

# **DATA MINING**

## **Pertemuan 2: Data and Data Preprocessing**

# Data pada Database

- Karakteristik:
  - Sangat rentan terhadap *noise*
  - Memiliki *missing/inconsistent data*
- Data yang berkualitas rendah akan menghasilkan hasil mining yang berkualitas rendah juga
- Solusi: melakukan pra-pemrosesan pada data terlebih dahulu sebelum melakukan proses mining

# Ringkasan Deskriptif Data

- Landasan analitik untuk data pre-processing, mencakup:
  - Pengukuran statistik dasar
    - Rata-rata, rata-rata terbobot, median, modus, range, kuartil, variance, standar deviasi
  - Representasi grafis, seperti:
    - Histogram, boxplot, scatter plot, matriks scatter-plot

# Contoh Representasi Grafis: Scatter Plot

- Scatter Plot dapat digunakan untuk melihat secara visual apakah ada korelasi pada data atau tidak

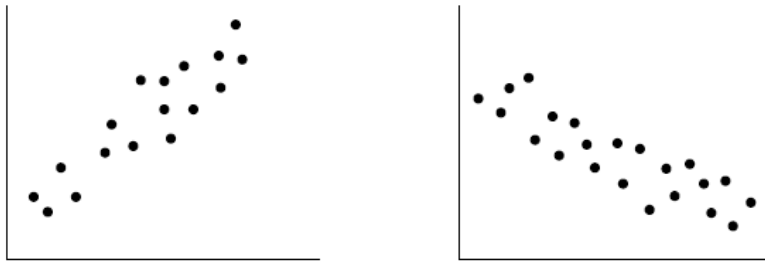


Figure 2.8 Scatter plots can be used to find (a) positive or (b) negative correlations between attributes.

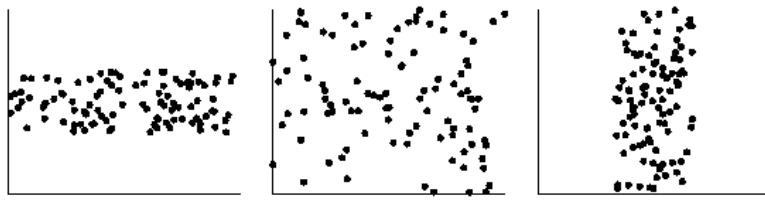
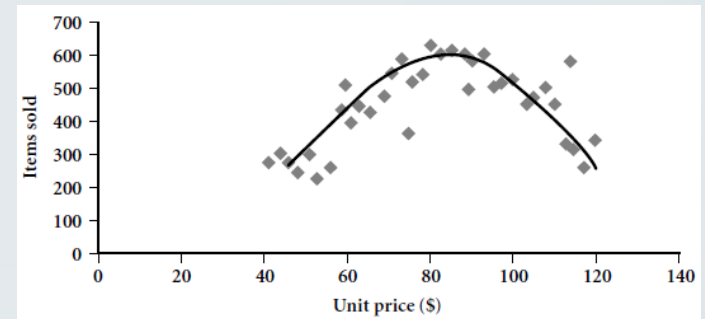


Figure 2.9 Three cases where there is no observed correlation between the two plotted attributes in each of the data sets.



# Data Preprocessing

- **Data cleaning**
  - Menghilangkan noise dan memperbaiki ketidakkonsistenan data
- **Data integration**
  - Melakukan proses *merge data* dari berbagai sumber menjadi 1 sumber → data warehouse
- **Data transformation**
  - Mengubah format data, misal melalui proses normalisasi
- **Data reduction**
  - Mengurangi data dengan menghilangkan beberapa atribut yang redundan, menggabung beberapa atribut
- **Data discretization**
  - Membagi range atribut ke dalam interval-interval

# 1. Data Cleaning

- Menangani missing values
  - Mengabaikan kolom
  - Mengisi nilai yang kosong secara manual
  - Menggunakan konstanta global (misal: “-” atau “ $\infty$ ”)
  - Mengisi dengan nilai rata2 atribut
  - Mengisi dengan nilai yang paling memungkinkan, misal dengan metode regresi atau aturan Bayes

