Chương 4.
Đánh giá giải pháp MH

Ts. Nguyễn An Tế

Khoa CNTT kinh doanh – ĐH Kinh tế TPHCM

tena@ueh.edu.vn

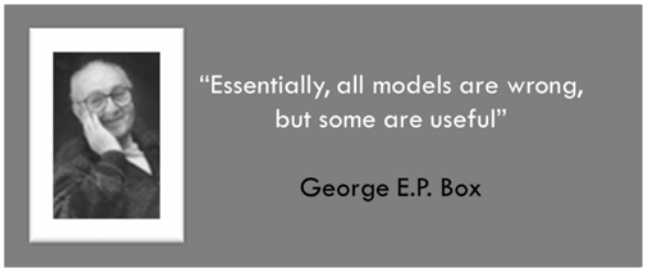
2025

Nội dung

1. **Đánh giá giải pháp máy học**
2. **Đánh giá mô hình phân lớp**
3. **Đánh giá mô hình hồi quy**
4. **Đánh giá mô hình gom cụm**

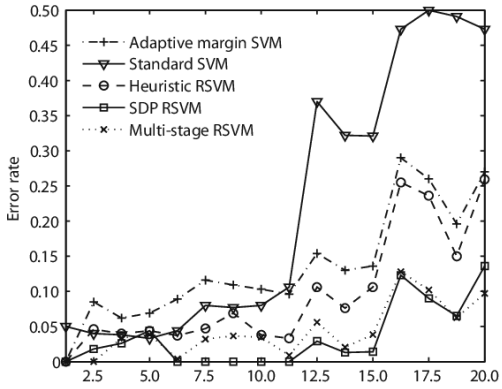
1. Đánh giá giải pháp máy học

- **Lựa chọn giải pháp**
  - giải pháp "chấp nhận được" hay giải pháp tối ưu ?



1. Đánh giá giải pháp máy học

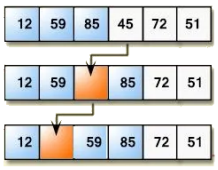
- **So sánh, chọn lựa các giải pháp (đa dạng)**
  - khác nhóm: *parametric*, *nonparametric*
  - cùng nhóm: *parameters*, *hyperparameters*



1. Đánh giá giải pháp máy học

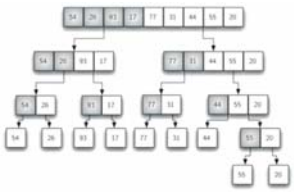
- **Đánh giá, so sánh các thuật toán sắp xếp**

Insertion sort



$O(n^2)$

Merge sort



$O(n \log(n))$

Machine Learning Algorithms

1. Đánh giá giải pháp máy học

- **Chất lượng của mô hình: nhiều tiêu chí** [Turney, 2000]
  - sự chính xác
  - sự hiệu quả: thời gian huấn luyện, bộ nhớ, …
  - sự nhạy cảm đối với dữ liệu nhiễu
  - khả năng diễn dịch, giải thích
  - . . .

1. Đánh giá giải pháp máy học

- **Đánh giá giải pháp máy học: thực nghiệm**
  - tỷ lệ % sai số
  - mức độ tin cậy (áp dụng thực tế)

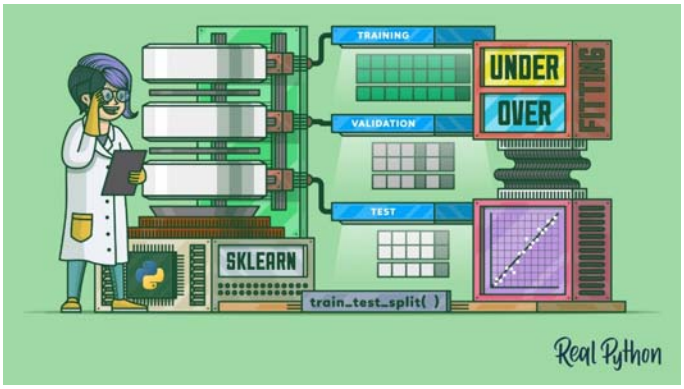
1. Đánh giá giải pháp máy học

- **Các loại chỉ số**
  - định lượng (*quantitative quality indicators*): thống kê
  - đồ họa/biểu đồ (*graphical indicators*): Confusion matrix, ROC curve, LIFT curve, . . .

- **Ứng dụng cho các loại bài toán: phân lớp, hồi quy, gom cụm**
  - đặc trưng riêng của từng loại bài toán
  - dùng chung: có thể cần phải "cải biên"

1. Đánh giá giải pháp máy học

- **Dữ liệu (DL) thực nghiệm**
  - *training set*: huấn luyện, xây dựng các mô hình ứng viên
  - *validation* / *development set*: kiểm định, chọn lựa mô hình
  - *test* / *publication set*: đánh giá mô hình (đã chọn)



1. Đánh giá giải pháp máy học

- **Dữ liệu (DL) thực nghiệm: vai trò của validation set**
  - kiểm định khả năng tổng quát hóa của mô hình
    → chọn mô hình tốt nhất trong số các mô hình ứng viên
  - tinh chỉnh cấu trúc hay *hyperparameters* (≠ *parameters*)

→ xem như validation set trở thành 1 thành phần của training set (tham gia xây dựng mô hình hoàn chỉnh cuối cùng)

1. Đánh giá giải pháp máy học

- **Dữ liệu (DL) thực nghiệm: vai trò của validation set**

  VD: Xây dựng mô hình hồi quy đa thức bậc k ∈ [2, d]

  B1. Sử dụng training set để xác định các hệ số cho mỗi bậc k

  B2. Sử dụng validation set để chọn ra mô hình bậc k tốt nhất (sai số thấp nhất trên validation set)

1. Đánh giá giải pháp máy học

- **Dữ liệu (DL) thực nghiệm: vai trò của validation set**

  VD: Xây dựng mô hình mạng nơ-ron đa tầng

  B1. Sử dụng training set để xác định vectơ trọng số w

  B2. Sử dụng validation set để xác định số hidden layers, learning rate, …

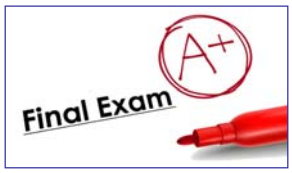
  VD: Xây dựng mô hình phân lớp k-NN

  B1. Sử dụng training set như lookup table

  B2. Sử dụng validation set để chọn công thức khoảng cách và số lượng láng giềng k

1. Đánh giá giải pháp máy học

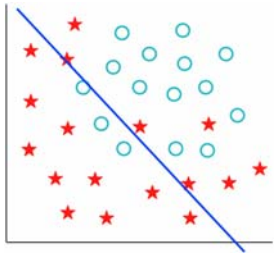
- **Dữ liệu (DL) thực nghiệm**



1. Đánh giá giải pháp máy học

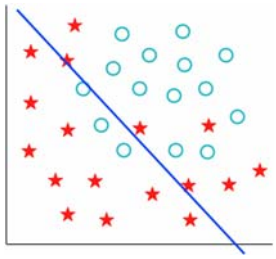
- **Hiện tượng** *underfitting*
  - mô hình chưa khớp với DL huấn luyện: chưa đủ độ phức tạp cần thiết để có thể "bao quát" được tập DL

