Kuliah 3

Data Mining:

Beberapa Konsep dan Teknik

— Chapter 2 —

Jiawei Han
Department of Computer Science
University of Illinois at Urbana-Champaign
www.cs.uiuc.edu/~hanj

Pra-proses data

- Mengapa perlu pra-proses data?
- Pembersihan data
- Integrasi dan transformasi data
- Reduksi data
- Diskretisasi dan konsep hirarki
- Kesimpulan

Mengapa perlu pra-proses data?

- Sering ditemukan data yang kurang baik (*data dirty*)
  - tidak lengkap: tidak ada nilai atribut, atribut yg diperlukan, atau hanya berisi data agregat
    - contoh, occupation=" "
  - *noisy*: mengandung *errors* atau *outliers*
    - contoh., Salary="-10"
  - tidak konsisten:
    - contoh., Age="42" Birthday="03/07/1997"
    - contoh., Was rating "1,2,3", now rating "A, B, C"
    - contoh., perbedaan antar *records* yang duplikasi

Mengapa Data Dirty?

- data yang tidak lengkap dapat disebabkan oleh:
  - "Tidak memungkinkan" saat proses pengumpulan data
  - adanya pertimbangan yang berbeda saat data dikumpulkan dan dianalisis
  - masalah pada manusia/perangkat keras/perangkat
- *Noisy data* (nilai salah) dapat berasal dari
  - kesalahan instrumen saat mengumpulkan data
  - *Human/computer error* saat *entry* data
  - kesalahan saat transmisi data
- inkonsistensi data dapat berasal dari
  - sumber data yang berbeda
  - pelanggaran *Functional dependency* (contoh., modifikasi beberapa data yang terhubung)
- duplikasi

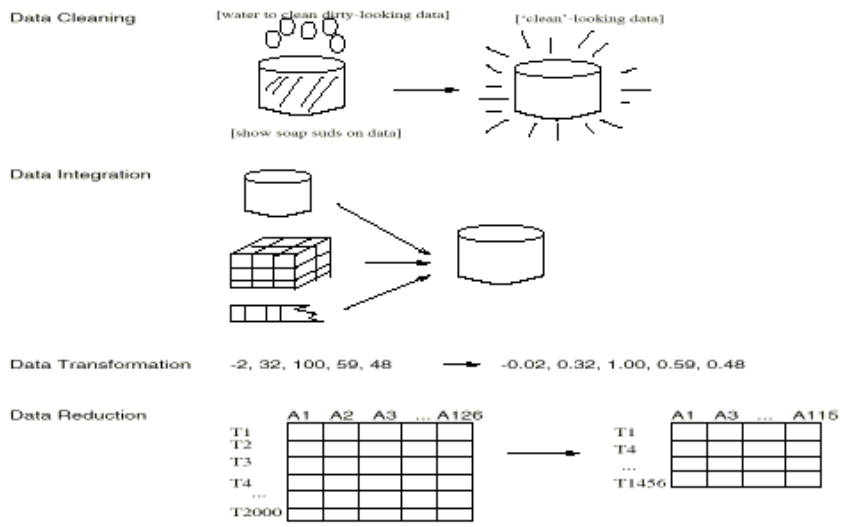
Mengapa pra-proses data penting?

- Tidak ada data yang berkualitas, tidak ada hasil proses *mining* yang berkualitas
  - keputusan yang berkualitas harus berdasarkan data yang berkualitas
    - contoh, duplikasi atau data yang tidak lengkap akan menyebabkan statistik yang salah atau menyesatkan.
  - *Data warehouse* memerlukan integrasi yang konsisten dari data yang berkualitas
- Ekstraksi data, pembersihan data dan transformasi data merupakan aktivitas utama dalam membangun data warehouse

Tugas utama dalam pra-proses data

- Pembersihan data (*Data cleaning*)
  - mengisi nilai yang kosong, *smooth noisy data*, identifikasi dan membuang *outliers*, dan mengatasi inkonsistensi
- Integrasi data (*Data integration*)
  - integrasi dari beberapa basisdata, *data cubes*, atau *files*
- Transformasi data (*Data transformation*)
  - normalisasi dan agregasi
- Reduksi data (*Data reduction*)
  - menghasilkan jumlah data yang lebih sedikit namun dapat memperoleh hasil analisis yang sama
- Diskretisasi data (*Data discretization*)
  - bagian dari reduksi data tetapi dengan kepentingan tertentu, khususnya untuk data numerik

