

Học Máy

(IT 4862)

Nguyễn Nhật Quang

quangnn-fit@mail.hut.edu.vn

Trường Đại học Bách Khoa Hà Nội
Viện Công nghệ thông tin và truyền thông
Năm học 2011-2012

Nội dung môn học:

- **Giới thiệu chung**
 - **Học máy**
 - **Công cụ WEKA**
- Đánh giá hiệu năng hệ thống học máy
- Các phương pháp học dựa trên xác suất
- Các phương pháp học có giám sát
- Các phương pháp học không giám sát
- Lọc cộng tác
- Học tăng cường

Giới thiệu về Học máy

- Học máy (Machine Learning – ML) là một lĩnh vực nghiên cứu của Trí tuệ nhân tạo (Artificial Intelligence – AI)
- Các định nghĩa về học máy
 - Một quá trình nhờ đó một hệ thống cải thiện hiệu suất (hiệu quả hoạt động) của nó [Simon, 1983]
 - Một quá trình mà một chương trình máy tính cải thiện hiệu suất của nó trong một công việc thông qua kinh nghiệm [Mitchell, 1997]
 - Việc lập trình các máy tính để tối ưu hóa một tiêu chí hiệu suất dựa trên các dữ liệu ví dụ hoặc kinh nghiệm trong quá khứ [Alpaydin, 2004]
- Biểu diễn một bài toán học máy [Mitchell, 1997]

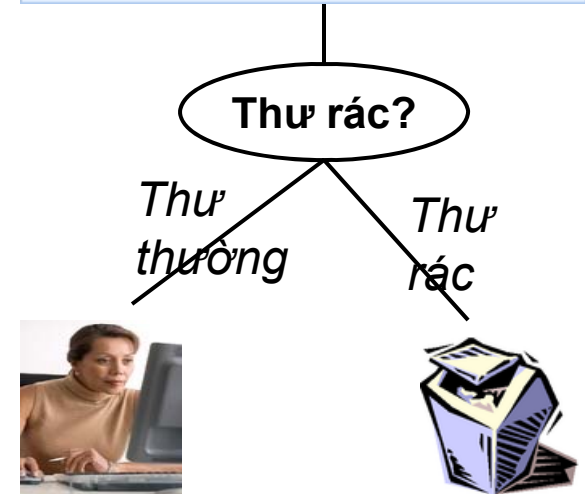
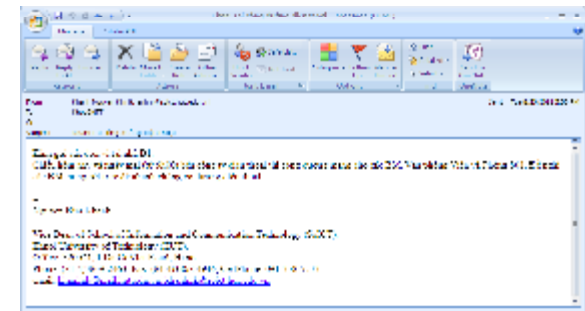
Học máy = Cải thiện hiệu quả một công việc thông qua kinh nghiệm

 - Một công việc (nhiệm vụ) T
 - Đối với các tiêu chí đánh giá hiệu năng P
 - Thông qua (sử dụng) kinh nghiệm E

Ví dụ bài toán học máy (1)

Lọc thư rác – Email spam filtering

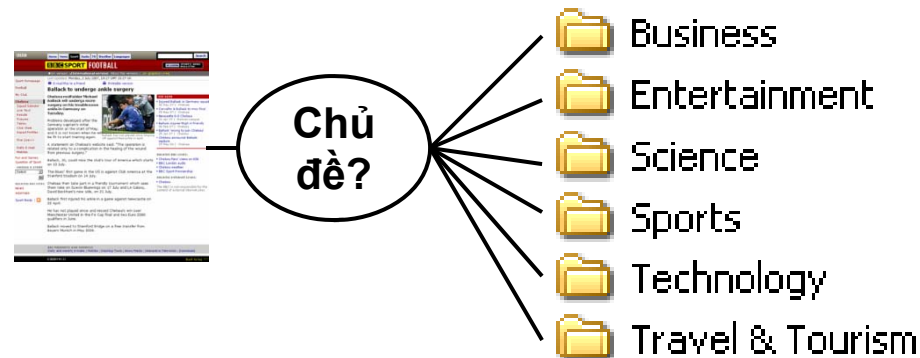
- ***T***: Dự đoán (để lọc) những thư điện tử nào là thư rác (spam email)
- ***P***: % of các thư điện tử gửi đến được phân loại chính xác
- ***E***: Một tập các thư điện tử (emails) mẫu, mỗi thư điện tử được biểu diễn bằng một tập thuộc tính (vd: tập từ khóa) và nhãn lớp (thư thường/thư rác) tương ứng



Ví dụ bài toán học máy (2)

