

# Dữ Liệu Lớn

## Prediction with Machine Learning

**Thân Quang Khoát**

*khoattq@soict.hust.edu.vn*

Viện Công nghệ thông tin và Truyền thông  
Trường Đại Học Bách Khoa Hà Nội

Năm 2019

# Nội dung khóa học

---

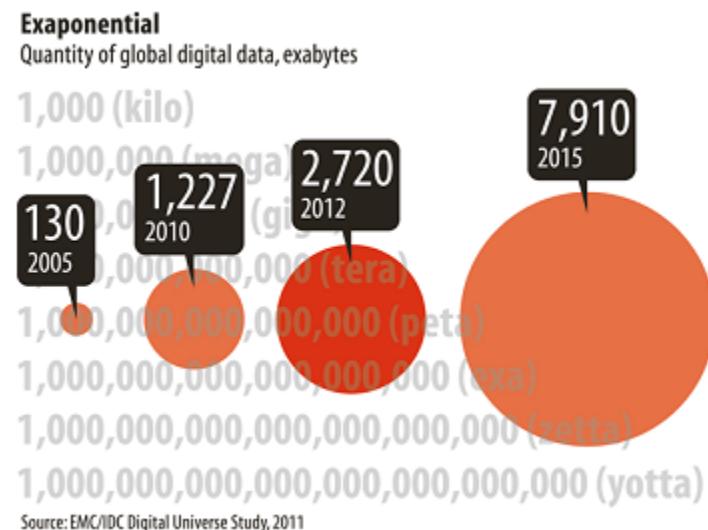
- Overview of data analytics/science
- Basic statistics
- Python and programming tools
- Exploratory data analysis
- Data integration and preprocessing
- **Prediction with machine learning**
- Data visualization
- Evaluation of analysis results
- Basics of natural language processing
- Anomaly detection
- Big data analysis
- Capstone project

# Tại sao nên biết Học Máy?

- Học máy (ML – Machine Learning):  
data mining, inference, prediction.
- ML là con đường hiệu quả để tạo ra các hệ thống thông minh,  
dịch vụ thông minh.
- ML cung cấp nền tảng và phương pháp cho Big Data.



**Each day:**  
230M tweets,  
2.7B comments to FB,  
86400 hours of video  
to YouTube

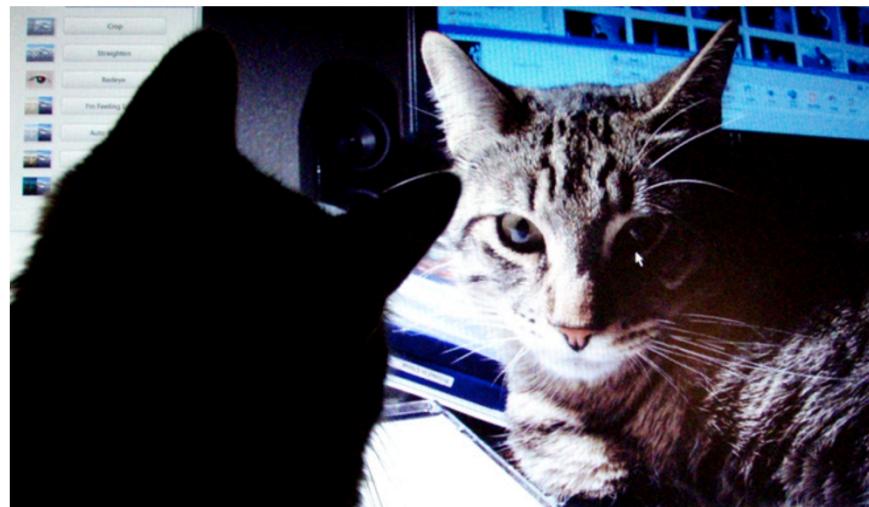


# Vài thành công: GoogleBrain (2012)

## Google's Artificial Brain Learns to Find Cat Videos

BY WIRED UK 06.26.12 | 11:15 AM | PERMALINK

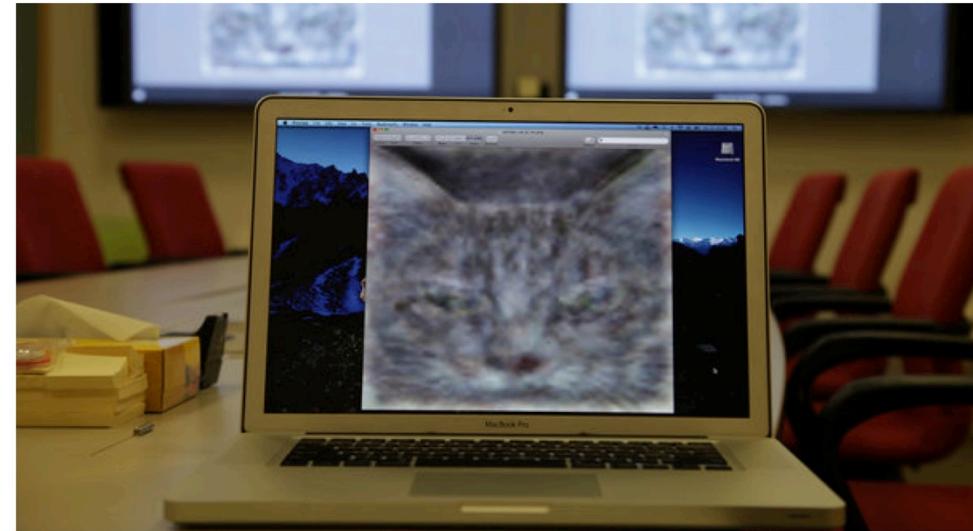
[Share](#) 138 [Tweet](#) 32 [g+1](#) 506 [in Share](#) 8 [Pin it](#)



By Liat Clark, Wired UK



How Many Computers to Identify a Cat? 16,000



An image of a cat that a neural network taught itself to recognize.

