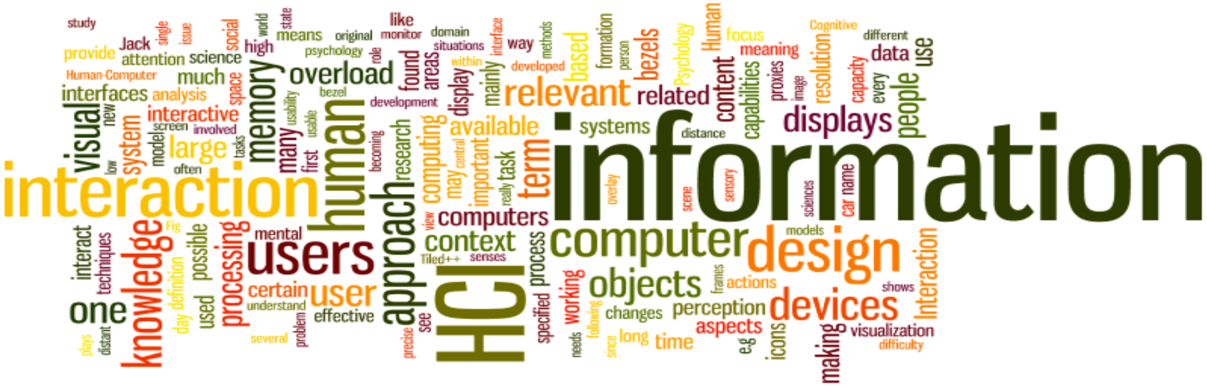
机器学习

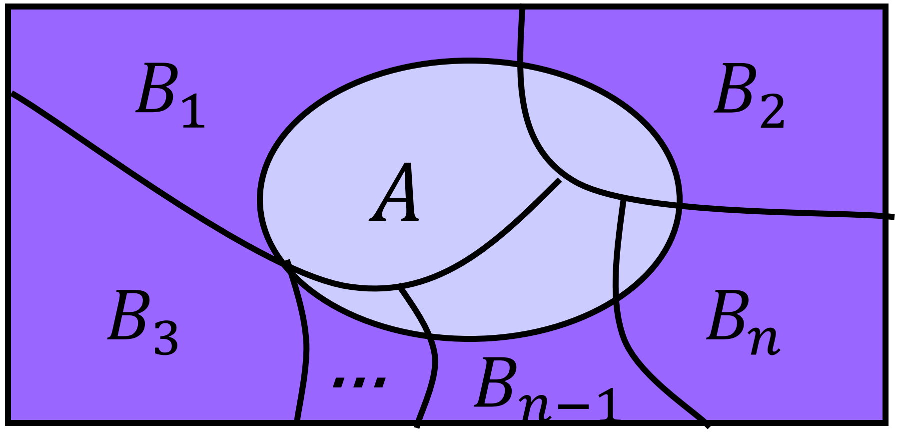
——第8章 贝叶斯学习——



倪张凯
zkni@tongji.edu.cn
https://eezkni.github.io/

全概率公式 Theorem of total probability

$$P(A) = P(B_1)\,P(A/B_1) + P(B_2)\,P(A/B_2) + \cdots + P(B_n)\,P(A/B_n)$$



- If B_1, B_2, … B_n are exhaustive, and mutually exclusive

$\sum_{i=1}^{n} P(B_i) = 1$, $\quad B_i\,B_j = \emptyset,\ i,j = 1,2,\ldots n;$

条件概率 Conditional Probability

- The conditional probability of A given B is the joint probability of A and B, divided by the marginal probability of B.

$$\text{条件概率}\quad p(A \mid B) = \frac{\overbrace{p(A,B)}^{\text{联合概率}}}{\underbrace{p(B)}_{\text{边缘概率（先验概率）}}}$$

独立 Independence

Two events are independent if the occurrence of one in no way affects the probability of the other

- Thus if A and B are statistically independent,

$$p(A\,|\,B)=\frac{p(A,B)}{p(B)}=\frac{p(A)p(B)}{p(B)}=p(A).$$

- However, if A and B are statistically dependent, then

$$p(A\,|\,B)\neq p(A).$$

贝叶斯定理 Bayes' Theorem

- Bayes' theorem is most commonly used to estimate the state of a hidden, causal variable ***h*** based on the measured state of an observable variable ***D***:

Likelihood(似然概率)

Prior
(假设h的先验概率)

$$\boxed{P(h \mid D)} = \frac{P(D \mid h)\, P(h)}{P(D)}$$

Posterior (后验概率)

Evidence
(数据D的先验概率)

条件概率 Conditional Probability

- The conditional probability of A given B is the joint probability of A and B, divided by the marginal probability of B.

联合概率

$$p(A|B) = \frac{p(A,B)}{p(B)}$$

边缘概率（先验概率）

条件概率

贝叶斯定理推导 Bayes' Theorem Deduction

- Bayes' Theorem is simply a consequence of the definition of conditional probabilities:

$$p(A|B)=\frac{p(A,B)}{p(B)}\rightarrow p(A,B)=p(A|B)p(B)$$

$$p(B|A)=\frac{p(A,B)}{p(A)}\rightarrow p(A,B)=p(B|A)p(A)$$

Thus $p(A|B)p(B)=p(B|A)p(A)$

$$\rightarrow \boxed{p(A|B)=\frac{p(B|A)p(A)}{p(B)}}$$

Bayes' Equation

最大后验假设 Maximum a Posteriori Hypothesis (MAP)

In many learning scenarios, the learner considers a set of hypotheses H and is interested in finding the most probable hypothesis $h \in H$ given the observed data D. Any such hypothesis is called *maximum a posteriori hypothesis*.

$$
\begin{aligned}
h_{MAP} &= \arg\max_{h \in H} P(h \mid D) \\
&= \arg\max_{h \in H} \frac{P(D \mid h) P(h)}{P(D)} \\
&= \arg\max_{h \in H} P(D \mid h) P(h)
\end{aligned}
$$

贝叶斯定理 Bayes' Theorem

- Bayes' theorem is most commonly used to estimate the state of a hidden, causal variable ***h*** based on the measured state of an observable variable ***D***:

Likelihood Can be formed

Prior

$$\boxed{P(h \mid D)} = \frac{\boxed{P(D \mid h)}\, P(h)}{P(D)}$$

difficult to estimate directly

Posterior

Evidence

Whereas the posterior *p(h|D)* is often difficult to estimate directly, reasonable models of the likelihood *p(D|h)* can often be formed. This is typically because *h* is causal on *D*.

Thus Bayes' theorem provides a means for estimating the posterior probability of the causal variable *h* based on observations *D*.

