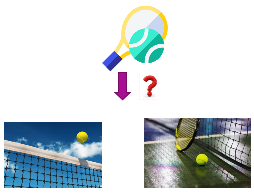
贝叶斯分类器

刘杰
人工智能学院

主要内容

- 贝叶斯决策论
- 朴素贝叶斯
- 文本分类实例

今天的天气是否适合打网球?



定义随机变量: y_0=Yes y_1=No

先验概率（Prior Probability）

- 先验或先验概率反映了我们在实际观察之前对某种状态的预期
- 在这个例子里，也就是适宜或不适宜打球天气的概率
- 比如在A地，温和的天气占大多数；在B地，阴雨天气居多；
- 先验概率记作：$P(y = yi)$或$P(yi)$

$$1 = \sum_{i=1}^{C} \mathrm{P}(y_i)$$

基于先验的决策

- 决策规则是基于输入所采取的特定行动
- 我们是否可以基于先验做出决策?
  - 北京的秋天天气晴朗
  - 中老年容易出现某些疾病
  - 。。。
- 可以，但是局限很大
  - 总是做出同样的预测
  - 如果先验概率是均匀的，那么规则效果不佳
  - 无法利用更多的信息

引入特征

- 特征：观测变量
- 特征空间：进行观测值采样的空间
- Tennis：

Day	Outlook	Temperature	Humidity	Wind	Play Tennis ?
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
3	Overcast	Hot	High	Weak	Yes
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	Yes
6	Rain	Cool	Normal	Strong	No
7	Overcast	Cool	Normal	Strong	Yes
8	Sunny	Mild	High	Weak	No
9	Sunny	Cool	Normal	Weak	Yes
10	Rain	Mild	Normal	Weak	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No

后验概率（Posterior Probability）

- 后验概率：给定观测向量x，某个特定类别的概率$P(y|\boldsymbol{x})$
- 贝叶斯定理

$$P(y,\boldsymbol{x}) = P(y|\boldsymbol{x})P(\boldsymbol{x}) = P(\boldsymbol{x}|y)P(y)$$

$$p(y|\boldsymbol{x}) = \frac{P(\boldsymbol{x}|y)P(y)}{p(\boldsymbol{x})}$$

$$= \frac{p(\boldsymbol{x}|y)P(y)}{\sum_i p(\boldsymbol{x}|yi)p(y_i)}$$

最大后验概率（MAP）

- 因此，我们希望最大化后验概率的类别作为预测结果

$$y^* = \arg\max_i P(y_i|\boldsymbol{x})$$

$$y^* = \begin{cases} y_1 & if\ P(y_1|\boldsymbol{x}) > P(y_2|\boldsymbol{x}) \\ y_2 & if\ P(y_2|\boldsymbol{x}) > P(y_1|\boldsymbol{x}) \end{cases}$$

风险

- 那么我们犯错的概率有多大?

$$P(err|x) = \begin{cases} P(y_2|\boldsymbol{x}) & if\ 决策为 y_1 \\ P(y_1|\boldsymbol{x}) & if\ 决策为 y_2 \end{cases}$$

$$P(err|\boldsymbol{x}) = \min[P(y_1|\boldsymbol{x}), P(y_2|\boldsymbol{x})]$$

损失

- 错误的分类会带来损失
  - 把病人误诊为健康
  - 把正常人误诊为病人
- 不同的错误带来的损失可能不同，记作λ_{ij}

条件风险

- 条件风险（期望损失）

$$\mathrm{R}(y_i|x) = \sum_{j=1}^{n} \lambda_{ij} P(yj|\boldsymbol{x})$$

- 0-1条件风险

$$\mathrm{R}(y_i|x) = 1 - P(yi|\boldsymbol{x})$$

条件风险

- 条件风险（期望损失）

$$\mathrm{R}(y_i|x) = \sum_{j=1}^{n} \lambda_{ij} P(yj|\boldsymbol{x})$$

- 0-1条件风险

$$\mathrm{R}(y_i|x) = 1 - P(yi|\boldsymbol{x})$$

- 贝叶斯最优分类： $h^*(\boldsymbol{x}) = \arg\max_{y\in Y} P(y|\boldsymbol{x})$

朴素贝叶斯

Likelihood　　Class Prior Probability

$$P(c\mid x)=\frac{P(x\mid c)P(c)}{P(x)}$$

Posterior Probability　　Predictor Prior Probability

$$P(c\mid \mathrm{X}) = P(x_1\mid c)\times P(x_2\mid c)\times\cdots\times P(x_n\mid c)\times P(c)$$

