Data Mining:

Concepts and Techniques

(3rd ed.)

MODULE 2

Chapter 2: Data Preprocessing

- Data Preprocessing
  - Types of data?
- Data Preprocessing: An Overview
  - Data Quality
  - Major Tasks in Data Preprocessing
- Data Cleaning
- Data Integration
- Data Reduction
- Data Transformation and Data Discretization
- Similarity and Dissimilarity measures.

What is Data?

- Collection of data objects and their attributes
- An attribute is a property or characteristic of an object
  - Examples: eye color of a person, temperature, etc.
  - Attribute is also known as variable, field, characteristic, or feature
- A collection of attributes describe an object
  - Object is also known as record, point, case, sample, entity, or instance

Attributes

Objects

Tid	**Refund**	**Marital Status**	**Taxable Income**	**Cheat**
1	Yes	Single	125K	**No**
2	No	Married	100K	**No**
3	No	Single	70K	**No**
4	Yes	Married	120K	**No**
5	No	Divorced	95K	**Yes**
6	No	Married	60K	**No**
7	Yes	Divorced	220K	**No**
8	No	Single	85K	**Yes**
9	No	Married	75K	**No**
10	No	Single	90K	**Yes**

Types of attributes ....

1) **Nominal:**
   - items differentiated by a simple naming system
   - They may have numbers assigned to them. But, they are not actual numbers. They simply capture and reference.
   - They are 'categorical' i.e, they belong to a definable category.
   - Ex:- ID numbers, eye color, zip codes

Types of attributes…

2) **Ordinal:**

- They have some kind of order by their position on the scale.
- order of items can be defined by assigning numbers – relative position.
- letters can also be assigned.
- they are ‘categorical’
- cannot do arithmetic – only ordering property.
- Ex:

  rankings (taste of potato chips on a scale from 1 – 10)

  Grades in {A, B, C, D, E}

  Height in {small, medium, large}

Types of attributes ….

3) Interval:

- Is measured along a scale in which each position is equidistant from one another.

- Distance between two pairs will be equivalent in some way.
  Cannot be multiplied, or divided.

- Ex:
  Calendar dates
  Temperature in celsius/fah

Types of attributes….

4) Ratio

- numbers can be compared as multiples of one another.
- One person can be twice as tall as another person
- Number zero has no meaning

Ex:

- Difference between a person of age 35 and a person of age 38 is same as difference between people who are 12 and 15. (35 to 38 = 3 , 12 to 15 = 3) 3:3 .
- Ratio data can be multiplied and divided.

Types of attributes…

- Interval and ratio data measure quantities and hence are quantitative.
- Ex: length, time, count

Types of attributes…

- Nominal (symbolic, categorical)
  - Values from an unordered set
  - Ex: {red, yellow, blue, ….}
- Ordinal :
  - Values from an ordered set
  - Ex: {good, better, best}
- Continuous : real numbers
  - Ex: {-9.8, 3.9,…..}

Discrete and Continuous Attributes

Depending on the number of values :-

- Discrete Attribute
  - Has only a finite or countably infinite set of values
  - Examples: zip codes, counts, or the set of words in a collection of documents
  - Often represented as integer variables.
  - Note: binary attributes are a special case of discrete attributes
- Continuous Attribute
  - Has real numbers as attribute values
  - Examples: temperature, height, or weight.
  - Practically, real values can only be measured and represented using a finite number of digits.
  - Continuous attributes are typically represented as floating-point variables.

Types of Attributes Summary

- There are different types of attributes

Categorical / qualitative

- Nominal
  - Examples: ID numbers, eye color, zip codes
- Ordinal
  - Examples: rankings (e.g., taste of potato chips on a scale from 1-10), grades, height in {tall, medium, short}

Numeric / quantitative

- Interval
  - Examples: calendar dates, temperatures in Celsius or Fahrenheit.
- Ratio
  - Examples: temperature in Kelvin, length, time, counts

Properties of Attribute Values

- The type of an attribute depends on which of the following properties it possesses:
  - Distinctness: $=$ $\neq$
  - Order: $<$ $>$
  - Addition: $+$ $-$
  - Multiplication: $*$ $/$

  - Nominal attribute: distinctness
  - Ordinal attribute: distinctness & order
  - Interval attribute: distinctness, order & addition
  - Ratio attribute: all 4 properties

Types of Attributes Summary

Classify the following attributes as :-

- binary, discrete or continuous
- Qualitative(nominal or ordinal) or quantitative (interval or ratio)

1) Age in years

   Ans: discrete, quantitative, ratio

2) Brightness as measured by a light meter

   Ans: continuous, quantitative, ratio

3) Bronze, silver and gold medals as awarded at Olympics

   Ans: Discrete, qualitative, ordinal.

Types of data sets

- **Record**
  - **Data Matrix**
  - **Document Data**
  - **Transaction Data**
- **Graph**
  - **World Wide Web**
  - **Molecular Structures**
- **Ordered**
  - **Spatial Data**
  - **Temporal Data**
  - **Sequential Data**
  - **Genetic Sequence Data**

Record Data

- Data that consists of a collection of records, each of which consists of a fixed set of attributes

Tid	**Refund**	**Marital Status**	**Taxable Income**	**Cheat**
1	Yes	Single	125K	**No**
2	No	Married	100K	**No**
3	No	Single	70K	**No**
4	Yes	Married	120K	**No**
5	No	Divorced	95K	**Yes**
6	No	Married	60K	**No**
7	Yes	Divorced	220K	**No**
8	No	Single	85K	**Yes**
9	No	Married	75K	**No**
10	No	Single	90K	**Yes**

Data Matrix

- If data objects have the same fixed set of numeric attributes, then the data objects can be thought of as points in a multi-dimensional space, where each dimension represents a distinct attribute
- Such data set can be represented by an m by n matrix, where there are m rows, one for each object, and n columns, one for each attribute

Projection of x Load	Projection of y load	Distance	Load	Thickness
10.23	5.27	15.22	2.7	1.2
12.65	6.25	16.22	2.2	1.1

Document Data

- Each document becomes a `term' vector,
  - each term is a component (attribute) of the vector,
  - the value of each component is the number of times the corresponding term occurs in the document.

	team	coach	pla y	ball	score	game	wi n	lost	timeout	season
Document 1	3	0	5	0	2	6	0	2	0	2
Document 2	0	7	0	2	1	0	0	3	0	0
Document 3	0	1	0	0	1	2	2	0	3	0

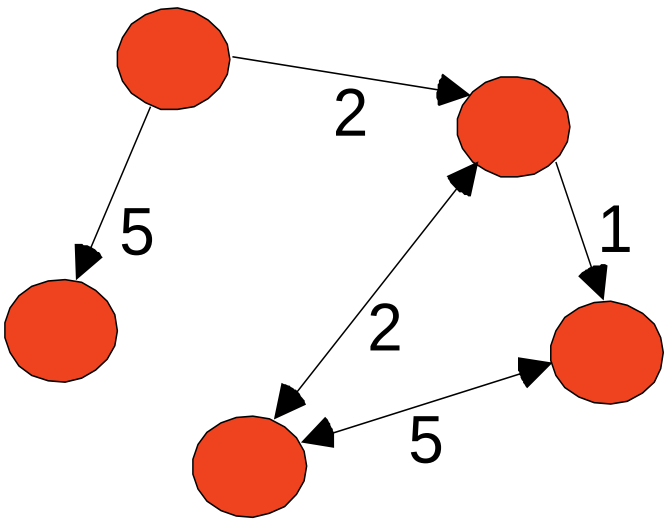
Transaction Data

- A special type of record data, where
  - each record (transaction) involves a set of items.
  - For example, consider a grocery store. The set of products purchased by a customer during one shopping trip constitute a transaction, while the individual products that were purchased are the items.

