Learning with Probabilities

CS194-10 Fall 2011 Lecture 15

Outline

♢ Bayesian learning
- – eliminates arbitrary loss functions and regularizers
- – facilitates incorporation of prior knowledge
- – quantifies hypothesis and prediction uncertainty
- – gives optimal predictions

♢ Maximum *a posteriori* and maximum likelihood learning

♢ Maximum likelihood parameter learning

Full Bayesian learning

View learning as Bayesian updating of a probability distribution over the hypothesis space

H is the hypothesis variable, values $h_1, h_2, \ldots$, prior $\mathbf{P}(H)$

ith observation x_i gives the outcome of random variable X_i
training data $\mathbf{X} = x_1, \ldots, x_N$

Given the data so far, each hypothesis has a posterior probability:

$$P(h_k|\mathbf{X}) = \alpha P(\mathbf{X}|h_k)P(h_k)$$

where $P(\mathbf{X}|h_k)$ is called the likelihood

Predictions use a likelihood-weighted average over the hypotheses:

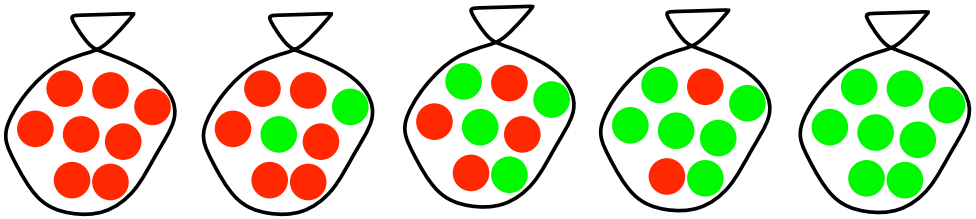
$$\mathbf{P}(X_{N+1}|\mathbf{X}) = \Sigma_k \, \mathbf{P}(X_{N+1}|\mathbf{X}, h_k)P(h_k|\mathbf{X}) = \Sigma_k \, \mathbf{P}(X_{N+1}|h_k)P(h_k|\mathbf{X})$$

No need to pick one best-guess hypothesis!

Example

Suppose there are five kinds of bags of candies:

10% are h_1: 100% cherry candies
20% are h_2: 75% cherry candies + 25% lime candies
40% are h_3: 50% cherry candies + 50% lime candies
20% are h_4: 25% cherry candies + 75% lime candies
10% are h_5: 100% lime candies



Then we observe candies drawn from some bag: 

What kind of bag is it? What flavour will the next candy be?