By JOHN MARKOFF

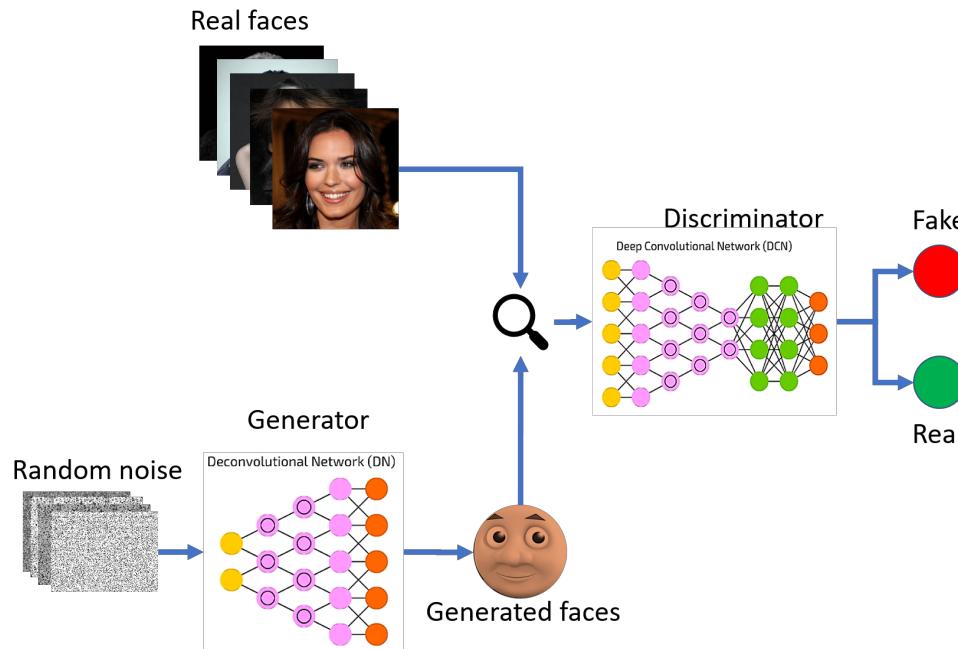
Published: June 25, 2012

Jim Wilson/The New York Times

# Vài thành công: GAN (2014)

## ■ Tạo Trí tưởng tượng (Imagination)

Ian Goodfellow

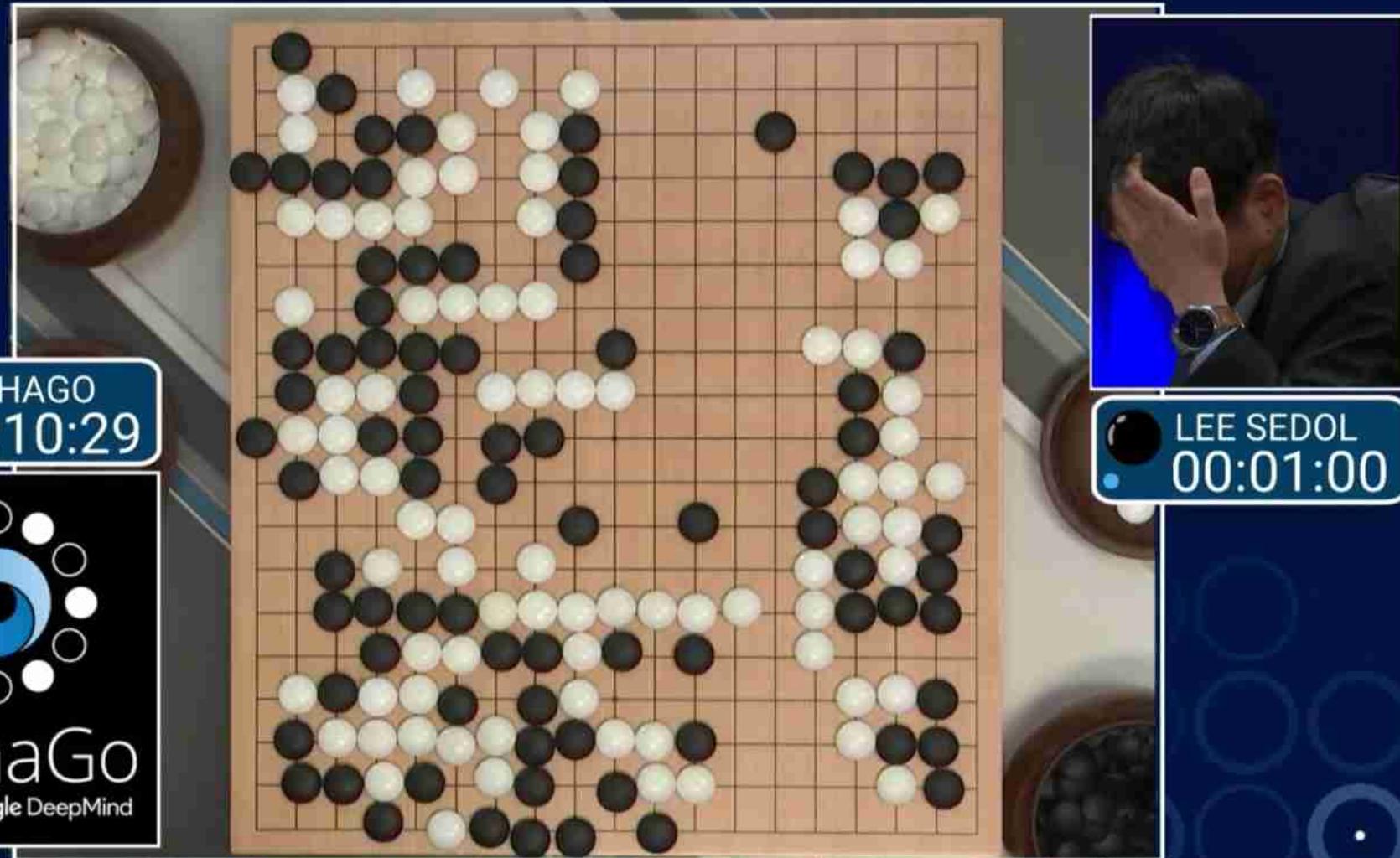


Artificial faces



Goodfellow, Ian, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. "Generative adversarial nets." In *NIPS*, pp. 2672-2680. 2014.

## Vài thành công: AlphaGo (2016)



# Machine Learning?

- Học máy (ML - Machine Learning) là một lĩnh vực nghiên cứu của Trí tuệ nhân tạo (Artificial Intelligence)
- Câu hỏi trung tâm của ML: [Mitchell, 2006]
  - *How can we build computer systems that automatically improve with experience, and what are the fundamental laws that govern all learning processes?*
- Vài quan điểm về học máy:
  - Build systems that automatically improve their performance [Simon, 1983].
  - Program computers to optimize a performance objective at some task, based on data and past experience [Alpaydin, 2010]



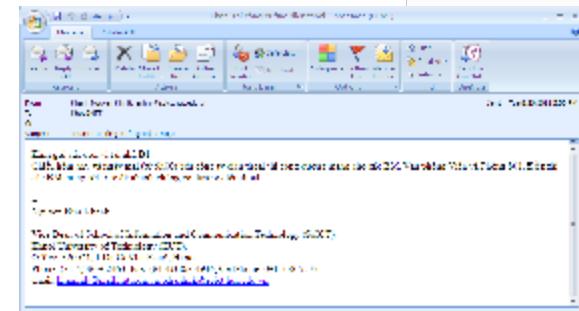
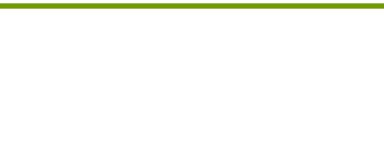
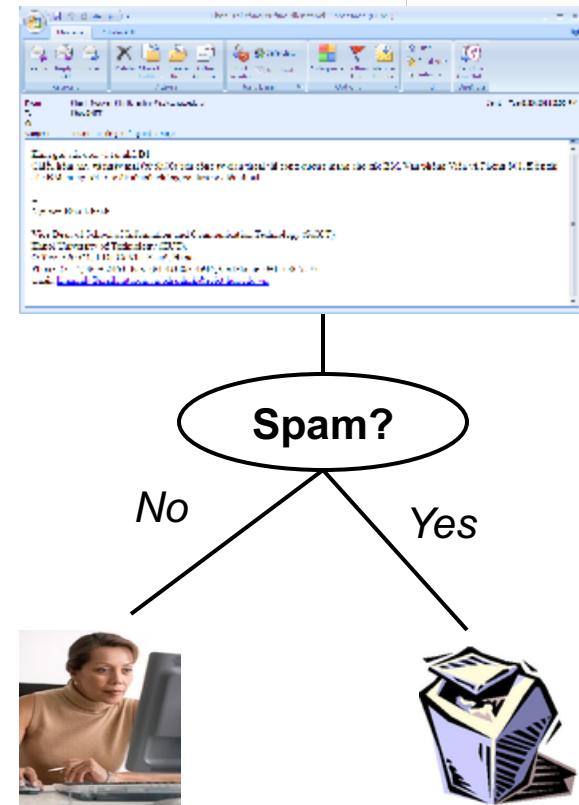
