
Università degli Studi di Milano
Master Degree in Computer Science

Information Management course

Teacher: Alberto Ceselli

Lecture 03 : 16/10/2014

Data Mining:

Concepts and Techniques


(3rd ed.)

— Chapter 3 —

Jiawei Han, Micheline Kamber, and Jian Pei
University of Illinois at Urbana-Champaign &
Simon Fraser University

©2011 Han, Kamber & Pei. All rights reserved.

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

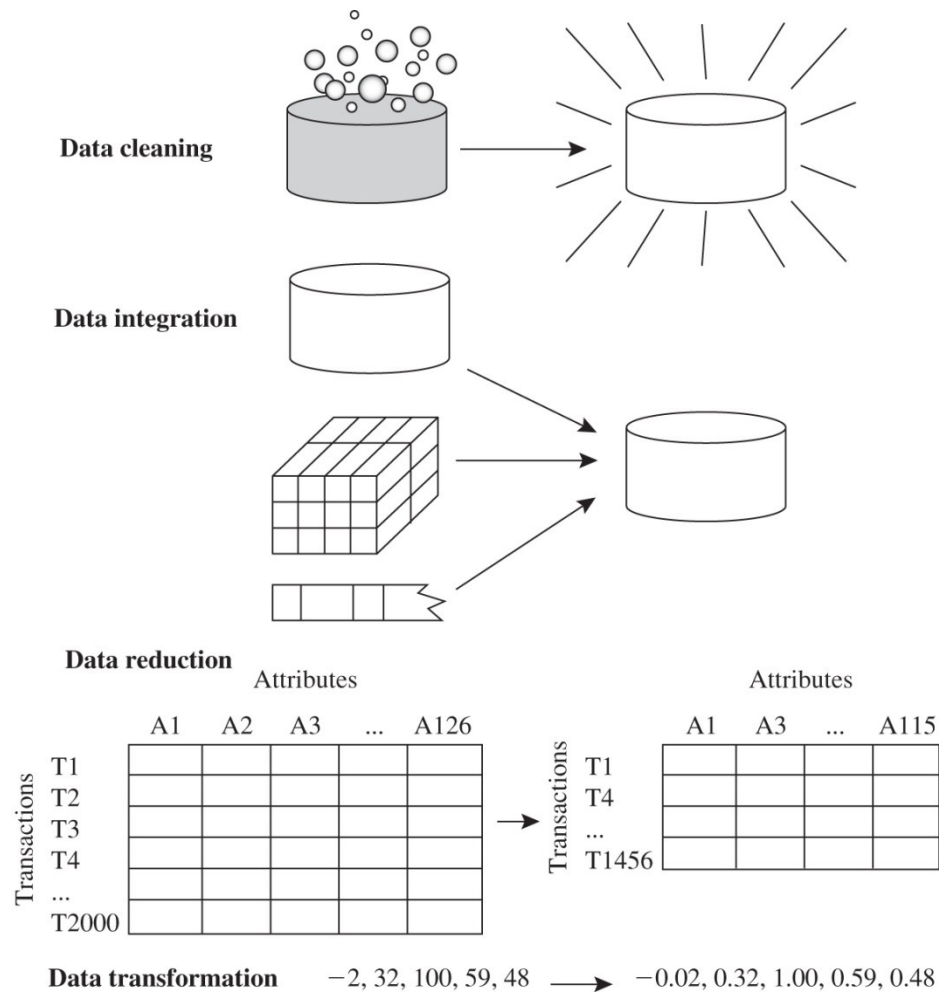
Data Quality: Why Preprocess the Data?

- 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

Major Tasks in Data Preprocessing



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 Cleaning

- Data in the Real World Is Dirty (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?

How to Handle Missing Data?

- Ignore the tuple (e.g. when class label is missing and doing classification) → simple, but loss of data
- Fill in the missing value manually
→ tedious + infeasible?
- Fill in it automatically with
 - global const (e.g., “unknown”) → a new class?!
 - the attribute mean or median
 - 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


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.
- Clustering
 - detect and remove outliers
- Regression
 - smooth by fitting the data into regression functions
- Filtering
 - Apply transforms (e.g whitening)
- Combined computer and human inspection

Data Cleaning as a Process

- Data discrepancy detection
 - Use knowledge about data → use **metadata** (e.g., domain, range, dependency, distribution) i.e. **know your data!**
 - 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) → already “data mining”
- Data migration and integration
 - Data migration tools: allow transformations to be specified
 - ETL (Extraction/Transformation/Loading) tools (GUI)
- Integration of the two processes
 - Iterative and interactive (e.g., Potter’s Wheels)

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 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 integrating 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 detected by *correlation analysis* and *covariance analysis*

Correlation Analysis (Nominal Data)

■ χ^2 (chi-square) test

- Attribute A has c values ($a_1 \dots a_c$)
- Attribute B has r values ($b_1 \dots b_r$)
- Build a contingency table [o_{ij}], having 1 row for each a_i , one col for each b_j
- o_{ij} is the observed frequency (number of tuples having value a_i for A and b_j for B)

$$e_{ij} = \frac{\text{count}(A=a_i) \times \text{count}(B=b_j)}{\text{num. data tuples}}$$

$$\chi^2 = \sum_i \sum_j \frac{(o_{ij} - e_{ij})^2}{e_{ij}}$$

Correlation Analysis (Nominal Data)

- The larger the χ^2 value, the more likely the variables are related
- The cells that contribute the most to the χ^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

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

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

- χ^2 (chi-square) calculation (numbers in parenthesis are e_{ij})

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

- K x K table → K categories → (K-1) degrees of freedom (1 in the example)
- From chi-square distribution, the value for rejecting hypothesis of independency at 0.001 significance level is 10.828 → **strong correlation**

Deg. freedom	0.00	0.02	0.06	0.15	0.46	1.07	1.64	2.71	3.84	6.64	10.83
1	0.10	0.21	0.45	0.71	1.39	2.41	3.22	4.60	5.99	9.21	13.82
2	0.35	0.58	1.01	1.42	2.37	3.66	4.64	6.25	7.82	11.34	16.27
3	0.71	1.06	1.65	2.20	3.36	4.88	5.99	7.78	9.49	13.28	18.47
4	1.14	1.61	2.34	3.00	4.35	6.06	7.29	9.24	11.07	15.09	20.52
5	1.63	2.20	3.07	3.83	5.35	7.23	8.56	10.64	12.59	16.81	22.46
6	2.17	2.83	3.82	4.67	6.35	8.38	9.80	12.02	14.07	18.48	24.32
7	2.73	3.49	4.59	5.53	7.34	9.52	11.03	13.36	15.51	20.09	26.12
8	3.32	4.17	5.38	6.39	8.34	10.66	12.24	14.68	16.92	21.67	27.88
9	3.94	4.80	6.10	7.27	9.34	11.78	13.44	15.99	18.31	23.21	29.59
10	4.60	5.58	7.00	8.33	10.59	12.91	14.68	17.36	19.68	24.72	31.57
p-val	0.95	0.9	0.8	0.7	0.5	0.3	0.2	0.1	0.05	0.01	0.001

