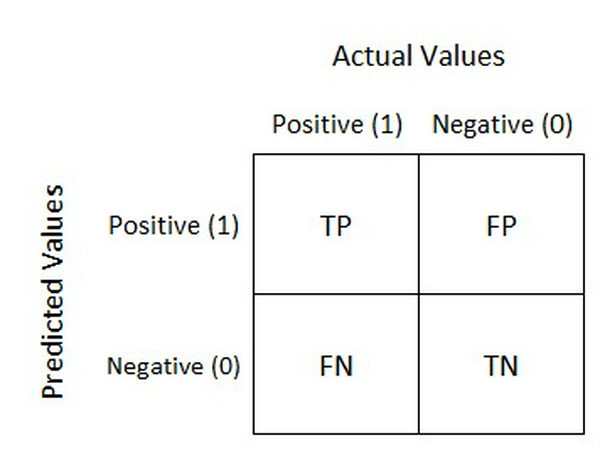
**PHÂN TÍCH, ĐÁNH GIÁ KẾT QUẢ RANDOM FOREST**

Thuật toán Random Forest có tính ưu việt hơn so với Decision tree do là từ nhiều Decision tree hoạt động dự đoán và tổng hợp các kết quả dự đoán trả về theo số đông từ các Decision tree để đưa đến dự đoán trực quan nhất cho một vấn đề đang muốn tiến hành phân tích. Tuy nhiên, thuật toán này không hẳn luôn có dộ chính xác hoàn toàn từ việc dự đoán so với kết quả thực tế xảy ra. Vì thế, ta cần tiến hành những phương thức đánh giá hiệu suất của thuật toán thông qua một vài chỉ số cơ bản nhằm có thể nhận biết phần trăm đúng hay sai và độ chênh lệch giữa dúng với sai của thuật toán. Thực hiện việc đánh giá, ta sẽ sử dụng Confusion matrix dể đánh giá kết quả phân loại qua việc xem xét những chỉ số về độ chính xác và độ bao quát của những dự đoán với từng lớp. Confusion matrix có bốn chỉ số:



* TP (true positive): số lượng dự đoán chính xác khi mô hình dự đoán đúng kết quả xảy ra trong thực tế
* TN (true negative): số lượng dự đoán chính xác gián tiếp
* FP (false positive): số lượng dự đoán sai lệch khi mô hình dự đoán sai kết quả thực tế
* FN (false negative): số lượng dự đoán sai lệch gián tiếp

Từ 4 chỉ số đã nêu, ta có một số phương pháp đánh giá mức độ tin cậy cho một mô hình:

* Precision: chuẩn xác khi phát hiện các điểm positive. Số càng cao thì mô hình nhận điểm positive càng chuẩn. Công thức triển khai:

FP càng thấp, Precision sẽ càng cao → mô hình càng chính xác vì FP là chỉ số dự đoán Positive sai so với dữ liệu thực nên khi chỉ số này càng thấp, tính chính xác khi dự đoán của mô hình sẽ càng nâng cao.

* Recall: khả năng phát hiện tất cả positive, tỷ lệ càng cao thì khả năng bỏ sót điểm positive càng thấp. Công thức triển khai:

FN càng thấp, Recall càng tăng → mô hình dự đoán chính xác cao

* F-Score: kết hợp 2 chỉ số Precision và Recall thành chỉ số duy nhất nhằm đánh giá độ tin cậy chung của mô hình. Công thức triển khai:

F-Score chỉ cao khi cả hai chỉ số Precision và Recall đều cao. Một trong hai thấp đều kéo F-Score xuống. Trường hợp xấu nhất khi một trong hai chỉ số là 0 thì F-Score sẽ là 0. Ngược lại, tốt nhất khi cả hai đạt 1, F-Score sẽ bằng 1.

Việc sử dụng F-Score sẽ là thước đo đáng tin cậy về hiệu năng của mô hình trong bài toán phân loại, đặc biệt dữ liệu một lớp lớn hơn gấp nhiều lần dữ liệu các lớp còn lại.

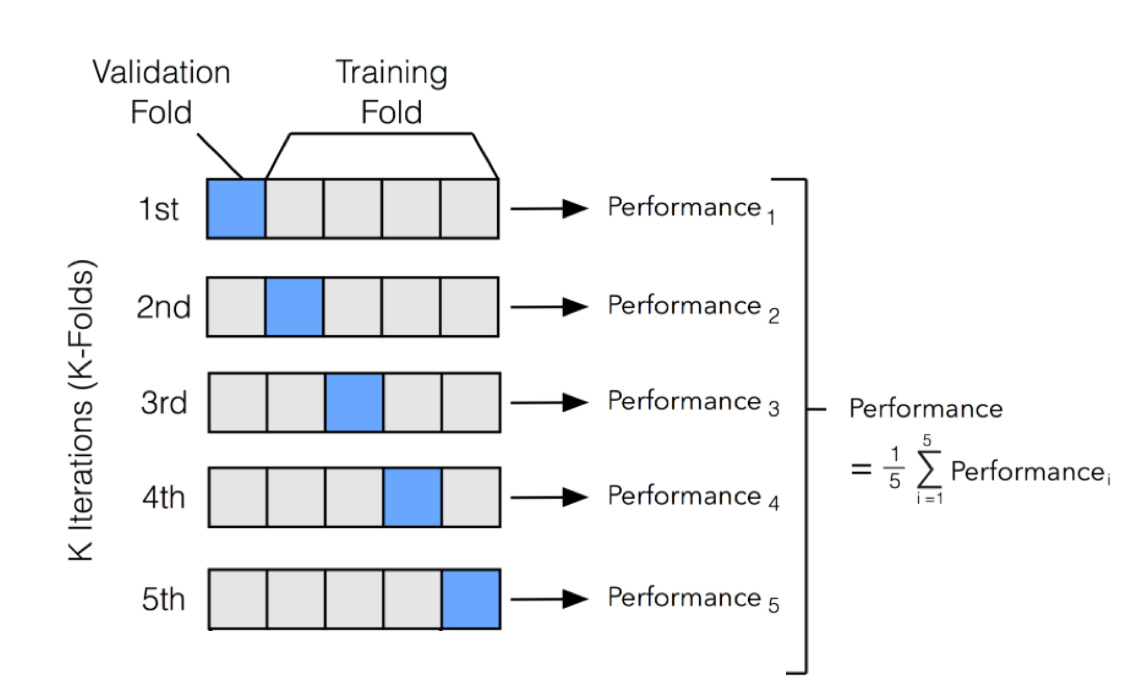
* Accuracy: độ chính xác mà mô hình dự đoán đúng với kết quả thực tế của bộ dữ liệu gốc. Công thức triển khai:

Ngoài Confusion matrix, ta cũng có thể dùng một vài độ đánh giá khác để mở rộng cách thức đánh giá và hình thành độ tin cậy dành cho Random Forset chằng hạn như Cross validation để khắc phục Overfitting

* Overfitting: hiện tượng xảy ra khi mô hình làm việc với tập dữ liệu huấn luyện có kết quả rất tốt nhưng khi làm việc với tập dữ liệu thực tế lại đưa ra những kết quả rất kém. Mô hình không xử lý được những bộ dữ liệu mới khi chưa được trải qua huấn luyện hay mô hình có tính phức tạp cao là yếu tố mà Overfitting biểu lộ.
* Cross validtion: giả mạo tình trạng dữ liệu mà mô hình chưa từng thấy, từ đây đánh giá khả năng tổng quát của mô hình

+ Holdout Validation: quá trình huấn luyện mô hình, một phần dữ liệu được giữ lại và không dùng để huấn luyện mô hình (tập kiểm thử). Mô hình đã được huấn luyện sẽ được đánh giá qua tập kiểm thử nhằm đo lường hiệu suất. Mô hình chỉ được đánh giá trên một tập kiểm thử cụ thể, dẫn đến đánh giá không chính xác hiệu suất mô hình

+ K-fold Cross-validation: dữ liệu chia thành k fold (phần) nhỏ, mô hình được huấn luyện qua k lần, mỗi lần sẽ dùng fold khác nhau nhằm làm tập kiểm thử, fold còn lại dùng cho huấn luyện. Kết quả của k lần huấn luyện được tính trung bình để có kết quả cuối cùng. Kỹ thuật này đánh giá mô hình qua nhiều tập kiểm thử khác nhau, hiệu suất được ước lượng sẽ chính xác hơn



Cross-Validation giúp đánh giá chính xác hiệu suất mô hình với nhiều tập dữ liệu và đảm bảo mô hình hoạt động tốt trên dữ liệu mới nên giảm nguy cơ Overfitting. Nó cũng hữu ích khi tối ưu hóa tham số của mô hình và đảm bảo tính ổn định của các đánh giá hiệu suất.

Tuy nhiên, Cross-Validation có một vài hạn chế cần lưu ý xem xét khi sử dụng khắc phục overfitting:

+ Cross-Validation dùng hiệu quả với dữ liệu số lượng lớn, trường hợp dữ liệu quá nhỏ, việc chia fold có thể làm giảm khả năng đại diện của mẫu, gây nhiễu loạn

+ Sử dụng K-fold Cross-Validation khi số lượng k lớn, mô hình đòi hỏi cần phải huấn luyện, đánh giá k lần gây tốn kém về tính toán, đặc biệt với mô hình phức tạp hay tập dữ liệu lớn

+ Quá trình thực hiện Cross-Validation có nguy cơ lựa chọn siêu tham số (hyperparameters) theo hiệu suất của tập kiểm thử có thể dẫn đến overfitting trên tập kiểm thử

+ Phương pháp chia fold cũng gây ảnh hưởng kết quả Cross-Validation vào trường hợp dữ liệu chia không ngẫu nhiện và có bias, có thể dẫn đến ước lượng sai lệch hiệu suất mô hình

+ Dữ liệu không đồng nhất, các fold có sự khác biệt và biến động lớn dẫn đến sự không chắc chắn trong việc đánh giá hiệu suất mô hình

Mặc dù có những hạn chế, Cross-Validation vẫn là công cụ quan trọng, đắc lực trong xây dựng mô hình lẫn đánh giá hiệu suất khi sử dụng đúng cách và đúng hoàn cảnh

Hiệu suất từ Random Forest phụ thuộc nhiều yếu tố từ số lượng Decision tree, số đặc trưng chọn ngẫu nhiên cho mỗi Decision tree. Điều chỉnh các đặc trưng hay yếu tố tác động đến các cây thì sẽ cải thiện hiệu suất cho Random Forest. Tùy thuộc vào mục đích thực hiện bài toán cụ thể, cần điều chỉnh hay thử nghiệm các tham số để đạt được trạng thái tốt nhất của mô hình.