机器学习

苏州大学计算机科学与技术学院
自然语言处理实验室
主讲：周夏冰
邮箱：zhouxiabing@suda.edu.cn

CONTENTS

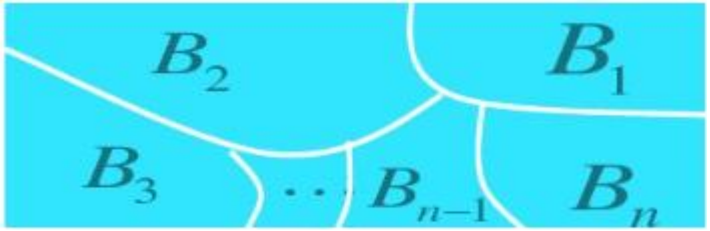
01

预备知识

预备知识

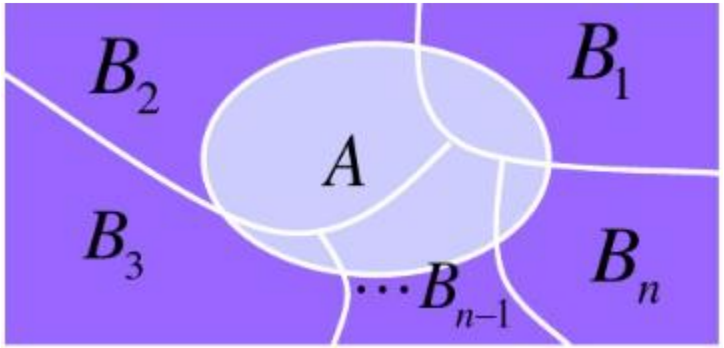
- 全概率公式
  - 完备事件群

$$B_iB_j = \varnothing \quad (i \neq j)$$
$$B_1 + B_2 + \cdots = \Omega$$



- 假设研究事件A，希望求出P(A)

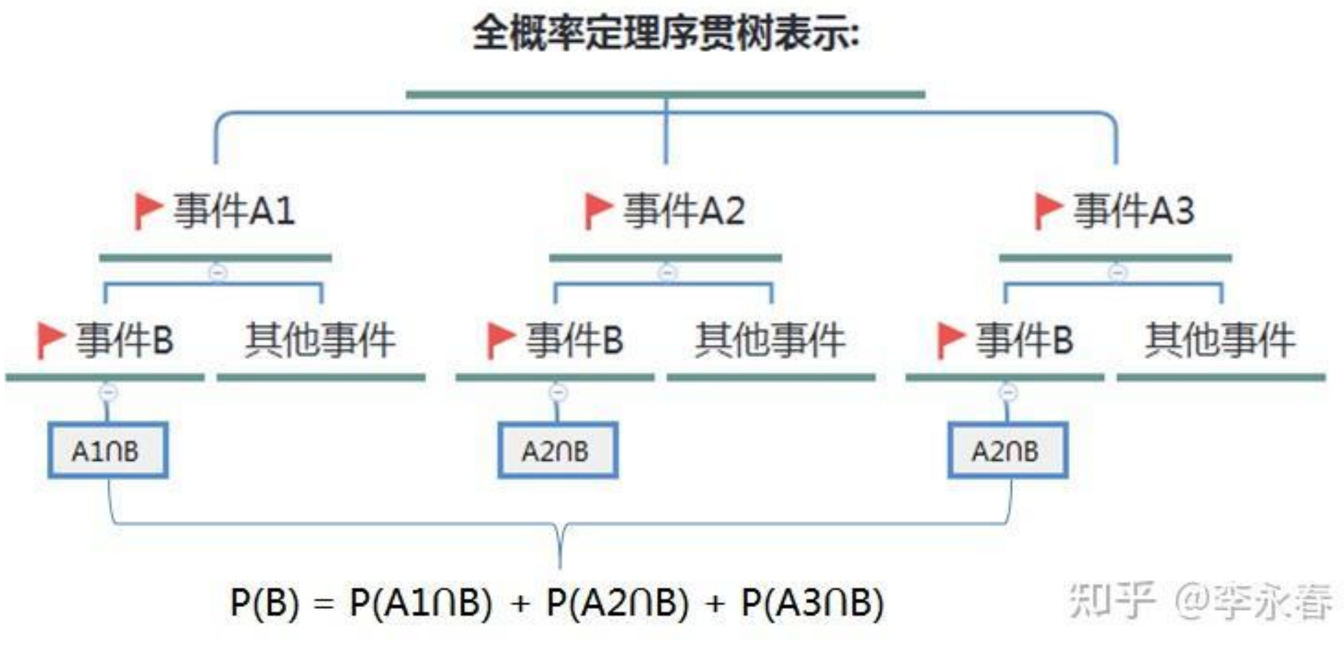
$$\begin{aligned} P(A) &= P(A\Omega) \\ &= P(AB_1 + AB_2 + AB_3 + \cdots) \\ &= P(AB_1) + P(AB_2) + P(AB_3) + \cdots \end{aligned}$$



$$\boxed{P(A) = P(B_1)P(A|B_1) + P(B_2)P(A|B_2) + P(B_3)P(A|B_3) + \cdots}$$

预备知识

- **全概率公式**
  - 全概率公式的主要用途在于它可以将一个复杂事件的概率计算问题，分解为若干个简单事件的概率计算问题，最后应用概率的可加性求出最终结果。



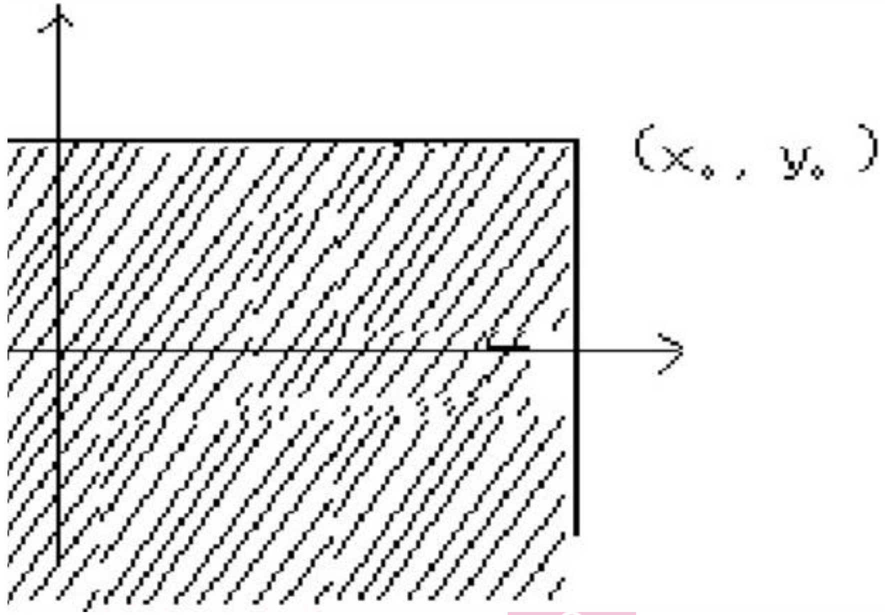
预备知识

- 联合概率密度
  - 两个及以上随机变量组成的随机向量的概率分布
- 离散型：
- $p(X=x_i,Y=y_j)=p_{ij}$
- 边缘分布：
- $p(X=x_i)=\sum_j p_{ij}=\sum_j p(x_i|y_j)p(y_j)$

