PART 3
DATA (CONTINUED)

NGURAH AGUS SANJAYA ER, S.KOM, M.KOM, PH.D
E-mail: agus.sanjaya@cs.unud.ac.id

Data Preprocessing: Motivation

- Data in the real world tends to be dirty
  - **Incomplete: lacking attribute values, lacking certain attributes of interest, or containing only aggregate data**
  - **Noisy: containing errors or outliers**
  - **Inconsistent: containing different formats in codes or names**
- No quality data, no quality mining results!
  - **Quality decisions must be based on quality data**
  - **Data warehouse needs consistent integration of quality data**

Major task in Data Preprocessing

- Data cleaning
  - **Fill in missing values, smooth noisy data, identify or remove outliers, and resolve inconsistencies**
- Data integration
  - **Integration of multiple databases, data cubes, or files**
- Data transformation
  - **Normalization and aggregation**
- Data reduction
  - **Obtains reduced representation in volume but produces the same or similar analytical results**
- Data discretization
  - **Part of data reduction (by data discretization) but with particular importance, especially for numerical data to generate hierarchy**

Data Preprocessing

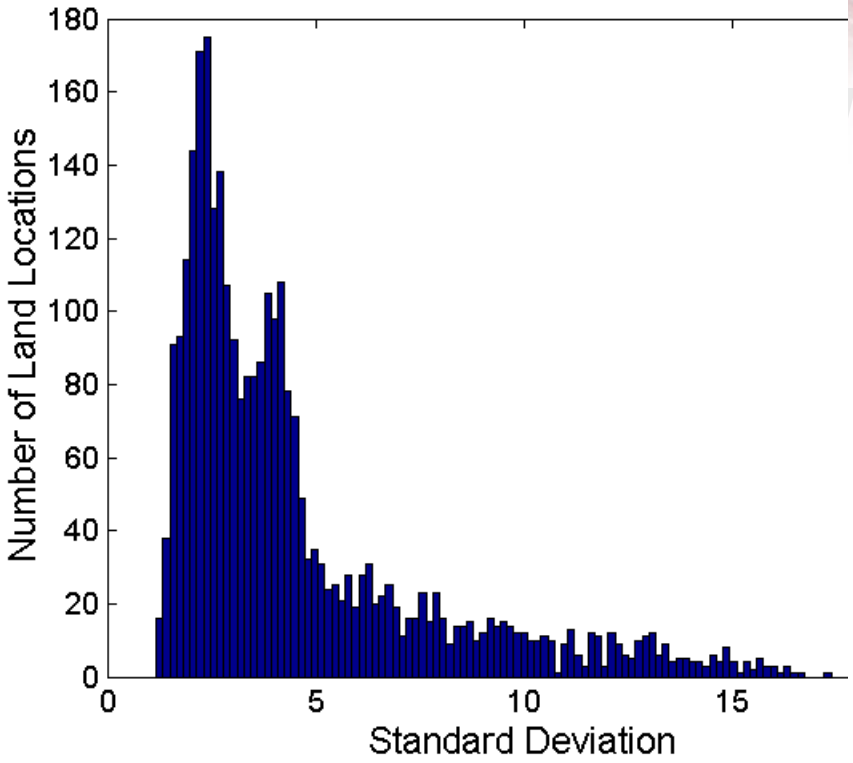
- Aggregation
- Sampling
- Feature subset selection
- Feature creation
- Dimensionality Reduction
- Discretization and Binarization
- Attribute Transformation

Aggregation / 1

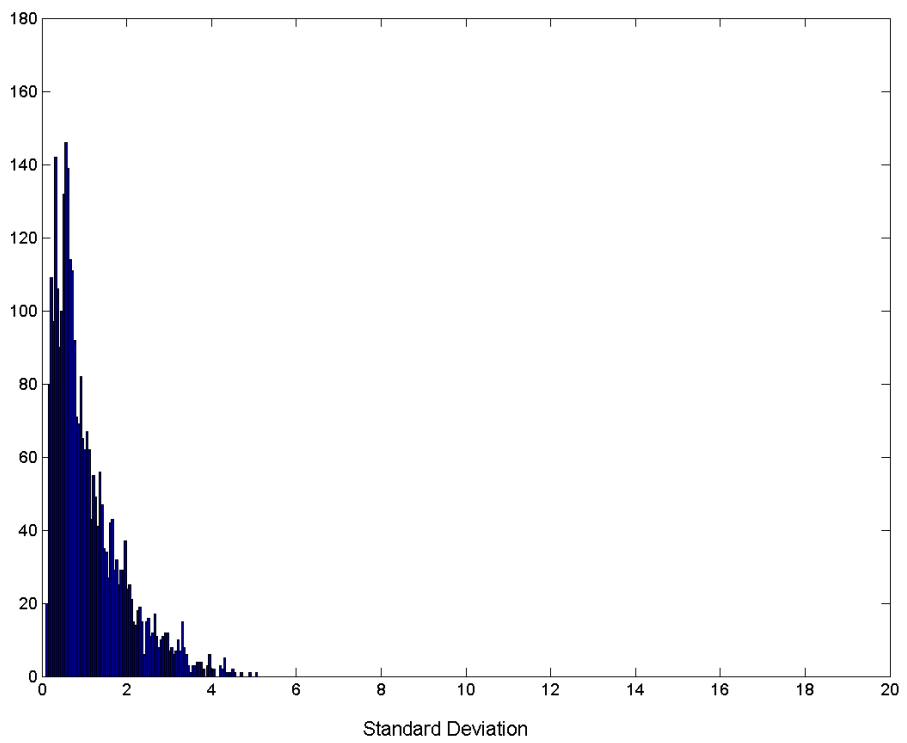
- Combining two or more attributes (or objects) into a single attribute (or object)

- Purpose
  - **Data reduction**
    - Reduce the number of attributes or objects
  - **Change of scale**
    - Cities aggregated into regions, states, countries, etc
  - **More “stable” data**
    - Aggregated data tends to have less variability

Variation of Precipitation in Australia



Standard Deviation of Average Monthly Precipitation



Standard Deviation of Average Yearly Precipitation

Sampling / 1

- **Sampling is the main technique employed for data selection.**
  - It is often used for both the preliminary investigation of the data and the final data analysis.

- **Statisticians sample because obtaining the entire set of data of interest is too expensive or time consuming.**

- **Sampling is used in data mining because processing the entire set of data of interest is too expensive or time consuming.**

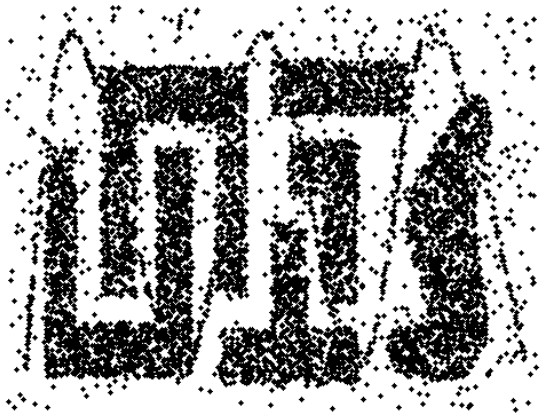
Sampling / 2

- The key principle for effective sampling is the following:
  - **Using a sample will work almost as well as using the entire data sets, if the sample is representative**
  - **A sample is representative if it has approximately the same property (of interest) as the original set of data**

