

# 機器學習算法常用指標總結

深度學習愛好者 2022-03-25 08:05

點擊上方“深度學習愛好者”，選擇加“星標”或“置頂”

重磅乾貨，第一時間送達

作者：Poll

鏈接：<https://www.cnblogs.com/maybe2030/p/5375175.html>

編輯：石頭

機器學習性能評價標準是模型優化的前提，在設計機器學習算法過程中，不同的問題需要用到不同的評價標準，本文對機器學習算法常用指標進行了總結。

## 閱讀目錄

- 1. TPR、FPR&TNR
- 2. 精確率Precision、召回率Recall和F1值
- 3. 綜合評價指標F-measure
- 4. ROC曲線和AUC
- 5. 參考內容

考慮一個二分問題，即將實例分成正類（positive）或負類（negative）。對一個二分問題來說，會出現四種情況。如果一個實例是正類並且也被預測成正類，即為真正類（True positive），如果實例是負類被預測成正類，稱之為假正類（False positive）。相應地，如果實例是負類被預測成負類，稱之為真負類（True negative），正類被預測成負類則為假負類（false negative）。

- TP：正確肯定的數目；
- FN：漏報，沒有正確找到的匹配的數目；
- FP：誤報，給出的匹配是不正確的；
- TN：正確拒絕的非匹配對數；

列聯表如下表所示，1代表正類，0代表負類：

	预测1	预测0
实际1	True Positive(TP)	False Negative(FN)
实际0	False Positive(FP)	True Negative(TN)

### 1. TPR、FPR&TNR

從列聯表引入兩個新名詞。其一是**真正類率(true positive rate , TPR)**，計算公式為

$$TPR = TP / (TP + FN)$$

刻畫的是分類器所識別出的正實例佔所有正實例的比例。

另外一個是**負正類率(false positive rate, FPR)**，計算公式為

$$FPR = FP / (FP + TN)$$

計算的是分類器錯認為正類的負實例佔所有負實例的比例。

還有一個真負類率 (True Negative Rate, TNR)，也稱為specificity，計算公式為

$$TNR = TN / (FP + TN) = 1 - FPR$$

## 2. 精確率Precision、召回率Recall和F1值

精確率（正確率）和召回率是廣泛用於信息檢索和統計學分類領域的兩個度量值，用來評價結果的質量。其中精度是檢索出相關文檔數與檢索出的文檔總數的比率，衡量的是檢索系統的查准率；召回率是指檢索出的相關文檔數和文檔庫中所有的相關文檔數的比率，衡量的是檢索系統的查全率。

一般來說，Precision就是檢索出來的條目（比如：文檔、網頁等）有多少是準確的，Recall就是所有準確的條目有多少被檢索出來了，兩者的定義分別如下：

$$Precision = \text{提取出的正確信息條數} / \text{提取出的信息條數}$$

$$Recall = \text{提取出的正確信息條數} / \text{樣本中的信息條數}$$

為了能夠評價不同算法的優劣，在Precision和Recall的基礎上提出了F1值的概念，來對Precision和Recall進行整體評價。F1的定義如下：

$$F1 \text{ 值} = \text{正確率} * \text{召回率} * 2 / (\text{正確率} + \text{召回率})$$

**不妨舉這樣一個例子：**

某池塘有1400條鯉魚，300只蝦，300只鰱。現在以捕鯉魚為目的。撒一大網，逮著了700條鯉魚，200只蝦，100只鰱。那麼，這些指標分別如下：

$$\text{正確率} = 700 / (700 + 200 + 100) = 70\%$$

$$\text{召回率} = 700 / 1400 = 50\%$$

$$F1 \text{ 值} = 70\% * 50\% * 2 / (70\% + 50\%) = 58.3\%$$

不妨看看如果把池子裡的所有的鯉魚、蝦和鰱都一網打盡，這些指標又有何變化：

$$\text{正確率} = 1400 / (1400 + 300 + 300) = 70\%$$

$$\text{召回率} = 1400 / 1400 = 100\%$$

$$F1 \text{ 值} = 70\% * 100\% * 2 / (70\% + 100\%) = 82.35\%$$

由此可見，正確率是評估捕獲的成果中目標成果所占得比例；召回率，顧名思義，就是從關注領域中，召回目標類別的比例；而F值，則是綜合這二者指標的評估指標，用於綜合反映整體的指標。

