1-1-2018

The Multilinear Structure of ReLU Networks

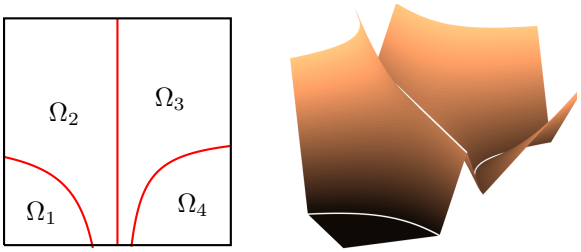
Thomas Laurent
Loyola Marymount University, thomas.laurent@lmu.edu

The Multilinear Structure of ReLU Networks

Thomas Laurent [*1] **James H. von Brecht** [*2]

Abstract

We study the loss surface of neural networks equipped with a hinge loss criterion and ReLU or leaky ReLU nonlinearities. Any such network defines a piecewise multilinear form in parameter space. By appealing to harmonic analysis we show that all local minima of such network are non-differentiable, except for those minima that occur in a region of parameter space where the loss surface is perfectly flat. Non-differentiable minima are therefore not technicalities or pathologies; they are heart of the problem when investigating the loss of ReLU networks. As a consequence, we must employ techniques from nonsmooth analysis to study these loss surfaces. We show how to apply these techniques in some illustrative cases.



(a) Parameter Space Ω (b) Loss Surface $\mathcal{L}(\boldsymbol{\omega})$

Figure 1. (a): Parameter space $\Omega = \mathbb{R}^2$ decomposes into a partition of four open cells Ω_u and a closed boundary set $\mathcal{N}$ (solid red lines). (b): The loss surface is smooth inside each Ω_u and non-differentiable on $\mathcal{N}$. It has two types of local minima, flat minima (cell Ω_1) and sharp minima (boundary between cells Ω_3 and Ω_4). The sharp minima must have non-zero loss.

1. Introduction

Empirical practice tends to show that modern neural networks have relatively benign loss surfaces, in the sense that training a deep network proves less challenging than the non-convex and non-smooth nature of the optimization would naïvely suggest. Many theoretical efforts, especially in recent years, have attempted to explain this phenomenon and, more broadly, the successful optimization of deep networks in general (Gori & Tesi, 1992; Choromanska et al., 2015; Kawaguchi, 2016; Safran & Shamir, 2016; Mei et al., 2016; Soltanolkotabi, 2017; Soudry & Hoffer, 2017; Du et al., 2017; Zhong et al., 2017; Tian, 2017; Li & Yuan, 2017; Zhou & Feng, 2017; Brutzkus et al., 2017). The properties of the loss surface of neural networks remain poorly understood despite these many efforts. Developing of a coherent mathematical understanding of them is therefore one of the major open problems in deep learning.

We focus on investigating the loss surfaces that arise from feed-forward neural networks where rectified linear units (ReLUs) $\sigma(x) := \max(x, 0)$ or leaky ReLUs $\sigma_\alpha(x) := \alpha \min(x, 0) + \max(x, 0)$ account for all nonlinearities present in the network. We allow the transformations defining the hidden-layers of the network to take the form of fully connected affine transformations or convolutional transformations. By employing a ReLU-based criterion we then obtain a loss with a consistent, homogeneous structure for the nonlinearities in the network. We elect to use the binary hinge loss

$$\ell(\hat{y}, y) := \sigma\big(1 - y\hat{y}\big) \tag{1}$$

for binary classification, where $\hat{y}$ denote the scalar output of the network and $y \in \{-1, 1\}$ denotes the target. Similarly, for multiclass classification we use the multiclass hinge loss,

$$\ell(\hat{\mathbf{y}}, r_0) = \textstyle\sum_{r \neq r_0} \sigma\big(1 + \hat{y}_r - \hat{y}_{r_0}\big) \tag{2}$$

where $\hat{\mathbf{y}} = (\hat{y}_1, \ldots, \hat{y}_R) \in \mathbb{R}^R$ denotes the vectorial output of the network and $r_0 \in \{1, \ldots, R\}$ denotes the target class.

To see the type of structure that emerges in these networks, let Ω denote the space of network parameters and let $\mathcal{L}(\boldsymbol{\omega})$ denote the loss. Due to the choices (1,2) of network criteria, all nonlinearities involved in $\mathcal{L}(\boldsymbol{\omega})$ are piecewise linear.

[*]Equal contribution [1]Department of Mathematics, Loyola Marymount University, Los Angeles, CA 90045, USA [2]Department of Mathematics and Statistics, California State University, Long Beach, Long Beach, CA 90840, USA. Correspondence to: Thomas Laurent <tlaurent@lmu.edu>, James H. von Brecht <james.vonbrecht@csulb.edu>.

Proceedings of the 35^{th} International Conference on Machine Learning, Stockholm, Sweden, PMLR 80, 2018.

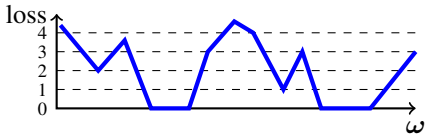


Figure 2. For a certain class of networks, flat local minima are always optimal whereas sharp ones are always sub-optimal.

These nonlinearities encode a partition of parameter space $\Omega = \Omega_1 \cup \cdots \cup \Omega_M \cup \mathcal{N}$ into a finite number of open *cells* Ω_u and a closed set $\mathcal{N}$ of cell boundaries (c.f. figure 1). A cell Ω_u corresponds to a given activation pattern of the nonlinearities, and so $\mathcal{L}(\boldsymbol{\omega})$ is smooth in the interior of cells and (potentially) non-differentiable on cell boundaries. This decomposition provides a description of the smooth (i.e. $\Omega \backslash \mathcal{N}$) and non-smooth (i.e. $\mathcal{N}$) parts of parameter space.

We begin by showing that the loss restricted to a cell Ω_u is a multilinear form. As multilinear forms are harmonic functions, an appeal to the strong maximum principle shows that non-trivial optima of the loss must happen on cell boundaries (i.e. the non-differentiable region $\mathcal{N}$ of the parameter space). In other words, ReLU networks with hinge loss criteria **do not have differentiable local minima**, except for those trivial ones that occur in regions of parameter space where the loss surface is perfectly flat. Figure 1b) provides a visual example of such a loss.

As a consequence the loss function has only two types of local minima. They are

- **Type I (Flat)**: Local minima that occur in a flat (i.e. constant loss) cell or on the boundary of a flat cell.
- **Type II (Sharp)**: Local minima on $\mathcal{N}$ that are not on the boundary of any flat cell.

