Analysing the dynamics of online learning in over-parameterised two-layer neural networks

Sebastian Goldt [1] **Madhu S. Advani** [2] **Andrew M. Saxe** [3] **Florent Krzakala** [4] **Lenka Zdeborová** [1]

Abstract

Deep neural networks achieve stellar generalisation even when they have enough parameters to easily fit all their training data. We study this phenomenon by analysing the dynamics of two-layer neural networks in a teacher-student setup, where one network, the student, is trained on data generated by another network, called the teacher. We show how the dynamics of stochastic gradient descent (SGD) in this model are captured by a set of differential equations. We calculate the generalisation error of students that have more parameters than their teacher and find that the asymptotic generalisation error increases with network size, keeping other relevant parameters constant. Our results indicate that achieving good generalisation in large neural networks depends on the interplay of at least the algorithm, its learning rate, the model architecture, and the data set.

1. Introduction

One hallmark of the deep neural networks behind state-of-the-art results in image classification and other domains is their size: their free parameters outnumber the samples in their training set by up to two orders of magnitude (LeCun et al., 2015; Simonyan & Zisserman, 2015). Statistical learning theory would suggest that such heavily over-parameterised networks should generalise poorly without further regularisation (Bartlett & Mendelson, 2003; Mohri et al., 2012), yet empirical studies consistently find that increasing the size of networks to the point where they can fit their training data and beyond does not impede their ability to generalise well (Neyshabur et al., 2015; Zhang et al., 2017; Arpit et al., 2017). Resolving this paradox is arguably one of the big challenges in the theory of deep learning. Here, we study the dynamics of neural networks in the teacher-student setup, which will give us a precise notion of over-parameterisation as follows.

We consider a supervised regression problem with training set $\{(x^\mu, y_B^\mu)_{\mu=1,\dots,P}\}$. The components of the inputs $x^\mu \in \mathbb{R}^N$ are i.i.d. draws from the standard normal distribution, which we denote $\mathcal{N}(0,1)$. The scalar outputs $y_B^\mu \equiv \phi(B, x^\mu) + \zeta^\mu$ are computed using a two-layer neural network with M hidden units and weights $B \in \mathbb{R}^{M\times N}$, called the *teacher*. We choose $\phi(B, x^\mu) = \sum_{m=1}^{M} g\left(B_m x^\mu/\sqrt{N}\right)$, where B_m is the mth row of B, and $g(\cdot)$ is the non-linear activation function of the teacher. The additive output noise ζ^μ is drawn from $\mathcal{N}(0, \sigma^2)$.

A second two-layer network with weights $w \in \mathbb{R}^{K\times N}$ and output $\phi(w, x)$, called the *student*, is then trained using SGD on the quadratic training loss $E(w) \propto \sum_{\mu=1}^{P} \left(\phi(w, x^\mu) - y_B^\mu\right)^2$. Crucially, the student network may have a larger number of hidden units $K \geq M$ than the teacher and thus be over-parameterised with respect to the generative model of its training data in a controlled way.

Main contributions. We derive a set of ordinary differential equations (ODEs) that track the generalisation error

$$\epsilon_g(w, B) \equiv \langle [\phi(w, x) - \phi(B, x)]^2/2 \rangle \qquad (1)$$

of a *typical* over-parameterised student during SGD. This description becomes exact for large input dimension N and data sets that are large enough to allow that we visit every sample only once before training converges. Using this framework, we analytically calculate the generalisation error after convergence ϵ_g^*. We find that with other relevant parameters held constant, the generalisation error increases at least linearly with the over-parameterisation $L \equiv K - M \geq 0$. For small learning rates η in particular, we have

$$\epsilon_g^* \sim \eta \sigma^2 L. \qquad (2)$$

Our model thus offers an interesting perspective on the implicit regularisation of SGD, which we will discuss in detail.

[1]Institut de Physique Théorique, CNRS, CEA, Université Paris-Saclay, 91191 Gif-sur-Yvette, France [2]Center for Brain Science, Harvard University, Cambridge, MA 02138, USA [3]Department of Experimental Psychology, University of Oxford, United Kingdom [4]Laboratoire de Physique Statistique, Sorbonne Universités, Université Pierre et Marie Curie Paris 6, Ecole Normale Supérieure, 75005 Paris, France. Correspondence to: Sebastian Goldt <sebastian.goldt@ipht.fr>.

Presented at the ICML 2019 Workshop on Theoretical Physics for Deep Learning. Copyright 2019 by the author(s).

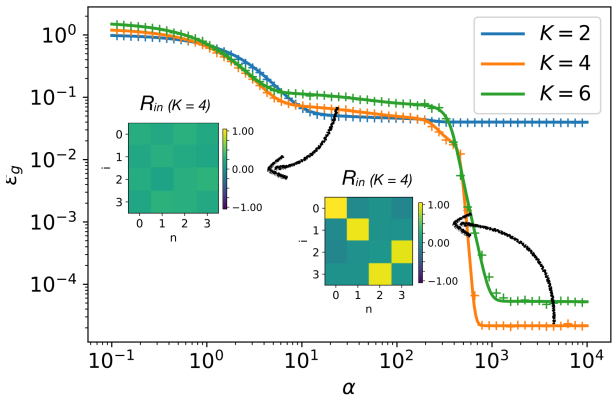

Figure 1. **The analytical description of the generalisation dynamics of sigmoidal networks matches experiments.** We plot the learning dynamics $\epsilon_g(\alpha)$ obtained by integration of the ODEs (5) (solid) and from a single run of SGD (3) (crosses) for students with different numbers of hidden units K. The insets show the values of the teacher-student overlaps R_{in} (4) at two times during training as indicated by the arrows. $N = 784, \kappa = 0$.

Related work. Our work builds on the seminal papers of Biehl & Schwarze (1995) and Saad & Solla (1995a;b), who studied the case $K = M$. Their work is part of a rich tradition of analysing average-case behaviour in learning and inference using statistical physics (Gardner & Derrida, 1989; Kinzel et al., 1990; Seung et al., 1992; Watkin et al., 1993; Engel & Van den Broeck, 2001) that has recently seen a surge of interest (Zdeborová & Krzakala, 2016; Advani & Ganguli, 2016; Chaudhari et al., 2017; Advani & Saxe, 2017; Aubin et al., 2018; Baity-Jesi et al., 2018). Two-layer neural networks have recently attracted a lot of interest as models of generalisation in neural networks across communities (Mei et al., 2018; Rotskoff & Vanden-Eijnden, 2018; Aubin et al., 2018; Chizat & Bach, 2018; Li & Liang, 2018).

2. The dynamics of online learning can be described in closed form

The weight updates of stochastic gradient descent on the quadratic training error $E(w)$ can be written as

$$w_k^{\mu+1} = w_k^\mu - \frac{\kappa}{N} w_k^\mu - \frac{\eta}{\sqrt{N}} x^\mu r_k^\mu \tag{3}$$

where $r_k^\mu \equiv g'(\lambda_k^\mu)\,[\phi(w, x^\mu) - y_B^\mu]$ and $\lambda_k^\mu \equiv w_k x^\mu/\sqrt{N}$. The scaling of the weight decay rate κ and the learning rate η are such that all terms remain of order 1 as $N \to \infty$. The initial weights w^0 are i.i.d. draws from $\mathcal{N}(0, 1)$. In indexing the steps of the algorithm by μ, we are using the fact that we visit every training sample only once during training until convergence of the generalisation error to its final value. This limit is known as one-shot or *online learning* and has been studied for models ranging from PCA (Oja & Karhunen, 1985; Wang et al., 2017) to generative adversarial networks (Wang et al., 2018).

A key insight is that $\epsilon_g(w, B)$ can be expressed as a function of only two sets of macroscopic variables,

$$Q_{ik} \equiv \frac{w_i w_k}{N} \quad \text{and} \quad R_{in} \equiv \frac{w_i B_n}{N}, \tag{4}$$

which are called *order parameters* in statistical physics. We give the full expression of $\epsilon_g(Q, R)$ in Eq. (S14). Intuitively, R_{in} measures the overlap or the similarity between the weights of the ith hidden unit of the student and the nth hidden unit of the teacher, while Q_{ik} quantifies the overlap of the weights of the ith and kth hidden unit of the student, resp.