# 1. Data Cleaning (lanjutan)

- Menangani noisy data
  - Noise: error random yang muncul pada data
  - Data yang *noisy* perlu dihaluskan (*smoothing*)
  - *Teknik 1: Binning*
    - Mengurutkan nilai data dan menempatkannya pada sejumlah “tong” atau “kantong” (*bin*)

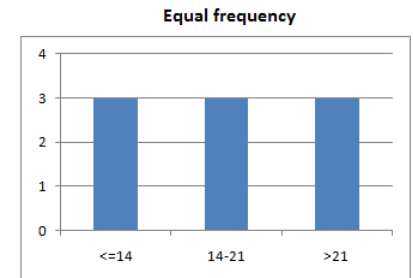
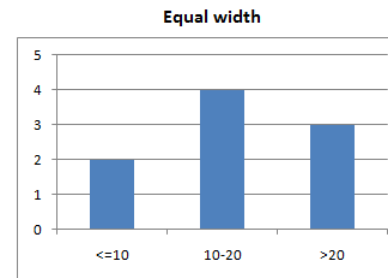
Sorted data for *price* (in dollars): 4, 8, 15, 21, 21, 24, 25, 28, 34

Partition into (equal-frequency) bins:

Bin 1: 4, 8, 15

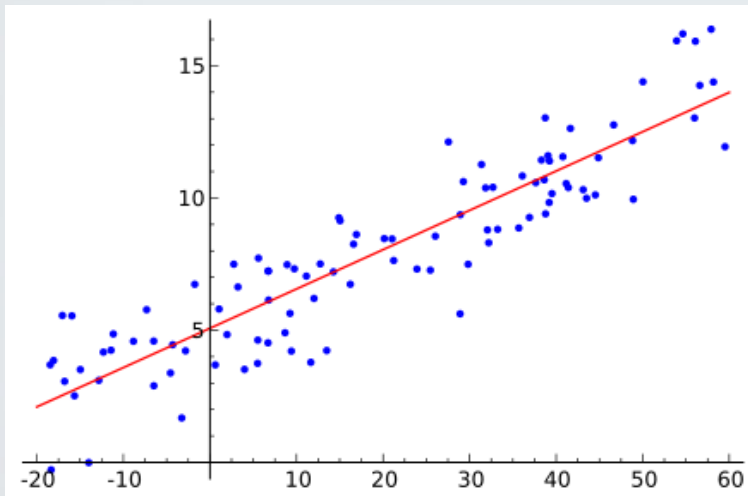
Bin 2: 21, 21, 24

Bin 3: 25, 28, 34

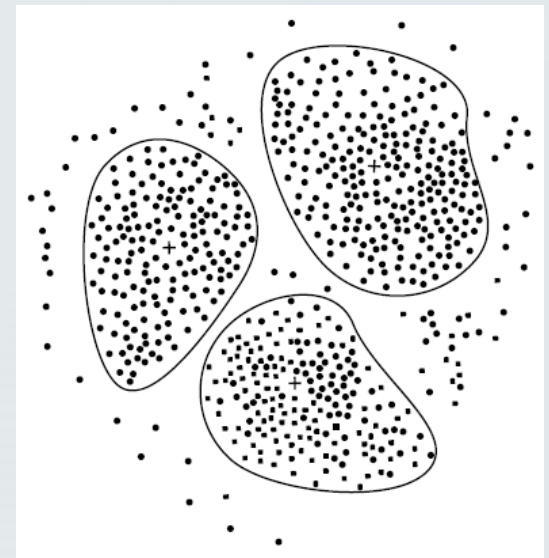


# 1. Data Cleaning (lanjutan)

- Teknik 2: Regresi
  - Memplot data pada sebuah fungsi. Contoh: dengan Regresi Linier