Bentuk dari pra-proses data



Pra-proses data

- Mengapa perlu pra-proses data?
- Pembersihan data
- Integrasi dan transformasi data
- Reduksi data
- Diskretisasi dan konsep hirarki
- Kesimpulan

Pembersihan data

- Pentingnya pembersihan data
  - "Pembersihan data merupakan salah satu dari tiga masalah utama dalam *data warehousing*"—Ralph Kimball
  - "Pembersihan data merupakan masalah utama dalam data warehousing"—DCI survey
- Tugas dalam pembersihan data
  - mengisi nilai yang kosong
  - identifikasi *outliers* dan *smooth out noisy data*
  - memperbaiki inkonsistensi data
  - mengatasi redundansi yang disebabkan oleh integrasi data

Data yang hilang/kosong

- Data tidak selalu tersedia
  - contoh., banyak record yang tidak memiliki nilai pada beberapa atribut, seperti penghasilan pelanggan pada data penjualan
- Data hilang/kosong dapat disebabkan oleh
  - kerusakan peralatan
  - nilai atribut dihapus karena inkonsistensi dengan data lain yang telah tersimpan
  - data tidak di-input karena adanya kesalahpahaman
  - data tertentu mungkin dianggap tidak penting saat di-entri
  - tidak mendaftarkan perubahan data
- Data yang hilang mungkin perlu disimpulkan

Bagaimana cara menangani data yg hilang/kosong?

- abaikan record tsb: umumnya dilakukan ketika tidak ada label kelas (misalkan utk kasus klasifikasi – tidak efektif ketika persentase nilai yang kosong untuk setiap atribut sangat berbeda).
- mengisi nilai yang kosong/hilang secara manual: *tedious* + *infeasible*?
- mengisi secara otomatis menggunakan:
  - konstanta global: misal, "unknown", kelas baru?!
  - rataan nilai atribut
  - rataan nilai atribut dari seluruh sample pada kelas yang sama – lebih baik
  - nilai yang paling mungkin: dugaan menggunakan formula Bayes atau *decision tree*

Noisy Data

- *Noise*: ragam atau kesalahan acak pada variable yang diukur
- kesalahan pada nilai atribut dapat disebabkan oleh
  - kesalahan pad instrument pengumpulan data
  - masalah saat *entry* data
  - masalah saat transmisi data
  - keterbatasan teknologi
  - inkonsistensi pada penamaan
- masalah lainnya yang memerlukan pembersihan data
  - duplikasi data
  - data tidak lengkap
  - inkonsistensi data

Bagaimana cara menangani *Noisy Data*?

- *Binning*
  - urutkan data dan partisi ke dalam (frekuensi yg sama) *bins*
  - selanjutnya, *smooth by bin means, smooth by bin median, smooth by bin boundaries*, etc.
- Regresi
  - *smooth by fitting the data into regression functions*
- *Clustering*
  - mendeteksi dan membuang *outliers*
- Kombinasi hasil pemeriksaan komputer dan manusia
  - deteksi nilai yang mencurigakan dan validasi oleh manusia (misal., berkaitan dengan *outliers*)