VD: Xây dựng mô hình hồi quy tuyến tính trên một mẫu dữ liệu đa thức bậc 3



1. Đánh giá giải pháp máy học

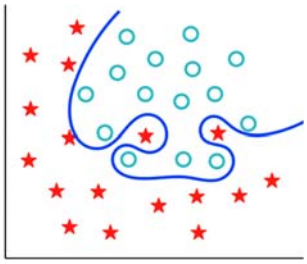
- **Hiện tượng** *underfitting*
  - train error và test error đều cao
  - tồn tại những điểm DL không thể phân lớp
  - khó dự đoán chính xác với DL mới



1. Đánh giá giải pháp máy học

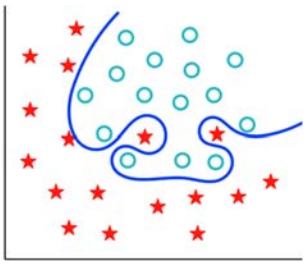
- **Hiện tượng** *overfitting*
  - mô hình quá phức tạp khi mô phỏng DL huấn luyện

VD: Xây dựng mô hình hồi quy đa thức bậc 6 cho mẫu dữ liệu mang bản chất là đa thức bậc 3 nhưng có chứa nhiễu



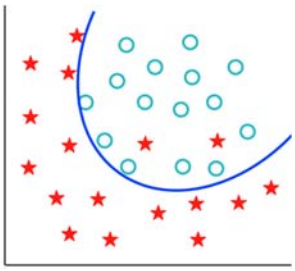
1. Đánh giá giải pháp máy học

- **Hiện tượng** *overfitting*
  - mô hình quá khớp với DL huấn luyện: dễ dự đoán nhầm nhiễu
  - train error thấp trong khi test error cao
  - lượng DL huấn luyện quá nhỏ, không đủ để thể hiện mô hình
  - khó phù hợp với tính tổng quát của dữ liệu mới



1. Đánh giá giải pháp máy học

- **Hiện tượng** *good fitting*
  - kết quả "hợp lý" với cả tập DL huấn luyện và DL mới
  - train error và test error đều thấp
  - sự hiệu quả
  - sự đơn giản



1. Đánh giá giải pháp máy học

- **Phương châm thực hiện:** *triple trade-off* [Dietterich, 2003]
  - độ phức tạp mô hình
  - quy mô của DL huấn luyện
  - khả năng tổng quát hóa (sai số) đối với những quan sát mới

1. Đánh giá giải pháp máy học

- **Không có "best learning algorithm" theo nghĩa tuyệt đối**
  - phụ thuộc lãnh vực
  - phụ thuộc datasets

Nội dung

1. Đánh giá giải pháp máy học
2. **Đánh giá mô hình phân lớp**
3. Đánh giá mô hình hồi quy
4. Đánh giá mô hình gom cụm

2.1 Các tiêu chí đánh giá

- **Ma trận nhầm lẫn** (*Confusion Matrix*)
  - ma trận vuông, kích thước mỗi chiều = số lớp
  - m_{ij} : số lượng items thuộc C_i nhưng bị dự đoán (SAI) thuộc C_j (hoặc chuyển vị)
  - các giá trị m_{ij} có thể được chuẩn hóa ([0, 1])

Dự đoán

CLASSES	A	B	C	Row totals
A	5	2	3	10
B	2	6	0	8
C	3	2	2	7
Column Totals	10	10	5	25

Thực tế

 # dự đoán đúng

 # items

2.1 Các tiêu chí đánh giá

- **Ma trận nhầm lẫn** (*Confusion Matrix*)**: nhị phân** { +, – }
  - phân loại sai lầm (*Error Types*) → so với *Hypothesis Testing* ?

	Dự đoán	
Thực tế	*Positive*	*Negative*
Positive	Kết luận đúng **True Positive** (TP)	Sai lầm loại II **False Negative** (FN)
Negative	Sai lầm loại I **False Positive** (FP)	Kết luận đúng **True Negative** (TN)

silence

noise

2.1 Các tiêu chí đánh giá

- **Ma trận nhầm lẫn** (*Confusion Matrix*)**: nhị phân** { +, – }
  - phân loại sai lầm (*Error Types*)

	Dự đoán	
Thực tế	*Positive*	*Negative*
Positive		
Negative		

2.1 Các tiêu chí đánh giá

- **Ma trận nhầm lẫn** (*Confusion Matrix*)**: nhị phân** { +, – }
  - chuẩn hóa (*Normalized Confusion Matrix*) → R: rate

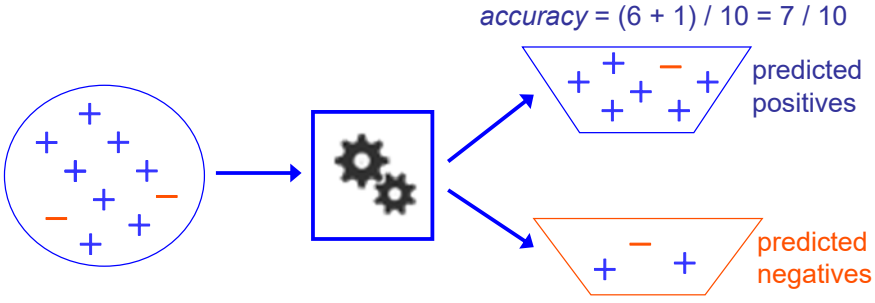
	Dự đoán	
Thực tế	*Positive*	*Negative*
Positive	**TPR** = TP / (TP + FN) *recall, sensitivity*	**FNR** = FN / (TP + FN) *miss detection rate*
Negative	**FPR** = FP / (FP + TN) *fall-out*	**TNR** = TN / (FP + TN) *specificity*

False Alarm Rate

2.1 Các tiêu chí đánh giá

- **Độ đo** *accuracy*
  - có phải toàn bộ quan sát đều được phân lớp đúng ?
  - tỉ lệ % các quan sát được phân lớp đúng

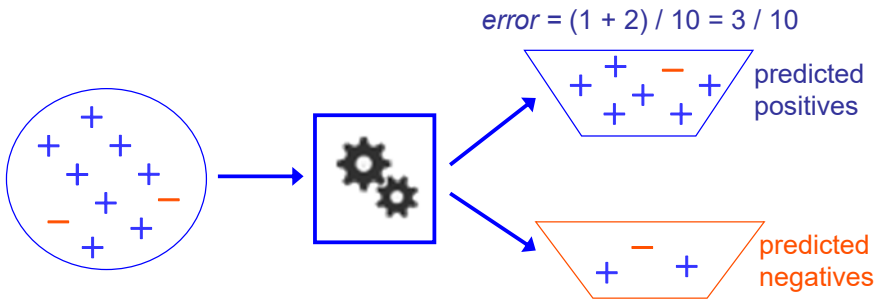
$$accuracy = \frac{TP + TN}{n}$$



2.1 Các tiêu chí đánh giá

- **Độ đo *error***
  - tỉ lệ % các quan sát bị phân lớp sai

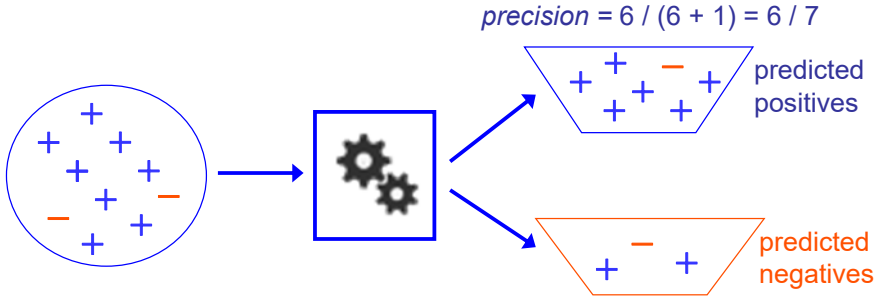
$$error = \frac{FP + FN}{n} = 1 - accuracy$$