- Teknik 3: Clustering
  - Mengelompokkan beberapa nilai yang mirip menjadi 1 kluster/kelompok
  - Dapat mendeteksi data *outlier*





## 2. Data Integration & Transformation

- Integrasi data: penggabungan beberapa data dari berbagai tabel
- Agar integrasi berjalan lancar, perlu untuk memperhatikan metadata tiap atribut (seperti: nama, makna, tipe data, range nilai)

## 2. Data Integration & Transformation (lanjutan)

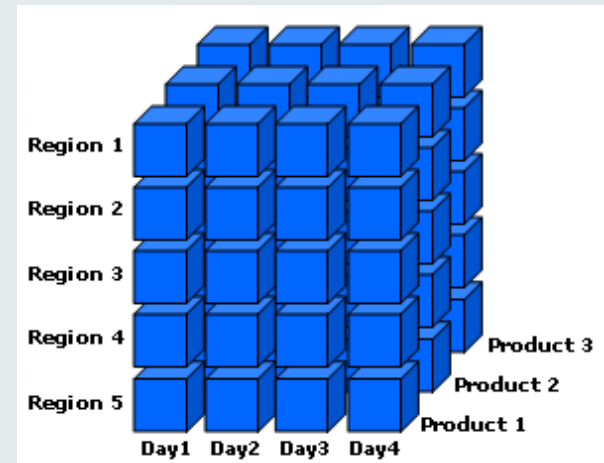
- Transformasi data: mentransformasi data ke bentuk yang sesuai untuk proses mining. Beberapa prosesnya:
  - **Smoothing** → binning, regression, clustering
  - **Aggregation** → merekap data (misal: bulanan, tahunan)
  - **Generalization** → nama jalan menjadi nama kota, nilai umur menjadi muda, paruh baya, tua
  - **Normalization** → data di-skala-kan agar berada pada range yang ditentukan, misal antara 0 dan 1
  - **Attribute construction** → menambahkan atribut baru dari sekumpulan atribut yang sudah ada

# 3. Data Reduction

- Mengurangi ukuran data agar lebih mudah di-mining.
- Harus tetap menjaga integritas data aslinya
- Beberapa teknik data reduction:
  - Data cube aggregation
  - Attribute subset selection
  - Dimensionality reduction
  - Numerosity reduction
  - Discretization dan concept hierarchy generation

# 3. Data Reduction (lanjutan)

- **Data Cube Aggregation**  
(Agregasi Data Kubik)
  - Menampilkan data dalam bentuk 3 dimensi
  - Memudahkan untuk melihat rekapan data



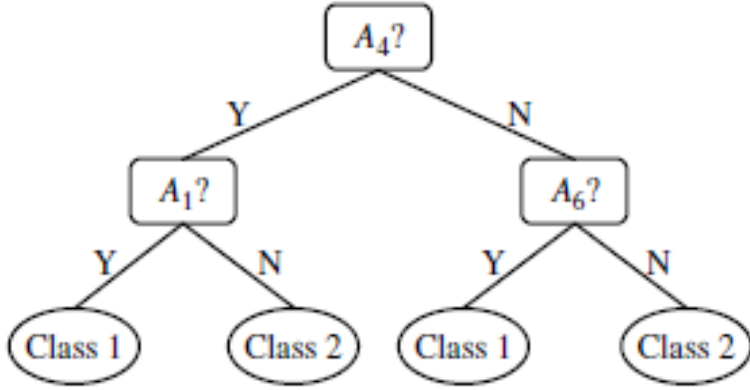
### 3. Data Reduction (lanjutan)

- **Attribute Subset Selection** (Pemilihan subhimpunan atribut)
  - Mengurangi ukuran data dengan menghilangkan atribut (atau dimensi) yang kurang relevan atau redundan
  - Tujuan: Mencari himpunan terkecil dari sekumpulan atribut yang probabilitas distribusinya sedekat mungkin dengan probabilitas distribusi dengan keseluruhan atribut

