The Role of the Learning Rate in Layered Neural Networks with ReLU Activation Function

Otavio Citton[1,2], Frederieke Richert[1] and Michael Biehl[1] *

1 - Bernoulli Institute for Mathematics, Computer Science and Artificial Intelligence, University of Groningen - The Netherlands

2 - Groningen Cognitive Systems and Materials Center (CogniGron), University of Groningen, Groningen, The Netherlands

Abstract. Using the statistical physics framework, we study the online learning dynamics in a particular case of shallow feed-forward neural networks with ReLU activation. By expanding the activation function in terms of Hermite polynomials we derive analytical results for the evolution of order parameters for any learning rate. Moreover, we compare our results with online gradient descent simulations and show how our method describes the typical learning curves. We also present results on how the learning rate affects the overall behavior of the network and its equilibria, showing different learning regimes and critical values of the learning rate.

1 Introduction

Neural networks are a central piece in the ongoing artificial intelligence revolution. Given the broad range of applications of these models, understanding their underlying mechanisms is instrumental not only to improve their performance but also to use them responsibly and to be aware of their limitations.

Statistical mechanics techniques borrowed from physics have proven useful in the study of neural networks [1, 2, 3, 4]. Here, we apply this framework to analyse the online dynamics of shallow feed-forward networks. Our focus is in the investigation of how networks with Rectifier Linear Unit (ReLU) activation behave when compared to the classical sigmoidal activation. The study of ReLU is of particular interest because it is widely used in applications and its definition is supposedly based on biological motivation. Works like [5, 6] show that these two activation functions produce fundamentally different results, which might have strong implications for applications.

So far, the analysis of online learning in shallow neural networks with ReLU activation in the statistical physics framework [7, 8] was restricted to small learning rates, η. The low learning rate regime results in a rescaling of the *training time* and produces results independent of η. Thus, to understand the role of realistic learning rates, the limit $\eta \to 0$ cannot be exploited.

Here we apply the representation proposed in [6], expanding the activation function in terms of Hermite polynomials. This enables us to determine the learning dynamics of a Soft Committee Machine (SCM) with ReLU activation for arbitrary learning rate. In addition, we compare our results with simulations of online gradient descent.

*This work is funded by NWO M1 grant OCENW.M20.287 and the Groningen Cognitive Systems and Materials Center (CogniGron).

2 Methods

The SCM is a special case of a two-layered fully connected feed-forward neural network, where only the weights connecting the input to the hidden layer are adaptive. These learnable parameters are denoted by $\mathbf{w} = \{\mathbf{w}_i \in \mathbb{R}^d\}_{i=1}^K$. For a given a d-dimensional input $\boldsymbol{\xi} \in \mathbb{R}^d$, the output of the network is given by $\sigma(\boldsymbol{\xi}, \mathbf{w}) = \sum_{i=1}^K g(x_i)$ with $x_i = \mathbf{w}_i \cdot \boldsymbol{\xi}$, where g is a non-linear activation function, that here will be considered to be the ReLU, $g(x) = \max(0, x)$.

Consider the learning problem of approximating a rule $\boldsymbol{\xi} \mapsto \tau(\boldsymbol{\xi})$. For the sake of mathematical modeling we assume that this rule can be fully realized in the space of possible SCMs, that is, we assume that there exists a *teacher* network with parameters $\mathbf{w}^* = \{\mathbf{w}_n^* \in \mathbb{R}^d\}_{n=1}^M$ and output $\tau(\boldsymbol{\xi}) = \sum_{n=1}^M g(y_n)$ with $y_n = \mathbf{w}_n^* \cdot \boldsymbol{\xi}$. Note that this network can differ from the student network by having a different number of internal units M, but still with ReLU activation function.