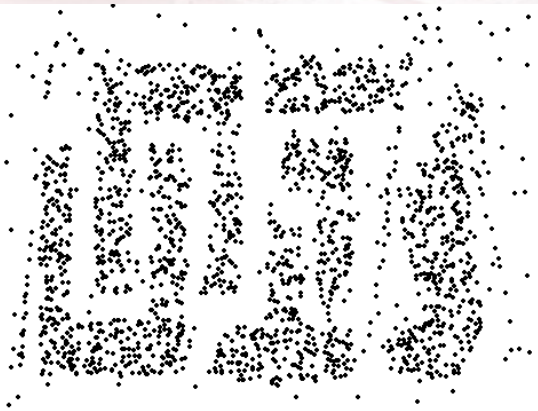
Types of Sampling

- Simple Random Sampling
  - **There is an equal probability of selecting any particular item**
- Sampling without replacement
  - **As each item is selected, it is removed from the population**
- Sampling with replacement
  - **Objects are not removed from the population as they are selected for the sample.**
    - In sampling with replacement, the same object can be picked up more than once
- Stratified sampling
  - **Split the data into several partitions; then draw random samples from each partition**

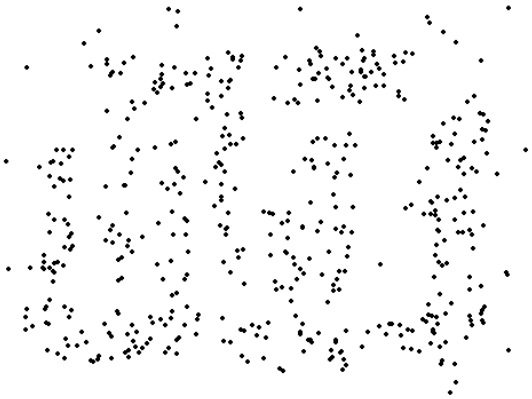
Sample Size / 1



8000 points



2000 Points



500 Points

Feature Subset Selection / 1

- Another way to reduce dimensionality of data

- Redundant features
  - **Duplicate much or all of the information contained in one or more other attributes**
  - **Example: purchase price of a product and the amount of sales tax paid**

- Irrelevant features
  - **Contain no information that is useful for the data mining task at hand**
  - **Example: students' ID is often irrelevant to the task of predicting students' GPA**

Feature Subset Selection / 1

- Techniques:
  - **Brute-force approch:**
    - Try all possible feature subsets as input to data mining algorithm
  - **Embedded approaches:**
    - Feature selection occurs naturally as part of the data mining algorithm
  - **Filter approaches:**
    - Features are selected before data mining algorithm is run
  - **Wrapper approaches:**
    - Use the data mining algorithm as a black box to find best subset of attributes

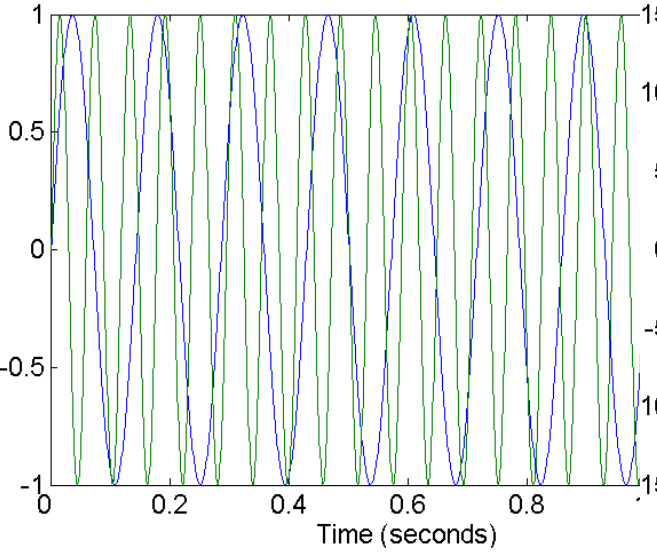
Feature Creation

- Create new attributes that can capture the important information in a data set much more efficiently than the original attributes

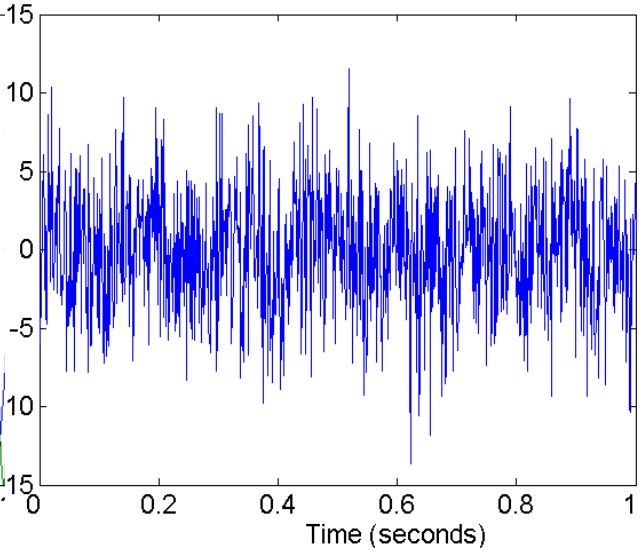
- Three general methodologies:
  - **Feature Extraction**
    - domain-specific
  - **Mapping Data to New Space**
  - **Feature Construction**
    - combining features

Mapping Data to a New Space

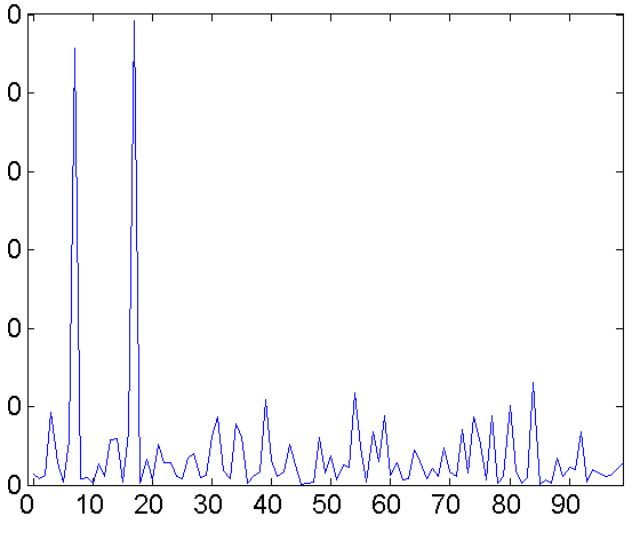
- Fourier transform
- Wavelet transform



Two Sine Waves



Two Sine Waves + Noise



Frequency

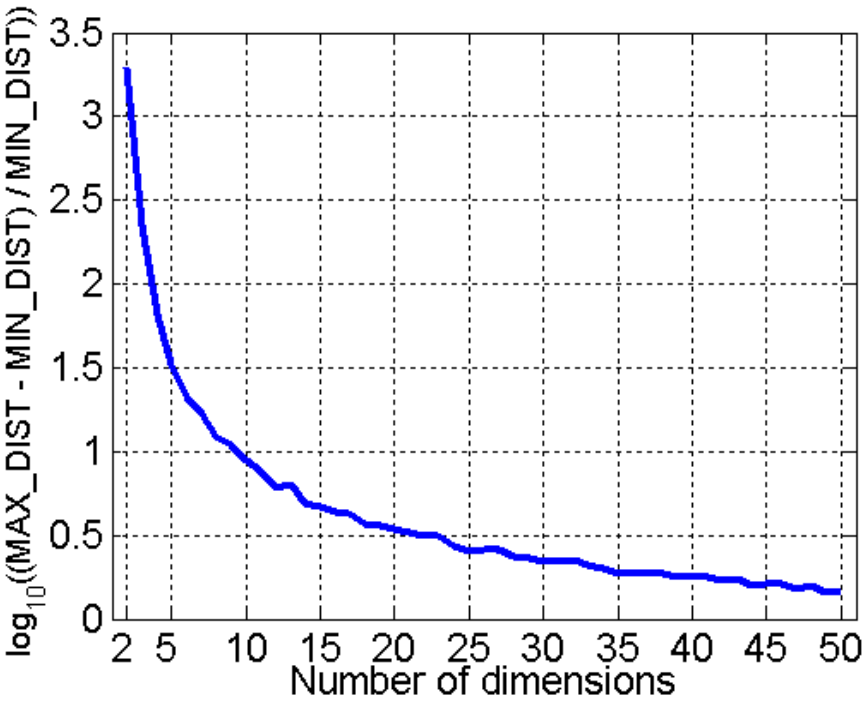
Dimensionality Reduction

- Purpose:
  - **Avoid curse of dimensionality**
  - **Reduce amount of time and memory required by data mining algorithms**
  - **Allow data to be more easily visualized**
  - **May help to eliminate irrelevant features or reduce noise**
- Techniques
  - **Principle Component Analysis**
  - **Singular Value Decomposition**
  - **Others: supervised and non-linear techniques**