# 3. Data Reduction (lanjutan)

- Beberapa metode *attribute subset selection*:
  - **Stepwise forward selection**
    - Mulai dari himpunan kosong dan menambahkan satu-persatu atribut
  - **Stepwise backward selection**
    - Mulai dari seluruh atribut dan mengurangi satu-persatu
  - **Combination of forward and backward elimination**
    - Pada tiap tahap, dapat menambahkan atribut yang baik dan mengurangi atribut yang buruk
  - **Decision tree induction**
    - Membuat flowchart tree untuk melakukan tes pada atribut

# 3. Data Reduction (lanjutan)

Forward selection	Backward elimination	Decision tree induction
<p>Initial attribute set:  <math>\{A_1, A_2, A_3, A_4, A_5, A_6\}</math></p> <p>Initial reduced set:  <math>\{\}</math>  <math>\Rightarrow \{A_1\}</math>  <math>\Rightarrow \{A_1, A_4\}</math>  <math>\Rightarrow</math> Reduced attribute set:  <math>\{A_1, A_4, A_6\}</math></p>	<p>Initial attribute set:  <math>\{A_1, A_2, A_3, A_4, A_5, A_6\}</math>  <math>\Rightarrow \{A_1, A_3, A_4, A_5, A_6\}</math>  <math>\Rightarrow \{A_1, A_4, A_5, A_6\}</math>  <math>\Rightarrow</math> Reduced attribute set:  <math>\{A_1, A_4, A_6\}</math></p>	<p>Initial attribute set:  <math>\{A_1, A_2, A_3, A_4, A_5, A_6\}</math></p>  <pre> graph TD     A4["A4?"] -- Y --&gt; A1["A1?"]     A4 -- N --&gt; A6["A6?"]     A1 -- Y --&gt; C1_1((Class 1))     A1 -- N --&gt; C2_1((Class 2))     A6 -- Y --&gt; C1_2((Class 1))     A6 -- N --&gt; C2_2((Class 2))     </pre> <p><math>\Rightarrow</math> Reduced attribute set:  <math>\{A_1, A_4, A_6\}</math></p>

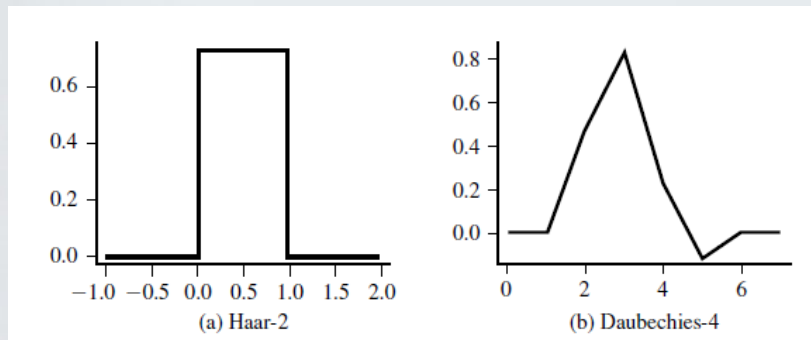
# 3. Data Reduction (lanjutan)

- ***Dimensionality Reduction*** (Reduksi Dimensi)
  - Data dikurangi dengan mengompresi data dengan cara encoding
  - Beberapa teknik dimensionality reduction:
    - Wavelet Transforms
      - Mentransformasi vektor data  $X$  ke vektor lain  $X'$  dengan koefisien wavelet
    - Principal Component Analysis (PCA)
      - Memetakan data dengan  $n$  atribut ke sejumlah  $k$  vektor dengan dimensi  $n$ , dimana  $k \leq n$
    - Singular Value Decomposition (SVD)
      - Faktorisasi dari matriks bilangan real atau kompleks



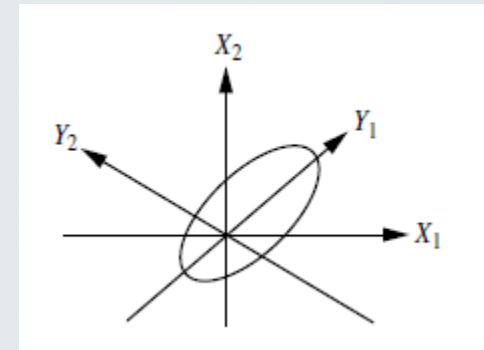
# 3. Data Reduction (lanjutan)