- Applies to learning tasks where each instance x is described by a conjunction of attribute values and where the target function $f(x)$ can take on any value from some finite set V
- Training examples are described by $<a_1, a_2, \ldots, a_n>$

Bayesian approach

$$\begin{aligned} v_{MAP} &= \underset{v_j\in V}{argmax}\ P(v_j|a_1,a_2,\ldots,a_n) \\ &= \underset{v_j\in V}{argmax}\ \frac{P(a_1,a_2,\ldots,a_n|v_j)P(v_j)}{P(a_1,a_2,\ldots,a_n)} \\ &= \underset{v_j\in V}{argmax}\ P(a_1,a_2,\ldots,a_n|v_j)P(v_j) \end{aligned}$$

朴素贝叶斯

- 训练数据集：

$$T=\{(x_1,y_1),(x_2,y_2),\cdots,(x_N,y_N)\}$$

- 由X和Y的联合概率分布P(X, Y)独立同分布产生
- 朴素贝叶斯通过训练数据集学习联合概率分布P(X,Y)，
  - 即先验概率分布：　$P(Y=c_k), k=1,2,\ldots,K$
  - 及条件概率分布：

$$P(X=x|Y=c_k)=P(X^{(1)}=x^{(1)},\ldots,X^{(n)}=x^{(n)}|Y=c_k), k=1,2,\ldots,K$$

  - 注意：*条件概率为指数级别的参数*：

$$K\prod_{j=1}^{n}S_j \qquad x_j\in\{a_{j1},a_{j2},\ldots,a_{jS_j}\}$$

基本方法

- 条件独立性假设：

$$\begin{aligned} P(X=x|Y=c_k) &= P(X^{(1)}=x^{(1)},\ldots,X^{(n)}=x^{(n)}|Y=c_k) \\ &= \prod_{j=1}^{n}P(X^{(j)}=x^{(j)}|Y=c_k) \end{aligned}$$

- “朴素”贝叶斯名字由来，牺牲分类准确性。
- 贝叶斯定理：$P(Y=c_k|X=x)=\dfrac{P(X=x|Y=c_k)P(Y=c_k)}{\sum_k P(X=x|Y=c_k)P(Y=c_k)}$
- 代入上式:$P(Y=c_k|X=x)=\dfrac{P(Y=c_k)\prod_j P(X^{(j)}=x^{(j)}|Y=c_k)}{\sum_k P(Y=c_k)\prod_j P(X^{(j)}=x^{(j)}|Y=c_k)}$

基本方法

- 贝叶斯分类器：

$$y=f(x)=\arg\max_{c_k}\frac{P(Y=c_k)\prod_j P(X^{(j)}=x^{(j)}|Y=c_k)}{\sum_k P(Y=c_k)\prod_j P(X^{(j)}=x^{(j)}|Y=c_k)}$$

- 分母对所有c_k都相同：

$$y=\arg\max_{c_k}P(Y=c_k)\prod_j P(X^{(j)}=x^{(j)}|Y=c_k)$$

后验概率最大化的含义：

朴素贝叶斯法将实例分到后验概率最大的类中，等价于期望风险最小化，假设选择0-1损失函数：$f(X)$为决策函数

$$L(Y,f(X))=\begin{cases}1, & Y\neq f(X)\\ 0, & Y=f(X)\end{cases}$$

联合分布P(X, Y)的期望风险函数：

$$R_{exp}(f)=E[L(Y,f(X))]$$

取条件期望风险：

$$R_{exp}(f)=E_X\sum_{k=1}^{K}[L(c_k,f(X))]P(c_k|X)$$

后验概率最大化的含义：

只需对$X = x$逐个极小化，得：

$$
\begin{aligned}
f(x) &= \arg\min_{y\in\mathcal{Y}} \sum_{k=1}^{K} L(c_k, y) P(c_k|X = x) \\
&= \arg\min_{y\in\mathcal{Y}} \sum_{k=1}^{K} P(y \neq c_k|X = x) \\
&= \arg\min_{y\in\mathcal{Y}} (1 - P(y = c_k|X = x)) \\
&= \arg\max_{y\in\mathcal{Y}} P(y = c_k|X = x)
\end{aligned}
$$