當然希望檢索結果Precision越高越好，同時Recall也越高越好，但事實上這兩者在某些情況下有矛盾的。比如極端情況下，我們只搜索出了一個結果，且是準確的，那麼Precision就是100%，但是Recall就很低；而如果我們把所有結果都返回，那麼比如Recall是100%，但是Precision就會很低。因此在不同的場合中需要自己判斷希望Precision比較高或是Recall比較高。如果是做實驗研究，可以繪製Precision-Recall曲線來幫助分析。

### 3. 綜合評價指標F-measure

Precision和Recall指標有時候會出現的矛盾的情況，這樣就需要綜合考慮他們，最常見的方法就是F-Measure（又稱為F-Score）。

F-Measure是Precision和Recall加權調和平均：

$$F = \frac{(\alpha^2 + 1)P * R}{\alpha^2(P + R)}$$

當參數 $\alpha = 1$ 時，就是最常見的F1。因此，F1綜合了P和R的結果，當F1較高時則能說明試驗方法比較有效。

### 4. ROC曲線和AUC

#### 4.1 為什麼引入ROC曲線？

Motivation1：在一個二分類模型中，對於所得到的連續結果，假設已確定一個閾值，比如說0.6，大於這個值的實例劃歸為正類，小於這個值則劃到負類中。如果減小閾值，減到0.5，固然能識別出更多的正類，也就是提高了識別出的正例佔所有正例的比類，即TPR，但同時也將更多的負實例當作了正實例，即提高了FPR。為了形象化這一變化，引入ROC，ROC曲線可以用於評價一個分類器。

Motivation2：在類不平衡的情況下，如正樣本90個，負樣本10個，直接把所有樣本分類為正樣本，得到識別率為90%。但這顯然是沒有意義的。單純根據Precision和Recall來衡量算法的優劣已經不能表徵這種病態問題。

#### 4.2 什麼是ROC曲線？

ROC (Receiver Operating Characteristic) 翻譯為“接受者操作特性曲線”。曲線由兩個變量1-specificity 和Sensitivity繪製。1-specificity=FPR，即負正類率。Sensitivity即是真正類率，TPR(True positive rate),反映了正類覆蓋程度。這個組合以1-specificity對sensitivity,即是以代價(costs)對收益(benefits)。