1 - Cum. Distr. Funct. =
significance level



Covariance (Numeric Data)

- Covariance:
 - Attributes A and B
 - $n \rightarrow$ number of tuples
 - \bar{A} and $\bar{B} \rightarrow$ respective means of A and B
 - σ_A and $\sigma_B \rightarrow$ the respective standard deviation of A and B

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

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

Covariance (Numeric Data)

- Covariance:

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

- **Dep. w. positive correlation** $\leftarrow \rightarrow$ **positive covariance**
If $\text{Cov}_{A,B} > 0$, then when A is larger (resp. smaller) than its expected value, B is larger (resp. smaller) as well
- **Dep. w. negative correlation** $\leftarrow \rightarrow$ **negative covariance**
If $\text{Cov}_{A,B} < 0$, then when A is larger than its expected value, B is likely to be smaller than its expected value (and vice versa)
- **Independence** $\rightarrow \text{Cov}_{A,B} = 0$ (but the converse is not always 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) a covariance of 0 does 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

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

- 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?
 - $E(A) = (2 + 3 + 5 + 4 + 6) / 5 = 20 / 5 = 4$
 - $E(B) = (5 + 8 + 10 + 11 + 14) / 5 = 48 / 5 = 9.6$
 - $Cov(A, B) = (2 \times 5 + 3 \times 8 + 5 \times 10 + 4 \times 11 + 6 \times 14) / 5 - 4 \times 9.6 = 4$
- Thus, A and B rise together since $Cov(A, B) > 0$.

Correlation Analysis (Numeric Data)

- Correlation coefficient (also called **Pearson's product moment coefficient**)
 - Attributes A and B
 - n → number of tuples
 - \bar{A} and \bar{B} → respective means of A and B
 - σ_A and σ_B → the respective standard deviation of A and B

$$\rho_{X,Y} = \frac{\text{cov}(X, Y)}{\sigma_X \sigma_Y} = \frac{E[(X - \mu_X)(Y - \mu_Y)]}{\sigma_X \sigma_Y}$$

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

Correlation Analysis (Numeric Data)

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

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

- 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

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

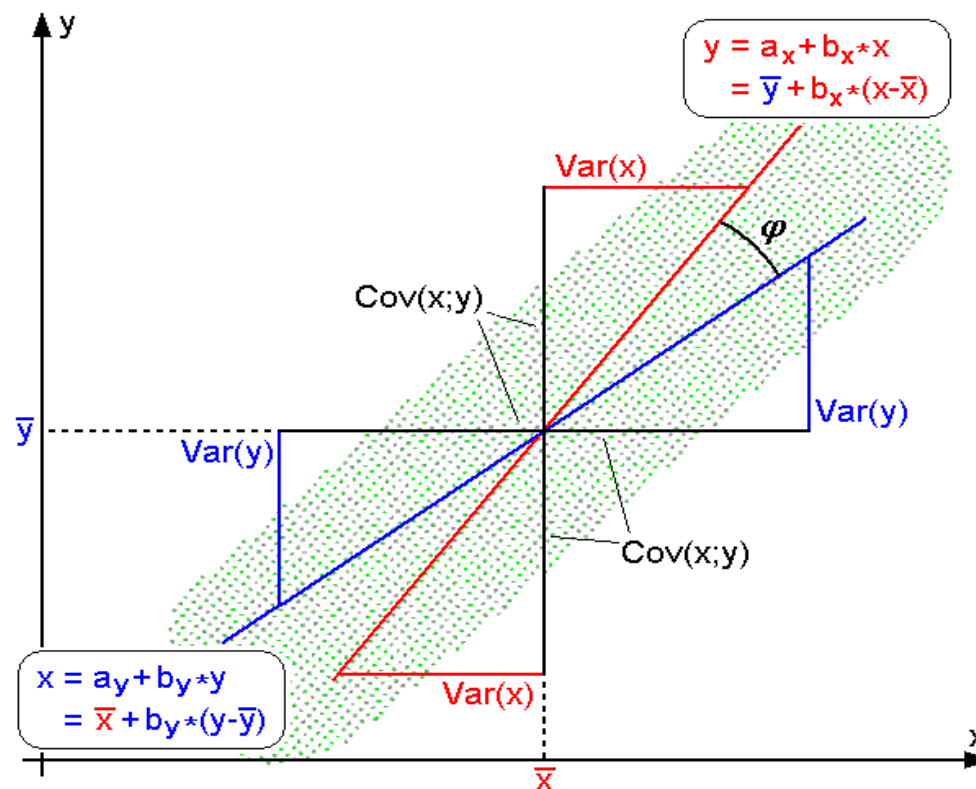
$$a'_k = (a_k - \text{mean}(A)) / \text{std}(A)$$