Metode Diskretisasi Sederhana: *Binning*

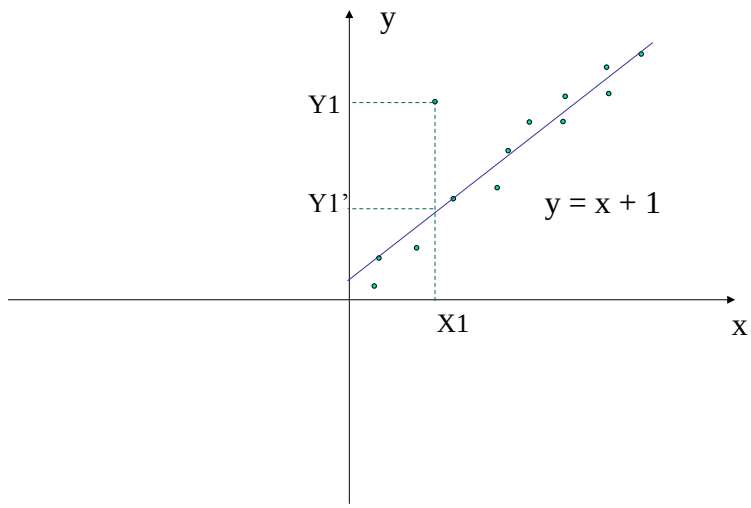
- Equal-width (*distance*) *partitioning*
  - membagi range nilai menjadi *N* intervals dengan ukuran yang sama: *uniform grid*
  - jika A dan B merupakan nilai atribut terkecil dan terbesar, lebar interval adalah: $W = (B - A)/N$.
  - Pendekatan paling mudah, namun *outliers* mungkin mendominasi penyajian
  - *Skewed data* tidak ditangani dengan baik
- Equal-depth (*frequency*) *partitioning*
  - membagi range nilai menjadi *N* intervals, setiap interval berisi kira-kira jumlah sampel yang sama
  - penskalaan data yang bagus
  - Mengelola atribut kategorik bisa jadi rumit

Metode Binning untuk *Smoothing* Data

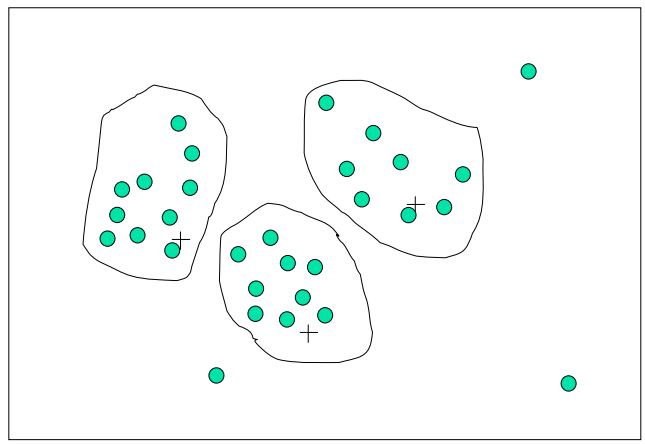
- Urutkan data (harga dalam dollars): 4, 8, 9, 15, 21, 21, 24, 25, 26, 28, 29, 34

* Partisi menjadi equal-frequency (equi-depth) *bins*:
  - Bin 1: 4, 8, 9, 15
  - Bin 2: 21, 21, 24, 25
  - Bin 3: 26, 28, 29, 34
* *Smoothing* dengan rataan di setiap *bin*:
  - Bin 1: 9, 9, 9, 9
  - Bin 2: 23, 23, 23, 23
  - Bin 3: 29, 29, 29, 29
* *Smoothing* dengan nilai batas *bin*:
  - Bin 1: 4, 4, 4, 15
  - Bin 2: 21, 21, 25, 25
  - Bin 3: 26, 26, 26, 34

Regresi



Analisis *cluster*



Pra-proses data

- Mengapa perlu pra-proses data?
- Pembersihan data
- Integrasi dan transformasi data
- Reduksi data
- Diskretisasi dan konsep hirarki
- Kesimpulan

Integrasi Data

- Integrasi data:
  - menggabungkan data dari beberapa sumber menjadi data yang koheren
- Skema integrasi: misal., A.cust-id ≡ B.cust-#
  - integrasi metadata dari sumber yang berbeda
- Masalah identifikasi entitas:
  - mengenali entitas di dunia nyata dari beberapa sumber data, contoh., Bill Clinton = William Clinton
- mendeteksi dan menyelesaikan *data value conflicts*
  - dari sumber yang berbeda, entitas yang sama memiliki nilai atribut yang berbeda
  - penyebab: representasi yang berbeda, perbedaan skala seperti *metric* vs. British *units*

Menangani redundansi pada integrasi data

- redundansi data sering terjadi saat proses integrasi dari beberapa basis data
  - *Object identification*: atribut atau objek yang sama memiliki nama yang berbeda di basis data yang berbeda
  - *Derivable data:* suatu atribut merupakan atribut "turunan" dari tabel lainnya, contoh., *annual revenue*
- Redundansi atribut dapat dideteksi dengan analisis korelasi (*correlation analysis)*
- proses integrasi data dari beberapa sumber yang dilakukan secara hati-hati dapat mengurangi/menghindari redundansi, inkonsistensi dan meningkatkan kecepatan dan kualitas saat mining

