One lecture on two-layer neural networks

Andrea Montanari*

August 17, 2018

Abstract

Notes for a lecture at the Cargèse Summer School 'Statistical Physics and Machine Learning Back Together,' August 21, 2018.

1 The basic problem

The basic problem of machine learning can be stated as follows. We are given data $\{(y_i, \boldsymbol{x}_i)\}_{1\le i\le n}$ which are independent and identically distributed (i.i.d.) from a common distribution $\mathbb{P}$. Here $\boldsymbol{x}_i \in \mathbb{R}^d$ is a feature vector (e.g. a descriptor of an image) and $y_i \in \mathbb{R}$ is a response variable or label (e.g. indicating what is the object depicted in image i). Based on these data, we want to come up with a function $\hat{f} : \mathbb{R}^d \to \mathbb{R}$ that models the dependency of y_i on $\boldsymbol{x}_i$. This allows, for instance to classify a new image. A crucial aspects of this problem is that the joint distribution $\mathbb{P}$ of $(y_i, \boldsymbol{x}_i)$ is unknown.

To make the setting more concrete, we can think (without loss of generality) that the relation between y_i and $\boldsymbol{x}_i$ is given by

$$y_i = f(\boldsymbol{x}_i) + z_i\,, \tag{1.1}$$

where $f : \mathbb{R}^d \to \mathbb{R}$ is a certain function that we want to learn, and the 'noise' z_i has zero expectation $\mathbb{E}(z_i) = 0$, $\mathrm{Var}(z_i) = \sigma^2$. We can think that the function $\hat{f}$ is parametrized by a vector of parameters $\boldsymbol{\theta} = (\theta_1, \dots, \theta_p)$. Namely $\hat{f} : (\boldsymbol{x}, \boldsymbol{\theta}) \mapsto \hat{f}(\boldsymbol{x}; \boldsymbol{\theta})$. Learning the function $\hat{f}$ is equivalent to learning the parameters $\boldsymbol{\theta}$.

Example 1.1. As an example, we can think of fitting a polynomial of maximum degree k. In this case

$$\hat{f}(\boldsymbol{x}; \boldsymbol{\theta}) = \sum_{\alpha:\,|\alpha|\le k} \theta_{\alpha_1,\dots,\alpha_d} x_1^{\alpha_1} \cdots x_d^{\alpha_d}\,, \tag{1.2}$$

where $\alpha = (\alpha_1, \dots, \alpha_d)$, and $|\alpha| = \sum_{i\le d} \alpha_i$.

We qualify the accuracy of such a predictor $\hat{f}(\,\cdot\,; \boldsymbol{\theta})$ via

$$R(\boldsymbol{\theta}) = \mathbb{E}\big\{\big[f(\boldsymbol{x}) - \hat{f}(\boldsymbol{x}; \boldsymbol{\theta})\big]^2\big\} + \sigma^2 \tag{1.3}$$

$$= \mathbb{E}\big\{\big[y - \hat{f}(\boldsymbol{x}; \boldsymbol{\theta})\big]^2\big\}\,. \tag{1.4}$$

[*] Department of Electrical Engineering and Department of Statistics, Stanford University

This quantity is known in the literature as 'prediction error,' 'test error,' or 'population risk' depending on the sub-community.

A classical approach consists in replacing the population risk by its empirical version, thus minimizing

$$\widehat{R}_n(\boldsymbol{\theta}) = \frac{1}{n}\sum_{i=1}^{n}\big[y_i - \hat{f}(\boldsymbol{x}_i;\boldsymbol{\theta})\big]^2 \, . \tag{1.5}$$

We will not pursue this direction further in these lecture, but instead consider a specific (efficient) algorithm, namely stochastic gradient descent.

2 The simplest neural network model

An important question is: how to construct a rich enough class of functions, as to fit complex data? Two-layers neural networks consider functions of the form [Ros62]

$$\hat{f}(\boldsymbol{x};\boldsymbol{\theta}) = \frac{1}{N}\sum_{i=1}^{N}\sigma_*(\boldsymbol{x};\boldsymbol{\theta}_i)\, . \tag{2.1}$$

Here N is the number of hidden units (neurons), $\sigma_* : \mathbb{R}^d \times \mathbb{R}^D \to \mathbb{R}$ is an activation function, and $\boldsymbol{\theta}_i \in \mathbb{R}^D$ are parameters, which we collectively denote by $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_N)$. The factor $(1/N)$ is introduced for convenience and can be eliminated by redefining the activation.

In fact, the above formulation is more general than what is used in practice. The standard choice is to take $\boldsymbol{\theta}_i = (a_i, b_i, \boldsymbol{w}_i)$, where a_i is the weight of unit i, b_i is an offset, and $\boldsymbol{w}_i \in \mathbb{R}^d$ is a weight vector, and

$$\sigma_*(\boldsymbol{x};\boldsymbol{\theta}_i) = a_i\, \sigma(\langle \boldsymbol{w}_i, \boldsymbol{x}\rangle + b_i)\, , \tag{2.2}$$

for some $\sigma : \mathbb{R} \to \mathbb{R}$. In this case of course $D = d + 2$. Standard examples are

$$\sigma(x) = \frac{1}{1+e^{-2x}} \qquad \text{(sigmoid)}, \tag{2.3}$$

$$\sigma(x) = \max(x, 0) \qquad \text{(Rectified Linear Unit, ReLU)}. \tag{2.4}$$

Is this class of functions rich enough? This question was studied in the nineties. Here is a basic result in this direction, from [Cyb89] (the original statement is slightly different).

