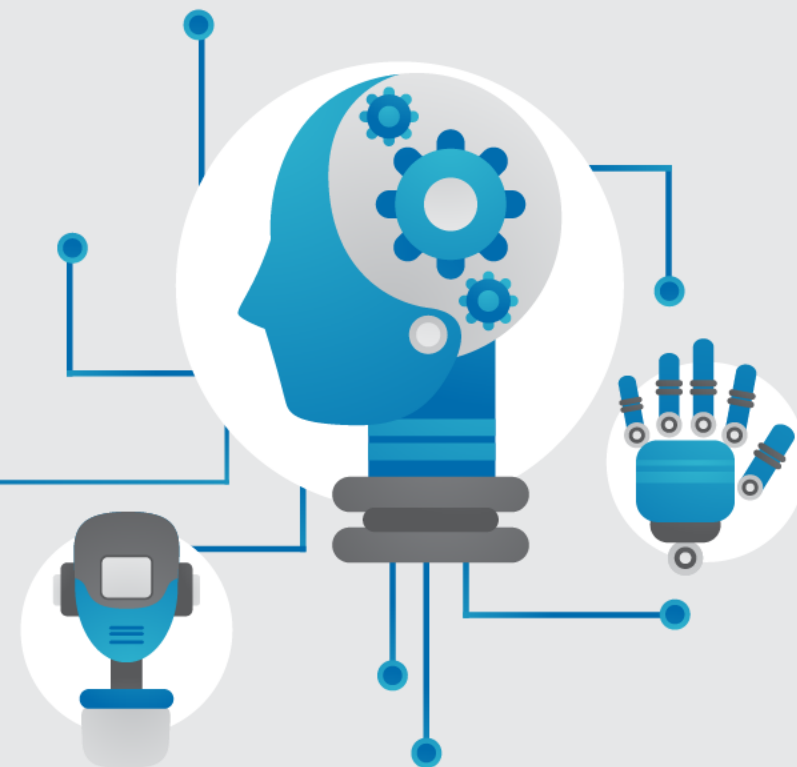


# 裝袋法 ( Bagging )

## 整體學習






# 裝袋法 ( Bagging )



- › **Bagging演算法 ( 裝袋演算法 ) = bootstrap aggregating** 是機器學習領域的一種集成學習 ( Ensemble learning ) 演算法，是由 Leo Breiman 於1994年提出。
- › Bagging演算法可與其他分類、回歸演算法結合，提高準確率、穩定性的同時，通過降低結果的方差，避免過擬合的發生。
- › `sklearn.ensemble.BaggingClassifier` 是裝袋法的實作

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

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

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

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

`sklearn.ensemble.BaggingClassifier`

## `sklearn.ensemble.BaggingClassifier`

```
class sklearn.ensemble.BaggingClassifier(base_estimator=None, n_estimators=10, max_samples=1.0, max_features=1.0,
bootstrap=True, bootstrap_features=False, oob_score=False, warm_start=False, n_jobs=None, random_state=None, verbose=0)
[source]
```

A Bagging classifier.



# 裝袋法步驟



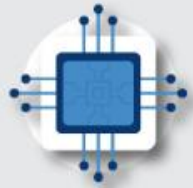
指定一個**基礎預測器 (base estimator)**，  
這個基礎預測器可以是任何**機器學習分類或  
迴歸演算法**。



對訓練數據集進行**隨機取樣**，並使用取樣後  
的數據子集，使用指定的基礎預測器訓練一個  
**預測器模型**。



預測時，每一個預測器模型都會做出預測，  
整體結果則是每個預測的**平均或投票結果**。



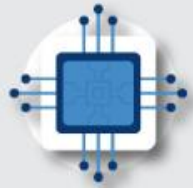
# Bagging Classifier 參數說明

機器學習實務



## › `class sklearn.ensemble.BaggingClassifier`

(base\_estimator= None, n\_estimators=10,  
max\_samples=1.0, max\_features=1.0,  
bootstrap=True, bootstrap\_features=False,  
oob\_score=False, warm\_start=False, n\_jobs=None,  
random\_state=None, verbose=0)



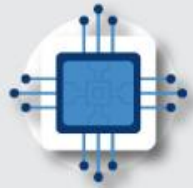
# Bagging Classifier 參數說明

機器學習實務



## › Bagging Classifier 類別常用參數

- ✓ base\_estimator
- ✓ n\_estimators
- ✓ max\_samples
- ✓ max\_features
- ✓ bootstrap
- ✓ bootstrap\_features
- ✓ oob\_score
- ✓ warm\_start



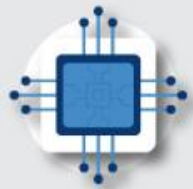
# 參數 base\_estimator



› **base\_estimator : object or None, optional (default=None)**

- **基礎預測器**，是一個 scikit-learn 的分類器或迴歸器。  
如果沒有設定，那麼預設使用決策樹（ Decision Tree ）。

```
from sklearn.ensemble import BaggingClassifier
from sklearn.svm import SVC
baggingClassifier = BaggingClassifier(base_estimator=SVC())
```



# 參數 n\_estimators



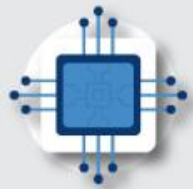
## › n\_estimators : int, optional (default=10)