Analisis Korelasi (Data Numerik)

- Koefisien korelasi (juga dikenal sbg Pearson's product moment coefficient)

$$r_{A,B} = \frac{\sum (A-\overline{A})(B-\overline{B})}{(n-1)\sigma_A \sigma_B} = \frac{\sum (AB) - n\overline{A}\overline{B}}{(n-1)\sigma_A \sigma_B}$$

dimana ***n*** adalah jumlah record, $\overline{A}$ dan $\overline{B}$ merupakan rataan dari A and B, $\boldsymbol{\sigma_A}$ dan $\boldsymbol{\sigma_B}$ adalah standar deviasi dari A and B, dan Σ(AB) adalah jumlah dari cross-product AB.

- Jika $r_{A,B} > 0$, A dan B berkorelasi positif (*A's values increase as B's*). Semakin tinggi nilai korelasi, semakin kuat korelasi A dan B.
- $r_{A,B} = 0$: *independent*; $r_{A,B} < 0$: berkorelasi negatif

Analisis Korelasi (Categorical Data)

- X^2 (chi-square) test

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

- Semakin besar nilai X^2, *the more likely the variables are related*
- *Cell* yang paling berkontribusi pada nilai X^2 adalah *cell* yg memiliki nilai aktual paling berbeda dari *expected*
- *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

Contoh Chi-Square

	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

- Perhitungan X^2 (chi-square)
  - (bilangan di dalam () merupakan *expected counts yang* dihitung berdasarkan distribusi data pada dua kategori)

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

- Hal ini menunjukkan: **like_science_fiction** dan **play_chess** *are correlated in the group*

Transformasi Data

- *Smoothing*: membuang *noise* dari data
- *Aggregation*: summarization, data cube construction
- Generalization: concept hierarchy climbing
- Normalization: scaled to fall within a small, specified range
  - min-max normalization
  - z-score normalization
  - normalization by decimal scaling
- Attribute/feature construction
  - New attributes constructed from the given ones

Transformasi Data: Normalisasi

- Min-max normalization: to [new_min$_A$, new_max$_A$]

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

  - Misalkan range pendapatan \$12,000 s.d. \$98,000 akan di-normalisasi ke [0.0, 1.0]. Maka, \$73,000 akan dipetakan ke: $\frac{73{,}600 - 12{,}000}{98{,}000 - 12{,}000}(1.0 - 0) + 0 = 0.716$
- Z-score normalization (μ: mean, σ: standard deviation):

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

  - Ex. Let μ = 54,000, σ = 16,000. Then $\frac{73{,}600 - 54{,}000}{16{,}000} = 1.225$
- Normalization by decimal scaling

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

Pra-proses data

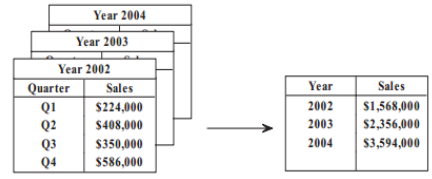
- Mengapa perlu pra-proses data?
- Pembersihan data
- Integrasi dan transformasi data
- Reduksi data
- Diskretisasi dan konsep hirarki
- Kesimpulan

Strategi untuk mereduksi data

- Mengapa reduksi data?
  - Basis data/data warehouse menyimpan terabytes of data
  - analisis/*mining* data memerlukan waktu proses yang lama jika diterapkan pada seluruh data
- Reduksi data
  - memperoleh representasi data yang lebih kecil namun dapat menghasilkan hasil analisis yang (hampir) sama
- Strategi reduksi data
  - Data cube aggregation:
  - **Dimensionality reduction** — misal., membuang atribut yang tidak penting
  - Data Compression
  - Numerosity reduction — misal., fit data into models
  - Discretization and concept hierarchy generation

