) \in \mathbb{R}^d$ with $\mu = 0, 1, \dots, n$ either from the isotropic Gaussian distribution $\mathbb{Q}_0$ (label $y^\mu = -1$) or from a *planted distribution* $\mathbb{Q}_{\text{plant}}$ (label $y^\mu = 1$). Under $\mathbb{Q}_0$, inputs have zero mean, an isotropic covariance equal to the identity $\mathbb{1}$, and all HOCs are zero. Under $\mathbb{Q}_{\text{plant}}$, the first few cumulants each carry a signal that distinguishes the inputs from those in $\mathbb{Q}_0$: a non-zero mean $m \in \mathbb{R}^d$, a covariance which is isotropic except in the direction $u \in \mathbb{R}^d$, and higher-order cumulants proportional to $v^{\otimes k}, k \geq 2$.

We sample an input from $\mathbb{Q}_{\text{plant}}$ by first drawing an i.i.d. Gaussian vector z^μ and two scalar latent variables, the normally distributed $\lambda^\mu \sim \mathcal{N}(0,1)$ and ν^μ, which is drawn from a non-Gaussian distribution. For concreteness, we will assume that $\nu^\mu = \pm 1$ with equal probability. Both latent variables are independent of z. Then, for $y^\mu = 1$, we have

$$x^\mu = \underbrace{\overbrace{\beta_m m}^{\text{mean}} + \overbrace{\sqrt{\beta_u}\lambda^\mu u}^{\text{covariance}}}_{\text{Gaussian}} + \underbrace{S(\sqrt{\beta_v}\nu^\mu v}_{\text{HOCs}} + z^\mu). \qquad (1)$$

where $\beta_i \geq 0$ are the signal-to-noise ratios associated to the three directions, or "spikes", m, u, v. The spikes are fixed and drawn uniformly from the unit sphere. We will sometimes force them to be orthogonal to one another. If $\beta_v = 0$, it is easy to verify that inputs are Gaussian with mean $\beta_m m$ and covariance $\mathbb{1} + \beta_u u u^\top$. If $\beta_v > 0$, inputs are non-Gaussian but the presence of the whitening matrix

$$S = \mathbb{1} - \frac{\beta_v}{1 + \beta_v + \sqrt{1 + \beta_v}} v v^\top \qquad (2)$$

removes the direction v from the covariance matrix, so that v cannot be recovered from the input covariance if the latent variables λ^ν, ν^μ are uncorrelated.

2.2. Neural networks take a long time to learn the cumulant spike in the vanilla MCM

We show the performance of a two-layer neural network trained on the MCM model in Figure 1A with signal-to-noise ratios $\beta_m = 1, \beta_u = 5, \beta_v = 10$ and independent latent variables $\lambda^\mu \sim \mathcal{N}(0,1)$ and $\nu^\mu = \pm 1$ with equal probability (red line). We can evaluate which of the three directions have been learnt by the network at any point in time by *evaluating* the same network on a reduced test set where only a subset of the spikes are present. Testing the network on a test set where the only difference between the two classes $\mathbb{Q}_0$ and $\mathbb{Q}_{\text{plant}}$ are the mean of the inputs (blue, $\beta_u = \beta_v = 0$) or the mean and covariance ($\beta_v = 0$), we find that two-layer networks learn about the different directions in a sequential way, learning first about the mean, then the

covariance, and finally the higher-order cumulants. This is an example of the distributional simplicity bias (Ingrosso & Goldt, 2022; Refinetti et al., 2023; Nestler et al., 2023; Belrose et al., 2024). However, note that the periods of sudden improvement, where the networks discovers a new direction in the data, are followed by long plateaus where the test error does not improve. The plateaus notably delay the learning of the direction v that is carried by the higher-order cumulants. We also see an "overfitting" on the censored test sets: as the network discovers a direction that is not present in the censored data sets, this appears as overfitting on the censored tests.

2.3. Correlated latent variables speed up learning non-Gaussian directions

As it stands, the MCM with orthogonal spikes and independent latent variables is a poor model for real data: neural networks show long plateaus between learning the Gaussian and non-Gaussian part of the data on the MCM, but if we train the same network on CIFAR10, we see a smooth decay of the test loss, see Figure 1B. Moreover, testing a network trained on CIFAR10 on a Gaussian mixture with the means fitted to CIFAR10 (blue) and a Gaussian mixture with the means and covariance fitted on CIFAR10 (orange) shows that the network goes smoothly and quickly from Gaussian to the non-Gaussian part of the data, without a plateau.

A natural idea improve the MCM, i.e. to make the loss curves of a neural network trained on the MCM resemble more the dynamics observed on CIFAR10, is to correlate the covariance and cumulant spikes: instead of choosing them to be orthogonal to each other, one could give them a finite overlap $u \cdot v = \rho$. However, a detailed analysis of the SGD dynamics in Section 3.2 will show that this does *not* speed up learning the non-Gaussian part of the data; instead, the crucial ingredient to speed up learning of non-Gaussian correlations is the correlation between latent variables. Setting for example

$$\nu^\mu = \text{sign}(\lambda^\mu), \tag{3}$$

we obtain the generalisation dynamics shown in red in Figure 1C: the plateaus have disappeared, and we get a behaviour that is very close to real data: a single exponential decay of the test loss.

2.4. A rigorous analysis of a single neuron quantifies the speed-up of correlated latents

We can make our experimental observations for two-layer networks rigorous in the simplest model of a neural network a single neuron $f(w, x) = \sigma(w \cdot x)$ trained using online projected stochastic gradient descent (also known as the spherical perceptron). At each step t of the algorithm, we sample a tuple (x_t, y_t) from the MCM (1) and update the weight w_t according to the following rule:

$$w_t = \begin{cases} w_0 \sim \text{Unif}\left(\mathbb{S}^{d-1}\right) & t = 0 \\ \tilde{w}_t = w_{t-1} - \frac{\delta}{d}\nabla_{\text{sph}}\left(\mathcal{L}(w, (x_t, y_t)\right) & t \geq 1 \\ w_t = \frac{\tilde{w}_t}{||\tilde{w}_t||}. \end{cases} \tag{4}$$

Here, ∇_{sph} is the spherical gradient defined by $\nabla_{\text{sph}} f(w) = (\mathbb{1} - ww^\top)\nabla f(w)$. We follow Damian et al. (2023) in using the *correlation loss* for our analysis,

$$\mathcal{L}(w, (x, y)) = 1 - yf(w, x). \tag{5}$$

In high dimensions, the typical overlap of the weight vector at initialisation and, say, the cumulant spike scales as $w_0 \cdot v \simeq d^{-1/2}$. We will say we the perceptron has learnt the direction v if we have "weakly recovered" the spike, i.e. when the overlap $\alpha_v \equiv w \cdot v \sim O(1)$. This transition from diminishing to macroscopic overlap marks the exit of the search phase of stochastic gradient descent, and it often requires most of the runtime of online SGD (Ben Arous et al., 2021).

In this setup, we can give a precise characterisation of the sample complexity of learning a single spike (either in the covariance or in the cumulant) and of learning in the presence of two spikes with independent or correlated latent variables. By looking at stochastic gradient descent, we can also make quantitative statements on the optimal learning rates to achieve weak recovery quickly. For simplicity, our perceptron analysis does not consider the spike in the mean, i.e. $\beta_m = 0$ throughout. This assumption is mainly to enhance the mathematical tractability of the model and we expect most of the following to hold also in the case $\beta_m \neq 0$. Our main theoretical results are then as follows:

Learning a single direction: If only the covariance *or* the cumulant are spiked, i.e. either $\beta_u = 0$ or $\beta_v = 0$, the analysis of Ben Arous et al. (2021) applies directly and we find that projected online SGD requires $n \gtrsim d \log^2 d$ samples to learn the covariance spike u, but $n \gtrsim d^3$ samples to learn the cumulant spike; see Proposition 1 and Proposition 2 for the precise scaling of the learning rate δ. This result establishes that in isolation, learning from the higher-order cumulant has a much higher sample complexity than learning from the covariance.

Independent latent variables: In the mixed cumulant model with spikes in the covariance and the HOCs, Proposition 3 shows that if $n \leq d^3$, it is impossible to have weak recovery of the cumulant spike for any learning rate $\delta_d = o\left(1/d\right)$ *if* the latent variables λ and ν are independent – hence learning from HOCs remains hard. For larger learning rates $1/d \leq \delta_d = o(1)$, the SGD noise becomes dominant after d/δ^2 steps and our analysis works only up to that horizon; we discuss this in Section 3.2.

Correlated latent variables: If instead both spikes are present and their latent variables have a positive correlation $\mathbb{E}[\lambda^\mu \nu^\mu] > 0$ (fixed, independent of d), proposition 4 shows that $n \gtrsim d \log^2 d$ samples are *sufficient to weakly recover both spikes* with the optimal learning rate $\delta \approx 1/\log d$. Moreover, even for sub-optimal learning rates $\delta = o\left(1/d\right)$, the time to reach weak recovery of the cumulant spike is $d^2 \log^2 d$, which is still faster that in the uncorrelated case. Proposition 4 also shows that the speed-up in terms of sample complexity required for weak recovery happens even when the amount of correlation is small compared to the signal carried by the cumulant spike. In other words, it does not affect the minimum landscape of the loss: if the global minimum is close to the cumulant spike, introducing correlations between the latents will not move that minimum to the covariance spike.

We give a full summary of the known results and our contributions in Table 1. In the following, we give precise statements of our theorems in Section 3 and discuss our results in the context of the wider literature, and the recently discussed "staircase phenonmenon", in Section 4.

3. Rigorous analysis of the perceptron: the non-Gaussian information exponent

We now present a detailed analysis of the simplest model where correlated latents speed up learning from higher-order cumulants in the data, the spherical perceptron. Before stating the precise theorems, we present the main ingredients of the analysis informally.

The key idea of our analysis is borrowed from Ben Arous et al. (2021): during the *search phase* of SGD, while the overlaps $\alpha_u = u \cdot w, \alpha_v = v \cdot w$ of the weight w with the covariance and cumulant spikes are small, i.e. $o(1)$ with respect to d, the dynamics of spherical SGD is driven by the low-order terms of the polynomial expansion of the *population loss* $\mathcal{L}(w) = \mathbb{E}[\mathcal{L}(w, (x, y)]$. In our case of a mixture classification task, it is useful to rewrite the expectation using the *likelihood ratio* between the isotropic and the planted distributions $L(x) := {\mathrm{d}\mathbb{Q}_{\text{plant}}}/{\mathrm{d}\mathbb{Q}_0}(x)$ such that all averages are taken with respect to the simple, isotropic Gaussian distribution:

$$\mathcal{L}(w) = 1 + \frac{1}{2}\mathbb{E}_{\mathbb{Q}_0}[\sigma(w \cdot x)] - \frac{1}{2}\mathbb{E}_{\mathbb{Q}_0}\left[L(x)\sigma(w \cdot x)\right]. \tag{6}$$

We can then expand the loss in Hermite polynomials, which form an orthonormal basis w.r.t. the standard Gaussian distribution, and prove that the loss depends only on α_u, α_v and can be expanded in the following way:

$$\mathcal{L}(w) = \ell(\alpha_u, \alpha_v) = \sum_{i,j=0}^{\infty} c_{ij}^L c_{i+j}^\sigma \alpha_u^i \alpha_v^j, \tag{7}$$

see Lemma 7 in the appendix. The degree of the lowest order term in this expansion is called the *information exponent* k of the loss and it rigorously determines the duration of the search phase before SGD *weakly recovers* the key directions (Ben Arous et al., 2021; Dandi et al., 2023), in the sense that for any time-dependent overlap $\alpha(t)$, there exists an $\eta > 0$ such that defining $\tau_\eta := \min\left\{t \geq 0 \middle| \, |\alpha(t)| \geq \eta\right\}$, we have

$$\lim_{d\to\infty} \mathbb{P}\left(\tau_\eta \leq n\right) = 1. \tag{8}$$

Since the time in online SGD is equivalent to the number of samples, the information exponent governs the sample complexity for weak recovery. Here, we are interested in finding out how the correlation of the latent variables changes the information exponent, and which consequences this change has on the duration of search phase of projected SGD as $d \to \infty$. Considering only the terms that give relevant contribution for the search phase, Equation (7) applied to the MCM model with $\beta_m = 0$ becomes (see Appendix A.2.1 for more details):

$$\ell(\alpha_u, \alpha_v) = -\left(c_{20}\alpha_u^2 + c_{11}\alpha_u\alpha_v + c_{04}\alpha_v^4\right) \tag{9}$$

where $c_{20}, c_{04} > 0$ as long as $\beta_u, \beta_v > 0$, whereas $c_{11} > 0$ if and only if the latent variables are positively correlated: $\mathbb{E}[\lambda\nu] > 0$. Note that switching on this correlation does not change the overall information exponent, which is still 2, but the *mixed term* $\alpha_u\alpha_v$ strongly impacts the direction v, along which $\partial_v \ell = c_{11}\alpha_u + 4c_{04}\alpha_v^3$ has degree 3 if $c_{11} = 0$ whereas it has degree 1 in case of positive correlation. This means that *positive correlation of latent variables lowers the information exponent along the non-Gaussian direction*.

