# Module 4 Tuần 1
# A Gentle Introduction Ada Boost

Time-Series Team

Ngày 5 tháng 9 năm 2025

Buổi học thứ 6 (ngày 5/9/2025) được chia thành 6 phần chính đi từ tổng quát đến chi tiết và công thức phía sau thuật toán này

- **Phần 1: Kĩ thuật Boosting**
- **Phần 2: Trực giác đằng sau AdaBoost**
- **Phần 3: Công thức toán đằng sau AdaBoost**

## Phần 1: Kĩ thuật Boosting

Một cây quyết định (gọi tắt là DS) đơn lẻ thường không đủ mạnh để xử lý dữ liệu thực tế: nếu quá nông (1 node) thì chỉ dựa trên 1 đặc trưng duy nhất dẫn đến bỏ sót thông tin quan trọng hoặc nếu quá sâu (full decision tree) thì cây dễ ghi nhớ dữ liệu huấn luyện và mất khả năng khái quát. Chính vì hạn chế này mà các phương pháp **Essemble Learning** ra đời, nhằm kết hợp nhiều mô hình con để tận dụng ưu điểm và hạn chế nhược điểm của từng mô hình đơn, chi tiết thế nào mình cùng đi tiếp nhé.

### 1.1 Điểm yếu của Decision Tree

Một cây quyết định quá nông gồm 1 node được gọi là **Stump** (dịch ra là 1 đoạn của cây) là chỉ chia dữ liệu dựa trên 1 đặc trưng duy nhất, ví dụ "nếu cân nặng $\geq$ 70kg thì dự đoán bệnh tim = "có" cho thấy rõ ràng các yếu tố như tuổi, huyết áp và động mạch bị tắc, etc.. không được cân nhắc do thiếu nhánh để quyết định. Do đó, cây nông bị underfitting.

Ngược lại, 1 DS quá sâu có thể tạo ra hàng chục hoặc hàng trăm nhánh, quá phù hợp với dữ liệu huấn luyện nhưng lại thất bại khi dự đoán dữ liệu mới dẫn đến overfitting.

| Mô hình | Hàm loss Regression | Ý nghĩa & Cải tiến |
|---|---|---|
| **Decision Tree** | $MSE = \frac{1}{n}\sum(y_i - \hat{y}_i)^2$ | Dự đoán giá trị trung bình ở mỗi lá. Cắt theo tiêu chí giảm MSE nhiều nhất |
| **Random Forest** | MSE (từng cây), dự đoán trung bình | Giảm overfitting bằng **bagging** (lấy mẫu bootstrap + chọn random feature). Không thay đổi loss |
| **AdaBoost** | Adaboost Loss | Được dùng trong việc cập nhật các sample |
| **Gradient Boosting** | Bất kỳ hàm khả vi | Gradient Boosting coi bài toán là tối ưu hàm loss bất kỳ qua gradient descent (MSE) |
| **XGBoost** | Hàm loss khả vi + Regularization | Thêm L1, L2 để kiểm soát độ phức tạp của cây, giảm overfitting |
| **LightGBM** | Như XGBoost | Như XGBoost |

Figure 1: Random Forest (nhiều Decision Tree vừa và sâu) vs Ada Boost (nhiều cây nông / Stump)