Posterior probability of hypotheses

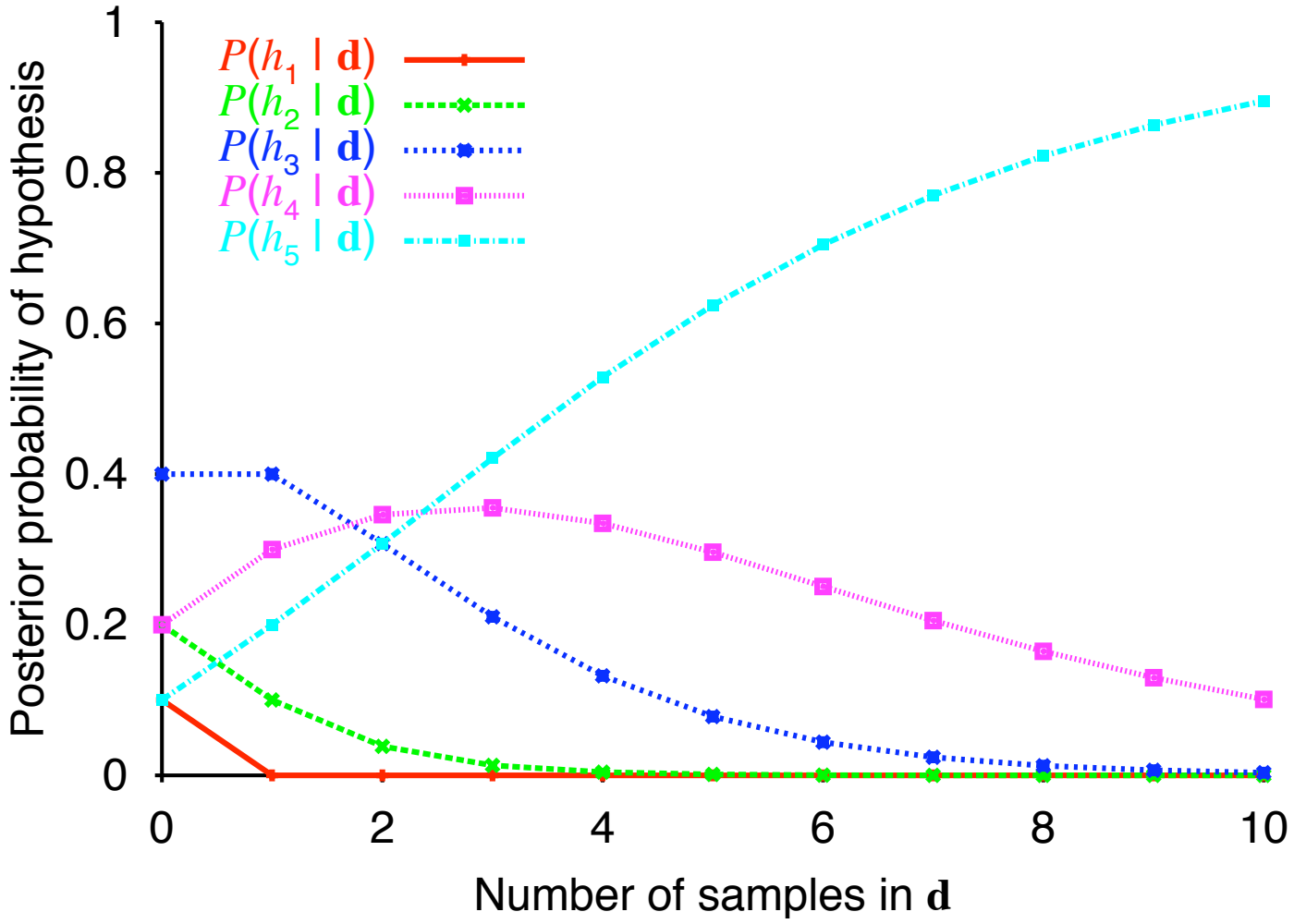
$$P(h_k|\mathbf{X}) = \alpha P(\mathbf{X}|h_k)P(h_k)$$

$$P(h_1 \mid \textsf{5 limes}) = \alpha P(\textsf{5 limes} \mid h_1)P(h_1) = \alpha \cdot 0.0^5 \cdot 0.1 = 0$$
$$P(h_2 \mid \textsf{5 limes}) = \alpha P(\textsf{5 limes} \mid h_2)P(h_2) = \alpha \cdot 0.25^5 \cdot 0.2 = 0.000195\alpha$$
$$P(h_3 \mid \textsf{5 limes}) = \alpha P(\textsf{5 limes} \mid h_3)P(h_3) = \alpha \cdot 0.5^5 \cdot 0.4 = 0.0125\alpha$$
$$P(h_4 \mid \textsf{5 limes}) = \alpha P(\textsf{5 limes} \mid h_4)P(h_4) = \alpha \cdot 0.75^5 \cdot 0.2 = 0.0475\alpha$$
$$P(h_5 \mid \textsf{5 limes}) = \alpha P(\textsf{5 limes} \mid h_5)P(h_5) = \alpha \cdot 1.0^5 \cdot 0.1 = 0.1\alpha$$

$$\alpha = 1/(0 + 0.000195 + 0.0125 + 0.0475 + 0.1) = 6.2424$$

$$P(h_1 \mid \textsf{5 limes}) = 0$$
$$P(h_2 \mid \textsf{5 limes}) = 0.00122$$
$$P(h_3 \mid \textsf{5 limes}) = 0.07803$$
$$P(h_4 \mid \textsf{5 limes}) = 0.29650$$
$$P(h_5 \mid \textsf{5 limes}) = 0.62424$$