Theorem 2.1 (Cybenko, 1989)**.** *Assume* $\mathbb{E}(f(x)^2) < \infty$, *and further assume* $\sigma : \mathbb{R} \to \mathbb{R}$ *to be continuous with* $\lim_{x\to\infty}\sigma(x) \to 1$ *and* $\lim_{x\to-\infty}\sigma(x) \to 0$. *Then, for any* ε, *there exists* $N = N(\varepsilon)$, *such that*

$$\inf_{\{(a_i,b_i,\boldsymbol{w}_i)\}} \mathbb{E}\left\{\left[f(\boldsymbol{x}) - \frac{1}{N}\sum_{i=1}^{N} a_i\,\sigma(\langle \boldsymbol{w}_i, \boldsymbol{x}\rangle + b_i)\right]\right\} \le \varepsilon\, . \tag{2.5}$$

In other words, we can approximate any 'reasonable' function arbitrarily well. Before discussing how this is proved, it is worth mentioning that this result is similar to something that you already know about: Fourier analysis. When N gets very large, you can replace the parameters $\boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_N$ by their density ρ (which is a probability measure in $\mathbb{R}^D$) and hence replace Eq. (2.1) by

$$\hat{f}(\boldsymbol{x}) = \int \sigma_*(\boldsymbol{x};\boldsymbol{\theta})\,\rho(\mathrm{d}\boldsymbol{\theta})\, . \tag{2.6}$$

When the neuron takes the form (2.2), we get therefore

$$\hat{f}(\boldsymbol{x};\rho) = \int a\sigma(\langle \boldsymbol{w}, \boldsymbol{x}\rangle + b)\,\rho(\mathrm{d}a, \mathrm{d}b, \mathrm{d}\boldsymbol{w}) \tag{2.7}$$

$$= \int \sigma(\langle \boldsymbol{w}, \boldsymbol{x}\rangle + b)\,\mu(\mathrm{d}b, \mathrm{d}\boldsymbol{w})\,, \tag{2.8}$$

$$\mu(b, w) = \int_{\mathbb{R}} a\,\rho(\mathrm{d}a, b, \boldsymbol{w})\,. \tag{2.9}$$

In other words μ is obtained by integrating out a. Technically, it is a signed measure. Take the special case $b = 0$ and $\sigma(x) = e^{ix}$. Then

$$\hat{f}(\boldsymbol{x};\mu) = \int e^{i\langle \boldsymbol{w}, \boldsymbol{x}\rangle}\mu(\mathrm{d}\boldsymbol{w})\,. \tag{2.10}$$

In other words, $\hat{f}$ is the Fourier transform of μ. Fourier analysis suggests that any (square integrable) function can be represented in this way.

Sketch of proof of Theorem 2.1. We let P denote the distribution of $\boldsymbol{x}$.

Let $\mathscr{L}$ be the linear space of functions that can be written as linear combinations of functions as (2.2)

$$\mathscr{L} = \left\{ \frac{1}{N}\sum_{i=1}^{N} a_i\,\sigma(\langle \boldsymbol{w}_i, \boldsymbol{x}\rangle + b_i) : \quad N \in \mathbb{N}, a_i, b_i \in \mathbb{R}, \boldsymbol{w}_i \in \mathbb{R}^d \right\}\,, \tag{2.11}$$

and denote by $\overline{\mathscr{L}}$ its closure (in $L^2(\mathsf{P})$). We want to prove that $\overline{\mathscr{L}} = L^2(\mathsf{P})$.

Assume by contradiction that there is $f \notin \overline{\mathscr{L}}$. Then there id an g that is orthogonal to $\overline{\mathscr{L}}$. Then, it is orthogonal to every activation function:

$$\int g(\boldsymbol{x})\sigma(\langle \boldsymbol{w}, \boldsymbol{x}\rangle + b)\,\mathsf{P}(\mathrm{d}\boldsymbol{x}) = 0\,. \tag{2.12}$$

We can take $\boldsymbol{w} = \alpha\boldsymbol{v}$, $b = -\alpha c$, $\alpha \to \infty$, to get

$$\int g(\boldsymbol{x})\mathbf{1}_{\{\langle \boldsymbol{v}, \boldsymbol{x}\rangle \ge c\}}\,\mathsf{P}(\mathrm{d}\boldsymbol{x}) = 0\,. \tag{2.13}$$

In other words the intergal of g over any half-space is zero. It is not hard to show that this implies $g(\boldsymbol{x}) = 0$ (for P-almost every $\boldsymbol{x}$). □

The next question is: how big should N be for 'reasonable' functions f? Andrew Barron proved a classical theorem about this problem. (Here $\mathsf{B}(\mathbf{0}, r)$ demotes the ball of radius r in d dimensions.)