It is not straightforward to link these changes of the population-loss series expansion to actual changes in SGD dynamics. The randomness of each sample comes into play and it must be estimated thanks to a careful selection of the learning rate δ (see Proposition 3 and Proposition 4). However, at the risk of oversimplification, we could imagine that the dynamic is updated descending the spherical gradient of the population loss, leading to the following system of ODEs that approximately hold when $\alpha_u, \alpha_v \leq \eta$

$$\begin{cases} \dot{\alpha}_u(t) = 2c_{20}\alpha_u + c_{11}\alpha_v + O(\eta^2) \\ \dot{\alpha}_v(t) = c_{11}\alpha_u + 4c_{04}\alpha_v^3 - 2c_{20}\alpha_u^2\alpha_v + O(\eta^4) \end{cases} \tag{10}$$

where the last term in the second equation is due to the distorting effect of the spherical gradient.

We can see that the behaviour of α_u is not affected too much by the presence of correlation (the integral lines starting at $\alpha_u(0) \approx d^{-1/2}$ reach order 1 in $t = d\log^2 d$). On the contrary the dynamic of α_v changes completely: if $c_{11} > 0$, α_v will grow at the same pace as α_u, otherwise it will be orders of magnitude slower (there is even the possibility that

it will never escape 0, since the term $-2c_{20}\alpha_u^2\alpha_v$ could push it back too fast). Before stating the theorems that make these intuitions precise, we have to state our main assumptions on the activation function:

Assumption 1 (Assumption on the student activation function σ)**.** *We require that the activation function $\sigma \in \mathscr{C}^1(\mathbb{R})$ fulfils the following conditions:*

$$\mathop{\mathbb{E}}_{z\sim\mathcal{N}(0,1)}[\sigma(z)h_2(z)] > 0, \mathop{\mathbb{E}}_{z\sim\mathcal{N}(0,1)}[\sigma(z)h_4(z)] < 0,$$
$$\sup_{w\in\mathbb{S}^{d-1}} \mathbb{E}\left[\sigma'(w\cdot x)^4\right] \le C, \sup_{w\in\mathbb{S}^{d-1}} \mathbb{E}\left[\sigma'(w\cdot x)^{8+\iota}\right] \le C,$$
$$\text{and} \sum_{k=0} k \mathop{\mathbb{E}}_{z\sim\mathcal{N}(0,1)}[h_k(z)\sigma(z)] < \infty. \tag{11}$$

Here, $(h_i)_i$ are normalised Hermite polynomials (see Appendix A.1) and $\iota > 0$.

Note that smoothed versions of the ReLU activation function satisfy these conditions. We remark that all Equation (11) are important requirements, activation functions that do not satisfy one or more of these conditions could lead to different dynamics. On the other hand the assumption $\sigma \in \mathscr{C}^1(\mathbb{R})$ is done for convenience and differentiability a.e. should be enough; for instance all the simulations of the present work were done with the ReLU activation function.

3.1. Learning a single direction

To establish the baselines for weak recovery in single index models, where inputs carry only a covariance spike or a spike in the higher-order cumulants, we apply the results of Ben Arous et al. (2021) directly to find the following SGD learning timescales:

Proposition 1 (Covariance spike only)**.** *Considering only the spike in the covariance the generative distribution is $y \sim$ Rademacher $(1/2)$,*

$$y = 1 \;\Rightarrow\; x^\mu = \sqrt{\beta}_u \lambda^\mu u + z^\mu, \qquad \lambda^\mu \sim \mathcal{N}(0,1)$$
$$y = -1 \;\Rightarrow\; x^\mu = z^\mu, \qquad z^\mu \sim \mathcal{N}(0, \mathbb{1}_d)$$

A spherical perceptron that satisfies Assumption 1 and is trained on the correlation loss (5) using online SGD (4) has the following result concerning the overlap with the hidden direction $\alpha_{u,t} := u \cdot w_t$:

- *if $\frac{1}{\log^2 d} \ll \delta_d \ll \frac{1}{\log d}$ and $n \gg d\log^2 d$ then there is **strong recovery** and*

$$|\alpha_{u,t}| \to 1 \qquad \textit{in probability and in } \mathscr{L}_p \;\forall p \ge 1$$

- *if $\delta_d = O(1)$ and $n \le d\log d$ then it is impossible to learn anything:*

$$|\alpha_{u,t}| \to 0 \qquad \textit{in probability and in } \mathscr{L}_p \;\forall p \ge 1$$

Proposition 2 (Cumulant spike only)**.** *Considering only the spike in the cumulants, letting $(y^\mu)_\mu, (\nu^\mu)_\mu$ be i.i.d. Rademacher$(1/2)$, $(z^\mu)_\mu$ be i.i.d $N(0, \mathbb{1}_d)$ and S as in (2), the generative distribution is:*

$$y^\mu = 1 \;\Rightarrow\; x^\mu = S\left(\sqrt{\beta}_v \nu^\mu v + z^\mu\right)$$
$$y^\mu = -1 \;\Rightarrow\; x^\mu = z^\mu,$$

A spherical perceptron that satisfies Assumption 1 and is trained on the correlation loss (5) using online SGD (4) has the following result concerning the overlap with the hidden direction $\alpha_{v,t} := u \cdot v_t$:

- *if $\frac{1}{d^2\log^2 d} \ll \delta_d \ll \frac{1}{d\log d}$ and $n \gg d^3\log^2 d$ then there is **weak recovery** of the cumulant spike in the sense of Equation (8);*
- *if $\delta_d = o(\frac{1}{d})$ and $n \ll d^3$ then it is impossible to learn anything:*

$$|\alpha_{v,t}| \to 0 \qquad \textit{in probability and in } \mathscr{L}_p \;\forall p \ge 1$$

Note that the thresholds found in Proposition 2 coincide with the ones found for tensor-PCA for a order 4 tensor (see proposition 2.8 in (Ben Arous et al., 2021)), which is exactly the order of the first non trivial cumulant of $\mathbb{Q}_{\text{plant}}$ in the spiked-cumulant model. We will also see what happens in case of larger step size than what considered in Proposition 2.

3.2. Learning two directions

The following two propositions apply to a perceptron trained on the MCM model of Equation (1) with $\beta_m = 0$, $\nu^\mu \sim \mathrm{Radem}(1/2)$. We first state a negative result: in the case of *independent* latent variables, the cumulant direction cannot be learned faster than in the one dimensional case (Proposition 2).

Proposition 3. *Consider a spherical perceptron that satisfies Assumption 1 and is trained on the MCM (1) with correlation loss (5) using online SGD (4), with λ^μ independent of ν^μ. Let $\delta_d = o(1)$ be the step size. As long as $n \ll \min\left(\frac{d}{\delta_d^2}, d^3\right)$, we have that*

$$\lim_{d\to\infty} \sup_{t\le n} |\alpha_{v,t}| = 0$$

where the limit holds in $\mathscr{L}^p$ for every $p \ge 1$.

The condition $n \ll d/\delta_d^2$ is due to *sample complexity horizon* in online SGD. As already pointed out by Ben Arous et al. (2021), it turns out that after a certain number of steps, when $n\delta_d^2/d$ becomes large, the noise term in the equations dominates the drift term and the dynamics starts to become completely random (mathematically, Doob's inequality (30)

Model	Step size	Weak recovery u	Weak recovery v
Covariance spike only	$\delta = O(1)$, $\delta \approx \frac{1}{\log d}$	$n \ll d \log d$, $n \gg d \log^2 d$	not present
Cumulants spike only	$\delta = o\left(\frac{1}{d}\right)$, $\delta \approx \frac{1}{d}$	not present	$n \ll d^3$ $n \gg d^3 \log^2 d$
Two spikes, independent latents	$\delta \approx \frac{1}{\log d}$	$n \gg d \log^2 d$	$n \ll d \log^2 d$
	$\delta = o\left(\frac{1}{d}\right)$, $\delta \approx \frac{1}{d}$	$n \gg d^2 \log d$	$n \ll d^3$
Two spikes, correlated latents	$\delta \approx \frac{1}{\log d}$	$n \gg d \log^2 d$	$n \gg d \log^2 d$
	$\delta \approx \frac{1}{d}$	$n \gg d^2 \log d$	$n \gg d^2 \log d$

Table 1. **Summary of positive and negative results on learning from higher-order cumulants.** We summarise the sample complexities required to reach weak recovery of the covariance spike u and the cumulant spike v in the different settings discussed in Section 3. Negative results mean that weak recovery is impossible below these sample complexities at the given learning rates. Positive results guarantee weak recovery with at least as many samples as stated. The first two lines are direct applications of results from Ben Arous et al. (2021) for data models with a single spike. The mixed cumulant model with independent latent variables is a direct corollary of these results. The notation $\delta \approx f(d)$ for the positive results means that $\delta = o(f(d))$, but the best sample complexity is achieved when δ is as close as possible to its bound. Note that for independent latents with large learning rate $\delta \approx 1/\log d$, our techniques only allow studying a time horizons up to $t \approx d \log^2 d$, after which the randomness of the SGD updates becomes too large to be controlled, see Section 3.2 for more details.

fails to provide a useful estimate on the noise after this point). So our results can prove the absence of weak recovery only up to the *sample complexity horizon*; after that our definition of *weak recovery* is not useful anymore: due to the increased randomness, it is possible that the weight vector attains, by pure chance, a finite overlap with the spiked directions, but that would be forgotten very easily, not leading to meaningful learning. If instead the learning rate is small enough, $\delta_d = o(\frac{1}{d})$, then the horizon becomes large enough to include $n \ll d^3$, which is the same regime of Proposition 2.

Now we state the positive results: the covariance spike can be weakly recovered in the same sample complexity as the one-dimensional case (Proposition 1). However, if the latent variables have positive correlation, the same sample complexity is also sufficient to weakly recover the cumulant spike.

Proposition 4. *In the setting described, let the total number of samples be $n = \theta_d d$, with $\theta_d \gtrsim \log^2 d$ and growing at most polynomially in d. The step size $(\delta_d)_d$ is chosen to satisfy:*

$$\frac{1}{\theta_d} \ll \delta_d \ll \frac{1}{\sqrt{\theta_d}} \tag{12}$$

*Projected SGD reaches **weak recovery** of the covariance spike in the sense of Equation (8) in a time $\tau_u \leq n$. Moreover, if the latent variables have positive correlation: $\mathbb{E}[\lambda\nu] > 0$, conditioning on having matching signs at initialisation: $\alpha_{u,0}\alpha_{v,0} > 0$, **weak recovery is reached also for the cumulant spike v in a time $\tau_v \leq n$.***

The initialisation assumption $\alpha_{u,0}\alpha_{v,0} > 0$ means that the second part of Proposition 4 can be applied on half of the runs on average. This requirement could be fundamental: in case of mismatched initialisation, the correlation of the latent variables would push both $\alpha_{u,t}$ and $\alpha_{v,t}$ towards 0, slowing down the process. Note that the dependence of SGD dynamics on initialisation is a recurring phenomenon that arises with complex tasks, it is for example a well-known effect when training a neural network on inputs on a XOR-like Gaussian mixture task (Refinetti et al., 2021; Ben Arous et al., 2022). It is likely that this initialisation problem is specifically due to having a single neuron network. In a two-layer neural network, overparametrisation usually helps since it is sufficient that there is a neuron with the right initialisation signs to drive the learning in the right direction (Refinetti et al., 2021). For instance, the simulations we performed for Figure 1 with wide two-layer networks did not exhibit initialisation problems.