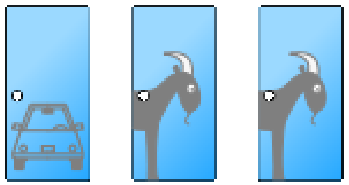
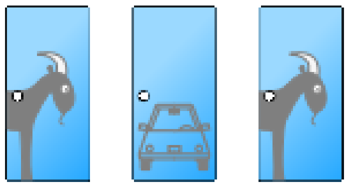
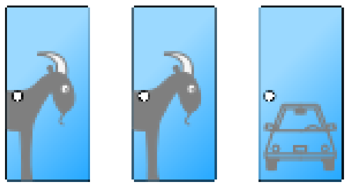
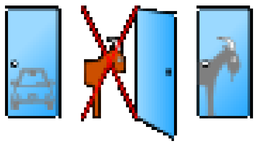
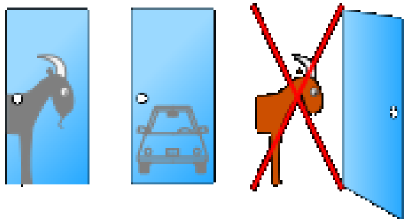
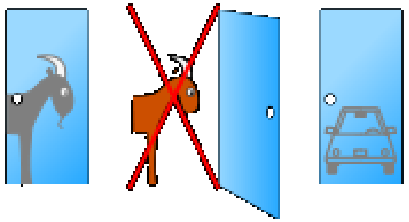
例子：三门问题 Example: The Monty Hall Problem

The Monty Hall problem is a famous probability puzzle, stemming from a television game show scenario.

The question is: Does switching doors increase the contestant's chances of winning the car?

In a study of 228 subjects, 13% chose to switch.

例子：三门问题 Example: The Monty Hall Problem

<table>
<tr><td colspan="2">Car hidden behind Door 1</td><td>Car hidden behind Door 2</td><td>Car hidden behind Door 3</td></tr>
<tr><td colspan="4">Player initially picks Door 1</td></tr>
<tr><td colspan="2"></td><td></td><td></td></tr>
<tr><td colspan="2">Host opens either Door 2 or 3</td><td>Host must open Door 3</td><td>Host must open Door 2</td></tr>
<tr><td></td><td></td><td></td><td></td></tr>
<tr><td>Switching loses with probability 1/6</td><td>Switching loses with probability 1/6</td><td>Switching wins with probability 1/3</td><td>Switching wins with probability 1/3</td></tr>
<tr><td colspan="2">Switching loses with probability 1/3</td><td colspan="2">Switching wins with probability 2/3</td></tr>
</table>

例子：三门问题 Example: The Monty Hall Problem

- Let's assume you initially select Door 1.
- Suppose that Monty then opens Door 2 to reveal a goat.
- We want to calculate the ***posterior*** probability that a car lies behind Door 1 & 3 after Monty has provided these new data.

$$P(h_1/D_2) = \frac{P(D_2/h_1)P(h_1)}{P(D_2)}$$

$$= \frac{P(D_2/h_1)P(h_1)}{P(D_2/h_1)P(h_1) + P(D_2/h_2)P(h_2) + +P(D_2/h_3)P(h_3)}$$

$$= \frac{\frac{1}{2} \times \frac{1}{3}}{\frac{1}{2} \times \frac{1}{3} + 0 \times \frac{1}{3} + 1 \times \frac{1}{3}} \quad = \frac{1}{3}$$

Let h_i represent the state that the car lies behind door $i, i \in [1,2,3]$.
Let D_i represent the event that the Monty opens door $i, i \in [1,2,3]$.

例子：三门问题 Example: The Monty Hall Problem

- Let's assume you initially select Door 1.
- Suppose that Monty then opens Door 2 to reveal a goat.
- We want to calculate the ***posterior*** probability that a car lies behind Door 1&3 after Monty has provided these new data.

$$P(h_3/D_2) = \frac{P(D_2/h_3)P(h_3)}{P(D_2)}$$

$$= \frac{P(D_2/h_3)P(h_3)}{P(D_2/h_1)P(h_1) + P(D_2/h_2)P(h_2) + +P(D_2/h_3)P(h_3)}$$

$$= \frac{1 \times \frac{1}{3}}{\frac{1}{2} \times \frac{1}{3} + 0 \times \frac{1}{3} + 1 \times \frac{1}{3}} = \frac{2}{3}$$

Let h_i represent the state that the car lies behind door $i, i \in [1,2,3]$.
Let D_i represent the event that the Monty opens door $i, i \in [1,2,3]$.

最大后验假设 Maximum a Posteriori Hypothesis (MAP)

In many learning scenarios, the learner considers a set of hypotheses H and is interested in finding the most probable hypothesis $h \in H$ given the observed data D. Any such hypothesis is called *maximum a posteriori hypothesis*.

$$\begin{aligned} h_{MAP} &= \arg\max_{h \in H} P(h \mid D) \\ &= \arg\max_{h \in H} \frac{P(D \mid h)\, P(h)}{P(D)} \\ &= \arg\max_{h \in H} P(D \mid h)\, P(h) \end{aligned}$$

给定训练数据，最可能的假设是什么？

最大后验假设 Maximum a Posteriori Hypothesis (MAP)

In many learning scenarios, the learner considers a set of hypotheses *H* and is interested in finding the most probable hypothesis $h \in H$ given the observed data *D*. Any such hypothesis is called ***maximum a posteriori hypothesis***.

$$
\begin{aligned}
h_{MAP} &= \arg\max_{h \in H} P(h \mid D) \\
&= \arg\max_{h \in H} \frac{P(D \mid h)\, P(h)}{P(D)} \\
&= \arg\max_{h \in H} P(D \mid h)\, \boxed{P(h)}
\end{aligned}
$$

If all of P(h) are equal

Maximum Likelihood hypothesis

$$h_{MLP} = \underset{h \in H}{\operatorname{argmax}}\, P(D/h)$$

例子:疾病诊断 Example： disease diagnosis

- A patient takes a lab test and the result comes back positive.
- It is known that the test returns a correct positive result in only 98% of the cases and a correct negative result in only 97% of the cases; Furthermore, only 0.008 of the entire population has this disease.

1. What is the probability that this patient has cancer?
2. What is the probability that he does not have cancer?
3. What is the diagnosis?

Brute-Force MAP学习算法

Brute-Force MAP Learning algorithm

- For each h in H, calculate the posterior probability

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

- Output the h_{MAP}

$$h_{MAP} = \arg\max_{h \in H} P(h \mid D)$$

- This algorithm requires significant computation, because it need to calculate each P(h|D). This is impractical for large hypothesis spaces.

最大后验假设 Maximum a Posteriori Hypothesis (MAP)

In many learning scenarios, the learner considers a set of hypotheses H and is interested in finding the most probable hypothesis $h \in H$ given the observed data D. Any such hypothesis is called *maximum a posteriori hypothesis*.

$$
\begin{aligned}
h_{MAP} &= \arg\max_{h \in H} P(h \mid D) \\
&= \arg\max_{h \in H} \frac{P(D \mid h)\, P(h)}{P(D)} \\
&= \arg\max_{h \in H} P(D \mid h)\, P(h)
\end{aligned}
$$

给定训练数据，最可能的分类是什么？

例子:最可能的分类 Example：most probable classification

- Given new instance x, what is its most probable classification?
  - $P(h_1|D)=0.4$, $h1(x)=+$
  - $P(h_2|D)=0.3$, $h2(x)=-$
  - $P(h_3|D)=0.3$, $h3(x)=-$

例子:最可能的分类 Example：most probable classification

- Given new instance x, what is its most probable classification?

