Module 6 - Tuần 1 - Các Thước Đo Đánh Giá Mô Hình Hồi Quy

(Evaluation Metrics for Regression)*

Time-Series Team

Ngày 21 tháng 11 năm 2025

1 Giới thiệu

Trong bất kỳ bài toán hồi quy nào – từ dự báo giá nhà, nhu cầu năng lượng cho đến dự đoán nhu cầu thuốc hay cường độ tải trọng trong kỹ thuật – việc **đánh giá mô hình** quan trọng không kém gì việc xây dựng mô hình. Không có mô hình nào dự đoán chính xác 100% mọi điểm dữ liệu. Câu hỏi thực tế hơn luôn là: mô hình sai lệch bao nhiêu so với thực tế, sai lệch đó có đủ nhỏ để chấp nhận trong bối cảnh ứng dụng, và thước đo nào phản ánh đúng "chất lượng dự đoán" mà ta quan tâm.

Bài blog này được xây dựng từ hai nguồn chính: một bài tổng quan học thuật về các thước đo cho biến liên tục trong mô hình hồi quy và học máy, và một bộ slide giảng dạy về *Evaluation Metrics for Regression (From metrics to loss functions)*, trong đó phân loại metric theo nhóm: sai số phụ thuộc thang đo, sai số phần trăm, sai số tương đối, thước đo tương đối và các dạng sai số được chuẩn hoá.

Nguồn tham khảo chính

Bài viết này dựa trên hai hướng tiếp cận:

1. Một bộ 14 thước đo lỗi cho biến liên tục, gồm các chỉ số như Mean Bias (MB), Mean Absolute Error (MAE / MAGE), Root Mean Squared Error (RMSE), các dạng sai số chuẩn hoá, hệ số tương quan Pearson R, VAF, v.v., kèm phân tích ưu/nhược và ví dụ số.
2. Bộ slide về *Evaluation Metrics for Regression*, trong đó hệ thống hoá metric theo nhóm (scale-dependent, percentage, relative, scaled errors, others) và liên kết chúng với hàm mất mát dùng để huấn luyện mô hình.

Mục tiêu là không chỉ liệt kê công thức, mà còn dẫn người đọc đi theo một câu chuyện: từ cách định nghĩa sai số, đến việc lựa chọn metric phù hợp cho từng tình huống, nhìn thấy rõ các bẫy phổ biến (như MAPE khi giá trị thực gần 0, hay RSE khi có outlier), và cuối cùng là kết nối metric đánh giá với hàm mất mát dùng trong tối ưu mô hình.

2 Thiết lập vấn đề và ký hiệu cơ bản

Giả sử ta có một tập dữ liệu gồm N quan sát. Tại mỗi quan sát i:

- r_i là giá trị **thực** (real / target),
- p_i là giá trị **dự đoán** (prediction),
- sai số tại điểm đó là $e_i = p_i - r_i$.

*Dựa trên Plevris et al., *Investigation of Performance Metrics in Regression Analysis and Machine Learning-Based Prediction Models*, và Muraina et al., *Data Analytics Evaluation Metrics Essentials: Measuring Model Performance in Classification and Regression*.

Ta ký hiệu vector:

$$\mathbf{r} = (r_1, \ldots, r_N)^\top, \quad \mathbf{p} = (p_1, \ldots, p_N)^\top.$$

Giá trị trung bình của biến thực và biến dự đoán lần lượt là:

$$\bar{r} = \frac{1}{N}\sum_{i=1}^{N} r_i, \qquad \bar{p} = \frac{1}{N}\sum_{i=1}^{N} p_i.$$

Trong các phần sau, ta sẽ sử dụng cùng một bộ ví dụ nhỏ để tính tay, nhằm so sánh hành vi của các metric dưới những kịch bản khác nhau.

Ví dụ dữ liệu cơ bản. Giả sử ta có 5 quan sát về giá nhà (đơn vị: nghìn USD):

$$\text{Actual } \mathbf{r} = (200, 300, 250, 500, 300),$$

và ba mô hình dự đoán:

$$\mathbf{p}^{(1)} = (210, 290, 260, 500, 300),$$
$$\mathbf{p}^{(2)} = (210, 290, 260, 500, 600),$$
$$\mathbf{p}^{(3)} = (210, 290, 260, 800, 600).$$

Mô hình (1) gần đúng với tất cả các điểm; mô hình (2) mắc một lỗi khá lớn ở điểm cuối; mô hình (3) mắc hai lỗi rất lớn. Ta sẽ quay lại ví dụ này nhiều lần để xem metric nào "nhạy" với từng loại sai số.

3 Sai số phụ thuộc thang đo: ME, MAE, MSE, RMSE

Mean Error (ME): đo thiên lệch trung bình

Mean Error (ME) đơn giản là trung bình cộng của sai số có dấu:

$$ME = \frac{1}{N}\sum_{i=1}^{N}(p_i - r_i) = \bar{p} - \bar{r}.$$

Nếu $ME > 0$, mô hình có xu hướng dự đoán cao hơn thực tế (overestimate); nếu $ME < 0$, mô hình thường dự đoán thấp hơn thực tế (underestimate). Khi $ME \approx 0$, ta nói rằng về trung bình mô hình không thiên lệch.

Áp dụng cho $\mathbf{p}^{(1)}$:

$$e^{(1)} = (10, -10, 10, 0, 0), \quad ME^{(1)} = \frac{10 - 10 + 10 + 0 + 0}{5} = 2.$$

Với $\mathbf{p}^{(2)}$:

$$e^{(2)} = (10, -10, 10, 0, 300), \quad ME^{(2)} = \frac{10 - 10 + 10 + 0 + 300}{5} = 62.$$

Ta thấy ME tăng mạnh ở mô hình (2) vì có một sai số rất lớn. Tuy nhiên, nếu sai số dương và âm cân bằng, ME có thể bằng 0 ngay cả khi mô hình rất tệ. Vì vậy ME chủ yếu dùng để phát hiện *hướng thiên lệch*, chứ không phải để đánh giá chất lượng tổng thể.

Mean Absolute Error (MAE): mức sai lệch trung bình dễ hiểu

MAE được định nghĩa là:

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|p_i - r_i|.$$

Với ví dụ trên:

$$MAE^{(1)} = \frac{|10| + |-10| + |10| + 0 + 0}{5} = \frac{30}{5} = 6,$$
$$MAE^{(2)} = \frac{|10| + |-10| + |10| + 0 + |300|}{5} = \frac{330}{5} = 66.$$

MAE cùng đơn vị với biến mục tiêu (nghìn USD), nên ta có thể nói: "trung bình, dự đoán lệch 6 nghìn USD" (mô hình 1) hay 66 nghìn USD (mô hình 2). MAE **không cho phép** sai số dương và âm triệt tiêu nhau, nên tránh được vấn đề của ME.

Tuy nhiên, ví dụ về giá nhà cũng cho thấy một điểm yếu khác của MAE: nếu ta có 4 lỗi nhỏ và 1 lỗi rất lớn, MAE vẫn chỉ là một con số trung bình duy nhất, không cho ta biết rằng tồn tại một quan sát có lỗi cực kỳ nghiêm trọng. Chẳng hạn, nếu ta thay mô hình (2) bằng mô hình (3) với hai outlier rất lớn, MAE vẫn chỉ phản ánh mức "trung bình", không kể được câu chuyện chi tiết về phân bố sai số. Ở ví dụ giá nhà, điểm yếu của MAE hiện lên rõ hơn khi ta nhìn kỹ vào từng sai số chứ không chỉ nhìn một con số trung bình. Nhớ lại bộ dữ liệu thực

$$\mathbf{r} = (200, 300, 250, 500, 300),$$

và hai mô hình:

$$\mathbf{p}^{(2)} = (210, 290, 260, 500, 600), \qquad \mathbf{p}^{(3)} = (210, 290, 260, 800, 600).$$

Với mô hình $\mathbf{p}^{(2)}$, sai số tại từng quan sát là:

$$e^{(2)} = \mathbf{p}^{(2)} - \mathbf{r} = (10, -10, 10, 0, 300),$$

nên sai số tuyệt đối là $(10, 10, 10, 0, 300)$. Tổng sai số tuyệt đối bằng $10+10+10+0+300 = 330$, vì vậy

$$MAE^{(2)} = \frac{330}{5} = 66.$$

Ta thấy có bốn điểm gần như "ổn" (lỗi chỉ 0 hoặc 10 nghìn USD), nhưng một điểm cuối cùng bị dự đoán lệch đến 300 nghìn USD. Tuy nhiên, khi nén tất cả vào MAE, ta chỉ còn lại con số 66, hoàn toàn không biết được rằng trong 5 căn nhà có một căn bị dự đoán cực kỳ sai.

Giờ nếu ta chuyển sang mô hình $\mathbf{p}^{(3)}$:

$$\mathbf{p}^{(3)} = (210, 290, 260, 800, 600),$$

thì sai số trở thành

$$e^{(3)} = \mathbf{p}^{(3)} - \mathbf{r} = (10, -10, 10, 300, 300),$$

và sai số tuyệt đối là $(10, 10, 10, 300, 300)$. Lúc này tổng sai số tuyệt đối bằng $10+10+10+300+300 = 630$, nên

$$MAE^{(3)} = \frac{630}{5} = 126.$$

Rõ ràng, MAE đã tăng từ 66 lên 126, phản ánh việc mô hình (3) tệ hơn mô hình (2). Tuy nhiên, MAE vẫn chỉ cho ta biết rằng "trung bình mỗi căn nhà sai khoảng 126 nghìn USD". Nó không nói rõ

rằng cấu trúc sai số đã thay đổi rất nhiều: thay vì chỉ có một outlier cực lớn như ở mô hình (2), giờ ta có tới hai căn nhà bị dự đoán lệch 300 nghìn USD ở mô hình (3), và vị trí của các outlier (căn nhà thứ tư và thứ năm) cũng mang ý nghĩa rất khác nếu đó là những căn giá cao, rủi ro lớn.

Nói cách khác, MAE gộp cả bốn lỗi nhỏ và một (hoặc hai) lỗi cực lớn vào một con số duy nhất. Nếu chỉ nhìn vào MAE mà không xem phân bố sai số (ví dụ: bảng lỗi, histogram hoặc boxplot residual), ta không biết được có bao nhiêu quan sát gần như chính xác, có bao nhiêu quan sát bị dự đoán "thảm hoạ", và những lỗi lớn đó nằm ở đâu (vùng giá thấp hay vùng giá rất cao).

Chính vì vậy, MAE rất tốt để tóm tắt "mức sai lệch trung bình", nhưng lại không kể được câu chuyện chi tiết về **phân bố sai số**: một mô hình có thể có cùng MAE với mô hình khác nhưng phân bố lỗi hoàn toàn khác nhau (nhiều lỗi vừa phải so với vài lỗi cực lớn). Trong thực hành, ta thường phải kết hợp MAE với các thông tin khác như RMSE, giá trị lỗi lớn nhất, các quantile của sai số hoặc biểu đồ residual để thấy được bức tranh đầy đủ hơn.

Mean Squared Error (MSE) và Root MSE (RMSE): phạt nặng outlier

MSE được tính bằng:

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(p_i - r_i)^2,$$

và RMSE là căn bậc hai:

$$RMSE = \sqrt{MSE}.$$

Tính tay cho hai mô hình:

$$MSE^{(1)} = \frac{10^2 + (-10)^2 + 10^2 + 0^2 + 0^2}{5} = \frac{100 + 100 + 100}{5} = 60, \quad RMSE^{(1)} \approx 7.75.$$

$$MSE^{(2)} = \frac{10^2 + (-10)^2 + 10^2 + 0^2 + 300^2}{5} = \frac{100 + 100 + 100 + 0 + 90000}{5} = 18060, \quad RMSE^{(2)} \approx 134.3.$$

Ở đây ta thấy rõ điểm mạnh của MSE/RMSE: một outlier duy nhất (300) làm RMSE tăng vọt, báo hiệu rằng mô hình mắc một lỗi cực kỳ nghiêm trọng ở đâu đó. Trong nhiều bài toán nhạy cảm với rủi ro (như dự báo tải trọng chịu lực hoặc liều thuốc), việc phạt nặng sai số lớn là điều mong muốn.

Đổi lại, MSE có đơn vị bình phương, khó diễn giải trực tiếp; RMSE diễn giải được nhưng lại rất nhạy cảm với outlier. Trong slide bài giảng, ví dụ về doanh số theo vùng minh hoạ rất rõ việc RMSE có thể bị chi phối gần như hoàn toàn bởi một vài vùng có sai số tuyệt đối quá lớn, trong khi RMSLE lại "dịu" hơn với những outlier này. Ta có thể cụ thể hoá ví dụ đó bằng một bộ số đơn giản như sau.

Giả sử doanh số thực tế của 5 vùng (tính theo đơn vị nghìn đơn vị sản phẩm) là

$$\mathbf{r} = (10,\ 12,\ 15,\ 18,\ 1000),$$

trong đó bốn vùng đầu là thị trường nhỏ đến trung bình, còn vùng thứ năm là một thị trường rất lớn. Mô hình dự đoán cho năm vùng lần lượt là

$$\mathbf{p} = (9,\ 11,\ 16,\ 17,\ 600).$$

Như vậy, bốn vùng đầu được dự đoán khá sát, sai số chỉ khoảng 1 đơn vị, nhưng vùng cuối cùng bị dự đoán thiếu tới 400 đơn vị (từ 1000 xuống 600).

Ta tính sai số tại từng vùng:

$$e_i = p_i - r_i \quad \Rightarrow \quad \mathbf{e} = (-1,\ -1,\ 1,\ -1,\ -400).$$

Sai số tuyệt đối tương ứng là

$$(|e_1|, |e_2|, |e_3|, |e_4|, |e_5|) = (1,\ 1,\ 1,\ 1,\ 400).$$

RMSE bị chi phối gần như hoàn toàn bởi vùng outlier.