推导出后验概率最大化准则：

$$f(x) = \arg\max_{c_k} P(c_k|X = x)$$

朴素贝叶斯法的参数估计

应用极大似然估计法估计相应的概率：

先验概率$P(Y = c_k)$的极大似然估计是：

$$P(Y = c_k) = \frac{\sum_{i=1}^{N} I(y_i = c_k)}{N}, k = 1, 2, ..., K$$

设第j个特征$x^{(j)}$可能取值的集合为：$\{a_{j1}, a_{j2}, ..., a_{jS_j}\}$

条件概率的极大似然估计：

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

$$j = 1, 2, ..., n; l = 1, 2, ..., S_j; k = 1, 2, ..., K$$

朴素贝叶斯法的参数估

- 学习与分类算法Naïve Bayes Algorithm：
- 输入：
  - 训练数据集
    $$T = \{(x_1, y_1), (x_2, y_2), \cdots, (x_N, y_N)\}$$
  - $x_i^{(j)}$ 第i个样本的第j个特征
    $$x_i = (x_i^{(1)}, x_i^{(2)}, ..., x_i^{(n)})^T$$
  - a_{jl} 第j个特征可能取的第l个值
    $$x_i^{(j)} \in \{a_{j1}, a_{j2}, ..., a_{(jS_i)}\}$$
- 输出：
  - x的分类 $y_i \in \{c_1, c_2, ..., c_K\}$

朴素贝叶斯法的参数估

- 步骤
- 1、计算先验概率和条件概率

$$P(y = c_k) = \frac{\sum_{i=1}^{N} I(y_i = c_k)}{N}, k = 1, 2, ..., K$$

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

$$j = 1, 2, ..., n; l = 1, 2, ..., S_j; k = 1, 2, ..., K$$

朴素贝叶斯法的参数估

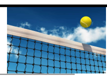
- 步骤
- 2、对于给定的实例 $x = (x_i^{(1)}, x_i^{(2)}, ..., x_i^{(n)})^T$
- 计算

$$P(Y = c_k) \prod_{j=1}^{n} P(X^{(j)} = x^{(j)}|Y = c_k), k = 1, 2, ..., K$$

- 3、确定x的类别

$$y = \arg\max_{c_k} P(Y = c_k) \prod_{j=1}^{n} P(X^{(j)} = x^{(j)}|Y = c_k)$$

Can we play tennis today?

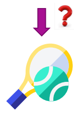


假设我们有一张表格，决定在某些情况下我们是否应该打网球。

这些可能是天气状况；温度；湿度和风力

Day	Outlook	Temperature	Humidity	Wind	Play Tennis ?
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
3	Overcast	Hot	High	Weak	Yes
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	Yes
6	Rain	Cool	Normal	Strong	No
7	Overcast	Cool	Normal	Strong	Yes
8	Sunny	Mild	High	Weak	No
9	Sunny	Cool	Normal	Weak	Yes
10	Rain	Mild	Normal	Weak	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No

X = (**Outlook** = Sunny, **Temperature** = Cool, **Humidity** = High, **Wind** = Strong)



例子

Day	Outlook	Temperature	Humidity	Wind	PlayTennis
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
3	Overcast	Hot	High	Weak	Yes
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	Yes
6	Rain	Cool	Normal	Strong	No
7	Overcast	Cool	Normal	Strong	Yes
8	Sunny	Mild	High	Weak	No
9	Sunny	Cool	Normal	Weak	Yes
10	Rain	Mild	Normal	Weak	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No

