proceedings

MDPI

Proceedings

Entropic Dynamics for Learning in Neural Networks and the Renormalization Group †

Nestor Caticha

Instituto de Fisica, Universidade de Sao Paulo, 05508-090 Sao Paulo, SP, Brazil; ncaticha@usp.br

† Presented at the 39th International Workshop on Bayesian Inference and Maximum Entropy Methods in Science and Engineering, Garching, Germany, 30 June–5 July 2019.

Published: 25 November 2019

Abstract: We study the dynamics of information processing in the continuous depth limit of deep feed-forward Neural Networks (NN) and find that it can be described in language similar to the Renormalization Group (RG). The association of concepts to patterns by NN is analogous to the identification of the few variables that characterize the thermodynamic state obtained by the RG from microstates. We encode the information about the weights of a NN in a Maxent family of distributions. The location hyper-parameters represent the weights estimates. Bayesian learning of new examples determine new constraints on the generators of the family, yielding a new pdf and in the ensuing entropic dynamics of learning, hyper-parameters change along the gradient of the evidence. For a feed-forward architecture the evidence can be written recursively from the evidence up to the previous layer convoluted with an aggregation kernel. The continuum limit leads to a diffusion-like PDE analogous to Wilson's RG but with an aggregation kernel that depends on the the weights of the NN, different from those that integrate out ultraviolet degrees of freedom. Approximations to the evidence can be obtained from solutions of the RG equation. Its derivatives with respect to the hyper-parameters, generate examples of Entropic Dynamics in Neural Networks Architectures (EDNNA) learning algorithms. For simple architectures, these algorithms can be shown to yield optimal generalization in student- teacher scenarios.

Keywords: Neural Networks; Renormalization Group; Entropic Dynamics; learning algorithms

1. Introduction

Neural networks are information processing systems that learn from examples. Loosely inspired in biological neural systems, they have been used for several types of problems such as classification, regression, dimensional reduction and clustering [1]. Biological systems selection is based on a measure of performance that combines not only accuracy but also ease of computation and implementation. Predictions based on expectations over posterior Bayesian distributions may lead to saturating bounds for optimal accuracy learning but will typically lack in ease of computation and speed in reaching a result. Neural networks are parametric models and if we don't address the determination of the architecture, which we don't in this paper, the problem of learning from examples is reduced to obtaining fast estimates of the weights or parameters, avoiding the integration over large dimensional spaces. The spectacular explosion of applications in several areas is witness to the fact that several training methods and large data sets are available. Despite these victories, the mechanisms of information dynamics processing remain obscure and despite several decades of theoretical analysis using methods of Statistical Mechanics, much remains to be understood. Here we study on-line learning in feed-forward architectures, where (input,output) examples are presented one at a time. Theoretical analysis is easier than for batch or off-line learning where the cost function depends on a large number of example pairs, however on-line accuracy performance remains high. This is in

part due to the fact that since the cost function changes from example to example, the local minima of the cost function that plague off-line learning are not so important. Local stationary points of the learning dynamics are still a problem, but good performances are possible. An important problem to be addressed is what cost function is the most appropriate. If an algorithm is going to be successful it has to approach Bayesian estimates for the available information. But any Bayes algorithm leads to high, even in the millions, dimensional integrals. Monte Carlo strategies cannot be used if simplicity is a requirement. The strategy to determine optimized algorithms for on-line learning has been studied in the past for restricted scenarios and architectures. We present a more general approach, with the following strategy. We are in a situation of incomplete information, thus a probability distribution represents, at a given point in the dynamics, what is known about the parameters. We have to commit to a family of distributions and we choose a Maxent family. Location hyperparameters give the current estimate of the weights. A new (input,output) example pair arrives and Bayes rule permits an update. The choice of the likelihood is a reflection of what we know about the architecture of the NN. In general it is not conjugated to the chosen family.

Still, the Bayes posterior, while not in the family, points to a unique member of the family, since it imposes new constraints on the expected values of the generators.

The resulting learning algorithm is the entropic dynamics imposed by the arrival of information in the examples that induces a change of the hyperparameters of the family. It turns out that changes in the weights are in the direction of decreasing the model Bayesian evidence and it is a stochastic gradient descent algorithm, where the cost function is the log evidence of the model.

