Data Preparation and Reduction

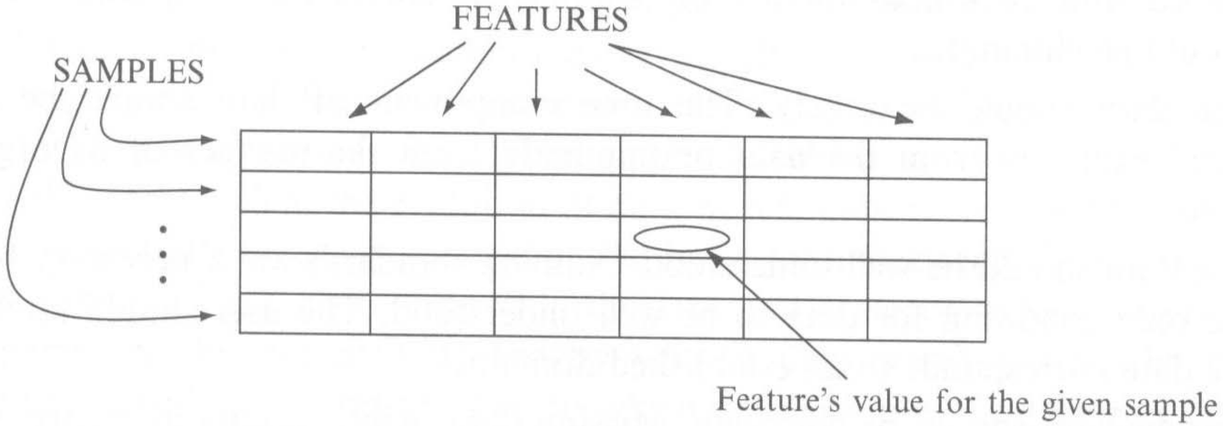
Berlin Chen 2004

References:

1. Data Mining: Concepts, Models, Methods and Algorithms, Chapters 2, 3
2. Data Mining: Concepts and Techniques, Chapters 3, 8

Data Samples

- Large amounts of samples with different types of features (attributes)
- Each sample is described with several features
  - Different types of values for every feature
    - Numeric: real-value or integer variables
      - Support “order” and “distance” relations
    - Categorical: symbolic variables
      - Support “equal” relation



Data Samples

- Another way of classification of variables
  - Continuous variables
    - Also called *quantitative* or *metric* variables
    - Measured using interval or ratio scales
      - Interval: e.g., temperature scale
      - Ratio: e.g., height, length,.. (has an absolute zero point)
  - Discrete variables
    - Also called *qualitative* variables
    - Measured using nonmetric scales (nominal, ordinal)
      - Nominal: e.g., (A,B,C, ...), (1,2,3, ...)
      - Ordinal: e.g., (young, middle-aged, old), (low, middle-class, upper-middle-class, rich), …
    - A special class of discrete variable: periodic variables
      - Weekdays (Monday, Tuesday,..): distance relation exists

Data Samples

- On additional dimension of classification of data
  - Static data
    - Attribute values do not change with time
  - Dynamic (temporal) data
    - Attribute values change with time

Curse of Dimensionality

- Data samples are very often high dimensional
  - Extremely large number of measurable features
  - The properties of high dimensional spaces often appear counterintuitive
  - High dimensional spaces have a larger surface area for a given volume
  - Look like a porcupine after visualization

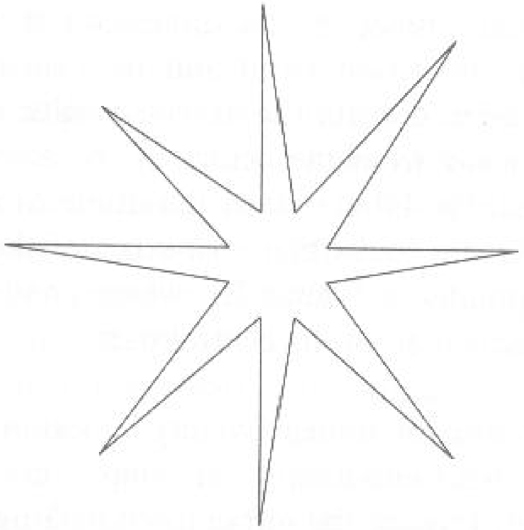


FIGURE 2.1 High-dimensional data looks conceptually like a porcupine

Curse of Dimensionality

- Four important properties of high dimensional data
  1. The size of a data set yielding the same density of data points in an *n*-dimensional space increases exponentially with dimensions
  2. A large radius is needed to enclose a fraction of the data points in a high dimensional space

With the same density

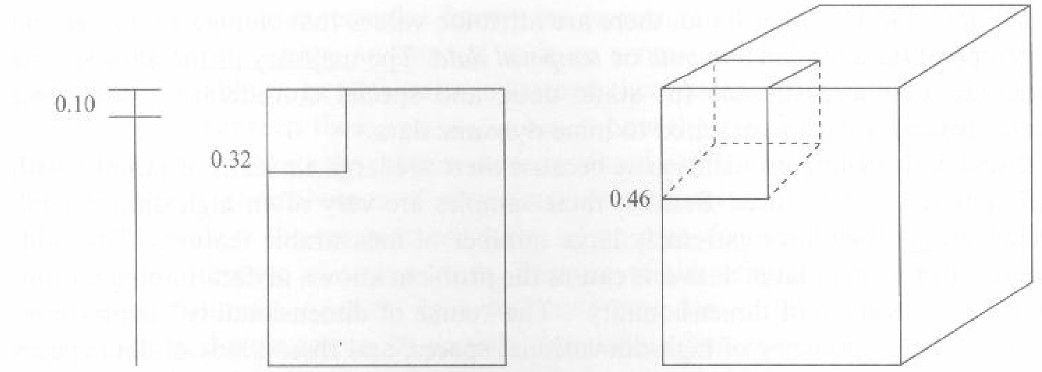


FIGURE 2.2 Regions enclose 10% of the samples for 1-, 2-, and 3-dimensional spaces

$$e_d(p) = p^{1/d}$$

dimensionality

fraction of samples

Curse of Dimensionality

3. Almost every point is closer to an edge than to another sample point in a high dimensional space

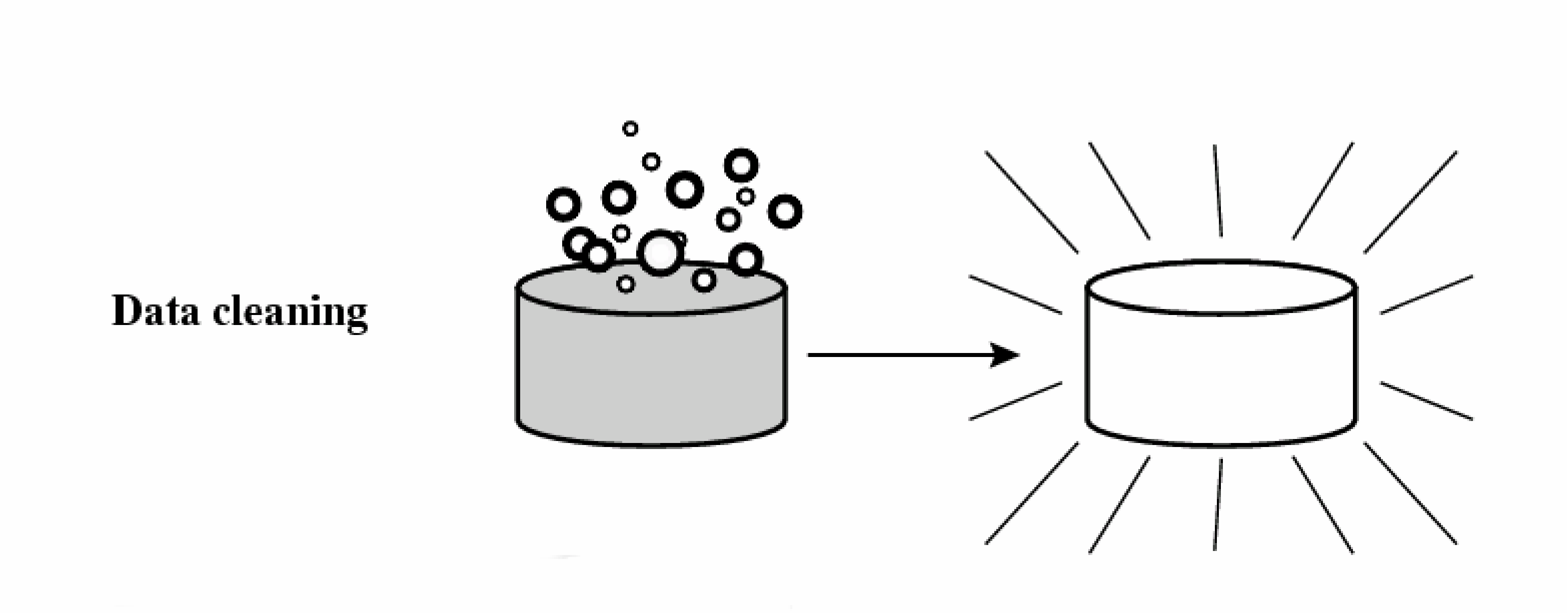
4. Almost every point is an outlier. The distance between the prediction point and the center of the classified points increases

