Data Mining for Model Identification

Diego Liberati
Italian National Research Council, Italy

INTRODUCTION

In many fields of research, as well as in everyday life, it often turns out that one has to face a huge amount of data, without an immediate grasp of an underlying simple structure, often existing. A typical example is the growing field of bio-informatics, where new technologies, like the so-called Micro-arrays, provide thousands of gene expressions data on a single cell in a simple and fast integrated way. On the other hand, the everyday consumer is involved in a process not so different from a logical point of view, when the data associated to his fidelity badge contribute to the large data base of many customers, whose underlying consuming trends are of interest to the distribution market.

After collecting so many variables (say gene expressions, or goods) for so many records (say patients, or customers), possibly with the help of wrapping or warehousing approaches, in order to mediate among different repositories, the problem arise of reconstructing a synthetic mathematical model capturing the most important relations between variables. To this purpose, two critical problems must be solved:

1 To select the most salient variables, in order to reduce the dimensionality of the problem, thus simplifying the understanding of the solution
2 To extract underlying rules implying conjunctions and/or disjunctions between such variables, in order to have a first idea of their even non linear relations, as a first step to design a representative model, whose variables will be the selected ones

When the candidate variables are selected, a mathematical model of the dynamics of the underlying generating framework is still to be produced. A first hypothesis of linearity may be investigated, usually being only a very rough approximation when the values of the variables are not close to the functioning point around which the linear approximation is computed.

On the other hand, to build a non linear model is far from being easy: the structure of the non linearity needs to be a priori known, which is not usually the case. A typical approach consists in exploiting a priori knowledge to define a tentative structure, and then to refine and modify it on the training subset of data, finally retaining the structure that best fits a cross-validation on the testing subset of data. The problem is even more complex when the collected data exhibit hybrid dynamics, i.e. their evolution in time is a sequence of smooth behaviors and abrupt changes.

BACKGROUND

Such tasks may be sequentially accomplished with various degree of success in a variety of ways. Principal components (O'Connel 1974) orders the variables from the most salient to the least one, but only under a linear framework (Liberati et al., 1992a).

Partial least squares (Dijkstra 1983) allow to extend to non linear models, provided that one has some a priori information on the structure of the involved non linearity: in fact, the regression equation needs to be written before identifying its parameters. Clustering may operate in an unsupervised way, without the a priori correct classification of a training set (Booley 1998).

Neural networks are known to learn the embedded rules, with the indirect possibility (Taha and Ghosh 1999) to make rules explicit or to underline the salient variables. Decision trees (Quinlan 1994) are a popular framework providing a satisfactory answer to both questions.

Systems identification (Söderström and Stoica, 1989) is widely and robustly addressed even in calculus tools like Matlab from Matworks.

MAIN FOCUS

More recently, a different approach has been suggested, named Hamming Clustering (Muselli & Liberati 2000). It is related to the classical theory exploited in minimizing the size of electronic circuits, with the additional care to obtain a final function able to generalize from the training data set to the most likely framework describing the actual properties of the data. Such approach enjoys the following remarkable two properties:

a) It is fast, exploting, after proper binary coding, just logical operations instead of floating point multiplications

b) It directly provides a logical understandable expression (Muselli & Liberati 2002), being the final synthesized function directly expressed as the OR of ANDs of the salient variables, possibly negated.

An alternative approach is to infer the model directly from the data via an identification algorithm capable to reconstruct a very general class of piece-wise affine models (Ferrari-Trecate et al., 2003). This method can be also exploited for the data-driven modeling of hybrid dynamical systems where logic phenomena interact with the evolution of continuous-valued variables.. Such approach will be concisely described later in the following, after a little more detailed drawing of the rules-oriented mining procedure. The last section will briefly discuss some applications.

HAMMING CLUSTERING: BINARY RULE GENERATION AND VARIABLE SELECTION WHILE MINING DATA

The approach followed by Hamming Clustering (HC) in mining the available data to select the salient variables and to build the desired set of rules consists of the three steps in Table 1.

- **Step 1:** A critical issue is the partition of a possibly continuous range in intervals, whose number and limits may affect the final result. The thermometer code may then be used to preserve ordering and distance (in case of nominal input variables, for which a natural ordering cannot be defined, the only-one code may instead be adopted). The metric used is the Hamming distance, computed as the number of different bits between binary strings: the training process does not require floating point computation, but only basic logic operations, for the sake of speed and insensitivity to precision.
- **Step 2:** Classical techniques of logical synthesis, such as ESPRESSO (Brayton at al 1984) or MINI (Hong 1997), are specifically designed to obtain the simplest AND-OR expression able to satisfy all the available input-output pairs, without an explicit attitude to generalize. To generalize and infer the underlying rules, at every iteration HC groups together, in a competitive way, binary strings having the same output and close to each other. A final pruning phase does simplify the resulting expression, further improving its generalization ability. Moreover, the minimization of the involved variables do intrinsically exclude the redundant ones, thus enhancing the very salient variables for the investigated problem. The quadratic computational cost allows to manage quite large datasets.
- **Step 3:** Each logical product directly provides an intelligible rule synthesizing a relevant aspect of the searched underlying system that is believed to generate the available samples.

Table 1. The tree steps executed by Hamming clustering to build the set of rules embedded in the mined data

1. The input variables are converted into binary strings via a coding designed to preserve distance and, if relevant, ordering.
2. The ‘OR of ANDs’ expression of a logical function is derived from the training examples coded in the binary form of step 1
3. In the OR final expression, each logical AND provides intelligible conjunctions or disjunctions of the involved variables, ruling the analyzed problem