Phân loại các trang Web

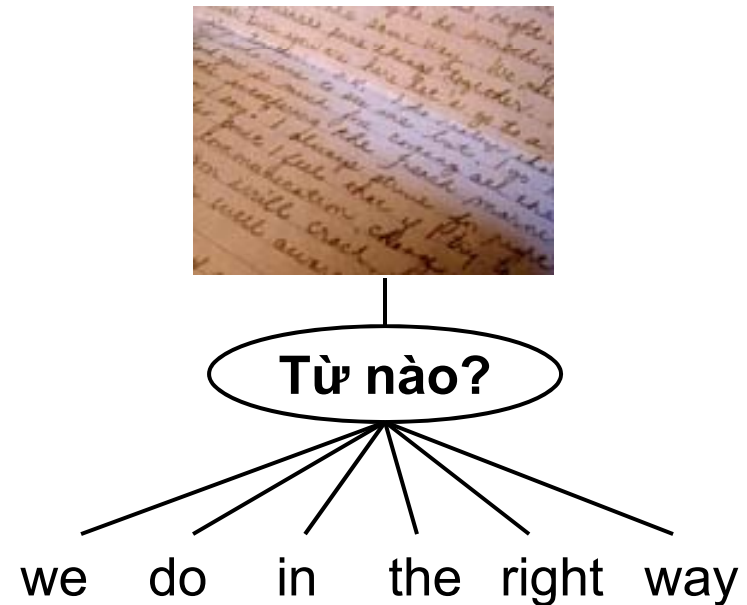
- **T**: Phân loại các trang Web theo các chủ đề đã định trước
- **P**: Tỷ lệ (%) các trang Web được phân loại chính xác
- **E**: Một tập các trang Web, trong đó mỗi trang Web gắn với một chủ đề



Ví dụ bài toán học máy (3)

Nhận dạng chữ viết tay

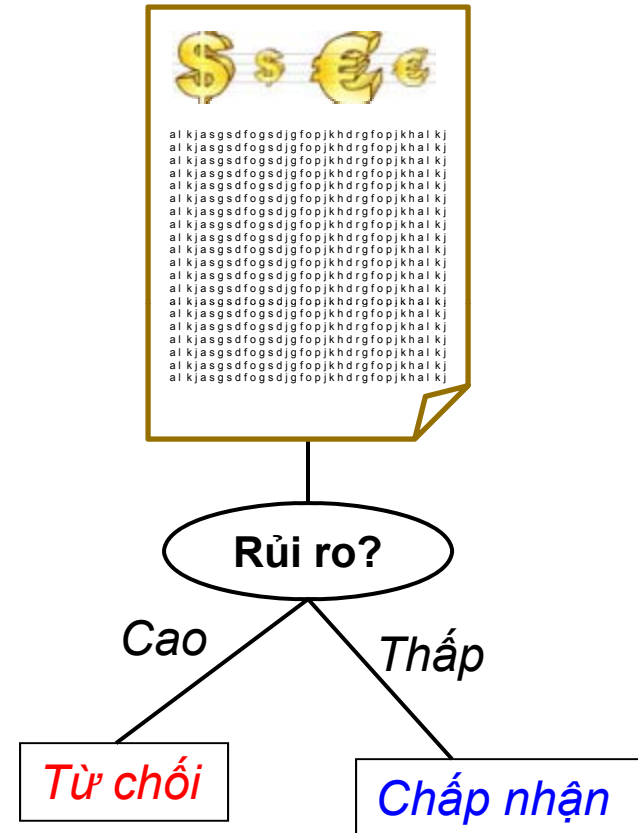
- **T**: Nhận dạng và phân loại các từ trong các ảnh chữ viết tay
- **P**: Tỷ lệ (%) các từ được nhận dạng và phân loại đúng
- **E**: Một tập các ảnh chữ viết tay, trong đó mỗi ảnh được gắn với một định danh của một từ