With the same number of samples

Central Tasks for Data Preparation

- Organize data into a standard form that is ready for processing by data-mining and other computer-based tools

- Prepare data set that lead to the best data-mining performances



Transformation of Raw Data

- Data transformation can involve the following
  - Normalizations
  - Data Smoothing
  - Differences and Ratios (attribute/feature construction)
  - ….

Attention should be paid to data transformation, because relatively simple transformations can sometimes be far more effective for the final performance !

Normalizations

- For data with features are based on distance computation between points in an *n*-dimensional space
  - Scaled values to a specific range, e.g., [-1,1] or [0,1]
  - Avoid overweighting those features that have large values

1. Decimal Scaling:
   - Move the decimal point but still preserves most of the original digital value

$$v'(i) = v(i)/10^{k}$$

$$\text{for small } k \text{ such that } \max(|v'|) < 1$$

$$\left.\begin{array}{r} \text{largest} = 455 \\ \text{smallest} = -834 \end{array}\right\} \Rightarrow k = 3 \qquad \left.\begin{array}{r} \text{largest} = 250 \\ \text{smallest} = 150 \end{array}\right\} \Rightarrow k = 3$$

$$(-0.834 \sim 0.455) \qquad (0.15 \sim 0.25)$$

Normalizations

2. Min-Max Normalization:

- Normalized to be in [0, 1]

$$v'(i) = \frac{v(i) - \min(v)}{(\max(v) - \min(v))}$$

- Normalized to be in [-1, 1]

$$v'(i) = 2\left[\frac{v(i) - \min(v)}{(\max(v) - \min(v))} - 0.5\right]$$

- The automatic computation of min and max value requires one additional search through the entire data set
- It may be dominated by the outliers
- It will encounter an “out of bounds” error !

Normalizations

3. Standard Deviation Normalization

- Also called *z-score* or *zero-mean* normalization
- The values of an attribute are normalized based on the mean and standard deviation of it
- Mean and standard deviation are first computed for the entire data set

$$v'(i) = \frac{v(i) - mean(v)}{sd(v)}$$

$$\bar{v} = mean(v) = \frac{\sum v}{n_v}$$

$$\sigma_v = sd(v) = \sqrt{\frac{\sum (v - \bar{v})^2}{n_v - 1}}$$

- E.g., the initial set of values of the attribute $v = \{1, 2, 3\}$ has

$$mean(v) = 1,\ \ sd(v) = 1 \text{ and new set of } v' = \{-1, 0, 1\}$$

Data Smoothing

- Minor differences between the values of a feature (attribute) are not significant and may degrade the performance of data mining
  - They may be caused by noises

- Reduce the number of distinct values for a feature
  - E.g., round the values to the given precision

$$F = \{0.93, 1.01, 1.001, 3.02, 2.99, 5.03, 5.01, 4.98\}$$
$$\Rightarrow F_{smoothed} = \{1.0, 1.0, 1.0, 3.0, 3.0, 5.0, 5.0, 5.0\}$$

  - The dimensionality of the data space is also reduced at the same time

Differences and Ratios

- Can be viewed as a kind of attribute/feature construction
  - New attributes are constructed from the given attributes
  - Can discover the missing information about the relationships between data attributes
  - Can be applied to the *input* and *output* features for data mining

- E.g.,
  1. Difference
     - E.g., "*s*(*t*+1) - *s*(*t*)", relative moves for control setting
  2. Ratio
     - E.g.,"*s*(*t*+1) / *s*(*t*)", levels of increase or decrease
     - E.g., Body-Mass Index (BMI) $Weight(Kg) \big/ Height(m^2)$

Missing Data

- In real-world application, the subset of samples or future cases with complete data may be relatively small
  - Some data mining methods accept missing values
  - Others require all values be available
    - Try to drop the samples or fill in the missing attribute values in during data preparation

Missing Data

- Two major ways to deal with missing data (values)
  1. Reduce the data set and eliminate all samples with missing values

  2. Find values for missing data
     - a. Domain experts examine and enter reasonable, probable, and expected values for the missing data

     - b. Replace missing values with some constants
       - b.1 Replace a missing value with a single global constant
       - b.2 Replace a missing value with its feature mean
       - b.3 Replace a missing value with its feature mean for the given class
       - b.4 Replace a missing value with the most probable value (e.g., according to the values of other attributes of the present data)

will bias the data

Missing Data

- The replaced value(s) (especially for b.1~b.3) will homogenize the cases / samples with missing values into an artificial class

- Other solutions

  1. "Don't Care"

     - Interpret missing values as "don't care" values

       $$\vec{x} = \langle 1,\ ?,\ 3 \rangle, \text{ with feature values in domain } [0,1,2,3,4\,]$$

       $$\Rightarrow \vec{x}_1 = \langle 1,\ 0,\ 3 \rangle, \vec{x}_2 = \langle 1,\ 1,\ 3 \rangle, \vec{x}_3 = \langle 1,\ 2,\ 3 \rangle, \vec{x}_4 = \langle 1,\ 3,\ 3 \rangle, \vec{x}_5 = \langle 1,\ 4,\ 3 \rangle$$

     - A explosion of artificial samples being generated !

  2. Generate multiple solutions of data-mining with and without missing-value features and then analyze and interpret them !