2.1 Các tiêu chí đánh giá

- **Độ đo *precision*: tập DL bất đối xứng (*imbalanced* / *skew data*)**
  - có phải toàn bộ *predicted positives* đều có bản chất + ?
  - tỉ lệ % *predicted positives* có bản chất +

$$precision = \frac{TP}{TP + FP} \neq \frac{TP + TN}{n} \quad (error\ rate)$$

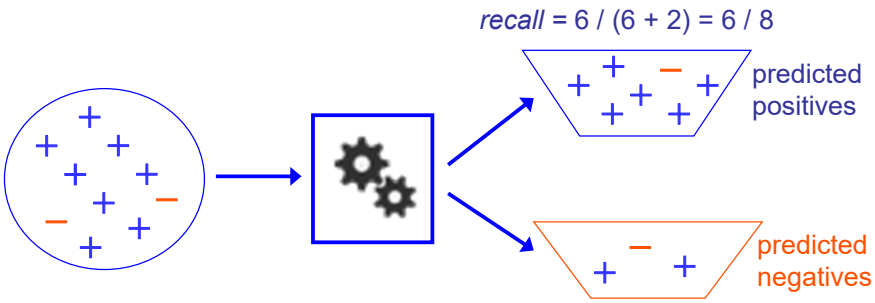


2.1 Các tiêu chí đánh giá

❑ **Độ đo *recall*: tập DL bất đối xứng**

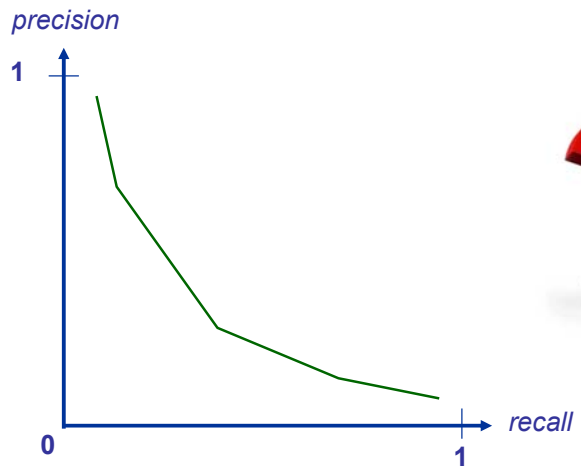
- có phải toàn bộ *positives* (bản chất) đều được nhận diện ?
- tỉ lệ % *positives* đã được nhận diện → độ "nhạy" (*sensitivity*)

$$recall = TPR = \frac{TP}{TP + FN}$$



2.1 Các tiêu chí đánh giá

❑ **Đồ thị tương quan giữa precision và recall**



2.1 Các tiêu chí đánh giá

- **Độ đo F (*F-measure*): điều hòa giữa precision và recall (*weighted harmonic mean*)**

$$F = \frac{precision.recall}{(1-\alpha).precision + \alpha.recall}$$

$$\alpha = 0 \;:\quad F = recall$$

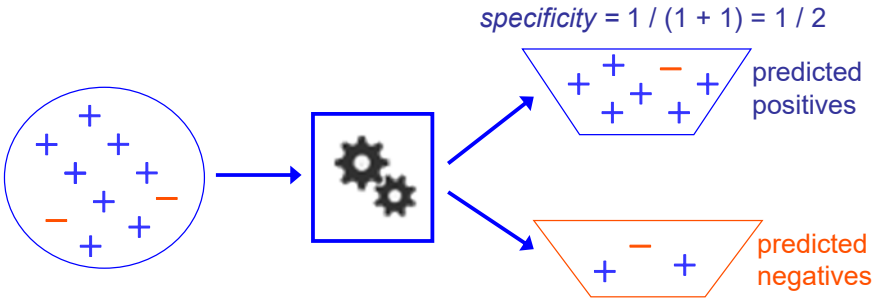
$$\alpha = 1 \;:\quad F = precision$$

$$\alpha = \frac{1}{2} \;:\quad F = 2.\frac{precision.recall}{precision + recall}$$

2.1 Các tiêu chí đánh giá

- **Độ đo *specificity*: tập DL bất đối xứng**
  - có phải toàn bộ *negatives* (bản chất) đều được nhận diện ?
  - tỉ lệ % *negatives* đã được nhận diện → "đặc trưng"

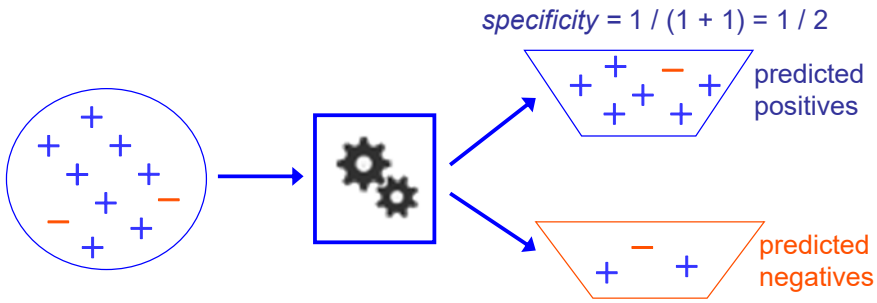
$$specificity = TNR = \frac{TN}{TN + FP}$$



2.1 Các tiêu chí đánh giá

- **Độ đo *fall-out*: tập DL bất đối xứng**
  - tỉ lệ % *negatives* không được nhận diện

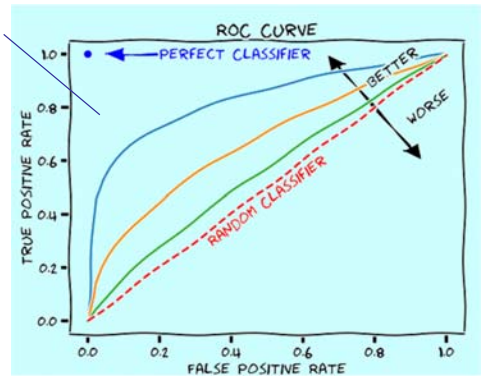
$$fallout = FPR = \frac{FP}{FP + TN} = 1 - specificity$$



2.1 Các tiêu chí đánh giá

- **Đường cong ROC** (*Receiver Operating Characteristic Curve*)
  - hiệu quả phân lớp nhị phân
  - ngưỡng quyết định: $P(y = 1 \mid x) \geq \beta$ (tách bạch tín hiệu-nhiễu)

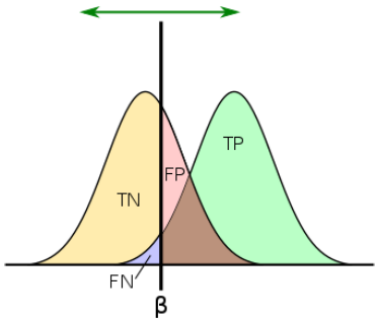
% không bỏ sót CAO
trong khi
% báo động nhầm THẤP

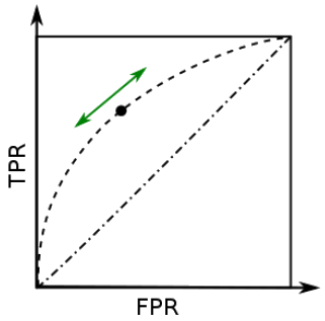


2.1 Các tiêu chí đánh giá

❑ **Đường cong ROC** (*Receiver Operating Characteristic Curve*)

