Data Mining: Data Reduction

Prof.Dr. Songül Varlı

Department of Computer Engineering

Yildiz Technical University

svarli@yildiz.edu.tr

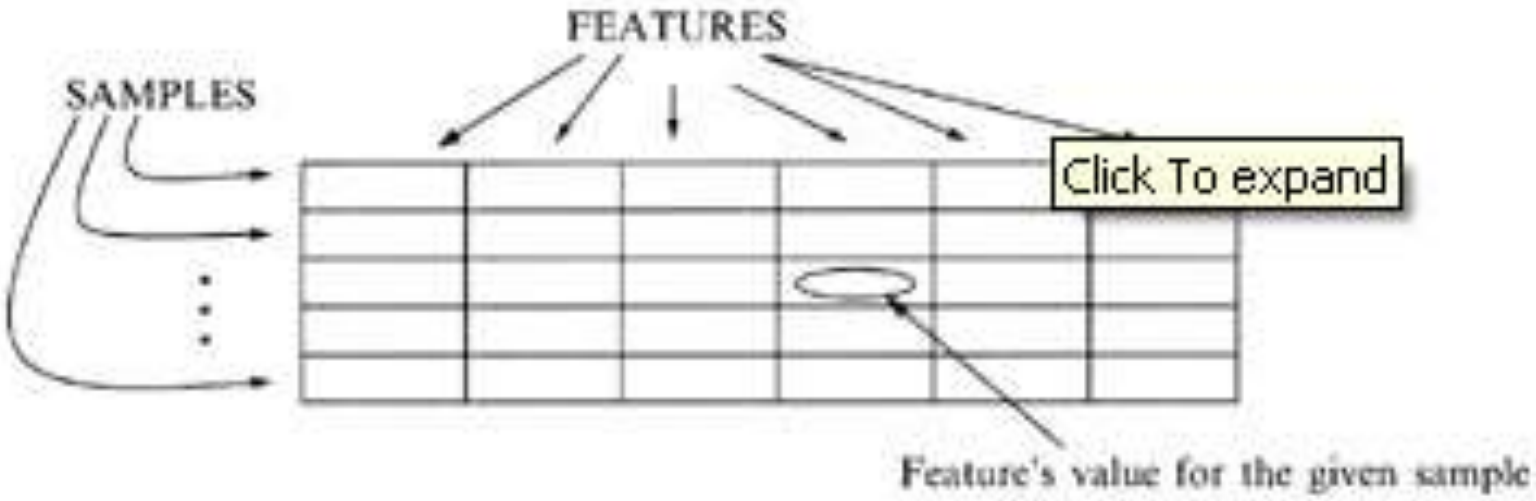
DATA REDUCTION

- Chapter Objectives
  - Identify the difference in dimensionality based on features, cases, and reduction of value techniques.
  - Explain the advantages of data reduction in the preprocessing phase of a data mining process.
  - Understand the basic principles of feature-selection and feature-composition tasks using corresponding statistical methods.
  - Apply and compare entropy-based technique and principal component analysis for feature ranking.
  - Understand the basic principles and implement ChiMerge and Bin-based techniques for reduction of discrete values.
  - Distinguish approaches in cases where reduction is based on incremental and average samples.

Dimensions of Large Data Sets

- In practice, the number of features can be as many as several hundreds. If we have only a few hundred samples for analysis, dimensionality reduction is required in order for any reliable model to be mined or to be of any practical use.
- On the other hand, data overload, because of high dimensionality, can make some data-mining algorithms nonapplicable, and the only solution is again a reduction of data dimensions.
- The three main dimensions of preprocessed data sets, usually represented in the form of flat files, are *columns* (features), *rows* (cases or samples), and *valves* of the features.

Dimensions of Large Data Sets



- The three main dimensions of preprocessed data sets, usually represented in the form of flat files, are *columns* (features), *rows* (cases or samples), and *values* of the features.
- Therefore, the three basic operations in a data-reduction process are delete a column, delete a row, and reduce the number of values in a column (smooth a feature).

