BÀI 4: HUẤN LUYỆN MÔ HÌNH VÀ LỖI

1. Ôn lại kiến thức đã học

Mô hình hồi quy tuyến tính và hàm mất mát

$$h_w(X) = w^T X$$

$$J(w) = \frac{1}{2m}\sum_{i=1}^{m}(h_w(x_i) - y_i)^2$$

Mô hình hồi quy Logistic và hàm mất mát:

$$h_w(X) = \frac{1}{1 + e^{-w^T X}}$$

$$J(w) = -\frac{1}{m}\sum_{i=1}^{m}\Big(y_i log\big(h_w(x_i)\big) + (1 - y_i) log\big(1 - h_w(x_i)\big)\Big)$$

2. Huấn luyện mô hình

Với tập dữ liệu ban đầu $D = \{(X, y) | X \in R^{m \times n} \text{ và } y \in R^m\}$ nếu như chúng ta đem toàn bộ đi huấn luyện mô hình như cách làm từ trước đến nay thì không thể đánh giá được hiệu năng của mô hình được huấn luyện. Do vậy tập dữ liệu *D* ban đầu được chia thành 2 tập dữ liệu là:

- Tập dữ liệu huấn luyện (training set): $D_{train} = \{(X_{train}, y_{train})\}$ thông thường $D_{train} = 70\% D$;
- Tập dữ liệu kiểm thử (test set): $D_{test} = \{(X_{test}, y_{test})\}$ thông thường $D_{test} = 30\% D$.
- *Ghi chú: việc phân chia tập dữ liệu D thành train – test được thực hiện với hàm train_test_split() của thư viện sklearn.*

Dựa vào 2 tập dữ liệu train – test này, mô hình được huấn luyện trên tập D_{train} tạm kí hiệu là h_w^{train}. Sau đó, mô hình h_w^{train} sẽ được đánh giá hiệu năng thông qua so sánh kết quả dự đoán của nó $h_w^{train}(X_{test})$ với giá trị thực sự tương ứng y_{test}. Độ sai khác giữa dự đoán của mô hình $h_w^{train}(X_{test})$ so với giá trị thực tế y_{test} được gọi là ***độ lỗi – error***. Có khá nhiều công thức tính độ lỗi được đề xuất. Trong phạm vi bài học này, chúng ta sử dụng công thức đánh giá độ lỗi mô hình *MSE.*

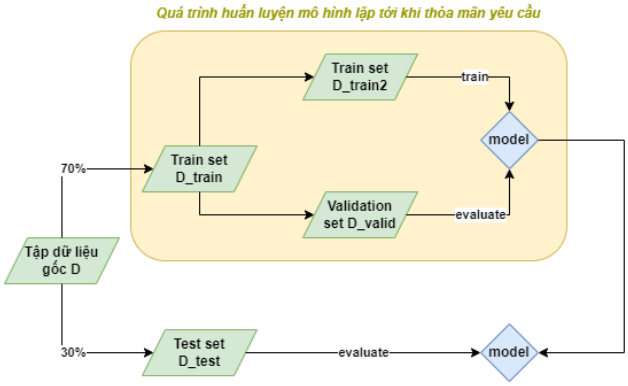
$$MSE = \frac{1}{|D_{test}|}\sum_{D_{test}}(h_w(X_{test}) - y_{test})^2$$

Vấn đề đặt ra là, *vậy thì làm thế nào để chắc chắn là mô hình h_w^{train} là mô hình được huấn luyện tốt theo nghĩa là hiệu năng của h_w^{train} cũng được đánh giá bằng chỉ số đo độ lỗi (ví dụ: MSE)?*

Để đạt được mục đích nêu trên, người ta áp dụng quy trình chia train – test một lần nữa với D_{train} thành D_{train2} và $D_{validation}$. Trong đó D_{train2} được sử dụng để huấn luyện mô hình như cách chúng ta vẫn làm từ trước đến nay. Mô hình thu được sẽ được đánh giá với $D_{validation}$ theo phương pháp đo độ lỗi (ví dụ: $MSE = \frac{1}{|D_{validation}|}\sum_{D_{validation}}\left(h_w^{train2}(X_{validation}) - y_{validation}\right)^2$). Quá trình huấn luyện sẽ được lặp lại cho đến khi *MSE* đạt đến ngưỡng chấp nhận được thì dừng lại. Cuối cùng mô hình h_w^{train} thu được tại bước lặp cuối cùng này sẽ được đánh giá hiệu năng với D_{test}.

$$MSE = \frac{1}{|D_{test}|}\sum_{D_{test}}(h_w^{train}(X_{test}) - y_{test})^2$$

Quá trình huấn luyện và kiểm thử mô hình như trên được tổng quát trong Hình 1:



Hình 1: Quy trình huấn luyện và kiểm thử mô hình đơn giản

2. Các hiện tượng xảy ra với mô hình được huấn luyện

Khi thực hiện huấn luyện và kiểm thử mô hình, dựa vào độ lỗi chúng ta có thể xác định được các hiện tượng sau:

- Một mô hình được gọi là ***tốt*** nếu cả độ lỗi huấn luyện và độ lỗi kiểm thử ***đều thấp***;
- Khi độ lỗi huấn luyện (***training error***) ***thấp*** mà độ lỗi kiểm thử (***test error***) ***cao*** thì mô hình đó bị ***overfitting***;

- Khi độ lỗi huấn luyện (***training error***) ***cao*** mà độ lỗi kiểm thử (***test error***) ***cao*** thì mô hình đó bị ***underfitting***.