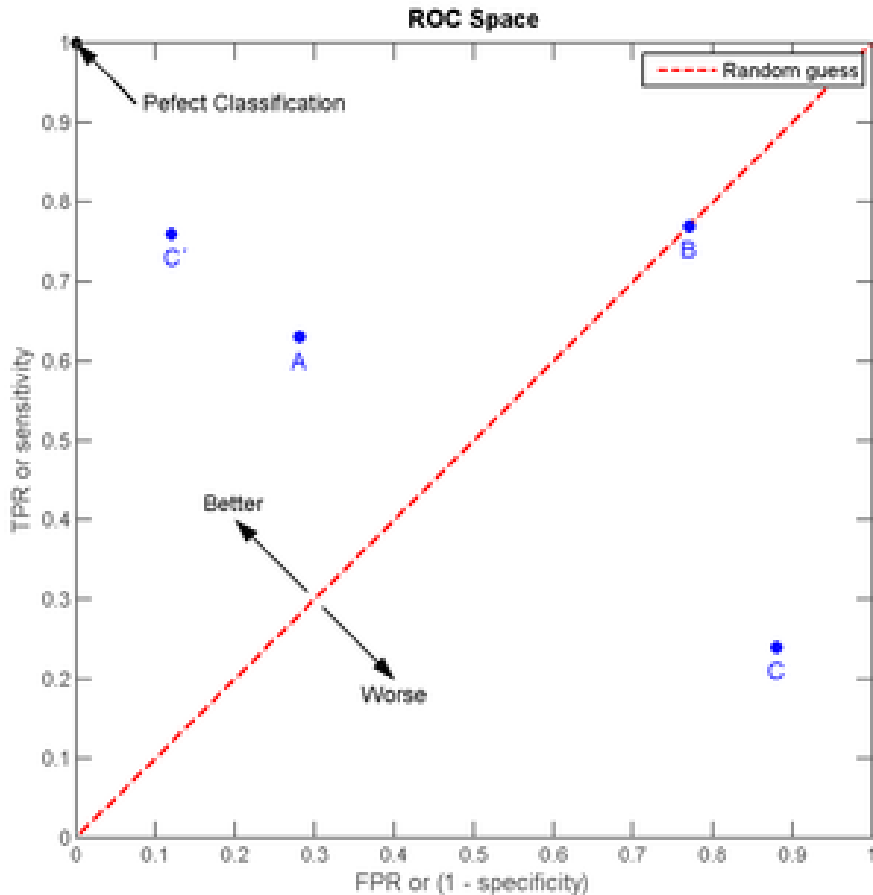
此外，ROC曲線還可以用來計算“均值平均精度”(mean average precision)，這是當你通過改變閾值來選擇最好的結果時所得到的平均精度 (PPV)。

为了更好地理解ROC曲線，我們使用具體的實例來說明：

如在醫學診斷中,判斷有病的樣本。那麼盡量把有病的揪出來是主要任務,也就是第一個指標TPR,要越高越好。而把沒病的樣本誤診為有病的,也就是第二個指標FPR,要越低越好。

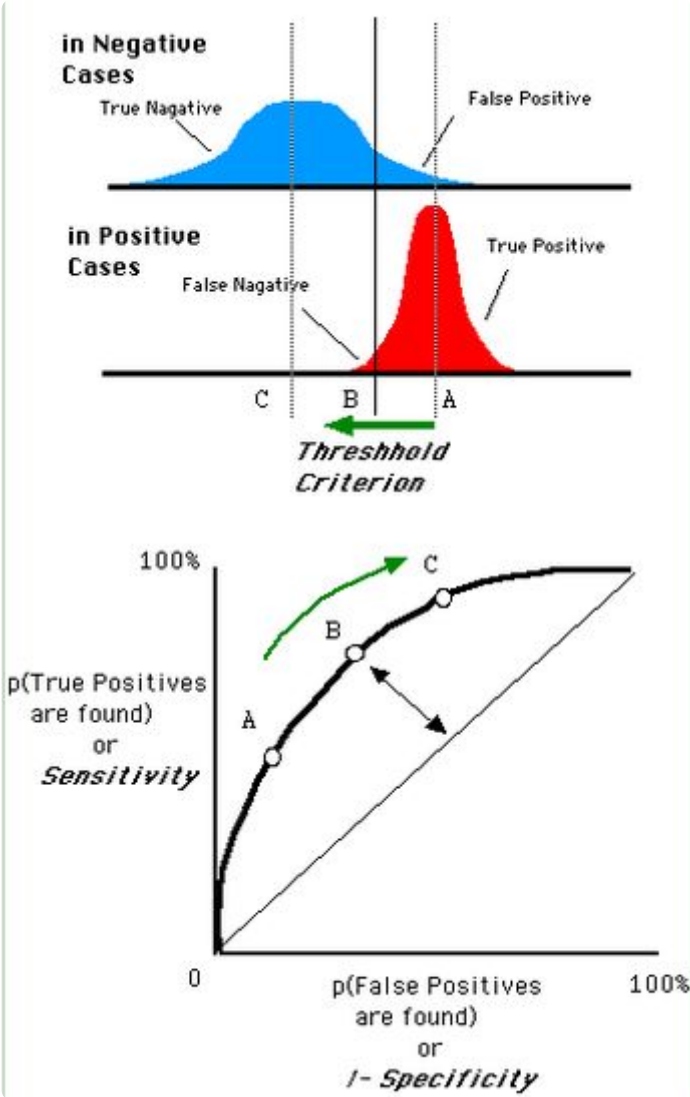
不難發現,這兩個指標之間是相互制約的。如果某個醫生對於有病的症狀比較敏感,稍微的小症狀都判斷為有病,那麼他的第一個指標應該會很高,但是第二個指標也就相應地變高。最極端的情況下,他把所有的樣本都看做有病,那麼第一個指標達到1,第二個指標也為1。

