

# Đánh giá mô hình phân loại trong Machine Learning

## 1 Đánh giá mô hình

Trong quá trình xây dựng model machine learning, chúng ta cần phải đánh giá model để biết được chất lượng của model, và để lựa chọn model phù hợp nhất với bài toán của mình. Bài viết này sẽ giới thiệu một số thước đo phổ biến để đánh giá model.

## 2 Bộ dữ liệu

**Bài toán:** Phân loại hồ sơ nợ xấu với nhãn: BAD - nợ xấu (positive), GOOD - không phải nợ xấu (negative)

**Bộ dữ liệu:**

- Tập train: 100 BAD và 900 GOOD
- Tập test: 15 BAD và 85 GOOD

**Kết quả phân loại:** trong bảng dưới đây

Predict/Actual		Actual	
		BAD (Positive)	GOOD (Negative)
Predict	BAD (Positive)	55 (TP - True Positive)	50 (FP - False Positive)
	GOOD (Negative)	45 (FN - False Negative)	850 (TN - True Negative)
Total		100	900

**Một số chỉ số:**

- TP (True Positive): Tổng số trường hợp dự báo đúng nhãn positive
- TN (True Negative): Tổng số trường hợp dự báo đúng nhãn negative
- FP (False Positive): Tổng số trường hợp dự báo nhãn negative thành positive
- FN (False Negative): Tổng số trường hợp dự báo nhãn positive thành negative

## 3 Accuracy

$$Accuracy = \frac{TP + TN}{total\_sample} = \frac{55 + 850}{100 + 900} = 90.5\%$$

```
# Example
from sklearn.metrics import accuracy_score
accuracy_score(y_true, y_pred)
```

Accuracy là metric khá được ưa chuộng vì nó có công thức tường minh và dễ diễn giải ý nghĩa. Tuy nhiên nó không quan tâm đến độ chính xác trên từng nhãn, do đó accuracy không phù hợp với những task mà tầm quan trọng của việc dự báo các nhãn không còn như nhau, ví dụ việc dự báo chính xác 1 bộ hồ sơ nợ xấu (positive) quan trọng hơn việc dự báo chính xác 1 bộ hồ sơ thông thường (negative).

## 4 Precision

$$Precision = \frac{TP}{total\_predicted\_positive} = \frac{TP}{TP + FP} = \frac{55}{55 + 50} = 52.4\%$$

Ý nghĩa: trong số các hồ sơ được dự đoán là positive thì có 52.4% số hồ sơ được dự đoán đúng.

## 5 Recall

$$Recall = \frac{TP}{total\_actual\_positive} = \frac{TP}{TP + FN} = \frac{55}{55 + 45} = 55\%$$

# Example

```
from sklearn.metrics import precision_recall_curve
precision_recall_curve(y_true, y_pred)
```

Ý nghĩa: trong số các hồ sơ có nhãn thực là positive thì có 55% số hồ sơ được dự đoán đúng.

Để tính recall chúng ta cần phải biết trước nhãn của tập dữ liệu, do đó recall được dùng đánh giá trên tập train và validation, còn precision được dùng đánh giá trên tập test.

## 6 F1 score

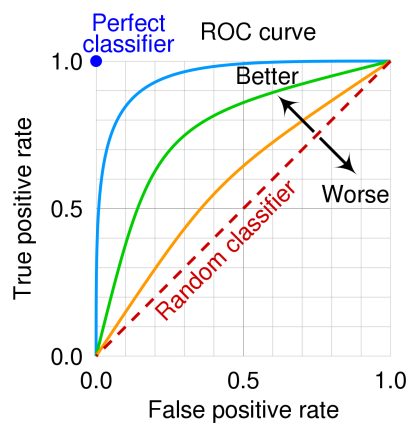
$$F_1 = \frac{2}{precision^{-1} + recall^{-1}} = \frac{2}{0.524^{-1} + 0.55^{-1}} = 53.7\%$$

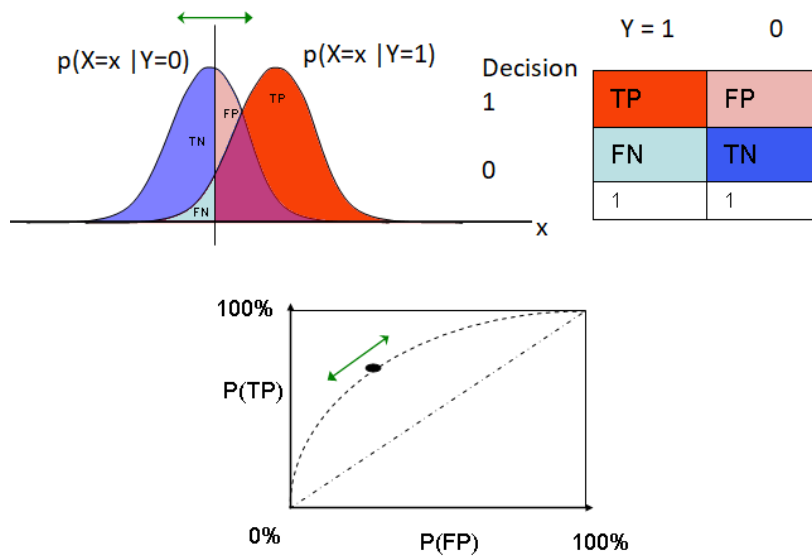
# Example

```
from sklearn.metrics import f1_score
f1_score(y_true, y_pred)
```

