



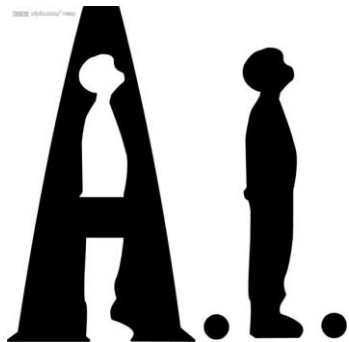
机器学习



讲师：曾江峰



E-mail: jfzeng@ccnu.edu.cn



大数据，成就未来

朴素贝叶斯



outline



1.1 引言

1.2 Naïve Bayes

1.3 案例分析

1.4 Naïve Bayes优缺点



图书馆的小姐姐 ——数学系小明的暗恋



小本本上的数据

天气	温度	是否周末	小姐姐来了么？
下雨	舒适	是	来啦
天晴	舒适	否	没来
天晴	舒适	是	没来
下雨	温度高	是	来啦
下雨	温度高	否	没来
天晴	舒适	是	来啦
...

小明今天周二（非周末），
起床发现，天气晴朗，温度
挺高（35℃），小姐姐今天
来不来图书馆？



贝叶斯定理

$$P(B|A) = \frac{P(A|B)P(B)}{P(A)}$$

$$P(\text{不去图书馆} | \text{非周末, 晴天, 温度高}) = \frac{P(\text{非周末, 晴天, 温度高} | \text{不去图书馆}) \times P(\text{不去图书馆})}{P(\text{非周末, 晴天, 温度高})}$$

VS

$$P(\text{去图书馆} | \text{非周末, 晴天, 温度高}) = \frac{P(\text{非周末, 晴天, 温度高} | \text{去图书馆}) \times P(\text{去图书馆})}{P(\text{非周末, 晴天, 温度高})}$$

outline



1.1 引言

1.2 Naïve Bayes

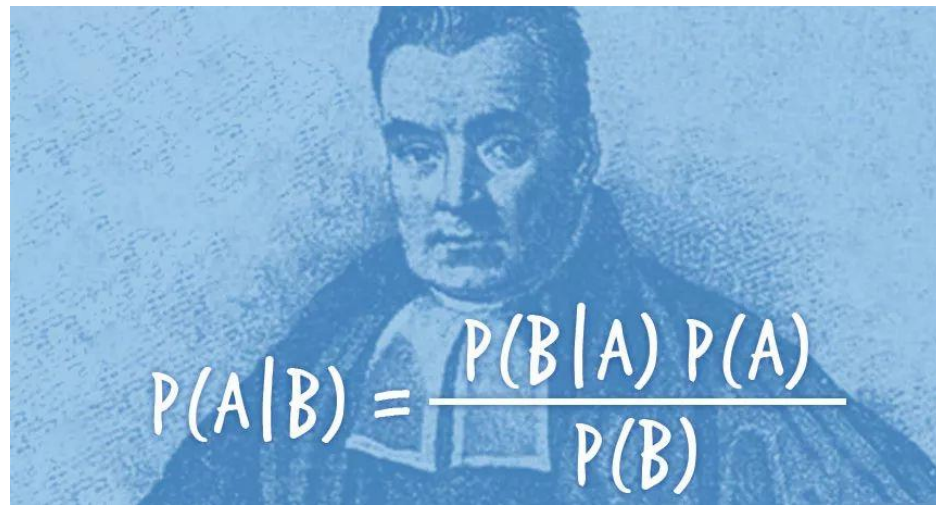
1.3 案例分析

1.4 Naïve Bayes优缺点



朴素贝叶斯

- 朴素贝叶斯是基于**贝叶斯定理**和**特征条件独立性假设**的分类算法，是统计学中的一种简单多分类算法。
- 现实生活中朴素贝叶斯算法应用广泛，如文本分类，垃圾邮件分类，信用评估，钓鱼网站检测等等。


