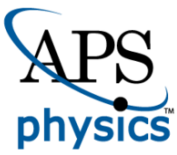

Role of Synaptic Stochasticity in Training Low-Precision Neural Networks

Carlo Baldassi, Federica Gerace, Hilbert J. Kappen, Carlo Lucibello, Luca Saglietti, Enzo Tartaglione, and Riccardo Zecchina

Phys. Rev. Lett. **120**, 268103 — Published 29 June 2018

DOI: 10.1103/PhysRevLett.120.268103

On the role of synaptic stochasticity in training low-precision neural networks

Carlo Baldassi,[1,2,3] Federica Gerace,[2,4] Hilbert J. Kappen,[5] Carlo Lucibello,[2,4] Luca Saglietti,[2,4] Enzo Tartaglione,[2,4] and Riccardo Zecchina[1,2,6]

[1] *Bocconi Institute for Data Science and Analytics, Bocconi University, Milano, Italy*
[2] *Italian Institute for Genomic Medicine, Torino, Italy*
[3] *Istituto Nazionale di Fisica Nucleare, Sezione di Torino, Italy*
[4] *Dept. of Applied Science and Technology, Politecnico di Torino, Torino, Italy*
[5] *Radboud University Nijmegen, Donders Institute for Brain, Cognition and Behaviour 6525 EZ Nijmegen, The Netherlands*
[6] *International Centre for Theoretical Physics, Trieste, Italy*

Stochasticity and limited precision of synaptic weights in neural network models are key aspects of both biological and hardware modeling of learning processes. Here we show that a neural network model with stochastic binary weights naturally gives prominence to exponentially rare dense regions of solutions with a number of desirable properties such as robustness and good generalization performance, while typical solutions are isolated and hard to find. Binary solutions of the standard perceptron problem are obtained from a simple gradient descent procedure on a set of real values parametrizing a probability distribution over the binary synapses. Both analytical and numerical results are presented. An algorithmic extension aimed at training discrete deep neural networks is also investigated.

Learning can be regarded as an optimization process over the connection weights of a neural network. In nature, synaptic weights are known to be plastic, low precision and unreliable, and it is an interesting issue to understand if this stochasticity can help learning or if it is an obstacle. The debate about this issue has a long history and is still unresolved (see [1] and references therein). Here, we provide quantitative evidence that the stochasticity associated with noisy low precision synapses can drive elementary supervised learning processes towards a particular type of solutions which, despite being rare, are robust to noise and generalize well — two crucial features for learning processes.

In recent years, multi-layer (*deep*) neural networks have gained prominence as powerful tools for tackling a large number of cognitive tasks [2]. In a K-class classification task, neural network architectures are typically trained as follows. For any input $x \in \mathcal{X}$ (the input space $\mathcal{X}$ typically being a tensor space) and for a given set of parameters W, called *synaptic weights*, the network defines a probability density function $P(y \,|\, x, W)$ over the K possible outcomes. This is done through composition of affine transformations involving the synaptic weights W, element wise non-linear operators, and finally a softmax operator that turns the outcome of previous operations into a probability density function [3]. The weights W are adjusted, in a supervised learning scenario, using a training set $\mathcal{D}$ of M known input-output associations, $\mathcal{D} = \{(x^\mu, y^\mu)\}_{\mu=1}^{M}$. The learning problem is reframed into the problem of maximizing a log-likelihood $\tilde{\mathcal{L}}\,(W)$ over the synaptic weights W:

$$\max_{W} \quad \tilde{\mathcal{L}}\,(W) \coloneqq \sum_{(x,y)\in\mathcal{D}} \log P\,(y \,|\, x, W) \tag{1}$$

The maximization problem is approximately solved using variants of the Stochastic Gradient Descent (SGD) procedure over the loss function $-\tilde{\mathcal{L}}\,(W)$ [4]. In a Bayesian approach instead one is interested in computing the posterior distribution $P\,(W \,|\, \mathcal{D}) \propto P\,(\mathcal{D} \,|\, W)\, P\,(W)$, where $P\,(W)$ is some prior over the weights W. In deep networks, unfortunately, the exact computation of $P\,(W \,|\, \mathcal{D})$ is typically infeasible and various approximated approaches have been proposed [5–7].