Time-Dependent Data

- Time-dependent relationships may exist in specific features of data samples
  - E.g., “temperature reading” and speech are a univariate time series, and video is a multivariate time series

$$X = \{t(0), t(1), t(2), t(3), t(4), t(5), t(6), t(7), t(8), t(9), t(10)\}$$

- Forecast or predict *t*(*n*+1) from previous values of the feature

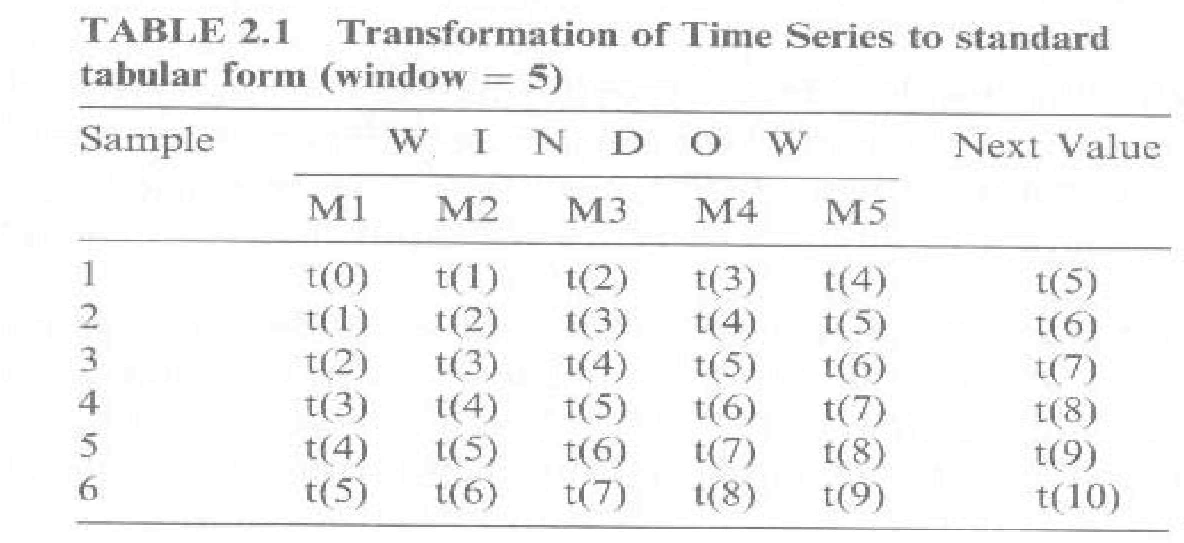


TABLE 2.1 Transformation of Time Series to standard tabular form (window = 5)

Sample	WINDOW					Next Value
	M1	M2	M3	M4	M5	
1	t(0)	t(1)	t(2)	t(3)	t(4)	t(5)
2	t(1)	t(2)	t(3)	t(4)	t(5)	t(6)
3	t(2)	t(3)	t(4)	t(5)	t(6)	t(7)
4	t(3)	t(4)	t(5)	t(6)	t(7)	t(8)
5	t(4)	t(5)	t(6)	t(7)	t(8)	t(9)
6	t(5)	t(6)	t(7)	t(8)	t(9)	t(10)

Time-Dependent Data

- Forecast or predict *t*(*n*+*j*) from previous values of the feature

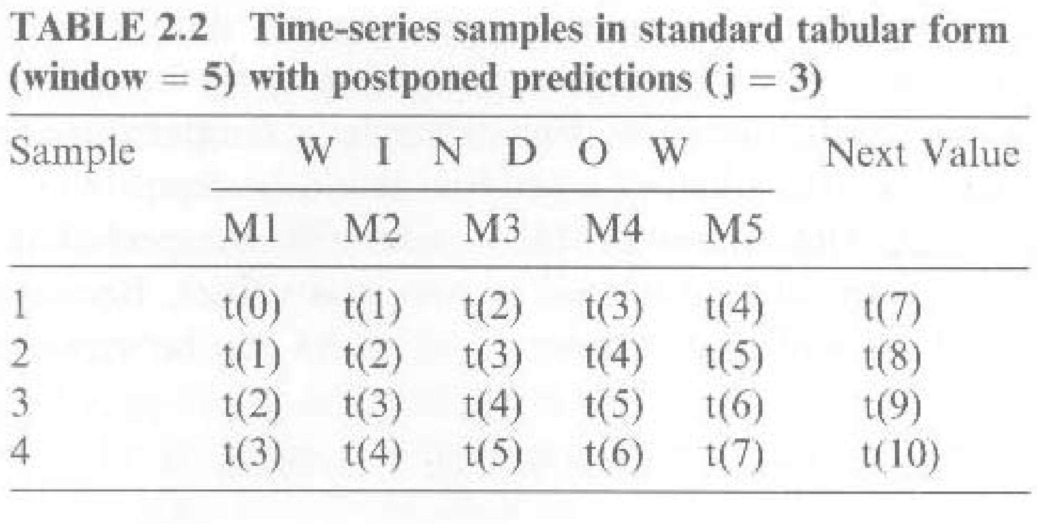


TABLE 2.2 Time-series samples in standard tabular form (window = 5) with postponed predictions (j = 3)

Sample	WINDOW					Next Value
	M1	M2	M3	M4	M5	
1	t(0)	t(1)	t(2)	t(3)	t(4)	t(7)
2	t(1)	t(2)	t(3)	t(4)	t(5)	t(8)
3	t(2)	t(3)	t(4)	t(5)	t(6)	t(9)
4	t(3)	t(4)	t(5)	t(6)	t(7)	t(10)

- As mentioned earlier, forecast or predict the differences or ratios of attribute values
  - *t*(*n*+1) - *t*(*n*)
  - *t*(*n*+1) / *t*(*n*)

Time-Dependent Data

- "Moving Average" (MA)– a single average summarizes the most *m* feature values for each case at each time moment *i*
  - Reduce the random variation and noise components

$$MA(i, m) = \frac{1}{m} \cdot \sum_{j=i-m+1}^{i} \tilde{t}(j),$$

$$\tilde{t}(j): \text{noisy data}$$

$$\tilde{t}(j) = t(j) + error, \quad \text{error is assumed to be a constant}$$

$$\Rightarrow MA(i, m) = \frac{1}{m} \cdot \sum_{j=i-m+1}^{i} \tilde{t}(j) = mean(j) + error$$

$$, where \quad mean(j) = \sum_{j=i-m+1}^{i} t(j)$$

$$\Rightarrow \tilde{t}(j) - MA(i, m) = t(j) - mean(j)$$

Time-Dependent Data

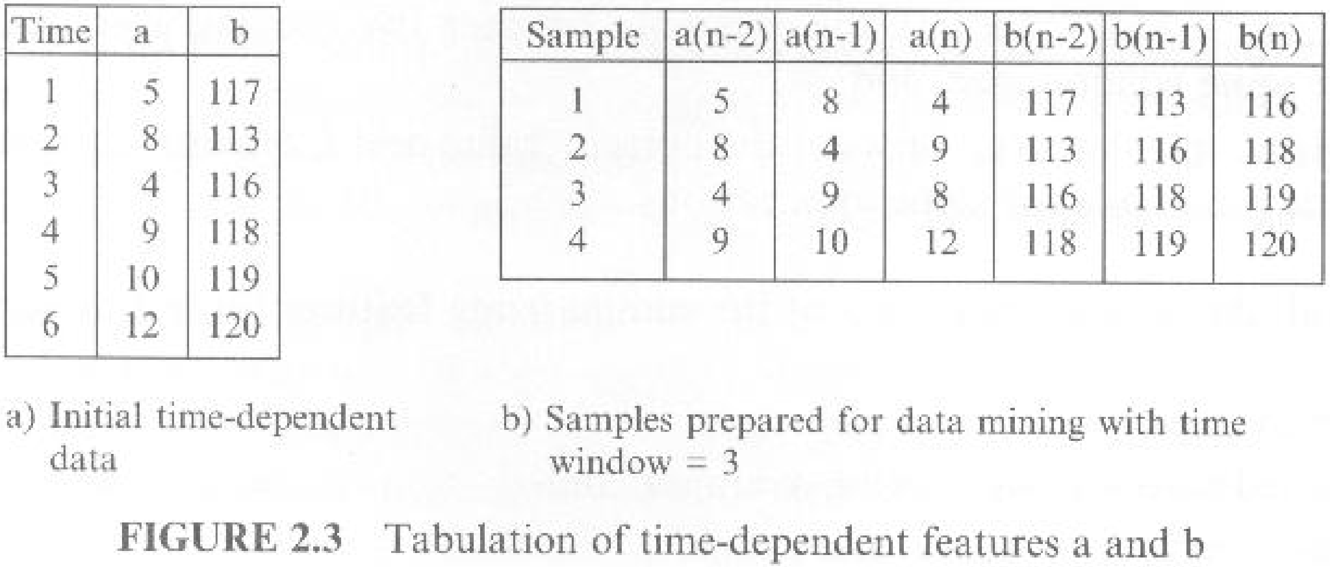
- “Exponential Moving Average” (EMA) – give more weight to the most recent time periods

$$EMA(i) = p \cdot t(i) + (1 - p) \cdot EMA(i-1)$$

$$EMA(i-1) = p \cdot t(i-1) + (1 - p) \cdot EMA(i-2)$$
$$.....$$
$$EMA(2) = p \cdot t(2) + (1 - p) \cdot EMA(1)$$
$$EMA(1) = t(1)$$

