OPTIMIZATION TECHNIQUES FOR SOLVING COMPLEX PROBLEMS

Edited by

Enrique Alba
University of Málaga

Christian Blum
Technical University of Catalonia

Pedro Isasi
University Carlos III of Madrid

Coromoto León
University of La Laguna

Juan Antonio Gómez
University of Extremadura

WILEY

A JOHN WILEY & SONS, INC., PUBLICATION

OPTIMIZATION TECHNIQUES FOR SOLVING COMPLEX PROBLEMS

WILEY SERIES ON PARALLEL AND DISTRIBUTED COMPUTING

Editor: Albert Y. Zomaya

A complete list of titles in this series appears at the end of this volume.

OPTIMIZATION TECHNIQUES FOR SOLVING COMPLEX PROBLEMS

Edited by

Enrique Alba
University of Málaga

Christian Blum
Technical University of Catalonia

Pedro Isasi
University Carlos III of Madrid

Coromoto León
University of La Laguna

Juan Antonio Gómez
University of Extremadura

WILEY

A JOHN WILEY & SONS, INC., PUBLICATION

Published by John Wiley & Sons, Inc., Hoboken, New Jersey.
Published simultaneously in Canada.

Library of Congress Cataloging-in-Publication Data:

Optimization techniques for solving complex problems / [edited by] Enrique Alba, Christian Blum, Pedro Isasi, Coromoto León, Juan Antonio Gómez

Includes bibliographic references and index.

ISBN 978-0-470-29332-4 (cloth)

Printed in the United States of America

10 9 8 7 6 5 4 3 2 1

Enrique Alba, *To my family*

Christian Blum, *To María and Marc*

Pedro Isasi, *To my family*

Coromoto León, *To Juana*

Juan Antonio Gómez, *To my family*

CONTENTS

CONTRIBUTORS

E. Alba, Universidad de Málaga, Dpto. de Lenguajes y Ciencias de la Computación, Málaga (Spain)

R. Aler, Universidad Carlos III de Madrid, Dpto. de Informática, Escuela Politécnica Superior, Madrid (Spain)

M. J. Blesa, Universitat Politècnia de Catalunya, Dpto. de Llenguatges i Sistemes Informàtics, Barcelona (Spain)

C. Blum, Universitat Politècnica de Catalunya, Dpto. de Llenguatges i Sistemes Informàtics, Barcelona (Spain)

J. Carretero, Universitat Politècnica de Catalunya, Dpto. d′Arquitectura de Computadors, Barcelona (Spain)

F. Chicano, Universidad de Málaga, Dpto. de Lenguajes y Ciencias de la Computación, Málaga (Spain)

C. Cotta, Universidad de Málaga, Dpto. de Lenguajes y Ciencias de la Computación, Málaga (Spain)

B. Dorronsoro, Université de Luxembourg (Luxembourg)

J. J. Durillo, Universidad de Málaga, Dpto. dé Lenguajes y Ciencias de la Computación, Málaga (Spain)

C. Estébanez, Universidad Carlos III de Madrid, Dpto. de Informática, Escuela Politécnica Superior, Madrid (Spain)

A. J. Fernández, Universidad de Málaga, Dpto. de Lenguajes y Ciencias de la Computación, Málaga (Spain)

F. Fernández, Universidad de Extremadura, Dpto. de Tecnologías de Computadores y Comunicaciones, Centro Universitario de Mérida, Mérida (Spain)

J. E. Gallardo, Universidad de Málaga, Dpto. de Lenguajes y Ciencias de la Computación, Málaga (Spain)

I. M. Galván, Universidad Carlos III de Madrid, Dpto. de Informática, Escuela Politécnica Superior, Madrid (Spain)

J. García-Nieto, Universidad de Málaga, Dpto. de Lenguajes y Ciencias de la Computación, Málaga (Spain)

A. Gómez, Centro de Investigaciones Energéticas, Medioambientales y Tecnológicas (CIEMAT), Centro Extremeño de Tecnologías Avanzadas, Trujillo (Spain)

