Partial local entropy and anisotropy in deep weight spaces

Daniele Musso*

Departamento de Física de Partículas,
Universidade de Santiago de Compostela (USC)
Instituto Galego de Física de Altas Enerxías (IGFAE)
E-15782 Santiago de Compostela, Spain
and
Inovalabs Digital S.L. (TECHEYE), E-36202 Vigo, Spain

Abstract

We refine a recently-proposed class of local entropic loss functions by restricting the smoothening regularization to only a subset of weights. The new loss functions are referred to as partial local entropies. They can adapt to the weight-space anisotropy, thus outperforming their isotropic counterparts. We support the theoretical analysis with experiments on image classification tasks performed with multi-layer, fully-connected neural networks. The present study suggests how to better exploit the anisotropic nature of deep landscapes and provides direct probes of the shape of the wide flat minima encountered by stochastic gradient descent algorithms. As a by-product, we observe an asymptotic dynamical regime at late training times where the temperature of all the layers obeys a common scaling rule.

*daniele.musso@usc.es, mudaniele@yahoo.com
1 Introduction

Recent studies on the weight space of deep neural networks [1,2] have highlighted the existence of rare subdominant clusters of configurations which yield a high test accuracy. Although these clusters constitute a deviation from typicality, they are efficiently encountered by stochastic gradient descent (SGD) algorithms and correspond to wide valleys of suitable loss functions, such as cross entropy [3].

An analogous circumstance occurs in the context of constraint satisfaction problems, where the chase after clusters of solutions is improved when the loss function gets supplemented by a term that encourages a local high density of solutions [4]. In order to find the number of solutions contained in a vicinity of a specific weight configuration, one can define a local solution-counting functional, namely, a local entropy.

Classification tasks performed by means of quantized neural networks (where the weights are discrete) can be interpreted as constraint satisfaction problems. There are however two reasons to generalize the concept of local entropy: First, classification problems are typically required to reach a high but not necessarily perfect accuracy; second, they are often approached with machines that have continuous weights. The strict counting of solutions of constraint satisfaction problems can therefore be relaxed to just an incentive which encourages the presence of high-accuracy configurations throughout a whole region in weight space. A local averaging of the loss, for instance, is expected to have such an effect, but other deformations of the loss yielding a local smoothening can be valid choices too.

A specific smoothening procedure of the loss function can be enforced by means of a spatial convolution with an Euclidean heat kernel, whose spread is controlled by a parameter $\gamma$,

$$
\mathcal{F}(\beta, \gamma; W) = -\log \int d^N W' \exp \left( -\beta \mathcal{L}(W') - \frac{\gamma}{2} ||W - W'||^2 \right),
$$

(1)

1Up to the numerical precision employed.
where both $W$ and $W'$ parametrize the $N$-dimensional weight space, $||.||_2$ represents the Euclidean norm and $\mathcal{L}$ is a generic loss function; adopting an energetic interpretation of the loss, the parameter $\beta$ corresponds to an inverse temperature. In the limit $\beta \to \infty$, the integral in (1) reduces to a weighted counting of the configurations $W'$ where the weighting decreases exponentially with the distance from $W$.\footnote{The functional $\mathcal{F}(\beta, \gamma; W)$ defined in (1) can be interpreted in analogy to a thermodynamical potential; as such, it should be referred to as local free entropy, this extra connotation is sometime omitted to avoid clutter.}

The smoothening introduced by (1) is isotropic in weight space. However, when optimizing with SGD, the gradient noise depends in general on both the position and the direction, this being actually a key factor for the success of SGD algorithms\footnote{One can rephrase such combinatorial complexity in terms of correlations among the input channels: the neurons in the first layer are sensitive to the inputs individually, so they respond to 1-point correlations; the neurons belonging to the $n$-th layer, instead, are sensitive to $n$-point correlations, that is, the joint correlations of $n$ inputs.}. Therefore, it is natural to expect that a refinement of the smoothening functional able to suitably exploit the anisotropy of gradient noise can significantly improve its regularizing effects. Besides, such refinement can furnish an interesting new probe of the weight space.