Dimensions of Large Data Sets

- For example, if samples in a data set have two features, person-height and person-weight, it is possible for some applications in the medical domain to replace these two features with only one, body-mass-index, which is proportional to the quotient of the initial two features.
- Final reduction of data does not reduce the quality of results; in some applications, the results of data mining are even improved.

Dimensions of Large Data Sets

Performing standard data-reduction operations (deleting rows, columns, or values) as a preparation for data mining, we need to know what we gain and/or lose with these activities. The overall comparison involves the following parameters for analysis:

- ***Computing time*** - Simpler data, a result of the data-reduction process, can hopefully lead to a reduction in the time taken for data mining.
- ***Predictive/descriptive accuracy*** - This is the dominant measure for most data mining models since it measures how well the data is summarized and generalized into the model.
- ***Representation of the data-mining model*** - The simplicity of representation, obtained usually with data reduction, often implies that a model can be better understood. The simplicity of the induced model and other results depends on its representation. Therefore, if the simplicity of representation improves, a relatively small decrease in accuracy may be tolerable.

Dimensions of Large Data Sets

- It would be ideal if we could achieve reduced time, improved accuracy, and simplified representation at the same time, using dimensionality reduction. More often, however, we gain in some and lose in others, and balance between them according to the application at hand.

Feature Reduction

- We want to choose features that are relevant to our data-mining application in order to achieve maximum performance with the minimum measurement and processing effort.
- Two standard tasks are associated with producing a reduced set of features, and they are classified as
- ***Feature selection*** - Based on the knowledge of the application domain and the goals of the mining effort, the human analyst may select a subset of the features found in the initial data set. The process of feature selection can be manual or supported by some automated procedures.
- ***Feature composition*** - There are transformations of data that can have a surprisingly strong impact on the results of data-mining methods. In this sense, the composition of features is a greater determining factor in the quality of data-mining results than the specific mining technique.

1-Feature Selection

The objective of **feature selection** is to find a subset of features with data-mining performances comparable to the full set of features.

- One possible technique for feature selection is based on comparison of ***means*** and ***variances***. To summarize the key characteristics of the distribution of values for a given feature, it is necessary to compute the mean value and the corresponding variance.
- The main weakness in this approach is that the distribution for the feature is not known. If it is assumed to be a normal curve, the statistics can work out very well, but this may in fact be a poor assumption.
- In general, if one feature describes different classes of entities, samples of different classes can be examined.
- The means of feature values are normalized by its variances and then compared.
  - If the means are far apart, interest in a feature increases; it has potential, in terms of its use in distinguishing between two classes.
  - If the means are indistinguishable, interest wanes in that feature.
- It is a heuristic, nonoptimal approach to feature selection, but it is consistent with practical experience in many data-mining applications in the triage of features.

1-Feature Selection

- Next equations formalize the test, where A and B are sets of feature values measured for two different classes, and n1 and n2 are the corresponding number of samples:

$$SE(A - B) = \sqrt{(var(A)/n_1 + var(B)/n_2)}$$

$$\text{TEST: } |mean(A) - mean(B)|/SE(A - B) > \text{threshold-value}$$

1-Feature Selection: Example

Table 3.1: Dataset with three features

X	Y	C
0.3	0.7	A
0.2	0.9	B
0.6	0.6	A
0.5	0.5	A
0.7	0.7	B
0.4	0.9	B

- A simple data set is given in Table above with two input features X and Y, and an additional feature C that classifies samples into two classes A and B. It is necessary to decide whether the features X and Y are candidates for reduction or not. Suppose that the threshold value of the applied test is 0.5.

