

Advanced Genetic Programming Based Machine Learning

Stephan Winkler · Michael Affenzeller · Stefan Wagner

Received: 1 November 2005 / Accepted: 1 December 2006
© Springer Science + Business Media B.V. 2007

Abstract A Genetic Programming based approach for solving classification problems is presented in this paper. Classification is understood as the act of placing an object into a set of categories, based on the object's properties; classification algorithms are designed to learn a function which maps a vector of object features into one of several classes. This is done by analyzing a set of input-output examples ("training samples") of the function. Here we present a method based on the theory of Genetic Algorithms and Genetic Programming that interprets classification problems as optimization problems: Each presented instance of the classification problem is interpreted as an instance of an optimization problem, and a solution is found by a heuristic optimization algorithm. The major new aspects presented in this paper are advanced algorithmic concepts as well as suitable genetic operators for this problem class (mainly the creation of new hypotheses by merging already existing ones and their detailed evaluation). The experimental part of the paper documents the results produced using new hybrid variants of Genetic Algorithms as well as investigated parameter settings. Graphical analysis is done using a novel multiclass classifier analysis concept based on the theory of Receiver Operating Characteristic curves.

Keywords Evolutionary algorithms · Genetic programming · Data mining

The work described in this paper was done within the Translational Research Project L282 "GP-Based Techniques for the Design of Virtual Sensors" sponsored by the Austrian Science Fund (FWF).

S. Winkler (✉) · M. Affenzeller · S. Wagner
Upper Austrian University of Applied Sciences, College of Information Technology
at Hagenberg, Hauptstraße 117, 4232 Hagenberg, Austria
e-mail: stephan@heuristiclab.com

M. Affenzeller
e-mail: michael@heuristiclab.com

S. Wagner
e-mail: stefan@heuristiclab.com

1 Introduction

In general, Data Mining is understood as the practice of automatically searching for patterns in large stores of data. Nowadays, incredibly large (and quickly growing) amounts of data are collected in commercial, administrative, and scientific databases. Several sciences (as for example molecular biology, genetics, astrophysics, and many others) produce extreme amounts of information which are often collected automatically. This is why it is impossible to analyze and exploit all these data manually; what is needed are intelligent computer systems that can extract useful information (such as general rules or interesting patterns) from large amounts of observations. In short, “data mining is the non-trivial process of identifying valid, novel, potentially useful, and ultimately understandable patterns in data” [8].

Classification is understood as the act of placing an object into a set of categories, based on the object’s properties. Objects are classified according to an (in most cases hierarchical) classification scheme also called taxonomy. Amongst many other possible applications, examples of taxonomic classification are biological classification (the act of categorizing and grouping living species of organisms), medical classification and security classification (where it is often necessary to classify objects or persons for deciding whether a problem might arise from the present situation or not). A statistical classification algorithm is supposed to take feature representations of objects and map them to a special, predefined classification label. Such classification algorithms are designed to learn (i.e. to approximate the behavior of) a function which maps a vector of object features into one of several classes; this is done by analyzing a set of input–output examples (“training samples”) of the function. Since statistical classification algorithms are supposed to “learn” such functions, we are dealing with a specific subarea of *Machine Learning* and, more generally, *Artificial Intelligence*.

There are several approaches which are nowadays used for solving data mining and, more specifically, classification problems. The most common ones are (as for example described in [23]) decision tree learning, instance-based learning, inductive logic programming (such as Prolog, e.g.) and reinforcement learning.

Unlike these methods, the approach we have designed is a genetic programming (GP) model including appropriate crossover and mutation operators for this problem. This GP approach, described in Section 3, has also been implemented as a part of the already existing “HeuristicLab,” a framework for prototyping and analyzing optimization techniques, for which – as described in [24] – both generic concepts of evolutionary algorithms and many functions to evaluate and analyze them are available. The programming language chosen for this project (and the HeuristicLab) is C# using the Microsoft .NET Framework 2.0.

Moreover, in addition to standard GP implementations, new generic concepts [3], based on evolutionary algorithms and developed to increase the quality of the produced solutions, were used and compared to the classical GP approach. These concepts are explained in detail in Section 4. Examples of the results of our test series and an overview of the operators and parameters used are given in Section 5.