# Máy học

- Ta nói một máy tính *có khả năng học* nếu nó tự cải thiện hiệu suất hoạt động  $P$  cho một công việc  $T$  cụ thể, dựa vào kinh nghiệm  $E$  của nó.
- Như vậy *một bài toán học máy* có thể biểu diễn bằng 1 bộ  $(T, P, E)$ 
  - $T$ : một công việc (nhiệm vụ)
  - $P$ : tiêu chí đánh giá hiệu năng
  - $E$ : kinh nghiệm

# Ví dụ thực tế (1)

## ■ Lọc thư rác (email spam filtering)

- $T$ : Dự đoán (để lọc) những thư điện tử nào là thư rác (spam email)
- $P$ : số lượng 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 thực tế (2)

### Gán nhãn ảnh

- **T:** đưa ra một vài mô tả ý nghĩa của 1 bức ảnh
- **P:** ?
- **E:** Một tập các bức ảnh, trong đó mỗi ảnh đã được gán một tập các từ mô tả ý nghĩa của chúng



FISH WATER OCEAN  
TREE CORAL



PEOPLE MARKET PATTERN  
TEXTILE DISPLAY



BIRDS NEST TREE  
BRANCH LEAVES

# Máy học gì?

- Học một ánh xạ (hàm):

$$f : x \mapsto y$$

- **x**: quan sát (dữ liệu), kinh nghiệm
- **y**: phán đoán, tri thức mới, kinh nghiệm mới, ...
- **Hồi quy** (regression): nếu y là một số thực
- **Phân loại** (classification): nếu y thuộc một tập rời rạc (tập nhãn lớp)

Anh ta thích nghe



+



→ Trẻ hay Già ?

