# TIỀN XỬ LÝ DỮ LIỆU

# Nội dung

- Đặc điểm chung của dữ liệu
- Vì sao phải tiền xử lý dữ liệu?
- Tóm tắt dữ liệu
- Làm sạch dữ liệu
- Tích hợp & biến đổi dữ liệu
- Thu gọn dữ liệu

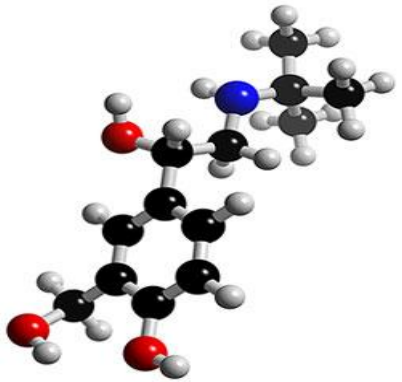
# Nội dung

- Đặc điểm chung của dữ liệu
- Vì sao phải tiền xử lý dữ liệu?
- Tóm tắt dữ liệu
- Làm sạch dữ liệu
- Tích hợp & biến đổi dữ liệu
- Thu gọn dữ liệu

# Các dạng dữ liệu phổ biến

- Dạng hồ sơ (record): văn bản, ma trận …
- Dạng đồ thị (graph): cấu trúc mạng, cấu trúc phân tử…
- Dạng có trật tự (ordered): chuỗi thời gian, chuỗi giao dịch…

| *TID* | *Items* |
|---|---|
| 1 | Bread, Coke, Milk |
| 2 | Beer, Bread |
| 3 | Beer, Coke, Diaper, Milk |
| 4 | Beer, Bread, Diaper, Milk |
| 5 | Coke, Diaper, Milk |



# Các loại thuộc tính tiêu biểu

- Bình thường
  - Nghề nghiệp, ID, màu mắt …
- Có thứ tự
  - Xếp hạng, điểm số, chiều cao …
- Nhị phân
  - Kết quả xét nghiệm …
- Khoảng
  - Nhiệt độ cơ thể, ngày tháng …
- …

# Các đặc điểm quan trọng của tập dữ liệu

- Số chiều (dimensionality)
- Sự phân bố (sparsity)
- Sự tương đồng (similarity)

# Nội dung

- Đặc điểm chung của dữ liệu
- Vì sao phải tiền xử lý dữ liệu?
- Tóm tắt dữ liệu
- Làm sạch dữ liệu
- Tích hợp & biến đổi dữ liệu
- Thu gọn dữ liệu

# Vì sao phải tiền xử lý dữ liệu?

- Dữ liệu không hoàn chỉnh
  - Vd: ID = “”
- Dữ liệu bị nhiễu
  - Vd: Luong = -1000
- Dữ liệu không nhất quán
  - Vd: 01/04/1989
- …

# Vì sao dữ liệu không sạch?

- Mục tiêu, cách nhìn khác nhau khi thu thập dữ liệu
- Lỗi trong quá trình truyền dữ liệu
- Dữ liệu từ nhiều nguồn khác nhau
- Các vấn đề từ con người, phần cứng, phần mềm
- …
- *Các record trùng nhau cũng phải được xóa*

# Vì sao tiền xử lý dữ liệu rất quan trọng?

- Dữ liệu không tốt
  - ➔ Việc khai thác dữ liệu không tốt
  - ➔ Người dùng không tin tưởng vào kết quả
- Trích dẫn, làm sạch, chuyển hóa … dữ liệu là công việc chính khi xây dựng một kho dữ liệu (data warehouse)

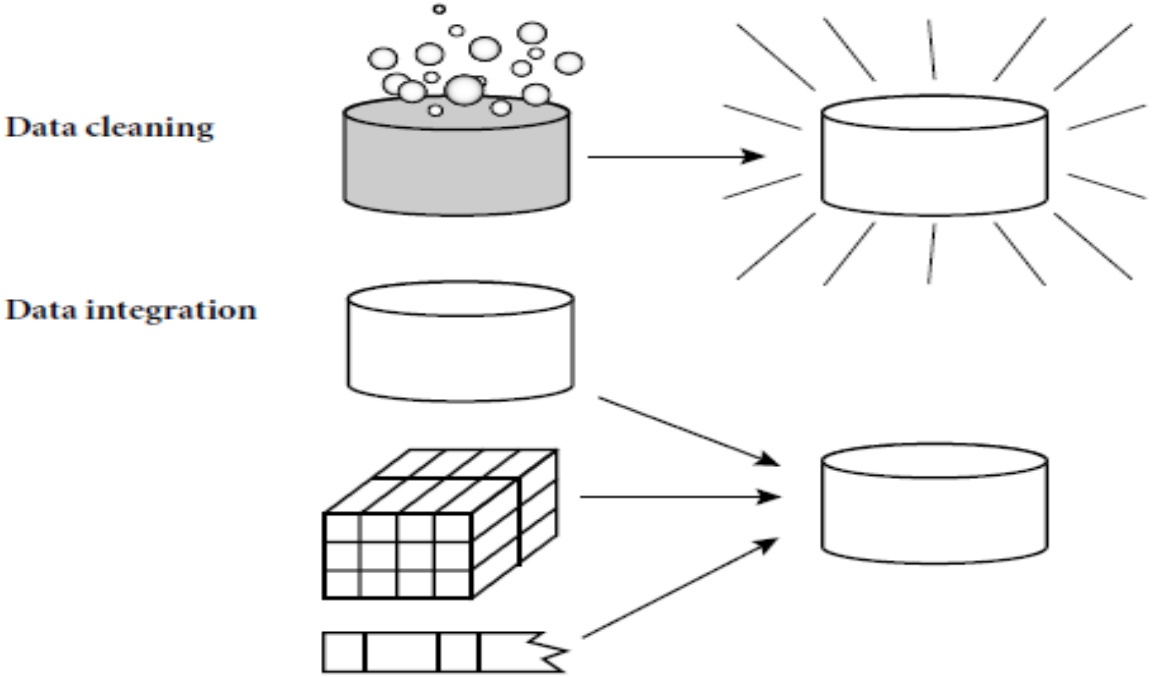
# Tiêu chuẩn đánh giá

- Chính xác
- Hoàn chỉnh
- Nhất quán
- Hợp thời
- Đáng tin
- Có giá trị
- Có thể hiểu được
- Có thể dùng được

# Các nhiệm vụ chính

- Làm sạch dữ liệu (data clearning)
- Tích hợp dữ liệu (data integration)
- Biến đổi dữ liệu (data transformation)
- Thu gọn dữ liệu (data reduction)

# Các nhiệm vụ chính (tt)



# Nội dung

- Đặc điểm chung của dữ liệu
- Vì sao phải tiền xử lý dữ liệu?
- Tóm tắt dữ liệu
- Làm sạch dữ liệu
- Tích hợp & biến đổi dữ liệu
- Thu gọn dữ liệu

# Tóm tắt dữ liệu

- Cung cấp cái nhìn chung nhất về dữ liệu
- Xác định các nhiễu (noisy) và phần tử cá biệt (outlier)
- Nội dung:
  - Đo lường giá trị trung tâm
  - Đo lường sự phân tán
  - Các dạng đồ thị

# Các loại độ đo

- Độ đo phân bố (distributive measure)
  - VD: sum, count, min, max …
- Độ đo đại số (algebraic measure)
  - VD: average ..
- Độ đo nguyên (holistic measure)
  - VD: median …

# Đo lường giá trị trung tâm

- Trung bình (mean)
  - Trung bình cộng (algebraic measure)

$$\bar{x} = \frac{1}{N}\sum_{i=1}^{N} x_i = \frac{x_1 + x_2 + \ldots + x_N}{N}$$

  - Trung bình cộng có trọng số (weighted arithmetic mean)

$$\bar{x} = \frac{\sum_{i=1}^{N} w_i x_i}{\sum_{i=1}^{N} w_i} = \frac{w_1 x_1 + w_2 x_2 + \ldots + w_N x_N}{w_1 + w_2 + \ldots + w_N}$$

# Đo lường giá trị trung tâm(tt)

- Trung bình (mean)
  - Phẩn tử cá biệt ?
    - Age: 3, 13, 15, 16, 19, 20, 21, 25, 70
  - Trimmed mean:
    - VD: 3, 13, 15, 16, 19, 20, 21, 25, 70
      → 13, 15, 16, 19, 20, 21, 25
      → mean

# Đo lường giá trị trung tâm(tt)

- Trung vị (median)
  - Cách 1
    - N giá trị phân biệt được sắp tăng dần
    - Nếu N lẻ, median là giá trị chính giữa
    - Nếu N chẵn, median là trung bình của 2 giá trị chính giữa
    - VD:
      - Age: 13, 15, 16, 19, 20, 21, 25

        → median = 19
      - Age: 13, 15, 16, 18, 20, 22, 25, 30

        →median = (18 + 20) / 2 = 19

# Đo lường giá trị trung tâm(tt)

- Trung vị (median)
  - Cách 2

| Age | Frequency |
|---|---|
| 1-5 | 200 |
| 5-15 | 450 |
| 15-20 | 300 |
| 20-50 | 1500 |
| 50-80 | 700 |

$$median = L_1 + \left( \frac{N/2 - (\sum freq)_l}{freq_{median}} \right) width$$

- $L_1$: chặn dưới của khoảng trung vị (median interval)
- N: Tổng tần số
- $(\sum freq)_l$: tổng tần số của tất cả các khoảng thấp hơn khoảng trung vị
- $freq_{median}$: tần số của khoảng trung vị
- width: chiều rộng cua khoảng trung vị

# Đo lường giá trị trung tâm(tt)

- VD:
  - $freq_{median} = 450$
  - ➔ Khoảng trung vị: 5-15
  - width = 10
  - $L_1 = 5$
  - N = 3150
  - $(\sum freq)_l = freq_{1\text{-}5} = 200$

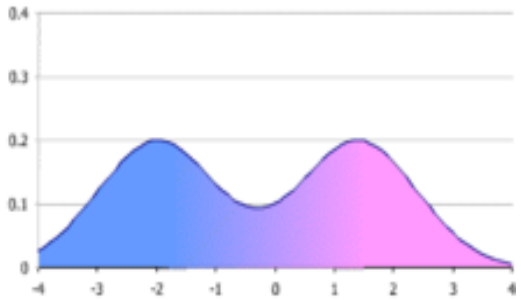
| Age | Frequency |
|---|---|
| 1-5 | 200 |
| 5-15 | 450 |
| 15-20 | 300 |
| 20-50 | 1500 |
| 50-80 | 700 |

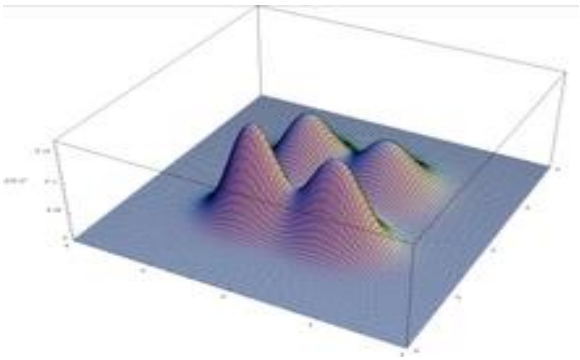
$$median = 5 + \left( \frac{3150/2 - 200}{450} \right) 10 \approx 35$$

# Đo lường giá trị trung tâm(tt)

- Mode:
  - Giá trị xảy ra với tần số lớn nhất
  - Unimodal, bimodal, trimodal, multimodal
  - Unimodal:

$$mean - mode = 3 \times (mean - median)$$



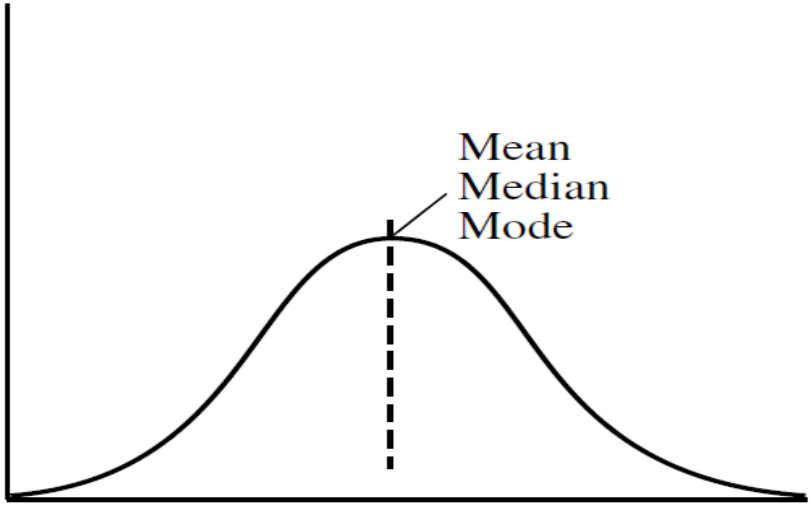



# Đo lường giá trị trung tâm(tt)

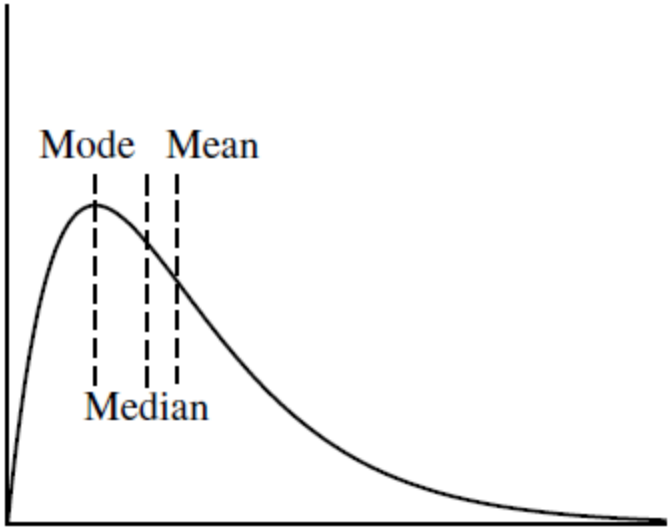
- Midrange:
  - *midrange = (min + max) / 2*
  - VD:
    - Age: 13, 15, 16, 19, 20, 21, 25

      → midrange = (13 + 25) / 2 = 19

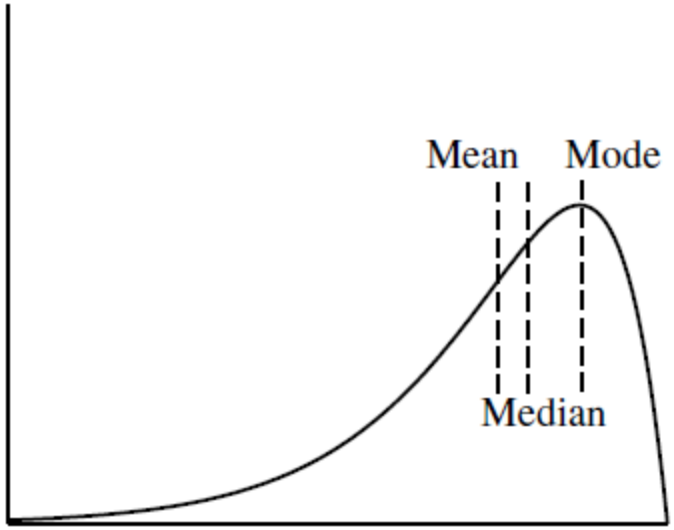
# Đo lường giá trị trung tâm(tt)




(a) symmetric data



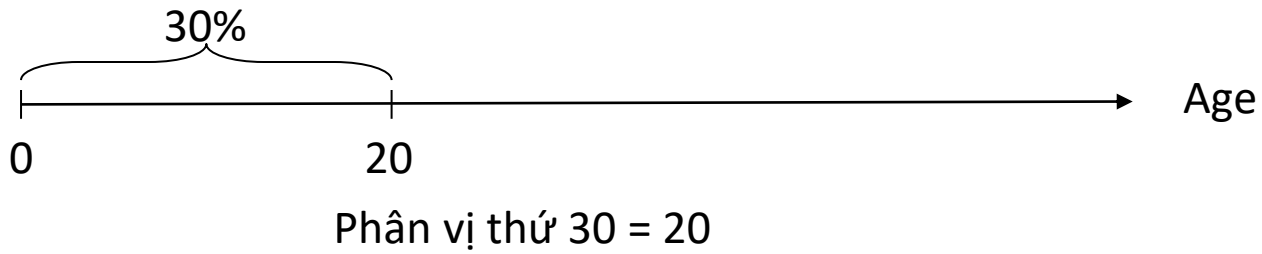

(b) positively skewed data




(c) negatively skewed data

# Đo lường sự phân tán dữ liệu

- Phân vị (Percentile)
  - Phân vị thứ k: giá trị $x_i$ có k% dữ liệu bé hơn hoặc bằng $x_i$



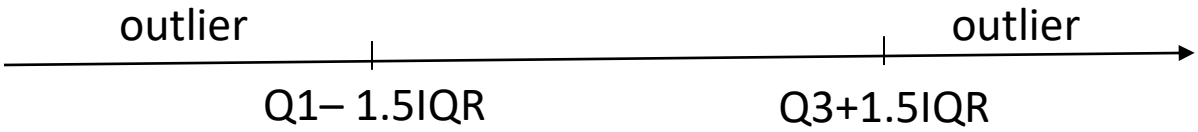

  - Median là phân vị thứ mấy?
    - A. 40
    - B. 50
    - C. 60
    - D. 70

# Đo lường sự phân tán dữ liệu(tt)

- Tứ phân vị (quartiles):
  - Q1: phân vị thứ 25
  - Q2: trung vị
  - Q3: phân vị thứ 75
- Dãy phân vị (Interquartile range):
  - IQR = Q3 – Q1
- Five – number summary
  - Minumum, $Q_1$, Median, $Q_3$, Maximum