The present paper focuses on partial, entropic and local smoothening, namely a smoothening analogous to (1) applied to just a subsets of weights. This allows one to address weight-space anisotropy in a direct and active way. We will loosely adopt the term partial local entropy to convey this idea irrespective of the details of the particular smoothening technique, as long as it corresponds to an incentive to local high density of high-accuracy configurations restricted to a subset of weights.\footnote{To this regard, two relevant examples are Kaiming weight initialization\footnote{To this regard, two relevant examples are Kaiming weight initialization and regularization by means of anisotropic noise injection.} and regularization by means of anisotropic noise injection.}

2 Anisotropy in weight space

By definition the neurons of a deep network are arranged on different layers and such architecture imposes a natural hierarchy among them, according to their depth within the network. In a fully-connected setting, the receptive field of each neuron coincides with the whole input, however deeper neurons are fed with signals that have been pre-processed by lower-lying neurons. Roughly, while the neurons in the first layer compute a weighted sum of the network inputs, the neurons in the second layer compute a weighted sum of the outputs of the first layer, that is, a weighted sum of a weighted sum of the network inputs. Such compositional nature of the operation performed by each subsequent layer suggests that the depth of the network corresponds to a hierarchy in combinatorial complexity\footnote{One can rephrase such combinatorial complexity in terms of correlations among the input channels: the neurons in the first layer are sensitive to the inputs individually, so they respond to 1-point correlations; the neurons belonging to the $n$-th layer, instead, are sensitive to $n$-point correlations, that is, the joint correlations of $n$ inputs.}. Any isotropic assumption about the weight space neglects this structural hierarchy, thereby it should be regarded with caution if not even suspicion.

Careful consideration of the weight space hierarchical anisotropy have led to important insight about the inner workings of neural networks\footnote{To this regard, two relevant examples are Kaiming weight initialization and regularization by means of anisotropic noise injection.} as well as improvements in their optimization.\footnote{To this regard, two relevant examples are Kaiming weight initialization and regularization by means of anisotropic noise injection.} Gradient noise depends on both position and direction and its covariance matrix is correlated to the Hessian matrix of the loss function, which leads SGD to escape exponentially fast from sharp minima\footnote{To this regard, two relevant examples are Kaiming weight initialization and regularization by means of anisotropic noise injection.}. Thus, it is fair to consider weight-space anisotropy as one of the main features at the root of the effectiveness of SGD algorithms in reaching high test accuracy and generalization.

2.1 Layer temperature and asymptotic exponential cooling

The learning dynamics of a deep neural network trained with SGD is in general a complex process. The system is out of equilibrium and, given the dependence of the gradient noise on the position in weight space, one cannot schematize the training as the evolution of a system in
contact with an equilibrium thermal reservoir. Nonetheless, it is still possible to try to define a temperature as the variance of the gradient noise when schematizing the training evolution in terms of a Brownian motion \[7^{11,12}\]. More precisely, one has to focus on the covariance matrix \(D(W)\) characterizing the stochastic Wiener process.

Let us focus for a moment on a specific fixed point \(W^*\) in weight space. Given the anisotropy of \(D(W^*)\), it is impossible to define a unique temperature characterizing all directions, but one can in principle still define a temperature for each direction. Since we are working in a space with very high dimensionality, this is hardly of any help. However, we should recall that there is a natural grouping of the directions in weight space provided by the layered structure of the network. Furthermore, it is possible to define layer variables which average over the weights belonging to the same layer. One can consider fluctuations of such layer variables that, due to the averaging over a layer, are expected to be stabler and reflect the hierarchy of the architecture. Accordingly, one can define a layer temperature corresponding to the variance of such layer averaging of gradients. This corresponds to regarding the layers as if they were the individual units of a neural network; despite being a crude approximation, this helps gaining useful insight about the training dynamics \[14\].

The layer temperature is a characterization of the noise of the training signal \(s_I\) through layer \(I\), defined as

\[
s_I = \frac{1}{N_I} \sum_{\omega \in \Omega_I} \nabla_{\omega} \mathcal{L}(W),
\]

where \(\Omega_I\) denotes the set of \(N_I\) weights connecting the \(I\)-th layer with its inputs and \(\mathcal{L}(W)\) is the loss evaluated at the weight configuration \(W\). The training signal and its noise evolve during optimization and it is possible to isolate different regimes in the training dynamics. In \[14\] the authors observed that a possibly generic dynamic transition occurs when the signal/noise ratio switches from being initially dominated by the signal to being later dominated by noise. This occurs quite abruptly (in terms of optimization time) and just after the moment when the training signal attains its maximum value, see Figure 1.