$$P(A|B) = \frac{P(B|A) P(A)}{P(B)}$$

$$P(AB|C) = P(A|C) * P(B|C)$$

贝叶斯定理

$$P(\text{类别} | \text{特征}) = \frac{P(\text{特征} | \text{类别}) P(\text{类别})}{P(\text{特征})}$$

$$D = \{ (x_1, y_1), (x_2, y_2), \dots, (x_n, y_n) \}$$

$$x_i = (x_{i1}, x_{i2}, \dots, x_{id})$$

$$\mathbf{c} \in (\mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_m)$$

The diagram shows the formula for Bayes' Theorem with labels and arrows pointing to each term:

$$P(\mathbf{c} | \mathbf{x}) = \frac{P(\mathbf{x} | \mathbf{c}) P(\mathbf{c})}{P(\mathbf{x})} = \frac{P(\mathbf{x} | \mathbf{c}) * P(\mathbf{c})}{\sum_{o=1}^m P(\mathbf{x} | \mathbf{c}_o) * P(\mathbf{c}_o)}$$

Labels and arrows:

- Likelihood** points to $P(\mathbf{x} | \mathbf{c})$
- Class Prior Probability** points to $P(\mathbf{c})$
- Posterior Probability** points to $P(\mathbf{c} | \mathbf{x})$
- Predictor Prior Probability** points to $P(\mathbf{x})$
- 全概率** (Total Probability) points to the denominator $\sum_{o=1}^m P(\mathbf{x} | \mathbf{c}_o) * P(\mathbf{c}_o)$

如此，贝叶斯公式的分子和分母的形式一致，计算思路一致。

先验概率

- 先验概率是指根据以往经验和分析得到的概率，先验，即先于验证，就是当前事件还没发生时做出的决断。
- 事件发生前的预判概率，可以是基于历史事件的统计，可以由背景常识得出。一般都是单独事件的概率。
- 令 D_c 表示训练集 D 中第 c 类样本组成的集合，若有充足的**独立同分布样本**，则可容易地估计出类先验概率

$$P(c) = \frac{|D_c|}{|D|}$$

后验概率

- 后验概率是指事情已经发生，要求这件事情发生的原因是由某个因素引起的可能性的大小。
- 也称作事件发生后的反向条件概率；或者说是基于先验概率求得的反向条件概率。

Naïve Bayes 数学推导

$$D = \{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$$

$$x_i = (x_{i1}, x_{i2}, \dots, x_{id})$$

$$\mathbf{c} \in (\mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_m)$$

- Step 1: 计算类先验概率

$$P(Y = \mathbf{c}_k) = \frac{1}{n} \sum_{i=1}^n I(y_i = \mathbf{c}_k)$$

- Step 2: 计算类条件概率

$$P(X = x_i | Y = \mathbf{c}_k) = P(x_{i1}, x_{i2}, \dots, x_{id} | \mathbf{c}_k)$$

根据特征条件独立性假设, 可知

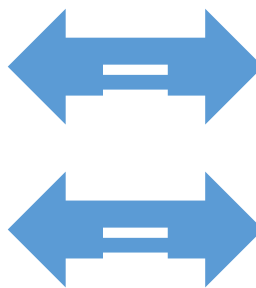
$$= \prod_{j=1}^d P(x_{ij} | \mathbf{c}_k)$$

Naïve Bayes 数学推导

- 给定样本 x_i 时，计算其后验概率，公式如下：

$$P(Y = c_k | X = x_i) = \frac{\prod_{j=1}^d P(x_{ij} | c_k) * P(c_k)}{\sum_{o=1}^m \prod_{j=1}^d P(x_{ij} | c_o) * P(c_o)}$$

- 目标：最大化这个后验概率即可

$$\hat{y} = \underset{c_k}{\operatorname{argmax}} \frac{\prod_{j=1}^d P(x_{ij} | c_k) * P(c_k)}{\sum_{o=1}^m \prod_{j=1}^d P(x_{ij} | c_o) * P(c_o)}$$

$$\hat{y} = \underset{c_k}{\operatorname{argmax}} \prod_{j=1}^d P(x_{ij} | c_k) * P(c_k)$$
$$\hat{y} = \underset{c_k}{\operatorname{argmax}} \left(\log P(c_k) + \sum_{j=1}^d \log P(x_{ij} | c_k) \right)$$

Example: 小明今天去不去图书馆?