# Đo lường sự phân tán dữ liệu(tt)

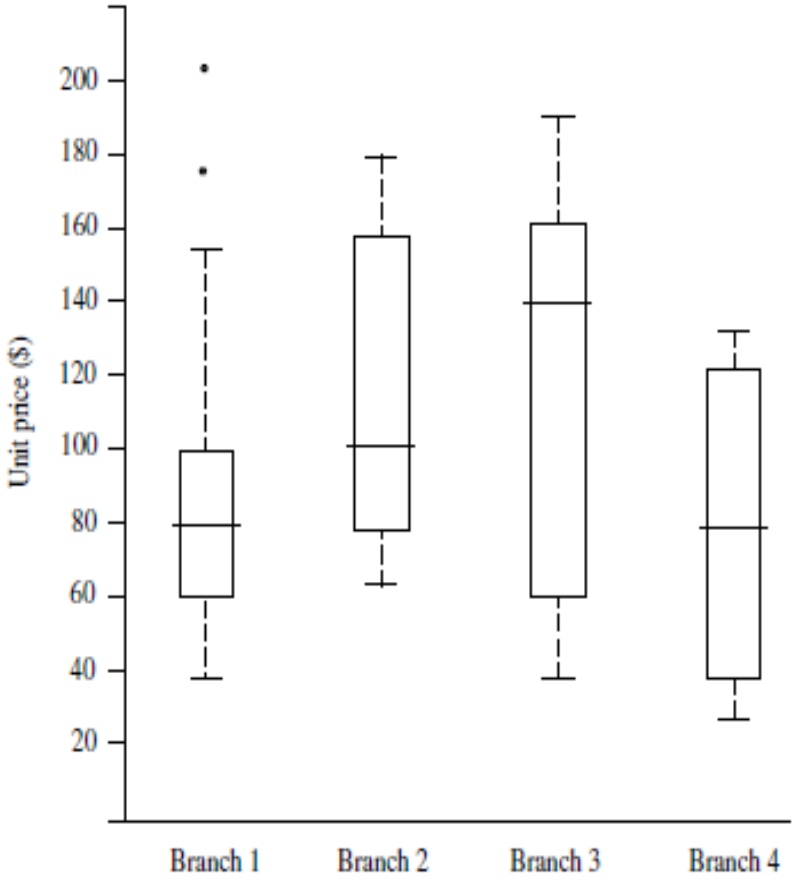
- Một giá trị là giá trị cá biệt (outlier) khi nó lớn hơn 1.5 x IQR + Q3 hay nhỏ hơn Q1 – 1.5 x IQR



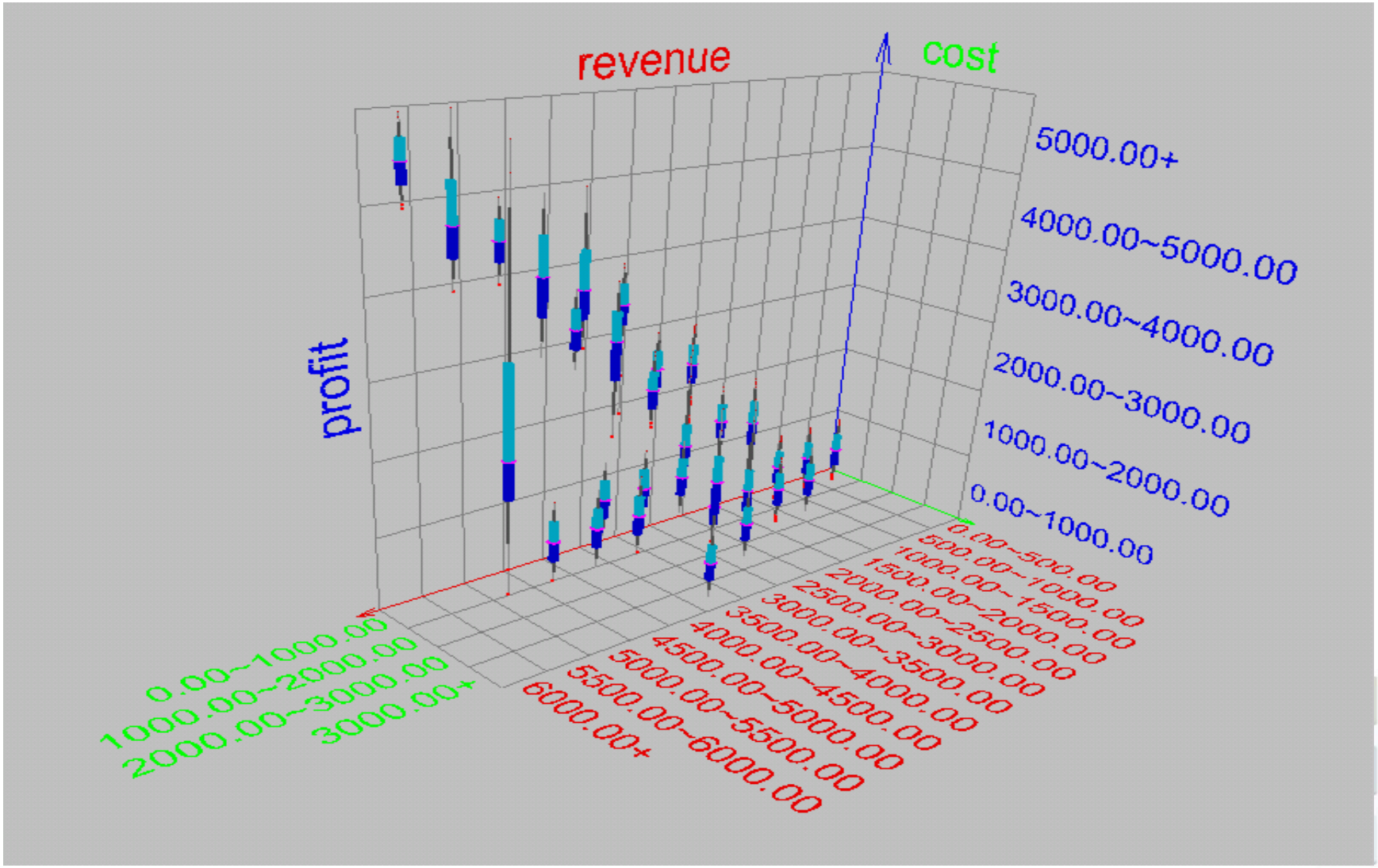

- VD:
  - Q1 = 60, Q3 = 100
  - ➔ IQR = 100 – 60 = 40
  - Xét giá trị 175: 175 > 1.5x40 + 100
  - ➔ 175 là outlier

# Boxplots

- Boxplots biểu diễn Five-number summary
- Đầu, cuối hộp là Q1 và Q3, chiều cao hộp = IQR
- Median là một đường nằm ngang
- Hai đường thẳng bên ngoài kéo dài đến min và max



# 3-D Boxplots


revenue
cost
profit
5000.00+
4000.00~5000.00
3000.00~4000.00
2000.00~3000.00
1000.00~2000.00
0.00~1000.00
0.00~500.00
500.00~1000.00
1000.00~1500.00
1500.00~2000.00
2000.00~2500.00
2500.00~3000.00
3000.00~3500.00
3500.00~4000.00
4000.00~4500.00
4500.00~5000.00
5000.00~5500.00
5500.00~6000.00
6000.00+
0.00~1000.00
1000.00~2000.00
2000.00~3000.00
3000.00+

# Đo lường sự phân tán dữ liệu(tt)

- Phương sai và độ lệch chuẩn:
  - Phương sai:

$$\sigma^2 = \frac{1}{N-1}\sum_{i=1}^{N}(x_i - \bar{x})^2$$

  - Độ lệch chuẩn: $\sigma$

# Đo lường sự phân tán dữ liệu(tt)

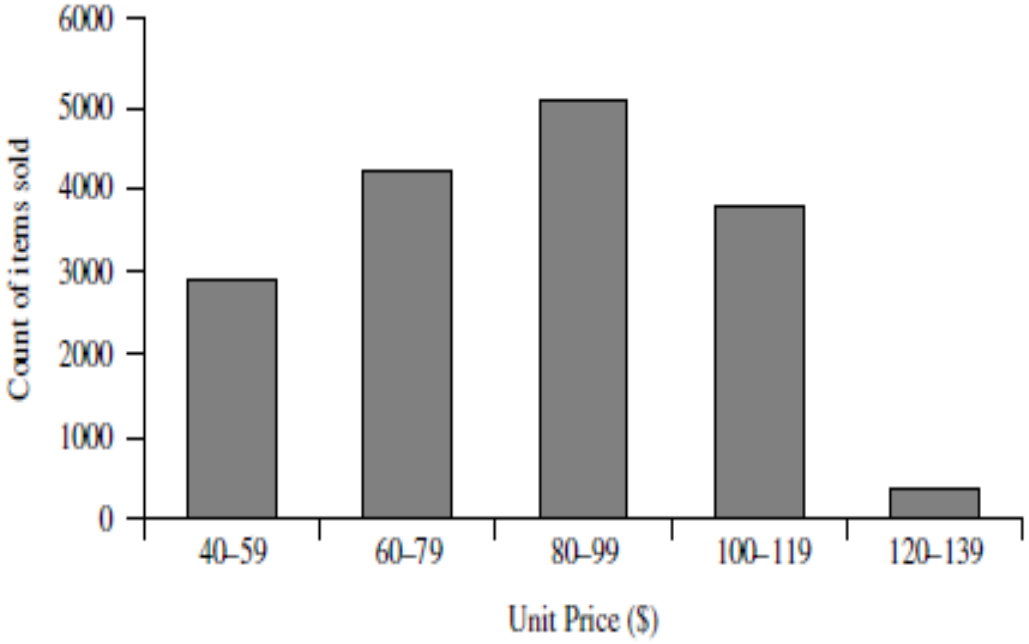
- Độ lệch chuẩn là thước đo độ lệch của từng giá trị với giá trị trung bình của nó
- Phương sai luôn lớn hơn hoặc bằng 0, bằng 0 khi mọi giá trị đều bằng nhau

# Các dạng đồ thị

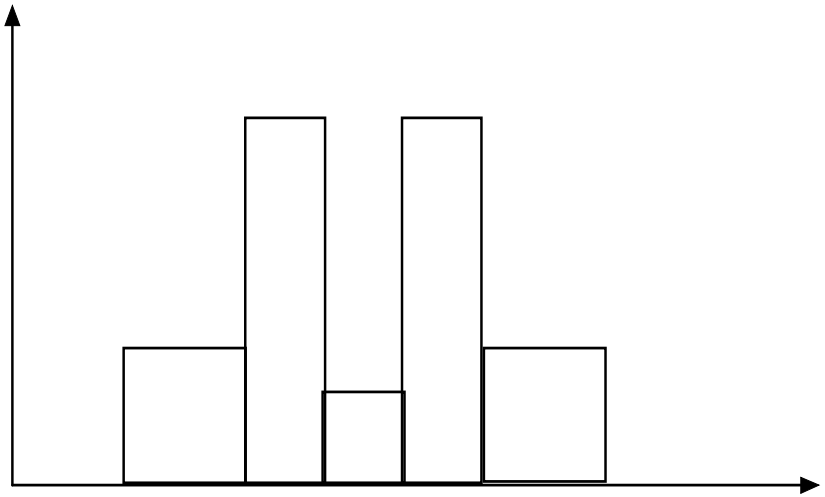
- Boxplots
- Histogram
- Đồ thị Quantile
- Đồ thị QQ
- Đồ thị phân tán (Scatter plot)
- Đường cong Loess

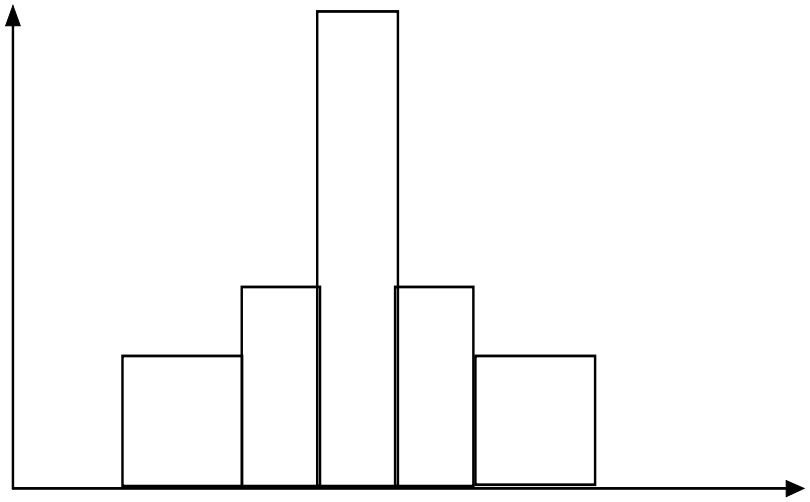
# Histogram

- Histogram: biểu đồ tần suất
- Trục x thể hiện giá trị
- Trục y thể hiện tần số xuất hiện của giá trị đó



# Histograms vs Boxplots



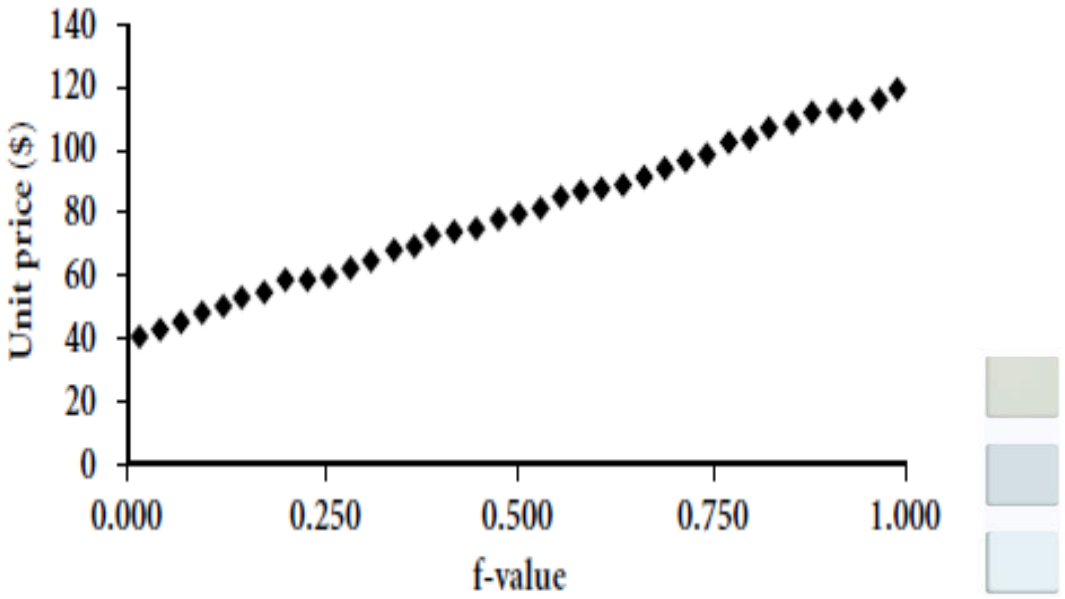


- 2 Histograms này có thể có cùng 1 Boxplots
  - Có cùng min, max, median, Q1, Q3
- Nhưng sự phân bố dữ liệu là khác nhau

# Đồ thị Quantile

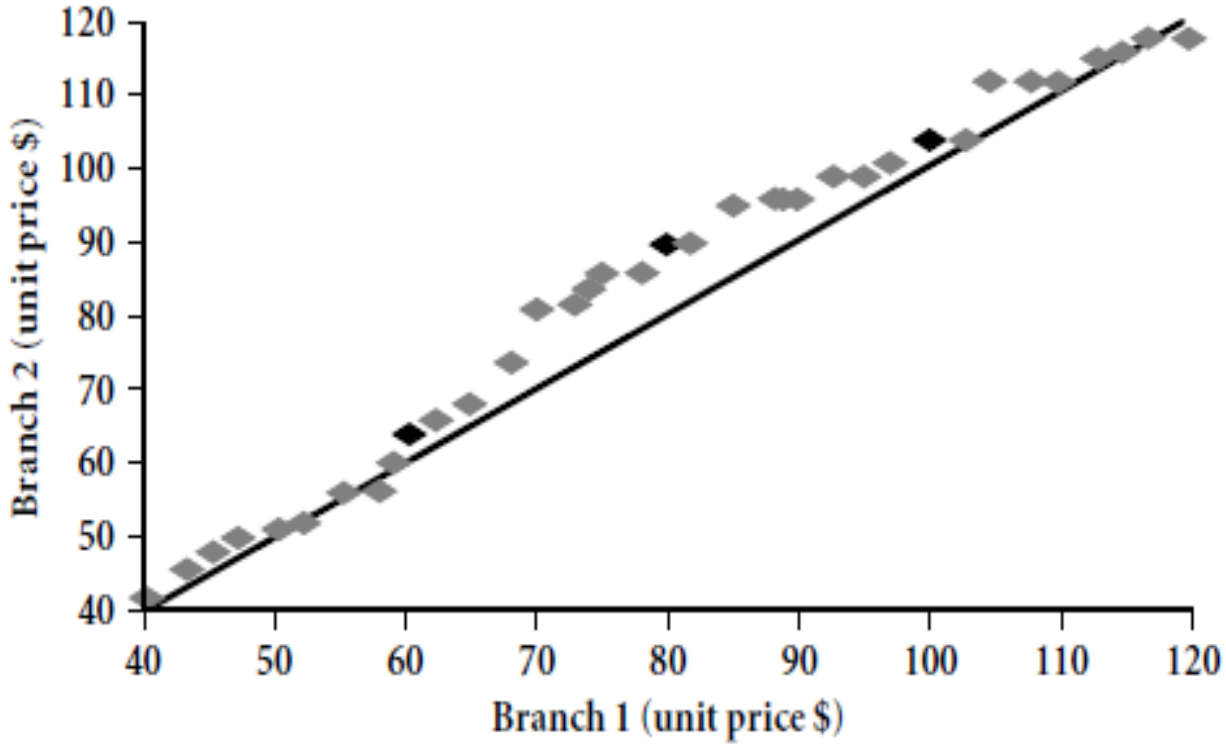
- Thể hiện toàn bộ sự phân bố của dữ liệu
- Thông tin trên đồ thị:
  - Mỗi giá trị $x_i$ được xắp tăng dần và đi liền với $f_i$
  - Tại mỗi điểm, có xấp xỉ 100 $f_i$% dữ liệu có giá trị nhỏ hơn hay bằng $x_i$

| *Unit price ($)* | *Count of items sold* |
|---|---|
| 40 | 275 |
| 43 | 300 |
| 47 | 250 |
| .. | .. |
| 74 | 360 |
| 75 | 515 |
| 78 | 540 |
| .. | .. |
| 115 | 320 |
| 117 | 270 |
| 120 | 350 |