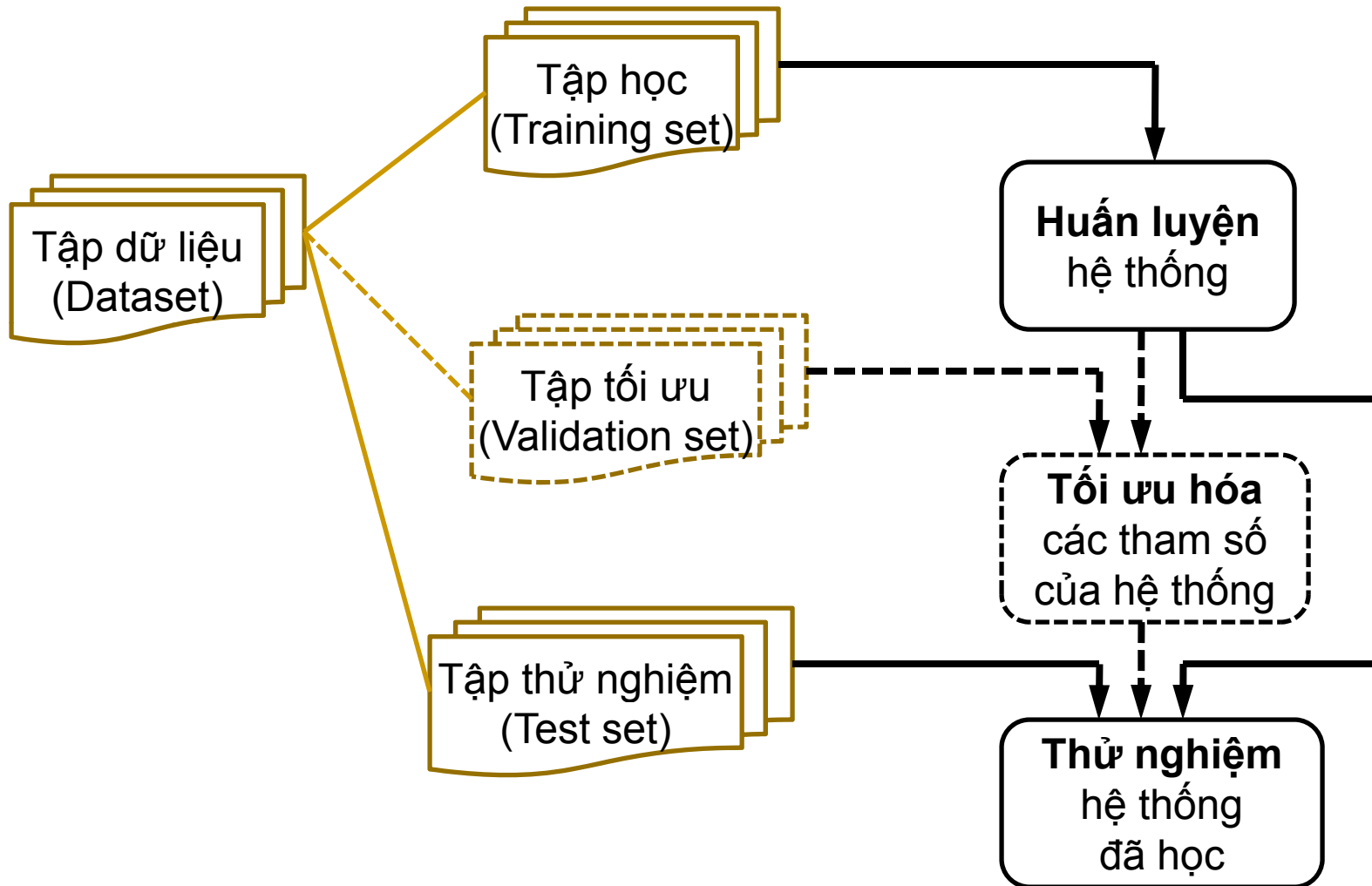
Ví dụ bài toán học máy (4)

Dự đoán rủi ro cho vay tài chính

- **T :** Xác định mức độ rủi ro (vd: cao/thấp) đối với các hồ sơ xin vay tài chính
- **P :** Tỷ lệ % các hồ sơ xin vay có mức độ rủi ro cao (không trả lại tiền vay) được xác định chính xác
- **E :** Một tập các hồ sơ xin vay; mỗi hồ sơ được biểu diễn bởi một tập các thuộc tính và mức độ rủi ro (cao/thấp)



Quá trình học máy



Các thành phần chính của bài toán học máy (1)

■ Lựa chọn các ví dụ học (training/learning examples)

- Các thông tin hướng dẫn quá trình học (training feedback) được chứa ngay trong các ví dụ học, hay là được cung cấp gián tiếp (vd: từ môi trường hoạt động)
- Các ví dụ học theo kiểu có giám sát (supervised) hay không có giám sát (unsupervised)
- Các ví dụ học phải tương thích với (đại diện cho) các ví dụ sẽ được sử dụng bởi hệ thống trong tương lai (future test examples)

■ Xác định hàm mục tiêu (giả thiết, khái niệm) cần học

- $F: X \rightarrow \{0,1\}$
- $F: X \rightarrow$ Một tập các nhãn lớp
- $F: X \rightarrow \mathbb{R}^+$ (miền các giá trị số thực dương)
- ...

Các thành phần chính của bài toán học máy (2)

- Lựa chọn cách biểu diễn cho hàm mục tiêu cần học
 - Hàm đa thức (a polynomial function)
 - Một tập các luật (a set of rules)
 - Một cây quyết định (a decision tree)
 - Một mạng nơ-ron nhân tạo (an artificial neural network)
 - ...
- Lựa chọn một giải thuật học máy có thể học (xấp xỉ) được hàm mục tiêu
 - Phương pháp học hồi quy (Regression-based)
 - Phương pháp học quy nạp luật (Rule induction)
 - Phương pháp học cây quyết định (ID3 hoặc C4.5)
 - Phương pháp học lan truyền ngược (Back-propagation)
 - ...