1-Feature Selection: Example

X_A={0.3, 0.6, 0.5} mean(X_A)=0.4667 Var(X_A)=0.0233

X_B={0.2, 0.7, 0.4} mean(X_B)=0.4333 Var(X_B)=0.6333

Y_A={0.7, 0.6, 0.5} mean(Y_A)=0.6 Var(Y_A)=0.01

Y_B={0.9, 0.7, 0.9} mean(Y_B)=0.833 Var(Y_B)=0.0133

1-Feature Selection: Example

$$SE(X_A - X_B) = \sqrt{(var(X_A)/n_1 + var(X_B)/n_2)} = \sqrt{(0.0233/3 + 0.6333/3)} = 0.4678$$

$$SE(Y_A - Y_B) = \sqrt{(var(Y_A)/n_1 + var(Y_B)/n_2)} = \sqrt{(0.01/3 + 0.0133/3)} = 0.0875$$

$$|mean(X_A) - mean(X_B)| / SE(X_A - X_B) = |0.4667 - 0.4333| / 0.4678 = 0.0735 < 0.5$$

$$|mean(Y_A) - mean(Y_B)| / SE(Y_A - Y_B) = |0.6 - 0.8333| / 0.0875 = 2.6667 > 0.5$$

- This analysis shows that X is a candidate for reduction because its mean values are close and, therefore, the final test is below the threshold value. On the other hand, the test for feature Y is significantly above the threshold value; this feature is not a candidate for reduction because it has the potential to be a distinguishing feature between two classes.

2-Feature Composition

- An alternative view of this process is to reduce feature dimensions by merging features instead of by deleting them.
- This process results in a new set of fewer features with totally new values.
- One well-known approach is merging by *principal components*. The features are examined collectively, merged, and transformed into a new set of features that, it is hoped, will retain the original information content in a reduced form. The basic transformation is linear.

Entropy Measure For Ranking Features

- A method for unsupervised feature selection or ranking based on entropy measure is a relatively simple technique; but with a large number of features its complexity increases significantly.
- The basic assumption is that all samples are given as vectors of a feature's values without any classification of output samples.
- The approach is based on the observation that removing an irrelevant feature, a redundant feature, or both from a set may not change the basic characteristics of the data set.
- The idea is to remove as many features as possible but yet maintain the level of distinction between the samples in the data set as if no features had been removed.
- The algorithm is based on a similarity measure S that is in inverse proportion to the distance D between two n-dimensional samples.

Entropy Measure For Ranking Features

- The distance measure D is small for close samples (close to zero) and large for distinct pairs (close to one). When the features are numeric, the similarity measure S of two samples can be defined as

$$S_{ij} = e^{-\alpha Dij}$$

- where Dij is the distance between samples xi and xj and α is a parameter mathematically expressed as

$$\alpha = -(\ln\ 0.5)/D$$

- D is the average distance among samples in the data set. Hence, α is determined by the data. But, in a successfully implemented practical application, it was used a constant value of α = 0.5.

Entropy Measure For Ranking Features: Normalized Euclidean distance measure

- Normalized Euclidean distance measure is used to calculate the distance Dij between two samples xi and xj:

$$D_{ij} = \left[\sum_{k=1}^{n} ((x_{ik} - x_{jk})/(max_k - min_k))^2 \right]^{1/2}$$

- where n is the number of dimensions and max_k and min_k are maximum and minimum values used for normalization of the k-th dimension.

Entropy Measure For Ranking Features: Similarity for nominal variables

- All features are not numeric. The similarity for nominal variables is measured directly using Hamming distance:

$$S_{ij} = \left(\sum_{k=1}^{n} |x_{ik} = x_{jk}| \right) / n$$

  - where $|x_{ik} = x_{jk}|$ is 1 if $x_{ik} = x_{jk}$, and 0 otherwise. The total number of variables is equal to n. For mixed data, we can discretize numeric values and transform numeric features into nominal features before we apply this similarity measure.

Entropy Measure For Ranking Features: Similarity for nominal variables

- This is an example of a simple data set with three categorical features; corresponding similarities are given in the left table.