我們以FPR為橫軸,TPR為縱軸,得到如下ROC空間。

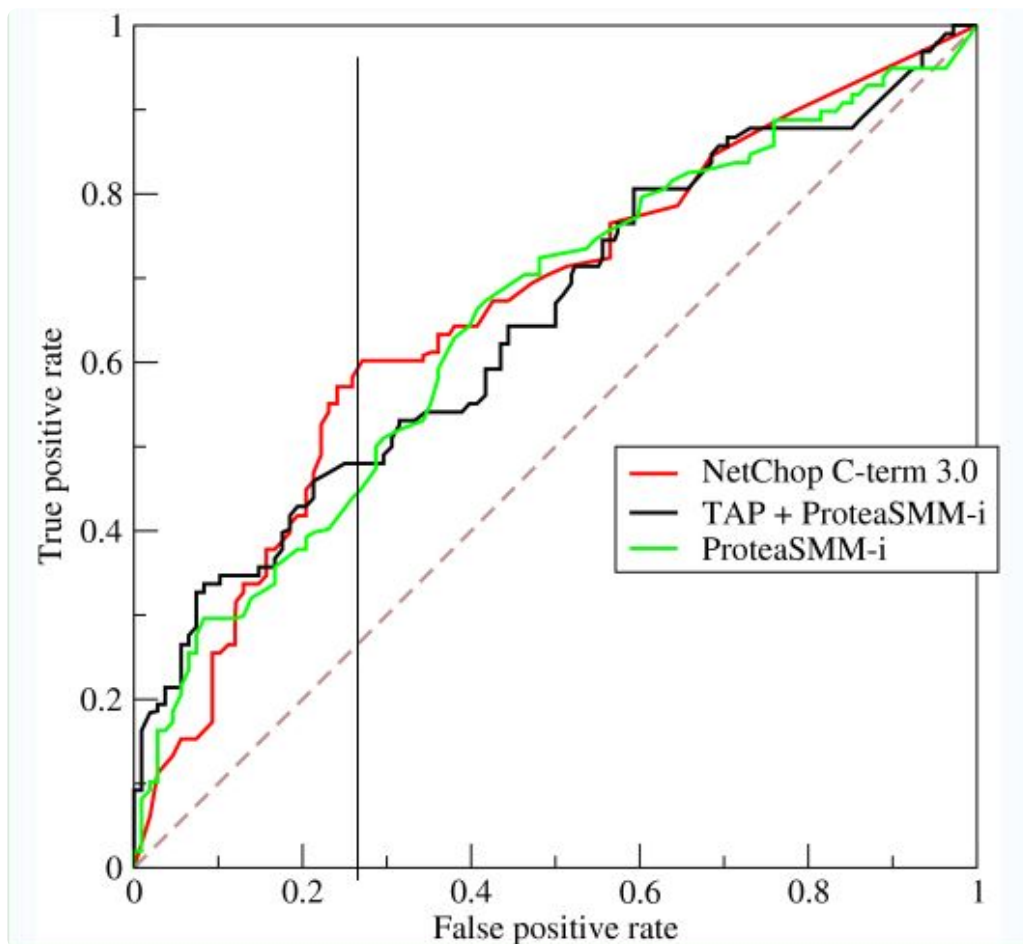


我們可以看出,左上角的點( $TPR=1, FPR=0$ ),為完美分類,也就是這個醫生醫術高明,診斷全對。點A( $TPR>FPR$ ),醫生A的判斷大體是正確的。中線上的點B( $TPR=FPR$ ),也就是醫生B全都是蒙的,蒙對一半,蒙錯一半;下半平面的點C( $TPR<FPR$ ),這個醫生說你有病,那麼你很可能沒有病,醫生C的話我們要反著聽,為真庸醫。上圖中一個閾值,得到一個點。現在我們需要一個獨立於閾值的評價指標來衡量這個醫生的醫術如何,也就是遍歷所有的閾值,得到ROC曲線。

還是一開始的那幅圖,假設如下就是某個醫生的診斷統計圖,直線代表閾值。我們遍歷所有的閾值,能夠在ROC平面上得到如下的ROC曲線。



曲線距離左上角越近, 證明分類器效果越好。



如上，是三條ROC曲線，在0.23處取一條直線。那麼，在同樣的低FPR=0.23的情況下，紅色分類器得到更高的PTR。也就表明，ROC越往上，分類器效果越好。我們用一個標量值AUC來量化它。

#### 4.3 什麼是AUC？

AUC值為ROC曲線所覆蓋的區域面積，顯然，AUC越大，分類器分類效果越好。

AUC = 1，是完美分類器，採用這個預測模型時，不管設定什麼閾值都能得出完美預測。絕大多數預測的場合，不存在完美分類器。

$0.5 < \text{AUC} < 1$ ，優於隨機猜測。這個分類器（模型）妥善設定閾值的話，能有預測價值。

AUC = 0.5，跟隨機猜測一樣（例：丟銅板），模型沒有預測價值。

AUC < 0.5，比隨機猜測還差；但只要總是反預測而行，就優於隨機猜測。

AUC的物理意義：假設分類器的輸出是樣本屬於正類的score（置信度），則AUC的物理意義為，任取一對（正、負）樣本，正樣本的score大於負樣本的score的概率。