Theorem 2.2 (Barron, 1993)**.** *Assume* P *t be supported on* $\mathsf{B}(0, r)$*, and let* $f : \mathbb{R}^d \to \mathbb{R}$ *be a function with Fourier transform* F*:* $f(\boldsymbol{x}) = \int e^{i\langle \boldsymbol{\omega}, \boldsymbol{z}\rangle}F(\boldsymbol{\omega})\mathrm{d}\boldsymbol{\omega}$*. Let* $\sigma : \mathbb{R} \to \mathbb{R}$ *be such that* $\lim_{t\to\infty}\sigma(t) = 1$, $\lim_{t\to\infty}\sigma(t) = 0$.

Define

$$N(\varepsilon) \equiv \frac{1}{\varepsilon}\left(2r\int \|\boldsymbol{\omega}\|_2\,|F(\boldsymbol{\omega})|\mathrm{d}\boldsymbol{\omega}\right)^2\,. \tag{2.14}$$

Then there exists a network of the form (2.11) with $N(\varepsilon)$ *hidden unit achieving error* $\mathbb{E}\{(\hat{f}(\boldsymbol{x};\boldsymbol{\theta}) - f(\boldsymbol{x}))\} \le \varepsilon$*.*

Of course there are interesting functions for which the number $N(\varepsilon)$ is very large (exponential in d) and require a very large two-layers network. On the other hand, they can be represented compactly with a larger number of layers. An example is constructed in [ES16].

3 Stochastic gradient descent

Suppose you want to minimize a smooth function $R(\boldsymbol{\theta})$. The simplest algorithm you might want to try is probably gradient descent:

$$\boldsymbol{\theta}^{k+1} = \boldsymbol{\theta}^k + s_k\, \boldsymbol{v}_k\,, \qquad \boldsymbol{v}_k = -\nabla R(\boldsymbol{\theta}^k)\,. \tag{3.1}$$

Here s_k is the step size. In order to ensure convergence s_k needs to decrease in the right way with k. This algorithm is more than 170 years old [Cau47], and there has been some progress since.

One major step forward has been the idea that we do not need to compute exact gradients. Suppose for instance that we are given noisy observations of the gradient

$$\boldsymbol{\theta}^{k+1} = \boldsymbol{\theta}^k + s_k\, \boldsymbol{v}_k\,, \qquad \boldsymbol{v}_k = -\nabla R(\boldsymbol{\theta}^k) + \boldsymbol{z}^k\,. \tag{3.2}$$

where $\boldsymbol{z}^k$ is i.i.d. noise (across time), with zero mean. It was first realized by Robbins and Monro [RM51] that the algorithm converges the same (with suitably chosen step sizes). The noise 'averages out.' The resulting algorithm is known as SGD (stochastic gradient descent).

In our case we cannot even evaluate $R(\boldsymbol{\theta})$, but we have samples $(y_i, \boldsymbol{x}_i)$, and

$$R_N(\boldsymbol{\theta}) = \mathbb{E}\ell(y_i, \boldsymbol{x}_i; \boldsymbol{\theta}),, \qquad \ell(y_i, \boldsymbol{x}_i; \boldsymbol{\theta}) = \Big(y_i - \hat{f}(\boldsymbol{x}_i; \boldsymbol{\theta})\Big)^2\,. \tag{3.3}$$

We can then implement SGD by taking a step in the direction of the gradient of $\ell(y_i, \boldsymbol{x}_i; \boldsymbol{\theta})$:

$$\boldsymbol{\theta}^{k+1} = \boldsymbol{\theta}^k - s_k\, \nabla_{\boldsymbol{\theta}}\ell(y_k, \boldsymbol{x}_k; \boldsymbol{\theta}^k)\,. \tag{3.4}$$

I will assume here that I make only one pass over the data and hence the gradients are really i.i.d. with $\mathbb{E}\nabla\ell(y_k, \boldsymbol{x}_k; \boldsymbol{\theta}) = \nabla R(\boldsymbol{\theta})$.

In other words, I am hoping to converge to a good θ fast enough so that I do not run out of data. In reality, multiple passes over data are often useful even in large scale applications. However, as a simplifying assumption this is not too bad.

When we specialize this algorithm to networks of the form (2.1), we get

$$\boldsymbol{\theta}_i^{k+1} = \boldsymbol{\theta}_i^k + 2s_k\, \nabla_{\boldsymbol{\theta}_i}\sigma_*(\boldsymbol{x}_k; \boldsymbol{\theta}_i^k)\, \left(y_k - \frac{1}{N}\sum_{i=1}^N \sigma_*(\boldsymbol{x}_k; \boldsymbol{\theta}_i^k)\right)\,. \tag{3.5}$$

