高斯朴素贝叶斯 (Gaussian Naive Bayes) 详细介绍

高斯朴素贝叶斯 (Gaussian Naive Bayes)

高斯朴素贝叶斯是朴素贝叶斯分类器的一种变体，它特别适用于**连续特征**的分类问题。与多项式朴素贝叶斯处理计数型数据不同，高斯朴素贝叶斯假设每个特征的概率分布服从**高斯分布（即正态分布）**。

核心思想

如同所有朴素贝叶斯分类器一样，高斯朴素贝叶斯也基于**朴素贝叶斯假设**，即给定类别的情况下，特征之间是条件独立的。这个“朴素”的假设极大地简化了模型的计算，使其在计算效率方面具有优势。

当特征是连续值时，我们不能像多项式朴素贝叶斯那样简单地统计频率。高斯朴素贝叶斯模型假设这些连续特征在每个类别下都服从高斯分布。

数学原理

对于一个给定的数据点 $x = (x_1, x_2, ..., x_n)$ 和一个类别 C_k，我们想要计算 $P(C_k|x)$。根据贝叶斯定理，我们有：

$$P(C_k|x) = \frac{P(x|C_k)P(C_k)}{P(x)}$$

其中：

- $P(C_k|x)$ 是后验概率，表示给定数据点 x 的情况下，它属于类别 C_k 的概率。
- $P(x|C_k)$ 是似然，表示给定类别 C_k 的情况下，数据点 x 出现的概率。
- $P(C_k)$ 是先验概率，表示类别 C_k 本身出现的概率。
- $P(x)$ 是证据，表示数据点 x 出现的概率。

由于 $P(x)$ 对于所有类别都是常数，我们只需要最大化 $P(x|C_k)P(C_k)$。
朴素贝叶斯假设：特征之间是条件独立的。

$$P(x|C_k) = P(x_1, x_2, ..., x_n|C_k) = \prod_{i=1}^{n} P(x_i|C_k)$$

其中 x_i 是数据点 x 的第 i 个特征。
高斯分布假设：对于每个特征 x_i 在给定类别 C_k 的情况下，其概率分布服从高斯分布。

$$P(x_i|C_k) = \frac{1}{\sqrt{2\pi\sigma_{ik}^2}} \exp\left(-\frac{(x_i - \mu_{ik})^2}{2\sigma_{ik}^2}\right)$$

其中：

- μ_{ik} 是特征 x_i 在类别 C_k 中的均值。
- σ_{ik}^2 是特征 x_i 在类别 C_k 中的方差。

这些参数 (μ_{ik} 和 σ_{ik}^2) 在训练阶段从数据中估计得到。对于每个类别 C_k 和每个特征 x_i，模型会计算其均值和方差。

先验概率 $P(C_k)$ 的估计：

$$P(C_k) = \frac{\text{Number of samples in } C_k}{\text{Total number of samples}}$$

最终，模型会选择使 $P(x|C_k)P(C_k)$ 最大的类别 C_k 作为预测结果。

Scikit-learn 中的实现

在 Scikit-learn 中，高斯朴素贝叶斯通过 `sklearn.naive_bayes.GaussianNB` 类实现。

主要参数：

`GaussianNB` 几乎没有可调整的参数，因为它的参数（均值和方差）是直接从训练数据中估计出来的。

- `priors`: 数组类型，默认值为 `None`。如果指定，则为每个类别设置先验概率。如果为 `None`，则根据训练数据中每个类别的样本比例自动计算先验概率。
- `var_smoothing`: 浮点型，默认值为 `1e-9`。这是一个添加到每个特征方差中的平滑项。它的作用是防止计算过程中出现零方差的情况，因为零方差会导致似然为无穷大或零，从而使模型不稳定。这个小值有助于数值稳定性。

适用场景

高斯朴素贝叶斯最常用于：

- **连续数值特征**的分类问题。
- 当你**假设特征服从正态分布**时，或者你希望一个简单、快速的基线模型时。

优点

- **实现简单，计算高效**：训练和预测速度快，尤其是在大型数据集上。
- **对小规模数据集表现良好**：当数据量不足以训练更复杂的模型时，朴素贝叶斯仍能提供合理的性能。
- **对高维数据表现良好**：虽然 “朴素” 假设简化了问题，但它在高维特征空间中仍然能有效工作。

缺点

- **“朴素”假设**：特征条件独立性假设在现实中很少完全成立。如果特征之间存在强烈的相关性，这可能会降低模型的性能。
- **对数据分布的严格假设**：假设特征服从高斯分布，如果实际数据分布与高斯分布偏差较大，模型的性能可能会受影响。在这种情况下，可能需要对数据进行转换（如对数转换）或使用其他更灵活的朴素贝叶斯变体。
- **概率估计可能不校准**：尽管它能给出概率预测，但这些概率可能未校准（即预测概率不直接对应真实事件频率），如之前讨论的，可能需要 `CalibratedClassifierCV` 进行校准。

与其他朴素贝叶斯变体的区别

- `MultinomialNB` **(多项式朴素贝叶斯)**：适用于**计数型离散特征**，如文本分类中的词频。假设特征服从多项式分布。
- `BernoulliNB` **(伯努利朴素贝叶斯)**：适用于**二元特征**（特征值为 0 或 1），表示特征是否存在。假设特征服从伯努利分布。
- `ComplementNB` **(补充朴素贝叶斯)**：也是为文本分类设计的变体，尤其适用于**不平衡数据集**。

总之，高斯朴素贝叶斯是一个简单而有效的分类器，适用于具有连续数值特征的数据集，但需要注意其对特征独立性和高斯分布的假设。