Outlook

	Yes	No	P(yes)	P(no)
Sunny	2	3	2/9	3/5
Overcast	4	0	4/9	0/5
Rainy	3	2	3/9	2/5
Total	9	5	100%	100%

Temperature

	Yes	No	P(yes)	P(no)
Hot	2	2	2/9	2/5
Mild	4	2	4/9	2/5
Cool	3	1	3/9	1/5
Total	9	5	100%	100%

例子

Day	Outlook	Temperature	Humidity	Wind	PlayTennis
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
3	Overcast	Hot	High	Weak	Yes
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	Yes
6	Rain	Cool	Normal	Strong	No
7	Overcast	Cool	Normal	Strong	Yes
8	Sunny	Mild	High	Weak	No
9	Sunny	Cool	Normal	Weak	Yes
10	Rain	Mild	Normal	Weak	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No

Humidity

	Yes	No	P(yes)	P(no)
High	3	4	3/9	4/5
Normal	6	1	6/9	1/5
Total	9	5	100%	100%

Wind

	Yes	No	P(yes)	P(no)
Weak	6	2	6/9	2/5
Strong	3	3	3/9	3/5
Total	9	5	100%	100%

Outlook

	Yes	No	P(yes)	P(no)
Sunny	2	3	2/9	3/5
Overcast	4	0	4/9	0/5
Rainy	3	2	3/9	2/5
Total	9	5	100%	100%

Temperature

	Yes	No	P(yes)	P(no)
Hot	2	2	2/9	2/5
Mild	4	2	4/9	2/5
Cool	3	1	3/9	1/5
Total	9	5	100%	100%

Prior

Play		P(Yes)/P(No)
Yes	9	9/14
No	5	5/14
Total	14	100%

Humidity

	Yes	No	P(yes)	P(no)
High	3	4	3/9	4/5
Normal	6	1	6/9	1/5
Total	9	5	100%	100%

Wind

	Yes	No	P(yes)	P(no)
Weak	6	2	6/9	2/5
Strong	3	3	3/9	3/5
Total	9	5	100%	100%

x = (Sunny, Hot, Normal, Weak)

$$y^* = \arg\max_{y \in \{yes, no\}} P(y)P(Sunny|y)P(Hot|y)P(Normal|y)P(Weak|y)$$

Outlook

	Yes	No	P(yes)	P(no)
Sunny	2	3	2/9	3/5
Overcast	4	0	4/9	0/5
Rainy	3	2	3/9	2/5
Total	9	5	100%	100%

Temperature

	Yes	No	P(yes)	P(no)
Hot	2	2	2/9	2/5
Mild	4	2	4/9	2/5
Cool	3	1	3/9	1/5
Total	9	5	100%	100%

Prior

Play		P(Yes)/P(No)
Yes	9	9/14
No	5	5/14
Total	14	100%

Humidity

	Yes	No	P(yes)	P(no)
High	3	4	3/9	4/5
Normal	6	1	6/9	1/5
Total	9	5	100%	100%

Wind

	Yes	No	P(yes)	P(no)
Weak	6	2	6/9	2/5
Strong	3	3	3/9	3/5
Total	9	5	100%	100%

x = (Sunny, Hot, Normal, Weak)

$$P(yes)P(sunny|yes)P(Hot|yes)P(Normal|yes)P(Weak|yes)$$

9/14 * 2/9 * 2/9 * 6/9 * 6/9 = 0.0141

$$P(no)P(sunny|no)P(Hot|no)P(Normal|no)P(Weak|no)$$

5/14 * 3/5 * 2/5 * 1/5 * 2/5 = 0.0069

Outlook

	Yes	No	P(yes)	P(no)
Sunny	2	3	2/9	3/5
Overcast	4	0	4/9	0/5
Rainy	3	2	3/9	2/5
Total	9	5	100%	100%

Temperature

	Yes	No	P(yes)	P(no)
Hot	2	2	2/9	2/5
Mild	4	2	4/9	2/5
Cool	3	1	3/9	1/5
Total	9	5	100%	100%