The denominator of the Bayes update can be interpreted either as the evidence of the model or alternatively as the predictive probability distribution of the output conditioned on the input and the weights. Once it is written as the marginalization over the internal representation, i.e. the activation values of the internal units, of the joint distribution of activities of the whole network, and under the supposition that the information flows only from one layer to the next, a Markov chain structure follows. Recursion relations of the partial evidence up to a given internal layer are obtained and in the continuous depth limit (CDL) a Fokker-Planck parabolic partial differential equation is obtained. It generalizes Wilson's Renormalization Group [2] diffusion equation for general kernels. The usual, e.g., majority rule that eliminates high frequency degrees of freedom are replaced by the weights of the NN. The RG dynamics can be seen as a classifier of Statistical Mechanics microstates into thermodynamics states. A NN extracts the relevant degrees of freedom that describe the macroscopic concept onto which an input pattern is to be assigned. The first authors to relate the RG and NN were [3] and [4] generating a large flow of ideas into the possible connections between these two areas [5–7].

2. Maxent Distributions and Bayesian Learning

Let $f_a(\boldsymbol{w})$, for $a = 1, ...K$, $\boldsymbol{w} \in \mathbb{R}^N$, be the generators of a family $\mathcal{Q}$ of distributions $Q(\boldsymbol{w}|\boldsymbol{\lambda})$. If information about $\boldsymbol{w}$ is given in the form of constraints $\mathbb{E}_Q(f_a) = F_a$, for the set of numbers $\{F_a\}_{a=1,K}$, the Maxent distribution is

$$Q(\boldsymbol{w}|\boldsymbol{\lambda}) = \frac{1}{z} \exp\left(-\sum_{i=1}^{K} \lambda_a f_a(\boldsymbol{w})\right), \tag{1}$$

where z ensures normalization. Then

$$\frac{\partial \ln z}{\partial \lambda_a} = -F_a \text{ and } \frac{\partial Q(\boldsymbol{w}|\boldsymbol{\lambda})}{\partial \lambda_a} = (-f_a + F_a) Q(\boldsymbol{w}|\boldsymbol{\lambda}). \tag{2}$$

Now consider a system learning a map from inputs x to outputs y, and the model is a known function which depends on a parameter array $\boldsymbol{w}$: $y = T(x; \boldsymbol{w})$. The aim of learning is to obtain the parameters from the information in the learning set $\mathcal{D}_n = \{(x_i, y_i)\}_{i=1,n}$. We want to obtain a distribution for the parameters and consider that up to $n-1$ examples the information is coded in a member of the Q

family: $Q(\boldsymbol{w}|\boldsymbol{\lambda}_{n-1}) = Q_{n-1}$. Calling the likelihood of the problem $L_n = P(y_n|x_n, \boldsymbol{w})$, the product rule permits the Bayesian updating

$$P_n = P(\boldsymbol{w}|\mathcal{D}_n) \quad = \quad \frac{Q_{n-1}L_n}{Z_n}, \tag{3}$$

where the partition function or the evidence is $Z(y_n|x_n, \boldsymbol{\lambda}_{n-1}) = \int Q_{n-1}L_n d\boldsymbol{w} = P(y_n|x_n\boldsymbol{\lambda}_{n-1})$. The Bayes posterior given by eq. 3 in general doesn't belong to the $\mathcal{Q}$ family. We have to choose the member of the family that is closest to the Bayes posterior. This is the Maxent posterior. The way to proceed is based on the fact that a member of the $\mathcal{Q}$ family is determined solely by the values of the constraints $\{F_a\}$. The Bayes posterior defines a set of values for the constraints $\{\langle f_a \rangle\}$. It points in a unique way to the Maxent posterior Q_n within the family $\mathcal{Q}$, obtained at the extreme of

$$S[Q_n||Q_{n-1}] = -\int Q_n \log \frac{Q_n}{Q_{n-1}} d\boldsymbol{w} - \Delta\lambda_a \left(\mathbb{E}_n(f_a) - \langle f_a \rangle\right), \tag{4}$$

subject to the only possible constraints on its expected values $\mathbb{E}_n(f_a)$ which are taken to be the Bayes posterior expected values $\langle f_a \rangle$. Then for every generator

$$\mathbb{E}_{Q_n}(f_a) \quad = \quad \int \frac{Q_{n-1}L_n}{Z_n} f_a(\boldsymbol{w}) d\boldsymbol{w} = \mathbb{E}_{P_n}(f_a) = F_a^n. \tag{5}$$