J. A. Gómez, Universidad de Extremadura, Dpto. de Tecnologías de Computadores y Comunicaciones, Escuela Politécnica, Cáceres (Spain)

J. M. Granado, Universidad de Extremadura, Dpto. de Ingeniería de Sistemas Informáticos y Telemáticos, Escuela Politécnica, Cáceres (Spain)

J. M. Guerra, Universidad Complutense de Madrid, Dpto. de Optica, Madrid (Spain)

J. L. Guisado, Universidad de Sevilla, Dpto. de Arquitectura y Tecnología de Computadors, Sevilla (Spain)

P. Isasi, Universidad Carlos III de Madrid, Dpto. de Informática, Escuela Politécnica Superior, Madrid (Spain)

M. D. Jaraiz, Universidad de Extremadura, Dpto. de Tecnologías de Computadores y Comunicaciones, Escuela Politécnica, Cáceres (Spain)

F. Jiménez-Morales, Universidad de Sevilla, Dpto. de Física de la Materia Condensada, Sevilla (Spain)

G. Leguizamón, Universidad Nacional de San Luis, Laboratorio de Investigación y Desarrollo en Inteligencia Computacional (LIDIC), San Luis (Argentina)

C. León, Universidad de La Laguna, Dpto. de Estadística, I.O. y Computación, La Laguna (Spain)

D. Lombraña, Universidad de Extremadura, Cátedra CETA-CIEMAT, Centro Universitario de Mérida, Mérida (Spain)

F. Luna, Universidad de Málaga, Dpto. de Lenguajes y Ciencias de la Computación, Málaga (Spain)

C. Luque, Universidad Carlos III de Madrid, Dpto. de Informática, Escuela Politécnica Superior, Madrid (Spain)

G. Luque, Universidad de Málaga, Dpto. de Lenguajes y Ciencias de la Computación, Málaga (Spain)

E. Lutton, Institut National de Recherche en Informatique et en Automatique (INRIA), Orsay (France)

G. Miranda, Universidad de La Laguna, Dpto. de Estadística, I.O. y Computación, La Laguna (Spain)

G. Molina, Universidad de Málaga, Dpto. de Lenguajes y Ciencias de la Computación, Málaga (Spain)

J. M. Molina, Universidad de Málaga, Dpto. de Lenguajes y Ciencias de la Computación, Málaga (Spain)

S. Molina, Universided de Nacional San Luis, Laboratorio de Investigación y Desarrollo en Inteligencia Computacional (LIDIC), San Luis (Argentina)

A. Mochón, Universidad Nacional de Educación a Distancia (UNED), Dpto. de Economía Aplicada e Historia Económica, Madrid (Spain)

A. J. Nebro, Universidad de Málaga, Dpto. de Lenguajes y Ciencias de la Computación, Málaga (Spain)

G. Olague, Centro de Investigación Cientifica y de Educación Superior de Ensenada (CICESE), Dpto. de Ciencias Informáticas, Ensenada (México)

G. Ordóñez, Universidad Nacional de San Luis, Laboratorio de Investigación y Desarrollo en Inteligencia Computacional (LIDIC), San Luis (Argentina)

C. B. Pérez, Centro de Investigación Cientifica y de Educación Superior de Ensenada (CICESE), Dpto. de Ciencias Informáticas, Ensenada (México)

D. Quintana, Universidad Carlos III de Madrid, Dpto. de Informática, Escuela Politécnica Superior, Madrid (Spain)

C. Rodríguez, Universidad de La Laguna, Dpto. de Estadística, I. O. y Computación, La Laguna (Spain)

M. Rubio, Centro de Investigaciones Energéticas, Medioambientales y Tecnológicas (CIEMAT), Centro Extremeño de Tecnologías Avanzadas, Trujillo (Spain)