The numerical studies that we performed suggests the generic presence of a further dynamic transition, occurring at later stages of the training. This eventual regime is characterized by an exponential decay of both signal and noise for all layers. Interestingly, the exponential contraction of the signal and the noise for all the layers is characterized by a single decay rate. At late times, the hierarchy between layers is therefore preserved and gets frozen: the dynamics of all the layers can in fact be described factorizing the common exponential decay.

As already stressed, even if the layer-wise account gives a very coarse-grained picture of the actual training dynamics, still it confirms the importance of anisotropy throughout the whole training process, including at asymptotic late times where the in-sample loss and the test error have long stabilized.

\[5\]We refer to \[12\] for the definition of the covariance matrix \(D(W)\). The analysis of a Brownian motion by means of the Fokker-Planck equation encodes both the noise anisotropy and its dependence on position through the covariance matrix \(D(W)\) \[12,13\].

\[6\]We underline that a direct analysis of the variance of the gradient noise for the single weights shows that in general the weights belonging to the same layer can not be characterized by a common temperature. Said otherwise, the possibility of defining a layer temperature does not imply thermal isotropy within the subspace spanned by the weights of the same layer.

\[7\]Interpreting the noise as a temperature and adopting a renormalization group language, the eventual exponential cooling is suggestive of an infrared fix point, where quantities evolve by a common rescaling without distortion at asymptotic low energies. Here it emerges a potential connection to studies of neural networks under the perspective of scaling rules, see for instance \[15,16\]. It is relevant to stress that Figure 1 has been obtained without adopting weight-decay regularization.
Figure 1: The absolute value of the training signal \(|s_I|\) defined in (2) where \(I \in \{1, 2, 3, 4\}\) labels the layers of the network is represented with solid lines; the dashed lines represent the associated standard deviations. The plot depicts a long training of a 4-layer fully-connected neural network on MNIST. Three distinct dynamical regimes emerge: 1) an early signal-dominated regime; 2) an intermediate noise-dominated regime; 3) an eventual noise-dominated regime where all quantities decay exponentially with a common rate (the vertical axis is logarithmic).

3 Partial local free entropy

For the sake of generality, the present section is rather technical. The reader who is just interested in the specific losses used in the experiments can jump to Section 4 and focus on the loss functions (22) and (23) without missing the core ideas.

We consider the cross-entropy loss \(\mathcal{L}_{c.e.}(W)\) as the baseline function to be smoothened; \(W\) is a vector indicating a configuration in weight space. We consider \(y\) additional configurations \(W + \Delta W^a\) with \(a = 1, ..., y\), shifted by a random vector \(\Delta W^a\). The loss corresponding to each configuration is supplemented by an additional term measuring its distance from the unperturbed point \(W\). For the moment we let the distance function \(d_{R,k}(\Delta W^a)\) be arbitrary but we assume it depends on two parameters, to be specified later. We consider the new loss

\[
M(R, k, y; W) \equiv -\log \left\{ \frac{1}{y + 1} \left[ \sum_{a=1}^{y} e^{-\mathcal{L}_{c.e.}(W + \Delta W^a)} - d_{R,k}(\Delta W^a) \right] \right\},
\]

(3)

normalized with respect to the number of sampling points \(y + 1\). Roughly, the loss \(M\) amounts to the logarithm of an average of exponentials. In the case of just one sampling point, \(y = 0\), \(M\) coincides with the baseline loss,

\[
M(R, k, y = 0; W) = \mathcal{L}_{c.e.}(W).
\]

(4)
We choose the following distance function
\[ d_{R,k}(\Delta W) \equiv -\log \prod_{i=1}^{N} \left[ (1 - \frac{1}{1 + e^{-2k(\Delta W_i - R)}}) \frac{1}{1 + e^{-2k(\Delta W_i + R)}} \right] , \]
which depends on two real parameters, \( R \) and \( k \). In the \( k \to \infty \) limit, the kernel
\[ K_{R,k}(\Delta W) \equiv e^{-d_{R,k}(\Delta W)} , \]
reduces to the characteristic function of the \( N \)-dimensional hyper-cube \( H_{W,R} \) centered in \( W \) with edges \( 2R \) long.\(^6\)

Thus, the parameter \( R \) represents the effective linear size of the support of the kernel \((6)\), while \( k \) controls its sharpness, see Figure \(2\). In the infinite sharpness limit, \( k \to \infty \), the random displacement vectors \( \Delta W^a \) in \((3)\) are sampling the hyper-cube \( H_{W,R} \) uniformly.