Các vấn đề trong Học máy (1)

- Giải thuật học máy (Learning algorithm)
 - Những giải thuật học máy nào có thể học (xấp xỉ) một hàm mục tiêu cần học?
 - Với những điều kiện nào, một giải thuật học máy đã chọn sẽ hội tụ (tiệm cận) hàm mục tiêu cần học?
 - Đối với một lĩnh vực bài toán cụ thể và đối với một cách biểu diễn các ví dụ (đối tượng) cụ thể, giải thuật học máy nào thực hiện tốt nhất?

Các vấn đề trong Học máy (2)

- Các ví dụ học (Training examples)
 - Bao nhiêu ví dụ học là đủ?
 - Kích thước của tập học (tập huấn luyện) ảnh hưởng thế nào đối với độ chính xác của hàm mục tiêu học được?
 - Các ví dụ lỗi (nhiều) và/hoặc các ví dụ thiếu giá trị thuộc tính (missing-value) ảnh hưởng thế nào đối với độ chính xác?

Các vấn đề trong Học máy (3)

- Quá trình học (Learning process)
 - Chiến lược tối ưu cho việc lựa chọn thứ tự sử dụng (khai thác) các ví dụ học?
 - Các chiến lược lựa chọn này làm thay đổi mức độ phức tạp của bài toán học máy như thế nào?
 - Các tri thức cụ thể của bài toán (ngoài các ví dụ học) có thể đóng góp thế nào đối với quá trình học?

Các vấn đề trong Học máy (4)

- Khả năng/giới hạn học (Learning capability)
 - Hàm mục tiêu nào mà hệ thống cần học?
 - Biểu diễn hàm mục tiêu: Khả năng biểu diễn (vd: hàm tuyến tính / hàm phi tuyến) vs. Độ phức tạp của giải thuật và quá trình học
 - Các giới hạn (trên lý thuyết) đối với khả năng học của các giải thuật học máy?
 - Khả năng khái quát hóa (generalize) của hệ thống từ các ví dụ học?
 - Để tránh vấn đề “over-fitting” (đạt độ chính xác cao trên tập học, nhưng đạt độ chính xác thấp trên tập thử nghiệm)
 - Khả năng hệ thống tự động thay đổi (thích nghi) biểu diễn (cấu trúc) bên trong của nó?
 - Để cải thiện khả năng (của hệ thống đối với việc) biểu diễn và học hàm mục tiêu

Vấn đề over-fitting (1)

- Một hàm mục tiêu (một giả thiết) học được h sẽ được gọi là **quá khớp/quá phù hợp (over-fit)** với một tập học nếu tồn tại một hàm mục tiêu khác h' sao cho:
 - h' kém phù hợp hơn (đạt độ chính xác kém hơn) h đối với tập học, nhưng
 - h' đạt độ chính xác cao hơn h đối với toàn bộ tập dữ liệu (bao gồm cả những ví dụ được sử dụng sau quá trình huấn luyện)
- Vấn đề over-fitting thường do các nguyên nhân:
 - Lỗi (nhiều) trong tập huấn luyện (do quá trình thu thập/xây dựng tập dữ liệu)
 - Số lượng các ví dụ học quá nhỏ, không đại diện cho toàn bộ tập (phân bố) của các ví dụ của bài toán học

Vấn đề over-fitting (2)

- Giả sử gọi D là tập toàn bộ các ví dụ, và D_{train} là tập các ví dụ học
- Giả sử gọi $\text{Err}_D(h)$ là mức lỗi mà giả thiết h sinh ra đối với tập D , và $\text{Err}_{D_{\text{train}}}(h)$ là mức lỗi mà giả thiết h sinh ra đối với tập D_{train}
- Giả thiết h quá khớp (quá phù hợp) tập học D_{train} nếu tồn tại một giả thiết khác h' :
 - $\text{Err}_{D_{\text{train}}}(h) < \text{Err}_{D_{\text{train}}}(h')$, và
 - $\text{Err}_D(h) > \text{Err}_D(h')$

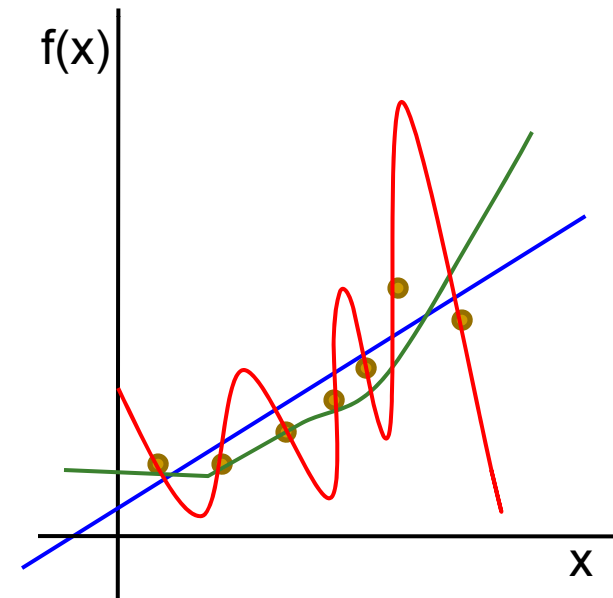
Vấn đề over-fitting (3)

- Trong số các giả thiết (hàm mục tiêu) học được, giả thiết (hàm mục tiêu) nào khái quát hóa tốt nhất từ các ví dụ học?