  - $P(h_1|D)=0.4$, $h1(x)=+$
  - $P(h_2|D)=0.3$, $h2(x)=-$
  - $P(h_3|D)=0.3$, $h3(x)=-$

$$\sum_{h_i \in H} P(+|h_i)P(h_i|D)=0.4$$

$$\sum_{h_i \in H} P(-|h_i)P(h_i|D)=0.6$$

$$h_{MAP}=\arg\max_{h \in H} P(h|D)=\mathrm{h1}(x)=+$$

MAP is not the most probable classification!

最大后验假设 Maximum a Posteriori Hypothesis (MAP)

In many learning scenarios, the learner considers a set of hypotheses *H* and is interested in finding the most probable hypothesis $h \in H$ given the observed data *D*. Any such hypothesis is called *maximum a posteriori hypothesis*.

$$
\begin{aligned}
h_{MAP} &= \arg\max_{h \in H} P(h \mid D) \\
&= \arg\max_{h \in H} \frac{P(D \mid h) P(h)}{P(D)} \\
&= \arg\max_{h \in H} P(D \mid h) P(h)
\end{aligned}
$$

MAP is not the most probable classification!

贝叶斯最优分类器 Bayes Optimal Classifier

- Bayes Optimal Classification: The most probable classification of a new instance is obtained by combining the predictions of all hypotheses, weighted by their posterior probabilities:

$$\underset{v_j \in V}{\operatorname{argmax}} \sum_{h_i \in H} P(v_j / h_i) P(h_i / D)$$

where v is the set of all the values a classification can take and v_j is one possible such classification.

例子:最可能的分类 Example：most probable classification

- Given new instance x, what is its most probable classification?

- $P(h_1|D)=0.4$, $h1(x)=+$
- $P(h_2|D)=0.3$, $h2(x)=-$
- $P(h_3|D)=0.3$, $h3(x)=-$

$$\sum_{h_i \in H} P(+|h_i)P(h_i|D) = 0.4$$

$$\sum_{h_i \in H} P(-|h_i)P(h_i|D) = 0.6$$

$$h_{MAP} = \arg\max_{h \in H} P(h|D) = \mathrm{h1}(x) = +$$

$$\arg\max_{v_j \in \{+,-\}, h_i \in H} \sum_{h_i \in H} P(v_j|h_i)P(h_i|D) = -$$

Bayes Optimal Classification gives us a lower bound on the classification error that can be obtained for a given problem.

贝叶斯最优分类器 Bayes Optimal Classifier

- Bayes Optimal Classification: The most probable classification of a new instance is obtained by combining the predictions of all hypotheses, weighted by their posterior probabilities:

$$\underset{v_j \in V}{\operatorname{argmax}} \sum_{h_i \in H} P(v_j/h_i)P(h_i/D)$$

where v is the set of all the values a classification can take and v_j is one possible such classification.

Unfortunately, Bayes Optimal Classifier is usually too costly to apply!

朴素贝叶斯分类器 Naive Bayes Classifier

- Let each instance *x* of a training set *D* be described by a conjunction of *m* attribute values <$x_1, x_2, .., x_m$> and let *f(x)*, the target function, be such that *f(x)* $\in$ *V*, a finite set.
- **Bayesian Approach:**

$$y = \underset{v_k \in V}{\operatorname{argmax}} P(v_k / x_1, x_2, \ldots, x_m) = \underset{v_k \in V}{\operatorname{argmax}} \frac{P(x_1, x_2, \ldots, x_m / v_k) P(v_k)}{P(x_1, x_2, \ldots, x_m)}$$

$$= \underset{v_k \in V}{\operatorname{argmax}} P(x_1, x_2, \ldots, x_m / v_k) P(v_k)$$

- **Naive Bayesian Approach:** assume that the attribute values are conditionally independent so that

$$P(x_1, x_2, \ldots, x_m / v_k) = \prod_{j=1}^{m} P(x_j / v_k)$$

- **Naive Bayes Classifier:** $y = \underset{v_k \in V}{\operatorname{argmax}} P(v_k) \prod_{j=1}^{m} P(x_j / v_k)$

例子：朴素贝叶斯分类 Example: Naive Bayes classification

Outlook	Temperature	Humidity	Windy	Class
sunny	hot	high	false	N
sunny	hot	high	true	N
overcast	hot	high	false	P
rain	mild	high	false	P
rain	cool	normal	false	P
rain	cool	normal	true	N
overcast	cool	normal	true	P
sunny	mild	high	false	N
sunny	cool	normal	false	P
rain	mild	normal	false	P
sunny	mild	normal	true	P
overcast	mild	high	true	P
overcast	hot	normal	false	P
rain	mild	high	true	N

Consider the weather data and we have to classify the instance:

< *Outlook* = *sunny*, *Temp* = *cool*, *Hum* = *high*, *Wind* = *strong*>

例子：朴素贝叶斯分类 Example: Naive Bayes classification

Outlook	Temperature	Humidity	Windy	Class
sunny	hot	high	false	N
sunny	hot	high	true	N
overcast	hot	high	false	P
rain	mild	high	false	P
rain	cool	normal	false	P
rain	cool	normal	true	N
overcast	cool	normal	true	P
sunny	mild	high	false	N
sunny	cool	normal	false	P
rain	mild	normal	false	P
sunny	mild	normal	true	P
overcast	mild	high	true	P
overcast	hot	normal	false	P
rain	mild	high	true	N

PlayTennis

$P(\text{yes}) = 9/14$
$P(\text{no}) = 5/14$

Outlook	
$P(\text{sunny}\mid\text{yes}) = 2/9$	$P(\text{sunny}\mid\text{no}) = 3/5$
$P(\text{overcast}\mid\text{yes}) = 4/9$	$P(\text{overcast}\mid\text{no}) = 0$
$P(\text{rain}\mid\text{yes}) = 3/9$	$P(\text{rain}\mid\text{no}) = 2/5$
Temp	
$P(\text{hot}\mid\text{yes}) = 2/9$	$P(\text{hot}\mid\text{no}) = 2/5$
$P(\text{mild}\mid\text{yes}) = 4/9$	$P(\text{mild}\mid\text{no}) = 2/5$
$P(\text{cool}\mid\text{yes}) = 3/9$	$P(\text{cool}\mid\text{no}) = 1/5$
Hum	
$P(\text{high}\mid\text{yes}) = 3/9$	$P(\text{high}\mid\text{no}) = 4/5$
$P(\text{normal}\mid\text{yes}) = 6/9$	$P(\text{normal}\mid\text{no}) = 2/5$
Windy	
$P(\text{true}\mid\text{yes}) = 3/9$	$P(\text{true}\mid\text{no}) = 3/5$
$P(\text{false}\mid\text{yes}) = 6/9$	$P(\text{false}\mid\text{no}) = 2/5$

例子：朴素贝叶斯分类 Example: Naive Bayes classification

Consider the weather data and we have to classify the instance:

< *Outlook* = *sunny*, *Temp* = *cool*, *Hum* = *high*, *Wind* = *strong*>

The task is to predict the value (*yes* or *no*) of the concept PlayTennis. We apply the naive bayes rule:

$$y = \underset{v_k \in \{\text{yes},no\}}{\operatorname{argmax}} P(v_\mathrm{k}) \prod_{j=1}^{m} P\left(x_j / v_k\right)$$

$$= \underset{v_k \in \{\text{yes},no\}}{\operatorname{argmax}} P(v_k) P(outlook = sunny/v_k) P(Temp = cool/v_k)$$

$$P(Hum = high/v_k) P(Wind = strong/v_k)$$

例子：朴素贝叶斯分类 Example: Naive Bayes classification

同济大学 TONGJI UNIVERSITY

$$y = \underset{v_k\in\{\text{yes},no\}}{\text{argmax}}\; P(v_\text{k})\prod_{j=1}^{m} P\left(x_j/v_k\right)$$

$$= \underset{v_k\in\{\text{yes},no\}}{\text{argmax}}\; P(v_k)P(outlook = sunny/v_k)P(Temp = cool/v_k)$$

$$P(Hum = high/v_k)P(Wind = strong/v_k)$$

$$P(yes)P(sunny \mid yes)P(cool \mid yes)P(high \mid yes)P(strong \mid yes) = .0053$$

$$P(no)p(sunny \mid no)P(cool \mid no)P(high \mid no)P(strong \mid no) = .0206$$

Thus, the naive Bayes classifier assigns the value ‘*no*’ to PlayTennis!

例子：朴素贝叶斯分类 Example: Naive Bayes classification

$$y = \underset{v_k \in \{\text{yes},no\}}{\text{argmax}} P(v_\text{k}) \prod_{j=1}^{m} P(x_j/v_k)$$

$$= \underset{v_k \in \{\text{yes},no\}}{\text{argmax}} P(v_k)P(outlook = sunny/v_k)P(Temp = cool/v_k)$$

$$P(Hum = high/v_k)P(Wind = strong/v_k)$$

$$P(yes)P(sunny \mid yes)P(cool \mid yes)P(high \mid yes)P(strong \mid yes) = .0053$$

$$P(no)p(sunny \mid no)P(cool \mid no)P(high \mid no)P(strong \mid no) = .0206$$

normalize

$$\frac{0.0206}{0.0206+0.0053} = 0.795$$

例子：文本分类 Example: Learning to classify text

- Target concept, $v_k \in \{like, dislike\}$
- Attributes to represent text documents
  - One attribute per word position in document
  - Vector of words for each document

This is an example document for the naïve bayes classifier. This document contains only one paragraph, or two sentences.

$$y = \operatorname*{argmax}_{v_k \in \{like,dislike\}} P(v_\mathrm{k}) \prod_{j=1}^{19} P\left(x_j = w_j / v_k\right)$$

$$= \operatorname*{argmax}_{v_k \in \{like,dislike\}} P(v_k) P(x_1 = this/v_k) \ldots P(x_{19} = setnences/v_k)$$

例子：文本分类 Example: Learning to classify text

- Target concept, $v_k \in \{like, dislike\}$
- Attributes to represent text documents
  - One attribute per word position in document
  - Vector of words for each document

This is an example document for the naïve bayes classifier. This document contains only one paragraph, or two sentences.

$$y = \underset{v_k \in \{like,dilike\}}{\operatorname{argmax}} P(v_\mathrm{k}) \prod_{j=1}^{19} P\left(x_j = w_j / v_k\right)$$

$$= \underset{v_k \in \{like,dilike\}}{\operatorname{argmax}} P(v_k) P(x_1 = this/v_k) \ldots P(x_{19} = setnences/v_k)$$

Assumption: independent of position

$$= \underset{v_k \in \{like,dilike\}}{\operatorname{argmax}} P(v_k) P(this/v_k) \ldots P(setnences/v_k)$$

例子：文本分类 Example: Learning to classify text

To estimate the probability $P(w_j/v_k)$ we use:

$$P(w_j/v_k) = \frac{\begin{array}{c} the\ number\ of\ times\ word\ w_j \\ in\ all\ the\ \ instances\ whose\ target\ value\ is\ v_k \end{array}}{\begin{array}{c} total\ number\ of\ the\ words \\ in\ all\ the\ \ instances\ whose\ target\ value\ is\ v_k \end{array}}$$

例子：文本分类 Example: Learning to classify text

To estimate the probability $P(w_j/v_k)$ we use:

$$P(w_j/v_k) = \frac{\begin{array}{c} the\ number\ of\ times\ word\ w_j \\ in\ all\ the\ \ instances\ whose\ target\ value\ is\ v_k \end{array}}{\begin{array}{c} total\ number\ of\ the\ words \\ in\ all\ the\ \ instances\ whose\ target\ value\ is\ v_k \end{array}}$$

Laplacian smoothing

$$P(w_j/v_k) = \frac{\begin{array}{c} the\ number\ of\ times\ word\ w_j \\ in\ all\ the\ \ instances\ whose\ target\ value\ is\ v_k \end{array} + \alpha}{\begin{array}{c} total\ number\ of\ the\ words \\ in\ all\ the\ \ instances\ whose\ target\ value\ is\ v_k \end{array} + \alpha * |Vocabulary|}$$

(usually set $\alpha = 1$)

朴素贝叶斯分类器 Naive Bayes Classifier

- Let each instance *x* of a training set *D* be described by a conjunction of *m* attribute values <$x_1, x_2, .., x_m$> and let *f(x)*, the target function, be such that *f(x)* $\in$ *V*, a finite set.
- **Bayesian Approach:**

$$y = \underset{v_k \in V}{\operatorname{argmax}} P(v_k / x_1, x_2, \dots, x_m) = \underset{v_k \in V}{\operatorname{argmax}} \frac{P(x_1, x_2, \dots, x_m / v_k) P(v_k)}{P(x_1, x_2, \dots, x_m)}$$

$$= \underset{v_k \in V}{\operatorname{argmax}} P(x_1, x_2, \dots, x_m / v_k) P(v_k)$$

- **Naive Bayesian Approach:** assume that the attribute values are conditionally independent so that

$$P(x_1, x_2, \dots, x_m / v_k) = \prod_{j=1}^{m} P(x_j / v_k)$$

- **Naive Bayes Classifier:** $y = \underset{v_k \in V}{\operatorname{argmax}} P(v_k) \prod_{j=1}^{m} P(x_j / v_k)$

不同的朴素贝叶斯分类器 Different NBs

- Gaussian NB
  – continuous features

$$P(x_i/v_k) = \frac{1}{\sqrt{2\pi\sigma_k^2}}\exp(-\frac{x_i-\mu_k}{2\sigma_k^2})$$