# Máy học từ đâu?

## ■ Học từ đâu?

- Từ các quan sát trong quá khứ (*tập học – training data set*).  
 $\{\{x_1, x_2, \dots, x_N\}; \{y_1, y_2, \dots, y_M\}\}$
- $x_i$  là các quan sát của  $x$  trong quá khứ
- $y_h$  là *nhãn (label)* hoặc *phản hồi (response)* hoặc *đầu ra (output)* tương ứng với  $x_h$ .

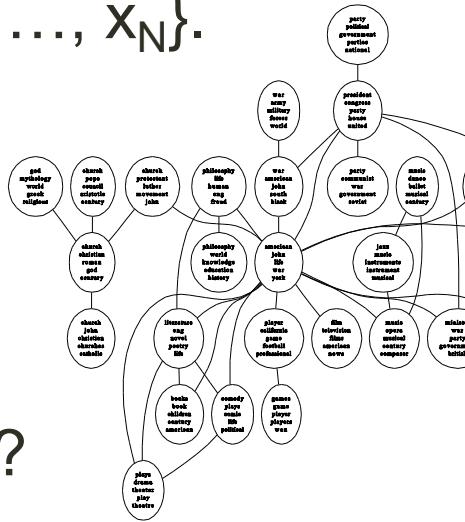
## ■ Sau khi đã học:

- Thu được một mô hình, kinh nghiệm, tri thức mới ( $f$ ).
- Dùng nó để *suy diễn (infer)* hoặc *phán đoán (predict)* cho quan sát trong tương lai.

$$y_z = f(z)$$

## Hai bài toán học cơ bản

- **Học có giám sát (supervised learning):** cần học một hàm  $y = f(x)$  từ tập học  $\{\{x_1, x_2, \dots, x_N\}; \{y_1, y_2, \dots, y_N\}\}$  sao cho  $y_i \approx f(x_i)$ .
    - *Phân loại* (phân lớp): nếu  $y$  chỉ nhận giá trị từ một tập rời rạc, chẳng hạn {cá, cây, quả, mèo}
    - *Hồi quy*: nếu  $y$  nhận giá trị số thực
  - **Học không giám sát (unsupervised learning):** cần học một hàm  $y = f(x)$  từ tập học cho trước  $\{x_1, x_2, \dots, x_N\}$ .
    - $Y$  có thể là các cụm dữ liệu.
    - $Y$  có thể là các cấu trúc ẩn.
  - **Học bán giám sát (semi-supervised learning)?**



# Supervised learning: Phân loại

- **Multi-class classification** (*phân loại nhiều lớp*):  
when the output  $y$  is one of the pre-defined labels

$\{c_1, c_2, \dots, c_L\}$

(mỗi đầu ra chỉ thuộc 1 lớp, mỗi quan sát x chỉ có 1 nhãn)

- Spam filtering:  $y$  in {spam, normal}
  - Financial risk estimation:  $y$  in {high, normal, no}
  - Discovery of network attacks: ?

- **Multi-label classification** (*phân loại đa nhãn*): when the output  $y$  is a subset of labels

(mỗi đầu ra là một tập nhỏ các lớp;  
mỗi quan sát x có thể có nhiều nhãn)

- Image tagging:  $y = \{\text{birds, nest, tree}\}$
  - sentiment analysis



# Supervised learning: Hồi quy

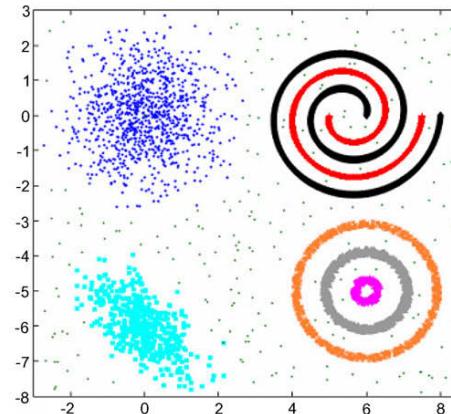
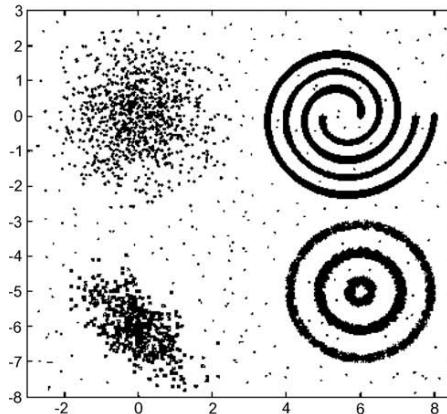
- #### ■ Phán đoán chỉ số chứng khoán



# Unsupervised learning: ví dụ (1)

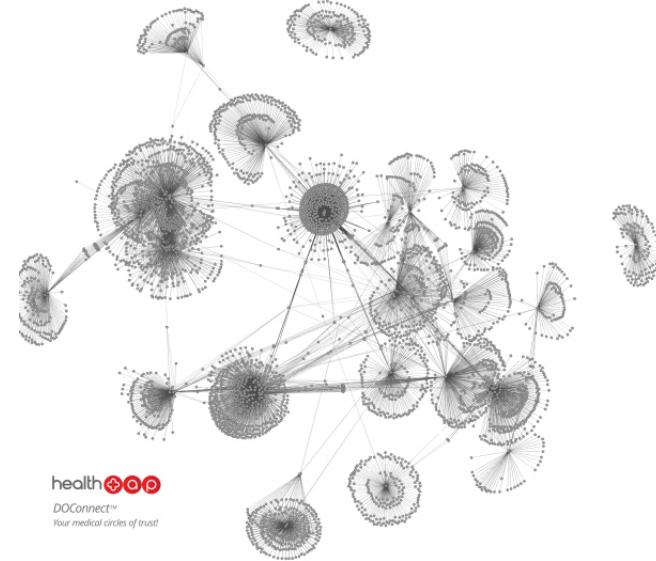
## ■ Gom nhóm dữ liệu vào các cụm (Clustering)