- tinh chỉnh mô hình → { (TPR_i, FPR_i) }

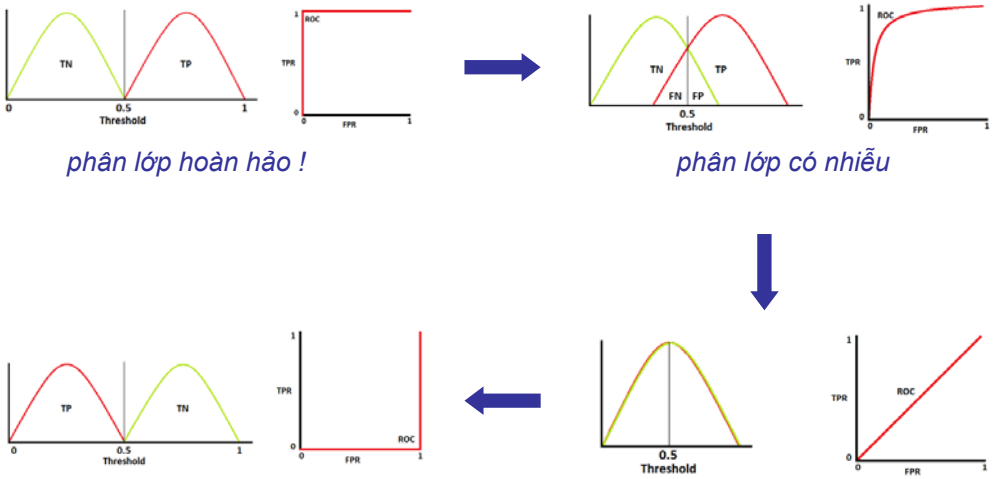




[*https://devopedia.org/roc-curve*]

2.1 Các tiêu chí đánh giá

❑ **Đường cong ROC** (*Receiver Operating Characteristic Curve*)



phân lớp hoàn hảo !

phân lớp có nhiễu

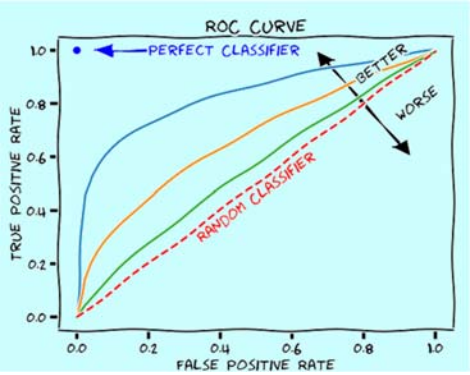
phân lớp đối nghịch

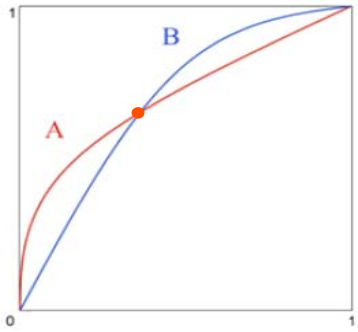
không tách bạch 2 lớp

[*https://www.i2tutorials.com/*]

2.1 Các tiêu chí đánh giá

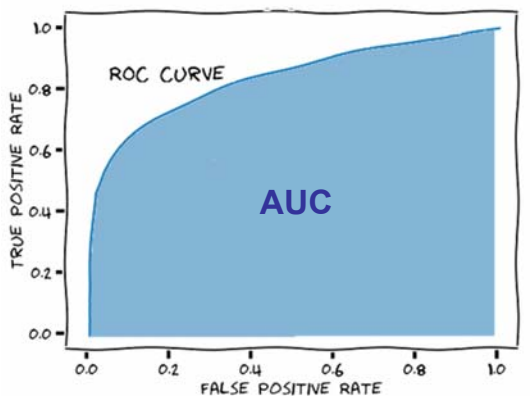
- **Đường cong ROC** (*Receiver Operating Characteristic Curve*)
  - so sánh 2 thuật giải A và B có ROC cắt nhau ?





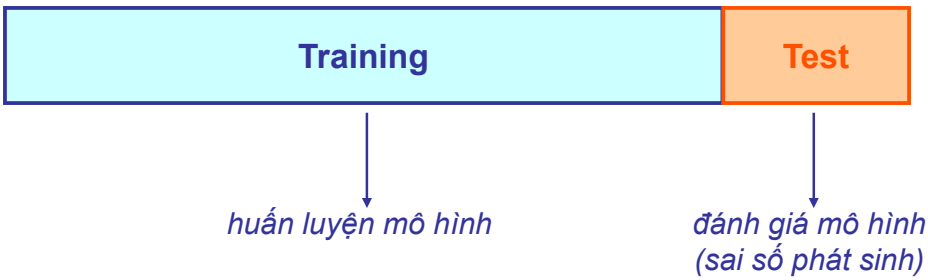
2.1 Các tiêu chí đánh giá

- **Miền AUC** (*Area Under the* [*ROC*] *Curve*)
  - diện tích ([0, 1]) bên dưới đường cong ROC



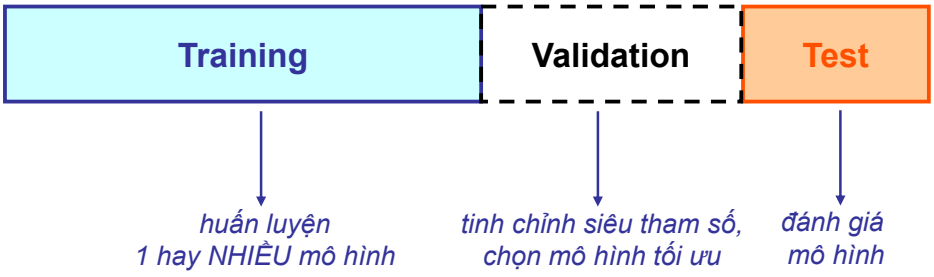
2.2 Cross-validation

- **Phương pháp** *Hold-out Cross-validation*
  - phân chia ngẫu nhiên dataset thành *training set* và *test set* (*hold-out set*) theo tỷ lệ xác đinh (training set >> test set 7:3)
  - đánh giá 1 mô hình, không có nhiều tham số cần tinh chỉnh



2.2 Cross-validation

- **Phương pháp** *Hold-out Cross-validation*
  - tạo *validation set* từ 1 phần của training set ban đầu
  - validation set: tinh chỉnh các siêu tham số, chọn mô hình tối ưu
  - validation set giúp làm giảm nhẹ hiện tượng overfitting



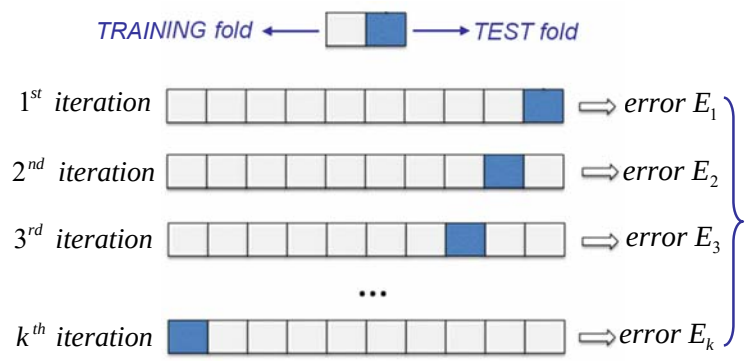
2.2 Cross-validation

- **Phương pháp** *Hold-out Cross-validation*
  - dataset nhỏ dễ bị tác động mạnh bởi những quan sát "đặc biệt" (noise, outliers)
  - sai số phát sinh từ test set phụ thuộc nhiều vào tỷ lệ "phân bố" các phần tử của các lớp trong training set và test set (VD: training set, hay test set, chứa quá ít phần tử + hay –)
  - không hiệu quả khi mô hình có nhiều tham số cần tinh chỉnh
  - không hiệu quả khi so sánh nhiều mô hình

  → *Repeated hold-out*: tính sai số trung bình trên k lần thực hiện
  Hold-out Cross-validation: tỷ lệ trùng lắp dữ liệu giữa những lần thực hiện ?