Prior

Play		P(Yes)/P(No)
Yes	9	9/14
No	5	5/14
Total	14	100%

Humidity

	Yes	No	P(yes)	P(no)
High	3	4	3/9	4/5
Normal	6	1	6/9	1/5
Total	9	5	100%	100%

Wind

	Yes	No	P(yes)	P(no)
Weak	6	2	6/9	2/5
Strong	3	3	3/9	3/5
Total	9	5	100%	100%

x = (Sunny, Hot, Normal, Weak)

$$P(yes)P(sunny|yes)P(cool|yes)P(high|yes)P(strong|yes) = 0.0069$$

$$P(no)P(sunny|no)P(cool|no)P(high|no)P(strong|no) = 0.0191$$

Outlook

	Yes	No	P(yes)	P(no)
Sunny	2	3	2/9	3/5
Overcast	4	0	4/9	0/5
Rainy	3	2	3/9	2/5
Total	9	5	100%	100%

Temperature

	Yes	No	P(yes)	P(no)
Hot	2	2	2/9	2/5
Mild	4	2	4/9	2/5
Cool	3	1	3/9	1/5
Total	9	5	100%	100%

Prior

Play		P(Yes)/P(No)
Yes	9	9/14
No	5	5/14
Total	14	100%

Humidity

	Yes	No	P(yes)	P(no)
High	3	4	3/9	4/5
Normal	6	1	6/9	1/5
Total	9	5	100%	100%

Wind

	Yes	No	P(yes)	P(no)
Weak	6	2	6/9	2/5
Strong	3	3	3/9	3/5
Total	9	5	100%	100%

x = (Overcast, Hot, Normal, Weak)

贝叶斯估计

考虑：用极大似然估计可能会出现所要估计的概率值为0的情况，这时会影响到后验概率的计算结果，使分类产生偏差.

解决这一问题的方法是采用拉普拉斯平滑。

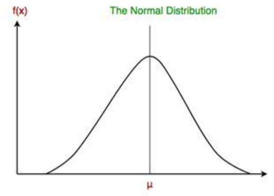
条件概率的贝叶斯估计：

$$P_\lambda(X^{(j)} = a_{jl}|Y = c_k) = \frac{\sum_{i=1}^{N} I(x_i^{(j)} = a_{jl}, y_i = c_k) + \lambda}{\sum_{i=1}^{N} I(y_i = c_k) + S_j\lambda}$$

先验概率的贝叶斯估计：

$$P_\lambda(Y = c_k) = \frac{\sum_{i=1}^{N} I(y_i = c_k) + \lambda}{N + K\lambda}$$

连续特征



$$P(x_i \mid y) = \frac{1}{\sqrt{2\pi\sigma_y^2}} \exp\left(-\frac{(x_i - \mu_y)^2}{2\sigma_y^2}\right)$$

文本分类

- 垃圾邮件分类
- 新闻报道分类
- 情感极性分类

我们如何使用朴素贝叶斯模型进行文本分类？

例子

以下是一部电影的影评以及每个影评的极性(积极/消极)。

TEXT	REVIEWS
"I liked the movie"	positive
"It's a good movie. Nice story"	positive
"Nice songs. But sadly boring ending. "	negative
"Hero's acting is bad but heroine looks good. Overall nice movie"	positive
"Sad, boring movie"	negative

如果需要区分Overall nice movie这句评论的极性，那么需要计算：

$P(positive|overall\ liked\ the\ movie)$ ---这句评论打积极标签的可能性

$P(negative|overall\ liked\ the\ movie)$ ---这句评论打消极标签的可能性

去除停顿词和词干提取

但是在进行计算之前，首先需要去除停顿词并进行词干提取。

- 去除停顿词：去除携带信息量极为有限的停顿词，例如either, else, ever等等，这些停顿词对于分类基本没有帮助。
- 词干提取：对词语去除词缀，从而得到词干的过程。

经过以上两步之后，得到：

TEXT	REVIEWS
"ilikedthemovi"	positive
"itsagoodmovienicestori"	positive
"nicesongsbutsadlyboringend"	negative
"herosactingisbadbutheroinelooksgoodoverallnicemovi"	positive
"sadboringmovi"	negative