$$b'_k = (b_k - \text{mean}(B)) / \text{std}(B)$$

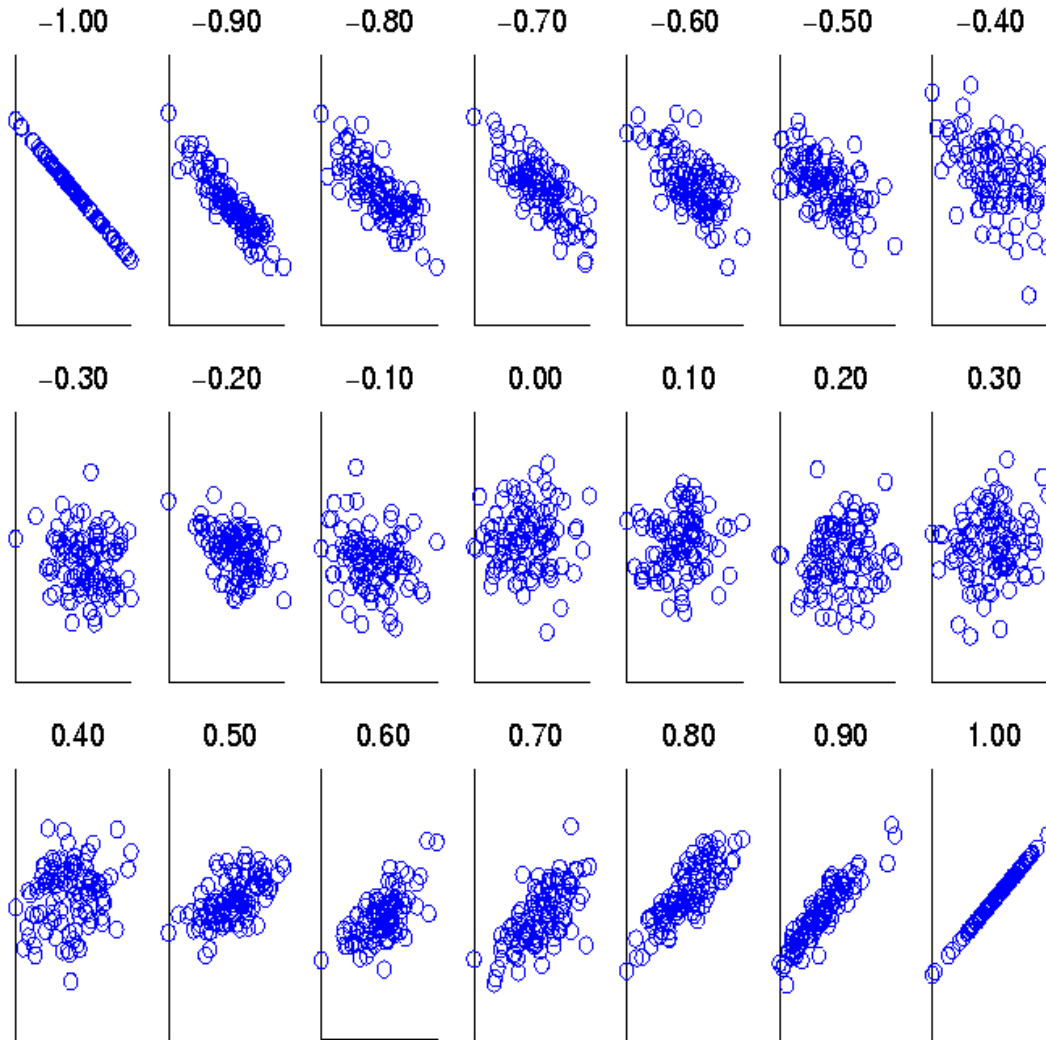
$$\text{correlation}(A, B) = A' \bullet B'$$

Correlation Analysis (Numeric Data)

- Geometrically: the cosine of the angle between the two vectors, after centering (or possible regression lines)




Visually Evaluating Correlation



Scatter plots showing the similarity from -1 to 1.

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? Computational issues in big data!
- Data reduction strategies
 - **Numerosity reduction** (or simply “Data Reduction” → red. the number of data objects)
 - Sampling
 - Histograms, clustering
 - Regression and Log-Linear Models
 - Data cube aggregation
 - **Dimensionality reduction** (→ red. the number of attributes)
 - Principal Components Analysis (PCA)
 - Feature subset selection, feature creation
 - Transforms (Fourier, Wavelet, Whitening ...)
 - **Data compression**

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? Computational issues in big data!
- Data reduction strategies
 - Numerosity reduction (or simply “Data Reduction” → red. the number of data objects)
 - Sampling
 - Histograms, clustering
 - Regression and Log-Linear Models
 - Data cube aggregation
 - Dimensionality reduction (→ red. the number of attributes)
 - Principal Components Analysis (PCA)
 - Feature subset selection, feature creation
 - Transforms (Fourier, Wavelet, Whitening ...)
 - Data compression

Numerosity Reduction: sampling

- Sampling: obtaining a small sample s to represent the whole data set N
- Allow a mining algorithm to run in complexity that is potentially sub-linear to the size of the data
- Key principle: Choose a **representative** subset of the data
 - Simple random sampling may have very poor performance in the presence of skew
 - Develop adaptive sampling methods, e.g., stratified sampling:
- Note: Sampling may not reduce database I/Os (page at a time)

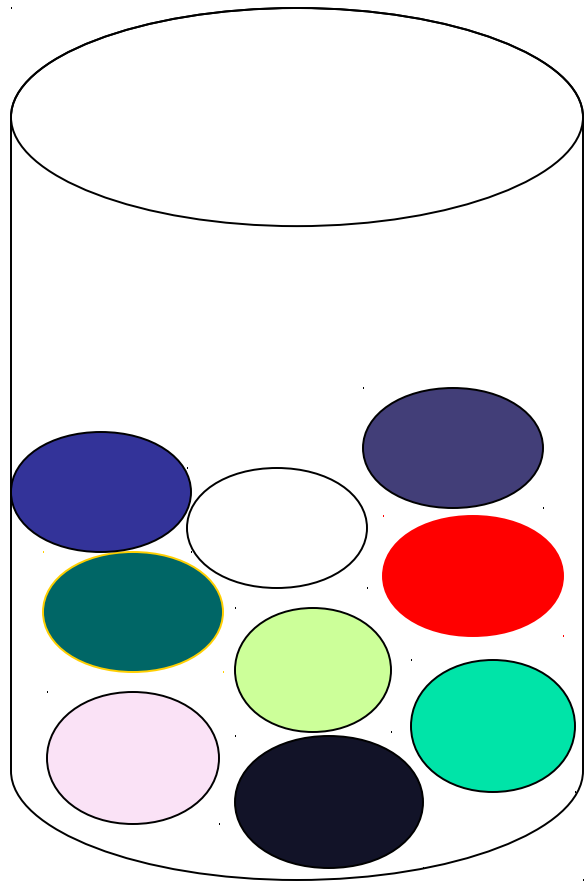
Sampling

- Beware! In general, sampling design (e.g. for surveys) is a serious issue:
 - Cochran, W.G. (1977). **Sampling techniques**, 3rd ed. New York: John Wiley & Sons
 - Lohr, S. (2009). **Sampling: Design and Analysis**. Duxbury Press
- in data sampling for automatic analyses we're more constrained (and therefore simplified)
- still we can exploit general techniques

Types of Sampling

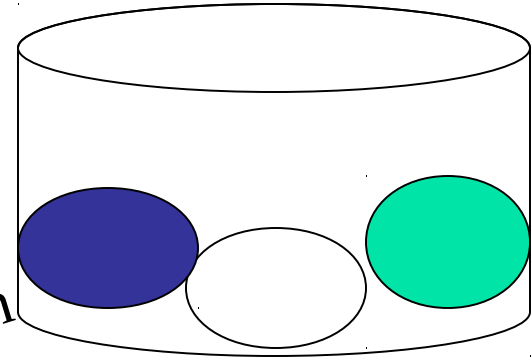
- **Systematic sampling**
 - Choose equally-spaced data objects (or even contiguous elements to reduce I/O)
- **Simple random sampling**
 - There is an equal probability of selecting any particular item
- **Sampling without / with replacement**
 - Once an object is selected, it is removed (resp. not removed) from the population
- **Stratified sampling:**
 - Partition the data set, and draw samples from each partition (proportionally, i.e., approximately the same percentage of the data)
 - Used in conjunction with skewed data

