

Intro. to machine learning (CSI 5325)

Lecture 12: Bayesian learning

Greg Hamerly

Some content from Tom Mitchell.

- 1 Bayes Theorem
- 2 MAP, ML hypotheses
- 3 MAP learners
- 4 ML learners

Two roles for Bayesian methods

Provides practical learning algorithms:

- Naive Bayes learning
- Bayesian belief network learning
- Combine prior knowledge (prior probabilities) with observed data
- Requires prior probabilities

Provides useful conceptual framework

- Provides “gold standard” for evaluating other learning algorithms
- Additional insight into Occam’s razor

Bayes theorem

$$P(h|D) = \frac{P(D|h)P(h)}{P(D)}$$

- $P(h)$ = prior probability of hypothesis h
- $P(D)$ = prior probability of training data D
- $P(h|D)$ = probability of h given D
- $P(D|h)$ = probability of D given h

Choosing hypotheses

Generally want most probable hypothesis given training data

Maximum a posteriori hypothesis h_{MAP} :

$$\begin{aligned}h_{MAP} &= \operatorname{argmax}_{h \in H} P(h|D) \\ &= \operatorname{argmax}_{h \in H} \frac{P(D|h)P(h)}{P(D)} \\ &= \operatorname{argmax}_{h \in H} P(D|h)P(h)\end{aligned}$$

If assume $P(h_i) = P(h_j)$ then can further simplify, and choose the *Maximum likelihood* (ML) hypothesis

$$h_{ML} = \operatorname{argmax}_{h_i \in H} P(D|h_i)$$

Bayes theorem

Does patient have cancer or not?

A patient takes a lab test and the result comes back positive. The test returns a correct positive result in only 98% of the cases in which the disease is actually present, and a correct negative result in only 97% of the cases in which the disease is not present. Furthermore, .008 of the entire population have this cancer.

$$P(\text{cancer}) =$$

$$P(+|\text{cancer}) =$$

$$P(+|\neg\text{cancer}) =$$

$$P(\neg\text{cancer}) =$$

$$P(-|\text{cancer}) =$$

$$P(-|\neg\text{cancer}) =$$

Basic formulas for probabilities

- *Product Rule*: probability $P(A \cap B)$ of a conjunction of two events A and B:

$$P(A \cap B) = P(A|B)P(B) = P(B|A)P(A)$$

- *Sum Rule*: probability of a disjunction of two events A and B:

$$P(A \cup B) = P(A) + P(B) - P(A \cap B)$$

- *Theorem of total probability*: if events A_1, \dots, A_n are mutually exclusive with $\sum_{i=1}^n P(A_i) = 1$, then

$$P(B) = \sum_{i=1}^n P(B|A_i)P(A_i)$$

Brute force MAP hypothesis learner

- 1 For each hypothesis h in H , calculate the posterior probability

$$P(h|D) = \frac{P(D|h)P(h)}{P(D)}$$

- 2 Output the hypothesis h_{MAP} with the highest posterior probability

$$h_{MAP} = \operatorname{argmax}_{h \in H} P(h|D)$$

Relation to concept learning

Consider our usual concept learning task

- instance space X , hypothesis space H , training examples D
- consider the *FindS* learning algorithm (outputs most specific hypothesis from the version space $VS_{H,D}$)

What would Bayes rule produce as the MAP hypothesis?

Does *FindS* output a MAP hypothesis??

Relation to concept learning

Assume fixed set of instances $\langle x_1, \dots, x_m \rangle$

Assume D is the set of classifications $D = \langle c(x_1), \dots, c(x_m) \rangle$

Choose $P(D|h)$:

Relation to concept learning

Assume fixed set of instances $\langle x_1, \dots, x_m \rangle$

Assume D is the set of classifications $D = \langle c(x_1), \dots, c(x_m) \rangle$

Choose $P(D|h)$

- $P(D|h) = 1$ if h consistent with D
- $P(D|h) = 0$ otherwise

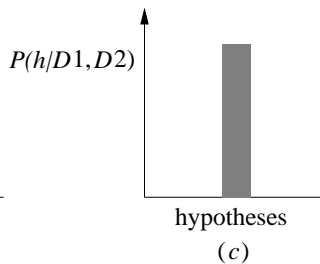
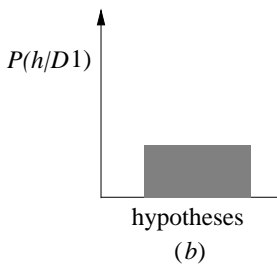
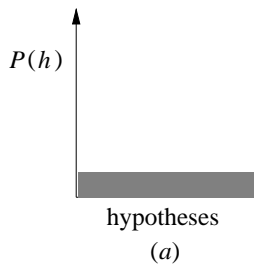
Choose $P(h)$ to be *uniform* distribution

- $P(h) = \frac{1}{|H|}$ for all h in H

Then,

$$P(h|D) = \begin{cases} \frac{1}{|V_{S_{H,D}}|} & \text{if } h \text{ is consistent with } D \\ 0 & \text{otherwise} \end{cases}$$

Evolution of posterior probabilities



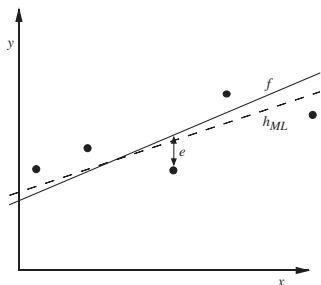
Learning a real-valued function

With some simple assumptions, maximum likelihood learning can be converted to squared-error minimization.

Occurs often in regression (i.e. learning real-valued functions).

- samples are independent
- true output signal is corrupted by noise
 - noise is Gaussian with zero mean
 - noise is i.i.d. (w.r.t. itself)
 - noise is independent of input

Learning a real-valued function



Consider any real-valued target function f

Training examples $\langle x_i, d_i \rangle$, where d_i is noisy training value:

- $d_i = f(x_i) + e_i$
- $e_i \sim N(0, \sigma^2)$ is a random variable (noise) drawn independently for each x_i according to some Gaussian distribution with $\mu = 0$, unknown σ^2

Learning a real-valued function

Training examples $\langle x_i, d_i \rangle$, where d_i is noisy training value

- $d_i = f(x_i) + e_i$
- 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:

$$h_{ML} = \operatorname{argmin}_{h \in H} \sum_{i=1}^m (d_i - h(x_i))^2$$

Learning a real-valued function

$$\begin{aligned}h_{ML} &= \operatorname{argmax}_{h \in H} p(D|h) \\ &= \operatorname{argmax}_{h \in H} \prod_{i=1}^m p(d_i|h) \\ &= \operatorname{argmax}_{h \in H} \prod_{i=1}^m \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2}\left(\frac{d_i - h(x_i)}{\sigma}\right)^2}\end{aligned}$$

Maximize natural log of this instead...

Learning a real-valued function

Maximize natural log of this instead...

$$\begin{aligned}h_{ML} &= \operatorname{argmax}_{h \in H} \sum_{i=1}^m \ln \frac{1}{\sqrt{2\pi\sigma^2}} - \frac{1}{2} \left(\frac{d_i - h(x_i)}{\sigma} \right)^2 \\&= \operatorname{argmax}_{h \in H} \sum_{i=1}^m -\frac{1}{2} \left(\frac{d_i - h(x_i)}{\sigma} \right)^2 \\&= \operatorname{argmax}_{h \in H} \sum_{i=1}^m -(d_i - h(x_i))^2 \\&= \operatorname{argmin}_{h \in H} \sum_{i=1}^m (d_i - h(x_i))^2\end{aligned}$$