We can obtain a closed set of differential equations for the time evolution of the order parameters Q and R by squaring the weight update (3) and taking its inner product with B_n, respectively. Then, an average over the inputs x, denoted by $\langle\cdot\rangle$, needs to be taken. We detail this procedure in Sec. A; it yields these equations of motion for R and Q:

$$\frac{\mathrm{d}R_{in}}{\mathrm{d}\alpha} = -\kappa R_{in} + \eta\langle r_i \nu_n\rangle \tag{5a}$$

$$\frac{\mathrm{d}Q_{ik}}{\mathrm{d}\alpha} = -2\kappa Q_{ik} + \eta\langle r_i \lambda_k\rangle + \eta\langle r_k \lambda_i\rangle + \eta^2\langle r_i r_k\rangle + \eta^2\sigma^2\langle g'(\lambda_i) g'(\lambda_k)\rangle \tag{5b}$$

where $\nu_n^\mu \equiv B_n x^\mu/\sqrt{N}$ and $\alpha = \mu/N$ becomes a continuous time-like variable in the limit $N \to \infty$. The averages $\langle\cdot\rangle$ can be evaluated analytically and the equations close for the choice $g(x) = \mathrm{erf}(x/\sqrt{2})$ (Biehl & Schwarze, 1995) and for linear networks. We plot $\epsilon_g(\alpha)$ obtained by numerically integrating[1] the ODEs (5) (solid) and from a single run of SGD (3) (crosses) for students with varying K in Fig. 1, which are in very good agreement. The equations (5) were first written by Saad & Solla (1995a), who also gave an extensive analysis of the special case $K = M$ (Saad & Solla, 1995b; 1997).

One notable feature of $\epsilon_g(\alpha)$ is the existence of a long plateau with sub-optimal generalisation error during training. During this period, the student "believes" that data are linearly separable and all of its hidden units have roughly the same overlap with all the hidden units of the teacher, $R_{in} = R = \mathrm{const.}$ (left inset in Fig. 1). As training continues, the student "specialises" and each of its hidden units ideally becomes strongly correlated with only one hidden unit of the teacher (right inset), as the generalisation error decreases exponentially to its final value. This effect is well-known for both batch and online learning (Engel & Van den Broeck, 2001) and will be key for our discussion in Sec. 3.

[1] We have packaged our experiments and our ODE integrator into a user-friendly Python library with example programs at https://github.com/sgoldt/pyscm

3. Generalisation error after online learning

Our goal is to extract the asymptotic generalisation error ϵ_g^* and in particular its scaling with L from the equations of motion (5). Our first contribution is to reduce the $K(K+M)$ equations to a set of eight coupled differential equations for any combination of K and M in Sec. B. This enables us to obtain a closed-form expression for ϵ_g^* as follows.

In the absence of output noise ($\sigma = 0$) and without weight decay ($\kappa = 0$), the generalisation error of a student with $K \geq M$ will asymptotically tend to zero as $\alpha \to \infty$. On the level of the order parameters, this corresponds to reaching a stable fixed point of (5) with $\epsilon_g(Q, R) = 0$. With small noise $\sigma > 0$, the order parameters instead converge to another, nearby fixed point with finite ϵ_g. The values of the order parameters at that fixed point can be obtained by perturbing Eqns. (5) to first order in σ, and the corresponding generalisation error $\epsilon_g(Q, R)$ turns out to be in excellent agreement with the generalisation error obtained when training a neural network using (3) starting from random initial conditions as described in Sec. 2.

3.1. Sigmoidal networks

We have performed this calculation for teacher and student networks with $g(x) = \mathrm{erf}(x/\sqrt{2})$. We relegate the details to Sec. B.2, and content us here to state the asymptotic value of the generalisation error to first order in σ^2,

$$\epsilon_g^* = \frac{\sigma^2 \eta}{2\pi} f(M, L, \eta) + \mathcal{O}(\sigma^3), \tag{6}$$

where $f(M, L, \eta)$ is a lengthy rational function of its variables. We plot our result in Fig. 2 together with the final generalisation error obtained in a single run of SGD (3) for a neural network with initial weights drawn i.i.d. from $\mathcal{N}(0, 1)$ and find excellent agreement, which we confirmed for a range of values for η, σ, and L.

One notable feature of Fig. 2 is that with all else being equal, SGD alone fails to regularise the student networks of increasing size in our setup, instead yielding students whose generalisation error increases at least linearly with L.

We can gain some intuition for this scaling by considering the asymptotic overlap matrices Q and R shown in the left half of Fig. 3. In the over-parameterised case, $L = K - M$ units of the student are effectively trying to specialise to hidden units of the teacher which do not exist, or equivalently, have weights zero. These L hidden units of the student do not carry any information about the teachers output, but they pick up fluctuations from output noise and thus increase ϵ_g*. This intuition is borne out by an expansion of ϵ_g^* in the limit of small learning rate η, which yields

$$\epsilon_g^* = \frac{\sigma^2 \eta}{2\pi} \left(L + \frac{M}{\sqrt{3}} \right) + \mathcal{O}(\eta^2), \tag{7}$$

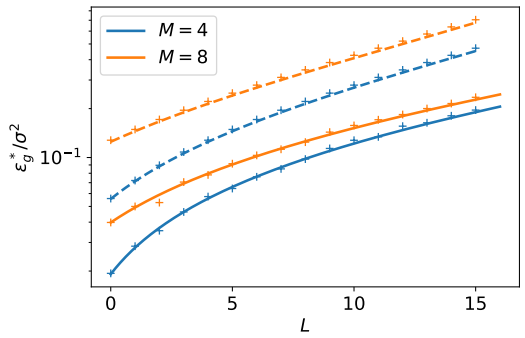

Figure 2. **Theoretical predictions for ϵ_g^* match experiments with sigmoidal and linear networks.** We plot theoretical predictions for ϵ_g^*/σ^2 for sigmoidal networks (Eq. (6), solid line) and linear networks (Eq. (8), dashed) together with the result from a single run of SGD (3) starting with random initial weights (crosses). Parameters: $N = 784, \eta = 0.05, \sigma = 0.01$.

which is indeed the sum of the error of M independent hidden units that are specialised to a single hidden unit of the teacher, and $L = K - M$ superfluous units contributing each the error of a hidden unit that is "learning" from a hidden unit with zero weights $B_m = 0$ (see also Sec. C).

3.2. Linear networks

One might suspect that part of the scaling $\epsilon_g^* \sim L$ in sigmoidal networks is due to the specialisation of the hidden units or the fact that teacher and student network can implement functions of different range if $K \neq M$. To test these hypotheses, we calculated ϵ_g^* for linear neural networks with $g(x) = x$ (Krogh & Hertz, 1992; Saxe et al., 2014). These networks lack a specialisation transition (Aubin et al., 2018) and their output range is set by the magnitude of their weights, rather than their number of hidden units. Following the same steps as for the sigmoidal networks, a perturbative calculation in the limit of small noise variance σ^2 yields

$$\epsilon_g^* = \frac{\eta \sigma^2 (L + M)}{4 - 2\eta(L + M)} + \mathcal{O}(\sigma^3). \tag{8}$$

This result is again in good agreement with the results of experiments, demonstrated in Fig. 2. In the limit of small learning rates η, Eq. (6) simplifies to yield the same scaling as for sigmoidal networks,

$$\epsilon_g^* = \frac{1}{4} \eta \sigma^2 (L + M) + \mathcal{O}\left(\eta^2\right). \tag{9}$$

This shows that the scaling ϵ_g^* L is not just a consequence of either specialisation or the mismatched range of the networks' output functions. Note that the optimal number of hidden units for linear networks is $K = 1$ for all M, because linear networks implement an effective linear transformation with an effective matrix $W = \sum_k w_k$. Adding hidden

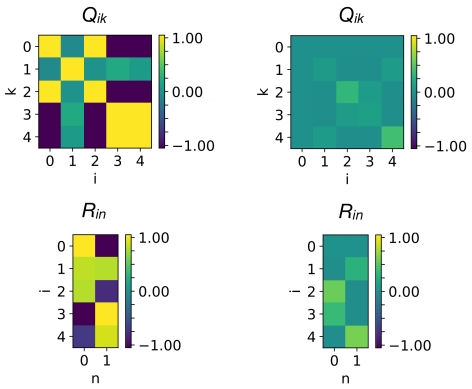

Figure 3. **Sigmoidal networks learn different representations from a noisy teacher than ReLU networks.** We show the final overlap matrices Q and R, Eq. (4), after convergence of online learning with $M = 2, K = 5$. With a sigmoidal activation function (left), the student shows clear signs of specialisation as described in Sec. 3.1. ReLU networks (right half) instead converge to solutions where all of the student's hidden units are used. Parameters: $N = 784, \eta = 0.3, \sigma = 0.1, \kappa = 0$.

units to a linear network hence does not augment the class of functions it can implement, but it adds redundant parameters which pick up fluctuations from the teacher's output noise, increasing ϵ_g.