Sampling: With or without Replacement

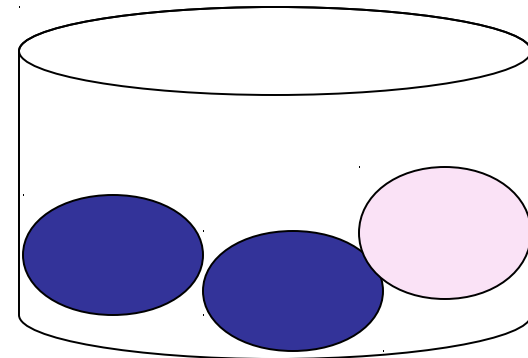


Raw Data

SRSWOR
(simple random
sample without
replacement)



SRSWR

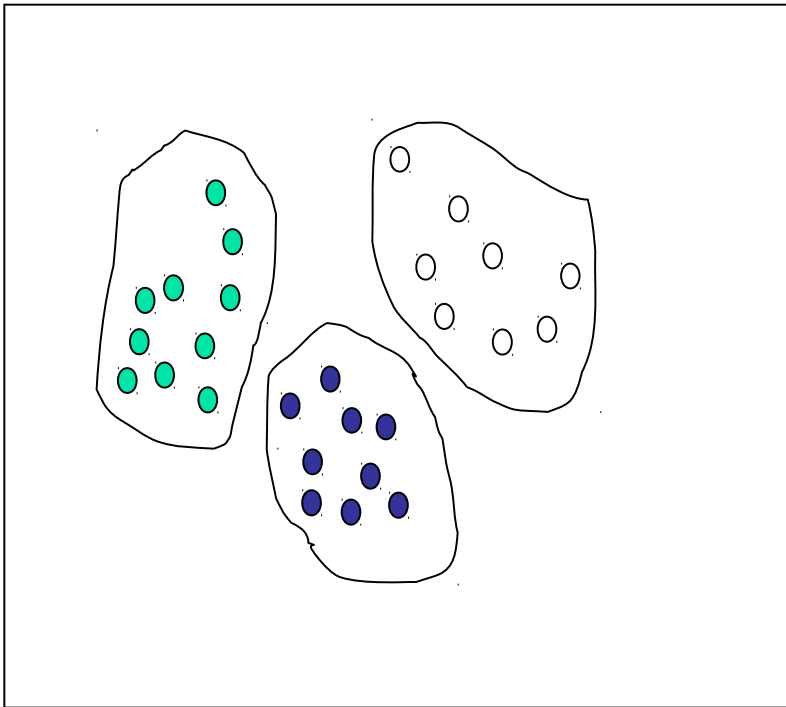


Types of Sampling

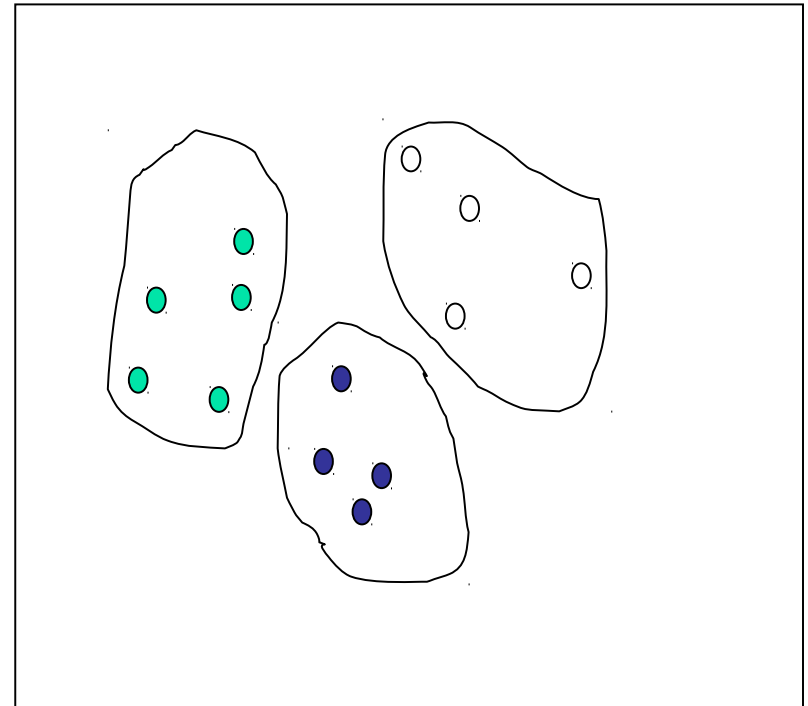
- **Stratified sampling:**
 - Choose a “category” attribute y
 - Partition the data set according to y values (strata)
 - Draw samples independently from each class (e.g. proportionally, i.e. approx. same % of data)
 - Better for skewed data
- **Clustered sampling:**
 - Cluster data, use cluster classes as category in a stratified sampling
- **Single stage / multi stage sampling:**
 - Perform hierarchical stratification or clustering
 - Sample recursively