$Y \backslash X$	x_1	$\cdots$	x_i	$\cdots$	$p_{\cdot j}$
y_1	p_{11}	$\cdots$	p_{i1}	$\cdots$	$p_{\cdot 1}$
$\vdots$	$\vdots$	$\cdots$	$\vdots$	$\cdots$	$\vdots$
y_j	p_{1j}	$\cdots$	p_{ij}	$\cdots$	$p_{\cdot}$
$\vdots$	$\vdots$	$\cdots$	$\vdots$	$\cdots$	$j\ \vdots$
$p_{i\cdot}$	$p_{1\cdot}$	$\cdots$	p_i	$\cdots$	1

预备知识

- **联合概率密度**
  - 两个及以上随机变量组成的随机向量的概率分布
- **连续型：**
- 联合分布函数$F(x,y)=P(X\le x,Y\le y)$
- $F(x,y)=\int_{-\infty}^{x}\int_{-\infty}^{y}f(u,v)dudv$
- 边缘概率：$F(x,\infty)$



预备知识

- 条件概率
  - 条件概率表示在条件Y=b成立的情况下，X=a的概率，记作P(X=a|Y=b)或P(a|b)
- 条件概率，边缘（全）概率，联合概率关系
  - $p(X=a|Y=b)=\frac{P(X=a,Y=b)}{P(Y=b)}$

预备知识

- 贝叶斯公式
  - 条件概率：$P(A|B)=\frac{P(AB)}{P(B)}$
  - 设$P(A)>0$，则$P(AB)=P(B|A)P(A)=P(A|B)P(B)$
  - $\boxed{P(B_i|A)=\frac{P(A|B_i)|P(B_i)}{P(A)}=\frac{P(A|B_i)P(B_i)}{\sum_{j=1}^{N}P(A|B_j)P(B_j)}}$

预备知识

- 由以往的数据分析得到的概率，叫先验概率
- 在得到信息之后再重新加以修正的概率叫做后验概率
- 贝叶斯定理
  - P(A)是A的先验概率或边缘概率
  - P(A|B)是已知B发生后A的条件概率，也由于得知B的取值而被称为A的后验概率
  - 简单的说，贝叶斯定理是基于假设的先验概率、给定假设下观察到不同数据的概率，提供了一种计算后验概率的方法

02

朴素贝叶斯

朴素贝叶斯

- 贝叶斯分类是一类分类算法的总称，这类算法均以贝叶斯定理为基础，故统称为贝叶斯分类
- 朴素贝叶斯分类是贝叶斯分类中最简单，也是常见的一种分类方法
- 基于贝叶斯定理和特征条件独立假设的分类

朴素贝叶斯

- **贝叶斯判定准则：最小化总体风险**

需在每个样本上选择那个能使后验概率最大的类别标记

- 假设损失函数$L(Y, f(X)) = \begin{cases} 1, Y \neq f(X) \\ 0, Y = f(X) \end{cases}$

- 在样本x 上的条件风险： $E_X \sum_{k=1}^{K} [L(c_k, f(X))] P(c_k|X)$

- $f(x) = \underset{y \in \mathcal{Y}}{\operatorname{argmin}} \sum_{k=1}^{K} [L(c_k, f(X))] P(c_k|X = x)$

$$= \underset{y \in \mathcal{Y}}{\operatorname{argmin}} \sum_{k=1}^{K} P(y \neq c_k|X = x) \quad = \underset{y \in \mathcal{Y}}{\operatorname{argmin}} (1 - P(y = c_k|X = x))$$

$$= \underset{y \in \mathcal{Y}}{\operatorname{argmax}} P(y = c_k|X = x)$$

朴素贝叶斯

- 假设c为类别，$x=\{x_1,\cdots,x_d\}$
- 有贝叶斯公式：$P(c|x)=\frac{P(c)P(x|c)}{P(x)}=\frac{P(c)}{P(x)}\prod_{i=1}^{d}P(x_i|c)$
- $f(x)=argmaxP(c)\prod_{i=1}^{d}P(x_i|c)$
- $P(Y=c_k)=\frac{\sum_{i=1}^{N}I(y_i=c_k)}{N},k=1,2,\cdots,K$

目的是：求对于输入x，输出的是哪一个类别，也就是依次考虑$P(Y=1|x),P(Y=2|x),\cdots,P(Y=c_K|x)$哪一个概率最高
对于它们而言，分母都是一样的，所以判断时候可以不考虑分母

离散

$$P(X^j=a_{ij}|Y=c_k)=\frac{\sum_{i=1}^{N}I(x_i^j=a_{ji},y_i=c_k)}{\sum_{i=1}^{N}I(y_i=c_k)}$$

连续

$$P(x_i|c)=\frac{1}{\sqrt{2\pi}\sigma_{c,i}}\exp(-\frac{(x_i-\mu_{c,i})^2}{2\sigma_{c,i}^2})$$

朴素贝叶斯

- 例1

帅？	性格好？	身高？	上进？	嫁与否
帅	不好	矮	不上进	不嫁
不帅	好	矮	上进	不嫁
帅	好	矮	上进	嫁
不帅	好	高	上进	嫁
帅	不好	矮	上进	不嫁
帅	不好	矮	上进	不嫁
帅	好	高	不上进	嫁
不帅	好	中	上进	嫁
帅	好	中	上进	嫁
不帅	不好	高	上进	嫁
帅	好	矮	不上进	不嫁
帅	好	矮	不上进	不嫁

- 男生向女生求婚，男生特点（帅，性格好，矮，不上进），问女生嫁不嫁？

P(嫁|帅，性格好，矮，不上进)?
P(不嫁|帅，性格好，矮，不上进)?

朴素贝叶斯

$P(c)\prod_{i=1}^{d} p(x_i|c)$

- **先验概率** P(嫁)和P(不嫁)　$\frac{1}{2}$
- P(帅，性格好，矮，不上进|嫁)= P(帅|嫁) × P(性格好|嫁) × P(矮|嫁) × P(不上进|嫁)

特征独立

帅？	性格好？	身高？	上进？	嫁与否
帅	不好	矮	不上进	不嫁
不帅	好	矮	上进	不嫁
帅	好	矮	上进	嫁
不帅	好	高	上进	嫁
帅	不好	矮	上进	不嫁
帅	不好	矮	上进	不嫁
帅	好	高	不上进	嫁
不帅	好	中	上进	嫁
帅	好	中	上进	嫁
不帅	不好	高	上进	嫁
帅	好	矮	不上进	不嫁
帅	好	矮	不上进	不嫁

朴素贝叶斯

- **先验概率** P(嫁)和P(不嫁)
- P(帅，性格好，矮，不上进|嫁)= P(帅|嫁) $\times$ P(性格好|嫁) $\times$ P(矮|嫁) $\times$ P(不上进|嫁)

特征独立

- P(嫁|帅，性格好，矮，不上进) $\approx \frac{1}{2} \times \frac{1}{2}$

朴素贝叶斯

- **先验概率** P(嫁)和P(不嫁)
- P(帅，性格好，矮，不上进|嫁)= P(帅|嫁) × P(性格好|嫁) × P(矮|嫁) × P(不上进|嫁)

特征独立

帅？	性格好？	身高？	上进？	嫁与否
帅	不好	矮	不上进	不嫁
不帅	好	矮	上进	不嫁
帅	好	矮	上进	嫁
不帅	好	高	上进	嫁
帅	不好	矮	上进	不嫁
帅	不好	矮	上进	不嫁
帅	好	高	不上进	嫁
不帅	好	中	上进	嫁
帅	好	中	上进	嫁
不帅	不好	高	上进	嫁
帅	好	矮	不上进	不嫁
帅	好	矮	不上进	不嫁