We then investigate type I and type II local minima in more detail. The investigation reveals a clean dichotomy. First and foremost,

Main Result 1. $\mathcal{L}(\boldsymbol{\omega}) > 0$ *at any type II local minimum.*

Importantly, if zero loss minimizers exist (which happens for most modern deep networks) then sharp local minima are always sub-optimal. This result applies to a quite general class of deep neural networks with fully connected or convolutional layers equipped with either ReLU or leaky ReLU nonlinearities. To obtain a converse we restrict our attention to fully connected networks with leaky ReLU nonlinearities. Under mild assumptions on the data we have

Main Result 2. $\mathcal{L}(\boldsymbol{\omega}) = 0$ *at any type I local minimum, while* $\mathcal{L}(\boldsymbol{\omega}) > 0$ *at any type II local minimum.*

Thus flat local minima are always optimal whereas sharp minima are always sub-optimal in the case where zero loss minimizers exist. Conversely, if zero loss minimizers do not exist then all local minima are sharp. See figure 2 for an illustration of such a loss surface.

All in all these results paint a striking picture. Networks with ReLU or leaky ReLU nonlinearities and hinge loss criteria have only two types of local minima. Sharp minima always have non-zero loss; they are undesirable. Conversely, flat minima are always optimal for certain classes of networks. In this case the structure of the loss (flat v.s. sharp) provides a perfect characterization of their quality (optimal v.s. sub-optimal).

This analysis also shows that local minima generically occur in the non-smooth region of parameter space. Analyzing them requires an invocation of machinery from non-smooth, non-convex analysis. We show how to apply these techniques to study non-smooth networks in the context of binary classification. We consider three specific scenarios to illustrate how nonlinearity and data complexity affect the loss surface of multilinear networks —

- **Scenario 1:** A deep linear network with arbitrary data.
- **Scenario 2:** A network with one hidden layer, leaky ReLUs and linearly separable data.
- **Scenario 3:** A network with one hidden layer, ReLUs and linearly separable data.

The nonlinearities $\sigma_\alpha(x)$ vary from the linear regime ($\alpha = 1$) to the leaky regime ($0 < \alpha < 1$) and finally to the ReLU regime ($\alpha = 0$) as we pass from the first to the third scenario. We show that no sub-optimal local minimizers exist in the first two scenarios. When passing to the case of paramount interest, i.e. the third scenario, a bifurcation occurs. Degeneracy in the nonlinearities (i.e. $\alpha = 0$) induces sub-optimal local minima in the loss surface. We also provide an explicit description of all such sub-optimal local optima. They correspond to the occurence of *dead data points*, i.e. when some data points do not activate any of the neurons of the hidden layer and are therefore ignored by the network. Our results for the second and third scenarios provide a mathematically precise formulation of a commonplace intuitive picture. A ReLU can completely "turn off," and sub-optimal minima correspond precisely to situations in which a data point turns off all ReLUs in the hidden layer. As leaky ReLUs have no completely "off" state, such networks therefore have no sub-optimal minima.

Finally, in section 4 we conclude by investigating the extent to which these phenomena do, or do not, persist when passing to the multiclass context. The loss surface of a multilinear network with the multiclass hinge loss (2) is fundamentally different than that of a binary classification problem. In particular, the picture that emerges from our two-class results does not extend to the multiclass hinge loss. Nevertheless, we show how to obtain a similar picture of critical points by modifying the training strategy applied to multiclass problems.

Many recent works theoretically investigate the loss surface of ReLU networks. The closest to ours is (Safran & Shamir, 2016), which uses ReLU nonlinearities to partition the parameter space into basins that, while similar in spirit, differ from our notion of cells. Works such as (Keskar et al., 2016; Chaudhari et al., 2017) have empirically investigated the notion of "width" of a local minimizer. Conjecturally, a "wide" local minimum should generalize better than a "narrow" one and might be more likely to attract the solution generated by a stochastic gradient descent algorithm. Our flat and sharp local minima are reminiscent of these notions. Finally, some prior works have proved variants of our results in smooth situations. For instance, (Brutzkus et al., 2017) derives results about the smooth local minima occurring in scenarios 2 and 3, but they do not investigate non-differentiable local minima. Additionally, (Kawaguchi, 2016) considers our first scenario with a mean squared error loss instead of the hinge loss, while (Frasconi et al., 1997) considers our second scenario with a smooth version of the hinge loss and with sigmoid nonlinearities. Our non-smooth analogues of these results require fundamentally different techniques. We prove all lemmas, theorems and corollaries in the appendix.

2. Global Structure of the Loss

We begin by describing the global structure of ReLU networks with hinge loss that arises due to their piecewise multilinear form. Let us start by rewriting (2) as

$$\begin{aligned}\ell(\hat{\mathbf{y}},\mathbf{y}) &= -1+\sum_{r=1}^{R}\sigma\big(1+\hat{y}_r-\langle\mathbf{y},\hat{\mathbf{y}}\rangle\big)\\ &= -1+\Big\langle\ \mathbf{1}\ ,\ \sigma\Big(\ (\mathrm{Id}-\mathbf{1}\otimes\mathbf{y})\hat{\mathbf{y}}+\mathbf{1}\ \Big)\Big\rangle \quad (3)\end{aligned}$$

where we now view the target $\mathbf{y}\in\{0,1\}^R$ as a one-hot vector that encodes for the desired class. The term $\mathbf{1}\otimes\mathbf{y}$ denotes the outer product between the constant vector $\mathbf{1}=(1,\ldots,1)^T$ and the target, while $\langle\mathbf{y},\hat{\mathbf{y}}\rangle$ refers to the usual Euclidean inner product. We consider a collection $(\mathbf{x}^{(i)},\mathbf{y}^{(i)})$ of N labeled data points fed through a neural network with L hidden layers,

$$\begin{aligned}\mathbf{x}^{(i,\ell)} &= \sigma_\alpha(W^{(\ell)}\mathbf{x}^{(i,\ell-1)}+\mathbf{b}^{(\ell)}) \qquad \text{for} \quad \ell\in[L]\\ \hat{\mathbf{y}}^{(i)} &= V\mathbf{x}^{(i,L)}+\mathbf{c}, \qquad (4)\end{aligned}$$

so that for $\ell\in[L]:=\{1,\ldots,L\}$ each $\mathbf{x}^{(i,\ell)}$ refers to feature vector of the i^{th} data point at the ℓ^{th} layer (with the convention that $\mathbf{x}^{(i,0)}=\mathbf{x}^{(i)}$) and $\hat{\mathbf{y}}^{(i)}$ refers to the output of the network for the i^{th} datum. By (3) we obtain

$$\mathcal{L}(\boldsymbol{\omega})=-1+\sum_i\mu^{(i)}\big\langle\mathbf{1},\sigma\big((\mathrm{Id}-\mathbf{1}\otimes\mathbf{y}^{(i)})\hat{\mathbf{y}}^{(i)}+\mathbf{1}\big)\big\rangle \quad (5)$$