Subtract from both sides F_a^{n-1}, and use equation 2, then

$$F_a^n - F_a^{n-1} \quad = \quad -\frac{\partial \ln Z}{\partial \lambda_a^{n-1}} \tag{6}$$

since the likelihood is independent of the Lagrange multiplier. This learning dynamics is deduced from entropy maximization and thus will be called Entropic dynamics. Learning occurs along the gradient of the log evidence. It will turn out that the sign is such that typically the evidence for the new model is higher than before learning. These equations hold for any family, but it is interesting to consider the case that will be most likely to be useful in practice, where the family is determined by the functions $f_0 = 1$, $f_i = w_i$ and $f_{ij} = w_i w_j$, for $i, j = 1, N$. The constraints after n examples are the normalization, $\mathbb{E}(w_i) = \hat{w}_{ni}$ and $\mathbb{E}(w_i w_j) = (\boldsymbol{C}_n)_{ij} + \hat{w}_{ni}\hat{w}_{nj}$. The result is the gaussian family $Q \propto \exp(-\lambda_0 - \sum_i \lambda_i w_i - \sum_{ij} \lambda_{ij} w_i w_j)$. The entropic dynamics update equations, driven by the arrival of the n^{th} example are

$$\hat{\boldsymbol{w}}_n \quad = \quad \hat{\boldsymbol{w}}_{n-1} + \boldsymbol{C}_{n-1}.\nabla_{\hat{w}_{n-1}} \log Z_n, \tag{7}$$

$$\boldsymbol{C}_n \quad = \quad \boldsymbol{C}_{n-1} + \boldsymbol{C}_{n-1}.\nabla^2_{\hat{w}_{n-1}} \log Z_n.\boldsymbol{C}_{n-1}. \tag{8}$$

For a layered network, these are the equations associated to the update of the weights afferent to a particular unit in layer d from unit i in layer $d-1$ and of the component of the covariance matrix describing the correlation between weights coming from units i and j. The update equations, induced by a maximum entropy approximation to Bayesian learning is the learning algorithm of the neural network which implements the map $y = T(x; \hat{\boldsymbol{w}})$.

An approximation to this scheme was found for simple networks with no hidden units using a variational procedure ([8]) and applied to several architectures [9–13]. Then Opper [14] showed the Bayesian connection, explored elsewhere [15]. Recently it has been applied to societies of interacting neural networks [16–19]. While [12] attacked the neural network with a hidden layer, the challenge remains to study networks with deep architectures.

3. Deep Multilayer Perceptron

In this section we show that the evidence for a multilayer feedforward neural network can be written recursively as a map. Actually we will get two maps that are essentially the same. This type of map is typical of Renormalization Group transformations and in a continuous limit representation of the neural network as a field theory, we will show that the map leads to a partial differential equation analogous to Wilson's diffusion-like RG equation.

We fix our attention at the n^{th} example, and hence don't write temporal (lower) indices anymore. A layer (upper) index now appears and x^d is the internal representation at the the unit layer d. Layers start with $d = 0$ and the depth of the network is D. Layer d weights are collectively denoted $\boldsymbol{w}^d$ and individually w_{ij}^d is the weight connecting unit i at layer $d-1$ to unit j at layer d. The data pair used for the learning step are X_0 and y. The distributions of the representation at the input is $\delta(x^0 - X^0)$ and at the output $\delta(x^D - y)$. The partition function $Z(y_n|x_n, \boldsymbol{\lambda}_{n-1})$ in Equation (3) is $Z(X^D|x^0, \boldsymbol{\lambda}) = \int Q(\boldsymbol{w}|\boldsymbol{\lambda}) L d\boldsymbol{w}$, where $Q(\boldsymbol{w}|\boldsymbol{\lambda})$ is the prior joint distribution of the weights over all the layers. We will eventually take this to be a product over layers, $Q(\boldsymbol{w}|\boldsymbol{\lambda}) = \prod_{d=1}^{D-1} Q(\boldsymbol{w}^d|\boldsymbol{\lambda}_d)$. which will permit a simpler analytical treatment, but it is not a necessity at this moment. To obtain the likelihood we marginalize the joint distribution of the internal representations $P(x^D, x^{D-1}....x^1|x^0, \boldsymbol{w}^1, ...\boldsymbol{w}^D)$ over all internal representations at the hidden units doing the same trick that leads to the Chapman-Kolmogorov equation