3.3. ReLU networks

The analytical calculation of ϵ_g^* described above for networks with ReLU activation $g(x) = \max(0, x)$ poses some additional technical challenges, so we resort to numerical experiments to investigate this case. We found numerically that the asymptotic generalisation error of a ReLU student learning from a ReLU teacher has the same scaling as the one we found analytically for networks with sigmoidal and linear activation functions: $\epsilon_g^* \sim \eta\sigma^2 L$ (see Fig. S3).

Looking at the final overlap matrices Q and R for ReLU networks in the right half of Fig. 3, we see that instead of the one-to-one specialisation of sigmoidal networks, all hidden units of the student have a finite overlap with some hidden unit of the teacher. This is a consequence of the fact that it is much simpler to re-express the sum of M ReLU units with $K \neq M$ ReLU units. However, there are still a lot of redundant degrees of freedom in the student, which all pick up fluctuations from the teacher's output noise and increase ϵ_g^*.

4. Discussion

We finally discuss the impact of several tweaks to our setup which we investigated numerically. The details of all experiments can be found in the supplementary material.

A standard regularisation method is introducing weight decay by choosing $\kappa > 0$ in Eq. (3). However, we did not find a scenario in our experiments where weight decay improved the performance of a student with $L > 0$ (Sec. D). We also made sure that our results persist when performing SGD with mini-batches, where the gradient estimate in Eq. (3) is averaged over several samples (x, y_B). While increasing the mini-batch size lowers the asymptotic generalisation error, it does not change the scaling of ϵ_g^* with L (Sec. E). We repeated our experiments using MNIST images as inputs x, while leaving all other aspects of our setup (regression task, y generated using a random teacher, etc.) the same. This allowed us to investigate the impact of higher-order correlations of the input distribution on the generalisation error of the student. Our experiments reproduced the same ϵ_g-L curve as having Gaussian inputs to within the experimental error (Sec. F).

We also analysed the impact of having a finite training set where we revisit examples before training converges. The behaviour of linear networks did not change qualitatively: the bigger the network, the worse the performance, and the optimal network has $K = 1$ hidden units for all M. However, for non-linear networks, the picture is more varied. For large training sets, we recover qualitatively the behaviour found in Sec. 3: the best generalisation is obtained by a student network with $K = M$, and the error increases with L. However, as we reduce the size of the training set, this is no longer true: as we detail in Sec. G, we found that for example for $P = 4$, the lowest asymptotic generalisation error is obtained with networks that have $K > M$. Thus the size of the training set with respect to the network has an important influence on the scaling of ϵ_g^* with L.

5. Concluding perspectives

We have studied the dynamics of online learning in two-layer neural networks within the teacher-student framework. We derived an analytical expression for the final generalisation error of a student with sigmoidal activation function in the limit of online learning with small noise, and found that it scales linearly with the network size. This result proved robust in experiments with weight decay, mini-batches or more realistic input data. However, for finite training sets with roughly as many samples as there are free parameters in the network, the final generalisation error can decrease with network size.

Our results clearly indicate that the regularisation of neural networks in our setting goes beyond the properties of SGD alone. Instead, a full understanding of the generalisation properties of deep networks requires taking into account the interplay of at least the algorithm, its learning rate, the model architecture, and the data set, setting up a formidable research programme for the future.

6. Acknowledgements

SG and LZ acknowledge funding from the ERC under the European Unions Horizon 2020 Research and Innovation Programme Grant Agreement 714608-SMiLe. MA thanks the Swartz Program in Theoretical Neuroscience at Harvard University for support. AS acknowledges funding by the European Research Council, grant 725937 NEUROABSTRACTION. FK acknowledges support from "Chaire de recherche sur les modles et sciences des donnes", Fondation CFM pour la Recherche-ENS, and from the French National Research Agency (ANR) grant PAIL.

References

Advani, M. and Saxe, A. M. High-dimensional dynamics of generalization error in neural networks. *arXiv:1710.03667*, 2017.

Advani, M. S. and Ganguli, S. Statistical mechanics of optimal convex inference in high dimensions. *Physical Review X*, 6(3):1–16, 2016.

Arpit, D., Jastrz, S., Kanwal, M. S., Maharaj, T., Fischer, A., Courville, A., and Bengio, Y. A Closer Look at Memorization in Deep Networks. In *Proceedings of the 34th International Conference on Machine Learning*, 2017.

Aubin, B., Maillard, A., Barbier, J., Krzakala, F., Macris, N., and Zdeborová, L. The committee machine: Computational to statistical gaps in learning a two-layers neural network. In *Advances in Neural Information Processing Systems 31*, pp. 3227–3238, 2018.

Baity-Jesi, M., Sagun, L., Geiger, M., Spigler, S., Arous, G. B., Cammarota, C., LeCun, Y., Wyart, M., and Biroli, G. Comparing Dynamics: Deep Neural Networks versus Glassy Systems. In *Proceedings of the 35th International Conference on Machine Learning*, 2018.

Bartlett, P. L. and Mendelson, S. Rademacher and Gaussian complexities: Risk bounds and structural results. *Journal of Machine Learning Research*, 3(3):463–482, 2003.

Biehl, M. and Schwarze, H. Learning by on-line gradient descent. *J. Phys. A. Math. Gen.*, 28(3):643–656, 1995.

Chaudhari, P., Choromanska, A., Soatto, S., LeCun, Y., Baldassi, C., Borgs, C., Chayes, J., Sagun, L., and Zecchina, R. Entropy-SGD: Biasing Gradient Descent Into Wide Valleys. In *ICLR*, 2017.

Chizat, L. and Bach, F. On the global convergence of gradient descent for over-parameterized models using optimal transport. In *Advances in Neural Information Processing Systems 31*, pp. 3040–3050, 2018.

Engel, A. and Van den Broeck, C. *Statistical Mechanics of Learning*. Cambridge University Press, 2001.

Gardner, E. and Derrida, B. Three unfinished works on the optimal storage capacity of networks. *Journal of Physics A: Mathematical and General*, 22(12):1983–1994, 1989.

Kinzel, W., Ruján, P., and Rujan, P. Improving a Network Generalization Ability by Selecting Examples. *EPL (Europhysics Letters)*, 13(5):473–477, 1990.

Krogh, A. and Hertz, J. A. Generalization in a linear perceptron in the presence of noise. *Journal of Physics A: Mathematical and General*, 25(5):1135–1147, 1992.

LeCun, Y., Bengio, Y., and Hinton, G. E. Deep learning. *Nature*, 521(7553):436–444, 2015.

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.

Mei, S., Montanari, A., and Nguyen, P.-M. A mean field view of the landscape of two-layer neural networks. *Proceedings of the National Academy of Sciences*, 115(33): E7665–E7671, 2018.

Mohri, M., Rostamizadeh, A., and Talwalkar, A. *Foundations of Machine Learning*. MIT Press, 2012.

Neyshabur, B., Tomioka, R., and Nathan Srebro. In search of the real inductive bias: On the role of implicit regularization in deep learning. In *ICLR*, 2015.

Oja, E. and Karhunen, J. On stochastic approximation of the eigenvectors and eigenvalues of the expectation of a random matrix. *Journal of Mathematical Analysis and Applications*, 106(1):69–84, 1985.

Rotskoff, G. M. and Vanden-Eijnden, E. Parameters as interacting particles: long time convergence and asymptotic error scaling of neural networks. In *Advances in neural information processing systems 31*, pp. 7146–7155, 2018.