for the loss $\mathcal{L}(\boldsymbol{\omega})$. The positive weights $\mu^{(i)}>0$ sum to one, say $\mu^{(i)}=1/N$ in the simplest case, but we allow for other choices to handle those situations, such as an unbalanced training set, in which non-homogeneous weights could be beneficial. The matrices $W^{(\ell)}$ and vector $\mathbf{b}^{(\ell)}$ appearing in (4) define the affine transformation at layer ℓ of the network, and V and $\mathbf{c}$ in (4) denote the weights and bias of the output layer. We allow for fully-connected as well as structured models, such as convolutional networks, by imposing the assumption that each $W^{(\ell)}$ is a matrix-valued function that depends *linearly* on some set of parameters $\omega^{(\ell)}$ —

$$W^{(\ell)}\big(c\omega^{(\ell)}+d\hat{\omega}^{(\ell)}\big)=cW^{(\ell)}\big(\omega^{(\ell)}\big)+dW^{(\ell)}\big(\hat{\omega}^{(\ell)}\big);$$

thus the collection

$$\boldsymbol{\omega}=(\omega^{(1)},\ldots,\omega^{(L)},V,\mathbf{b}^{(1)},\ldots,\mathbf{b}^{(L)},\mathbf{c})\in\Omega$$

represents the parameters of the network and Ω denotes parameter space. As the slope α of the nonlinearity decreases from $\alpha=1$ to $\alpha=0$ the network transitions from a deep linear architecture to a standard ReLU network. Finally, we let d_ℓ denote the dimension of the features at layer ℓ of the network, with the convention that $d_0=d$ (dimension of the input data) and $d_{L+1}=R$ (number of classes). We use $D=d_1+\ldots+d_{L+1}$ for the total number of neurons.

2.1. Partitioning Ω into Cells

The nonlinearities $\sigma_\alpha(x)$ and $\sigma(x)$ account for the only sources of nondifferentiabilty in the loss of a ReLU network. To track these potential sources of nondifferentiability, for a given a data point $\mathbf{x}^{(i)}$ we define the functions

$$\begin{aligned}\mathbf{s}^{(i,\ell)}(\boldsymbol{\omega}) &:= \mathrm{sign}(W^{(\ell)}\mathbf{x}^{(i,\ell-1)}+\mathbf{b}^{\ell}) \qquad \text{for} \quad \ell\in[L]\\ \mathbf{s}^{(i,L+1)}(\boldsymbol{\omega}) &:= \mathrm{sign}\Big(\ (\mathrm{Id}-\mathbf{1}\otimes\mathbf{y}^{(i)})\ \hat{\mathbf{y}}^{(i)}+\mathbf{1}\ \Big), \qquad (6)\end{aligned}$$

where $\mathrm{sign}(x)$ stands for the signum function that vanishes at zero. The function $\mathbf{s}^{(i,\ell)}$ describes how data point $\mathbf{x}^{(i)}$ activates the d_ℓ neurons at the ℓ^{th} layer, while $\mathbf{s}^{(i,L+1)}(\boldsymbol{\omega})$ describes the corresponding "activation" of the loss. These activations take one of three possible states, the fully active state (encoded by a one), the fully inactive state (encoded by a minus one), or an in-between state (encoded by a zero). We then collect all of these functions into a single *signature function*

$$\begin{aligned}\mathcal{S}(\boldsymbol{\omega})=\Big(&\mathbf{s}^{(1,1)}(\boldsymbol{\omega}),\ldots,\mathbf{s}^{(1,L+1)}(\boldsymbol{\omega});\ldots\ldots;\\ &\mathbf{s}^{(N,1)}(\boldsymbol{\omega}),\ldots,\mathbf{s}^{(N,L+1)}(\boldsymbol{\omega})\Big)\end{aligned}$$

to obtain a function $\mathcal{S}:\Omega\mapsto\{-1,\,0,\,1\}^{ND}$ since there are a total of D neurons and N data points. If $\mathcal{S}(\boldsymbol{\omega})$ belongs to the subset $\{-1,1\}^{ND}$ of $\{-1,\,0,\,1\}^{ND}$ then none of the ND entries of $\mathcal{S}(\boldsymbol{\omega})$ vanish, and as a consequence, all of the nonlinearities are differentiable near $\boldsymbol{\omega}$; the loss $\mathcal{L}$ is smooth near such points. With this in mind, for a given

$u \in \{-1,1\}^{ND}$ we define the cell Ω_u as the (possibly empty) set

$$\Omega_u := \mathcal{S}^{-1}(u) := \{\boldsymbol{\omega} \in \Omega : \mathcal{S}(\boldsymbol{\omega}) = u\}$$

of parameter space. By choice $\mathcal{L}$ is smooth on each non-empty cell Ω_u, and so the cells Ω_u provide us with a partition of the parameter space

$$\Omega = \left(\bigcup_{u \in \{-1,1\}^{ND}} \Omega_u \right) \bigcup \mathcal{N}$$

into smooth and potentially non-smooth regions. The set $\mathcal{N}$ contains those $\boldsymbol{\omega}$ for which at least one of the ND entries of $\mathcal{S}(\boldsymbol{\omega})$ takes the value 0, which implies that at least one of the nonlinearities is non-differentiable at such a point. Thus $\mathcal{N}$ consists of points at which the loss is potentially non-differentiable. The following lemma collects the various properties of the cells Ω_u and of $\mathcal{N}$ that we will need.

Lemma 1. *For each $u \in \{-1,1\}^{ND}$ the cell Ω_u is an open set. If $u \neq u'$ then Ω_u and $\Omega_{u'}$ are disjoint. The set $\mathcal{N}$ is closed and has Lebesgue measure 0.*

2.2. Flat and Sharp Minima

Recall that a function $\phi : \mathbb{R}^{d_1} \times \ldots \times \mathbb{R}^{d_n} \to \mathbb{R}$ is a multilinear form if it is linear with respect to each of its inputs when the other inputs are fixed. That is,

$$\phi(\mathbf{v}_1, \ldots, c\mathbf{v}_k + d\mathbf{w}_k, \ldots, \mathbf{v}_n) = c\phi(\mathbf{v}_1, \ldots, \mathbf{v}_k, \ldots, \mathbf{v}_n) + d\phi(\mathbf{v}_1, \ldots, \mathbf{w}_k, \ldots, \mathbf{v}_n).$$

Our first theorem forms the basis for our analytical results. It states that, up to a constant, the loss restricted to a fixed cell Ω_u is a sum of multilinear forms.