Taking an infinite number of sampling points,
\[ M(R, k, y; W) \xrightarrow{y \to +\infty} F(R, k; W) , \]
where
\[ F(R, k; W) \equiv -\log \int d^N W' e^{-\mathcal{L}_{c.e.}(W')} K_{R,k}(W' - W) , \]
defines a parametric family of local free entropies, in analogy with \((1)\). Taking the \( k \to \infty \) limit of \((10)\), one obtains
\[ \lim_{k \to +\infty} F(R, k; W) = -\log \int_{H_{W,R}} d^N W' e^{-\mathcal{L}_{c.e.}(W')} . \]

To recapitulate, in the limit of large number of sampling points, \( y \to \infty \), the loss \( M(R, k, y; W) \) approximates a parametric family of free local entropy functions \((10)\) parametrized by the effective linear size \( R \) of the smoothening region (in weight-space) and the sharpness \( k \) of the associated kernel \((6)\).

In order to define partial local free entropies we have just to generalize the passages above to the case where only a subset of weights is smoothened over. We can define a discrete indicator function \( U \) taking values in \{0, 1\} and defined on the \( N \) dimensions of weight space: it takes value 1 on the directions along which we smoothen the loss, and 0 on the remaining directions in weight space. Thinking to \( U \) as an \( N \)-dimensional vector, it provides an un-normalized projector onto the subset of weights considered for smoothening. We can thus define a restricted version of the distance function \( d_{R,k}(\Delta W) \),
\[ d_{R,k}^{[U]}(\Delta W) \equiv d_{R,k}((\Delta W \cdot U) U) , \]
\[8\]Recall that the Heaviside step function \( \Theta(x) \) can be obtained as the limit of infinite sharpness for a sigmoid function, namely
\[ \Theta(x) = \lim_{k \to +\infty} \frac{1}{1 + e^{-2kx}} . \]

\[9\]The particular local free entropy specified in \((1)\) is associated to a different choice of distance, namely
\[ d(\gamma; \Delta W) = \gamma \|\Delta W\|^2 . \]

\[10\]
where \( \cdot \) indicates the scalar product of \( \mathbb{R}^N \) in the \( N \)-dimensional weight space.

Adopting the restricted distance (13), we can repeat the same steps as above: first consider

\[
\mathcal{M}^{[U]}(R, k, y; W) \equiv -\log \left\{ \frac{1}{y+1} \left[ e^{-\mathcal{L}_{c.e.}(W)} + \sum_{a=1}^{y} e^{-\mathcal{L}_{c.e.}(W+\Delta W^a)} - d_{R,k}^{[U]}(\Delta W^a) \right] \right\},
\]

(14)

then take the \( y \to \infty \) limit,

\[
\mathcal{M}^{[U]}(R, k, y; W) \xrightarrow{y \to \infty} \mathcal{F}^{[U]}(R, k; W), \tag{15}
\]

where

\[
\mathcal{F}^{[U]}(R, k; W) \equiv -\log \int d^N W' e^{-\mathcal{L}_{c.e.}(W')} K_{R,k}^{[U]}(W'-W), \tag{16}
\]

represents a parametric family of partial local free entropies. Eventually, take the \( k \to \infty \) limit,

\[
\mathcal{F}^{[U]}(R; W) \equiv \lim_{k \to \infty} \mathcal{F}^{[U]}(R, k; W), \tag{17}
\]

where

\[
\mathcal{F}^{[U]}(R; W) \equiv -\log \int_{H_{W,R}^{[U]}} d^N W' e^{-\mathcal{L}_{c.e.}(W')}; \tag{18}
\]

the integration region \( H_{W,R}^{[U]} \) is a hyper-cube extended only in the directions along which \( U \) is non-null.

