Lecture 9 - Learning with Uncertainty (I)

Jesse Hoey
School of Computer Science
University of Waterloo

February 16, 2018

Readings: Poole & Mackworth (2nd Ed.) Chapt. 10.1, 10.4
Bayesian Learning

Basic premise:
- have a number of hypotheses or models
- don’t know which one is “correct”
- Bayesians assume all are correct to a certain degree
- Have a distribution over the models
- Compute expected prediction given this average

Suppose $X$ is input features, and $Y$ is target feature, $d = \{x_1, y_1, x_2, y_2, \ldots, x_N, y_N\}$ is evidence (data), $x$ is a new input, and we want to know corresponding output $y$. We sum over all models, $m \in M$

$$P(Y|x, d) = \sum_{m \in M} P(Y, m|x, d)$$

$$= \sum_{m \in M} P(Y|m, x, d)P(m|x, d)$$

$$= \sum_{m \in M} P(Y|m, x)P(m|d)$$
Have a bag of Candy with 2 flavors (Lime, Cherry)
Sold in bags with different ratios
- 100% cherry
- 75% cherry + 25% lime
- 50% cherry + 50% lime
- 25% cherry + 75% lime
- 100% lime

With a random sample - what ratio is in the bag?
Hypothesis $H$: probabilistic theory about the world

- $h_1$: 100% cherry
- $h_2$: 75% cherry + 25% lime
- $h_3$: 50% cherry + 50% lime
- $h_4$: 25% cherry + 75% lime
- $h_5$: 100% lime

Data $D$: evidence about the world

- $d_1$: 1st candy is lime
- $d_2$: 2nd candy is lime
- $d_3$: 3rd candy is lime
- ...

Statistical Learning
Bayesian Learning

- Prior: $P(H)$
- Likelihood: $P(d|H)$
- Evidence: $d = \{d_1, d_2, \ldots, d_n\}$

Bayesian learning: update the posterior (Bayes’ theorem)

$$P(H|d) \propto P(d|H)P(H)$$
Bayesian Prediction

- want to predict $X$: (e.g. next candy)

$$P(X|d) = \sum_i P(X|d, h_i)P(h_i|d)$$

$$= \sum_i P(X|h_i)P(h_i|d)$$

- Predictions are weighted averages of the predictions of the individual hypotheses
- Hypotheses serve as "intermediaries" between raw data and prediction
Posterior

Posteriors given data generated from $h_5$

- $P(h_1|d)$
- $P(h_2|d)$
- $P(h_3|d)$
- $P(h_4|d)$
- $P(h_5|d)$

Number of samples vs. $P(h|d)$ for each hypothesis $h_i$.
Bayesian Learning

Bayesian learning properties:

- **Optimal**: given prior, no other prediction is correct more often than the Bayesian one
- **No overfitting**: prior/likelihood both penalise complex hypotheses

Price to pay:

- Bayesian learning may be intractable when hypothesis space is large
- Sum over hypotheses space may be intractable

Solution: approximate Bayesian learning
Maximum a posteriori

- Idea: make prediction based on most probable hypothesis: $h_{\text{MAP}}$
- $h_{\text{MAP}} = \arg\max_i P(h_i|d)$
- $P(X|d) \approx P(X|h_{\text{MAP}})$
- Contrast with Bayesian learning where all hypotheses are used
Posterior

Posteriors given data generated from $h_5$
MAP properties

- MAP prediction less accurate than Bayesian one since it relies only on one hypothesis
- MAP and Bayesian predictions converge as data increases
- **no overfitting** (as in Bayesian learning)
- Finding $h_{MAP}$ may be intractable:

$$h_{MAP} = \arg\max_h P(h|d)$$
$$= \arg\max_h P(h)P(d|h)$$
$$= \arg\max_h P(h) \prod_i P(d_i|h)$$

product induces a non-linear optimisation
- can take the log to linearise

$$h_{MAP} = \arg\max_h \left[ \log P(h) + \sum_i \log P(d_i|h) \right]$$
Maximum Likelihood (ML)

- Idea: Simplify MAP by assuming uniform prior (i.e. $P(h_i) = P(h_j) \forall i, j$)

  $$h_{MAP} = \arg\max_h P(h)P(d|h)$$

  $$h_{ML} = \arg\max_h P(d|h)$$

- Make prediction based on $h_{ML}$ only

  $$P(X|d) \approx P(X|h_{ML})$$
ML Properties

- ML prediction **less accurate** than Bayesian or MAP predictions since it ignores prior and relies on one hypothesis.
- but ML, MAP and Bayesian converge as the amount of data increases.
- more susceptible to **overfitting**: no prior.
- $h_{ML}$ is often easier to find than $h_{MAP}$.