Remark 5. *Optimising in θ in Proposition 4, we get that if $\frac{1}{\log^2 d} \ll \delta_d \ll \frac{1}{\log d}$, then $n \gg d \log^2 d$ guarantees to have weak recovery. This optimal scaling of δ_d does not coincide with the best learning rate scaling for Proposition 3, since the sample complexity horizon is very short. However, by taking the slightly sub-optimal learning rate $\delta_d = \frac{1}{d \log d}$, we get that on one hand Proposition 3 applies and we can infer that, with independent latent variables, it takes at least d^3 samples/steps to learn direction v. On the other hand, by Proposition 4 we know that when the latent variables have (even a small) correlation, in $n \approx d^2 \log^2 d$ projected SGD attains weak recovery of both spikes.*

This completes our analysis of the search phase of SGD, which established rigorously how correlated latent variables speed up the weak recovery of the cumulant spike. We

finally note that this analysis can only be applied to the *search phase* of SGD. The expansion in Equation (7) breaks down as soon as α_u, α_v become macroscopic, since there are infinitely many terms that are not negligible along the non-Gaussian direction v in that case. Hence, even though the correlation between latent variables is key in the early stages of learning, it can become a sub-leading contribution in the later stages of SGD where the direction of weight updates is greatly influenced by terms due to HOCs. A thorough analysis of the descent phase after weak recovery will require different techniques: the SGD dynamical system involving the complete expression of the population loss should be studied, we leave this for future work.

However, the following simple example highlights the richness of the dynamics in this regime. Consider a sequence (β_u^m, β_v^m) such that $(\beta_u^0, \beta_v^0) = (1, 0)$ and $\lim_{m\to\infty}(\beta_u^m, \beta_v^m) = (0, 1)$. The global minimum of the population loss will move continuously from $w = u$ to $w = v$. However, looking at the dynamics of Equation (10), the change will be only at the level of the coefficients, and as long as none of them vanishes, the dynamics will be qualitatively the same. So changing the signal-to-noise ratios in this regime will affect only the the descent phase of the problem.

4. Discussion: hierarchies of learning with neural networks

4.1. Learning functions of increasing complexity and the staircase phenomenon

There are several well-known hierarchies that characterise learning in neural networks. In a seminal contribution, Saad & Solla (1995a;b) characterised a "specialisation" transition where two-layer neural networks with a few hidden neurons go from performing like an effective (generalised) linear model to a non-linear model during training with online SGD. Notably, this transition occurs after a long plateau in the generalisation error. A similar learning of functions of increasing complexity was shown experimentally in convolutional neural networks by Kalimeris et al. (2019).

More recently, Abbé et al. (2021; 2022) studied the problem of learning a target function over binary inputs that depend only on a small number of coordinates. They showed that two-layer neural networks can learn k-sparse functions (i.e. those functions that depend only on k coordinates) with sample complexity $n \gtrsim d$, rather than the $n \gtrsim d^k$ sample complexity of linear methods operating on a fixed feature space, if the target functions fulfil a (merged) *staircase* property. Abbé et al. (2023) extended this analysis to study the saddle-to-saddle dynamics of two-layer neural networks for specific target link functions. Similar saddle-to-saddle dynamics have been described by Jacot et al. (2021) in linear networks and by Boursier et al. (2022) for two-layer ReLU networks.

A detailed description of the staircase phenomenon when learning a multi-index target function over Gaussian inputs was given by Dandi et al. (2023), who studied feature learning via a few steps of single-pass, large-batch, gradient descent[1]. Their analysis showed how subsequent steps of gradient descent allow for learning new perpendicular directions of the target function if those directions are linearly connected to the previously learned directions. Bietti et al. (2023) also consider multi-index target function over Gaussian inputs, but instead look at the gradient flow for two-layer neural networks and provide a complete picture of the timescales of the ensuing saddle-to-saddle dynamics. Berthier et al. (2023) analyse the gradient flow of two-layer neural networks when both layers are trained simultaneously. They perform a perturbative expansion in the (small) learning rate of the second layer and find the timescales over which the neural network learns the Hermite coefficients of the target function sequentially.

4.1.1. Relation between the mixed cumulant and teacher-student models

Given these results, it is natural to ask how the staircases in teacher-student setups relate to the hierarchical learning in the mixed-cumulant model. A teacher model with three "spikes" (or teacher weight vectors) where each spike is learnt with a different sample complexity, analoguously to the uncorrelated MCM, is the function

$$y^*(x) = h_1(m \cdot x) + h_2(u \cdot x) + h_4(v \cdot x), \qquad (13)$$

where h_k is the kth Hermite polynomial (see Appendix A.1 for details) and the inputs x are drawn i.i.d. from the standard Gaussian distribution with identity covariance. In the language of Ben Arous et al. (2021), a single-index model with activation h_k has information exponent k, so taken individually, the three target functions in Equation (13) with directions m, u and v would be learnt with $n \gtrsim d, d\log^2 d$, and d^3 samples by a spherical perceptron using online SGD, exactly as in our mixed cumulant model with independent latent variables. We show the sequential learning of the different directions of such a teacher for a two-layer neural network in Figure 2A. In particular, we plot the test error with respect to the teacher $y^*(x)$ (13) for a two-layer network trained directly on the teacher (red), and for two-layer networks trained on only the first / the first two Hermite polynomials of Figure 2 (blue and green, respectively). Their performance suggests that the three directions m, v, u are learnt sequentially, which is further supported by showing

[1]Note that a single step of gradient descent was studied in a similar setting by Ba et al. (2022); Damian et al. (2022) without analysing the staircase phenomenon.

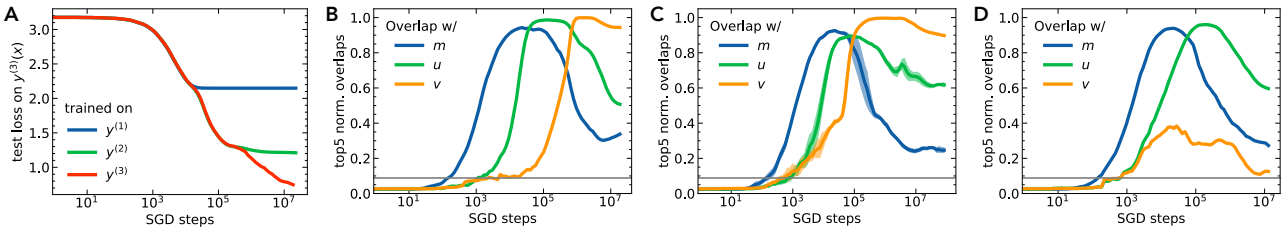

Figure 2. **Staircases in the teacher-student setup. A** Test accuracy of the same two-layer neural networks as in Figure 1 evaluated on the degree-4 target function $y^*(x)$ (13) during training on the target functions $y^{(1)}(x) = h_1(m \cdot x)$ (blue), $y^{(2)}(x) = h_1(m \cdot x) + h_2(u \cdot x)$ (green), and the teacher function Equation (13) (red). Inputs are drawn from the standard multivariate Gaussian distribution. **B-D** We show the the average of the top-5 largest normalised overlaps $w_k \cdot u$ of the weights of the kth hidden neuron w_k and the three directions that need to be learnt for three different target functions: the teacher function in Equation (13) (**B**), the same teacher with inputs that have a covariance $\mathbb{1} + uv^\top + vu^\top$ (**C**), and a teacher with mixed terms, Equation (14) (**D**). The dashed black line is at $d^{-1/2}$, the threshold for weak recovery. *Parameters:* Simulation parameters as in Figure 1: $d = 128, m = 512$ hidden neurons, ReLU activation function. Full details in Appendix B.1.

the maximum normalised overlap of the hidden neurons with the different spikes shown in Figure 2B.

How can we speed up learning of the direction u with information exponent 4 in a way that is analogous to the correlated latents? The latent variables of the MCM correspond to the pre-activations of the target function in the sense that both are low-dimensional projections of the inputs that determine the label. In the standard teacher-student setup, where inputs are drawn i.i.d. from a multivariate normal distribution with identity covariance, and target functions are of two-layer networks of the form $y^*(x) = \sum_k v_k \sigma_k(u_k \cdot x)$, it is not possible to have correlated pre-activations. Instead, we can speed up learning of the direction v by introducing a mixed term in the target function,

$$y^*(x) = h_1(m \cdot x) + h_1(u \cdot x) h_1(v \cdot x) + h_2(u \cdot x) + h_4(v \cdot x), \tag{14}$$

as is apparent from the overlap plot in Figure 2C. Dandi et al. (2023) and Bietti & Bach (2021) discuss the speed-up of learning this type of target function in the one-step and gradient flow frameworks, respectively, whereas our analysis focuses on online stochastic gradient descent with non-Gaussian inputs. An alternative for correlating the pre-activations is to keep the target function Equation (13) while drawing inputs from a normal distribution with covariance $\mathbb{1} + uv^\top$, and we see in Figure 2D that it does indeed speed up learning of the cumulant direction. This result is similar to the setting of Mousavi-Hosseini et al. (2024), who found that "spiking" of the input covariance can speed up learning a single-index model with a two-layer network.

So in summary, we see that the accelerated learning requires fine-tuning between the target function and the input structure. The mixed cumulant model does not require an explicit target function and instead directly highlights the importance of the correlation in the latent variables of the inputs to distinguish different classes of inputs.

4.2. The role of Gaussian fluctuations with correlated latents

Correlation between latent variables changes the covariance of the inputs compared to independent latents. A quick calculation shows that the covariance of the inputs under the planted distribution $C_{uv} \equiv \underset{\mathbb{Q}_{\text{plant}}}{\text{cov}}(x, x)$ becomes

$$C_{uv} = \mathbb{1} + \beta_u u u^\top + \sqrt{\frac{\beta_u \beta_v}{1 + \beta_v}}\, \mathbb{E}\, \lambda \nu \left(uv^\top + vu^\top\right). \tag{15}$$

In other words, we can weakly recover the cumulant spike by computing the leading eigenvector of the covariance of the inputs. Does that mean that the speed-up in learning due to correlated latents is simply due to the amount of information about the cumulant spike v that is exposed in the covariance matrix?

We test this hypothesis by taking the two-layer neural networks trained on the full MCM model and evaluating them on an equivalent Gaussian model, where we replace inputs from the planted distribution $\mathbb{Q}_{\text{plant}}$ of the MCM with samples from a multivariate normal distribution with mean $\beta_m u_m$ and covariance C_{uv}. We plot the corresponding test losses for independent and correlated latents in orange in Figure 1. When latent variables are independent, the Gaussian approximation breaks down after after a long plateau at $\approx 10^7$ steps, precisely when the network discovers the non-Gaussian fluctuations due to the cumulant spike. For correlated latent variables on the other hand, the Gaussian approximation of the data only holds for $\approx 10^4$ steps. This suggests that the presence of the cumulant spike in the covariance gives some of the neurons a finite overlap $w_k \cdot v$ with the cumulant spike, and this initial overlap speeds up the recovery of the cumulant spike using information from the higher-order cumulants of the data which is inaccessible at those time scales when latent variables are

uncorrelated. The presence of correlated latent variables therefore genuinely accelerates the learning from higher-order, non-Gaussian correlations in the inputs.

4.3. Further related work

Learning polynomials of increasing degree with kernel methods A series of works analysing the performance of kernel methods (Dietrich et al., 1999; Ghorbani et al., 2019; 2020; Bordelon et al., 2020; Spigler et al., 2020; Xiao et al., 2022; Cui et al., 2023) revealed that kernels (or linearised neural networks in the "lazy" regime (Li & Yuan, 2017; Jacot et al., 2018; Arora et al., 2019; Li & Liang, 2018; Chizat et al., 2019)) require $n \gtrsim d^\ell$ samples to learn the ℓth Hermite polynomial approximation of the target function. Here, we focus instead on the feature learning regime where neural networks can access features from higher-order cumulants much faster.

Tensor PCA with side information From the perspective of unsupervised learning, the MCM model is closely related to models studied in random matrix theory. When $\beta_m = \beta_\nu = 0$, the data distribution boils down to the well known Spiked Wishart model, that exhibits the *BBP phase transition* (Baik et al., 2005), which predicts thresholds on β_c for detection, at linear sample complexity $n \approx d$. If instead only $\beta_\nu \neq 0$, the MCM model corresponds to a tensor PCA problem, where one seeks to recover a signal $\xi \in \mathbb{R}^d$ from a noisy order-p tensor $T = \xi^{\otimes p} + \Delta$ with suitable noise tensor Δ (that in most of the cases is assumed to be Gaussian distributed). Indeed, in the MCM model the HOCs spike v could be retrieved by performing tensor PCA on the empirical kurtosis tensor.

The notion of correlated latent variables that we consider in the present work is reminiscent to the concept of *side information* that Richard & Montanari (2014) considered for tensor PCA. The side information would be an additional source of information on the spike via a Gaussian channel, $y = \gamma\xi + g$, where g is a Gaussian noise vector and $\gamma > 0$; using this estimate as an initialisation for AMP leads to a huge improvement in estimation. This joint model can be seen as a rank-1 version of a topic modelling method analysed by Anandkumar et al. (2014). In a similar vein, Sarao Mannelli et al. (2020) introduced the spiked matrix-tensor model, in which the statistician tries to recover ξ via the observation of a spiked matrix $M \propto \xi\xi^\top + \Delta^{(1)}$ and an order-p tensor $T \propto \xi^{\otimes p} + \Delta^{(2)}$ with appropriately scaled noise matrix / tensor $\Delta^{(1)}$ and $\Delta^{(2)}$, respectively. They analyse the optimisation landscape of the problem and the performance of the Langevin algorithm (Sarao Mannelli et al., 2020) and of gradient descent (Sarao Mannelli et al., 2019b;a). The main difference to our model is that these works consider recovering a *single* direction that spikes both the matrix and the tensor; we consider the case where two orthogonal directions are encoded as principal components of the covariance matrix and the higher-order cumulant tensors of the data.