Y. Sáez, Universidad Carlos III de Madrid, Dpto. de Informática, Escuela Politécnica Superior, Madrid (Spain)

C. Salto, Universidad Nacional de La Pampa, Facultad de Ingeniería, General Pico (Argentina)

J. M. Sánchez, Universidad de Extremadura, Dpto. de Tecnologías de Computadores y Comunicaciones, Escuela Politécnica, Cáceres (Spain)

J. M. Valls, Universidad Carlos III de Madrid, Dpto. de Informática, Escuela Politécnica Superior, Madrid (Spain)

M. A. Vega, Universidad de Extremadura, Dpto. de Tecnologías de Computadores y Comunicaciones, Escuela Politécnica, Cáceres (Spain)

F. Xhafa, Universitat Politècnia de Catalunya, Dpto. de Llenguatges i Sistemes Informàtics, Barcelona (Spain)

FOREWORD

The topic of optimization, especially in the context of solving complex problems, is of utmost importance to most practitioners who deal with a variety of optimization tasks in real-world settings. These practitioners need a set of new tools for extending existing algorithms and developing new algorithms to address a variety of real-world problems. This book addresses these very issues.

The first part of the book covers many new ideas, algorithms, and techniques. These include modern heuristic methods such as genetic programming, neural networks, genetic algorithms, and hybrid evolutionary algorithms, as well as classic methods such as divide and conquer, branch and bound, dynamic programming, and cryptographic algorithms. Many of these are extended by new paradigms (e.g., new metaheuristics for multiobjective optimization, dynamic optimization) and they address many important and practical issues (e.g., constrained optimization, optimization of time series).

The second part of the book concentrates on various applications and indicates the applicability of these new tools for solving complex real-world problems. These applications include DNA sequencing and reconstruction, location of antennas in telecommunication networks, job scheduling, cutting and packing problems, multidimensional knapsack problems, and image processing, to name a few.

The third and final part of the book includes information on the possibility of remote optimization through use of the Internet. This is definitely an interesting option, as there is a growing need for such services.

I am sure that you will find this book useful and interesting, as it presents a variety of available techniques and some areas of potential applications.

Zbigniew Michalewicz

University of Adelaide, Australia
February 2008

PREFACE

This book is the result of an ambitious project to bring together various visions of many researchers in both fundamental and applied issues of computational methods, with a main focus on optimization. The large number of such techniques and their wide applicability make it worthwhile (although difficult) to present in a single volume some core ideas leading to the creation of new algorithms and their application to new real-world tasks.

In addition to researchers interested mainly in algorithmic aspects of computational methods, there are many researchers whose daily work is rather application-driven, with the requirement to apply existing techniques efficiently but having neither the time, the resources, nor the interest in algorithmic aspects. This book is intended to serve all of them, since these two points of view are addressed in most of the chapters. Since the book has these two parts (fundamentals and applications), readers may use chapters of either part to enhance their understanding of modern applications and of optimization techniques simultaneously.

Since this is an edited volume, we were able to profit from a large number of researchers as well as from new research lines on related topics that have begun recently; this is an important added value that an authored book would probably not provide to such an extent. This can easily be understood by listing the diverse domains considered: telecommunications, bioinformatics, economy, cutting, packing, cryptography, hardware, laser industry, scheduling, and many more.

We express our profound appreciation to all who have contributed a chapter to this book, since any merit the work deserves must be credited to them. Also, we thank the research groups that contributed to the book for their efforts and for their help in making this project successful. We also appreciate the support received from Wiley during the entire editing process, as well as the decisive endorsement by Professor A. Zomaya that made this idea a reality. To all, thank you very much.

Enrique Alba
Christian Blum
Pedro Isasi
Coromoto León
Juan Antonio Gómez

February 2008

PART I

METHODOLOGIES FOR COMPLEX PROBLEM SOLVING

CHAPTER 1

Generating Automatic Projections by Means of Genetic Programming

C. ESTÉBANEZ and R. ALER
Universidad Carlos III de Madrid, Spain

1.1 INTRODUCTION

The aim of inductive machine learning (ML) is to generate models that can make predictions from analysis of data sets. These data sets consist of a number of instances or examples, each example described by a set of attributes. It is known that the quality or relevance of the attributes of a data set is a key issue when trying to obtain models with a satisfactory level of generalization. There are many techniques of feature extraction, construction, and selection [1] that try to improve the representation of data sets, thus increasing the prediction capabilities of traditional ML algorithms. These techniques work by filtering nonrelevant attributes or by recombining the original attributes into higher-quality ones. Some of these techniques were created in an automatic way by means of genetic programming (GP).

