Logistic Regression Overfitting và Regularization Classification Metrics

VietAI Teaching Team

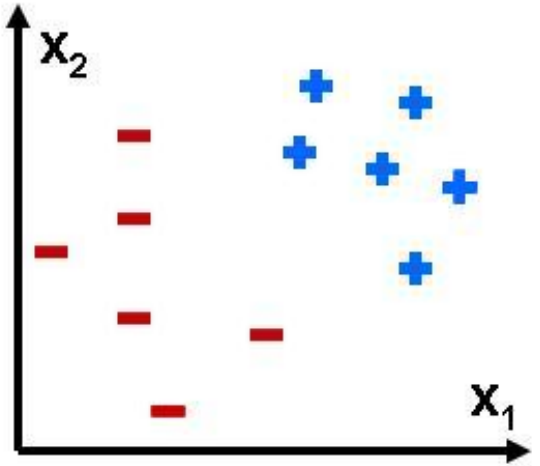
Logistic Regression

Logistic Regression

1. Bài toán Classification
2. Logistic Regression Model
   a) Hàm Sigmoid
   b) Cost Function
   c) Gradient Descent
3. Multiclass Classification

1 Bài toán Classification

- Trong bài toán Supervised Learning, khi tập giá trị dự đoán (Y) rời rạc, bài toán được gọi là ***Classification***
- Xét trường hợp đơn giản nhất khi $Y = \{0, 1\}$, bài toán được gọi là ***Binary Classification***.
- "0" được gọi là ***negative class***, "1" được gọi là ***positive class***.



2 Logistic Regression Model

- Bài toán classification được đưa về bài toán dự đoán xác suất của positive class:

$$P(y=1|x) = h_w(x) = g(w^\top x) = \frac{1}{1+\exp(-w^\top x)}$$

$$P(y=0|x) = 1 - P(y=1|x) = 1 - h_w(x)$$

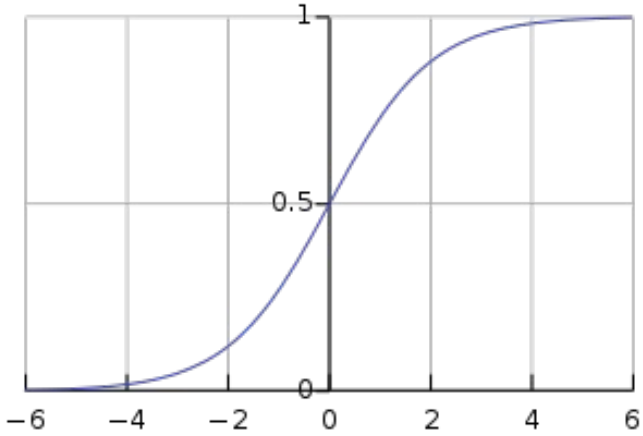
2 Hàm Sigmoid

- Hàm *g* được gọi là hàm ***sigmoid*** hay ***logistic***.

$$g(z) = \frac{1}{1 + e^{-z}}$$

- Đạo hàm của hàm sigmoid:

$$g'(z) = g(z)\,(1 - g(z))$$



2 Cost function

- Cost function được xây dựng để tối ưu hóa xác suất (mô hình dự đoán) của các lớp đúng.
- Cost function cần cực tiểu hóa trong mô hình Logistic Regression:

$$J(w) = -\frac{1}{m}\sum_{i=1}^{m}[y^{(i)}\log h_w(x^{(i)}) + (1-y^{(i)})\log(1-h_w(x^{(i)}))]$$

2 Gradient Descent

- Lặp đến khi hội tụ công thức:

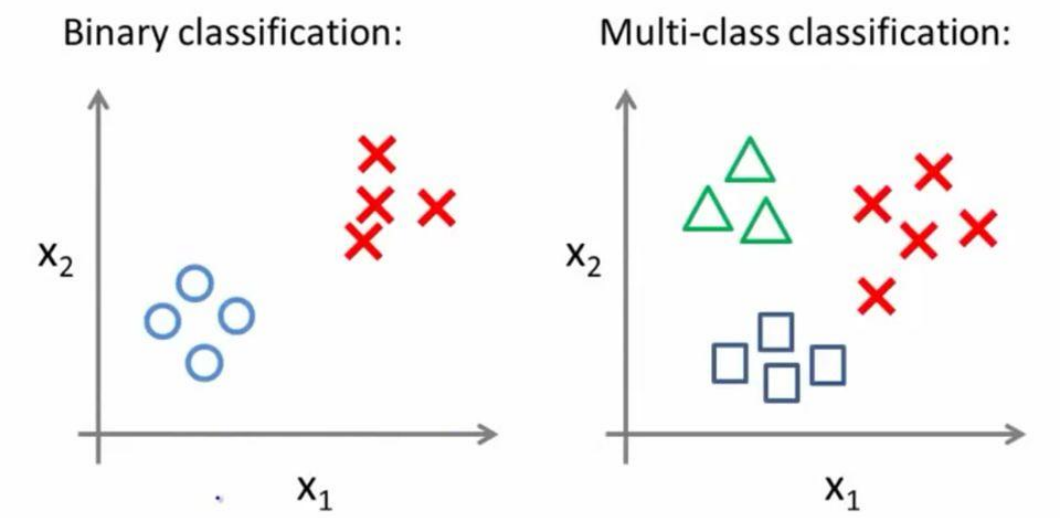
$$w_{new} = w_{old} - \alpha \nabla J(w_{old})$$