Theorem 1 (Multilinear Structure of the Loss)**.** *For each cell Ω_u there exist multilinear forms $\phi_0^u, \ldots, \phi_{L+1}^u$ and a constant ϕ_{L+2}^u such that*

$$\begin{aligned}\mathcal{L}|_{\Omega_u}(\omega^{(1)}, \ldots, \omega^{(L)}, V, \mathbf{b}^{(1)}, \ldots, \mathbf{b}^{(L)}, \mathbf{c}) = \\ \phi_0^u(\omega^{(1)}, \omega^{(2)}, \omega^{(3)}, \omega^{(4)} \ldots, \omega^{(L)}, V) \\ +\phi_1^u(\mathbf{b}^{(1)}, \omega^{(2)}, \omega^{(3)}, \omega^{(4)} \ldots, \omega^{(L)}, V) \\ +\phi_2^u(\mathbf{b}^{(2)}, \omega^{(3)}, \omega^{(4)} \ldots, \omega^{(L)}, V) \\ \vdots \\ +\phi_{L-1}^u(\mathbf{b}^{(L-1)}, \omega^{(L)}, V) \\ +\phi_L^u(\mathbf{b}^{(L)}, V) \\ +\phi_{L+1}^u(\mathbf{c}) \\ +\phi_{L+2}^u.\end{aligned}$$

The proof relies on the fact that the signature function $\mathcal{S}(\boldsymbol{\omega})$ is constant inside a fixed cell Ω_u, and so the network reduces to a succession of affine transformations. These combine to produce a sum of multilinear forms. Appealing to properties of multilinear forms then gives two important corollaries. Multilinear forms are harmonic functions. Using the strong maximum principle for harmonic functions[1] we show that $\mathcal{L}$ does not have differentiable optima, except for the trivial flat ones.

Corollary 1 (No Differentiable Extrema)**.** *Local minima and maxima of the loss* (5) *occur only on the boundary set $\mathcal{N}$ or on those cells Ω_u where the loss is constant. In the latter case, $\mathcal{L}|_{\Omega_u}(\boldsymbol{\omega}) = \phi_{L+2}^u$.*

Our second corollary reveals the saddle-like structure of the loss.

Corollary 2 (Saddle-like Structure of the Loss)**.** *If $\boldsymbol{\omega} \in \Omega \setminus \mathcal{N}$ and the Hessian matrix $D^2\mathcal{L}(\boldsymbol{\omega})$ does not vanish, then it must have at least one strictly positive and one strictly negative eigenvalue.*

These corollaries have implications for various optimization algorithms. At a local minimum $D^2\mathcal{L}$ either vanishes (flat local minima) or does not exist (sharp local minima). Therefore local minima do not carry any second order information. Moreover, away from minima the Hessian is never positive definite and is typically indefinite. Thus an optimization algorithm using second-order (i.e. Hessian) information must pay close attention to both the indefinite and non-differentiable nature of the loss.

To investigate type I/II minima in greater depth we must there exploit the multilinear structure of $\mathcal{L}$ itself. Our first result along these lines concerns type II local minima.

Theorem 2. *If $\boldsymbol{\omega}$ is a type II local minimum then $\mathcal{L}(\boldsymbol{\omega}) > 0$.*

Modern networks of the form (5) typically have zero loss global minimizers. For any such network type II (i.e. sharp) local minimizers are therefore always sub-optimal. A converse of theorem 2 holds for a restricted class of networks. That is, type I (i.e. flat) local minimizers are always optimal. To make this precise we need a mild assumption on the data.

Definition 1. *Fix $\alpha > 0$ and a collection of weighted data points $(\mu^{(i)}, \mathbf{x}^{(i)}, \mathbf{y}^{(i)})$. The weighted data are* **rare** *if there exist N coeffecients $\lambda^{(i)} \in \{1, \alpha, \ldots, \alpha^L\}$ and a non-zero collection of NR scalars $\varepsilon^{(i,r)} \in \{0,1\}$ so that the system*

$$\begin{aligned}\varepsilon^{(i)} &= \sum_{r:\mathbf{y}_r^{(i)}=0} \varepsilon^{(i,r)} \\ \sum_{i:\mathbf{y}_r^{(i)}=1} \lambda^{(i)}\mu^{(i)}\varepsilon^{(i)}\mathbf{x}^{(i)} &= \sum_{i:\mathbf{y}_r^{(i)}=0} \lambda^{(i)}\mu^{(i)}\varepsilon^{(i,r)}\mathbf{x}^{(i)}\end{aligned}$$

[1] The strong maximum principle states that a non-constant harmonic function cannot attain a local minimum or a local maximum at an interior point of an open, connected set.

$$\sum_{i:\mathbf{y}_r^{(i)}=1} \lambda^{(i)}\mu^{(i)}\varepsilon^{(i)} = \sum_{i:\mathbf{y}_r^{(i)}=0} \lambda^{(i)}\mu^{(i)}\varepsilon^{(i,r)} \tag{7}$$

holds $\forall r \in [R]$. *The data are* **generic** *if they are not rare.*

As the possible choices of $\lambda^{(i)}, \varepsilon^{(i,r)}$ take on at most a finite set of values, rare data points $(\mu^{(i)}, \mathbf{x}^{(i)}, \mathbf{y}^{(i)})$ must satisfy one of a given finite set of linear combinations. Thus (7) represents the exceptional case, and most data are generic. For example, if the $\mathbf{x}^{(i)} \sim X^{(i)}$ come from indepenendent samples of atomless random variables $X^{(i)}$ they are generic with probability one. Similarly, a small perturbation in the weights $\mu^{(i)}$ will usually transform data from rare to generic.

Theorem 3. *Consider the loss* (5) *for a fully connected network. Assume that* $\alpha > 0$ *and that the data points* $(\mathbf{x}^{(i)}, \mathbf{y}^{(i)})$ *are generic. Then* $\mathcal{L}(\boldsymbol{\omega}) = 0$ *at any type I local minimum.*

For most data we may pair this result with its counterpart for fully connected networks and obtain a clear picture. Desirable (zero loss) minima are always flat, while undesireable (positive loss) minima are always sharp. Analyzing sub-optimal minima therefore requires handling the non-smooth case, and we now turn to this task.

3. Critical Point Analysis

In this section we use machinery from non-smooth analysis (see chapter 6 of (Borwein & Lewis, 2010) for a good reference) to study critical points of the loss surface of such piecewise multilinear networks. We consider three scenarios by traveling from the deep linear case ($\alpha = 1$) and passing through the leaky ReLU case ($0 < \alpha < 1$) before arriving at the most common case ($\alpha = 0$) of ReLU networks. We intend this journey to highlight how the loss surface changes as the level of nonlinearity increases. A deep linear network has a trivial loss surface, in that local and global minima coincide (see theorem 100 in the appendix for a precise statement and its proof). If we impose further assumptions, namely linearly separable data in a one-hidden layer network, this benign structure persists into the leaky ReLU regime. When we arrive at $\alpha = 0$ a bifurcation occurs, and sub-optimal local minima suddenly appear in classical ReLU networks.