# Đồ thị Quantile-Quantile (Q-Q)

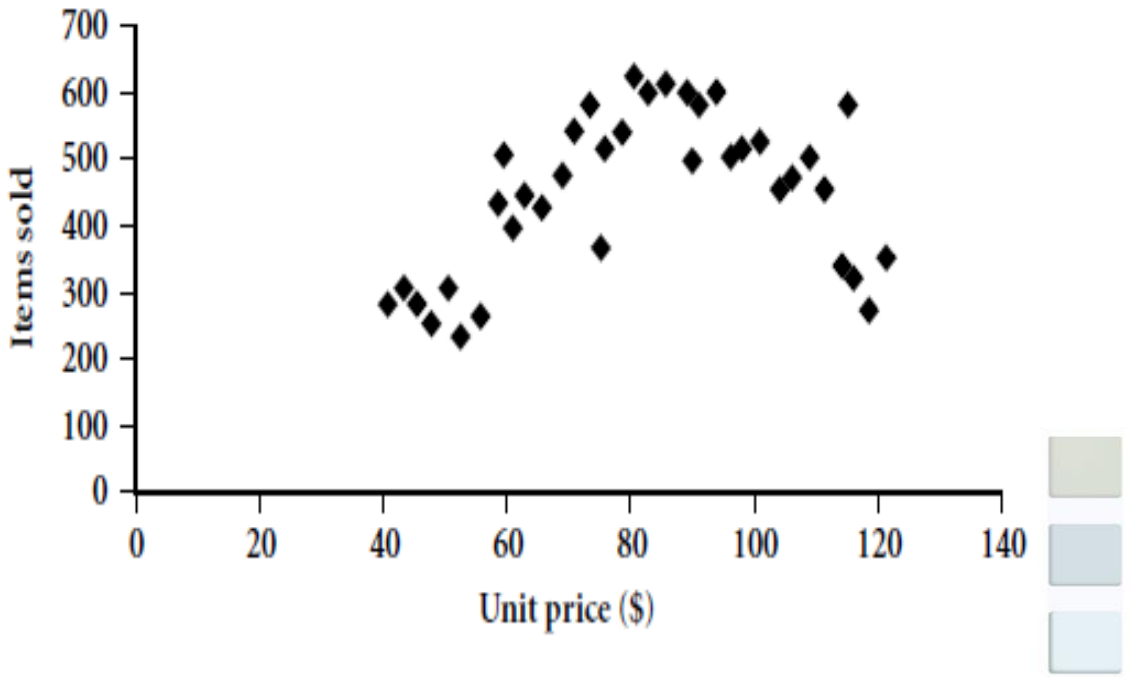
- Kết hợp 2 đồ thị Quantile
- Thể hiện, so sánh xác suất phân bố giữa các thành phần



# Đồ thị phân tán (Scatter plot)

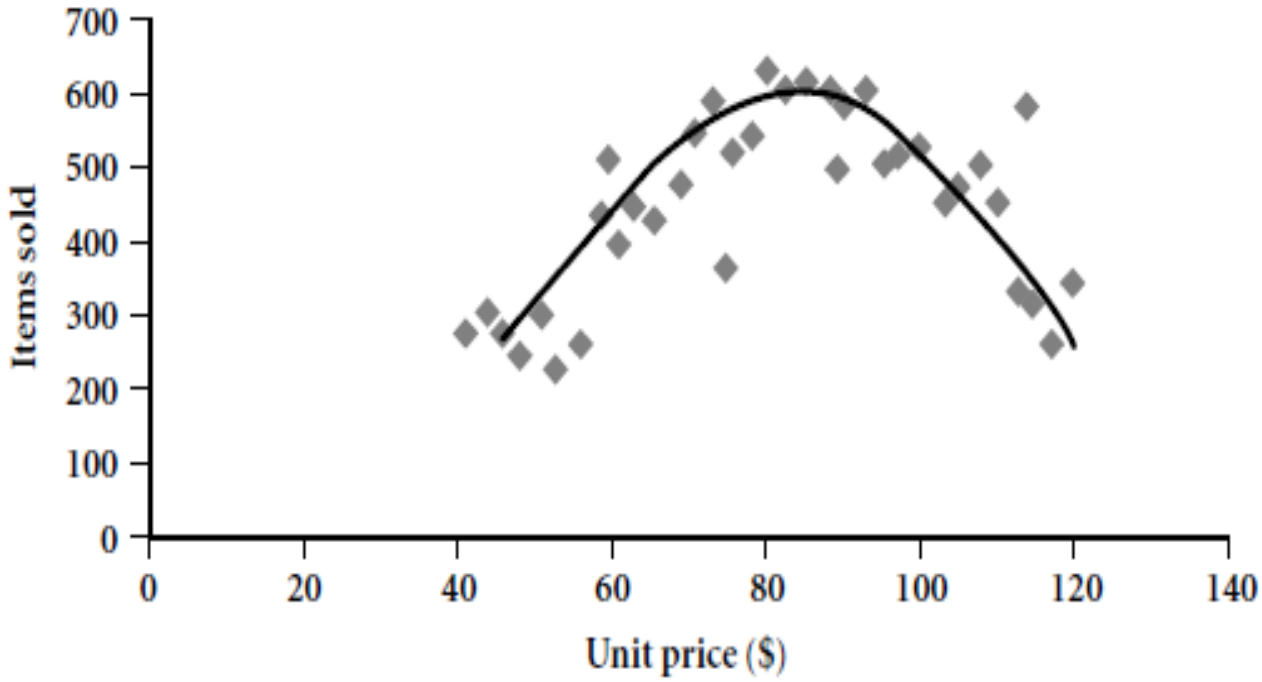
- Mô tả tổng quan (điểm, outlier) về phân bố của dữ liệu có thể thể hiện ở dạng 2 biến
- Mỗi điểm là một cặp giá trị

| *Unit price ($)* | *Count of items sold* |
|---|---|
| 40 | 275 |
| 43 | 300 |
| 47 | 250 |
| .. | .. |
| 74 | 360 |
| 75 | 515 |
| 78 | 540 |
| .. | .. |
| 115 | 320 |
| 117 | 270 |
| 120 | 350 |

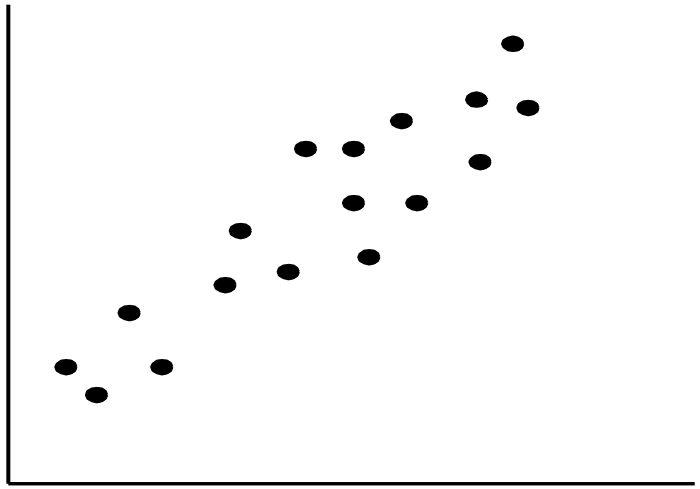


# Đường cong Loess (Loess curve)

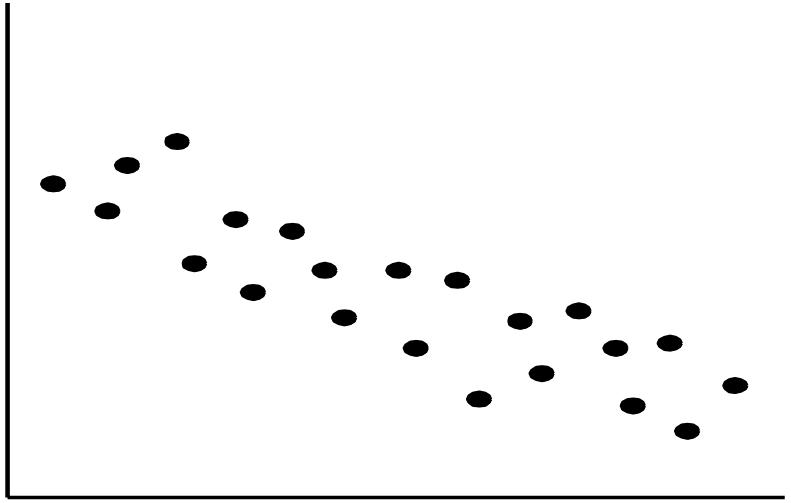
- Bổ sung thêm một đường cong trơn vào đồ thị phân tán để có cái nhìn tốt hơn về sự phụ thuộc



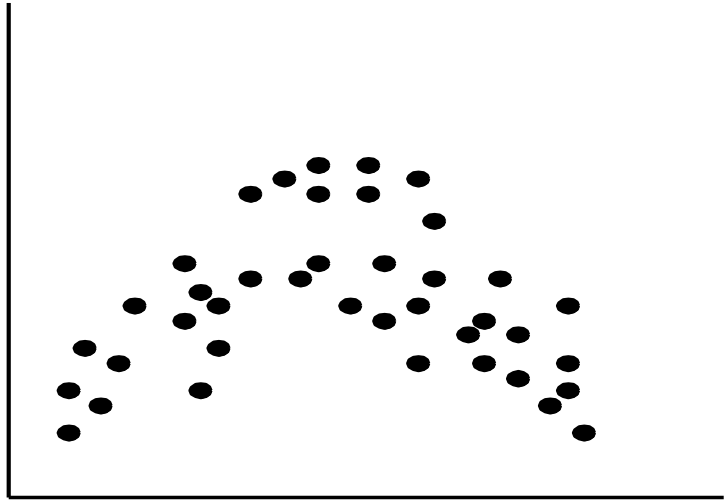
# Dữ liệu tương quan



Hình 1



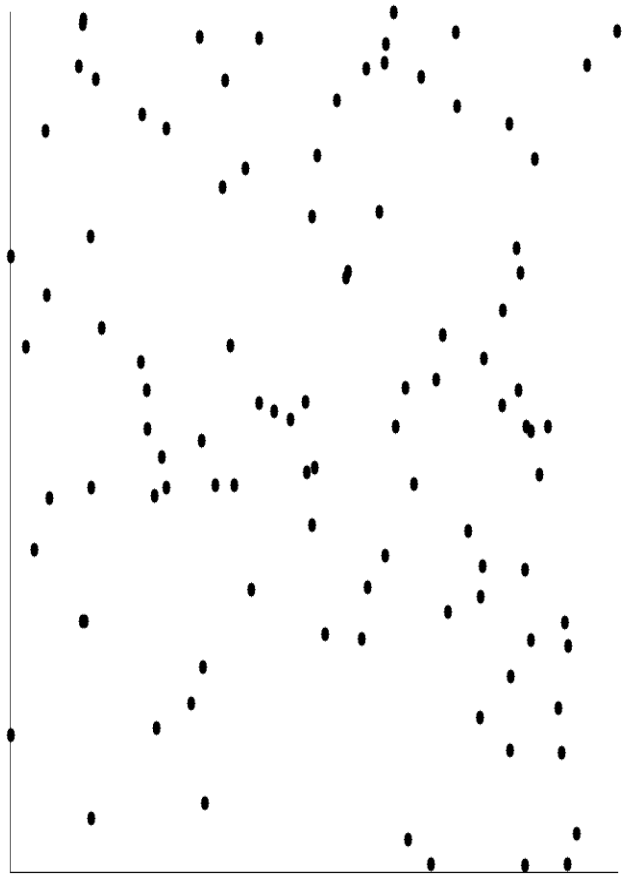
Hình 2

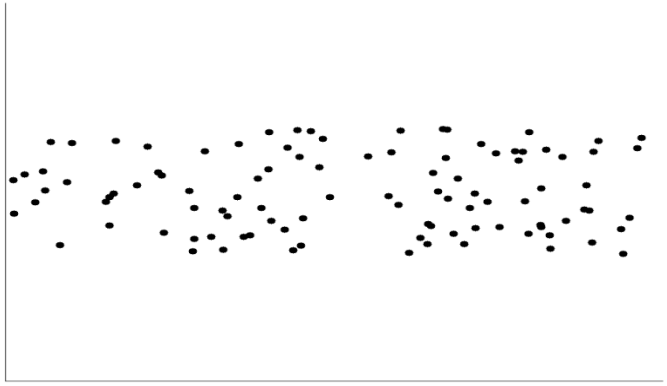


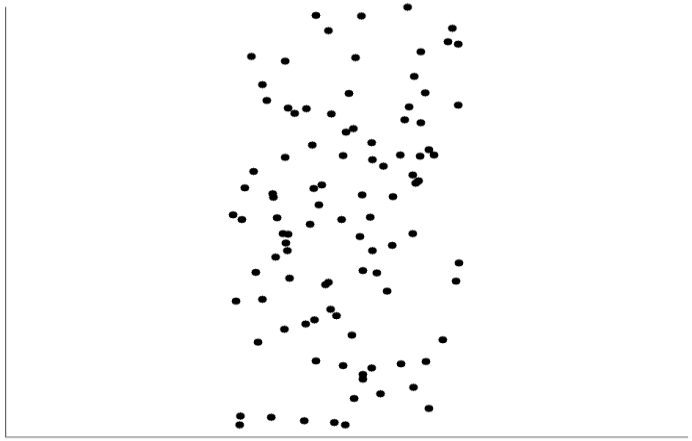
Hình 3

- H1. Dữ liệu tương quan theo kiểu “positive”
- H2. Dữ liệu tương quan theo kiểu “negative”
- H3. Dữ liệu tương quan theo cả 2 dạng

# Dữ liệu không tương quan



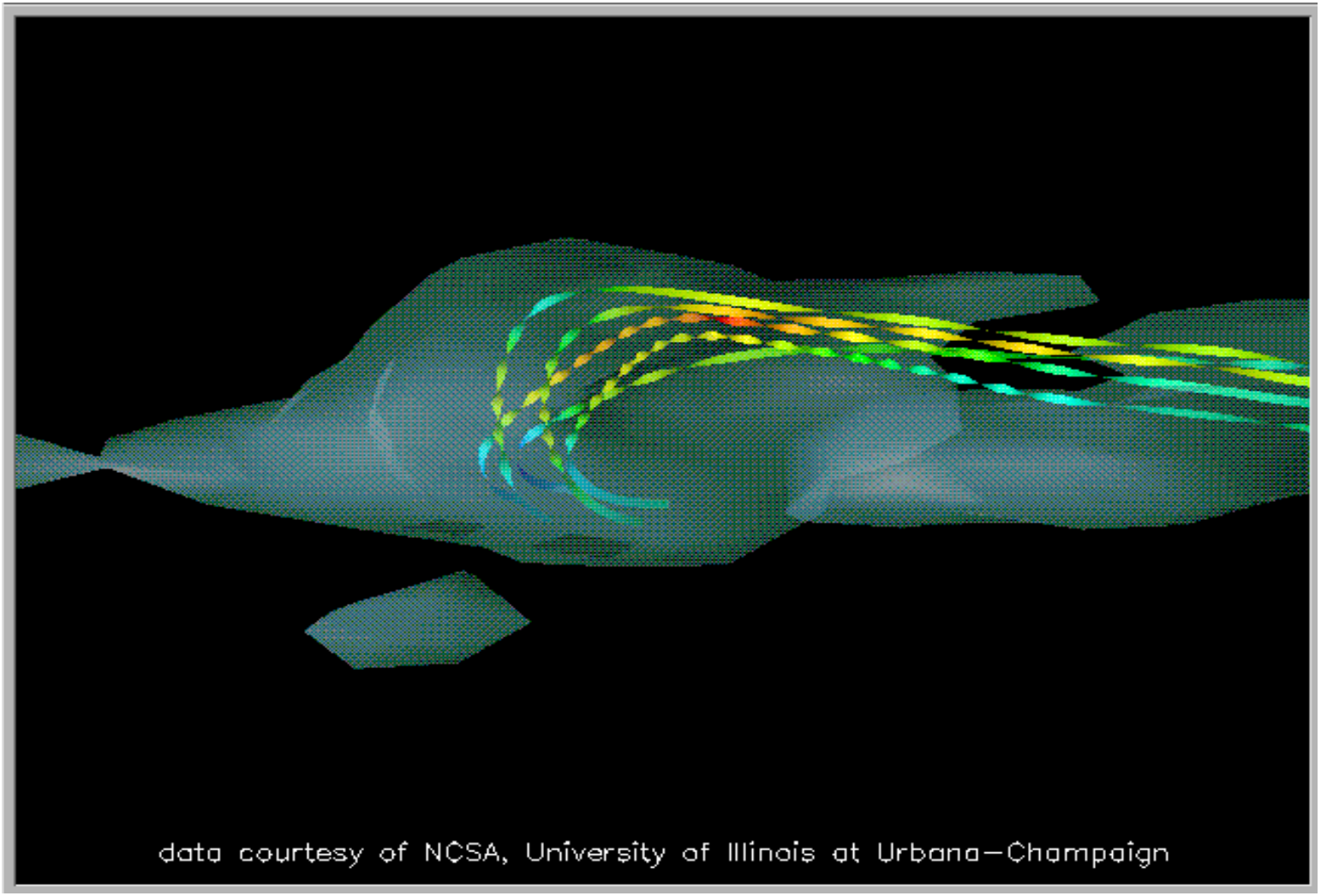




# Mô hình hóa dữ liệu (data visualization)

- Tại sao cần mô hình hóa?
  - Ánh xạ dữ liệu sang một không gian mới để có cái nhìn toàn diện hơn
  - Thể hiện mối tương quan, cấu trúc, ràng buộc  của dữ liệu
  - Giúp tìm ra các phần dữ liệu đáng quan tâm
- Các phương pháp:
  - Kỹ thuật hình học (Geometric techniques)
  - Kỹ thuật dựa trên biểu tượng(Icon-based techniques)
  - Kỹ thuật phân cấp (Hierarchical techniques)