$$L = P(x^D|x^0 = X^0, \boldsymbol{w}^1, ...\boldsymbol{w}^D) = \int P(x^D, x^{D-1}, ...x^1|x^0 = X^0, \boldsymbol{w}^1, ...\boldsymbol{w}^D) \prod_{d=1}^{D-1} dx^d. \tag{9}$$

The evidence can be written as

$$Z_D(x^D|X^0, \boldsymbol{\lambda}) = \int Q^T(x^D, x^{D-1}....x^1|x^0 = X^0, \boldsymbol{\lambda}) \prod_{d=1}^{D-1} dx^d. \tag{10}$$

where

$$Q^T(x^D, x^{D-1}....x^1|x^0 = X^0, \boldsymbol{\lambda}) = \int P(x^D, x^{D-1}....x^1|x^0 = X^0, \boldsymbol{w}^1, ...\boldsymbol{w}^D) \times \prod_{d=1}^{D-1} Q(\boldsymbol{w}^d|\boldsymbol{\lambda}^d) d\boldsymbol{w}^d \tag{11}$$

is the joint transition distribution. Define the partially integrated Z_d for any $d = 1....D$

$$Z_d(x^D, x^{D-1}, ...x^d|x^0, \boldsymbol{\lambda}) = \int Q^T(x^D, x^{D-1}....x^1|x^0 = X^0, \boldsymbol{\lambda}) \prod_{d'=1}^{d-1} dx^{d'}. \tag{12}$$

It satisfies the recursion

$$Z_d = \int Z_{d-1} dx^{d-1}. \tag{13}$$

and the evidence is

$$Z_D = \int Z_d \prod_{d'=d}^{D-1} dx^{d'} \tag{14}$$

At this point this is analogous to a Statistical Mechanics (SM) or euclidean field theory (EFT) partition function in which all field configurations with momentum components above a cutoff have been integrated out. The equivalent of the effective action of the EFT, or the renormalized hamiltonian in the SM is $-\log Z_d$.

Now we get a similar map, where the renormalization group transformation of the internal representations can be seen. Recall the likelihood in equation 9 and use the product rule

$$L = P(x^D|x^0, \boldsymbol{w}^1, ...\boldsymbol{w}^D) = \int P(x^D|x^{D-1}w_D)P(x^{D-1}....x^1|x^0, \boldsymbol{w}^1, ...\boldsymbol{w}^D) \prod_{d=1}^{D-1} dx^d$$

and finally

$$L = P(x^D|x^0, \boldsymbol{w}^1, ...\boldsymbol{w}^D) = \int \prod_{d=1}^{D-1} P(x^{d+1}|x^d, \boldsymbol{w}^{d+1})dx^d$$

Since the prior is also a product, then the partition function $Z_D = Z_D(x^D = y|x^0 = X^0, \{\lambda^d\})$ is given by

$$Z_D = \int \prod_{d=1}^{D} Q_d(\boldsymbol{w}^d|\lambda^d)P(x^d|x^{d-1}, \boldsymbol{w}^d) \prod_{d=1}^{D} dx^{d-1}d\boldsymbol{w}^d \tag{15}$$

We integrate over x_0 and x^D with the constraints that their distribution are deltas at the input X^0 and output y.

$$Z_D = \prod_{d=1}^{D} \int d\boldsymbol{w}^d \left[\int dx^{d-1} Q_d(\boldsymbol{w}^d|\lambda^d)P(x^d|x^{d-1}, \boldsymbol{w}^d) \right]$$

Define the evidence up to a given layer $\rho(x^d)$, with initial condition $\rho(x^0) = \delta(x^0 - X^0)$ and the map

$$\rho(x^{d+1}) = \int \rho(x^d)P(x^{d+1}|x^d\boldsymbol{w}^{d+1})Q_{d+1}(\boldsymbol{w}^{d+1}|\lambda^{d+1})dx^d d\boldsymbol{w}^{d+1} \tag{16}$$