To begin, we recall that for a Lipschitz but non-differentiable function $f(\boldsymbol{\omega})$ the *Clarke subdifferential* $\partial_0 f(\boldsymbol{\omega})$ of f at a point $\boldsymbol{\omega} \in \Omega$ provides a generalization of both the gradient $\nabla f(\boldsymbol{\omega})$ and the usual subdifferential $\partial f(\boldsymbol{\omega})$ of a convex function. The Clarke subdifferential is defined as follow (c.f. page 133 of (Borwein & Lewis, 2010)):

Definition 2 (Clarke Subdifferential and Critical Points)**.** *Assume that a function* $f : \Omega \mapsto \mathbb{R}$ *is locally Lipschitz around* $\boldsymbol{\omega} \in \Omega$*, and differentiable on* $\Omega \setminus \mathcal{M}$ *where* $\mathcal{M}$ *is a set of Lebesgue measure zero. Then the convex hull*

$$\partial_0 f(\boldsymbol{\omega}) := \text{c.h.}\left\{\lim_k \nabla f(\boldsymbol{\omega}_k) : \boldsymbol{\omega}_k \to \boldsymbol{\omega}, \boldsymbol{\omega}_k \notin \mathcal{M}\right\}$$

is the ***Clarke Subdifferential*** *of* f *at* $\boldsymbol{\omega}$*. In addition, if*

$$\mathbf{0} \in \partial_0 f(\boldsymbol{\omega}), \tag{8}$$

then $\boldsymbol{\omega}$ *is a* ***critical point*** *of* f *in the Clarke sense.*

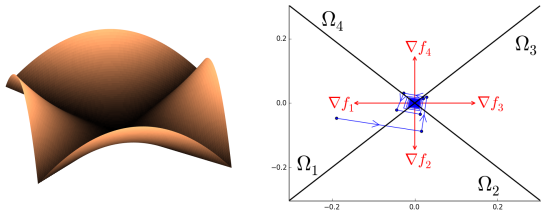


Figure 3. Illustration of the Clarke Subdifferential

The definition of critical point is a consistent one, in that (8) must hold whenever $\boldsymbol{\omega}$ is a local minimum (c.f. page 125 of (Borwein & Lewis, 2010)). Thus the set of all critical points contains the set of all local minima. Figure 3 provides an illustration of the Clarke Subdifferential. It depicts a function $f : \mathbb{R}^2 \mapsto \mathbb{R}$ with global minimum at the origin, which therefore defines a critical point in the Clarke sense. While the gradient of $f(\mathbf{x})$ itself does not exist at $\mathbf{0}$, its restrictions $f_k := f|_{\Omega_k}$ to the four cells Ω_k neighboring $\mathbf{0}$ have well-defined gradients $\nabla f_k(\mathbf{0})$ (shown in red) at the critical point. By definition the Clarke subdifferential $\partial_0 f(\mathbf{0})$ of f at $\mathbf{0}$ consists of all convex combinations

$$\theta_1 \nabla f_1(\mathbf{0}) + \theta_2 \nabla f_2(\mathbf{0}) + \theta_3 \nabla f_3(\mathbf{0}) + \theta_4 \nabla f_4(\mathbf{0})$$

of these gradients; that some such combination vanishes (say, $\frac{1}{2}\nabla f_1(\mathbf{0}) + \frac{1}{2}\nabla f_3(\mathbf{0}) = \mathbf{0}$) means that $\mathbf{0}$ satisfies the definition of a critical point. Moreover, an element of the subdifferential $\partial_0 f$ naturally arises from gradient descent. A gradient-based optimization path $\mathbf{x}^{(j+1)} = \mathbf{x}^{(j)} - dt^{(j)}\nabla f(\mathbf{x}^{(j)})$ (shown in blue) asymptotically builds, by successive accumulation at each step, a convex combination of the ∇f_k whose corresponding weights θ_k represent the fraction of time the optimization spends in each cell.

We may now show how to apply these tools in the study of ReLU Networks. We first analyze the leaky regime ($0 < \alpha < 1$) and then analyze the ordinary ReLU case ($\alpha = 0$).

Leaky Networks ($0 < \alpha < 1$)**:** Take $0 < \alpha < 1$ and consider the corresponding loss $\mathcal{L}(W, \mathbf{v}, \mathbf{b}, c) =$

$$\sum \mu^{(i)}\, \sigma\left[\, 1 - y^{(i)}\left\{\mathbf{v}^T \sigma_\alpha(W\mathbf{x}^{(i)} + \mathbf{b}) + c\right\}\right] \tag{9}$$

associated to a fully connected network with one hidden layer. We shall also assume the data $\{\mathbf{x}^{(i)}\}$ are linearly separable. In this setting we have

Theorem 4 (Leaky ReLU Networks)**.** *Consider the loss* (9) *with* $\alpha > 0$ *and data* $\mathbf{x}^{(i)}, i \in [N]$ *that are linearly separable. Assume that* $\boldsymbol{\omega} = (W, \mathbf{v}, \mathbf{b}, c)$ *is any critical point of the loss in the Clarke sense. Then either* $\mathbf{v} = \mathbf{0}$ *or* $\boldsymbol{\omega}$ *is a global minimum.*

The loss in this scenario has two type of critical points. Critical points with $\mathbf{v} = \mathbf{0}$ correspond to a trivial network in which all data points are mapped to a constant; all other critical points are global minima. If we further assume equally weighted classes

$$\sum_{i:y^{(i)}=1} \mu^{(i)} = \sum_{i:y^{(i)}=-1} \mu^{(i)}$$

then all local minima are global minima —

Theorem 5 (Leaky ReLU Networks with Equal Weight)**.** *Consider the loss* (9) *with* $\alpha > 0$ *and data* $\mathbf{x}^{(i)}, i \in [N]$ *that are linearly separable. Assume that the* $\mu^{(i)}$ *weight both classes equally. Then every local minimum of* $\mathcal{L}(\boldsymbol{\omega})$ *is a global minimum.*

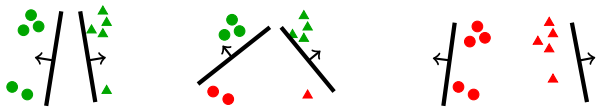
In other words, the loss surface is trivial when $0 < \alpha \leq 1$.

ReLU Networks ($\alpha = 0$)**:** This is the case of paramount interest. When passing from $\alpha > 0$ to $\alpha = 0$ a structural bifurcation occurs in the loss surface — ReLU nonlinearities generate non-optimal local minima even in a one hidden layer network with separable data. Our analysis provides an explicit description of all the critical points of such loss surfaces, which allows us to precisely understand the way in which sub-optimality occurs.