### 3.1 A simpler entropic loss

It is interesting to seek for a simpler loss which could somehow preserve the smoothening effect of partial local free entropy. To this purpose, one can define an averaged loss over an \( N \)-dimensional vicinity in weight space –this imitating the effects of local entropy– or to a lower-dimensional vicinity – this instead imitating partial local entropy. We focus on the latter case and define

\[
\mathcal{L}^{[U]}(R, k, y; W) \equiv \frac{1}{y+1} \left[ \mathcal{L}_{c.e.}(W) + \sum_{a=1}^{y} \mathcal{L}_{c.e.}(W + \Delta W^a) K_{R,k}^{[U]}(\Delta W^a) \right]. \tag{19}
\]
Considering the \( k \to \infty \) limit one obtains

\[
\hat{L}^{[U]}(R, y; W) \equiv \frac{1}{y+1} \left[ L_{c.e.}(W) + \sum_{a=1}^{y} L_{c.e.}(W + \Delta^{[U]}W^a) \right],
\]

(20)

where \( \Delta^{[U]} \) means simply that the random vectors are sampled within the hyper-cube \( H_{W,R}^{[U]} \), centered in \( W \) and extending along the direction indicated by the vector \( U \), its edges being \( 2R \) long. In the limit of infinite samples, we have

\[
\hat{L}^{[U]}(R, W) \xrightarrow{y \to +\infty} \int_{H_{W,R}^{[U]}} d^NW' \; L_{c.e.}(W'),
\]

(21)

and the loss reduces to a simple local average along a subset of directions in weight space.

### 4 Experiments on MNIST and Fashion-MNIST

The focus of the experiments is on layer-wise partial entropy regularizations for multi-layer, fully-connected neural networks trained on image classification tasks. Namely, we considered partial local entropies where the subset of weights chosen for smoothening coincides with whole layers. We consider the 10-class classification tasks associated with MNIST [18] and Fashion-MNIST [19] datasets, whose input images are 28 pixels wide and 28 pixels height. We consider both 2-layer and 3-layer fully-connected neural networks with continuous weights having a further 10-neuron output layer. All layers except the last have 784 = 28\(^2\) neurons and are structurally identical, apart from their different depth within the network. The following hyper-parameters have been kept fixed for all the experiments: learning rate \( \eta = 0.0001 \), momentum \( \mu = 0.9 \), mini-batch size 256 and trained for 120 epochs.

We considered two loss functions, a partial local exponential average loss (PLEA)

\[
L_{\text{PLEA}}(W) = -\log \left\{ \frac{1}{1 + y} \left[ e^{-L_{c.e.}(W)} + \sum_{a=1}^{y} e^{-L_{c.e.}(W + \Delta W^a)} \right] \right\},
\]

(22)

and a partial local average loss (PLA)

\[
L_{\text{PLA}}(W) = \frac{1}{1 + y} \left[ L_{c.e.}(W) + \sum_{a=1}^{y} L_{c.e.}(W + \Delta W^a) \right],
\]

(23)

where \( L_{c.e.} \) is the cross-entropy loss and \( \Delta W^a \) is a random vector sampled in a vicinity of \( W \). Such a vicinity is a hyper-cube centered in \( W \) with edge \( 2R \) and extending only along a subspace of the \( N \)-dimensional weight space. Notice that in this way the regularizations of the cross-entropy \( L_{c.e.} \) given by (22) and (23) enforce an anisotropic bias.

In the experiments reported below we consider only subspaces associated to one or more layers at a time. Apart from the entropic smoothening, we do not enforce any further regularization, in particular we do not use weight decay.

---

\(^{10}\) The loss function defined in (19) can be related to the robust ensemble studied in [5], which in turns is similar to the elastic averaging proposed in [17].

\(^{11}\) We performed the experiments with single floating point numerical precision.

\(^{12}\) The losses (22) and (23) correspond to infinite sharpness limits, \( k \to \infty \), of (14) and (19), respectively. See Section 3 for more details.

\(^{13}\) Throughout the present paper the weight space spanned by \( W \) is formed only by the synaptic coefficients connecting different layers, while it excludes biases. Despite these latter are present and trained over, we do not smooth over them.
4.1 Results

The experiments suggest two main conclusions:

- In general, the entropic regularizations \( (22) \) and \( (23) \) improve test accuracy. The effect increases rapidly with the size \( R \) of the smoothening region, up to a maximum size beyond which performance gets degraded.

- When implemented on suitable subsets of weights (e.g. single layers), the entropic regularizations outperform significantly their isotropic counterparts.

The first point means that smoothening improves performance up to a point beyond which its averaging effect distorts the original loss landscape too heavily. The second point means that the strong differences in the role played by the various weights affect the loss landscape and the effectiveness of regularization. This implies that the shape of the wide flat minima encountered by SGD optimization is relevant, not only their extension. Another generic conclusion suggested by the experiments is that the layer-wise entropic regularization is more effective when performed on deeper levels. This harmonizes with the intuitive idea that deeper weights are associated to more complex features, which –in a reliable classification– should be progressively more robust.