- Discover the data groups/clusters



## ■ Phát hiện cộng đồng

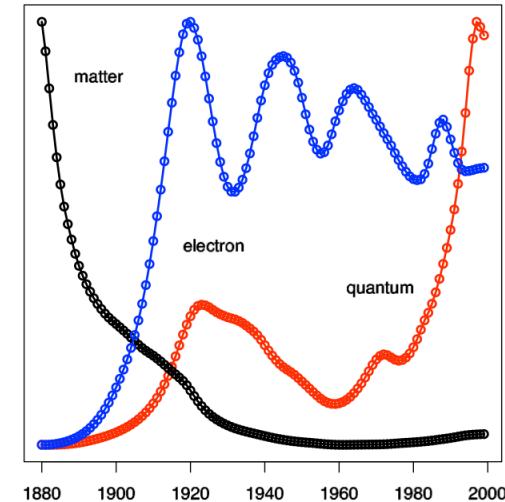
- Detect communities in online social networks



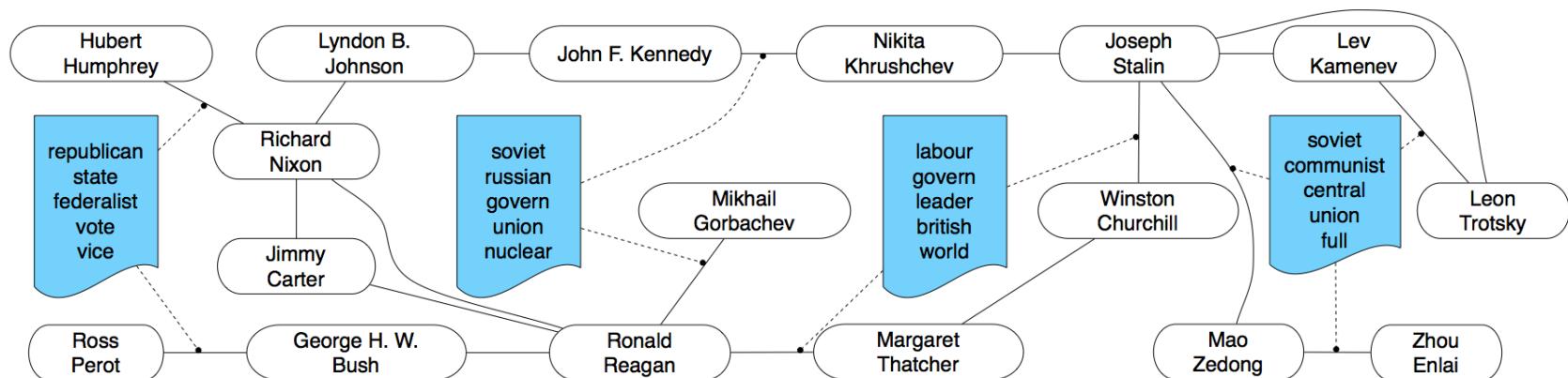
# Unsupervised learning: ví dụ (2)

## ■ Trends detection

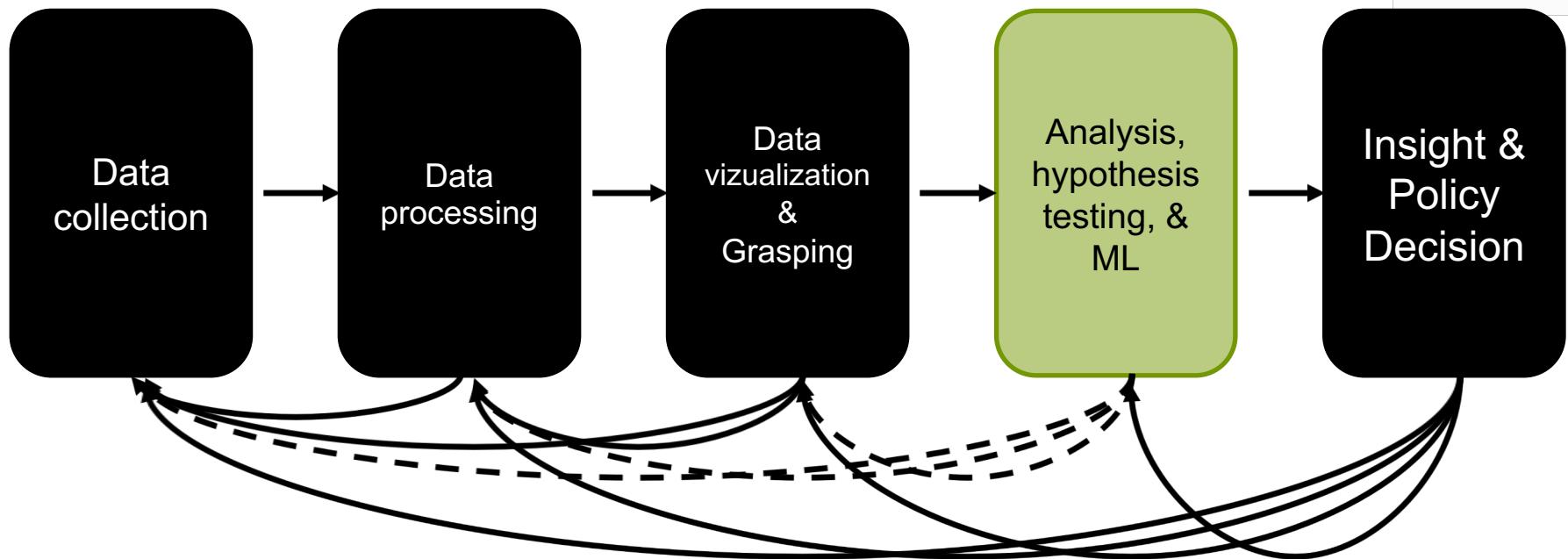
- Discover the trends, demands, future needs of online users