In order to describe this structure let us briefly assume that we have a simplified model with two hidden neurons, no output bias and uniform weights. If $\mathbf{w}_k$ denotes the k^{th} row of W then we have the loss

$$\mathcal{L}(W, \mathbf{v}, \mathbf{b}) = \frac{1}{N} \sum \sigma(1 - y^{(i)} \hat{y}^{(i)}), \quad \text{where}$$
$$\hat{y}^{(i)} = \sum_{k=1}^{2} v_k \sigma\Big(\langle \mathbf{w}_k, \mathbf{x}^{(i)} \rangle + b_k\Big) \tag{10}$$

for such a network. Each hidden neuron has an associated hyperplane $\langle \mathbf{w}_k, \cdot \rangle + b_k$ as well as a scalar weight v_k used to form the output. Figure 4 shows three different local minima of such a network. The first panel, figure 4(a), shows a global minimum where all the data points have zero loss. Figure 4(b) shows a sub-optimal local minimum. All unsolved data points, namely those that contribute a non-zero value to the loss, lie on the "blind side" of the two hyperplanes. For each of these data points the corresponding network output $\hat{y}^{(i)}$ vanishes and so the loss is $\sigma(\ 1 - y^{(i)} \hat{y}^{(i)}) = 1$ for these unsolved points. Small perturbations of the hyperplanes or of the values of the v_k do not change the fact that these data points lie on the blind side of the two hyperplanes. Their loss will not decrease under small perturbations, and so the configuration is, in fact, a local minimum. The same reasoning shows that the configuration in figure 4(c), in which no data point is classified correctly, is also a local minimum.



(a) $\mathcal{L} = 0$ (b) $\mathcal{L} = 3/10$ (c) $\mathcal{L} = 10/10$

Figure 4. Three different local minima of the loss $\mathcal{L}(\boldsymbol{\omega})$ for a network with two hidden neurons and standard ReLU nonlinearities. Points belonging to class +1 (resp. -1) are denoted by circles (resp. triangles). Data points for which the loss is zero (solved points) are colored in green, while data points with non-zero loss (unsolved points) are in red. The unsolved data points always lie on the blind side of both hyperplanes.

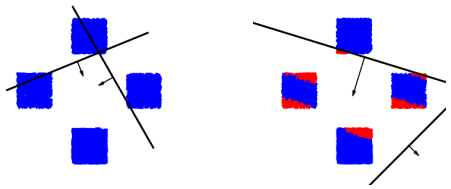
Despite the presence of sub-optimal local minimizers, the local minima depicted in figure 4 are somehow trivial cases. They simply come from the fact that, due to inactive ReLUs, some data points are completely ignored by the network, and this fact cannot be changed by small perturbations. The next theorem essentially shows that these are the only possible sub-optimal local minima that occur. Moreover, the result holds for the case (9) of interest and not just the simplified model.

Theorem 6 (ReLU networks)**.** *Consider the loss* (9) *with* $\alpha = 0$ *and data* $\mathbf{x}^{(i)}, i \in [N]$ *that are linearly separable. Assume that* $\boldsymbol{\omega} = (W, \mathbf{v}, \mathbf{b}, c)$ *is a critical point in the Clarke sense, and that* $\mathbf{x}^{(i)}$ *is any data point that contributes a nonzero value to the loss. Then for each hidden neuron* $k \in [K]$ *either*

$$\textit{(i)}\ \langle \mathbf{w}_k, \mathbf{x}^{(i)} \rangle + b_k \leq 0, \quad \textit{or} \quad \textit{(ii)}\ v_k = 0.$$

If $v_k = 0$ then the k^{th} hidden neuron is unused when forming network predictions. In this case we may say the k^{th} hyperplane is *inactive*, while if $v_k \neq 0$ the corresponding hyperplane is *active*. Theorem 6 therefore states that **if a data point $\mathbf{x}^{(i)}$ is unsolved it must lie on the blind side of every active hyperplane.** So all critical points, including local minima, obey the property sketched in figure 4.

When taken together, theorems 5 and 6 provide rigorous mathematical ground for the common view that dead or inactive neurons can cause difficulties in optimizing neural networks, and that using leaky ReLU networks can overcome these difficulties. The former have sub-optimal local minimizers exactly when a data point does not activate any of the ReLUs in the hidden layer, but this situation never occurs with leaky ReLUs and so neither do sub-optima minima.



(a) $\mathcal{L}(\boldsymbol{\omega}) = 0$ (b) $\mathcal{L}(\boldsymbol{\omega}) > 0$

Figure 5. Four-way classification with multiclass hinge loss (2). At left — a global minimizer. At right — a sub-optimal local minimizer where the analogue of theorem 6 fails.

4. Exact Penalties and Multi-Class Structure

These results give a clear illustration of how nonlinearity and data complexity combine to produce local minimizers in the loss surface for binary classification tasks. While we might try to analyze multi-class tasks by following down the same path, such an effort would unfotunately bring us to a quite different destination. Specifically, the conclusion of theorem 6 fails for multi-class case; in the presence of three or more classes a critical point may exhibit active yet unsolved data points (c.f. figure 5). This phenomenon is inherent to multi-class tasks in a certain sense, for if we use the same features $\mathbf{x}^{(i,\ell)}$ (c.f. (4)) in a multi-layer ReLU network but apply a different network criterion $\bar{\ell}(\mathbf{y}, \hat{\mathbf{y}})$ then the phenomenon persists. For example, using the one-versus-all criterion

$$\bar{\ell}(\hat{\mathbf{y}}, \mathbf{y}) := \textstyle\sum_r \mu^{(i,r)} \sigma\left(1 + \hat{y}_r^{(i)}(-1)^{y_r^{(i)}}\right), \qquad (11)$$

in place of the hinge loss (2) still gives rise to a network with non-trivial critical points (similar to figure 5) despite its more "binary" structure. In this way, the emergence of non-trivial critical points reflects the nature of multi-class tasks rather than some pathology of the hinge-loss network criterion itself.

To arrive at the same destination our analysis must therefore take a more circumlocuitous route. As these counter-examples suggest, if the loss $\mathcal{L}(\boldsymbol{\omega})$ has non-trivial critical points then we must avoid non-trivial critical points by modifing the *training strategy* instead. We shall employ the one-versus-all criterion (11) for this task, as this choice will allows us to directly leverage our binary analyses.

Let us begin this process by recalling that

$$\mathbf{x}^{(i,L)}\big(\omega^{(1)}, \dots, \omega^{(L)}, \mathbf{b}^{(1)}, \dots, \mathbf{b}^{(L)}\big)$$

and $\hat{\mathbf{y}}^{(i)} = V\mathbf{x}^{(i,L)} + \mathbf{c}$ denote the features and predictions of the network with L hidden layers, respectively. The sub-collection of parameters

$$\breve{\boldsymbol{\omega}} := \big(\omega^{(1)}, \dots, \omega^{(L)}, \mathbf{b}^{(1)}, \dots, \mathbf{b}^{(L)}\big)$$

therefore determine a set of features $\mathbf{x}^{(i,L)}$ for the network while the parameters $V, \mathbf{c}$ determine a set of one-versus-all classifiers utilizing these features. We may write the loss for the r^{th} class as

$$\mathcal{L}^{(r)}(\breve{\boldsymbol{\omega}}, \mathbf{v}_r, c_r) = \sum \mu^{(i,r)} \sigma\left(1 + \hat{y}_r^{(i)}(-1)^{y_r^{(i)}}\right) \qquad (12)$$

