Database Sampling for Data Mining

Patricia E.N. Lutu
University of Pretoria, South Africa

INTRODUCTION

In data mining, sampling may be used as a technique for reducing the amount of data presented to a data mining algorithm. Other strategies for data reduction include dimension reduction, data compression, and discretisation. For sampling, the aim is to draw, from a database, a random sample, which has the same characteristics as the original database. This chapter looks at the sampling methods that are traditionally available from the area of statistics, how these methods have been adapted to database sampling in general, and database sampling for data mining in particular.

BACKGROUND

Given the rate at which database/data warehouse sizes are growing, attempts at creating faster/more efficient algorithms that can process massive data sets, may eventually become futile exercises. Modern database and data warehouse sizes are in the region of 10’s or 100’s of terabytes, and sizes continue to grow. A query issued on such a database/data warehouse could easily return several millions of records.

While the costs of data storage continue to decrease, the analysis of data continues to be hard. This is the case for even traditionally simple problems requiring aggregation, for example, the computation of a mean value for some database attribute. In the case of data mining, the computation of very sophisticated functions, on very large number of database records, can take several hours, or even days. For inductive algorithms, the problem of lengthy computations is compounded by the fact that many iterations are needed in order to measure the training accuracy as well as the generalization accuracy.

There is plenty of evidence to suggest that for inductive data mining, the learning curve flattens after only a small percentage of the available data, from a large data set has been processed (Catlett 1991; Kohavi 1996; Provost et al, 1999). The problem of overfitting (Dietterich, 1995), also dictates that the mining of massive data sets should be avoided. The sampling of databases has been studied by researchers for some time. For data mining, sampling should be used as a data reduction technique, allowing a data set to be represented by a much smaller random sample that can be processed much more efficiently.

MAIN THRUST OF THE CHAPTER

There are a number of key issues to be considered before obtaining a suitable random sample for a data mining task. It is essential to understand the strengths and weaknesses of each sampling method. It is also essential to understand which sampling methods are more suitable to the type of data to be processed and the data mining algorithm to be employed. For research purposes, we need to look at a variety of sampling methods used by statisticians, and attempt to adapt them to sampling for data mining.

Some Basics of Statistical Sampling Theory

In statistics, the theory of sampling, also known as statistical estimation, or the representative method, deals with the study of suitable methods of selecting a representative sample of a population, in order to study or estimate values of specific characteristics of the population (Neyman, 1934). Since the characteristics being studied can only be estimated from the sample, confidence intervals are calculated to give the range of values within which the actual value will fall, with a given probability.

There are a number of sampling methods discussed in the literature, for example the book by Rao (Rao, 2000). Some methods appear to be more suited to
Database Sampling Methods

Database sampling has been practiced for many years for purposes of estimating aggregate query results, database auditing, query optimization, and, obtaining samples for further statistical processing (Olken, 1993). Static sampling (Olken, 1993) and adaptive (dynamic) sampling (Haas & Swami, 1992) are two alternatives for obtaining samples for data mining tasks. In recent years, many studies have been conducted in applying sampling to inductive and non-inductive data mining (John & Langley, 1996; Provost et al, 1999; Toivonen, 1996).

Simple Random Sampling

Simple random sampling is by far, the simplest method of sampling a database. Simple random sampling may be implemented using sequential random sampling or reservoir sampling. For sequential random sampling, the problem is to draw a random sample of size n without replacement, from a file containing N records. The simplest sequential random sampling method is due to Fan et al (1962) and Jones (1962). An independent uniform random variate (from the uniform interval (0,1) ) is generated for each record in the file to determine whether the record should be included in the sample. If m records have already been chosen from among the first t records in the file, the (t+1)th record is chosen with probability (RQsize / RMsize), where RQsize = (n-m) is the number of records that still need to be chosen for the sample, and RMsize = (N-t) is the number of records in the file, still to be processed. This sampling method is commonly referred to as method S (Vitter, 1987).

The reservoir sampling method (Fan et al, 1962; Jones, 1962; Vitter, 1985 & 1987) is a sequential sampling method over a finite population of database records, with an unknown population size. Olken, 1993) discuss its use in sampling of database query outputs on the fly. This technique produces a sample of size S, by initially placing the first S records of the database/file/query in the reservoir. For each subsequent kth database record, that record is accepted with probability S / k. If accepted, it replaces a randomly selected record in the reservoir.

Acceptance/Rejection sampling (A/R sampling) can be used to obtain weighted samples (Olken 1993). For a weighted random sample, the probabilities of inclusion of the elements of the population are not uniform. For database sampling, the inclusion probability of a data record is proportional to some weight calculated from the record’s attributes. Suppose that one database record rj is to be drawn from a file of n records with the probability of inclusion being proportional to the weight wj. This may be done by generating a uniformly distributed random integer j, 1 ≤ j ≤ n and then accepting the sampled record rj with probability αj = wj /