We consider a loss or error function given by the square deviation of the outputs $\epsilon(\boldsymbol{\xi}, \mathbf{w}) = \frac{1}{2}\big[\sigma(\boldsymbol{\xi}, \mathbf{w}) - \tau(\boldsymbol{\xi})\big]^2$, and the performance of the network is measured by the generalization error: the average error over the input distribution $\epsilon_g(\mathbf{w}) = \langle \epsilon(\boldsymbol{\xi}, \mathbf{w}) \rangle_{\boldsymbol{\xi}}$. Assuming that the inputs have i.i.d. components, we can compute ϵ_g analytically, obtaining an expression that only depends on

$$\mathbf{C} = \begin{pmatrix} \mathbf{Q} & \mathbf{R} \\ \mathbf{R}^\top & \mathbf{T} \end{pmatrix}, \text{where the} \quad Q_{ik} = \mathbf{w}_i \cdot \mathbf{w}_k,\ R_{in} = \mathbf{w}_i \cdot \mathbf{w}_n^* \text{ and } T_{nm} = \mathbf{w}_n^* \cdot \mathbf{w}_m^*$$

play the role of order parameters. See [9] for the full form of $\epsilon_g(\mathbf{C})$.

2.1 Online learning dynamics

Let us consider the online gradient descent dynamics, where at each step only one example $(\boldsymbol{\xi}^\mu, \tau^\mu)$ is presented, and the update of the learnable parameters is given by

$$\mathbf{w}_k^{\mu+1} = \mathbf{w}_k^\mu - \tfrac{\eta}{d} \nabla_{\mathbf{w}_k} \epsilon(\boldsymbol{\xi}^\mu, \mathbf{w}^\mu),$$

where η denotes the learning rate and μ indexes the example.

It was shown in [3, 4] that using the above dynamics for the weight variables, the corresponding ordinary differential equations (ODE) for the order parameters is given by

$$\tfrac{\mathrm{d}R_{in}}{\mathrm{d}\alpha} = \eta \langle \delta_i y_n \rangle, \qquad \tfrac{\mathrm{d}Q_{ik}}{\mathrm{d}\alpha} = \eta \langle \delta_i x_k + \delta_k x_i \rangle + \eta^2 \langle \delta_i \delta_k \rangle,$$

where $\delta_i = g'(x_i)\Big(\sum_n g(y_n) - \sum_k g(x_k)\Big)$, and $\langle \cdot \rangle$ denotes averages over the $(K+M)$-dimensional normal distribution with zero mean and covariance $\mathbf{C}$. To obtain the ODE we define the "time" variable $\alpha = \mu/d$ as the scaled example number, take the thermodynamic limit $d \to \infty$, and obtain the averages in the r.h.s. using the self-averaging property of the order parameters for large systems. Saad & Solla showed in [4] that all averages can be written explicitly as follows,

where sums are over $m, n = 1, \ldots, M$ and $j, l = 1, \ldots, K$:

$$\begin{aligned}
\langle \delta_i y_n \rangle &= \textstyle\sum_m I_3(\mathbf{C}_{i,K+n,K+m}) - \sum_j I_3(\mathbf{C}_{i,K+n,j}) \\
\langle \delta_i x_k + \delta_k x_i \rangle &= \textstyle\sum_m \left[I_3(\mathbf{C}_{i,k,K+m}) + I_3(\mathbf{C}_{k,i,K+m}) \right] - \sum_j \left[I_3(\mathbf{C}_{i,k,j}) + I_3(\mathbf{C}_{k,i,j}) \right] \\
\langle \delta_i \delta_k \rangle &= \textstyle\sum_{n,m} I_4(\mathbf{C}_{i,k,K+n,K+m}) - 2 \sum_{j,n} I_4(\mathbf{C}_{i,k,j,K+n}) + \sum_{j,l} I_4(\mathbf{C}_{i,k,j,l})
\end{aligned}$$

with the averages I_3 and I_4 given by $I_3(\mathbf{A}) = \int g'(z_1) z_2 g(z_3) P(\mathbf{z}|\mathbf{A}) \mathrm{d}z_1 \mathrm{d}z_2 \mathrm{d}z_3$ and $I_4(\mathbf{A}) = \int g'(z_1) g'(z_2) g(z_3) g(z_4) P(\mathbf{z}|\mathbf{A}) \mathrm{d}z_1 \mathrm{d}z_2 \mathrm{d}z_3 \mathrm{d}z_4$, where $P(\mathbf{z}|\mathbf{A})$ represents a three- or four-dimensional Gaussian distribution with zero mean and covariance matrix $\mathbf{A}$.

