Training a feedforward neural net

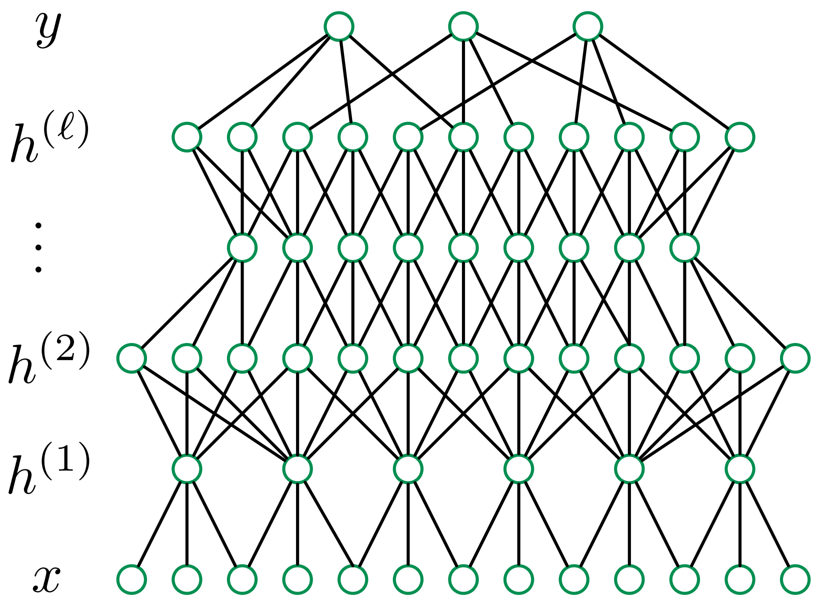
Sanjoy Dasgupta

University of California, San Diego

Topics we'll cover

1. The loss function
2. Back-propagation
3. Early stopping and dropout

Feedforward nets



The loss function

Classification problem with k labels.

- Parameters of entire net: W
- For any input x, net computes probabilities of labels:

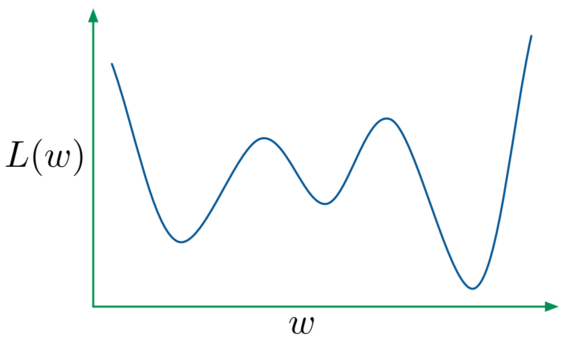
$$\Pr_W(\text{label} = j|x)$$

- Given data set $(x^{(1)}, y^{(1)}), \ldots, (x^{(n)}, y^{(n)})$, loss function:

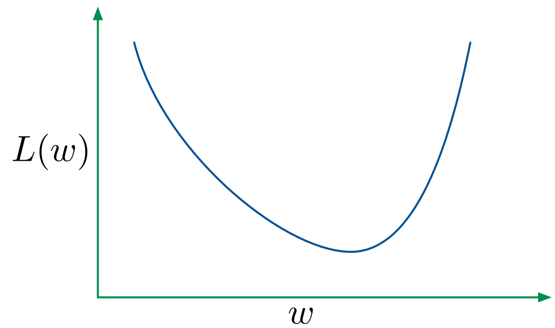
$$L(W) = -\sum_{i=1}^{n} \ln \Pr_W(y^{(i)}|x^{(i)})$$

(sometimes called **cross-entropy**).