# Mô hình hóa dữ liệu

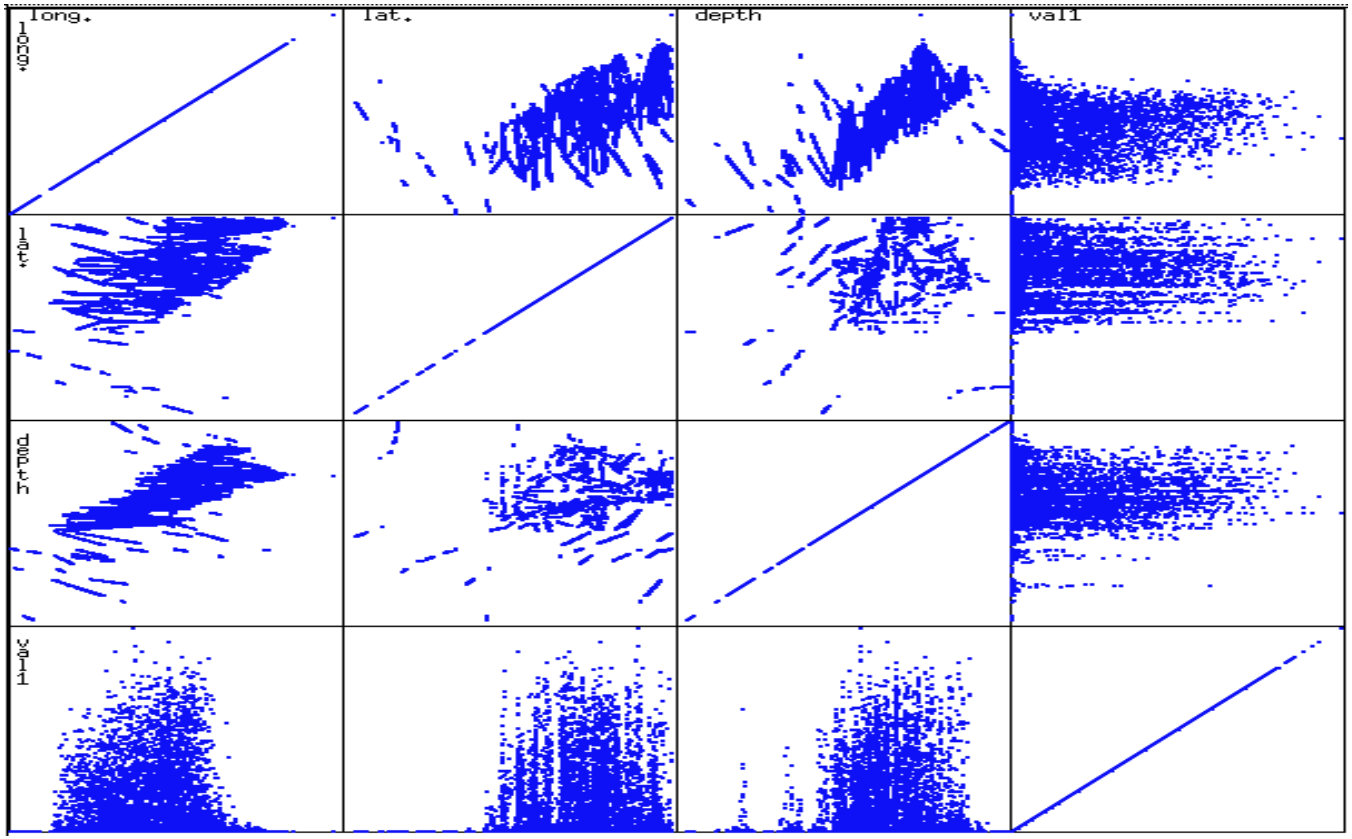


# Kỹ thuật hình học

- Mô hình hóa bằng các phép biến đổi hình học  để chiếu dữ liệu vào không gian thể hiện
- Phương pháp:
  - Landscapes
  - Scatterplot matrices
  - Parallel coordinates
  - ...

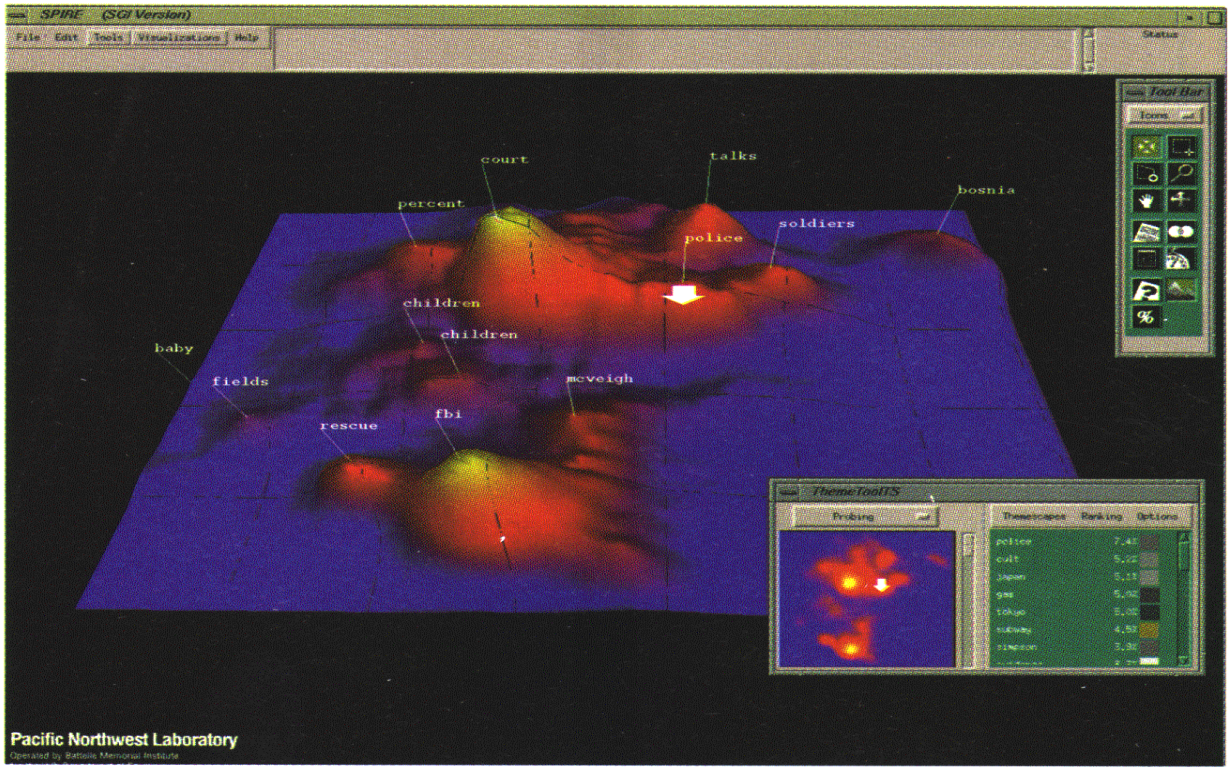
# Scatterplot Matrices

- Kết hợp các đồ thị phân tán (Scattet plot) thành một ma trận



# Landscapes

- Mô hình hóa dữ liệu bằng cách phối cảnh
- Dữ liệu phải được chuyển về không gian (2D hay 3D) mà vẫn lưu trữ được đặc điểm của dữ liệu

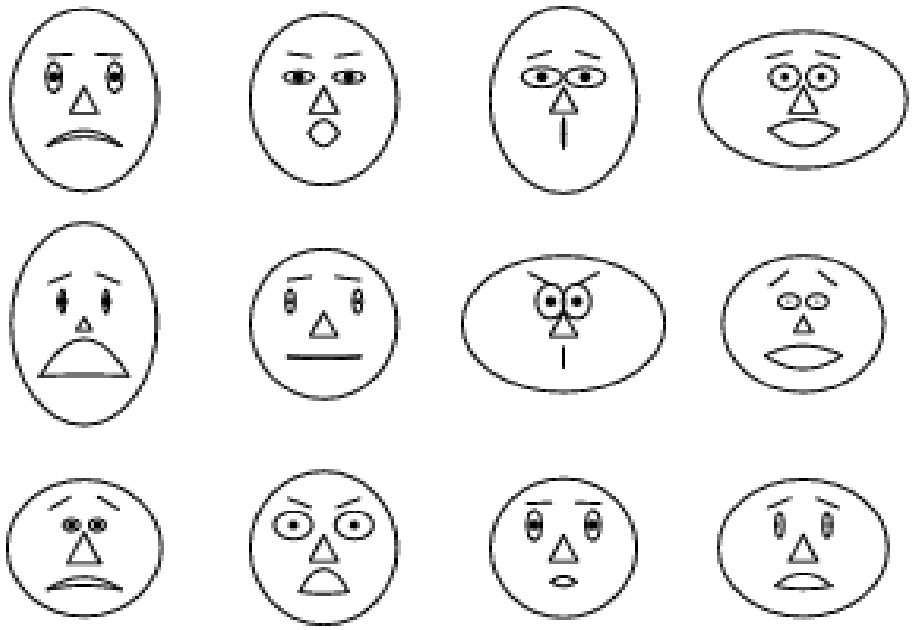


# Kỹ thuật dựa trên biểu tượng

- Mô hình hóa dựa vào đặc trưng của biểu tượng
- Phương pháp:
  - Chernoff Faces
  - Stick Figures
  - Shape Coding
  - …

# Chernoff Faces

- Một phương pháp biểu diễn dự liệu 2 chiều dựa vào các đặc trưng nổi bật của dữ liệu

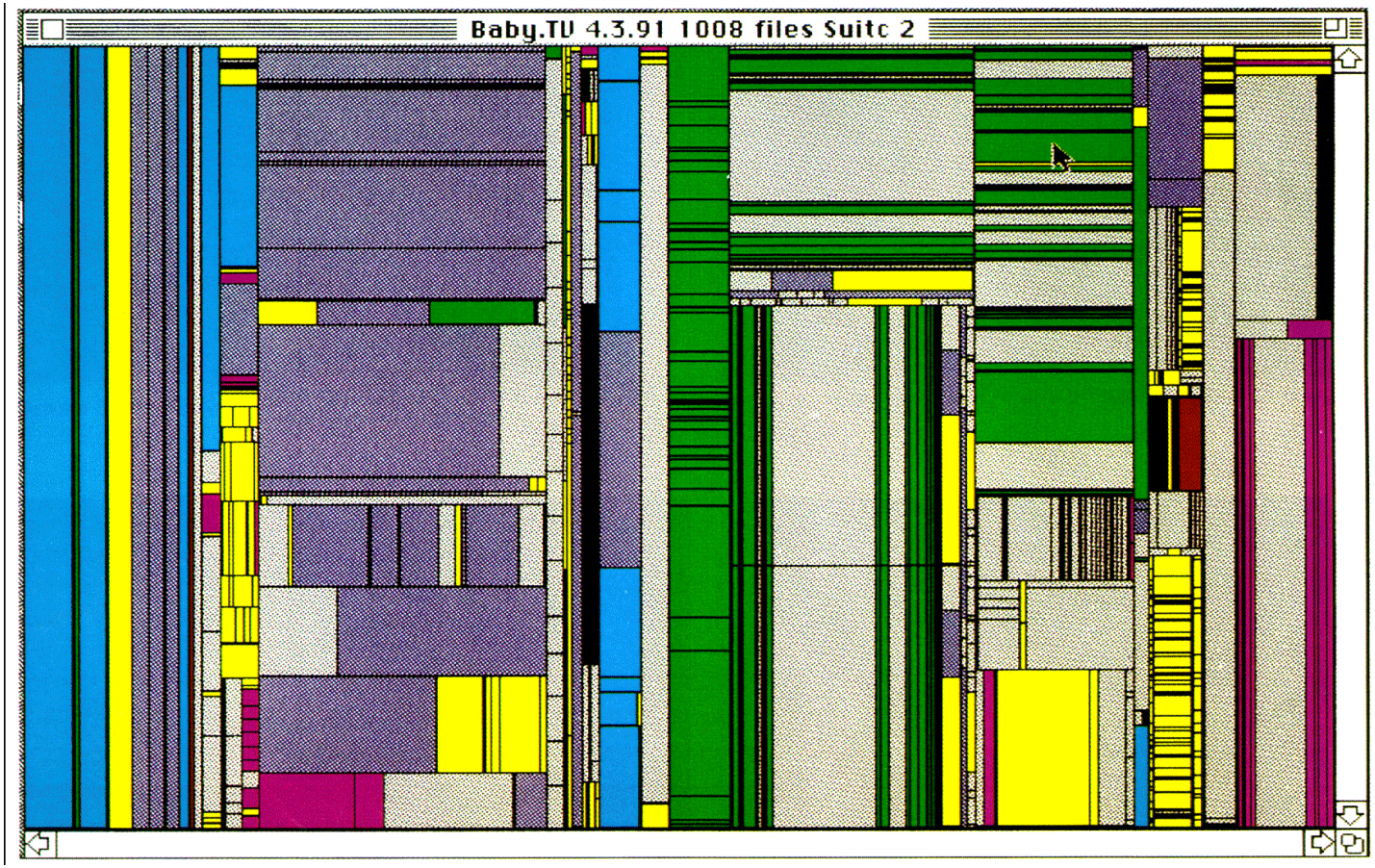


# Phương pháp phân cấp

- Mô hình hóa dữ liệu bằng cách phân cấp ra thành từng phần nhỏ
- Methods
  - Treemap
  - Dimensional Stacking
  - Worlds-within-Worlds

# Tree-Map

- Lấp đầy đồ thị bằng các phần thuộc về một vùng dữ liệu có sẵn



# Nội dung

- Đặc điểm chung của dữ liệu
- Vì sao phải tiền xử lý dữ liệu?
- Tóm tắt dữ liệu
- Làm sạch dữ liệu
- Tích hợp & biến đổi dữ liệu
- Thu gọn dữ liệu

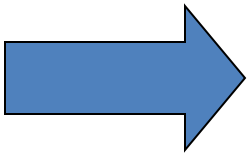
# Làm sạch dữ liệu

- Tầm quan trọng
  - "Data cleaning is one of the three biggest problems in data warehousing"—Ralph Kimball
  - "Data cleaning is the number one problem in data warehousing"—DCI survey
- Nội dung
  - Làm đầy giá trị bị thiếu
  - Làm mịn nhiễu, giá trị cá biệt và sự không nhất quán
  - Qúa trình làm sạch dữ liệu

# Làm đầy giá trị bị thiếu

- Bỏ qua những bộ giá trị bị thiếu
- Làm đầy bằng tay (khả thi ?)
- Làm đầy bằng hằng số toàn cục
  - Unknow
  - $\infty$

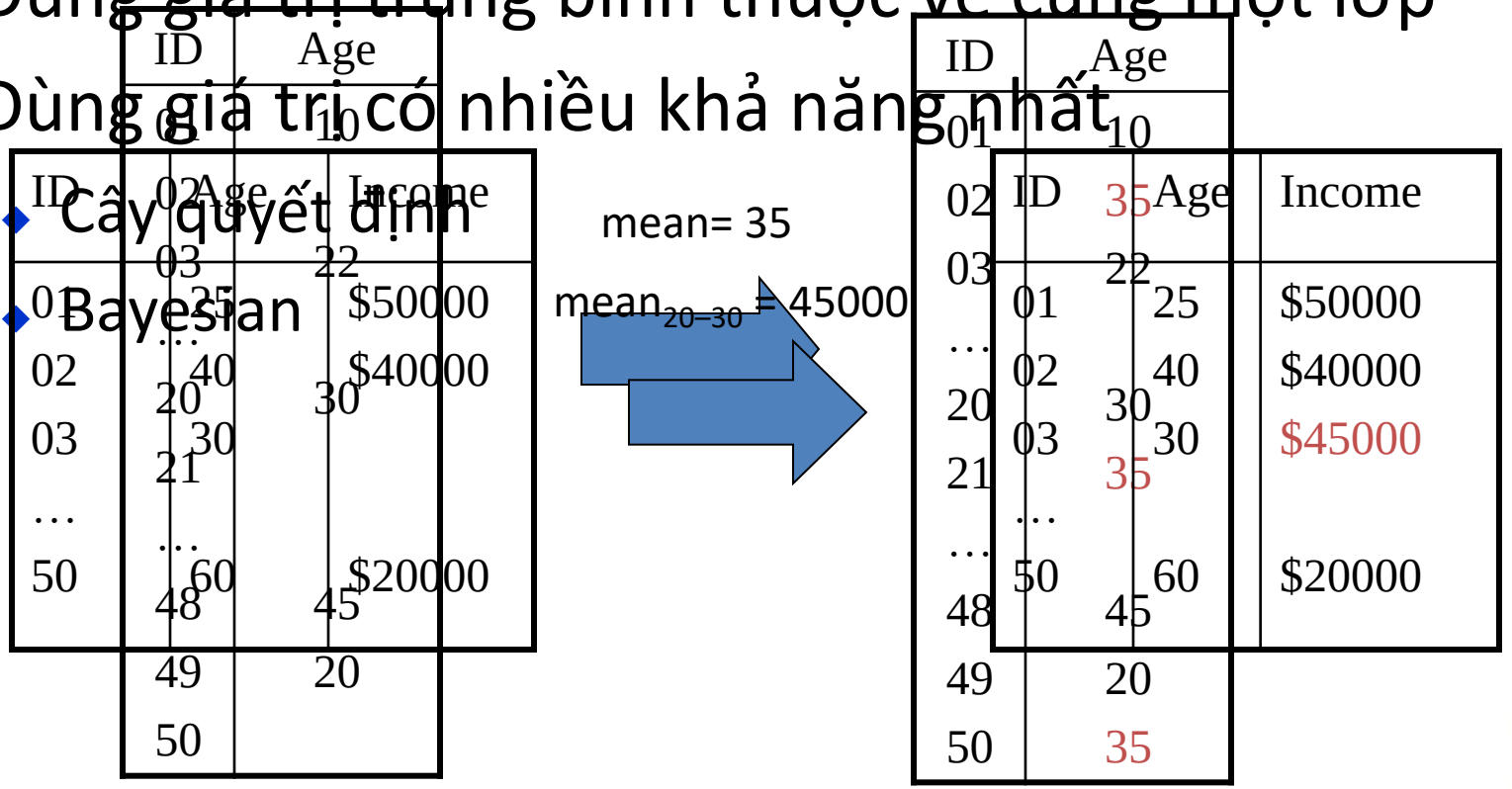
| ID | Age |
|---|---|
| 01 | 10 |
| 02 | |
| 03 | 22 |
| … | |
| 20 | 30 |
| 21 | |
| … | |
| 48 | 45 |
| 49 | 20 |
| 50 | |



| ID | Age |
|---|---|
| 01 | 10 |
| 02 | Unknown |
| 03 | 22 |
| … | |
| 20 | 30 |
| 21 | Unknown |
| … | |
| 48 | 45 |
| 49 | 20 |
| 50 | Unknown |