and then form the sum over classes

$$\bar{\mathcal{L}}(\boldsymbol{\omega}) := (\mathcal{L}^{(1)} + \cdots + \mathcal{L}^{(R)})(\boldsymbol{\omega})$$

to recover the total objective. We then seek to minimize $\bar{\mathcal{L}}$ by applying a soft-penalty approach. We introduce the R replicates

$$\breve{\boldsymbol{\omega}}^{(r)} = \big(\omega^{(1,r)}, \dots, \omega^{(L,r)}, \mathbf{b}^{(1,r)}, \dots, \mathbf{b}^{(L,r)}\big) \quad r \in [R]$$

of the hidden-layer parameters $\breve{\boldsymbol{\omega}}$ and include a soft ℓ^2-penalty $\mathcal{R}\big(\breve{\boldsymbol{\omega}}^{(1)}, \dots, \breve{\boldsymbol{\omega}}^{(R)}\big) :=$

$$\frac{R}{R-1} \sum_{\ell=1}^{L} \sum_{r=1}^{R} \|\omega^{(\ell,r)} - \bar{\omega}^{(\ell)}\|^2 + \|\mathbf{b}^{(\ell,r)} - \bar{\mathbf{b}}^{(\ell)}\|^2$$

to enforce that the replicated parameters $\omega^{(\ell,r)}, \mathbf{b}^{(\ell,r)}$ remain close to their corresponding means $(\bar{\omega}^{(\ell)}, \bar{\mathbf{b}}^{(\ell)})$ across classes. Our training strategy then proceeds to minimize the penalized loss $\mathcal{E}_\gamma\big(\boldsymbol{\omega}^{(1)}, \dots, \boldsymbol{\omega}^{(R)}\big) :=$

$$\textstyle\sum_r \mathcal{L}^{(r)}\big(\boldsymbol{\omega}^{(r)}\big) + \gamma \mathcal{R}\big(\breve{\boldsymbol{\omega}}^{(1)}, \dots, \breve{\boldsymbol{\omega}}^{(R)}\big) \qquad (13)$$

for $\gamma > 0$ some parameter controlling the strength of the penalty. Remarkably, utilizing this strategy yields

Theorem 7 (Exact Penalty and Recovery of Two-Class Structure)**.** *If $\gamma > 0$ then the following hold for* (13) *—*

*(i) The penalty is exact, that is, at **any** critical point $\big(\boldsymbol{\omega}^{(1)}, \dots, \boldsymbol{\omega}^{(R)}\big)$ of $\mathcal{E}_\gamma$ the equalities*

$$\omega^{(\ell,1)} = \cdots = \omega^{(\ell,R)} = \bar{\omega}^{(\ell)} := \frac{1}{R}\sum_{r=1}^{R} \omega^{(\ell,r)}$$

$$\mathbf{b}^{(\ell,1)} = \cdots = \mathbf{b}^{(\ell,R)} = \bar{\mathbf{b}}^{(\ell)} := \frac{1}{R}\sum_{r=1}^{R} \mathbf{b}^{(\ell,r)}$$

hold for all $\ell \in [L]$.

*(ii) At **any** critical point of $\mathcal{E}_\gamma$ the two-class critical point relations $\mathbf{0} \in \partial_0 \mathcal{L}^{(r)}(\breve{\boldsymbol{\omega}}, \mathbf{v}_r, c_r)$ hold for all $r \in [R]$.*

In other words, applying a soft-penalty approach to minimizing the original problem (12) actually yields an exact penalty method. By (i), at critical points we obtain a common set of features $\mathbf{x}^{(i,L)}$ for each of the R binary classification problems. Moreover, by (ii) these features simultaneously yield critical points

$$\mathbf{0} \in \partial_0 \mathcal{L}^{(r)}\big(\breve{\boldsymbol{\omega}}, \mathbf{v}_r, c_r\big) \qquad (14)$$

for *all* of these binary classification problems. The fact that (14) may fail for critical points of $\bar{\mathcal{L}}$ is responsible

for the presence of non-trivial critical points in the context of a network with one hidden layer. We may therefore interpret (ii) as saying that a training strategy that uses the penalty approach will avoid pathological critical points where $\mathbf{0} \in \partial_0 \bar{\mathcal{L}}(\boldsymbol{\omega})$ holds but (14) does not. In this way the penalty approach provides a path forward for studying multi-class problems. Regardless of the number L of hidden layers, it allows us to form an understanding of the family of critical points (14) by reducing to a study of critical points of binary classification problems. This allows us to extend the analyses of the previous section to the multi-class context.

We may now pursue an analysis of multi-class problems by traveling along the same path that we followed for binary classification. That is, a deep linear network ($\alpha = 1$) once again has a trivial loss surface (see corollaries 100 and 101 in the appendix for precise statements and proofs). By imposing the same further assumptions, namely linearly separable data in a one-hidden layer network, we may extend this benign structure into the leaky ReLU regime. Finally, when $\alpha = 0$ sub-optimal local minima appear; we may characterize them in a manner analogous to the binary case.

To be precise, recall the loss

$$\mathcal{L}(\boldsymbol{\omega}) = \sum_{r=1}^{R} \mathcal{L}^{(r)}(\boldsymbol{\omega}) \quad \text{for} \tag{15}$$
$$\mathcal{L}^{(r)}(\boldsymbol{\omega}) := \sum \mu^{(i,r)} \sigma\big(1 - y^{(i,r)}(\langle \mathbf{v}_r, \mathbf{x}^{(i,1)} \rangle + c_r)\big)$$

that results from the features $\mathbf{x}^{(i,1)} = \sigma_\alpha(W\mathbf{x}^{(i)} + \mathbf{b})$ of a ReLU network with one hidden layer. If the positive weights $\mu^{(i,r)} > 0$ satisfy

$$\sum_{y^{(i,r)}=1} \mu^{(i,r)} = \sum_{y^{(i,r)}=-1} \mu^{(i,r)} = \frac{1}{2}$$

then we say that the $\mu^{(i,r)}$ give equal weight to all classes. Appealing to the critical point relations (14) yields the following corollary. It gives the precise structure that emerges from the leaky regime $0 < \alpha < 1$ with separable data —

Corollary 3 (Multiclass with $0 < \alpha < 1$)**.** *Consider the loss* (15) *and its corresponding penalty* (13) *with* $\gamma > 0, 0 < \alpha < 1$ *and data* $\mathbf{x}^{(i)}, i \in [N]$ *that are linearly separable.*

(i) Assume that $\boldsymbol{\omega} = (\boldsymbol{\omega}^{(1)}, \ldots, \boldsymbol{\omega}^{(R)})$ *is a critical point of* $\mathcal{E}_\gamma$ *in the Clarke sense. If* $\mathbf{v}^{(r)} \neq \mathbf{0}$ *for all* $r \in [R]$ *then* $\boldsymbol{\omega}$ *is a global minimum of* $\mathcal{L}$ *and of* $\mathcal{E}_\gamma$.