Nature of the loss function



versus



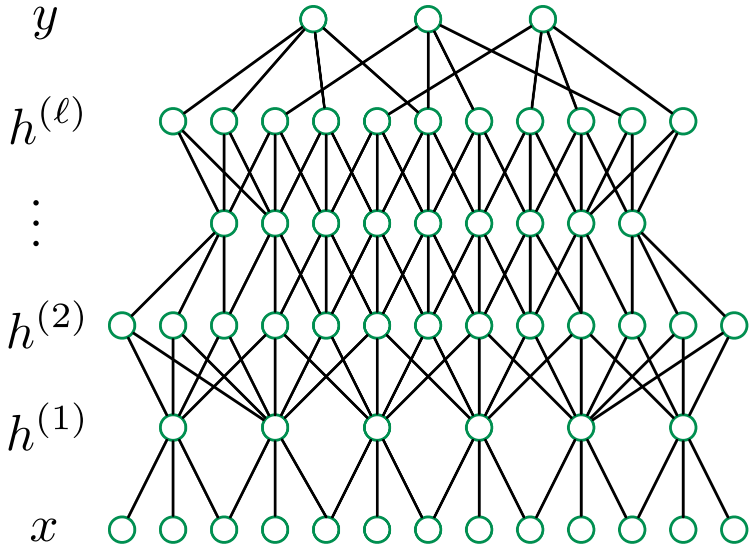
Variants of gradient descent

Initialize W and then repeatedly update.

1. Gradient descent
   Each update involves the entire training set.
2. Stochastic gradient descent
   Each update involves a single data point.
3. Mini-batch stochastic gradient descent
   Each update involves a modest, fixed number of data points.

Derivative of the loss function

Update for a specific parameter: derivative of loss function wrt that parameter.

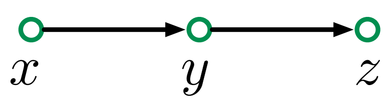


Chain rule

1. Suppose $h(x) = g(f(x))$, where $x \in \mathbb{R}$ and $f, g : \mathbb{R} \to \mathbb{R}$.

   Then: $h'(x) = g'(f(x))\, f'(x)$

2. Suppose z is a function of y, which is a function of x.

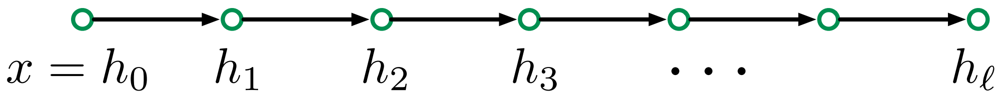


   Then:

$$\frac{dz}{dx} = \frac{dz}{dy}\,\frac{dy}{dx}$$

A single chain of nodes

A neural net with one node per hidden layer:



For a specific input x,

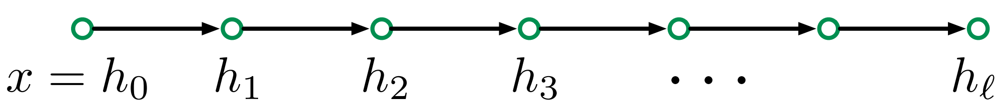
- $h_i = \sigma(w_i h_{i-1} + b_i)$
- The loss L can be gleaned from h_ℓ

To compute dL/dw_i we just need dL/dh_i:

$$\frac{dL}{dw_i} = \frac{dL}{dh_i}\,\frac{dh_i}{dw_i} = \frac{dL}{dh_i}\,\sigma'(w_i h_{i-1} + b_i)\,h_{i-1}$$

Backpropagation

- On a single forward pass, compute all the h_i.
- On a single backward pass, compute $dL/dh_\ell, \ldots, dL/dh_1$



From $h_{i+1} = \sigma(w_{i+1} h_i + b_{i+1})$, we have

$$\frac{dL}{dh_i} = \frac{dL}{dh_{i+1}}\,\frac{dh_{i+1}}{dh_i} = \frac{dL}{dh_{i+1}}\,\sigma'(w_{i+1} h_i + b_{i+1})\,w_{i+1}$$

Improving generalization

1. **Early stopping**
   - Validation set to better track error rate
   - Revert to earlier model when recent training hasn't improved error

2. **Dropout**
   During training, delete each hidden unit with probability 1/2, independently.