Dựa vào số liệu thống kê về kết quả dự đoán của mô hình so với giá trị thực tế tương ứng, hai thông số độ lệch (Bias) và phương sai (Variance) được sử dụng để mô tả tóm tắt kết quả dự đoán với ý nghĩa như sau:

- Độ lệch (Bias) thể hiện sự chênh lệch giữa giá trị trung bình mà mô hình dự đoán so với giá trị thực tế tương ứng;
- Phương sai (Variance) thể hiện độ phân tán của giá trị mà mô hình dự đoán so với giá trị thực tế tương ứng.

3. Các phương pháp hạn chế overfitting

Cách 1: Giảm số lượng thuộc tính:

- Lựa chọn thuộc tính nào được giữ lại, thuộc tính nào phải loại bỏ (bằng tay);
- Áp dụng thuật toán lựa chọn thuộc tính.

Cách 2: Điều tiết (regularization)

- Giữ lại tất cả các thuộc tính nhưng giảm giá trị của hệ số w_i;
- Phương pháp này hữu hiệu khi có nhiều thuộc tính (x_i).

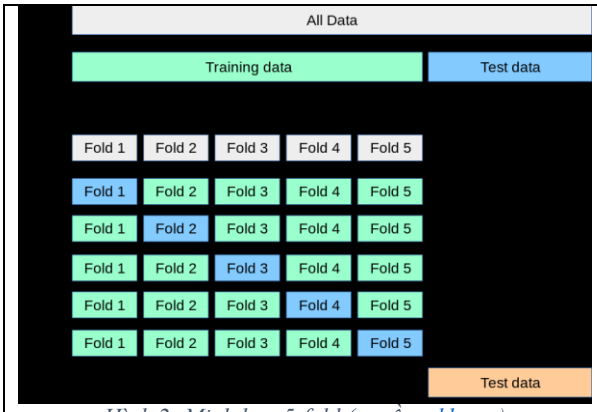
Cách 3: Phương pháp validation (10-fold cross-validation)

- Căn cứ nhận định: "Một mô hình được gọi là ***tốt*** nếu cả độ lỗi huấn liệu và độ lỗi kiểm thử ***đều thấp***".
- Người ta lấy ra từ training set 1 tập nhỏ gọi là ***validation set*** – đóng vai trò là test set trong quá trình huấn luyện. Phần còn lại của training set đóng vai trò là tập huấn luyện thực thụ.
- Dựa trên 2 tập dữ liệu training set mới và validation set quá trình huấn luyện sẽ được lặp lại cho đến khi độ lỗi trên 2 tập này đều thấp thì dừng lại.
- Mô hình thu được sẽ được đem đánh giá lại trên tập test set.
- Cross-validation là cải tiến của phương pháp validation. Áp dụng cho lượng dữ liệu của validation set là nhỏ. Người ta chia training set ra làm k tập con không giao

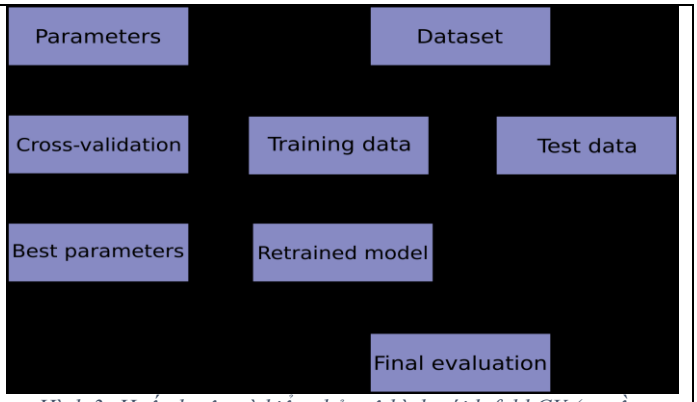
nhau (k-fold cross validation). Tại mỗi bước lặp, lấy ra 1 tập con trong k tập để làm tập validation.

- Trong thực tế, k thường được chọn là 10. Chúng ta có phương pháp 10-fold cross-validation.

Đối với lĩnh vực học máy, phương pháp k-fold cross validation được sử dụng khá phổ biến hơn cả. Một số hình ảnh minh họa của phương pháp này lấy từ scikit-learn.



Hình 2: Minh họa 5-fold (nguồn: sklearn)



Hình 3: Huấn luyện và kiểm thử mô hình với k-fold CV (nguồn: sklearn)