CS 591.03
Introduction to Data Mining
Instructor: Abdullah Mueen

LECTURE 3: DATA TRANSFORMATION AND DIMENSIONALITY REDUCTION
Chapter 3: Data Preprocessing

Data Preprocessing: An Overview

- Data Quality
- Major Tasks in Data Preprocessing

Data Cleaning

Data Integration

Data Reduction

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
Chapter 3: Data Preprocessing

Data Preprocessing: An Overview

- Data Quality
- Major Tasks in Data Preprocessing

Data Cleaning

Data Integration

Data Reduction

Summary
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)
Chapter 3: Data Preprocessing

Data Preprocessing: An Overview

- Data Quality
- Major Tasks in Data Preprocessing

Data Cleaning

Data Integration

Data Reduction

Summary
Chapter 3: Data Preprocessing

Data Preprocessing: An Overview

◦ Data Quality

◦ Major Tasks in Data Preprocessing

Data Cleaning

Data Integration

Data Reduction

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 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**
What Is Wavelet Transform?

Decomposes a signal into different frequency subbands
  ◦ Applicable to n-dimensional signals

Data are transformed to preserve relative distance between objects at different levels of resolution

Allow natural clusters to become more distinguishable

Used for image compression
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

<table>
<thead>
<tr>
<th>Resolution</th>
<th>Averages</th>
<th>Detail Coefficients</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>[2, 2, 0, 2, 3, 5, 4, 4]]</td>
<td>[0, -1, -1, 0]</td>
</tr>
<tr>
<td>4</td>
<td>[2, 1, 4, 4]</td>
<td>[\frac{1}{2}, 0]</td>
</tr>
<tr>
<td>2</td>
<td>[1\frac{1}{2}, 4]</td>
<td>[-1\frac{1}{4}]</td>
</tr>
<tr>
<td>1</td>
<td>[2\frac{3}{4}]</td>
<td></td>
</tr>
</tbody>
</table>
Example Steps for Haar transform in MATLAB

\[ N = [\sqrt{8}, \sqrt{8}, 2, 2, \sqrt{2}, \sqrt{2}, \sqrt{2}, \sqrt{2}]; \]

\[ H_8 = [1 1 1 1 1 1 1 1; 1 1 1 -1 -1 -1 -1 -1; 1 1 -1 -1 0 0 0 0; 0 0 0 0 1 -1 -1 -1; 1 -1 0 0 0 0 0 0; 0 0 1 -1 0 0 0 0; 0 0 0 1 -1 0 0 0; 0 0 0 0 0 1 -1 -1]; \]

\[ X = x*H_8'./N; \]

\[ Y = y*H_8'./N; \]

\[ \text{sum}((X-Y).^2) \]

\[ \text{sum}((x-y).^2) \]
Why Wavelet Transform?

Use hat-shape filters
- Emphasize region where points cluster
- Suppress weaker information in their boundaries

Effective removal of outliers
- Insensitive to noise, insensitive to input order

Multi-resolution
- Detect arbitrary shaped clusters at different scales

Efficient
- Complexity O(N)

Only applicable to low dimensional data
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.
Example: Steps for PCA in Matlab

```matlab
X(1,:) = round(sin(0:2*pi+1)*1000)+ rand(1,8)*100
X(2,:) = round(sin(0:2*pi+1)*1000)+ rand(1,8)*100
X(3,:) = round(sin(0:2*pi+1)*1000)+ rand(1,8)*100
X(4,:) = round(sin(0:2*pi+1)*1000)+ rand(1,8)*100
X(5,:) = round(sin(0:5/2:(5*pi+4))*200)+ rand(1,8)*100
X(6,:) = round(sin(0:5/2:(5*pi+4))*200)+ rand(1,8)*100
X(7,:) = round(sin(0:5/2:(5*pi+4))*200)+ rand(1,8)*100
X(8,:) = round(sin(0:5/2:(5*pi+4))*200)+ rand(1,8)*100
plot(X')
for i = 1:8
    X(:,i) = zNorm(X(:,i)); --Note the change from the earlier version of the slide where it was X(i,:) = zNorm(X(i,:));
end
C = (X'*X) – Note the change from the earlier version of the slide where it was C= (X*X');
[V,D] = eig(C);
V(:,7)
X*V(:,7)
plot(X*V(:,7),X*V(:,8),'*')
```

```matlab
function y = zNorm(x)

    y = (x-mean(x))./std(x,1);
```

Principal Component Analysis (Steps)

Given $N$ data vectors from $n$-dimensions, find $k \leq n$ 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 on “Advanced Classification”)
  - 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)
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 10
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)
Types of Sampling

**Simple random sampling**
- There is an equal probability of selecting any particular item

**Sampling without replacement**
- Once an object is selected, it is removed from the population

**Sampling with replacement**
- A selected object is 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
Sampling: Cluster or Stratified Sampling

Raw Data

Cluster/Stratified Sample
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

Dimensionality and numerosity reduction may also be considered as forms of data compression
Data Compression

Original Data → Compressed Data

lossless

Original Data Approximated

lossy