朴素贝叶斯

- **先验概率** $\boldsymbol{P}$(嫁)和$\boldsymbol{P}$(不嫁)
- $\boldsymbol{P}$(帅，性格好，矮，不上进|嫁)= $\boldsymbol{P}$(帅|嫁) × $\boldsymbol{P}$(性格好|嫁) × $\boldsymbol{P}$(矮|嫁) × $\boldsymbol{P}$(不上进|嫁)
- $\boldsymbol{P}$(嫁|帅，性格好，矮，不上进) ≈ $\frac{1}{2} \times \frac{1}{2} \times \frac{5}{6}$

特征独立

朴素贝叶斯

- **先验概率** P(嫁)和P(不嫁)
- P(帅，性格好，矮，不上进|嫁)= P(帅|嫁) × P(性格好|嫁) × P(矮|嫁) × P(不上进|嫁)

特征独立

帅？	性格好？	身高？	上进？	嫁与否
帅	不好	矮	不上进	不嫁
不帅	好	矮	上进	不嫁
$\boxed{\text{帅}}$	$\boxed{\text{好}}$	$\boxed{\text{矮}}$	$\boxed{\text{上进}}$	$\boxed{\text{嫁}}$
不帅	好	高	上进	嫁
帅	不好	矮	上进	不嫁
帅	不好	矮	上进	不嫁
帅	好	高	不上进	嫁
不帅	好	中	上进	嫁
帅	好	中	上进	嫁
不帅	不好	高	上进	嫁
帅	好	矮	不上进	不嫁
帅	好	矮	不上进	不嫁

朴素贝叶斯

- **先验概率** $\boldsymbol{P}$(嫁)和$\boldsymbol{P}$(不嫁)
- $\boldsymbol{P}$(帅，性格好，矮，不上进|嫁)= $\boldsymbol{P}$(帅|嫁) × $\boldsymbol{P}$(性格好|嫁) × $\boldsymbol{P}$(矮|嫁) × $\boldsymbol{P}$(不上进|嫁)
- $\boldsymbol{P}$(嫁|帅，性格好，矮，不上进) $\approx \frac{1}{2} \times \frac{1}{2} \times \frac{5}{6} \times \frac{1}{6}$

特征独立

朴素贝叶斯

- **先验概率** P(嫁)和P(不嫁)
- P(帅，性格好，矮，不上进|嫁)= P(帅|嫁) × P(性格好|嫁) × P(矮|嫁) × P(不上进|嫁)

特征独立

帅？	性格好？	身高？	上进？	嫁与否
帅	不好	矮	不上进	不嫁
不帅	好	矮	上进	不嫁
帅	好	矮	上进	嫁
不帅	好	高	上进	嫁
帅	不好	矮	上进	不嫁
帅	不好	矮	上进	不嫁
帅	好	高	不上进	嫁
不帅	好	中	上进	嫁
帅	好	中	上进	嫁
不帅	不好	高	上进	嫁
帅	好	矮	不上进	不嫁
帅	好	矮	不上进	不嫁

朴素贝叶斯

- **先验概率** P(嫁)和P(不嫁)
- P(帅，性格好，矮，不上进|嫁)= P(帅|嫁) × P(性格好|嫁) × P(矮|嫁) × P(不上进|嫁)
- P(嫁|帅，性格好，矮，不上进) $\approx \frac{1}{2} \times \frac{1}{2} \times \frac{5}{6} \times \frac{1}{6} \times \frac{1}{6} \approx 0.0058$

特征独立

朴素贝叶斯

- **先验概率** $\boldsymbol{P}$(嫁)**和**$\boldsymbol{P}$(不嫁)
- $\boldsymbol{P}$(帅，性格好，矮，不上进|嫁)= $\boldsymbol{P}$(帅|嫁) × $\boldsymbol{P}$(性格好|嫁) × $\boldsymbol{P}$(矮|嫁) × $\boldsymbol{P}$(不上进|嫁)
- $\boldsymbol{P}$(嫁|帅，性格好，矮，不上进) $\approx \frac{\mathbf{1}}{\mathbf{2}} \times \frac{1}{2} \times \frac{5}{6} \times \frac{1}{6} \times \frac{1}{6} \approx \mathbf{0.0058}$
- $\boldsymbol{P}$(不嫁|帅，性格好，矮，不上进) $\approx \mathbf{0.104}$

特征独立

朴素贝叶斯

- 假设
- $\mu = 0.53$
- $\sigma^2 = 0.076$

每个类别都需要计算分布

- $P(\text{长相} = 0.5|\text{嫁}) = \frac{1}{\sqrt{2\pi \times 0.076}} e^{-\frac{(0.5-0.53)^2}{2\times 0.076}}$

$= 5.2817$

长相	结果
0.1	嫁
0.33	嫁
0.4	嫁
0.7	嫁
0.75	嫁
0.9	嫁

朴素贝叶斯

- $\boldsymbol{argmax\, P(c)\prod_{i=1}^{d} p(x_i|c)}$
- 拉普拉斯平滑/修正
  - $p(c)=\frac{|D_c|+\lambda}{|D|+N\lambda}$ 类别数量 $P(嫁)=\frac{6+1}{12+2}=\frac{1}{2}$
  - $p(x_i|c)=\frac{|D_{c,x_i}|+\lambda}{|D_c|+N_i\lambda}$ $P(帅|嫁)=\frac{3+1}{6+2}=\frac{1}{2}$

X_i的取值数量

- 修正了因为训练集不充分导致概率估计为0的情况

文本朴素贝叶斯分类

- 文字表示
  - **One-Hot向量**
    - **词表[absence, abundant, ……, woman, wonder, ……, zoo]**

$$
\begin{array}{ccc}
\text{absence} & \text{woman} & \text{orange} \\
\begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} &
\begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 1 \\ \vdots \\ 0 \end{bmatrix} &
\begin{bmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}
\end{array}
$$

文本朴素贝叶斯分类

- 词袋模型（Bag-of-Words）
  - Bob likes to play basketball, Jim likes too. [0 1 1 0 0 1 2 1 1 1]
  - Bob also likes to play football games. [1 0 1 1 1 0 1 1 1 0]
  - 词汇表：
    - [1: also; 2: basketball; 3. bob; 4: football; 5: games; 6: jim; 7: likes; 8: play; 9: to; 10: too]

文本朴素贝叶斯分类

- **TF-IDF模型**
  - **TF (Term frequency)：词频**

$$w_j = \begin{cases} count(t_j), \text{文本含有词项}j \\ 0, \text{文本不含词项}j \end{cases}$$

  - **IDF (Inverse document frequency)：逆向文件频率**

$$\mathrm{idf}(\mathrm{t}) = \log\frac{1+\mathrm{n}}{1+\mathrm{df}(\mathrm{t})} + 1$$

df(t)代表包含单词t的文档的个数

$$w_j = \begin{cases} count(t_j) \times idf(t_j), \text{文本含有词项}j \\ 0, \text{文本不含词项}j \end{cases}$$

文本朴素贝叶斯分类

- **TF-IDF模型**

	TF	TF-IDF
This is the first document.	[0 1 1 1 0 1 0 1]	[0 1 1.5108 1.2231 0 1 0 1.2231]
This is the second document.	[0 1 0 1 1 1 0 1]	[0 1 0 1.2231 1.9163 1 0 1.2231]
And the third document.	[1 1 0 0 0 1 1 0]	[1.9163 1 0 0 0 1 1.9163 0]
Is this the first document?	[0 1 1 1 0 1 0 1]	[0 1 1.5108 1.2231 0 1 0 1.2231]