Trade off giữa precision và recall: model sẽ lựa chọn ngưỡng (mặc định 0.5) để quyết định nhãn, do đó precision và recall không có định mà biến đổi theo ngưỡng, và khi precision cao thì recall thấp và ngược lại. Khi không biết đánh giá model theo precision hay recall sẽ phù hợp hơn, chúng ta sẽ kết hợp cả *precision* và *recall*, đó là  $F_1$ . Quy ước nếu  $precision = 0$  hoặc  $recall = 0$  thì  $F_1 = 0$ , và giá trị  $F_1$  nằm giữa *precision* và *recall*.

## 7 AUC





ROC (Receiving Operating Curve) là đường cong đồng biến biểu diễn khả năng phân loại của model tại các ngưỡng khác nhau dựa trên 2 chỉ số:

- TPR (True Positive Rate): recall
- FPR (False Positive Rate): tỷ lệ dự báo sai các trường hợp thực tế là negative thành positive trên tổng số các trường hợp thực tế là negative.

$$FPR = \frac{FP}{total\_negative}$$

AUC (Area Under Curve) là phần diện tích phần nằm dưới ROC có giá trị trong đoạn  $[0, 1]$ . AUC càng lớn khi ROC càng tiệm cận đường  $y = 1$  và khả năng phân loại của mô hình càng tốt. Khi ROC nằm sát đường chéo thì model là model phân loại ngẫu nhiên.

```
# Example
from sklearn.metrics import auc, roc_curve
fpr, tpr, thres = roc_curve(y_true, y_pred)
auc(fpr, tpr)

# Draw ROC
def _plot_roc_curve(fpr, tpr, thres):
    roc = plt.figure(figsize = (10, 8))
    plt.plot(fpr, tpr, 'b-', label = 'ROC')
    plt.plot([0, 1], [0, 1], '--')
    plt.axis([0, 1, 0, 1])
    plt.xlabel('False Positive Rate')
    plt.ylabel('True Positive Rate')
    plt.title('ROC Curve')

_plot_roc_curve(fpr, tpr, thres)
```

## 8 MRR - Mean Reciprocal Rank

$$MRR = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \frac{1}{rank_i}$$

Example:

$$MRR = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \frac{1}{rank_i} = \frac{1}{3} \left( \frac{1}{3} + \frac{1}{2} + \frac{1}{1} \right) = \frac{11}{18} \approx 0.61$$

Query	Proposed Results	Correct response	Rank	Reciprocal rank
cat	catten, cati, <b>cats</b>	cats	3	1/3
torus	torii, <b>tori</b> , toruses	tori	2	1/2
virus	<b>viruses</b> , virii, viri	viruses	1	1

## 9 nDCG@p - Normalized Discounted Cumulative Gain

DCG là độ đo chất lượng xếp hạng. Nó thường được sử dụng trong các bài toán truy xuất thông tin như đo độ hiệu quả của thuật toán tìm kiếm bằng cách xếp hạng các bài viết mà nó hiển thị theo mức độ liên quan với từ khóa tìm kiếm. Cụ thể, hãy xét  $p = 5$  bài viết output cùng mức độ liên quan với từ khóa tìm kiếm:  $(D_1, 3), (D_2, 2), (D_3, 0), (D_4, 0), (D_5, 1)$

$$CG_p = \sum_{i=1}^p rel_i = 3 + 2 + 0 + 0 + 1 = 6$$

$$DCG_p = \sum_{i=1}^p \frac{rel_i}{\log_2(i+1)} = \frac{3}{\log_2 2} + \frac{2}{\log_2 3} + \frac{0}{\log_2 4} + \frac{0}{\log_2 5} + \frac{1}{\log_2 6} \approx 4.67$$

$$IDCG_p = \sum_{i=1}^p \frac{rel_{i\_order}}{\log_2(i+1)} = \frac{3}{\log_2 2} + \frac{2}{\log_2 3} + \frac{1}{\log_2 4} + \frac{0}{\log_2 5} + \frac{0}{\log_2 6} \approx 4.76$$

$$nDCG@p = \frac{DCG_p}{IDCG_p} = \frac{4.67}{4.76} \approx 0.98$$

## References

1. <https://phamdinhhkhanh.github.io/2020/08/13/ModelMetric.html>
2. [https://en.wikipedia.org/wiki/Mean\\_reciprocal\\_rank](https://en.wikipedia.org/wiki/Mean_reciprocal_rank)
3. <https://www.geeksforgeeks.org/normalized-discounted-cumulative-gain-multilabel-ranking-metrics-ml/>

Chu Dinh Duc  
25/12/2022