GP is an evolutionary technique for evolving symbolic programs [2]. Most research has focused on evolving functional expressions, but the use of loops and recursion has also been considered [3]. Evolving circuits are also among the successes of GP [4]. In this work we present a method for attribute generation based on GP called the GPPE (genetic programming projection engine). Our aim is to evolve symbolic mathematical expressions that are able to transform data sets by representing data on a new space, with a new set of attributes created by GP. The goal of the transformation is to be able to obtain higher accuracy in the target space than in the original space. The dimensions of the new data space can be equal to, larger, or smaller than those of the original. Thus, we also intend that GPPE be used as a dimension reduction technique as

Optimization Techniques for Solving Complex Problems, Edited by Enrique Alba, Christian Blum, Pedro Isasi, Coromoto León, and Juan Antonio Gómez

well as creating highly predictive attributes. Although GPPE can either increase or reduce dimensionality, the work presented in this chapter focuses on reducing the number of dimensions dramatically while attempting to improve, or at least maintain, the accuracy obtained using the original data.

In the case of dimension reduction, the newly created attributes should contain all the information present in the original attributes, but in a more compact way. To force the creation of a few attributes with a high information content, we have established that the data in the projected space must follow a nearly linear path. To test GPPE for dimensionality reduction, we have applied it to two types of data mining domains: classification and regression. In classification, linear behavior will be measured by a fast classification algorithm based on selecting the nearest class centroid. In regression, linear behavior will be determined by simple linear regression in the projected space.

GP is very suitable for generating feature extractors, and some work has been done in this field. In the following section we overview briefly some approaches proposed in the literature. Then, in Section 1.4 we focus on GPPE, which can be used in both the classification and regression domains, and we show some experimental results in Section 1.5. We finish with our conclusions and some suggestions for future work.

1.2 BACKGROUND

There are many different constructive induction algorithms, using a wide variety of approaches. Liu et al. [1] provide a good starting point for the exploration of research into feature extraction, construction, and selection. Their book compiles contributions from researchers in this field and offers a very interesting general view. Here we discuss only works that use GP or any other evolutionary strategy, and we focus on those that are among the most interesting for us because they bear some resemblance to GPPE.

Otero et al. [5] use typed GP for building feature extractors. The functions are arithmetic and relational operators, and the terminals are the original (continuous) attributes of the original data set. Each individual is an attribute, and the fitness function uses the information gain ratio. Testing results using C4.5 show some improvements in some UCI domains. In Krawiec's work [6], each individual contains several subtrees, one per feature. C4.5 is used to classify in feature space. Their work allows us to cross over subtrees from different features.

Shafti and Pérez [7] discuss the importance of applying GA as a global search strategy for constructive induction (CI) methods and the advantages of using these strategies instead of using classic greedy methods. They also present MFE2/GA, a CI method that uses GA to search through the space of different combination of attribute subsets and functions defined over them. MFE2/GA uses a nonalgebraic form of representation to extract complex interactions between the original attributes of the problem.

Kuscu [8] introduced the GCI system. GCI is a CI method based on GP. It is similar to GPPE in the sense that it uses basic arithmetic operators and the fitness is computed measuring the performance of an ML algorithm (a quick-prop net) using the attributes generated. However, each individual represents a new attribute instead of a new attribute set. In this way, GCI can only generate new attributes that are added to the original ones, thus increasing the dimensionality of the problem. The possibility of reducing the number of attributes of the problem is mentioned only as possible and very interesting future work.