5. Concluding perspectives

To achieve good performance, neural networks need to unwrap the higher-order correlations of their training data to extract features that are pertinent for a given task. We have shown that neural networks exploit correlations between the latent variables corresponding to these directions to speed up learning. In particular, our analysis of the spherical perceptron showed that correlations between the latent variables corresponding to the directions of two cumulants of order p and $q > p$, respectively, will speed up the learning of the direction from the qth cumulant by lowering its information exponent.

Our results open up several research directions. First, it will be intriguing to extend our analysis from single-index to shallow models. For two-layer neural networks, a key difficulty in applying mean-field techniques to analyse the impact of data structure on the dynamics of learning is the breakdown of the Gaussian equivalence principle (Mei & Montanari, 2022; Goldt et al., 2020; 2022; Gerace et al., 2020; Hu & Lu, 2022; Pesce et al., 2023) that we discussed in Section 4.2. In the presence of non-Gaussian pre-activations $w_k \cdot x$, it is not immediately clear what the right order parameters are that completely capture the dynamics. An analysis of the dynamics in the one-step framework of Ba et al. (2022); Dandi et al. (2023) is another promising approach to tackle the dynamics of two-layer networks. Other intriguing directions include determining the hardness of learning from higher-order cumulants when re-using data (Dandi et al., 2024) and from the perspective of generative exponents (Damian et al., 2024). For deeper neural networks, it is important to investigate the role of different layers in processing the HOCs of the data (even numerically), for example along the lines of Fischer et al. (2022). Finally, it will be intriguing to investigate how correlated latent variables emerge both in real data and in (hierarchical) models of synthetic data.

Acknowledgements

We thank Joan Bruna and Loucas Pillaud-Vivien for valuable discussions. SG acknowledges co-funding from Next Generation EU, in the context of the National Recovery and Resilience Plan, Investment PE1 – Project FAIR "Future Artificial Intelligence Research".

Code availability

We provide code to reproduce our experiments via GitHub at https://github.com/anon/correlatedlatents.

Impact statement

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

References

Abbé, E., Boix-Adsera, E., Brennan, M. S., Bresler, G., and Nagaraj, D. The staircase property: How hierarchical structure can guide deep learning. *Advances in Neural Information Processing Systems*, 34:26989–27002, 2021.

Abbé, E., Adsera, E. B., and Misiakiewicz, T. The merged-staircase property: a necessary and nearly sufficient condition for sgd learning of sparse functions on two-layer neural networks. In *Conference on Learning Theory*, pp. 4782–4887. PMLR, 2022.

Abbé, E., Adserà, E. B., and Misiakiewicz, T. Sgd learning on neural networks: leap complexity and saddle-to-saddle dynamics. In Neu, G. and Rosasco, L. (eds.), *Proceedings of Thirty Sixth Conference on Learning Theory*, volume 195 of *Proceedings of Machine Learning Research*, pp. 2552–2623. PMLR, 2023. URL https://proceedings.mlr.press/v195/abbe23a.html.

Anandkumar, A., Ge, R., Hsu, D., Kakade, S. M., and Telgarsky, M. Tensor decompositions for learning latent variable models. *Journal of machine learning research*, 15:2773–2832, 2014.

Arora, S., Du, S. S., Hu, W., Li, Z., Salakhutdinov, R. R., and Wang, R. On exact computation with an infinitely wide neural net. *Advances in neural information processing systems*, 32, 2019.

Ba, J., Erdogdu, M. A., Suzuki, T., Wang, Z., Wu, D., and Yang, G. High-dimensional asymptotics of feature learning: How one gradient step improves the representation. *Advances in Neural Information Processing Systems*, 35: 37932–37946, 2022.

Baik, J., Arous, G. B., and Péché, S. Phase transition of the largest eigenvalue for nonnull complex sample covariance matrices. *The Annals of Probability*, 33(5):1643 – 1697, 2005. doi: 10.1214/009117905000000233. URL https://doi.org/10.1214/009117905000000233.

Barak, B., Hopkins, S., Kelner, J., Kothari, P. K., Moitra, A., and Potechin, A. A nearly tight sum-of-squares lower bound for the planted clique problem. *SIAM Journal on Computing*, 48:687–735, 2019.

Belrose, N., Pope, Q., Quirke, L., Mallen, A., and Fern, X. Neural networks learn statistics of increasing complexity. *arXiv:2402.04362*, 2024.

Ben Arous, G., Gheissari, R., and Jagannath, A. Online stochastic gradient descent on non-convex losses from high-dimensional inference. *J. Mach. Learn. Res.*, 22(1), 2021.

Ben Arous, G., Gheissari, R., and Jagannath, A. High-dimensional limit theorems for sgd: Effective dynamics and critical scaling. In Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., and Oh, A. (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 25349–25362. Curran Associates, Inc., 2022.

Berthier, R., Montanari, A., and Zhou, K. Learning time-scales in two-layers neural networks. *arXiv preprint arXiv:2303.00055*, 2023.

Bietti, A. and Bach, F. Deep equals shallow for reLU networks in kernel regimes. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=aDjoksTpXOP.

Bietti, A., Bruna, J., and Pillaud-Vivien, L. On learning gaussian multi-index models with gradient flow. *arXiv preprint arXiv:2310.19793*, 2023.

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

Boursier, E., Pillaud-Vivien, L., and Flammarion, N. Gradient flow dynamics of shallow relu networks for square loss and orthogonal inputs. *Advances in Neural Information Processing Systems*, 35:20105–20118, 2022.

Chizat, L., Oyallon, E., and Bach, F. On lazy training in differentiable programming. In Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/ae614c557843b1df326cb29c57225459-Paper.pdf.

Cui, H., Krzakala, F., and Zdeborová, L. Optimal learning of deep random networks of extensive-width. *arXiv:2302.00375*, 2023.

Damian, A., Lee, J., and Soltanolkotabi, M. Neural networks can learn representations with gradient descent. In *Conference on Learning Theory*, pp. 5413–5452. PMLR, 2022.

Damian, A., Nichani, E., Ge, R., and Lee, J. D. Smoothing the landscape boosts the signal for sgd: Optimal sample complexity for learning single index models. *arXiv:2305.10633*, 2023.

Damian, A., Pillaud-Vivien, L., Lee, J. D., and Bruna, J. The computational complexity of learning gaussian single-index models. *arXiv preprint arXiv:2403.05529*, 2024.

Dandi, Y., Krzakala, F., Loureiro, B., Pesce, L., and Stephan, L. Learning two-layer neural networks, one (giant) step at a time. *arXiv:2305.18270*, 2023.

Dandi, Y., Troiani, E., Arnaboldi, L., Pesce, L., Zdeborová, L., and Krzakala, F. The benefits of reusing batches for gradient descent in two-layer networks: Breaking the curse of information and leap exponents. *arXiv preprint arXiv:2402.03220*, 2024.

Dietrich, R., Opper, M., and Sompolinsky, H. Statistical mechanics of support vector networks. *Physical review letters*, 82(14):2975, 1999.

Fischer, K., René, A., Keup, C., Layer, M., Dahmen, D., and Helias, M. Decomposing neural networks as mappings of correlation functions. *Physical Review Research*, 4(4): 043143, 2022.

Gerace, F., Loureiro, B., Krzakala, F., Mézard, M., and Zdeborová, L. Generalisation error in learning with random features and the hidden manifold model. In *International Conference on Machine Learning*, pp. 3452–3462. PMLR, 2020.

Ghorbani, B., Mei, S., Misiakiewicz, T., and Montanari, A. Limitations of lazy training of two-layers neural network. In *Advances in Neural Information Processing Systems*, volume 32, pp. 9111–9121, 2019.

Ghorbani, B., Mei, S., Misiakiewicz, T., and Montanari, A. When do neural networks outperform kernel methods? In *Advances in Neural Information Processing Systems*, volume 33, 2020.

Goldt, S., Mézard, M., Krzakala, F., and Zdeborová, L. Modeling the influence of data structure on learning in neural networks: The hidden manifold model. *Physical Review X*, 10(4):041044, 2020.

Goldt, S., Loureiro, B., Reeves, G., Krzakala, F., Mézard, M., and Zdeborová, L. The gaussian equivalence of generative models for learning with shallow neural networks. In *Mathematical and Scientific Machine Learning*, pp. 426–471. PMLR, 2022.

Hopkins, S. *Statistical inference and the sum of squares method*. PhD thesis, Cornell University, 2018.

Hopkins, S. B. and Steurer, D. Bayesian estimation from few samples: com- munity detection and related problems. *arXiv:1710.00264*, 2017.

Hopkins, S. B., Kothari, P. K., Potechin, A., Raghavendra, P., Schramm, T., and Steurer, D. The power of sum-of-squares for detecting hidden structures. In *IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*, pp. 720—-731, 2017.

Hu, H. and Lu, Y. M. Universality laws for high-dimensional learning with random features. *IEEE Transactions on Information Theory*, 69(3):1932–1964, 2022.

Ingrosso, A. and Goldt, S. Data-driven emergence of convolutional structure in neural networks. *Proceedings of the National Academy of Sciences*, 119(40):e2201854119, 2022.

Jacot, A., Gabriel, F., and Hongler, C. Neural tangent kernel: Convergence and generalization in neural networks. In Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper_files/paper/2018/file/5a4be1fa34e62bb8a6ec6b91d2462f5a-Paper.pdf.

Jacot, A., Ged, F., Şimşek, B., Hongler, C., and Gabriel, F. Saddle-to-saddle dynamics in deep linear networks: Small initialization training, symmetry, and sparsity. *arXiv preprint arXiv:2106.15933*, 2021.

Kalimeris, D., Kaplun, G., Nakkiran, P., Edelman, B., Yang, T., Barak, B., and Zhang, H. Sgd on neural networks learns functions of increasing complexity. *Advances in neural information processing systems*, 32, 2019.

Kunisky, D., Wein, A. S., and Bandeira, A. S. Notes on computational hardness of hypothesis testing: Predictions using the low-degree likelihood ratio. In *ISAAC Congress (International Society for Analysis, its Applications and Computation)*, pp. 1–50. Springer, 2019.

Li, Y. and Liang, Y. Learning Overparameterized Neural Networks via Stochastic Gradient Descent on Structured Data. In *Advances in Neural Information Processing Systems 31*, 2018.

Li, Y. and Yuan, Y. Convergence analysis of two-layer neural networks with relu activation. *Advances in neural information processing systems*, 30, 2017.

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

Merger, C., René, A., Fischer, K., Bouss, P., Nestler, S., Dahmen, D., Honerkamp, C., and Helias, M. Learning interacting theories from data. *Phys. Rev. X*, 13:041033, 2023. doi: 10.1103/PhysRevX.13.041033. URL https://link.aps.org/doi/10.1103/PhysRevX.13.041033.

Mousavi-Hosseini, A., Wu, D., Suzuki, T., and Erdogdu, M. A. Gradient-based feature learning under structured data. *Advances in Neural Information Processing Systems*, 36, 2024.

Nestler, S., Helias, M., and Gilson, M. Statistical temporal pattern extraction by neuronal architecture. *Physical Review Research*, 5(3):033177, 2023.

Pesce, L., Krzakala, F., Loureiro, B., and Stephan, L. Are gaussian data all you need? extents and limits of universality in high-dimensional generalized linear estimation. *arXiv:2302.08923*, 2023.

Refinetti, M., Goldt, S., Krzakala, F., and Zdeborová, L. Classifying high-dimensional gaussian mixtures: Where kernel methods fail and neural networks succeed. In *International Conference on Machine Learning*, pp. 8936–8947. PMLR, 2021.

Refinetti, M., Ingrosso, A., and Goldt, S. Neural networks trained with sgd learn distributions of increasing complexity. In *International Conference on Machine Learning*, pp. 28843–28863. PMLR, 2023.

Richard, E. and Montanari, A. A statistical model for tensor pca. *Advances in neural information processing systems*, 27, 2014.

Riegler, P. and Biehl, M. On-line backpropagation in two-layered neural networks. *Journal of Physics A: Mathematical and General*, 28(20):L507, 1995.