Time-Dependent Data

- Example: multivariate time series



Time	a	b
1	5	117
2	8	113
3	4	116
4	9	118
5	10	119
6	12	120

a) Initial time-dependent data

Sample	a(n-2)	a(n-1)	a(n)	b(n-2)	b(n-1)	b(n)
1	5	8	4	117	113	116
2	8	4	9	113	116	118
3	4	9	8	116	118	119
4	9	10	12	118	119	120

b) Samples prepared for data mining with time window = 3

FIGURE 2.3 Tabulation of time-dependent features a and b

High dimensions of data generated during the transformation of time-dependent can be reduced through "data reduction"

Outlier Analysis

- Outliers
  - Data samples that do not comply the general behavior of the data model and are significantly different or inconsistent with the remaining set of data
  - E.g., a person's age is "-999", the number of children for one person is "25", ….

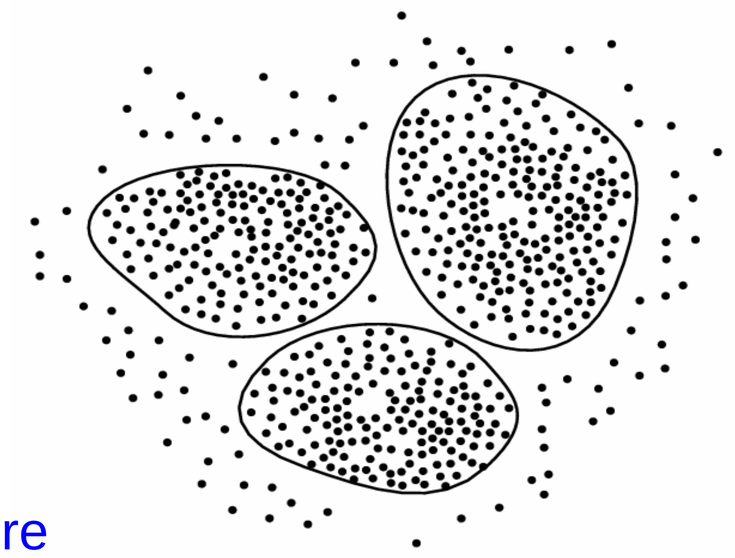
- Many data-mining algorithms try to minimize the influence of outliers or eliminate them all together
  - However, it could result in the loss of important hidden information
  - "one person's noise could be another person's signal", e.g., outliers may indicate abnormal activity

Outlier Analysis

- Applications:
  - Credit card fraud detection
  - Telecom fraud detection
  - Customer segmentation
  - Medical analysis

Outlier Analysis

- Outlier detection/mining
  - Given a set of *n* samples, and *k*, the expected number of outliers, find the top *k* samples that are considerably dissimilar, exceptional, or inconsistent with respect to the remaining data
  - Can be viewed as two subproblems
    - Define what can be considered as inconsistent in a given data set
      - Nontrivial
    - Find an efficient method to mine the outliers so defined
      - Three methods introduced here

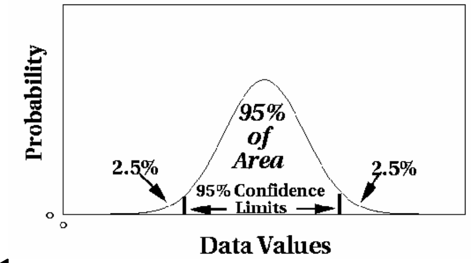


Visual detection of outlier ?

Outlier Analysis

1. Statistical-based Outlier Detection

- Assume a distribution or probability model for the given data set and then identifies outliers with respect to the model using a *discordance* test
  - Data distribution (e.g. normal distribution)
  - Distribution parameters: mean, variance



$$Age = \{3, 56, 23, 39, 156, 52, 41, 22, 9, 28, 139, 31,$$
$$55, 20, -67, 37, 11, 55, 45, 37\}$$

$$Mean = 39.9$$

$$Standard\ d\ eviation = 45.65$$

$$Threshold = Mean \pm 2 \times Standard\ d\ eviation$$

$[-54., 131.2] \Rightarrow [0, 131.2]$ Age is always greater than zero !

$\Rightarrow outliers : 156, 139, -67$

Outlier Analysis

1. Statistical-based Outlier Detection
   - Drawbacks
     - Most tests are for single attribute
     - In many cases, data distribution may not be known

Outlier Analysis

2. Distance-based Outlier Detection

- A sample s_i in a data *S* is an outlier if at least a fraction *p* of the objects in *S* lies at a distance greater than *d,* denoted as *DB<p, d>*

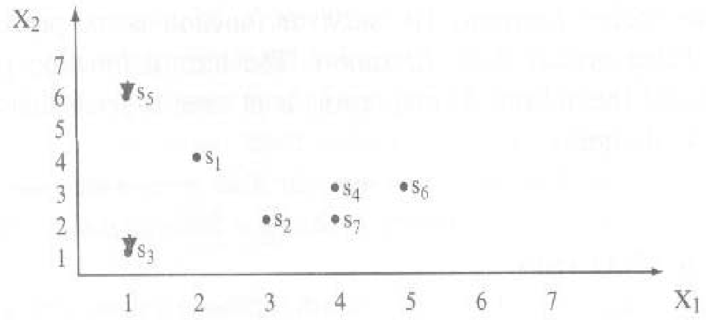


FIGURE 2.4 Visualization of two-dimensional data set for outlier detection

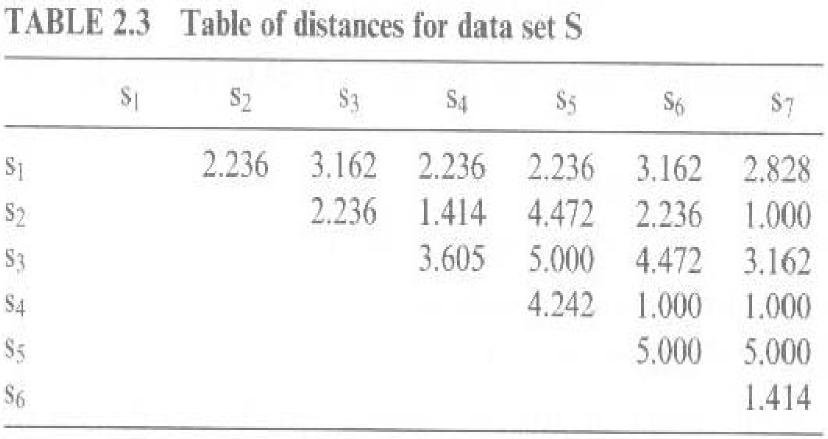


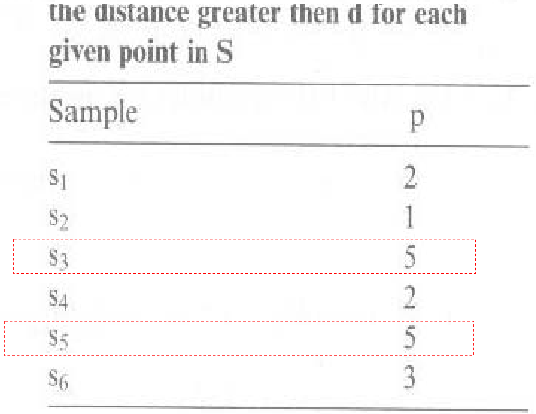
TABLE 2.3 Table of distances for data set S

	s_1	s_2	s_3	s_4	s_5	s_6	s_7
s_1		2.236	3.162	2.236	2.236	3.162	2.828
s_2			2.236	1.414	4.472	2.236	1.000
s_3				3.605	5.000	4.472	3.162
s_4					4.242	1.000	1.000
s_5						5.000	5.000
s_6							1.414