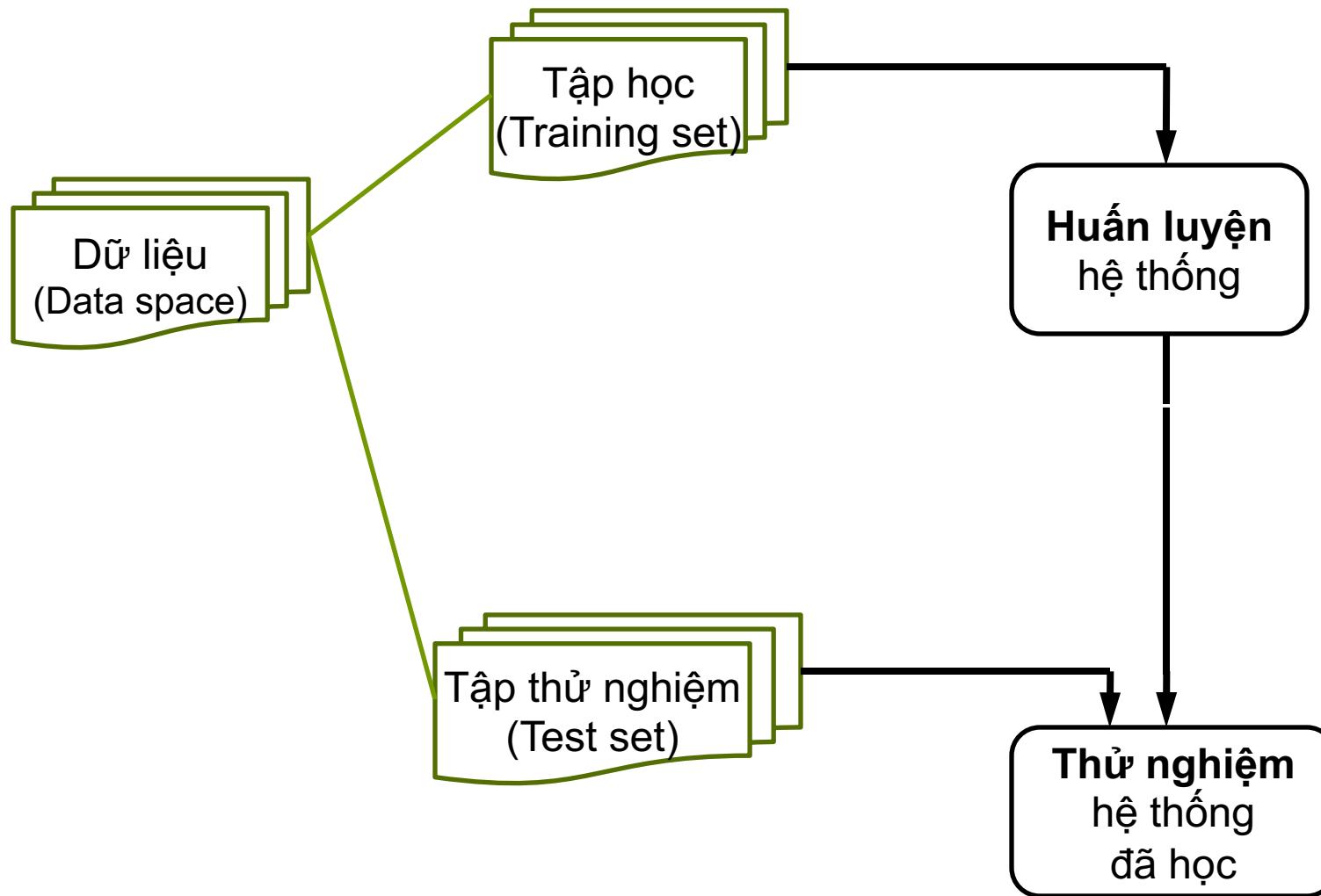
## ■ Entity-interaction analysis



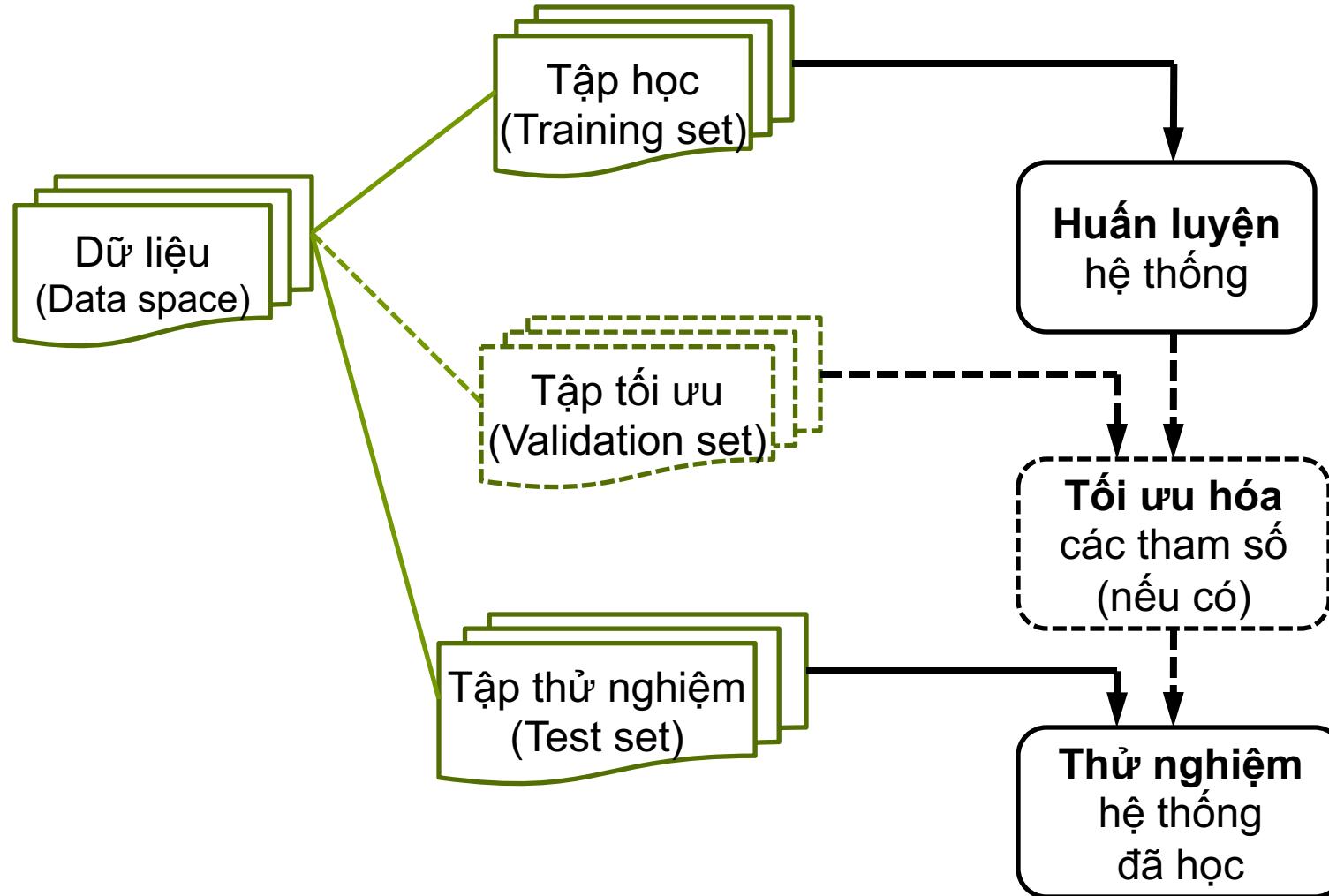
# Học máy nằm ở đâu?



# Quá trình học máy: cơ bản



# Quá trình học máy: toàn diện



# Thiết kế một hệ thống học (1)

- Một số vấn đề quan trọng cần được xem xét kỹ khi thiết kế một hệ thống học
- **Lựa chọn tập học (training examples/data):**
  - Tập học có ảnh hưởng lớn đến hiệu quả của hệ thống học.
  - Liệu ta có thu thập được nhãn cho dữ liệu huấn luyện?
  - Các ví dụ học nên tương thích với (đại diện cho) các ví dụ sẽ được làm việc bởi hệ thống trong tương lai (future test examples)
- **Xác định được bài toán học máy nào?**
  - Phân loại?  $F: X \rightarrow \{0,1\}$ ;  $F: X \rightarrow \text{set of labels/tags}$
  - Hồi quy?  $F: X \rightarrow \mathbb{R}$
  - Phân cụm? (không thu thập được đầu ra  $y$ )
  - ...

# Thiết kế một hệ thống học (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)
  - ...

# Vài 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) đến hàm mục tiêu cần học?
  - Đối với một lĩnh vực 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?
- **No-free-lunch theorem** [Wolpert and Macready, 2005]:  
*If an algorithm performs well on a certain class of problems then it necessarily pays for that with degraded performance on the set of all remaining problems.*
- ❖ **No algorithm can beat another on all domains.**  
(không có thuật toán nào luôn hiệu quả nhất trên mọi miền ứng dụng)