## Transformasi Wavelet



Contoh nama Wavelet

## PCA



$Y_1$  dan  $Y_2$  adalah dua komponen prinsip pertama dari data

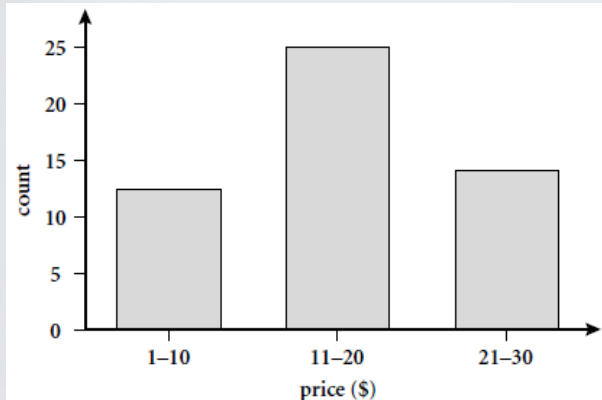
PCA bagus digunakan untuk data yang renggang (*sparse*)

Wavelet bagus digunakan untuk data yang memiliki dimensionalitas tinggi

## 4. Numerosity Reduction

- Menerapkan teknik untuk menyimpan data dalam representasi yang lebih kecil
- Dapat berupa *parametric* dan *non-parametric*:
  - Parametric: menggunakan model untuk mengestimasi data (sehingga hanya parameter data yang disimpan, bukan data yang sesungguhnya). Contoh: model log-linier dan regresi (linier atau multiple linier)
  - Non-parametric: menyimpan representasi tereduksi dari data. Contoh: histogram, clustering, sampling

# 4. Numerosity Reduction (lanjutan)



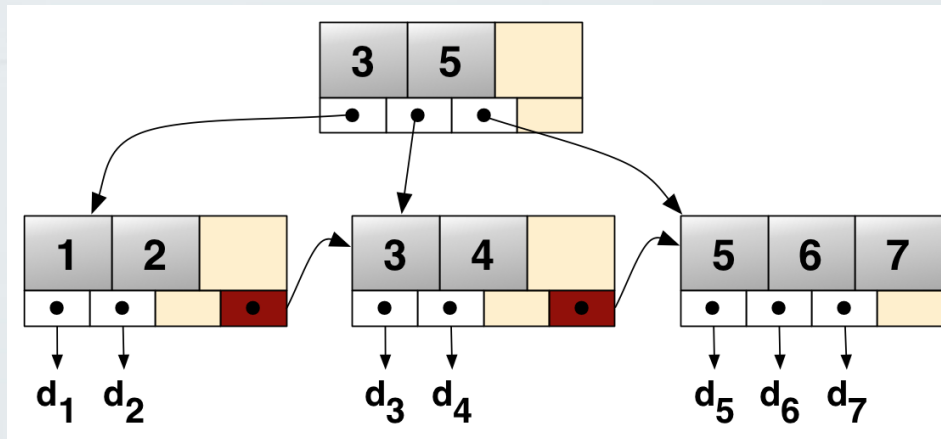
Histogram

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

Stratified Sampling



Clustering: B+ tree

# 5. Data Discretization

- Mengubah data numerik ke data nominal (membagi ke dalam interval)
- Beberapa teknikanya:
  - **Binning**
  - **Histogram Analysis**
  - **Entropy-based Discretization** : menggunakan nilai entropi atribut untuk mempartisi range atribut
  - **$\chi^2$ -merging** (chi square merging): melakukan tes  $\chi^2$  untuk menemukan interval tetangga terdekat dan melakukan proses merging untuk membentuk interval yang lebih besar
  - **Cluster Analysis**: menggunakan algoritma clustering untuk mengelompokkan atribut

# Pre-processing pada Weka

Weka Explorer

Preprocess | Classify | Cluster | Associate | Select attributes | Visualize

Open file... Open URL... Open DB... Generate... Undo Edit... Save...