Hu [9] introduced another CI method based on GP: GPCI. As in GCI, in GPCI each individual represents a newly generated attribute. The fitness of an individual is evaluated by combining two functions: an absolute measure and a relative measure. The absolute measure evaluates the quality of a new attribute using a gain ratio. The relative measure evaluates the improvement of the attribute over its parents. A function set is formed by two Boolean operators: AND and NOT. GPCI is applied to 12 UCI domains and compared with two other CI methods, achieving some competitive results.

Howley and Madden [10] used GP to evolve kernels for support vector machines. Both scalar and vector operations are used in the function set. Fitness is computed from SVM performance using a GP-evolved kernel. The hyperplane margin is used as a tiebreaker to avoid overfitting. Although evolved kernels are not forced by the fitness function to satisfy standard properties (such as Mercer's property) and therefore the evolved individuals are not proper kernels, results in the testing data sets are very good compared to those of standard kernels. We believe that evolving proper distance functions or kernels is difficult because some properties (such as transitivity or Mercer's property) are not easy to impose on the fitness computation.

Eads et al. [11] used GP to construct features to classify time series. Individuals were made of several subtrees returning scalars (one per feature). The function set contained typical signal-processing primitives (e.g., convolution), together with statistical and arithmetic operations. SVM was then used for classification in feature space. Cross-validation on training data was used as a fitness function. The system did not outperform the SVM, but managed to reduce dimensionality. This means that it constructed good features to classify time series. However, only some specific time series domains have been tested. Similarly, Harvey et al. [12] and Szymanski et al. [13] assemble image-processing primitives (e.g., edge detectors) to extract multiple features from the same scene to classify terrains containing objects of interest (i.e., golf courses, forests, etc.). Linear fixed-length representations are used for the GP trees. A Fisher linear discriminant is used for fitness computation. Results are quite encouraging but are restricted to image-processing domains.

Results from the literature show that, in general, the GP projection approach has merit and obtains reasonable results, but that more research is needed. New variations of the idea and more domains should be tested. Regression problems are not considered in any of the works reviewed, and we believe that a lot more research on this topic is also needed.

1.3 DOMAINS

In this chapter we are interested in applying GPPE to two classical prediction tasks: classification and regression. We have used bankruptcy prediction as the classification domain and IPO underpricing prediction as the regression domain.

1.3.1 Bankruptcy Prediction

In general terms, the bankruptcy prediction problem attempts to determine the financial health of a company, and whether or not it will soon collapse. In this chapter we use a data set provided and described by Vieira et al. [14]. This data set studies the influence of several financial and economical variables on the financial health of a company. It includes data on 1158 companies, half of which are in a bankruptcy situation (class 0) and the rest of which have good financial health (class 1). Companies are characterized by 40 numerical attributes [14]. For validation purposes we have divided the data set into a training set and a test set, containing 766 (64%) and 400 (36%) instances, respectively.

1.3.2 IPO Underpricing Prediction

IPO underpricing is an interesting and important phenomenon in the stock market. The academic literature has long documented the existence of important price gains in the first trading day of initial public offerings (IPOs). That is, there is usually a big difference between the offering price and the closing price at the end of the first trading day. In this chapter we have used a data set composed of 1000 companies entering the U.S. stock market for the first time, between April 1996 and November 1999 [15]. Each company is characterized by seven explicative variables: underwriter prestige, price range width, price adjustment, offer price, retained stock, offer size, and relation to the tech sector. The target variable is a real number which measures the profits that could be obtained by purchasing the shares at the offering price and selling them soon after dealing begins. For validation purposes we have divided the data set into a training set and a test set, containing 800 (80%) and 200 (20%) instances, respectively.

1.4 ALGORITHMIC PROPOSAL

In this section we describe the genetic programming projection engine (GPPE). GPPE is based on GP. Only a brief summary of GP is provided here. The reader is encouraged to consult Koza's book [2] for more information.

GP has three main elements:

1. A population of individuals, in this case, computer programs
2. A fitness function, used to measure the goodness of the computer program represented by the individual