4 Mean field limit: Statics

Let's reconsider the population risk, which we denote by $R_N(\boldsymbol{\theta})$, to emphasize the dependence on the number of neurons. By expanding the square, we get the expression

$$R_N(\boldsymbol{\theta}) \equiv R_{\#} + \frac{2}{N}\sum_{i=1}^N V(\boldsymbol{\theta}_i) + \frac{1}{N^2}\sum_{i,j=1}^N U(\boldsymbol{\theta}_i, \boldsymbol{\theta}_j)\,, \tag{4.1}$$

$$R_{\#} = \mathbb{E}\{y^2\}\,, \qquad V(\boldsymbol{\theta}) = -\mathbb{E}\{y\,\sigma_*(\boldsymbol{x}; \boldsymbol{\theta})\}\,, \tag{4.2}$$

$$U(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2) = \mathbb{E}\{\sigma_*(\boldsymbol{x}; \boldsymbol{\theta}_1)\sigma_*(\boldsymbol{x}; \boldsymbol{\theta}_2)\}\,. \tag{4.3}$$

In physical terms, $R_N(\boldsymbol{\theta})$ is the energy of a system of N particles in D dimensions, interacting via pairwise potentials $U(\boldsymbol{\theta}_i, \boldsymbol{\theta}_j)$, ad moving in an external potential $V(\boldsymbol{\theta}_i)$. An important observation is that the kernel U is positive semidefinite, i.e. for any (bounded, compactly supported) function h, we have

$$\int\int U(\boldsymbol{\theta}_1,\boldsymbol{\theta}_2)\, h(\boldsymbol{\theta}_1)\, h(\boldsymbol{\theta}_2)\, \mathrm{d}\boldsymbol{\theta}_1 \mathrm{d}\boldsymbol{\theta}_2 \geq 0\,. \tag{4.4}$$

Physically, this corresponds to U being a repulsive interaction (in an average sense).

For large N, it makes sense to replace the positions $\boldsymbol{\theta}_1,\dots,\boldsymbol{\theta}_N$ by a density $\rho \in \mathscr{P}(\mathbb{R}^D)$ (we will denote by $\mathscr{P}(\Omega)$ the space of probability distributions in Ω), defined by

$$R(\rho) \equiv R_{\#} + 2\int V(\boldsymbol{\theta})\ \rho(\mathrm{d}\boldsymbol{\theta}) + \int U(\boldsymbol{\theta}_1,\boldsymbol{\theta}_2)\ \rho(\mathrm{d}\boldsymbol{\theta}_1)\, \rho(\mathrm{d}\boldsymbol{\theta}_2)\,. \tag{4.5}$$

The following simple result shows that there is not much difference in minimizing $R_N(\boldsymbol{\theta})$, versus minimizing $R(\rho)$.

Proposition 4.1. *Assume that there exists $\varepsilon_0 > 0$ such that, for any $\rho \in \mathscr{P}(\mathbb{R}^D)$ such that $R(\rho) \leq \inf_\rho R(\rho) + \varepsilon_0$ we have $\int U(\boldsymbol{\theta},\boldsymbol{\theta})\, \rho(\mathrm{d}\boldsymbol{\theta}) \leq K$. Then*

$$\big|\inf_{\boldsymbol{\theta}} R_N(\boldsymbol{\theta}) - \inf_{\rho} R(\rho)\big| \leq \frac{K}{N}\,. \tag{4.6}$$

For future reference, it is useful to define the functional derivative

$$\Psi(\boldsymbol{\theta};\rho) \equiv \frac{1}{2}\frac{\delta R(\rho)}{\delta\rho(\boldsymbol{\theta})} = V(\boldsymbol{\theta}) + \int U(\boldsymbol{\theta},\boldsymbol{\theta}')\ \rho(\mathrm{d}\boldsymbol{\theta}')\,. \tag{4.7}$$

This can be interpreted as the additional energy of adding a single particle at $\boldsymbol{\theta} \in \mathbb{R}^D$. Global minima are distributions ρ_* such that

$$\mathrm{supp}(\rho_*) \subseteq \arg\min_{\boldsymbol{\theta}\in\mathbb{R}^D} \Psi(\boldsymbol{\theta};\rho_*)\,. \tag{4.8}$$

In other words the energy cannot be decreased by moving an infinitesimal mass from $\mathrm{supp}(\rho_*)$ elsewhere.

5 Mean field limits: Dynamics

Consider now the SGD dynamics (3.5), and set for simplicity $s_k = \varepsilon/2 \ll 1$. We will define a time variable t by letting $k = \lfloor k/\varepsilon \rfloor$. This describes a set of N particles, with velocity of particle i given by

$$\boldsymbol{v}_i^k = \nabla_{\boldsymbol{\theta}_i}(y_k \sigma_*(\boldsymbol{x}_k;\boldsymbol{\theta}_i^k)) - \frac{1}{N}\sum_{j=1}^N \nabla_{\boldsymbol{\theta}_i}\sigma_*(\boldsymbol{x}_k;\boldsymbol{\theta}_i^k)\sigma_*(\boldsymbol{x}_k;\boldsymbol{\theta}_j^k)\,. \tag{5.1}$$

If we take expectation over $y_k, \boldsymbol{x}_k$, given the past (denoted by $\mathcal{F}_k$), we get

$$\mathbb{E}(\boldsymbol{v}_i^k|\mathcal{F}_k) = -\nabla_{\boldsymbol{\theta}_i} V(\boldsymbol{\theta}_i^k) - \frac{1}{N}\sum_{j=1}^N \nabla_{\boldsymbol{\theta}_i} U(\boldsymbol{\theta}_i^k,\boldsymbol{\theta}_j^k)\,. \tag{5.2}$$

