Foundations of Deep Learning **Fall 2024**

Exercise 6: Neural Tangent Kernel

Lecturer: Aurelien Lucchi

Problem 1 (NTK for two-layer ReLU network):

Consider a two-layer neural network with the second layer fixed,

$$f(\mathbf{W}, \mathbf{x}) = \frac{1}{\sqrt{m}} \sum_{r=1}^{m} \sigma(\mathbf{w}_r^\top \mathbf{x})$$

where $\mathbf{x} \in \mathbb{R}^d$ is the input, $\mathbf{W} = (\mathbf{w}_r)_{r=1}^m \in \mathbb{R}^{m \times d}$ is a matrix containing the weight vectors $\mathbf{w}_r \in \mathbb{R}^d$ of the first layer, and σ is the ReLU activation function. Throughout this problem, only the first layer (with weight vectors $\mathbf{w}_r,\ r = 1, ..., m$) is trained. We are given a training dataset $(\mathbf{x}_i, y_i)$ where each $\mathbf{x}_i \in \mathbb{R}^d$ is a feature vector such that $\|\mathbf{x}_i\| = 1$, and $y_i \in \mathbb{R}$ is the corresponding target label. We consider the following square loss:

$$\ell(\mathbf{W}) = \frac{1}{2} \sum_{i=1}^{n} \left(f(\mathbf{W}, \mathbf{x}_i) - y_i\right)^2 .$$

We optimize over each $\mathbf{w}_r$ using gradient flow:

$$\frac{d\mathbf{w}_r(t)}{dt} = -\frac{\partial \ell(\mathbf{W}(t))}{\partial \mathbf{w}_r(t)} \in \mathbb{R}^d$$

for $r = 1, ..., m$.

a) We denote $u_i(t) = f(\mathbf{W}(t), \mathbf{x}_i)$ the prediction on input $\mathbf{x}_i$ at time t. Show that, at any time t, we have

$$\frac{\partial \ell(\mathbf{W}(t))}{\partial \mathbf{w}_r(t)} = \sum_{i=1}^{n} \left(u_i(t) - y_i\right) \frac{\partial f(\mathbf{W}(t), \mathbf{x}_i)}{\partial \mathbf{w}_r(t)} \in \mathbb{R}^d$$

b) Let $\mathbf{u}(t) = (u_1(t), \ldots, u_n(t)) \in \mathbb{R}^n$ be the prediction vector at time t. Show that, using chain rule on $\mathbf{u}(t)$, the dynamics of the predictions can be written as

$$\frac{d}{dt}\mathbf{u}(t) = \mathbf{H}(t)(\mathbf{y} - \mathbf{u}(t)),$$

where $\mathbf{H}(t)$ is an $n \times n$ matrix with (i, j)-th entry

$$\mathbf{H}_{ij}(t) = \sum_{r=1}^{m} \left\langle \frac{\partial f(\mathbf{W}(t), \mathbf{x}_i)}{\partial \mathbf{w}_r(t)}, \frac{\partial f(\mathbf{W}(t), \mathbf{x}_j)}{\partial \mathbf{w}_r(t)} \right\rangle$$

Note that the so-called Gram matrix $\mathbf{H}$ defined above is essentially the neural tangent kernel on the training data.

c) Show that, at any time t, we have the following expression of the entries in the Gram matrix $\mathbf{H}$:

$$\mathbf{H}_{ij}(t) = \frac{1}{m} \sum_{r=1}^{m} \mathbf{x}_i^\top \mathbf{x}_j \mathbb{I}\left\{\mathbf{w}_r(t)^\top \mathbf{x}_i \geq 0, \mathbf{w}_r(t)^\top \mathbf{x}_j \geq 0\right\}$$

where the indicator $\mathbb{I}\{A\}$ is 1 when the the constraint A holds; 0 otherwise.