Đầu tiên, ta tính RMSE:

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N} e_i^2}.$$

Bình phương sai số từng vùng là

$$(e_1^2, e_2^2, e_3^2, e_4^2, e_5^2) = (1^2, 1^2, 1^2, 1^2, 400^2) = (1,\ 1,\ 1,\ 1,\ 160000).$$

Tổng bình phương sai số là

$$\sum_{i=1}^{5} e_i^2 = 1 + 1 + 1 + 1 + 160000 = 160004,$$

nên

$$RMSE = \sqrt{\frac{160004}{5}} \approx 178{,}9.$$

Nếu nhìn vào cấu trúc này, ta thấy bốn vùng đầu đóng góp tổng cộng chỉ $1+1+1+1 = 4$ vào tổng bình phương sai số, trong khi riêng vùng thứ năm đóng góp tới 160000. Nói cách khác, hơn 99,99% giá trị bên trong dấu căn của RMSE đến từ *một* vùng duy nhất. Dù mô hình làm khá tốt ở bốn vùng nhỏ, RMSE hầu như chỉ phản ánh mức độ "thảm hoạ" ở vùng lớn, gần như bỏ qua thông tin rằng các vùng còn lại được dự đoán ổn.

RMSLE giảm bớt sức nặng của vùng outlier nhờ thang log.

RMSLE được định nghĩa là

$$RMSLE = \sqrt{\frac{1}{N}\sum_{i=1}^{N} \left(\log(r_i + 1) - \log(p_i + 1)\right)^2}.$$

Thay vì làm việc trên thang tuyến tính của doanh số, RMSLE đưa cả giá trị thực và dự đoán lên thang log. Điều này khiến sự khác biệt giữa những con số rất lớn được "nén" lại.

Ta tính sai số trên thang log cho từng vùng:

$$d_i = \log(r_i + 1) - \log(p_i + 1),$$

và bình phương của chúng. Bốn vùng đầu có giá trị nhỏ nên chênh lệch log giữa r_i và p_i cũng rất nhỏ; bình phương của các chênh lệch này vào khoảng 0.0091, 0.0064, 0.0037 và 0.0029. Riêng vùng thứ năm, do từ $r_5 = 1000$ xuống $p_5 = 600$, chênh lệch log lớn hơn, bình phương sai số log vào khoảng 0.26. Tổng bình phương sai số log là xấp xỉ

$$0.0091 + 0.0064 + 0.0037 + 0.0029 + 0.2603 \approx 0.2824,$$

nên

$$RMSLE \approx \sqrt{\frac{0.2824}{5}} \approx 0.238.$$

Ở đây, vùng thứ năm vẫn đóng góp phần lớn (khoảng 92%) tổng bình phương sai số log, nhưng tỷ trọng này đã *nhỏ hơn rất nhiều* so với tỷ trọng trong RMSE, nơi nó chiếm gần như toàn bộ. Đồng thời,

giá trị tuyệt đối của RMSLE (≈ 0.238) nằm trên một thang "vừa phải", phản ánh rằng mô hình có một sai số tương đối đáng kể ở vùng lớn nhưng không đến mức "nổ tung" như RMSE $\approx$ 178,9 trên thang doanh số.

Như vậy, ví dụ doanh số theo vùng cho thấy sự khác biệt rõ ràng trong cách hai metric nhìn vào cùng một lỗi lớn. RMSE, vốn làm việc trên bình phương sai số tuyến tính, để cho vùng outlier chi phối gần như toàn bộ đánh giá, khiến mô hình trông rất tệ về mặt tổng thể. RMSLE, nhờ phép biến đổi log, giảm bớt tác động tuyệt đối của sai số ở vùng có doanh số khổng lồ, để bức tranh không chỉ xoay quanh một vài điểm cực trị. Điều này đặc biệt hữu ích trong các bài toán doanh số, nơi chênh lệch vài trăm đơn vị ở một thị trường lớn đôi khi không "tai hại" như chênh lệch vài chục đơn vị ở một thị trường nhỏ, nếu ta nhìn mọi thứ trên thang tăng trưởng theo tỷ lệ.

MAE vs. RMSE: chọn ai?

Một cách thực tế để hiểu rõ mối quan hệ giữa MAE và RMSE là đặt chúng cạnh nhau và hỏi: *tại sao cùng một bộ sai số mà hai con số này lại khác nhau nhiều như vậy?*

Về mặt toán học, MAE là trung bình của các giá trị tuyệt đối $|e_i|$, trong khi RMSE là căn bậc hai của trung bình các bình phương e_i^2. Khi bình phương, những sai số lớn bị khuếch đại rất mạnh: sai số 10 khi bình phương thành 100, nhưng sai số 100 khi bình phương thành 10 000. Điều này có nghĩa là trong RMSE, một vài lỗi cực lớn có thể thống trị gần như toàn bộ giá trị trung bình, trong khi với MAE, chúng chỉ đóng góp tuyến tính như mọi lỗi khác.

Giả sử ta có hai phân bố sai số khác nhau nhưng cùng số lượng quan sát. Ở phân bố thứ nhất, mọi sai số đều quanh quẩn mức 5–10 đơn vị; ở phân bố thứ hai, đa số sai số chỉ khoảng 1–2 đơn vị nhưng thi thoảng xuất hiện một lỗi cỡ 100. Nếu ta tính MAE cho cả hai trường hợp, kết quả có thể không chênh lệch quá lớn: ở phân bố đầu, MAE có thể quanh 7; ở phân bố thứ hai, MAE có thể nhích lên 10–15 do bị kéo bởi vài lỗi lớn. Nhưng nếu tính RMSE, bức tranh sẽ khác hẳn. Trong phân bố đầu, RMSE sẽ chỉ nhỉnh hơn MAE một chút, vì không có sai số nào đủ lớn để chi phối bình phương. Trong phân bố thứ hai, chỉ cần một vài lỗi cỡ 100 đã đủ làm RMSE bật lên rất cao, có thể lên 40, 50 hoặc hơn, trong khi MAE vẫn ở mức khá "dễ chịu".

Chính vì thế, khi ta so sánh MAE và RMSE trên cùng một tập lỗi, nếu thấy RMSE lớn hơn MAE rất nhiều (chẳng hạn MAE = 5 nhưng RMSE = 20), đó gần như là lời cảnh báo rằng phân phối sai số có đuôi rất dày hoặc đang chứa vài outlier cực lớn. MAE lúc này giống như người kể chuyện hiền lành, mô tả rằng "trung bình sai không quá nhiều", còn RMSE là người nhấn mạnh: "đúng là đa số ổn, nhưng thỉnh thoảng có những cú sai kinh khủng".

Ở góc độ thiết kế hệ thống, ta có thể xem MAE là thước đo tương đối "robust": mỗi đơn vị sai số được đối xử như nhau, nên metric này phản ánh tốt mức sai lệch điển hình mà người dùng thường gặp. Ngược lại, RMSE được thiết kế để nhấn mạnh các lỗi lớn. Nếu bài toán yêu cầu *an toàn trong trường hợp xấu nhất* (ví dụ dự báo tải trọng, liều thuốc, hay rủi ro tài chính), ta thường quan tâm tới việc hạn chế những cú sai lớn và do đó RMSE trở nên rất phù hợp. Ngược lại, nếu ta chỉ cần bảo đảm rằng "phần lớn thời gian mô hình dự đoán vừa phải, thỉnh thoảng lạc một chút cũng được", MAE có thể là thước đo phù hợp hơn, vì nó không để outlier lấn át toàn bộ đánh giá.

Tóm lại, khoảng cách giữa MAE và RMSE không chỉ là chuyện con số, mà là một chỉ báo về hình dáng phân phối sai số. Khi hai giá trị này gần nhau, ta có thể yên tâm rằng sai số được phân bố khá đồng đều. Khi RMSE bỏ xa MAE, ta nên dừng lại nhìn kỹ residual: ở đâu đó đang tồn tại những dự đoán rất tệ mà MAE một mình không kể hết câu chuyện.

4 Sai số phần trăm: MAPE và sMAPE

MAPE: trực quan nhưng nhiều bẫy

Mean Absolute Percentage Error (MAPE) được định nghĩa:

$$MAPE = \frac{100}{N}\sum_{i=1}^{N}\left|\frac{r_i - p_i}{r_i}\right|.$$

Ưu điểm rõ ràng nhất là tính trực quan: ta nói được ngay "mô hình sai trung bình 12%" hay 30%. MAPE không phụ thuộc vào đơn vị đo, nên có thể so sánh giữa các tập dữ liệu khác nhau.

Tuy nhiên, slide bài giảng chỉ ra những vấn đề rất quan trọng: nếu r_i rất nhỏ hoặc bằng 0, $\frac{|r_i - p_i|}{r_i}$ có thể bùng nổ vô hạn. Chẳng hạn, nếu $r_i = 1$ và $p_i = 2$, sai số tuyệt đối chỉ là 1, nhưng sai số phần trăm là 100%. Điều này khiến MAPE cực kỳ nhạy với các quan sát có giá trị gần 0, làm méo toàn bộ metric.

Ví dụ tính tay đơn giản. Giả sử ta dự báo nhu cầu thuốc (đơn vị: viên):

$$\mathbf{r} = (100, 100, 2), \quad \mathbf{p} = (110, 90, 4).$$

Ta có:

$$MAPE = \frac{100}{3}\left(\frac{|100-110|}{100} + \frac{|100-90|}{100} + \frac{|2-4|}{2}\right) = \frac{100}{3}(0.1 + 0.1 + 1) \approx 40\%.$$

Một lỗi tuyệt đối 2 viên (từ 2 lên 4) ở ngày thứ ba, vốn có thể không quá nghiêm trọng về mặt thực tế, lại chiếm tỷ trọng rất lớn trong MAPE vì mẫu số nhỏ. Đây chính là bẫy mà slide minh hoạ khi so sánh nhiều kịch bản dự báo với cùng một sai số tuyệt đối nhưng giá trị thật khác nhau: sai số phần trăm biến thiên rất mạnh theo độ lớn của r_i.

sMAPE: cố gắng "cân bằng" nhưng không hoàn hảo

Symmetric MAPE (sMAPE) được đề xuất để khắc phục phần nào vấn đề của MAPE bằng cách dùng trung bình của r_i và p_i ở mẫu số:

$$sMAPE = \frac{200}{N}\sum_{i=1}^{N}\frac{|r_i - p_i|}{|r_i| + |p_i|}.$$

Nhờ vậy, khi $r_i = 0$ nhưng p_i nhỏ, mẫu số vẫn dương và metric không bùng nổ vô hạn. Đồng thời, việc dùng $r_i + p_i$ ở dưới khiến sMAPE đối xử cân bằng hơn giữa over-forecast và under-forecast.

Tuy nhiên, sMAPE vẫn có những hành vi khó trực giác. Trong slide, hai mô hình khác nhau (một mô hình dự đoán 10 thay vì 1000, một mô hình dự đoán 1000 thay vì 10) có cùng sMAPE khoảng 196%. Trong khi về ý nghĩa thực tế, dự đoán 10 cho 1000 khác rất nhiều so với dự đoán 1000 cho 10, nhưng sMAPE không phân biệt rõ được. Điều này nhắc ta rằng metric nào cũng mang một "góc nhìn" riêng, không có thước đo hoàn hảo cho mọi tình huống. Một ví dụ rất cực đoan nhưng giúp nhìn rõ giới hạn của sMAPE là trường hợp chỉ có *một* quan sát, nhưng mô hình dự đoán "ngược đời". Giả sử giá trị thực là $r = 1000$ và mô hình dự đoán $p = 10$. Khi đó, sai số tuyệt đối là $|r - p| = |1000 - 10| = 990$. Theo định nghĩa,

$$sMAPE = 200 \times \frac{|r-p|}{|r| + |p|} = 200 \times \frac{990}{1000 + 10} = 200 \times \frac{990}{1010}.$$

Ta có $\frac{990}{1010} \approx 0{,}9802$, nên

$$sMAPE \approx 200 \times 0{,}9802 \approx 196{,}04\%.$$

Bây giờ, ta đảo ngược tình huống: giá trị thực là $r = 10$ nhưng mô hình lại dự đoán $p = 1000$. Sai số tuyệt đối vẫn là $|r - p| = |10 - 1000| = 990$. Áp dụng lại công thức sMAPE:

$$sMAPE = 200 \times \frac{|r - p|}{|r| + |p|} = 200 \times \frac{990}{10 + 1000} = 200 \times \frac{990}{1010}.$$

Mẫu số $10 + 1000$ vẫn bằng 1010, tử số vẫn là 990, nên ta thu được đúng cùng một giá trị

$$sMAPE \approx 196{,}04\%.$$

Về mặt toán học, điều này không có gì lạ: trong công thức của sMAPE, hai số r và p chỉ xuất hiện dưới dạng $|r - p|$ ở tử và $|r| + |p|$ ở mẫu. Nếu ta hoán đổi vị trí của chúng (từ $r = 1000, p = 10$ sang $r = 10, p = 1000$), cả tử số lẫn mẫu số đều không thay đổi, nên sMAPE cho cùng một kết quả. sMAPE thực sự "đối xứng" giữa over-forecast và under-forecast theo đúng nghĩa đen.

Tuy nhiên, nếu nhìn từ góc độ thực tế, dự đoán 10 cho một giá trị thật là 1000 và dự đoán 1000 cho một giá trị thật là 10 không hề giống nhau. Trong bối cảnh dự báo nhu cầu, chẳng hạn, trường hợp đầu có thể khiến doanh nghiệp bị thiếu hàng trầm trọng, mất doanh thu lớn vì chỉ chuẩn bị 10 đơn vị thay vì 1000. Trường hợp thứ hai lại dẫn đến tồn kho khổng lồ, chi phí lưu kho, hư hỏng, rủi ro tài chính khác. Mức độ rủi ro, loại rủi ro, và cảm nhận của người sử dụng mô hình ở hai tình huống này có thể rất khác nhau.

Thế nhưng, sMAPE gói cả hai trường hợp này vào cùng một con số xấp xỉ 196%. Nó chỉ nói rằng "mô hình sai gần gấp đôi mức tối đa có thể trên thang đối xứng này", nhưng không cho biết *theo hướng nào* (thừa nhiều hay thiếu nhiều), cũng không phân biệt được bối cảnh nào là nguy hiểm hơn. Từ ví dụ này, ta thấy rõ một giới hạn: sMAPE được thiết kế để đối xử công bằng giữa over-forecast và under-forecast về mặt toán học, nhưng chính sự đối xứng đó khiến nó không phản ánh hết ý nghĩa thực tiễn của từng kiểu sai lầm.

