Bayesian Learning

Complicated derivation of known things.

- Maximal aposteriory probability hypothesis (MAP) (nejpravděpodobnější hypotéza)
- Maximum likelihood hypothesis (ML) (maximálně věrohodná hypotéza)
- Bayes optimal prediction
- EM algorithm
Our favorite candy comes in two flavors: cherry and lime, both in the same wrapper.

They are in a bag in one of following rations of cherry candies and prior probability of bags:

<table>
<thead>
<tr>
<th>hypothesis (bag type)</th>
<th>$h_1$</th>
<th>$h_2$</th>
<th>$h_3$</th>
<th>$h_4$</th>
<th>$h_5$</th>
</tr>
</thead>
<tbody>
<tr>
<td>cherry</td>
<td>100%</td>
<td>75%</td>
<td>50%</td>
<td>25%</td>
<td>0%</td>
</tr>
<tr>
<td>prior probability $h_i$</td>
<td>10%</td>
<td>20%</td>
<td>40%</td>
<td>20%</td>
<td>10%</td>
</tr>
</tbody>
</table>

The first candy is cherry. What is the probability next candy from the same bag is cherry?
Maximum aposteriory probability hypothesis (MAP)

- We assume large bags of bonbons, the result of one missing bonbon in the bag is negligible.
- Recall Bayes formula:

  \[ P(h_i | B = c) = \frac{P(B = c | h_i) \cdot P(h_i)}{\sum_{j=1,...,5} P(B = c | h_j) \cdot P(h_j)} = \frac{P(B = c | h_i) \cdot P(h_i)}{P(B = c)} \]

- We look for the MAP hypothesis
  \[ \text{argmax}_i P(h_i | B = c) = \text{argmax}_i P(B = c | h_i) \cdot P(h_i). \]
- Aposteriory probabilities of hypotheses are in the following table.
Candy Example: Aposteriori Probability of Hypotheses

| index | prior $P(h_i)$ | cherry ration $P(B = c|h_i)$ | cherry AND $h_i$ $P(B = c|h_i) \cdot P(h_i)$ | Aposteriori prob. $h_i$ $P(h_i|B = c)$ |
|-------|---------------|-----------------------------|---------------------------------------------|-----------------------------------------|
| 1     | 0.1           | 1                           | 0.1                                         | 0.2                                     |
| 2     | 0.2           | 0.75                        | 0.15                                        | 0.3                                     |
| 3     | 0.4           | 0.5                         | 0.2                                         | **0.4**                                 |
| 4     | 0.2           | 0.25                        | 0.05                                        | 0.1                                     |
| 5     | 0.1           | 0                           | 0                                           | 0                                       |

Which hypothesis is most probable?

$$h_{MAP} = \arg\max_i P(data|h_i) \cdot P(h_i)$$

What is the prediction of a new candy according the most probable hypothesis $h_{MAP}$?
MAP hypothesis maximizes:

\[ h_{MAP} = \underset{i}{\text{argmax}} P(data|h_i) \cdot P(h_i) \]

therefore minimizes:

\[ h_{MAP} = \underset{h}{\text{argmax}} P(data|h)P(h) \]
\[ = \underset{h}{\text{argmin}} \left[ -\log_2 P(data|h) - \log_2 P(h) \right] \]

**Minimal Description Length principle**: Find an encoding \( h \), such that the complexity of the encoding \( h \) plus the complexity of the data described by this encoding is minimal.

- Hypothesis complexity (lower bound of it) is \(-\log_2 P(h)\) measured in binary encoding.
- Data description complexity (lower bound of it) is \(-\log_2 P(data|h)\) measured in binary encoding.
Bayesian optimal prediction is weighted average of predictions of all hypotheses:

\[
P(N = c | data) = \sum_{j=1,\ldots,5} P(N = c | h_j, data) \cdot P(h_j | data)
\]

\[
= \sum_{j=1,\ldots,5} P(N = c | h_j) \cdot P(h_j | data)
\]

▶ If our model is correct, no prediction has smaller expected error than Bayesian optimal prediction.

▶ We always assume i.i.d. data, independently identically distributed. This allows the last equation above.
Candy Example: Bayesian Optimal Prediction

\[
P(h_i|B = c) \cdot P(N = c|h_i) = \sum_{i} 0.645
\]

| i | \(P(h_i|B = c)\) | \(P(N = c|h_i)\) | \(P(N = c|h_i) \cdot P(h_i|B = c)\) |
|---|------------------|------------------|---------------------------------|
| 1 | 0.2              | 1                | 0.2                             |
| 2 | 0.3              | 0.75             | 0.225                           |
| 3 | 0.4              | 0.5              | 0.2                             |
| 4 | 0.1              | 0.25             | 0.02                            |
| 5 | 0                | 0                | 0                               |
| \(\sum\) |                  |                  | 0.645                           |
Maximum Likelihood Estimate (ML)

- Usually, we do not know prior probabilities of hypotheses.
- Setting all prior probabilities equal leads to **Maximum Likelihood Estimate**

\[ h_{ML} = \arg\max_i P(data|h_i) \]

- Find the ML estimate:

<table>
<thead>
<tr>
<th>index</th>
<th>prior</th>
<th>cherry ration</th>
<th>cherry AND ( h_i )</th>
<th>Aposteriori prob. ( h_i )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( i )</td>
<td>( P(h_i) )</td>
<td>( P(B = c</td>
<td>h_i) )</td>
<td>( P(B = c</td>
</tr>
<tr>
<td>1</td>
<td>0.1</td>
<td>1</td>
<td>0.1</td>
<td>0.2</td>
</tr>
<tr>
<td>2</td>
<td>0.2</td>
<td>0.75</td>
<td>0.15</td>
<td>0.3</td>
</tr>
<tr>
<td>3</td>
<td>0.4</td>
<td>0.5</td>
<td>0.2</td>
<td>0.4</td>
</tr>
<tr>
<td>4</td>
<td>0.2</td>
<td>0.25</td>
<td>0.05</td>
<td>0.1</td>
</tr>
<tr>
<td>5</td>
<td>0.1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
New producer on the market. We do not know the ratios of bonbons, any $h_\theta$, kde $\theta \in \langle 0; 1 \rangle$ is possible, any prior probabilities $h_\theta$ are possible.

We look for maximum likelihood estimate.

For a given hypothesis $h_\theta$, the probability of a cherry candy is $\theta$, of lime candy $1 - \theta$.

Probability of a sequence of $c$ cherry and $l$ lime candies is:

$$P(data|h_\theta) = \theta^c \cdot (1 - \theta)^l$$
ML Estimate of Parameter $\theta$

- Probability of a sequence of $c$ cherry and $l$ lime candies is:

$$P(data|h_\theta) = \theta^c \cdot (1 - \theta)^l$$

- Usual trick is to take logarithm:

$$L(data|h_\theta) = c \cdot \log_2 \theta + l \cdot \log_2(1 - \theta)$$

- To find the maximum of $L$ (log likelihood) with respect to $\theta$ we set the derivative equal to 0:

$$\frac{\partial L(data|h_\theta)}{\partial \theta} = \frac{c}{\theta} - \frac{l}{1 - \theta}$$

$$\frac{c}{\theta} = \frac{l}{1 - \theta}$$

$$\theta = \frac{c}{c + l}$$
Producer introduced two colors of wrappers - red $r$ and green $g$.

Both flavors are wrapped in both wrappers, but with different probability of the red/green wrapper.

We need three parameters to model this situation:

\[
\begin{align*}
P(B = c) & \quad P(W = r | B = c) & \quad P(W = r | B = l) \\
\theta_0 & \quad \theta_1 & \quad \theta_2
\end{align*}
\]

Following table denotes observed frequencies:

<table>
<thead>
<tr>
<th>wrapper \ flavor</th>
<th>cherry</th>
<th>lime</th>
</tr>
</thead>
<tbody>
<tr>
<td>red</td>
<td>$r_c$</td>
<td>$r_l$</td>
</tr>
<tr>
<td>green</td>
<td>$g_c$</td>
<td>$g_l$</td>
</tr>
</tbody>
</table>
ML Estimate of Multiple Parameters

Parameters are:

| $P(B = c)$ | $P(W = r | B = c)$ | $P(W = r | B = l)$ |
|------------|--------------------|--------------------|
| $\theta_0$ | $\theta_1$         | $\theta_2$         |

Probability of the hypothesis $h_{\theta_0, \theta_1, \theta_2}$ is:

$$P(data | h_{\theta_0, \theta_1, \theta_2}) = \theta_1^r \cdot (1 - \theta_1)^g_c \cdot \theta_0^{r_c + g_c} \cdot \theta_2^{r_l} \cdot (1 - \theta_2)^g_l \cdot (1 - \theta_0)^{r_l + g_l}$$

$$L(data | h_{\theta_0, \theta_1, \theta_2}) = r_c \log_2 \theta_1 + g_c \log_2 (1 - \theta_1) + (r_c + g_c) \log_2 \theta_0$$
$$+ r_l \log_2 \theta_2 + g_l \log_2 (1 - \theta_2) + (r_l + g_l) \log_2 (1 - \theta_0)$$

We look for maximum:

$$\frac{\partial L(data | h_{\theta_0, \theta_1, \theta_2})}{\partial \theta_0} = \frac{r_c + g_c}{\theta_0} - \frac{r_l + g_l}{1 - \theta_0}$$

$$\theta_0 = \frac{(r_c + g_c)}{r_c + g_c + r_l + g_l}$$

$$\frac{\partial L(data | h_{\theta_0, \theta_1, \theta_2})}{\partial \theta_2} = \frac{r_l}{\theta_2} - \frac{g_l}{1 - \theta_2}$$

$$\theta_2 = \frac{r_l}{r_l + g_l}$$
Discrete Variables

- Maximum Likelihood estimate is the ratio of frequencies.
- **Naive Bayes Model, Bayes Classifier** assumes independent features given the class variable.
  - Calculate prior probability of classes $P(c_i)$
  - For each feature $f$, calculate for each class the probability of this feature $P(f|c_i)$
  - For a new observation of features $f$ predict the most probable class
    $\text{argmax}_{c_i} P(f|c_i) \cdot P(c_i)$. 
- Bayesian Networks learn more complex (in)dependencies between features.
ML Estimate of Gaussian Distribution Parameters