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

Graph Data

- Examples: Generic graph and HTML Links



•a graph is sometimes a more convenient and powerful representation of data

• can be used to capture relationship between data objects.

•Data objects themselves can be graphs.
•Ex: set of linked web pages can be represented as graphs

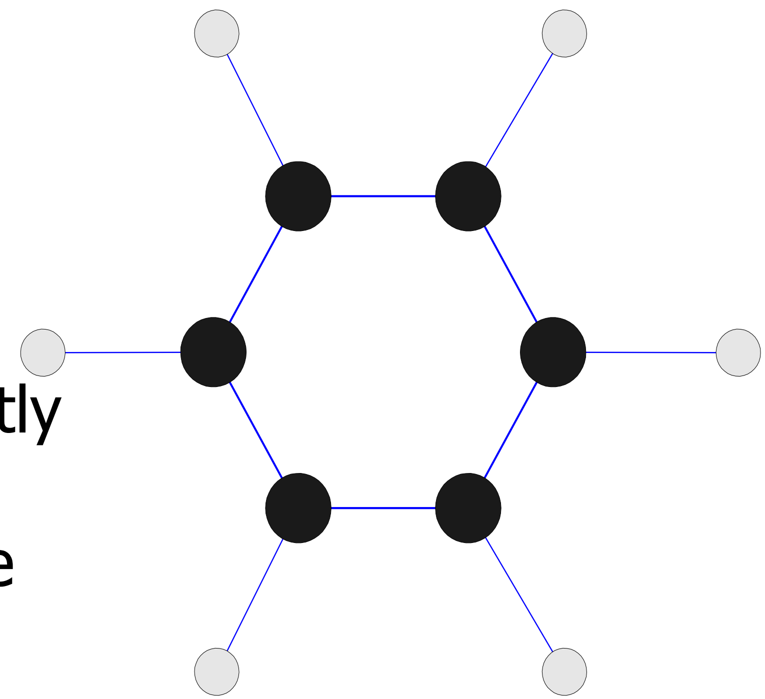
Chemical Data as a Graph

Data with objects that are graphs:-

- Objects have sub-objects that have relationships
- Ex : structure of chemical compounds

  Nodes – atoms

  Links – chemical compounds
- Benzene Molecule: C_6H_6
- **Mining Substructures**
- Which substructures occur frequently in a chemical compound?
- Is the presence of any substructure associated with any other?



Ordered Data

- Attributes have relationships that involve order in time/space
- Extension of a record data
- Each record has a time associated with it
- Each attribute can also be given a time stamp.

Ordered Data

- Sequences of transactions

Items/Events

(A B) (D) (C E)
(B D) (C) (E)
(C D) (B) (A E)

An element of
the sequence

- **Patterns ?**
- People who buy DVD player tend to buy DVDs in the period immediately following the purchase of DVD player.

Ordered Data

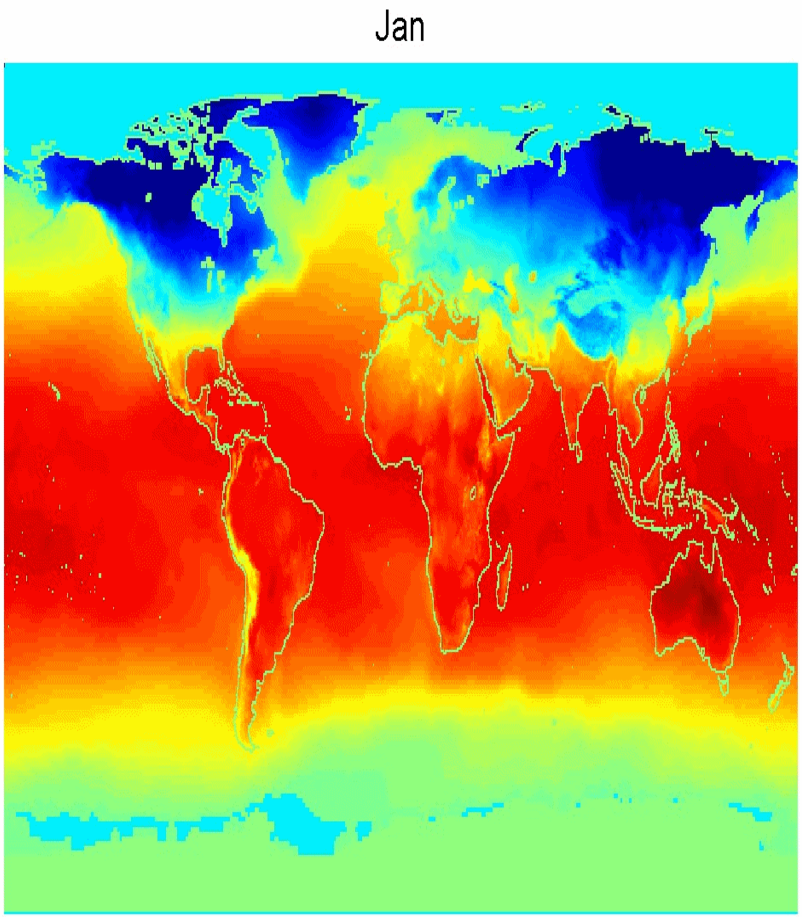
- Genomic sequence data – sequences of individual entities, letters/words. No time stamp
- Ex: genetic info of animals/plants in the form of sequences of genes/nucleotides.