特征工程

- 从数据中找出特征，使机器学习算法能够正常运行。

在影评的例子里面有影评的文本，所以需要把这段文本转换成能够参与计算的数字。而在本例当中，可以将每一个文本视为一组单词，因而特征可以是对每一个单词的计数，那么：

$$P(positive|overall\ liked\ the\ movie) = \frac{P(overall\ liked\ the\ movie|positive) * P(positive)}{P(overall\ liked\ the\ movie)}$$

$$P(negative|overall\ liked\ the\ movie) = \frac{P(overall\ liked\ the\ movie|negative) * P(negative)}{P(overall\ liked\ the\ movie)}$$

而对于分类器而言，需要找出积极与消极哪个标签的概率更大，所以可以去掉相同的除数，即比较两者的分子。

这样存在一个问题:

- "overall liked the movie"并没有在我们的训练集里面出现，所以概率为0，因而无法进行比较。

“Naive”假设

- 假设文本中的每一个单词都独立于其他单词存在，每一个单词都与其他不同单词无关。

根据这个假设，得：

$$P(overall\ liked\ the\ movie) = P(overall) * P(liked) * P(the) * P(movie)$$

根据贝叶斯定理：

$$P(overall\ liked\ the\ movie|positive) = P(overall|positive) * P(liked|positive) * P(the|positive) * P(movie|positive)$$

- 这样这些单词就在训练数据中出现了很多次了，从而可以进行计算了。

计算

- 首先先计算每个标签得先验概率，对于训练数据中给定得文本句子：

$$P(positive) = \frac{3}{5} \qquad P(nagetive) = \frac{2}{5}$$

- 然后计算 $P(*|positive)$ 的概率，在标签为积极的本文当中，共有17个单词，所以：

$$P(overall|positive) = \frac{1}{17} \qquad P(liked|positive) = \frac{1}{17}$$

$$P(the|positive) = \frac{2}{17} \qquad P(movie|positive) = \frac{3}{17}$$

- 如果出现概率为0的情况，可以使用拉普拉斯平滑，在计数的时候给每一个都加上1,这样就不会出现概率为0的情况。
- 同时为了平衡，可以将所有可能出现的单词总数加到分母当中,这样除数就不会大于1了，在本例当中，所以可能出现单词的总数为21。

计算

- 应用平滑得：

Word	P(WORD | POSITIVE)	P(WORD | NEGATIVE)
overall	$\frac{1+1}{17+21}$	$\frac{0+1}{7+21}$
liked	$\frac{1+1}{17+21}$	$\frac{0+1}{7+21}$
the	$\frac{2+1}{17+21}$	$\frac{0+1}{7+21}$
movie	$\frac{3+1}{17+21}$	$\frac{1+1}{7+21}$

- 最后将概率相乘,看看积极与消极哪个概率更大一些：

$$P(overall|positive)*P(liked|positive)*P(the|positive)*P(movie|positive)*P(postive) = 1.38 * 10^{-5} = 0.0000138$$

$$P(overall|negative)*P(liked|negative)*P(the|negative)*P(movie|negative)*P(negative) = 0.13 * 10^{-5} = 0.0000013$$

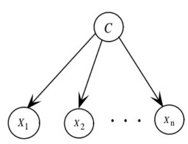
- 所以分类器给“overall liked the movie”积极的标签。

20NG

Given 1000 training documents from each group
Learn to classify new documents according to which newsgroup it came from

comp.graphics	misc.forsale
comp.os.ms-windows.misc	rec.autos
comp.sys.ibm.pc.hardware	rec.motorcycles
comp.sys.mac.hardware	rec.sport.baseball
comp.windows.x	rec.sport.hockey
alt.atheism	sci.space
soc.religion.christian	sci.crypt
talk.religion.misc	sci.electronics
talk.politics.mideast	sci.med
talk.politics.misc	
talk.politics.guns	

朴素贝叶斯网络分类器



$$C^* = \arg\max P(c|x_1, x_2, ..., x_n) = \arg\max \prod_{i=1}^{n} P(x_i|c)P(c)$$

- Q&A?