$\mathbf{C}_{a,b,\ldots}$ represents a lower dimensional correlation matrix (three- or four-dimensional in our case), obtained from $\mathbf{C}$ by selecting the rows and columns corresponding to the elements $a, b, \ldots$.

2.2 Hermite polynomial representation

While integrals I_3 can be analytically calculated for the ReLU, the calculation of the I_4 is challenging and previous studies of neural networks with ReLU activation were limited to the small learning rate regime [8, 7]. In this work we extend the method introduced in [6] and propose a representation of I_4 in terms of a power series using Hermite polynomials.

The Hermite polynomials $\{H_n\}_{n\geq 0}$ form an orthogonal basis of the $L^2(\mathbb{R})$ Hilbert space with respect to the inner product $\langle f, g \rangle = \frac{1}{\sqrt{2\pi}} \int f(x) g(x) e^{-\frac{1}{2}x^2} \mathrm{d}x$: any function $f \in L^2(\mathbb{R})$, such that $\langle f, f \rangle < \infty$, can be expressed as a generalized Fourier series in this basis as: $f(x) = \sum_{n=0}^{\infty} \frac{\langle H_n, f \rangle}{n!} H_n(x)$. In particular, for the ReLU, $g(x) = \max(0, x)$, the coefficients can be obtained analytically:

$$\langle H_0, g \rangle = 1/\sqrt{2\pi},\ \langle H_1, g \rangle = 1/2, \quad \text{and} \quad \langle H_n, g \rangle = (-1)^n H_{n-2}(0)/\sqrt{2\pi} \quad \text{for } n \geq 2$$

where the $H_n(0) = (-1)^{n/2}(n-1)!!$ for even n, and zero otherwise.

I_4 can be calculated by means of the so called Mehler's kernel or, more precisely, its higher-dim. generalization, the Kibble-Slepian formula [10, 11]. This kernel is constructed using the Hermite polynomials and it allows us to decouple the variables of the multivariate Gaussian distribution, by paying the price of introducing an infinite series for each pair correlation present.

Using the four-dim. Kibble-Slepian formula, for a symmetric correlation matrix $\mathbf{\Sigma}$ with $\Sigma_{ij} = \delta_{i,j} + \rho_{ij}(1 - \delta_{i,j})$ and $|\rho_{ij}| < 1$, I_4 can be represented as

$$I_4(\mathbf{\Sigma}) = \left(\prod \textstyle\sum_{n_{ij}=0}^{\infty} \frac{\rho_{ij}^{n_{ij}}}{n_{ij}!} \right) \langle H_{n_1}, g' \rangle \langle H_{n_2}, g' \rangle \langle H_{n_3}, g \rangle \langle H_{n_4}, g \rangle , \tag{1}$$

where $n_i = \sum_{j \neq i} n_{ij}$, with $n_{ij} = n_{ji}$ and the product runs over all index pairs $(i, j) \in \{1, \ldots, 4\}^2$ satisfying $i < j$.

It is worth noting that the numerical implementation of the series (1) is not straightforward. First, when we approximate the series at a given order N we consider all sets of indices $\{n_{ij}\}$ satisfying $n_1, n_2 \leq N - 1$ and $n_3, n_4 \leq N$, such that we always have terms of the same order when expanding g and g'.

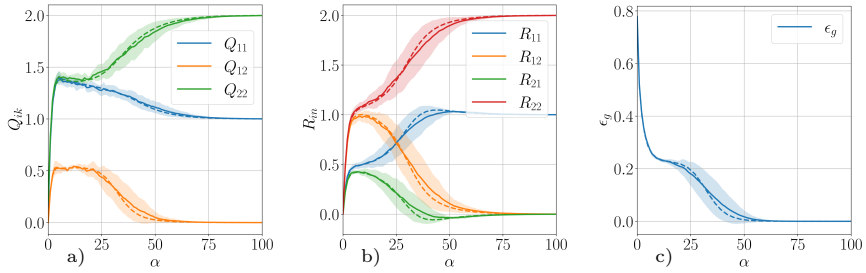

Figure 1: Learning curves. (a) time evolution of the order parameters $\{Q_{ik}\}$. (b) time evolution of the order parameters $\{R_{in}\}$. (c) time evolution of the generalization error ϵ_g. The dashed curves denote the ODE solution, and solid lines the average over 10 online gradient descent simulations with $d = 1000$ and $\eta = 1$ with the corresponding standard deviation.