- 小明首先对小本本上的数据进行统计分析，结果如下：

小姐姐60天里来了43天，还有17天没来。

小姐姐来的43天中，有28天天晴，有15天下雨。

小姐姐没来的17天中，有12天下雨，有5天天晴。

小姐姐来的43天中，有32天是工作日，有11天是双休日。

小姐姐没来的17天中，有12天是工作日，有5天是双休日。

小姐姐来的43天中，有25天温度高，有18天温度舒适。

小姐姐没来的17天中，有9天温度高，有8天温度舒适。

Example: 小明今天去不去图书馆?

$$P(\text{非周末} \mid \text{不去图书馆}) = 12/17$$

$$P(\text{晴天} \mid \text{不去图书馆}) = 5/17$$

$$P(\text{温度高} \mid \text{不去图书馆}) = 9/17$$

$$P(\text{非周末} \mid \text{不去图书馆}) * P(\text{晴天} \mid \text{不去图书馆}) * P(\text{温度高} \mid \text{不去图书馆}) \\ = 12/17 * 5/17 * 9/17 = \mathbf{0.1099}$$

$$P(\text{非周末} \mid \text{去图书馆}) = 32/43$$

$$P(\text{晴天} \mid \text{去图书馆}) = 28/43$$

$$P(\text{温度高} \mid \text{去图书馆}) = 25/43$$

$$P(\text{非周末} \mid \text{去图书馆}) * P(\text{晴天} \mid \text{去图书馆}) * P(\text{温度高} \mid \text{去图书馆}) \\ = 32/43 * 28/43 * 25/43 = \mathbf{0.2817}$$

$$P(\text{非周末} \mid \text{不去图书馆}) * P(\text{晴天} \mid \text{不去图书馆}) * P(\text{温度高} \mid \text{不去图书馆}) * P(\text{不去图书馆}) \\ = 0.1099 * P(\text{不去图书馆}) = 0.1099 * 17/60 = \mathbf{0.031}$$

$$P(\text{非周末} \mid \text{去图书馆}) * P(\text{晴天} \mid \text{去图书馆}) * P(\text{温度高} \mid \text{去图书馆}) * P(\text{去图书馆}) \\ = 0.2817 * P(\text{去图书馆}) = 0.2817 * 43/60 = \mathbf{0.202} > 0.031$$

结论

今天周二（非周末），天气晴朗，温度挺高的（35°C），小明掐指一算，图书馆走起！

:: Naïve Bayes

Example : Can we play tennis today ?

Lets say we have a table that decided if we should play tennis under certain circumstances. These could be the **outlook** of the weather; the **temperature**; the **humidity** and the strength of the **wind**:

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

there are 5 cases of not being able to play a game, and 9 cases of being able to play a game.



If we were given a new instance :

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

We want to know if we can play a game or not ?

:: Naïve Bayes

Example : Can we play tennis today ?

Here we have 4 attributes. What we need to do is to create “look-up tables” for each of these attributes, and write in the probability that a game of tennis will be played based on this attribute.