Saad, D. and Solla, S. Exact Solution for On-Line Learning in Multilayer Neural Networks. *Phys. Rev. Lett.*, 74(21): 4337–4340, 1995a.

Saad, D. and Solla, S. On-line learning in soft committee machines. *Phys. Rev. E*, 52(4):4225–4243, 1995b.

Sarao Mannelli, S., Biroli, G., Cammarota, C., Krzakala, F., and Zdeborová, L. Who is afraid of big bad minima? analysis of gradient-flow in spiked matrix-tensor models. *Advances in Neural Information Processing Systems*, 32, 2019a.

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

Sarao Mannelli, S., Biroli, G., Cammarota, C., Krzakala, F., Urbani, P., and Zdeborová, L. Marvels and pitfalls of the langevin algorithm in noisy high-dimensional inference. *Physical Review X*, 10(1):011057, 2020.

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

Székely, E., Bardone, L., Gerace, F., and Goldt, S. Learning from higher-order statistics, efficiently: hypothesis tests, random features, and neural networks. *arXiv2312.14922*, 2023.

Xiao, L., Hu, H., Misiakiewicz, T., Lu, Y., and Pennington, J. Precise learning curves and higher-order scalings for dot-product kernel regression. *Advances in Neural Information Processing Systems*, 35:4558–4570, 2022.

A. Dynamical analysis

A.1. Expansions with Hermite polynomials

In this section we will present how expansion formulas like (7) can be obtained. The idea is that, thanks to the fact that the null hypothesis distribution $\mathbb{Q}_0$ is a standard Gaussian, and that the planted distribution $\mathbb{Q}_{\text{plant}}$ is still quite close to a standard Gaussian, with signal on just 1 or 2 directions, it is possible to use *Hermite polynomials* to expand in polynomial series all the functions of interest.

This use of *Hermite polynomials* is, at this point, well known, and we refer for instance to (Székely et al., 2023), (Kunisky et al., 2019), (Dandi et al., 2023) for more details on this kind of application of Hermite polynomials. We will just recall the properties that we need:

- $(h_n)_{n\in\mathbb{N}}$ is a family of polynomials in which h_k has degree k.
- they form an *orthonormal* basis for $\mathscr{L}^2(\mathbb{R}, \mathcal{N}(0,1))$, which is the Hilbert space of square integrable function with the product:
$$\langle h_i, h_j\rangle = \mathbb{E}_{z\sim\mathcal{N}(0,1)}\left[h_i(z)h_j(z)\right] = \delta_{ij}$$
- this can be generalised to higher dimensions. $(H_{\boldsymbol{\alpha}})_{\boldsymbol{\alpha}\in\mathbb{N}^d}$ such that:
$$H_{\boldsymbol{\alpha}}(x_1,\dots,x_d) = \prod_{i=1}^{d} h_{\boldsymbol{\alpha}_i}(x_i)$$
form an orthonormal basis for the space $\mathscr{L}^2\left(\mathbb{R}^d, \mathcal{N}\left(0, \mathbb{1}_{d\times d}\right)\right)$

Moreover, it will be useful the following rewriting in our notation of lemma 1 in (Dandi et al., 2023):

Lemma 6. *Suppose $g \in \mathscr{L}^2(\mathbb{R}, \mathcal{N}(0,1))$ and $w \in \mathbb{R}$, with $||w|| = 1$, then $f : \mathbb{R}^d \to \mathbb{R}$ defined as $f(x) := g(w\cdot x)$ belongs to $\mathscr{L}^2(\mathbb{R}^d, \mathcal{N}(0, \mathbb{1}_d))$ and has Hermite coefficients:*

$$C^f_{\boldsymbol{\alpha}} := \mathop{\mathbb{E}}_{z\sim\mathcal{N}(0,\mathbb{1}_d)}\left[H_{\boldsymbol{\alpha}}(z)f(z)\right] = \mathop{\mathbb{E}}_{z\sim\mathcal{N}(0,1)}\left[h_{|\boldsymbol{\alpha}|}(z)g(z)\right]\prod_{i=1}^{d} w_i^{\boldsymbol{\alpha}_i}, \quad \boldsymbol{\alpha}\in\mathbb{N}^d \tag{16}$$

Suppose now $g \in \mathscr{L}^2(\mathbb{R}^2, \mathcal{N}(0, \mathbb{1}_2))$ and $u, v \in \mathbb{R}$, with $||u|| = ||v|| = 1$ and $u\cdot v$, then $f : \mathbb{R}^d \to \mathbb{R}$ defined as $f(x) := g(u\cdot x, v\cdot x)$ belongs to $\mathscr{L}^2(\mathbb{R}^d, \mathcal{N}(0, \mathbb{1}_d))$ and has Hermite coefficients:

$$C^f_{\boldsymbol{\alpha}} = \sum_{i+j=|\boldsymbol{\alpha}|} \mathop{\mathbb{E}}_{z\sim\mathcal{N}(0,\mathbb{1}_2)}\left[H_{(i,j)}(z)g(z)\right]\prod_{m=1}^{d}\left(\sum_{\substack{i_m+j_m=\boldsymbol{\alpha}_m\\ \sum_m i_m=i, \sum_m j_m=j}} u_m^{i_m} v_m^{j_m}\right) \tag{17}$$

First we will present the expansion procedure informally, and then in Lemma 7 we will see the exact statement that will be used in the subsequent proofs.

Assume the learning algorithm is a perceptron $f(w,x) = \sigma(w\cdot x)$, it can be expanded in Hermite basis with appropriate coefficients:

$$f(w,x) = \sum_{k=0}^{\infty} c_k^{\sigma} h_k(w\cdot x)$$

where

$$c_k^{\sigma} = \mathop{\mathbb{E}}_{z\sim\mathcal{N}(0,1)}\left[\sigma(z)h_k(z)\right] \tag{18}$$

Suppose that the data distribution is the MCM as described in Section 2 (for simplicity we will assume that $\beta_m = 0$; there are only the covariance and HOCs spikes). To be able to carry on the expansion it is useful to consider (as done in (Kunisky et al., 2019) and (Székely et al., 2023)) the *likelihood ratio* of $\mathbb{Q}_{\text{plant}}$ with respect to $\mathbb{Q}_0$:

$$L(x) := \frac{\mathrm{d}\mathbb{Q}_{\text{plant}}(x)}{\mathrm{d}\mathbb{Q}_0(x)} \tag{19}$$

Note that since $\mathbb{Q}_{\text{plant}}$ is a perturbation of a standard Gaussian only along the directions u, v; then the likelihood ratio L will depend only on the projections $y_u := x \cdot u, y_v = x \cdot v$. Assuming u, v to be orthogonal and the non Gaussianity ν to have a distribution that belongs to $\mathscr{L}^2(\mathbb{R}, \mathcal{N}(0,1))$, we can have an expansion in Hermite basis:

$$L(x) = \sum_{i,j=0}^{\infty} c_{ij}^L h_i(y_u) h_j(y_v)$$

where

$$c_{ij}^L = \mathop{\mathbb{E}}_{(z_1,z_2)\sim\mathcal{N}(0,\mathbb{1}_{2\times 2})} [L(u \cdot z_1 + v \cdot z_2) h_i(z_1) h_j(z_2)] \tag{20}$$

So assuming to use a *correlation loss*:

$$\mathcal{L}(w, (x, y)) = 1 - y f(w, x)$$

The population loss is:

$$\begin{aligned}
\mathcal{L}(w) &= \mathbb{E}_{x,y}[\mathcal{L}(w, (x, y)] = 1 + \frac{1}{2}\mathbb{E}_{\mathbb{Q}_0}[\sigma(w \cdot x)] - \frac{1}{2}\mathbb{E}_{\mathbb{Q}_0}[L(x)\sigma(w \cdot x)] \\
&= \mathbb{E}_{x,y}[\mathcal{L}(w, (x, y)] = 1 + \frac{1}{2}\mathop{\mathbb{E}}_{z\sim\mathcal{N}(0,1)}[\sigma(z)] - \frac{1}{2}\mathbb{E}_{\mathbb{Q}_0}[L(x)\sigma(w \cdot x)] \\
&= 1 + \frac{c_0^\sigma}{2} - \frac{1}{2}\mathbb{E}_{\mathbb{Q}_0}\left[\left(\sum_{i,j=0}^{\infty} c_{ij}^L h_i(x \cdot u) h_j(x \cdot v)\right)\left(\sum_{k=0}^{\infty} c_k^\sigma h_k(w \cdot x)\right)\right]
\end{aligned} \tag{21}$$

The following lemma deals with the exchanging the integral and the series.

Lemma 7. *Let $\mathbb{P}$ be such that the likelihood ratio is square integrable $L \in \mathscr{L}^2\left(\mathbb{R}^d, \mathbb{Q}_d\right)$ and depends on x only through its projection along 2 orthogonal directions u, v:*

$$L(x) = l(u \cdot x, v \cdot x) = \sum_{i,j=0}^{\infty} c_{ij}^L h_i(x \cdot u) h_j(x \cdot v) \tag{22}$$

Suppose $\sigma \in \mathscr{L}^2(\mathbb{R}, \mathbb{Q}) \cap \mathscr{C}^1(\mathbb{R})$, with derivative $\sigma' \in \mathscr{L}^2(\mathbb{R}, \mathbb{Q}) \cap \mathscr{C}^0(\mathbb{R})$, and expansions

$$\sigma(y) = \sum_{k=0}^{\infty} c_k^\sigma h_k(y) \tag{23}$$

then, defining $\alpha_u := u \cdot w$ and $\alpha_v := v \cdot w$, the following identity holds

$$\mathbb{E}_{\mathbb{Q}_0}[L(x)\sigma(w \cdot x)] = \sum_{i,j}^{\infty} c_{ij}^L c_{i+j}^\sigma \alpha_u^i \alpha_v^j \tag{24}$$

Moreover, if $\sum_k k c_k^\sigma < \infty$, the population loss $\mathcal{L}(w) \in \mathscr{C}^1(\mathcal{C}, \mathbb{R})$, where $\mathcal{C} := \left\{w \in \mathbb{S}^{d-1} | \alpha_u < \frac{1}{2}, \alpha_v < \frac{1}{2}\right\}$ and it is also possible to switch expectations and derivatives, to get the expression:

$$\begin{aligned}
\nabla\mathcal{L}(w) &= \nabla\mathbb{E}_{\mathbb{Q}}[L(x)\mathcal{L}(w, (x, y))] = \mathbb{E}_{\mathbb{Q}}[L(x)\nabla\mathcal{L}(w, (x, y))] = \\
&= -\frac{1}{2}\left[\left(\sum_{i=1}^{\infty} i c_{i0}^L c_i^\sigma \alpha_u^{i-1}\right) u + \left(\sum_{j=1}^{\infty} j c_{0j}^L c_j^\sigma \alpha_v^{j-1}\right) v + \left(\sum_{i,j=1}^{\infty} c_{ij}^L c_{i+j}^\sigma \alpha_u^{i-1} \alpha_v^{j-1} (i\alpha_v u + j\alpha_u v)\right)\right]
\end{aligned} \tag{25}$$

Proof. First focus on (24). We know that both L and $\sigma(w \cdot _)$ belong to $\mathscr{L}^2\left(\mathbb{R}^d, \mathbb{Q}_0\right)$, hence we can see the integral in (24)

as the $\mathscr{L}^2$ inner product, and compute it using the series expansion and Lemma 6.

$$\begin{aligned}
\mathbb{E}_{\mathbb{Q}_0}\left[L(x)\sigma(w\cdot x)\right] &= \sum_{\boldsymbol{\alpha}\in\mathbb{N}^d} c^L_{\boldsymbol{\alpha}} c^{\sigma(w\cdot)}_{\boldsymbol{\alpha}} = \sum_{k=0}^{\infty}\sum_{|\boldsymbol{\alpha}|=k} c^L_{\boldsymbol{\alpha}} c^{\sigma}_{k} \prod_{l=1}^{d} w_l^{\boldsymbol{\alpha}_l} \\
&= \sum_{k}\sum_{i+j=k} c^L_{ij} c^{\sigma}_{k} \sum_{|\boldsymbol{\alpha}|=k}\prod_{m=1}^{d} w_m^{\boldsymbol{\alpha}_m} \left(\sum_{\substack{i_m+j_m=\boldsymbol{\alpha}_m \\ \sum_m i_m=i, \sum_m j_m=j}} u_m^{i_m} v_m^{j_m}\right) \\
&= \sum_{k}\sum_{i+j=k} c^L_{ij} c^{\sigma}_{k} \prod_{m=1}^{d} \left(\sum_{\substack{\sum_m i_m=i \\ \sum_m j_m=j}} (u_m w_m)^{i_m} (v_m w_m)^{j_m}\right) \\
&= \sum_{k}\sum_{i+j=k} c^L_{ij} c^{\sigma}_{k} \alpha_u^i \alpha_v^j
\end{aligned}$$

Moreover it can be verified directly that, under the assumption that $\sum_k k c_k^\sigma < \infty$, and $w \in \mathcal{C}$ the series in (25) converges uniformly, hence also the second part of the statement holds. □

A.2. Properties of the MCM model

In this section we will apply the Hermite expansion to the Mixed Cumulant Model and note the key properties that will allow to prove the Propositions of Section 3.