Điều này nhắc ta rằng mỗi metric luôn mang theo một "góc nhìn" cụ thể về sai số. sMAPE coi việc dự đoán quá cao và quá thấp là tương đương nếu xét theo độ chênh lệch tuyệt đối và tổng độ lớn của hai con số. Cách nhìn này phù hợp khi chi phí của việc dự đoán thừa và thiếu gần như giống nhau. Ngược lại, trong những bài toán mà thiếu hàng nguy hiểm hơn tồn kho, hoặc ngược lại, ta cần những metric khác (hoặc bổ sung thêm phân tích chi phí) để phân biệt hai loại lỗi đó. Không có thước đo nào hoàn hảo cho mọi tình huống; điều quan trọng là hiểu rõ "lăng kính" mà mỗi metric đang sử dụng, để không bị con số đẹp trên giấy che mất câu chuyện thực sự của dữ liệu.

5 Sai số tương đối và thước đo tương đối: MRAE, GMRAE, RelMAE, RSE

MRAE: so với một mô hình mốc

Mean Relative Absolute Error (MRAE) không chỉ nhìn vào sai số của mô hình, mà còn so sánh với sai số của một *mô hình mốc* (benchmark), thường là phương pháp đơn giản như naive forecast:

$$MRAE = \frac{1}{N} \sum_{i=1}^{N} \frac{|r_i - p_i|}{|r_i - b_i|},$$

trong đó b_i là dự đoán của mô hình mốc.

Nếu $MRAE < 1$, mô hình đang đánh giá tốt hơn benchmark; nếu $MRAE > 1$, mô hình tệ hơn mốc. Về mặt câu chuyện, MRAE trả lời câu hỏi: "Mô hình của tôi tốt hơn hay tệ hơn so với một cách dự báo cực kỳ đơn giản?"

Nhưng MRAE cũng rất nhạy với mẫu số nhỏ. Trong slide, vấn đề của các chỉ số dạng "tương đối so với benchmark" như MRAE được minh hoạ bằng một tình huống khá tinh tế: khi mô hình mốc b_i dự báo gần như hoàn hảo so với giá trị thực r_i, sai số $|r_i - b_i|$ ở mẫu số trở nên rất nhỏ, khiến tỉ lệ

$$\frac{|r_i - p_i|}{|r_i - b_i|}$$

dễ dàng "nổ tung" chỉ vì mô hình của ta không tốt bằng một dự báo gần như hoàn hảo. Chỉ số khi đó bị phóng đại, trông vô cùng tệ, dù trên thực tế sai số của mô hình không quá khủng khiếp nếu ta nhìn ở thang đo gốc.

Ta có thể cụ thể hoá bằng một ví dụ đơn giản. Giả sử giá trị thực tại một thời điểm là

$$r_i = 100.$$

Mô hình mốc (benchmark) sử dụng một cách dự báo rất thông minh, ví dụ như trung bình trượt có trọng số, và vô tình cho kết quả gần như chính xác:

$$b_i = 99.$$

Sai số của benchmark khi đó là

$$|r_i - b_i| = |100 - 99| = 1.$$

Mô hình của chúng ta, trong khi đó, dự đoán

$$p_i = 95.$$

Sai số tuyệt đối của mô hình là

$$|r_i - p_i| = |100 - 95| = 5.$$

Nếu ta nhìn trên thang đo gốc, việc sai 5 đơn vị quanh mốc 100 không phải là một thảm hoạ; MAE ở mức 5 vẫn có thể được coi là chấp nhận được trong nhiều bối cảnh. Nhưng khi ta tính MRAE tại điểm này,

$$\text{tỉ lệ tương đối} = \frac{|r_i - p_i|}{|r_i - b_i|} = \frac{5}{1} = 5,$$

nghĩa là chỉ số riêng ở điểm này nói rằng mô hình của ta tệ gấp 5 lần benchmark. Nếu tập dữ liệu chỉ có vài quan sát như vậy, giá trị MRAE trung bình sẽ bị kéo lên rất cao, tạo cảm giác mô hình là "thảm hoạ" so với mốc, dù trên thực tế nó chỉ đơn giản là không tốt bằng một dự báo gần như hoàn hảo.

Vấn đề trở nên nghiêm trọng hơn khi sai số của benchmark còn nhỏ hơn nữa. Hãy tưởng tượng benchmark đoán đúng đến mức

$$b_i = 99{,}9,$$

khi đó

$$|r_i - b_i| = |100 - 99{,}9| = 0{,}1.$$

Nếu mô hình của ta vẫn dự đoán $p_i = 95$, sai số tuyệt đối vẫn là 5. Nhưng tỉ lệ tương đối giờ là

$$\frac{|r_i - p_i|}{|r_i - b_i|} = \frac{5}{0{,}1} = 50.$$

Chỉ vì benchmark may mắn (hoặc rất tinh vi) đoán gần như chính xác, mẫu số trong tỉ lệ trở nên cực nhỏ. Kết quả là MRAE tại điểm này vọt lên 50, gợi ý rằng mô hình của ta tệ gấp 50 lần benchmark. Nhìn vào con số 50, nhiều người sẽ nghĩ mô hình gần như vô dụng, trong khi trên thực tế nó chỉ sai 5 đơn vị quanh mức 100.

Hiện tượng này cho thấy một điểm yếu mang tính cấu trúc của các metric dạng

$$\frac{|r_i - p_i|}{|r_i - b_i|} :$$

chúng cực kỳ nhạy với những tình huống mà benchmark làm *quá tốt*. Khi $|r_i - b_i|$ tiến gần 0, bất cứ sai số nào của mô hình cũng bị "phóng đại" lên tỉ lệ rất lớn, dẫn đến một chỉ số tổng hợp bị lệch hẳn về những điểm này. Nếu trong cả chuỗi thời gian, benchmark có thêm vài lần đoán gần như hoàn hảo, MRAE trung bình có thể bị chi phối bởi những quan sát hiếm hoi đó, dù phần lớn thời gian mô hình của ta hoạt động ở mức chấp nhận được.

Điều này không có nghĩa là MRAE vô dụng. Khi benchmark là một mô hình đơn giản (như naive forecast) và hiếm khi chính xác tuyệt đối, MRAE là một thước đo hữu ích để trả lời câu hỏi "mô hình của tôi tốt hơn baseline bao nhiêu lần". Tuy nhiên, ví dụ trên nhắc chúng ta rằng nếu benchmark đôi khi rất gần với dữ liệu thật, đặc biệt trong những khoảng thời gian ít nhiễu, các metric chia cho $|r_i - b_i|$ cần được diễn giải thận trọng. Thay vì nhìn vào MRAE một cách mù quáng, ta nên kết hợp với các chỉ số khác như MAE, RMSE, hoặc thậm chí xem trực tiếp phân bố $|r_i - b_i|$ để hiểu bối cảnh: benchmark có thực sự là một "đối thủ công bằng" hay chỉ tình cờ cực kỳ may mắn ở một vài điểm dữ liệu.

GMRAE: lấy trung bình hình học để giảm ảnh hưởng outlier

Geometric Mean Relative Absolute Error (GMRAE) dùng trung bình hình học của các tỉ lệ thay vì trung bình cộng:

$$GMRAE = \left(\prod_{i=1}^{N} \frac{|r_i - p_i|}{|r_i - b_i|} \right)^{1/N} .$$

Việc dùng tích và căn bậc N làm giảm ảnh hưởng của một vài tỉ lệ quá lớn hoặc quá nhỏ, nên GMRAE thường ổn định hơn MRAE với outlier. Tuy nhiên, như slide minh hoạ, trung bình hình học lại nhạy với các giá trị rất nhỏ và có thể tạo cảm giác "cân bằng" hơn thực tế: một lỗi cực nhỏ tại một thời điểm nào đó có thể kéo GMRAE xuống, làm ta nghĩ rằng mô hình ổn định, trong khi thực tế tồn tại vài lỗi rất lớn ở các thời điểm khác. Ta có thể minh hoạ vấn đề này bằng một ví dụ số cụ thể với bốn thời điểm dự báo. Tại mỗi thời điểm i, ta xét tỉ lệ

$$q_i = \frac{|r_i - p_i|}{|r_i - b_i|}$$

giữa sai số tuyệt đối của mô hình và sai số tuyệt đối của benchmark. Giả sử trong bốn thời điểm, ta thu được các giá trị:

$$q_1 = 0{,}01, \qquad q_2 = 5, \qquad q_3 = 10, \qquad q_4 = 10.$$

Diễn giải một cách đơn giản, điều này có nghĩa là:

- Ở thời điểm thứ nhất, mô hình tốt hơn benchmark rất nhiều (sai số chỉ bằng 1% sai số của benchmark).
- Ở thời điểm thứ hai, mô hình tệ hơn benchmark 5 lần.
- Ở hai thời điểm cuối, mô hình tệ hơn benchmark tới 10 lần.

Nếu ta dùng **MRAE** (trung bình cộng), giá trị thu được là

$$MRAE = \frac{q_1 + q_2 + q_3 + q_4}{4} = \frac{0{,}01 + 5 + 10 + 10}{4} = \frac{25{,}01}{4} \approx 6{,}25.$$

Con số 6,25 phản ánh khá rõ ràng rằng, trung bình, mô hình tệ hơn benchmark nhiều lần; nó bị kéo mạnh bởi hai lỗi rất lớn (gấp 10 lần) và một lỗi khá lớn (gấp 5 lần).

Tuy nhiên, nếu ta dùng **GMRAE** (trung bình hình học), ta lại có:

$$GMRAE = (q_1 \cdot q_2 \cdot q_3 \cdot q_4)^{1/4} = (0{,}01 \times 5 \times 10 \times 10)^{1/4}.$$

Tích trong ngoặc là

$$0{,}01 \times 5 \times 10 \times 10 = 0{,}01 \times 500 = 5,$$

nên

$$GMRAE = 5^{1/4} \approx 1{,}50.$$

Ở đây ta thấy hiệu ứng rất rõ: chỉ vì có một thời điểm đầu tiên mà mô hình tốt vượt trội (tỉ lệ 0,01), tích $q_1q_2q_3q_4$ bị kéo xuống khá mạnh, khiến căn bậc bốn của tích chỉ còn khoảng 1,5. Nhìn vào GMRAE $\approx$ 1,5, ta dễ có cảm giác rằng mô hình "chỉ tệ hơn benchmark một chút" và tương đối *cân bằng*. Thực tế, trong ba thời điểm còn lại, mô hình tệ hơn benchmark 5–10 lần, một mức chênh lệch rất đáng lo ngại.

Ví dụ này cho thấy trung bình hình học nhạy với các giá trị rất nhỏ: chỉ cần một điểm dự đoán gần như hoàn hảo (so với benchmark) là đủ kéo GMRAE xuống đáng kể, che bớt ấn tượng về những lỗi rất lớn ở các thời điểm khác. Nếu ta chỉ nhìn vào GMRAE, có thể đi đến kết luận lạc quan rằng mô hình "ổn định quanh mức 1,5 lần tệ hơn benchmark", trong khi phân bố thực sự của tỷ lệ q_i lại rất mất cân đối: một điểm cực tốt và nhiều điểm cực tệ.

Điều này không có nghĩa GMRAE là một metric "xấu", mà chỉ nhắc ta rằng mỗi chỉ số đều mang một góc nhìn riêng. GMRAE ưu ái các mô hình có vài điểm cực tốt (tỉ lệ rất nhỏ), và "giảm nhẹ" ảnh hưởng của các điểm cực xấu. Nếu mục tiêu của ta là đánh giá mức độ tệ nhất hoặc mức độ tệ trung bình theo nghĩa tuyến tính, MRAE hoặc các thống kê khác (như quantile của q_i) sẽ phù hợp hơn; còn nếu ta quan tâm đến hiệu suất tương đối "điển hình" trên thang nhân, GMRAE lại là một công cụ hữu ích.

RelMAE: so sánh MAE với benchmark

Một cách nhìn khác là Relative Mean Absolute Error (RelMAE):

$$RelMAE = \frac{MAE_{\text{model}}}{MAE_{\text{benchmark}}} = \frac{\sum |r_i - p_i|}{\sum |r_i - b_i|}.$$

Nếu $RelMAE < 1$, mô hình tốt hơn benchmark; nếu $RelMAE > 1$, mô hình tệ hơn. Metric này đơn giản, dễ hiểu và không phụ thuộc vào đơn vị. Tuy nhiên, ví dụ trong slide cho thấy RelMAE cũng có thể che mờ lỗi cực lớn: một lỗi 100 ở chỉ một thời điểm có thể bị "dìm" xuống bởi nhiều lỗi nhỏ ở các thời điểm khác, khiến RelMAE chỉ nhỉnh hơn 1 một chút và không phản ánh đúng mức độ nghiêm trọng của outlier. Một cách khác để nhìn thấy hạn chế của các chỉ số tương đối so với benchmark là thông qua **RelMAE**. Nhắc lại, RelMAE được định nghĩa là tỷ số giữa MAE của mô hình và MAE của benchmark:

$$RelMAE = \frac{MAE_{\text{model}}}{MAE_{\text{benchmark}}} = \frac{\sum_{i=1}^{N} |r_i - p_i|}{\sum_{i=1}^{N} |r_i - b_i|}.$$

Nếu RelMAE nhỏ hơn 1, ta kết luận mô hình tốt hơn benchmark; nếu lớn hơn 1, mô hình kém hơn. Về mặt trực giác, đây là một thước đo rất tiện: chỉ cần một con số để biết mình "hơn hay kém baseline bao nhiêu lần". Tuy nhiên, ví dụ trong slide cho thấy RelMAE cũng có thể *che mờ* những lỗi cực lớn, vì nó chỉ nhìn tổng sai số tuyệt đối, không quan tâm sai số đó phân bố như thế nào theo thời gian.