Posterior probability of hypotheses

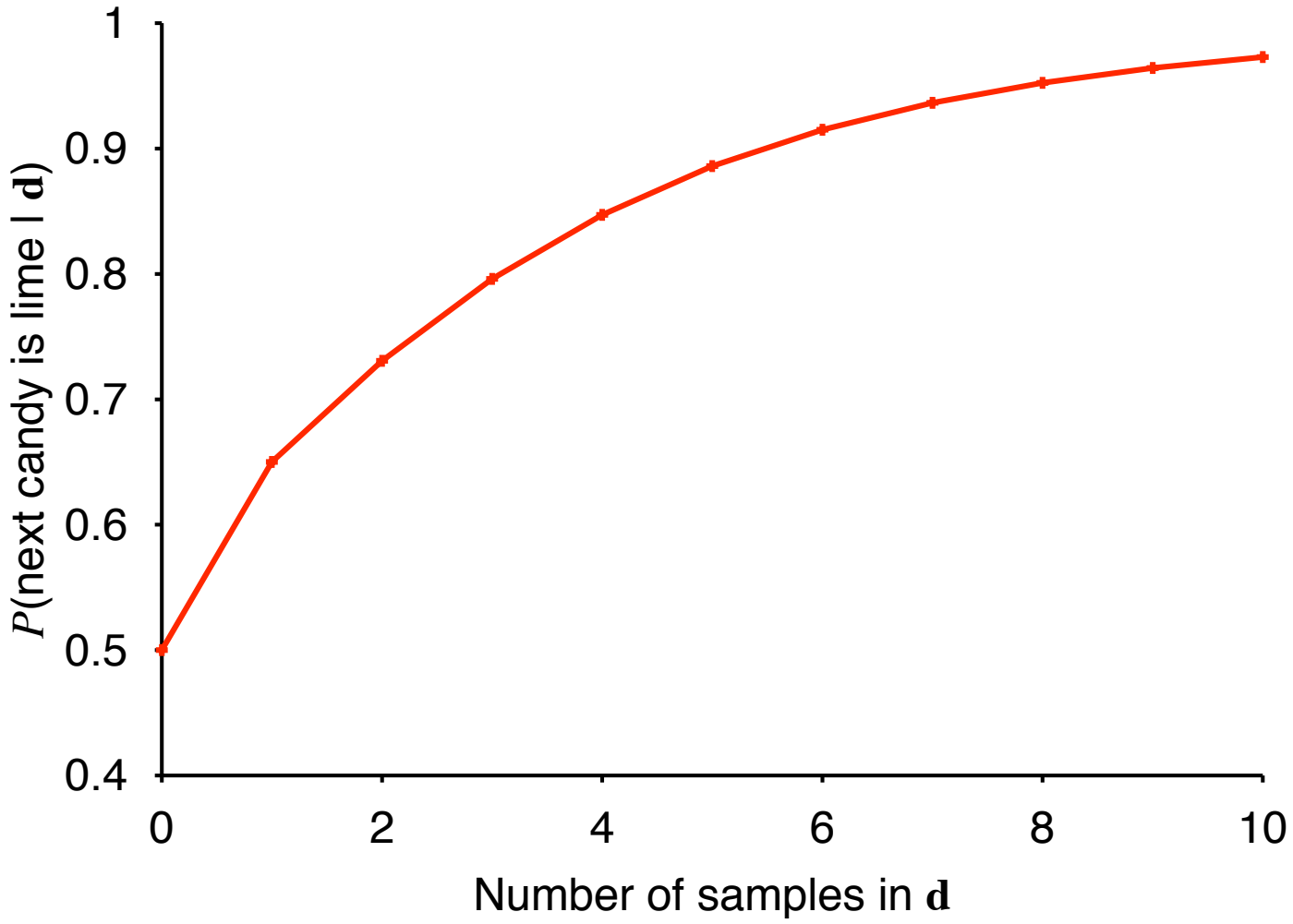


Prediction probability

$$\mathbf{P}(X_{N+1}|\mathbf{X}) = \Sigma_k \, \mathbf{P}(X_{N+1}|\mathbf{X}, h_k)P(h_k|\mathbf{X}) = \Sigma_k \, \mathbf{P}(X_{N+1}|h_k)P(h_k|\mathbf{X})$$

$$\begin{aligned}
P(\text{lime on 6} \mid \text{5 limes}) &= P(\text{lime on 6} \mid h_1)P(h_1 \mid \text{5 limes}) \\
&+ P(\text{lime on 6} \mid h_2)P(h_2 \mid \text{5 limes}) \\
&+ P(\text{lime on 6} \mid h_3)P(h_3 \mid \text{5 limes}) \\
&+ P(\text{lime on 6} \mid h_4)P(h_4 \mid \text{5 limes}) \\
&+ P(\text{lime on 6} \mid h_5)P(h_5 \mid \text{5 limes}) \\
&= 0 \times 0 \\
&+ 0.25 \times 0.00122 \\
&+ 0.5 \times 0.07830 \\
&+ 0.75 \times 0.29650 \\
&+ 1.0 \times 0.62424 \\
&= 0.88607
\end{aligned}$$