Shallow neural network models, such as the perceptron model for binary classification, are amenable to analytic treatment while exposing a rich phenomenology. They have attracted great attention from the statistical physics community for many decades [8–16]. In the perceptron problem we have binary outputs $y \in \{-1, +1\}$,

while inputs x and weights W are N-components vectors. Under some statistical assumptions on the training set $\mathcal{D}$ and for large N, single variable marginal probabilities $P(W_i \,|\, \mathcal{D})$ can be computed efficiently, using Belief Propagation [17–19]. The learning dynamics has also been analyzed, in particular in the online learning setting [11, 20]. In a slightly different perspective the perceptron problem is often framed as the task of minimizing the error-counting Hamiltonian

$$\min_{W} \quad \mathcal{H}(W) \coloneqq \sum_{(x,y)\in\mathcal{D}} \Theta\left(-y \sum_{i=1}^{N} W_i\, x_i\right), \qquad (2)$$

where $\Theta(x)$ is the Heaviside step function, $\Theta(x) = 1$ if $x > 0$ and 0 otherwise. As a constraint satisfaction problem, it is said to be satisfiable (SAT) if zero energy (i.e. $\mathcal{H}(W) = 0$) configurations exists, unsatisfiable (UNSAT) otherwise. We call *solutions* such configurations. Statistical physics analysis, assuming random and uncorrelated $\mathcal{D}$, shows a sharp threshold at a certain $\alpha_c = M/N$, when N grows large, separating a SAT phase from an UNSAT one. Moreover, restricting the synaptic space to binary values,$W_i = \pm 1$, leads to a more complex scenario: most solutions are essentially isolated and computationally hard to find [13, 21]. Some efficient algorithms do exist though [12, 22] and generally land in a region dense of solutions. This apparent inconsistency has been solved through a large deviation analysis which revealed the existence of sub-dominant and dense regions of solutions [14, 23]. This analysis introduced the concept of Local Entropy [14] which subsequently led to other algorithmic developments [24–26] (see also [27] for related analysis).

In the generalization perspective, solutions within a dense region may be loosely considered as representative of the entire region itself, and therefore act as better pointwise predictors than isolated solutions, since the optimal Bayesian predictor is obtained averaging all solutions [14].

Here, we propose a method to solve the binary perceptron problem (2) through a relaxation to a distributional space. We introduce a perceptron problem with stochastic discrete weights, and show how the learning process is naturally driven towards dense regions of solutions, even in the regime in which they are exponentially rare compared to the isolated ones. In perspective, the same approach can be extended to the general learning problem (1), as we will show.

Denote with $Q_\theta(W)$ a family of probability distributions over W parametrized by a set of variables θ. Consider the following problem:

$$\max_{\theta} \quad \mathcal{L}(\theta) \coloneqq \sum_{(x,y)\in\mathcal{D}} \log \mathbb{E}_{W\sim Q_\theta} P(y \,|\, x, W) \qquad (3)$$

Here $\mathcal{L}(\theta)$ is the log-likelihood of a model where for each training example $(x, y) \in \mathcal{D}$ the synaptic weights are independently sampled according to $Q_\theta(W)$. Within this scheme two class predictors can be devised for any input x: $\hat{y}_1(x) = \operatorname{argmax}_y P(y \,|\, x, \hat{W})$, where $\hat{W} = \operatorname{argmax}_W Q_\theta(W)$, and $\hat{y}_2(x) = \operatorname{argmax}_y \int \mathrm{d}W\, P(y \,|\, x, W)\, Q_\theta(W)$. In this paper we will analyze the quality of the training error given by the first predictor. Generally, dealing with Problem (3) is more difficult than dealing with Problem (1), since it retains some of the difficulties of the computation of $P(W \,|\, \mathcal{D})$. Also notice that for any maximizer $W^\star$ of Problem (1) we have that $\delta(W - W^\star)$ is a maximizer of Problem (3) provided that it belongs to the parametric family, as can be shown using Jensen's inequality. Problem (3) is a "distributional" relaxation of Problem (1).