- Multinomial NB
  – multivariate discrete features

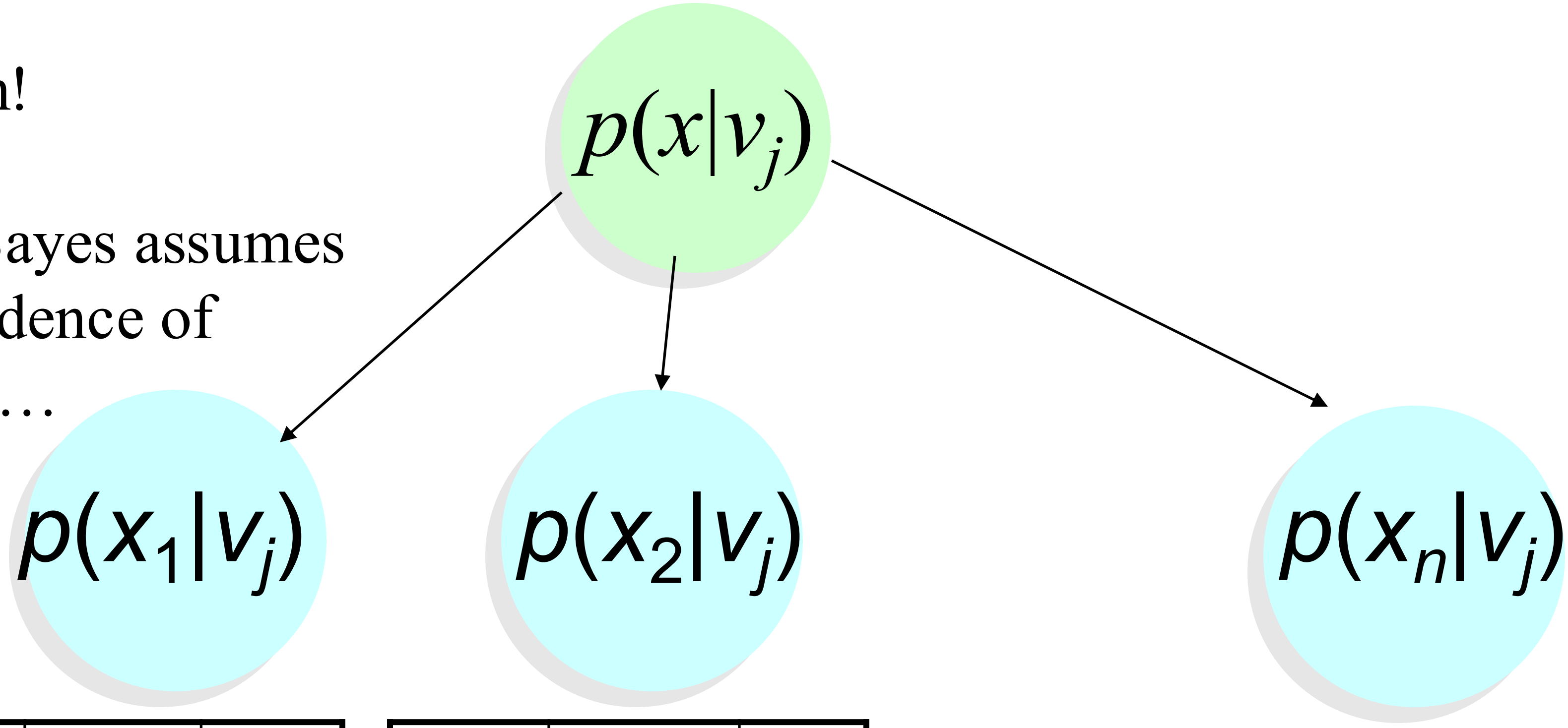
$$P(x_i/v_k) = \frac{N_{x_{i,v_k}} + \alpha}{N_{v_k} + \alpha * n}$$

- Bernoulli NB
  – binary discrete or very sparse multivariate discrete features
- ….

朴素贝叶斯分类器问题 NB’s Problem

Problem!

Naïve Bayes assumes independence of features…



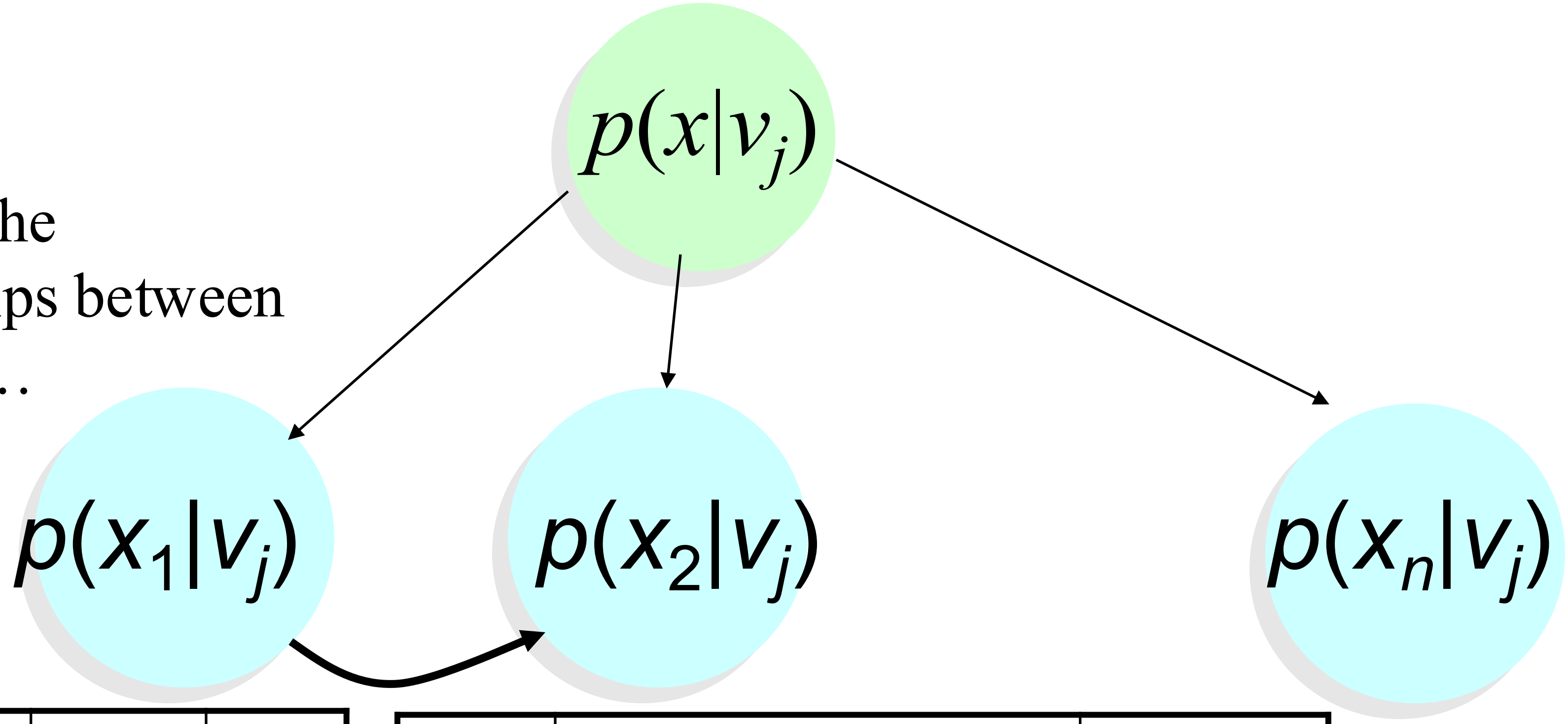
Sex	**Over 6 foot**	
Male	Yes	0.15
	No	0.85
Female	Yes	0.01
	No	0.99

Sex	**Over 200 pounds**	
Male	Yes	0.11
	No	0.80
Female	Yes	0.05
	No	0.95

朴素贝叶斯分类器问题 NB’s Problem

Solution

Consider the relationships between attributes…



Sex	**Over 6 foot**	
Male	Yes	0.15
	No	0.85
Female	Yes	0.01
	No	0.99

Sex	**Over 200 pounds**	
Male	Yes and **Over 6 foot**	0.11
	No and **Over 6 foot**	0.59
	Yes and NOT **Over 6 foot**	0.05
	No and NOT **Over 6 foot**	0.35
Female	Yes and **Over 6 foot**	0.01