- **词汇表：**
  - [1: and; 2: document; 3. first; 4: is; 5: second; 6: the; 7: third; 8: this]

idf [1.9163 1 1.5108 1.2231 1.9163 1 1.9163 1.2231]

文本朴素贝叶斯分类

- **N-gram模型**
  - “明天”“有”“可能”“会”“下雨”
  - 词汇表(n=2)
    - [（“明天”“有”），（“有”“可能”），（“可能”“会”），（“会”“下雨”）]

文本朴素贝叶斯分类

- from sklearn.feature_extraction.text import CountVectorizer
- from sklearn.feature_extraction.text import TfidfTransformer

- 词汇表获得:

```
corpus=[
    'This is the first document.',
    'This is the second document.',
    'And the third document.',
    'Is this the first document?'
]

vectorizer=CountVectorizer()
x=vectorizer.fit_transform(corpus)
word=vectorizer.get_feature_names_out()
```

- x.toarray()
- **Tf-idf转换**

```
transformers=TfidfTransformer()
tfidf=transformers.fit_transform(x)
tfidf.toarray()
```

Sklearn的idf计算会有一个正则化

$$v_{norm} = \frac{v}{\sqrt{v_1^2 + v_2^2 + \ldots + v_n^2}}$$

文本朴素贝叶斯分类

- from sklearn.feature_extraction.text import CountVectorizer
- from sklearn.feature_extraction.text import TfidfTransformer

- 词汇表获得：

```
corpus=[
    'This is the first document.',
    'This is the second document.',
    'And the third document.',
    'Is this the first document?'
]

vectorizer=CountVectorizer()
x=vectorizer.fit_transform(corpus)
word=vectorizer.get_feature_names_out()
```

Sklearn的idf计算会有一个正则化

$$\mathrm{v}_{\mathrm{norm}}=\frac{\mathrm{v}}{\sqrt{\mathrm{v}_1^2+\mathrm{v}_2^2+\ldots+\mathrm{v}_\mathrm{n}^2}}$$

```
[[0.     0.3708 0.5602 0.4535 0.     0.3708 0.     0.4535]
 [0.     0.3397 0.     0.4155 0.651  0.3397 0.     0.4155]
 [0.6269 0.3271 0.     0.     0.     0.3271 0.6269 0.    ]
 [0.     0.3708 0.5602 0.4535 0.     0.3708 0.     0.4535]]
```

Sklearn朴素贝叶斯

- naive_bayes
  - GaussianNB 高斯分布下的朴素贝叶斯
  - BernouliNB 伯努利分布下的朴素贝叶斯
  - MultinomialNB 多项式分布下的朴素贝叶斯

Sklearn朴素贝叶斯

- **naive_bayes**
  - **GaussianNB 高斯分布下的朴素贝叶斯**
    - $p(x_i|y)$ 高斯分布
  - **BernouliNB 伯努利分布下的朴素贝叶斯**
    - 以文档为粒度
  - **MultinomialNB 多项式分布下的朴素贝叶斯**
    - 以单词为粒度

	文档	标签
1	Chinese Beijing Chinese	Yes
2	Chinese Chinese Shanghai	Yes
3	Chinese Macao	Yes
4	Tokyo Japan Chinese	No

Yes: 3个文档；No：1个文档

$p(yes) = \frac{3}{4}$，$p(no) = \frac{1}{4}$

$p(Chinese|yes) = \frac{3+1}{3+2}$

平滑处理，2表示存在或不存在两个值

Sklearn朴素贝叶斯

- **naive_bayes**
  - **GaussianNB 高斯分布下的朴素贝叶斯**
    - $p(x_i|y)$ 高斯分布
  - **BernouliNB 伯努利分布下的朴素贝叶斯**
    - 以文档为粒度
  - **MultinomialNB 多项式分布下的朴素贝叶斯**
    - 以单词为粒度
  - 使用if-idf时，权重是当做出现的次数来使用的

	文档	标签
1	Chinese Beijing Chinese	Yes
2	Chinese Chinese Shanghai	Yes
3	Chinese Macao	Yes
4	Tokyo Japan Chinese	No

Yes: 8个单词；No：3个单词

$p(yes)=\frac{8}{11}$，$p(no)=\frac{3}{11}$

$p(Chinese|yes)=\frac{5+1}{8+6}$

平滑处理，（Chinese, Beijing, Shanghai, Macao, Tokyo, Japan）6个单词

实验作业2

- 朴素贝叶斯实现文本分类任务，并在测试集中给出每个类别的R、P、F1值
  - 数据（资料）：4news-bydata-train/4news-bydata-test, 英文文本四分类数据集

rec.sport.baseball #运动——篮球
sci.space #科学技术——太空
talk.politics.guns #政治——枪支
talk.politics.mideast #政治——中东

```
From: tedward@cs.cornell.edu (Edward [Ted] Fischer)
Subject: Re: Rickey Henderson
Organization: Cornell Univ. CS Dept, Ithaca NY 14853
Distribution: usa
Lines: 12

In article <1993Apr5.173500.26383@ra.msstate.edu> js1@Isis.MsState.Edu (Jiann-m
Su) writes:
>I say buy out Henderson's contract and let him go bag groceries.  Next
>season, you'll be able to sign him for nothing.  That goes for any bitching
>ball player.

I doubt Henderson would clear waivers.  And if he did, he would
instantly be signed for the major league minimum, with Oakland picking
up the remaining $3 million tab.

Some GMs value on-field performance too...

-Valentine
```

提交实验报告pdf
包含实验设置（不同的文本表示方法，不同的朴素贝叶斯函数，数据、参数等）、实验分析、结果展示、必要设计模块解释

提交截止时间4.25，pdf上传到学习通对应题目位置即可）

03

贝叶斯网络

贝叶斯网络(Bayesian Network)

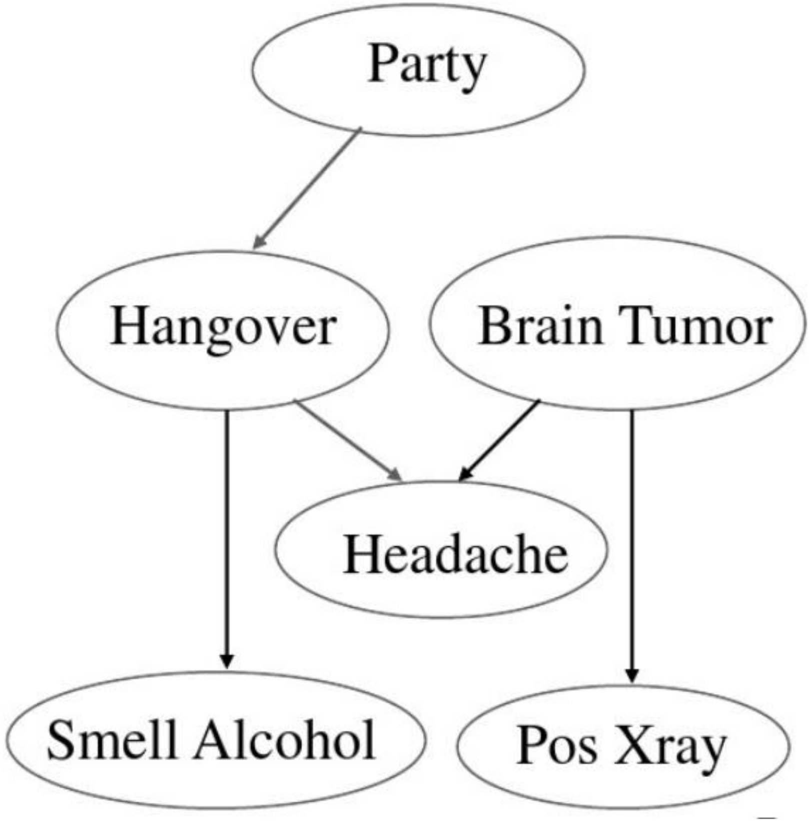
- 贝叶斯网络也成为信念网络(belief network)
- 贝叶斯网络借助有向无环图来刻画属性之间的依赖关系，并使用条件概率表来表述属性的联合概率分布
- 贝叶斯网络是将概率统计应用于复杂领域进行不确定性推理和数据分析的工具
- BN是一种系统地描述随机变量之间关系的工具，建立BN的目的主要是进行概率推理