An important detail of the experimental setups is that all layers have the same number of neurons, 784. Thus, when comparing quantities associated to different layers, we are actually probing the mere effect of depth. A direct comparison between structurally different layers would instead be more difficult to interpret.

4.2 2-layer fully-connected neural network on Fashion-MNIST

We considered 2-layer, fully-connected neural networks adopting both PLEA loss function \( (22) \) and PLE loss function \( (3) \). The results obtained with the two loss functions are qualitatively analogous.

We measured the test accuracy reached by three versions of the same 2-layer network as we moved the regularization radius \( R \), the three versions differ simply by the choice of the weight subspace considered for smoothening: either (i) the whole first layer; (ii) the whole second layer; (iii) both layers (isotropic choice). The results are reported in Figure 3 and Figure 4 (left plot). Regularization on the 2nd layer alone proved to be the best strategy for both choices of loss functions and in the entire range of \( R \) probed by the experiments. The isotropic regularization can outperform the regularization on the 1st layer alone, but only at very small values for \( R \). In fact, the isotropic choice leads soon to degraded results as \( R \) increases, while the single-layer regularizations continue to improve the test accuracy, showing a saturating behavior.

4.3 3-layer fully-connected neural network on MNIST

The experiments on the 3-layer fully-connected neural networks confirm and extend the results obtained for its 2-layer counterpart. They are depicted in Figure 5 (right plot). In particular, the isotropic choice proves to be the worst among all the possible choices of subsets of the width of the smoothening region \( R \) is sufficiently big. Moreover, there is an articulated interplay of regimes as \( R \) varies: at the lowest values of \( R \) the best choice consists in regularizing with respect to the 1st and 3rd layers jointly; at large values of \( R \), regularizing with respect to the 2nd or 3rd layer alone proves to be the best choice. Also the performance hierarchy among the sub-optimal regularizations schemes changes as \( R \) moves showing a complicated structure.

\[ \text{i.e. the parameter encoding the linear size of the smoothening region; see Section 3 for details.} \]

\[ \text{Recall that we consider only subsets of weights associated to one or more whole layers.} \]
Figure 3: Comparison of the best (left) and final(right) test accuracy reached by a 2-layer fully-connected neural network on Fashion-MNIST. The lines correspond to three different PLA losses (see 23) obtained by smoothening the cross entropy respectively on the 1st, the 2nd or both layers.

Figure 4: Left plot: best test accuracy reached during training by a 2-layer fully-connected neural network over Fashion-MNIST. The three line correspond to three different PLEA regularization schemes (see Eq.(3) where smoothening is performed on the 1st layer alone, on the 2nd layer alone or on both layers, respectively. Right plot: best test accuracy reached by a 3-layer fully-connected neural network on MNIST. The lines represents different PLA regularization schemes according to an RGB color nomenclature where Red corresponds to the 1st layer, Green to the 2nd and Blue to the 3rd.
Figure 5: Comparison of the test accuracy performance obtained with a bi-layer fully-connected neural network on Fashion-MNIST and trained with PLA loss (see Eq. (23)). The lighter line refer to finer sampling, \( y = 8 \), while the darker line refers to \( y = 4 \). There is no strong sensitivity to the sample size.

4.4 Finer sampling

In order to test whether the decrease in accuracy associated to regularizing on multiple layers is due to insufficient sampling (\textit{i.e.} too small \( y \), see (22) and (23)) we repeated the experiments performed with the 2-layer fully-connected neural network on Fashion-MNIST with PLA loss doubling the number of sampling points \( y \). The results obtained with \( y = 8 \) are comparable to those obtained with \( y = 4 \), see Figure 5; this hints to the fact that the sampling of the smoothening neighborhood can not explain the poor performance of multi-layer regularization.

5 Discussion

The local smoothening of the loss function improves the chase for wide flat minima \cite{5, 12}, which is already a strength of the standard stochastic gradient descent algorithm \cite{6}. This is the reason for adopting local entropy losses to drive an efficient optimization of deep neural networks. We elaborate and refine the concept of local entropy to the purpose of leveraging the intrinsically anisotropic nature of deep weight spaces. Concretely, we propose to restrict local entropy losses to suitable sub-spaces of weights. This explores, addresses and exploits the intrinsic anisotropic nature of deep weight spaces. We showed that an anisotropic regularization can implement useful biases on the shape of the wide flat minima encountered by SGD optimization.