Sample	F_1	F_2	F_3
R_1	A	X	1
R_2	B	Y	2
R_3	C	Y	2
R_4	B	X	1
R_5	C	Z	3

a) Data set with three categorical features

→

	R_1	R_2	R_3	R_4	R_5
R_1		0/3	0/3	2/3	0/3
R_2			2/3	1/3	0/3
R_3				0/3	1/3
R_4					0/3

b) A table of similarity measures S_{ij} between samples

Entropy Measure For Ranking Features:

- From information theory, we know that entropy is a global measure, and that it is less for ordered configurations and higher for disordered configurations.
- The proposed technique compares the entropy measure for a given data set before and after removal of a feature. If the two measures are close, then the reduced set of features will satisfactorily approximate the original set.
- For a data set of N samples, the entropy measure is

$$E = -\sum_{i=1}^{N-1}\sum_{j=I+1}^{N}(S_{ij} \times \log S_{ij} + (1 - S_{ij}) \times \log(1 - S_{ij}))$$

- where Sij is the similarity between samples xi and xj. This measure is computed in each of the iterations as a basis for deciding the ranking of features. We rank features by gradually removing the least important feature in maintaining the order in the configurations of data. The steps of the algorithm are base on sequential backward ranking, and they have been successfully tested on several real-world applications

Entropy Measure For Ranking Features:

1. Start with the initial full set of features F.
2. For each feature f $\in$ F, remove one feature f from F and obtain a subset F_f. Find the difference between entropy for F and entropy for all F_f. In our example, we have to compare the differences $(E_F - E_{F-F1})$, $(E_F - E_{F-F2})$, and $(E_F - E_{F-F3})$.
3. Let f_k be a feature such that the difference between entropy for F and entropy for F_{fk} is minimum.
4. Update the set of features F = F – {Fk}, where – is a difference operation on sets. In our example, if the difference $(E_F - E_{F-F1})$ is minimum, then the reduced set of features is {F2, F3}. F1 becomes the bottom of the ranked list.
5. Repeat steps 2-4 until there is only one feature in F.

Values Reduction

- A reduction in the number of discrete values for a given feature is based on the second set of techniques in the data-reduction phase; these are the ***feature-discretization techniques***.
- The task of feature-discretization techniques is to discretize the values of continuous features into a small number of intervals, where each interval is mapped to a discrete symbol.
- The benefits of these techniques are simplified data description and easy-to-understand data and final data-mining results. Also, more datamining techniques are applicable with discrete feature values.
- An "old fashioned" discretization is made manually, based on our a priori knowledge about the feature. For example, using common sense or consensus, a person's age, given at the beginning of a data-mining process as a continuous value (between 0 and 150 years) may be classified into categorical segments: child, adolescent, adult, middle age, and elderly. Cut off points are subjectively defined

Values Reduction

- Two main questions exist about this reduction process:
  1. What are the cut-off points?
  2. How does one select representatives of intervals?

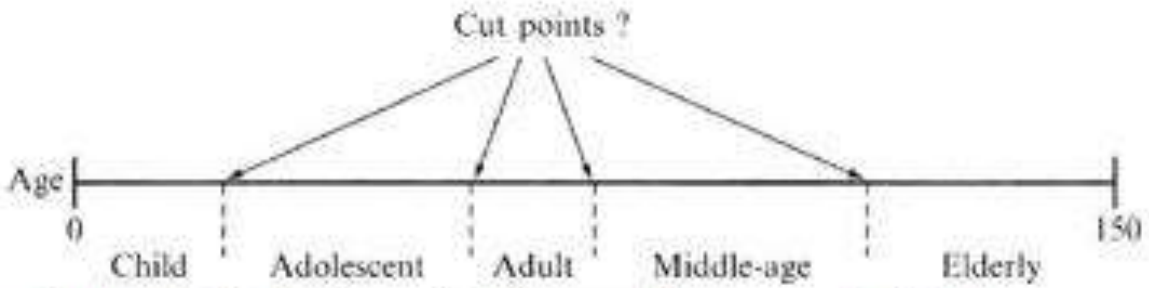


Figure 3.3: Discretization of the *age* feature

Values Reduction:

- Suppose that a feature has a range of numeric values, and that these values can be ordered from the smallest to the largest using standard greater-than and less-than operators.
- This leads naturally to the concept of *placing the values in bins*- partitioning into groups with close values.
- Typically, these bins have a close number of elements. All values in a bin will be merged into a single concept represented by a single value- usually either the mean or median of the bin's values. The mean or the mode is effective for a moderate or large number of bins.
- When the number of bins is small, the closest boundaries of each bin can be candidates for representatives in a given bin.