- If *DB<p, d>=DB<4, 3>*

$$d = \left[(x_1 - x_1)^2 + (y_1 - y_1)^2 \right]^{1/2}$$

- Outliers: s_3, s_5



the distance greater then d for each given point in S

Sample	p
s_1	2
s_2	1
s_3	5
s_4	2
s_5	5
s_6	3

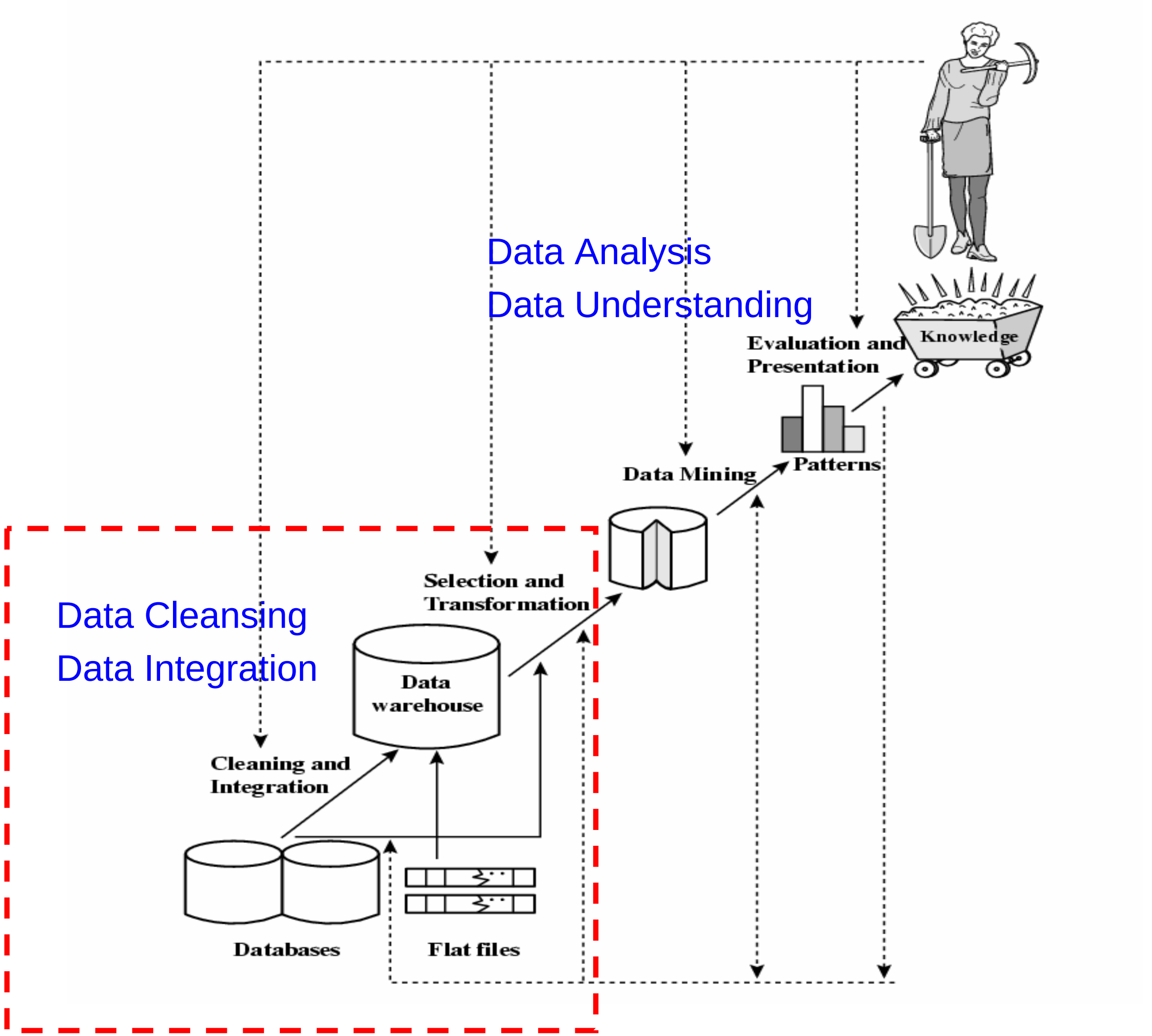
Outlier Analysis

3. Deviation-based Outlier Detection

- Define the basic characteristics of the sample set, and all samples that deviate from these characteristics are outliers

- The "sequence exception technique"
  - Based on a dissimilarity function, e.g., variance $\frac{1}{n}\sum_{i=1}^{n}\left(x_i - \overline{x}\right)^2$
  - Find the smallest subset of samples whose removal results in the greatest reduction of the dissimilarity function for the residual set (a NP-hard problem)

Where Are We Now ?



Data Analysis

Data Understanding

Data Cleansing

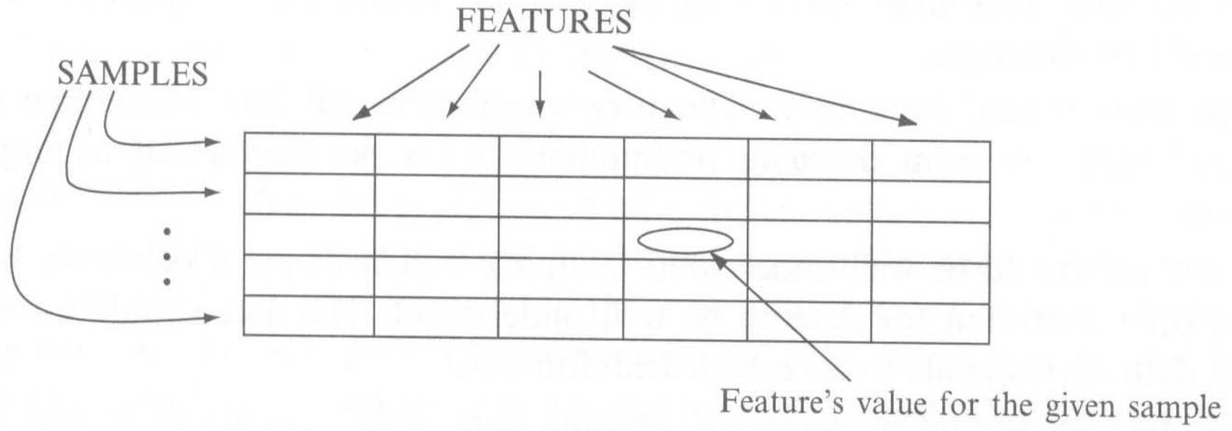
Data Integration

Data Preparation and Reduction

Introduction to Data Reduction

- Three dimensions of data sets
  - Rows (cases, samples, examples)
  - Columns (features)
  - *Values* of the features

- We hope that the final reduction doesn't reduce the quality of results, instead the results of data mining can be even improved



Introduction to Data Reduction

- Three basic operations in data reduction
  - Delete a column
  - Delete a row
  - Reduce the number of values in a column

Preserve the characteristic of original data
Delete the nonessential data

- Gains or losses with data reduction
  - Computing time
    - Tradeoff existed for preprocessing and data-mining phases
  - Predictive/descriptive accuracy
    - Faster and more accurate model estimation
  - Representation of the data-mining model
    - Simplicity of model representation (model can be better understood)
      - Tradeoff between simplicity and accuracy