We have mainly explored the layer-wise implementations of partial local entropies; although there is room for finer analyses resolving smaller sub-spaces, the layer-wise approach is both natural (\textit{i.e.} well-adapted to the architecture of deep networks) and informative.

In the present paper we have applied partial entropic regularizations to particular kinds of fully-connected neural networks only, they can be however employed to the optimization of wider classes of learning machines, \textit{e.g.} autoencoders \cite{20}. In particular, the specific layer-wise entropic regularizations proposed in the present study apply in any context involving a layered neural network.
5.1 Direct analysis in the language of statistical physics

The study of local entropic regularizations is a very active research front in machine learning, especially in connection to statistical physics [1, 3–5, 17, 20–23]. Wide flat minima have been shown to be a structural characteristic of deep networks and their correlation with good generalization performance has been established [1, 3]. In some simple setups, it is even possible to estimate analytically the hyper-volume of the clusters of configurations giving rise to the relevant wide flat minima [3, 24]. The theoretical framework on which the calculations are based has been developed for the study of disordered systems in condensed matter, mainly spin glasses (see [25] and references therein). It is called replica approach and replica symmetry breaking, the two applying respectively to two different regimes separated by a clustering transition in the space of relevant minima [25].

Despite the approach being very refined, two simplifying assumptions are generically considered [28]: (i) average over (typically Gaussian) input; (ii) tree-like architectures. The first is a crucial point for the development of the replica formalism, but it essentially washes out completely the information about the dataset. This is not always undesirable, in fact it allows for the characterization of structural properties of the machines that hold true per se independently of the dataset. It however constitutes a limitation whenever the actual information provided by the input is important. Considering a tree-like architecture is very helpful to simplify the computations, in fact avoiding loops in the network often opens the possibility of exact computations by, for instance, belief propagation algorithms [3, 28]. Nevertheless, adopting a tree-like network as a proxy for a fully-connected one can be a crude simplification which is expected to deviate more significantly as the depth of the system is increased.

In order to explain the experiments described in Section 4, it would be desirable to have a direct control on the shape of the relevant clusters of weight configurations reached upon SGD training, or at least an estimation thereof. This could be seen as a refinement of the estimation of the clusters size [3, 24], as such it is likely to be a very demanding endeavour up to the point that it becomes natural to ask whether some simpler –though possibly rougher– approach is viable.

6 Acknowledgements

I would like to thank Riccardo Argurio, Javier Más, Giorgio Musso, Alfonso Ramallo and Hernán Serrano for interesting discussions.

References

[1] C. Baldassi, A. Ingrosso, C. Lucibello, L. Saglietti, and R. Zecchina, “Subdominant dense clusters allow for simple learning and high computational performance in neural networks with discrete synapses,” Physical review letters, vol. 115 12, p. 128101, 2015.

[2] P. Chaudhari, A. Choromanska, S. Soatto, Y. LeCun, C. Baldassi, C. Borgs, J. Chayes, L. Sagun, and R. Zecchina, “Entropy-SGD: Biasing Gradient Descent Into Wide Valleys,” arXiv e-prints, p. arXiv:1611.01838, Nov. 2016.

[3] C. Baldassi, E. M. Malatesta, and R. Zecchina, “Properties of the geometry of solutions and capacity of multilayer neural networks with rectified linear unit activations,” Physical Review Letters, vol. 123, Oct 2019.

An analogous transition in K-SAT problems has been studied in [26, 27].
[4] C. Baldassi, A. Ingrosso, C. Lucibello, L. Saglietti, and R. Zecchina, “Local entropy as a measure for sampling solutions in constraint satisfaction problems,” *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2016, p. 023301, 2016.

[5] C. Baldassi, C. Borgs, J. T. Chayes, A. Ingrosso, C. Lucibello, L. Saglietti, and R. Zecchina, “Unreasonable effectiveness of learning neural networks: From accessible states and robust ensembles to basic algorithmic schemes,” *Proceedings of the National Academy of Sciences*, vol. 113, p. E7655–E7662, Nov 2016.

[6] Z. Xie, I. Sato, and M. Sugiyama, “A diffusion theory for deep learning dynamics: Stochastic gradient descent escapes from sharp minima exponentially fast,” *ArXiv*, vol. abs/2002.03495, 2020.