If at time k the density of particles is approximately $\rho_{t=k\varepsilon}$, then

$$\mathbb{E}(\boldsymbol{v}_i^k|\mathcal{F}_k) \approx \boldsymbol{v}(\boldsymbol{\theta}_i^k;\rho_t) = \nabla\Psi(\theta_i^k;\rho_t)\,. \tag{5.3}$$

The density ρ_t should satisfy the continuity equation $\partial_t\rho_t(\boldsymbol{\theta}) + \nabla\cdot(\rho_t(\boldsymbol{\theta})\boldsymbol{v}(\boldsymbol{\theta};\rho_t))$, which we can rewrite more explicitly as

$$\partial_t\rho_t(\boldsymbol{\theta}) = \nabla_{\boldsymbol{\theta}}\cdot\Big(\rho_t(\boldsymbol{\theta})\nabla_{\boldsymbol{\theta}}\Psi(\boldsymbol{\theta};\rho_t)\Big)\,,. \tag{5.4}$$

The fixed points of this dynamics are densities ρ_* such that all mass sits in zero velocity positions

$$\mathrm{supp}(\rho_*) \subseteq \Big\{\boldsymbol{\theta}\in\mathbb{R}^D:\ \nabla\Psi(\boldsymbol{\theta};\rho_*) = 0\Big\}\,. \tag{5.5}$$

A quantitative statement is given by the following result from [MMN18].

Theorem 5.1. *Assume that the following conditions hold:*

A2. *The activation function $(\boldsymbol{x},\boldsymbol{\theta})\mapsto\sigma_*(\boldsymbol{x};\boldsymbol{\theta})$ is bounded, with sub-Gaussian gradient: $\|\sigma_*\|_\infty \le K_2$, $\|\nabla_{\boldsymbol{\theta}}\sigma_*(\boldsymbol{X};\boldsymbol{\theta})\|_{\psi_2}\le K_2$. Labels are bounded $|y_k|\le K_2$.*

A3. *The gradients $\boldsymbol{\theta}\mapsto\nabla V(\boldsymbol{\theta})$, $(\boldsymbol{\theta}_1,\boldsymbol{\theta}_2)\mapsto\nabla_{\boldsymbol{\theta}_1}U(\boldsymbol{\theta}_1,\boldsymbol{\theta}_2)$ are bounded, Lipschitz continuous (namely $\|\nabla_{\boldsymbol{\theta}}V(\boldsymbol{\theta})\|_2$, $\|\nabla_{\boldsymbol{\theta}_1}U(\boldsymbol{\theta}_1,\boldsymbol{\theta}_2)\|_2\le K_3$, $\|\nabla_{\boldsymbol{\theta}}V(\boldsymbol{\theta})-\nabla_{\boldsymbol{\theta}}V(\boldsymbol{\theta}')\|_2\le K_3\|\boldsymbol{\theta}-\boldsymbol{\theta}'\|_2$, $\|\nabla_{\boldsymbol{\theta}_1}U(\boldsymbol{\theta}_1,\boldsymbol{\theta}_2)-\nabla_{\boldsymbol{\theta}_1}U(\boldsymbol{\theta}_1',\boldsymbol{\theta}_2')\|_2\le K_3\|(\boldsymbol{\theta}_1,\boldsymbol{\theta}_2)-(\boldsymbol{\theta}_1',\boldsymbol{\theta}_2')\|_2$).*

For $\rho_0\in\mathscr{P}(\mathbb{R}^D)$, consider SGD with initialization $(\boldsymbol{\theta}_i^0)_{i\le N}\sim_{iid}\rho_0$ and step size $s_k=\varepsilon/2$. For $t\ge 0$, let ρ_t be the solution of PDE (5.4). Then, there exists a constant C (depending uniquely on the parameters K_i of conditions A1-A2) such that, for any $f:\mathbb{R}^D\times\mathbb{R}\to\mathbb{R}$, with $\|f\|_\infty,\|f\|_{\mathrm{Lip}}\le 1$, $\varepsilon\le 1$,

$$\sup_{k\in[0,T/\varepsilon]\cap\mathbb{N}}\big|R_N(\boldsymbol{\theta}^k)-R(\rho_{k\varepsilon})\big| \le Ce^{CT}\sqrt{1/N\vee\varepsilon}\cdot\Big[\sqrt{D+\log(N/\varepsilon)}+z\Big]\,, \tag{5.6}$$

with probability $1-e^{-z^2}$.

Remarkably, the PDE approximation is accurate as soon as $N\gg D$, $\varepsilon\ll 1/D$.

Related results were recently (independently) proven in [RVE18, SS18, CB18]. See also [WML17] for a similar approach, although in a different context.

6 Examples

By using the PDE description we can prove convergence (or non-convergence) of SGD to a global optimum in some specific examples. Figures 6.1, 6.2, 6.3 show some comparison of SGD simulations with the PDE solution.