Lưu ý: Mục tiêu của học máy là để đạt được độ chính xác cao trong dự đoán đối với các ví dụ sau này, không phải đối với các ví dụ học

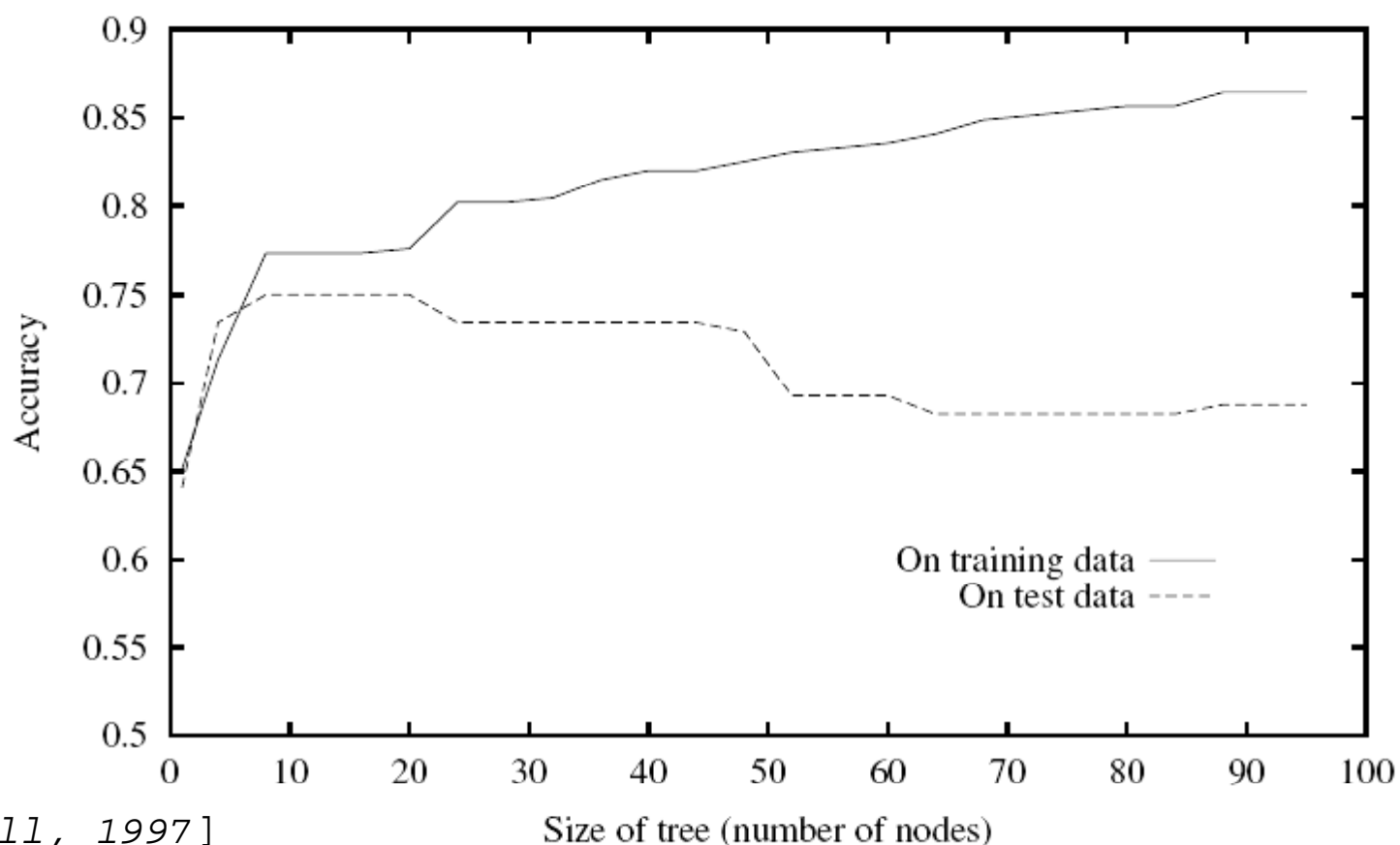
- **Occam's razor:** Ưu tiên chọn hàm mục tiêu đơn giản nhất phù hợp (không nhất thiết hoàn hảo) với các ví dụ học
 - Khái quát hóa tốt hơn
 - Dễ giải thích/diễn giải hơn
 - Độ phức tạp tính toán ít hơn

Hàm mục tiêu $f(x)$ nào đạt độ chính xác cao nhất đối với các ví dụ sau này?