Sampling: Cluster or Stratified Sampling

Raw Data

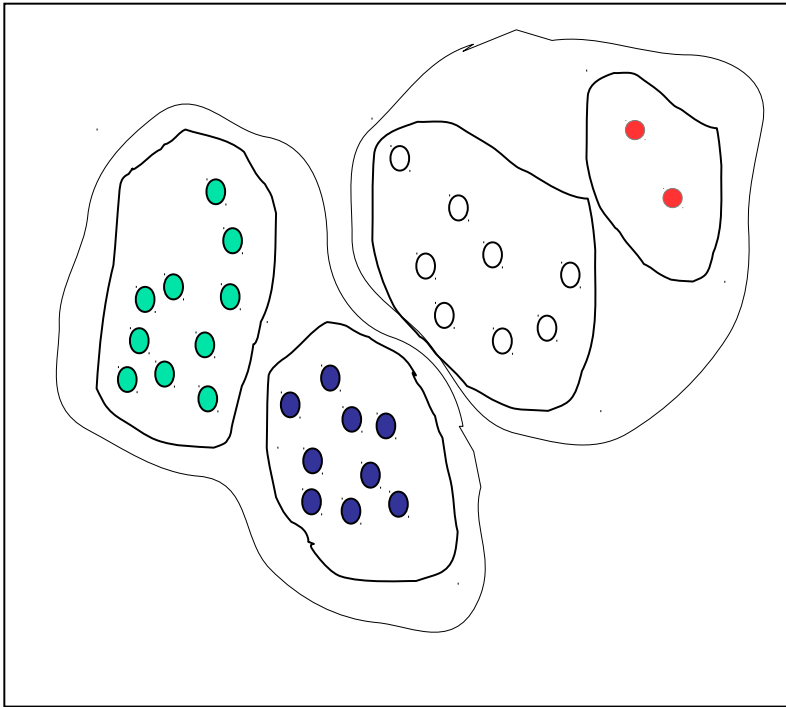


Cluster/Stratified Sample

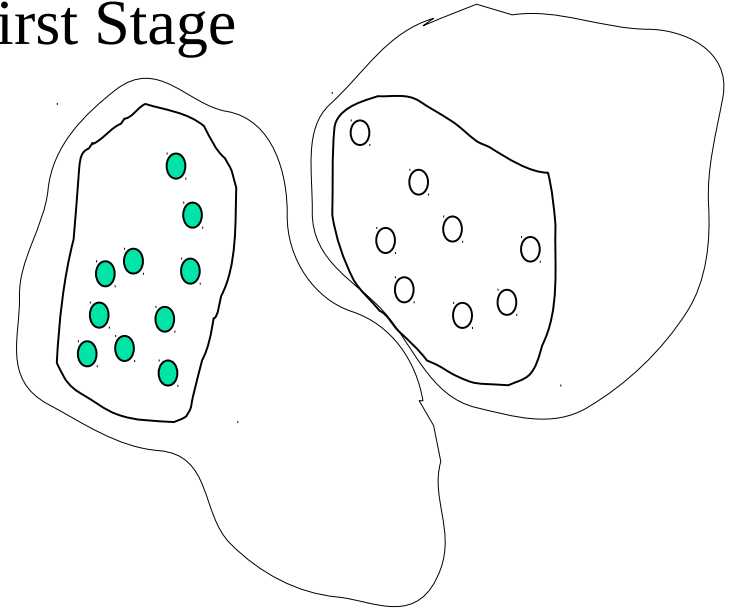


Sampling: Cluster or Stratified Sampling

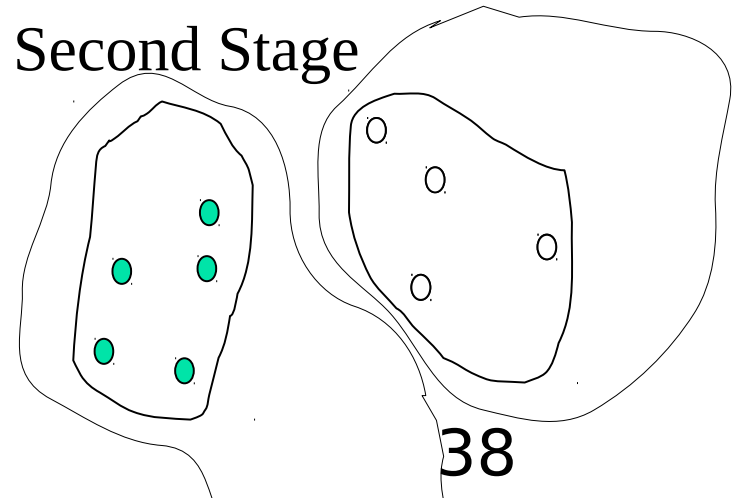
Raw Data



First Stage



Second Stage



Example: estimating sample size

- Generally speaking: we want to select n data objects from N
 - to estimate the value P of a parameter of (the probability distribution of) an attribute
 - with a value p computed (by a consistent estimator) on the sample only
 - up to a given precision δ with a certain probability $(1-\alpha)$
 - **Then** $\Pr(|p-P| \geq \delta) \leq \alpha$
- If we assume our estimator to be asymptotically normal, and the attribute p.d.f. to have variance σ^2
 - $z(x)$: value of the normal curve in x
 - $n_0 \geq z(\alpha/2)^2 \sigma^2 / \delta^2$; $n \geq n_0 / (1 + n_0 / N)$

Numerosity Reduction: change representation

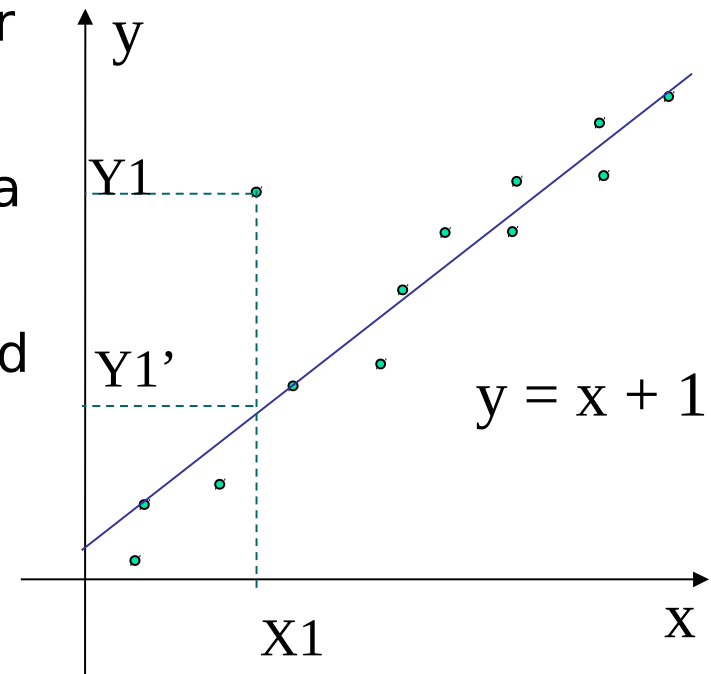
- Reduce data volume by choosing alternative, *smaller forms* of data representation
- **Parametric methods** (e.g., regression)
 - Assume the data fits some model, estimate model parameters, store only the parameters, and discard the data (except possible outliers)
 - Ex.: Log-linear models
- **Non-parametric** 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 “predictor” feature (variable) vector X
- **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