A.2.1. Hermite coefficients

Starting from the single direction models we have that:

- If $\beta_m = \beta_v = 0$ only the covariance spike survives and the model is Gaussian, hence the population loss is:

$$\mathcal{L}(w) = \ell(\alpha_u) = 1 - \frac{\beta_u c_2^\sigma}{4}\alpha_u^2 \tag{26}$$

- if $\beta_m = \beta_u = 0$ only the cumulants spike survives. To compute the Hermite coefficients we can rely on lemma 13 form (Székely et al., 2023) to get:

$$\mathcal{L}(w) = \ell(\alpha_v) = 1 - \frac{1}{2}\sum_{j\geq 3}\frac{c_j^\sigma}{j!}\left(\frac{\beta_v}{1+\beta_v}\right)^{j/2}\mathbb{E}[h_j(\nu)]\alpha_v^j \tag{27}$$

  So if $\nu =$Radem$(1/2)$ we have that the leading term is

$$\mathcal{L}(w) = \ell(\alpha_v) = 1 + \frac{c_4^\sigma}{4!}\left(\frac{\beta_v}{1+\beta_v}\right)^2\alpha_v^4 + o(\alpha_v^4)$$

  Note the change of sign, due to $\mathbb{E}[h_4(\nu)] = -2$. This is why we need to ask $c_4^\sigma < 0$ to have that the population loss is decreasing for small α_v.

- consider now the case with both spike and *independent* latent variables λ, ν. It is easy to see that, thanks to the independence, the coefficients split $c^L_{ij} = c^{cov}_i c^{cumulant}_j$, so the leading terms are:

$$\mathcal{L}(w) = \ell(\alpha_u, \alpha_v) = 1 - \frac{1}{2}\underbrace{\frac{\beta_u c_2^\sigma}{4}}_{c_{20}}\alpha_u^2 - \underbrace{\frac{-c_4^\sigma}{4!}\left(\frac{\beta_v}{1+\beta_v}\right)^2}_{c_{04}}\alpha_v^4 + o(\alpha_v^4) \tag{28}$$

- in case of latent variables with positive correlation, we need to add mixed terms, the leading one is $c_{11} = c_2^\sigma \sqrt{\frac{\beta_u \beta_v}{1+\beta_v}} \mathbb{E}[\lambda\nu]$. So the first few terms of the expansion of the population loss are:

$$\mathcal{L}(w) = \ell(\alpha_u, \alpha_v) = 1 - \frac{1}{2}\left(c_{20}\alpha_u^2 + c_{11}\alpha_u\alpha_v + c_{04}\alpha_v^4\right) + o(\alpha_u\alpha_v) \tag{29}$$

  note that the presence of $c_{11} > 0$ makes the term $c_{04}\alpha_v^4$ to become a sub-leading contribution, lowering the information exponent on the v direction.

A.2.2. Assumption on the sample wise error

An important quantity for proving the Propositions in Section 3 is the *sample-wise error*

$$H_d^\mu(w) := \mathcal{L}_d\left(w, (x^\mu, y^\mu)\right) - \mathcal{L}_d(w)$$

we will ask the same assumptions of (Ben Arous et al., 2021):

$$\sup_{w\in\mathbb{S}^{d-1}} \mathbb{E}\left[(\nabla_{sph} H_d(w)\cdot u)^2\right] \le C_1$$
$$\sup_{w\in\mathbb{S}^{d-1}} \mathbb{E}\left[(\nabla_{sph} H_d(w)\cdot v)^2\right] \le C_1$$
$$\sup_{w\in\mathbb{S}^{d-1}} \mathbb{E}\left[||\nabla_{sph} H_d(w)||^{4+\iota}\right] \le C_2 d^{4+\iota/2} \qquad \text{for some } \iota > 0$$

simple calculations ensure that it is sufficient to ask that

- $\mathbb{Q}_{\text{plant}}$ has finite moments up to the 8-th order,which is satisfied when taking $\nu \sim$Rademacher$(1/2)$.
- the following holds:

$$\sup_{w\in\mathbb{S}^{d-1}} \mathbb{E}\left[\sigma'(w\cdot x)^4\right] \le C$$
$$\sup_{w\in\mathbb{S}^{d-1}} \mathbb{E}\left[\sigma'(w\cdot x)^{8+2\iota}\right] \le C$$

  which we have assumed in (11)

A.3. Proof for the SGD analysis

Here we will provide proofs for the Propositions presented in Section 3. All of them rely heavily on the thorough analysis of spherical perceptron dynamics proved in (Ben Arous et al., 2021).

Note that Proposition 1 is a well known result, very close to the examples provided in section 2 of (Ben Arous et al., 2021), and we can verify that a straightforward application of theorems 1.3 and 1.4 from (Ben Arous et al., 2021), in the case $k = 2$, proves it.

A different situation happens for Proposition 2, the spiked cumulant model is quite recent so its SGD dynamics have still to be investigated. However it is very quick to verify that also in this case it is possible to apply theorems 1.3 and 1.4 from (Ben Arous et al., 2021). Assumption B in (Ben Arous et al., 2021) is met thanks to Appendix A.2.2. The monotonicity assumption could be harder to check, but since we are interested only in the search phase, it is sufficient to satisfy that $\mathcal{L}(w) = \ell(\alpha_v)$ is monotone only in $(0, \rho)$ for any $\rho > 0$ (assumption A_ρ introduced in section 3.1 of (Ben Arous et al., 2021)), which is clear from (27). Hence we can apply the case $k = 4$ of theorems 1.3 and 1.4 from (Ben Arous et al., 2021), to conclude the proof.

A.3.1. Proof of proposition 3

This proposition can be considered a Corollary of *Theorem 1.4* from (Ben Arous et al., 2021), the only differences is that the data distribution has an additional direction u and that we are considering also lower learning rates $\delta = o(1)$ instead of $\delta = o(\frac{1}{d})$ (note that at page 10 in (Ben Arous et al., 2021) this extension to larger learning rates is already mentioned).

On one hand we will verify that thanks to the fact that the auxiliary variable λ, ν are independent, then the additional direction u makes no impact in the dynamic of the overlap α_v. On the other hand, the condition $n \leq \frac{d}{\delta_d^2}$ ensures that we are always inside the time horizon considered in (Ben Arous et al., 2021). We will discuss how to deal with these two changes in the next two paragrphs.

Direction u can be neglected Since $\nabla_{sph}\mathcal{L}(w_t, (x^t, y^t))$ is orthogonal to w_t we have that $||\tilde{\alpha}_{v,t}|| = ||v \cdot \tilde{w}_t|| \geq 1$. Hence we have that it $\alpha_{v,t} > 0$, then

$$\begin{aligned}\alpha_{v,t} \leq \tilde{\alpha}_{v,t} &= \alpha_{v,t-1} - \frac{\delta}{d}\nabla_{sph}\mathcal{L}(w_{t-1}, (x^t, y^t)) \cdot v \\ &= \alpha_{v,t-1} - \frac{\delta}{d}\nabla_{sph}\mathcal{L}(w_{t-1}) \cdot v - \frac{\delta}{d}\nabla_{sph}H^t(w_{t-1}) \cdot v\end{aligned}$$

Since the sample-wise error satisfies the same properties required in (Ben Arous et al., 2021) (as checked in Appendix A.2.2) we just need to ensure that the drift term also can be handled in the same way as in (Ben Arous et al., 2021). There, the monotonicity assumption on the population loss allowed to estimate $-v \cdot \nabla_{sph}\mathcal{L}(w) \leq C\alpha_v^3$ for $\alpha_v > 0$. So the aim will be to get a similar bound. Note that we can assume α_v smaller than η, hence we can apply (28) and Lemma 7 to get:

$$\begin{aligned}-\nabla_{sph}\mathcal{L}(w) \cdot v &= ((v - \alpha_v w)^\top \nabla L(w) \\ &= \partial_v \ell - \alpha_v \left(\alpha_u \partial_u \ell + \alpha_v \partial_v \ell\right) \\ &= \ 4c_{04}\alpha_v^3 - c_{20}\alpha_u^2\alpha_v + o(\eta^3) \\ &\leq 5c_{04}\alpha_v^3\end{aligned}$$

We can now use this inequality to reach the same estimates as in p 37 of (Ben Arous et al., 2021), and carry on with their proof, in the case $\delta = o(\frac{1}{d})$. We only need to show that we can take larger step size, at the cost of taking $n \ll \frac{d}{\delta_d^2}$.

Larger step size We note that in the proof of *Theorem 1.4* from (Ben Arous et al., 2021) the main use of the assumptions $\delta = o(\frac{1}{d})$ was to ensure that $\frac{n}{d}\delta_d^2 = o(1)$ which is required to use the following form of Doob's maximal inequality for martingales:

$$\sup_{w_o \in \mathbb{S}^{d-1}} \mathbb{P}_{w_0}\left(\max_{t \leq n} \frac{\delta}{d}\left|\sum_{j=0}^{t-1} \nabla_{sph}H^{j+1}(w_j) \cdot u\right| \geq r\right) \leq \frac{2n\delta^2 C}{d^2 r^2}, \tag{30}$$

where $\mathbb{P}_{w_0}$ means that we are conditioning that the initial weight is w_0, C comes from Appendix A.2.2 and $r > 0$ is a free parameter. To carry out the proof, the noise level needs to be of order $\frac{1}{\sqrt{d}}$, hence we want to be able to take $r \approx d^{-1/2}$ and have that the probability in Equation (30) goes to 0, which is guaranteed by the condition $\frac{n}{d}\delta_d^2 = o(1)$. Hence we can follow (Ben Arous et al., 2021) p 38 and get that the trajectory will be below η for all $t \leq \tilde{t} := C_\eta \frac{d^2}{\delta}$, which is larger than our horizon, completing the proof.

A.3.2. Proof of proposition 4

We will focus on the case of correlated latent variables and prove weak recovery of both spikes. This will be a proof also of the weak recovery of the covariance spike in the case of independent latent variable (which is actually much simpler and is essentially a straight-forward verification of (Ben Arous et al., 2021) theorem 1.3 to our slightly different setting).

First note that we can restrict to initialisation with positive overlap, $\alpha_{u,0}, \alpha_{v,0} > 0$: if for example the initialisation was $\alpha_{u,0} < 0$ then we can just substitute $\hat{u} := -u$, $\hat{\lambda} = -\lambda$ so that the sign changes cancel each other and $\hat{x} = x$ but $\hat{\alpha}_{u,0} = -\alpha_{u,0}$. In the case of uncorrelated latents, the distribution would not change: $\hat{x} \sim x$, due to the symmetry of the latent variables and independents of the two directions, hence the new problem is equivalent to the previous one.

In case of positive correlation of the latent variables, we need to use our assumption that there is no sign mismatch in the initialisation. Unlike the uncorrelated case, we cannot change the sign of one of the axes without changing also the other, since it would flip the sign of the correlation. So there are only two cases: either $\alpha_{u,0}, \alpha_{v,0} > 0$, which is the desired initialisation; or $\alpha_{u,0}, \alpha_{v,0} < 0$, where we can flip the signs along *both the spiked directions*:

$$\hat{u} = -u,\ \hat{\lambda} = -\lambda, \quad \hat{v} = -v,\ \hat{\mu} = -\mu$$

So that $\hat{x} = x$ and $(\hat{\lambda}, \hat{\nu}) \sim (\lambda, \nu)$ keeping a positive correlation and $\hat{\alpha}_{u/v,0} = -\alpha_{u/v,0}$.

So from now on we can assume $\alpha_{u,0}, \alpha_{v,0} > 0$. Hence, for d large enough, $w_0 \in K_\eta := \left\{w : (\alpha_u, \alpha_v) \in [0, \eta]^2\right\}$, for $\eta > 0$ small, so we are able to use the expansions from Appendix A.2.1.

Recall from the statement that we assume $n = \theta_d d$, with $\theta_d \gtrsim \log^2 d$, and choose the step size $\delta = \delta_d$ such that ${}^1\!/_{\theta_d} \ll \delta \ll {}^1\!/_{\sqrt{\theta_d}}$. To prove weak recovery, we will show that there are $\eta_u, \eta_v \leq \eta$ such that, defining the τ_u, τ_v as the first times in which $\alpha_{u,t} \geq \eta_u$ and $\alpha_{v,t} \geq \eta_v$, then the probability that $\tau_u, \tau_v \leq n$ goes to 1 as $d \to \infty$.