Vấn đề over-fitting – Ví dụ

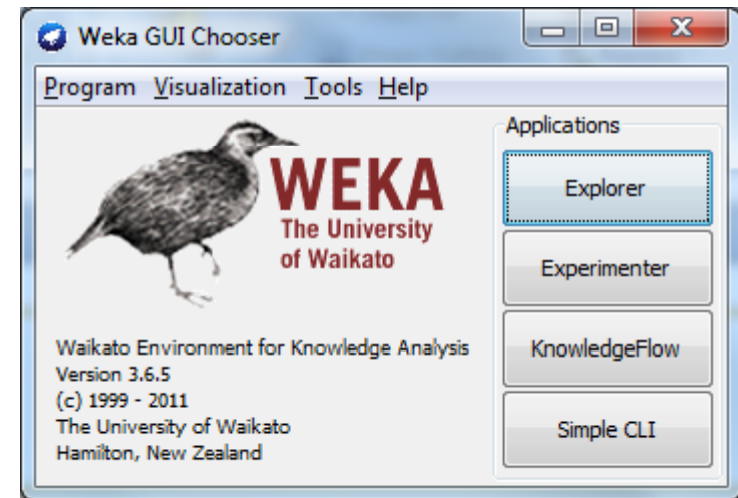
Tiếp tục quá trình học cây quyết định sẽ làm giảm độ chính xác đối với tập thử nghiệm mặc dù tăng độ chính xác đối với tập học



[Mitchell, 1997]

WEKA – Giới thiệu

- WEKA là một công cụ phần mềm viết bằng Java, phục vụ lĩnh vực học máy và khai phá dữ liệu
- Có thể tải về từ địa chỉ:
<http://www.cs.waikato.ac.nz/ml/weka/>



- Các tính năng chính
 - Một tập các công cụ tiền xử lý dữ liệu, các giải thuật học máy, khai phá dữ liệu, và các phương pháp thí nghiệm đánh giá
 - Giao diện đồ họa (gồm cả tính năng hiển thị hóa dữ liệu)
 - Môi trường cho phép so sánh các giải thuật học máy và khai phá dữ liệu

WEKA – Các môi trường chính

- Simple CLI

Giao diện đơn giản kiểu dòng lệnh (như MS-DOS)

- **Explorer** (chúng ta sẽ chủ yếu sử dụng môi trường này!)

Môi trường cho phép sử dụng tất cả các khả năng của WEKA để khám phá dữ liệu

- Experimenter

Môi trường cho phép tiến hành các thí nghiệm và thực hiện các kiểm tra thống kê (statistical tests) giữa các mô hình học máy

- KnowledgeFlow

Môi trường cho phép bạn tương tác đồ họa kiểu kéo/thả để thiết kế các bước (các thành phần) của một thí nghiệm

WEKA – Môi trường Explorer

The screenshot displays the Weka Explorer application window. The top menu bar includes Preprocess, Classify, Cluster, Associate, Select attributes, and Visualize. Below the menu is a toolbar with buttons for Open file..., Open URL..., Open DB..., Undo, Edit..., and Save....

The Filter section shows the selected filter: **AttributeSelection** -E "weka.attributeSelection.CfsSubsetEval" -S "weka.attributeSelection.BestFirst -D 1 -N 5". An Apply button is next to it.

The Current relation section shows: Relation: weather, Instances: 14, Attributes: 5.

The Attributes section has buttons for All, None, and Invert. Below is a table of attributes:

No.	Name
1	<input checked="" type="checkbox"/> outlook
2	<input type="checkbox"/> temperature
3	<input type="checkbox"/> humidity
4	<input type="checkbox"/> windy
5	<input type="checkbox"/> play

A Remove button is located below the attributes table.

The Selected attribute section shows details for the selected attribute 'outlook': Name: outlook, Missing: 0 (0%), Distinct: 3, Type: Nominal, Unique: 0 (0%). Below this is a table:

Label	Count
sunny	5
overcast	4
rainy	5

The Class section shows: Class: play (Nom). A Visualize All button is next to it.

The visualization area displays a bar chart with three bars representing the 'outlook' attribute values: sunny (5), overcast (4), and rainy (5). Each bar is stacked with blue at the bottom and red at the top. The total count for each bar is 5.

The Status bar at the bottom shows: OK, Log, and a small icon with 'x 0'.