Curse of Dimensionality

- When dimensionality increases, data becomes increasingly sparse in the space that it occupies

- Definitions of density and distance between points, which is critical for clustering and outlier detection, become less meaningful



- **Randomly generate 500 points**
- **Compute difference between max and min distance between any pair of points**

Binarization & Discretization

- Some data mining algorithms especially classification require that the data be in the form of categorical attribute.
- Algorithms that find association patterns require that the data be in the form of binary attribute.
- Continuous → categorical attribute (discretization)
- Continuous / discrete → one / more binary values (binarization)

Binarization

Categorical Value	Integer Value	X1	X2	X3
Awful	0	0	0	0
Poor	1	0	0	1
OK	2	0	1	0
Good	3	0	1	1
Great	4	1	0	0

Conversion of a categorical attribute to three binary atributes

Categorical Value	Integer Value	X1	X2	X3	X4	X5
Awful	0	1	0	0	0	0
Poor	1	0	1	0	0	0
OK	2	0	0	1	0	0
Good	3	0	0	0	1	0
Great	4	0	0	0	0	1

Conversion of a categorical attribute to five asymmetric binary attributes

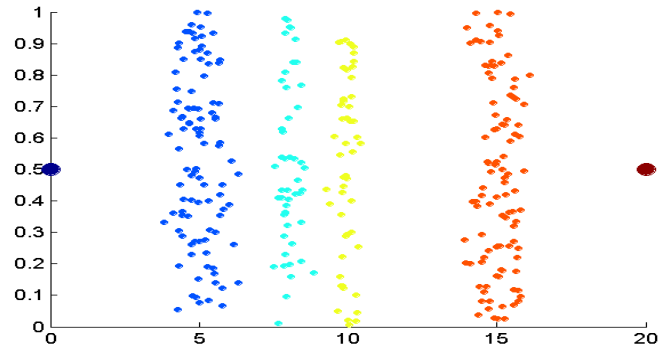
Discretization of Continuous Attributes

- Typically applied to attributes that are used in classification or association analysis.
- Two subtasks: how many categories and how to map the values of continuous attribute to these categories.
- In the first step: sort the values of the continuous attributes, they are divided into n intervals by specifying n-1 split points.
- In the second step: all values in one interval are mapped to the same categorical value.
- Therefore the problem of discretization is one of deciding how many split points to choose and where to put them.

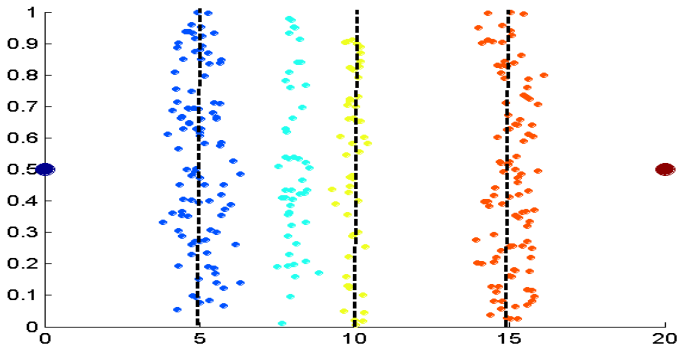
Types of Discretization

- Two types: supervised and unsupervised.
- A basic distinction between discretization methods for classifications is whether the class information is used (supervised) or not used (unsupervised).
- Unsupervised:
  - Equal width → divides the range of attribute into a user-specified number of intervals each having the same width (badly affected by outliers).
  - Equal frequency/depth → put the same number of objects into each interval.
- Examples of unsupervised discretization → K-Means (method of clustering) can also be used.

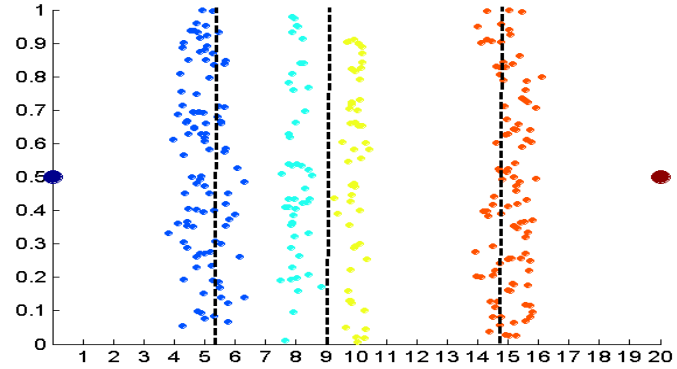
Discretization Without Using Class Labels (Upsupervised Discretization)