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

Outlook	Play = Yes	Play = No	Total
Sunny	2/9	3/5	5/14
Overcast	4/9	0/5	4/14
Rain	3/9	2/5	5/14

Temperature	Play = Yes	Play = No	Total
Hot	2/9	2/5	4/14
Mild	4/9	2/5	6/14
Cool	3/9	1/5	4/14

Humidity	Play = Yes	Play = No	Total
High	3/9	4/5	7/14
Normal	6/9	1/5	7/14

Wind	Play = Yes	Play = No	Total
Strong	3/9	3/5	6/14
Weak	6/9	2/5	8/14

If we were given a new instance :

$X = (\text{Outlook} = \text{Sunny}, \text{Temperature} = \text{Cool}, \text{Humidity} = \text{High}, \text{Wind} = \text{Strong})$, can we play the game ?

Firstly we look at the probability that we can play the game

- $P(\text{Outlook}=\text{Sunny} \mid \text{Play}=\text{Yes}) = 2/9$
- $P(\text{Temperature}=\text{Cool} \mid \text{Play}=\text{Yes}) = 3/9$
- $P(\text{Humidity}=\text{High} \mid \text{Play}=\text{Yes}) = 3/9$
- $P(\text{Wind}=\text{Strong} \mid \text{Play}=\text{Yes}) = 3/9$
- $P(\text{Play}=\text{Yes}) = 9/14$

Next we consider the fact that we cannot play a game:

- $P(\text{Outlook}=\text{Sunny} \mid \text{Play}=\text{No}) = 3/5$
- $P(\text{Temperature}=\text{Cool} \mid \text{Play}=\text{No}) = 1/5$
- $P(\text{Humidity}=\text{High} \mid \text{Play}=\text{No}) = 4/5$
- $P(\text{Wind}=\text{Strong} \mid \text{Play}=\text{No}) = 3/5$
- $P(\text{Play}=\text{No}) = 5/14$

:: Naïve Bayes

Example : Can we play tennis today ?

If $X = (\text{Outlook} = \text{Sunny}, \text{Temperature} = \text{Cool}, \text{Humidity} = \text{High}, \text{Wind} = \text{Strong})$, then

$$\begin{aligned} \bullet \quad P(\text{Play}=\text{Yes} \mid X) &= P(\text{Play}=\text{Yes} \mid \text{Outlook} = \text{Sunny}, \text{Temperature} = \text{Cool}, \text{Humidity} = \text{High}, \text{Wind} = \text{Strong}) \\ &= \frac{P(\text{Outlook} = \text{Sunny}, \text{Temperature} = \text{Cool}, \text{Humidity} = \text{High}, \text{Wind} = \text{Strong} \mid \text{Play}=\text{Yes}) * P(\text{Play}=\text{Yes})}{P(\text{Outlook} = \text{Sunny}, \text{Temperature} = \text{Cool}, \text{Humidity} = \text{High}, \text{Wind} = \text{Strong})} \\ &= \frac{P(\text{Outlook} = \text{Sunny} \mid \text{Play}=\text{Yes}) * P(\text{Temperature} = \text{Cool} \mid \text{Play}=\text{Yes}) * P(\text{Humidity} = \text{High} \mid \text{Play}=\text{Yes}) * P(\text{Wind} = \text{Strong} \mid \text{Play}=\text{Yes}) * P(\text{Play}=\text{Yes})}{P(\text{Outlook}=\text{Sunny}) * P(\text{Temperature}=\text{Cool}) * P(\text{Humidity}=\text{High}) * P(\text{Wind}=\text{Strong})} \\ &= \frac{(2/9) * (3/9) * (3/9) * (3/9) * (9/14)}{(5/14) * (4/14) * (7/14) * (6/14)} \\ &= \frac{0.0053}{0.02186} = \mathbf{0.2424} \end{aligned}$$

$$\begin{aligned} \bullet \quad P(\text{Play}=\text{No} \mid X) &= P(\text{Play}=\text{NO} \mid \text{Outlook} = \text{Sunny}, \text{Temperature} = \text{Cool}, \text{Humidity} = \text{High}, \text{Wind} = \text{Strong}) \\ &= \frac{(3/5) * (1/5) * (4/5) * (3/5) * (5/14)}{(5/14) * (4/14) * (7/14) * (6/14)} = \frac{0.0206}{0.02186} = \mathbf{0.9421} \end{aligned}$$



- $P(\text{Play}=\text{Yes} \mid X) = 0.2424$
- $P(\text{Play}=\text{No} \mid X) = 0.9421$

Since 0.9421 is greater than 0.2424 then the answer is 'no', we cannot play a game of tennis today.