贝叶斯网络

- 概率图模型
- 有向图模型（贝叶斯网络）
  - 隐马尔可夫模型，马尔可夫随机过程
- 无向图模型（马尔可夫网络）
  - 条件随机场

贝叶斯网络

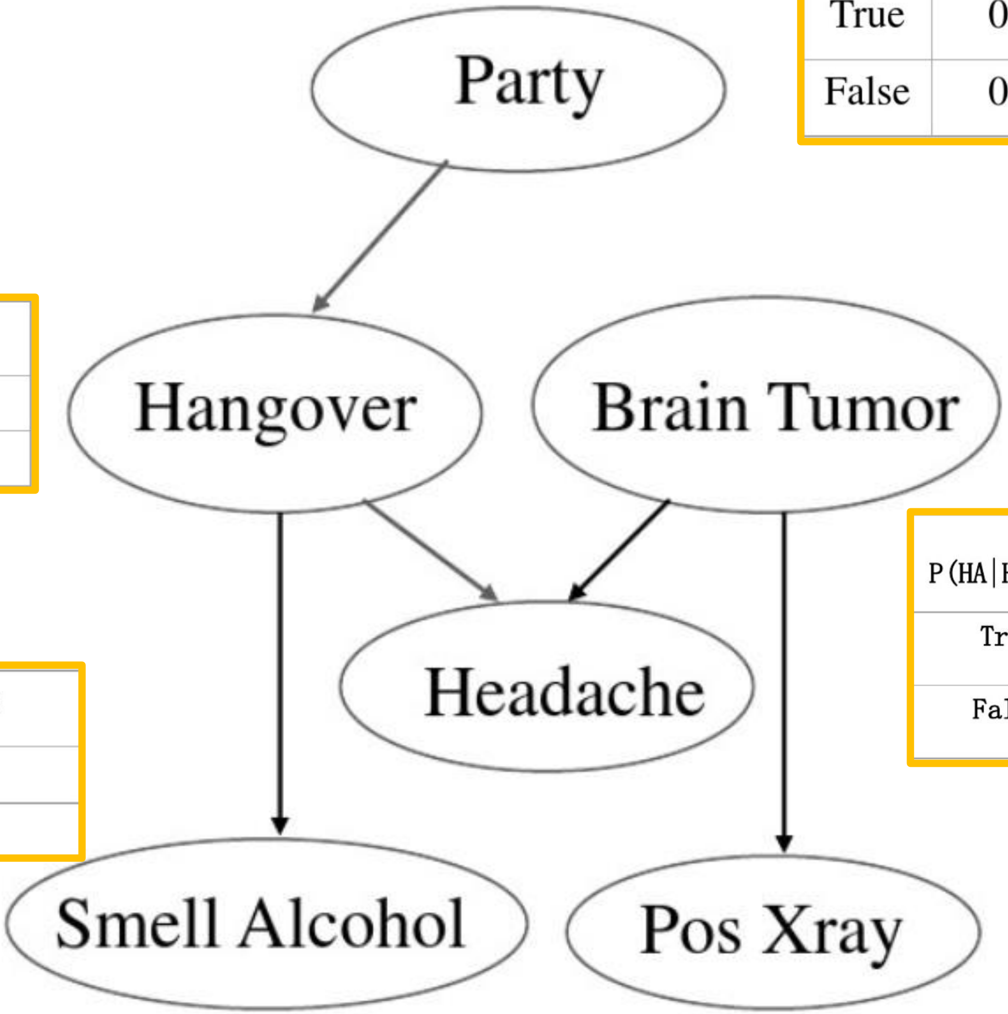
- 如图
  - 参加晚会(party)
  - 宿醉(Hangover)
  - 头疼(Headache)
  - 患脑瘤(Brain tumor)
  - 有酒精味(Smell alcohol)
  - X射线检查呈阳性(Pos Xray)



贝叶斯网络

- 条件概率表

	P(PT)	P(BT)
True	0.200	0.001
False	0.800	0.999



P(HO|PT)	PT=True	PT=False
True	0. 700	0
False	0. 300	1. 000

P(HA|HO, BT)	HO=True BT=True	HO=True BT=False	HO=False BT=True	HO=False BT=False
True	0. 990	0. 700	0. 900	0. 020
False	0. 010	0. 300	0. 100	0. 980

P(SA|HO)	HO=True	HO=False
True	0. 800	0. 100
False	0. 200	0. 900

P(PX|BT)	BT=True	BT=False
True	0. 980	0. 010
False	0. 020	0. 990

贝叶斯网络

- 贝叶斯网络结果有效地表达了属性间地条件独立性，给定父节点集，贝叶斯网络假设每个属性与它的非后裔属性独立
- $P(x_1, x_2, \cdots, x_d) = \prod_{i=1}^{d} P(x_i | \pi_i)$

贝叶斯网络

- 已知结构，预测/推断
- 结构学习

贝叶斯网络——预测

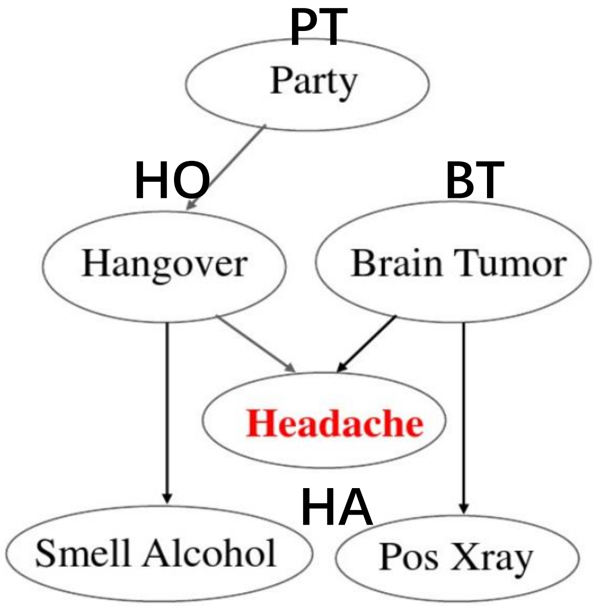
- 自顶向下
  - 已知某些原因节点的情况，可以预测结果结点的概率
    - 参加晚会情况下，头疼发生的概率

贝叶斯网络——预测

- 例：已知参加聚会且没有脑瘤（P(PT)=1, P(-BT)=1），计算HA的概率

P(HO|PT)	PT=True	PT=False
True	0.700	0
False	0.300	1.000

P(HA|HO,BT)	HO=True BT=True	HO=True BT=False	HO=False BT=True	HO=False BT=False
True	0.990	0.700	0.900	0.020
False	0.010	0.300	0.100	0.980