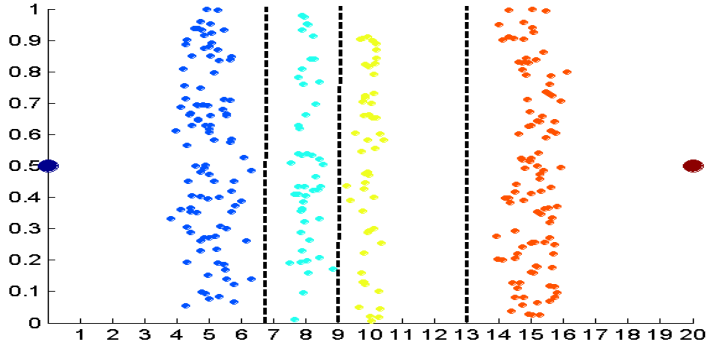
Data



Equal interval width



Equal frequency



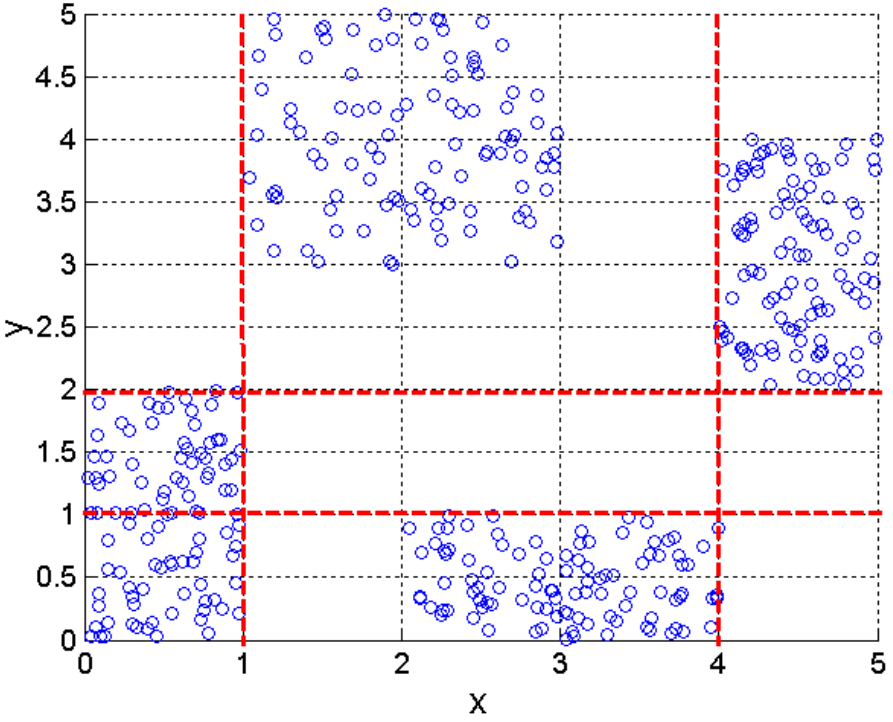
K-means

Supervised Discretization

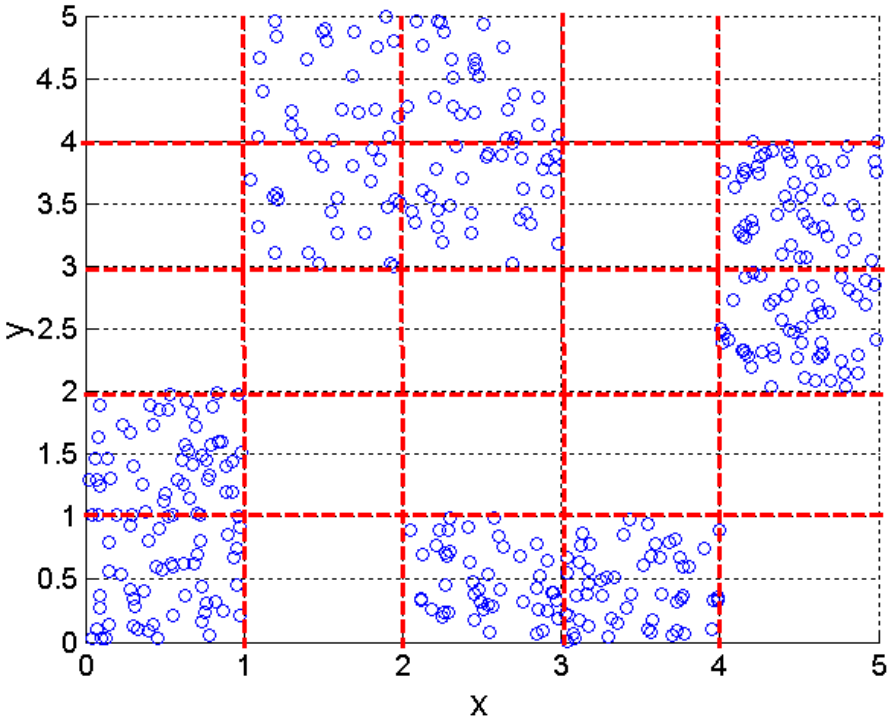
- Unsupervised discretization methods described above are better than no discretization, but interval constructed without no knowledge of the class labels often contains a mixture of class labels.
- How to solve this? Place the splits in a way that maximizes the purity of the intervals. Problem: how to decide the purity of an interval and the minimum size of an interval.
- To overcome such concerns, some statistically based approaches start with each attribute value as a separate interval and create larger intervals by merging adjacent intervals that are similar according to a statistical test → entropy based.

Discretization Using Class Labels (Supervised Discretization)

- Entropy based approach



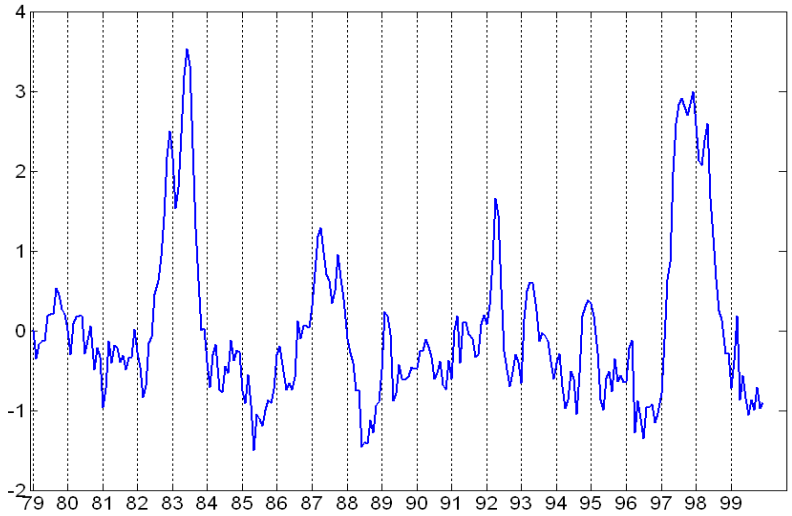
3 categories for both x and y



5 categories for both x and y

Attribute Transformation

- A function that maps the entire set of values of a given attribute to a new set of replacement values such that each old value can be identified with one of the new values
  - **Simple functions: x^k, log(x), e^x, |x|**
  - **Standardization and Normalization**



Attribute Transformation: Normalization /1

- Min-max normalization

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

Suppose that the minimum and maximum values for the attribute income are $12,000 and $98,000, respectively.

➔ Map income to the range [0.0, 1.0]. By min-max normalization, a value of $73,600 for income is transformed to:

(73600-12000)/(98000-12000)*(1.0-0.0)+0=0.716.

Attribute Transformation: Normalization/2

- Z-score normalization

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

- Decimal scaling normalization

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

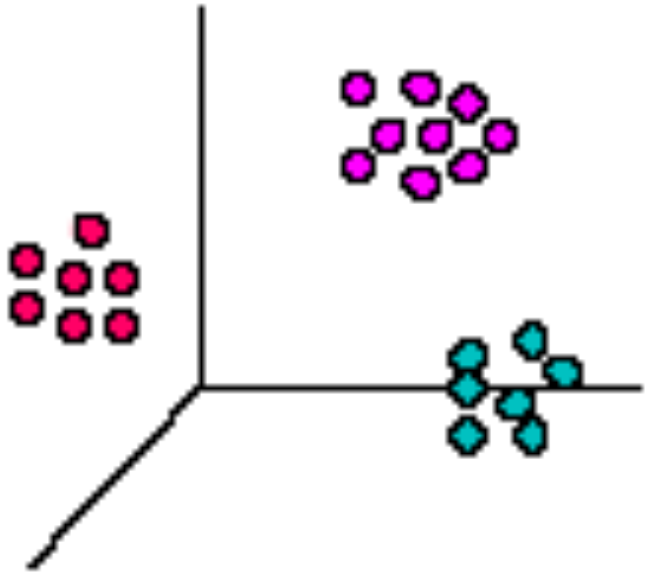
Where j is the smallest integer such that $Max(|v'|) < 1$

Similarity and Dissimilarity / 1

- Similarity
  - **Numerical measure of how alike two data objects are.**
  - **Is higher when objects are more alike.**
  - **Often falls in the range [0,1]**
- Dissimilarity
  - **Numerical measure of how different are two data objects**
  - **Lower when objects are more alike**
  - **Minimum dissimilarity is often 0**
  - **Upper limit varies**
- Proximity refers to a similarity or dissimilarity

Similarity and Dissimilarity / 2

- Used by a number of data mining techniques
  - **Clustering**
  - **Anomaly detection**
  - **Nearest-neighbor classification**



Similarity/Dissimilarity for Simple Attributes

p and *q* are the attribute values for two data objects.