The last step for the map of a network of depth D is for $x^D = y$ leading to the evidence of the model defined by the architecture of the network with weight and hyperparameters given by the set of λ_d:

$$Z_D(y) = \rho(x^D) = \int \rho(x^{D-1})P(x^D|x^{D-1}\boldsymbol{w}^D)Q_D(\boldsymbol{w}^D|\lambda^D)dx^{D-1}d\boldsymbol{w}^D \tag{17}$$

Define a layer to layer transition distribution

$$Q_{d-1}^T(x^d|x^{d-1}\lambda^d) = \int P(x^d|x^{d-1}, \boldsymbol{w}^d)Q_d(\boldsymbol{w}^d|\lambda^d)d\boldsymbol{w}^d \tag{18}$$

$$\tag{19}$$

then, we have a map that gives the evidence after d layers as an integral over internal representations at layer $d-1$ of the evidence at layer $d-1$ with a kernel Q^T that implements an aggregation RG-like step:

$$\rho(x^d) = \int dx^{d-1}\rho(x^{d-1})Q_{d-1}^T(x^d|x^{d-1}, \lambda^d) \tag{20}$$

We have obtained two RG-like maps, Equations (13) and (20). Z_d depends on all internal representations from layer d to D and on all the hyperparameters $\boldsymbol{\lambda}$. The simpler ρ_d only depends on the internal representation at layer d and on the hyperparameters of the previous layers. The map for Z_d is simpler and the map for ρ_d requires, at each step the input on the transition distribution $Q^T(x^d|x^{d-1}, \lambda^d)$. The transition distribution describes the renormalization group like transformation implemented by the neural network that takes the internal representation at one layer to the next. It is simple to see that

$$Z_d = \rho(x^d) \prod_{d' \geq d}^{D} Q^T(x^{d'+1}|x^{d'}\lambda^d) \tag{21}$$

3.1. Generalized RG Differential Equation of a Neural Network in the Continuous Depth Limit

The layer index is obviously discrete, but we can take the continuous limit, where now layers are represented by a time like τ variable. A discrete variable i still labels the units. The evidence at depth τ is related to the evidence at depth τ_0 by a generalization of Equation (20):

$$\rho(\boldsymbol{x},\tau) \quad = \quad \int Q^T(\boldsymbol{x}(\tau)|\boldsymbol{x}'(\tau_0),\lambda)\rho(\boldsymbol{x}',\tau_0)D\boldsymbol{x}', \tag{22}$$

where the integration measure $D\boldsymbol{x} = \prod_i dx_i$. The distribution $Q^T(\boldsymbol{x}(\tau)|\boldsymbol{x}'(\tau_0),\lambda)$ is the probability, that a network with parameters λ, conditional on being in state $\boldsymbol{x}'$ at τ_0 has an internal representation $\boldsymbol{x}$ at depth τ. It must satisfy the composition law

$$Q^T(\boldsymbol{x}(\tau+\Delta\tau)|\boldsymbol{x}'(\tau_0),\lambda) \quad = \quad \int Q^T(\boldsymbol{x}(\tau+\Delta\tau)|\boldsymbol{z}(\tau),\lambda)Q^T(\boldsymbol{z}(\tau)|\boldsymbol{x}'(\tau_0),\lambda)D\boldsymbol{z}$$

For a deterministic neural network, conditional on the weights $\boldsymbol{w}$, the evolution of the internal representation is given by the transfer function. To obtain a well behaved limit it is supposed to vary slowly:

$$x_i(\tau+\Delta\tau) = T_i(\boldsymbol{x}(\tau),\boldsymbol{w}) = x_i(\tau) + \Delta\tau\tilde{b}_i(\boldsymbol{x}(\tau),\boldsymbol{w}), \tag{23}$$

so that interpretation of $\tilde{\boldsymbol{b}}$ is the gradient of the transfer function. The transition distribution is

$$Q^T(\boldsymbol{x}|\tau,\boldsymbol{x}',\tau_0,\lambda) \quad = \quad \int \prod_{\tau'\in[\tau_0,\tau]} \delta\left(\boldsymbol{x}(\tau+\Delta\tau) - T(\boldsymbol{x}'(\tau),\boldsymbol{w})\right) Q(\boldsymbol{w}|\lambda,\tau)d\boldsymbol{w}_{\tau'}, \tag{24}$$

