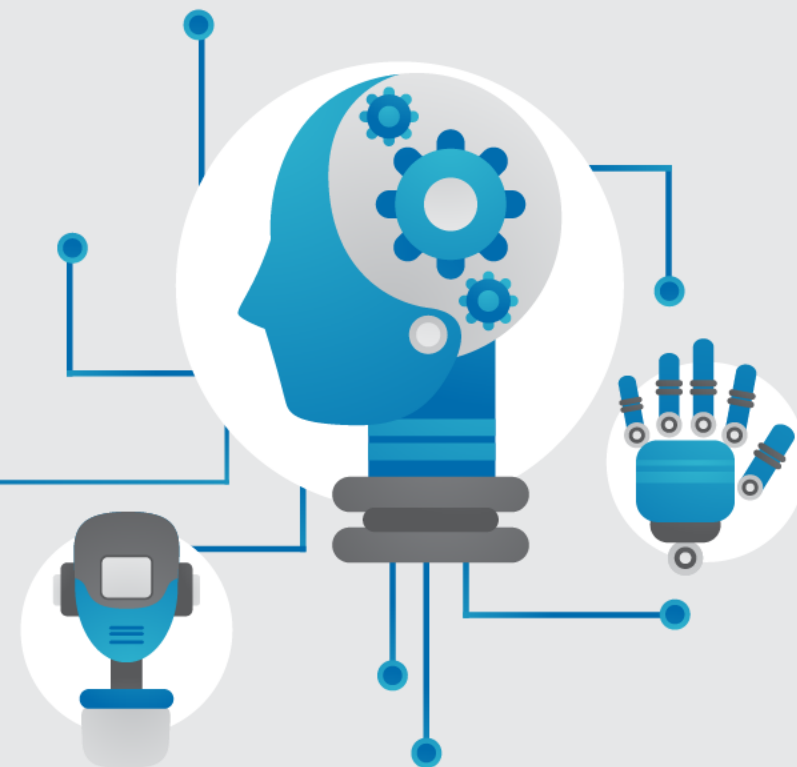


# Scikit-learn GaussianNB 和 MultinomialNB





# GaussianNB 介紹

機器學習實務



- › 貝式分類器乃是根據貝氏定理（ Bayes' theorem ）為基礎，用以判斷未知類別的資料應該最接近哪一個類別。
- › `sklearn.naive_bayes.GaussianNB` 為貝氏分類器演算法的實作之一，假設特徵的先驗機率為**常態分佈**，主要使用於**連續形資料**。

[Install](#) [User Guide](#) [API](#) [Examples](#) [More](#)  [Go](#)

[Prev](#) [Up](#) [Next](#)

scikit-learn 0.22.2  
[Other versions](#)

Please [cite us](#) if you use the software.

`sklearn.naive_bayes.GaussianNB`  
Examples using  
`sklearn.naive_bayes.GaussianNB`

## `sklearn.naive_bayes.GaussianNB`

```
class sklearn.naive_bayes.GaussianNB(priors=None, var_smoothing=1e-09)
```

[\[source\]](#)

Gaussian Naive Bayes (GaussianNB)

Can perform online updates to model parameters via [partial\\_fit](#). For details on algorithm used to update feature means and variance online, see Stanford CS tech report STAN-CS-79-773 by Chan, Golub, and LeVeque:  
<http://i.stanford.edu/pub/cstr/reports/cs/tr/79/773/CS-TR-79-773.pdf>

Read more in the [User Guide](#).



# GaussianNB 參數說明

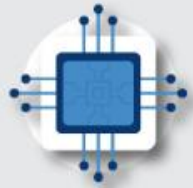


› `class sklearn.naive_bayes.GaussianNB(priors=None, var_smoothing=1e-09)`

› GaussianNB 類別常用參數

- Priors
- var\_smoothing





# GaussianNB 參數說明

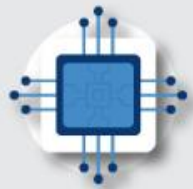


- › **priors : array-like, shape (n\_classes,)**

類別的先驗機率。如果指定，則先驗機率不會隨著資料調整。

- › **var\_smoothing : float, optional (default=1e-9)**

在估計變異數的時候，為了追求估計的穩定性，將所有特徵的變異數中最大的變異數，以某個比例添加到估計的變異數中。這個比例，由var\_smoothing參數控制。




# MultinomialNB 介紹

機器學習實務



- › `sklearn.naive_bayes.MultinomialNB` 也是貝氏分類器演算法的實作之一，假設特徵的先驗機率為**多項式分佈**，主要使用於**離散形資料**。

 [Install](#) [User Guide](#) [API](#) [Examples](#) [More](#)  [Go](#)

[Prev](#) [Up](#) [Next](#)

**scikit-learn 0.22.2**  
[Other versions](#)

Please **cite us** if you use the software.

`sklearn.naive_bayes.MultinomialNB`  
[Examples using sklearn.naive\\_bayes.MultinomialNB](#)

[Toggle Menu](#)

## `sklearn.naive_bayes.MultinomialNB`

```
class sklearn.naive_bayes.MultinomialNB(alpha=1.0, fit_prior=True, class_prior=None)
```

[\[source\]](#)

Naive Bayes classifier for multinomial models

The multinomial Naive Bayes classifier is suitable for classification with discrete features (e.g., word counts for text classification). The multinomial distribution normally requires integer feature counts. However, in practice, fractional counts such as tf-idf may also work.