Optimizing $\mathcal{L}(\theta)$ instead of $\tilde{\mathcal{L}}(W)$ may seem an unnecessary complication. In this paper we argue that there are two reasons for dealing with this kind of task. First, when the configuration space of each synapse is restricted to discrete values, the network cannot be trained with SGD procedures. The problem, while being very important for computational efficiency and memory gains, has been tackled only very recently [5, 28]. Since variables θ typically lie in a continuous manifold instead, standard continuous optimization tools can be applied to $\mathcal{L}(\theta)$. Also, the learning dynamics on $\mathcal{L}(\theta)$ enjoys some additional properties when compared to the dynamics on $\tilde{\mathcal{L}}(W)$. In the latter case additional regularizers, such as dropout and L_2 norm, are commonly used to improve generalization properties [4]. The SGD in the θ-space instead already incorporates the kind of natural regularization intrinsic in the Bayesian approach and the robustness associated to high local entropy [14]. Here we make a case for these arguments by a numerical

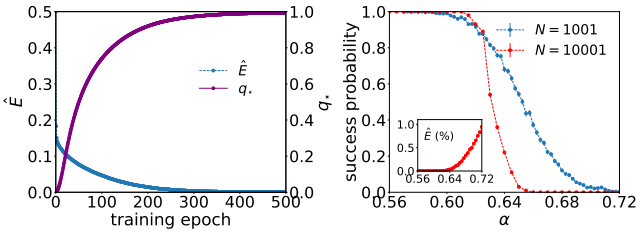

Figure 1. (*Left*) The training error and the squared norm against the number of training epochs, for $\alpha = 0.55$ and $N = 10001$, averaged over 100 samples. (*Right*) Success probability in the classification task as a function of the load α for networks of size $N = 1001, 10001$ averaging 1000 and 100 samples respectively. In the inset we show the average training error at the end of GD as a function of α.

and analytical study of the proposed approach for the binary perceptron. We also present promising preliminary numerical results on deeper networks.

Learning for the Stochastic Perceptron. Following the above discussion, we now introduce our binary stochastic perceptron model. For each input x presented, N synaptic weights $W = (W_1, \dots, W_N)$, $W_i \in \{-1, +1\}$, are randomly extracted according to the distribution

$$Q_m(W) = \prod_{i=1}^{N} \left[\frac{1+m_i}{2} \delta_{W_i,+1} + \frac{1-m_i}{2} \delta_{W_i,-1} \right] \quad (4)$$

where $\delta_{a,b}$ is the Kronecker delta symbol. We will refer to the set $m = (m_i)_i$, where $m_i \in [-1, 1]$ $\forall i$, as the magnetizations or the control parameters. We choose the probability $P(y \,|\, x, W)$ on the class $y \in \{-1, +1\}$ for a given input x as follows:

$$P(y \,|\, x, W) = \Theta\left(y \sum_{i=1}^{N} W_i \, x_i \right). \quad (5)$$

While other possibilities for $P(y \,|\, x, W)$ could be considered, this particular choice is directly related to the form of the Hamiltonian in Problem (2), which we ultimately aim to solve. Given a training set $\mathcal{D} = \{(x^\mu, y^\mu)\}_{\mu=1}^{M}$, we can then compute the log-likelihood function of Eq. (3), with the additional assumption that N is large and the central limit theorem applies. It reads