Attribute Type	Dissimilarity	Similarity
Nominal	$d = \begin{cases} 0 & \text{if } p = q \\ 1 & \text{if } p \neq q \end{cases}$	$s = \begin{cases} 1 & \text{if } p = q \\ 0 & \text{if } p \neq q \end{cases}$
Ordinal	$d = \frac{\lvert p-q \rvert}{n-1}$ (values mapped to integers 0 to $n-1$, where n is the number of values)	$s = 1 - \frac{\lvert p-q \rvert}{n-1}$
Interval or Ratio	$d = \lvert p - q \rvert$	$s = -d$, $s = \frac{1}{1+d}$ or $s = 1 - \frac{d - min\_d}{max\_d - min\_d}$

Similarity & dissimilarity for simple attributes

Similarity and Dissimilarity for Data Objects

- Dissimilarity (mostly for continuous data)
  - **Euclidean distance**
  - **Minkowski distance**
  - **Mahalanobis distance**
- Similarity
  - **Binary data (SMC, Jaccard, cosine, Hamming)**
  - **Continuous data (Tanimoto, correlation)**

Euclidean Distance / 1

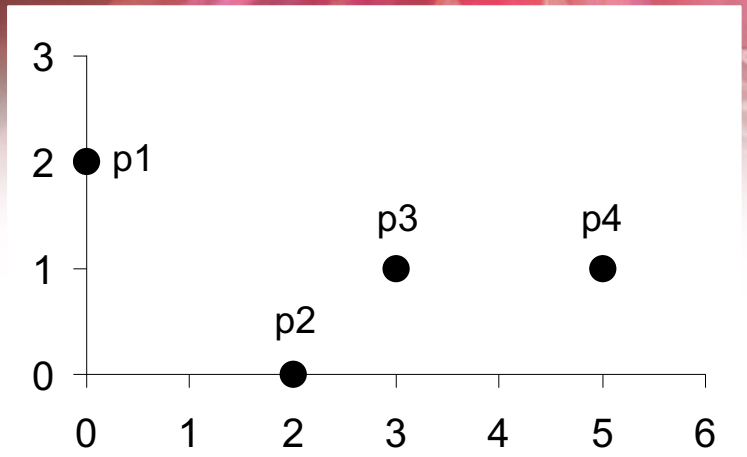
- Euclidean Distance

$$dist = \sqrt{\sum_{k=1}^{n} (p_k - q_k)^2}$$

Where *n* is the number of dimensions (attributes) and p_k and q_k are the k^{th} attributes (components) of data objects *p* and *q*, respectively

- Standardization is necessary, if scales differ.

Euclidean Distance



point	x	y
p1	0	2
p2	2	0
p3	3	1
p4	5	1

	p1	p2	p3	p4
p1	0	2.828	3.162	5.099
p2	2.828	0	1.414	3.162
p3	3.162	1.414	0	2
p4	5.099	3.162	2	0

Distance Matrix

Minkowski Distance / 1

- Minkowski Distance is a generalization of Euclidean Distance

$$dist = \left(\sum_{k=1}^{n} | p_k - q_k |^r \right)^{\frac{1}{r}}$$

Where *r* is a parameter, *n* is the number of dimensions (attributes) and p_k and q_k are, respectively, the kth attributes (components) or data objects *p* and *q*.

Minkowski Distance / 2

- $r = 1$. City block (Manhattan, taxicab, L_1 norm) distance.
  - **A common example of this is the Hamming distance, which is just the number of bits that are different between two binary vectors**
- $r = 2$. Euclidean distance
- $r \rightarrow \infty$. "supremum" (L_{max} norm, L_∞ norm) distance.
  - **This is the maximum difference between any component of the vectors**
- Do not confuse *r* with *n*, i.e., all these distances are defined for all numbers of dimensions.

Minkowski Distance: Examples

point	x	y
p1	0	2
p2	2	0
p3	3	1
p4	5	1

L1	p1	p2	p3	p4
p1	0	4	4	6
p2	4	0	2	4
p3	4	2	0	2
p4	6	4	2	0

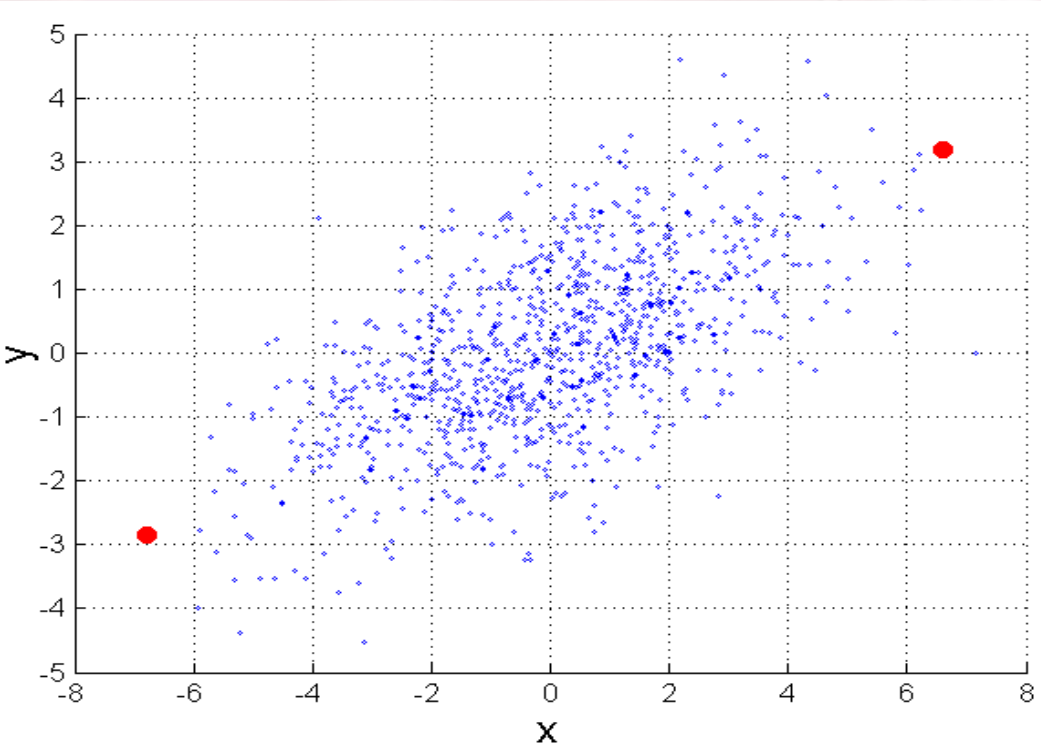
L2	p1	p2	p3	p4
p1	0	2.828	3.162	5.099
p2	2.828	0	1.414	3.162
p3	3.162	1.414	0	2
p4	5.099	3.162	2	0