GGTTCCGCCTTCAGCCCCGCGCC
CGCAGGGCCCGCCCCGCGCCGTC
GAGAAGGGCCCGCCTGGCGGGCG
GGGGGAGGCGGGGCCGCCCGAGC
CCAACCGAGTCCGACCAGGTGCC
CCCTCTGCTCGGCCTAGACCTGA
GCTCATTAGGCGGCAGCGGACAG
GCCAAGTAGAACACGCGAAGCGC
TGGGCTGCCTGCTGCGACCAGGG

- **Human genetic code sequence- 4 genes : A, T, G and C.**
- Mining on gene data – capture structure and properties of genes
- **Mining biological sequence data - BIOINFORMATICS**

Ordered Data

- Spatio-Temporal Data
- Spatial attributes
  – position and area
- Ex: weather data
- Earth sciences data
- - measures temp and pressure measured at points on latitude - longitude



Average Monthly Temperature of land and ocean

Time-series data

- A special type of sequential data
- Each record is a time-series i.e. a series of measurements taken over time
- Ex: financial data set has objects which are the time series of the daily prices of various stocks.
- Have temporal autocorrelation
- If two measurements are close in time, then their values are often similar.

Chapter 2: Data Preprocessing

- Data Preprocessing
  - Types of data?
- Data Preprocessing
  - Data Quality
  - Major Tasks in Data Preprocessing
- Data Cleaning
- Data Integration
- Data Reduction
- Data Transformation and Data Discretization
- Similarity and Dissimilarity measures.

Data Quality: Why Preprocess the Data?

- Measures for data quality: A multidimensional view
  - Accuracy: correct or wrong, accurate or not
  - Completeness: not recorded, unavailable, …
  - Consistency: some modified but some not, dangling, …
  - Timeliness: timely update?
  - Believability: how trustable the data are correct?
  - Interpretability: how easily the data can be understood?

Major Tasks 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 reduction**
  - Dimensionality reduction
  - Numerosity reduction
  - Data compression
- **Data transformation and data discretization**
  - Normalization
  - Concept hierarchy generation

Chapter 2: Data Preprocessing

- Data Preprocessing
  - Types of data?
- Data Preprocessing: An Overview
  - Data Quality
  - Major Tasks in Data Preprocessing
- Data Cleaning
- Data Integration
- Data Reduction
- Data Transformation and Data Discretization
- Similarity and Dissimilarity measures.

Data Cleaning

- Data in the Real World Is Dirty: Lots of potentially incorrect data, e.g., instrument faulty, human or computer error, transmission error
  - incomplete: lacking attribute values, lacking certain attributes of interest, or containing only aggregate data
    - e.g., *Occupation*=" " (missing data)
  - noisy: containing noise, errors, or outliers
    - e.g., *Salary*="−10" (an error)
  - inconsistent: containing discrepancies in codes or names, e.g.,
    - *Age*="42", *Birthday*="03/07/2010"
    - Was rating "1, 2, 3", now rating "A, B, C"
    - discrepancy between duplicate records
  - Intentional (e.g., *disguised missing* data)
    - Jan. 1 as everyone's birthday?

Incomplete (Missing) Data

- Data is not always available
  - E.g., many tuples have no recorded value for several attributes, such as customer income in sales data
- Missing data may be due to
  - equipment malfunction
  - inconsistent with other recorded data and thus deleted
  - data not entered due to misunderstanding
  - certain data may not be considered important at the time of entry
  - not register history or changes of the data
- Missing data may need to be inferred

How to Handle Missing Data?

- Ignore the tuple: usually done when class label is missing (when doing classification)—not effective when the % of missing values per attribute varies considerably
- Fill in the missing value manually: tedious + infeasible?
- Fill in it automatically with
  - a global constant : e.g., “unknown”, a new class?!
  - the attribute mean
  - the attribute mean for all samples belonging to the same class: smarter
  - the most probable value: inference-based such as Bayesian formula or decision tree

Noisy Data

- Noise: random error or variance in a measured variable
- Incorrect attribute values may be due to
  - faulty data collection instruments
  - data entry problems
  - data transmission problems
  - technology limitation
  - inconsistency in naming convention
- Other data problems which require data cleaning
  - duplicate records
  - incomplete data
  - inconsistent data

How to Handle Noisy Data?

- Binning
  - first sort data and partition into (equal-frequency) bins
  - then one can smooth by bin means, smooth by bin median, smooth by bin boundaries, etc.
- Regression
  - smooth by fitting the data into regression functions
- Clustering
  - detect and remove outliers
- Combined computer and human inspection
  - detect suspicious values and check by human (e.g., deal with possible outliers)

Data Cleaning as a Process

- Data discrepancy detection
  - Use metadata (e.g., domain, range, dependency, distribution)
  - Check field overloading
  - Check uniqueness rule, consecutive rule and null rule
  - Use commercial tools
    - Data scrubbing: use simple domain knowledge (e.g., postal code, spell-check) to detect errors and make corrections
    - Data auditing: by analyzing data to discover rules and relationship to detect violators (e.g., correlation and clustering to find outliers)
- Data migration and integration
  - Data migration tools: allow transformations to be specified
  - ETL (Extraction/Transformation/Loading) tools: allow users to specify transformations through a graphical user interface
- Integration of the two processes
  - Iterative and interactive (e.g., Potter's Wheels)

Chapter 3: Data Preprocessing

- **Data Preprocessing**
  - **Types of data?**
- **Data Preprocessing**
  - **Data Quality**
  - **Major Tasks in Data Preprocessing**
- **Data Cleaning**
- **Data Integration**
- **Data Reduction**
- **Data Transformation and Data Discretization**
- **Similarity and Dissimilarity measures.**

Data Integration

- **Data integration**:
  - Combines data from multiple sources into a coherent store
- Schema integration: e.g., A.cust-id $\equiv$ B.cust-#
  - Integrate metadata from different sources
- Entity identification problem:
  - Identify real world entities from multiple data sources, e.g., Bill Clinton = William Clinton
- Detecting and resolving data value conflicts
  - For the same real world entity, attribute values from different sources are different
  - Possible reasons: different representations, different scales, e.g., metric vs. British units

Handling Redundancy in Data Integration

- Redundant data occur often when integration of multiple databases
  - *Object identification*: The same attribute or object may have different names in different databases
  - *Derivable data:* One attribute may be a "derived" attribute in another table, e.g., annual revenue
- Redundant attributes may be able to be detected by *correlation analysis* and *covariance analysis*
- Careful integration of the data from multiple sources may help reduce/avoid redundancies and inconsistencies and improve mining speed and quality

Correlation Analysis (Nominal Data)

- **X^2 (chi-square) test**

$$\chi^2 = \sum \frac{(Observed - Expected)^2}{Expected}$$

- The larger the X^2 value, the more likely the variables are related
- The cells that contribute the most to the X^2 value are those whose actual count is very different from the expected count
- Correlation does not imply causality
  - # of hospitals and # of car-theft in a city are correlated
  - Both are causally linked to the third variable: population

Chi-Square Calculation: An Example

	Play chess	Not play chess	Sum (row)
Like science fiction	250(90)	200(360)	450
Not like science fiction	50(210)	1000(840)	1050
Sum(col.)	300	1200	1500