贝叶斯置信网 Bayesian Belief Networks(Bayes nets)

- Naive assumption of conditional independency too restrictive
- But it's intractable without some such assumptions…
- Bayesian belief networks describe conditional independence among *subsets* of variables
- Allows combining prior knowledge about causal relationships among variables with observed data

条件独立 Conditional Independence

Definition: X is conditionally independent of Y given Z is the probability distribution governing X is independent of the value of Y given the value of Z, that is, if

$\forall\ x_i, y_j, z_k \quad P(X=x_i|Y=y_j,Z=z_k) = P(X=x_i|Z=z_k)$

or more compactly P(X|Y,Z) = P(X|Z)

Example: *Thunder* is conditionally independent of *Rain* given *Lightning*

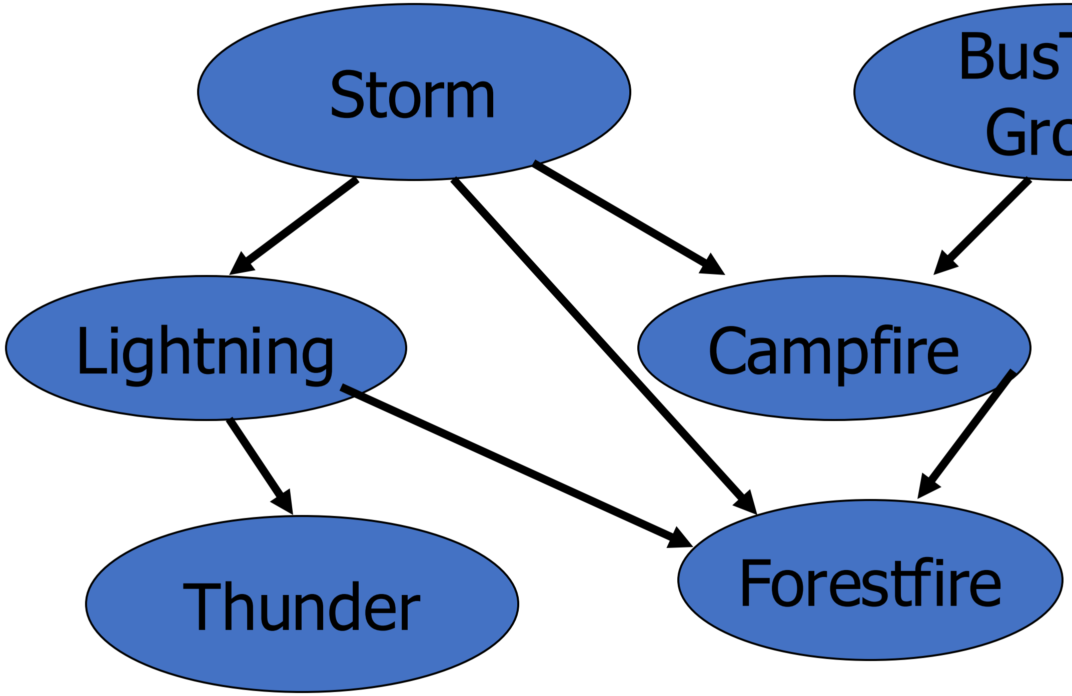
P(*Thunder* |*Rain*, *Lightning*) = P(*Thunder* |*Lightning*)

Notice: P(*Thunder* |*Rain*) ≠ P(*Thunder*)

Naive bayes uses cond. Indep. to justify:

$$\begin{aligned} P(A_1, A_2 \mid V) &= P(A_1 \mid A_2, V) P(A_2 \mid V) \\ &= P(A_1 \mid V) P(A_2 \mid V) \end{aligned}$$

贝叶斯信念网 Bayesian Belief Networks (BBNs)



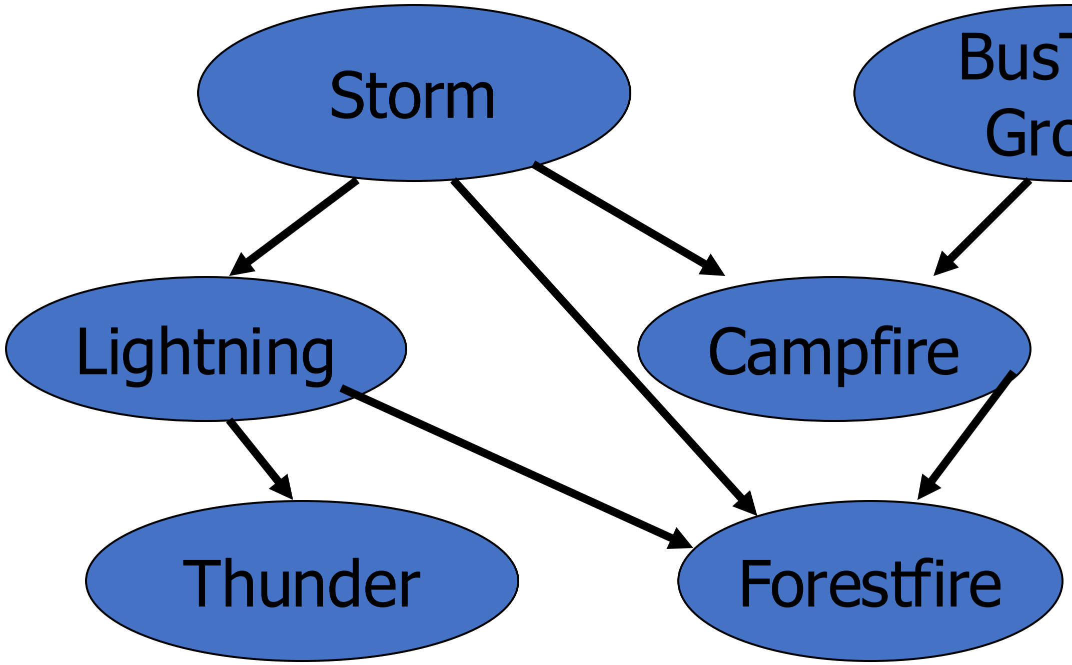
Campfire

	S,B	S,¬B	¬S,B	S, ¬B
C	0.4	0.1	0.8	0.2
¬C	0.6	0.9	0.2	0.8

Network represents a set of conditional independence assertions:

- Each node is conditionally independent of its non-descendants, given its immediate predecessors. (directed acyclic graph)

贝叶斯信念网 Bayesian Belief Networks (BBNs)



Campfire

	S,B	S,¬B	¬S,B	S, ¬B
C	0.4	0.1	0.8	0.2
¬C	0.6	0.9	0.2	0.8

P(C|S,B)

Network represents joint probability distribution over all variables

- P(Storm, BusGroup, Lightning, Campfire, Thunder, Forestfire)
- $P(y_1,\ldots,y_n) = \prod_{i=1}^{n} P(y_i|Parents(Y_i))$
- joint distribution is fully defined by graph plus $P(y_i|Parents(Y_i))$

贝叶斯信念网 Bayesian Belief Networks (BBNs)

1. Graphical model that represents probabilistic relationships among a set of variables.
2. Composed of a Directed Acyclic Graph (DAG) and Conditional Probability Distributions (CPDs).