Introduction to Data Reduction

- Recommended characteristics of data-reduction algorithms
  - Measure quality
    - Quality of approximated results using a reduced data set can be determined precisely
  - Recognizable quality
    - Quality of approximated results can be determined at preprocessing phrase
  - Monotonicity
    - Iterative, and monotonically decreasing in time and quality
  - Consistency
    - Quality of approximated results is correlated with computation time and input data quality
  - Diminishing returns
    - Significant improvement in early iterations and which diminished over time
  - Interruptability
    - Can be stopped at any time and provide some answers
  - Preemptability
    - Can be suspended and resumed with minimal overhead

Feature Reduction

- Also called “column reduction”
- Two standard tasks for producing a reduced feature set
  - Feature selection
    - – Objective: find a subset of features with performances comparable to the full set of features
  - Feature composition (do not discuss it here!)
    - – New features/attributes are constructed from the given/old features/attributes and those given ones are discarded later on !
    - – For example
      - » Body-Mass Index (BMI) $Weight(Kg)/Height(m^2)$
      - » New features/dimensions retained after principal component analysis (PCA)
    - – Interdisciplinary approaches and domain knowledge

Feature selection

- Select a subset of the features based domain knowledge and data-mining goals

$\{A_1, A_2, A_3\}$
$\Rightarrow \{0,0,0\}, \{1,0,0\}, \{0,1,0\}, \ldots, \{1,1,1\}$

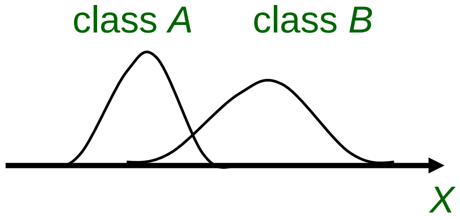
- Can be viewed as a search problem
- Manual or automated, supervised or unsupervised
- Methods can be classified as
  - Feature ranking algorithms
  - Minimum subset algorithms

Need a feature-evaluation scheme

  - Button-up: starts with an empty set and fill it in by choosing the most relevant features from the initial set of features
  - Top-down: begin with a full set of original features and remove one-by-one those that are irrelevant

Supervised Feature Selection

- Method 1: simply based on comparison of means and variances
  - Assume the distribution of the feature form a normal curve
  - Feature means of different categories/classes are normalized and then compared
    - If means are far apart → interest in a feature increases
    - If means are indistinguishable → interest wanes in that feature



$$SE\left(X_A - X_B\right) = \sqrt{\frac{var\left(X_A\right)}{n_{X,A}} + \frac{var\left(X_B\right)}{n_{X,B}}}$$

$$TEST\ : \frac{\left|mean\left(X_A\right) - mean\left(X_B\right)\right|}{SE\left(X_A - X_B\right)} > \text{threshold - value}$$

  - Simple but effective
  - Without taking into consideration relationship to other features

Supervised Feature Selection

- Example: threshold - value = 0.5

$$\bar{x} = mean\ (x) = \frac{\sum x}{n_x}$$

$$\text{var}\ (x) = \frac{\sum (x - \bar{x})^2}{n_x - 1}$$

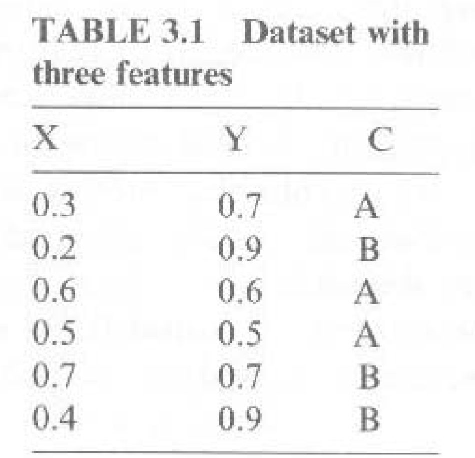


TABLE 3.1 Dataset with three features

X	Y	C
0.3	0.7	A
0.2	0.9	B
0.6	0.6	A
0.5	0.5	A
0.7	0.7	B
0.4	0.9	B

$$X_A = \{0.3, 0.6, 0.5\},\ n_{X,A} = 3$$

$$X_B = \{0.2, 0.7, 0.4\},\ n_{X,B} = 3$$

$$Y_A = \{0.7, 0.6, 0.5\},\ n_{Y,A} = 3$$

$$Y_B = \{0.9, 0.7, 0.9\},\ n_{Y,B} = 3$$

$$SE(X_A - X_B) = \sqrt{\frac{var(X_A)}{n_{X,A}} + \frac{var(X_B)}{n_{X,B}}} = \sqrt{\frac{0.0233}{3} + \frac{0.6333}{3}} = 0.4678$$

$$\frac{|mean(X_A) - mean(X_B)|}{SE(X_A - X_B)} = \frac{|0.4667 - 0.4333|}{0.4678} = 0.0735 < 0.5$$

$$SE(Y_A - Y_B) = \sqrt{\frac{var(Y_A)}{n_{Y,A}} + \frac{var(Y_B)}{n_{Y,B}}} = \sqrt{\frac{0.010}{3} + \frac{0.0133}{3}} = 0.0875$$

$$\frac{|mean(Y_A) - mean(Y_B)|}{SE(Y_A - Y_B)} = \frac{|0.600 - 0.8333|}{0.0875} = 2.6667 > 0.5$$ ✓

Supervised Feature Selection

- Method 2: feature examined collectively instead of independently, additional information can be obtained

$C : m \times m$ covariance matrix, each entry $C_{i,j}$ stands for the correlation between two features i, j ⇦ *m* features are selected

$$C_{i,j} = \frac{1}{n} \sum_{k=1}^{n} \left(v(k,i) - m(i)\right) \cdot \left(v(k,i) - m(i)\right)$$

number of samples

$v(k,i)$: the value of feature i of sample k

$m(i)$: mean of feature i

$$DM = (M_1 - M_2)(C_1 + C_2)^{-1}(M_1 - M_2)^T$$

⇦ distance measure for multivariate variables

- M1, M2, C1, C2, are respectively mean vectors and covariance matrices for class 1 and class 2
- A subset set of features are selected for this measure (maximizing *DM*)
    - All subsets should be evaluated ! (how to do ?)

Review: Entropy

- Three interpretations for quantity of information
  1. The amount of **uncertainty** before seeing an event
  2. The amount of **surprise** when seeing an event
  3. The amount of **information** after seeing an event

- The definition of information: $define \quad 0\log_2 0 = 0$

$$I(x_i) = \log_2 \frac{1}{P(x_i)} = -\log_2 P(x_i)$$

  - $P(x_i)$ the probability of an event x_i

- Entropy: the average amount of information

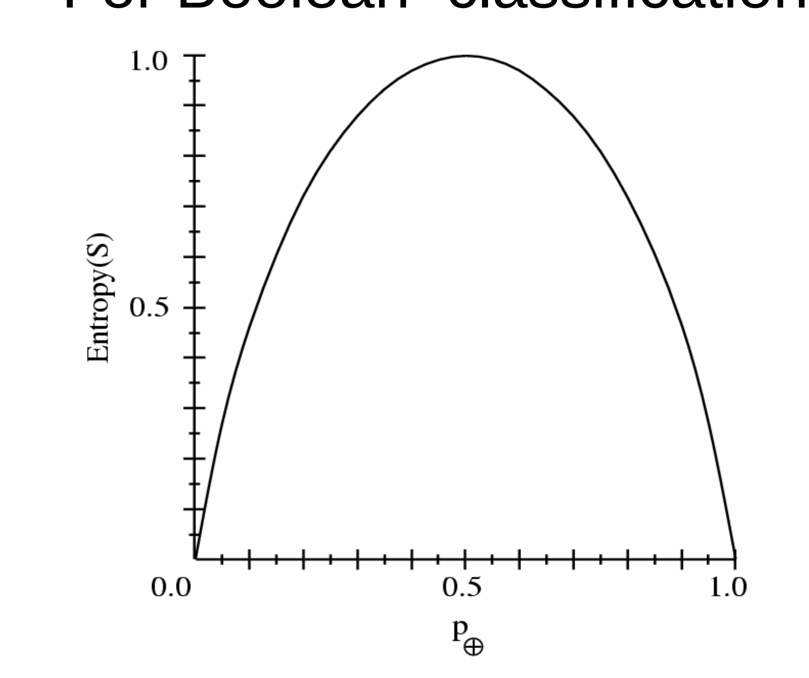
$$H(X) = E[I(X)]_X = E[-\log_2 P(x_i)]_X = \sum_{x_i} -P(x_i)\cdot \log_2 P(x_i)$$

$$\text{where } X = \{x_1, x_2, \ldots, x_i, \ldots\}$$

  - Have maximum value when the probability (mass) function is a uniform distribution