Weak recovery of the covariance spike Given $\gamma > 0$, let $E_\gamma := \left\{w \mid \alpha_u \geq \frac{\gamma}{\sqrt{d}}, \alpha_v \geq \frac{\gamma}{\sqrt{d}}\right\}$. Since we consider the limit $d \to \infty$ and $w_0 \sim$Unif$(\mathbb{S}^{d-1})$, we can apply Poincaré lemma and have that for any $\varepsilon > 0$ we can find $\gamma > 0$ such that for all d sufficiently large

$$\mathbb{P}\left(w_0 \in E_\gamma\right) \geq 1 - \varepsilon \tag{31}$$

So, fix $\varepsilon > 0$ and from now on we will assume that $w \in E_\gamma$ for some $\gamma > 0$ sufficiently small (or d sufficiently large) so that (31) holds.

The first step is to prove that the *difference inequality* from *Proposition 4.1* from (Ben Arous et al., 2021) holds also in our case for $(\alpha_{u,t})_{t>0}$. So starting from the update equation (4) and taking the scalar product by u we get

$$\begin{aligned}\tilde{\alpha}_{u,t+1} &= \alpha_{u,t} - \frac{\delta}{d} u \cdot \nabla_{sph} \mathcal{L}\left(w, (x_t, y_t)\right) \\ &= \alpha_{u,t} - \frac{\delta}{d} u \cdot \left(\nabla_{sph} \mathcal{L}\left(w\right) + \nabla_{sph} H^t(w)\right)\end{aligned}$$

Where we used the definition of *sample-wise loss* $H^t(w) = \mathcal{L}\left(w, (x_t, y_t)\right) - \mathcal{L}\left(w\right)$. Now we need to normalise $\alpha_{u,t} = {}^{\tilde{\alpha}_{u,t}}\!/_{||\tilde{\alpha}_{u,t}||}$ but instead we just estimate the denominator, using the fact that if $\alpha_u, \alpha_v \leq \frac{1}{2}$ by Lemma 7 $\mathcal{L} \in \mathscr{C}^1(\mathcal{C}, \mathbb{R})$. So restricting in $\mathcal{K}_\eta$ for any $\eta < \frac{1}{2}$ we have that there exists $A > 0$:

$$\sup_{w \in \mathcal{K}_\eta} |\nabla_{sph} \mathcal{L}\left(w\right)| \leq A \tag{32}$$

and letting

$$\Theta_t := \left| \frac{1}{\sqrt{d}} \nabla_{sph} H^t\left(w_{t-1}\right) \right|^2 \tag{33}$$

It is straightforward to estimate:

$$1 \leq |\tilde{\alpha}_{u,t}| \leq 1 + \underbrace{\delta^2 \left(\frac{A}{d^2} + \frac{\Theta_t}{d}\right)}_{\zeta}$$

Then we can use the chain of inequalities: $\frac{1}{|\tilde{\alpha}_{u,t}|} \geq \frac{1}{1+\zeta} \geq 1 - \zeta$ to get:

$$\begin{aligned}\alpha_{u,t} \geq \alpha_{u,t-1} - \frac{\delta}{d} \nabla_{sph} \mathcal{L}(w) \cdot u - \frac{\delta}{d} u \cdot \nabla_{sph} H^t(w_{t-1}) - \delta^2 \left(\frac{A}{d^2} + \frac{\Theta_t}{d}\right) |\alpha_{u,t-1}| \\ - \delta^3 \left(\frac{A}{d^2} + \frac{\Theta_t}{d}\right) \left(\frac{1}{d} |\nabla_{sph} \mathcal{L}(w) \cdot u| + \frac{1}{d} |u \cdot \nabla_{sph} H^t(w_{t-1})|\right)\end{aligned} \tag{34}$$

Now, one of the key ideas employed in (Ben Arous et al., 2021) is to split the random term

$$\delta^2 \frac{\Theta_t}{d} |\alpha_{u,t-1}| = \delta^2 \frac{\Theta_t \mathbb{1}_{\Theta_t \leq \hat{\Theta}}}{d} |\alpha_{u,t-1}| + \delta^2 \frac{\Theta_t \mathbb{1}_{\Theta_t \geq \hat{\Theta}}}{d} |\alpha_{u,t-1}| \tag{35}$$

Where $\hat{\Theta}$ is a real parameter to be tuned carefully. So we can regroup the terms in (34) in two families (for the sake of

brevity, we use the notation $\Lambda := |\nabla_{sph}\mathcal{L}(w)\cdot u| + |u\cdot\nabla_{sph}H^t(w_{t-1})|$):

$$\alpha_{u,t} \geq \alpha_{u,t-1} \overbrace{-\frac{\delta}{d}\nabla_{sph}\mathcal{L}(w_{t-1})\cdot u - \delta^2\frac{\Theta_t \mathbb{1}_{\Theta_t\leq\hat{\Theta}}}{d}|\alpha_{u,t-1}|}^{F_1} + \underbrace{-\frac{\delta}{d}u\cdot\nabla_{sph}H^t(w_{t-1}) - \delta^2\left(\frac{A}{d^2}+\frac{\Theta_t\mathbb{1}_{\Theta_t\geq\hat{\Theta}}}{d}\right)|\alpha_{u,t-1}| - \frac{\Lambda\delta^3}{d}\left(\frac{A}{d^2}+\frac{\Theta_t}{d}\right)}_{F_2} \tag{36}$$

The family F_2 is made up by terms that are either always negligible, or that can cause problems only if *very unlikely realisations* of H and Θ happen. They are estimated in lemmas 4.3 and 4.5 of (Ben Arous et al., 2021), and, since our assumption A.2.2 implies assumption B in (Ben Arous et al., 2021), we are able to replicate those estimates also in our case.

On the contrary, to handle F_1 we need to make an adjustment: (Ben Arous et al., 2021) relied on their formula (4.12), that was derived from the monotonicity and mono-dimensionality of the population loss, which we do not have. However, consider the gradient of the population loss in direction u. Applying Lemma 7 and substituting the values from Appendix A.2.1 we have that $-u\nabla_{sph}\mathcal{L}(w) = \frac{1}{2}(c_{20}\alpha_u + c_{11}\alpha_v) + o(\eta)$, since both $c_{20} = 2\beta_u c_2^\sigma > 0$ and $c_{11} > 0$, taking η small enough and assuming $(\alpha_u, \alpha_v) \in \mathcal{K}_\eta$ we have that:

$$\frac{1}{4}\left(c_{20}\alpha_u + c_{11}\alpha_v\right) \leq -u\nabla_{sph}\mathcal{L}(w) \leq c_{20}\alpha_u + c_{11}\alpha_v \tag{37}$$

Hence, if we assume $\alpha_v \geq 0$, we get $-u\nabla_{sph}\mathcal{L}(w) \geq \frac{1}{4}c_{20}\alpha_u$, which is exactly what we needed to replicate equation (4.12) in (Ben Arous et al., 2021), so as long as $\alpha_{u/v,t-1} > 0$ we have that:

$$\alpha_{u,t} \geq \alpha_{u,t-1} - \frac{\delta}{d}\left(\frac{c_{20}}{4} - \delta\Theta_t\mathbb{1}_{\Theta_t\leq\hat{\Theta}}\right)\alpha_{u,t-1} + F_2 \tag{38}$$

Hence we are back on the setting studied in (Ben Arous et al., 2021), and using that $\delta \lesssim \frac{1}{\log d}$ we can apply also lemma 4.2 and proposition 4.4 in (Ben Arous et al., 2021).

So, conditioning on positivity of $(\alpha_{v,t})_{t\leq\tau}$ we can apply to $(\alpha_{u,t})_t$ *Proposition 4.1* from (Ben Arous et al., 2021), that implies

$$\lim_{d\to\infty}\inf_{w_0\in E_{\gamma/\sqrt{d}}}\mathbb{P}_{w_0}\left(\alpha_{u,t} \geq \frac{\alpha_{u,0}}{2} + \frac{\delta}{8d}\sum_{j=0}^{t-1}c_{20}\alpha_{u,j}\forall\, t\leq\tau \,\middle|\, \alpha_{v,t}\geq 0\,\forall t\leq\tau\right) = 1$$
$$\text{where } \tau \text{ is a stopping time defined as} \qquad \tau := \inf\left\{t\leq n \middle| \alpha_{u,t} \leq \frac{\gamma}{2\sqrt{d}} \text{ or } \alpha_{u,t}\geq\eta\right\} \tag{39}$$

where $\mathbb{P}_{w_0}$ denotes the probability conditioned on starting from w_0.

Recall discrete Gronwall inequality

$$x_n \geq a + b\sum_{i=0}^{n-1}x_i \quad\Longrightarrow\quad x_n \geq a\,(1+b)^n\,, \qquad a,b>0$$

Using it, we get that, if the event in (39) happens, then

$$\alpha_{u,t} \geq \frac{\alpha_{u,0}}{2}\left(1+\frac{\delta c_{20}}{8d}\right)^t \qquad \forall t\leq\tau \tag{40}$$

So, recalling that we are conditioning to be in E_γ, so $\alpha_{u,0} \geq \frac{\gamma}{\sqrt{d}}$, (40) proves that $\alpha_{u,t}$ will have surpassed η_u after a number of steps t_u^*:

$$t_u^* = \frac{\log\left(2\eta_u\sqrt{d}\right) - \log\gamma}{\log\left(1+\frac{\delta c_{20}}{8d}\right)} \approx C\frac{d}{\delta}\log d \tag{41}$$

This proves weak recovery of the covariance spike in the time require ending the first part of Proposition 4.

Weak recovery of the cumulant spike Now turn to $(\alpha_{v,t})_t$. Starting from the equivalent of (36), we cannot apply the same reasoning this time because

$$-v \cdot \nabla_{sph}\mathcal{L}(w) = c_{11}\alpha_u(1+o(\eta)) + 4c_{04}\alpha_v^3(1+o(\eta)) \qquad \alpha_u, \alpha_v \leq \eta. \tag{42}$$

There is no linear term in α_v, hence we cannot regroup the term $\delta\Theta_t \mathbb{1}_{\Theta_t \leq \hat{\Theta}}|\alpha_{v,t-1}|$ as did in (38).

We will instead prove a the following:

$$\alpha_{v,t} \geq \frac{\tilde{\gamma}}{2\sqrt{d}}\left(1 + \frac{\delta \min(c_{20}, c_{11})}{8d}\right)^t \tag{43}$$

for some $\tilde{\gamma}$ small enough. To show it, we apply lemma 4.3, 4.5 from (Ben Arous et al., 2021), and (42), to get that:

$$\lim_{d\to\infty} \inf_{w_0 \in E_{\gamma/\sqrt{d}}} \mathbb{P}_{w_0}\left(\alpha_{v,t} \geq \frac{\alpha_{v,0}}{2} + \frac{\delta}{4d}\sum_{j=0}^{t-1} c_{11}\alpha_{u,j} + 4c_{04}\alpha_{v,j}^3 - \frac{\delta^2}{d}\mathbb{1}_{\Theta_j \leq \hat{\Theta}}\alpha_{v,j} \quad \forall\, t \leq \tilde{\tau}\right) = 1$$

$$\text{where } \tilde{\tau} \text{ is a stopping time defined as} \qquad \tilde{\tau} := \inf\left\{t \leq n \middle| \, \alpha_{v,t} \leq \frac{\gamma}{2\sqrt{d}} \text{ or } \alpha_{v,t} \geq \eta\right\}$$

We need to divide in 2 cases to be able to carry one the estimates: suppose first that $\alpha_{u,s} \geq \alpha_{v,s}$ for all $s \leq t$. Hence $\delta^2\alpha_{v,t} \leq \delta^2\alpha_{u,t}$ hence we can also apply proposition 4.4 to get:

$$\begin{gathered}\lim_{d\to\infty} \inf_{w_0 \in E_{\gamma/\sqrt{d}}} \mathbb{P}_{w_0}\left(\alpha_{v,t} \geq \frac{\alpha_{v,0}}{2} + \frac{\delta}{8d}\sum_{j=0}^{t-1} c_{11}\alpha_{u,j} \quad \forall\, t \leq \mathcal{S}_u\right) = 1 \\ \text{where } \mathcal{S}_u \text{ is defined as } \mathcal{S}_u := \inf\left\{t \leq n \middle| \, \alpha_{v,t} \leq \frac{\tilde{\gamma}}{2\sqrt{d}} \text{ or } \alpha_{v,t} \geq \eta \text{ or } \alpha_{v,t} \geq \alpha_{u,t}\right\}\end{gathered} \tag{44}$$

This implies that as long as α_v is below α_u, it satisfies the same kind of Gronwall inequality, with different coefficients, leading to:

$$\alpha_{v,t} \geq \frac{\alpha_{v,0}}{2}\left(1 + \frac{\delta c_{11}}{8d}\right)^t \qquad t \leq \mathcal{S}_u$$

so it satisfies (43) for this selection of times. Now note that we actually proved a stronger property. If at any time t: $\frac{\tilde{\gamma}}{2\sqrt{d}}\left(1 + \frac{\delta \min(c_{11}, c_{20})}{8d}\right)^t \leq \alpha_{v,t} \leq \alpha_{u,t}$, then we know that same reasoning applies and, calling $\mathcal{S}_u^{(2)}$ the next instant such that with $\alpha_{v,\mathcal{S}_u^{(2)}} \geq \alpha_{u,\mathcal{S}_u^{(2)}}$ we get that for all times $t \leq s \leq \mathcal{S}_u^{(2)}$:

$$\alpha_{v,s} \geq \frac{\alpha_{v,t}}{2}\left(1 + \frac{\delta c_{11}}{8d}\right)^{s-t}$$

hence $\alpha_{v,s}$ satisfies (43).