WEKA – Môi trường Explorer

- **Preprocess**

Để chọn và thay đổi (xử lý) dữ liệu làm việc

- **Classify**

Để huấn luyện và kiểm tra các mô hình học máy (phân loại, hoặc hồi quy/dự đoán)

- **Cluster**

Để học các nhóm từ dữ liệu (phân cụm)

- **Associate**

Để khám phá các luật kết hợp từ dữ liệu

- **Select attributes**

Để xác định và lựa chọn các thuộc tính liên quan (quan trọng) nhất của dữ liệu

- **Visualize**

Để xem (hiển thị) biểu đồ tương tác 2 chiều đối với dữ liệu

WEKA – Khuôn dạng của tập dữ liệu

- WEKA chỉ làm việc với các tập tin văn bản (text) có khuôn dạng *ARFF*
- Ví dụ của một tập dữ liệu

@relation weather ← Tên của tập dữ liệu

@attribute outlook {sunny, overcast, rainy} ← Thuộc tính kiểu định danh

@attribute temperature real ← Thuộc tính kiểu số

@attribute humidity real

@attribute windy {TRUE, FALSE}

@attribute play {yes, no} ← Thuộc tính phân lớp (mặc định là thuộc tính cuối cùng)

@data

sunny, 85, 85, FALSE, no

overcast, 83, 86, FALSE, yes

...

← Các ví dụ (instances)

WEKA Explorer: Tiền xử lý dữ liệu

- Dữ liệu có thể được nhập vào (imported) từ một tập tin có khuôn dạng: ARFF, CSV
- Dữ liệu cũng có thể được đọc vào từ một địa chỉ URL, hoặc từ một cơ sở dữ liệu thông qua JDBC
- Các công cụ tiền xử lý dữ liệu của WEKA được gọi là *filters*
 - Rời rạc hóa (Discretization)
 - Chuẩn hóa (Normalization)
 - Lấy mẫu (Re-sampling)
 - Lựa chọn thuộc tính (Attribute selection)
 - Chuyển đổi (Transforming) và kết hợp (Combining) các thuộc tính
 - ...

→ *Hãy xem giao diện của WEKA Explorer...*

WEKA Explorer: Các bộ phân lớp (1)

- Các bộ phân lớp (Classifiers) của WEKA tương ứng với các mô hình dự đoán các đại lượng kiểu định danh (phân lớp) hoặc các đại lượng kiểu số (hồi quy/dự đoán)
 - Các kỹ thuật phân lớp được hỗ trợ bởi WEKA
 - Naïve Bayes classifier and Bayesian networks
 - Decision trees
 - Instance-based classifiers
 - Support vector machines
 - Neural networks
 - ...
- *Hãy xem giao diện của WEKA Explorer...*

WEKA Explorer: Các bộ phân lớp (2)

- Lựa chọn một bộ phân lớp (classifier)
- Lựa chọn các tùy chọn cho việc kiểm tra (test options)
 - **Use training set.** Bộ phân loại học được sẽ được đánh giá trên tập học
 - **Supplied test set.** Sử dụng một tập dữ liệu khác (với tập học) để cho việc đánh giá
 - **Cross-validation.** Tập dữ liệu sẽ được chia đều thành k tập (folds) có kích thước xấp xỉ nhau, và bộ phân loại học được sẽ được đánh giá bởi phương pháp *cross-validation*
 - **Percentage split.** Chỉ định tỷ lệ phân chia tập dữ liệu đối với việc đánh giá

WEKA Explorer: Các bộ phân lớp (3)

■ More options...

- **Output model.** Hiển thị bộ phân lớp học được
- **Output per-class stats.** Hiển thị các thông tin thống kê về precision/recall đối với mỗi lớp
- **Output entropy evaluation measures.** Hiển thị đánh giá độ hỗn tạp (entropy) của tập dữ liệu
- **Output confusion matrix.** Hiển thị thông tin về ma trận lỗi phân lớp (confusion matrix) đối với phân lớp học được
- **Store predictions for visualization.** Các dự đoán của bộ phân lớp được lưu lại trong bộ nhớ, để có thể được hiển thị sau đó
- **Output predictions.** Hiển thị chi tiết các dự đoán đối với tập kiểm tra
- **Cost-sensitive evaluation.** Các lỗi (của bộ phân lớp) được xác định dựa trên ma trận chi phí (cost matrix) chỉ định
- **Random seed for XVal / % Split.** Chỉ định giá trị *random seed* được sử dụng cho quá trình lựa chọn ngẫu nhiên các ví dụ cho tập kiểm tra

WEKA Explorer: Các bộ phân lớp (4)

- **Classifier output** hiển thị các thông tin quan trọng
 - **Run information.** Các tùy chọn đối với mô hình học, tên của tập dữ liệu, số lượng các ví dụ, các thuộc tính, và f.f. thí nghiệm
 - **Classifier model (full training set).** Biểu diễn (dạng text) của bộ phân lớp học được
 - **Predictions on test data.** Thông tin chi tiết về các dự đoán của bộ phân lớp đối với tập kiểm tra
 - **Summary.** Các thống kê về mức độ chính xác của bộ phân lớp, đối với f.f. thí nghiệm đã chọn
 - **Detailed Accuracy By Class.** Thông tin chi tiết về mức độ chính xác của bộ phân lớp đối với mỗi lớp
 - **Confusion Matrix.** Các thành phần của ma trận này thể hiện số lượng các ví dụ kiểm tra (test instances) được phân lớp đúng và bị phân lớp sai

WEKA Explorer: Các bộ phân lớp (5)