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

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

Saad, D. and Solla, S. A. Learning with Noise and Regularizers Multilayer Neural Networks. In *Advances in Neural Information Processing Systems 9*, pp. 260–266, 1997.

Saxe, A. M., McClelland, J. L., and Ganguli, S. Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. In *ICLR*, 2014.

Seung, H. S., Sompolinsky, H., and Tishby, N. Statistical mechanics of learning from examples. *Physical Review A*, 45(8):6056–6091, 1992.

Simonyan, K. and Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. In *International Conference on Learning Representations*, 2015.

Wang, C., Mattingly, J., and Lu, Y. M. Scaling Limit: Exact and Tractable Analysis of Online Learning Algorithms with Applications to Regularized Regression and PCA. *arXiv:1712.04332*, 2017.

Wang, C., Hu, H., and Lu, Y. M. A Solvable High-Dimensional Model of GAN. *arXiv:1805.08349*, 2018.

Watkin, T. L. H., Rau, A., and Biehl, M. The statistical mechanics of learning a rule. *Reviews of Modern Physics*, 65(2):499–556, 1993.

Zdeborová, L. and Krzakala, F. Statistical physics of inference: thresholds and algorithms. *Adv. Phys.*, 65(5): 453–552, 2016.

Zhang, C., Bengio, S., Hardt, M., Recht, B., and Vinyals, O. Understanding deep learning requires rethinking generalization. In *ICLR*, 2017.

Supplemental material

A. Derivation of the ODE description of the generalisation dynamics of online learning

We will now show how to derive ODEs that describe the dynamics of online learning in two-layer neural networks, following the seminal work by Biehl and Schwarze (Biehl & Schwarze, 1995) and Saad and Solla (Saad & Solla, 1995a;b). We focus on the teacher-student setup introduced in the main paper, where a student network with weights $w \in \mathbb{R}^{K\times N}$ and output

$$\phi(w,x) = \sum_{k=1}^{K} g\left(\frac{w_k x}{\sqrt{N}}\right) \tag{S1}$$

is trained on samples (x^μ, y^μ) generated by another two-layer network with weights $B \in \mathbb{R}^{M\times N}$, the teacher, according to

$$y_B^\mu(x^\mu) \equiv \phi(B, x^\mu) + \zeta^\mu. \tag{S2}$$

Here, ζ^μ is normally distributed with mean 0 and variance σ^2. We will make two technical assumptions, namely having a large network ($N \to \infty$) and a data set that is large enough to allow that we visit every sample only once before training converges.

A.1. Expressing the generalisation error in terms of order parameters

To make this section self-consistent, we briefly recapitulate how the assumptions stated above allow to rewrite the generalisation error in terms of a number of *order parameters*. We have

$$\epsilon_g \equiv \frac{1}{2}\left\langle \left[\phi(w,x) - \phi(B,x)\right]^2 \right\rangle \tag{S3}$$

$$= \frac{1}{2}\left\langle \left[\sum_{k=1}^{K} g\left(\lambda_k^\mu\right) - \sum_{m=1}^{M} g(\nu_m^\mu)\right]^2 \right\rangle, \tag{S4}$$

where we have introduced the local fields

$$\lambda_k^\mu \equiv \frac{w_k x^\mu}{\sqrt{N}}, \tag{S5}$$

$$\nu_m^\mu \equiv \frac{B_m x^\mu}{\sqrt{N}}. \tag{S6}$$

Here and throughout this paper, we will use the indices $i, j, k, \ldots$ to refer to hidden units of the student, and indices $n, m, \ldots$ to denote hidden units of the teacher. Since the input x^μ only appears in ϵ_g only via products with the weights of the teacher and the student, we can replace the high-dimensional average $\langle\cdot\rangle$ over the input distribution $p(x)$ by an average over the $K + M$ local fields λ_k^μ and ν_m^μ. The assumption that the training set is large enough to allow that we visit every sample in the training set only once guarantees that the inputs and the weights of the networks are uncorrelated. Taking the limit $N \to \infty$ ensures that the local fields are jointly normally distributed with mean zero ($\langle x_n\rangle = 0$). Their covariance is also easily found: writing w_{ka} for the ath component of the kth weight vector, we have

$$\langle \lambda_k \lambda_l \rangle = \frac{\sum_{a,b}^{N} w_{ka} w_{lb} \langle x_a x_b\rangle}{N} = \frac{w_k w_l}{N} \equiv Q_{kl}, \tag{S7}$$

since $\langle x_a x_b\rangle = \delta_{ab}$. Likewise, we define

$$\langle \nu_n \nu_m\rangle = \frac{B_n B_m}{N} \equiv T_{nm}, \quad \langle \lambda_k \nu_m\rangle = \frac{w_k B_m}{N} \equiv R_{km}. \tag{S8}$$

The variables R_{in}, Q_{ik}, and T_{nm} are called *order parameters* in statistical physics and measure the overlap between student and teacher weight vectors w_i and B_n and their self-overlaps, respectively. Crucially, from Eq. (S4) we see that they are sufficient to determine the generalisation error ϵ_g. We can thus write the generalisation error as

$$\epsilon_g = \frac{1}{2}\sum_{i,k} I_2(i,k) + \frac{1}{2}\sum_{n,m} I_2(n,m) - \sum_{i,n} I_2(i,n), \tag{S9}$$

where we have defined

$$I_2(i,k) \equiv \langle g(\lambda_i) g(\lambda_k)\rangle = \frac{2}{\pi}\arcsin\frac{Q_{ik}}{\sqrt{1+Q_{ii}}\sqrt{1+Q_{kk}}}. \tag{S10}$$

The average in Eq. (S10) is taken over a normal distribution for the local fields λ_i and λ_k with mean $(0,0)$ and covariance matrix

$$C_2 = \begin{pmatrix} Q_{ii} & Q_{ik} \\ Q_{ik} & Q_{kk} \end{pmatrix}. \tag{S11}$$

Since we are using the indices $i, j, \ldots$ for student units and $n, m, \ldots$ for teacher hidden units, we have

$$I_2(i,n) = \langle g(\lambda_i) g(\nu_m)\rangle, \tag{S12}$$

where the covariance matrix of the joint of distribution λ_i and ν_m is given by

$$C_2 = \begin{pmatrix} Q_{ii} & R_{in} \\ T_{in} & T_{nn} \end{pmatrix}. \tag{S13}$$

and likewise for $I_2(n,m)$. We will use this convention to denote integrals throughout this section. For the generalisation error, this means that it can be expressed in terms of the order parameters alone as

$$\epsilon_g = \frac{1}{\pi}\sum_{i,k}\arcsin\frac{Q_{ik}}{\sqrt{1+Q_{ii}}\sqrt{1+Q_{kk}}} + \frac{1}{\pi}\sum_{n,m}\arcsin\frac{T_{nm}}{\sqrt{1+T_{nn}}\sqrt{1+T_{mm}}} - \frac{2}{\pi}\sum_{i,n}\arcsin\frac{R_{in}}{\sqrt{1+Q_{ii}}\sqrt{1+T_{nn}}}. \quad \text{(S14)}$$

Note that our results for large N are valid for any input distributions that has the same mean and variance, for example equiprobable binary inputs $x_n = \pm 1$.