- Đạo hàm riêng của các phần tử trong w

$$\frac{\partial J(w)}{\partial w_j} = \frac{1}{m} \sum_{i=1}^{m} (h_w(x^{(i)}) - y^{(i)}) x_j^{(i)}$$

3 Multiclass Classification

- Trong bài Classification, nếu y có thể nhận nhiều giá trị rời rạc khác nhau (Y = {0, 1, 2 ...}) khi đó bài toán được gọi là Multiclass Classification.

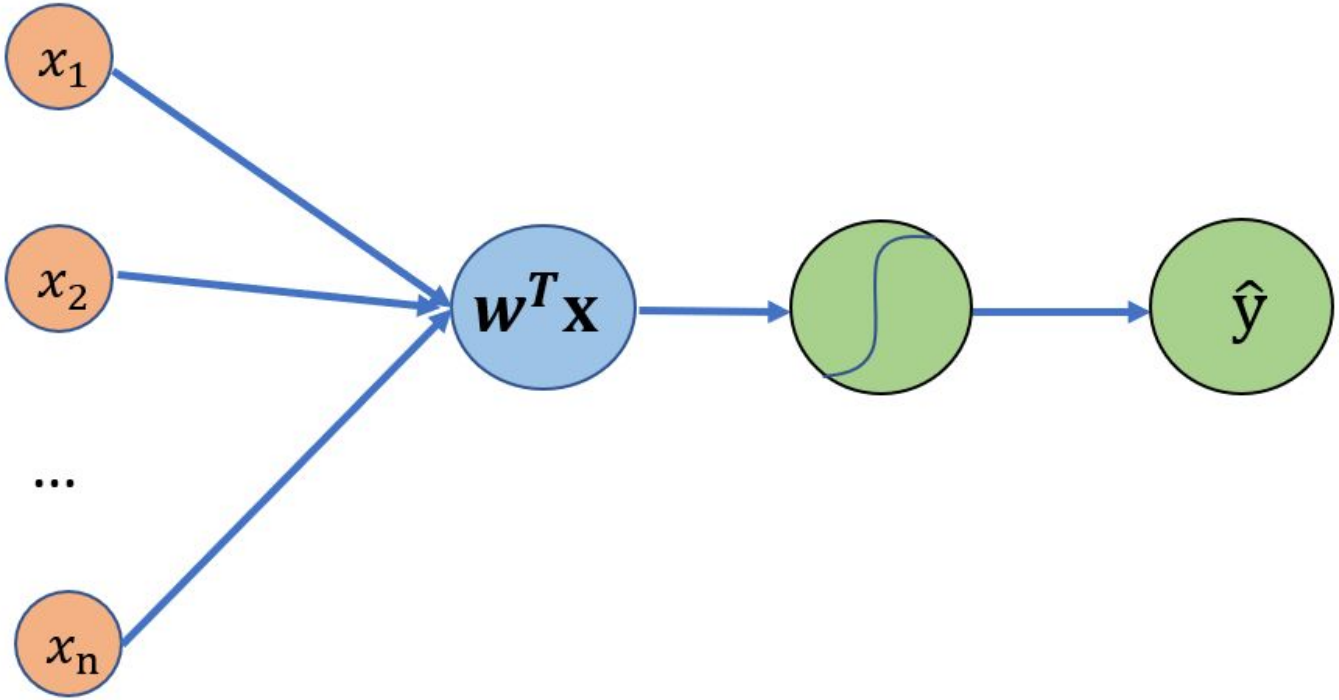


3 Multiclass Classification

