Neural Tangent Kernel

Setup: $\mathcal{D} := \{x_i, y_i\}$, $i = 1, \dots, n$, $\quad y = \underbrace{f_\theta(x)}_{\text{MLP}} + \varepsilon$

Network outputs: $f(x; \theta(t))$

Training via gradient gr

$$\mathcal{L}(\theta) = \frac{1}{2} \sum_{i=1}^{n} \left[f(x_i; \theta(t)) - y_i \right]^2, \quad \frac{\partial \mathcal{L}}{\partial \theta} \longrightarrow \theta^{n+1} = \theta^{n} - \eta \nabla_\theta \mathcal{L}(\theta^{n}$$

Gradient flow: $$\boxed{\frac{d\theta}{dt} = - \nabla_\theta \mathcal{L}(\theta)}$$

forward Euler discretization

- Derive the evolution of $f(x; \theta(t))$:

$$\frac{d f(x; \theta(t))}{dt} = \frac{d f(x; \theta(t))}{d\theta}^{T} \cdot \frac{d\theta}{dt}$$

$$= - \frac{d f(x; \theta(t))}{d\theta}^{T} \underbrace{\nabla_\theta \mathcal{L}(\theta)}$$

$$= - \frac{d f(x; \theta(t))}{d\theta}^{T} \sum_{i=1}^{n} \left[f(x_i; \theta(t)) - y_i \right] \frac{d f(x_i; \theta(t))}{d\theta}$$

$$\Rightarrow \quad \boxed{\frac{d f(X; \theta(t))}{dt} = - K_t(X, x) \left[f(X; \theta(t)) - y \right]} \quad (1)$$

where $X := \begin{bmatrix} x_{11} & \cdots & x_{1d} \\ \vdots & & \end{bmatrix}$ $\quad y = \begin{bmatrix} y_1 \\ \vdots \end{bmatrix}$

$$\underset{n\times d}{}\begin{bmatrix} x_{n1} & \cdots & x_{nd} \end{bmatrix}, \quad \begin{bmatrix} \vdots \\ y_n \end{bmatrix}$$

$$\underbrace{\underset{n\times n}{K_t(X,X)_{ij}} = \left\langle \frac{df(x_i;\theta(t))}{d\theta}, \frac{df(x_j;\theta(t))}{d\theta} \right\rangle}_{\text{Neural Tangent Kernel}}$$

Remark #1: At the infinite width and infinitesimaly small learning rate, the NTK K_t converges to a deterministi kernel that remains constant during training. $K_t = K(0) = K$

Remark #2:

$$\boxed{f(x^*;\theta(t)) \approx K_t(x^*,X)\,K_t^{-1}(X,X)\left(I - e^{-K_t t}\right)}$$

$$\approx \underbrace{K^*(x^*,X)\,K^{*-1}(X,X)\,y}_{\text{Kernel regression}}$$

Remark #3: $\boxed{\underset{n\times n}{K^*} = Q^T \Lambda Q}$ $\begin{cases} Q: \text{orthogonal} \\ \Lambda: \text{diagonal} \end{cases}$

$$\boxed{Q^T\big(\underbrace{f(X;\theta(t)) - y}_{\text{training error}}\big) = -e^{\Lambda t} Q^T y}$$

$$\Longrightarrow \boxed{f(X;\theta(t)) - y = \sum_{i=1}^{n}\left(e^{-\lambda_i t} q_i^T y\right) q_i} \Longrightarrow \text{Spectra Bias}$$