Filter

- weka
  - filters
    - AllFilter
    - MultiFilter
  - supervised
    - attribute
      - AddClassification
      - AttributeSelection
      - ClassOrder
      - Discretize
      - NominalToBinary
      - PLSFilter
  - instance
  - unsupervised
    - attribute
      - Add
      - AddCluster
      - AddExpression
      - AddID
      - AddNoise
      - AddValues
      - Center
      - ChangeDateFormat
      - ClassAssigner
      - ClusterMembership

Pattern

Selected attribute

Name: sepal.length  
Missing: 0 (0%)  
Distinct: 35  
Type: Numeric  
Unique: 9 (6%)

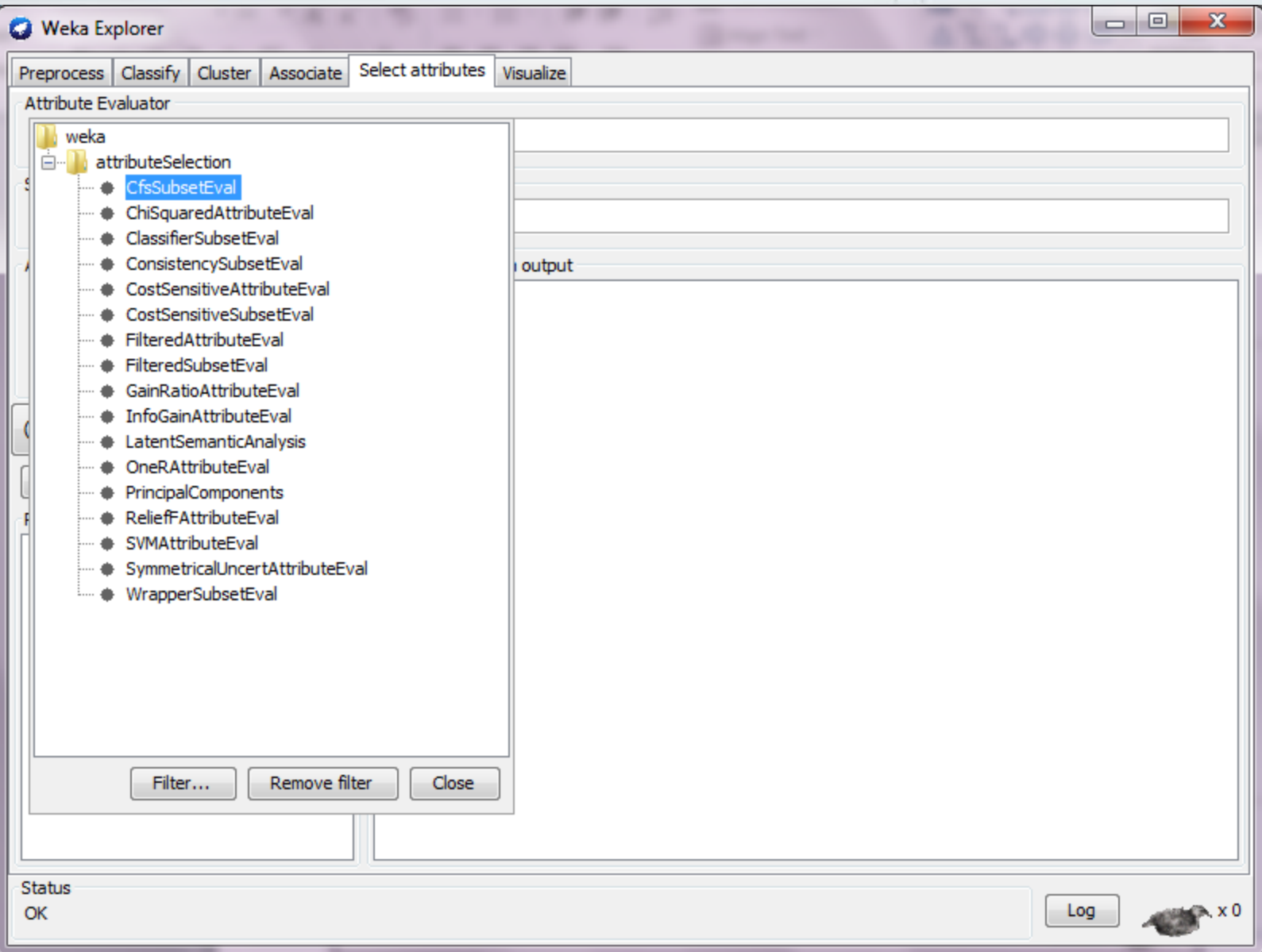
Statistic	Value
Minimum	4.3
Maximum	7.9
Mean	5.843
StdDev	0.828