$$\mathcal{L}(m) = \sum_{(x,y)\in\mathcal{D}} \log H\left(-\frac{y \sum_i m_i \, x_i}{\sqrt{\sum_i (1 - m_i^2) \, x_i^2}} \right), \quad (6)$$

where $H(x) := \int_x^\infty dz \, e^{-z^2/2}/\sqrt{2\pi}$. Minimizing $-\mathcal{L}(m)$ instead of finding the solutions of Problem (2) allows us to use the simplest method for approximately solving continuous optimization problems, the Gradient Descent (GD) algorithm:

$$m_i^{t+1} \leftarrow \mathrm{clip}\left(m_i^t + \eta \, \partial_{m_i} \mathcal{L}(m^t) \right). \quad (7)$$

We could have adopted the more efficient SGD approach, however in our case simple GD is already effective. In the last equation η is a suitable learning rate and $\mathrm{clip}(x) := \max(-1, \min(1, x))$, applied element-wise. Parameters are randomly initialized to small values, $m_i^0 \sim \mathcal{N}(0, N^{-1})$. At any epoch t in the GD dynamics a binarized configuration $\hat{W}_i^t = \mathrm{sign}(m_i^t)$ can be used to compute the training error $\hat{E}^t = \frac{1}{M} \mathcal{H}\left(\hat{W}^t\right)$. We consider a training set $\mathcal{D}$ where each input component x_i^μ is sampled uniformly and independently in $\{-1, 1\}$ (with this choice we can set $y^\mu = 1 \, \forall \mu$ without loss of generality). The evolution of the network during GD is shown in Fig. 1. The training error goes progressively to zero while the mean squared norm of the control variables $q_\star^t = \frac{1}{N} \sum_i (m_i^t)^2$ approaches one. Therefore the distribution Q_m concentrates around a single configuration as the training is progressing. This natural flow is similar to the annealing of the coupling parameter manually performed in local entropy inspired algorithms [25, 26]. We also show in Fig. 1 the probability over the realizations of $\mathcal{D}$ of finding a solution of the binary problem as function of the load $\alpha = M/N$. The algorithmic capacity of GD is approximately $\alpha_{GD} \approx 0.63$. This value has to be compared to the theoretical capacity $\alpha_c \approx 0.83$, above which there are almost surely no solutions [9], and state-of-the-art algorithms based on message passing heuristics for which we have a range of capacities $\alpha_{MP} \in [0.6, 0.74]$ [12, 22, 29]. Therefore GD reaches loads only slightly worse than those reached by much more fine tuned algorithms, a surprising results for such a simple procedure. Also, for α slightly above α_{GD}

the training error remains comparably low, as shown in Fig. 1. In our experiments most variants of the GD procedure of Eq. (7) performed just as well: e.g. SGD ors GD computed on the fields $h_i^t = \tanh^{-1}(m_i^t)$ rather than the magnetizations[30]. Other updates rules for the control parameters can be derived as multiple pass of on-line Bayesian learning [31, 32]. Variations of rule (7) towards biological plausibility are discussed in the SM [33].

Deep Networks. We applied our framework to deep neural networks with binary stochastic weights and sign activation functions. Using an uncorrelated neuron approximation, as in Ref. [6], we trained the network using the standard SGD algorithm with backpropagation. We give the details in the SM. On the MNIST benchmark problem [34], using a network with three hidden layers we achieved $\sim 1.7\%$ test error, a very good result for a network with binary weights and activations and with no convolutional layers [35]. No other existing approach to the binary perceptron problem has been extended yet to deeper settings.

