Available online at www.sciencedirect.com
ScienceDirect
Physica A 370 (2006) 104–108
PHYSICA A
www.elsevier.com/locate/physa

Performance of genetic programming to extract the trend in noisy data series

A. Borrelli[a], I. De Falco[b], A. Della Cioppa[c], M. Nicodemi[a,*], G. Trautteur[a]

[a]Department of Physical Sciences, University of Naples "Federico II", INFN, Naples, Italy
[b]ICAR, National Research Council of Italy, Naples, Italy
[c]Natural Computation Lab, DIIIE, University of Salerno, Fisciano (SA), Italy

Available online 11 May 2006

Abstract

In this paper an approach based on genetic programming for forecasting stochastic time series is outlined. To obtain a suitable test-bed some well-known time series are dressed with noise. The GP approach is endowed with a multiobjective scheme relying on statistical properties of the faced series, i.e., on their momenta. Finally, the method is applied to the MIB30 Index series.

Keywords: Multiobjective genetic programming; Stochastic time series

1. Introduction

The prediction of spatio-temporal patterns is relevant in a variety of applications in fields ranging from engineering to economics. Different methods have been investigated over the years [1]. Recently, the flexibility of evolutionary search methods is being exploited, and the genetic programming (GP) framework [2] has been applied in various fields with excellent results and to forecasting as well [3–5]. The goal is to exploit GP's flexible tree structure for building a time series prediction model. The motivation of this choice is that GP has many advantages over the other classically used forecasting models: (i) generation of explicit model representations amenable to easy human comprehension, (ii) automatic discovering of the model structure from the given data, (iii) adaptive evolutionary search that allows to escape trapping in suboptimal, unsatisfactory local solutions, (iv) no need for specific knowledge. GP does not guarantee to find the global optimum, nonetheless it usually allows to retrieve a satisfactory solution in a reasonable computation time.

2. Genetic programming

GP is a heuristic optimization technique relying on the simulation in a computer of the mechanisms typical of the natural evolution. GP deals with a population of solutions whose genotypes are programs represented

*Corresponding author.
E-mail address: nicodem@na.infn.it (M. Nicodemi).

0378-4371/$ - see front matter © 2006 Elsevier B.V. All rights reserved.
doi:10.1016/j.physa.2006.04.025

by tree structures, differing in shape and size, in which the internal nodes denote the functions whereas the leaves are terminals (constants and variables). Once loaded the problem data (e.g. time series values), GP procedure is

(1) generate at random an initial population of composite functions representing potential solutions (e.g. forecasting models);
(2) evaluate the individuals by using the *fitness function*;
(3) at each generation repeat until a new population is filled:
   - choose an operator among *crossover*, *copy* and *mutation*;
   - *select* a suitable number of individuals in current population;
   - apply the chosen operator to produce the offspring;
   - insert the offspring in the new population;
   - evaluate the models by means of *fitness*;
(4) repeat step (3) until a maximum number of generations is reached.

Selection is a mechanism which should allow to choose the individuals which will undergo reproduction among those in the current population.

In *crossover*, two parents are selected and a subtree is picked in each one. Then crossover swaps the nodes and their relative subtrees from one parent to the other. If a too-deep offspring is generated, it is replaced by one of the parents.

In *copy*, an individual in the current population is chosen, and then this operator copies it without changes into the new population.

In *mutation* a tree is selected, then mutation randomly picks a node, and a new subtree is generated starting from it. If this replacement violates the depth limit, mutation just reproduces the original tree into the new generation.

GP for data series. The aim is the implementation of a GP able to automatically provide the modeling of a time series and its prediction.

The evolving population is constituted in this case by programs representing the forecasting models as trees with variable depth. Each model is composed by elementary functions and terminals. The function set chosen is $\{+, -, *, /, \sin, \cos, \exp, \mathrm{rlog}, \mathrm{abs}, \mathrm{sqrt}\}$. The terminal set chosen contains the independent variable x and a symbol N representing an integer in a range $[D_{\min}, D_{\max}]$.

If S is the model represented by a generic individual and f is the function standing for a time series with n points, the fitness we choose is the following:

$$\phi = \|S - f\|^2 = \frac{1}{n}\sum_{i=1}^{n}(S(i) - f(i))^2 \tag{1}$$

3. GP results

The experiments have concerned a wide set of 50 known functions typically used in literature and the GP parameters are given the "optimal" values found in the literature as well.

GP on known test functions. The first quantity we show is the average, over our test set of functions, of the adjusted fitness, Φ, as a function of time, t, i.e., the number of generations starting from the initial random population:

$$\Phi(t) = \langle 1 + \|S_b(t) - f\|^2\rangle^{-1}, \tag{2}$$

where $S_b(t)$ is the best individual at generation t and f is the function to fit.

$\Phi(t)$ increases with time and can be well fitted by a stretched exponential: $\Phi_\infty - \Delta\Phi \exp[-(t/\tau)^\beta]$. In the present case, we find that the asymptotic level is $\Phi_\infty = 0.55$, and the time to approach it is $\tau \simeq 150$ ($\Delta\Phi \simeq 0.18$, and $\beta \simeq 0.3$). Interestingly, Φ_∞ is definitely below one (i.e., its maximal possible value found for perfect fitting