A.2. ODEs for the evolution of the order parameters

Expressing the generalisation error in terms of the order parameters as we have in Eq. (S14) is of course only useful if we can track the evolution of the order parameters over time. We can derive ODEs that allow us to do precisely that by first writing again the SGD update of the weights:

$$w_k^{\mu+1} = w_k^\mu - \frac{\kappa}{N}w_k^\mu - \frac{\eta}{\sqrt{N}}x^\mu r_k^\mu, \quad \text{(S15)}$$

where μ is a running index counting the weight updates or, equivalently, the samples used so far, and

$$r_k^\mu \equiv g'(\lambda_k^\mu)\left[\phi(w,x^\mu) - y_B^\mu\right]. \quad \text{(S16)}$$

From this equation, we can obtain differential equations for the time evolution of the order parameters Q by squaring the weight update (S15) and for R taking the inner product of (S15) with B_n, respectively, which yields the Eqns. (12) of the main text and which we state again for completeness:

$$\frac{\mathrm{d}R_{in}}{\mathrm{d}\alpha} = -\kappa R_{in} + \eta\langle r_i\nu_n\rangle \quad \text{(S17a)}$$

$$\frac{\mathrm{d}Q_{ik}}{\mathrm{d}\alpha} = -2\kappa Q_{ik} + \eta\langle r_i\lambda_k\rangle + \eta\langle r_k\lambda_i\rangle + \eta^2\langle r_i r_k\rangle + \eta^2\sigma^2\langle g'(\lambda_i)g'(\lambda_k)\rangle \quad \text{(S17b)}$$

where $\alpha = \mu/N$ becomes a continuous time-like variable in the limit $N\to\infty$. These equations are valid for any choice of activation functions g_1 and g_2. To make progress however, *i.e.* to obtain a closed set of differential equations for Q and R, we need to evaluate the averages $\langle\cdot\rangle$ over the local fields. In particular, we have to compute three types of averages:

$$I_3 = \langle g'(a)bg'(c)\rangle, \quad \text{(S18)}$$

where a is one the local fields of the student, while b and c can be local fields of either the student or the teacher;

$$I_4 = \langle g'(a)g'(b)g(c)g(d)\rangle, \quad \text{(S19)}$$

where a and b are local fields of the student, while c and d can be local fields of both; and finally

$$J_2 = \langle g'(a)g'(b)\rangle, \quad \text{(S20)}$$

where a and b are local fields of the teacher. In each of these integrals, the average is taken with respect to a multivariate normal distribution for the local fields with zero mean and a covariance matrix whose entries are chosen in the same way as discussed for I_2.

We can re-write Eqns. (S17) with these definitions in a more explicit form as (Saad & Solla, 1995a;b)

$$\frac{\mathrm{d}R_{in}}{\mathrm{d}\alpha} = -\kappa R_{in} + \eta\left(\sum_m I_3(i,n,m) - \sum_j I_3(i,n,j)\right), \quad \text{(S21)}$$

$$\frac{\mathrm{d}Q_{ik}}{\mathrm{d}\alpha} = -2\kappa Q_{ik} + \eta^2\sigma^2 J_2(i,k) + \eta\left(\sum_m I_3(i,k,m) - \sum_j I_3(i,k,j)\right) + \eta\left(\sum_m I_3(k,i,m) - \sum_j I_3(k,i,j)\right) + \eta^2\left(\sum_{n,m} I_4(i,k,n,m) - 2\sum_{j,n} I_4(i,k,j,n) + \sum_{j,l} I_4(i,k,j,l)\right). \quad \text{(S22)}$$

The explicit form of the integrals $I_2(\cdot)$, $I_3(\cdot)$, $I_4(\cdot)$ and $J_2(\cdot)$ is given in Sec. H for the case $g(x) = \mathrm{erf}\left(x/\sqrt{2}\right)$. Solving these equations numerically for Q and R and substituting their values in to the expression for the generalisation error (S9) gives the full generalisation dynamics of the student. We show the resulting learning curves together with the result of a single experiment in Fig. 2 of the main text. We have bundled our simulation software and our ODE integrator as a user-friendly Python package[2]. In Sec. B, we discuss how to extract information from them in an analytical way.

B. Calculation of ϵ_g in the limit of small noise

Our aim is to understand the asymptotic value of the generalisation error

$$\epsilon_g^* \equiv \lim_{\alpha\to\infty}\epsilon_g(\alpha). \quad \text{(S23)}$$

We focus on students that have more hidden units than the teacher, $K \geq M$. These students are thus over-parameterised *with respect to the generative model of the*

[2]To download, visit https://github.com/sgoldt/pyscm

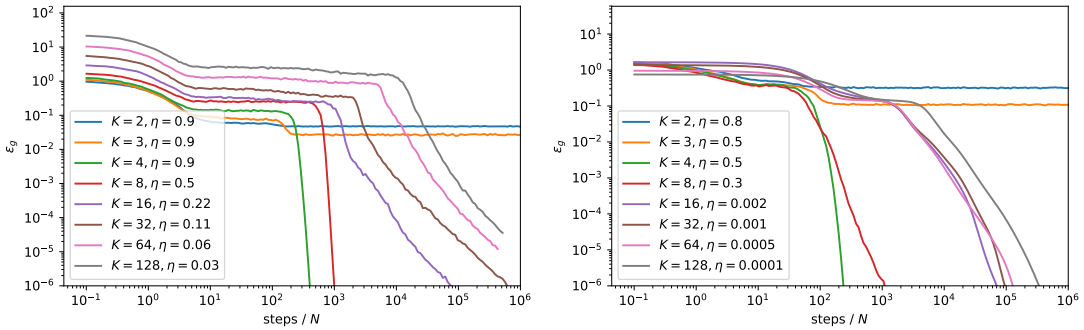

Figure S1. **Over-parametrised networks with sigmoidal or ReLU activations learn perfectly from a noiseless teacher.** The generalisation dynamics for students with sigmoidal (left) and ReLU activation function (right) for various K learning from a teacher with $M = 4$ is shown. In all cases, the generalisation error eventually decays exponentially towards zero. ($N = 784$)

data and we define

$$L \equiv K - M \tag{S24}$$

as the number of additional hidden units in the student network. In this section, we focus on the sigmoidal activation function

$$g(x) = \mathrm{erf}\left(x/\sqrt{2}\right), \tag{S25}$$

unless stated otherwise.

Eqns. (S21) are a useful tool to analyse the generalisation dynamics and they allowed Saad and Solla to gain plenty of analytical insight into the special case $K = M$ (Saad & Solla, 1995a;b). However, they are also a bit unwieldy. In particular, the number of ODEs that we need to solve grows with K and M as $K^2 + KM$. To gain some analytical insight, we make use of the symmetries in the problem, *e.g.* the permutation symmetry of the hidden units of the student, and re-parametrised the matrices Q_{ik} and R_{in} in terms of eight order parameters that obey a set of self-consistent ODEs for any $K > M$. We choose the following parameterisation with eight order parameters:

$$Q_{ij} = \begin{cases} Q & i = j \leq M, \\ C & i \neq j;\ i, j \leq M, \\ D & i > M, j \leq M \quad \text{or} \quad i \leq M, j > M, \\ E & i = j > M, \\ F & i \neq j;\ i, j > M, \end{cases} \tag{S26}$$

$$R_{in} = \begin{cases} R & i = n, \\ S & i \neq n;\ i \leq M, \\ U & i > M, \end{cases} \tag{S27}$$

which in matrix form for the case $M = 3$ and $K = 5$ read:

$$R = \begin{pmatrix} R & S & S \\ S & R & S \\ S & S & R \\ U & U & U \\ U & U & U \end{pmatrix} \quad \text{and} \quad Q = \begin{pmatrix} Q & C & C & D & D \\ C & Q & C & D & D \\ C & C & Q & D & D \\ D & D & D & E & F \\ D & D & D & F & E \end{pmatrix} \tag{S28}$$

We choose this number of order parameters and this particular setup for the overlap matrices Q and R for two reasons: it is the smallest number of variables for which we were able to self-consistently close the equations of motion (S21), and they agree with numerical evidence obtained from integrating the full equations of motion (S21).

By substituting this ansatz into the equations of motion (S21), we find a set of eight ODEs for the order parameters. These equations are rather unwieldy and some of them do not even fit on one page, which is why we do not print them here in full; instead, we provide a *Mathematica* notebook where they can be found and interacted with[3]. These equations allow for a detailed analysis of the effect of over-parameterisation on the asymptotic performance of the student, as we will discuss now.

B.1. Heavily over-parameterised students can learn perfectly from a noiseless teacher using online learning

For a teacher with $T_{nm} = \delta_{nm}$ and in the absence of noise in the teacher's outputs ($\sigma = 0$), there exists a fixed point of the ODEs with $R = Q = 1$, $C = D = E = F = 0$, and perfect generalisation $\epsilon_g = 0$. Online learning will find this fixed point, as is demonstrated in Fig. S1, where we plot the generalisation dynamics of a student with K hidden units

[3]For the duration of the review process, we have appended the contents of the Mathematica notebook to this supplementary material.

learning from a teacher with $M = 4$ hidden units for both Erf and ReLU activation functions. More precisely, after a plateau whose length depends on the size of the network for the sigmoidal network, the generalisation error eventually begins an exponential decay to the optimal solution with zero generalisation error. The learning rates are chosen such that learning converges, but aren't optimised otherwise.