Key Concept:

1. Encodes causal relationships via its graphical structure.
2. Provides probabilities for each variable given the state of its parent variables.

贝叶斯信念网的应用 Applications of BBNs

Medical Diagnosis:

Diagnose diseases based on the probabilistic relationships between diseases and symptoms.

Recommendation Systems:

Predict preferences or interests based on user behavior and attributes of items.

Risk Assessment:

Evaluate potential risks in fields like finance, insurance, etc.

Fault Diagnosis:

simulate different mechanical components and their interactions, predicting failures and suggesting maintenance actions.

…

贝叶斯信念网的优点和挑战
Advantages and Challenges of BBNs

Advantages:

1. High interpretability due to its graphical structure.
2. Naturally handles uncertainty and missing data.
3. Incorporates prior knowledge seamlessly.

Challenges:

1. Scalability: Inference can be complex and time-consuming for large networks.
2. Data sparsity: Might lead to overfitting when learning parameters for intricate networks with insufficient data.

贝叶斯网的推理 Inference in Bayesian Network

How can one infer the (probabilities of) values of one or more network variables, given observed values of others?

- If only one variable with unknown value, easy to infer it
- Exact inference may not be feasible in large or highly complex networks, hence approximate methods are often favored.

In practice, can succeed in many cases

- Learning the parameter:
  - Maximum Likelihood Estimation (MLE) , Bayesian Estimation (BE), Expectation-Maximization (EM) …
- Learning the Structure (heuristic algorithms)
  - Score-based methods; Constraint-based methods; Hybrid methods…

期望最大化算法 Expectation Maximization Algorithm

The EM (Expectation-Maximization) algorithm is a powerful and iterative approach to statistical estimation in cases where data are incomplete or have some missing or hidden parts.

when to use

- data is only partially observable
- unsupervised clustering: target value unobservable
- supervised learning: some instance attributes unobservable

applications

- training Bayesian Belief Networks
- unsupervised clustering
- learning hidden Markov models
- …

期望最大化算法 Expectation Maximization Algorithm

The algorithm consists of two main steps that are repeated until convergence:

1. **Expectation (E) step**: Given the current estimates of parameters, calculate the expected values of the hidden variables. This step involves creating a function for the expectation of the log-likelihood evaluated using the current estimate for the parameters.
2. **Maximization (M) step**: Maximize the expectation function from the E step to find the new estimates of the parameters. This maximizes the likelihood of the data given the expected values of the hidden variables from the E step.

The idea is that each iteration will increase the likelihood of the data incrementally, and under certain conditions, the algorithm is guaranteed to converge to a (local) maximum.

例子：硬币投掷实验 Example：Coin Tossing Experiment

Esitimate $\theta = <\theta_A, \theta_B>$ when X, Z are known.

		Coin A	Coin B
B	H T T T H H T H T H		5 H, 5 T
A	H H H H T H H H H H	9 H, 1 T	
A	H T H H H H H T H H	8 H, 2 T	
B	H T H T T T H H T T		4 H, 6 T
A	T H H H T H H H T H	7 H, 3 T	
		24 H, 6 T	9 H, 11 T

5 sets, 10 tosses per set

$$\hat{\theta}_A = \frac{\#\text{ of heads using coin A}}{\text{total }\#\text{ of flips using coin A}}$$

$$\hat{\theta}_B = \frac{\#\text{ of heads using coin B}}{\text{total }\#\text{ of flips using coin B}}$$

例子：硬币投掷实验 Example：Coin Tossing Experiment

Esitimate θ=< θ_A, θ_B> when X, Z are known.

		Coin A	Coin B
B	H T T T H H T H T H		5 H, 5 T
A	H H H H T H H H H H	9 H, 1 T	
A	H T H H H H H T H H	8 H, 2 T	
B	H T H T T T H H T T		4 H, 6 T
A	T H H H T H H H T H	7 H, 3 T	
		24 H, 6 T	9 H, 11 T

5 sets, 10 tosses per set

$$\hat{\theta}_A = \frac{\text{\# of heads using coin A}}{\text{total \# of flips using coin A}}$$

$$\hat{\theta}_B = \frac{\text{\# of heads using coin B}}{\text{total \# of flips using coin B}}$$

$$\hat{\theta}_A = \frac{24}{24+6} = 0.80$$

$$\hat{\theta}_B = \frac{9}{9+11} = 0.45$$

- X=(x1,x2, x3,x4,x5), xi is the number of heads observed during the i-th set of tosses
  xi $\in$ {0,1,2,3,4,5,6,7,8,9,10}
- Z =(z1,z2, z3,z4,z5), zi is the identity of the coin used during the i-th set of tosses.
  zi $\in$ {A,B}

例子：硬币投掷实验 Example：Coin Tossing Experiment

Esitimate θ=< θ_A, θ_B> when X is observable, Z is unobservable

	Coin A	Coin B
H T T T H H T H T H		5 H, 5 T
H H H H T H H H H H	9 H, 1 T	
H T H H H H H T H H	8 H, 2 T	
H T H T T T H H T T		4 H, 6 T
T H H H T H H H T H	7 H, 3 T	
	24 H, 6 T	9 H, 11 T

5 sets, 10 tosses per set

$$\hat{\theta}_A = \frac{\text{\# of heads using coin A}}{\text{total \# of flips using coin A}}$$

