# ****、參數****

### ****max\_depth****：

決策樹**最大深度**，【int,  None】。**默認值是‘None’**。一般數據比較少或者特徵少的時候可以不用管這個值，如果模型樣本數量多，特徵也多時，推薦限制這個最大深度，具體取值取決於數據的分佈。**常用的可以取值10-100之間**，常用來**解決過擬合**。

### ****criterion****：

特徵選擇標準，【entropy, gini】。**默認gini，即CART算法。**

**找出最佳節點及分枝的指標為＂不純度＂，不純度越低，決策樹對訓練集得的擬合越好。決策樹就是追求不純度相關指標的最優化**

**樹中的每個節點都會有一個不純度，子節點的不純度定低於父節點，葉子節點的不純度是最低的。**

**"entropy": 使用信息熵**

**信息熵對不純度叫基尼係數強(對不純度懲罰最強)，容易過擬合。實際使用中兩者效果基本一樣**

**"gini"：使用基尼係數，適用於數據維度大，噪音很大的數據。**

### ****splitter****：

**特徵劃分標準**，【best, random】。best在特徵的所有劃分點中找出**最優的劃分點**，random隨機的在部分劃分點中找局部最優的劃分點。**默認的‘best’適合樣本量不大**的時候，而如果樣本數據量非常大，此時決策樹構建推薦‘random’。

### ****min\_samples\_split****：

**內部節點（即判斷條件）**再劃分所需最小樣本數，【int, float】。默認值爲2。如果是int，則取傳入值本身作爲最小樣本數；如果是float，則取ceil(min\_samples\_split\*樣本數量)作爲最小樣本數。（向上取整）

### ****min\_samples\_leaf****：

**葉子節點（即分類）**最少樣本數。如果是int，則取傳入值本身作爲最小樣本數；如果是float，則取ceil(min\_samples\_leaf\*樣本數量)的值作爲最小樣本數。這個值限制了葉子節點最少的樣本數，如果某葉子節點數目小於樣本數，則會和兄弟節點一起被剪枝。

### ****min\_weight\_fraction\_leaf****：

**葉子節點（即分類）最小的樣本權重和**，【float】。這個值限制了葉子節點所有樣本權重和的最小值，如果小於這個值，則會和兄弟節點一起被剪枝。**默認是0**，就是不考慮權重問題，所有樣本的權重相同。

一般來說如果我們**有較多樣本有缺失值**或者**分類樹樣本的分佈類別偏差很大**，就會引入樣本權重，這時就要注意此值。

### ****max\_features****：

在劃分數據集時**考慮的最多的特徵值數量**，【int值】。在每次split時最大特徵數；【float值】表示百分數，即（max\_features\*n\_features）

### ****max\_leaf\_nodes****：

**最大葉子節點數**。【int, None】，通過設置最大葉子節點數，可以**防止過擬合**。默認值None，默認情況下不設置最大葉子節點數。如果加了限制，算法會建立在最大葉子節點數內最優的決策樹。如果特徵不多，可以不考慮這個值，但是如果特徵多，可以加限制，**具體的值可以通過交叉驗證得到**。

### ****min\_impurity\_decrease****：

**節點劃分最小不純度**，【float】。默認值爲‘0’。限制決策樹的增長，節點的不純度（**基尼係數，信息增益，均方差，絕對差**）必須大於這個閾值，否則該節點不再生成子節點。

### ****class\_weight****：

類別權重，【dict, list of dicts, balanced】，默認爲None。（不適用於迴歸樹，sklearn.tree.DecisionTreeRegressor）

**指定樣本各類別的權重**，**主要是爲了防止訓練集某些類別的樣本過多，導致訓練的決策樹過於偏向這些類別**。**balanced**，算法自己計算權重，樣本量少的類別所對應的樣本權重會更高。如果樣本類別分佈沒有明顯的偏倚，則可以不管這個參數。

### ****random\_state****：

【int, randomSate instance, None】，默認是None

### 模型調參注意事項：

1、當樣本少數量但是樣本特徵非常多的時候，決策樹很容易過擬合，一般來說，樣**本數比特徵數多一些會比較容易建立健壯的模型**2、如果樣本數量少但是樣本特徵非常多，在擬合決策樹模型前，推薦**先做維度規約**，比如主成分分析（PCA），特徵選擇（Losso）或者獨立成分分析（ICA）。這樣特徵的維度會大大減小。再來擬合決策樹模型效果會好。  
3、推薦多用決策樹的可視化，同時先限制決策樹的深度（比如最多3層），這樣可以先觀察下生成的決策樹裏數據的初步擬合情況，然後再決定是否要增加深度。  
4、在訓練模型先，**注意觀察樣本的類別情況（主要指分類樹），如果類別分佈非常不均勻**，就要考慮用**class\_weight**來限制模型過於偏向樣本多的類別。  
5、決策樹的數組使用的是numpy的float32類型，如果訓練數據不是這樣的格式，算法會先做copy再運行。  
6、如果輸入的樣本矩陣是稀疏的，推薦在擬合前調用csc\_matrix稀疏化，在預測前調用csr\_matrix稀疏化。

### 其他：

如果使用默認DecisionTreeClassifier的參數，得到的AUC較低，很可能是因爲出現過擬合，需調整默認參數，避免過擬合。