A simple example in which we prove global convergence [MMN18], is the following distribution

With probability 1/2: $y=+1$, $\boldsymbol{x}\sim\mathsf{N}(0,(1+\Delta)^2\boldsymbol{I}_d)$

With probability 1/2: $y=-1$, $\boldsymbol{x}\sim\mathsf{N}(0,(1-\Delta)^2\boldsymbol{I}_d)$.

We choose an activation function without offset or output weights, namely $\sigma_*(\boldsymbol{x};\boldsymbol{\theta}_i)=\sigma(\langle\boldsymbol{w}_i,\boldsymbol{x}\rangle)$. While qualitatively similar results are obtained for other choices of σ, we will use a simple piecewise linear function as a running example: $\sigma(t)=s_1$ if $t\le t_1$, $\sigma(t)=s_2$ if $t\ge t_2$, and $\sigma(t)$ interpolated linearly for $t\in(t_1,t_2)$. In simulations we use $t_1=0.5$, $t_2=1.5$, $s_1=-2.5$, $s_2=7.5$.

Figure 6.3 shows instead one example in which SGD fails unless it is initialized close to a global optimum. The data have the same distribution as above, but we selected a non-monotone activation.

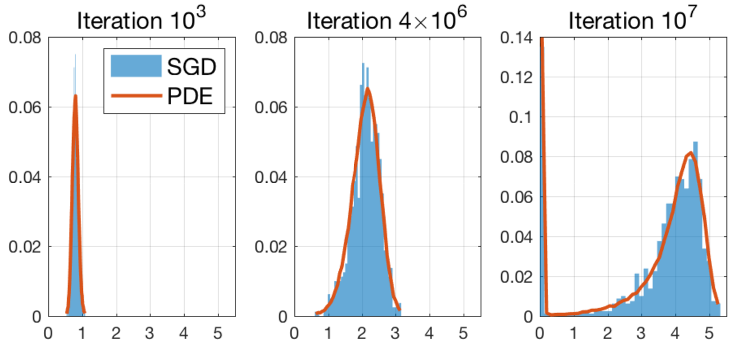

Figure 6.1: Evolution of the radial distribution $\overline{\rho}_t$ for the isotropic Gaussian model, with $\Delta = 0.8$. Histograms are obtained from SGD experiments with $d = 40$, $N = 800$, initial weights distribution $\rho_0 = \mathsf{N}(\mathbf{0}, 0.8^2/d \cdot \boldsymbol{I}_d)$, step size $\epsilon = 10^{-6}$. Continuous lines correspond to a numerical solution of the PDE (5.4).

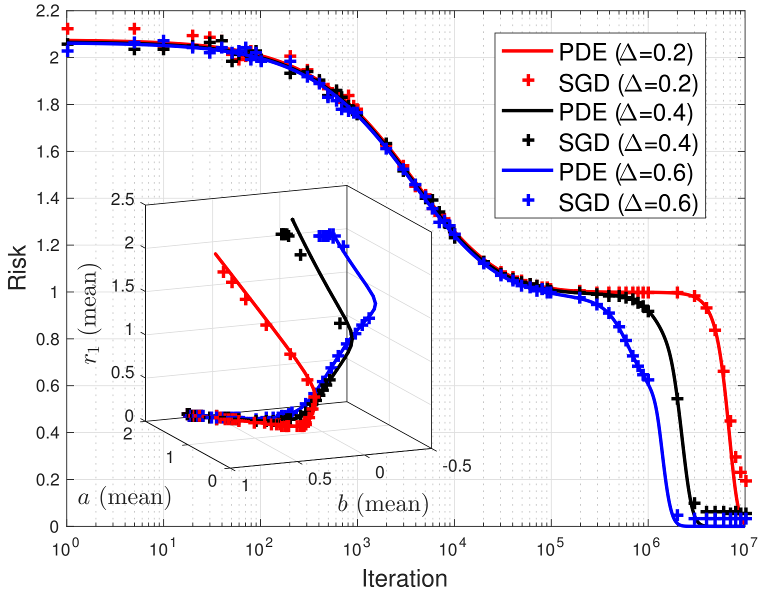

Figure 6.2: Evolution of the population risk for a variable selection problem using a two-layers neural network with ReLU activations. Here $d = 320$, $s_0 = 60$, $N = 800$, and we used $\xi(t) = t^{-1/4}$ and $\varepsilon = 2 \times 10^{-4}$ to set the step size. Numerical simulations using SGD (one run per data point) are marked "+", and curves are solutions of the reduced PDE with $d = \infty$. Inset: evolution of three parameters of the reduced distribution $\overline{\rho}_t$ (average output weights a, average offsets b and average ℓ_2 norm in the relevant subspace r_1) for the same setting.