Hãy xét một ví dụ cụ thể với 6 thời điểm dự báo. Giá trị thực r_i có thể là bất kỳ, ở đây ta chỉ tập trung vào độ lớn sai số. Giả sử benchmark có sai số tuyệt đối tại 6 thời điểm là:

$$|r_i - b_i| : \quad 10,\ 10,\ 10,\ 10,\ 10,\ 10.$$

Benchmark ổn định, luôn sai 10 đơn vị ở mỗi thời điểm, nên

$$MAE_{\text{benchmark}} = \frac{10+10+10+10+10+10}{6} = 10.$$

Mô hình của ta có sai số như sau:

$$|r_i - p_i| : \quad 8,\ 8,\ 8,\ 8,\ 8,\ 100.$$

Ở 5 thời điểm đầu, mô hình tốt hơn benchmark (chỉ sai 8 thay vì 10). Nhưng ở thời điểm cuối, nó mắc một lỗi cực lớn: sai 100 đơn vị. MAE của mô hình là:

$$\sum_{i=1}^{6} |r_i - p_i| = 8+8+8+8+8+100 = 140,$$

nên

$$MAE_{\text{model}} = \frac{140}{6} \approx 23{,}33.$$

Từ đây, RelMAE bằng:

$$RelMAE = \frac{23{,}33}{10} \approx 2{,}33.$$

Con số 2,33 nói rằng *trung bình* mô hình tệ hơn benchmark khoảng 2,3 lần về sai số tuyệt đối. Đó đã là một cảnh báo khá rõ. Tuy nhiên, hãy tinh chỉnh ví dụ một chút để thấy rõ hơn việc outlier có thể bị "dìm".

Giả sử benchmark lại tệ hơn một chút, với sai số:

$$|r_i - b_i| : \quad 20,\ 20,\ 20,\ 20,\ 20,\ 20,$$

nên

$$MAE_{\text{benchmark}} = 20.$$

Mô hình vẫn giữ sai số

$$|r_i - p_i| : \quad 8,\ 8,\ 8,\ 8,\ 8,\ 100,$$

tức là ở 5 thời điểm đầu nó vượt trội benchmark (8 so với 20), chỉ có một thời điểm cuối cùng là rất tệ. Khi đó:

$$\sum_{i=1}^{6} |r_i - p_i| = 140, \qquad MAE_{\text{model}} \approx 23{,}33.$$

Và

$$\sum_{i=1}^{6} |r_i - b_i| = 20 \times 6 = 120, \qquad MAE_{\text{benchmark}} = 20.$$

Tỷ số RelMAE trở thành:

$$RelMAE = \frac{23{,}33}{20} \approx 1{,}17.$$

Nhìn vào RelMAE $\approx$ 1,17, ta dễ diễn giải rằng mô hình "chỉ tệ hơn benchmark khoảng 17%" về mặt MAE. Điều này nghe có vẻ không quá kinh khủng, đặc biệt nếu ta chỉ nhìn con số 1,17 mà không

xem chi tiết từng thời điểm. Nhưng phân tích kỹ hơn, ta thấy bức tranh hoàn toàn khác: trong 5/6 thời điểm, mô hình rất tốt, sai số chỉ bằng 40% sai số của benchmark (8 so với 20), nhưng đổi lại, ở 1 thời điểm, mô hình mắc lỗi *cực lớn* (100 đơn vị). Nếu bối cảnh ứng dụng là một hệ thống nhạy cảm, ví dụ dự báo liều thuốc, dự trữ hàng hoá giá trị cao, hay cảnh báo rủi ro, thì chỉ một lỗi 100 duy nhất này cũng có thể gây hậu quả nghiêm trọng.

RelMAE, vì chỉ quan tâm tới *tổng* sai số tuyệt đối, đã để cho 5 lỗi nhỏ "dìm" bớt ảnh hưởng của một lỗi cực lớn. Trung bình cộng 5 lần rất tốt và 1 lần rất tệ cho ra một con số chỉ hơi cao hơn baseline; trong khi về mặt rủi ro, có thể ta sẵn sàng chấp nhận một mô hình luôn sai 20 nhưng ổn định, hơn là một mô hình thường sai 8 nhưng thỉnh thoảng "nổ" 100.

Ví dụ này cho thấy RelMAE có xu hướng trơn hoá mọi thứ về một hệ số tương đối duy nhất, nên dễ che khuất các outlier. Nếu chỉ nhìn vào RelMAE mà không kiểm tra phân bố sai số theo thời gian (hoặc không kiểm tra thêm các thống kê như lỗi lớn nhất, quantile của lỗi), ta có thể đánh giá thấp mức độ nghiêm trọng của một vài sai số cực lớn. Cũng giống như các metric khác, RelMAE rất hữu ích để so sánh tổng thể mô hình với benchmark, nhưng không nên được sử dụng một mình khi bài toán nhạy cảm với lỗi cực trị. Trong những tình huống như vậy, cần kết hợp thêm các thước đo khác hoặc phân tích tập trung vào các trường hợp xấu nhất để có cái nhìn đầy đủ hơn.

Relative Squared Error (RSE): liên hệ với R^2

Relative Squared Error (RSE) so sánh tổng bình phương sai số của mô hình với tổng bình phương sai số của một baseline đơn giản là dự đoán bằng trung bình:

$$RSE = \frac{\sum_{i=1}^{N}(r_i - p_i)^2}{\sum_{i=1}^{N}(r_i - \bar{r})^2}.$$

RSE gần 0 nghĩa là mô hình rất tốt (sai số bình phương nhỏ), RSE gần 1 nghĩa là không tốt hơn nhiều so với việc đoán bằng trung bình, và RSE lớn hơn 1 nghĩa là mô hình tệ hơn baseline.

Trong slide, hai case study với và không có outlier cho thấy một hiện tượng thú vị: khi thêm một outlier rất lớn vào cả mô hình và benchmark, tổng sai số bình phương của cả hai đều tăng mạnh, nhưng theo cách nào đó khiến tỉ lệ RSE tiến gần 1. Kết quả là, RSE giảm từ 2.22 xuống còn khoảng 0.99 khi thêm outlier, tạo cảm giác mô hình "tiến bộ", trong khi thực tế cả hai (mô hình và benchmark) đều trở nên tệ hơn nhiều. Đây là ví dụ rõ ràng về việc một metric tương đối có thể đánh lừa ta nếu cả mô hình và baseline đều bị kéo bởi outlier theo hướng giống nhau. Hiện tượng thú vị của RSE (Relative Squared Error) được thấy rất rõ khi ta so sánh hai case study: một bộ dữ liệu không có outlier và một bộ dữ liệu được thêm cùng một outlier rất lớn vào cả mô hình lẫn baseline. Nhớ lại định nghĩa:

$$RSE = \frac{\sum_{i=1}^{N}(r_i - p_i)^2}{\sum_{i=1}^{N}(r_i - \bar{r})^2},$$

trong đó tử số là tổng sai số bình phương của mô hình (RSS), còn mẫu số chính là tổng bình phương sai số của *baseline dự đoán bằng trung bình* (TSS). Nói cách khác, RSE so sánh mô hình với một baseline "ngây thơ" là luôn đoán $\bar{r}$.

Case 1: không có outlier, mô hình tệ hơn nhiều so với baseline.

Xét bộ dữ liệu đầu tiên với bốn quan sát:

$$\mathbf{r}^{(1)} = (10,\ 12,\ 15,\ 18), \quad \mathbf{p}^{(1)} = (14,\ 16,\ 20,\ 24).$$

Trung bình của giá trị thực là

$$\bar{r}^{(1)} = \frac{10 + 12 + 15 + 18}{4} = 13{,}75.$$

Tổng bình phương sai số so với trung bình (TSS – tức chất lượng baseline "đoán bằng $\bar{r}$") là:

$$TSS^{(1)} = \sum_{i=1}^{4}(r_i^{(1)} - \bar{r}^{(1)})^2 = (10-13{,}75)^2 + (12-13{,}75)^2 + (15-13{,}75)^2 + (18-13{,}75)^2 = 36{,}75.$$

Sai số của mô hình tại từng điểm là

$$e_i^{(1)} = p_i^{(1)} - r_i^{(1)} = (4,\ 4,\ 5,\ 6),$$

nên tổng sai số bình phương (RSS) là

$$RSS^{(1)} = 4^2 + 4^2 + 5^2 + 6^2 = 16 + 16 + 25 + 36 = 93.$$

Từ đó,

$$RSE^{(1)} = \frac{RSS^{(1)}}{TSS^{(1)}} = \frac{93}{36{,}75} \approx 2{,}53.$$

Giá trị $RSE^{(1)} > 1$ cho thấy mô hình đang tệ hơn rất nhiều so với baseline "chỉ đoán mỗi $\bar{r}$". Trong case này, trực giác và con số đều khớp: baseline đoán cũng không đến nỗi tệ, trong khi mô hình hầu như luôn lệch ra xa khỏi r_i.

Case 2: thêm một outlier lớn cho cả dữ liệu và mô hình.

Bây giờ, ta thêm một quan sát thứ năm là một outlier rất lớn. Giá trị thực mới:

$$\mathbf{r}^{(2)} = (10,\ 12,\ 15,\ 18,\ 500),$$

và mô hình dự đoán:

$$\mathbf{p}^{(2)} = (14,\ 16,\ 20,\ 24,\ 50).$$

Hãy chú ý: ở bốn điểm đầu, mô hình vẫn tệ như cũ (sai 4, 4, 5, 6), và ở điểm thứ năm mô hình cũng *sai rất lớn* (500 so với 50 là chênh 450 đơn vị). Tức là cả baseline lẫn mô hình đều sẽ bị outlier này kéo cho "tệ đi" rất nhiều về mặt sai số tuyệt đối.

Trước hết, tính lại trung bình mới:

$$\bar{r}^{(2)} = \frac{10 + 12 + 15 + 18 + 500}{5} = \frac{555}{5} = 111.$$

Tổng bình phương sai số so với trung bình (TSS) lúc này là:

$$\begin{aligned} TSS^{(2)} &= (10-111)^2 + (12-111)^2 + (15-111)^2 + (18-111)^2 + (500-111)^2 \\ &= 101^2 + 99^2 + 96^2 + 93^2 + 389^2 \\ &= 10\,201 + 9\,801 + 9\,216 + 8\,649 + 151\,321 \\ &= 189\,188. \end{aligned}$$

Baseline "đoán bằng $\bar{r}^{(2)} = 111$" giờ đây cũng mắc những lỗi khổng lồ: ở cả 4 điểm nhỏ đầu, nó đoán cao hơn nhiều (từ 10–18 mà đoán 111), còn ở điểm outlier thứ năm, nó đoán thấp hơn khá nhiều (từ 500 về 111). Kết quả là TSS tăng từ 36,75 lên 189 188, tức baseline trở nên tệ hơn rất nhiều.

Sai số của mô hình mới là:

$$e_i^{(2)} = p_i^{(2)} - r_i^{(2)} = (4,\ 4,\ 5,\ 6,\ -450),$$

nên tổng bình phương sai số:

$$\begin{aligned} RSS^{(2)} &= 4^2 + 4^2 + 5^2 + 6^2 + (-450)^2 \\ &= 16 + 16 + 25 + 36 + 202\,500 \\ &= 202\,593. \end{aligned}$$

Rõ ràng, mô hình cũng tệ đi rất nhiều về mặt tuyệt đối: RSS nhảy từ 93 lên hơn 202 nghìn.

Tuy nhiên, khi ta tính RSE:

$$RSE^{(2)} = \frac{RSS^{(2)}}{TSS^{(2)}} = \frac{202\,593}{189\,188} \approx 1{,}07.$$

Hiệu ứng "tiến bộ ảo" của RSE.

Nếu đặt hai case cạnh nhau:

$$RSE^{(1)} \approx 2{,}53 \quad \text{(không có outlier)}, \qquad RSE^{(2)} \approx 1{,}07 \quad \text{(có outlier)}.$$

Ta thấy một điều rất nghịch lý: sau khi thêm một outlier khổng lồ, khiến *cả* baseline lẫn mô hình đều tệ hơn nhiều (TSS tăng từ 36,75 lên 189 188, RSS tăng từ 93 lên 202 593), thì RSE lại *giảm mạnh* từ khoảng 2,53 xuống còn 1,07, tiến rất gần tới 1. Nếu chỉ nhìn vào RSE, ta có thể kết luận rằng "mô hình đã tiến bộ rõ rệt so với baseline", vì từ chỗ tệ hơn baseline hơn hai lần, giờ chỉ tệ hơn khoảng 7%.

Thực tế, điều xảy ra là outlier kéo cả hai về phía xấu đi, nhưng baseline (dự đoán bằng trung bình) bị outlier đó phạt nặng hơn mô hình. Baseline phải "cân bằng" toàn bộ dữ liệu, nên khi xuất hiện một giá trị 500 bên cạnh các giá trị 10–18, trung bình bị đẩy lên một mức mà làm cho cả các điểm nhỏ lẫn điểm lớn đều bị lỗi lớn. Mô hình, trong ví dụ này, vô tình (hoặc cố ý) dự đoán 50 cho điểm 500: sai rất nặng, nhưng vẫn "ít tệ" hơn so với baseline dự đoán 111. Tỉ số RSS/TSS vì thế giảm xuống, dù cả hai sai số tuyệt đối đều bùng nổ.

Đây chính là hiệu ứng "tiến bộ ảo" của RSE (và nói chung là mọi metric tương đối so với một baseline cụ thể): nếu cả mô hình lẫn baseline đều bị kéo bởi outlier theo hướng giống nhau, nhưng baseline chịu tác động nặng hơn một chút, thì RSE sẽ giảm, tạo cảm giác mô hình *tốt hơn tương đối*, dù trên thang đo tuyệt đối, mọi thứ đều trở nên tệ hơn rất nhiều. Nếu ta chỉ nhìn vào RSE mà không kiểm tra TSS, RSS, hoặc không xem trực tiếp các outlier trong residual, rất dễ đi đến kết luận sai về chất lượng thực sự của mô hình.

