Bayesian Learning

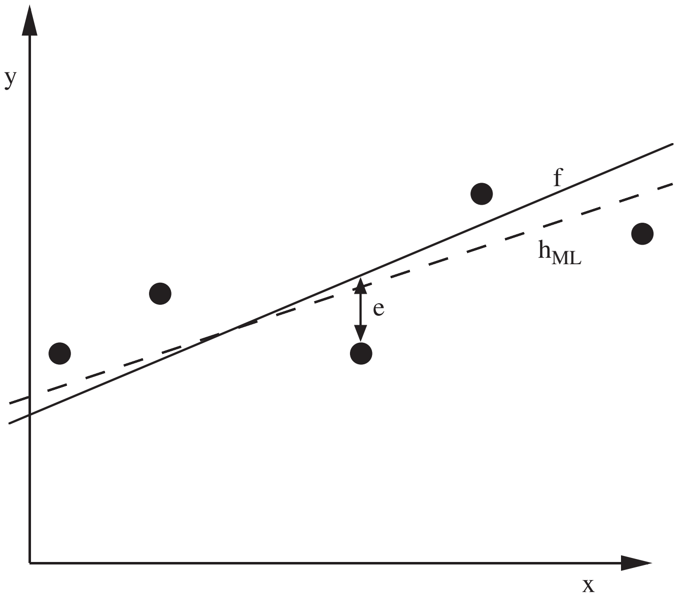
Last time:

- Bayes rule
- Maximum a posteriori (MAP) hypotheses
- Maximum likelihood (ML) hypotheses
- Bayesian perspective on consistent concept learner
- Minimizing squared error as maximizing likelihood

Today:

- More on maximizing likelihood for neural networks

Min squared error and Max likelihood



Consider any real-valued target function f

Training examples $\langle x_i, \hat{f}(x_i) \rangle$, where

- $\hat{f}(x_i) = f(x_i) + e_i$
- e_i is random variable (noise) drawn independently for each x_i according to some Gaussian distribution with mean=0

Then the maximum likelihood hypothesis h_{ML} is the one that minimizes the sum of squared errors:

$$h_{ML} = \arg\min_{h \in H} \sum_{i=1}^{m} \left(\hat{f}(x_i) - h(x_i) \right)^2$$

Why isn't $h_{ML} = f$?

Learning Probabilistic Functions and Max Likelihood

Suppose we want to learn a probabilistic (non-deterministic) function $f : X \rightarrow \{0, 1\}$, such as

- initial patient symptoms $\rightarrow$ patient will recover
- credit history $\rightarrow$ person will repay new loan

General setting:

- Wish to interpret network output as *probability* of a 1 given input x (i.e., target function $X \rightarrow P(f(x) = 1)$
- Note brute force method: estimate $P(f(x) = 1)$ based on frequencies of 1's for all x's in training set, train network to fit these frequencies
- We can do better...
- Train network directly using observed values 0 or 1

What is maximum likelihood hypothesis (neural network) given training data D?

Maximum Likelihood Hypothesis

Assume data $D = \{\langle x_1, d_1\rangle \ldots \langle x_m, d_m\rangle\}$, where d_i is observed 0 or 1 value for x_i

What's $P(D|h)$?

Assuming x_i drawn independently,

$$P(D|h) = \prod_{i=1}^{m} P(x_i, d_i|h) = \prod_{i=1}^{m} P(d_i|h, x_i)$$

Here h is the learned network, so $h(x_i) = P(d_i = 1|h)$, and

$$P(d_i|h, x_i) = \begin{cases} h(x_i) & \text{if } d_i = 1 \\ \\ (1 - h(x_i)) & \text{if } d_i = 0 \end{cases}$$

Which we can rewrite as

$$P(d_i|h, x_i) = h(x_i)^{d_i}(1 - h(x_i))^{1-d_i}$$

Binomial distribution:

$$P(D|h) = \prod_{i=1}^{m} h(x_i)^{d_i}(1 - h(x_i))^{1-d_i}$$

Maximum Likelihood Hypothesis

Maximum likelihood hypothesis h maximizes $P(D|h)$, so let's maximize

$$P(D|h) = \prod_{i=1}^{m} h(x_i)^{d_i} (1 - h(x_i))^{1-d_i}$$

Same as maximizing log:

$$\ln P(D|h) = \sum_{i=1}^{m} d_i \ln h(x_i) + (1 - d_i) \ln(1 - h(x_i))$$

negation of this is called *cross entropy* (why?)

Can use gradient descent, by solving for $\frac{\partial \ln P(D|h)}{\partial w_{j,k}}$

$$\frac{\partial \ln P(D|h)}{\partial w_{j,k}} = \sum_{i=1}^{m} \frac{\partial \ln P(D|h)}{\partial h(x_i)} \frac{\partial h(x_i)}{\partial w_{j,k}}$$

$$\frac{\partial \ln P(D|h)}{\partial w_{j,k}} = \sum_{i=1}^{m} \frac{d_i - h(x_i)}{h(x_i)(1 - h(x_i))} \frac{\partial h(x_i)}{\partial w_{j,k}}$$

Maximum Likelihood Hypothesis, contd

$$\frac{\partial \ln P(D|h)}{\partial w_{j,k}} = \sum_{i=1}^{m} \frac{d_i - h(x_i)}{h(x_i)(1-h(x_i))} \frac{\partial h(x_i)}{\partial w_{j,k}}$$

Simple case: output layer units

$$\frac{\partial h(x_i)}{\partial w_{j,k}} = \sigma'(x_i) x_{i,j,k} = h(x_i)(1-h(x_i)) x_{i,j,k}$$

therefore

$$\frac{\partial \ln P(D|h)}{\partial w_{j,k}} = \sum_{i=1}^{m} (d_i - h(x_i))\ x_{i,j,k}$$

Cool! Gradient descent weight tuning rule to maximize $P(D|h)$

$$w_{j,k} \leftarrow w_{j,k} + \sum_{i=1}^{m} (d_i - h(x_i))\ x_{i,j,k}$$