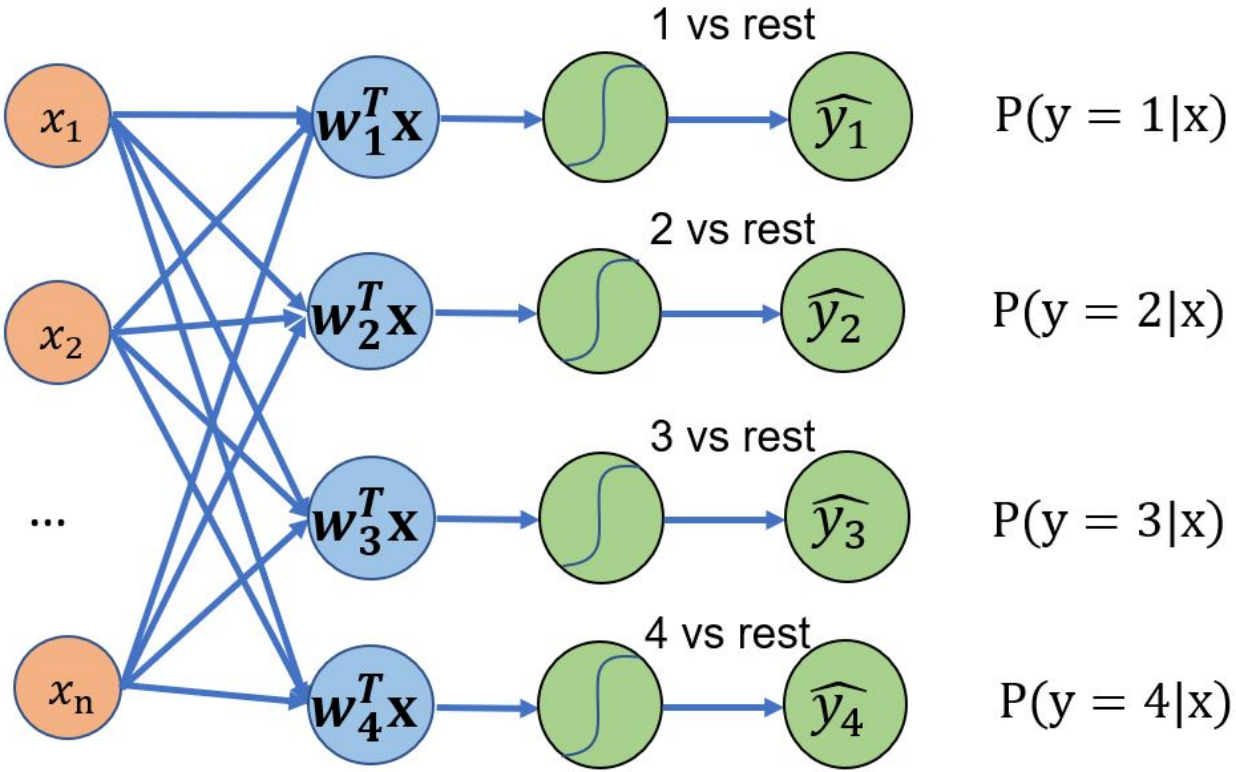
- Đối với bài toán Multiclass Classification, Logistic Regression có thể áp dụng với kĩ thuật One-vs-Rest (còn gọi là One-vs-All)
- Huấn luyện mô hình $h_w^{(k)}(x)$ cho mỗi class *k* để dự đoán xác suất *y* = *k*
- Quá trình dự đoán cho giá trị đầu vào *x* mới được thực hiện bằng cách tính xác suất cho tất các class và chọn class có xác suất cao nhất