- X^2 (chi-square) calculation (numbers in parenthesis are expected counts calculated based on the data distribution in the two categories)

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

- It shows that like_science_fiction and play_chess are correlated in the group

Correlation Analysis (Numeric Data)

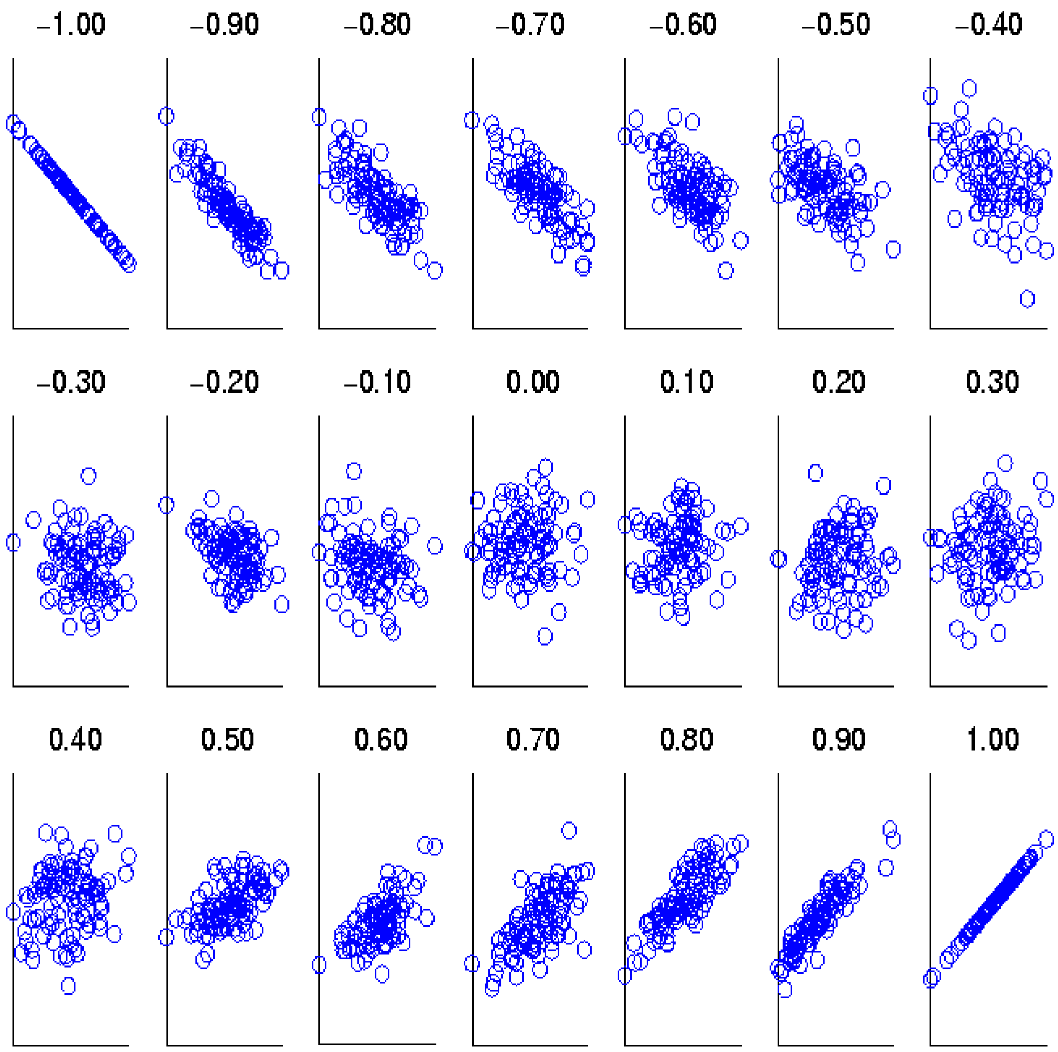
- Correlation coefficient (also called Pearson's product moment coefficient)

$$r_{A,B} = \frac{\sum_{i=1}^{n}(a_i - \bar{A})(b_i - \bar{B})}{(n-1)\sigma_A \sigma_B} = \frac{\sum_{i=1}^{n}(a_i b_i) - n\bar{A}\bar{B}}{(n-1)\sigma_A \sigma_B}$$

where n is the number of tuples, $\bar{A}$ and $\bar{B}$ are the respective means of A and B, σ_A and σ_B are the respective standard deviation of A and B, and $\Sigma(a_i b_i)$ is the sum of the AB cross-product.

- If $r_{A,B} > 0$, A and B are positively correlated (A's values increase as B's). The higher, the stronger correlation.
- $r_{A,B} = 0$: independent; $r_{AB} < 0$: negatively correlated

Visually Evaluating Correlation



Scatter plots showing the similarity from –1 to 1.

Correlation (viewed as linear relationship)

- Correlation measures the linear relationship between objects
- To compute correlation, we standardize data objects, A and B, and then take their dot product

$$a'_k = (a_k - mean(A)) / std(A)$$

$$b'_k = (b_k - mean(B)) / std(B)$$

$$correlation(A, B) = A' \bullet B'$$

Covariance (Numeric Data)

- Covariance is similar to correlation

$$Cov(A,B) = E((A-\bar{A})(B-\bar{B})) = \frac{\sum_{i=1}^{n}(a_i-\bar{A})(b_i-\bar{B})}{n}$$

Correlation coefficient: $$r_{A,B} = \frac{Cov(A,B)}{\sigma_A \sigma_B}$$

where n is the number of tuples, $\overline{A}$ and $\overline{B}$ are the respective mean or **expected values** of A and B, σ_A and σ_B are the respective standard deviation of A and B.

- **Positive covariance**: If $Cov_{A,B} > 0$, then A and B both tend to be larger than their expected values.
- **Negative covariance**: If $Cov_{A,B} < 0$ then if A is larger than its expected value, B is likely to be smaller than its expected value.
- **Independence**: $Cov_{A,B} = 0$ but the converse is not true:
  - Some pairs of random variables may have a covariance of 0 but are not independent. Only under some additional assumptions (e.g., the data follow multivariate normal distributions) does a covariance of 0 imply independence

Co-Variance: An Example

$$Cov(A,B) = E((A-\bar{A})(B-\bar{B})) = \frac{\sum_{i=1}^{n}(a_i-\bar{A})(b_i-\bar{B})}{n}$$

- It can be simplified in computation as

$$Cov(A,B) = E(A \cdot B) - \bar{A}\bar{B}$$

- Suppose two stocks A and B have the following values in one week: (2, 5), (3, 8), (5, 10), (4, 11), (6, 14).
- Question: If the stocks are affected by the same industry trends, will their prices rise or fall together?
  - E(A) = (2 + 3 + 5 + 4 + 6)/ 5 = 20/5 = 4
  - E(B) = (5 + 8 + 10 + 11 + 14) /5 = 48/5 = 9.6
  - Cov(A,B) = (2×5+3×8+5×10+4×11+6×14)/5 − 4 × 9.6 = 4
- Thus, A and B rise together since Cov(A, B) > 0.