# Làm đầy giá trị bị thiếu

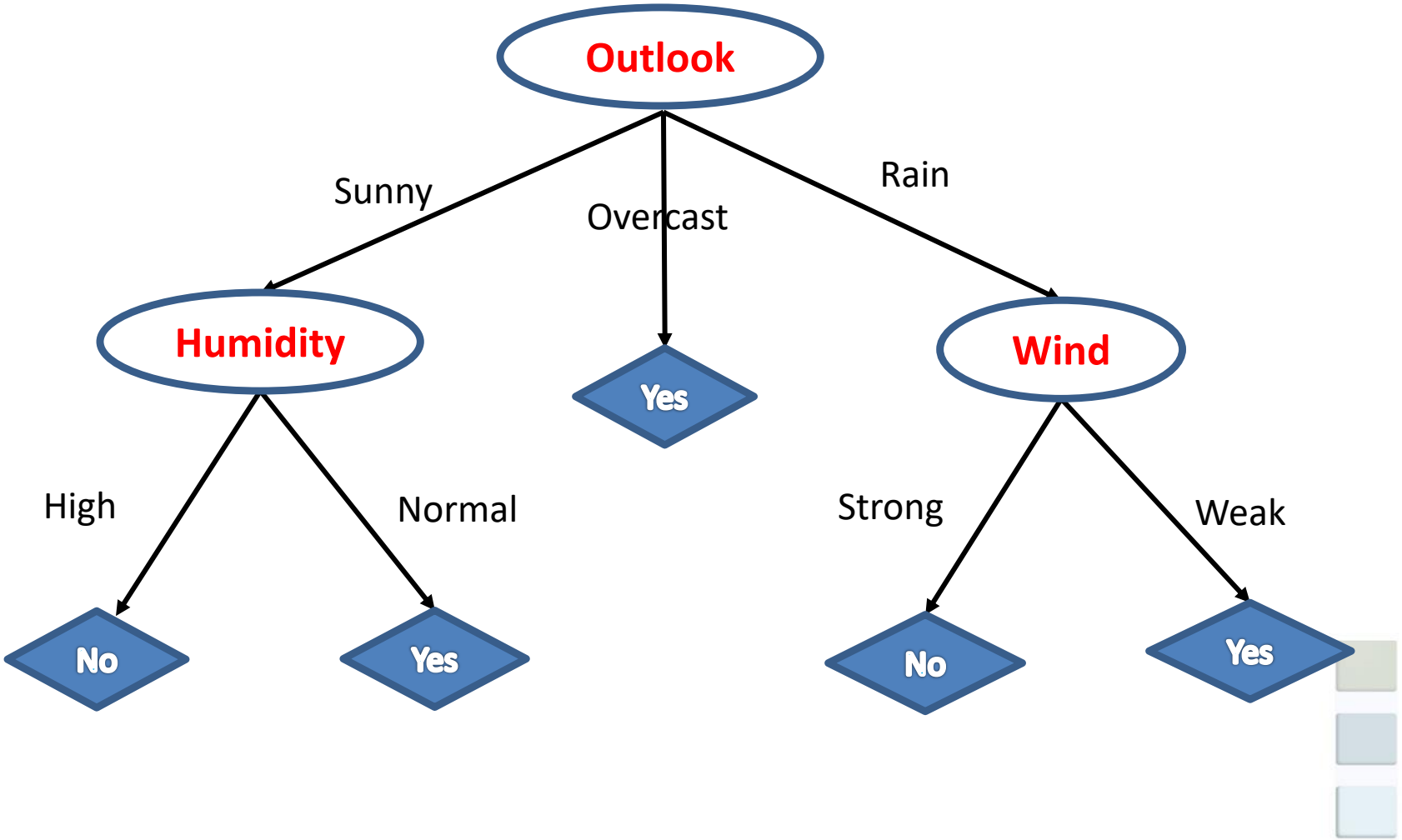
- Dùng giá trị trung bình (mean) toàn cục
- Dùng giá trị trung bình thuộc về cùng một lớp
- Dùng giá trị có nhiều khả năng nhất
  - Cây quyết định
  - Bayesian



# Làm đầy giá trị bị thiếu

| Day | Outlook | Temp | Humidity | Wind | Play? |
|---|---|---|---|---|---|
| 1 | Sunny | Hot | High | Weak | No |
| 2 | Sunny | Hot | High | Strong | No |
| 3 | Overcast | Hot | High | Weak | Yes |
| 4 | Rain | Mild | High | Weak | Yes |
| 5 | Rain | Cool | Normal | Weak | Yes |
| 6 | Rain | Cool | Normal | Strong | No |
| 7 | Overcast | Cool | Normal | Strong | Yes |
| 8 | Sunny | Mild | High | Weak | No |
| 9 | Sunny | Cool | Normal | Weak | Yes |
| 10 | Rain | Mild | Normal | Weak | Yes |
| 11 | Sunny | Mild | Normal | Strong | Yes |
| 12 | Overcast | Mild | High | Strong | Yes |
| 13 | Overcast | Hot | Normal | Weak | Yes |
| 14 | Rain | Mild | High | Strong | No |
| **15** | **Sunny** | **Hot** | **High** | **Weak** | **?** |

# Làm đầy giá trị bị thiếu



# Làm mịn nhiễu

- Bining
  - Là phương pháp làm mịn nhiễu cục bộ
  - Dữ liệu được sắp xếp vào các bin và được làm mịn tại từng bin
- Phương pháp
  - Bin means
  - Bin median
  - Bin boundaries
  - …

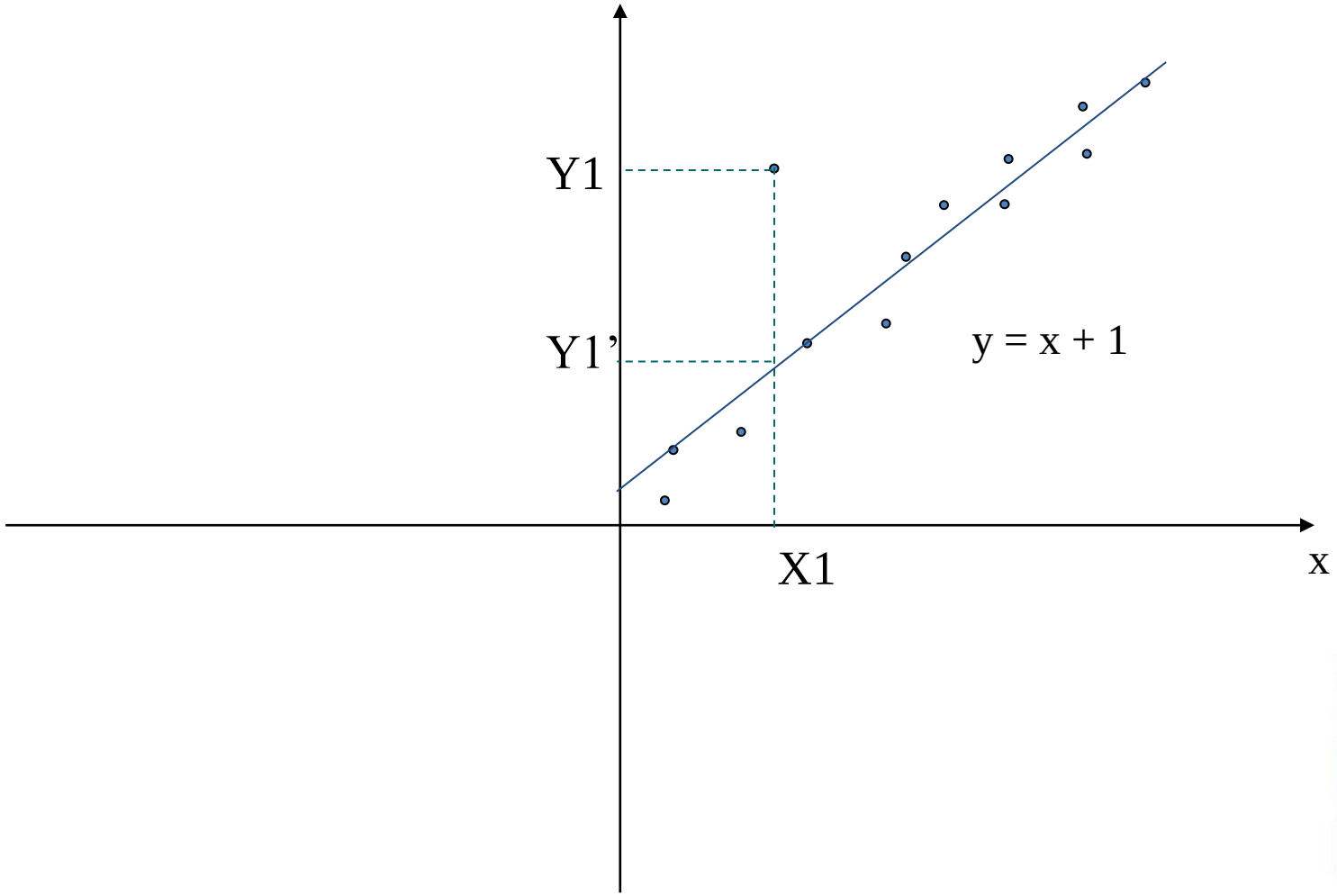
# Làm mịn nhiễu

- Ví dụ:
  - Dữ liệu về giá đã sắp xếp: 4, 8, 9, 15, 21, 21, 24, 25, 26, 28, 29, 34
  - Phân chia vào các bin có tần số bằng nhau:
    - Bin 1: 4, 8, 9, 15
    - Bin 2: 21, 21, 24, 25
    - Bin 3: 26, 28, 29, 34
  - Làm mịn bằng bin means:
    - Bin 1: 9, 9, 9, 9
    - Bin 2: 23, 23, 23, 23
    - Bin 3: 29, 29, 29, 29
  - Làm mịn bằng bin boundaries:
    - Bin 1: 4, 4, 4, 15
    - Bin 2: 21, 21, 25, 25
    - Bin 3: 26, 26, 26, 34

# Làm mịn nhiễu (tt)

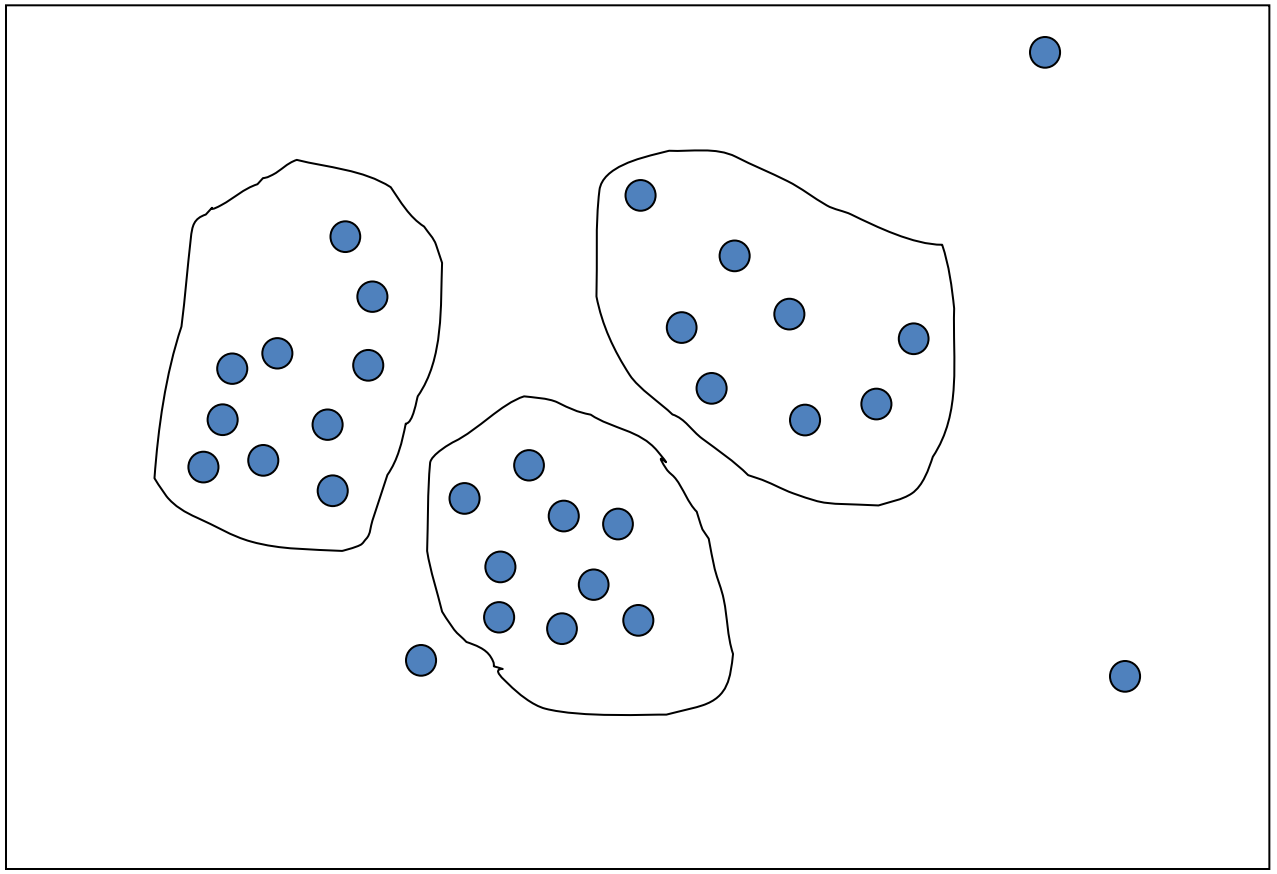
- Hồi quy (regression) : Làm mịn dữ liệu bằng cách so khớp dữ liệu với hàm hồi qui
  - Hồi quy tuyến tính (Linear regression)
  - Hồi quy tuyến tính phức hợp (Multiple linear regression)

# Làm mịn nhiễu (tt)



# Làm mịn nhiễu (tt)

- Gom nhóm (Clustering): Phát hiện và loại bỏ giá trị cá biệt



# Tiến trình làm sạch dữ liệu

- Phát hiện sự không nhất quán
  - Dùng siêu dữ liệu (các miền, phụ thuộc …)
  - Kiểm tra overloading
  - Kiểm ra ràng buộc(tính duy nhất, liên tục, khác rỗng…)
  - Dùng các công cụ thương mại
    - Data scrubbing: Dùng các thông tin cơ bản (mã số, chính tả …) để kiểm tra dữ liệu
    - Data auditing: Phân tích dữ liệu để tìm luật, phát hiện nhiễu và giá trị cá biệt

# Tiến trình làm sạch dữ liệu(tt)

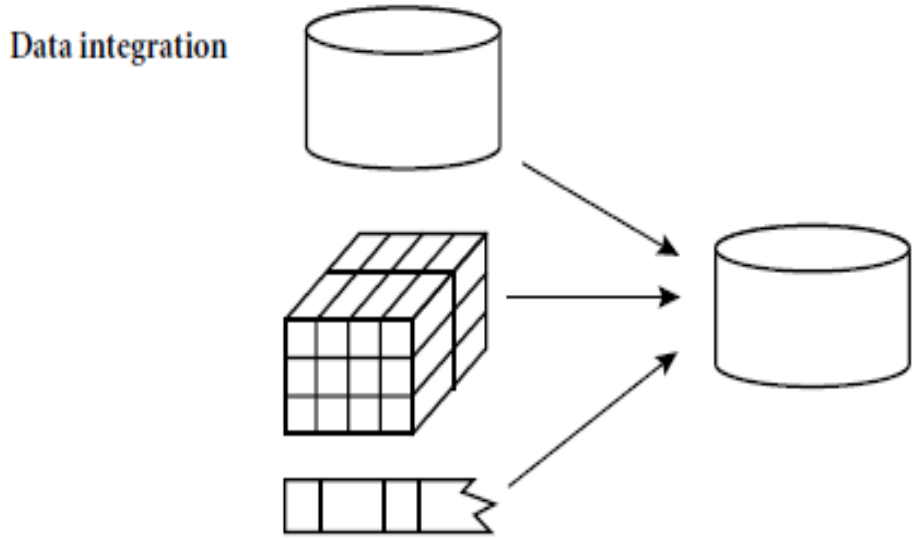
- Di chuyển và tích hợp dữ liệu
  - Công cụ di chuyển dữ liệu: Cho phép chuyển dữ liệu từ dạng này sang dạng khác
  - ETL (Extraction/Transformation/Loading): Chuyển đổi dữ liệu 1 cách chuyên biệt

# Nội dung

- Đặc điểm chung của dữ liệu
- Vì sao phải tiền xử lý dữ liệu?
- Tóm tắt dữ liệu
- Làm sạch dữ liệu
- Tích hợp & biến đổi dữ liệu
- Thu gọn dữ liệu

# Tích hợp dữ liệu

- Kết hợp dữ liệu từ nhiều nguồn khác nhau vào một nguồn thống nhất




- Nguồn sau cùng phải rõ ràng, mạch lạc (data warehouse)

# Tích hợp dữ liệu(tt)

- Các vấn đề cần quan tâm
  - Thống nhất lược đồ (schema integration)
    - customer_id = cust_number ?
  - Định danh thực thể (entity identification)
    - Bill Clinton = William Clinton?
  - Xung đột dữ liệu
    - Khác biệt đơn vị đo
    - Ràng buộc khóa ngoại
    - Phụ thuộc hàm
    - …
  - Dư thừa dữ liệu

# Dư thừa dữ liệu

- Dư thừa dữ liệu xảy ra khi tích hợp dữ liệu từ nhiều cơ sở dữ liệu khác nhau
  - Nhận dạng đối tượng: Cùng một đối tượng có thể có tên gọi khác nhau tại các CSDL khác nhau
  - Dữ liệu suy diễn: Một số thuộc tính có thể được suy ra từ các bảng khác nhau
- Các thuộc tính dư thừa có thể được phát hiện bằng phân tích tương quan (correlation analysis)

# Phân tích tương quan

- Đối với dữ liệu dạng số (numerical data)
  - Phân tích hệ số tương quan
- Đối với dữ liệu dạng loại (categorical data)
  - Kiểm định Chi-square

# Phân tích hệ số tương quan

- Pearson's product moment coefficient

$$r_{A,B} = \frac{\sum (A - \overline{A})(B - \overline{B})}{(n-1)\sigma_A \sigma_B} = \frac{\sum (AB) - n\overline{A}\overline{B}}{(n-1)\sigma_A \sigma_B}$$

  - n: số bộ dữ liệu
  - $\overline{A}$ và $\overline{B}$ là trung bình (mean) của A, B
  - $\sigma_A$ và $\sigma_B$ là độ lệch chuẩn của A, B
  - Σ(AB) là tổng của tích vô hướng AB