Statistical mechanics Analysis. We now proceed with the analytical investigation of the equilibrium properties of the stochastic perceptron, which partly motivates the good performance of the GD dynamics. The starting point of the analysis is the partition function

$$Z = \int_\Omega \prod_i \mathrm{d}m_i \; \delta\left(\sum_i m_i^2 - q_\star N\right) e^{\beta \mathcal{L}(m)} \qquad (8)$$

where $\Omega = [-1,1]^N$, β is an inverse temperature, and we constrained the squared norm to $q_\star N$ in order to mimic the natural flow of $q_\star^t$ in the training process. The dependence on the training set $\mathcal{D}$ is implicit in last equation. We shall denote with $\mathbb{E}_\mathcal{D}$ the average over the training sets with i.i.d. input and output components uniform in $\{-1,1\}$. We investigate the average properties of the system for large N and fixed load $\alpha = M/N$ using the replica method in the Replica Symmetric (RS) ansatz [36]. Unfortunately the RS solution becomes locally unstable for very large β. Therefore, instead of taking the infinite β limit to maximize the likelihood we will present the results obtained for β large but still in the RS region. The details of the free energy calculation and of the stability check can be found in the SM.

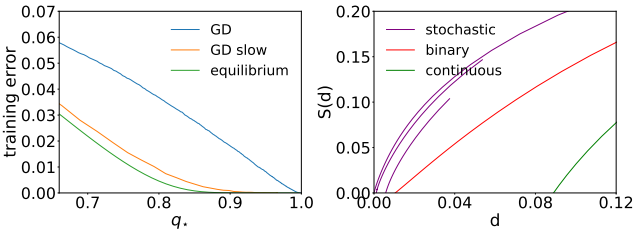

Figure 2. (*Left*) Energy of the Binarized Configuration versus the control variable $q_\star$. We show the equilibrium prediction of Eq. (9), and numerical results from the GD algorithm and a GD algorithm variant where after each update we rescale the norm of m to $q_\star$ until convergence before moving to the next value of $q_\star$ according to a certain schedule. The results are averaged over 20 random realizations of the training set with $N = 10001$. (*Right*) Entropy of binary solutions at fixed distance d from BCs of the spherical, binary and stochastic perceptron ($q_\star = 0.7$, 0.8 and 0.9 from bottom to top) at thermodynamic equilibrium. In both figures $\alpha = 0.55$, also $\beta = 20$ for the stochastic perceptron and $\beta = \infty$ for the spherical and binary ones.

Energy of the Binarized Configuration. We now analyze some properties of the mode of the distribution $Q_m(W)$, namely $\hat{W}_i = \mathrm{sign}(m_i)$, that we call Binarized Configuration (BC). The average training error per pattern is:

$$E = \lim_{N\to\infty} \frac{1}{\alpha N} \mathbb{E}_\mathcal{D}\left[\sum_{(x,y)\in\mathcal{D}} \left\langle \Theta\left(-y \sum_i \mathrm{sign}(m_i)\, x_i\right)\right\rangle\right] \qquad (9)$$

where $\langle \bullet \rangle$ is the thermal average over m according to the partition function (8), which implicitly depends on $\mathcal{D}$, $q_\star$ and β. The last equation can be computed analytically within the replica framework (see SM). In Fig. 2 (*Left*) we show that for large β the BC becomes a solution of the problem when $q_\star$ approaches one. This is compared to the values of the training error obtained from GD dynamics at corresponding values of $q_\star$, and a modified GD dynamics where we let the system equilibrate at fixed $q_\star$. The latter case, although we are at finite N and we are considering a dynamical process that could suffer the presence of local minima, is in rea-

sonable agreement with the equilibrium result of Eq. (9).

