

Những Điểm Đặc Biệt (Quirks) của XGBoost

MỤC TIÊU BÀI HỌC

Sau bài học này, bạn sẽ:

- Biết được 3 đặc điểm quan trọng khi sử dụng XGBoost.
- Hiểu cách xử lý biến phân loại, giá trị thiếu (NA), và mối quan hệ phi tuyến.
- Biết cách tránh sai lầm khi tiền xử lý dữ liệu cho XGBoost.

I. GIỚI THIỆU NGẮN GỌN

XGBoost là một thuật toán mạnh mẽ nhưng có một số **đặc điểm riêng biệt (quirks)** mà người dùng cần hiểu rõ để sử dụng hiệu quả. Cụ thể có **3 điểm chính** như sau:

BIẾN PHÂN LOẠI (Categorical Variables)

Vấn đề:

- XGBoost không hiểu kiểu dữ liệu “character” hay “factor”** (chuỗi hoặc phân loại).
- Bạn cần **chuyển đổi chúng thành biến giả (dummy variables)** trước khi đưa vào mô hình.

Ví dụ:

Giả sử bạn có cột **“Giới tính”** với 2 giá trị:

- Nam
- Nữ

Bạn cần chuyển thành:

Giới tính_Nam	Giới tính_Nữ
1	0

Giới tính_Nam	Giới tính_Nữ
0	1

“ Tuy nhiên, bạn **không nên giữ cả 2 cột** này cùng lúc!

Đây là cái gọi là “**Dummy Variable Trap**”, vì việc giữ cả 2 cột sẽ tạo ra **đa cộng tuyến (multicollinearity)**.

Giải pháp:

- Giữ lại 1 cột duy nhất (ví dụ chỉ giữ "Giới tính_Nam")
- Cột còn lại sẽ được suy ra.

II. GIÁ TRỊ THIẾU (Missing Values - NA)

Điểm đặc biệt của XGBoost:

- **Không cần loại bỏ hay thay thế NA.**
- XGBoost **coi giá trị NA là một thông tin riêng biệt.**

Ví dụ minh họa:

Giả sử bạn có cột “Thu nhập”:

ID	Thu nhập
1	5 triệu
2	NA
3	7 triệu
4	NA

Nếu bạn đang dùng hồi quy tuyến tính hay Random Forest, bạn **phải thay thế NA** bằng trung bình hoặc loại bỏ dòng.

☐ Nhưng với XGBoost:

- Nó **xem NA là một nhánh riêng trong cây quyết định.**
- Giúp giữ lại ý nghĩa của việc “không có thông tin”.

Tại sao điều này quan trọng?

Trong thực tế:

- Nhiều khách hàng không cung cấp đầy đủ dữ liệu.
- Việc giữ nguyên NA sẽ giúp mô hình hiểu rằng **“việc không cung cấp thông tin” là một tín hiệu riêng biệt.**

III. MỐI QUAN HỆ PHI TUYẾN (Non-linearity)

Lợi thế của XGBoost:

- Hiểu được mối quan hệ **phi tuyến (non-linear)** giữa biến độc lập và biến phụ thuộc.

Ví dụ:

Bạn có thể gặp:

- Mối quan hệ hình chữ U (U-shape)
- Mối quan hệ hình chữ S (S-shape)

Với các mô hình hồi quy tuyến tính, bạn phải:

- Biến đổi biến thủ công (thêm x^2 , \log , v.v.)

❌ Nhưng XGBoost thì:

- Tự động học được mối quan hệ phức tạp đó** thông qua cây quyết định.

IV. TỔNG HỢP QUA VÍ DỤ THỰC TẾ

Giả sử bạn có tập dữ liệu về khách hàng:

Tuổi	Giới tính	Thu nhập	Mua hàng
25	Nam	5 triệu	1
32	Nữ	NA	0
40	Nam	10 triệu	1

Bạn cần làm gì?

- Giới tính** → chuyển thành biến giả: giữ "Giới_tính_Nam"
- Thu nhập NA** → giữ nguyên (không thay thế)
- Dùng XGBoost để học mô hình → mô hình vẫn hoạt động tốt

KẾT LUẬN

Đặc điểm	Mô tả	Xử lý như thế nào
Biến phân loại	Không hỗ trợ trực tiếp	Dùng dummy variables, tránh trap

Đặc điểm	Mô tả	Xử lý như thế nào
NA (giá trị thiếu)	XGBoost tự xử lý	Không cần thay thế
Phi tuyến	Học tốt các mối quan hệ phi tuyến	Không cần biến đổi thủ công

Tác giả: **Đỗ Ngọc Tú**
Công Ty Phần Mềm **VHTSoft**

Phiên bản #1
Được tạo 15 tháng 5 2025 09:49:12 bởi Đỗ Ngọc Tú
Được cập nhật 15 tháng 5 2025 09:57:20 bởi Đỗ Ngọc Tú