Chapter 3: Data Preprocessing

- Data Preprocessing: An Overview
  - Data Quality
  - Major Tasks in Data Preprocessing
- Data Cleaning
- Data Integration
- Data Reduction
- Data Transformation and Data Discretization
- Summary

Data Reduction Strategies

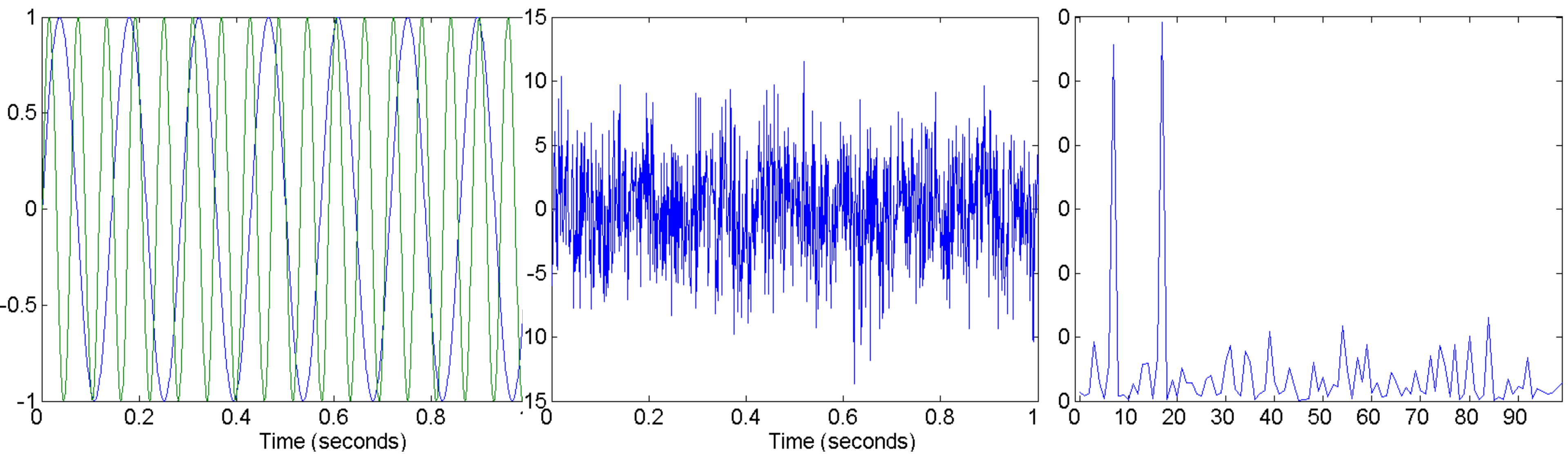
- **Data reduction**: Obtain a reduced representation of the data set that is much smaller in volume but yet produces the same (or almost the same) analytical results
- Why data reduction? — A database/data warehouse may store terabytes of data. Complex data analysis may take a very long time to run on the complete data set.
- Data reduction strategies
  - Dimensionality reduction, e.g., remove unimportant attributes
    - Wavelet transforms
    - Principal Components Analysis (PCA)
    - Feature subset selection, feature creation
  - Numerosity reduction (some simply call it: Data Reduction)
    - Regression and Log-Linear Models
    - Histograms, clustering, sampling
    - Data cube aggregation
  - Data compression

Data Reduction 1: Dimensionality Reduction

- **Curse of dimensionality**
  - When dimensionality increases, data becomes increasingly sparse
  - Density and distance between points, which is critical to clustering, outlier analysis, becomes less meaningful
  - The possible combinations of subspaces will grow exponentially
- **Dimensionality reduction**
  - Avoid the curse of dimensionality
  - Help eliminate irrelevant features and reduce noise
  - Reduce time and space required in data mining
  - Allow easier visualization
- **Dimensionality reduction techniques**
  - Wavelet transforms
  - Principal Component Analysis
  - Supervised and nonlinear techniques (e.g., feature selection)

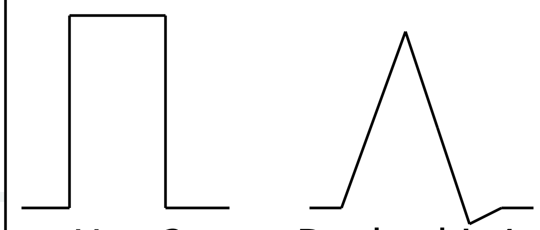
Mapping Data to a New Space

- Fourier transform
- Wavelet transform



Two Sine Waves **Two Sine Waves + Noise** **Frequency**

Wavelet Transformation



- Discrete wavelet transform (DWT) for linear signal processing, multi-resolution analysis
- Compressed approximation: store only a small fraction of the strongest of the wavelet coefficients
- Similar to discrete Fourier transform (DFT), but better lossy compression, localized in space
- Method:
  - Length, L, must be an integer power of 2 (padding with 0's, when necessary)
  - Each transform has 2 functions: smoothing, difference
  - Applies to pairs of data, resulting in two set of data of length L/2
  - Applies two functions recursively, until reaches the desired length