Geometrical structure of the solution space. Most solutions of the binary perceptron problem are isolated [13], while a subdominant but still exponentially large number belongs to a dense connected region [14]. Solutions in the dense region are the only ones that are algorithmically accessible. Here we show that BCs of the stochastic binary perceptron typically belong to the dense region, provided $q_\star$ is high enough. To prove this we count the number of solutions at a fixed Hamming distance d from typical BC (this corresponds to fixing an overlap $p = 1-2d$). Following the approach of Franz and Parisi [37] we introduce the constrained partition function

$$\mathcal{Z}(d,m) = \sum_{W} \prod_{(x,y)\in\mathcal{D}} \Theta\left(y \sum_i W_i x_i\right) \times \delta\left(N\left(1-2d\right) - \sum_i \mathrm{sign}\left(m_i\right) W_i\right), \quad (10)$$

where the sum is over the $\{-1,+1\}^N$ binary configurations. The Franz-Parisi entropy $\mathcal{S}\left(d\right)$ is then given by

$$\mathcal{S}(d) = \lim_{N\to\infty} \frac{1}{N} \mathbb{E}_{\mathcal{D}} \left\langle \log\, \mathcal{Z}\left(d,m\right)\right\rangle . \quad (11)$$

We show how to compute $\mathcal{S}\left(d\right)$ in the SM. In Fig. 2 (*Right*) we compare $\mathcal{S}\left(d\right)$ for the stochastic perceptron with the analogous entropies obtained substituting the expectation $\langle\bullet\rangle$ over m in Eq. (11) with a uniform sampling from the solution space of the spherical (the model of Ref. [8]) and the binary (as in Ref. [13]) perceptron. The distance gap between the BC and the nearest binary solutions (i.e., the value of the distance after which $\mathcal{S}(d)$ becomes positive) vanishes as $q_\star$ is increased: in this regime the BC belongs to the dense cluster and we have an exponential number of solutions at any distance $d > 0$. Typical binary solutions and binarized solutions of the continuous perceptron are isolated instead (finite gap, corresponding to $S(d) = 0$ at small distances). In the SM we provide additional numerical results on the properties of the energetic landscape in the neighborhood of different types of solutions, showing that solutions in flatter basins achieve better generalization than those in sharp ones.

Conclusions. Our analysis shows that stochasticity in the synaptic connections may play a fundamental role in learning processes, by effectively reweighting the error loss function, enhancing dense, robust regions, suppressing narrow local minima and improving generalization.

The simple perceptron model allowed us to derive analytical results as well as to perform numerical tests. Moreover, as we show in the SM, when considering discretized priors, there exist a connection with the dropout procedure which is popular in modern deep learning practice. However, the most promising immediate application is in the deep learning scenario, where this framework can be extended adapting the tools developed in Refs. [6, 7], and where we already achieved state-of-the-art results in our preliminary investigations.

Hopefully, the general mechanism shown here can also help to shed some light on biological learning processes, where the role of low precision and stochasticity is still an open question. Finally, we note that this procedure is not limited to neural network models; for instance, application to constraint satisfaction problems is straightforward.

CB, HJB and RZ acknowledge ONR Grant N00014-17-1-2569.

[1] H Sebastian Seung. Learning in spiking neural networks by reinforcement of stochastic synaptic transmission. *Neuron*, 40(6):1063–1073, 2003.

[2] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

[3] David JC MacKay. *Information theory, inference and learning algorithms*. Cambridge university press, 2003.

[4] Léon Bottou, Frank E Curtis, and Jorge Nocedal. Optimization methods for large-scale machine learning. *arXiv preprint arXiv:1606.04838*, 2016.

[5] Daniel Soudry, Itay Hubara, and R Meir. Expectation Backpropagation: parameter-free training of multilayer neural networks with real and discrete weights. *Neural Information Processing Systems 2014*, 2(1):1–9, 2014.

[6] José Miguel Hernández-Lobato and Ryan P Adams. Probabilistic Backpropagation for Scalable Learning of

Bayesian Neural Networks. *Journal of Machine Learning Research*, 37:1–6, feb 2015.

[7] Oran Shayar, Dan Levi, and Ethan Fetaya. Learning discrete weights using the local reparameterization trick. *ArXiv e-print*, 2017.

[8] Elizabeth Gardner. The space of interactions in neural network models, 1988.

[9] Werner Krauth and Marc Mézard. Storage capacity of memory networks with binary couplings, 1989.