- $P(+HA) = P(+HA|+HO)P(+HO) + P(+HA|-HO)P(-HO)$

$$= 0.7 \times 0.7 + 0.02 \times 0.3 = 0.55$$

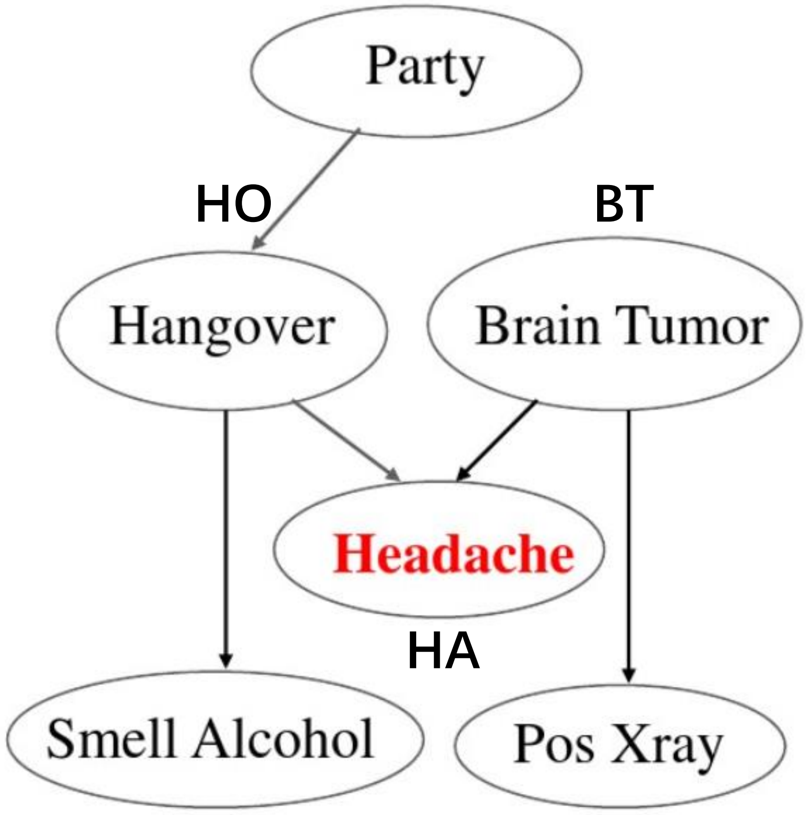
贝叶斯网络——预测

- 自顶向下
  - 已知某些原因节点的情况，可以预测结果结点的概率
    - 参加晚会情况下，头疼发生的概率
  - 在不知任何节点信息的情况下，可以预测网络某个节点发生的概率
    - 即使不知道任何节点发生与否的信息，仍然可以计算节点HA发生的概率

贝叶斯网络——预测

- 例2：计算HA的概率

P(HA|HO,BT)	HO=True		HO=False	
	BT=True	BT=False	BT=True	BT=False
True	0.990	0.700	0.900	0.020
False	0.010	0.300	0.100	0.980

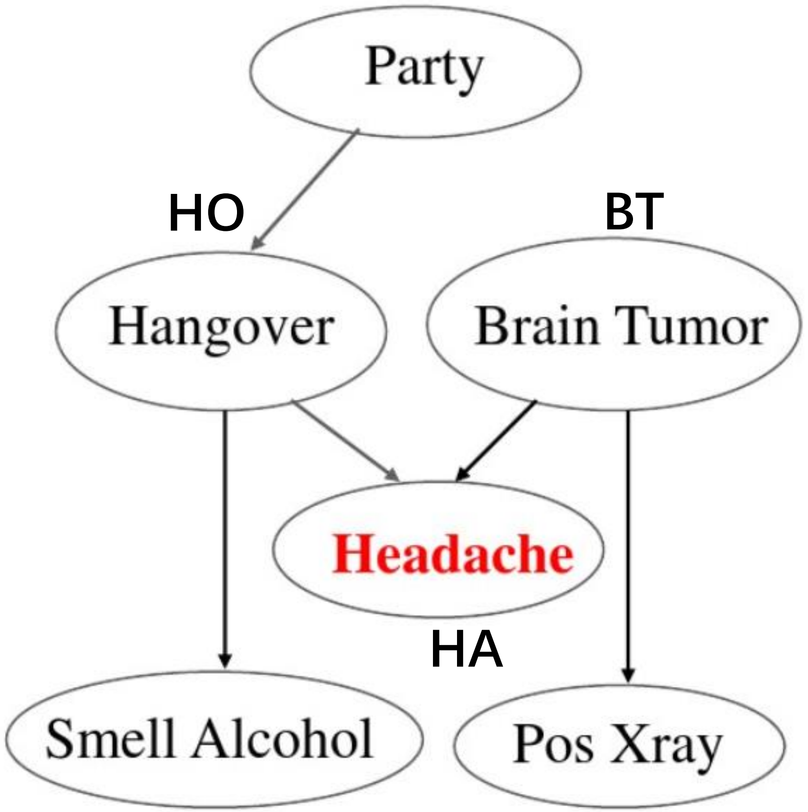


- $P(+HA) = P(+HO)P(+BT) \times 0.99 + P(+HO) \times P(-BT) \times 0.7 + P(-HO) \times P(+BT) \times 0.9 + P(-HO) \times P(-BT) \times 0.02$

贝叶斯网络——预测

- 例2：计算HA的概率

	P(PT)	P(BT)
True	0.200	0.001
False	0.800	0.999



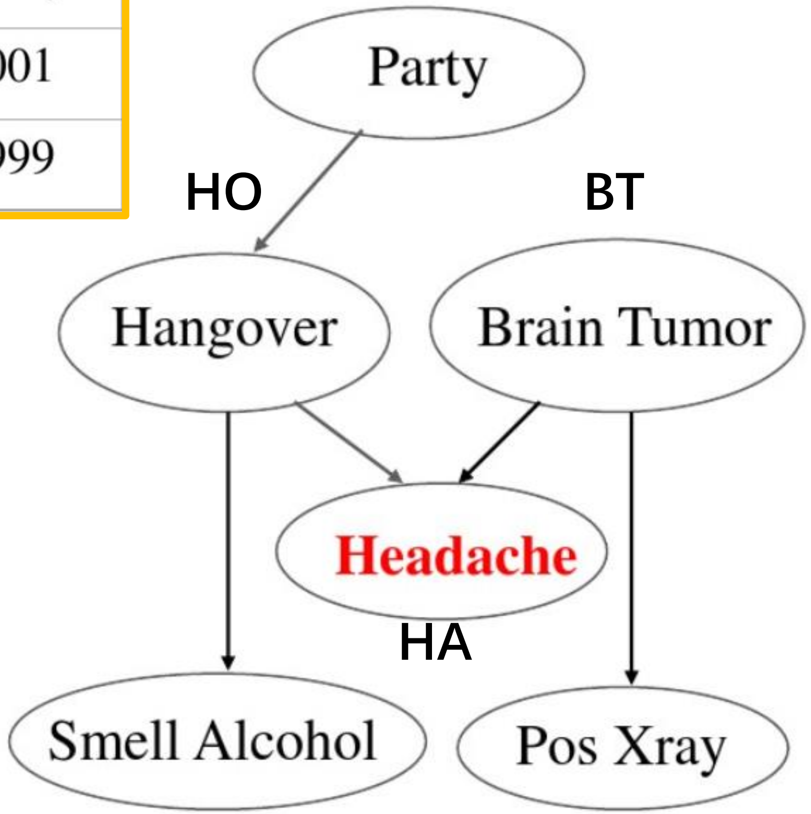
- $P(+HA) = P(+HO)P(+BT) \times 0.99 + P(+HO) \times P(-BT) \times 0.7 + P(-HO) \times P(+BT) \times 0.9 + P(-HO) \times P(-BT) \times 0.02$