B.2. Perturbative solution of the ODEs

We have calculated the asymptotic value of the generalisation error ϵ_g^* for a teacher with $T_{nm} = \delta_{nm}$ to first order in the variance of the noise σ^2. To do so, we performed a perturbative expansion around the fixed point

$$R_0 = Q_0 = 1, \tag{S29}$$

$$S_0 = U_0 = C_0 = D_0 = E_0 = F_0 = 0, \tag{S30}$$

with the ansatz

$$X = X_0 + \sigma^2 X_1 \tag{S31}$$

for all the order parameters. Writing the ODEs to first order σ^2 and solving for their steady state where $X'(\alpha) = 0$ yielded a fixed point with an asymptotic generalisation error

$$\epsilon_g^* = \frac{\sigma^2 \eta}{2\pi} f(M, L, \eta) + \mathcal{O}(\sigma^3). \tag{S32}$$

$f(M, L, \eta)$ is an unwieldy rational function of its variables. Due to its length, we do not print it here in full; instead, we give the full function in a *Mathematica* notebook, which for the duration of the review process is appended to this supplementary material. Here, we plot the results in various forms in Fig. S2. We note in particular the following points:

ϵ_g^* increases with L, η The two plots on the left show that the generalisation error increases monotonically with both L and η while keeping the other fixed, respectively, for teachers with $M = 2$ (red) and $M = 16$ (blue)

The role of the learning rate η Mitigating this effect by decreasing the learning rate η for larger students raises two problems: a lower learning rate yields longer training times, and longer training times imply that more data is required until convergence. This is in agreement with statistical learning theory, where models with more parameters generalise just as well as smaller ones given enough data, despite having a higher complexity class as measured by VC dimension or Rademacher complexity (Mohri et al., 2012), for example. Furthermore, we show in Sec. B.2 that even with $\eta \sim 1/K$, the generalisation error increases with L before plateauing at a constant value. Moreover, we show in Fig. S4 that the asymptotic generalisation error (S32) of a student trained using SGD with learning rate $\eta = 1/K$ still increases with L before plateauing at a constant value that is independent of M.

Divergence at large η Our perturbative result diverges for large L, or equivalently, for a large learning rate that depends on the number of hidden units $L \sim K$. For the special case $K = M$, the learning rate η_{div} at which our perturbative result diverges is precisely the maximum learning rate η_{max} for which the exponential convergence to the optimal solution is still guaranteed for $\sigma = 0$ (Saad & Solla, 1995b)

$$\eta_{\mathrm{max}} = \frac{\sqrt{3}\pi}{M + 3/\sqrt{5} - 1} \tag{S33}$$

as we show in the right-most plot of Fig. S2.

Expansion for small η In the limit of small learning rates, which is the most relevant in practice and which from the plots in Fig. S2 dominates the behaviour of ϵ_g^* outside of the divergence, the generalisation error is linear in the learning rate. Expanding ϵ_g^* to first order in the learning rate reveals a particularly revealing form,

$$\epsilon_g^* = \frac{\sigma^2 \eta}{2\pi} \left(L + \frac{M}{\sqrt{3}} \right) + \mathcal{O}(\eta^2) \tag{S34}$$

with second-order corrections that are quadratic in L. This is actually the sum of the asymptotic generalisation errors of M continuous perceptrons that are learning from a teacher with $T = 1$ and L continuous perceptrons with $T = 0$ as we calculate in Sec. C. This neat result is a consequence of the specialisation that is typical of SCMs with sigmoidal activation functions as we discussed in the main text.

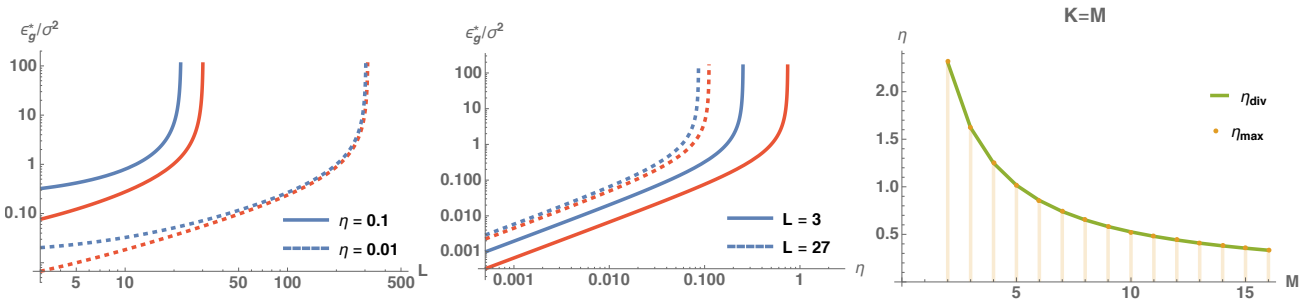

Figure S2. **The final generalisation error of over-parameterised Erf networks scales linearly with the learning rate, the variance of the teacher's output noise, and L.** We plot ϵ_g^*/σ^2 in the limit of small noise, Eq. (S32), for $M = 2$ (red) and $M = 16$ (blue). It is clear that generalisation error increases with the number of superfluous units L at fixed learning rate (*left*) and the learning rate η (*middle*). *Right:* For $K = M$, the learning rate $\eta_{\rm div}$ at which our perturbative result diverges is precisely the maximum learning rate $\eta_{\rm max}$ at which the exponential convergence to the optimal solution is guaranteed for $\sigma = 0$, Eq. (S33)

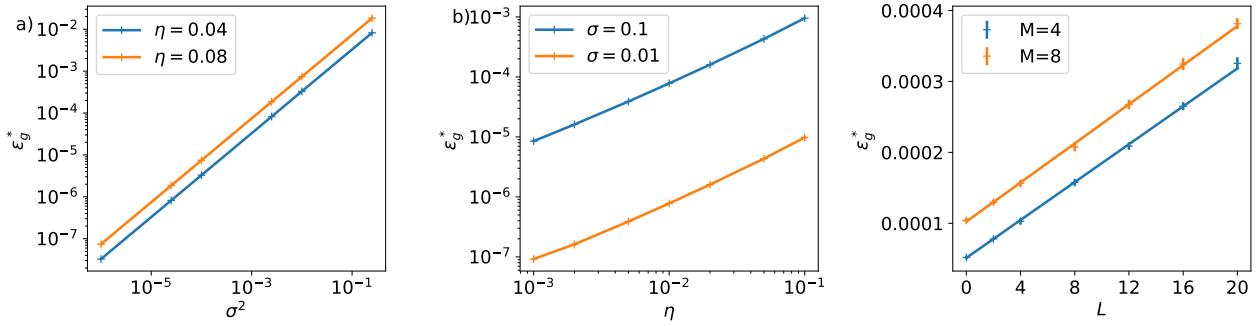

Figure S3. **The final generalisation error of over-parametrised ReLU networks scales as $\epsilon_g^* \sim \eta\sigma^2 L$.** Simulations confirm that the asymptotic generalisation error ϵ_g^* of a ReLU student learning from a ReLU teacher scales with the learning rate η, the variance of the teacher's output noise σ^2 and the number of additional hidden units as $\epsilon_g \sim \eta\sigma^2 L$, which is the same scaling as the one found analytically for sigmoidal networks in Eq. (S34). Straight lines are linear fits to the data, with slope 1 in (a) and (b). Parameters: $M = 2, K = 6$ (a, b) and $M = 4, 16$; $K = M + L$ (c); in all figures, $N = 784, \kappa = 0$.

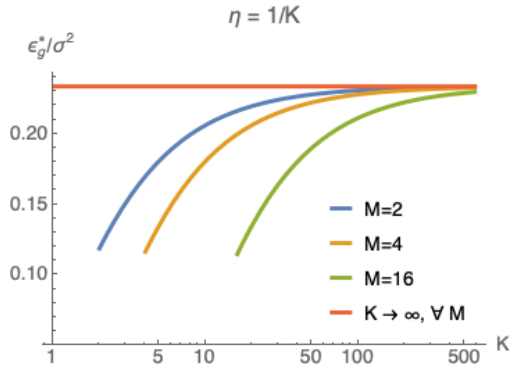

Figure S4. **Asymptotic generalisation error for sigmoidal networks with learning rate $\eta = 1/K$.** We plot the asymptotic generalisation error ϵ_g^* (S32) over σ^2 of a student with a varying number of hidden units trained on data generated by teachers with $M = 2, 4, 16$ using SGD with learning rate $1/K$. The generalisation error still increases with K, before plateauing at a constant value that is independent of M. $\kappa = 0$.