7 Gradient flows

The PDE (5.4) has an interesting structure that its inherited from its origin as a description of SGD. In a single sentence:

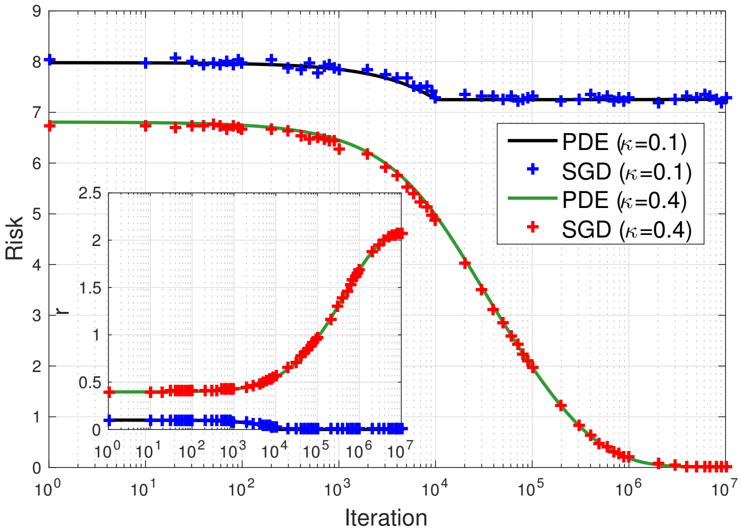

Figure 6.3: Separating two isotropic Gaussians, with a non-monotone activation function (see text for details). Here $N = 800$, $d = 320$, $\Delta = 0.5$. The main frame presents the evolution of the population risk along the SGD trajectory, starting from two different initializations of $(\boldsymbol{w}_i^0)_{i\le N} \sim_{iid} \mathsf{N}(\mathbf{0}, \kappa^2/d \cdot \mathbf{I}_d)$ for either $\kappa = 0.1$ or $\kappa = 0.4$. In the inset, we plot the evolution of the average of $\|\boldsymbol{w}\|_2$ for the same conditions. Symbols are empirical results. Continuous lines are prediction obtained by solving the PDE (5.4).

Equation (5.4) is the gradient flow for $R(\rho)$ in Wassersein metric.

It is worth to try to understand this statement because it is related to some interesting mathematics [AGS08]. This point of view was first developed in a seminal paper by Jordan, Kinderlehrer and Otto [JKO98], which was about the Fokker-Planck equation (which is relater to the case $U = 0$ of our equation, see also next section).

Let us take a detour and consider continuous time gradient descent for a function $F : \mathbb{R}^d \to \mathbb{R}$:

$$\dot{\boldsymbol{x}}(t) = -\nabla F(\boldsymbol{x}(t))\,. \tag{7.1}$$

The resulting dynamics is also referred to as 'gradient flow.' It turns out that there is a more general way to think about this flow. The key observation is that, for small ε,

$$\boldsymbol{x}(t+\varepsilon) \approx \arg\min_{\boldsymbol{z}\in\mathbb{R}^d} \left\{ F(\boldsymbol{z}) + \frac{1}{2\varepsilon}\|\boldsymbol{z} - \boldsymbol{x}(t)\|_2^2 \right\}\,. \tag{7.2}$$

We can reverse the point of view and use this as a definition for the gradient flow. This point of view is more general. Namely, for any distance function $d(\,\cdot\,,\,\cdot\,)$, we can define a trajectory by letting

$$\boldsymbol{x}_\varepsilon((k+1)\varepsilon) \approx \arg\min_{\boldsymbol{z}\in\mathbb{R}^d} \left\{ F(\boldsymbol{z}) + \frac{1}{2\varepsilon}\, d(\boldsymbol{z}, \boldsymbol{x}_\varepsilon(t))^2 \right\}\,, \tag{7.3}$$

and interpolating linearly between these points. We then take the limit $\varepsilon \to 0$ and get a continuous trajectory.

Therefore, we can (try to) define a gradient flow for any space, any cost function and any metric (any distance) on that space. The (5.4) is a gradient flow with the following ingredients

- The space is the space of probability measures ρ in $\mathbb{R}^D$ with finite second moment $\int \|\boldsymbol{\theta}\|_2^2 \rho(\mathrm{d}\boldsymbol{\theta}) < \infty$, denoted by $\mathscr{P}_2(\mathbb{R}^d)$.

- The cost is the risk $R(\rho)$.

- The metric is the Wasserstein distance W_2 between probability distributions:

$$W_2(\mu, \nu) = \Big(\inf_{\gamma \in \mathcal{C}(\mu,\nu)} \int \|\boldsymbol{x} - \boldsymbol{y}\|_2^2 \gamma(\mathrm{d}\boldsymbol{x}, \mathrm{d}\boldsymbol{y}) \Big)^{1/2}. \tag{7.4}$$

  where the infimum is over all the couplings of μ and ν, i.e. over all probability distributions γ on $\mathbb{R}^d \times \mathbb{R}^d$ whose marginals are equal to μ and ν.