Values Reduction:

- For example, if a set of values for a given feature f is

  {3, 2, 1, 5, 4, 3, 1, 7, 5, 3}, then, after sorting, these values will be organized into an ordered set:

  {1, 1, 2, 3, 3, 3, 4, 5, 5, 7}

- Now, it is possible to split the total set of values into three bins with a close number of elements:

  {1, 1, 2, 3, 3, 3, 4, 5, 5, 7}

  BIN_1 BIN_2 BIN_3

- In the next step, different representatives can be selected for each bin. If the smoothing is performed based on bin modes, the new set of values for each bin will be

  {1, 1, 1, 3, 3, 3, 5, 5, 5, 5}

  BIN_1 BIN_2 BIN_3

Values Reduction:

- If the smoothing is performed based on mean values, then the new distribution for reduced set of values will be

{1.33, 1.33, 1.33, 3, 3, 3, 5.25, 5.25, 5, 25}

BIN_1 BIN_2 BIN_3

- and finally, if all the values in a bin are replaced by the closest of the boundary values, the new set will be

{1, 1, 2, 3, 3, 3, 4, 4, 4, 7}

BIN_1 BIN_2 BIN_3

Feature Discretization: ChiMerge Technique

- ChiMerge is one automated discretization algorithm that analyzes the quality of multiple intervals for a given feature by using χ2 statistics.
- The algorithm determines similarities between distributions of data in two adjacent intervals based on output classification of samples.
- If the conclusion of the χ2 test is that the output class is independent of the feature's intervals, then the intervals should be merged; otherwise, it indicates that the difference between intervals is statistically significant, and no merger will be performed.

Feature Discretization: ChiMerge Technique

- ChiMerge algorithm consists of three basic steps for discretization:
  1. Sort the data for the given feature in ascending order.
  2. Define initial intervals so that every value of the feature is in a separate interval.
  3. Repeat until no χ^2 of any two adjacent intervals is less then threshold value.
- After each merger, χ^2 tests for the remaining intervals are calculated, and two adjacent features with the smallest χ^2 values are found. If the calculated χ^2 is less than the threshold, merge these intervals. If no merge is possible, and the number of intervals is greater than the user-defined maximum, increase the threshold value.

Feature Discretization: ChiMerge Technique

- The χ2 test or contingency-table test is used in the methodology for determining the independence of two adjacent intervals.
- When the data are summarized in a contingency table, the χ2 test is given by the formula:

$$\chi^2 = \sum_{i=1}^{2} \sum_{j=1}^{k} (A_{ij} - E_{ij})^2 / E_{ij}$$

where

k = the number of classes,

A_{ij} = the number of instances in the i-th interval, j-th class,

E_{ij} = the expected frequency of A_{ij}, which is computed as $(R_i \cdot C_j) N$,

R_i = the number of instances in the i-th interval = ΣA_{ij}, $j = 1, \ldots, k$,

C_j = the number of instances in the j-th class = ΣA_{ij}, $i = 1, 2$,

N = the total number of instances = ΣR_i, $i = 1, 2$.

Feature Discretization: ChiMerge Technique

- For a classification problem with two classes (k = 2), where the merging of two intervals is analyzed, the contingency table for 2 × 2 data has the form given in Table. A_{11} represents the number of samples in the first interval belonging to the first class, A_{12} is the number of samples in the first interval belonging to the second class, A_{21} is the number of samples in the second interval belonging to the first class, and finally A_{22} is the number of samples in the second interval belonging to the second class.