Từ ví dụ này, bài học là: các metric tương đối so với baseline, như RSE, rất hữu ích để xem mô hình có vượt qua một mốc đơn giản hay không, nhưng chúng không thể thay thế cho việc phân tích trực tiếp sai số tuyệt đối và cấu trúc outlier. Khi dữ liệu có khả năng xuất hiện các giá trị cực đoan, ta cần đặc biệt thận trọng khi diễn giải những chỉ số kiểu "tỉ số" như vậy.

6 Sai số chuẩn hoá theo thang: MASE, RMSSE

Một vấn đề lớn với MAE, MSE, RMSE là chúng phụ thuộc vào thang đo của dữ liệu: cùng một mô hình có thể trông "tệ" nếu đơn vị là nghìn, nhưng lại "ổn" nếu đơn vị là triệu. Để giải quyết, ta dùng các metric được chuẩn hoá bởi một đại lượng đặc trưng (thường là MAE in-sample của một naive baseline).

MASE: Mean Absolute Scaled Error

Mean Absolute Scaled Error (MASE) được định nghĩa:

$$MASE = \frac{\frac{1}{N}\sum_{i=1}^{N} |r_i - p_i|}{\frac{1}{N-1}\sum_{t=2}^{N} |r_t - r_{t-1}|}.$$

Tử số là MAE của mô hình; mẫu số là MAE của mô hình naive dùng giá trị trước đó làm dự báo. Nếu $MASE < 1$, mô hình tốt hơn naive; $MASE > 1$ nghĩa là dự báo còn tệ hơn cả việc chỉ lấy giá trị trước đó.

Ví dụ trong slide sử dụng cùng một tập dữ liệu gồm 5 quan sát liên tiếp để minh hoạ cách tính MASE một cách cụ thể. Ý tưởng của MASE là *chuẩn hoá* MAE của mô hình bằng một mốc tham chiếu rất đơn giản: sai số tuyệt đối trung bình của mô hình **naive** dự báo giá trị hôm nay bằng giá trị hôm qua. Nhờ vậy, ta không chỉ biết mô hình sai bao nhiêu, mà còn biết nó tốt hơn hay tệ hơn so với một chiến lược rất ngây thơ.

Giả sử ta có chuỗi thời gian nhu cầu (hoặc doanh số) theo ngày:

$$r_1 = 100, \quad r_2 = 110, \quad r_3 = 105, \quad r_4 = 120, \quad r_5 = 115.$$

Bước đầu tiên để tính MASE là tính **mẫu số**, tức là MAE của mô hình naive dùng giá trị hôm trước để dự báo hôm sau. Khi đó, dự báo naive tại các thời điểm từ 2 đến 5 là:

$$\hat{r}_2^{(\text{naive})} = r_1 = 100, \quad \hat{r}_3^{(\text{naive})} = r_2 = 110, \quad \hat{r}_4^{(\text{naive})} = r_3 = 105, \quad \hat{r}_5^{(\text{naive})} = r_4 = 120.$$

Sai số tuyệt đối của mô hình naive tại từng điểm là:

$$|r_2 - \hat{r}_2^{(\text{naive})}| = |110 - 100| = 10,$$

$$|r_3 - \hat{r}_3^{(\text{naive})}| = |105 - 110| = 5,$$

$$|r_4 - \hat{r}_4^{(\text{naive})}| = |120 - 105| = 15,$$

$$|r_5 - \hat{r}_5^{(\text{naive})}| = |115 - 120| = 5.$$

Tổng sai số tuyệt đối của naive là $10 + 5 + 15 + 5 = 35$. Vì chỉ có 4 dự báo (từ thời điểm 2 đến 5), MAE của naive bằng:

$$\text{MAE}_{\text{naive}} = \frac{|r_2 - r_1| + |r_3 - r_2| + |r_4 - r_3| + |r_5 - r_4|}{4} = \frac{35}{4} = 8{,}75.$$

Biểu thức trong slide

$$\frac{|r_2 - r_1| + |r_3 - r_2| + |r_4 - r_3| + |r_5 - r_4|}{4}$$

chính là cách viết cô đọng của $\text{MAE}_{\text{naive}}$ với dự báo kiểu "hôm nay bằng hôm qua".

Bây giờ, giả sử ta có hai mô hình dự báo khác nhau cho cùng chuỗi này.

Mô hình A dự báo:

$$p_1^{(A)} = 100, \quad p_2^{(A)} = 108, \quad p_3^{(A)} = 107, \quad p_4^{(A)} = 118, \quad p_5^{(A)} = 117.$$

Sai số tuyệt đối của mô hình A là:

$$|r_1 - p_1^{(A)}| = |100 - 100| = 0,$$

$$|r_2 - p_2^{(A)}| = |110 - 108| = 2,$$

$$|r_3 - p_3^{(A)}| = |105 - 107| = 2,$$

$$|r_4 - p_4^{(A)}| = |120 - 118| = 2,$$

$$|r_5 - p_5^{(A)}| = |115 - 117| = 2.$$

Tổng sai số tuyệt đối là $0+2+2+2+2=8$, nên

$$\text{MAE}_A = \frac{8}{5} = 1{,}6.$$

MASE của mô hình A sẽ là MAE của A chia cho MAE của naive:

$$MASE_A = \frac{\text{MAE}_A}{\text{MAE}_{\text{naive}}} = \frac{1{,}6}{8{,}75} \approx 0{,}18.$$

Giá trị $MASE_A \approx 0{,}18$ nhỏ hơn 1 rất nhiều, có thể diễn giải là: mô hình A chỉ mắc khoảng 18% mức sai số trung bình của một chiến lược naive, tức là tốt hơn naive khoảng $\frac{1}{0{,}18} \approx 5{,}5$ lần.

Mô hình B lại dự đoán kém hơn. Giả sử:

$$p_1^{(B)} = 105, \quad p_2^{(B)} = 115, \quad p_3^{(B)} = 110, \quad p_4^{(B)} = 125, \quad p_5^{(B)} = 120.$$

Sai số tuyệt đối:

$$|r_1 - p_1^{(B)}| = |100 - 105| = 5,$$
$$|r_2 - p_2^{(B)}| = |110 - 115| = 5,$$
$$|r_3 - p_3^{(B)}| = |105 - 110| = 5,$$
$$|r_4 - p_4^{(B)}| = |120 - 125| = 5,$$
$$|r_5 - p_5^{(B)}| = |115 - 120| = 5.$$

Tổng sai số tuyệt đối là $5+5+5+5+5=25$, nên

$$\text{MAE}_B = \frac{25}{5} = 5.$$

MASE của mô hình B là:

$$MASE_B = \frac{\text{MAE}_B}{\text{MAE}_{\text{naive}}} = \frac{5}{8{,}75} \approx 0{,}57.$$

Mô hình B vẫn tốt hơn naive (vì $MASE_B < 1$), nhưng kém mô hình A (vì $0{,}57 > 0{,}18$). Thay vì nói rời rạc rằng "MAE của A là 1,6 còn của B là 5", MASE cho ta một thước *chuẩn hoá*: A chỉ mắc 18% sai số của naive, còn B mắc 57% sai số của naive.

Điểm mạnh của MASE là mẫu số $\frac{|r_2-r_1|+\cdots+|r_5-r_4|}{4}$ phản ánh mức độ biến động nội tại của chính chuỗi đó. Nếu chuỗi có biến động cao (các chênh lệch $|r_t - r_{t-1}|$ lớn), chiến lược naive cũng sẽ mắc sai số lớn; nếu chuỗi rất ổn định, chiến lược naive sẽ rất chính xác. Khi ta chuẩn hoá MAE của mô hình theo mức sai số "tự nhiên" này, ta có thể so sánh MASE của nhiều chuỗi khác nhau trên cùng một thang: một mô hình có $MASE = 0{,}5$ nghĩa là nó sai bằng một nửa mô hình naive, bất kể chuỗi đang ở thang đo 10, 100 hay 10 000.

Nhờ vậy, MASE trở thành một "chuẩn chung" rất hữu ích: nó trả lời câu hỏi *"mô hình của tôi tốt hơn dự báo đơn giản bao nhiêu lần"* theo một cách không phụ thuộc vào đơn vị đo lường, cho phép so sánh hiệu quả mô hình giữa các chuỗi thời gian có thang đo và độ biến động khác nhau.

RMSSE: Root Mean Squared Scaled Error

RMSSE là phiên bản "bình phương" của MASE:

$$RMSSE = \sqrt{\frac{\frac{1}{N}\sum_{i=1}^{N}(r_i - p_i)^2}{\frac{1}{N-1}\sum_{t=2}^{N}(r_t - r_{t-1})^2}}.$$

RMSSE thừa hưởng tính nhạy với outlier của RMSE, nhưng sử dụng mẫu số để chuẩn hoá, giúp so sánh giữa các chuỗi có thang đo khác nhau. Hai metric MASE và RMSSE đặc biệt phổ biến trong bối cảnh đánh giá mô hình dự báo chuỗi thời gian trên nhiều sản phẩm/danh mục với mức doanh số rất khác nhau. Để thấy rõ kết luận này, ta xây một ví dụ cụ thể với **hai sản phẩm** có thang đo doanh số rất khác nhau, nhưng cùng áp dụng RMSSE để so sánh.

Bước 1: Định nghĩa RMSSE

RMSSE (Root Mean Squared Scaled Error) là phiên bản "chuẩn hoá" của RMSE, thường được định nghĩa (trong bối cảnh chuỗi thời gian) là:

$$RMSSE = \sqrt{\frac{\frac{1}{h}\sum_{t=T+1}^{T+h}(r_t - p_t)^2}{\frac{1}{T-1}\sum_{t=2}^{T}(r_t - r_{t-1})^2}},$$

trong đó:

- r_t là giá trị thật trong giai đoạn forecast (từ $T+1$ đến $T+h$),
- p_t là dự báo của mô hình,
- mẫu số là *MSE của mô hình naive* "hôm nay bằng hôm qua" tính trên giai đoạn lịch sử (từ 1 đến T).

Nếu bỏ phần căn bậc hai, ta thu được *MSSE* (Mean Squared Scaled Error); RMSSE đơn giản là $\sqrt{MSSE}$.

Bước 2: Hai sản phẩm có thang đo rất khác nhau

Giả sử ta có hai sản phẩm:

- Sản phẩm A: một mặt hàng bán lẻ với doanh số nhỏ (vài chục đơn vị/ngày).
- Sản phẩm B: một mặt hàng bán buôn với doanh số rất lớn (vài nghìn đơn vị/ngày).

Sản phẩm A (thang nhỏ). Giả sử dữ liệu lịch sử 5 ngày của sản phẩm A là:

$$r_{\text{hist}}^{(A)} = (10,\ 12,\ 11,\ 13,\ 12).$$

Ta có 4 chênh lệch liên tiếp:

$$|12-10| = 2, \quad |11-12| = 1, \quad |13-11| = 2, \quad |12-13| = 1.$$

Bình phương chênh lệch:

$$2^2 = 4, \quad 1^2 = 1, \quad 2^2 = 4, \quad 1^2 = 1.$$

Tổng bằng $4 + 1 + 4 + 1 = 10$. MSE của naive trên giai đoạn lịch sử là:

$$\text{MSE}_{\text{naive}}^{(A)} = \frac{1}{T-1}\sum_{t=2}^{5}(r_t^{(A)} - r_{t-1}^{(A)})^2 = \frac{10}{4} = 2{,}5.$$

Giả sử giai đoạn forecast có 3 ngày ($h = 3$), với giá trị thật và dự báo từ mô hình:

$$r_{\text{forecast}}^{(A)} = (11,\ 14,\ 13), \qquad p_{\text{forecast}}^{(A)} = (10,\ 16,\ 10).$$

Sai số bình phương:

$$(11-10)^2 = 1, \quad (14-16)^2 = 4, \quad (13-10)^2 = 9.$$

Tổng là $1 + 4 + 9 = 14$, nên

$$\text{MSE}_{\text{model}}^{(A)} = \frac{14}{3} \approx 4{,}67.$$

Từ đó, MSSE và RMSSE cho sản phẩm A là:

$$MSSE^{(A)} = \frac{\text{MSE}_{\text{model}}^{(A)}}{\text{MSE}_{\text{naive}}^{(A)}} = \frac{4{,}67}{2{,}5} \approx 1{,}87,$$

$$RMSSE^{(A)} = \sqrt{MSSE^{(A)}} \approx \sqrt{1{,}87} \approx 1{,}37.$$

Diễn giải: mô hình A có sai số bình phương trung bình lớn hơn naive khoảng 1,87 lần (RMSSE khoảng 1,37).

Sản phẩm B (thang lớn, có outlier). Giờ xét sản phẩm B với doanh số lớn, dữ liệu lịch sử 5 ngày:

$$r_{\text{hist}}^{(B)} = (1000,\ 1020,\ 980,\ 1010,\ 995).$$

Chênh lệch liên tiếp:

$$|1020 - 1000| = 20, \quad |980 - 1020| = 40, \quad |1010 - 980| = 30, \quad |995 - 1010| = 15.$$

Bình phương:

$$20^2 = 400, \quad 40^2 = 1600, \quad 30^2 = 900, \quad 15^2 = 225.$$

Tổng bằng $400 + 1600 + 900 + 225 = 3125$. MSE của naive:

$$\text{MSE}_{\text{naive}}^{(B)} = \frac{3125}{4} = 781{,}25.$$

Giai đoạn forecast 3 ngày của sản phẩm B (có một outlier rất lớn):

$$r_{\text{forecast}}^{(B)} = (990,\ 5000,\ 1005), \qquad p_{\text{forecast}}^{(B)} = (1000,\ 3000,\ 900).$$

Sai số bình phương:

$$(990 - 1000)^2 = 10^2 = 100,$$

$$(5000 - 3000)^2 = 2000^2 = 4{,}000{,}000,$$

$$(1005 - 900)^2 = 105^2 = 11{,}025.$$

Tổng sai số bình phương là

$$100 + 4{,}000{,}000 + 11{,}025 = 4{,}011{,}125,$$

nên

$$\text{MSE}_{\text{model}}^{(B)} = \frac{4{,}011{,}125}{3} \approx 1{,}337{,}042.$$

Từ đó:

$$MSSE^{(B)} = \frac{\text{MSE}_{\text{model}}^{(B)}}{\text{MSE}_{\text{naive}}^{(B)}} = \frac{1{,}337{,}042}{781{,}25} \approx 1{,}71,$$

$$RMSSE^{(B)} = \sqrt{MSSE^{(B)}} \approx \sqrt{1{,}71} \approx 1{,}31.$$

Bước 3: RMSSE vừa nhạy với outlier, vừa so sánh được giữa các thang đo

Nếu ta chỉ nhìn **RMSE** trên thang gốc:

$$RMSE^{(A)} = \sqrt{4{,}67} \approx 2{,}16 \quad \text{(đơn vị: sản phẩm A)},$$

$$RMSE^{(B)} = \sqrt{1{,}337{,}042} \approx 1156 \quad \text{(đơn vị: sản phẩm B)}.$$

Hai con số này nằm ở hai thang rất khác nhau (khoảng 2 so với hơn 1000), nên *không thể* so sánh trực tiếp: việc sai 2 đơn vị trên thang 10–20 không cùng ý nghĩa với sai 1000+ đơn vị trên thang 1000–5000.