2.2 Cross-validation

- **Phương pháp** *k-Fold Cross-validation* [Russell+, 19.4]
  - chia dataset thành k phần (*fold*) bằng nhau
  - gia tăng độ tin cậy về chất lượng mô hình



$$E = \frac{1}{k}\sum_{i=1}^{k} E_i$$

  - k = N : *Leave-One-Out Cross-Validtion* (*LOOCV*)

2.2 Cross-validation

- **Phương pháp** *Regularization* [Breiman, 1998]
  - "trừng phạt" độ phức tạp mô hình: *augmented error function*

    sai số ε' = sai số trên DL ε + (λ * độ phức tạp)

    λ : trọng số ≥ 0 (hyperparameter)

2.3 Grid Search

- **Tham số** (*parameter*)
  - giá trị được xác định hoàn toàn dựa trên tập dữ liệu
  - giá trị được xác định một cách tự động, phù hợp với mô hình hay thuật toán máy học

  → *parametric* (fixed number), *nonparametric* (variable number)

  VD:
  - các trọng số trong mạng nơ-ron
  - các vectơ hỗ trợ trong SVM
  - các hệ số trong hồi quy tuyến tính hay hồi quy đa thức

2.3 Grid Search

- **Siêu tham số** (*hyperparameter*)
  - vai trò đối với 1 lớp các thuật toán hay mô hình
  - giá trị có thể được xác định không dựa trên chính tập dữ liệu
  - giá trị có thể được xác định một cách thủ công hoặc tự động

  VD:
  - số lượng láng giềng trong k-NN
  - hệ số học (*learning rate*) trong Gradient Descent
  - C trong Soft Margin SVM
  - số lượng layer(s) trong mạng nơ-ron

2.3 Grid Search

- **Phương pháp** *Grid Search*
  - mô hình có nhiều hyperparameters
  - hyperparameter(s): số thực (giá trị liên tục)
  - chạy thử nghiệm trên tổ hợp các giá trị của hyperparameters (tích Đề-các giữa các tập giá trị)

2.3 Grid Search

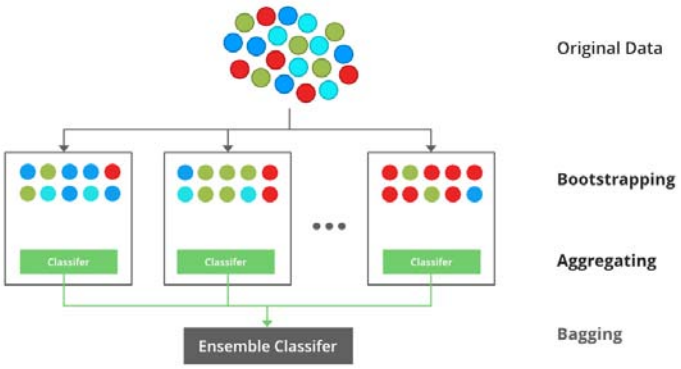
- **Phương pháp** *Random Search*
  - chọn ngẫu nhiên những giá trị của hyperparameter(s)
    → hiệu quả khi hyperparameter(s) lấy giá trị liên tục
- **Phương pháp** *Hand-tuning*
  - chọn lựa giá trị theo kinh nghiệm
  - chọn lựa giá trị theo thông lệ
- . . .

2.4 Bagging và Boosting

- **Phương pháp học kết hợp** (*Ensemble Learning*): **phối hợp nhiều mô hình khác nhau**
  - *homogeneous ensemble*: các mô hình được xây dựng từ cùng 1 thuật toán nhưng dựa trên nhiều tập dữ liệu khác nhau
    - bagging, boosting, …
  - *heterogeneous ensemble*: các mô hình được xây dựng từ nhiều thuật toán khác nhau
    - weighted, cascade, switching, stacking, . . .

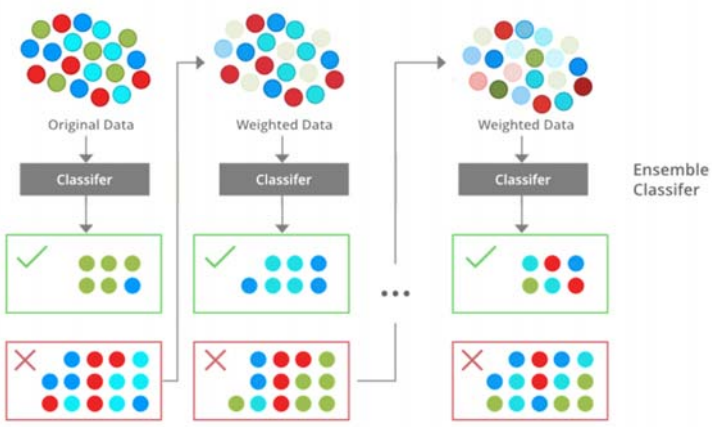
2.4 Bagging và Boosting

- **Phương pháp *Bagging* (song song)**
  - kỹ thuật boostrap samples
  - thêm nhiễu



2.4 Bagging và Boosting

- **Phương pháp *Boosting* (tuần tự)**
  - đánh lại các trọng số cho các quan sát sau mỗi lần huấn luyện
  - tập trung vào những quan sát đặc biệt, thường bị học sai



2.5 Hypothesis testing

- **Áp dụng kiểm định giả thuyết để đánh giá giải pháp MH**
  - classification: error rates, …
  - regression: squared errors, …
  - unsupervised learning: log likelihoods, …
  - reinforcement learning: expected reward, …

2.5 Hypothesis testing

- **Kiểm định nhị thức** (*Binomial Test*)

  Training set T, validation set V với |V| = N

  Ước lượng xác suất phạm sai lầm khi phân lớp p ?

  $\forall v_t \in V$: $x^t = 1$ nếu v_t bị phân lớp sai; ngược lại: $x^t = 0$

  Biến ngẫu nhiên = tổng sai số (*independent and identically distributed* – i.i.d):

$$X = \sum_{t=1}^{N} x^t$$

  Phân phối nhị thức: $P(X = j) = \binom{N}{j} p^j (1-p)^{N-j}$

2.5 Hypothesis testing

- **Kiểm định nhị thức** (*Binomial Test*)

  Các giả thuyết kiểm định:

$$\begin{cases} H_0: & p \le p_0 \\ H_1: & p > p_0 \end{cases}$$

  Mức độ tin cậy (*confidence level*): (1 - α)

  Miền bác bỏ (*reject region*) của kiểm định bên phải: $[z_\alpha, +\infty)$

2.5 Hypothesis testing

- **Kiểm định nhị thức** (*Binomial Test*)

