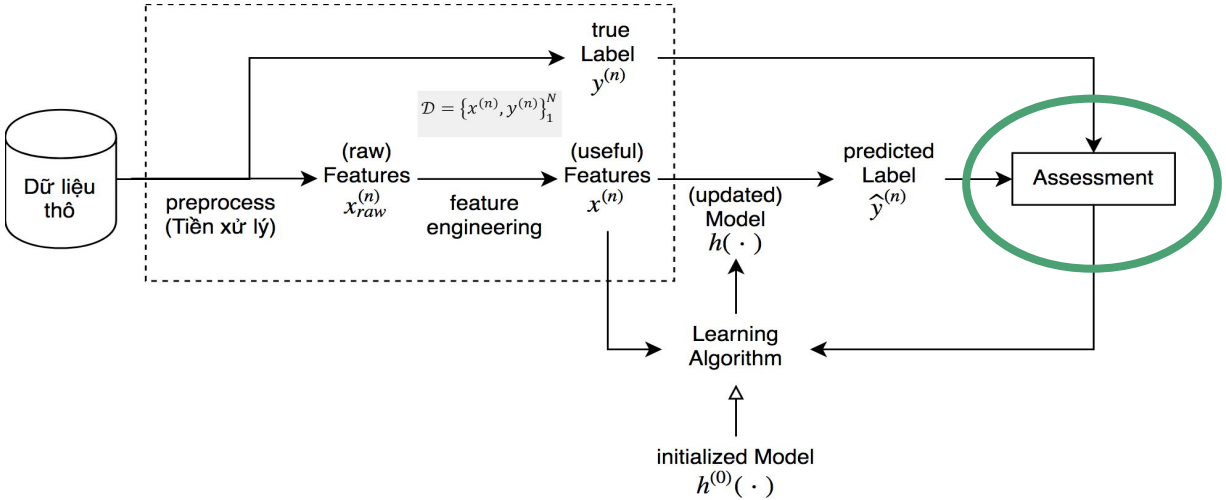
VietAI

Bài 5: Đánh giá chất lượng & Hiệu chỉnh Mô hình ML

Tuần 3A

Nội dung chính

1. Vấn đề Overfitting & Hiệu chỉnh mô hình - Model Regularization
2. Phương pháp Cross-Validation



1. Vấn đề Overfitting
& Hiệu chỉnh mô hình - Regularization

Overfitting

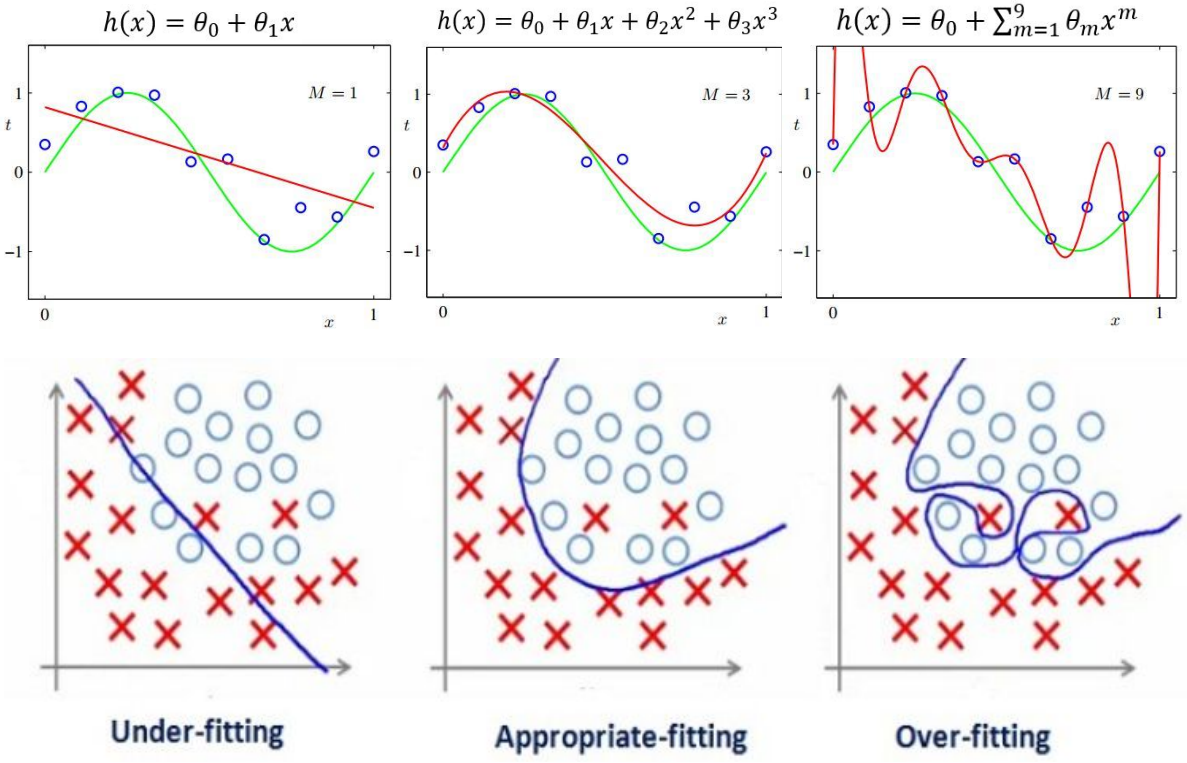
Định nghĩa

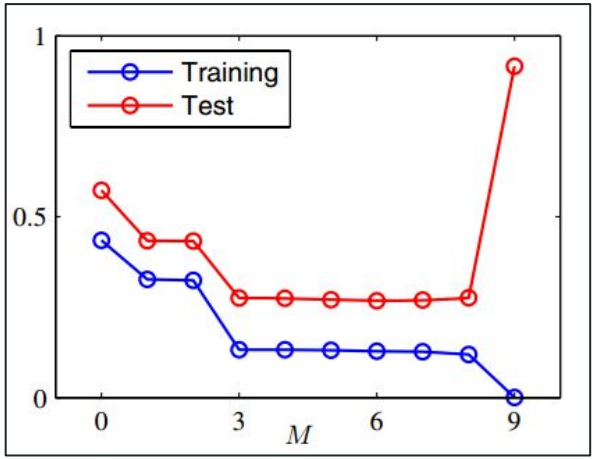
Overfitting là hiện tượng xảy ra khi một mô hình ML cố gắng fit tất cả các điểm dữ liệu thuộc ***training set*** - *tập huấn luyện*, khiến mô hình học được dù có độ lỗi thấp trên training set đó, nhưng lại có độ lỗi lớn trên những tập dữ liệu khác

- Overfitting là vấn đề chung mà *tất cả* mô hình ML đều phải giải quyết
- Khi mô hình ML học được có độ lỗi lớn trên cả training set và các tập dữ liệu khác, mô hình ML đó đang **underfitting**

Overfitting

Ví dụ





Với ví dụ như hình minh họa, mô hình đa thức bậc 9 bị overfit; mô hình bậc 0-2 bị underfit; mô hình bậc 3-8 là phù hợp.

(source: Bishop, 2006)

Overfitting

Một số cách khắc phục vấn đề Overfitting

Mô hình ML rất dễ bị overfit khi “N << p” - là khi số điểm dữ liệu training ít hơn nhiều so với độ phức tạp của mô hình. Do đó có những cách khắc phục sau:

- Thêm dữ liệu huấn luyện
  - Thu thập / crawl thêm dữ liệu
  - Biến đổi các điểm dữ liệu training set đã có vd. xoay / lật ngang ảnh
  - Dùng các mô hình sinh ảnh / text / giọng nói / ... để tạo thêm dữ liệu synthetic
- Giảm độ phức tạp của mô hình bằng cách
  - Giảm số lượng features; hoặc
  - Giữ nguyên mô hình, thêm **đại lượng Hiệu chỉnh - *Regularization term*** - vào Cost function

