-to high dimensionality data

Sampling → Let D is data set contains N tuples.

1) simple random sample without replacement (SRSWOR) of size S → This is done by selecting 'S' no. of tuples from N (S<N) of D, where probability of drawing any tuple in D is 1/N. All tuples are equally likely.

2) simple random sample with replacement (SRSWR) of size S → same as above, but each time a tuple is drawn from D, it is recorded and then replaced, so that it may be drawn again.

3) cluster sample → The tuples in D are grouped into M mutually disjoint clusters, then as SRS of S clusters can be obtained where S<M.

Stratified sample

Here D is divided into mutually disjoint parts called 'Strata'. Then a stratified sample is generated by obtaining an simple random sample at each stratum.

This helps ensure a representative sample, especially when data are skewed.

Top-down split,

Let D consist of data tuples defined by a set of attributes A and a class-label attribute. The basic method for discretization of an attribute A within the set is as follows:

1) A within the set is as follows:

> Each value of A can be considered as a potential interval boundary or split-point to partition the range of A. That is, a split point for A can partition the tuples in D into two subsets, A ≤ split_point and A > split_point, creating a binary discretization

2) Let we have two classes C_1 and C_2 we want all tuples of C_1 will fall into one partition and all tuples of C_2 will fall into other partition. But the 1st partition containing all tuples of C_1 and few tuples of C_2. The amount of more information, still we need for a perfect partition is known as expected information requirement.

Given by

$$Info_A(D) = \frac{|D_1|}{|D|} entropy(D_1) + \frac{|D_2|}{|D|} entropy(D_2)$$

where D_1 and D_2 correspond to tuples in D satisfying A ≤ split-point, and A > split-point.

|D| = total no. of tuples in D

m classes, then the entropy is

$$entropy(D) = -\sum_{i=1}^{m} p_i \log_2(p_i)$$

where p_i is the probability of class C_i in D_1

$$= \frac{\text{total no. of tuples of class } C_i \text{ in } D_1}{\text{total no. of tuples in } D_1}$$

Therefore, when selecting a split point for attribute A, we would pick the attribute value that gives the minimum $info_A(D)$.

3) The process of determining a split point is recursively applied to each partition, until some stopping criteria is met.

2.2 Suppose that the values for a given set of data are grouped into intervals. The intervals and corresponding frequencies are as follows.

age	*frequency*
1–5	200
5–15	450
15–20	300
20–50	1500
50–80	700
80–110	44

Compute an *approximate median* value for the data.

2.3 Give three additional commonly used statistical measures (i.e., not illustrated in this chapter) for the characterization of *data dispersion,* and discuss how they can be computed efficiently in large databases.

2.4 Suppose that the data for analysis includes the attribute *age*. The *age* values for the data tuples are (in increasing order) 13, 15, 16, 16, 19, 20, 20, 21, 22, 22, 25, 25, 25, 25, 30, 33, 33, 35, 35, 35, 35, 36, 40, 45, 46, 52, 70.

(a) What is the *mean* of the data? What is the *median*?
(b) What is the *mode* of the data? Comment on the data's modality (i.e., bimodal, trimodal, etc.).
(c) What is the *midrange* of the data?
(d) Can you find (roughly) the first quartile ($Q1$) and the third quartile ($Q3$) of the data?
(e) Give the *five-number summary* of the data.
(f) Show a *boxplot* of the data.
(g) How is a *quantile-quantile plot* different from a *quantile plot*?

2.5 In many applications, new data sets are incrementally added to the existing large data sets. Thus an important consideration for computing descriptive data summary is whether a measure can be computed efficiently in incremental manner. Use *count, standard deviation,* and *median* as examples to show that a distributive or algebraic measure facilitates efficient incremental computation, whereas a holistic measure does not.

2.6 In real-world data, tuples with *missing values* for some attributes are a common occurrence. Describe various methods for handling this problem.