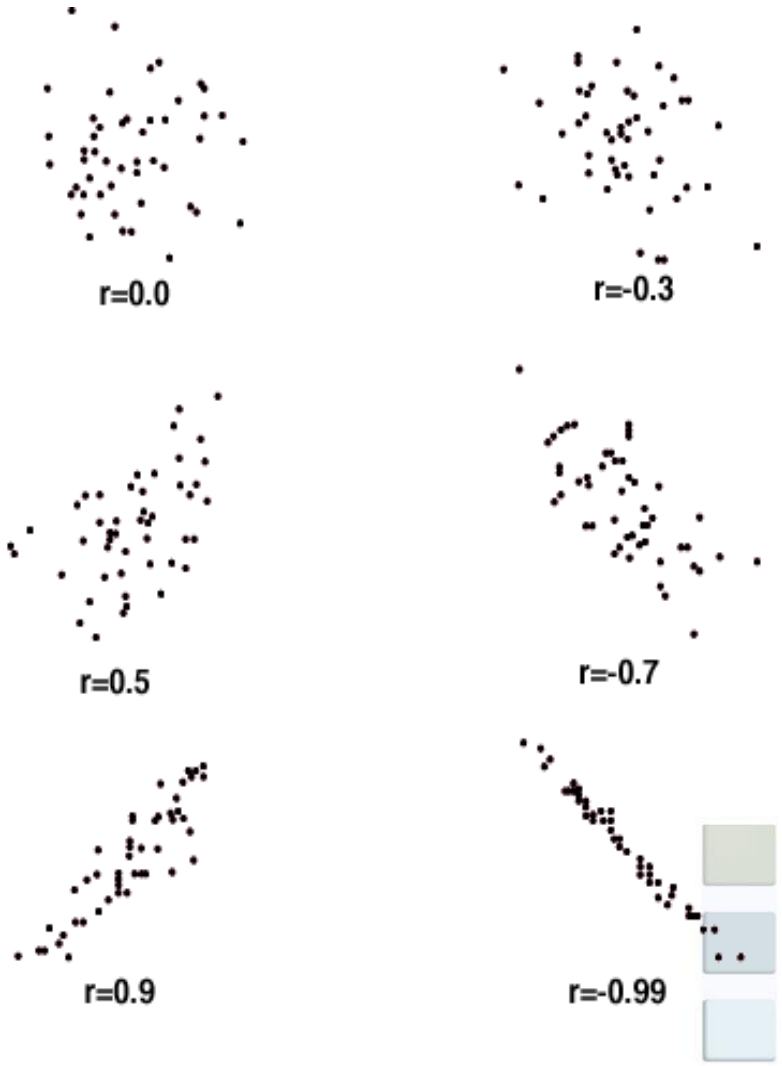
# Phân tích hệ số tương quan(tt)

$$r_{A,B} = \frac{\sum (A - \overline{A})(B - \overline{B})}{(n-1)\sigma_A \sigma_B} = \frac{\sum (AB) - n\overline{A}\,\overline{B}}{(n-1)\sigma_A \sigma_B}$$

- $r_{A,B} > 0$: A, B có tương quan cùng chiều
- $r_{A,B} = 0$: A, B độc lập
- $r_{A,B} < 0$: A, B tương quan ngược chiều

# Phân tích hệ số tương quan(tt)

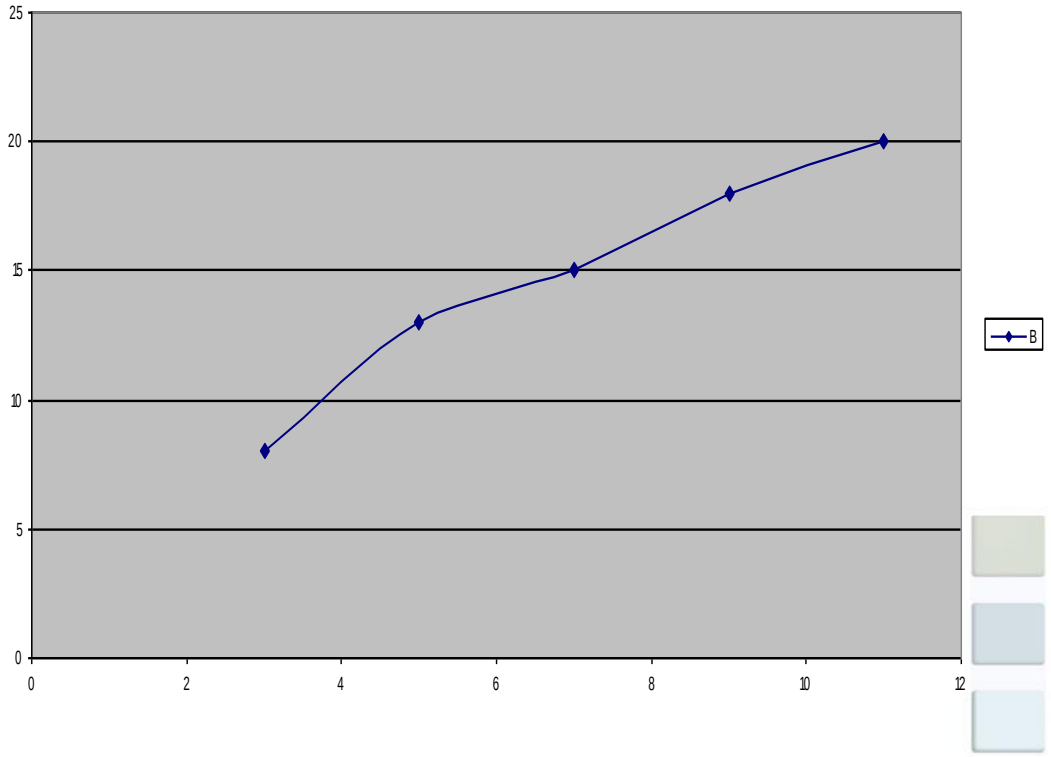
- |r| <= 1
- |r|>0.8: tương quan mạnh
- |r| từ 0.4 đến 0.8: tương quan trung bình
- |r|<0.4: tương quan yếu



# Phân tích hệ số tương quan(tt)

| | | | | | |
|---|---|---|---|---|---|
| A | 3 | 5 | 7 | 9 | 11 |
| B | 8 | 13 | 15 | 18 | 20 |

- n = 5
- Σ(XY) = 576
- $\overline{A} = 7$
- $\overline{B} = 14.8$
- $\sigma_A = 2.82$
- $\sigma_B = 4.16$
- $r_{A,B} = 0.984$
- ➔ Tương quan mạnh



# Kiểm định Chi-Square

- Là một phép kiểm định trong thống kê toán học
- Kiểm định A, B có độc lập hay không
- Giả thiết: A, B độc lập
- Bậc tự do:
  - A có c giá trị riêng biệt
  - B có r giá trị riêng biệt
  - ➔ Bậc tự do của phép kiểm định là (r-1)(c-1)

# Kiểm định Chi-Square

$$\chi^2 = \sum_{i=1}^{c} \sum_{j=1}^{r} \frac{(O_{ij} - E_{ij})^2}{E_{ij}}$$

- $O_{ij}$: Tần số quan sát thực của sự kiện ($A_i$, $B_j$)
- $E_{ij}$: Tần số mong đợi của sự kiện ($A_i$, $B_j$)

- $\chi^2$ càng lớn thì khả năng các biến có quan hệ với nhau càng lớn

# Kiểm định Chi-Square

$$E_{ij} = \frac{count(A = a_i) \times count(B = b_j)}{N}$$

- N: Tổng số bộ của dữ liệu
- count(A=$a_i$): số bộ dữ liệu trong A bằng $a_i$
- count(B=$b_j$): số bộ dữ liệu trong B bằng $b_i$

# Kiểm định Chi-Square

- Ví dụ:

| | male | female | Total |
|---|---|---|---|
| **fiction** | **250** | **200** | 450 |
| **non_fiction** | **50** | **1000** | 1050 |
| Total | 300 | 1200 | 1500 |

- 2 thuộc tính *gender* và *preferred_reading* có độc lập không?

# Kiểm định Chi-Square

- Tính các tần số mong đợi
- E(male, fiction)

$$E_{11} = \frac{count(male) \times count(fiction)}{N} = \frac{300 \times 450}{1500} = 90$$

| | male | female | Total |
|---|---|---|---|
| fiction | 250 (90) | 200 (360) | 450 |
| non_fiction | 50 (210) | 1000 (840) | 1050 |
| Total | 300 | 1200 | 1500 |

# Kiểm định Chi-Square

- Ví dụ (tt):

| | male | female | Total |
|---|---|---|---|
| **fiction** | **250 (90)** | **200 (360)** | 450 |
| **non_fiction** | **50 (210)** | **1000 (840)** | 1050 |
| Total | 300 | 1200 | 1500 |

$$\chi^2 = \frac{(250-90)^2}{90} + \frac{(50-210)^2}{210} + \frac{(200-360)^2}{360} + \frac{(1000-840)^2}{840} = 507.93$$

- Bậc tự do: (2-1) x (2-1) = 1
- Với 1 bậc tự do, giá trị $\chi^2$ cần để bác bỏ giả thiết độc lập của 2 thuộc tính với độ chính xác 0.001 là 10.828

# Kiểm định Chi-Square

- X2 = 507.93 > 10.828: Bác bỏ giả thiết *gender* và *preferred_reading* độc lập
- *gender* và *preferred_reading* có mối tương quan mạnh
- Các biến tương quan với nhau không có nghĩa là giữa chúng có quan hệ nhân quả
- Ví dụ:
  - num_hospital
  - thefts
  - hospital

# Chuyển đổi dữ liệu

- Chuyển đổi giá trị ban đầu của dữ liệu sang những giá trị mới thích hợp hơn
- Công việc
  - Làm mịn (smoothing)
  - Tổng hợp dữ liệu (aggregation)
  - Tổng quát hóa dữ liệu (generalization)
  - Chuẩn hóa dữ liệu (normalization)
  - Xây dựng thuộc tính mới (attribute construction)

# Chuẩn hóa dữ liệu

- Thu hẹp dữ liệu về một khoảng nhất định
- Phương pháp
  - Chuẩn hóa min-max
  - Chuẩn hóa z-score
  - Chuẩn hóa bằng tỷ lệ thập phân

# Chuẩn hóa min-max

$$v' = \frac{v - min_A}{max_A - min_A}(new\_max_A - new\_min_A) + new\_min_A$$

- Ví dụ:
  - Tuổi: 13, 15, 16, 16, 19, 20, 21, 22, 25, 30, 33, 35, 35, 35, 35, 36, 45, 52, 70
  - Yêu cầu: chuẩn hoá giá trị 35
- Chuẩn hóa min-max
  - min-max: [0…1]

$$v' = \frac{35 - 13}{70 - 13} \times (1 - 0) + 0 = 0.38$$

# Chuẩn hóa z-score

$$v' = \frac{v - \mu_A}{\sigma_A}$$

- Ví dụ:
  - Tuổi: 13, 15, 16, 16, 19, 20, 21, 22, 25, 30, 33, 35, 35, 35, 35, 36, 45, 52, 70
  - Yêu cầu: chuẩn hoá giá trị 35
- Chuẩn hóa z-score
  - $\mu$ = 30.15
  - $\sigma$ = 14.43

$$v' = \frac{35 - 30.15}{14.43} = 0.336$$

# Chuẩn hóa bằng tỷ lệ thập phân

$$v' = \frac{v}{10^j}$$

- j là số nguyên nhỏ nhất sao cho Max(|v'|) < 1
- Ví dụ:
  - Tuổi: 13, 15, 16, 16, 19, 20, 21, 22, 25, 30, 33, 35, 35, 35, 35, 36, 45, 52, 70
  - Yêu cầu: chuẩn hoá giá trị 35
- Chuẩn hóa:
  - max|v| = 70<100 → j=2

$$v' = \frac{35}{10^2} = 0.35$$

# Nội dung

- Đặc điểm chung của dữ liệu
- Vì sao phải tiền xử lý dữ liệu?
- Tóm tắt dữ liệu
- Làm sạch dữ liệu
- Tích hợp & biến đổi dữ liệu
- Thu gọn dữ liệu

# Thu gọn dữ liệu

- Tại sao phải thu gọn dữ liệu?
  - Dữ liệu trong thực tế là rất lớn (terabytes)
  - Không thể mining trên tập dữ liệu này (thời gian + chi phí)
- Thu gọn dữ liệu: Làm giảm kích thước tập dữ liệu nhưng vẫn cho ra kết quả gần giống khi khai thác

# Thu gọn dữ liệu(tt)

- Nội dung:
  - Xây dựng tập thuộc tính.
  - Giảm chiều dữ liệu.
  - Giảm kích thước dữ liệu

# Xây dựng tập thuộc tính

- Ý **tưởng**: Chọn ra một tập thuộc tính nhỏ nhất đủ để mô tả xấp xỉ sự phân bố của các mẫu dữ liệu theo một tiêu chuẩn nào đó
- **Mục tiêu**: Đơn giản hóa các mẫu dữ liệu, giảm chi phí tính toán, đảm bảo tính chính xác khi phân tích

# Xây dựng tập thuộc tính

- **Hướng giải quyết** : Tìm cách loại bỏ những thuộc tính có ảnh hưởng không đáng kể đến sự phân bố của các mẫu dữ liệu
- Phương pháp
  - Stepwise forward selection
  - Stepwise backward elimination
  - Kết hợp forward selection và backward elimination
  - Cây quyết định: ID3, C4.5, CART

# Stepwise forward selection

- Đi từ tập thuộc tính ban đầu là rỗng, sau mỗi lần lặp, bổ sung thêm các thuộc tính mới
- Mô hình:
  - Tập thuộc tính ban đầu: {A, B, C, D, E, F, G, H}
  - Tập thuộc tính chọn lọc:
    - {}
    - {A}
    - {A,  C}
    - {A, C, F}

# Stepwise backward elimination

- Khởi đầu là toàn bộ tập thuộc tính, sau mỗi bước thuộc tính tồi nhất sẽ bị bỏ đi
- Mô hình
  - Tập thuộc tính ban đầu: {A, B, C, D, E, F, G, H}
  - {A, B, C, D, F, G, H}
  - {A, C, D, F, H}
  - {A, C, F, H}
  - {A, C, F}

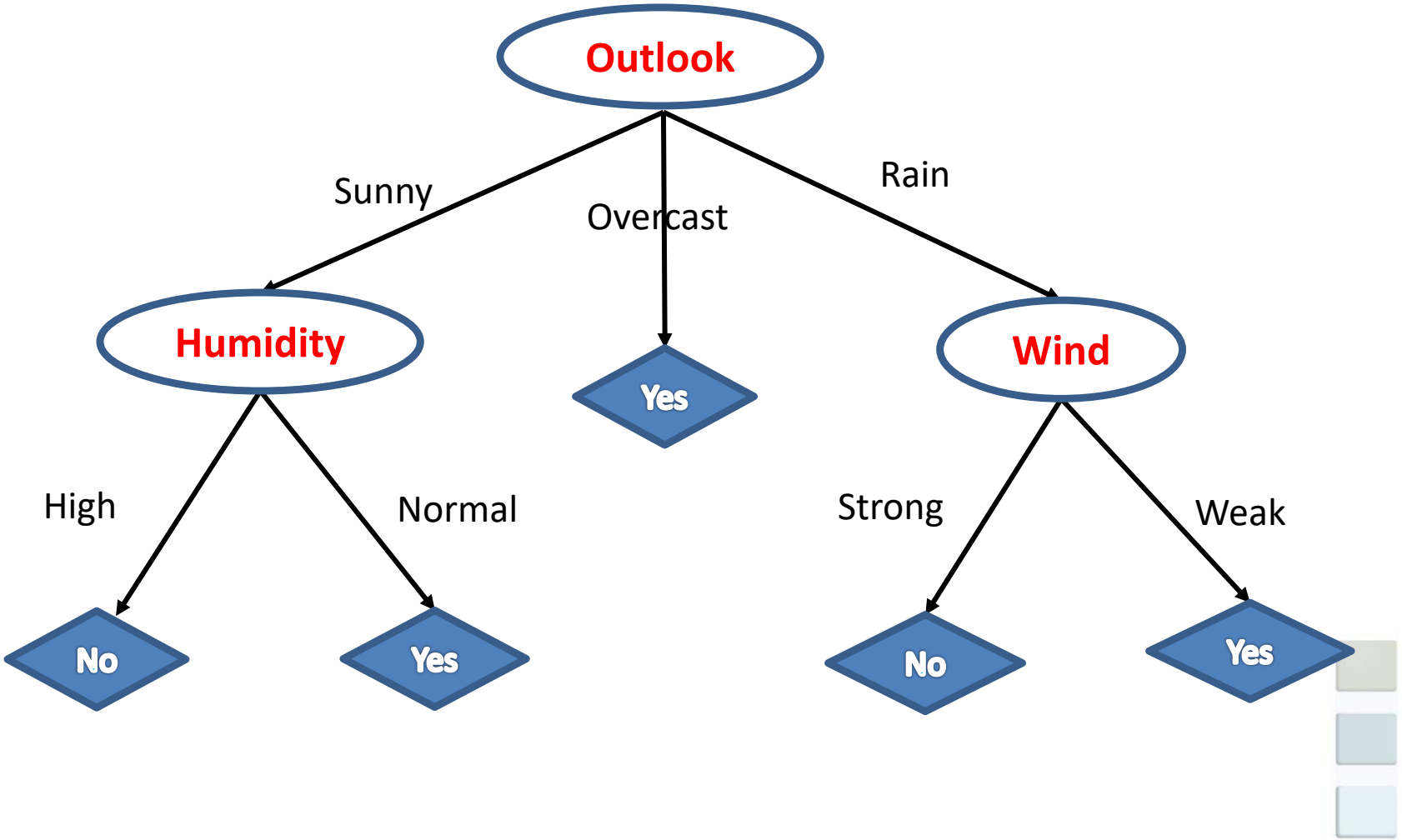
# Forward selection & backward elimination

- Kết hợp giữa forward selection & backward elimination
- Tại mỗi bước, các thuộc tính tốt sẽ được thêm vào, và các thuộc tính xấu sẽ bị bỏ đi

# Cây quyết định

| Day | Outlook | Temp | Humidity | Wind | Play? |
|---|---|---|---|---|---|
| 1 | Sunny | Hot | High | Weak | No |
| 2 | Sunny | Hot | High | Strong | No |
| 3 | Overcast | Hot | High | Weak | Yes |
| 4 | Rain | Mild | High | Weak | Yes |
| 5 | Rain | Cool | Normal | Weak | Yes |
| 6 | Rain | Cool | Normal | Strong | No |
| 7 | Overcast | Cool | Normal | Strong | Yes |
| 8 | Sunny | Mild | High | Weak | No |
| 9 | Sunny | Cool | Normal | Weak | Yes |
| 10 | Rain | Mild | Normal | Weak | Yes |
| 11 | Sunny | Mild | Normal | Strong | Yes |
| 12 | Overcast | Mild | High | Strong | Yes |
| 13 | Overcast | Hot | Normal | Weak | Yes |
| 14 | Rain | Mild | High | Strong | No |

# Cây quyết định(tt)



# Giảm chiều của dữ liệu

- **Ý tưởng**: Tìm những phương pháp biến đổi, biểu diễn dữ liệu dưới dạng "nén"
- **Mục đích**: Giảm chiều của các mẫu dữ liệu nhằm giảm chi phí tính toán mà không làm giảm độ chính xác khi phân tích dữ liệu

# Giảm chiều của dữ liệu(tt)

- “Lossless”: sau khi biến đổi có thể khôi phục lại dữ liệu như trạng thái ban đầu
- “Lossy”: sau khi biến đổi không thể khôi phục lại dữ liệu như trạng thái ban đầu
- Trong thực tế, phép biến đổi dạng lossy được quan tâm nhiều hơn lossless

# Giảm chiều của dữ liệu(tt)

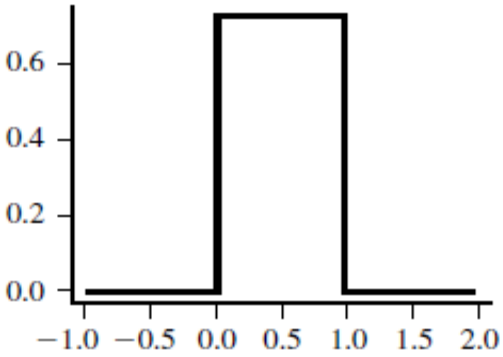
- 2 phương pháp chính
  - Phân tích thành phần chính (Principle Component Analysis-PCA)
  - Biến đổi wavelet rời rạc (Discrete Wavelet Transform)

# Biến đổi wavelet rời rạc

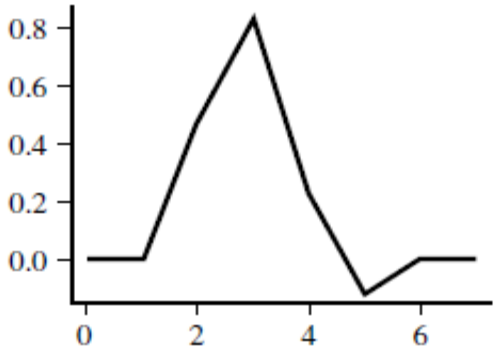
- **Ý tưởng:** Nén dữ liệu
- **Phương pháp:**
  - Từ vector dữ liệu ($x_1$, $x_2$, …, $x_N$)
  - ➔ ($y_1$, $y_2$, …, $y_N$), trong đó $y_i$ là các hệ số wavelet.
  - Những $y_i$ nhỏ hơn ngưỡng (threshold) cho trước được làm tròn về 0.