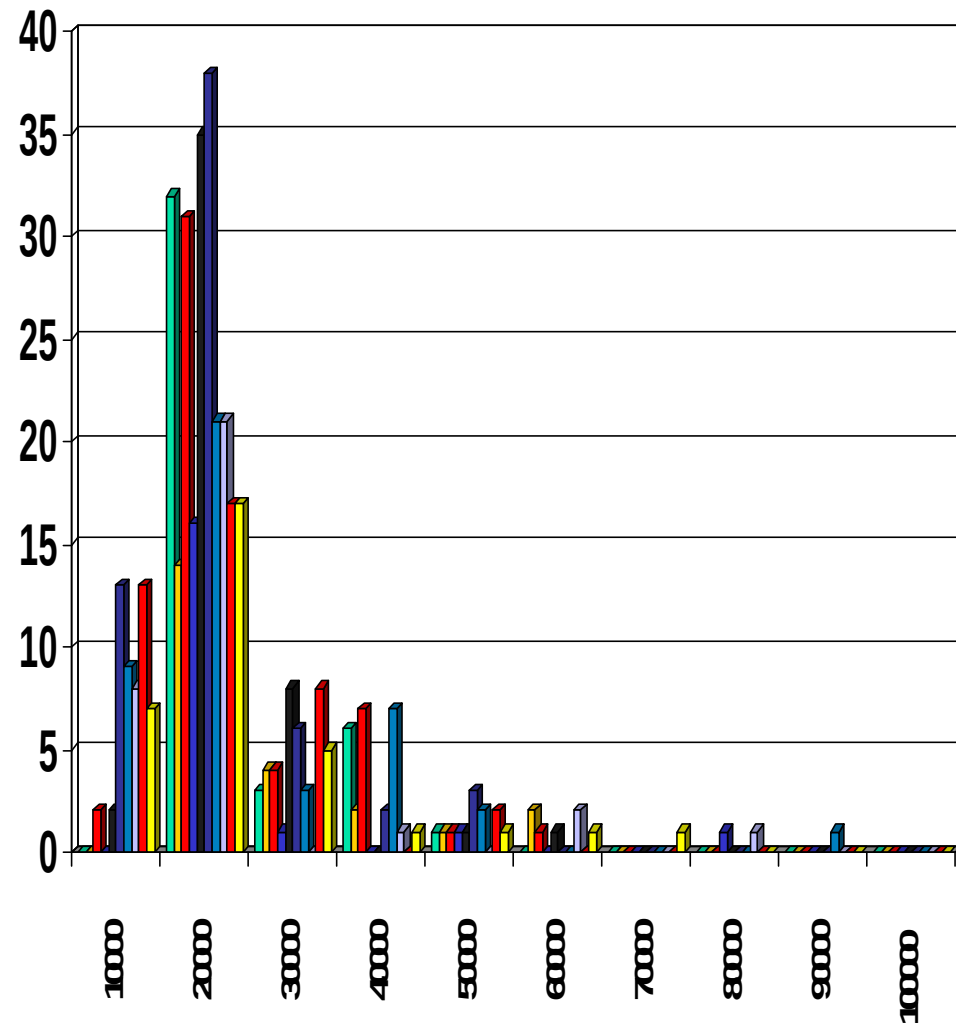
- Used for prediction (including forecasting of time-series data), inference, hypothesis testing, and modeling of causal relationships

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 $Y_1, Y_2, \dots, X_1, X_2, \dots$
- Multiple regression: $Y = b_0 + b_1 X_1 + b_2 X_2$
 - Many nonlinear functions can be transformed as above
- Log-linear models:
 - Approximate discrete multidimensional prob. 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)



Clustering

- Partition data set into clusters based on similarity, and store cluster representation (e.g., centroid and diameter) only
- Can be very effective if data is clustered but not if data is “smeared”
- Can have hierarchical clustering and be stored in multi-dimensional index tree structures
- There are many choices of clustering definitions and clustering algorithms
 - *We will have some dedicated lectures for clustering algorithms*

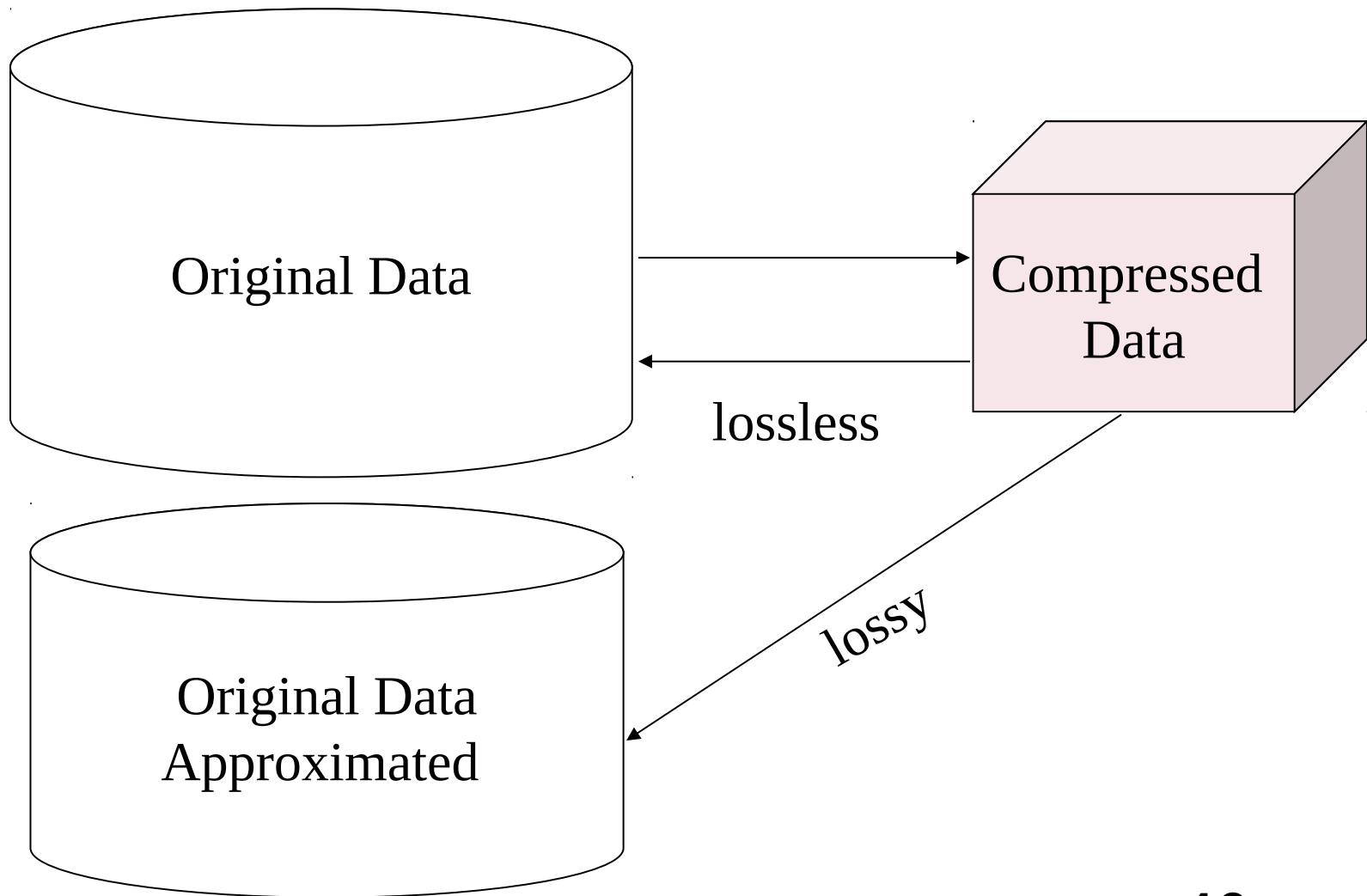
Data Reduction Strategies

- **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? Computational issues in big data!
- Data reduction strategies
 - **Numerosity reduction** (or simply “Data Reduction” → red. the number of data objects)
 - Sampling
 - Histograms, clustering
 - Regression and Log-Linear Models
 - Data cube aggregation
 - **Dimensionality reduction** (→ red. the number of attributes)
 - Principal Components Analysis (PCA)
 - Feature subset selection, feature creation
 - Transforms (Fourier, Wavelet, Whitening ...)
 - Data compression