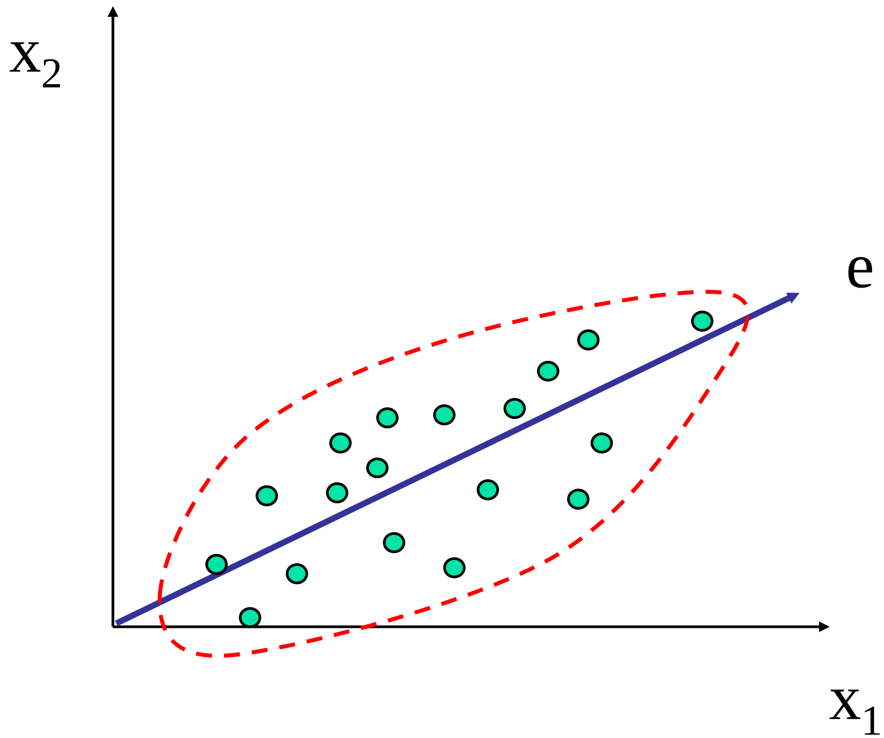
Wavelet Decomposition

- Wavelets: A math tool for space-efficient hierarchical decomposition of functions
- S = [2, 2, 0, 2, 3, 5, 4, 4] can be transformed to $S_\wedge$ = [$2^3/_4$, $-1^1/_4$, $^1/_2$, 0, 0, -1, -1, 0]
- Compression: many small detail coefficients can be replaced by 0's, and only the significant coefficients are retained

Resolution	Averages	Detail Coefficients
8	$[2, 2, 0, 2, 3, 5, 4, 4]$	
4	$[2, 1, 4, 4]$	$[0, -1, -1, 0]$
2	$[1\frac{1}{2}, 4]$	$[\frac{1}{2}, 0]$
1	$[2\frac{3}{4}]$	$[-1\frac{1}{4}]$

Principal Component Analysis (PCA)

- Find a projection that captures the largest amount of variation in data
- The original data are projected onto a much smaller space, resulting in dimensionality reduction. We find the eigenvectors of the covariance matrix, and these eigenvectors define the new space



Principal Component Analysis (Steps)

- Given *N* data vectors from *n*-dimensions, find $k \le n$ orthogonal vectors (*principal components*) that can be best used to represent data
  - Normalize input data: Each attribute falls within the same range
  - Compute *k* orthonormal (unit) vectors, i.e., *principal components*
  - Each input data (vector) is a linear combination of the *k* principal component vectors
  - The principal components are sorted in order of decreasing "significance" or strength
  - Since the components are sorted, the size of the data can be reduced by eliminating the *weak components*, i.e., those with low variance (i.e., using the strongest principal components, it is possible to reconstruct a good approximation of the original data)
- Works for numeric data only

Attribute Subset Selection

- Another way to reduce dimensionality of data
- Redundant attributes
  - Duplicate much or all of the information contained in one or more other attributes
  - E.g., purchase price of a product and the amount of sales tax paid
- Irrelevant attributes
  - Contain no information that is useful for the data mining task at hand
  - E.g., students' ID is often irrelevant to the task of predicting students' GPA

Heuristic Search in Attribute Selection

- There are 2^d possible attribute combinations of d attributes
- Typical heuristic attribute selection methods:
  - Best single attribute under the attribute independence assumption: choose by significance tests
  - Best step-wise feature selection:
    - The best single-attribute is picked first
    - Then next best attribute condition to the first, ...
  - Step-wise attribute elimination:
    - Repeatedly eliminate the worst attribute
  - Best combined attribute selection and elimination
  - Optimal branch and bound:
    - Use attribute elimination and backtracking

Attribute Creation (Feature Generation)

- Create new attributes (features) that can capture the important information in a data set more effectively than the original ones
- Three general methodologies
  - Attribute extraction
    - Domain-specific
  - Mapping data to new space (see: data reduction)
    - E.g., Fourier transformation, wavelet transformation, manifold approaches (not covered)
  - Attribute construction
    - Combining features (see: discriminative frequent patterns in Chapter 7)
    - Data discretization

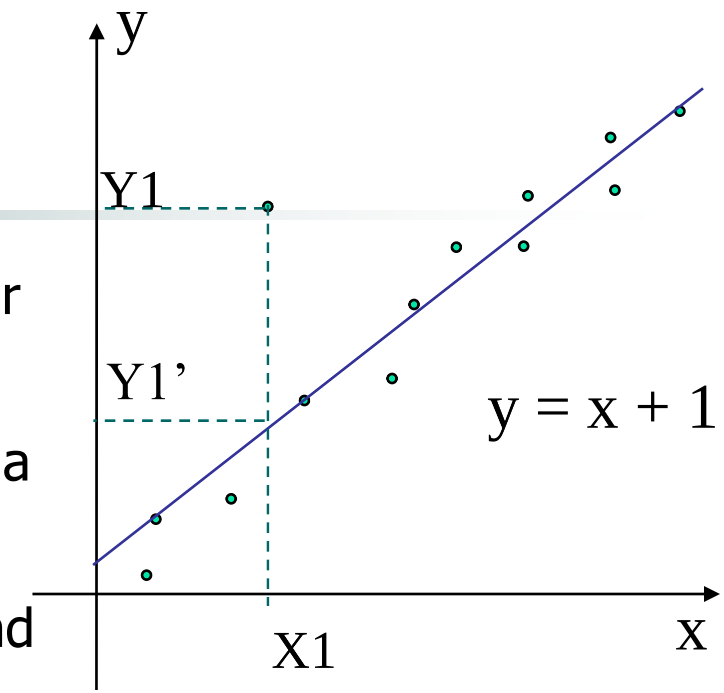
Data Reduction 2: Numerosity Reduction

- Reduce data volume by choosing alternative, *smaller forms* of data representation
- **Parametric methods** (e.g., regression)
  - Assume the data fits some model, estimate model parameters, store only the parameters, and discard the data (except possible outliers)
  - Ex.: Log-linear models—obtain value at a point in *m*-D space as the product on appropriate marginal subspaces
- **Non-parametric** methods
  - Do not assume models
  - Major families: histograms, clustering, sampling, …

Parametric Data Reduction: Regression and Log-Linear Models

- **Linear regression**
  - Data modeled to fit a straight line
  - Often uses the least-square method to fit the line
- **Multiple regression**
  - Allows a response variable Y to be modeled as a linear function of multidimensional feature vector
- **Log-linear model**
  - Approximates discrete multidimensional probability distributions