L_∞	p1	p2	p3	p4
p1	0	2	3	5
p2	2	0	1	3
p3	3	1	0	2
p4	5	3	2	0

Distance Matrix

Mahalanobis Distance / 1

$$Mahalanobis(p, q) = (p-q)\Sigma^{-1}(p-q)^T$$

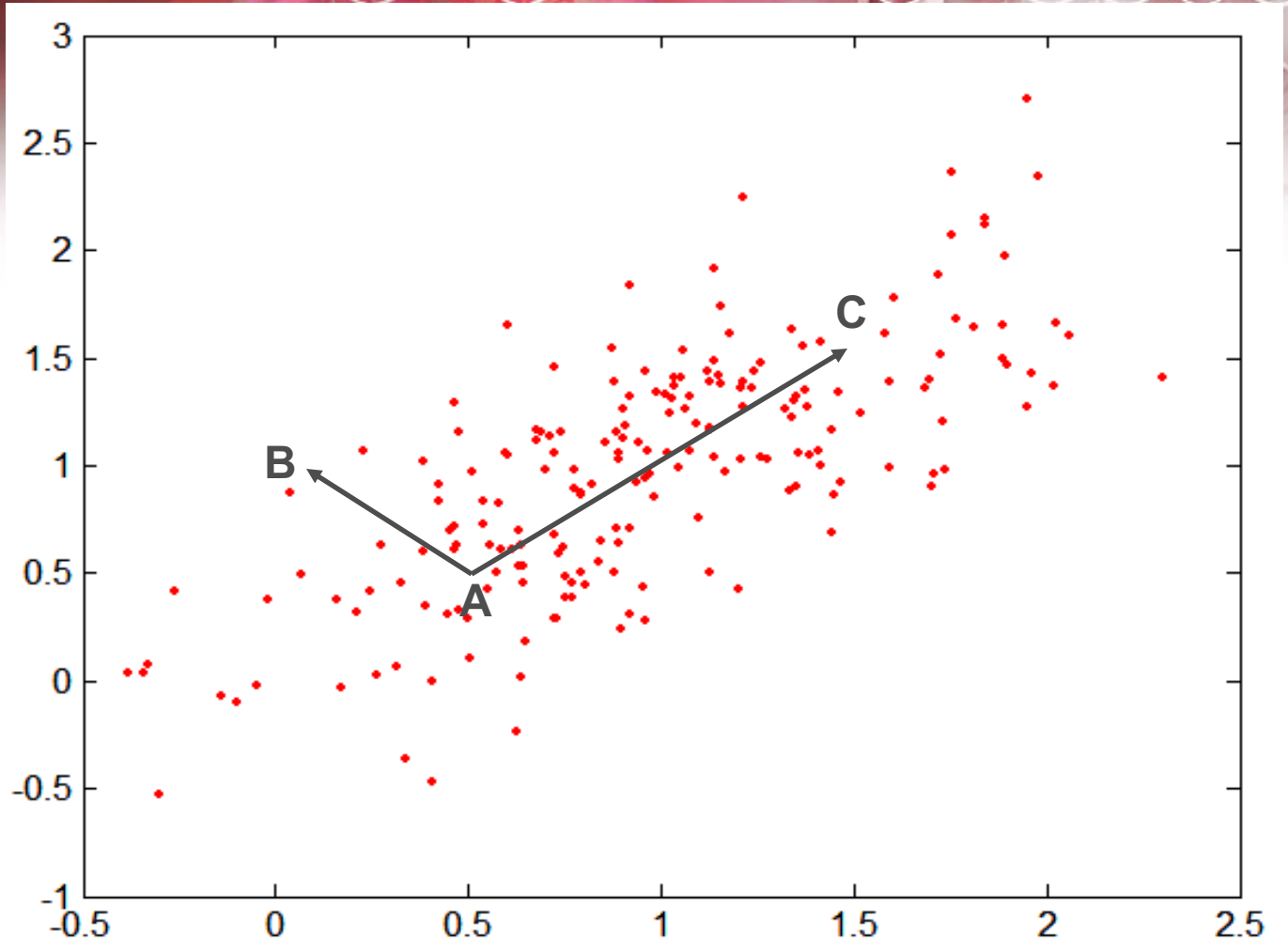


Σ^{-1} is the covariance matrix of the input data *X*

$$\Sigma_{j,k} = \frac{1}{n-1}\sum_{i=1}^{n}(X_{ij} - \overline{X}_j)(X_{ik} - \overline{X}_k)$$

For red points, the Euclidean distance is 14.7, Mahalanobis distance is 6.

Mahalanobis Distance / 2



Covariance Matrix:

$$\Sigma = \begin{bmatrix} 0.3 & 0.2 \\ 0.2 & 0.3 \end{bmatrix}$$

A: (0.5, 0.5)

B: (0, 1)

C: (1.5, 1.5)

Mahal(A,B) = 5

Mahal(A,C) = 4

Common Properties of a Distance

- Distances, such as the Euclidean distance, have some well known properties.

  1. **$d(p, q) \geq 0$ for all p and q and $d(p, q) = 0$ only if $p = q$. (positivity)**
  2. **$d(p, q) = d(q, p)$ for all p and q. (symmetry)**
  3. **$d(p, r) \leq d(p, q) + d(q, r)$ for all points p, q, and r. (triangle inequality)**

  where $d(p, q)$ is the distance (dissimilarity) between points (data objects), p and q.

- A distance that satisfies these properties is called as a metric

Common Properties of a Similarity

- Similarities, also have some well known properties.

1. ***s(p, q) = 1* (or maximum similarity) only if *p* = *q*.**

2. ***s(p, q) = s(q, p)* for all *p* and *q*. (Symmetry)**

where *s(p, q)* is the similarity between points (data objects), *p* and *q*.

Similarity Between Binary Vectors

- Common situation is that objects, *p* and *q*, have only binary attributes
- Compute similarities using the following quantities

M_{01} = the number of attributes where p was 0 and q was 1
M_{10} = the number of attributes where p was 1 and q was 0
M_{00} = the number of attributes where p was 0 and q was 0
M_{11} = the number of attributes where p was 1 and q was 1

- Simple Matching and Jaccard Coefficients

SMC = number of matches / number of attributes
= $(M_{11} + M_{00}) / (M_{01} + M_{10} + M_{11} + M_{00})$

JC = number of 11 matches / number of not-both-zero attributes values
= $(M_{11}) / (M_{01} + M_{10} + M_{11})$

SMC versus Jaccard: Example

p = 1 0 0 0 0 0 0 0 0 0

q = 0 0 0 0 0 0 1 0 0 1

M_{01} = 2 (the number of attributes where p was 0 and q was 1)

M_{10} = 1 (the number of attributes where p was 1 and q was 0)

M_{00} = 7 (the number of attributes where p was 0 and q was 0)

M_{11} = 0 (the number of attributes where p was 1 and q was 1)

$$SMC = (M_{11} + M_{00})/(M_{01} + M_{10} + M_{11} + M_{00}) = (0+7) / (2+1+0+7) = 0.7$$

$$JC = (M_{11}) / (M_{01} + M_{10} + M_{11}) = 0 / (2 + 1 + 0) = 0$$

Cosine Similarity

- If d_1 and d_2 are two document vectors, then

$$\cos(d_1, d_2) = (d_1 \bullet d_2) / \|d_1\| \|d_2\|,$$

where $\bullet$ indicates vector dot product and $\|d\|$ is the length of vector d.