## Vài 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?

# Vài 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?

# Vài vấn đề trong Học máy (4)

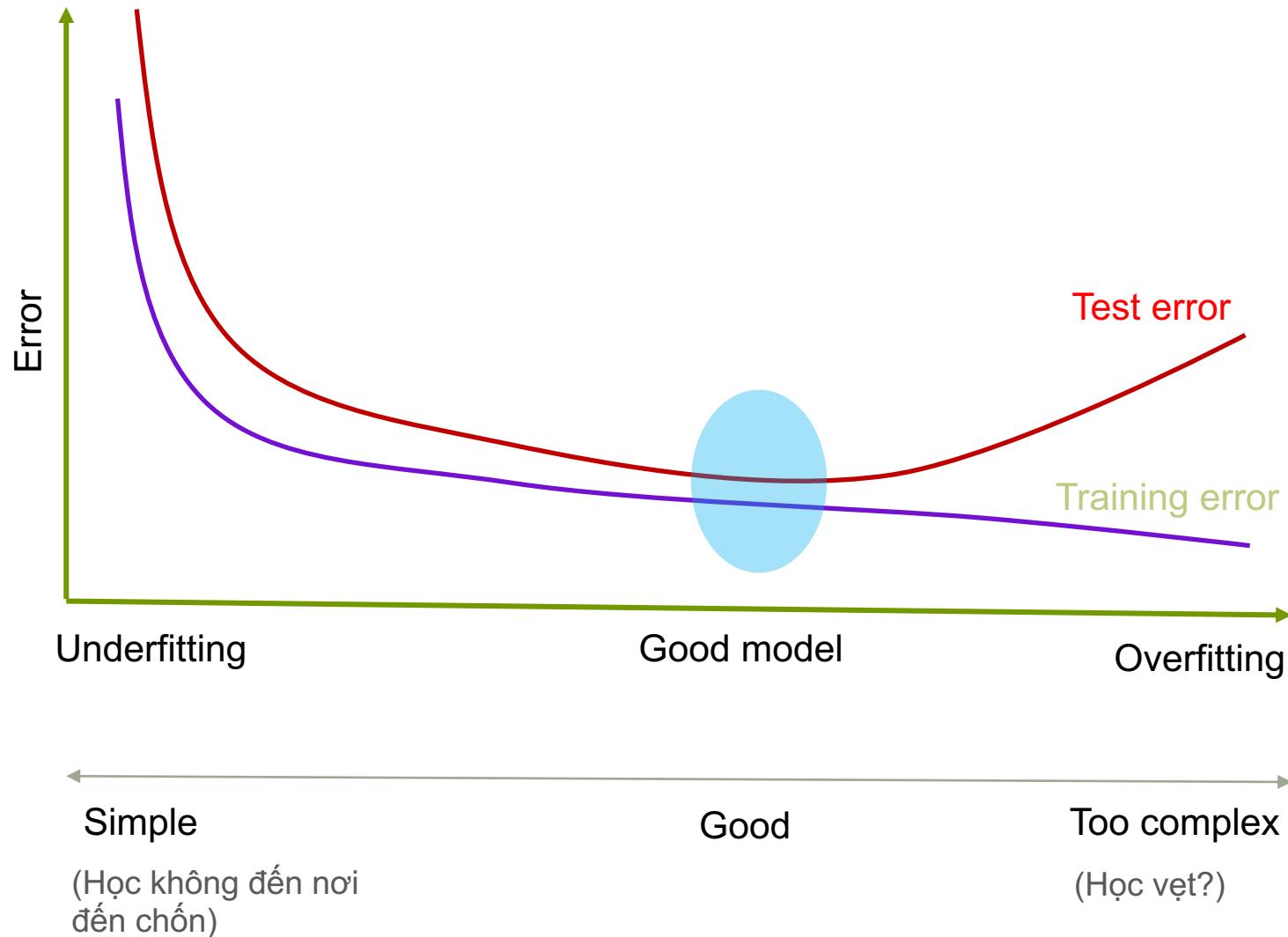
## ■ Khả năng/giới hạn học (Learnability)

- 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 **Tổng quát hóa (generalization)** của hệ thống?
  - 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

# Overfitting (quá khớp, quá khít)

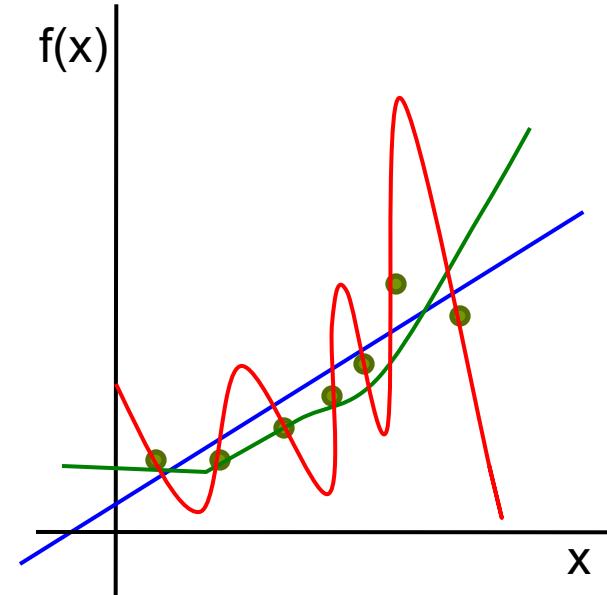
- Hàm  $h$  được gọi là *overfitting* nếu tồn tại hàm  $g$  mà:
  - $g$  có thể tồi hơn  $h$  đối với tập huấn luyện,
  - nhưng  $g$  tốt hơn  $h$  đối với dữ liệu tương lai.
- A learning algorithm is said to overfit relative to another one if it is *more accurate in fitting* known data, but *less accurate in predicting* unseen data.
- Vài nguyên nhân gây ra Overfitting:
  - Hàm  $h$  quá phức tạp
  - 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 overfitting: minh họa



# Overfitting: Regularization

- Trong số rất nhiều hàm thì hàm nào có khả năng tổng quát cao nhất khi học từ tập dữ liệu cho trước?
  - *Tổng quát hóa là mục tiêu chính của học máy.*
  - Tức là, khả năng phán đoán tốt với dữ liệu tương lai.
- **Regularization:** cách dung phổ biến
  - Là cách hạn chế không gian học hàm  $f$ .



# Tài liệu tham khảo

---

- Alpaydin E. (2010). Introduction to Machine Learning. The MIT Press.
- Mitchell, T. M. (1997). Machine learning. *McGraw Hill*.
- Mitchell, T. M. (2006). *The discipline of machine learning*. Carnegie Mellon University, School of Computer Science, Machine Learning Department.
- Simon H.A. (1983). 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.
- Wolpert, D.H., Macready, W.G. (1997), "No Free Lunch Theorems for Optimization", *IEEE Transactions on Evolutionary Computation* 1, 67.