Hint: The derivative $\sigma'(z)$ of the ReLU function is the step function $\text{step}(z) = \begin{cases} 1, & x > 0 \\ 0, & x < 0 \end{cases}$.

d) If we assume the weight vectors $\mathbf{w}_r$ are initialized as independent standard Gaussian vectors, i.e. $\mathbf{w}_r(0) \sim \mathcal{N}(0, \mathbf{I})$ for $r = 1, ..., m$, and we take the the limit of the layer width $m \to \infty$, we have the so-called neural tangent kernel (NTK) matrix $\mathbf{H}^\infty$ with entries:

$$\mathbf{H}_{ij}^\infty = \lim_{m \to \infty} \mathbf{H}_{ij}(0) = \mathbb{E}_{\mathbf{w} \sim \mathcal{N}(0, \mathbf{I})} \left[\mathbf{x}_i^\top \mathbf{x}_j \mathbb{I}\left\{\mathbf{w}^\top \mathbf{x}_i \geq 0, \mathbf{w}^\top \mathbf{x}_j \geq 0\right\}\right]$$

Show that we have the following closed-form for the entries of NTK $\mathbf{H}^\infty$:

$$\mathbf{H}_{ij}^\infty = \frac{\mathbf{x}_i^\top \mathbf{x}_j \left(\pi - \arccos(\mathbf{x}_i^\top \mathbf{x}_j)\right)}{2\pi}, \quad \forall i, j = 1, ..., n. \tag{1}$$

Problem 2 (Upper bound of classification error):

Assume we have a dataset with n data points, $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$ where each input $\mathbf{x}_i \in \mathbb{R}^d$ has norm 1 and $y_i = \beta^\top \mathbf{x}_i$ for some $\beta \in \mathbb{R}^d$. Write $\mathbf{X} = (\mathbf{x}_i)_{i=1}^n \in \mathbb{R}^{n \times d}$.

We consider the same two-layer ReLU network used in the previous problem. An upper bound on the classification error of the NTK is given by

$$\frac{\sqrt{2\mathbf{y}^\top (\mathbf{H}^\infty)^{-1}\mathbf{y} \cdot \mathrm{tr}(\mathbf{H}^\infty)}}{n},$$

where the NTK matrix $\mathbf{H}^\infty$ is defined in Eq. (1).

This complexity measure can be derived using *Rademacher complexity* bounds, which will be the subject of a later course. Our goal will be to derive a bound on this complexity measure that decreases as n increases. This will imply that the NTK can achieve zero classification error given a sufficient large number of datapoints n.

a) First, show that $\mathbf{H}^\infty$ admits the following expression of entries:

$$\mathbf{H}_{ij}^\infty = \frac{\mathbf{x}_i^\top \mathbf{x}_j}{4} + \frac{1}{2\pi} \sum_{l=0}^{\infty} \frac{(2l)!}{2^{2l}(l!)^2} \frac{(\mathbf{x}_i^\top \mathbf{x}_j)^{2l+2}}{2l+1}.$$

Hint: Use the following Taylor approximation of $\arccos(z)$:

$$\arccos(z) = \frac{\pi}{2} - \sum_{l=0}^{\infty} \frac{(2l)!}{2^{2l}(l!)^2} \frac{z^{2l+1}}{2l+1}.$$

b) Using the following facts: (You do not need to prove them)

- $4\mathbf{K}^{-1} - (\mathbf{H}^\infty)^{-1}$ is a positive semi-definite matrix; (why?)
- $\mathrm{tr}(\mathbf{H}^\infty) \le n$; (why?)
- the operator norm of the matrix $\mathbf{X}(\mathbf{X}\mathbf{X}^\top)^{-1}\mathbf{X}^\top$ is less than or equal to 1. (why?)

and prove the following upper bound on the classification error of the NTK :

$$\frac{\sqrt{2\mathbf{y}^\top (\mathbf{H}^\infty)^{-1}\mathbf{y} \cdot \mathrm{tr}(\mathbf{H}^\infty)}}{n} \le \frac{2\sqrt{2}\|\beta\|_2}{\sqrt{n}}.$$