Read more in the [User Guide](#).

|                    |  |
|--------------------|--|
| <b>Parameters:</b> | <b>alpha : float, optional (default=1.0)</b><br>Additive (Laplace/Lidstone) smoothing parameter (0 for no smoothing).  |
|                    | <b>fit_prior : boolean, optional (default=True)</b><br>Whether to learn class prior probabilities or not. If false, a uniform prior will be used.                                  |
|                    | <b>class_prior : array-like, size (n_classes,), optional (default=None)</b><br>Prior probabilities of the classes. If specified the priors are not adjusted according to the data. |
| <b>Attributes:</b> | <b>class_count_ : array, shape (n_classes,)</b><br>Number of samples encountered for each class during fitting. This value is weighted by the sample weight when provided.         |



# MultinomialNB 參數說明

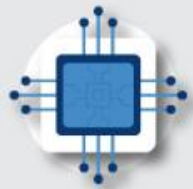


› `class sklearn.naive_bayes.MultinomialNB(alpha=1.0, fit_prior=True, class_prior=None)`

› MultinomialNB 類別常用參數

- `alpha`
- `fit_prior`
- `class_prior`





# MultinomialNB 參數說明



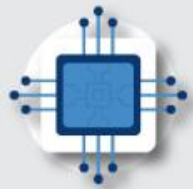
## › **alpha : float, optional (default=1.0)**

其實就是添加拉普拉斯 ( Laplas ) 平滑，即為上述公式中的 $\lambda$ ，如果這個參數設置為0，就是不添加平滑。

## › **fit\_prior : boolean, optional (default=True)**

- 布林參數，表示是否要考慮先驗機率。
- 如果是false，則所有的樣本類別輸出都有相同的類別先驗機率，否則可以自己用第三個參數class\_prior輸入先驗機率，或者不輸入第三個參數class\_prior，讓MultinomialNB自行從訓練集樣本來計算先驗機率。





# MultinomialNB 參數說明

機器學習實務



› **class\_prior** : array-like, size (n\_classes,), optional  
(default=None)

- 類別的先驗機率





# GaussianNB 函式說明

機器學習實務



## › GaussianNB 常用函式

- fit
- predict
- score



# 訓練 ( fit )



- › 指令：fit(self, X, y[, sample\_weight])
- › 參數
  - x：訓練輸入樣本
  - y：目標值（分類中的類標籤）
- › 回傳：訓練後的GaussianNB物件
- › 說明：根據訓練集（x，y）建立一個Gaussian Naive Bayes
- › 範例程式

```
from sklearn.naive_bayes import GaussianNB  
gaussianNB = GaussianNB()  
gaussianNB.fit(X_train, y_train)
```

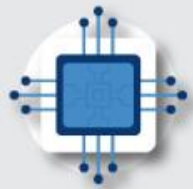


# 預測 ( predict )



- › 指令 : `predict(self, X)`
- › 參數
  - `x` : 輸入樣本
- › 回傳 : 預測的類別
- › 範例程式

```
from sklearn.naive_bayes import GaussianNB  
gaussianNB = GaussianNB()  
gaussianNB.fit(X_train, y_train)  
predictions = gaussianNB.predict(x_test)
```



# 評分 ( score )



› 指令：score(self, X, y[, sample\_weight])

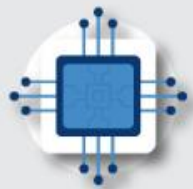
› 參數

- x：測試樣本
- y：測試樣本的正确答案

› 回傳：測試樣本的平均準確度

› 範例程式

```
from sklearn.naive_bayes import GaussianNB
gaussianNB = GaussianNB()
gaussianNB.fit(X_train, y_train)
accuracy = gaussianNB.score(x_test, y_test)
```



# 程式範例 ( IRIS )



## › 程式碼

```
import numpy as np
import matplotlib.pyplot as plt
from sklearn.naive_bayes import GaussianNB
from sklearn import datasets

# 載入資料
iris = datasets.load_iris()
X = iris.data[:, :2] # 只取前兩種特徵
Y = iris.target

# 建立 Gaussian Naive Bayes
gaussianNB = GaussianNB()

# 進行訓練
gaussianNB.fit(X, Y)

# 繪製座標軸
x_min, x_max = X[:, 0].min() - .5, X[:, 0].max() + .5
y_min, y_max = X[:, 1].min() - .5, X[:, 1].max() + .5
h = .02 # 單位間隔
xx, yy = np.meshgrid(np.arange(x_min, x_max, h), np.arange(y_min, y_max, h))
```



# 程式範例 ( IRIS )



## › 程式碼

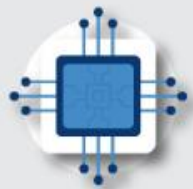
```
# 進行預測
Z = gaussianNB.predict(np.c_[xx.ravel(), yy.ravel()])

# 繪製預測結果
Z = Z.reshape(xx.shape)
plt.figure(1, figsize=(4, 3))
plt.pcolormesh(xx, yy, Z, cmap=plt.cm.Paired)

plt.scatter(X[:, 0], X[:, 1], c=Y, edgecolors='k', cmap=plt.cm.Paired)
plt.xlabel('Sepal Length')
plt.ylabel('Sepal width')

plt.xlim(xx.min(), xx.max())
plt.ylim(yy.min(), yy.max())
plt.xticks(())
plt.yticks(())

plt.show()
```



# 程式範例 ( IRIS )

機器學習實務



› 輸出結果