Regularization

Ví dụ: Hiệu chỉnh Mô hình Linear Regression

Thêm đại lượng **Hiệu chỉnh L2** dựa trên (squared) L2-norm của parameter $\boldsymbol{\theta}$

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

- Hệ số hiệu chỉnh λ là **hyperparameter** - "siêu tham số"
- Thuật toán Gradient Descent áp dụng khi có Regularization term như sau
  - `Lặp đến khi hội tụ:`

$$\theta_0 := \theta_0 - \alpha\frac{1}{m}\sum_{i=1}^{m}\left(h_\theta(x^{(i)}) - y^{(i)}\right)x_0^{(i)}$$

$$\theta_j := \theta_j - \alpha\left[\frac{1}{m}\sum_{i=1}^{m}\left(h_\theta(x^{(i)}) - y^{(i)}\right)x_j^{(i)} + \frac{\lambda}{m}\theta_j\right]$$

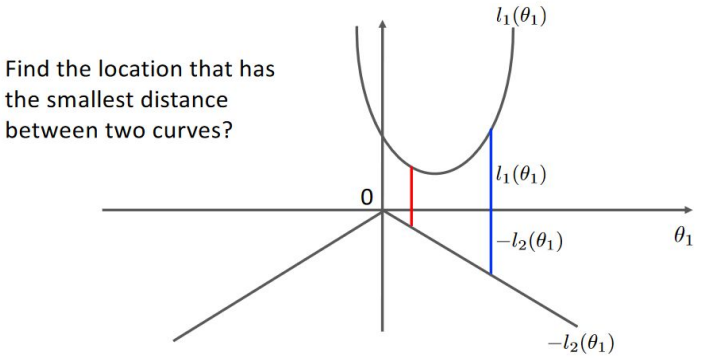
- Có thể sử dụng đại lượng **Hiệu chỉnh L1** $\lambda\sum_{j=1}^{n}|\theta_j|$

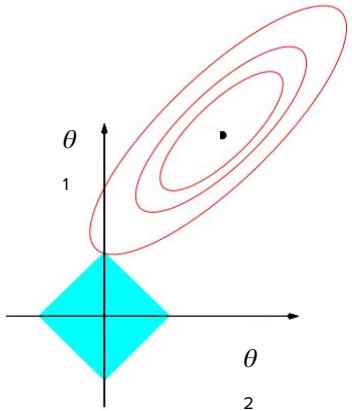
Regularization

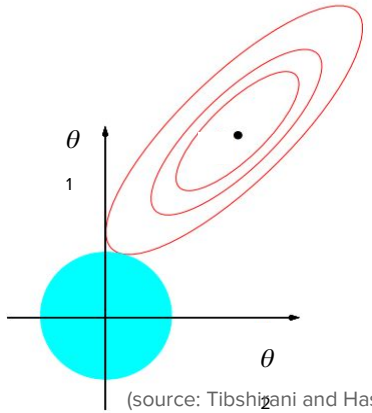
Nền tảng lý thuyết của Hiệu chỉnh L1/L2 dưới các góc nhìn (đọc thêm)

- Góc nhìn Toán tối ưu: *Đánh phạt / Giới hạn "độ lớn" của vector trọng số* $\boldsymbol{\theta}$

$$\min_{\boldsymbol{\theta}} \underbrace{\sum_{i=1}^{m}\left(y^{(i)} - \boldsymbol{\theta}^T x^{(i)}\right)^2}_{l_1(\boldsymbol{\theta})} + \underbrace{\lambda\|\boldsymbol{\theta}\|_1}_{l_2(\boldsymbol{\theta})} \quad \longleftarrow \quad \min_{\boldsymbol{\theta}} \sum_{i=1}^{m}\left(y^{(i)} - \boldsymbol{\theta}^T x^{(i)}\right)^2 \text{ với điều kiện } \|\boldsymbol{\theta}\|_1 \leq \lambda$$