Table 3.4: A contingency table for 2 × 2 categorical data

	Class 1	Class 2	Σ
Interval-1	A_{11}	A_{12}	R_1
Interval-2	A_{21}	A_{22}	R_2
Σ	C_1	C_2	N

Feature Discretization: ChiMerge Technique

- We will analyze the ChiMerge algorithm using one relatively simple example, where the database consists of 12 two-dimensional samples with only one continuous feature (F) and an output classification feature (K).
- The two values 1 and 2 for the feature K represent the two classes to which the samples belong. The initial data set, already sorted with respect to the continuous numeric feature F, is given in Table.

ChiMerge Example

- Example: Dataon the sorted continuous feature F with corresponding Classes K. Analyze the Chi-Merge algorithm.

Sample ID	Feature F	Class (K)
1	1	1
2	3	2
3	7	1
4	8	1
5	9	1
6	11	2
7	23	2
8	37	1
9	39	2
1	45	1
11	46	1
12	59	1

ChiMerge Example:

	K=1	K=2	Σ
Interval [7.5, 8.5]	$A_{1,1}=1$	$A_{1,2}=0$	$R_1=1$
Interval [8.5, 10]	$A_{2,1}=1$	$A_{2,2}=0$	$R_2=1$
Σ	$C_1=2$	$C_2=0$	N=2

$$X^2 = \sum_{i=1}^{2} \sum_{j=1}^{k} (A_{i,j} - E_{i,j})/E_{i,j}$$

$X^2=(1-1)^2/1+ (0-0.1)^2/0.1+ (1-1)^2/1 + (0-0.1)^2/0.1$

$X^2=0.2$

For the degree of freedom d=1 and $X^2=0.2<2.706$, we can conclude that there are no significant differences in relative class frequencies and the selected intervals can be merged. And Interval will be [7.5, 10]

ChiMerge Example

- The merging process will be applied in one iteration only for two adjacent intervals with a minimum X^2 and, at the same time with X^2 > threshold value. The iterative process will continue with the next adjacent intervals that have the minimum X2.
- We will Show one additional step in somewhere in the middle of the merging process, where the intervals [0, 7.5] and [7.5, 10] are analized.

ChiMerge Example:

	K=1	K=2	Σ
Interval [0, 7.5]	$A_{1,1}=2$	$A_{1,2}=1$	$R_1=3$
Interval [7.5, 10]	$A_{2,1}=2$	$A_{2,2}=0$	$R_2=2$
Σ	$C_1=4$	$C_2=1$	N=5

$$X^2 = \sum_{i=1}^{2} \sum_{j=1}^{k} (A_{i,j} - E_{i,j})/E_{i,j}$$

$X^2=(2-2.4)^2/2.4+ (1-0.6)^2/0.6+ (2-1.6)^2/1.6 + (0-0.4)^2/0.4$

$X^2=0.834$

Selected intervals should be merged into one because for the degree of freedom d=1 and $X^2=0.834<2.706$

ChiMerge Example

- In our example, with the given threshold value for X2, the algorithm will define a final discretization with three intervals :
- [0, 10], [10, 42], [42, 60], where 60 is supposed to be the maximum value for the feature F. We can assign to this intervals coded valeus **1, 2, 3** or descriptive linguistic values **low, medium and high**.

ChiMerge Example:

	K=1	K=2	Σ
Interval [0, 10]	$A_{1,1}$=4	$A_{1,2}$=1	R_1=5
Interval [10, 42]	$A_{2,1}$=1	$A_{2,2}$=3	R_2=4
Σ	C_1=5	C_2=4	N=

$$X^2 = \sum_{i=1}^{2}\sum_{j=1}^{k}(A_{i,j} - E_{i,j})/E_{i,j}$$

$X^2=(4-2.78)^2/2.78 + (1-2.22)^2/2.22 + (1-2.22)^2/2.22 + (3-1.78)^2/1.78$

$X^2=2.72$

$X^2=2.72>2.706$, the conclusion is that significant differences between two intervals exist, the merging is nor recommended!