- Example:

d_1 = **3 2 0 5 0 0 0 2 0 0**
d_2 = **1 0 0 0 0 0 0 1 0 2**

$d_1 \bullet d_2$= 3*1 + 2*0 + 0*0 + 5*0 + 0*0 + 0*0 + 0*0 + 2*1 + 0*0 + 0*2 = 5

$\|d_1\|$ = $(3*3+2*2+0*0+5*5+0*0+0*0+0*0+2*2+0*0+0*0)^{0.5}$ = $(42)^{0.5}$ = 6.481

$\|d_2\|$ = $(1*1+0*0+0*0+0*0+0*0+0*0+0*0+1*1+0*0+2*2)^{0.5}$ = $(6)^{0.5}$ = 2.245

$$\cos(d_1, d_2) = 5 / (6.481 + 2.245) = 0.3150$$

Extended Jaccard Coefficient (Tanimoto)

- Variation of Jaccard for continuous or count attributes
  - **Reduces to Jaccard for binary attributes**

$$T(p,q) = \frac{p \bullet q}{\|p\|^2 + \|q\|^2 - p \bullet q}$$

Correlation

- Correlation measures the linear relationship between objects
- To compute correlation, data objects *p* and *q* must be standardized, and then take their dot product

$$p'_k = (p_k - mean(p)) / std(p)$$

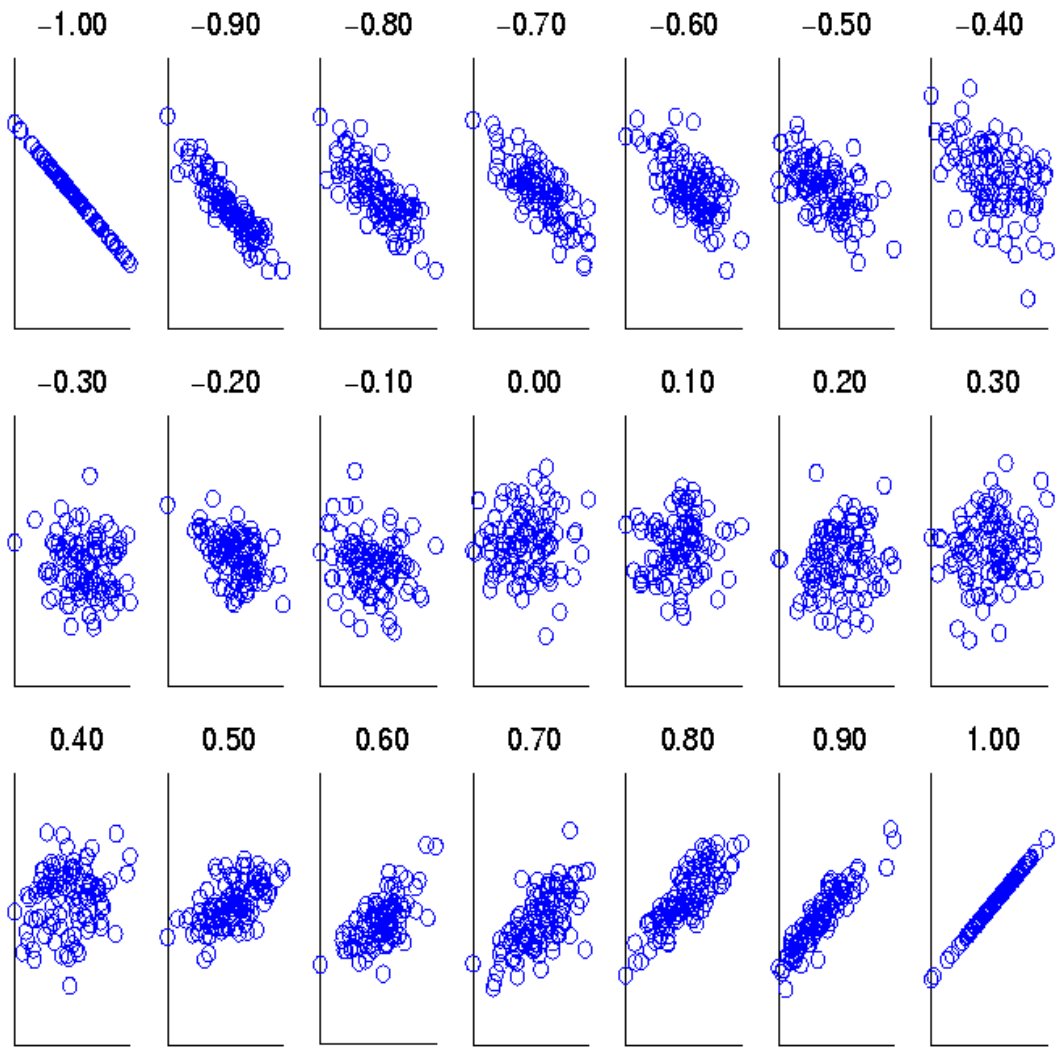
$$q'_k = (q_k - mean(q)) / std(q)$$

$$correlation(p,q) = p' \bullet q'$$

$$mean(x) = \sum_{k=1}^{n} x_k \qquad std(x) = \sqrt{\frac{\sum_{k=1}^{n} (x_k - mean(x))^2}{n-1}}$$

- If corr. > 0, then A and B are positively correlated. The higher the value, the more each attribute implies the other
- If corr. < 0, then A and B are negatively correlated
- If corr. = 0, then A and B are independent (no correlation)

Visually Evaluating Correlation



Scatter plots showing the similarity from –1 to 1.

Drawback of Correlation

- X = (-3, -2, -1, 0, 1, 2, 3)
- Y = (9, 4, 1, 0, 1, 4, 9) ➔ $Y = X^2$
- Mean(X) = 0, Mean(Y) = 4

- Correlation

  = (-3)(5)+(-2)(0)+(-1)(-3)+(0)(-4)+(1)(-3)+(2)(0)+(3)(5)

  = 0

General Approach for Combining Similarities

- Sometimes attributes are of many different types, but an overall similarity is needed.

1. For the k^{th} attribute, compute a similarity, s_k, in the range $[0, 1]$.
2. Define an indicator variable, δ_k, for the k_{th} attribute as follows:

$$\delta_k = \begin{cases} 0 & \text{if the } k^{th} \text{ attribute is a binary asymmetric attribute and both objects have a value of 0, or if one of the objects has a missing values for the } k^{th} \text{ attribute} \\ 1 & \text{otherwise} \end{cases}$$

3. Compute the overall similarity between the two objects using the following formula:

$$similarity(p, q) = \frac{\sum_{k=1}^{n} \delta_k s_k}{\sum_{k=1}^{n} \delta_k}$$

Using Weights to Combine Similarities

- May not want to treat all attributes the same.
  - **Use weights w_k which are between 0 and 1 and sum to**

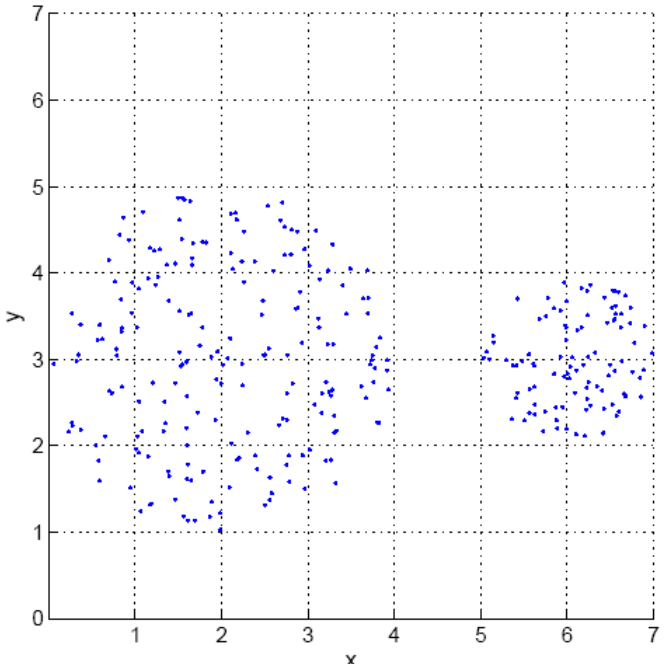
$$similarity(p,q) = \frac{\sum_{k=1}^{n} w_k \delta_k s_k}{\sum_{k=1}^{n} \delta_k}$$

$$distance(p,q) = \left(\sum_{k=1}^{n} w_k |p_k - q_k|^r \right)^{1/r}$$

Density

- Density-based clustering require a notion of density
- Examples:
  - **Euclidean density**
    - Euclidean density = number of points per unit volume
  - **Probability density**
  - **Graph-based density**