Prediction probability



Learning from positive examples only

Example from Tenenbaum via Murphy, Ch.3:

Given examples of some unknown class, a predefined subset of $\{1, \ldots, 100\}$, output a hypothesis as to what the class is

E.g., $\{16, 8, 2, 64\}$

Boolean classification problem; simplest consistent solution is "everything."

[This is the basis for Chomsky's "Poverty of the Stimulus" argument purporting to prove that humans must have innate grammatical knowledge]

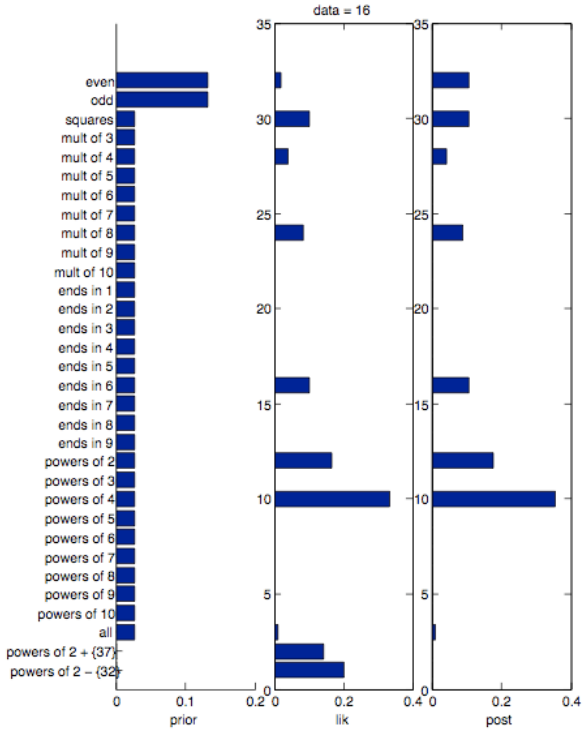
Bayesian counterargument

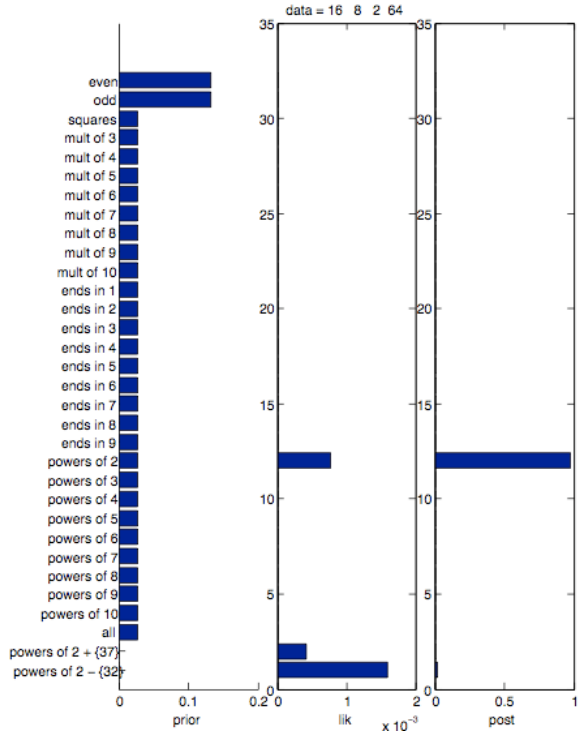
Assuming numbers are sampled uniformly from the class:

$P(\{16, 8, 2, 64\} \mid \text{powers of 2}) = 7^{-4} \approx 4.2 \times 10^{-4}$