Class: class (Nom) Visualize All

16 30 34 28 25 10 7

4.3 6.1 7.9

Status OK Log x 0



# Quiz

- Dalam data riil, sering ditemui tabel yang memiliki *missing value*. Bagaimana cara untuk menangani masalah ini?
- Apa yang dimaksud dengan *data cube* (kubik data)?
- Apa yang dimaksud dengan outlier?
- Bagaimana cara kerja dari teknik Attribute Subset Selection?
- Apa yang dimaksud dengan *binning*?

# Tambahan: Normalisasi *min-max* untuk transformasi data

- Misal *minA* dan *maxA* adalah nilai min dan max dari sebuah atribut A
- Normalisasi min-max memetakan sebuah nilai  $v$  ke  $v'$  dalam range [*new\_minA*, *newMaxA*] dengan rumus:

$$v' = \frac{v - \text{min}_A}{\text{max}_A - \text{min}_A} (\text{new\_max}_A - \text{new\_min}_A) + \text{new\_min}_A$$



# Contoh

- Misal nilai min dan max dari sebuah atribut gaji adalah \$12,000 dan \$98,000. Kita ingin memetakan atribut gaji ke dalam range [0.0, 1.0]
- Dengan normalisasi min-max, gaji sebesar \$73,600 ditransformasi menjadi bernilai 
$$\frac{73,600 - 12,000}{98,000 - 12,000} (1.0 - 0.0) + 0 = 0.716$$

# Tambahan: Normalisasi z-score untuk transformasi data

- Nilai dari atribut A dinormalisasi menggunakan rata-rata (*mean*) dan standar deviasi dari A. Nilai  $v$  ditransformasi ke  $v'$  dengan persamaan:

$$v' = \frac{v - \bar{A}}{\sigma_A}$$

$\bar{A}$  = rata – rata (*mean*) data pada atribut A

$\sigma_A$  = Standar deviasi data pada atribut A

# Contoh

- Misal nilai rata-rata dan standar deviasi dari atribut gaji sebesar \$54,000 dan \$16,000. Dengan normalisasi z-score, maka nilai \$73,600 ditransformasi menjadi  $\frac{73,600 - 54,000}{16,000} = 1.225$
- Metode normalisasi z-score berguna ketika nilai min dan max dari A tidak diketahui, atau terdapat outlier yang mendominasi normalisasi min-max
- Nilai z positif jika di atas rata-rata dan negatif jika di bawah rata-rata

# Latihan

- Diketahui data usia (dalam tahun) sebagai berikut (urut menaik): 13, 15, 16, 16, 19, 20, 20, 21, 22, 22, 25, 25, 25, 25, 30, 33, 33, 35, 35, 35, 35, 36, 40, 45, 46, 52, 70.
  - Lakukan normalisasi min-max untuk mentransformasi usia 35 ke range [0.0, 1.0]
  - Gunakan normalisasi z-score untuk mentransformasi usia 35 jika diketahui standar deviasinya sebesar 12.94 tahun

# Next Week

- Exploratory Data Analysis