Data Compression

- String compression
 - There are extensive theories and well-tuned algorithms
 - Typically lossless, but only limited manipulation is possible without expansion
- Audio/video compression
 - Typically lossy compression, with progressive refinement
 - Sometimes small fragments of signal can be reconstructed without reconstructing the whole
- Time sequence is not audio
 - Typically short and vary slowly with time

Data Compression




Data Reduction Strategies

- Data reduction strategies
 - **Numerosity reduction** (or simply “Data Reduction” → red. the number of data objects)
 - Sampling
 - Histograms, clustering
 - Regression and Log-Linear Models
 - Data cube aggregation
 - **Dimensionality reduction** (→ red. the number of attributes)
 - Principal Components Analysis (PCA)
 - Feature subset selection, feature creation
 - Transforms (Fourier, Wavelet, Whitening ...)

TOPIC OF THE NEXT SET OF LECTURES
 - **Data compression**

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

- A function that maps the entire set of values of a given attribute to a new set of replacement values s.t. each old value can be identified with one of the new values
- Methods
 - Smoothing: Remove noise from data
 - Attribute/feature construction
 - New attributes constructed from the given ones
 - Aggregation: Summarization, data cube construction
 - Normalization: Scaled to fall within a smaller, specified range (min-max normalization; z-score normalization; normalization by decimal scaling)
 - Discretization: Concept hierarchy climbing

Normalization

- **Min-max normalization:** to $[new_min_A, new_max_A]$

$$v' = \frac{v - min_A}{max_A - min_A} (new_max_A - new_min_A) + new_min_A$$

- Ex. Let income range \$12,000 to \$98,000 normalized to $[0.0, 1.0]$. Then \$73,600 is mapped to $\frac{73,600 - 12,000}{98,000 - 12,000} (1.0 - 0) + 0 = 0.716$

- **Z-score normalization** (μ : mean, σ : standard deviation):

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

- Ex. Let $\mu = 54,000$, $\sigma = 16,000$. Then $\frac{73,600 - 54,000}{16,000} = 1.225$

- **Normalization by decimal scaling**

$$v' = \frac{v}{10^j} \quad \text{Where } j \text{ is the smallest integer such that } \text{Max}(|v'|) < 1$$

Discretization

- Three types of attributes
 - Nominal—values from an unordered set, e.g. color
 - Ordinal—values from an ordered set, e.g. rank
 - Numeric—real numbers, e.g., integer or real numbers
- 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
 - Supervised vs. unsupervised
 - Split (top-down) vs. merge (bottom-up)
 - Discretization can be performed recursively on an attribute
 - Prepare for further analysis, e.g., classification

Data Discretization Methods

- Typical methods: All the methods can be applied recursively
 - **Binning**
 - Top-down split, unsupervised
 - **Histogram analysis**
 - Top-down split, unsupervised
 - **Clustering analysis** (unsupervised, top-down split or bottom-up merge)
 - **Decision-tree analysis** (supervised, top-down split)
 - **Correlation (e.g., χ^2) analysis** (unsupervised, bottom-up merge)

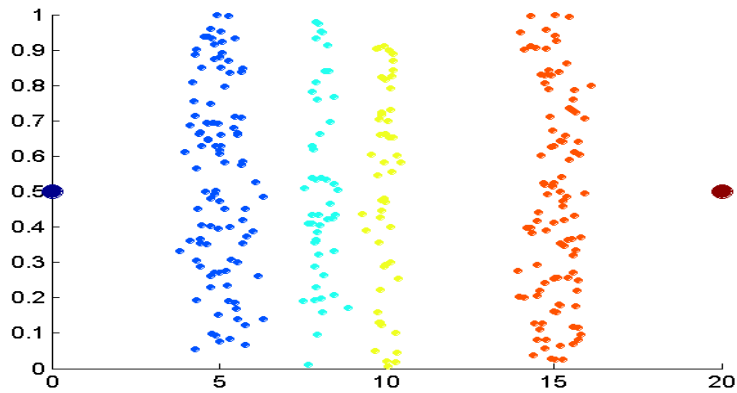
Simple Discretization: Binning

- **Equal-width** (distance) partitioning
 - 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, but outliers may dominate presentation
 - Skewed data is not handled well
- **Equal-depth** (frequency) partitioning
 - Divides the range into N intervals, each containing approximately same number of samples
 - Good data scaling
 - Managing categorical attributes can be tricky

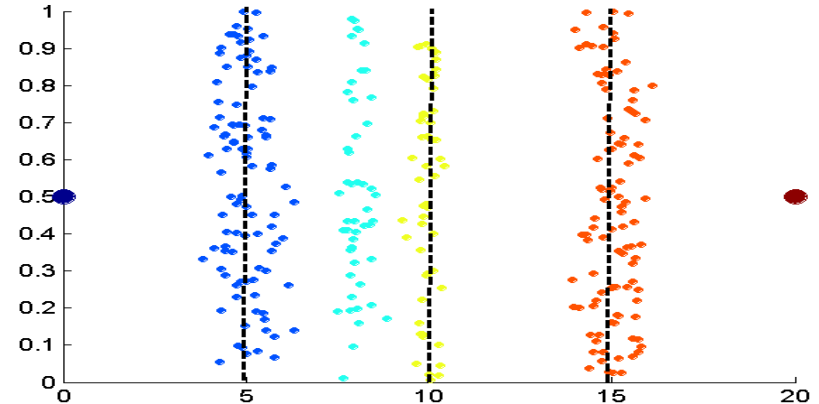
Binning Methods for Data Smoothing

- Sorted data for price (in dollars): 4, 8, 9, 15, 21, 21, 24, 25, 26, 28, 29, 34
- * Partition into equal-frequency (**equi-depth**) bins:
 - Bin 1: 4, 8, 9, 15
 - Bin 2: 21, 21, 24, 25
 - Bin 3: 26, 28, 29, 34
- * Smoothing by **bin means**:
 - Bin 1: 9, 9, 9, 9
 - Bin 2: 23, 23, 23, 23
 - Bin 3: 29, 29, 29, 29
- * Smoothing by **bin boundaries**:
 - Bin 1: 4, 4, 4, 15
 - Bin 2: 21, 21, 25, 25
 - Bin 3: 26, 26, 26, 34

