Numeric Minimization of Loss: ①

We have a neural network model with parameters b, w.

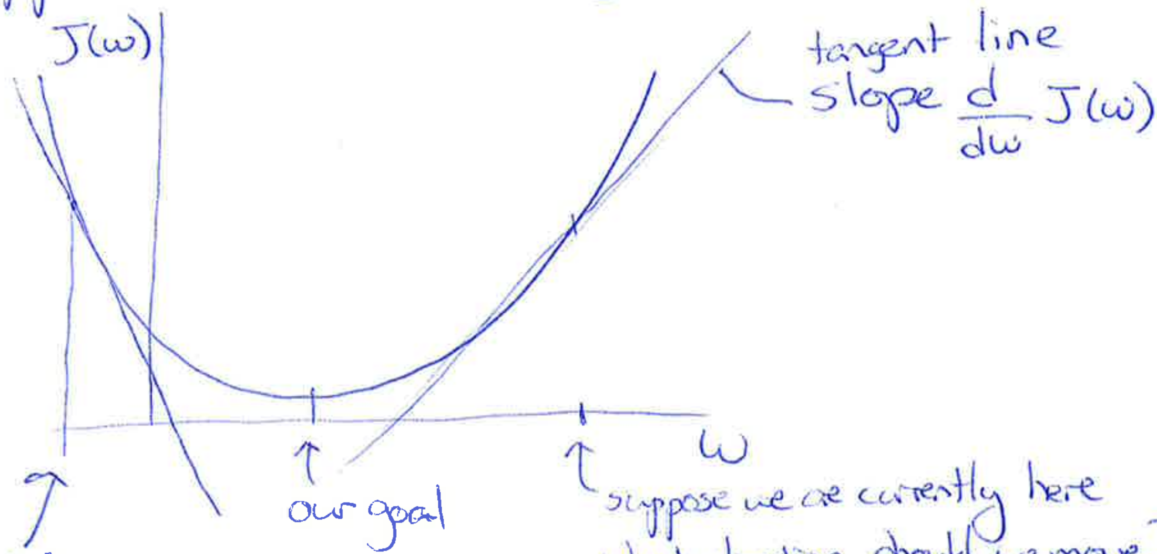
We pick b and w by <u>maximizing</u> <u>likelihood</u>: $\mathcal{L}(b,w)$

- want parameter values for which the probability of the observed data is largest

Equivalently, <u>minimize</u> the <u>negative</u> <u>log-likelihood</u> $J(b,w)$

- log-likelihood is generally more consistently sloped / doesn't have flat regions
- minimizing the negative is just custom / historical practice - not critical.

Suppose no b and only one w:



If we start here:

- slope < 0
- J(w) increases if w increases, decreases if w decreases
- should move in opposite direction as $\frac{d}{dw} J(w)$

suppose we are currently here
what direction should we move?

- slope > 0
- ~~increase~~ if J(w) increases if w increases, decreases if w decreases
- should move in opposite direction as $\frac{d}{dw} J(w)$

"Gradient" Descent with 1 parameter w

Inputs:

- initial value of w
- learning rate α
- number of iterations to run n_iter

Outputs:

- estimate of w

Algorithm:

for i in 0, 1, ..., n_iter - 1:

$$w = w - \alpha \cdot \frac{d}{dw} J(w)$$

↑ how much should we move in each step?

What if we have multiple parameters?

- Each parameter moves ~~according to~~ in direction opposite to the partial derivative of J w.r.t. that parameter.

$$\begin{cases} w = w - \alpha \cdot \dfrac{\partial}{\partial w} J(b, w) \\ b = b - \alpha \cdot \dfrac{\partial}{\partial b} J(b, w) \end{cases}$$

$$\begin{bmatrix} w \\ b \end{bmatrix} = \begin{bmatrix} w \\ b \end{bmatrix} - \alpha \cdot \underbrace{\nabla J(b, w)}_{\begin{bmatrix} \frac{\partial}{\partial w} J(b,w) \\ \frac{\partial}{\partial w} J(b,w) \end{bmatrix}}$$

How to compute the gradient?

