ECE/CS 559 Lecture 8 Th 9/19

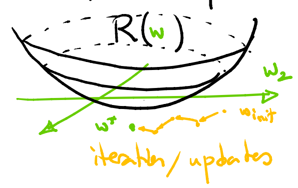
Last time: Linear Regression with Squared Loss

- Data: $\{(x,y), \dots\}$ $x \in \mathbb{R}^n$ $y \in \mathbb{R}^m$
- Predictor: $f(x;W) = Wx$ $W \in \mathbb{R}^{m \times n}$
- Goal: $\underset{W}{\text{minimize}}\ R(W) = \sum_{(x,y) \in \text{Data}} \underbrace{\|y - Wx\|^2}_{\ell(y, Wx) \equiv \ell((x,y); W)}$

$$m\left[\hat{W}\right] \cdot \underbrace{\left[\cdots X \cdots\right]}_{|\text{Data}|} = m\underbrace{\left[\cdots Y \cdots\right]}_{|\text{Data}|}$$

$$W = Y \boxed{X^T(XX^T)^{-1}}$$ Pseudo inverse

it's a quadratic equation



① **Gradient Descent**

Goal: $\underset{W}{\text{minimize}}\ R(W) = \sum_{(x,y) \in \text{Data}} \overbrace{\|y - f(x;W)\|^2}^{\ell(y, f(x;W))}$

Idea: Set the derivative to zero! → reduce step-by-step.

Gradient: $\nabla_W R = \left[\frac{\partial R}{\partial w_1}, \dots, \frac{\partial R}{\partial w_k}\right]$

Each step: Change W by ΔW : $\boxed{W' \leftarrow W + \Delta W}$

$$R(W') = R(W + \Delta W) = \underbrace{R(W) + \nabla R(W)^T \Delta W}_{\text{Taylor expansion}} + \underbrace{O(\|\Delta W\|^2)}_{\text{ignore}}$$

How should we choose ΔW to get the best reduction?

- To $\underset{\Delta W}{\text{minimize}}\ \nabla R^T \Delta W$, let $\Delta W \underset{\text{proportional}}{\propto} -\nabla_W R$: $\boxed{W' \leftarrow W - \eta \nabla R(W)}$

1-D Example: $R(w) = w^4$ $\frac{dR}{dw} = 4w^3$ $\eta = \frac{1}{8}$

$w(0) = 1$

$$w(1) \leftarrow w(0) - \nabla R(w(0)) \cdot \eta$$
$$= 1 - 4 \cdot 1^3 \cdot \frac{1}{8} = \frac{1}{2}$$
$$w(2) \leftarrow w(1) - \nabla R(w(1)) \cdot \eta$$
$$= \frac{1}{2} - 4\left(\frac{1}{2}\right)^3 \cdot \frac{1}{8} = \frac{7}{16}$$

- What's happening?

$$w(t) \leftarrow w(t-1) - 4\, w(t-1)^3 \eta = [1 - \underbrace{4\eta\, w(t-1)^2}]\, w(t-1)$$

- If η is small enough, always diminishes. (<1) Since bounded from below (by 0) will converge.
- But will it converge to 0? Yes, think about why.
- What if η is too big? Try $\eta = \frac{1}{2}$ with $w(0) = 1$.

② **Widrow-Hoff LMS Algorithm**

- Let's apply gradient descent to linear regression. Recall:
- $\nabla R(W) = -2 \sum_{(x,y) \in \text{Data}} (y - Wx)x^T$ same shape as W

$$W \longleftarrow W - \eta \nabla R(W) = W + \underset{\text{absorb}}{2}\eta \sum_{(x,y)} (y - Wx)x^T$$

- When $y \in \mathbb{R}^1$ then $W = w^T$ and $y - w^Tx \in \mathbb{R}$

The updates become: $w \leftarrow w + \eta \sum_{(x,y)} (y - w^Tx)\, x$

- We could do this 1 data point at a time:

For each $(x,y) \in$ Data: $w \leftarrow w + \eta(y - w^T x)\, x$

Similar to the Perceptron learning algorithm!

③ **Delta Rule**

- Squared loss for single neuron with differentiable $\varphi(\cdot)$:

$$R(w) = \sum_{(x,y)} (y - \varphi(w^Tx))^2$$

$$\nabla R(w) = -2 \sum_{(x,y)} (y - \varphi(w^Tx))\, \varphi'(w^Tx)\, x^T$$ chain rule

$$w \leftarrow w + 2\eta \sum_{(x,y)} (y - \varphi(w^Tx))\, \varphi'(w^Tx)\, x$$

- 1 data point at a time:

$$w \leftarrow w + \eta\,(y - \varphi(w^Tx))\, \varphi'(w^Tx)\, x$$

Example $\varphi(\cdot)$:

- Sigmoid: $\varphi(v) = \frac{1}{1 + e^{-av}}$ Since $1 - \varphi(v) = \frac{e^{-av}}{1 + e^{-av}}$

$$\varphi'(v) = \frac{0 - (-a e^{-av})}{(1 + e^{-av})^2} = \frac{a e^{-av}}{(1 + e^{-av})^2} = a\,\varphi(v)\,(1 - \varphi(v))$$

- ReLU: $\varphi(v) = \begin{cases} v & v \geq 0 \\ 0 & \text{otherwise} \end{cases}$

$$\varphi'(v) = \begin{cases} 1 & v \geq 0 \\ 0 & \text{otherwise} \end{cases} = \text{step}(v)$$