$$\arg\max_k(h_w^{(k)}(x))$$

3 Multiclass Classification



3 Multiclass Classification



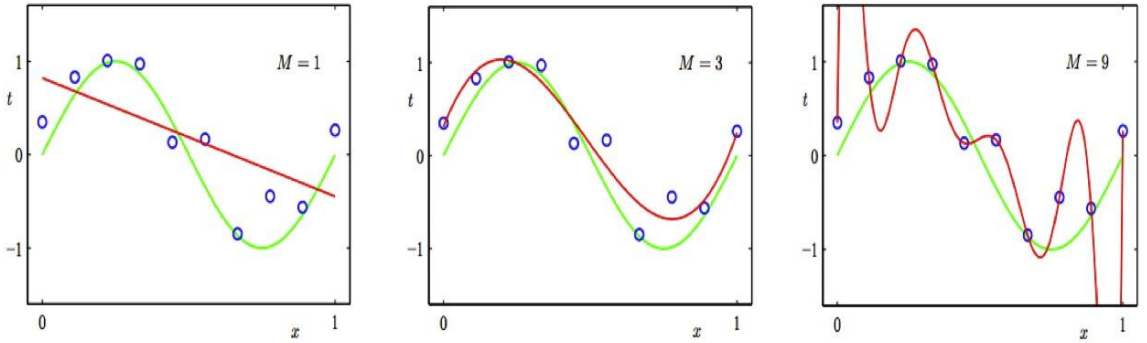
Overfitting và Regularization

Overfitting và Regularization

1. Overfitting
2. Regularization

1 Overfitting

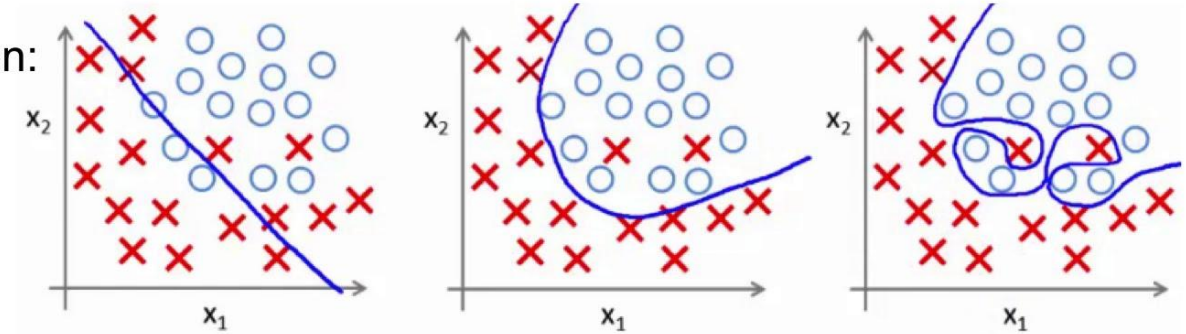
Regression:



predictor too inflexible:
cannot capture pattern

predictor too flexible:
fits noise in the data

Classification:



1 Overfitting

- Khi có quá nhiều features và parameters, mô hình có độ lỗi rất thấp trên tập huấn luyện nhưng có độ lỗi lớn trên thực tế.
- Cách khắc phục:
  - Giảm số lượng features và parameters
  - Áp dụng kĩ thuật Regularization
  - Thêm dữ liệu huấn luyện.

2 Regularization

- Linear Regression:

$$J(w) = \frac{1}{2m}\left[\sum_{i=1}^{m}\left(h_w(x^{(i)}) - y^{(i)}\right)^2 + \lambda\sum_{j=1}^{n} w_j^2\right]$$

- Logistic Regression:

$$J(w) = -\frac{1}{m}\sum_{i=1}^{m}\left(y^{(i)}\log h_w\left(x^{(i)}\right) + \left(1-y^{(i)}\right)\log\left(1-h_w\left(x^{(i)}\right)\right)\right) + \frac{\lambda}{2m}\sum_{j=1}^{n} w_j^2$$

Một số độ đo cho bài toán Classification

Overfitting và Regularization

1. Accuracy
2. Precision và Recall
3. F1-Score
4. Áp dụng cho nhiều lớp

1 Accuracy

- Accuracy là độ đo thường gặp nhất trong bài toán classification nó được tính dựa trên số mẫu dữ liệu được phân lớp đúng chia tổng số mẫu.
- Ví dụ: classifier thực hiện dự đoán 100 mẫu dữ liệu, trong đó có 75 mẫu dự đoán đúng lớp của nó, khi đó classifier này có accuracy là 75%.

2 Precision và Recall

		actual value: *p*	actual value: *n*	total
prediction outcome	*p'*	True Positive	False Positive	P'
	n'	False Negative	True Negative	N'
	total	P	N	

$$Precision = \frac{TP}{P'} = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{P} = \frac{TP}{TP + FN}$$

3 F1-score

- F1-score là độ đo "trung hòa" cả Precision lẫn Recall, được tính bằng công thức trung bình điều hòa (harmonic mean) của Precision và Recall.

$$F_1\text{-score} = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

4 Áp dụng cho nhiều lớp

- Cách kết hợp kết quả Precision, Recall, F1-Score cho nhiều lớp:
  - ***Micro-averaged***: Tính tổng các đại lượng TP, FP, TN, FN trên tất cả các lớp, sau đó mới tính các độ đo Precision, Recall, F1-Score dựa trên TP, FP, TN, FN thu được.
  - ***Macro-averaged***: Tính các độ đo Precision, Recall, F1-Score cho từng lớp rồi lấy kết quả trung bình
  - ***Weighted***: Tính các độ đo Precision, Recall, F1-Score cho từng lớp rồi tính trung bình theo trọng số cho trước (thường là dựa theo tỉ lệ của lớp tương ứng trong tập dữ liệu)

Tài liệu tham khảo

- UFLDL Tutorial - Logistic Regression - Stanford University
- Machine Learning - CS229 - Stanford University