2.7 Using the data for *age* given in Exercise 2.4, answer the following.

(a) Use *smoothing by bin means* to smooth the data, using a bin depth of 3. Illustrate your steps. Comment on the effect of this technique for the given data.
(b) How might you determine *outliers* in the data?
(c) What other methods are there for *data smoothing*?

2.8 Discuss issues to consider during *data integration*.

2.9 Suppose a hospital tested the age and body fat data for 18 randomly selected adults with the following result:

age	23	23	27	27	39	41	47	49	50
%fat	9.5	26.5	7.8	17.8	31.4	25.9	27.4	27.2	31.2
age	52	54	54	56	57	58	58	60	61
%fat	34.6	42.5	28.8	33.4	30.2	34.1	32.9	41.2	35.7

(a) Calculate the mean, median, and standard deviation of *age* and *%fat*.
(b) Draw the boxplots for *age* and *%fat*.
(c) Draw a *scatter plot* and a *q-q plot* based on these two variables.
(d) Normalize the two variables based on *z-score normalization*.
(e) Calculate the *correlation coefficient* (Pearson's product moment coefficient). Are these two variables positively or negatively correlated?

2.10 What are the value ranges of the following *normalization methods*?

(a) min-max normalization
(b) z-score normalization
(c) normalization by decimal scaling

2.11 Use the two methods below to *normalize* the following group of data:
200, 300, 400, 600, 1000

(a) min-max normalization by setting $min = 0$ and $max = 1$
(b) z-score normalization

2.12 Using the data for *age* given in Exercise 2.4, answer the following:

(a) Use min-max normalization to transform the value 35 for *age* onto the range [0.0, 1.0].
(b) Use z-score normalization to transform the value 35 for *age*, where the standard deviation of *age* is 12.94 years.
(c) Use normalization by decimal scaling to transform the value 35 for *age*.
(d) Comment on which method you would prefer to use for the given data, giving reasons as to why.

2.13 Use a flowchart to summarize the following procedures for *attribute subset selection*:

(a) stepwise forward selection
(b) stepwise backward elimination
(c) a combination of forward selection and backward elimination

2.14 Suppose a group of 12 *sales price* records has been sorted as follows:
5, 10, 11, 13, 15, 35, 50, 55, 72, 92, 204, 215
Partition them into three bins by each of the following methods:

(a) equal-frequency (equidepth) partitioning

(b) equal-width partitioning

(c) clustering

2.15 Using the data for *age* given in Exercise 2.4,

(a) Plot an equal-width histogram of width 10.

(b) Sketch examples of each of the following sampling techniques: SRSWOR, SRSW cluster sampling, stratified sampling. Use samples of size 5 and the strata "youth "middle-aged," and "senior."

2.16 [Contributed by Chen Chen] The *median* is one of the most important holistic me sures in data analysis. Propose several methods for median approximation. Analyze the respective complexity under different parameter settings and decide to what extent t real value can be approximated. Moreover, suggest a heuristic strategy to balance betwe accuracy and complexity and then apply it to all methods you have given.

2.17 [Contributed by Deng Cai] It is important to define or select similarity measures in da analysis. However, there is no commonly accepted subjective similarity measure. Usin different similarity measures may deduce different results. Nonetheless, some apparent different similarity measures may be equivalent after some transformation.

Suppose we have the following two-dimensional data set:

	A_1	A_2
x_1	1.5	1.7
x_2	2	1.9
x_3	1.6	1.8
x_4	1.2	1.5
x_5	1.5	1.0

(a) Consider the data as two-dimensional data points. Given a new data point, $x =$ (1.4, 1.6) as a query, rank the database points based on similarity with the query usin (1) Euclidean distance (Equation 7.5), and (2) cosine similarity (Equation 7.16).

(b) Normalize the data set to make the norm of each data point equal to 1. Use Euclidea distance on the transformed data to rank the data points.

2.18 ChiMerge [Ker92] is a supervised, bottom-up (i.e., merge-based) *data discretizatio* method. It relies on χ^2 analysis: adjacent intervals with the least χ^2 values are merge together until the stopping criterion is satisfied.