Khi chuyển sang **RMSSE**, cả hai được chuẩn hoá theo mức biến động nội tại của từng chuỗi:

$$RMSSE^{(A)} \approx 1{,}37, \qquad RMSSE^{(B)} \approx 1{,}31.$$

Hai giá trị này bỗng nhiên trở nên *so sánh được*: cả hai mô hình đều có lỗi bình phương trung bình lớn hơn naive (RMSSE > 1), và mức độ *tệ hơn naive* khá tương đương nhau (khoảng 1,3–1,4 lần). Nói cách khác, dù sản phẩm B có một outlier rất lớn (đơn đặt hàng 5000 so với dự báo 3000), RMSSE không bị số tuyệt đối 2000 "làm nổ tung" chỉ số, vì nó nhìn mọi thứ trên nền *mức độ dao động bình thường* của chuỗi đó (được đo bằng MSE của naive).

Đồng thời, RMSSE vẫn **thừa hưởng tính nhạy với outlier của RMSE**: trong sản phẩm B, sai số 2000 được bình phương thành 4 triệu, chi phối gần như toàn bộ $\text{MSE}_{\text{model}}^{(B)}$. Nếu trong forecast của B không có outlier đó mà chỉ toàn sai số nhỏ, RMSSE của B sẽ giảm đáng kể. Điều này cho thấy RMSSE vẫn "đau" với các lỗi cực lớn (như RMSE), nhưng nhờ được chia cho $\text{MSE}_{\text{naive}}$, nó kể câu chuyện trên thang *"so với biến động tự nhiên của chuỗi"* thay vì trên thang tuyệt đối.

Bước 4: Liên hệ MASE và RMSSE trong thực tế

Trong các hệ thống dự báo chuỗi thời gian đa sản phẩm (ví dụ: hàng nghìn SKU với doanh số từ vài đơn vị đến hàng chục nghìn đơn vị mỗi kỳ), việc báo cáo MAE hay RMSE thô cho từng chuỗi gần như vô nghĩa khi so sánh chéo. Một mô hình có RMSE = 100 trên mặt hàng doanh số trung bình 10 000 có thể *tốt hơn nhiều* so với một mô hình có RMSE = 10 trên mặt hàng doanh số trung bình 20.

Chính vì vậy:

- **MASE** chuẩn hoá sai số tuyệt đối theo sai số "tự nhiên" của dự báo naive, giúp trả lời: *"mô hình sai bao nhiêu lần so với một chiến lược cực kỳ đơn giản"*.
- **RMSSE** làm điều tương tự cho sai số bình phương, giữ lại độ nhạy với outlier của RMSE, nhưng trên một thang có thể so sánh giữa các chuỗi.

Hai metric này đặc biệt phổ biến trong các bài toán như Demand Forecasting, Inventory Planning, hay Sales Forecasting, nơi người ta cần xếp hạng và so sánh hiệu suất của cùng một mô hình (hoặc nhiều mô hình) trên hàng trăm, hàng nghìn chuỗi thời gian khác nhau, với thang đo và mức dao động rất đa dạng. Thay vì đắm chìm trong một rừng MAE/RMSE không cùng đơn vị, các nhà phân tích chỉ cần nhìn vào MASE và RMSSE để xem, trên từng sản phẩm, mô hình đang *tốt hơn hay tệ hơn naive bao nhiêu lần*, và tổng thể hệ thống forecast đang hoạt động ra sao.

7 Các metric khác: RMSLE, Percentage Better, liên hệ với loss function

RMSLE: giảm ảnh hưởng của outlier theo log-scale

Root Mean Squared Logarithmic Error (RMSLE) được định nghĩa:

$$RMSLE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\log(r_i+1) - \log(p_i+1)\right)^2}.$$

Nhờ phép log, RMSLE tập trung hơn vào sai số tương đối (theo tỷ lệ) và giảm ảnh hưởng của các outlier rất lớn. Slide về doanh số theo vùng minh hoạ rằng trong khi RMSE bị kéo lên rất cao bởi một vài vùng có dự đoán lệch xa, RMSLE lại giữ giá trị ở mức hợp lý hơn, phản ánh rằng về mặt *tỷ lệ*, mô hình không tệ đến vậy ở hầu hết các vùng.

Ví dụ. Giả sử doanh số thực (triệu USD) là $(2, 3, 5, 10)$ và dự đoán là $(2.1, 3.2, 5.5, 20)$. RMSE sẽ bị chi phối bởi sai số 10 ở điểm cuối, trong khi RMSLE làm giảm độ nặng của sai số này vì $\log(10+1)$ và $\log(20+1)$ không cách xa nhau nhiều bằng 10 và 20 trên thang tuyến tính. Điều này phù hợp trong những bài toán mà chúng ta quan tâm đến tỷ lệ tăng trưởng (growth rate) hơn là chênh lệch tuyệt đối.

Percentage Better (PB): đếm số lần "thắng" benchmark

Thay vì tính trung bình lỗi, chỉ số Percentage Better (PB) đếm tỷ lệ số lần mô hình có lỗi nhỏ hơn benchmark:

$$PB(MAE) = \frac{100}{N}\sum_{i=1}^{N}\mathbf{1}\left(|r_i - p_i| < |r_i - b_i|\right).$$

Nếu $PB = 60\%$, ta hiểu đơn giản là "trong 60% số lần, mô hình dự đoán tốt hơn mô hình mốc". Metric này không nhìn vào độ lớn sai số, mà chỉ quan tâm đến việc "thắng/thua" từng lần, nên phù hợp khi ta muốn tối đa hoá xác suất dự đoán tốt hơn baseline, chứ không tối ưu giá trị kỳ vọng của sai số.

Pearson R: cần nhưng chưa đủ

Hệ số tương quan Pearson:

$$R = \frac{\sum_{i=1}^{N}(p_i - \bar{p})(r_i - \bar{r})}{\sqrt{\sum_{i=1}^{N}(p_i - \bar{p})^2}\sqrt{\sum_{i=1}^{N}(r_i - \bar{r})^2}},$$

đo mức độ tương quan tuyến tính giữa dự đoán và giá trị thật.

Ví dụ. Dùng lại dữ liệu giá nhà với mô hình (1):

$$\mathbf{r} = (200, 300, 250, 500, 300), \quad \mathbf{p}^{(1)} = (210, 290, 260, 500, 300).$$

Ta tính:

$$\bar{r} = \frac{200+300+250+500+300}{5} = 310, \quad \bar{p}^{(1)} = \frac{210+290+260+500+300}{5} = 312.$$

Sau đó tính tích lệch chuẩn và v.v. (bạn có thể trình bày chi tiết trong phụ lục nếu muốn). Kết quả R sẽ rất gần 1, vì mô hình (1) bám rất tốt xu hướng của dữ liệu.

Case mô hình gấp đôi. Giờ xét mô hình (3): $\mathbf{p}^{(3)} = 2\mathbf{r} = (400, 600, 500, 1000, 600)$.

Ta có:

$$\bar{p}^{(3)} = \frac{400 + 600 + 500 + 1000 + 600}{5} = 620.$$

Nếu bạn tính R giữa $\mathbf{r}$ và $\mathbf{p}^{(3)}$, bạn sẽ thu được $R = 1$ (tương quan tuyến tính hoàn hảo), dù sai số tuyệt đối rất lớn:

$$e^{(3)} = (200, 300, 250, 500, 300).$$

MAE lúc này là 310 nghìn USD, cực kỳ tệ về mặt thực tế. Ví dụ này cho thấy: R **cao không bảo đảm dự đoán tốt**, nó chỉ nói rằng mối quan hệ tuyến tính giữa r và p rất chặt (ở đây là $p = 2r$).

Variance Accounted For (VAF)

Variance Accounted For được định nghĩa:

$$VAF = 1 - \frac{\sum_{i=1}^{N}(p_i - r_i)^2}{\sum_{i=1}^{N}(r_i - \bar{r})^2}.$$

Nếu tử số nhỏ (sai số bình phương nhỏ), VAF gần 1; nếu tử số lớn, VAF có thể bằng 0 hoặc âm (tức là mô hình tệ hơn cả việc đoán bằng trung bình). VAF về mặt công thức giống với cách định nghĩa R^2 trong hồi quy tuyến tính, nhưng ở đây không cần giả định mô hình là đường thẳng.

R^2 trong hồi quy tuyến tính

Trong hồi quy tuyến tính đơn, ta có:

$$R^2 = \frac{ESS}{TSS} = 1 - \frac{RSS}{TSS},$$

trong đó:

$$TSS = \sum_{i=1}^{N}(r_i - \bar{r})^2, \quad RSS = \sum_{i=1}^{N}(r_i - p_i)^2, \quad ESS = TSS - RSS.$$

Với mô hình tuyến tính phù hợp, có thể chứng minh rằng:

$$R^2 = R^2_{\text{Pearson}},$$

tức là bình phương hệ số tương quan giữa r và p.

Ví dụ. Giả sử:

$$r = (1, 2, 3), \quad p = (1.2, 1.9, 3.1).$$

Ta có:

$$\bar{r} = 2, \quad TSS = (1-2)^2 + (2-2)^2 + (3-2)^2 = 2.$$

Sai số:

$$RSS = (1 - 1.2)^2 + (2 - 1.9)^2 + (3 - 3.1)^2 = 0.01 + 0.01 + 0.01 = 0.03.$$

Do đó:

$$R^2 = 1 - \frac{0.03}{2} = 0.985.$$

Nếu bạn tính Pearson R cho r và p, bạn sẽ thu được $R \approx 0.992$ và $R^2_{\text{Pearson}} \approx 0.985$, khớp với R^2 từ TSS/RSS.

Khi R^2 không còn là R^2 nữa

Khi mô hình *không phải* là hồi quy tuyến tính (ví dụ: mạng nơ-ron, cây quyết định, mô hình phi tuyến bất kỳ), nhiều thư viện vẫn trả về một giá trị dạng:

$$RSquared = 1 - \frac{\sum(r_i - p_i)^2}{\sum(r_i - \bar{r})^2},$$

và gọi đó là "R^2". Về mặt số học, nó giống VAF, nhưng không còn đảm bảo bằng bình phương hệ số tương quan nữa. Giá trị này có thể lớn hơn 1 hoặc âm, nếu mô hình rất tệ. Vấn đề xảy ra khi ta đọc kết quả mà *tưởng* nó là "R^2 kinh điển" trong hồi quy tuyến tính, rồi suy diễn như trong giáo trình hồi quy. Vì vậy:

- Với mô hình phi tuyến, hãy coi $1 - RSS/TSS$ là một dạng chỉ số tương đối (giống VAF).
- Không nên dùng R^2 như **thước đo duy nhất** để đánh giá mô hình.
- Nên kết hợp thêm MAE, RMSE, IOA, VAF, v.v.

Adjusted R^2: phạt các biến vô nghĩa

Khi có nhiều biến độc lập, R^2 luôn tăng khi thêm biến, kể cả biến vô nghĩa. Vì vậy, ta dùng Adjusted R^2:

$$R^2_{\text{adj}} = 1 - \frac{(1 - R^2)(N - 1)}{N - k - 1},$$

trong đó k là số biến độc lập. Adjusted R^2 "phạt" mô hình khi thêm biến mà không cải thiện thực sự. Nếu một biến mới làm R^2 tăng nhưng không đủ mạnh để bù cho việc mất thêm một bậc tự do, Adjusted R^2 sẽ giảm.

Ví dụ. Mô hình A dùng 1 biến, cho $R^2 = 0.70$ với $N = 100$. Mô hình B thêm 1 biến nữa, R^2 tăng lên 0.72. Tuy nhiên, khi tính Adjusted R^2, có thể xảy ra:

$$R^2_{\text{adj,A}} = 0.695, \quad R^2_{\text{adj,B}} = 0.690.$$

Lúc này, dù R^2 của B cao hơn, Adjusted R^2 của B thấp hơn A, hàm ý rằng biến mới thực chất không giúp mô hình giải thích dữ liệu tốt hơn khi tính đến độ phức tạp.

8 Taylor Diagram: nhìn ba thống kê trong một hình

Taylor diagram cho phép hiển thị đồng thời:

- Độ lệch chuẩn của dự đoán σ_p,
- Hệ số tương quan R,
- CRMSD (Centered Root Mean Square Difference) giữa p và r.

Dựa trên quan hệ:

$$CRMSD^2 = \sigma_r^2 + \sigma_p^2 - 2\sigma_r\sigma_p R.$$

Trong biểu đồ:

- Điểm "REF" (dữ liệu thật) nằm trên trục hoành với độ lệch chuẩn σ_r, $R = 1$, $CRMSD = 0$.
- Mỗi mô hình tương ứng một điểm khác; càng gần REF thì mô hình càng tốt.

Đây là công cụ trực quan mạnh để so sánh nhiều mô hình hồi quy cùng lúc.