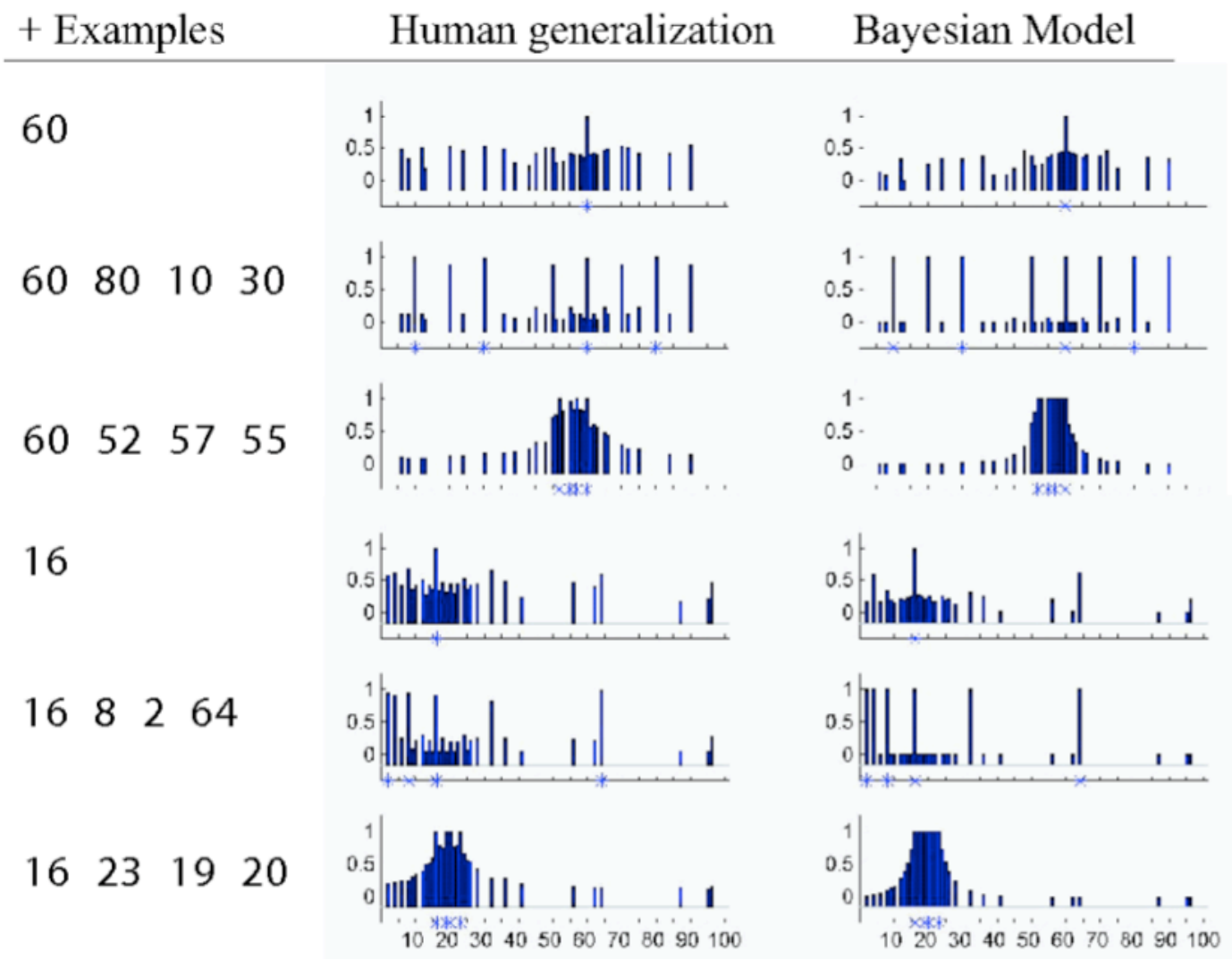
$P(\{16, 8, 2, 64\} \mid \text{everything}) = 100^{-4} = 10^{-8}$

This difference far outweighs any reasonable simplicity-based prior





Bayes vs. Humans



MAP approximation

Summing over the hypothesis space is often intractable
(e.g., 18,446,744,073,709,551,616 Boolean functions of 6 attributes)

Maximum a posteriori (MAP) learning: choose h_{MAP} maximizing $P(h_k|\mathbf{X})$