- Assume $x$ to have a Gaussian distribution with unknown parameters $\mu$ and $\sigma$.
- Therefore $h_{\mu,\sigma} = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$.
- We observed $x_1, \ldots, x_n$.
- Log likelihood is:

$$L = \sum_{j=1}^{N} \log \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{(x_j-\mu)^2}{2\sigma^2}}$$

$$= N \cdot (\log \frac{1}{\sqrt{2\pi\sigma}}) - \sum_{j=1}^{N} \frac{(x_j - \mu)^2}{2\sigma^2}$$

- Find the maximum.
Remark: Bayesian Parameter Learning

- We represent probability distribution on parameters.
- For binary features, Beta function is used, $a$ is the number of positive examples, $b$ the number of negative examples.

$$\text{beta}[a, b](\theta) = \alpha \theta^{a-1} (1 - \theta)^{b-1}$$

- (For categorical features, Dirichlet priors and multinomial distribution is used. (Dirichlet-multinomial distribution).
- For gaussian, $\mu$ has gaussian prior, $\frac{1}{\sigma}$ has gamma prior (to stay in exponential family).)
EM - Algorithm

- EM algorithm is used for estimation of unobserved variables (for example, cluster membership).
- It is an iterative algorithm with two steps:
  - **Estimate**, fills in the unobserved data based on current M model, and
  - **Maximize**, finds maximum (log)likelihood model given the data filled in E step.
Reasons for Modelling Unobserved Variables

- It may be useful.
- This variable makes many features conditionally independent (that is, simplifies the model).
- Often, mixtures of Gaussians are used. It is also our example: clustering.
Estimate Step

- I have a model from the previous step (at the beginning, we may choose random cluster centers and/or uniformly distributed values or values based on sample mean and variance.
- Use weighted data, each row $i$ with unobserved variables filled by $j$ is the weight $\gamma_{ij}$.
- Estimate step: For each data row:
  - Calculate the conditional probability of possible values of unobserved variables given the model.
- Maximize step: for some models we know:
  - gaussians - mean and standard deviation are maximum likelihood estimates of $\mu, \sigma$,
  - discrete - the ratios of observed counts
Mixture of Two Gaussians, one input feature $x$

- Model parameters: $\pi, \mu_1, \sigma_1^2, \mu_2, \sigma_2^2$, initialize $\mu$ randomly, $\pi = 0.5$, $\sigma = \text{sample variance}$

- Estimate – step: fill the data, estimate weights of filled samples:

$$\gamma_i = \frac{\pi \phi_{\theta_2}(x_i)}{(1 - \pi) \phi_{\theta_1}(x_i) + \pi \phi_{\theta_2}(x_i)}$$

- Maximize – step: estimate new model,

$$\mu_1 = \frac{\sum_{i=1}^{N} (1 - \gamma_i) x_i}{\sum_{i=1}^{N} (1 - \gamma_i)}$$

$$\sigma_2^2 = \frac{\sum_{i=1}^{N} \gamma_i (x_i - \mu_2)^2}{\sum_{i=1}^{N} \gamma_i}$$

$$\pi = \frac{\sum_{i=1}^{N} \gamma_i}{N}$$

- iterate E–M until convergence.
Mixture of $K$ Gaussians

- Model parameters $\pi_1, \ldots, \pi_k, \mu_1, \ldots, \mu_k, \Sigma_1, \ldots, \Sigma_k$ such that $\sum_{k=1}^{K} \pi_k = 1$.

- Estimate: weights of unobserved 'fill-ins' $k$ of variable $C$:

$$p_{ik} = P(C = k|x_i) = \alpha \cdot P(x_i|C = k) \cdot P(C = k)$$

$$= \frac{\pi_k \phi_{\theta_k}(x_i)}{\sum_{l=1}^{K} \pi_l \phi_{\theta_l}(x_i)}$$

$$p_k = \sum_{i=1}^{N} p_{ik}$$

- Maximize: mean, variance and cluster 'prior' for each cluster $k$:

$$\mu_k \leftarrow \sum_{i} \frac{p_{ik}}{p_k} x_i$$

$$\Sigma_k \leftarrow \sum_{i} \frac{p_{ik}}{p_k} (x_i - \mu_k)(x_i - \mu_k)^T$$

$$\pi_k \leftarrow \frac{p_k}{\sum_{l=1}^{K} p_l}$$
library('mclust')
x=c(-0.39,0.12,0.94,1.67,1.76,2.44,3.72,4.28,
4.92,5.53,0.06,0.48,1.01,1.68,1.80,3.25,
4.12,4.60,5.28,6.22)
mmm=Mclust(x, modelNames='V',G=2)
plot(mmm, what=c('density'))
R – Learned Model

Command:
mmm$parameters

Answer:
$pro [1] 0.5536728 0.4463272
$mean
1 2
1.080118 4.652344
$variance$sigmasq
[1] 0.8064871 0.8240747