[10] Manfred Opper and Ole Winther. A Mean Field Approach to Bayes Learning in Feed-Forward Neural Networks. *Physical Review Letters*, 76(11):1964–1967, mar 1996.

[11] Sara Solla and Ole Winther. Optimal perceptron learning: an online Bayesian approach. In David Saad, editor, *On-Line Learning in Neural Networks*, pages 1–20. Cambridge University Press, 1998.

[12] Alfredo Braunstein and Riccardo Zecchina. Learning by Message Passing in Networks of Discrete Synapses. *Physical Review Letters*, 96(3):030201, jan 2006.

[13] Haiping Huang and Yoshiyuki Kabashima. Origin of the computational hardness for learning with binary synapses. *Physical Review E*, 90(5):052813, nov 2014.

[14] Carlo Baldassi, Alessandro Ingrosso, Carlo Lucibello, Luca Saglietti, and Riccardo Zecchina. Subdominant Dense Clusters Allow for Simple Learning and High Computational Performance in Neural Networks with Discrete Synapses. *Physical Review Letters*, 115(12):128101, 2015.

[15] Silvio Franz and Giorgio Parisi. The simplest model of jamming. *Journal of Physics A: Mathematical and Theoretical*, 49(14):145001, apr 2016.

[16] Simona Cocco, Rémi Monasson, and Riccardo Zecchina. Analytical and numerical study of internal representations in multilayer neural networks with binary weights. *Physical Review E*, 54(1):717–736, jul 1996.

[17] Marc Mézard. The space of interactions in neural networks: Gardner's computation with the cavity method. *Journal of Physics A: Mathematical and General*, 22(12):2181–2190, jun 1989.

[18] Giorgio Parisi, Marc Mézard, and Miguel Angel Virasoro. *Spin glass theory and beyond*. World Scientific Singapore, 1987.

[19] Andrea Montanari and Marc Mézard. *Information, Physics and Computation*. Oxford Univ. Press, 2009.

[20] David Saad. *On-line learning in neural networks*. Cambridge University Press, 1998.

[21] Lenka Zdeborová and Marc Mézard. Constraint satisfaction problems with isolated solutions are hard. *Journal of Statistical Mechanics: Theory and Experiment*, 2008(12):P12004, oct 2008.

[22] Carlo Baldassi, Alfredo Braunstein, Nicolas Brunel, and Riccardo Zecchina. Efficient supervised learning in networks with binary synapses. *Proceedings of the National Academy of Sciences*, 104(26):11079–11084, jun 2007.

[23] Carlo Baldassi, Federica Gerace, Carlo Lucibello, Luca Saglietti, and Riccardo Zecchina. Learning may need only a few bits of synaptic precision. *Physical Review E*, 93(5):052313, 2016.

[24] Carlo Baldassi, Alessandro Ingrosso, Carlo Lucibello, Luca Saglietti, and Riccardo Zecchina. Local entropy as a measure for sampling solutions in constraint satisfaction problems. *Journal of Statistical Mechanics: Theory and Experiment*, 2016(2):023301, 2016.

[25] Carlo Baldassi, Christian Borgs, Jennifer T. Chayes, Alessandro Ingrosso, Carlo Lucibello, Luca Saglietti, and Riccardo Zecchina. Unreasonable effectiveness of learning neural networks: From accessible states and robust ensembles to basic algorithmic schemes. *Proceedings of the National Academy of Sciences*, 113(48):E7655–E7662, nov 2016.

[26] Pratik Chaudhari, Anna Choromanska, Stefano Soatto, Yann LeCun, Carlo Baldassi, Christian Borgs, Jennifer T. Chayes, Levent Sagun, and Riccardo Zecchina. Entropy-SGD: Biasing Gradient Descent Into Wide Valleys. *ArXiv e-prints*, nov 2016.