Note: forward propagation is basically function composition:

$$\begin{cases} z^{[1]} = b^{[1]} + w^{[1]T} a^{[0]} \\ a^{[1]} = g^{[1]}(z^{[1]}) \end{cases}$$

tanh

$$\begin{cases} z^{[2]} = b^{[2]} + w^{[2]T} a^{[1]} \\ a^{[2]} = g^{[2]}(z^{[2]}) \end{cases}$$

$$\begin{cases} z^{[3]} = b^{[3]} + w^{[3]T} a^{[2]} \\ a^{[3]} = g^{[3]}(z^{[3]}) \end{cases}$$

sigmoid

Put these things together:

$$a^{[3]} = g^{[3]}\left(b^{[3]} + w^{[3]T} \cdot g^{[2]}\left(b^{[2]} + w^{[2]T} g^{[1]}\left(b^{[1]} + w^{[1]T} a^{[0]}\right)\right)\right)$$

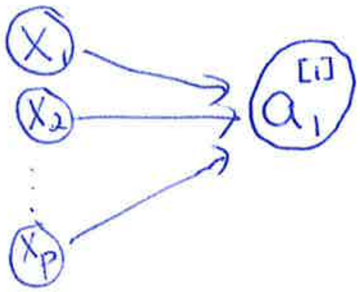
basically a composition of nonlinear activation functions

$\implies$ we need to use the chain rule

The backpropagation algorithm calculates the gradient of $J(b, w)$ by repeated application of the chain rule.

Gradient calculation for logistic regression:

p inputs, 1 layer with sigmoid activation:



All the steps to calculate loss for one observation:

$$\cancel{z_1^{(i)[1]} = b_1^{[1]} + (w_1^{[1]})^T x^{(i)} \rightarrow a_1^{(i)[1]} = \sigma(z_1^{(i)[1]}) \rightarrow J^{(i)}(w,b)}$$

$$z_1^{(i)[1]} = b_1^{[1]} + (w_1^{[1]})^T x^{(i)} \rightarrow a_1^{(i)[1]} = \sigma(z_1^{(i)[1]}) \rightarrow J^{(i)}(w,b) = y^{(i)} \cdot \log\{a_1^{(i)[1]}\} + (1-y^{(i)})\log\{1-a_1^{(i)[1]}\}$$

All parameters: $b_1^{[1]}$, $w_1^{[1]} = \begin{bmatrix} w_1^{[1]} \\ \vdots \\ w_p^{[1]} \end{bmatrix}$

For this observation, partial derivatives are step at a time:

For w_1: $$\frac{\partial J^{(i)}}{\partial w_{11}^{[1]}} = \frac{\partial J^{(i)}}{\partial a_1^{(i)[1]}} \cdot \frac{\partial a_1^{(i)[1]}}{\partial z_1^{(i)[1]}} \cdot \frac{\partial z_1^{(i)[1]}}{\partial w_{11}^{[1]}}$$

For w_2: $$\frac{\partial J^{(i)}}{\partial w_{12}^{[1]}} = \underbrace{\frac{\partial J^{(i)}}{\partial a_1^{(i)[1]}} \cdot \frac{\partial a_1^{(i)[1]}}{\partial z_1^{(i)[1]}}} \cdot \frac{\partial z_1^{(i)[1]}}{\partial w_{12}^{[1]}}$$

Note: this appears in all these expressions $\left(\text{it is } \frac{\partial J^{(i)}}{\partial z_1^{(i)[1]}}\right)$

For b: $$\frac{\partial J^{(i)}}{\partial w_{12}^{[1]}} = \underbrace{\frac{\partial J^{(i)}}{\partial a_1^{(i)[1]}} \cdot \frac{\partial a_1^{(i)[1]}}{\partial z_1^{(i)[1]}}} \cdot \frac{\partial z_1^{(i)[1]}}{\partial b_1^{[1]}}$$

Key idea of backpropagation: save "intermediate" partial derivatives that we will want to reuse, to reduce extra calculations