- 基礎預測器的數量，預設值為10。
- 通常，預測器愈多，整體模型的變異愈小。

```
from sklearn.ensemble import BaggingClassifier
from sklearn.svm import SVC
baggingClassifier = BaggingClassifier(base_estimator=SVC(), n_estimators=10, verbose=2)
```

```
Building estimator 1 of 10 for this parallel run (total 10)...
Building estimator 2 of 10 for this parallel run (total 10)...
Building estimator 3 of 10 for this parallel run (total 10)...
Building estimator 4 of 10 for this parallel run (total 10)...
Building estimator 5 of 10 for this parallel run (total 10)...
Building estimator 6 of 10 for this parallel run (total 10)...
Building estimator 7 of 10 for this parallel run (total 10)...
Building estimator 8 of 10 for this parallel run (total 10)...
Building estimator 9 of 10 for this parallel run (total 10)...
Building estimator 10 of 10 for this parallel run (total 10)...
```





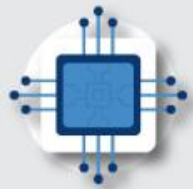
# 參數 max\_samples



› max\_samples : int or float, optional (default=1.0)

- 每個用於訓練基礎預測器的**數據子集的樣本數量**
- 可以是**浮點數**（ 0.0至1.0，表示取樣樣本佔所有樣本的比例 ），也可以是**整數**（ 表示取樣樣本的實際數量 ）。
- 如果輸入了**1**而不是**1.0**，那麼每個數據子集僅包含1個樣本，會導致嚴重失誤。





# 參數 max\_features



› max\_features : int or float, optional (default=1.0)

- 每個數據子集的特徵數量
- 可以是浮點數（0.0至1.0，表示選取的特徵數量佔所有特徵數量的比例），也可以是整數（表示選取的特徵數量的實際數量）。



# 參數 bootstrap



› **bootstrap** : boolean, optional (default=True)

- 在隨機選取樣本時是否可以重複選取（ sample with replacement ）。





# 參數 bootstrap\_features

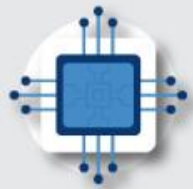
機器學習實務



› bootstrap\_features : boolean, optional (default=False)

- 在隨機選取**特徵**時是否可以重複選取



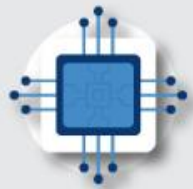


# 參數 oob\_score



› oob\_score : bool, optional (default=False)

- 是否計算 out-of-bag 分數
- 每個基礎預測器都只在子數據集上訓練。可以用剩下的樣本（ out-of-bag samples ），來估算其泛化誤差（ generalization error ）。

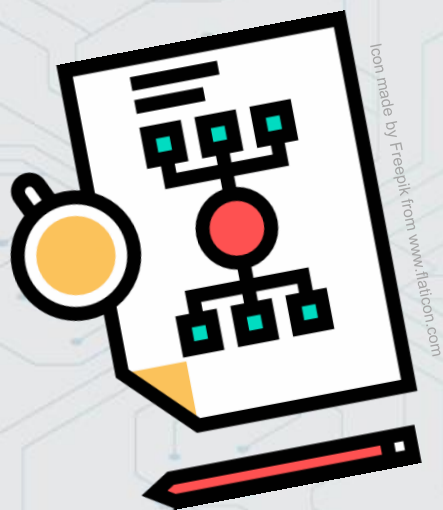


# 參數 warm\_start



› warm\_start : bool, optional (default=False)

- 如果是 True，在下一次使用 fit 方法時，除了保留原有的預測器模型，將再增加 n\_estimators 個新的基礎預測器模型





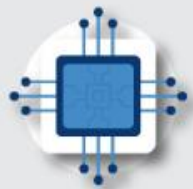
# Bagging Classifier 函式說明

機器學習實務



## › 常用函式

- fit
- predict
- score



# 訓練



- › 指令：`fit(self, X, y, sample_weight=None)`
- › 參數
  - X: 訓練向量
  - y: 相對於X的目標向量
- › 回傳：訓練後的bagging classifier 模型物件
- › 說明：從訓練中建立一個預測器的裝袋集合





# 訓練



## › 範例程式

```
from sklearn.ensemble import BaggingClassifier  
from sklearn.svm import SVC  
baggingClassifier =  
BaggingClassifier(base_estimator=SVC(),  
n_estimators=10)  
baggingClassifier.fit(X_train, y_train)
```



# 預測



- › 指令：`predict(self, X)`
- › 參數
  - `X`: 輸入樣本
- › 回傳：每個樣本的預測類別標籤
- › 說明：預測`X`中樣本的類別標籤



# 預測



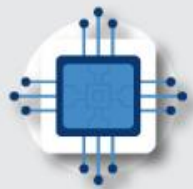
## › 範例程式

```
from sklearn.ensemble import BaggingClassifier
from sklearn.svm import SVC

baggingClassifier =
BaggingClassifier(base_estimator=SVC(),
n_estimators=10)

baggingClassifier.fit(X_train, y_train)

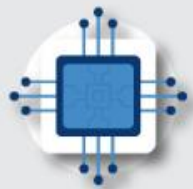
predictions = baggingClassifier.predict(X_test)
```



# 評分



- › 指令：`score(self, X, y, sample_weight=None)`
- › 參數
  - `X`：測試樣本
  - `y`：測試樣本的正確答案
- › 回傳：測試樣本的平均準確度
- › 說明：返回給定測試數據和標籤上的平均準確度



# 評分



## › 範例程式

```
from sklearn.ensemble import BaggingClassifier
from sklearn.svm import SVC
baggingClassifier =
BaggingClassifier(base_estimator=SVC(), n_estimators=10)
baggingClassifier.fit(X_train, y_train)

accuracy = baggingClassifier.score(X_test, y_test)
```