Consider now the other case: if at any instant t we have that $\alpha_{v,t-1} \geq \alpha_{u,t-1}$ we want to prove that it cannot happen that $\alpha_{v,t} \leq \frac{\tilde{\gamma}}{2\sqrt{d}}\left(1 + \frac{\delta \min(c_{20}, c_{11})}{8d}\right)^t$. To do this we use an estimate on the update equation (it can be easily verified that comes from the application of lemma 4.3,4.5 to formula (4.11) of (Ben Arous et al., 2021), with their choice $\hat{\Theta} = d^{\frac{1}{2}-\frac{1}{4}\iota}$):

$$\begin{aligned}\alpha_{v,t} &\geq \left(1 - \frac{\delta^2}{d^{1/2+1/4\iota}}\right)\alpha_{v,t-1} + \frac{c_{11}\delta}{8d}\alpha_{u,t-1} - \frac{\tilde{\gamma}}{5\sqrt{d}} \\ &\geq \left(1 - \frac{\delta^2}{d^{1/2+1/4\iota}} + \frac{c_{11}\delta}{8d}\right)\alpha_{u,t-1} - \frac{\tilde{\gamma}}{5\sqrt{d}} \\ &\geq \left(1 - O\left(d^{-1/2}\right)\right)\frac{\alpha_{u,0}}{2}\left(1 + \frac{\delta c_{20}}{8d}\right)^t - \frac{\tilde{\gamma}}{5\sqrt{d}}\end{aligned}$$

Which verifies our requirement by taking $\tilde{\gamma}$ small enough with respect to γ and d large enough.

Hence we verified that conditioning on the the events (31) and (39) (that have arbitrarily large probabilities), also inequality (43) holds. This means that we can drop all the positivity requirements and have that, as long as $0 < \alpha_{u,t}, \alpha_{v,t} \leq \eta$, then with high probability $(\alpha_{u,t})_t$ satisfies (40) until it reaches η_u and $(\alpha_{v,t})_t$ satisfies (43) until it reaches η_v, hence we need to take n larger than the maximum of those bounds, which is (43):

$$n > t_v^* = \frac{\log\left(2\eta\sqrt{d}\right) - \log\tilde{\gamma}}{\log\left(1 + \frac{\delta\min(c_{20},c_{11})}{8d}\right)} \approx C\frac{d}{\delta\min(c_{20},c_{11})}\log d \tag{45}$$

Where the first constant C is just a number, does not depend on $\eta, \tilde{\gamma}$, for d large enough. Substituting the constraint $\delta \ll \frac{1}{\log d}$ (that we used in order to apply proposition 4.1 from (Ben Arous et al., 2021)) we get that $n \gtrsim d\log^2 d$ is a sufficient sample complexity for weak recovery if δ is as close as possible to the upper bound $\frac{1}{\log d}$. If instead we take a smaller learning rate $\delta \approx \frac{1}{d^2}$ we have that if $n \gg d^2\log d$ weak recovery is guaranteed, as shown in Table 1.

So the proposition is proved under the assumption that all the trajectories live in K_η with η sufficiently small. To conclude we need to make sure that it cannot happen that one of $(\alpha_{u/v,t})_t$ surpasses η while the other is still close to 0 and far away from $\eta_{u/v}$. Note that up until now we did not choose explicitly values for $\eta_{u/v}$, so it will be sufficient to take them way smaller than η to be sure that there will be no problems.

To be more precise, we will use the following bound on the maximum progress achieved in a fixed time interval.

Bound on maximum progress Supposing $\alpha_{u,t} > 0$ (all that we will do here works also for α_v, t), since the direction of the spherical gradient is tangent to the sphere it follows that $||\tilde{w}_{u,t}|| > 1$, hence

$$\alpha_{u,t} \leq \tilde{\alpha}_{u,t} = \alpha_{u,t-1} - \frac{\delta}{d}\nabla_{sph}\mathcal{L}(w_{t-1})\cdot u - \frac{\delta}{d}u\cdot\nabla_{sph}H^t(w_{t-1}) \tag{46}$$

Applying this inequality iteratively in a time interval $[t_1, t_2]$, assuming $\alpha_{u,t} > 0$ in all the interval, we get that

$$\alpha_{u,t_2} \leq \alpha_{u,t_1} - \sum_{j=t_1}^{t_2-1}\frac{\delta}{d}\nabla_{sph}\mathcal{L}(w_j)\cdot u + \frac{\delta}{d}u\cdot\nabla_{sph}H^{j+1}(w_j)$$

To estimate the martingale term we use the estimate from lemma 4.5 in (Ben Arous et al., 2021) (which is an application of Doob maximal inequality):

$$\sup_{T\leq n}\sup_{w_o\in\mathbb{S}^{d-1}}\mathbb{P}_{w_0}\left(\max_{t\leq T}\frac{1}{\sqrt{T}}\left|\sum_{j=0}^{t-1}\nabla_{sph}H^{j+1}(w_j)\cdot u\right| \geq r\right) \leq \frac{2C_1}{r^2}, \tag{47}$$

where $r > 0$ is a positive parameter and C_1 comes from assumption A.2.2. Together with (32), we have that with probability larger than $1 - \frac{2C_1}{r^2}$

$$\alpha_{u,t_2} - \alpha_{u,t_1} \leq \frac{\delta}{Ad}(t_2 - t_1) + \frac{\delta}{d}\sqrt{t_2 - t_1}r \tag{48}$$

assume that at t_1 we have that $\alpha_{u,t_1} \geq \eta_1$ and that at $\alpha_{u,t_2} = \eta_2$ while $\alpha_{u,t} \geq \eta_1$ for all $t \in [t_1, t_2]$, choosing $r = \sqrt{t_2 - t_1}$ we get that with probability larger than $1 - \frac{2C_1}{t_2 - t_1}$

$$t_2 - t_1 \geq C(\eta_2 - \eta_1)\frac{d}{\delta} \gg C(\eta_2 - \eta_1)d\log d \tag{49}$$

so in the limit $d \to \infty$ with probability converging to 1 $t_2 - t_1 \gg d\log d$.

Application of the bound Suppose that at time t_2 $\alpha_{u,t_2} = \eta =: \eta_u$, hence there exists t_1 such that $\alpha_{u,t} \geq \frac{\eta}{2}$ for all $t \in [t_1, t_2]$. By (49), $t_2 - t_1 \geq \frac{C\eta}{2}d\log d$. If $\alpha_{v,t}$ is larger than $\alpha_{u,t}$ for any t in this interval, the statement is already verified by taking $\eta_v = \eta_u$, so assume $\alpha_{v,t} \leq \alpha_{u,t}$ for $t \in [t_1, t_2]$. We can apply Equation (44) to get that with probability converging to 1 in the limit $d \to \infty$:

$$\alpha_{v,t_2} \geq \frac{\alpha_{v,t_1}}{2} + \frac{\delta}{8d}\sum_{j=t_1}^{t_2-1}c_{11}\alpha_{u,j} \geq c_{11}\frac{\delta\eta_u}{16d}(t_2 - t_1) \geq C\eta_u^2$$

where the constant C does not depend on d, hence we can take $\eta_v := C\eta_u^2$ there is weak recovery for the cumulant spike v even in the "unlucky" case where the overlap along u reaches η very fast.

The vice versa can be done analogously: assume that at time t_2 $\alpha_{v,t_2} = \eta =: \eta_v$. Again, let t_1 such that $\alpha_{v,t} \geq \frac{\eta}{2}$ for all $t \in [t_1, t_2]$. By Equation (49), $t_2 - t_1 \geq \frac{C\eta}{2} d \log d$. Now note that by (37) $-u \cdot \nabla_{\text{sph}} L(w) \geq \frac{c_{11}\alpha_{v,t}}{4}$, so the same reasoning as before applies leading to weak recovery with $\eta_u := C\eta_v^2$.

B. Details on the experiments

B.1. Experiments with two-layer networks

Training In all our experiments with two-layer neural networks on *synthetic* data, be it on the mixed cumulant model or in the teacher-student setup, we trained two-layer neural networks using online stochastic gradient with mini-batch size 1. Note that we used different learning rates for the first and second layer, with the second layer learning rate $\eta_2 = \epsilon\eta_1$ and $\epsilon = 0.01$ unless otherwise noted. It has been noted several times that rescaling the second-layer learning rate in this way ensures convergence to a well-defined mean-field limit (Riegler & Biehl, 1995; Berthier et al., 2023)

B.1.1. FIGURE 1

CIFAR10 We trained a two-layer neural network with $m = 32^2 * 4 = 4096$ hidden neurons on grayscale CIFAR10 images to ensure the same ratio between hidden neurons and input dimension as in our experiments with synthetic data. We set the mini-batch size to 128, weight decay to $5 \cdot 10^{-4}$, and momentum to 0.9. Images were transformed to grayscale using the pyTorch conversion function. We trained for 200 epochs on the cross-entropy loss and used a cosine learning rate scheduler. The final test accuracy of the networks was just below 50%.

Synthetic data distributions We trained the two-layer neural network on a data set sampled from the mixed cumulant model with signal-to-noise ratios $\beta_m = 1, \beta_u = 5, \beta_v = 10$. The test error on this data set is shown in red ("Full data"). For the independent latent variables, we used $\lambda^\mu \sim \mathcal{N}(0,1)$ and $\nu^\mu = \pm 1$ with equal probability. For correlated latent variables, we set $\nu^\mu = \text{sign}(\lambda^\mu)$. We also show the test loss of the same network evaluated on the following censored data sets:

- $\beta_m = 1, \beta_u = 0, \beta_v = 0$ (blue, "mean only")
- $\beta_m = 1, \beta_u = 5, \beta_v = 0$ (green, "mean + cov")
- A Gaussian equivalent model (orange), where we sample inputs for the spiked class from a Gaussian distribution with mean $\beta_m m$ and a covariance

$$\text{cov}_{\mathbb{P}}(x,x) = \mathbb{1} + \beta_u uu^\top + \sqrt{\beta_u\beta_v(1-\gamma)^2}\, \mathbb{E}\, \lambda\nu \left(uv^\top + vu^\top\right) \tag{50}$$

B.1.2. TEACHER-STUDENT SETUP (FIGURE 2)

We trained the same two-layer network with the same hyperparameters as in Figure 1. Defining the pre-activations $\lambda_i = \langle u_i, x\rangle$ for the three spikes u_i, the tasks were:

A inputs $x \sim \mathcal{N}(0, \mathbb{1})$, teacher

$$y(x) = h_1(\lambda_1) + h_2(\lambda_2) + h_4(\lambda_3) \tag{51}$$

B For the simulation shown in A, We computed the absolute normalised overlaps $|\langle w_k, u_i\rangle|/\|w_k\|$ of all the first-layer weights w_k with a given spike, and took the average over the five highest values.

C Same as in B, except that we sampled inputs from a normal distribution with zero mean and covariance $\mathbb{1} + \gamma(uv^\top + vu^\top)$, which introduces correlations between the pre-activations λ_2 and λ_3 of the teacher model.

D Same as in B, except that we trained on isotropic Gaussian inputs $\mathcal{N}(0, \mathbb{1})$ and instead added a co-linear cross-term between spikes u and v, akin to the cross-term that is found in our expansion of the mixed cumulant task,

$$y(x) = h_1(\lambda_1) + h_1(\lambda_2)h_1(\lambda_3) + h_2(\lambda_2) + h_4(\lambda_3) \tag{52}$$