For logistic regression,

$$J^{(i)}(b,w) = -\left\{ y^{(i)} \log\left(a_1^{(i)[1]}\right) + (1-y^{(i)}) \log\left(1-a_1^{(i)[1]}\right) \right\}$$

$$= -\left\{ y^{(i)} \log\left[\frac{e^z}{1+e^z}\right] + (1-y^{(i)}) \cdot \log\left[\frac{1}{1+e^z}\right] \right\}$$

all z's are really $z_1^{(i)[1]}$

$$= -\left\{ y^{(i)} \cdot \left[\log(e^z) - \log(1+e^z)\right] - (1-y^{(i)}) \cdot \log(1+e^z) \right\}$$

$$= -\left\{ y^{(i)} \cdot z - \cancel{y^{(i)} \log(1+e^z)} - \log(1+e^z) + \cancel{y^{(i)} \log(1+e^z)} \right\}$$

$$= -\left\{ y^{(i)} \cdot z - \log(1+e^z) \right\}$$

$$\frac{d}{dz} J^{(i)}(b,w) = -\left\{ y^{(i)} - \frac{1}{1+e^z} \cdot e^z \right\}$$

$$= -\left\{ y^{(i)} - \frac{e^z}{1+e^z} \right\} = -\left\{ y^{(i)} - a_1^{(i)[1]} \right\} = a - y$$

We also need $\dfrac{\partial z_1^{(i)[1]}}{\partial b_1^{[1]}}$ and $\dfrac{\partial z_1^{(i)[1]}}{\partial w_{11}^{[1]}} \cdots \dfrac{\partial z_1^{(i)[1]}}{\partial w_{1p}^{[1]}}$

$$z_1^{(i)[1]} = b_1^{[1]} + w_1^{[1]T} x^{(i)} = b_1^{[1]} + w_{11}^{[1]} x_1^{(i)} + \cdots + w_{1p}^{[1]} x_p^{(i)}$$

so $\dfrac{\partial z_1^{(i)[1]}}{\partial b_1^{[1]}} = 1$, $\nabla_w z_1^{(i)[1]} = \begin{bmatrix} x_1^{(i)} \\ x_2^{(i)} \\ \vdots \\ x_p^{(i)} \end{bmatrix} = x^{(i)}$

Arrange everything with observations next to eachother!

$$y = [y^{(1)} \cdots y^{(m)}] \qquad a_1^{[1]} = [a_1^{(1)[1]} \cdots a_1^{(m)[1]}]$$

$$X = [x^{(1)} \cdots x^{(m)}]$$

$$dJdz = y - a_1^{[1]}$$ ← specific to sigmoid activation

$$= [y^{(1)} - a_1^{(1)[1]} \cdots y^{(m)} - a_1^{(m)[1]}]$$

dJdb = np.~~mean~~ sum(dJdz)

Goal:

$$dJdw = \begin{bmatrix} \frac{\partial J^{(1)}}{\partial z^{(1)}} \cdot \frac{\partial z^{(1)}}{\partial w_1} + \cdots + \frac{\partial J^{(m)}}{\partial z^{(m)}} \frac{\partial z^{(m)}}{\partial w_1} \\ \vdots \\ \frac{\partial J^{(1)}}{\partial z^{(1)}} \cdot \frac{\partial z^{(1)}}{\partial w_p} + \cdots + \frac{\partial J^{(m)}}{\partial z^{(m)}} \cdot \frac{\partial z^{(m)}}{\partial w_p} \end{bmatrix}$$

$$= \begin{bmatrix} \frac{\partial z^{(1)}}{\partial w_1} & \cdots & \frac{\partial z^{(m)}}{\partial w_1} \\ \vdots & \ddots & \vdots \\ \frac{\partial z^{(1)}}{\partial w_p} & \cdots & \frac{\partial z^{(m)}}{\partial w_p} \end{bmatrix} \begin{bmatrix} \frac{\partial J^{(1)}}{\partial z^{(1)}} \\ \vdots \\ \frac{\partial J^{(m)}}{\partial z^{(m)}} \end{bmatrix}$$

$$= X \begin{bmatrix} \frac{\partial J^{(1)}}{\partial z^{(1)}} \\ \vdots \\ \frac{\partial J^{(m)}}{\partial z^{(m)}} \end{bmatrix}$$

= np.dot(X, dJdz.T)