Data Cube Aggregation

- Level terendah dari *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



Seleksi Subset Atribut

- Seleksi fitur (pemilihan subset atribut):
  - Select a minimum set of features such that the probability distribution of different classes given the values for those features is as close as possible to the original distribution given the values of all features
  - reduce # of patterns in the patterns, mudah dimengerti
- Heuristic methods (due to exponential # of choices):
  - Step-wise forward selection
  - Step-wise backward elimination
  - Combining forward selection and backward elimination
  - Decision-tree induction

Seleksi Subset Atribut

Forward selection

Initial attribute set:
{A1,A2,A3,A4, A5, A6}

Initial reduced set:
{}
=> {A1}
=> {A1, A4}
=> Reduced attribute set:
{ A1, A4, A6}

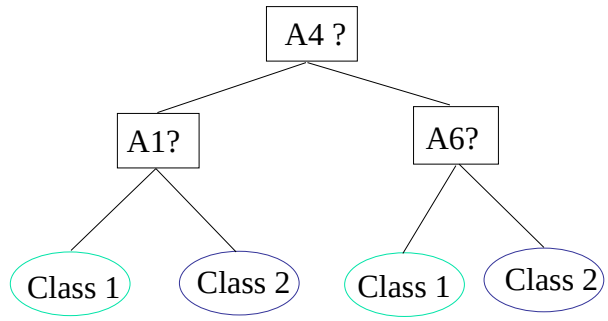
Backward elimination

Initial attribute set:
{A1,A2,A3,A4, A5, A6}

=>{A1,A3,A4, A5, A6}
=>{A1,A4, A5, A6}
=> Reduced attribute set:
{A1,A4, A6}

Example of Decision Tree Induction

Initial attribute set:
{A1, A2, A3, A4, A5, A6}

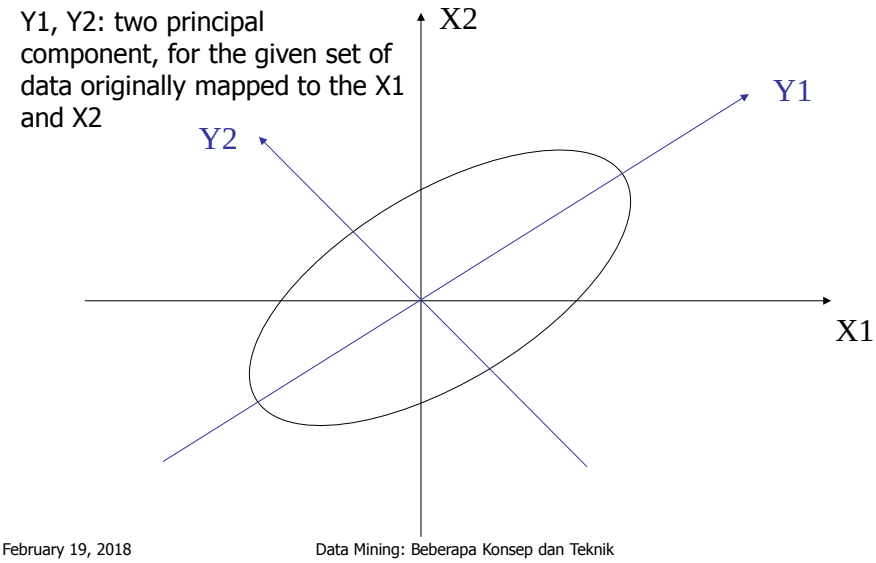


------> Reduced attribute set: {A1, A4, A6}

Dimensionality Reduction: Principal Component Analysis (PCA)

- Diberikan *N* data vectors dengan *n*-dimensions, tentukan $k \leq n$ orthogonal vectors (*principal components*) terbaik yang dapat digunakan untuk merepresentasikan data
- Tahapan:
  - Normalisasi input data: Each attribute falls within the same range
  - Hitung *k* orthonormal (unit) vectors, i.e., *principal components*
  - Setiap input data (vector) merupakan kombinasi linear dari *k* principal component vectors
  - Principal components diurutkan mulai dari yg terbesar ke yg terkecil
  - 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
- Hanya berlaku untuk data numeric data
- Digunakan ketika dimensi data besar