拉普拉斯修正

$$\hat{y} = \underset{c_k}{arg\max} \prod_{j=1}^d P(x_{ij}|c_k) * P(c_k)$$

- 防止连乘过程中出现0的乘数

$$P(c_k) = \frac{|D_c|}{|D|}$$



$$\hat{P}(c_k) = \frac{|D_{c_k}| + 1}{|D| + N_c}$$

N_c 表示类别的个数

$$P(x_{ij}|c_k) = \frac{|D_{c_k, x_{ij}}|}{|D_c|}$$



$$\hat{P}(x_{ij}|c_k) = \frac{|D_{c_k, x_{ij}}| + 1}{|D_c| + N_j}$$

N_j 表示第j个特征的取值个数

三种常用的朴素贝叶斯实现算法

- Gaussian Naive Bayes: 假设类条件概率分布为高斯分布，实现的朴素贝叶斯算法称为高斯朴素贝叶斯算法。
- Bernoulli Naive Bayes: 假设类条件概率分布为伯努利分布，实现的朴素贝叶斯算法称为伯努利朴素贝叶斯算法。
- Multinomial Naive Bayes: 假设类条件概率分布为多项式分布，实现的朴素贝叶斯算法称为多项式朴素贝叶斯算法。

$$\begin{aligned} P(X = \mathbf{x}_i | Y = c_k) &= P(x_{i1}, x_{i2}, \dots, x_{id} | c_k) \\ &= \prod_{j=1}^d P(x_{ij} | c_k) \end{aligned}$$

朴素的含义

- 朴素贝叶斯中的“朴素”的含义，即各个特征之间条件独立性假设。
- 特征条件独立性假设是说用于分类的特征在类确定的条件下都是独立的，该假设使得朴素贝叶斯的学习成为可能。
- 这个强假设使得对问题的求解变得更加简单，因此朴素贝叶斯算法逻辑简单，易于计算实现！
- 然而，这个假设往往在实际应用中是不成立的！

outline



1.1 引言

1.2 Naïve Bayes

1.3 案例分析

1.4 Naïve Bayes优缺点



案例分析——新闻标题分类

训练数据	类别
那部让人感动的电影名作重映	电影
华丽的动作电影首映	电影
复映的名作感动了世界	电影
沙尘暴笼罩着火星	宇宙
火星探测终于重新开始	宇宙
VR中看到的火星沙尘暴让人感动	宇宙

验证数据	类别
复映的动作电影名作让人感动	? ?

特征设计

训练数据	名作	电影	华丽	动作	世界	感动	沙尘暴	火星	探测	重新开始	VR	类别
那部让人感动的电影名作重映	1	1	0	0	0	1	0	0	0	0	0	1
华丽的动作电影首映	0	1	1	1	0	0	0	0	0	0	0	1
复映的名作感动了世界	1	0	0	0	1	1	0	0	0	0	0	1
沙尘暴笼罩着火星	0	0	0	0	0	0	1	1	0	0	0	0
火星探测终于重新开始	0	0	0	0	0	0	0	1	1	1	0	0
VR中看到的火星沙尘暴让人感动	0	0	0	0	0	1	1	1	0	0	1	0

```
from sklearn.naive_bayes import MultinomialNB

# 数据生成
X_train = [[1, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0],
            [0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0],
            [1, 0, 0, 0, 1, 1, 0, 0, 0, 0, 0],
            [0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0],
            [0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0],
            [0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 1]]
y_train = [1, 1, 1, 0, 0, 0]
model = MultinomialNB()
model.fit(X_train, y_train) # 训练
model.predict([[1, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0]]) # 评估
```