C. Asymptotic generalisation error of a noisy continuous perceptron

What is the asymptotic generalisation for a continuous perceptron, *i.e.* a network with $K = 1$, in a teacher-student scenario when the teacher has some additive Gaussian output noise? In this section, we repeat a calculation by Biehl & Schwarze (1995) where the teacher's outputs are given by

$$y_B = g\left(\frac{Bx}{\sqrt{N}}\right) + \zeta \tag{S35}$$

where ζ is again a Gaussian r.v. with mean 0 and variance σ^2. We keep denoting the weights of the student by w and the weights of the teacher by B. To analyse the generalisation dynamics, we introduce the order parameters

$$R \equiv \frac{wB}{N}, \qquad Q \equiv \frac{ww}{N} \quad \text{and} \quad T \equiv \frac{BB}{N}. \tag{S36}$$

and we explicitly do not fix T for the moment. For $g(x) = \operatorname{erf}\left(x/\sqrt{2}\right)$, they obey the following equations of motion:

$$\frac{dR}{dt} = \frac{2\eta}{\pi\left(Q(t)+1\right)}\left(\frac{TQ(t) - R(t)^2 + T}{\sqrt{(T+1)Q(t) - R(t)^2 + T + 1}} - \frac{R(t)}{\sqrt{2Q(t)+1}}\right) \tag{S37}$$

$$\begin{aligned}\frac{dQ}{dt} =& \frac{4\eta}{\pi(Q(t)+1)}\left(\frac{R(t)}{\sqrt{2(Q(t)+1) - R(t)^2}} - \frac{Q(t)}{\sqrt{2Q(t)+1}}\right) \\ &+ \frac{4\eta^2}{\pi^2\sqrt{2Q(t)+1}}\left[-2\arcsin\left(\frac{R(t)}{\sqrt{(6Q(t)+2)(2Q(t) - R(t)^2+1)}}\right)\right. \\ &\left. + \arcsin\left(\frac{2\left(Q(t) - R(t)^2\right)+1}{2\left(2Q(t) - R(t)^2+1\right)}\right) + \arcsin\left(\frac{Q(t)}{3Q(t)+1}\right)\right] \\ &+ \frac{2\eta^2\sigma^2}{\pi\sqrt{2Q(t)+1}}.\end{aligned} \tag{S38}$$

The equations of motion have a fixed point at $Q = R = T$ which has perfect generalisation for $\sigma = 0$. We hence make a perturbative ansatz in σ^2

$$Q(t) = T + \sigma^2 q(t) \tag{S39}$$

$$R(t) = T + \sigma^2 r(t) \tag{S40}$$

and find for the asymptotic generalisation error

$$\epsilon_g^* = \frac{\eta\sigma^2(4T+1)}{2\sqrt{2T+1}\left(-\eta\sqrt{8T^2+6T+1} + 4\pi T + \pi\right)} + \mathcal{O}\left(\sigma^3\right). \tag{S41}$$

To first order in the learning rate, this reads

$$\epsilon_g^* = \frac{\eta\sigma^2}{2\pi\sqrt{2T+1}}, \tag{S42}$$

which should be compared to the corresponding result for the full SCMs, Eq. (S34).

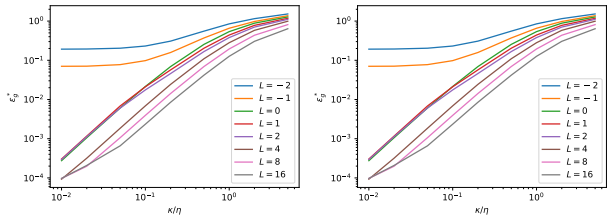

Figure S5. **Weight decay.** We plot the final generalisation error ϵ_g^* of a student with a varying number of hidden units trained on data generated by a teacher with $M = 4$ using SGD with weight decay. The generalisation error clearly increases with the weight decay constant κ. Parameters: $N = 784, \eta = 0.1, sigma = 0.01$.

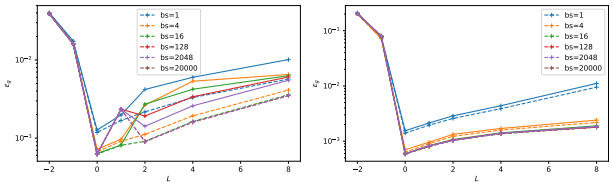

Figure S6. **SGD with mini-batches shows the same qualitative behaviour as online learning** We show the asymptotic generalisation error ϵ_g^* for students with sigmoidal (left) and ReLU activation function (right) for various K learning from a teacher with $M = 4$. Between the curves, we change the size of the mini-batch used at each step of SGD from 1 (online learning) to 20 000. Parameters: $N = 500, \eta = 0.2, \sigma = 0.1, \kappa = 0$.

D. Regularisation by weight decay does not help

A natural strategy to avoid the pitfalls of overfitting is to regularise the weights, for example by using explicit weight decay by choosing $\kappa > 0$. We have not found a setup where adding weight decay *improved* the asymptotic generalisation error of a student compared to a student that was trained without weight decay in our setup. As a consequence, weight decay completely fails to mitigate the increase of ϵ_g^* with L. We show the results of an illustrative experiment in Fig. S5.

E. SGD with mini-batches

One key characteristic of online learning is that we evaluate the gradient of the loss function using a single sample from the training step per step. In practice, it is more common to actually use a number of samples $b > 1$ to estimate the gradient at every step. To be more precise, the weight update

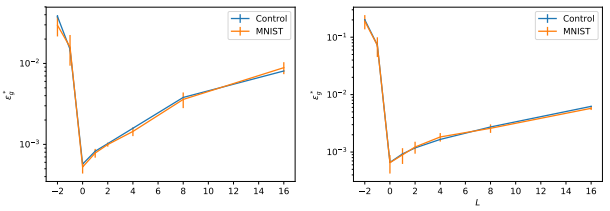

Figure S7. **Higher-order correlations in the input data do not play a role for the asymptotic generalisation.** We plot the final generalisation error ϵ_g^* after online learning of a student of various sizes from a teacher with $M = 4$ using Gaussian inputs (blue) and MNIST images (red) for training and testing. $N = 784, \eta = 0.1, \sigma = 0.1, \kappa = 0$.

equation for SGD with mini-batches would read:

$$w_k^{\mu+1} = w_k^\mu - \frac{\kappa}{N} w_k^\mu - \frac{\eta}{b\sqrt{N}} \sum_{\ell=1}^{b} x^{\mu,\ell} g'(\lambda_k^{\mu,\ell}) \left[\phi(w, x^{\mu,\ell}) - y_B^{\mu,\ell} \right]. \quad \text{(S43)}$$

where $x^{\mu,\ell}$ is the ℓth input from the mini-batch used in the mth step of SGD, $\lambda_k^{\mu,\ell}$ is the local field of the kth student unit for the ℓth sample in the mini-batch, etc. Note that when we use every sample only once during training, using mini-batches of size b increases the amount of data required by a factor b when keeping the number of steps constant.

We show the asymptotic generalisation error of student networks of varying size trained using SGD with mini-batches and a teacher with $M = 4$ in Fig. S6. Two trends are visible: first, using increasing the size of the mini-batches decreases the asymptotic generalisation error ϵ_g^* up to a certain mini-batch size, after which the gains in generalisation error become minimal; and second, the shape of the $\epsilon_g^* - L$ curve is the same for all mini-batch sizes, with the minimal generalisation error attained by a network with $K = M$.

F. Using MNIST images for training and testing

In the derivation of the ODE description of online learning for the main text, we noted that only the first two moments of the input distribution matter for the learning dynamics and for the final generalisation error. The reason for this is that the inputs only appear in the equations of motion for the order parameters as a product with the weights of either the teacher or the student. Now since they are – by assumption – uncorrelated with those weights, this product is the sum of large number of random variables and hence distributed by the central limit theorem.