  Ước lượng điểm: $\bar{p} = \frac{X}{N}$

  Theo định lý giới hạn trung tâm (*Central Limit Theorem*),

  với N đủ lớn: $\frac{X}{N} \sim N(\mu = p_0, \sigma^2 = p_0(1-p_0)/N)$

  $\Rightarrow$ $\frac{X/N - p_0}{\sqrt{p_0(1-p_0)/N}} \sim Z$

  Điều kiện bác bỏ: $\frac{\sqrt{N}}{\sigma}(\bar{p} - p_0) > z_\alpha$ (trị tới hạn – *critical value*)

2.5 Hypothesis testing

- **Kiểm định t-Test với k-fold cross-validation**

  Training set T_i, validation set V_i với i=1,…, k

  $\forall v_t \in V_i$: $x_i^t = 1$ nếu v_t bị phân lớp sai; ngược lại: $x_i^t = 0$

  Xác suất phạm sai lầm trên V_i: $p_i = \frac{1}{N}\sum_{t=1}^{N} x_i^t$

  Ước lượng điểm: $\bar{p} = \frac{1}{k}\sum_{i=1}^{k} p_i \qquad s^2 = \frac{1}{k-1}\sum_{i=1}^{k}(p_i - \bar{p})^2$

  Bậc tự do (*degree of freedom*): (k – 1)

  Điều kiện bác bỏ: $\frac{\sqrt{N}(\bar{p} - p_0)}{S} \geq t_{\alpha,k-1}$ (trị tới hạn)

2.5 Hypothesis testing

- **So sánh 2 thuật toán**
  - McNemar's test
  - k-fold cross-validation paired t-test
  - 5 x 2 cross-validation paired t-test
  - cross-validation paired F-test
- **So sánh nhiều hơn 2 thuật toán:** *ANOVA* (*ANalysis Of VAriance*)
  - One-way ANOVA

Nội dung

1. Đánh giá giải pháp máy học
2. Đánh giá mô hình phân lớp
3. **Đánh giá mô hình hồi quy**
4. Đánh giá mô hình gom cụm

3. Đánh giá mô hình hồi quy

- **Phương châm**
  - không chệch (*unbiased*): hồi quy trên nhiều mẫu cùng quy mô → giá trị trung bình của các tham số
  - vững chắc (*consistent*): gia tăng quy mô của mẫu
  - hiệu quả (*efficient*): các tham số tốt nhất đối với phương pháp hồi quy đã chọn

3.1 Các tiêu chí

- **Độ đo sai số** *Sum of Squared Error – SSE*

$$SSE = \sum_{i=1}^{N} (\hat{y}_i - y_i)^2$$

- **Độ đo sai số** *Mean Squared Error – MSE* (*L_2 loss*)

$$MSE = \frac{1}{N} \sum_{i=1}^{N} (\hat{y}_i - y_i)^2$$

- **Độ đo sai số** *Mean Absolute Error – MAE* (*L_1 loss*)

$$MAE = \frac{1}{N} \sum_{i=1}^{N} |\hat{y}_i - y_i|$$

3.1 Các tiêu chí

- **Độ đo sai số** *Relative Squared Error – RSE*

$$RSE = \frac{\sum_{i=1}^{n} (\hat{y}_i - y_i)^2}{\sum_{i=1}^{n} (\bar{y}_i - y_i)^2}$$

MSE: *regression errors*
nhạy cảm với *mean*, *scale*

σ^2: *total errors*

- không phụ thuộc vào scale (đơn vị, thứ nguyên) như MSE

3.1 Các tiêu chí

❑ **Độ đo *R-Square* – R^2 (*Coefficient of Determination*)**

$$R^2 = 1 - RSE = 1 - \frac{\sum_{i=1}^{n}(\hat{y}_i - y_i)^2}{\sum_{i=1}^{n}(\bar{y}_i - y_i)^2}$$

- mức độ phù hợp (độ chệch) giữa mô hình với tập dữ liệu
- tỷ lệ % biến thiên của biến phụ thuộc y được "giải thích" bằng các biến độc lập X trong mô hình hồi quy
- mô hình tốt với $R^2 \rightarrow 1$

3.1 Các tiêu chí

❑ **Phân rã sai số**

Training set: $T = \{(x^{(i)}, y_i)\}_{i=1}^{N} \quad x^{(i)} \in \mathbb{R}^d$

Hàm *f* phát sinh ra T: $y = f(x) + \varepsilon$

Nhiễu ε của các $(x^{(i)}, y_i)$ với μ = 0 và σ^2

Mục tiêu tối thiểu hóa $(y - \hat{f}(x))^2$ trên cả 2 training, test sets với $\hat{f}(x)$ là hàm hồi quy

Kỳ vọng sai số trên các tập huấn luyện N quan sát:

$$E[(y - \hat{f}(x))^2] = \underbrace{(E[\hat{f}(x)] - f(x))^2}_{bias^2} + \underbrace{E[(E[\hat{f}(x)] - \hat{f}(x))^2]}_{variance} + \sigma^2$$

σ^2: *irreducible erorr*

3.1 Các tiêu chí

- **Độ chệch** (*bias*)
  - mức độ chênh lệch giữa giá trị hồi quy so với giá trị thực tế của dữ liệu huấn luyện
  - bias *thấp* → mô hình khớp với dữ liệu huấn luyện
  - sai số xuất phát từ những giả thuyết sai lầm dẫn đến mô hình không tốt

    VD: dữ liệu đa thức bậc 2 được giả định là dữ liệu tuyến tính
  - bias *lớn* có nguy cơ không thể hiện được mối quan hệ quan trọng giữa features X và biến độc lập y

3.1 Các tiêu chí

- **Phương sai** (*variance*)
  - độ phân tán của các giá trị hồi quy quanh giá trị trung bình
  - sai số xuất phát từ sự nhạy cảm quá mức đối với 1 số thay đổi bất thường trong dữ liệu huấn luyện
  - phương sai *lớn* thể hiện mức độ đao động lớn trong dự đoán → tổng quát hóa thấp (tốt khi train nhưng kém khi test)

3.1 Các tiêu chí

- **Tính toán độ chệch và phương sai** [Andrew Ng]

  VD:

  Sai số trên training set: β = error(training set) = 15%

  Sai số trên test set: δ = error(test set) = 16%

  Thông thường: $\delta \geq \beta$ (có những quan sát chưa được học)

  Phân tích: $\delta = \beta + 1\%$

  "tệ" hơn so với training set

3.1 Các tiêu chí

- **Tính toán độ chệch và phương sai** [Andrew Ng]

  Giả sử: training, test sets cùng phân phối

  bias = β

  variance = $(\delta - \beta)$

  $\Rightarrow$ Áp dụng cho các mô hình phân lớp ?

3.1 Các tiêu chí

- **Đánh đổi** (trade-off) **giữa bias và variance**
  - những phương pháp làm giảm bias
  - những phương pháp làm giảm variance

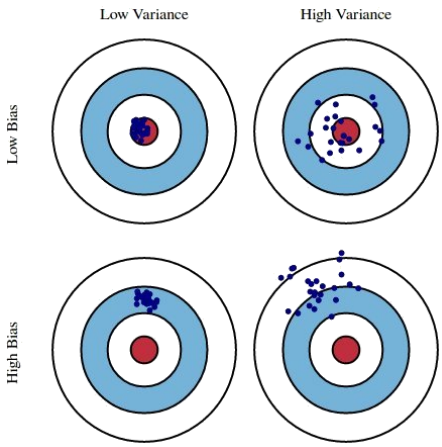
Ockham's razor (1324): "*plurality should not be posited without necessity*."

→ lời giải thích đơn giản nhất thường là xác đáng nhất !