Regression Analysis



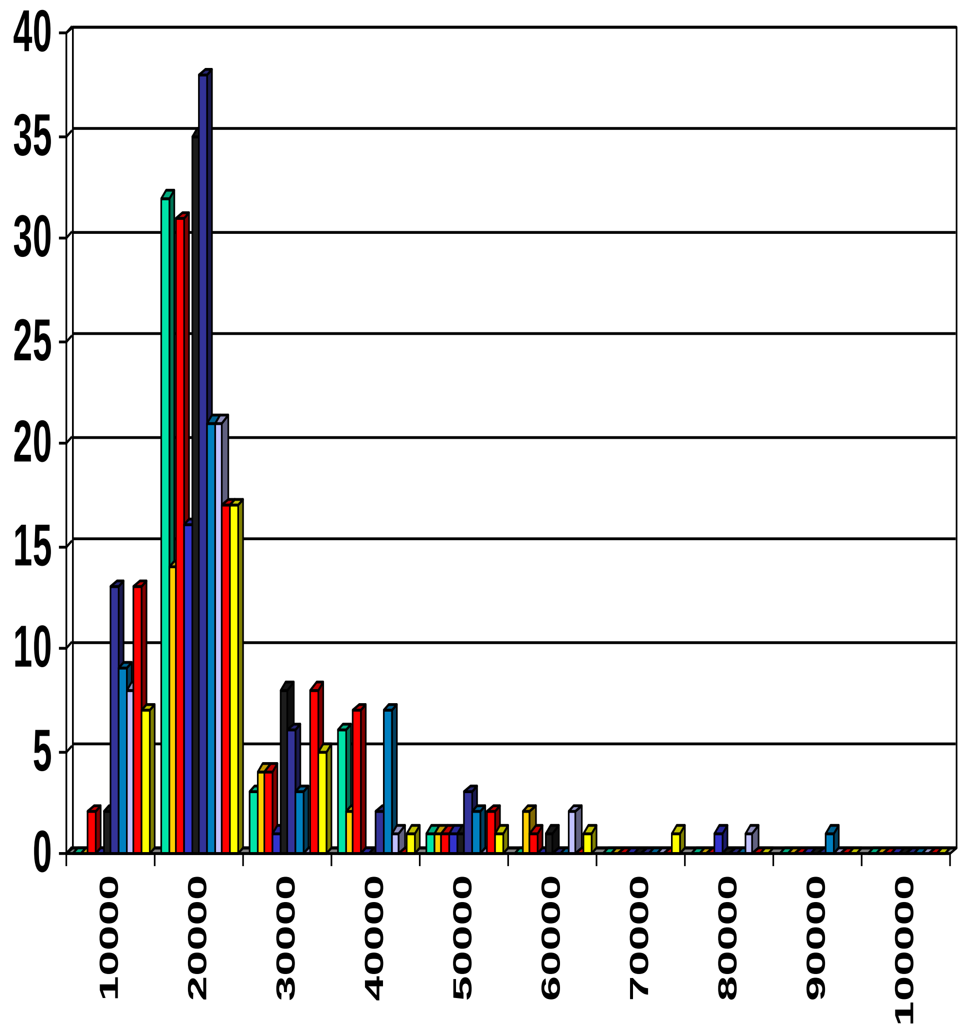
- Regression analysis: A collective name for techniques for the modeling and analysis of numerical data consisting of values of a ***dependent variable*** (also called ***response variable*** or *measurement*) and of one or more *independent variables* (aka. ***explanatory variables*** or ***predictors***)
- The parameters are estimated so as to give a "**best fit**" of the data
- Most commonly the best fit is evaluated by using the ***least squares method***, but other criteria have also been used
- Used for prediction (including forecasting of time-series data), inference, hypothesis testing, and modeling of causal relationships

Regress Analysis and Log-Linear Models

- Linear regression: $Y = w X + b$
  - Two regression coefficients, w and b, specify the line and are to be estimated by using the data at hand
  - Using the least squares criterion to the known values of $Y_1, Y_2, \ldots, X_1, X_2, \ldots$.
- Multiple regression: $Y = b_0 + b_1 X_1 + b_2 X_2$
  - Many nonlinear functions can be transformed into the above
- Log-linear models:
  - Approximate discrete multidimensional probability distributions
  - Estimate the probability of each point (tuple) in a multi-dimensional space for a set of discretized attributes, based on a smaller subset of dimensional combinations
  - Useful for dimensionality reduction and data smoothing

Histogram Analysis

- Divide data into buckets and store average (sum) for each bucket
- Partitioning rules:
  - Equal-width: equal bucket range
  - Equal-frequency (or equal-depth)



Data Cube Aggregation

- The lowest level of a data cube (base cuboid)
  - The aggregated data for an individual entity of interest
  - E.g., a customer in a phone calling data warehouse
- Multiple levels of aggregation in data cubes
  - Further reduce the size of data to deal with
- Reference appropriate levels
  - Use the smallest representation which is enough to solve the task
- Queries regarding aggregated information should be answered using data cube, when possible

Chapter 3: Data Preprocessing

- Data Preprocessing: An Overview
  - Data Quality
  - Major Tasks in Data Preprocessing
- Data Cleaning
- Data Integration
- Data Reduction
- Data Transformation and Data Discretization
- Summary

Data Transformation

- Discretization
  - Supervised
    - Entropy – based
  - Unsupervised
    - Equal width and equal frequency
- Normalization
  - Min-max
  - Z-score
  - Decimal scaling
- Binarization

Discretization/Quantization

- **Three types of attributes:**
  - Nominal – values from an unordered set
  - Ordinal – values from an ordered set
  - Continuous – real numbers

- **Discretization :**
  - Divide the range of a continuous attribute into intervals
  - Some classification algos only accept categorical attributes
  - Reduce data size by discretization
  - Prepare for further analysis

Transformation by Discretization

Some Algorithms require nominal/ discrete attributes

Discretizing Numeric Attributes

- We can turn a numeric attribute into a nominal/categorical one by using some sort of *discretization*.
- This involves dividing the range of possible values into subranges called *buckets* or *bins*.
  - example: an *age* attribute could be divided into these bins:

    child: 0-12
    teen: 12-17
    young: 18-35
    middle: 36-59
    senior: 60-

Discretization methods

- **Unsupervised**
  - Independent of the class label
  - Ex: Equal width binning, equal frequency binning
- **Supervised**
  - Dependent on the class label
  - Ex: entropy based binning

Unsupervised Discretization

Equal-width binning divides the range of possible values into N subranges of the same size.

- bin width = (max value – min value) / N
- example: if the observed values are all between 0-100, we could create 5 bins as follows:

  width = (100 – 0)/5 = 20

  bins: [0-20], (20-40], (40-60], (60-80], (80-100]

  [or] means the endpoint is included

  (or) means the endpoint is not included

- typically, the first and last bins are extended to allow for values outside the range of observed values

  (-infinity-20], (20-40], (40-60], (60-80], (80-infinity)

Equal Width binning…

- **Advantages :**
  - Simple and easy to implement
  - Produce a reasonable abstraction of data

- **Disadvantages :**
  - Unsupervised
  - Where does N come from?
  - If there are many occurrences of one range in the data set, it would be useless for the data mining task.

Simple Discretization Methods (cont.)