Roughly speaking, we have the following interpretation of the PDE (5.4):

$$\rho_{t+\varepsilon} \approx \arg\min_{\rho \in \mathbb{R}^d} \left\{ R(\rho) + \frac{1}{2\varepsilon} W_2(\rho, \rho_t)^2 \right\}. \tag{7.5}$$

8 Noisy SGD

The above framework can be generalized to an interesting variant of SGD, whereby at each step we add noise $\boldsymbol{g}_i^k \sim \mathsf{N}(0, \boldsymbol{I}_D)$:

$$\boldsymbol{\theta}_i^{k+1} = \boldsymbol{\theta}_i^k + 2s_k \, \nabla_{\boldsymbol{\theta}_i} \sigma_*(\boldsymbol{x}_k; \boldsymbol{\theta}_i^k) \left(y_k - \frac{1}{N} \sum_{i=1}^N \sigma_*(\boldsymbol{x}_k; \boldsymbol{\theta}_i^k) \right) + \sqrt{2Ts_k} \, \boldsymbol{g}_i^k \, . \tag{8.1}$$

Unsurprisingly, this results in a diffusion term added to the PDE:

$$\partial_t \rho_t(\boldsymbol{\theta}) = \nabla_{\boldsymbol{\theta}} \cdot \Big(\rho_t(\boldsymbol{\theta}) \nabla_{\boldsymbol{\theta}} \Psi(\boldsymbol{\theta}; \rho_t) \Big) + T \, \Delta \rho_t(\boldsymbol{\theta}) \, . \tag{8.2}$$

Also this has the interpretation of gradient flow, although now the quantity that is minimized is a free energy

$$F(\rho) = \frac{1}{2} \, R(\rho) - T \, S(\rho) \, , \tag{8.3}$$

$$S(\rho) = - \int \rho(\boldsymbol{\theta}) \log \rho(\boldsymbol{\theta}) \, \mathrm{d}\boldsymbol{\theta} \, . \tag{8.4}$$

This free energy is strongly convex (thanks to the fact that U is PSD) and has a unique minimizer ρ_* that solves the following self-consistent Boltzmann equation, for $\beta = 1/T$,

$$\rho_*(\boldsymbol{\theta}) = \frac{1}{Z(\beta)} \exp\Big\{ - \beta \Psi(\boldsymbol{\theta}; \rho_*) \Big\} \, . \tag{8.5}$$

The free energy is monotone decreasing along the solutions of the PDE, and we can compute the rate of free energy dissipation

$$\frac{\mathrm{d}F(\rho_t)}{\mathrm{d}t} = - \int_{\mathbb{R}^D} \|\nabla_{\boldsymbol{\theta}} (\Psi(\boldsymbol{\theta}; \rho_t) + T \, \log \rho_t(\boldsymbol{\theta}))\|_2^2 \rho_t(\boldsymbol{\theta}) \mathrm{d}\boldsymbol{\theta}. \tag{8.6}$$

It follows from this expression that the only fixed point of the dynamics is the unique solution of (8.5): if $\rho_t \neq \rho_*$, then the entropy dissipation is strictly positive. Hence we conclude that ρ_t converges to the global optimum in a time that can depend on D but does not depend on N! We state this below, referring to [MMN18] for a more formal version.

Theorem 8.1. *For smooth U, V, there exists $t_*(\delta,\beta), \beta_*(\varepsilon) < \infty$ such that, for any $\beta > \beta_*(\varepsilon)$, $t > t_*(\delta,\beta)$, we have*

$$R(\rho_t) \le R(\rho) + \delta\,. \tag{8.7}$$

In particular, SGD reaches a near optimum in time independent of the number of neurons.

References

[AGS08] Luigi Ambrosio, Nicola Gigli, and Giuseppe Savaré, *Gradient flows: in metric spaces and in the space of probability measures*, Springer Science & Business Media, 2008.

[Cau47] Augustin Cauchy, *Méthode générale pour la résolution des systemes d'équations simultanées*, Comp. Rend. Sci. Paris **25** (1847), no. 1847, 536–538.

[CB18] Lenaic Chizat and Francis Bach, *On the global convergence of gradient descent for over-parameterized models using optimal transport*, arXiv:1805.09545 (2018).

[Cyb89] George Cybenko, *Approximation by superpositions of a sigmoidal function*, Mathematics of control, signals and systems **2** (1989), no. 4, 303–314.

[ES16] Ronen Eldan and Ohad Shamir, *The power of depth for feedforward neural networks*, Conference on Learning Theory, 2016, pp. 907–940.

[JKO98] Richard Jordan, David Kinderlehrer, and Felix Otto, *The variational formulation of the fokker–planck equation*, SIAM journal on mathematical analysis **29** (1998), no. 1, 1–17.

[MMN18] Song Mei, Andrea Montanari, and Phan-Minh Nguyen, *A mean field view of the landscape of two-layer neural networks*, Proceedings of the National Academy of Sciences **115** (2018), no. 33, E7665–E7671.

[RM51] Herbert Robbins and Sutton Monro, *A stochastic approximation method*, The annals of mathematical statistics (1951), 400–407.

[Ros62] Frank Rosenblatt, *Principles of neurodynamics*, Spartan Book, 1962.

[RVE18] Grant M Rotskoff and Eric Vanden-Eijnden, *Neural networks as interacting particle systems: Asymptotic convexity of the loss landscape and universal scaling of the approximation error*, arXiv:1805.00915 (2018).

[SS18] Justin Sirignano and Konstantinos Spiliopoulos, *Mean field analysis of neural networks*, arXiv:1805.01053 (2018).

[WML17] Chuang Wang, Jonathan Mattingly, and Yue M Lu, *Scaling limit: Exact and tractable analysis of online learning algorithms with applications to regularized regression and pca*, arXiv:1712.04332 (2017).