Principal Component Analysis



Numerosity Reduction

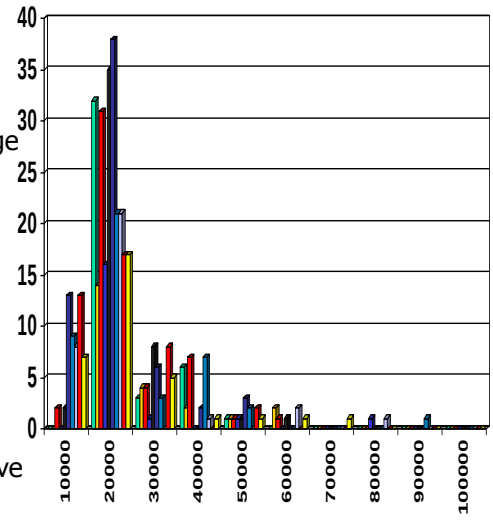
- Reduce data volume by choosing alternative, smaller forms of data representation
- Parametric methods
  - Assume the data fits some model, estimate model parameters, store only the parameters, and discard the data (except possible outliers)
  - Example: Log-linear models—obtain value at a point in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

Metode Reduksi Data (1): Regresi dan Model Log-Linear

- Linear regression: Data are 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.
  - It can be used to estimate the probability of each point in a multidimensional space for a set of discretized attributes, based on a smaller subset of dimensional combinations

Metode Reduksi Data (2) : Histograms

- Divide data into buckets and store average (sum) for each bucket
- Partitioning rules:
  - Equal-width: equal bucket range
  - Equal-frequency (or equal-depth)
  - V-optimal: with the least *histogram variance* (weighted sum of the original values that each bucket represents)
  - MaxDiff: set bucket boundary between each pair for pairs have the β–1 largest differences



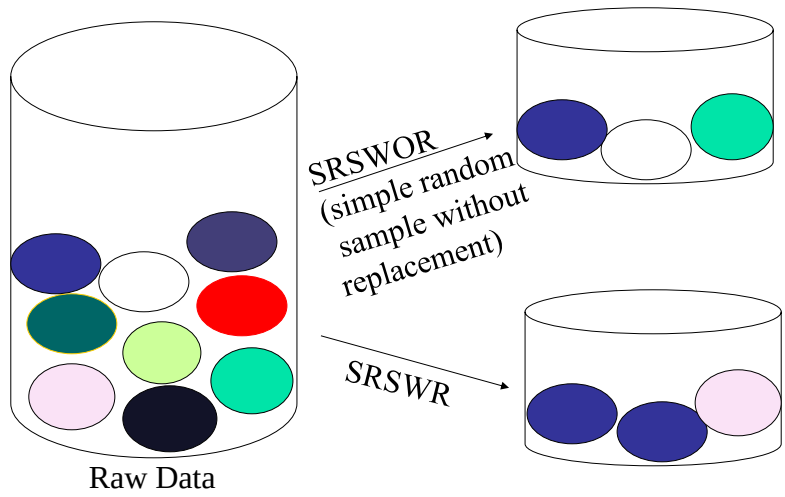
Metode Reduksi Data (3): *Clustering*

- 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
- Cluster analysis will be studied in depth in Chapter 7

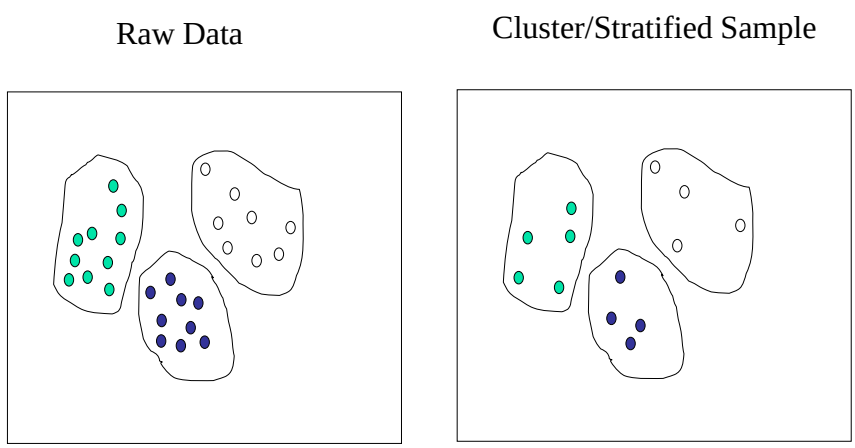
Metode Reduksi Data (4): *Sampling*

- 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
- Choose a representative subset of the data
  - Simple random sampling may have very poor performance in the presence of skew
- Develop adaptive sampling methods
  - Stratified sampling:
    - Approximate the percentage of each class (or subpopulation of interest) in the overall database
    - Used in conjunction with skewed data
- Note: Sampling may not reduce database I/Os (page at a time)