"the supreme goal of all theory is to make the irreducible basic elements as simple and as few as possible without having to surrender the adequate representation of a single datum of experience." [Einstein, 1933]

3.1 Các tiêu chí

- **Đánh đổi** (trade-off) **giữa bias và variance**
  - mô hình quá đơn giản → bias lớn
  - tăng độ phức tạp nhằm giảm bias → variance lớn

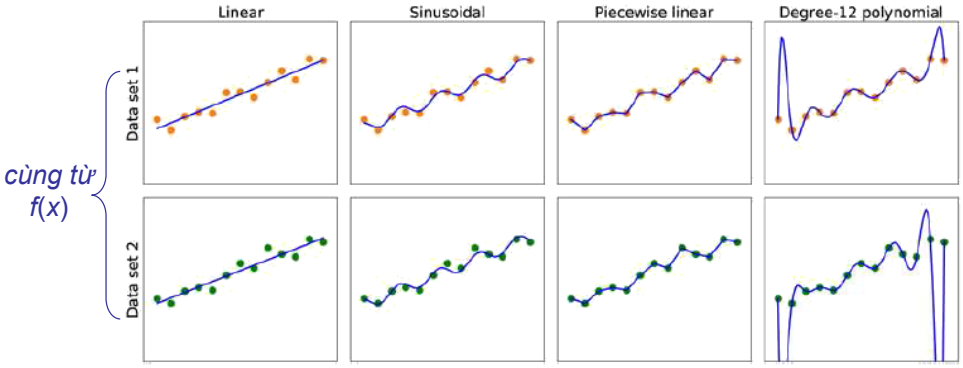


3.1 Các tiêu chí

- **Quan điểm về độ chệch và phương sai** [Russell+]

  *"By **bias** we mean (loosely) the tendency of a predictive hypothesis to deviate from the expected value when averaged over different training sets. Bias often results from restrictions imposed by the hypothesis space."*

  *"By **variance** we mean the amount of change in the hypothesis due to fluctuation in the training data."*



3.2 Các phương pháp

- **Cải biên những phương pháp cho mô hình phân lớp**
- **Sử dụng các biểu đồ trực quan**
  - regression plot
  - scatter plot
  - residual plot
  - distribution plot
  - . . .

3.2 Các phương pháp

❑ **Chọn mô hình theo phương pháp *ELBOW***

- validation set đóng vai trò như test set:

  variance δ = error(validation set)

  total error $\varepsilon = (\delta + \beta)$

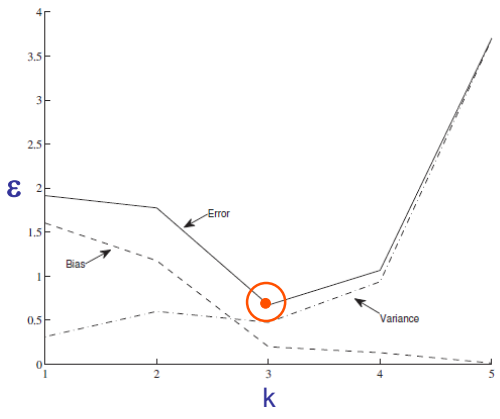
- độ phức tạp của mô hình ↗: bias ↘ trong khi variance ↗

3.2 Các phương pháp

❑ **Chọn mô hình theo phương pháp *ELBOW***

VD: Hồi quy đa thức bậc k ∈ [1, 8]

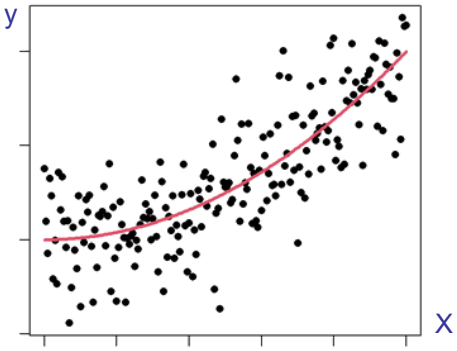
- khi bậc k càng ↗ thì bias ↘ trong khi variance ↗
- chọn k = 3 (elbow)



[Alpaydin]

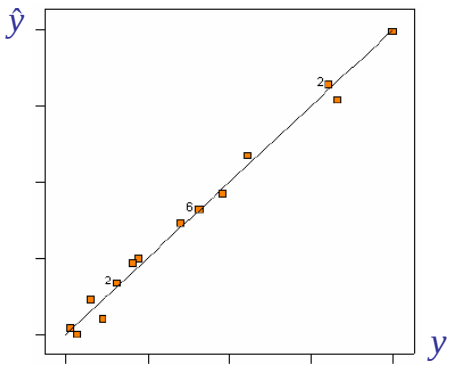
3.2 Các phương pháp

- **Biểu đồ hồi quy** (*Regression Plot*)
  - trục hoành: biến độc lập X
  - trục tung: biến phụ thuộc y
  - đường hồi quy



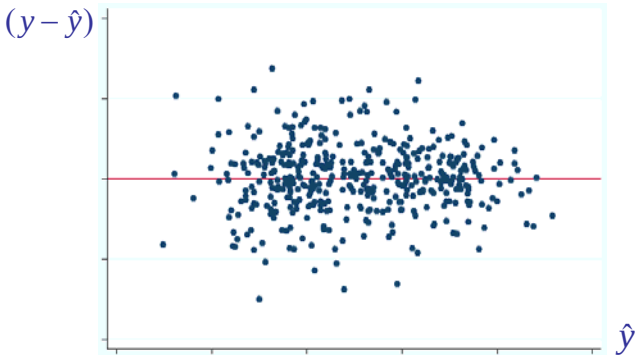
3.2 Các phương pháp

- **Biểu đồ phân tán** (*Scatter Plot*)
  - trục hoành: giá trị thực tế của biến phụ thuộc y
  - trục tung: giá trị hồi quy (dự đoán) $\hat{y}$ của biến phụ thuộc y
  - đường phân giác (y = X)



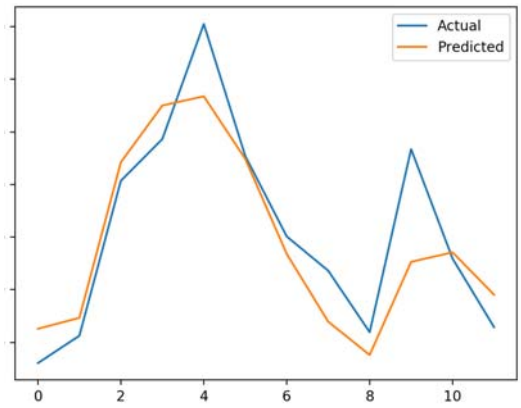
3.2 Các phương pháp

- **Biểu đồ phần dư** (*Residual Plot*)
  - trục hoành: $\hat{y}$
  - trục tung: phần dư $(y - \hat{y})$
  - các điểm rải đều ngẫu nhiên theo trục hoành: phù hợp với mô hình tuyến tính



3.2 Các phương pháp

- **Biểu đồ phân phối** (*Distribution Plot*)
  - *Multiple Linear Regression*: nhiều biến phụ thuộc
  - so sánh phân phối của giá trị hồi quy với giá trị thực tế của y

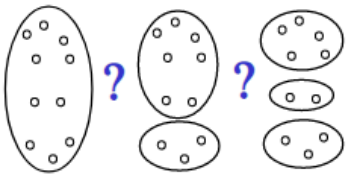


Nội dung

1. Đánh giá giải pháp máy học
2. Đánh giá mô hình phân lớp
3. Đánh giá mô hình hồi quy
4. **Đánh giá mô hình gom cụm**