#### 4.4 怎樣計算AUC？

第一種方法：AUC為ROC曲線下的面積，那我們直接計算面積可得。面積為一個個小的梯形面積之和。計算的精度與閾值的精度有關。

第二種方法：根據AUC的物理意義，我們計算正樣本score大於負樣本的score的概率。取 $N \times M$ （ $N$ 為正樣本數， $M$ 為負樣本數）個二元組，比較score，最後得到AUC。時間複雜度為 $O(N \times M)$ 。

第三種方法：與第二種方法相似，直接計算正樣本score大於負樣本的概率。我們首先把所有樣本按照score排序，依次用rank表示他們，如最大score的樣本，rank=n (n=N+M)，其次為n-1。那麼對於正樣本中rank最大的樣本，rank\_max，有M-1個其他正樣本比他score小，那麼就有(rank\_max-1)-(M-1)個負樣本比他score小。其次為(rank\_second-1)-(M-2)。最後我們得到正樣本大於負樣本的概率為

$$\frac{\sum_{\text{所有正样本}} \text{rank} - M(M+1)/2}{M * N}$$

時間複雜度為O(N+M)。

## 5. 參考內容

1. 機器學習指標大匯總：<http://www.36dsj.com/archives/42271>

### 下載1: Pytoch常用函數手冊

在「深度學習愛好者」公眾號后台留言：**Pytorch常用函數手冊**，即可下載全網第一份Pytorch常用函數手冊，涵蓋**Tensors介紹、基礎函數介紹、數據處理函數、優化函數、CUDA編程、多線程處理**等十四章章內容。



聲明：部分內容來源於網絡，僅供讀者學術交流之目的。文章版權歸原作者所有。如有不妥，請聯繫刪除。

喜歡此內容的人還喜歡

一文梳理深度學習算法演進

蘑菇先生學習記

---

機器學習二

七鹿山妖

---

機器學習一

七鹿山妖