Second, the series is only guaranteed to converge for $|\rho_{ij}| < 1$, so every time we encounter a singular matrix the expression above cannot be applied. Fortunately, in these cases, I_4 can be reduced to a three-dim. integral that can be calculated analytically.

3 Results

The method for the integration of the dynamics described here rely on the numerical truncation of the series (1) at a finite order. In [6] it was shown how the truncation error for a similar series decays with N, the maximum order of the expansion. In our case, an analytical expression for I_4 is unavailable so a similar analysis is not possible. Thus, to validate our results we compare them with online gradient descent simulations, with $\xi_i^\mu \sim \mathcal{N}(0, 1)$, for $i = 1, \ldots, d$.

Figure 1 shows a comparison between the learning curves obtained by numerically integrating the ODE and the average over 10 runs of online gradient descent simulations. The results correspond to a setting with a student network with $K = 2$ internal units learning from a graded teacher with $M = 2$ internal units and overlaps given by $T_{nm} = n\delta_{n,m}$, and learning rate $\eta = 1$. To compute I_4 we expand the activation function up to order $N = 10$ in the Hermite polynomial basis, and for the simulations $d = 1000$ was chosen as the input dimension. The initial condition for the order parameters are $Q_{ik}(0) = k \cdot 10^{-1}\delta_{i,k}$ and $R_{in}(0) = 10^{-3}\delta_{i,n}$. To achieve the same initial conditions in the simulations, we use the Gram-Schmidt based initialization scheme used by Straat & Biehl in [7], and implement a correction that takes into account finite size effects during the first steps of the dynamics, described in [12].

We found very different behaviors of the learning curves for different values of the learning rate η, depicted in Figure 2a. For small values of η we recover the results from the small learning rate limit [7], with the presence of a plateau. As we increase η, we notice a shrinking length and increasing height of the plateau,

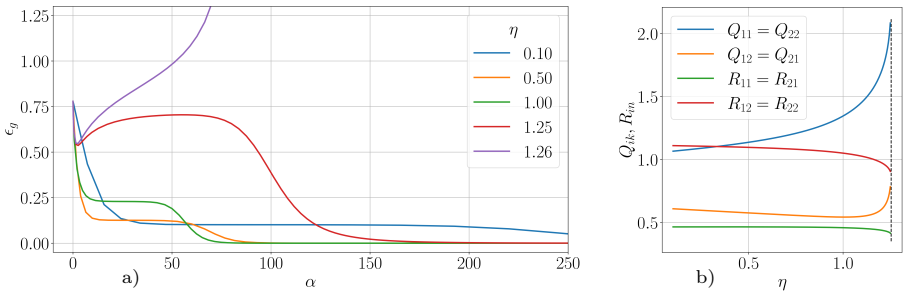

Figure 2: (a) shows the evolution of ϵ_g according to the ODE for different η. (b) shows the dependence of the symmetric plateau fixed point on η. The dashed line marks the learning rate above which the symmetric plateau ceases to exist.

similar to networks with sigmoidal activation, up to $\eta \approx 1.2175$. Above this value we find a regime of non-monotonic generalization error, with an interval of α where the network performance gets worse with more examples, resembling a double descent behavior that is not present in the same system with sigmoidal activation. For learning rates greater than a critical value $\eta_c \approx 1.2575$, the dynamics is characterized by an uncontrollable growth of the weight vector norms and, consequently, of the generalization error.