I.e., maximize $P(\mathbf{X}|h_k)P(h_k)$
or minimize $-\log P(\mathbf{X}|h_k) - \log P(h_k)$

. . . or, in information theory terms, minimize
bits to encode data given hypothesis + bits to encode hypothesis
This is the basic idea of minimum description length (MDL) learning

In science experiments, "inputs" are fixed, deterministic h predicts "outputs"
$\Rightarrow$ $P(\mathbf{X}|h_k)$ is 1 if consistent, 0 otherwise
$\Rightarrow$ MAP = simplest consistent hypothesis

ML approximation

For large data sets, prior becomes irrelevant

Maximum likelihood (ML) learning: choose h_{ML} maximizing $P(\mathbf{X}|h_k)$

I.e., simply get the best fit to the data; identical to MAP for uniform prior (which is reasonable if all hypotheses are of the same complexity)

ML is the "standard" (non-Bayesian) statistical learning method

A simple generative model: Bernoulli

A generative model is a probability model from which the probability of any observable data set can be derived

[Usually contrasted with a discriminative or conditional model, which gives only the probability for the "output" given the observable "inputs"]

E.g., Bernoulli$[\theta]$ model:

$$P(X_i = 1) = \theta; \; P(X_i = 0) = 1 - \theta$$

or $P(X_i = x_i) = \theta^{x_i}(1-\theta)^{1-x_i}$

Suppose we get a bag of candy from a new manufacturer;
fraction θ of cherry candies

Any θ is possible: continuum of hypotheses h_θ

ML estimation of Bernoulli model

Suppose we unwrap N candies, c cherries and $\ell = N - c$ limes
These are i.i.d. (independent, identically distributed) observations, so

$$P(\mathbf{X}|h_\theta) = \prod_{i=1}^{N} P(x_i|h_\theta) = \theta^{\sum_i x_i}(1-\theta)^{N-\sum_i x_i} = \theta^c \cdot (1-\theta)^\ell$$

Maximize this w.r.t. θ—which is easier for the log-likelihood:

$$L(\mathbf{X}|h_\theta) = \log P(\mathbf{X}|h_\theta) = \sum_{i=1}^{N} \log P(x_i|h_\theta) = c\log\theta + \ell\log(1-\theta)$$

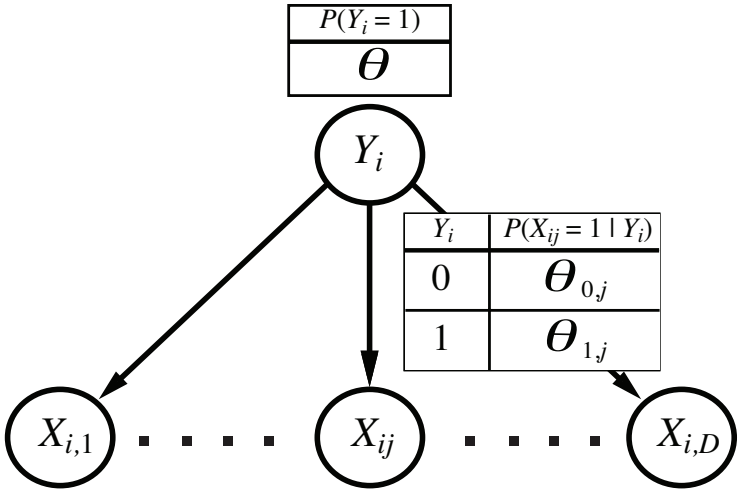
$$\frac{dL(\mathbf{X}|h_\theta)}{d\theta} = \frac{c}{\theta} - \frac{\ell}{1-\theta} = 0 \quad \Rightarrow \quad \theta = \frac{c}{c+\ell} = \frac{c}{N}$$

Seems sensible, but causes problems with 0 counts!

