Statistical learning
Homework

50 points, due in one week

16.3

16.11

17.4 (with value iteration, not policy iteration)

18.4

18.10
Finished testing, not finished assigning grades, but ready to send you feedback on how your programs performed.

Ryan Carr ran the tests. Sometime today he’ll send you email to let you know how your program performed. Some worked fine, some had some minor errors, some had major errors.

If your program had errors, we’ll give you a large amount of partial credit if you correct your programs by 11:59pm on Friday. Send them to jryancarr@gmail.com

I’ve posted a copy of the game supervisor and two dummy agents:

http://www.cs.umd.edu/~nau/cmsc421/project2/ct4.lisp
http://www.cs.umd.edu/~nau/cmsc421/project2/dummy-player.lisp
http://www.cs.umd.edu/~nau/cmsc421/project2/dummy-player1.lisp
Outline

◊ Bayesian learning

◊ Maximum a posteriori and maximum likelihood learning

◊ Bayes net learning
  – ML parameter learning with complete data
  – linear regression
Bayesian learning

Suppose we have a number of different hypotheses for the outcome of a random event:

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

**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

Given a randomly chosen bag, which kind is it?
Bayesian learning

To gather data, start drawing some of the candies from the bag

\[ d_1 \ d_2 \ d_3 \ d_4 \ d_5 \ d_6 \ d_7 \ d_8 \ d_9 \ d_{10} \]

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

---

View learning as Bayesian updating of a probability distribution over the hypothesis space (set of all hypotheses) \( \{h_1, \ldots, h_n\} \)

- Training data \( \mathbf{d} = \{d_1, \ldots, d_N\} \)
- \( j \)th observation \( d_j \) is the outcome of a random variable \( D_j \)

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

\[
P(h_i|\mathbf{d}) = \alpha P(\mathbf{d}|h_i)P(h_i)
\]

where \( P(\mathbf{d}|h_i) \) is called the likelihood.
Posterior probability of hypotheses

<table>
<thead>
<tr>
<th>$p_0$</th>
<th>$H$</th>
<th>$ch$</th>
<th>$li$</th>
</tr>
</thead>
<tbody>
<tr>
<td>.1</td>
<td>$h_1$</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>.2</td>
<td>$h_2$</td>
<td>.75</td>
<td>.25</td>
</tr>
<tr>
<td>.4</td>
<td>$h_3$</td>
<td>.5</td>
<td>.5</td>
</tr>
<tr>
<td>.2</td>
<td>$h_4$</td>
<td>.25</td>
<td>.75</td>
</tr>
<tr>
<td>.1</td>
<td>$h_5$</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

Suppose the bag is all lime:

$P(h_1 | d)$
$P(h_2 | d)$
$P(h_3 | d)$
$P(h_4 | d)$
$P(h_5 | d)$
Prediction probability

For predictions, use a likelihood-weighted average over all the hypotheses:

\[ P(X|d) = \sum_i P(X|d, h_i)P(h_i|d) = \sum_i P(X|h_i)P(h_i|d) \]

No need to pick one best-guess hypothesis!
MAP approximation

Summing over the hypothesis space,
\[ \sum_i P(X|h_i)P(h_i|d), \]
is often intractable
(e.g., 18,446,744,073,709,551,616 Boolean functions of 6 attributes)

**Maximum a posteriori** (MAP) learning: choose \( h_i \) that maximizes \( P(h_i|d) \)
i.e., maximize \( P(d|h_i)P(h_i) \)
ML approximation

For large data sets, the prior probability $P(h_i)$ becomes irrelevant

**Maximum likelihood** (ML) learning: choose $h_i$ that maximizes $P(d|h_i)$
  i.e., simply get the best fit to the data

Identical to MAP for uniform prior distribution
  (which is reasonable if we have no reason to believe any hypothesis is
  any more likely than any other hypothesis

ML is the “standard” (non-Bayesian) statistical learning method
ML parameter learning in Bayes nets

Bag from a new manufacturer; fraction $\theta$ of cherry candies?

Any $\theta$ is possible: continuum of hypotheses $h_\theta$
$\theta$ is a parameter for this simple (binomial) family of models

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

$$P(d|h_\theta) = \prod_{j=1}^{N} P(d_j|h_\theta) = \theta^c \cdot (1 - \theta)^\ell$$

Maximize this w.r.t. $\theta$. Easier to maximize the log-likelihood:

$$L(d|h_\theta) = \log P(d|h_\theta) = \sum_{j=1}^{N} \log P(d_j|h_\theta) = c \log \theta + \ell \log(1 - \theta)$$

$$\frac{dL(d|h_\theta)}{d\theta} = \frac{c}{\theta} - \frac{\ell}{1 - \theta} = 0 \quad \Rightarrow \quad \ldots$$
ML parameter learning in Bayes nets

Bag from a new manufacturer; fraction $\theta$ of cherry candies?

Any $\theta$ is possible: continuum of hypotheses $h_\theta$
$\theta$ is a parameter for this simple (binomial) family of models

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

$$P(d|h_\theta) = \prod_{j=1}^{N} P(d_j|h_\theta) = \theta^c \cdot (1-\theta)^\ell$$

Maximize this w.r.t. $\theta$. Easier to maximize the log-likelihood:

$$L(d|h_\theta) = \log P(d|h_\theta) = \sum_{j=1}^{N} \log P(d_j|h_\theta) = c \log \theta + \ell \log(1-\theta)$$

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

Duh ...
Multiple parameters

Suppose candy comes in red and green wrappers.

Red/green wrapper depends probabilistically on flavor:

Likelihood of green-wrapped cherry candy:

\[ P(F = \text{cherry}, W = \text{green}|h_{\theta,\theta_1,\theta_2}) = P(F = \text{cherry}|h_{\theta,\theta_1,\theta_2})P(W = \text{green}|F = \text{cherry}, h_{\theta,\theta_1,\theta_2}) = \theta \cdot (1 - \theta_1) \]

\( N \) candies, \( r_c \) red-wrapped cherry candies, etc.:

\[ P(d|h_{\theta,\theta_1,\theta_2}) = \theta^c(1 - \theta)^\ell \cdot \theta_1^{rc}(1 - \theta_1)^{gc} \cdot \theta_2^{r\ell}(1 - \theta_2)^{g\ell} \]

\[ L = [c \log \theta + \ell \log(1 - \theta)] + [r_c \log \theta_1 + g_c \log(1 - \theta_1)] + [r_\ell \log \theta_2 + g_\ell \log(1 - \theta_2)] \]
Multiple parameters contd.

\[ L = [c \log \theta + \ell \log(1 - \theta)] + [r_c \log \theta_1 + g_c \log(1 - \theta_1)] + [r_\ell \log \theta_2 + g_\ell \log(1 - \theta_2)] \]

Derivatives of \( L \) contain only the relevant parameter:

\[
\frac{\partial L}{\partial \theta} = \frac{c}{\theta} - \frac{\ell}{1 - \theta} = 0 \quad \Rightarrow \quad \theta = \frac{c}{c + \ell}
\]

\[
\frac{\partial L}{\partial \theta_1} = \frac{r_c}{\theta_1} - \frac{g_c}{1 - \theta_1} = 0 \quad \Rightarrow \quad \theta_1 = \frac{r_c}{r_c + g_c}
\]

\[
\frac{\partial L}{\partial \theta_2} = \frac{r_\ell}{\theta_2} - \frac{g_\ell}{1 - \theta_2} = 0 \quad \Rightarrow \quad \theta_2 = \frac{r_\ell}{r_\ell + g_\ell}
\]

Thus the parameters can be learned separately.
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