[27] Alfredo Braunstein, Luca Dall'Asta, Guilhem Semerjian, and Lenka Zdeborová. The large deviations of the whitening process in random constraint satisfaction problems. *Journal of Statistical Mechanics: Theory and Experiment*, 2016(5):053401, may 2016.

[28] Matthieu Courbariaux, Itay Hubara, Daniel Soudry, Ran El-Yaniv, and Yoshua Bengio. Binarized Neural Networks: Training Deep Neural Networks with Weights and Activations Constrained to +1 or -1. *ArXiv e-prints*, page 9, feb 2016.

[29] Carlo Baldassi and Alfredo Braunstein. A max-sum algorithm for training discrete neural networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2015(8):P08008, 2015.

[30] This has the advantage that it doesn't require clipping.

[31] Manfred Opper and Ole Winther. A Bayesian approach to on-line learning. In David Saad, editor, *On-line learning in neural networks*, pages 363–378. 1998.

[32] Thomas P Minka. Expectation Propagation for Approximate Bayesian Inference F d. *Uncertainty in Artificial Intelligence (UAI)*, 17(2):362–369, 2001.

[33] Supplemental Material contains further discussions on biologically plausible algorithms, details of the replica analysis and algorithmic extensions to deep networks. It also provides the additional Refs. [38–48].

[34] Yann LeCun, Corinna Cortes, and Christopher JC Burges. Mnist handwritten digit database. *AT&T Labs [Online]. Available: http://yann. lecun. com/exdb/mnist*, 2, 2010.

[35] Carlo Baldassi, Christian Borgs, Jennifer T. Chayes,

Carlo Lucibello, Luca Saglietti, Enzo Tartaglione, and Riccardo Zecchina. *In preparation.*

[36] Marc Mézard and Andrea Montanari. *Information, Physics, and Computation.* Oxford University Press, January 2009.

[37] Silvio Franz and Giorgio Parisi. Recipes for metastable states in spin glasses. *Journal de Physique I*, 5(11):1401–1415, 1995.

[38] Carlo Baldassi. Generalization Learning in a Perceptron with Binary Synapses. *Journal of Statistical Physics*, 136(5):902–916, sep 2009.

[39] Carlo Baldassi and Riccardo Zecchina. Efficiency of quantum vs. classical annealing in nonconvex learning problems. *Proceedings of the National Academy of Sciences*, 115(7):1457–1462, feb 2018.

[40] Thomas M Bartol, Cailey Bromer, Justin P Kinney, Michael A Chirillo, Jennifer N Bourne, Kristen M Harris, and Terrence J Sejnowski. Hippocampal spine head sizes are highly precise. *bioRxiv*, page 016329, 2015.

[41] Andreas Engel. *Statistical mechanics of learning.* Cambridge University Press, 2001.

[42] Elizabeth Gardner and Bernard Derrida. Optimal storage properties of neural network models, 1988.

[43] Heinz Horner. Dynamics of learning for the binary perceptron problem. *Zeitschrift für Physik B Condensed Matter*, 86(2):291–308, 1992.

[44] Yonatan Loewenstein and H. Sebastian Seung. Operant matching is a generic outcome of synaptic plasticity based on the covariance between reward and neural activity. *Proceedings of the National Academy of Sciences*, 103(41):15224–15229, 2006.

[45] Daniel H. O'Connor, Gayle M. Wittenberg, and Samuel S.-H. Wang. Graded bidirectional synaptic plasticity is composed of switch-like unitary events. *Proceedings of the National Academy of Sciences*, 102(27):9679–9684, 2005.

[46] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 2014.

[47] Li Wan, Matthew Zeiler, Sixin Zhang, Yann Lecun, and Rob Fergus. Regularization of neural networks using dropconnect. In *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*, 2013.

[48] Deniz Yuret. Knet: beginning deep learning with 100 lines of julia. In *Machine Learning Systems Workshop at NIPS 2016*, 2016.