\*Thuật toán Random Forest

-Sử dụng Cây quyết định(decision tree).

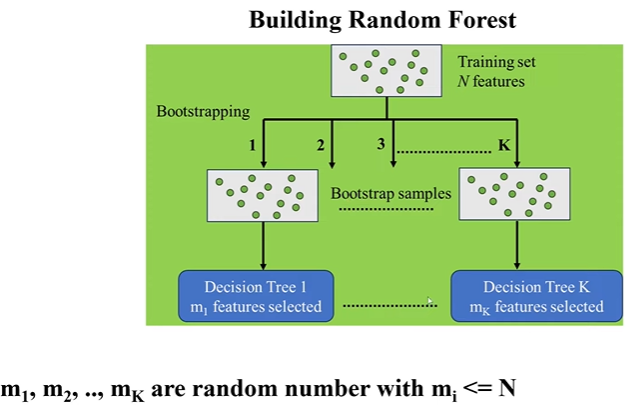
+Ưu điểm:dễ dàng xây dựng và sử dụng và giải thích

+Nhược điểm: rất nhạy cảm với training data(nếu thay đổi training data thì result sẽ khác) => sai số phương sai khá lớn(high Variance)=>Overfitting.

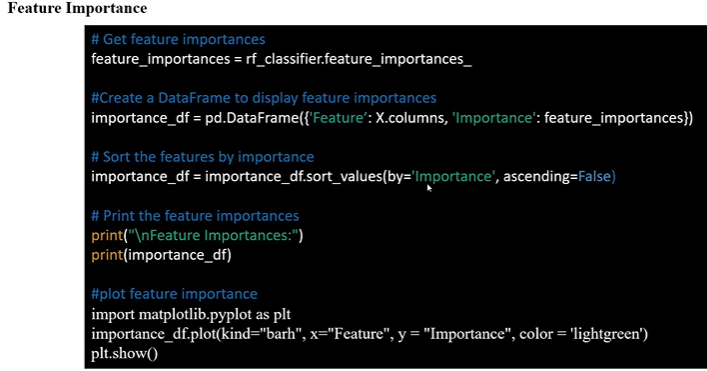
-Random Forest sẽ tận dụng ưu điểm và khắc phục nhược điểm của decision tree.

-Ý tưởng: xây dựng nhiều cây khác nhau trong 1 bộ dữ liệu sau đó tổng hợp tất cả các cây đưa ra predict cuối cùng (bằng vote hoặc lấy trung bình).

+Random các mẫu(bootstrap samples) để thực hiện cho các cây training nhưng các bootstrap samples sẽ được random các features cho mỗi cây=>Random forest làm cho đa dạng dữ liệu và đa dạng cây giúp giảm overfitting và kết quả tốt hơn.



-Có hàm feature\_importances\_ để lấy các feature quan trọng.



-Các tham số:n\_estimation:số cây(100-1000) default= 100.

+Bootstrap:Đây là cách tạo ra mẫu của câyTrue(default=True)

+Max\_features: số feature tối đa của mỗi mẫu có thể là từ 1 đến n-features (default=sqrt(n-features)).

+oob\_score: để đánh giá độ chính xác của mô hình không sử dụng test set (default=False).

+random\_state: được sử dụng để kiểm soát quá trình sinh số ngẫu nhiên. Ex random\_state=42. Đảm bảo rằng mỗi khi mô hình được huấn luyện lại, nó sẽ tạo ra cùng một bộ cây ngẫu nhiên và kết quả sẽ giống nhau, giúp việc so sánh mô hình trở nên nhất quán.

+Max\_depth:10-20(default=None).

+Min\_sample\_split:2-20(default=2).

+min\_samples\_leaf:Số lượng mẫu tối thiểu cho 1 nút lá :1-5(default=1).

+Ba giá trị này chỉ cần setup 1 giá trị là đủ