朴素贝叶斯模型

1. 朴素贝叶斯模型简单代码演示

In [1]:

```
from sklearn.naive_bayes import GaussianNB #使用高斯贝叶斯分类器，假设概率满足高斯分布
X = [[1, 2], [3, 4], [5, 6], [7, 8], [9, 10]] #X是特征变量，共有2个特征
y = [0, 0, 0, 1, 1] #y是目标变量，共有2个类别——0和1

model = GaussianNB()
model.fit(X, y)

print(model.predict([[5, 5]])) #用predict()函数进行预测
```

```
[0]
```

2.案例实战 - 肿瘤预测模型

肿瘤性质的判断影响着患者的治疗方式和痊愈速度。传统的做法是医生根据数十个指标来判断肿瘤的性质，预测效果依赖于医生的个人经验而且效率较低，而通过机器学习，我们有望能快速预测肿瘤的性质。

2.1 读取数据

首先通过如下代码导入某医院乳腺肿瘤患者的6个特征维度及肿瘤性质的数据。共569个患者，其中良性肿瘤358例、恶性肿瘤211例。

In [2]:

```
import pandas as pd
df = pd.read_excel('肿瘤数据.xlsx')
df.head() #通过打印df.head()来查看前5行数据
```

Out[2]:

	最大周长	最大凹陷度	平均凹陷度	最大面积	最大半径	平均灰度值	肿瘤性质
0	184.60	0.2654	0.14710	2019.0	25.38	17.33	0
1	158.80	0.1860	0.07017	1956.0	24.99	23.41	0
2	152.50	0.2430	0.12790	1709.0	23.57	25.53	1
3	98.87	0.2575	0.10520	567.7	14.91	26.50	0
4	152.20	0.1625	0.10430	1575.0	22.54	16.67	0

2.2 划分特征变量和目标变量

In [3]:

```
X = df.drop(columns='肿瘤性质') #用drop()函数删除“肿瘤性质”列，将剩下的数据作为特征变量赋给变
y = df['肿瘤性质']   #通过DataFrame提取列的方式提取“肿瘤性质”列的数据作为目标变量，并赋给变量
```

2.3 模型搭建

2.3.1 划分训练集和测试集

In [4]:

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=1) #设定t
```

2.3.2 朴素贝叶斯模型

In [5]:

```
from sklearn.naive_bayes import GaussianNB
nb_clf = GaussianNB()  # 高斯朴素贝叶斯模型
nb_clf.fit(X_train,y_train)
```

Out[5]:

```
GaussianNB()
```

In [6]:

```
y_pred = nb_clf.predict(X_test)
print(y_pred[:100])
```

```
[1 0 1 0 1 0 0 0 1 1 1 0 0 1 1 1 1 1 1 0 1 1 0 1 0 1 1 0 0 0 0 1 0 0 1 1 0
 1 1 1 1 1 1 1 1 0 1 1 1 0 0 0 1 1 1 1 1 1 0 1 1 1 0 1 1 1 1 1 0 1 1 1 1 1 0
 1 0 1 1 1 0 1 0 1 0 1 1 0 1 0 1 1 0 1 1 0 1 1 1 1 1]
```

In [7]:

```
a = pd.DataFrame()  # 创建一个空DataFrame
a['预测值'] = list(y_pred)
a['实际值'] = list(y_test)
a.head()
```

Out[7]:

	预测值	实际值
0	1	1
1	0	0
2	1	1
3	0	0
4	1	0

In [8]:

```
from sklearn.metrics import accuracy_score
score = accuracy_score(y_pred, y_test)
print(score)
```

```
0.9473684210526315
```

In [ ]:

```

```