Data Mining with Cubegrades

Amin A. Abdulghani
Data Mining Engineer, USA

INTRODUCTION

A lot of interest has been expressed in database mining using association rules (Agrawal, Imielinski, & Swami, 1993). In this chapter, we provide a different view of the association rules, referred to as *cubegrades* (Imielinski, Khachiyan, & Abdulghani, 2002) .

An example of a typical *association rule* states that, say, 23% of supermarket transactions (so called market basket data) which buy bread and butter buy also cereal (that percentage is called *confidence*) and that 10% of all transactions buy bread and butter (this is called *support*). Bread and butter represent the body of the rule and cereal constitutes the consequent of the rule. This statement is typically represented as a probabilistic rule. But *association rules* can also be viewed as statements about how the cell representing the body of the rule is affected by specializing it by adding an extra constraint expressed by the rule's consequent. Indeed, the confidence of an association rule can be viewed as the ratio of the support drop, when the cell corresponding to the body of a rule (in our case the cell of transactions buying bread and butter) is augmented with its consequent (in this case cereal). This interpretation gives association rules a "dynamic flavor" reflected in a hypothetical change of support affected by specializing the body cell to a cell whose description is a union of body and consequent descriptors. For example, our earlier association rule can be interpreted as saying that the count of transactions buying bread and butter drops to 23% of the original when restricted (rolled down) to the transactions buying bread, butter and cereal. In other words, this rule states how the count of transactions supporting buyers of bread and butter is affected by buying cereal as well.

With such interpretation in mind, a much more general view of association rules can be taken, when support (count) can be replaced by an arbitrary measure or aggregate and the specialization operation can be substituted with a different "delta" operation. Cubegrades capture this generalization. Conceptually, this is very similar to the notion of gradients used in calculus. By definition the gradient of a function between the domain points x_1 and x_2 measures the ratio of the *delta change* in the function value over the *delta change* between the points. For a given point x and function f(), it can be interpreted as a statement of how a change in the value of x (Δx), affects a change of value in the function ($\Delta f(x)$).

From another viewpoint, *cubegrades* can also be considered as defining a primitive for *data cubes*. Consider a 3-D cube model shown in Figure 1 representing sales data. It has three dimensions year, product and location. The measurement of interest is total sales. In olap terminology, since this cube models the base data, it forms a 3-D *base cuboid*. A *cuboid* in general is a group-by of a subset of dimensions of the base data, obtained by aggregating all tuples on these dimensions. So, for example for our sales data we have three 2-d cuboids namely (year, product), (product, location) and (year, location), three 1-d cuboids (year), (location) and (product) and one 0-d cuboid in which aggregation is performed on the whole data. For base data, with n diemensions, the union of of all *k*-dimensional (k<=n) cuboids forms *an n-dimensional data cube*. A *cell* represents an association of a measure *m* (e.g., total sales) with a member of every dimension in a cuboid e.g. C1(product="toys", location="NJ", year="2004"). The dimensions not present in the cell are aggregated over all possible members. For example, you can have a two-dimensional (2-D) cell, C2(product="toys", year="2004"). Here, the implicit value for the dimension location is '*', and the measure *m* (e.g., total sales) is aggregated over all locations. Any of the standard aggregate functions such as count, total, average, minimum, or maximum can be used for aggregating. Suppose the sales for toys in 2004 for NJ, NY, PA were \$2.5M, \$3.5M, \$1.5M respectively and that the aggregating function is total. Then, the measure value for cell C2 is \$7.5M.

The scope of interest in OLAP is to evaluate one or more measure values of the cells in the cube. Cubegrades allow a broader, more dynamic view. In addition to evaluating the measure values in a cell,

Figure 1. A 3-D base cuboid with an example 3-D cell.

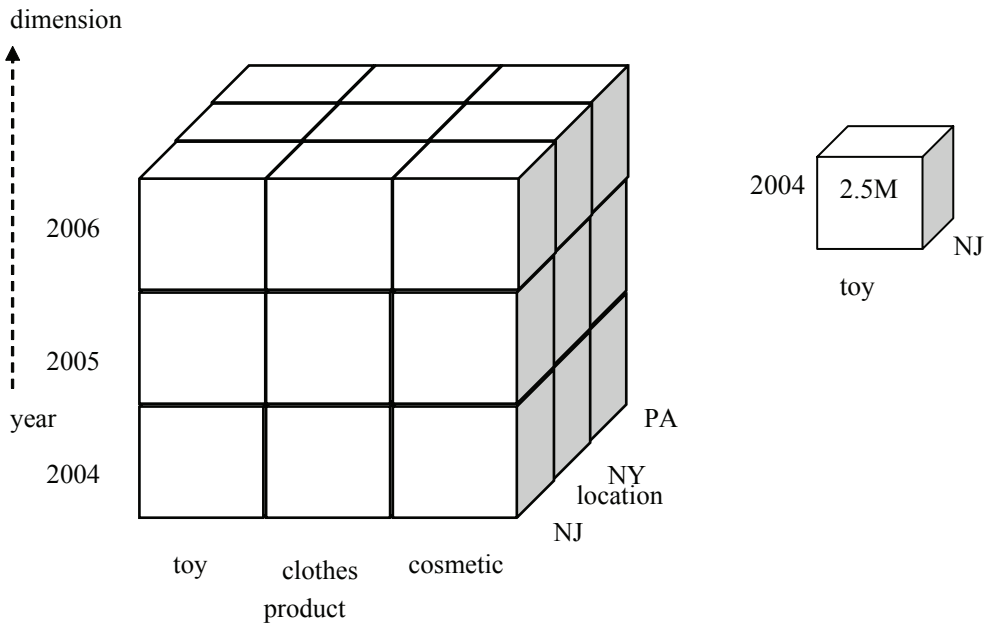
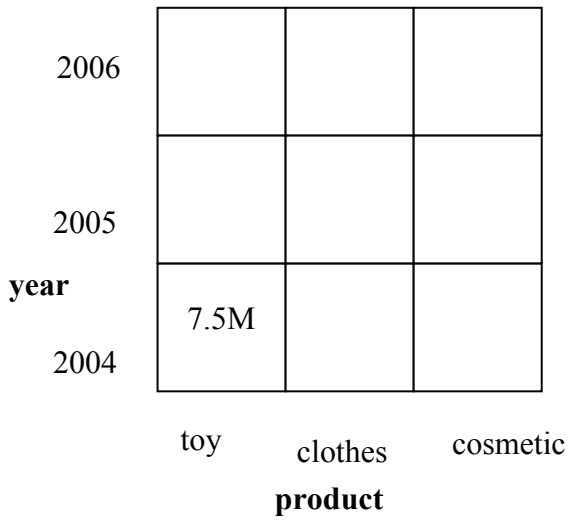


Figure 2. An example 2-D cuboid on (product, year) for the 3-D cube in Figure 1 (location=''); total sales needs to be aggregated (e.g., SUM)*



they evaluate how the measure values change or are affected in response to a change in the dimensions of a cell. Traditionally, OLAP have had operators such as drill downs, rollups defined, but the cubegrade operator differs from them as it returns a value measuring the effect of the operation. There have been additional operators proposed to evaluate/measure cell *interestingness* (Sarawagi, 2000; Sarawagi, Agrawal, & Megiddo, 1998). For example, Sarawagi et al., (1998) computes anticipated value for a cell using the neighborhood values, and a cell is considered an exception if its value is significantly different from its anticipated value. The difference is that cubegrades perform a direct cell to cell comparison.