# Biến đổi wavelet rời rạc

- Có nhiều dạng biến đổi wavelet, mỗi dạng được đặc trưng bởi một phép biến đổi riêng.
- Các dạng biến đổi wavelet thông dụng :
  - Haar-2.
  - Daubechies-4.
  - Daubechies-6.



(a) Haar-2



(b) Daubechies-4

# Haar - 2

- Biến đổi Haar – 2 được đặc trưng bởi ma trận

$$H_2 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}.$$

- Mỗi mẫu dữ liệu sẽ được biểu diễn dưới dạng vector : $\{x_0, x_1, \ldots, x_{2n}, x_{2n+1}\}$.
- Nếu số thành phần của x không là lũy thừa của 2 thì ta thêm 0 vào cho đủ.

# Haar – 2(tt)

- Các bước biến đổi :
  - B1: Gom nhóm 2 thành phần kề nhau $(x_0, x_1)$ , …, $(x_{2n}, x_{2n+1})$
  - B2: Với mỗi vector $(x_{2i}, x_{2i+1})$ ta nhân vào bên phải $H_2$ và thu được $(x_{2i} + x_{2i+1}, x_{2i} - x_{2i+1})$
  - B3: Sắp xếp lại $(s_0, s_1, …, s_n, t_0, t_1, …, t_n)$ trong đó :
    - $s_i = x_{2i} + x_{2i+1}$
    - $t_i = x_{2i} - x_{2i+1}$
  - B4: Giữ nguyên {t} và lặp lại quá trình trên cho $\{s_0, s_1,…, s_n\}$ nếu $n > 2$. Bản chất là giữ lại các hiệu $t_i$ và tiếp tục biến đổi các tổng $s_i$

# Haar – 2(tt)

- Cho mẫu dữ liệu : $\{a_0, a_1, a_2, a_3\}$
  - Biến đổi:

$$\begin{pmatrix} +1 & +1 \\ +1 & -1 \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \end{pmatrix} = \begin{pmatrix} a_0 + a_1 \\ a_0 - a_1 \end{pmatrix}$$

$$\begin{pmatrix} +1 & +1 \\ +1 & -1 \end{pmatrix} \begin{pmatrix} a_2 \\ a_3 \end{pmatrix} = \begin{pmatrix} a_2 + a_3 \\ a_2 - a_3 \end{pmatrix}$$

  - Thu được: $\{ a_0 - a_1, a_0 + a_1, a_2 - a_3, a_2 + a_3\}$
  - Sắp xếp lại: $\{ a_0 - a_1, a_2 - a_3, a_0 + a_1, a_2 + a_3 \}$
  - Không biến đổi tiếp vì $\{ a_0 - a_1, a_2 - a_3 \}$ có độ dài bằng 2

# Haar – 2(tt)

- $\{ a_0 - a_1, a_2 - a_3, a_0 + a_1, a_2 + a_3 \}$
- Nếu $a_0 - a_1 \rightarrow 0$

➔ Ta thu được dãy $\{a_2 - a_3, a_0 + a_1, a_2 + a_3 \}$

➔ Giảm chiều dữ liệu.

# PCA

- Phương pháp hiệu quả để phân tích dữ liệu đa chiều.
- Phân tích sự nhất quán và khác biệt của các mẫu dữ liệu.
- **Ý tưởng**: Chiếu dữ liệu lên một không gian ít chiều hơn mà không làm mất mát quá nhiều thông tin.

# PCA(tt)

- **Nội dung:** Phân tích các thành phần chính của dữ liệu thông qua các bước sau:
  - Chuẩn hóa dữ liệu.
  - Tính ma trận hiệp phương sai (Covariance).
  - Xác định vector đặc trưng và trị riêng
  - Chọn thành phần chính (chọn vector đặc trưng).
  - Lấy đặc trưng mẫu.

# PCA(tt)

- **Chuẩn hóa:** Lấy giá trị đang xét trừ đi trung bình mẫu của từng thành phần.
- Với mỗi mẫu ($x_i$, $y_i$) :

$$x_i = x_i - \bar{x}$$

$$y_i = y_i - \bar{y}$$

# PCA(tt)

- Dữ liệu sau khi chuẩn hóa

Data =

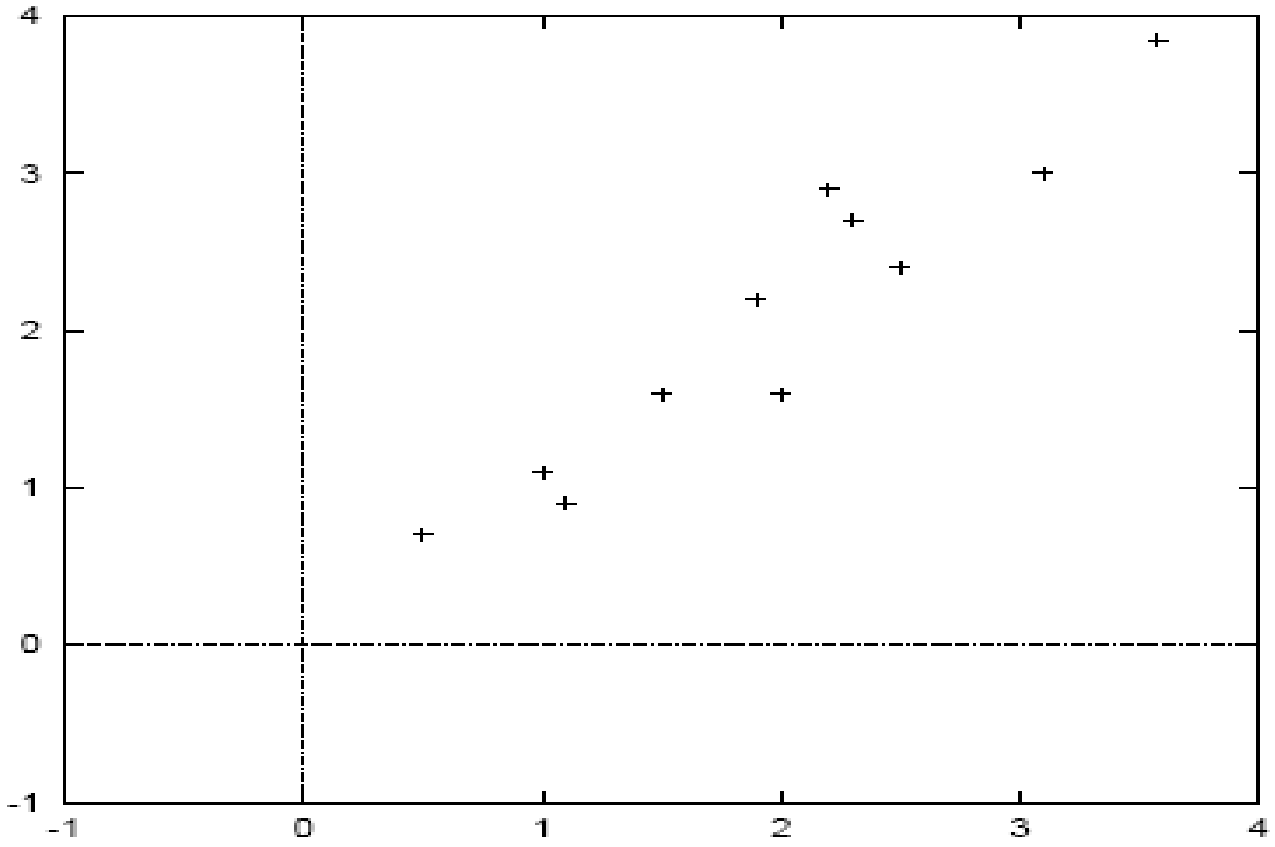
| $x$ | $y$ |
|---|---|
| 2.5 | 2.4 |
| 0.5 | 0.7 |
| 2.2 | 2.9 |
| 1.9 | 2.2 |
| 3.1 | 3.0 |
| 2.3 | 2.7 |
| 2 | 1.6 |
| 1 | 1.1 |
| 1.5 | 1.6 |
| 1.1 | 0.9 |

DataAdjust =

| $x$ | $y$ |
|---|---|
| .69 | .49 |
| -1.31 | -1.21 |
| .39 | .99 |
| .09 | .29 |
| 1.29 | 1.09 |
| .49 | .79 |
| .19 | -.31 |
| -.81 | -.81 |
| -.31 | -.31 |
| -.71 | -1.01 |

# PCA(tt)

- Đồ thị thể hiện phân bố dữ liệu sau khi chuẩn hóa



# PCA(tt)

- Phương sai
- Hiệp phương sai: hiệp phương sai là độ đo sự biến thiên cùng nhau của hai biến ngẫu nhiên

$$\text{cov} = (x - \bar{x})(y - \bar{y})$$

$$\text{cov} = \frac{\sum_{i=1}^{N} (x_i - \bar{x})(y_i - \bar{y})}{N - 1}$$

# PCA(tt)

- Ma trận hiệp phương sai

$$\Sigma = \begin{pmatrix} \sigma_{11} & \cdots & \sigma_{1n} \\ \sigma_{21} & \sigma_{22} & \\ \vdots & \ddots & \vdots \\ \sigma_{n1} & \cdots & \sigma_{pp} \end{pmatrix}$$

- $\sigma_{ii}$ : hiệp phương sai của (X,X) → Chính là phương sai của X
- $\sigma_{ij}$ : hiệp phương sai của (X,Y)

# PCA(tt)

- Ví dụ: Tính ma trận hiệp phương sai
- Cho ma trận:

$$\mathbf{X} = \begin{bmatrix} 4.0 & 2.0 & .60 \\ 4.2 & 2.1 & .59 \\ 3.9 & 2.0 & .58 \\ 4.3 & 2.1 & .62 \\ 4.1 & 2.2 & .63 \end{bmatrix}$$

- Vector trung bình của X:

$$\bar{\mathbf{x}} = \begin{bmatrix} 4.10 & 2.08 & .604 \end{bmatrix}$$

# PCA(tt)

$$\mathrm{cov} = \frac{\sum_{i=1}^{N}(x_i - \bar{x})(y_i - \bar{y})}{N-1}$$

$$\mathrm{cov}_{11} = \sigma_{11} = \frac{(4.0-4.1)^2 + (4.2-4.1)^2 + (3.9-4.1)^2 + (4.3-4.1)^2 + (4.1-4.1)^2}{5-1} = 0.025$$

$$\mathrm{cov}_{12} = \sigma_{12} = \frac{(4.0-4.1)(2.0-2.08) + (4.2-4.1)(2.1-2.08)}{5-1} +$$

$$\frac{(3.9-4.1)(2.0-2.08) + (4.3-4.1)(2.1-2.08)}{5-1} +$$

$$\frac{(4.1-4.1)(2.2-2.08)}{5-1} = 0.0075$$

# PCA(tt)

- Ma trận hiệp phương sai:

$$S = \begin{bmatrix} 0.025 & 0.0075 & 0.00175 \\ 0.0075 & 0.0070 & 0.00135 \\ 0.00175 & 0.00135 & 0.00043 \end{bmatrix}$$

$$cov = \begin{pmatrix} .616555556 & .615444444 \\ .615444444 & .716555556 \end{pmatrix}$$

# PCA(tt)

- **Vector đặc trưng (Eigenvectors)**
  - Cho ma trận vuông A, một vector C được gọi là vector đăng trưng của A nếu và chỉ nếu tồn tại một số $\alpha$ sao cho:

$$AC = \lambda C$$

  - $\lambda$ được gọi là trị riêng (hay giá trị đặc trưng-eigenvalue) của A

$$\begin{pmatrix} 2 & 3 \\ 2 & 1 \end{pmatrix} \times \begin{pmatrix} 1 \\ 3 \end{pmatrix} = \begin{pmatrix} 11 \\ 5 \end{pmatrix}$$

$$\begin{pmatrix} 2 & 3 \\ 2 & 1 \end{pmatrix} \times \begin{pmatrix} 3 \\ 2 \end{pmatrix} = \begin{pmatrix} 12 \\ 8 \end{pmatrix} = 4 \times \begin{pmatrix} 3 \\ 2 \end{pmatrix}$$

# PCA(tt)

- **Tính vector đặc trưng và trị riêng?**
- Cho ma trận: $\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}$

$$\det \begin{bmatrix} 2-\lambda & 1 \\ 1 & 2-\lambda \end{bmatrix} = 0,$$
$$\Rightarrow (2-\lambda)^2 - 1 = 0,$$
$$\Rightarrow \lambda^2 - 4\lambda + 3 = 0.$$

$$\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = 3 \begin{bmatrix} x \\ y \end{bmatrix}. \qquad \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 3 \\ 3 \end{bmatrix} = 3 \begin{bmatrix} 1 \\ 1 \end{bmatrix}.$$

# PCA(tt)

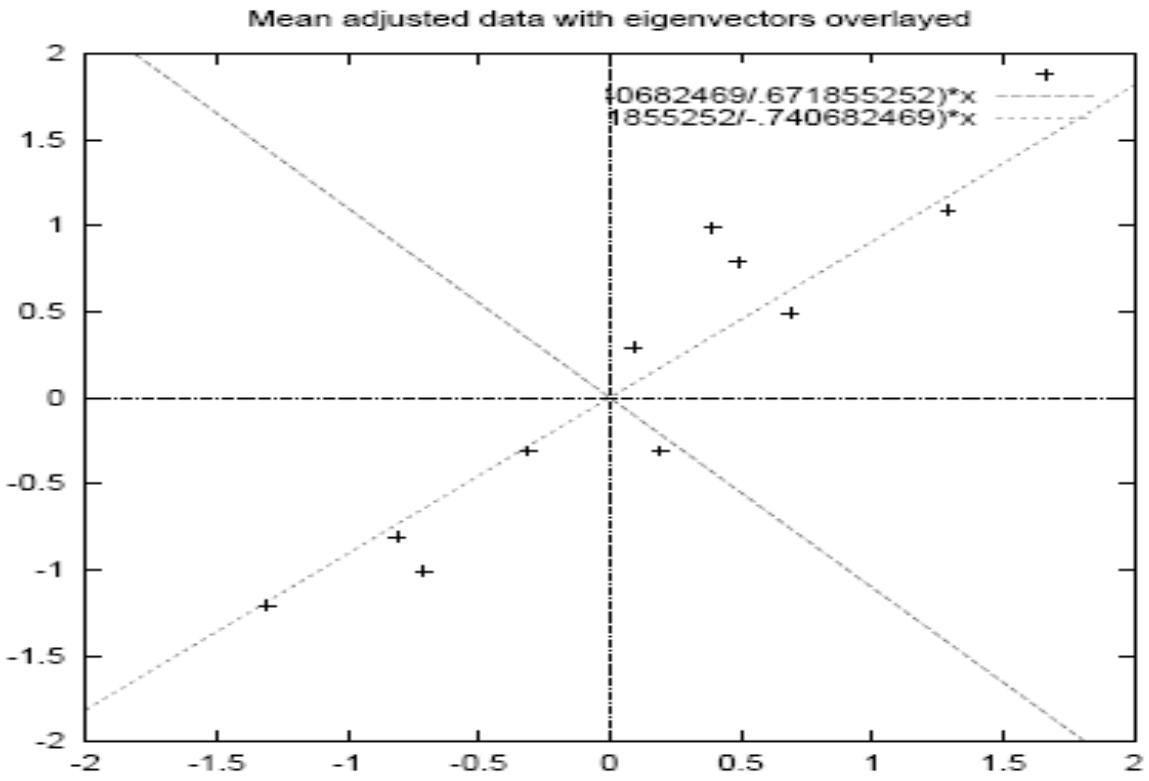
- Ví dụ (tt)

$$eigenvalues = \begin{pmatrix} .0490833989 \\ 1.28402771 \end{pmatrix}$$

$$eigenvectors = \begin{pmatrix} -.735178656 & -.677873399 \\ .677873399 & -.735178656 \end{pmatrix}$$

# PCA(tt)

- Chọn thành phần chính và thiết lập vector đặc trưng



# PCA(tt)

- **Rút ra vector đặc trưng:**

$$FeatureVector = (eig_1\ eig_2\ eig_3\ ....\ eig_n)$$

- Các vector được sắp từ trái qua phải giảm dần theo giá trị của trị riêng.
- Để giảm chiều dữ liệu, người ta định nghĩa một ngưỡng và loại đi các vector đặc trưng có trị riêng nhỏ hơn ngưỡng đó.