We can also study the fixed points of the dynamics and their respective stability. For a graded teacher and the number of internal units as described above, we found two stable fixed points that correspond to perfect alignment with the teacher vectors and, thus, present perfect generalization. We also have a fixed point responsible for the symmetric plateau observed in the dynamics. Figure 2b shows how this fixed point changes with the learning rate η. This is an unstable point with several (but not all) negative eigenvalues, implying that there are several directions that bring the order parameters very close to it before the positive eigenvalue sends the trajectories away from it, explaining the observed time spent in the plateau. This fixed point disappears at a learning rate $\eta \approx 1.2545$. There is yet another unstable fixed point with identical student weight vectors, but it is not visited with the initial conditions used here.

The equilibrium points of the online gradient dynamics depends in a non-trivial way on the learning rate, and for large values of η other fixed points appear, making the analysis very convoluted and beyond the scope of this work. A complete study of the role these fixed points play in the training dynamics is left to a future project.

4 Conclusions

By applying the representation of the activation function in terms of Hermite polynomials as introduced in [6], we obtain learning curves for neural networks with ReLU activation at arbitrary learning rate η. To validate our method we

also compare the results with online gradient descent simulations which match the integration of the ODE very well.

For learning rates above a critical value η_c the system no longer converges and we observe an uncontrollable growth of the student vector norms. This behavior is also observed for the system with sigmoidal activation as shown in [4]. However, here we also observe a diverging generalization error since the ReLU is an unbounded function.

We also found a range of values of $\eta \lesssim \eta_c$, where ϵ_g ceases to decrease monotonically with α and presents an interval of α where the performance gets worse with more training data, resembling a double descent behavior [13, 14].

The derivation of I_4 can be performed for other activation functions, including cases of mismatch. However, the ReLU has some simplifying properties such as homogeneity, $g(\lambda x) = \lambda g(x)$ and $\langle H_n, g \rangle = 0$ for odd $n \geq 3$. Without these simplifications, the time to integrate the ODE increases substantially.

References

[1] H. S. Seung, H. Sompolinsky, and N. Tishby. Statistical mechanics of learning from examples. *Phys. Rev. A*, 45:6056–6091, 1992.

[2] T. L. H. Watkin, A. Rau, and M. Biehl. The statistical mechanics of learning a rule. *Rev. Mod. Phys.*, 65:499–556, 1993.

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

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

[5] E. Oostwal, M. Straat, and M. Biehl. Hidden unit specialization in layered neural networks: Relu vs. sigmoidal activation. *Physica A: Stat. Mech. Appl.*, 564:125517, 2021.

[6] O. Citton, F. Richert, and M. Biehl. On-line Learning Dynamics in Layered Neural Networks with Arbitrary Activation Functions. In M. Verleysen, editor, *Proc. European Symposium on Artificial Neural Networks (ESANN)*, pages 437–442, 2024.

[7] M. Straat and M. Biehl. On-line learning dynamics of ReLU neural networks using statistical physics techniques. In M. Verleysen, editor, *Proc. European Symposium on Artificial Neural Networks (ESANN)*, pages 517–522, 2019.

[8] S. Goldt, M. Advani, A. M. Saxe, F. Krzakala, and L. Zdeborová. Dynamics of stochastic gradient descent for two-layer neural networks in the teacher-student setup. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

[9] M. Straat, F. Abadi, Z. Kan, C. Göpfert, B. Hammer, and M. Biehl. Supervised learning in the presence of concept drift: a modeling framework. *Neural Comput. Appl.*, 34:101–118, 2022.

[10] W. F. Kibble. An extension of a theorem of Mehler's on Hermite polynomials. *Mathematical Proceedings of the Cambridge Philosophical Society*, 41(1):12–15, 1945.

[11] D. Slepian. On the Symmetrized Kronecker Power of a Matrix and Extensions of Mehler's Formula for Hermite Polynomials. *SIAM J. Math. Analysis*, 3(4):606–616, 1972.

[12] M. Biehl, P. Riegler, and C. Wöhler. Transient dynamics of on-line learning in two-layered neural networks. *J. Phys. A: Math. Gen.*, 29(16):4769, 1996.

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

[14] T. Viering and M. Loog. The shape of learning curves: A review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(6):7799–7819, 2023.