- *Equal-frequency* or *equal-height binning* divides the range of possible values into N bins, each of which holds the same number of training instances.
  - example: let's say we have 10 training examples with the following values for the attribute that we're discretizing:

    5, 7, 12, 35, 65, 82, 84, 88, 90, 95

    to create 5 bins, we would divide up the range of values so that each bin holds 2 of the training examples:

    5, 7, | 12, 35, | 65, 82, | 84, 88, | 90, 95

- To select the boundary values for the bins, this method typically chooses a value halfway between the training examples on either side of the boundary.
  - examples: $(7 + 12)/2 = 9.5$ $(35 + 65)/2 = 50$

Entropy Based Discretization- Supervised

- Uses the class info present in the data
- Entropy(info content) is calculated based on the class label
- Tries to find the best split so that bins are as pure as possible.
- Pure bin : majority of the values in a bin should correspond to the same class.
- Purity of a bin is measured using its entropy
- Entropy
  - Zero – perfectly pure bin
  - Max (1) – impure – equal class distribution

Entropy Based Discretization- Supervised…

- Ex: Two class problem, classes are + and –
- If entropy of a bin is 0, then all values of the bin $\in C_{+}$ or C_{-}.
- Entropy is 1, if bin is mixed. Half of the bin $\in C_{+}$ and other half $\in C_{-}$.

Pure Bin

10	15	7
+	+	+

Totally Impure Bin

10	6	13	7
+	-	-	+

Entropy

- k = no. of different class labels
- m_i = no. of values in the i^{th} interval of a partition.
- m_{ij} = no. of values of class j in i^{th} interval

  where j = 1 to k

Then entropy of the i^{th} interval is

$$e_i = -\sum_{j=1}^{k} p_{ij}\, log_2 p_{ij}$$

Where $p_{ij} = \frac{m_{ij}}{m_i}$ is the probability of class j in i^{th} interval.

Procedure…

1. Sort the attb values to be discretized, S
2. Bisect the initial values so that the resulting two intervals have minimum entropy.
   i. Consider each value T as a possible split point Where T = midpoint of each consecutive attb values.
   ii. Compute the information gain before and after choosing T as a split point.

      Gain = E(S) – E(T,S)

   i. Select the best T which gives the highest info gain as the optimum split.

3. Repeat step 2 with another interval (highest entropy) until a user specified no. of intervals is reached or some stopping criterion is met.

Normalization

- Scale attribute values to fall within a small-specified range.
  - Min-max
  - Z-score
  - Decimal scaling
- Ex : kNN classifiers

Customer	**Age**	**Income**	**Purchased Product?**
1	45	75K	Book
2	39	100K	TV
3	39	150K	DVD
4	58	51K	???

Data Transformation: Normalization

- min-max normalization

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

where min_A and max_A are the minimum and maximum values of attribute A, and $[new\_min_A, new\_max_A]$ is the new range

- Example: Attribute *income* has values
  - \$12,000, \$20,000, \$25,000, \$30,000, \$45,000, \$60,000, \$73,600, \$98,000
  - normalized into values in range [0, 1]:

    0, 0.093, 0.151, 0.209, 0.384, 0.558, 0.716, 1

- Problems:
  - "Out of bounds" error occurs if a future input case falls outside the original range for A
  - A too big or too small value could be noise. If they are used as min or max value for normalization, the results are not reliable.

Data Transformation: Normalization (Cont

- z-score normalization

$$v' = \frac{v - mean_A}{s_A}$$

where $mean_A$ is the mean of attribute A and s_A is the standard deviation of A (suppose values are : v_1, v_2, …, v_n):

$$s_A = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(v_i - mean_A)^2}$$

- Example:
  - The mean and standard deviation of the attribute *income* are 45,450 and 22,000
  - With z-score normalization, the values are transformed into:

    -1.52, -1.16, -0.93, -0.7, -0.02, 0.66, 1.28, 2.39
- Advantages:
  - useful when the actual min and max are unknown
  - better deal with outliers than min-max normalization

Data Transformation: Normalization (Contd.)

- Normalization by decimal scaling

$$v_i' = \frac{v_i}{10^k}$$

where k is the smallest integer such that Max($|v_i'|$)<1

- Example:
  - Suppose the recorded values of A range from -986 to 97
  - The maximum absolute value of A is 986.
  - Then k=3
  - -986 is normalized to -0.986 and 97 is normalized to 0.097

$V_i' = \frac{V_i}{10^k}$ k = ? If max($|V_i'|$) < 1 Ans : k = 3

Binarization

- Transforming a continuous or discrete attribute into one or more binary attribute
- Why?
  - ARM can be done only on binarized data.
  - But i/p data set may have numeric/discrete attributes

Binarizing a categorical data:-

If the categorical attb has m distinct values,

1. assign a unique integer from 0 to m-1 to each value.
2. Represent each integer using unique bit combinations

Binarization....

- Ex :

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

Summary

- **Data quality**: accuracy, completeness, consistency, timeliness, believability, interpretability
- **Data cleaning**: e.g. missing/noisy values, outliers
- **Data integration** from multiple sources:
  - Entity identification problem
  - Remove redundancies
  - Detect inconsistencies
- **Data reduction**
  - Dimensionality reduction
  - Numerosity reduction
  - Data compression
- **Data transformation and data discretization**
  - Normalization
  - Concept hierarchy generation

References

- D. P. Ballou and G. K. Tayi. Enhancing data quality in data warehouse environments. Comm. of ACM, 42:73-78, 1999
- A. Bruce, D. Donoho, and H.-Y. Gao. Wavelet analysis. *IEEE Spectrum*, Oct 1996
- T. Dasu and T. Johnson. Exploratory Data Mining and Data Cleaning. John Wiley, 2003
- J. Devore and R. Peck. *Statistics: The Exploration and Analysis of Data*. Duxbury Press, 1997.
- H. Galhardas, D. Florescu, D. Shasha, E. Simon, and C.-A. Saita. Declarative data cleaning: Language, model, and algorithms. *VLDB'01*
- M. Hua and J. Pei. Cleaning disguised missing data: A heuristic approach. *KDD'07*
- H. V. Jagadish, et al., Special Issue on Data Reduction Techniques. Bulletin of the Technical Committee on Data Engineering, 20(4), Dec. 1997
- H. Liu and H. Motoda (eds.). *Feature Extraction, Construction, and Selection: A Data Mining Perspective*. Kluwer Academic, 1998
- J. E. Olson. *Data Quality: The Accuracy Dimension*. Morgan Kaufmann, 2003
- D. Pyle. Data Preparation for Data Mining. Morgan Kaufmann, 1999
- V. Raman and J. Hellerstein. Potters Wheel: An Interactive Framework for Data Cleaning and Transformation, VLDB'2001
- T. Redman. *Data Quality: The Field Guide*. Digital Press (Elsevier), 2001
- R. Wang, V. Storey, and C. Firth. A framework for analysis of data quality research. IEEE Trans. Knowledge and Data Engineering, 7:623-640, 1995