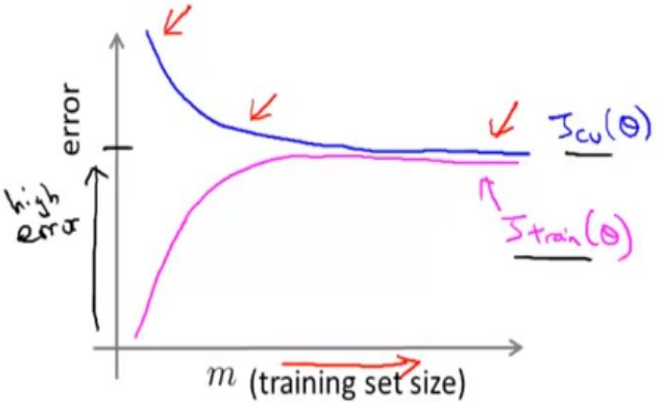


(source: Tibshirani and Hastie)

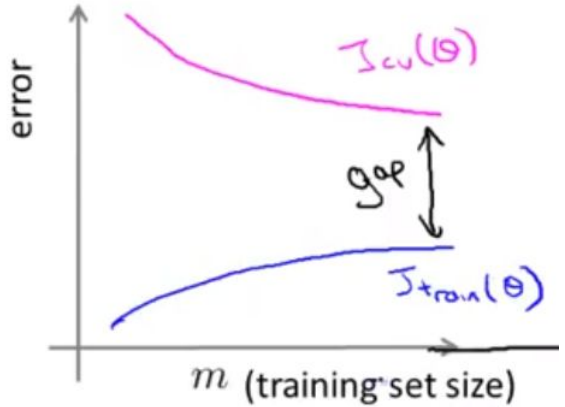
Minh Hoai Nguyen - Machine Learning Fall 18 – Stony Brook University 19

- Góc nhìn Mô hình xác suất: "*Prior - thông tin tiên nghiệm (trước khi có dữ liệu) - cho biết phân bố xác suất các giá trị của* $\boldsymbol{\theta}$ *có mật độ tập trung quanh giá trị* **0**"
  - $\boldsymbol{\theta}$~Laplace($\mathbf{0}$, $\lambda^{-1}\mathbf{I}$) ➔ Hiệu chỉnh L1; $\boldsymbol{\theta}$~$\mathcal{N}$($\mathbf{0}$, $\lambda^{-1}\mathbf{I}$) ➔ Hiệu chỉnh L2

Có cần thu thập thêm dữ liệu?



Mô hình hiện tại có dấu hiệu *underfitting*, thu thập thêm dữ liệu huấn luyện *không* cải thiện được thêm chất lượng mô hình



Mô hình hiện tại có dấu hiệu *overfitting*, thu thập thêm dữ liệu huấn luyện *có thể* cải thiện được thêm chất lượng mô hình

(source: Andrew Ng)