■ **Result list** cung cấp một số chức năng hữu ích

- **Save model.** Lưu lại mô hình tương ứng với bộ phân lớp học được vào trong một tập tin nhị phân (binary file)
- **Load model.** Đọc lại một mô hình đã được học trước đó từ một tập tin nhị phân
- **Re-evaluate model on current test set.** Đánh giá một mô hình (bộ phân lớp) học được trước đó đối với tập kiểm tra (test set) hiện tại
- **Visualize classifier errors.** Hiển thị cửa sổ biểu đồ thể hiện các kết quả của việc phân lớp

Các ví dụ được phân lớp chính xác sẽ được biểu diễn bằng ký hiệu bởi dấu chéo (x), còn các ví dụ bị phân lớp sai sẽ được biểu diễn bằng ký hiệu ô vuông (□)

- ...

WEKA Explorer: Các bộ phân cụm (1)

- Các bộ phân cụm (Cluster builders) của WEKA tương ứng với các mô hình tìm các nhóm của các ví dụ tương tự đối với một tập dữ liệu
 - Các kỹ thuật phân cụm được hỗ trợ bởi WEKA
 - Expectation maximization (EM)
 - k-Means
 - ...
 - Các bộ phân cụm có thể được hiển thị kết quả và so sánh với các cụm (lớp) thực tế
- *Hãy xem giao diện của WEKA Explorer ...*

WEKA Explorer: Các bộ phân cụm (2)

- Lựa chọn một bộ phân cụm (cluster builder)
- Lựa chọn chế độ phân cụm (cluster mode)
 - **Use training set.** Các cụm học được sẽ được kiểm tra đối với tập học
 - **Supplied test set.** Sử dụng một tập dữ liệu khác để kiểm tra các cụm học được
 - **Percentage split.** Chỉ định tỷ lệ phân chia tập dữ liệu ban đầu cho việc xây dựng tập kiểm tra
 - **Classes to clusters evaluation.** So sánh độ chính xác của các cụm học được đối với các lớp được chỉ định
- Store clusters for visualization
 - Lưu lại các bộ phân lớp trong bộ nhớ, để có thể hiển thị sau đó
- Ignore attributes
 - Lựa chọn các thuộc tính sẽ không tham gia vào quá trình học các cụm

WEKA Explorer: Phát hiện luật kết hợp

- Lựa chọn một mô hình (giải thuật) phát hiện luật kết hợp
- **Associator output** hiển thị các thông tin quan trọng
 - **Run information.** Các tùy chọn đối với mô hình phát hiện luật kết hợp, tên của tập dữ liệu, số lượng các ví dụ, các thuộc tính
 - **Associator model (full training set).** Biểu diễn (dạng text) của tập các luật kết hợp phát hiện được
 - Độ hỗ trợ tối thiểu (minimum support)
 - Độ tin cậy tối thiểu (minimum confidence)
 - Kích thước của các tập mục thường xuyên (large/frequent itemsets)
 - Liệt kê các luật kết hợp tìm được

→ *Hãy xem giao diện của WEKA Explorer...*

WEKA Explorer: Lựa chọn thuộc tính

- Để xác định những thuộc tính nào là quan trọng nhất
 - Trong WEKA, một phương pháp lựa chọn thuộc tính (attribute selection) bao gồm 2 phần:
 - *Attribute Evaluator*. Để xác định một phương pháp đánh giá mức độ phù hợp của các thuộc tính
Vd: correlation-based, wrapper, information gain, chi-squared,...
 - *Search Method*. Để xác định một phương pháp (thứ tự) xét các thuộc tính
Vd: best-first, random, exhaustive, ranking,...
- *Hãy xem giao diện của WEKA Explorer...*

WEKA Explorer: Hiển thị dữ liệu

- Hiển thị dữ liệu rất cần thiết trong thực tế
 - Giúp dễ xác định mức độ khó khăn của bài toán học
- WEKA có thể hiển thị
 - Mỗi thuộc tính riêng lẻ (1-D visualization)
 - Một cặp thuộc tính (2-D visualization)
- Các giá trị (các nhãn) lớp khác nhau sẽ được hiển thị bằng các màu khác nhau
- Thanh trượt **Jitter** hỗ trợ việc hiển thị rõ ràng hơn, khi có quá nhiều ví dụ (điểm) tập trung xung quanh một vị trí trên biểu đồ
- Tính năng phóng to/thu nhỏ (bằng cách tăng/giảm giá trị của **PlotSize** và **PointSize**)

→ *Hãy xem giao diện của WEKA Explorer...*

References

- E. Alpaydin. *Introduction to Machine Learning*. The MIT Press, 2004.
- T. M. Mitchell. *Machine Learning*. McGraw-Hill, 1997.
- H. A. Simon. *Why Should Machines Learn?* In R. S. Michalski, J. Carbonell, and T. M. Mitchell (Eds.): *Machine learning: An artificial intelligence approach*, chapter 2, pp. 25-38. Morgan Kaufmann, 1983.