# PCA(tt)

- Đối với ví dụ minh họa, vector đặc trưng sẽ là :

$$\begin{pmatrix} -.677873399 & -.735178656 \\ -.735178656 & .677873399 \end{pmatrix}$$

- Ta có thể rút gọn bằng cách loại đi một vector đặc trưng như sau

$$\begin{pmatrix} -.677873399 \\ -.735178656 \end{pmatrix}$$

# PCA(tt)

- Lấy đặc trưng mẫu

$$FinalData = RowFeatureVector \times RowDataAdjust$$

  - FinalData sẽ là ma trận biểu diễn đặc trưng dữ liệu.
  - RowFeatureVector là chuyển vị của FeatureVector.
  - Tương tự với RowDataAdjust.

- Khôi phục dữ liệu

$$RowDataAdjust = RowFeatureVector^{-1} \times FinalData$$

# PCA trong nhận dạng khuôn mặt

- Demo

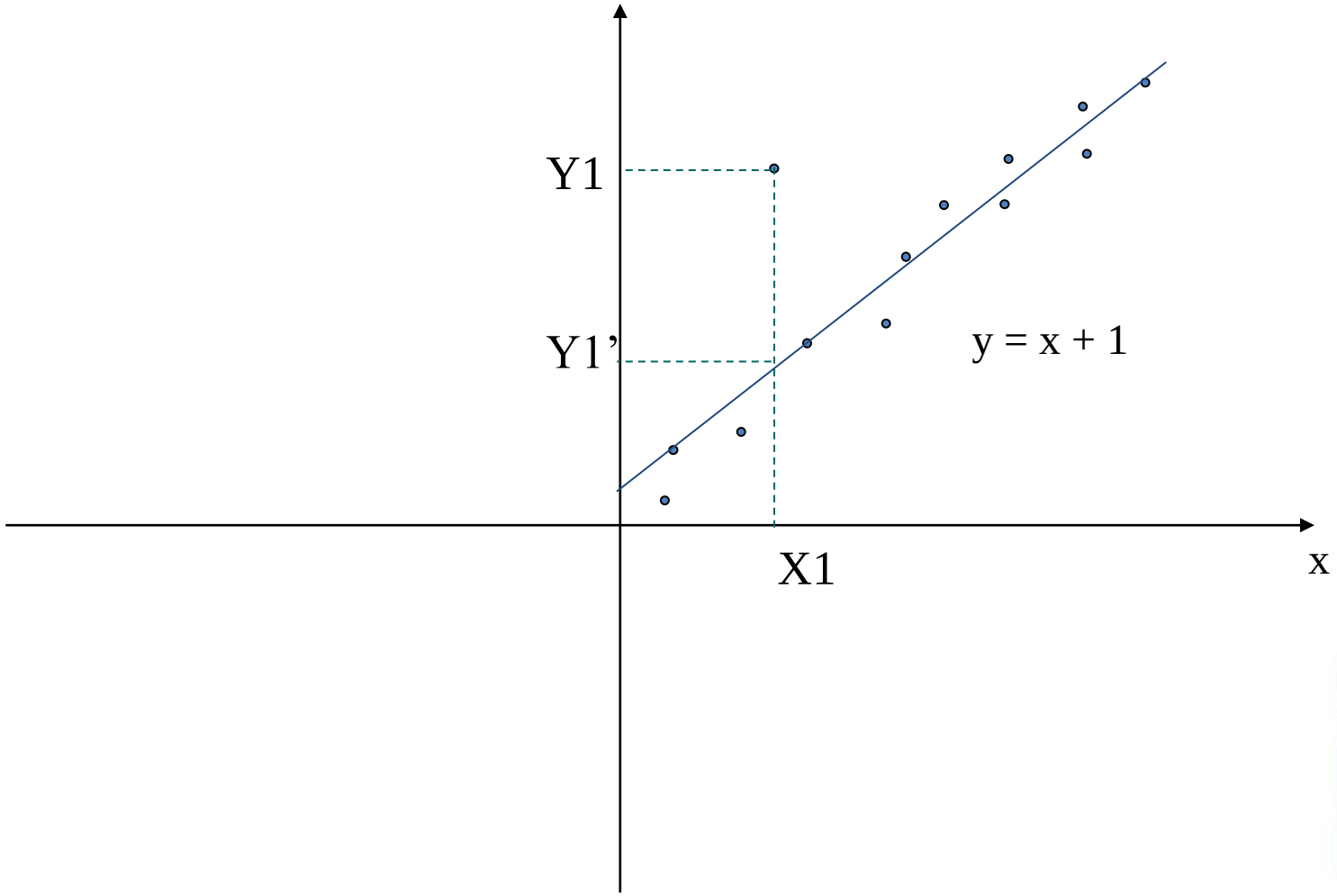
# Giảm kích thước dữ liệu

- **Ý tưởng:** Thay vì lưu trữ toàn bộ dữ liệu thì tìm cách biểu diễn dữ liệu dưới dạng những tham số đặc trưng của dữ liệu.
- **Phương pháp:** Gồm hai loại chính:
  - Phương pháp có tham số (Parametric Method): Hồi qui (regression), mô hình logarithm tuyến tính (Log-Linear Model)
  - Phương pháp không tham số (Nonparametric Method): Clustering, Sampling, Histogram

# Hồi qui (regression)

- Hồi quy thường được áp dụng để xấp xỉ ra dạng tham số của dữ liệu cho trước
- Ví dụ đơn giản là "fit" những điểm cho trước trên mặt phẳng vào một đường thẳng, nói cách khác là xấp xỉ ra một đường thẳng: y = ax+ b
- Quá trình xấp xỉ đó sẽ hướng tới việc tối ưu một tiêu chuẩn nào đó

# Hồi qui (regression)



# Hồi qui (regression)

- Ví dụ:

| X Value | Y Value | X*Y | X*X |
|---|---|---|---|
| 60 | 3.1 | 60 * 3.1 = 186 | 60 * 60 = 3600 |
| 61 | 3.6 | 61 * 3.6 = 219.6 | 61 * 61 = 3721 |
| 62 | 3.8 | 62 * 3.8 = 235.6 | 62 * 62 = 3844 |
| 63 | 4 | 63 * 4 = 252 | 63 * 63 = 3969 |
| 65 | 4.1 | 65 * 4.1 = 266.5 | 65 * 65 = 4225 |

$$\sum X = 311 \qquad \sum Y = 18.6$$

$$\sum X^2 = 19359 \qquad \sum XY = 1159.7$$

# Hồi qui (regression)

$$a = \frac{\sum XY - \sum X \sum Y}{N\sum X^2 - \left(\sum X\right)^2}$$

$$b = \frac{\sum Y - a\sum X}{N}$$

- a = 0.19, b = -8.098
- y = 0.19x - 8.098
- Với x = 64 ta có:
  - y = 0.19(0.64) - 8.098 = 4.06

# Mô hình logarithm tuyến tính

- **Ý tưởng:** Tìm ra các mối quan hệ logarithm tuyến tính giữa các thuộc tính trong các mẫu dữ liệu

$$\log(y) = a_0 + a_1x_1 + a_2x_2 + \ldots + a_Nx_N$$

  - $x_i$ là các thuộc tính độc lập quan hệ.
  - y là thuộc tính phụ thuộc quan hệ đối với $\{x_i\}$.
  - $\{a_i\}$ là các tham số logarithm tuyến tính
- **Phương pháp:** Tính các hệ dựa trên các mẫu dữ liệu sẵn có theo một tiêu chuẩn nào đó

# Mô hình logarithm tuyến tính

- Ví dụ:

| Y | X |
|---|---|
| 35 | 9.48 |
| 40 | 9.83 |
| 50 | 10.43 |
| 55 | 10.68 |
| 70 | 11.32 |
| 75 | 11.51 |

  - Có sự liên quan giữa X và Y.
  - X tăng thì Y tăng
  - X tăng chậm thì Y tăng chậm và ngược lại
  - ➔ Giữa X, Y có mối liên hệ khá chặt chẽ

# Mô hình logarithm tuyến tính

- Xét Ln(y) = ax + b  (1)

| Ln(Y) | X |
|---|---|
| 3.55 | 9.48 |
| 3.68 | 9.83 |
| 3.91 | 10.43 |
| 4.00 | 10.68 |
| 4.24 | 11.32 |
| 4.31 | 11.51 |

- Đặt z = ln(y)

➔ (1) trở thành z = ax + b

➔ Mô hình hồi quy

# Mô hình logarithm tuyến tính

$$a = \frac{\sum XY - \left(\sum X\right)\left(\sum Y\right)}{N\sum X^2 - \left(\sum X\right)^2}$$

$$b = \frac{\sum Y - a\sum X}{N}$$

- a = 0,375; b = 0
- ln(y) = 0.375x
  - x = 9.5
  - Ln(y) = 0.375*9.5 = 3.5625
  - y = $e^{3.5625}$ = 35.3

# Gom nhóm

- **Ý tưởng:** Gom các đối tượng tương tự nhau thành một nhóm
- **Mục tiêu:** Thay vì lưu trữ toàn bộ dữ liệu, ta chỉ lưu trữ dạng biểu diễn đặc trưng của các nhóm.
- **Hiệu quả:** Phụ thuộc vào bản chất của dữ liệu. (Hiệu quả đối với dữ liệu có cấu trúc, có tổ chức và kém hiệu quả đối với những dữ liệu phi cấu trúc)

# Gom nhóm(tt)

- **Phương pháp:** Gồm một số phương pháp phổ biến sau :
  - K-Means Clustering
  - Fuzzy C-Means Clustering
  - Hierachical
  - Mixture of Gaussian

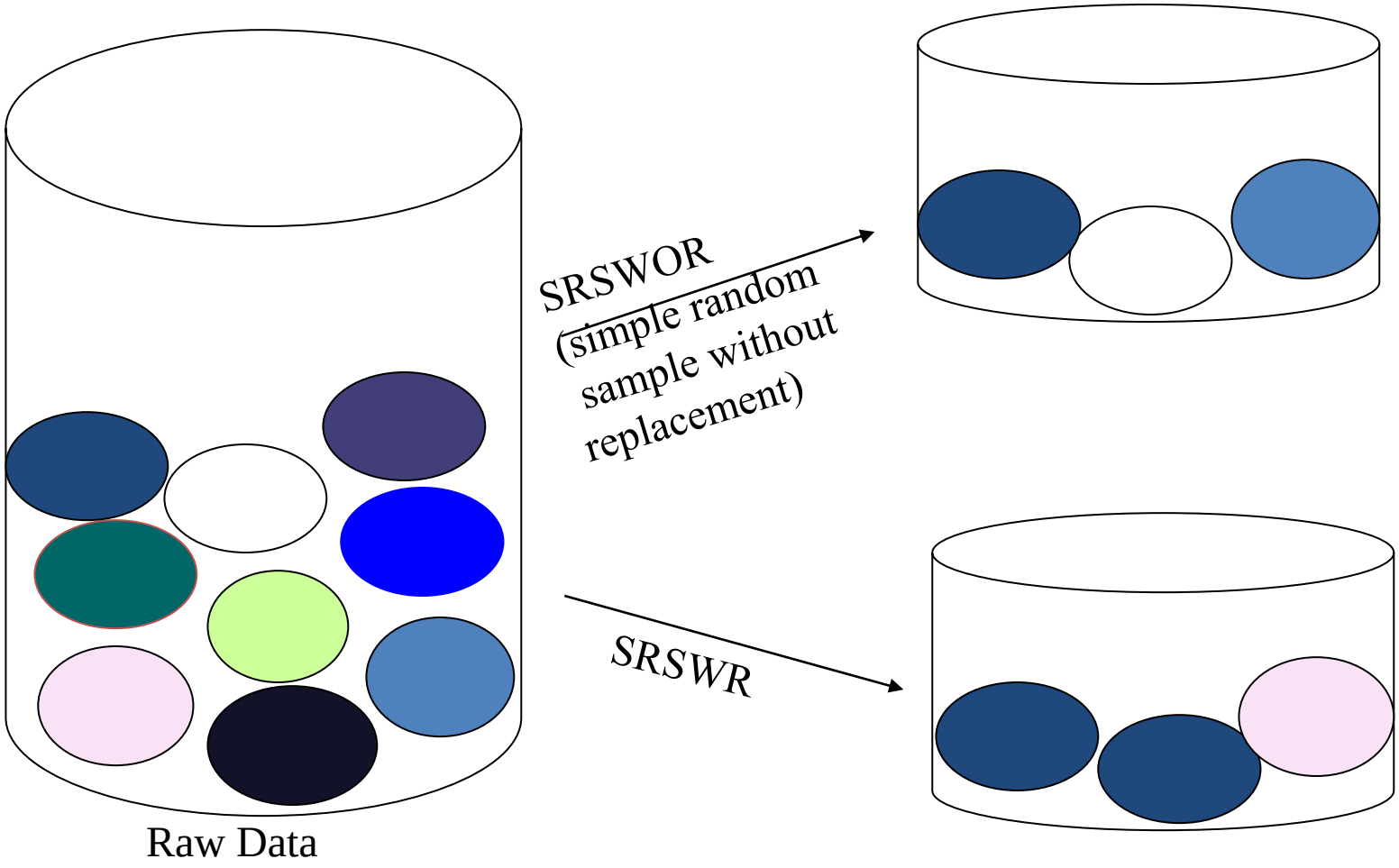
# Lấy mẫu (Sampling)

- Ý tưởng: Lấy một mẫu nhỏ để đại diện cho toàn bộ tập dữ liệu
- Nguyên lý:
  - Lấy mẫu theo kiểu ngẫu nhiên sẽ cho hiệu quả kém
  - Cần phát triển các phương pháp lấy mẫu tối ưu

# Các dạng lấy mẫu

- Lấy ngẫu nhiên
  - Xác xuất lấy là như nhau với mỗi item
- Lấy không đặt lại
  - Khi được chọn, mẫu được lấy sẽ bị loại khỏi tập ban đầu
- Lấy có đặt lại
  - Mẫu được chọn không bị loại khỏi tập ban đầu
- Lấy phân tầng
  - Phân tầng tập dữ liệu, sau đó lấy mẫu ở từng tầng

# Lấy mẫu có/không đặt lại



# Lấy mẫu phân tầng

Stratified sample
(according to *age*)

| | |
|---|---|
| T38 | youth |
| T256 | youth |
| T307 | youth |
| T391 | youth |
| T96 | middle_aged |
| T117 | middle_aged |
| T138 | middle_aged |
| T263 | middle_aged |
| T290 | middle_aged |
| T308 | middle_aged |
| T326 | middle_aged |
| T387 | middle_aged |
| T69 | senior |
| T284 | senior |

| | |
|---|---|
| T38 | youth |
| T391 | youth |
| T117 | middle_aged |
| T138 | middle_aged |
| T290 | middle_aged |
| T326 | middle_aged |
| T69 | senior |

# Rời rạc hóa dữ liệu

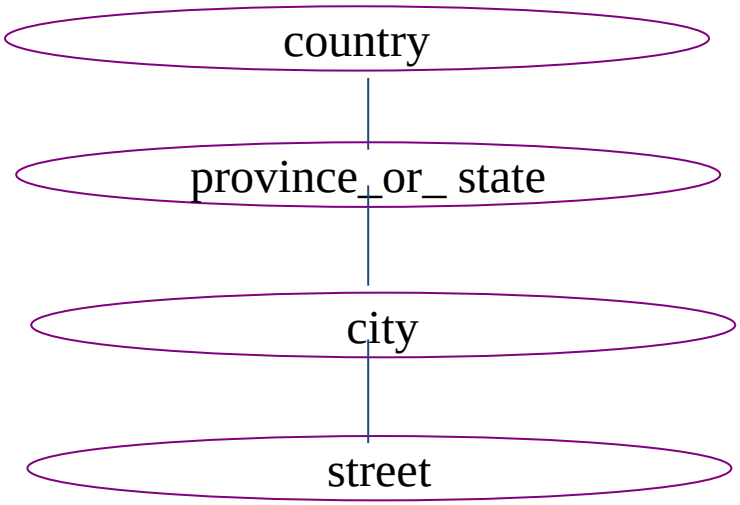
- 3 loại thuộc tính
  - Nominal: màu sắc, nghề nghiệp …
  - Ordinal: ngày tháng, điểm số…
  - Continuous: số thực …
- Rời rạc hoá:
  - Chia các thuộc tính liên tục ra thành từng khoảng
  - Môi số thuật toán phân lớp chỉ dừng được với dữ liệu dạng loai
  - Giảm kích thước dữ liệu
  - …

# Phân cấp khái niệm

- Ý tưởng: Giảm kích thước dữ liệu bằng cách thu thập và thay thế các khái nhiệm ở mức thấp thành các khái niệm ở mức cao hơn
- Khái niệm ở mức thấp: 4, 5, 6, 17, 22, 24, 55, 66
- Khái niệm ở mức cao: Trẻ, thanh niên, già …

# Phân cấp khái niệm

- Mô hình phân cấp




15 distinct values

365 distinct values

3567 distinct values

674,339 distinct values

# Rời rạc hóa và phân cấp khái niệm

- Tổng kết các phương pháp
  - Binning
    - Top-down, unsupervised
  - Histogram
    - Top-down, unsupervised
  - Clustering
    - Either top-down split or bottom-up merge, unsupervised
  - Entropy-based:
    - Top-down, supervised
  - …

# Rời rạc hóa và phân cấp khái niệm

- Tổng kết các phương pháp
  - Binning
    - Top-down, unsupervised
  - Histogram
    - Top-down, unsupervised
  - Clustering
    - Either top-down split or bottom-up merge, unsupervised
  - Entropy-based:
    - Top-down, supervised
  - …

# Tài liệu tham khảo


[1] *Data mining concepts and techniques*, Jiawei Han, Micheline Kamber, 2006

[2] *Data mining slide*, Jiavei Han, 2008

[3] *Haar Wavelet Analysis*, Albert Boggess Francis J. Narcowich, 2001

[4] *Haar Wavelets*, Jyun-Ming Chen, 2001

[5] http://mathworld.wolfram.com

[6] http://www.aiaccess.net/English/home.htm


Q&A

# CẢM ƠN THẦY VÀ CÁC BẠN ĐÃ CHÚ Ý THEO DÕI