Review: Entropy

- For Boolean classification (0 or 1)



$$Entropy(X) = -p_1 \log_2 p_1 - p_2 \log_2 p_2$$

- Entropy can be expressed as the minimum number of bits of information needed to encode the classification of an arbitrary number of examples
  - If c classes are generated, the maximum of Entropy can be

$$Entropy(X) = \log_2 c$$

Unsupervised Feature Selection

- Method 1: Entropy measure for ranking features
  - Assumptions
    - All samples are given as vectors of feature values without any categorical information
    - The removal of an irrelevant (redundant) feature may not change the basic characteristics of the data set
      - basic characteristics → the similarity measure between any pair of samples
    - Use entropy to observe the change of global information before and after removal of a specific feature
      - Higher entropy for disordered configurations
      - Less entropy for ordered configurations
  - Rank features by iteratively (gradually) removing the least important feature in maintaining the configuration order

Unsupervised Feature Selection

- Method 1: Entropy measure for ranking features (cont.)
  - Distance measure between two samples x_i and x_j

$$D_{ij} = \left[\sum_{k=1}^{n} \left(\left(x_{ik} - x_{jk} \right) / \left(\max_k - \min_k \right) \right)^2 \right]^{1/2}$$

number of features

  - Change the distance measure to likelihood of proximity/similarity using exponential operator (function)

$$S_{ij} = \exp\left(-\alpha D_{ij}\right)$$

α is simply set to 0.5

or is set as $-\left(\ln 0.5\right)/D_{average}$

ranging between 0 ~1

  - $S_{ij} \approx 1$: x_i and x_j is very similar
  - $S_{ij} \approx 0$: x_i and x_j is very dissimilar

- For Categorical (nominal/nonmetric) features

$$S_{ij} = \left(\sum_{k=1}^{n} \left| x_{ik} = x_{jk} \right| \right) / n,$$

ranging between 0 ~1

$$\left| x_{ik} = x_{jk} \right| = 1 \text{ if } x_{ik} = x_{jk} \text{ and } 0 \text{ otherwise}$$

Unsupervised Feature Selection

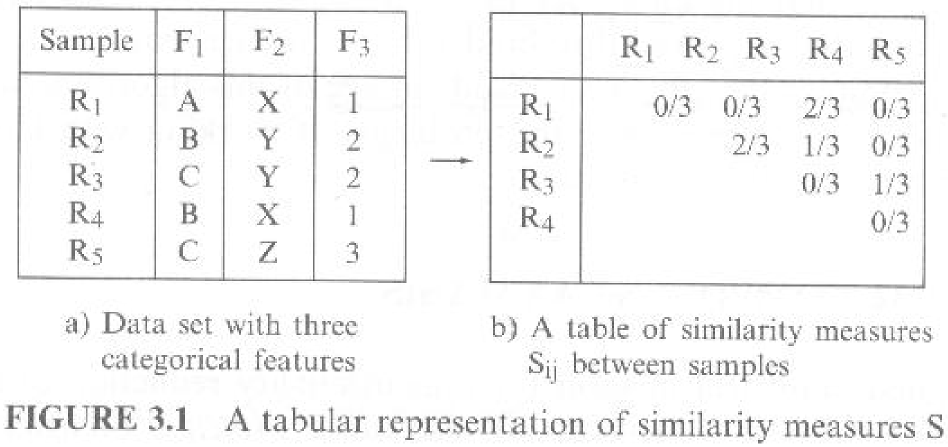
– Use entropy to monitor the changes in proximity between any sample pair in the data set

$$E = -\sum_{i=1}^{N-1}\sum_{j=i}^{N-1}\left(S_{ij}\log S_{ij} + \left(1 - S_{ij}\right)\log\left(1 - S_{ij}\right)\right)$$

Likelihood of being similar

Likelihood of being dissimilar

– Example: a simple data set with three categorical features



Sample	F1	F2	F3
R1	A	X	1
R2	B	Y	2
R3	C	Y	2
R4	B	X	1
R5	C	Z	3

a) Data set with three categorical features

→

	R1	R2	R3	R4	R5
R1		0/3	0/3	2/3	0/3
R2			2/3	1/3	0/3
R3				0/3	1/3
R4					0/3

b) A table of similarity measures S_{ij} between samples

FIGURE 3.1 A tabular representation of similarity measures S

Unsupervised Feature Selection

- Method 1: Entropy measure for ranking features (cont.)
  - Algorithm
    1. Start with the initial set of features F

    2. For each feature f in F, remove f from F and obtain a subset F_f . Find the difference between entropy for F and F_f

       $$\left| E_F - E_{F-f} \right|$$

    3. Find f_k such that its removal makes the entropy difference is minimum, check if the difference is less then the threshold

    4. If so, update the feature set as $F' = F - f_k$ and repeat steps 2~4 until only one feature is retained; otherwise, stop !

Disadvantage: the computational complexity is higher !

Value Reduction

- Also called Feature Discretization
- Goal: discretize the value of continuous features into a small number of intervals, where each interval is mapped to a discrete symbol
  - Simplify the tasks of data description and understanding
  - E.g., a person's age can be ranged from 0 ~ 150
    - Classified into categorical segments: "child, adolescent, adult, middle age, elderly"

Two main questions:

1. What are the cutoff points?
2. How to select representatives of intervals

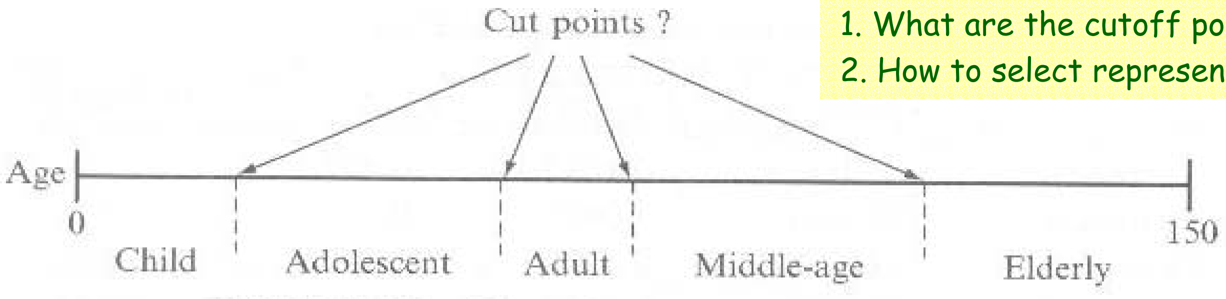


FIGURE 3.3 Discretization of the *age* feature

Unsupervised Value Reduction

- Method 1: Simple data reduction (value smoothing)
  - Also called number approximation by rounding
  - Reduce the number of distinct values for a feature
  - E.g., round the values to the given precision

$$f = \{0.93, 1.01, 1.001, 3.02, 2.99, 5.03, 5.01, 4.98\}$$
$$\Rightarrow f_{smoothed} = \{1.0, 1.0, 1.0, 3.0, 3.0, 5.0, 5.0, 5.0\}$$