案例分析——新闻文本分类

- 下载新闻数据集20newsgroups
- 将文本转换为特征向量（CountVectorizer）
- 训练模型
- 验证模型
- 打印分类准确率和详细分类性能报告

```
# 从sklearn.datasets里导入新闻数据抓取器fetch_20newsgroups。
from sklearn.datasets import fetch_20newsgroups
# 从sklearn.model_selection 导入 train_test_split。
from sklearn.model_selection import train_test_split
# 从sklearn.feature_extraction.text里导入用于文本特征向量转化模块将文本转化为特征向量。属于特征抽取
from sklearn.feature_extraction.text import CountVectorizer
# 从sklearn.naive_bayes里导入朴素贝叶斯模型。
from sklearn.naive_bayes import MultinomialNB
# 从sklearn.metrics里导入classification_report用于详细的分类性能报告。
from sklearn.metrics import classification_report

# 与之前预存的数据不同，fetch_20newsgroups需要即时从互联网下载数据。
news = fetch_20newsgroups(subset='all')
# 随机采样25%的数据样本作为测试集。
X_train, X_test, y_train, y_test = train_test_split(news.data, news.target, test_size=0.25, random_state=33)
vec = CountVectorizer()
X_train = vec.fit_transform(X_train)
X_test = vec.transform(X_test)


# 从使用默认配置初始化朴素贝叶斯模型。
mnb = MultinomialNB()
# 利用训练数据对模型参数进行估计。
mnb.fit(X_train, y_train)
# 对测试样本进行类别预测，结果存储在变量y_predict中。
y_predict = mnb.predict(X_test)

print('The accuracy of Naive Bayes Classifier is', mnb.score(X_test, y_test))
print(classification_report(y_test, y_predict, target_names = news.target_names))
```

基于**scikit-learn**的模型实现

- 课后作业：基于sklearn的鸢尾花分类
- 要求：
 - 按照4/1分训练和测试；
 - 调用GaussianNB模型；
 - 打印出准确率；
 - 从sklearn.metrics里导入classification_report用于详细的分类性能报告。

outline



1.1 引言

1.2 Naïve Bayes

1.3 案例分析

1.4 Naïve Bayes优缺点

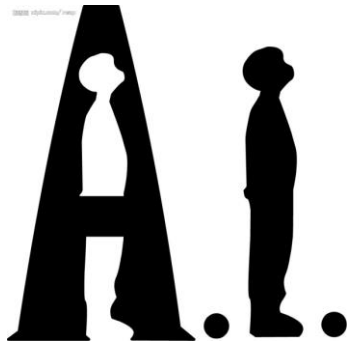


优点

- 1、朴素贝叶斯模型发源于古典数学理论，有稳定的分类效率。
- 2、对大数量训练和查询时具有较高的速度。即使使用超大规模的训练集，针对每个项目通常也只会相对较少的特征数，并且对项目的训练和分类也仅仅是特征概率的数学运算而已。
- 3、对小规模的数据表现很好，能处理多分类任务，适合增量式训练（即可以实时地对新增的样本进行训练）。
- 4、对缺失数据不太敏感，算法也比较简单，常用于文本分类。
- 5、朴素贝叶斯对结果解释容易理解。

缺点

- 1、由于使用了样本属性独立性的假设，所以在属性个数比较多或者属性之间相关性较大时，分类效果不好。
- 2、需要计算先验概率，且先验概率很多时候取决于假设，假设的模型可以有很多种，因此在某些时候会由于假设的先验模型的原因导致预测效果不佳。
- 3、由于我们是通过先验和数据来决定后验的概率从而决定分类，所以分类决策存在一定的错误率。
- 4、对输入数据的表达形式很敏感。



大数据，成就未来



Thank you!