9 Loss vs Metric: train bằng gì, đánh giá bằng gì

Một điểm quan trọng: **loss function** dùng để *train* mô hình, còn **evaluation metric** dùng để *đánh giá* sau train. Loss cần khả vi (hoặc ít nhất là tối ưu được bằng thuật toán mình chọn), trong khi metric không nhất thiết phải khả vi.

- MSE/RMSE thường được dùng như loss vì dễ đạo hàm và tối ưu.
- MAE khó tối ưu hơn (gradient không trơn tại 0), nhưng nếu mục tiêu business là giảm sai số tuyệt đối trung bình, việc dùng MAE làm loss lại rất hợp lý.
- Huber loss là một lựa chọn dung hoà: nhỏ thì giống MSE, lớn thì giống MAE.

Huber loss:

$$L_\delta(r_i, p_i) = \begin{cases} \frac{1}{2}(r_i - p_i)^2, & \text{nếu } |r_i - p_i| \le \delta, \\ \delta|r_i - p_i| - \frac{1}{2}\delta^2, & \text{ngược lại.} \end{cases}$$

Điều quan trọng là: nếu ta *đánh giá* mô hình bằng MAE, nhưng lại *train* bằng MSE, thì mô hình sẽ tối ưu theo mean trong khi ta quan tâm tới median. Sự thiếu nhất quán này có thể khiến mô hình "tốt theo loss" nhưng không tốt theo metric mà business dùng để đo lường thành công. Một cách dễ hiểu nhất để thấy vai trò khác nhau giữa **loss function** và **evaluation metric** là dựng một ví dụ số đơn giản, rồi cố tình cho *loss* và *metric* "nhìn" theo hai hướng khác nhau.

Giả sử ta đang làm một bài toán hồi quy 1 chiều rất đơn giản: dự đoán một giá trị duy nhất c (ví dụ, "mức nhu cầu trung bình") cho tất cả các quan sát. Ta có 5 giá trị thực:

$$r = (10,\ 11,\ 9,\ 10,\ 100).$$

Bốn điểm đầu quanh quanh 9–11, còn điểm thứ năm là một **outlier** rất lớn (100).

a. Nếu dùng MSE làm loss: mô hình sẽ "ưu tiên" mean

Ta giả sử mô hình chỉ học một hằng số c và ta train bằng **MSE**:

$$L_{\text{MSE}}(c) = \frac{1}{5}\sum_{i=1}^{5}(r_i - c)^2.$$

Ta biết kết quả tối ưu (về mặt lý thuyết) sẽ là

$$c^\star_{\text{MSE}} = \bar{r} = \frac{10 + 11 + 9 + 10 + 100}{5} = \frac{140}{5} = 28.$$

Vậy mô hình "tối ưu theo MSE" sẽ dự đoán

$$p_i = 28 \quad \text{cho mọi } i.$$

Ta tính **MAE** (metric) của mô hình này:

$$MAE(c = 28) = \frac{1}{5}\left(|10 - 28| + |11 - 28| + |9 - 28| + |10 - 28| + |100 - 28|\right).$$

Các sai số tuyệt đối là:

$$18,\ 17,\ 19,\ 18,\ 72.$$

Tổng sai số tuyệt đối:

$$18 + 17 + 19 + 18 + 72 = 144,$$

nên

$$MAE(c = 28) = \frac{144}{5} = 28{,}8.$$

Nhìn vào $c = 28$, ta thấy:

- Bốn quan sát "bình thường" quanh 9–11 bị dự đoán lệch rất xa.
- Chỉ có outlier 100 là gần hơn (sai 72) so với nếu ta chọn một giá trị nhỏ.
- Tuy nhiên, **vì loss là MSE**, thuật toán đã chọn "hy sinh" bốn điểm bình thường để giảm bớt lỗi *bình phương* ở điểm 100.

Nếu business quan tâm tới "trung bình bình phương sai số" (ví dụ trong một số bài toán đo năng lượng, power loss, v.v.), thì cách làm này là hợp lý. Nhưng nếu business đo lường bằng MAE ("trung bình sai lệch tuyệt đối"), đây chưa chắc là best choice.

b. Nếu dùng MAE làm loss: mô hình sẽ "ưu tiên" median

Giờ ta tưởng tượng ta **train bằng MAE**:

$$L_{\text{MAE}}(c) = \frac{1}{5}\sum_{i=1}^{5} |r_i - c|.$$

Về lý thuyết, nghiệm tối ưu là **median** của tập r. Sắp xếp:

$$(9,\ 10,\ 10,\ 11,\ 100),$$

median là 10. Vậy mô hình "tối ưu theo MAE" sẽ dự đoán:

$$p_i = 10 \quad \text{cho mọi } i.$$

Ta tính lại MAE:

$$MAE(c = 10) = \frac{1}{5}\left(|10 - 10| + |11 - 10| + |9 - 10| + |10 - 10| + |100 - 10|\right).$$

Các sai số tuyệt đối:

$$0,\ 1,\ 1,\ 0,\ 90.$$

Tổng là 92, nên:

$$MAE(c = 10) = \frac{92}{5} = 18{,}4.$$

So sánh:

$$MAE(c = 28) = 28{,}8 \quad \text{(train bằng MSE, tối ưu mean)},$$

$$MAE(c = 10) = 18{,}4 \quad \text{(train bằng MAE, tối ưu median)}.$$

Nếu **metric đánh giá mà business dùng là MAE**, mô hình tối ưu MSE rõ ràng *tệ hơn* mô hình tối ưu MAE. Mô hình MSE chịu ảnh hưởng mạnh của outlier 100, trong khi mô hình MAE "bỏ qua" phần nào outlier và tập trung phục vụ nhóm khách hàng chính quanh 9–11.

Đây là ví dụ rõ ràng:

- Nếu ta *train bằng MSE* nhưng *đánh giá bằng MAE*, mô hình "tối ưu theo loss" chưa chắc là mô hình tốt nhất theo metric mà business quan tâm.
- MSE hướng mô hình về mean, còn MAE hướng mô hình về median. Nếu business dùng MAE để báo cáo, mà đội kỹ thuật lại tối ưu MSE, thì mục tiêu của hai bên đang lệch nhau.

c. Huber loss: dung hoà giữa MSE và MAE

Nhớ lại Huber loss:

$$L_\delta(r_i, p_i) = \begin{cases} \frac{1}{2}(r_i - p_i)^2, & \text{nếu } |r_i - p_i| \le \delta, \\ \delta|r_i - p_i| - \frac{1}{2}\delta^2, & \text{ngược lại.} \end{cases}$$

Ta chọn một ngưỡng δ, ví dụ $\delta = 5$. Khi đó:

- Nếu sai số $|r_i - p_i| \le 5$, ta dùng loss dạng bình phương (giống MSE) để giữ tính trơn, gradient mượt.
- Nếu sai số lớn hơn 5, loss chuyển sang dạng tuyến tính (gần giống MAE), giảm bớt ảnh hưởng của outlier.

Ta so sánh Huber loss cho hai mô hình $c = 28$ và $c = 10$ trên cùng dữ liệu $r = (10, 11, 9, 10, 100)$, với $\delta = 5$.

Trường hợp $c = 28$: Sai số:

$$(10 - 28,\ 11 - 28,\ 9 - 28,\ 10 - 28,\ 100 - 28) = (-18,\ -17,\ -19,\ -18,\ 72).$$

Tuyệt đối:

$$18,\ 17,\ 19,\ 18,\ 72.$$

Tất cả đều lớn hơn $\delta = 5$, nên mọi điểm đều rơi vào nhánh $|r_i - p_i| > \delta$:

$$L_\delta(r_i, p_i) = \delta|r_i - p_i| - \tfrac{1}{2}\delta^2.$$

Với $\delta = 5$, ta có:

$$L_\delta(10, 28) = 5 \cdot 18 - \tfrac{1}{2} \cdot 25 = 90 - 12{,}5 = 77{,}5,$$
$$L_\delta(11, 28) = 5 \cdot 17 - 12{,}5 = 72{,}5,$$
$$L_\delta(9, 28) = 5 \cdot 19 - 12{,}5 = 82{,}5,$$
$$L_\delta(10, 28) = 5 \cdot 18 - 12{,}5 = 77{,}5,$$
$$L_\delta(100, 28) = 5 \cdot 72 - 12{,}5 = 360 - 12{,}5 = 347{,}5.$$

Tổng:

$$\sum L_\delta(r_i, 28) = 77{,}5 + 72{,}5 + 82{,}5 + 77{,}5 + 347{,}5 = 657{,}5,$$

Huber loss trung bình:

$$\overline{L_\delta}(c = 28) = \frac{657{,}5}{5} = 131{,}5.$$

Trường hợp $c = 10$: Sai số:

$$(10 - 10,\ 11 - 10,\ 9 - 10,\ 10 - 10,\ 100 - 10) = (0,\ 1,\ -1,\ 0,\ 90).$$

Tuyệt đối:

$$0,\ 1,\ 1,\ 0,\ 90.$$

Với $\delta = 5$:

- Bốn điểm đầu có $|e_i| \le 5$ dùng nhánh MSE.
- Điểm thứ năm có $|e_5| = 90 > 5$ dùng nhánh MAE.

Ta tính:

$$L_\delta(10, 10) = \tfrac{1}{2}(0)^2 = 0,$$
$$L_\delta(11, 10) = \tfrac{1}{2}(1)^2 = 0{,}5,$$
$$L_\delta(9, 10) = \tfrac{1}{2}(1)^2 = 0{,}5,$$
$$L_\delta(10, 10) = 0,$$
$$L_\delta(100, 10) = 5 \cdot 90 - 12{,}5 = 450 - 12{,}5 = 437{,}5.$$

Tổng:

$$\sum L_\delta(r_i, 10) = 0 + 0{,}5 + 0{,}5 + 0 + 437{,}5 = 438{,}5,$$

Huber loss trung bình:

$$\overline{L_\delta}(c = 10) = \frac{438{,}5}{5} = 87{,}7.$$

So sánh:

$$\overline{L_\delta}(c = 28) = 131{,}5, \qquad \overline{L_\delta}(c = 10) = 87{,}7.$$

Với Huber loss, mô hình $c = 10$ (gần median) vẫn "thắng" mô hình $c = 28$, nhưng:

- Outlier 100 bị phạt mạnh (vì $|e_5| = 90 > \delta$, rơi vào nhánh tuyến tính).
- Các sai số nhỏ (0, 1, 1, 0) được nhìn dưới dạng bình phương, giúp gradient trơn và ổn định trong quá trình tối ưu.

d. Kết luận từ ví dụ

Ví dụ nhỏ này cho thấy:

- **Loss function** (MSE, MAE, Huber, …) quyết định *mô hình học cái gì* trong quá trình train.
- **Evaluation metric** (MAE, RMSE, MAPE, …) quyết định *ta đánh giá mô hình như thế nào* sau khi train xong.
- Nếu ta *train bằng MSE* nhưng *đánh giá bằng MAE*, mô hình sẽ ưu tiên mean trong khi business đo thành công bằng median. Sự lệch pha này có thể khiến mô hình "đẹp trên loss" nhưng không tối ưu trên metric thực sự quan trọng.
- Huber loss là một ví dụ cho cách thiết kế loss gần với metric mong muốn (giảm ảnh hưởng outlier như MAE) nhưng vẫn giữ tính trơn để tối ưu bằng gradient (giống MSE).

Điều quan trọng trong thực tế là: ngay từ khi thiết kế hệ thống, ta nên xác định rõ *metric mà business dùng để đo lường thành công* (MAE, MAPE, RMSE, …), rồi chọn hoặc thiết kế loss function sao cho "nhìn cùng hướng" với metric đó, thay vì để loss và metric kéo mô hình theo hai mục tiêu khác nhau.

10 Một cảnh báo nhỏ: đừng dùng "accuracy" cho hồi quy

Lưu ý thường gặp

Trong classification, "accuracy" là metric phổ biến: tỷ lệ dự đoán đúng trên tổng mẫu. Trong regression, khái niệm "dự đoán đúng tuyệt đối" gần như vô nghĩa (hiếm khi p_i trùng chính xác r_i), nên **không dùng accuracy** để nói về chất lượng mô hình hồi quy.
Cách nói như "mô hình đạt accuracy 90% trong bài toán dự báo giá" là sai khái niệm. Thay vào đó, hãy nói: MAE = 2.3, RMSE = 3.1, $R^2 = 0.82$, VAF = 0.80, ...

Thông điệp rất thực tế: **không metric nào đủ để một mình "kể hết câu chuyện"**. Mỗi chỉ số chỉ cho ta một lát cắt của hiệu năng mô hình, giống như nhìn một căn phòng qua một ô cửa sổ. Nếu ta chỉ bám vào một con số duy nhất, rất dễ hiểu sai hoặc bỏ lỡ những rủi ro quan trọng.

10.1 Classification: tại sao không thể chỉ nhìn accuracy

Trong bài toán phân loại, nhiều người có xu hướng chăm chăm nhìn vào *accuracy* vì nó dễ hiểu: tỷ lệ dự đoán đúng trên tổng số mẫu. Tuy nhiên, bài conference nhắc lại một ví dụ kinh điển:

Giả sử ta có một bài toán phân loại bệnh, trong đó chỉ có 1% bệnh nhân là dương tính, 99% là âm tính. Nếu mô hình *luôn* dự đoán "âm tính" cho tất cả mọi người, accuracy sẽ là 99%. Con số này trông rất "đẹp", nhưng mô hình hoàn toàn vô dụng: nó không phát hiện được bất kỳ ca bệnh nào.