贝叶斯网络—预测

	P(PT)	P(BT)
True	0.200	0.001
False	0.800	0.999

- 例2：计算HA的概率

P(HO|PT)	PT=True	PT=False
True	0. 700	0
False	0. 300	1. 000

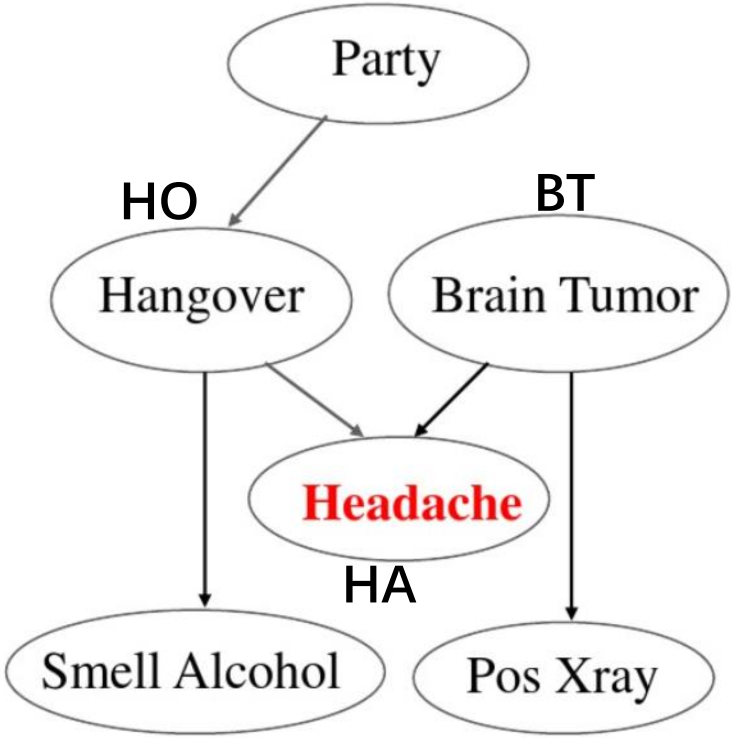


- $P(+HO) = P(+PT) \times 0.7 + P(-PT) \times 0 = 0.2 \times 0.7 = 0.14$
- $P(-HO) = 1 - 0.14 = 0.86$

贝叶斯网络—预测

- 例2：计算HA的概率

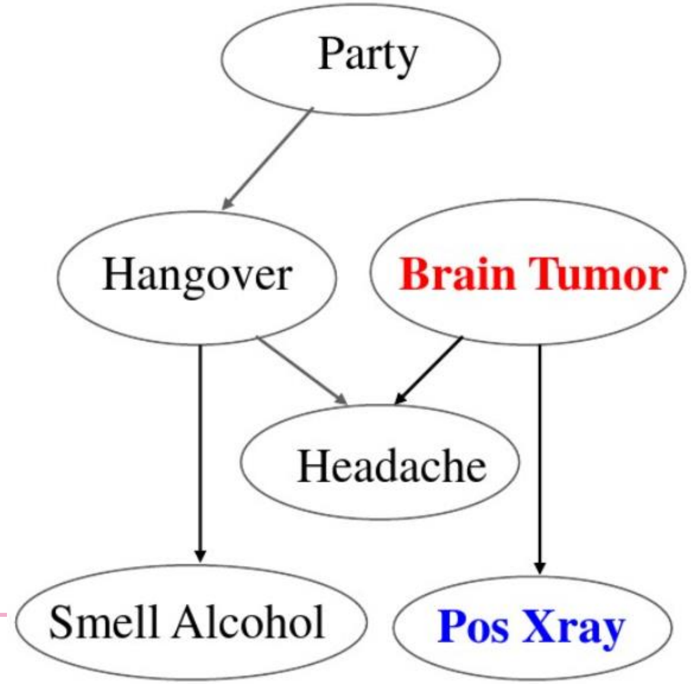
P(HA|HO, BT)	HO=True BT=True	HO=True BT=False	HO=False BT=True	HO=False BT=False
True	0.990	0.700	0.900	0.020
False	0.010	0.300	0.100	0.980



- $P(+HA) = P(+HO)P(+BT) \times 0.99 + P(+HO) \times P(-BT) \times 0.7 + P(-HO) \times P(+BT) \times 0.9 + P(-HO) \times P(-BT) \times 0.02$
- $= 0.14 \times 0.001 \times 0.99 + 0.14 \times 0.999 \times 0.7 + 0.86 \times 0.001 \times 0.9 + 0.86 \times 0.999 \times 0.02$
- $= 0.116$
- $P(-HA) = 1 - 0.116 = 0.884$

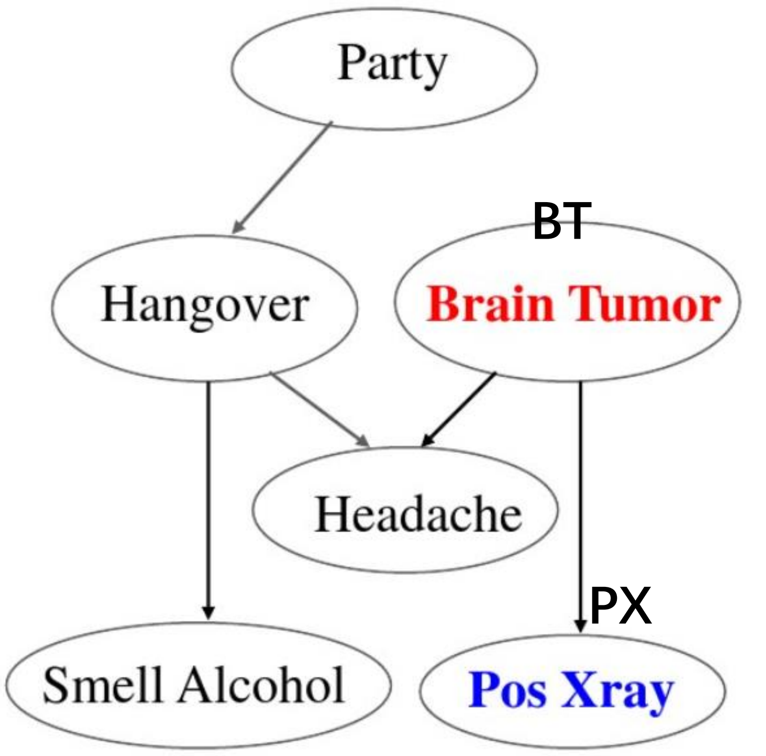
贝叶斯网络—推断

- 自底向上
  - 已知结果节点发生的情况，来推断条件节点发生的概率
- 例4：已知X光检查呈阳性，患脑瘤的概率



贝叶斯网络——推断

- 例4 X射线成阳性，患脑瘤的概率是多少
- P(+BT|+PX)
- $P(+BT|+PX) = \frac{P(+PX|+BT)P(+BT)}{P(+PX)}$
- $P(+PX) = P(+PX|+BT) \times P(+BT) + P(+PX|-BT) \times P(-BT)$