[7] W. E, “A proposal on machine learning via dynamical systems,” *Communications in Mathematics and Statistics*, vol. 5, pp. 1–11, 02 2017.

[8] M. Riesenhuber and T. Poggio, “Riesenhuber, m. and poggio, t. hierarchical models of object recognition in cortex. nat. neurosci. 2, 10191025,” *Nature neuroscience*, vol. 2, pp. 1019–25, 12 1999.

[9] K. He, X. Zhang, S. Ren, and J. Sun, “Delving deep into rectifiers: Surpassing human-level performance on imagenet classification,” *CoRR*, vol. abs/1502.01852, 2015.

[10] Z. Zhu, J. Wu, B. Yu, L. Wu, and J. Ma, “The Anisotropic Noise in Stochastic Gradient Descent: Its Behavior of Escaping from Sharp Minima and Regularization Effects,” *arXiv e-prints*, p. arXiv:1803.00195, Feb. 2018.

[11] D. Musso, “Stochastic gradient descent with random learning rate,” 3 2020.

[12] P. Chaudhari and S. Soatto, “Stochastic gradient descent performs variational inference, converges to limit cycles for deep networks,” *CoRR*, vol. abs/1710.11029, 2017.

[13] P. [da Silva], L. [da Silva], E. Lenzi, R. Mendes, and L. Malacarne, “Anomalous diffusion and anisotropic nonlinear fokkerplanck equation,” *Physica A: Statistical Mechanics and its Applications*, vol. 342, no. 1, pp. 16 – 21, 2004. Proceedings of the VIII Latin American Workshop on Nonlinear Phenomena.

[14] R. Shwartz-Ziv and N. Tishby, “Opening the black box of deep neural networks via information,” *CoRR*, vol. abs/1703.00810, 2017.

[15] S. L. Smith and Q. V. Le, “A Bayesian Perspective on Generalization and Stochastic Gradient Descent,” *arXiv e-prints*, p. arXiv:1710.06451, Oct. 2017.

[16] U. Sharma and J. Kaplan, “A neural scaling law from the dimension of the data manifold,” 2020.

[17] S. Zhang, A. Choromanska, and Y. LeCun, “Deep learning with elastic averaging sgd,” 2014.

[18] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” *Proceedings of the IEEE*, vol. 86, pp. 2278–2324, Nov 1998.

[19] H. Xiao, K. Rasul, and R. Vollgraf, “Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms,” 2017.
[20] M. Negri, D. Bergamini, C. Baldassi, R. Zecchina, and C. Feinauer, “Natural representation of composite data with replicated autoencoders,” 2019.

[21] P. Chaudhari, A. Choromanska, S. Soatto, Y. LeCun, C. Baldassi, C. Borgs, J. Chayes, L. Sagun, and R. Zecchina, “Entropy-sgd: Biasing gradient descent into wide valleys,” 2016.

[22] C. Baldassi, R. D. Vecchia, C. Lucibello, and R. Zecchina, “Clustering of solutions in the symmetric binary perceptron,” 2019.

[23] F. Pittorino, C. Lucibello, C. Feinauer, E. M. Malatesta, G. Perugini, C. Baldassi, M. Negri, E. Demyanenko, and R. Zecchina, “Entropic gradient descent algorithms and wide flat minima,” 2020.

[24] E. Barkai, D. Hansel, and I. Kanter, “Statistical mechanics of a multilayered neural network,” Phys. Rev. Lett., vol. 65, pp. 2312–2315, Oct 1990.

[25] M. Mézard, G. Parisi, and M. Virasoro, Spin Glass Theory And Beyond: An Introduction To The Replica Method And Its Applications. World Scientific Lecture Notes In Physics, World Scientific Publishing Company, 1987.

[26] M. Mézard, G. Parisi, and R. Zecchina, “Analytic and algorithmic solution of random satisfiability problems,” Science, vol. 297, no. 5582, pp. 812–815, 2002.

[27] F. Krzakala, A. Montanari, F. Ricci-Tersenghi, G. Semerjian, and L. Zdeborová, “Gibbs states and the set of solutions of random constraint satisfaction problems,” Proceedings of the National Academy of Sciences, vol. 104, no. 25, pp. 10318–10323, 2007.

[28] M. Mézard and A. Montanari, Information, Physics, and Computation. Oxford Graduate Texts, OUP Oxford, 2009.