obtained by integrating over all configuration of the weights in the slice. We have chosen a Gaussian family to represent the informational state of the network, which now takes the form of a product of Gaussians for all τ slices:

$$Q(\boldsymbol{w}|\lambda,\tau) \propto \prod_\tau \exp -\frac{1}{2}\{\Delta\boldsymbol{w}\cdot C_\tau^{-1}\cdot\Delta\boldsymbol{w}\}$$

where $\Delta\boldsymbol{w} = \boldsymbol{w} - \hat{\boldsymbol{w}}_\tau$ and $\lambda = \{\hat{\boldsymbol{w}}_\tau, C_\tau\}$ for all values of τ, but only the hyperparameters of the particular slice under consideration matters. To define the continuous limit we impose that the limits below exit:

$$\begin{aligned}
\lim_{\Delta\tau\downarrow 0}\frac{1}{\Delta\tau}\int Q^T(\boldsymbol{x}|\tau+\Delta\tau,\boldsymbol{x}',\tau,\lambda)(\boldsymbol{x}-\boldsymbol{x}')D\boldsymbol{x} \quad &= \\
\mathbb{E}_{\boldsymbol{w}}[\tilde{b}(\boldsymbol{x}(\tau),\boldsymbol{w}] = \boldsymbol{b}(\boldsymbol{x}',\tau,\lambda), & \\
\lim_{\Delta\tau\downarrow 0}\frac{1}{\Delta\tau}\int Q^T(\boldsymbol{x}|\tau+\Delta\tau,\boldsymbol{x}',\tau,\lambda)(x_i-x_i')(x_j-x_j')D\boldsymbol{x} \quad &= \\
\mathbb{E}_{\boldsymbol{w}}[\tilde{b}_i(\boldsymbol{x}(\tau),w)\tilde{b}_j(\boldsymbol{x}(\tau),\boldsymbol{w})] = B_{ij}(\boldsymbol{x}',\tau,\lambda). &
\end{aligned} \tag{25}$$

At each layer the drift vector $\boldsymbol{b}(\boldsymbol{x}',\tau,\lambda)$ is the expected value of the change in internal representation and the diffusion matrix $B_{ij}(\boldsymbol{x}',\tau,\lambda)$ to the expectation of quadratic change, which are related to the expected values of the gradient and Hessian of the transfer function respectively. As usual, take the time derivative of the expected value, with respect to $Q^T(\boldsymbol{x}|\boldsymbol{x}',\lambda)$ of a well behaved test function $g(\boldsymbol{x})$. Taylor expand $g(\boldsymbol{x})$ around $\boldsymbol{x}'$ and integrate by parts, use that $g(\boldsymbol{x})$ is arbitrary and obtain that Q^T satisfies a parabolic PDE and so does the evidence (see Equation (22))

$$\frac{\partial\rho(\boldsymbol{x},\tau)}{\partial\tau} = -\frac{\partial}{\partial x_i}(b_i(\boldsymbol{x},\tau,\lambda)\rho(\boldsymbol{x},\tau)) + \frac{1}{2}\frac{\partial^2}{\partial x_i\partial x_j}(B_{ij}(\boldsymbol{x},\tau,\lambda)\rho(\boldsymbol{x},\tau)). \tag{26}$$

The long time limit of Equation (26) is the predictive distribution $\rho(\boldsymbol{y},\tau = D) = P(\boldsymbol{y}|\boldsymbol{x}_0,\lambda)$. Equation (26) is a generalization of an analogous diffusion equation which appears in Wilson's

incomplete integration formulation of the renormalization group (e.g., [2]). It extends the type of transformation by permitting that the transformations that leads from τ to $\tau + d\tau$ are not a simple spatial average, which would eliminate high spatial frequency components. Instead, the transformations are mediated by the weights $\hat{w}$. It differs from the usual statistical mechanics or field theories also in the following sense. In those approaches, the transformation $\hat{w}$ is known and uniform and the aim is to obtain the final ρ_D, which describes the infrared limit or the thermodynamics of the theory. In supervised learning in neural networks, the starting point, defined by the input X^0 and the output Y are given. The problem is to find the correct set of weights $\hat{w}$ that implements the correct input-output association. There are two regimes for the neural network. In the learning phase the set of examples is a set of microscopic-macroscopic variables that describe a task. The aim of learning is to determine the appropriate generalized RG transformation that maps from the microscopic description to the macroscopic. After learning, the network is used to find out, for the current RG transformation, the unknown macroscopic generalized thermodynamics or infrared properties associated to the microstate. The next step is to derive optimized learning algorithms, from the solutions of Equation (26) and the EDNNA learning described by (7) and (8).