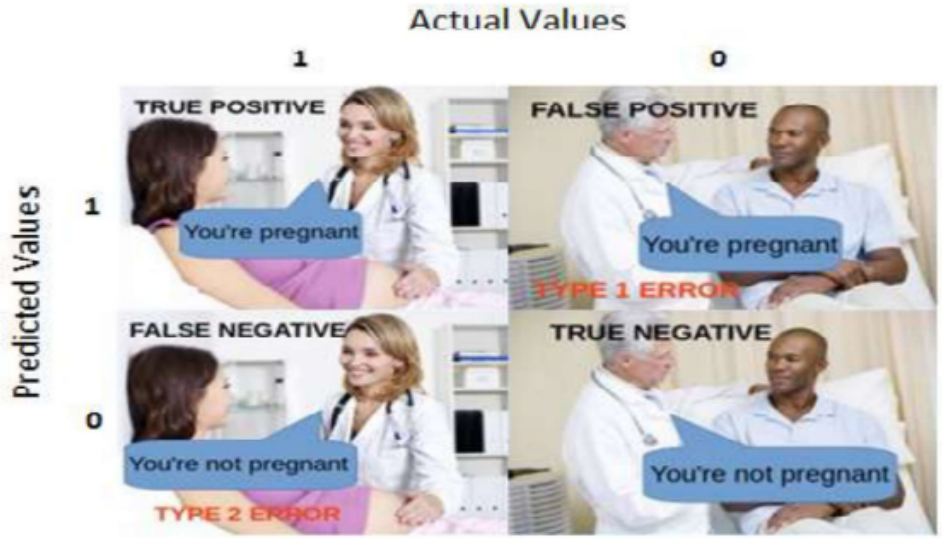
Chính vì vậy, trong classification người ta luôn đề xuất bộ chỉ số tối thiểu gồm:

- **Accuracy**: nhìn tổng thể mô hình đúng bao nhiêu phần trăm trên cả lớp dương và âm.
- **Precision**: trong số những mẫu mô hình dự đoán là dương, có bao nhiêu phần trăm thực sự dương. Nó trả lời câu hỏi "khi mô hình nói 'có bệnh', ta tin được tới mức nào?".
- **Recall**: trong số những mẫu thực sự dương, mô hình bắt được bao nhiêu phần trăm. Nó trả lời câu hỏi "mô hình bỏ sót bao nhiêu ca bệnh?".
- **F1-score**: trung bình điều hoà giữa precision và recall, giúp cân bằng hai yếu tố này khi ta muốn tối ưu cả phát hiện đúng và tránh báo động giả.

Ví dụ: nếu một mô hình ung thư có accuracy 99% nhưng recall chỉ 5%, nghĩa là nó bỏ *lỡ* 95% ca bệnh; một mô hình khác có accuracy 96% nhưng recall 90% và precision 85% có thể là lựa chọn tốt hơn nhiều, dù nhìn vào accuracy thuần tuý thì có vẻ "kém hơn". Bức thông điệp ở đây là: **accuracy một mình thường là không đủ, phải nhìn kèm precision, recall, F1 (và đôi khi thêm ROC-AUC)** để hiểu thật sự mô hình đang làm gì.

Để quan sát hơn sự khác nhau giữa các loại lỗi trong bài toán phân loại (false positive, false negative, true positive, true negative), ta xem xét ví dụ kinh điển về chẩn đoán *mang thai* trong Hình **??**.

Trong ví dụ này, trục dọc thể hiện **giá trị dự đoán** của mô hình (bác sĩ nói "bạn có thai" hay "bạn không có thai"), còn trục ngang thể hiện **giá trị thực tế** (thực sự có thai hay không):



Hình 1: Ví dụ về Confusion Matrix

- **True Positive (TP)**: góc trên bên trái. Bệnh nhân thực sự có thai (actual = 1) và bác sĩ cũng nói "Bạn có thai" (predicted = 1). Đây là trường hợp dự đoán đúng dương tính.
- **True Negative (TN)**: góc dưới bên phải. Bệnh nhân không có thai (actual = 0) và bác sĩ nói "Bạn không có thai" (predicted = 0). Đây là dự đoán đúng âm tính.
- **False Positive (FP) – Type I Error**: góc trên bên phải. Bệnh nhân *không* có thai (actual = 0) nhưng bác sĩ lại nói "Bạn có thai" (predicted = 1). Đây là *lỗi báo động giả* (Type I): mô hình kết luận có sự kiện trong khi thực tế không có. Trong nhiều bài toán y khoa, false positive gây lo lắng, tốn thêm xét nghiệm, nhưng vẫn có thể chấp nhận nếu giúp tăng khả năng phát hiện bệnh sớm.
- **False Negative (FN) – Type II Error**: góc dưới bên trái. Bệnh nhân thực sự có thai (actual = 1) nhưng bác sĩ nói "Bạn không có thai" (predicted = 0). Đây là *lỗi bỏ sót* (Type II): mô hình không phát hiện ra sự kiện quan trọng. Trong ví dụ mang thai (hoặc ung thư, bệnh nặng), false negative thường nguy hiểm hơn nhiều vì bệnh nhân không được chăm sóc, điều trị kịp thời.

Hình minh hoạ giúp ta thấy vì sao trong bài toán phân loại không thể chỉ nhìn **accuracy**. Một mô hình có accuracy cao nhưng:

- nếu chủ yếu mắc lỗi **false positive** (Type I), chi phí là *báo động giả*, làm tốn thêm kiểm tra, chi phí vận hành;
- nếu chủ yếu mắc lỗi **false negative** (Type II), chi phí là *bỏ sót ca bệnh*, có thể gây hậu quả nghiêm trọng cho bệnh nhân.

Các metric như **precision**, **recall** và **F1-score** chính là cách lượng hoá những trade-off này: precision nhạy với false positive, còn recall nhạy với false negative. Ví dụ mang thai trong Hình 1 vì thế là một minh hoạ trực quan cho việc **phải nhìn nhiều metric cùng lúc**, thay vì chỉ dựa vào một con số duy nhất như accuracy khi đánh giá mô hình phân loại.

10.2 Regression: một câu chuyện nhiều lớp với MAE, RMSE, R^2, VAF và các metric chuẩn hoá

Trong regression, bài conference đề xuất một "bộ khung" tương tự: không chỉ báo cáo một con số duy nhất như RMSE hoặc R^2, mà nên kết hợp nhiều metric để kể một câu chuyện đầy đủ hơn về sai số.

- **MAE** cho ta biết "trung bình dự đoán lệch bao nhiêu đơn vị" so với thực tế. Nó dễ hiểu, không nhạy quá mức với outlier và bám sát trực giác của người dùng. Ví dụ: "trung bình mô hình dự báo sai khoảng 2 °C" hoặc "sai khoảng 5 nghìn USD".
- **RMSE** phóng đại các sai số lớn do bình phương, nên phản ánh tốt hơn rủi ro của các *outlier*: nếu RMSE cao hơn nhiều so với MAE, đó gần như là đèn vàng báo hiệu có những trường hợp mô hình dự đoán rất tệ mà MAE đang "làm mịn" đi.
- R^2 **và Adjusted** R^2 cho ta góc nhìn về tỷ lệ phương sai của dữ liệu được mô hình giải thích. Ví dụ: $R^2 = 0{,}85$ nghĩa là 85% biến thiên của biến mục tiêu được mô hình nắm bắt. Adjusted R^2 giúp cân nhắc thêm độ phức tạp của mô hình khi có nhiều biến độc lập, tránh việc R^2 tăng ảo chỉ vì thêm biến không cần thiết.
- **VAF (Variance Accounted For)** tương tự như một dạng R^2 tính trên tổng thể, cho biết mô hình giảm được bao nhiêu phần trăm tổng bình phương sai số so với việc chỉ đoán bằng trung bình. VAF âm là dấu hiệu mô hình còn tệ hơn cả baseline trung bình.
- **Các metric chuẩn hoá** (như MAPE, MASE, RMSSE, NME, FGE, …) giúp đặt các mô hình trên nhiều thang đo khác nhau lên cùng một chuẩn, đặc biệt hữu ích khi ta có nhiều series/doanh mục với mức độ và đơn vị rất khác nhau. Ví dụ: MASE = 0,5 nghĩa là mô hình chỉ sai bằng một nửa so với dự báo naive, dù chuỗi là điện năng, doanh số hay nhiệt độ.

Hãy tưởng tượng ta đang so sánh hai mô hình dự báo nhu cầu sản phẩm:

- Mô hình X có MAE = 5, RMSE = 6, $R^2 = 0{,}9$, MASE = 0,4.
- Mô hình Y có MAE = 4, RMSE = 10, $R^2 = 0{,}92$, MASE = 0,3.

Nếu chỉ nhìn R^2, mô hình Y trông "nhỉnh hơn" (0,92 so với 0,9). Nếu chỉ nhìn MAE, mô hình Y cũng có vẻ tốt hơn (4 so với 5). Nhưng khi ta nhìn vào RMSE, Y có RMSE = 10, cao hơn rất nhiều so với X (6). Điều đó ngụ ý rằng mô hình Y, dù trung bình sai ít hơn, lại có vài trường hợp sai cực lớn, làm RMSE bật cao. Nếu đây là bài toán nhạy cảm với rủi ro (ví dụ thiếu hàng, quá tải, hay rủi ro tài chính), mô hình X có thể là lựa chọn an toàn hơn, dù MAE và R^2 thấp hơn một chút.

Nếu thêm MASE = 0,4 (X) và 0,3 (Y), ta biết rằng cả hai đều tốt hơn naive khá nhiều (chỉ sai 30–40% sai số của mô hình "hôm nay bằng hôm qua"). Khi nhìn toàn bộ bộ metric, ta có thể kể một câu chuyện rõ ràng cho business:

- Cả hai mô hình đều vượt xa baseline (MASE < 1).
- Mô hình Y trung bình sai ít hơn và giải thích được nhiều phương sai hơn.
- Nhưng mô hình Y cũng có đuôi sai số nặng hơn (RMSE cao hơn nhiều), chứa các trường hợp "dự báo thảm hoạ" mà X tránh được.

Tóm lại: metric là câu chữ, mô hình là nội dung

Chọn metric là chọn cách kể chuyện về mô hình. Với classification, ta cần cả accuracy, precision, recall, F1 (và đôi khi thêm ROC-AUC) để hiểu mô hình vừa "đúng nhiều", vừa "bắt được ca hiếm" và "không báo động giả quá nhiều". Với regression, ta cần ít nhất một metric về độ lệch (MAE), một metric nhạy với outlier (RMSE), một metric về phương sai giải thích (R^2 hoặc VAF), và, nếu có nhiều series/thang đo, thêm các metric chuẩn hoá như MASE/RMSSE.

Không có một con số nào đủ để kết luận "mô hình tốt hay xấu". Chỉ khi đặt nhiều metric cạnh nhau, ta mới có thể kể một câu chuyện trung thực, đầy đủ và có ý nghĩa với bối cảnh business: mô hình sai bao nhiêu, sai theo kiểu gì, có bỏ sót ca quan trọng không, có gây ra rủi ro cực đoan không, và có thực sự đáng để đưa vào vận hành hay chưa.

11 Chọn thước đo nào cho bài toán?

Không có một metric nào là "tốt nhất" cho mọi bài toán. Một số gợi ý thực tế:

- **Đánh giá cơ bản**: luôn bắt đầu với MAE và RMSE, kèm theo R hoặc R^2 để xem mô hình bắt được xu hướng tổng thể hay không.
- **Dữ liệu có giá trị gần 0**: tránh dùng MAPE thuần tuý; cân nhắc NME, FGE, hoặc chỉ số không chia trực tiếp cho r_i.
- **Chi phí sai số lớn rất cao**: ưu tiên RMSE (hoặc loss bậc cao hơn), vì nó phạt mạnh outlier.
- **So sánh nhiều mô hình/trên nhiều series**: sử dụng thêm IOA, VAF và các metric chuẩn hoá như MASE, RMSSE; Taylor diagram để có cái nhìn đa chiều.
- **Mô hình phi tuyến phức tạp**: đừng chỉ báo cáo một con số "R^2". Hãy ghi rõ công thức, và luôn kèm MAE, RMSE, VAF, IOA, v.v.

12 Kết luận: chọn metric như chọn góc nhìn

Qua hành trình đi qua ME, MAE, MSE/RMSE, MAPE, sMAPE, các dạng sai số tương đối (MRAE, GMRAE, RelMAE, RSE), sai số chuẩn hoá (MASE, RMSSE) và các metric đặc biệt như RMSLE hay PB, ta thấy mỗi thước đo giống như một cặp kính khác nhau để nhìn vào mô hình.

- ME cho ta biết mô hình đang thiên lệch về phía nào, nhưng dễ bị sai số dương và âm triệt tiêu.
- MAE dễ hiểu, ít nhạy với outlier nhưng che mờ thông tin về phân bố sai số (đặc biệt là các lỗi cực lớn).
- MSE/RMSE phạt nặng outlier, phù hợp khi sai số lớn là nguy hiểm, nhưng có thể bị chi phối bởi một vài điểm dữ liệu.
- MAPE và sMAPE mang lại trực giác phần trăm, nhưng phải cực kỳ cẩn thận khi dữ liệu có giá trị gần 0 hoặc phân bố không cân bằng.
- Metric tương đối (MRAE, RelMAE, RSE) hữu ích khi so với một baseline cụ thể, nhưng dễ bị đánh lừa khi cả mô hình lẫn baseline đều chịu ảnh hưởng của outlier.
- MASE và RMSSE giải quyết bài toán khác thang đo, giúp so sánh mô hình trên nhiều chuỗi khác nhau.

- RMSLE phù hợp khi ta quan tâm đến tỉ lệ và muốn giảm ảnh hưởng của những giá trị cực lớn.

Điều quan trọng nhất không phải là thuộc lòng công thức, mà là hiểu câu hỏi mà metric đang trả lời. Trong một dự án thực tế, câu hỏi có thể là: "Tôi có dám sử dụng dự báo này để đặt hàng thêm 1000 đơn vị sản phẩm không?", hoặc "Tôi có thể tin vào mô hình này khi nó báo nguy cơ quá tải cầu trong tuần tới không?". Mỗi câu hỏi tương ứng với một mức độ nhạy cảm khác nhau với outlier, với bias, với sai số tương đối hay tuyệt đối. Cuối cùng, hãy nhớ rằng metric đánh giá và loss function không nên tách rời. Nếu ta đo lường thành công bằng MAE, việc tối ưu mô hình theo MSE có thể dẫn ta đến một nghiệm tốt theo trung bình nhưng lại không tối ưu theo median. Khi metric và loss "nhìn về hai hướng khác nhau", mô hình sẽ rất khó đạt tới đúng loại hành vi mà ta thực sự mong muốn.