Introduction to Machine Learning Cheatsheet

Functions

softmax: $\delta(\mathbf{z})_j = \frac{e^{z_j}}{\sum_{k=1}^K e^{z_k}}$

$$\mathbf{ReLU}(x) = \max(x, 0)$$

sigmoid: $\theta(z) = \frac{1}{1+e^{-z}}$ $\qquad \tanh(s) = \frac{e^s - e^{-s}}{e^s + e^{-s}}$

$$\mathbf{CE}(t,s) = -\frac{1}{N}\sum_{n=1}^{N} t \log s$$

Derivatives

$$(\log x)' = 1/x$$

$$(f \cdot g)' = f'g + fg'$$

$$\left(\frac{f}{g}\right)' = \frac{f'g - fg'}{g^2} \qquad \text{ReLU}(x)' = \begin{cases} 1, x > 0 \\ 0, x < 0 \end{cases}$$

$$(e^x)' = e^x \qquad \tanh'(s) = 1 - \tanh^2(s)$$

$$\text{chain rule: } \frac{d}{dx} f(g(x)) = \frac{df(g)}{dg} \cdot \frac{dg(x)}{dx}$$

$$\text{sigmoid: } \frac{d\theta(x)}{dx} = \theta(x)(1-\theta(x))$$

$$\text{CE: } \frac{d}{dx}\mathbf{CE}(t,s) = -t/s$$

$$\text{softmax: } \frac{\partial \delta(\mathbf{z})}{\partial z_j} = \begin{cases} z_i(1-z_j) & , i = j \\ -z_i z_j & , i \neq j \end{cases}$$

$$\text{sigmoid-CE: } \frac{\partial L}{\partial w_j} = -y_j(1-y)$$

$$\text{softmax-CE: } \frac{\partial L}{\partial w_j} = \hat{y}_j - y_j$$

Models

Perceptron

Params: $\mathbf{w} = (w_0, \cdots, w_d) \in \mathbb{R}^{d+1}$
Hypothesis: $\hat{y} = h_w(\mathbf{x}) = \mathbf{sign}(\mathbf{w}^\intercal x)$
Loss: $E_{in}(\mathbf{w}) = \frac{1}{N}\sum_{n=1}^N \mathbb{I}\{y_n \neq \hat{y}_n\}$

Linear Regression

Params: $\mathbf{w} = (w_0, \cdots, w_n) \in \mathbb{R}^{d+1}$
Hypothesis: $\hat{y} = \mathbf{X}^\intercal \mathbf{w}$
Loss(MSE):
$E_{in}(\mathbf{w}) = \frac{1}{N}\|\mathbf{y} - \hat{\mathbf{y}}\|^2 = \frac{1}{N}\sum_{n=1}^N (y_n - \hat{y}_n)^2$
Optimize: $\mathbf{w}^* = \arg\min_{\mathbf{w}\in\mathbb{R}^{d+1}} E_{in}(\mathbf{w})$
Solution: $\mathbf{w}_{LS} = (\mathbf{X}^\intercal\mathbf{X})^{-1}\mathbf{X}^\intercal\mathbf{y}$

Linear Regression w/ Regularization

Optimize: $\mathbf{w}^* = \arg\min_{\mathbf{w}\in\mathbb{R}^{d+1}} \{E_{in}(\mathbf{w}) + \lambda\|\mathbf{w}\|^2\}$
Solution: $\mathbf{w}_{LS} = (\mathbf{X}^\intercal\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^\intercal\mathbf{y}$

Logistic Regression (binary)

$\hat{P}_{\mathbf{w}}(y|\mathbf{x}) = \frac{e^{y\mathbf{w}^\intercal\mathbf{x}}}{1+e^{y\mathbf{w}^\intercal\mathbf{x}}}$
Loss: "logloss" $\frac{1}{N}\sum_{n=1}^N \left(-\log \hat{P}_{\mathbf{w}}(y|\mathbf{x})\right)$
$= \frac{1}{N}\sum_{n=1}^N \log(1 + e^{-y_n \mathbf{w}^\intercal \mathbf{x}_n})$

Algorithms

PLA

1. Check if $E_{in}(\mathbf{w}) = 0$, STOP if yes
2. Use *mis-classified* $(\mathbf{x}_n, y_n)$, update $\mathbf{w} \leftarrow \mathbf{w} + y_n\mathbf{x}_n$.

Pocket

Initialize $\mathbf{w}$
for $t = 0, 1, \cdots, T-1$ **do**

1. Run PLA for one update to obtain $\mathbf{w}(t+1)$
2. Evaulate $E_{in}(\mathbf{w}(t+1))$
3. **if** $E_{in}(\mathbf{w}(t+1)) < E_{in}(\mathbf{w})$ **then** $\mathbf{w} \leftarrow \mathbf{w}(t+1)$

SGD

1. $\mathbf{g}_t = \nabla e_n(\mathbf{w}_t)$
2. $\mathbf{w}_{t+1} = \mathbf{w}_t - \epsilon\mathbf{g}_t$

SGD w/ Momentum

1. $\mathbf{g}_t = \nabla e_n(\mathbf{w}_t)$
2. $\mathbf{v}_t = -\epsilon_t\mathbf{g}_t + \mu\mathbf{v}_{t-1}$ ($\mathbf{v}_0 = 0, \mu \approx 0.9$)
3. $\mathbf{w}_{t+1} = \mathbf{w}_t + \mathbf{v}_t$

Neskrov Momentum

1. $\mathbf{v}_t = \mu\mathbf{v}_{t_1} - \epsilon_t\nabla e_n(\mathbf{w}_t + \mu\mathbf{v}_{t-1})$
2. $\mathbf{w}_{t+1} = \mathbf{w}_t + \mathbf{v}_t = \mathbf{w}_t + \mu\mathbf{v}_{t-1} - \epsilon_t\nabla e_n(\cdot)$

NN Complexity

\# of compulations $= \sum_l (d^{(l-1)+1}) \cdot d_{(l)} + \sum_l d^{(l)}$
$=$ \# of edges + \# of nodes$= Q + V \approx Q$.

Back Propagation

To compute $\frac{\partial e(\Omega)}{\partial w_{i,j}^{(l)}}$:

1. $\frac{\partial e(\Omega)}{\partial w_{i,j}^{(l)}} = \frac{\partial e(\Omega)}{\partial s_j^{(l)}} \cdot \frac{\partial s_j^{(l)}}{\partial w_{i,j}^{(l)}} = \delta_j \cdot x_i^{(l-1)}$
2. δ_j: Backward message at node j in layer l
3. Intermediate: $\delta_j = \frac{\partial e(\Omega)}{\partial x_i^{(l)}} \cdot \frac{\partial x_i^{(l)}}{\partial s_i^{(l)}} = \frac{\partial e(\Omega)}{\partial x_i^{(l)}} \cdot \theta'(s_i^{(l)})$
4. $\frac{\partial e(\Omega)}{\partial x_i^{(l)}} = \sum_{j=1}^{d^{(l+1)}} \frac{\partial e(\Omega)}{\partial s_j^{(l+1)}} \cdot \frac{\partial s_j^{(l+1)}}{\partial x_i^{(l)}} = \sum \delta_j^{(l+1)} \cdot w_{i,j}^{(l+1)}$.
5. $\theta_i^{(l)} = \left(\sum \theta_j^{(l+1)} \cdot w_{i,j}^{(l+1)}\right) \cdot \theta'(s_i^{(l)})$.