Available online at www.sciencedirect.com
ScienceDirect
Computational Statistics & Data Analysis 51 (2006) 398–416
COMPUTATIONAL STATISTICS & DATA ANALYSIS
www.elsevier.com/locate/csda

Design of local fuzzy models using evolutionary algorithms

Piero P. Bonissone*, Anil Varma, Kareem S. Aggour, Feng Xue

General Electric Global Research, One Research Circle, Niskayuna, NY 12309, USA

Available online 11 May 2006

Abstract

The application of local fuzzy models to determine the remaining life of a unit in a fleet of vehicles is described. Instead of developing individual models based on the track history of each unit or developing a global model based on the collective track history of the fleet, local fuzzy models are used based on clusters of peers—similar units with comparable utilization and performance characteristics. A local fuzzy performance model is created for each cluster of peers. This is combined with an evolutionary framework to maintain the models. A process has been defined to generate a collection of competing models, evaluate their performance in light of the currently available data, refine the best models using evolutionary search, and select the best one after a finite number of iterations. This process is repeated periodically to automatically update and improve the overall model. To illustrate this methodology an asset selection problem has been identified: given a fleet of industrial vehicles (diesel electric locomotives), select the best subset for mission-critical utilization. To this end, the remaining life of each unit in the fleet is predicted. The fleet is then sorted using this prediction and the highest ranked units are selected. A series of experiments using data from locomotive operations was conducted and the results from an initial validation exercise are presented. The approach of constructing local predictive models using fuzzy similarity with neighboring points along appropriate dimensions is not specific to any asset type and may be applied to any problem where the premise of similarity along chosen attribute dimensions implies similarity in predicted future behavior.

Keywords: Fuzzy models; Evolutionary algorithms; Instance-based models; Similarity measures; Prediction

1. Introduction

Soft computing (SC) literature is expanding at a rapid pace, as evidenced by the numerous congresses, books, and journals devoted to the topic. Its original definition provided by Zadeh (1994) denotes systems that "... *exploit the tolerance for imprecision*, *uncertainty*, *and partial truth to achieve tractability*, *robustness*, *low solution cost*, *and better rapport with reality*." As discussed in previous articles (Bonissone, 1997; Bonissone et al., 1999), we view soft computing as the synergistic association of computing methodologies that includes as its principal members *fuzzy logic*, *neurocomputing*, *evolutionary computing*, and *probabilistic computing*. We also stress the synergy derived from hybrid SC systems that are based on a loose or tight integration of their constituent technologies. This integration provides complementary reasoning and search methods that allow us to combine domain knowledge and empirical data to develop flexible computing tools and solve complex problems.

* Corresponding author. Tel.: +1 518 387 5155; fax: +1 518 387 6845.
E-mail address: bonissone@research.ge.com (P.P. Bonissone).

0167-9473/$ - see front matter © 2006 Elsevier B.V. All rights reserved.
doi:10.1016/j.csda.2006.04.013

1.1. Fuzzy and statistical models

From the breadth of SC components, we focus on the use of (local) fuzzy models in our case study with the use of evolutionary computing as a methodology to design the local fuzzy models. It has been noted that analytical models capture well understood systems and relations involving few variables, while statistical models operate at the other extreme, dealing with large amounts of randomness and large numbers of variables (Klir and Yuan, 1995). These two extremes correspond to the concepts of *organized simplicity* and *disorganized complexity*, originally proposed by Weaver (1948). Fuzzy models are commonly used outside of these two extremes, when there is enough vagueness in the relationships to prevent us from applying analytical models, and yet there is not enough data to allow for meaningful statistical analysis. In the literature we can find many hybrid fuzzy-statistical approaches, such as statistical decisions using fuzzy probabilities or fuzzy utilities, testing fuzzy hypotheses, and regression analysis with fuzzy data, to name a few. Readers interested in the intersection and synergy between fuzziness and statistics should refer to (Corral et al., 1998; Ralescu, 1996). In our work we focus on the development of local fuzzy models, which are closely related to kernel regression and locally weighted regression. We will review the latter under related work in Section 3.2, while the former will be described in Section 4.2.

1.2. Characteristics of real world applications

When addressing real-world problems, we are typically dealing with systems that are ill defined, difficult to model, and possess large solution spaces. Under these circumstances, precise models are usually impractical, too expensive, or non-existent. Therefore, we need to generate approximate solutions by leveraging the two types of resources that are generally available: *problem domain knowledge* of the process (or product) and *field data* that characterize the system's behavior. The relevant available domain knowledge is typically a combination of first principles and empirical knowledge. This knowledge is often incomplete and sometimes erroneous. The available data are typically a collection of input–output measurements representing instances of the system's behavior, and are generally incomplete and noisy. Soft computing is a flexible framework in which we can find a broad spectrum of design choices to perform the integration of knowledge and data in the construction of approximate models.

In real-world applications, before we can use a model in a production environment we must address the model's entire lifecycle from its design and implementation to its validation, tuning, production testing, use, monitoring and maintenance. By maintenance we mean all the steps required to keep the model vital (e.g., non-obsolete) and able to adapt to changes. Two reasons justify our focus on maintenance. Over the lifecycle of the model, maintenance costs are the most expensive (as software maintenance is the most expensive in the life of a software system). Second, when dealing with mission-critical software we need to guarantee continuous operation or at least fast recovery from system failures or model obsolescence to avoid lost revenues and other business costs.

In the next section we describe a variety of soft computing models and some typical search methods that could be used to generate those models. Then we cover related work in case-based reasoning, statistics, and evolutionary search for model design. In Section 4 we describe the fuzzy instance based models, while in Section 5 we explain the use of evolutionary algorithms for their design. Section 6 illustrates a case study with a fleet of locomotives and describes the experiments and results performed to validate our approach. In Section 7 we show the impact of model updates, and in Section 8 we present our conclusions.

2. Models and fuzzy models

2.1. Model generation

In general terms, we can consider a model to be characterized by its *representation* (structural and parametric information) and its associated *reasoning mechanism*, which is usually related to the representation. The generation (and updating) of an optimal model requires a *search* method to define the model's representation and to characterize its reasoning mechanism. We will illustrate this concept with examples taken from conventional and soft computing models.