ECE/CS 559 Lecture 7 9/17

Last time: Perceptron Learning Example — Supervised Learning

- Data: $\{(x,y),\dots\}$ $x \in \mathcal{X}$ (feature space), $y \in \mathcal{Y}$ (label space)
- Task: Predict/imitate y using only x.
- How: Use a predictor — neural network $y_w(x) \equiv f(x;w)$
- Loss: how far y and $f(x;w)$ are: $\ell(y,f) = \mathbb{1}\{y \neq f\}$ or $\|y-f\|^2$
- Risk: (empirical) $R(w) = \frac{1}{|\text{Data}|} \sum_{(x,y)\in\text{Data}} \ell(y, f(x,w))$
- Goal: Minimize (empirical) risk, make few mistakes/errors / stay close to y.

- Types of algorithms:
  - Online: - See (x,y), update w - See (x',y'), update w - See (x'',y''), update w ⋮
  - Batch: - Get Data $= \{(x,y),\dots\}$ • Epoch 1: use Data, update w • Epoch 2: use Data, update w ⋮

① Linear Regression with Squared Loss

- Data: $\{(x,y),\dots\}$ $x \in \mathbb{R}^n$ $y \in \mathbb{R}^m$
- Predictor: $f(x;w) = Wx$ $W \in \mathbb{R}^{m\times n}$ (this is a neuron with $g(v) = v$, identity activation)
- Goal: $\min_w R(w) = \sum_{(x,y)\in\text{Data}} \underbrace{\|y - Wx\|^2}_{\ell(y,Wx)}$

Analytic Solution: Optimality conditions, unconstrained $\Rightarrow \nabla_w R = 0$ (gradient)

- Single data point $\ell = \sum_{k=1}^{m} \left(y_k - \sum_{j=1}^{n} w_{kj} x_j\right)^2$

$$\frac{d\ell}{dw_{kj}} = -2x_j\left(y_k - \sum_{j'=1}^{n} w_{kj'}x_{j'}\right) = -2y_k x_j + \sum_{j'=1}^{n} w_{kj'}(x_j x_{j'})$$

$$\nabla_w \ell = -2yx^T + Wxx^T$$

- All data points: $R(w) = \sum_{(x,y)\in\text{Data}} \ell(y,Wx) \Rightarrow \nabla_w R = \sum_{(x,y)} \nabla_w \ell$

$$\nabla_w R = \sum_{(x,y)\in\text{Data}} -2yx^T + 2W \sum_{x\in\text{Data}} xx^T$$

$$= -2\underbrace{\begin{bmatrix} 1 & 1 & \cdots & 1 \\ y_1 & y_2 & \cdots & y_{|\text{Data}|} \end{bmatrix}}_{|\text{Data}|} \begin{bmatrix} -x_1- \\ -x_2- \\ \vdots \\ -x_{|\text{Data}|}- \end{bmatrix} + 2W \begin{bmatrix} 1 & 1 & \cdots & 1 \\ x_1 & x_2 & \cdots & x_{|\text{Data}|} \end{bmatrix} \begin{bmatrix} -x_1- \\ -x_2- \\ \vdots \\ -x_{|\text{Data}|}- \end{bmatrix}$$

$$= -2YX^T + 2WXX^T$$

- $\nabla_w R = 0 \Rightarrow -2YX^T + 2WXX^T = 0$

$$\Rightarrow W = Y\underbrace{X^T(XX^T)^{-1}}_{\text{Moore-Penrose Pseudo inverse}}$$

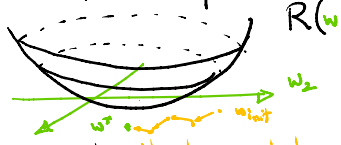
Compare to: $Wx \simeq Y$ not invertible if $|\text{Data}| > n$

$$\underbrace{\begin{bmatrix} w \end{bmatrix}}_{m\times n} \underbrace{\begin{bmatrix} x_1 & \cdots & x_{|\text{Data}|} \end{bmatrix}}_{n\times|\text{Data}|} \simeq \underbrace{\begin{bmatrix} y_1 & \cdots & y_{|\text{Data}|} \end{bmatrix}}_{m\times|\text{Data}|}$$

but it's "like" we inverted it above.

- Where are the updates? One single batch update!

Intuition: it's a quadratic equation



- Other shapes → solve by iterative updates

② Gradient Descent

Goal: $\min_w R(w) = \sum_{(x,y)\in\text{Data}} \underbrace{\|y - f(x;w)\|^2}_{\ell(y,f(x;w))}$

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: $w' \leftarrow w + \Delta w$.

$$R(w') = R(w+\Delta w) = R(w) + \nabla_w R(w)^T \Delta w + O(\|\Delta w\|^2)$$

(Taylor expansion; ignore)

How should we choose Δw to get the best reduction?

- To minimize $\nabla_w R^T \Delta w$ over Δw, let $\Delta w \propto -\nabla_w R$: $w' \leftarrow w - \eta \nabla R(w)$ (proportional)

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

$w(0) = 1$

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

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

- What's happening?

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

- 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 $w(0) = 1$.