VietAI

2. Cross-Validation

Cross-Validation

Giới thiệu chung

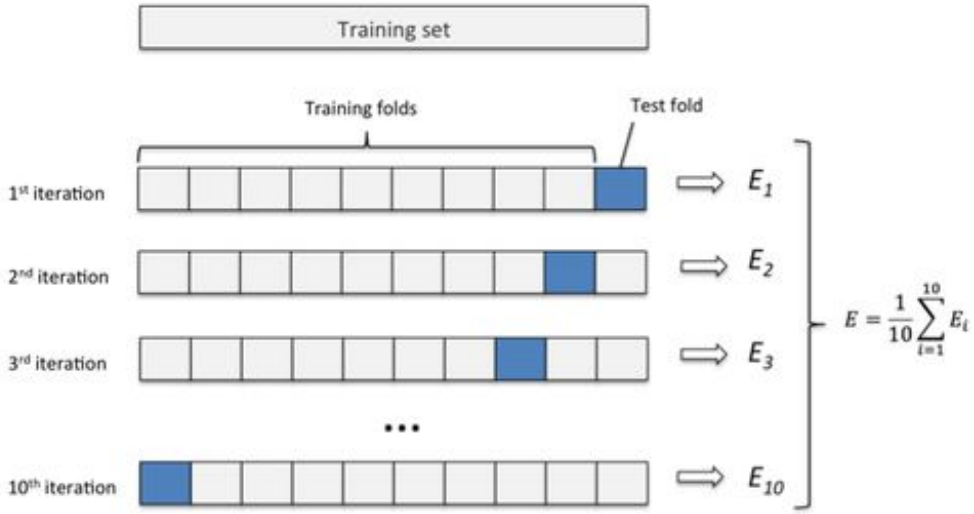
- Cross-Validation - CV - là phương pháp đánh giá chất lượng mô hình ML (*đã* được huấn luyện với 1 training set) trên 1 hoặc nhiều tập dữ liệu **chưa từng gặp** trong quá trình huấn luyện mô hình. Những tập dữ liệu không thuộc training set này gọi là *out-of-sample set(s)*
  - **Metrics** sử dụng để đánh giá chất lượng có thể (1) chính là cost function vd. MSE; hoặc (2) một số chỉ số khác phù hợp hơn (sẽ giới thiệu trong Lecture 6)
- CV giúp ta
  - ước lượng **generalization error** - độ lỗi / độ chính xác trên *tất cả* các tập dữ liệu chưa từng gặp i.e. *khả năng tổng quát hóa* của mô hình. Các out-of-sample set(s) dùng để ước lượng generalization error bằng CV được gọi là **Test set**(s)
  - xác định giá trị tối ưu cho các **hyperparameter** - các "siêu tham số" của quá trình huấn luyện mô hình (vd. λ, α) mà ta không thể dùng Thuật toán học như Gradient Descent để tìm giá trị tối ưu. Các out-of-sample set(s) dùng để xác định giá trị tối ưu cho các hyperparameter bằng CV được gọi là **Validation set**(s) hoặc **Dev set**(s)

Cross-Validation

Biến thể: K-Fold CV

K-Fold Cross Validation được thực hiện bằng cách chia tập dữ liệu thành K phần bằng nhau. Thực hiện K lần việc huấn luyện mô hình, mỗi lần sử dụng K - 1 phần để huấn luyện và 1 phần để evaluate.

<u>*Câu hỏi*</u>: Làm thế nào để đồng thời xác định các hyperparameter tối ưu, và ước lượng generalization error chỉ bằng K-Fold CV?



Vd. Dùng 10-fold CV để ước lượng Generalization error.

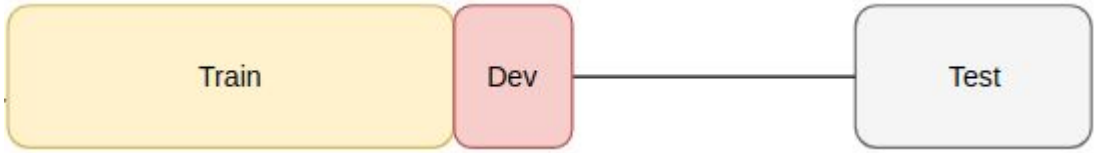
Cross-Validation

Biến thể: Hold-out CV

Về căn bản là 2-fold CV. Cần đảm bảo train / dev / test set có đủ tính đại diện cho bài toán cần giải quyết

Để đồng thời xác định các hyperparameter tối ưu, và ước lượng generalization error bằng chỉ bằng Hold-out CV, có thể thực hiện như sau:

1. Chia Training set thành 1 (sub) Train set và 1 Dev set
2. Huấn luyện mô hình trên (sub) Train set với các giá trị hyperparameter khác nhau, và cross-validate trên Dev set để xác định giá trị tối ưu
3. *(tùy chọn)* Huấn luyện lại mô hình với hyperparameter tối ưu trên Training set gốc
4. Tính độ lỗi trên Test set



(image credit: Kevin Zakka)

Tài liệu tham khảo

1. *Lecture 10 - Advice for Applying Machine Learning* - **Machine Learning** (Coursera) by Andrew Ng
2. *Chương 1.1*, **Pattern Recognition and Machine Learning** (Book) by Christopher Bishop, 2006
3. *Chương 3.4*; 7.5 **The Elements of Statistical Learning** (Book) by Jerome H. Friedman, Robert Tibshirani, and Trevor Hastie, 2001