$$h_{ML} = \arg\max_h \sum_i \log P(d_i | h)$$
Generalise the hypothesis space to a continuous quantity

- \( P(\text{Flavour} = \text{cherry}) = \theta \) (like a “coin weight”)
- \( P(\text{Flavour} = \text{lime}) = (1 - \theta) \)
- \( P(k \text{ lime}, n \text{ cherry}) = \theta^n(1 - \theta)^k \) (one order)
- \( P(k \text{ lime}, n \text{ cherry}) = \binom{n + k}{k} \theta^n(1 - \theta)^k \) (any order)
The Beta distribution $B(\theta, a, b) = \theta^{a-1}(1 - \theta)^{b-1}$
Bayesian classifiers

- Idea: if you knew the classification you could predict the values of features.

\[ P(Class|X_1 \ldots X_n) \propto P(X_1, \ldots , X_n|Class)P(Class) \]

- **Naive Bayesian classifier:** \( X_i \) are independent of each other given the class.
  Requires: \( P(Class) \) and \( P(X_i|Class) \) for each \( X_i \).

\[ P(Class|X_1 \ldots X_n) \propto \prod_i P(X_i|Class)P(Class) \]
Naïve Bayes classifier

- Predict class $C$ based on attributes $A_i$
- Parameters:
  \[
  \theta = P(C = true) \\
  \theta_{i1} = P(A_i = true|C = true) \\
  \theta_{i0} = P(A_i = true|C = false)
  \]
- Assumption: $A_i$s are independent given $C$. 

![Diagram of Naïve Bayes classifier]


Naïve Bayes classifier

<table>
<thead>
<tr>
<th>Action</th>
<th>Author</th>
<th>Thread</th>
<th>Length</th>
<th>Where</th>
</tr>
</thead>
<tbody>
<tr>
<td>e1</td>
<td>skips</td>
<td>known</td>
<td>new</td>
<td>long</td>
</tr>
<tr>
<td>e2</td>
<td>reads</td>
<td>unknown</td>
<td>new</td>
<td>short</td>
</tr>
<tr>
<td>e3</td>
<td>skips</td>
<td>unknown</td>
<td>old</td>
<td>long</td>
</tr>
</tbody>
</table>
| ...    | ...    | ...    | ...    | ...   | ...

ML sets

- $\theta$ to relative frequency of reads, skips
- $\theta_{i1}$ to relative frequency of $A_i$ given reads, skips

$$\theta_{i1} = \frac{\text{number of articles that are read and have } A_i = true}{\text{number of articles that are read}}$$

$$\theta_{i0} = \frac{\text{number of articles that are skipped and have } A_i = true}{\text{number of articles that are skipped}}$$
Bayesian Network Parameter Learning (ML)

For fully observed data

- Parameters $\theta_{V, \text{pa}(V)=v^i}$
- CPTs $\theta_{V, \text{pa}(V)=v} = P(V|\text{Pa}(V) = v)$
- Data $d$:
  
  $d_1 = < V_1 = v_{1,1}, V_2 = v_{2,1}, \ldots, V_n = v_{n,1} >$

  $d_2 = < V_2 = v_{1,2}, V_2 = v_{2,2}, \ldots, V_n = v_{n,2} >$

  \ldots

- Maximum likelihood: Set $\theta_{V, \text{pa}(V)=v}$ to the relative frequency of values of $V$ given the values $v$ of the parents of $V$
Occam’s Razor

e.g. from MacKay
www.inference.phy.cam.ac.uk/mackay/itila/book.html
Occam’s Razor

e.g. from MacKay
www.inference.phy.cam.ac.uk/mackay/itila/book.html

Figure 28.2. How many boxes are behind the tree?
Occam’s Razor

\[
P(D|H_1) \quad \text{Evidence} \quad P(D|H_2)
\]

\[C_1 \quad D\]
Minimum Description Length

Bayesian learning: update the posterior (Bayes’ theorem)

\[ P(H|d) = kP(d|H)P(H) \]

So

\[ -\log P(H|d) = - \log P(d|H) - \log P(H) \]

- first term: number of bits to encode the data given the model
- second term: number of bits to encode the model
- **MDL principle** is to choose the model that minimizes the number of bits it takes to describe both the model and the data given the model.
- generally, as second term increases, the first term decreases (reduced residuals)
- second term: part of Occam factor
Learning under Uncertainty II (Poole & Mackworth (2nd Ed.) chapter 10.2,10.3,10.5)