We have checked how our results change when this assump-

tion breaks down in one example where we train a network on a finite data set with non-trivial higher order moments, namely the images of the MNIST data set. We studied the very same setup that we discuss throughout this work, namely the supervised learning of a regression task in the teacher-student scenario. We *only* replace the the inputs, which would have been i.i.d. draws from the standard normal distribution, with the images of the MNIST data set. In particular, this means that we do not care about the labels of the images. Figure S7 shows a plot of the resulting final generalisation against L for both the MNIST data set and a data set of the same size, comprised of i.i.d. draws from the standard normal distribution, which are in good agreement.

G. The scaling of ϵ^* with L for finite training sets

In practice, a single sample of the training data set will be visited several times during training. After a first pass through the training set, the online assumption that an incoming sample (x, y_B) is uncorrelated to the weights of the network thus breaks down. A complete analytical treatment in this setting remains an open problem, so to study this practically relevant setup, we turn to simulations. We keep the setup described in Secs. 1 and 2, but simply reduce the number of samples in the training data set P. Our focus is again on the final generalisation error after convergence ϵ_g^* for linear, sigmoidal and ReLU networks, which we plot from left to right as a function of L in Fig. S8.

Linear networks show a similar behaviour to the setup with a very large training set discussed in Sec. 3.2: the bigger the network, the worse the performance for both $P = 4$ and $P = 50$. Again, the optimal network has $K = 1$ hidden units, irrespective of the size of the teacher. However, for non-linear networks, the picture is more varied: For large training sets, where the number of samples easily outnumber the free parameters in the network ($P = 50$, red curve; this corresponds roughly to learning a data set of the size of MNIST), the behaviour is qualitatively described by our theory from Sec. 3: the best generalisation is obtained by a network that matches the teacher size, $K = M$. However, as we reduce the size of the training set, this is no longer true. For $P = 4$, for example, the best generalisation is obtained with networks that have $K > M$. Thus the size of the training set with respect to the network has an important influence on the scaling of ϵ_g^* with L. Note that the early-stopping generalisation error, which we define as the minimal generalisation error over the duration of training, shows qualitatively the same behaviour as ϵ_g^* (see supplementary material for additional information.)

G.1. Early-stopping generalisation error for finite training sets

A common way to prevent over-fitting of a neural network when training with a finite training set in practice is early stopping, where the training is stopped before the training error has converged to its final value yet. The idea behind early-stopping is thus to stop training before over-fitting sets in. For the purpose of our analysis of the generalisation of two-layer networks trained on a fixed finite data set in Sec. 4 of the main text, we define the early-stopping generalisation error $\hat{\epsilon}_g$ as the minimum of ϵ_g during the whole training process. In Fig. S8, we reproduce Fig. 6 from the main text at the bottom and plot $\hat{\epsilon}_g$ obtained from the very same experiments at the top. While the ReLU networks showed very little to no over-training, the sigmoidal networks showed more significant over-training. However, the qualitative dependence of the generalisation errors on L was observed to be the same in this experiment. In particular, the early-stopping generalisation error also shows two different regimes, one where increasing the network hurts generalisation ($P \gg K$), and one where it improves generalisation or at least doesn't seem to affect it much (small $P \sim K$).

H. Explicit form of the integrals appearing in the equations of motion of sigmoidal networks

To be as self-contained as possible, here we collect the explicit forms of the integrals I_2, I_3, I_4 and J_2 that appear in the equations of motion for the order parameters and the generalisation error for networks with $g(x) = \operatorname{erf}\left(x/\sqrt{2}\right)$, see Eq. (S21). They were first given by (Biehl & Schwarze, 1995; Saad & Solla, 1995a). Each average $\langle\cdot\rangle$ is taken w.r.t. a multivariate normal distribution with mean 0 and covariance matrix $C \in \mathbb{R}^n$, whose components we denote with small letters. The integration variables u, v are always components of λ, while w and z can be components of either λ or ν.

$$J_2 \equiv \langle g'(u)g'(v)\rangle = \frac{2}{\pi}\left(1 + c_{11} + c_{22} + c_{11}c_{22} - c_{12}^2\right)^{-1/2} \tag{S44}$$

$$I_2 \equiv \frac{1}{2}\langle g(w)g(z)\rangle = \frac{1}{\pi}\arcsin\frac{c_{12}}{\sqrt{1+c_{11}}\sqrt{1+c_{12}}}. \tag{S45}$$

$$I_3 \equiv \langle g'(u)wg(z)\rangle = \frac{2}{\pi}\frac{1}{\sqrt{\Lambda_3}}\frac{c_{23}(1+c_{11}) - c_{12}c_{13}}{1+c_{11}} \tag{S46}$$

$$I_4 \equiv \langle g'(u)g'(v)g(w)g(z)\rangle = \frac{4}{\pi^2}\frac{1}{\sqrt{\Lambda_4}}\arcsin\left(\frac{\Lambda_0}{\sqrt{\Lambda_1\Lambda 2}}\right) \tag{S47}$$

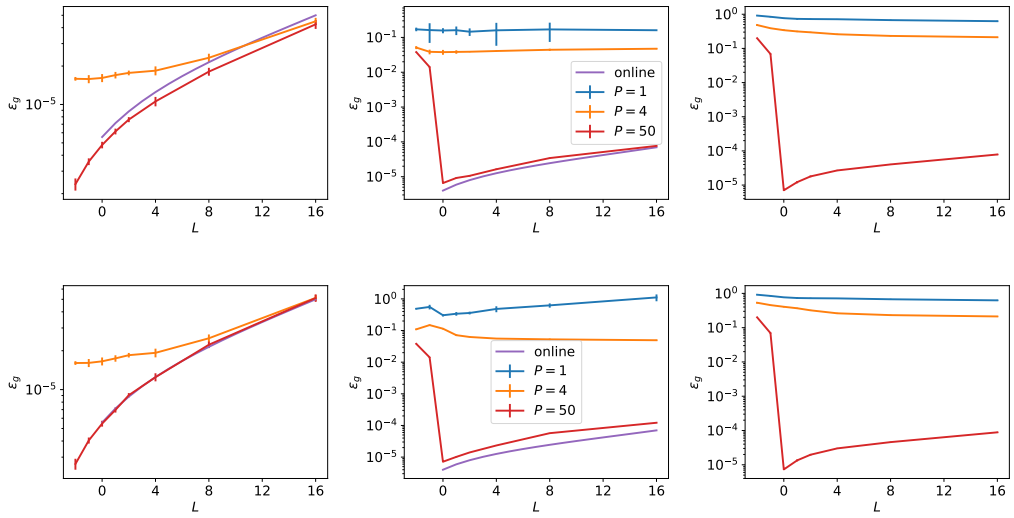

Figure S8. **The scaling of ϵ_g^* with L shows a similar dependence on the size of the training set for early-stopping (top) and final (bottom) generalisation error.** We plot the asymptotic and the early-stopping generalisation error after SGD with a finite training set containing PN samples (linear, sigmoidal and ReLU networks from left to right). The result for online learning for linear and sigmoidal networks, Eqns. (6) and (8) of the main text, are plotted in violet. Error bars indicate one standard deviation over 10 simulations, each with a different training set; many of them are too small to be clearly visible. Parameters: $N = 784, M = 4, \eta = 0.1, \sigma = 0.01$.

where

$$\Lambda_4 = (1 + c_{11})(1 + c_{22}) - c_{12}^2 \tag{S48}$$

and

$$\begin{aligned}\Lambda_0 = {} & \Lambda_4 c_{34} - c_{23}c_{24}(1 + c_{11}) - c_{13}c_{14}(1 + c_{22}) \\ & + c_{12}c_{13}c_{24} + c_{12}c_{14}c_{23}\end{aligned} \tag{S49}$$

$$\begin{aligned}\Lambda_1 = {} & \Lambda_4(1 + c_{33}) - c_{23}^2(1 + c_{11}) - c_{13}^2(1 + c_{22}) \\ & + 2c_{12}c_{13}c_{23}\end{aligned} \tag{S50}$$

$$\begin{aligned}\Lambda_2 = {} & \Lambda_4(1 + c_{44}) - c_{24}^2(1 + c_{11}) - c_{14}^2(1 + c_{22}) \\ & + 2c_{12}c_{14}c_{24}\end{aligned} \tag{S51}$$