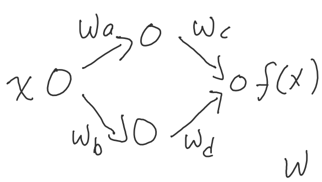
NTK Basic Understanding

Thursday, November 11, 2021 2:53 PM

Given a simple single layer netwok.



If we look at the weights during gradient descent.

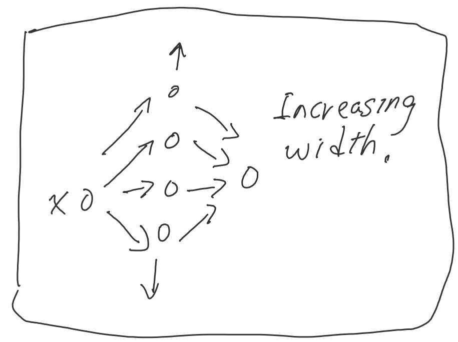
$W = [w_a, w_b, w_c, w_d]$ ← This vector would change over training.

If we let

w_0 = initial weight

w_Δ = change in weight $\Longrightarrow$ $w_0 + w_\Delta = w_f$

w_f = final weight



- It has been visually observed that as the width of the network increase w_Δ becomes smaller and smaller
- In fact, w_0 becomes approximately w_f where $w_0 \approx w_f$ as $w_\Delta \to 0$ (1)
- The NTK paper proves that statement (1) is True.
- The paper takes a step further linking it to

<u>Kernel Regression.</u>

- A quick recap of linear regression, we have the loss $\mathcal{L}$ as

$$\min_w \frac{1}{2n} \sum_i (w^T x_i - y_i)^2 = \min_w \mathcal{L}(w)$$

If we solve this via Gradient Descent.

$$w_{n+1} = w_n - \eta \nabla \mathcal{L}(w) \quad \leftarrow \eta \text{ is a small constant.}$$

$$\nabla \mathcal{L}(w) = \frac{d\mathcal{L}}{dw} \frac{1}{2n} \sum_i (w^T x_i - y_i)^2$$

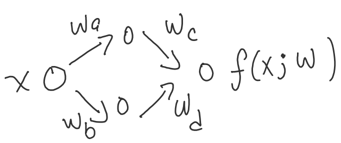
$$= \frac{1}{n} \sum_i (w^T x_i - y_i) x_i$$

- To extend linear regression to kernel regression we adjust $\nabla \mathcal{L}(w)$ to

$$\nabla \mathcal{L}(w) = \frac{1}{n} \sum_i \left(w^T \phi(x_i) - y_i \right) \phi(x_i)$$

$\phi(\cdot)$ is the feature map of a kernel.

Now let's go back to NTK



- Assume we know and fix the data X, the network becomes a function with respect to w, or $f(x;w)$ becomes $f(w)$.
- Since we know that as width $\to \infty$ $w_\Delta \to 0$. This implies that the change $f(w)$ with respect to w is small.
- This allows us to approximate $f(w)$ via its 1st order Taylor Approximation. around w_0

$$f(w) \approx f(w_0) + \nabla f(w_0)^T (w - w_0) + \underbrace{\cancel{\text{higher Terms}}}_{\text{Assume } 0}$$

$$\approx f(w_0) + \nabla f(w_0)^T w - \nabla f(w_0)^T w_0$$

this the only variable Everything else is a constant

$$\approx \nabla f(w_0)^T w + \overbrace{f(w_0) - \nabla f(w_0)^T w_0}^{\text{let this constant be } C}$$

$$\hat{f}(w) = \nabla f(w_0)^T w + C$$

Now that we have the approximate network let's perform regression with it.

$$\min_w \mathcal{L}(w) = \min_w \frac{1}{2n} \sum_i \left[\hat{f}(w; x_i) - y_i \right]^2$$

$$\mathcal{L}(w) = \frac{1}{2n} \sum_i \left[\nabla f(w_0)^T w + c - y_i \right]^2$$
$$= \frac{1}{2n} \sum_i \left[w^T \nabla f(w_0) + c - y_i \right]^2$$

Similarly, if we want to perform GD, we must find $\nabla \mathcal{L}(w)$

$$\nabla \mathcal{L}(w) = \frac{1}{n} \sum_i \left[w^T \nabla f(w_0) + c - y_i \right] \nabla f(w_0)$$

Note that $\nabla f(w_0)$ is still a function with x so it is also $\nabla f(w_0 ; x)$, since w_0 is now fixed we will rename the function $\nabla f(w_0 ; x) := \phi(x)$ resulting in

$$\nabla \mathcal{L}(w ; x) = \frac{1}{n} \sum_i \left[w^T \phi(x) + c - y_i \right] \phi(x)$$

The constant can be directly add into $w^T \phi(x)$ by

$$\begin{bmatrix} 1 & w^T \end{bmatrix} \begin{bmatrix} c \\ \phi(x) \end{bmatrix} = \text{the new } w^T \phi(x)$$

So the constant can be ignored, resulting

$$\nabla \mathcal{L}(w ; x) = \frac{1}{n} \sum_i \left[w^T \phi(x) - y_i \right] \phi(x)$$

Notice how this is identical to <u>kernel regression</u>

$$\nabla \mathcal{L}(x) = \frac{1}{n} \sum_i \left[w^T \phi(x) - y_i \right] \phi(x)$$

The kernel for NTK is

$$\phi(x) = \nabla_w f(x) \longrightarrow$$

Therefore the NTK kernel

$$K(x_i, x_j) = \langle \nabla_w f(x_i), \nabla_w f(x_j) \rangle$$