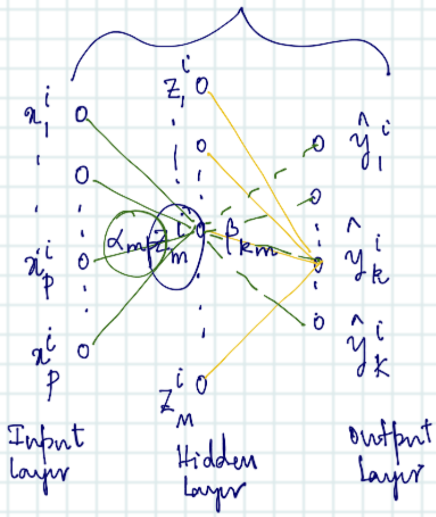
12/9/23

Deep Learning

- Recap ✓
- Loss function ✓
- Gradient descent ✓
- Back propagation
- Recap:



Multi layer perceptron

$$D = \{(\underset{\text{data point}}{\underline{x}^1}, \underset{\text{ground truth label}}{\underline{y}^1}), \ldots, (\underline{x}^n, \underline{y}^n)\}$$

$$\underline{x}^i = [1 \; x_1^i \; \ldots \; x_p^i]^T$$

$$\underline{z}^i = [1 \; z_1^i \; \ldots \; z_M^i]^T$$

$$\underline{\hat{y}}^i = [\hat{y}_1^i \; \ldots \; \hat{y}_K]^T$$

$$\underline{\alpha}_m = [\underset{\text{bias}}{\alpha_{m0}}, \alpha_{m1}, \ldots, \alpha_{mp}]^T$$

$$\underline{\beta}_k = [\beta_{k0}, \beta_{k1}, \ldots, \beta_{kM}]^T$$

$$\theta = \{\underline{\alpha}_1, \ldots \underline{\alpha}_M, \underline{\beta}_1 \ldots \underline{\beta}_K\}$$

$$z_m^i = \sigma(\langle \underline{\alpha}_m, \underline{x}^i \rangle)$$

$$\hat{y}_k^i = \text{softmax}(\langle \underline{\beta}_k, \underline{z}^i \rangle) = g_k(\langle \underline{\beta}_k, \underline{z}^i \rangle)$$

- Loss function: $\underset{\text{Risk}}{R(\theta)}$ or $\underset{\text{loss}}{\mathcal{L}(\theta)} = \sum_{i=1}^{n} d(\underline{y}^i, \underline{\hat{y}}^i)$

$$= \sum_{i=1}^{n} \sum_{k=1}^{K} d(y_k^i, \hat{y}_k^i)$$

For simplicity, let $d(y_k^i, \hat{y}_k^i) = (y_k^i - \hat{y}_k^i)^2$

$$\therefore \quad R(\theta) = \sum_{i=1}^{n} \sum_{k=1}^{K} (y_k^i - \hat{y}_k^i)^2$$

$$R(\theta) = \sum_{i=1}^{n} \sum_{k=1}^{K} (y_k^i - \underbrace{f_k(\underline{x}^i; \theta)}_{\text{machine}})^2$$

- Find $\underline{\theta}$ that minimizes the loss or risk $R(\theta)$; $\theta^* = \underset{\theta \in \Theta}{\arg\min}\ R(\theta)$
- <u>Exercise:</u> Show that $f_k(\underline{x}^i; \theta)$ is a non-convex function of θ.
- We will rely on gradient based techniques for find a local optimum.

$$\theta^{(r)} = \theta^{(r-1)} - \eta^{(r)} \nabla_{\theta^{(r-1)}} R(\theta)$$

$$\begin{bmatrix} \underline{\alpha}_1 \\ \underline{\alpha}_2 \\ \vdots \\ \underline{\alpha}_M \\ \underline{\beta}_1 \\ \underline{\beta}_2 \\ \vdots \\ \underline{\beta}_K \end{bmatrix}^{(r)} = \begin{bmatrix} \underline{\alpha}_1 \\ \vdots \\ \underline{\alpha}_M \\ \underline{\beta}_1 \\ \vdots \\ \underline{\beta}_K \end{bmatrix}^{(r-1)} - \eta^{(r)} \begin{bmatrix} \frac{\partial R(\theta)}{\partial \alpha_{11}} \\ \vdots \\ \frac{\partial R(\theta)}{\partial \alpha_{mp}} \\ \vdots \\ \frac{\partial R(\theta)}{\partial \beta_{km}} \\ \frac{\partial R(\theta)}{\partial \beta_{KM}} \end{bmatrix}^{(r-1)}$$

- Find $\dfrac{\partial R(\theta)}{\partial \beta_{km}} = \dfrac{\partial}{\partial \beta_{km}} \sum_{i=1}^{n} \sum_{j=1}^{K} (y_j^i - \hat{y}_j^i)^2$

$$\frac{\partial R(\theta)}{\partial \beta_{km}} = -\sum_{i=1}^{n} 2(y_k^i - \hat{y}_k^i) \cdot g_k'(\langle \underline{\beta}_k, \underline{z}^i \rangle) \cdot z_m^i$$

↳ what happens when g_k is softmax?

$$\frac{\partial R(\theta)}{\partial \alpha_{mp}} = \frac{\partial}{\partial \alpha_{mp}} \sum_{i=1}^{n} \sum_{k=1}^{K} (y_k^i - \hat{y}_k^i)^2$$

$$= \underbrace{-2 \sum_{i=1}^{n} \sum_{k=1}^{K} (y_k^i - \hat{y}_k^i) \cdot g_k'(\langle \underline{\beta}_k, \underline{z}^i \rangle) \cdot \beta_{km}\, \sigma'(\langle \underline{\alpha}_m, \underline{x}^i \rangle) \cdot x_p^i}_{\text{Heart of back prop}}$$