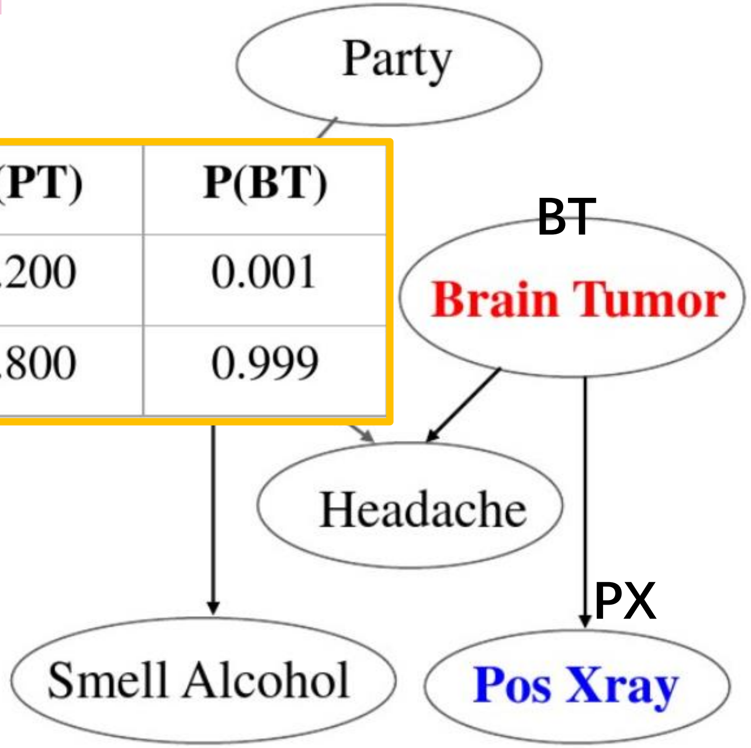


贝叶斯网络—推断

P(PX|BT)	BT=True	BT=False
True	0.980	0.010
False	0.020	0.990

- 例 少

	P(PT)	P(BT)
True	0.200	0.001
False	0.800	0.999



- $\mathbf{P(+BT|+PX)}$

- $P(+BT|+PX) = \dfrac{P(+PX|+BT)P(+BT)}{P(+PX)}$

- $P(+PX) = P(+PX|+BT) \times P(+BT) + P(+PX|-BT) \times P(-BT)$

$$= 0.98 \times 0.001 + 0.01 \times 0.999 = 0.011$$

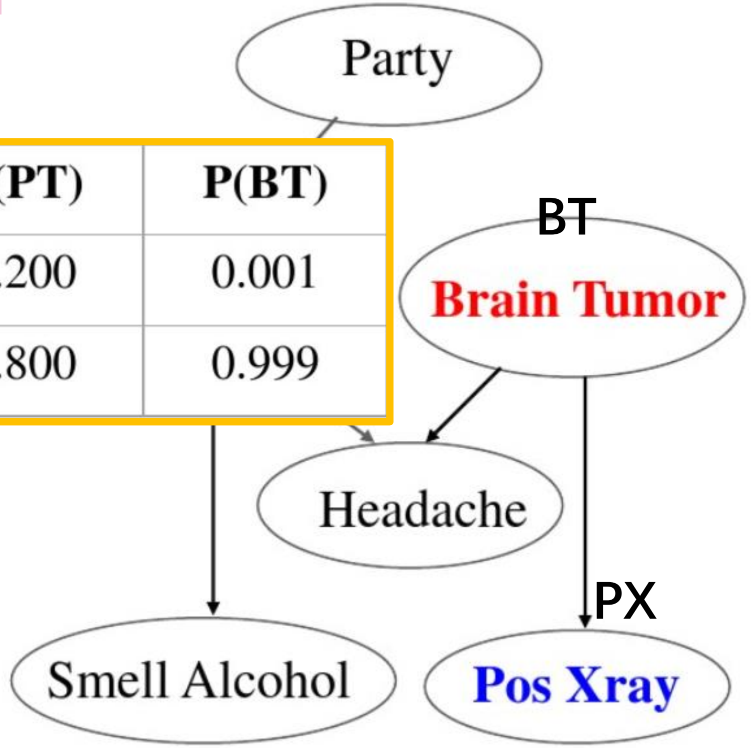
贝叶斯网络—推断

- 例

P(PX|BT)	BT=True	BT=False
True	0. 980	0. 010
False	0. 020	0. 990

- P(

	P(PT)	P(BT)
True	0.200	0.001
False	0.800	0.999



- $P(+BT|+PX) = \frac{P(+PX|+BT)P(+BT)}{P(+PX)}$

- $P(+PX) = P(+PX|+BT) \times P(+BT) + P(+PX|-BT) \times P(-BT)$

$$= 0.98 \times 0.001 + 0.01 \times 0.999 = 0.011$$

- $P(+BT|+PX) = \frac{0.98 \times 0.001}{0.011} = 0.891$

课堂练习

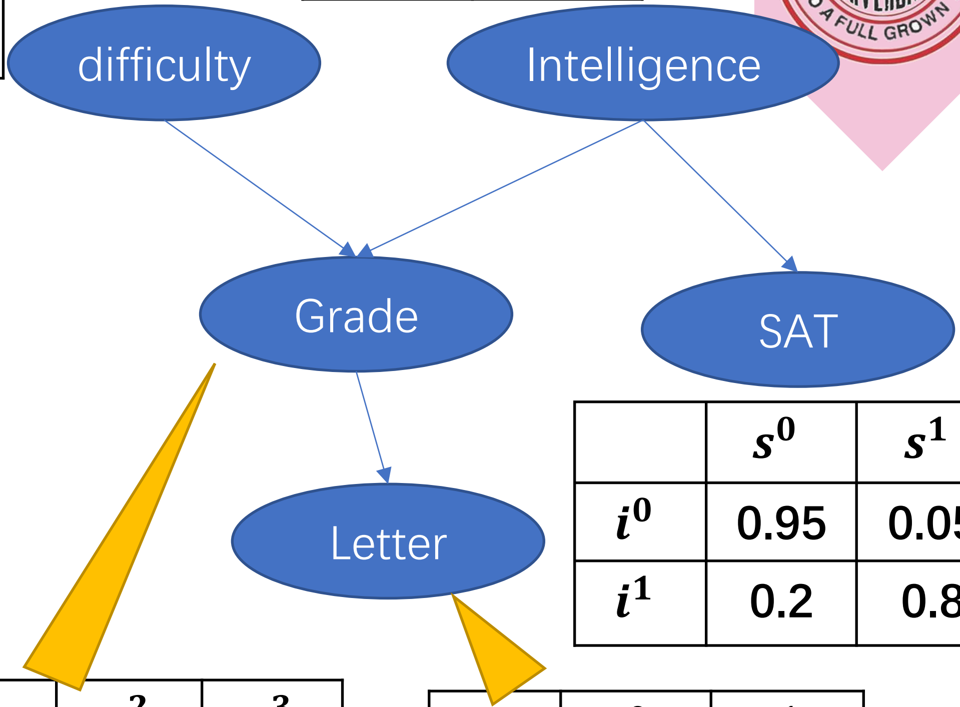
（1）学生George有多大可能从课程Econ101教授那里获得一封好的推荐信(l^1)

（2）若George不聪明(i^0)，概率会降到多少

（3）George去招聘，招聘官相信George有30%的高智商，但看到Econ101的课程得了g^3，判断George具有高智商的概率

d^0	d^1
0.6	0.4

i^0	i^1
0.7	0.3



	s^0	s^1
i^0	0.95	0.05
i^1	0.2	0.8

	g^1	g^2	g^3
i^0,d^0	0.3	0.4	0.3
i^0,d^1	0.05	0.25	0.7
i^1,d^0	0.9	0.08	0.02
i^1,d^1	0.5	0.3	0.2

	l^0	l^1
g^1	0.1	0.9
g^2	0.4	0.6
g^3	0.99	0.01

贝叶斯网络推断

- 当网络结点多，连接稠密，难以进行精确推断，通常此时采用近似推断，通过降低精度，达到有限时间内求解
- 如吉布斯采样