References

1. Goodfellow, I.; Bengio, Y.; Courville, A. *Deep Learning*; MIT Press: Cambridge, MA, USA, 2016. Available online: http://www.deeplearningbook.org (accessed on 19 November 2019).
2. Wilson, K.G.; Kogut, J. The renormalization group and the ϵ expansion. *Phys. Rep.* **1974**, *12*, 75–199. doi:doi:10.1016/0370-1573(74)90023-4.
3. Bény, C. Deep Learning and the Renormalization Group. Available online: https://arxiv.org/abs/1301.3124 (accessed on 19 November 2019).
4. Mehta, P.; Schwab, D.J. An exact mapping between the Variational Renormalization Group and Deep Learning. *arXiv* **2014**, arXiv:1410.3831. [arXiv:1410.3831].
5. Koch-Janusz, M.; Ringel, Z. Mutual information, neural networks and the renormalization group. *Nat. Phys.* **2018**, *14*, 578–582. doi:10.1038/s41567-018-0081-4.
6. Li, S.H.; Wang, L. Neural Network Renormalization Group. *Phys. Rev. Lett.* **2018**, *121*, 260601. doi:10.1103/PhysRevLett.121.260601.
7. Lin, H.W.; Tegmark, M.; Rolnick, D. Why Does Deep and Cheap Learning Work So Well? *J. Stat. Phys.* **2017**, *168*, 1223–1247. doi:10.1007/s10955-017-1836-5.
8. Kinouchi, O.; Caticha, N. Optimal generalization in perceptrons. *J. Phys. A* **1992**, *25*, 6243.
9. Biehl, M.; Riegler, P. On-Line Learning with a Preceptron. *Europhys. Lett.* **1994**, *28*, 525.
10. Kinouchi, O.; Caticha, N. Lower Bounds for Generalization with Drifting Rules. *J. Phys. A* **1993**, *26*, 6161.
11. Copelli, M.; Caticha, N. On-line learning in the Committee Machine. *J. Phys. A* **1995**, *28*, 1615.
12. Vicente, R.; Caticha, N. Functional optimization of online algorithms in multilayer neural networks. *J. Phys. A Gen. Phys.* **1997**, *30*. doi:10.1088/0305-4470/30/17/002.
13. Caticha, N.; de Oliveira, E. Gradient descent learning in and out of equilibrium. *Phys. Rev. E* **2001**, *63*, 061905.
14. Opper, M. *A Bayesian Approach to Online Learning in On-line Learning in Neural Networks*; Saad, D., Ed.; Cambridge University Press: Cambridge, UK, 1998.
15. Solla, S.A.; Winther, O. Optimal online learning: A Bayesian approach. *Comput. Phys. Commun.* **1999**, *121–122*, 94–97.
16. Caticha, N.; Vicente, R. Agent-based Social Psychology: From Neurocognitive Processes to Social Data. *Adv. Complex Syst.* **2011**, *14*, 711–731.
17. Vicente, R.; Susemihl, A.; Jerico, J.; Caticha, N. Moral foundations in an interacting neural networks society: A statistical mechanics analysis. *Phys. A Stat. Mech. Appl.* **2014**, *400*, 124–138. doi:doi:10.1016/j.physa.2014.01.013.

18. Caticha, N.; Cesar, J.; Vicente, R. For whom will the Bayesian agents vote? *Front. Phys.* **2015**, *3*. doi:10.3389/fphy.2015.00025.
19. Caticha, N.; Alves, F. Trust, law and ideology in a NN agent model of the US Appellate Courts. In *ESANN 2019 Proceedings, Proceedings of the European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning, Bruges, Belgium, 24–26 April 2019*; pp. 511–516. ISBN 978-287-587-065-0. Available online: https://www.elen.ucl.ac.be/Proceedings/esann/esannpdf/es2019-72.pdf (accessed on 19 November 2019).