Available online at www.sciencedirect.com
SCIENCE DIRECT®
Neurocomputing 64 (2005) 529–536
NEUROCOMPUTING
www.elsevier.com/locate/neucom

Letters

A fast fixed-point algorithm for complexity pursuit

Zhenwei Shi[a,b,*], Huanwen Tang[a], Yiyuan Tang[b,c,d]

[a]*Institute of Computational Biology and Bioinformatics, Dalian University of Technology, Dalian 116023, P.R. China*
[b]*Institute of Neuroinformatics, Dalian University of Technology, Dalian 116023, P.R. China*
[c]*Laboratory of Visual Information Processing, The Chinese Academy of Sciences, Beijing 100101, P.R. China*
[d]*Key Lab for Mental Health, The Chinese Academy of Sciences, Beijing 100101, P.R. China*

Received 1 November 2004; received in revised form 27 November 2004; accepted 28 November 2004
Communicated by R.W. Newcomb
Available online 19 January 2005

Abstract

Complexity pursuit is a recently developed algorithm using the gradient descent for separating interesting components from time series. It is an extension of projection pursuit to time series data and the method is closely related to blind separation of time-dependent source signals and independent component analysis (ICA). In this paper, a fixed-point algorithm for complexity pursuit is introduced. The fixed-point algorithm inherits the advantages of the well-known FastICA algorithm in ICA, which is very simple, converges fast, and does not need choose any learning step sizes.

Keywords: Independent component analysis; Blind source separation; Complexity pursuit; Projection pursuit; Time series

*Corresponding author. Institute of Computational Biology and Bioinformatics, Dalian University of Technology, Dalian 116023, P.R. China.
E-mail addresses: szw1977@yahoo.com.cn (Z. Shi), yy2100@163.net (Y. Tang).

0925-2312/$ - see front matter
doi:10.1016/j.neucom.2004.11.028

1. Introduction

Independent component analysis (ICA) has been widely applied to blind source separation, blind deconvolution, and feature extraction, and so on. The model of ICA consists of mixing independent random variables, usually linearly [1,2,4,5,9,11,13,18,19,21,22]. In many applications, however, what is mixed is not random variables but time signals, or time series. ICA in its basic form ignores any time structure and uses only the nongaussianity criteria. It is to be noted that under certain restrictions, it is also possible to estimate the independent components using the time-dependency information alone [3,15,16,20]. However, a principled way of combining both of these estimation criteria (nongaussianity and time-correlations) has been introduced by Hyvärinen in the complexity pursuit algorithm [10]. Complexity pursuit is an extension of projection pursuit [6] to time series, that is, signals with time structure. The goal is to find projections of time series that have interesting structure, defined using criteria related to Kolmogoroff complexity [17] or coding length. Time series which have the lowest coding complexity are considered the most interesting. Hyvärinen derived a simple approximation of Kolmogoroff complexity that takes into account both the nongaussianity and the autocorrelations of the time series. He developed a gradient ascent algorithm for its approximative optimization. The method is closely related to blind separation of time-dependent source signals and ICA.

In this paper, motivated by the work of Hyvärinen, we propose a fixed-point algorithm for complexity pursuit. The fixed-point algorithm inherits the advantages of the well-known FastICA algorithm [9,11] in ICA.

2. Complexity pursuit

Assume that the observed data are multivariate time series $x(t)$, that is, a vector of time signals. The basic idea in the complexity pursuit is to find projections $w^T x(t)$ such that the Kolmogoroff complexity of the projection is minimized. We search for projections that can be easily coded in the complexity pursuit. This is a general-purpose measure and is probably connected to information-processing principles used in the brain [10].

First, we derive an approximation of the Kolmogoroff complexity of a scalar signal $y(t)(t = 1, \ldots, T)$ along similar lines as Hyvärinen [10]. For simplicity, the signal is assumed to have zero mean and unit variance.

We consider predictive coding of the signal. The value $y(t)$ is predicted from the preceding values by some function to be specified:

$$\hat{y}(t) = f(y(t-1), \ldots, y(1)). \tag{1}$$

To code the actual value $y(t)$, the residual

$$\delta y(t) = y(t) - \hat{y}(t) \tag{2}$$