Sampling: dengan atau tanpa penggantian



Sampling: Cluster atau Stratified Sampling



Pra-proses data

- Mengapa perlu pra-proses data?
- Pembersihan data
- Integrasi dan transformasi data
- Reduksi data
- Diskretisasi dan konsep hirarki
- Kesimpulan

Diskretisasi

- Tiga tipe atribut:
  - Nominal — values from an unordered set, e.g., color, profession
  - Ordinal — values from an ordered set, e.g., military or academic rank
  - Continuous — real numbers, e.g., integer or real numbers
- Discretization:
  - Divide the range of a continuous attribute into intervals
  - Some classification algorithms only accept categorical attributes.
  - Reduce data size by discretization
  - Prepare for further analysis

Discretization and Concept Hierarchy

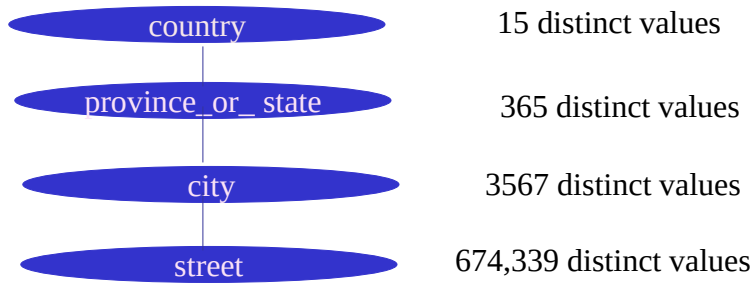
- Discretization
  - Reduce the number of values for a given continuous attribute by dividing the range of the attribute into intervals
  - Interval labels can then be used to replace actual data values
  - Supervised vs. unsupervised
  - Split (top-down) vs. merge (bottom-up)
  - Discretization can be performed recursively on an attribute
- 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 young, middle-aged, or senior)

Concept Hierarchy Generation for Categorical 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
  - {Urbana, Champaign, Chicago} < Illinois
- Specification of only a partial set of attributes
  - E.g., only street < city, not others
- 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



Pra-proses data

- Mengapa perlu pra-proses data?
- Pembersihan data
- Integrasi dan transformasi data
- Reduksi data
- Diskretisasi dan konsep hirarki
- Kesimpulan

Kesimpulan

- Data preparation or preprocessing is a big issue for both data warehousing and data mining
- Discriptive data summarization is need for quality data preprocessing
- Data preparation includes
  - Data cleaning and data integration
  - Data reduction and feature selection
  - Discretization
- A lot a methods have been developed but data preprocessing still an active area of research

Referensi

- D. P. Ballou and G. K. Tayi. Enhancing data quality in data warehouse environments. Communications of ACM, 42:73-78, 1999
- T. Dasu and T. Johnson. Exploratory Data Mining and Data Cleaning. John Wiley & Sons, 2003
- T. Dasu, T. Johnson, S. Muthukrishnan, V. Shkapenyuk. Mining Database Structure; Or, How to Build a Data Quality Browser. SIGMOD'02.
- H.V. Jagadish et al., Special Issue on Data Reduction Techniques. Bulletin of the Technical Committee on Data Engineering, 20(4), December 1997
- D. Pyle. Data Preparation for Data Mining. Morgan Kaufmann, 1999
- E. Rahm and H. H. Do. Data Cleaning: Problems and Current Approaches. *IEEE Bulletin of the Technical Committee on Data Engineering. Vol.23, No.4*
- V. Raman and J. Hellerstein. Potters Wheel: An Interactive Framework for Data Cleaning and Transformation, VLDB'2001
- T. Redman. Data Quality: Management and Technology. Bantam Books, 1992
- Y. Wand and R. Wang. Anchoring data quality dimensions ontological foundations. Communications of ACM, 39:86-95, 1996
- 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