- Simplest approach is to divide region into a number of rectangular cells of equal volume and define density as # of points the cell contains



Cell-based density

0	0	0	0	0	0	0
0	0	0	0	0	0	0
4	17	18	6	0	0	0
14	14	13	13	0	18	27
11	18	10	21	0	24	31
3	20	14	4	0	0	0
0	0	0	0	0	0	0

Point counts for each grid cell

Euclidean Density – Center-based

- Euclidean density is the number of points within a specified radius of the point

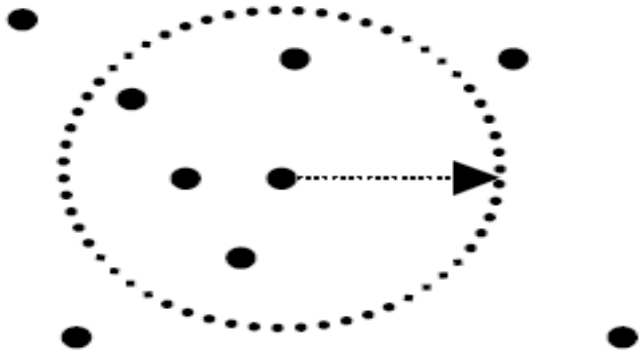


Figure 7.14. Illustration of center-based density.

Illustration of center-based density

Discretization and Concept Hierarchy

- Discretization
  - Divide the range of a continuous attribute into intervals
  - Interval labels can then be used to replace actual data values
  - Reduce data size by discretization
- Concept hierarchies
  - Reduce the data by collecting and replacing low level concepts (such as numeric values for the attribute age) by higher level concepts (such as young, middle-aged, or senior)

Discretization and Concept Hierarchy Generation for Numeric Data

- Binning
- Histogram analysis
- Clustering analysis

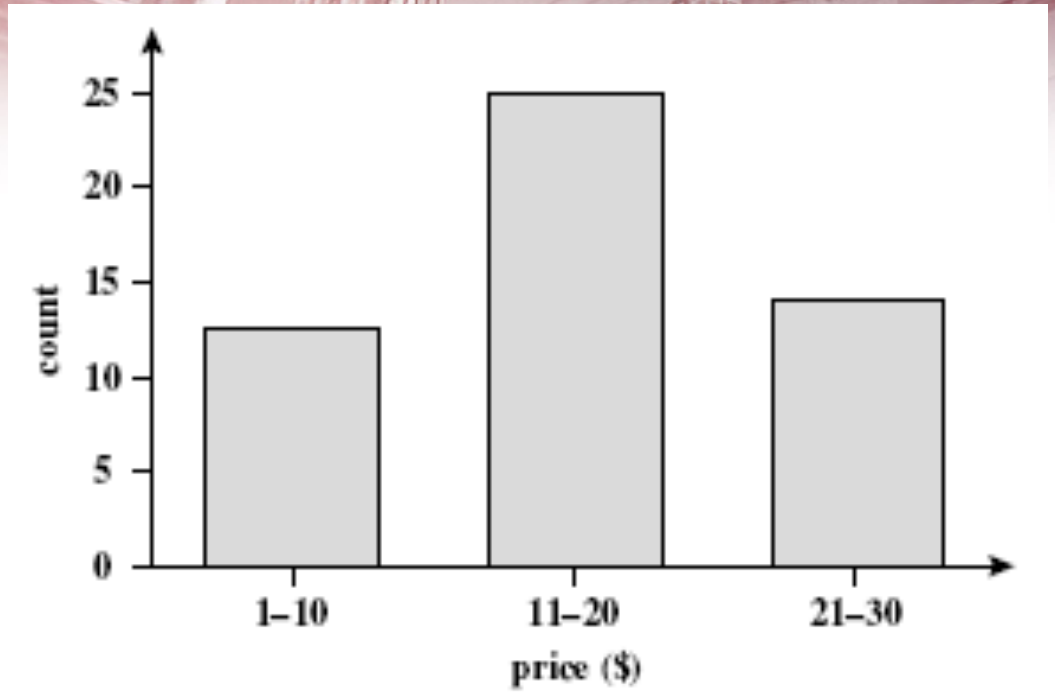
Simple Discretization Methods: Binning

- Equal-width (distance) partitioning:
  - **It divides the range into *N* intervals of equal size: uniform grid**
  - **if *A* and *B* are the lowest and highest values of the attribute, the width of intervals will be: *W* = (*B*-*A*)/*N*.**
  - **The most straightforward**

- Equal-depth (frequency) partitioning:
  - **It divides the range into *N* intervals, each containing approximately same number of samples**

Histograms

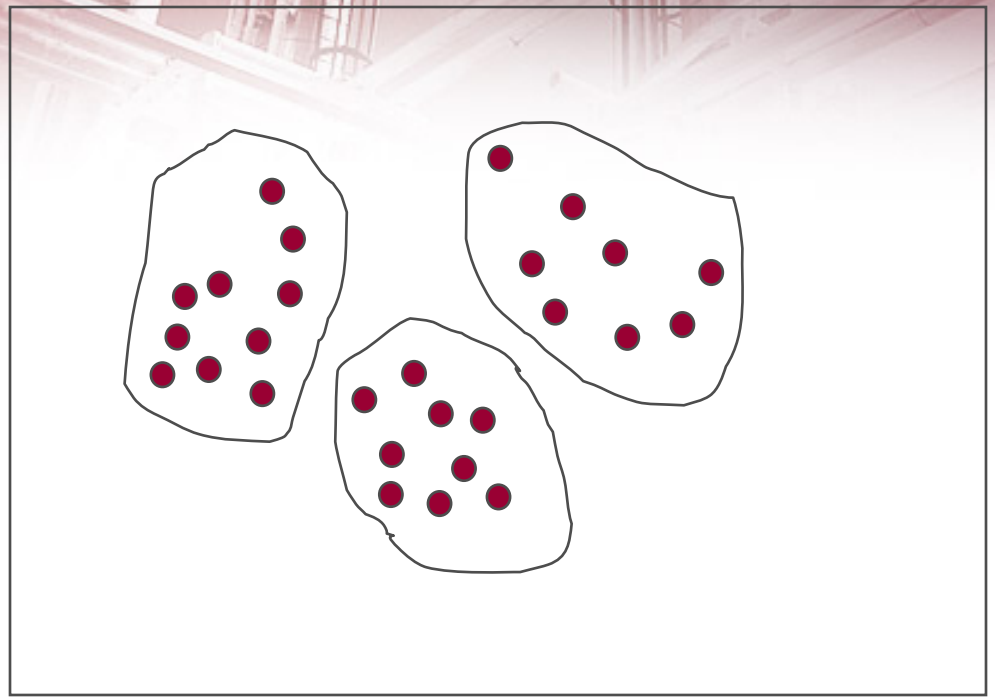
- A popular data reduction technique
- Divide continuous data into buckets and store average/sum for each bucket
- Buckets can be partitioned based on, for example, *equal-width* or *equal-frequency* method



Example of an equal-width histogram

Clustering

- Partition data set into clusters, and one can store cluster representation only
- Can be very effective if data is clustered but not if data is “smeared”
- There are many choices of clustering definitions and clustering algorithms



Concept Hierarchy Generation for Categorical Data / 1

- Specification of a partial ordering of attributes explicitly at the schema level by users or experts
- Specification of a portion of a hierarchy by explicit data grouping
- Specification of a set of attributes, but not of their partial ordering
- Specification of only a partial set of attributes

Concept Hierarchy Generation for Categorical Data / 2

Concept hierarchy can be automatically generated based on the number of distinct values per attribute in the given attribute set. The attribute with the most distinct values is placed at the lowest level of the hierarchy.