Naive Bayes models

Generative model for discrete (often Boolean) classification problems:

- Each example has discrete class variable Y_i
- Each example has discrete or continuous attributes X_{ij}, $j = 1, \ldots, D$
- Attributes are **conditionally independent** given the class value:



$$P(y_i, x_{i,1}, \ldots, x_{i,D}) = P(y_i) \prod_{j=1}^{D} P(x_{ij} \mid y_i)$$
$$= \theta^{y_i}(1-\theta)^{1-y_i} \prod_{j=1}^{D} \theta_{y_i,j}^{x_{ij}} (1-\theta_{y_i,j})^{1-x_{ij}}$$

ML estimation of Naive Bayes models

Likelihood is
$P(\mathbf{X} \mid h_\theta) = \prod_{i=1}^{N} \theta^{y_i}(1-\theta)^{1-y_i} \prod_{j=1}^{D} \theta_{y_i,j}^{x_{ij}}(1-\theta_{y_i,j})^{1-x_{ij}}$

Log likelihood is
$L = \log P(\mathbf{X} \mid h_\theta) =$
$\sum_{i=1}^{N} y_i \log\theta + (1-y_i)\log(1-\theta) + \sum_{j=1}^{D} x_{ij}\log\theta_{y_i,j} + (1-x_{ij})\log(1-\theta_{y_i,j})$

This has parameters in separate terms, so derivatives are decoupled:

$$\frac{\partial L}{\partial \theta} = \sum_{i=1}^{N} \frac{y_i}{\theta} - \frac{1-y_i}{1-\theta} = \frac{N_1}{\theta} - \frac{N-N_1}{1-\theta}$$
$$\frac{\partial L}{\partial \theta_{yj}} = \sum_{i:y_i=y} \frac{x_{ij}}{\theta_{yj}} - \frac{1-x_{ij}}{1-\theta_{yj}} = \frac{N_{yj}}{\theta_{yj}} - \frac{N_y - N_{yj}}{1-\theta_{yj}}$$

where N_y = number of examples with class label y
and N_{yj} = number of examples with class label y and value 1 for X_{ij}

ML estimation contd.

Setting derivatives to zero:

$$\theta = N_1/N \qquad \text{as before}$$
$$\theta_{yj} = N_{yj}/N_y$$

I.e., count the fraction of each class with jth attribute set to 1

$\Rightarrow$ $O(ND)$ time to train the model

Example: 1000 cherry and lime candies, wrapped in red or green wrappers by the Surprise Candy Company

400 cherry, of which 300 have red wrappers and 100 green wrappers
600 lime, of which 120 have red wrappers and 480 green wrappers

$$\theta = P(Flavor = cherry) = 400/100 = 0.40$$
$$\theta_{11} = P(Wrapper = red \mid Flavor = cherry) = 300/400 = 0.75$$
$$\theta_{01} = P(Wrapper = red \mid Flavor = lime) = 120/600 = 0.20$$

Classifying a new example

$$\begin{aligned} P(Y=1 \mid x_1,\ldots,x_D) &= \alpha\ P(x_1,\ldots,x_D \mid Y=1)P(Y=1) \\ &= \alpha\ \theta\ \textstyle\prod_{j=1}^{D} \theta_{1j}^{x_j}(1-\theta_{1j})^{1-x_j} \end{aligned}$$

$$\begin{aligned} &\log P(Y=1 \mid x_1,\ldots,x_D) \\ &= \log\alpha + \log\theta + \textstyle\sum_{j=1}^{D} x_j \log\theta_{1j} + (1-x_j)\log(1-\theta_{1j}) \\ &= \left(\log\alpha + \log\theta + \textstyle\sum_{j=1}^{D}(1-\theta_{1j})\right) + \textstyle\sum_{j=1}^{D} x_j\,(\log(\theta_{1j}/(1-\theta_{1j}))) \end{aligned}$$

The set of points where

$$P(Y=1 \mid x_1,\ldots,x_D) = P(Y=0 \mid x_1,\ldots,x_D) = 0.5$$

is a **linear separator**! (But location is sensitive to class prior.)

Summary

Full Bayesian learning gives best possible predictions but is intractable

MAP learning balances complexity with accuracy on training data

Maximum likelihood assumes uniform prior, OK for large data sets

1. Choose a parameterized family of models to describe the data
requires substantial insight and sometimes new models

2. Write down the likelihood of the data as a function of the parameters
may require summing over hidden variables, i.e., inference

3. Write down the derivative of the log likelihood w.r.t. each parameter

4. Find the parameter values such that the derivatives are zero
may be hard/impossible; modern optimization techniques help

Naive Bayes is a simple generative model with a very fast training method that finds a linear separator in input feature space

and provides probabilistic predictions