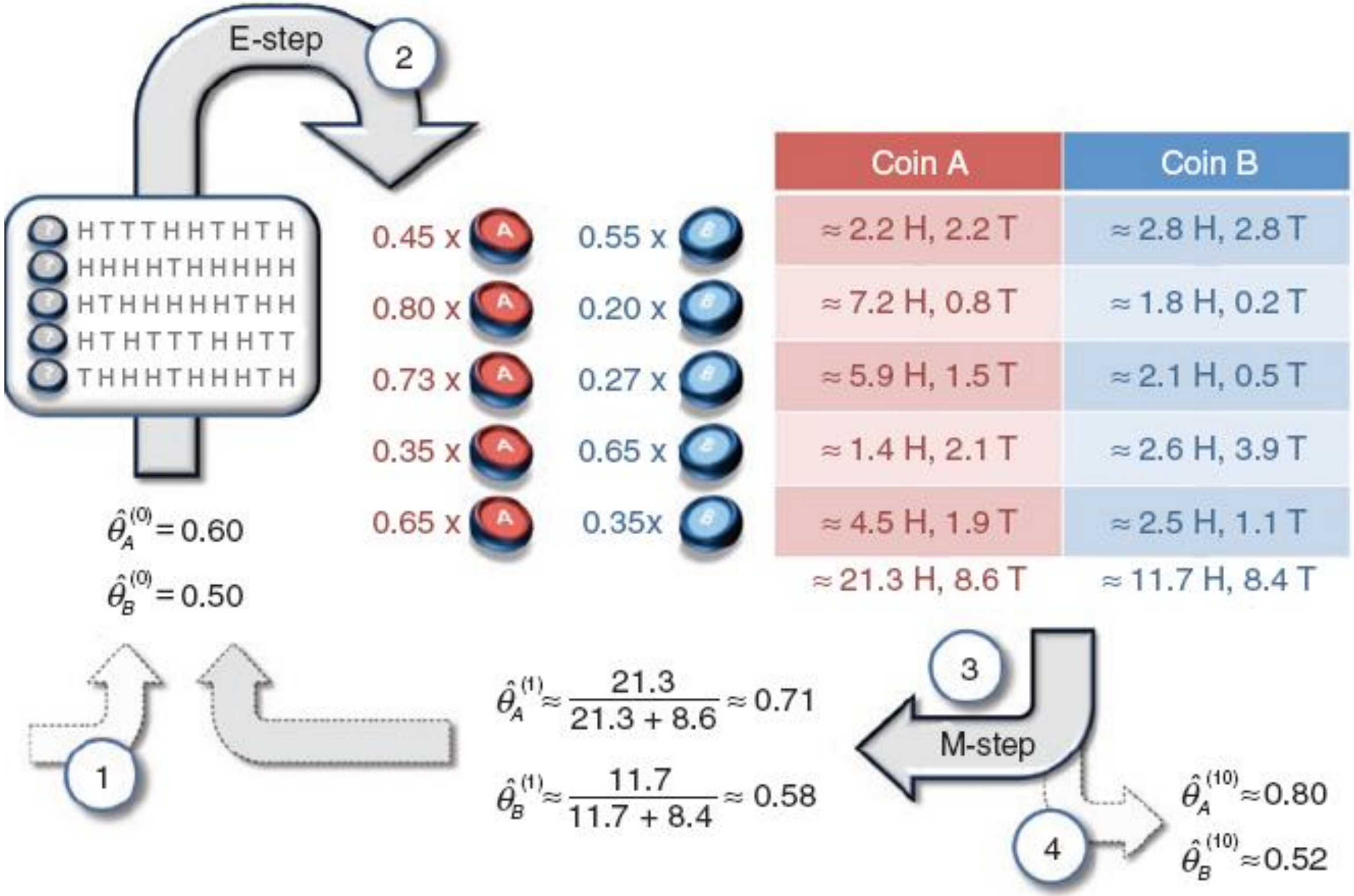
$$\hat{\theta}_B = \frac{\text{\# of heads using coin B}}{\text{total \# of flips using coin B}}$$

- X=(x1,x2, x3,x4,x5), xi is the number of heads observed during the ith set of tosses
  xi $\in$ {0,1,2,3,4,5,6,7,8,9,10}

- Z =(z1,z2, z3,z4,z5), zi is the identity of the coin used during the ith set of tosses.
  zi $\in$ {A,B}

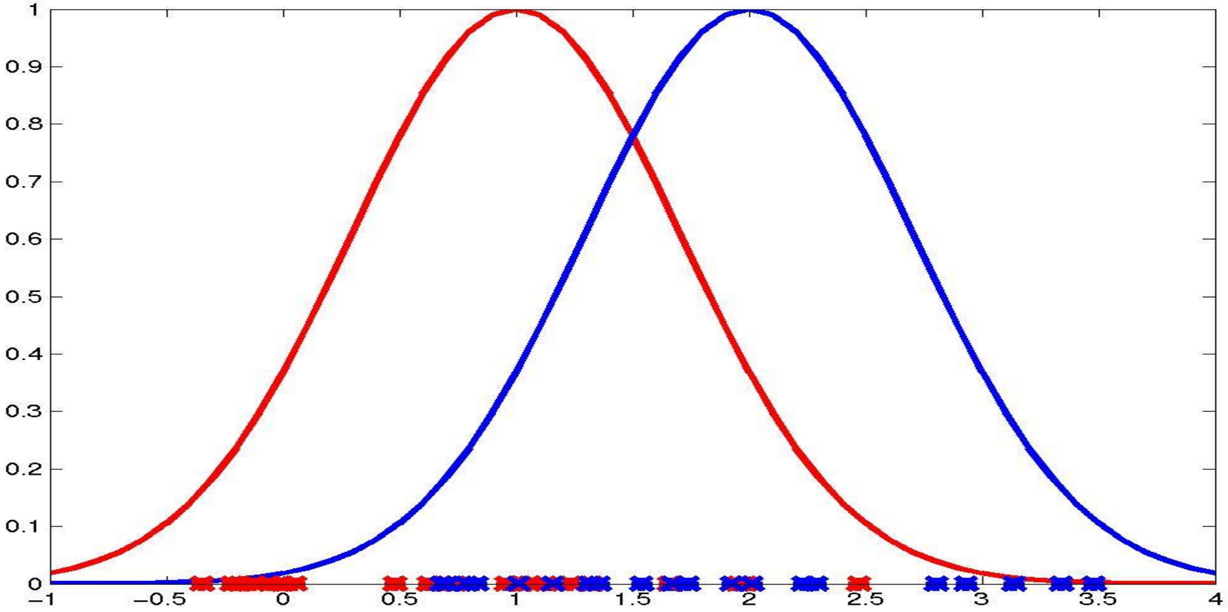
例子：硬币投掷实验 Example：Coin Tossing Experiment

Esitimate θ=< θ_A, θ_B> when X is observable, Z is unobservable



混合高斯模型 Generating Data from Mixture of Gaussians

$$f(x) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$



Each instance x generated by

- choosing one of the k Gaussians at random
- Generating an instance according to that Gaussian

EM算法估计高斯混合模型参数

EM for Estimating GMM Parameters

- EM is iterative technique designed for probabilistic models.

Given:

- instances from X generated by mixture of k Gaussians
- unknown means <$\mu_1,\ldots,\mu_k$ > of the k Gaussians
- don't know which instance x_i was generated by which Gaussian

Determine:

- maximum likelihood estimates of < $\mu_1,\ldots,\mu_k$ >

Think of full description of each instance as y_i =< x_i, z_{i1}, z_{i2} >

- z_{ij} is 1 if x_i generated by j-th Gaussian
- x_i observable
- z_{ij} unobservable

EM算法估计高斯混合模型参数

EM for Estimating GMM Parameters

EM algorithm: pick random initial

h=< μ_1, μ_2 > then iterate

- **E step**:

Calculate the expected value E[z_{ij}] of each hidden variable z_{ij}, assuming the current hypothesis

- h =< μ_1, μ_2 > holds.

$$E[z_{ij}] = \frac{p(x = x_i \mid \mu = \mu_j)}{\sum_{n=1}^{2} p(x = x_i \mid \mu = \mu_n)} = \frac{e^{-\frac{1}{2\sigma^2}(x_i - \mu_j)^2}}{\sum_{n=1}^{2} e^{-\frac{1}{2\sigma^2}(x_i - \mu_n)^2}}$$

- **M step**:

Calculate a new maximum likelihood hypothesis h'=< μ_1', μ_2' > assuming the value taken on by each hidden variable z_{ij} is its expected value E[z_{ij}] calculated in the E-step.

Replace h=< μ_1, μ_2 > by h'=< μ_1', μ_2' >

$$\mu_j = \frac{\sum_{i=1}^{m} E[z_{ij}] x_i}{\sum_{i=1}^{m} E[z_{ij}]}$$