(ii) Assume that the $\mu^{(i,r)}$ *give equal weight to all classes. If* $\boldsymbol{\omega} = (\boldsymbol{\omega}^{(1)}, \ldots, \boldsymbol{\omega}^{(R)})$ *is a local minimum of* $\mathcal{E}_\gamma$ *and* $\mathbf{v}_r = \mathbf{0}$ *for some* $r \in [R]$ *then* $\boldsymbol{\omega}$ *is a global minimum of* $\mathcal{L}$ *and of* $\mathcal{E}_\gamma$.

Finally, when arriving at the standard ReLU nonlinearity $\alpha = 0$ a bifurcation occurs. Sub-optimal local minimizers of $\mathcal{E}_\gamma$ can exist, but once again the manner in which these sub-optimal solutions appear is easy to describe. We let $\ell^{(i,r)}(\boldsymbol{\omega})$ denote the contribution of the i^{th} data point $\mathbf{x}^{(i)}$ to the loss $\mathcal{L}^{(r)}$ for the r^{th} class, so that $\mathcal{L}^{(r)}(\boldsymbol{\omega}) = \sum_i \mu^{(i,r)} \ell^{(i,r)}(\boldsymbol{\omega})$ gives the total loss. Appealing directly to the family of critical point relations $\mathbf{0} \in \partial_0 \mathcal{L}^{(r)}\big(\breve{\boldsymbol{\omega}}, \mathbf{v}_r, c_r\big)$ furnished by theorem 7 yields our final corollary in the multiclass setting.

Corollary 4 (Multiclass with $\alpha = 0$)**.** *Consider the loss* (15) *and its corresponding penalty* (13) *with* $\gamma > 0, \alpha = 0$ *and data* $\mathbf{x}^{(i)}, i \in [N]$ *that are linearly separable. Assume that* $\boldsymbol{\omega} = (\boldsymbol{\omega}^{(1)}, \ldots, \boldsymbol{\omega}^{(R)})$ *is any critical point of* $\mathcal{E}_\gamma$ *in the Clarke sense. Then* $\ell^{(i,r)} > 0$

$$\implies \quad (\mathbf{v}_r)_k \, \sigma(\langle \mathbf{w}_k, \mathbf{x}^{(i)} \rangle + b_k) = 0 \quad \textit{for all} \quad k \in [K].$$

5. Conclusion

We conclude by painting the overall picture that emerges from our analyses. The loss of a ReLU network is a multilinear form inside each cell. Multilinear forms are harmonic functions, and so maxima or minima simply cannot occur in the interior of a cell unless the loss is constant on the entire cell. This simple harmonic analysis reasoning leads to the following striking fact. ReLU networks **do not have differentiable minima**, except for trivial cases. This reasoning is valid for any convolutional or fully connected network, with plain or leaky ReLUs, and with binary or multiclass hinge loss. Dealing with non-differentiable minima is therefore not a technicality; it is the heart of the matter.

Given this dichotomy between trivial, differentiable minima on one hand and nontrivial, nondifferentiable minima on the other, it is natural to try and characterise these two classes of minima more precisely. We show that global minima with zero loss must be trivial, while minima with nonzero loss are necessarily nondifferentiable for many fully connected networks. In particular, if a network has no zero loss minimizers then all minima are nondifferentiable.

Finally, our analysis clearly shows that local minima of ReLU networks are generically nondifferentiable. They cannot be waved away as a technicality, so any study of the loss surface of such network must invoke nonsmooth analysis. We show how to properly use this machinery (e.g. Clark subdifferentials) to study ReLU networks. Our goal is twofold. First, we prove that a bifurcation occurs when passing from leaky ReLU to ReLU nonlinearities, as suboptimal minima suddenly appear in the latter case. Secondly, and perhaps more importantly, we show how to apply nonsmooth analysis in familiar settings so that future researchers can adapt and extend our techniques.

References

Borwein, J. and Lewis, A. S. *Convex analysis and nonlinear optimization: theory and examples. Second Edition*.

Springer Science & Business Media, 2010.

Brutzkus, A., Globerson, A., Malach, E., and Shalev-Shwartz, S. Sgd learns over-parameterized networks that provably generalize on linearly separable data. *arXiv preprint arXiv:1710.10174*, 2017.

Chaudhari, P., Choromanska, A., Soatto, S., and LeCun, Y. Entropy-sgd: Biasing gradient descent into wide valleys. In *International Conference on Learning Representations*, 2017.

Choromanska, A., Henaff, M., Mathieu, M., Arous, G. B., and LeCun, Y. The loss surfaces of multilayer networks. In *Artificial Intelligence and Statistics*, pp. 192–204, 2015.

Du, S. S., Lee, J. D., and Tian, Y. When is a convolutional filter easy to learn? *arXiv preprint arXiv:1709.06129*, 2017.

Frasconi, P., Gori, M., and Tesi, A. Successes and failures of backpropagation: A theoretical. *Progress in Neural Networks: Architecture*, 5:205, 1997.

Gori, M. and Tesi, A. On the problem of local minima in backpropagation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(1):76–86, 1992.

Kawaguchi, K. Deep learning without poor local minima. In *Advances in Neural Information Processing Systems*, pp. 586–594, 2016.

Keskar, N. S., Mudigere, D., Nocedal, J., Smelyanskiy, M., and Tang, P. T. P. On large-batch training for deep learning: Generalization gap and sharp minima. In *International Conference on Learning Representations*, 2016.

Li, Y. and Yuan, Y. Convergence analysis of two-layer neural networks with relu activation. In *Advances in Neural Information Processing Systems*, pp. 597–607, 2017.

Mei, S., Bai, Y., and Montanari, A. The landscape of empirical risk for non-convex losses. *arXiv preprint arXiv:1607.06534*, 2016.

Safran, I. and Shamir, O. On the quality of the initial basin in overspecified neural networks. In *International Conference on Machine Learning*, pp. 774–782, 2016.

Soltanolkotabi, M. Learning relus via gradient descent. In *Advances in Neural Information Processing Systems*, pp. 2004–2014, 2017.

Soudry, D. and Hoffer, E. Exponentially vanishing sub-optimal local minima in multilayer neural networks. *arXiv preprint arXiv:1702.05777*, 2017.

Tian, Y. An analytical formula of population gradient for two-layered relu network and its applications in convergence and critical point analysis. In *International Conference on Machine Learning*, 2017.

Zhong, K., Song, Z., Jain, P., Bartlett, P. L., and Dhillon, I. S. Recovery guarantees for one-hidden-layer neural networks. In *International Conference on Machine Learning*, 2017.

Zhou, P. and Feng, J. The landscape of deep learning algorithms. *arXiv preprint arXiv:1705.07038*, 2017.