Math of ML : Exercises 11 *

December 1, 2025

Before attempting the exercise, read the following blog posts:

- https://francisbach.com/gradient-descent-neural-networks-global-convergence/
- https://francisbach.com/gradient-descent-for-wide-two-layer-neural-networks-implicit-bias/

Exercise 1. *Consider a two-layer ReLU neural network*

$$f(x) = \frac{1}{m}\sum_{i=1}^{n} a_i \sigma_{\text{relu}}(b_i^\top x),$$

where $x \in \mathbb{R}^d$ is the input vector, $b_i \in \mathbb{R}^d$, $i = 1, \dots, m$ are the neurons in the first layer of the network, $\sigma_{\text{relu}}(x) = \max\{0, x\}$ is the ReLU activation function, and $a_i \in \mathbb{R}, i = 1, \dots, m$ are the second layer weights. The trainable model parameters are $\theta = (a_i, b_i)_{i=1}^m$ and we will denote the neural network with weights θ by f_θ. Finally, let the loss function be the quadratic loss: $R(\theta) = \frac{1}{n}\sum_{i=1}^n (f_\theta(X_i) - Y_i)^2$, where $(X_i, Y_i)_{i=1}^n$ is the set of observations. Perform the following simulations.

1. *(Role of the initialization scale) Let $\alpha > 0$ be the initialization scale parameter. For $j = 1, \dots, m$, initialize the weights $b_j = \alpha \tilde{b}_j$, and $a_j = \alpha \tilde{a}_j$, where the weights $\tilde{b}_j$ are sampled from the uniform distribution on the unit sphere $\mathcal{S}^{d-1} \subseteq \mathbb{R}^d$ and $\tilde{a}_j$ are distributed uniformly on $\{-1, +1\}$.*

   Fix $d = 2$ and generate the data by letting $X_i \sim \mathcal{N}(0, I_2)$ and $Y_i = f_{\theta^*}(X_i)$, where θ^* is some two-layer ReLU neural network with 5 neurons and the weights initialized via the distribution described above (setting $\alpha = 1$).

   Train a two-layer ReLU neural network by running gradient descent with the above-described initialization scheme with a large number of neurons m. Display the learned neural network using a colour plot (see the first figure in the second blog post linked above for an example). Observe how the learned function changes depending on the initialization scale parameter α. Which values of α lead to the best performance? How does the learned neural network behave for $\alpha = \sqrt{m}$?

2. *(Adaptivity to unknown linear structures) Consider the simulation setting described above. This time, let $\tilde{X}_i = ((X_i)_1, (X_i)_2, (Z_i)_1, \dots, (Z_i)_8)^\top$, where $Z_i \in \mathbb{R}^8$ is distributed as $\mathcal{N}(0, I_8)$. Thus, the outputs Y_i only depend on the first two coordinates of $\tilde{X}_i$.*

   Repeat the simulations of the previous exercise for the data $(\tilde{X}_i, Y_i)_{i=1}^n$. How does the initialization scale α affect the learned neural network?

3. *(Necessity of overparametrization for global convergence) Reproduce the figures displayed at the end of the first blog post linked above.*

*Lénaïc Chizat EPFL lenaic.chizat@epfl.ch