4. Đánh giá mô hình gom cụm

- **Bài toán phân cụm rất khó đánh giá chất lượng**
  - không có đáp án



- **Các tiêu chí chất lượng (nội tại)**
  - độ nén (*compactness*): các đối tượng trong cụm phải gần nhau
  - độ phân tách (*separation*): các cụm phải tách rời nhau (rõ ràng)

4.1 Phương pháp đánh giá trong

- **Phương pháp đánh giá trong** (*internal validation*)
  - không có thông tin từ bên ngoài
  - chủ yếu dựa trên ma trận xấp xỉ (*proximity matrix*)
  - tối ưu hóa độ tương đồng, độ phân tách
- **Một số độ đo**
  - Silhouette index, Dunn's index
  - Hubert's statistic
  - F-ratio
  - DBI (Davies Bouldin Index)

4.1 Phương pháp đánh giá trong

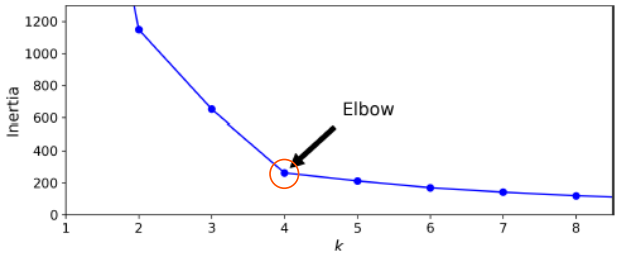
- **Phương pháp ELBOW**

Sum of Squared Error (*cluster inertia*):

$$SSE_k = \sum_{i=1}^{n}\sum_{j=1}^{k} w_{ij} \left\| x^{(i)} - \mu_j \right\|^2 \qquad w_{ij} = \begin{cases} 1 & x^{(i)} \in \text{cluster } C_j \\ 0 & x^{(i)} \notin \text{cluster } C_j \end{cases}$$

$$\mu_j = centroid(C_j)$$

- khi số cụm k ↗ thì SSE_k có xu thế ↘ vì các quan sát sẽ gần với trọng tâm hơn



[*Géron*]

4.1 Phương pháp đánh giá trong

❑ **Độ đo** *Silhouette Score*

$$Silhouette\_coef(x) = \frac{(b_x - a_x)}{\max(a_x, b_x)}$$

a_x: khoảng cách trung bình từ x đến các quan sát cùng cluster (*mean intra-cluster distance*)

b_x: khoảng cách trung bình từ x đến các quan sát thuộc cluster gần nhất (*mean nearest-cluster distance*)

Silhouette_coef(x) $\in$ [–1, 1]

→ 1 : x được phân cụm tốt vì $b_x >> a_x$

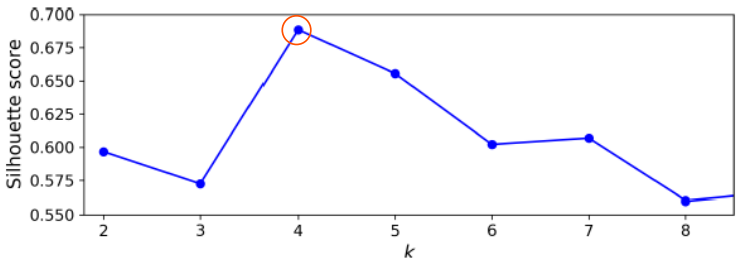
= 0 : x nằm gần đường biên giữa 2 clusters

→ –1: x thuộc cluster không phù hợp

4.1 Phương pháp đánh giá trong

❑ **Độ đo** *Silhouette Score*

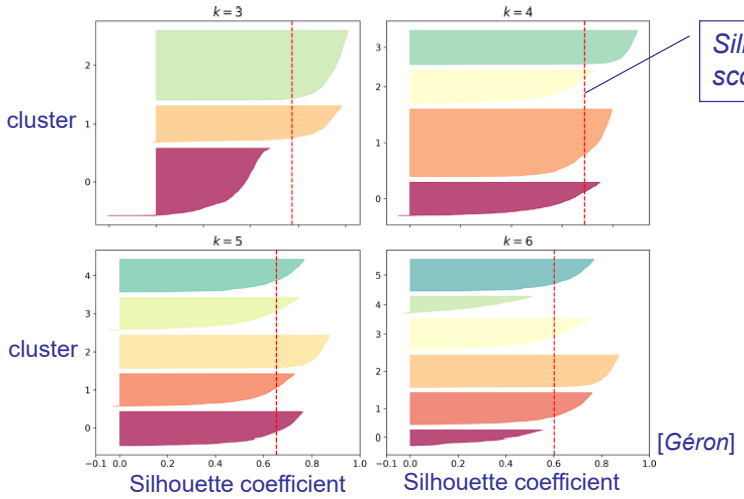
$$Silhouette\_score = \frac{1}{n}\sum_{i=1}^{1} Silhouette\_coef(x^{(i)})$$



[*Géron*]

4.1 Phương pháp đánh giá trong

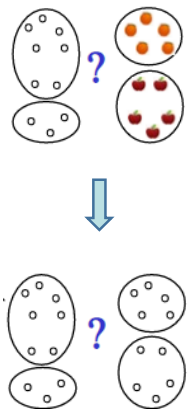
- **Biểu đồ *Silhouette* (*Silhouette diagram*)**
  - mỗi cluster: sắp xếp $x^{(i)}$ giảm dần theo Silhouette_coef
  - clustering kém: nhiều $x^{(i)}$ có Silhouette_coef ngắn hơn vạch ---



[Géron]

4.2 Phương pháp đánh giá ngoài

- **Phương pháp đánh giá ngoài** (*external validation*)
  - dựa vào cấu trúc/xu hướng gom cụm được xác định trước
  - so sánh mức độ sai khác giữa các cụm
  - so sánh với kết quả mẫu



- **Một số độ đo**
  - Rand index (statistic)
  - Jaccard coefficient
  - Folkes
  - Mallows index

4.2 Phương pháp đánh giá ngoài

- **Chỉ số Rand** (*Rand index*)

  R: phân hoạch các quan sát $x^{(i)}$ thành r clusters

  S: phân hoạch các quan sát thành s clusters

  a: số $(x^{(i)}, x^{(j)})$ thuộc cùng cluster trong R lẫn trong S

  b: số $(x^{(i)}, x^{(j)})$ thuộc $\neq$ cluster trong R lẫn trong S

  c: số $(x^{(i)}, x^{(j)})$ thuộc cùng cluster trong R và $\neq$ cluster trong S

  d: số $(x^{(i)}, x^{(j)})$ thuộc $\neq$ cluster trong R và cùng cluster trong S

$$R = \frac{a+b}{a+b+c+d} = \frac{a+b}{\binom{n}{2}} \in [0,1]$$

4.3 Phương pháp đánh giá tương đối

- **Phương pháp đánh giá tương đối** (*relative validation*)
  - so sánh kết quả của những bộ giá trị tham số khác nhau (*grid search*, *random search*, …)
  - so sánh kết quả gom cụm giữa các phương pháp khác nhau

Tài liệu tham khảo

Alpaydin, *Introduction to Machine Learning*, 4rd Edition, 2020.

Géron, *Hands-on ML with Scikit-Learn, Keras and TensorFlow*, 2nd Edition, 2019.

Mitchell, *Machine Learning*, 1st Edition, 1997.

Russell and Norvig, *Artificial Intelligence: A Modern Approach*, 4th Edition, 2020.

Thảo luận