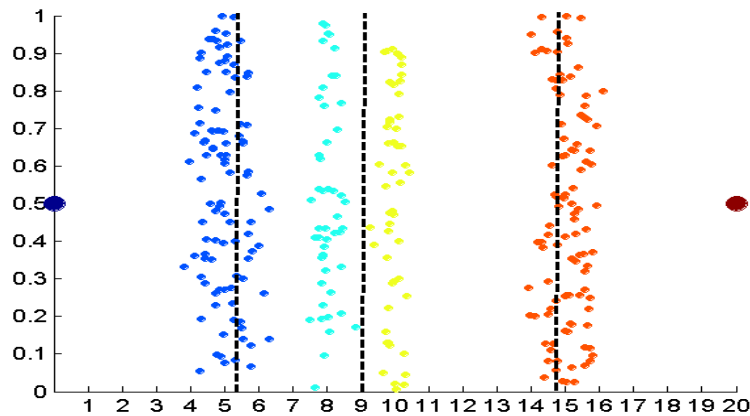
Discretization Without Using Class Labels (Binning vs. Clustering)



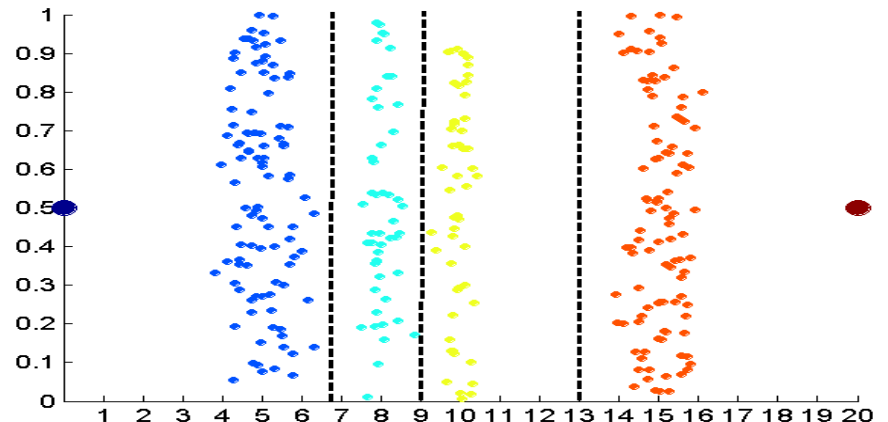
Data



Equal width (binning)



Equal frequency (binning)



K-means clustering leads to better results

Discretization by Classification & Correlation Analysis

- Classification (e.g., decision tree analysis)
 - Supervised: Given class labels, e.g., cancerous vs. benign
 - Using *entropy* to determine split point (discretization point); Top-down, recursive split

Details to be covered later in the course

- Correlation analysis (e.g., Chi-merge: χ^2 -based discretization)
 - Supervised: use class information
 - Bottom-up merge: find the best neighboring intervals (those having similar distributions of classes, i.e., low χ^2 values) to merge
 - Merge performed recursively, until a stopping condition

Concept Hierarchy Generation

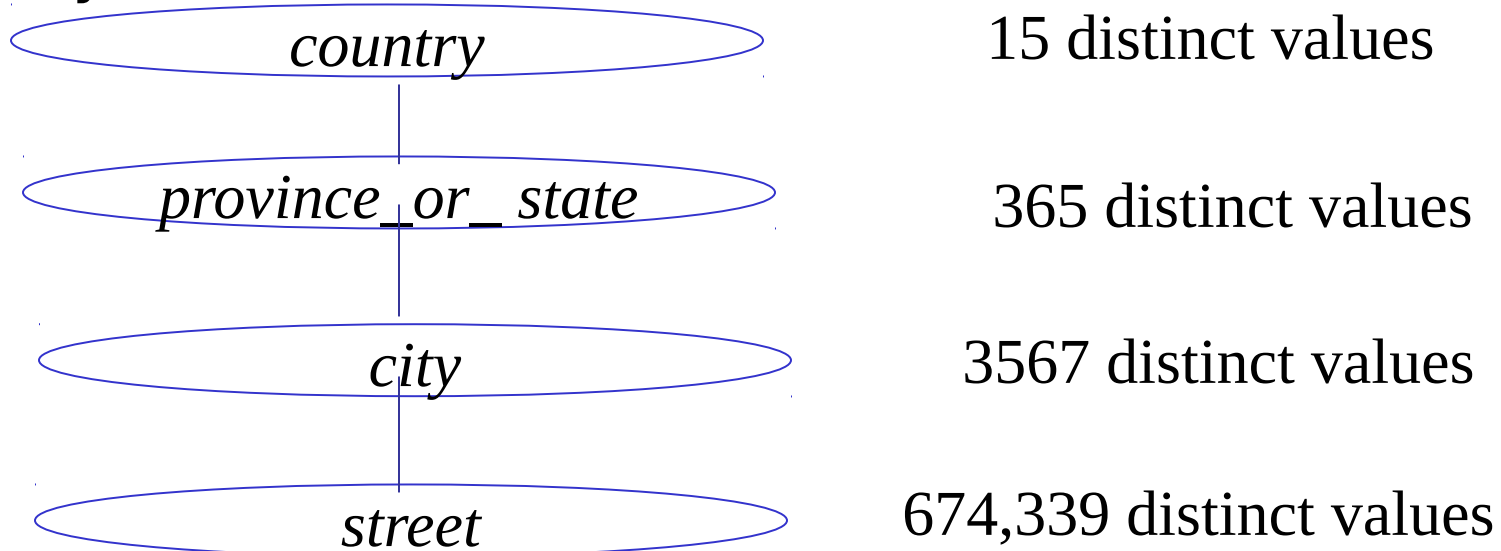
- **Concept hierarchy** organizes concepts (i.e., attribute values) hierarchically and is usually associated with each dimension in a data warehouse
- Concept hierarchies facilitate drilling and rolling in data warehouses to view data in multiple granularity
- Concept hierarchy formation: Recursively reduce the data by collecting and replacing low level concepts (such as numeric values for *age*) by higher level concepts (such as *youth, adult, or senior*)
- Concept hierarchies can be explicitly specified by domain experts and/or data warehouse designers
- Concept hierarchy can be automatically formed for both numeric and nominal data. For numeric data, use discretization methods shown.

Concept Hierarchy Generation for Nominal Data


- Specification of a partial/total ordering of attributes explicitly at the schema level by users or experts
 - $street < city < state < country$
- Specification of a hierarchy for a set of values by explicit data grouping
 - $\{Cremona, Lodi, Milano\} < Lombardia$
- Automatic generation of hierarchies (or attribute levels) by the analysis of the number of distinct values
 - E.g., for a set of attributes: $\{street, city, state, country\}$

Automatic Concept Hierarchy Generation

- Some hierarchies can be automatically generated based on the analysis of the number of distinct values per attribute in the data set
 - The attribute with the most distinct values is placed at the lowest level of the hierarchy
 - Exceptions, e.g., weekday, month, quarter, year



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 

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