  - Properties
    - Each feature is smoothed independently of other features
    - Performed only once without iterations
    - The number of data samples (cases) may be also reduced at the same time

Unsupervised Value Reduction

- Method 2: Placing the value in bins
  - Order the numeric values using great-than or less-than operators
  - Partition the ordered value list into groups with close values
    - Also, these bins have close number of elements
  - All values in a bin is merged into a single concept represented by a single value, for example:
    - Mean or median/mode of the bin’s value
    - The closest boundaries of each bin

$$f = \{3,2,1,5,4,3,1,7,5,3\}$$

$$\overset{ordering}{\Rightarrow} \{1,1,2,3,3,3,4,5,5,7\}$$

Based on what criterion ?

$$\overset{splitting}{\Rightarrow} \{\overset{\text{BIN 1}}{1,1,2} \quad \overset{\text{BIN 2}}{3,3,3} \quad \overset{\text{BIN 3}}{4,5,5,7}\}$$

$$\text{Smoothing based on mean values} \Rightarrow \{\overset{\text{BIN 1}}{1.33,1.33,1.33} \quad \overset{\text{BIN2}}{3,3,3} \quad \overset{\text{BIN 3}}{5.25,5.25,5.25,5.25}\}$$

$$\text{Smoothing based on bin modes} \Rightarrow \{\overset{\text{BIN 1}}{1,1,1} \quad \overset{\text{BIN2}}{3,3,3} \quad \overset{\text{BIN 3}}{5,5,5,5}\}$$

$$\text{Smoothing based on boundary values} \Rightarrow \{\overset{\text{BIN 1}}{1,1,2} \quad \overset{\text{BIN2}}{3,3,3} \quad \overset{\text{BIN 3}}{\boxed{4,4,7,7}}\}$$

Unsupervised Value Reduction

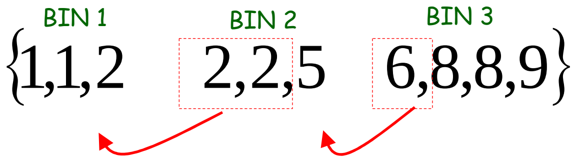
- Method 2: Placing the value in bins (cont.)
  - How to determine the optimal selection of *k* bins
    - Criterion: minimize the average distance of a value from its bin mean or median
      - Squared distance for a bin mean
      - Absolute distance for a bin median
    - Algorithm
      1. Sort all values for a given feature
      2. Assign approximately equal numbers of sorted adjacent value (v_i) to each bin, the number of bin is given in advance
      3. Move a border element v_i from one bin to the next (or previous) when that will reduce the global distance error (ER)

Unsupervised Value Reduction

- Method 2: Placing the value in bins (cont.)
  - Example

$$f = \{5,1,8,2,2,9,2,1,8,6\}$$

$$\overset{ordering}{\Rightarrow} \quad \{1,1,2,2,2,5,6,8,8,9\}$$



$$\overset{splitting\ /\ Initializing}{\Rightarrow} \quad \{\overset{\text{BIN 1}}{1,1,2} \quad \overset{\text{BIN 2}}{\boxed{2},2,\boxed{5}} \quad \overset{\text{BIN 3}}{\boxed{6},8,8,9}\}$$

Absolute distance to bin modes

$$ER = (0+0+1)+(0+0+3)+(2+0+0+1)=7$$

....

$$\Rightarrow \quad \{\overset{\text{BIN 1}}{1,1,2,2,2} \quad \overset{\text{BIN 2}}{5,6} \quad \overset{\text{BIN 3}}{8,8,9}\}$$

Absolute distance to bin modes

$$ER = (1+1+0+0+0)+(0+1)+(0+0+1)=4$$

$$\Rightarrow \text{corresponding modes} \{2,5,8\}$$

In real-world applications, the number of distinct values is controlled to be 50 ~ 100

Supervised Value Reduction

- Method 3: ChiMerge technique
  - An automated discretization algorithm that analyzes the quality of multiple intervals for a given feature using χ^2 statistics
  - Determine similarities between distributions of data in two adjacent intervals based on output classification of samples
    - If the χ^2 test indicates that the output class is independent of the feature's intervals, merge them; otherwise, stop merging!

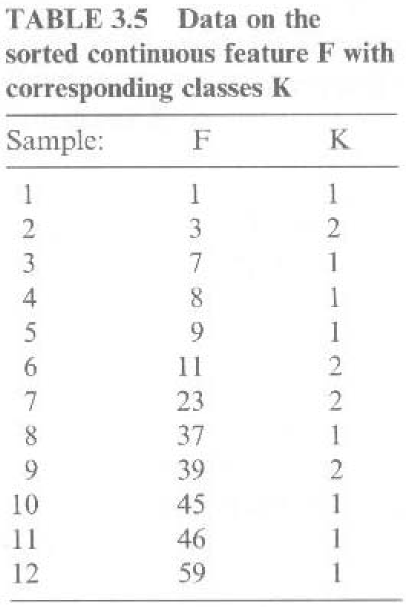


TABLE 3.5 Data on the sorted continuous feature F with corresponding classes K

Sample:	F	K
1	1	1
2	3	2
3	7	1
4	8	1
5	9	1
6	11	2
7	23	2
8	37	1
9	39	2
10	45	1
11	46	1
12	59	1

initial interval points :

0, 2, 5, 7.5, 8.5, 10, ...,60

Supervised Value Reduction

- Method 3: ChiMerge technique (cont.)
  - Algorithm
    1. Sort the data for the given feature in ascending order
    2. Define initial intervals so that every value of the feature is in a separate interval
    3. Repeat until no χ^2 of any two adjacent intervals is less then threshold value
  - Notes: a detailed explanation to χ^2 statistics/test and the introduced algorithm will be given in this course later on!

Case Reduction

- Also called "raw reduction"
- Premise: the largest and the most critical dimension in the initial data set is the number of cases or samples
  - The number of rows in the tabular representation of data
- Simple case reduction can be done in the preprocessing phase
  - Elimination of outliers
  - Elimination of samples with missing feature values

} There will be many samples remained !

- Or, case reduction achieved by using a sampled subset of samples (called an estimator) to provide some information about the entire data set (using sampling methods)
  - Reduced cost, greater speed, greater scope, even higher accuracy ?

Case Reduction

- Method 1: Systematic sampling
  - The simplest sampling technique
  - If 50% of a data set should be selected, simply take every other sample in a data set (e.g., 任兩個samples取其一)
  - There will be a problem, if the data set posses some regularities

Case Reduction

- Method 2: Random sampling
  - Every sample from the initial data set has the same chance of being selected in the subset
  - Two variants:

    1. Random sampling without replacement
       - Select *n* distinct samples form *N* initial samples without repetition
       - Avoid any bias in a selection

    2. Random sampling with replacement
       - All samples are given really equal chance of being selected, any of samples can be selected more than once

Case Reduction

- Method 2: Random sampling (cont.)
  - Notice that random sampling is an iterative process which may have two forms

    1. Incremental sampling 10%, 20%, 33%, 50%, 67%, 100%
        - Perform data mining on increasing larger random subsets to observe the trends in performances
        - Stop when no progress is made

    2. Average sampling
        - Solutions found from many random subsets of samples are averaged and voted
            - Regression problems → averaging
            - Classification problems → voting
        - Drawback: the repetitive process of data mining on smaller sets of samples

Case Reduction

- Method 3: Stratified sampling
  - The entire data set is split into non-overlapping subsets or strata
  - Sampling is performed for each different strata independently of each other
  - Combine all small subsets from different strata to form the final, total subset of samples
  - Better than random sampling if the strata is relatively homogeneous

- Method 4: Inverse sampling
  - Used when a feature in a data set occurs with rare frequency

    (not enough information can be given to estimate a feature value)
  - Sampling start with the smallest subset and it continues until some conditions about the required number of feature values are satisfied