Available online at www.sciencedirect.com
ScienceDirect
Applied Soft Computing 7 (2007) 471–479
Applied Soft Computing
www.elsevier.com/locate/asoc

Genetic programming neural networks: A powerful bioinformatics tool for human genetics

Marylyn D. Ritchie [a,*], Alison A. Motsinger [a], William S. Bush [a], Christopher S. Coffey [b], Jason H. Moore [c]

[a] *Center for Human Genetics Research, Department of Molecular Physiology and Biophysics, Vanderbilt University, 519 Light Hall, Nashville, TN 37232, United States*
[b] *Department of Biostatistics, University of Alabama at Birmingham, Ryals Public Health Bldg., Rm. 327M, Birmingham, AL, 35294, United States*
[c] *Computational Genetics Laboratory, Department of Genetics, 706 Rubin Bldg., HB7937, Dartmouth-Hitchcock Medical Center, One Medical Center Dr. Lebanon, NH 03756, United States*

Received 28 February 2005; accepted 16 January 2006

Abstract

The identification of genes that influence the risk of common, complex disease primarily through interactions with other genes and environmental factors remains a statistical and computational challenge in genetic epidemiology. This challenge is partly due to the limitations of parametric statistical methods for detecting genetic effects that are dependent solely or partially on interactions. We have previously introduced a genetic programming neural network (GPNN) as a method for optimizing the architecture of a neural network to improve the identification of genetic and gene–environment combinations associated with disease risk. Previous empirical studies suggest GPNN has excellent power for identifying gene–gene and gene–environment interactions. The goal of this study was to compare the power of GPNN to stepwise logistic regression (SLR) and classification and regression trees (CART) for identifying gene–gene and gene–environment interactions. SLR and CART are standard methods of analysis for genetic association studies. Using simulated data, we show that GPNN has higher power to identify gene–gene and gene–environment interactions than SLR and CART. These results indicate that GPNN may be a useful pattern recognition approach for detecting gene–gene and gene–environment interactions in studies of human disease.

Keywords: Nueral networks; Genetic programming; Bioinformatics; Epistasis; Gene–gene interactions

1. Introduction

One goal of genetic epidemiology is to identify genes associated with common, complex multifactorial diseases. Success in achieving this goal will depend on a research strategy that recognizes and addresses the importance of interactions among multiple genetic and environmental factors in the etiology of diseases such as essential hypertension [8,15]. One traditional approach to modeling the relationship between discrete predictors such as genotypes and discrete clinical outcomes is logistic regression [7]. Logistic regression is a parametric statistical approach for relating one or more independent or explanatory variables (e.g. genotypes) to a dependent or outcome variable (e.g. disease status) that follows a binomial distribution. However, as reviewed by Moore and Williams [15], the number of possible interaction terms grows exponentially as each additional main effect is included in the logistic regression model. Thus, logistic regression is limited in its ability to deal with interactions involving many factors. Having too many independent variables in relation to the number of observed outcome events is a well-recognized problem [3,17] and is an example of the curse of dimensionality [2].

In response to this limitation, Ritchie et al. [19] developed a genetic programming optimized neural network (GPNN). Neural networks (NN) have been utilized in genetic epidemiology, however, with little success. A potential weakness in the previous NN applications is the poor specification of NN

* Corresponding author. Tel.: +1 615 343 6549; fax: +1 615 343 8619.
E-mail addresses: ritchie@chgr.mc.vanderbilt.edu (M.D. Ritchie), motsinger@chgr.mc.vanderbilt.edu (A.A. Motsinger), wbush@chgr.mc.vanderbilt.edu (W.S. Bush), CCoffey@ms.soph.uab.edu (C.S. Coffey), Jason.H.Moore@dartmouth.edu (J.H. Moore).

1568-4946/$ – see front matter
doi:10.1016/j.asoc.2006.01.013

architecture. GPNN was developed in an attempt to improve upon the trial-and-error process of choosing an optimal architecture for a pure feed-forward back propagation neural network. The GPNN optimizes the inputs from a larger pool of variables, the weights, and the connectivity of the network including the number of hidden layers and the number of nodes in the hidden layer. Thus, the algorithm attempts to generate optimal neural network architecture for a given data set. This is an advantage over the traditional back propagation NN in which the inputs and architecture are pre-specified and only the weights are optimized.

Although previous empirical studies suggest GPNN has excellent power for identifying gene–gene interactions, a comparison of GPNN with traditional statistical methods has not yet been performed. The goal of the present study was to compare the power of GPNN to that of stepwise logistic regression (SLR) and classification and regression trees (CART) for identifying gene–gene and gene–environment interactions using data simulated from a variety of interaction models. This study is motivated by the number of studies in human genetics where SLR and CART have been applied. We wanted to determine if GPNN is more powerful than the status quo in the field. We find that GPNN has higher power to detect gene–gene and gene–environment interactions than stepwise logistic regression and classification and regression trees. These results demonstrate that GPNN may be an important pattern recognition tool for future studies in genetic epidemiology.

2. Methods

2.1. A genetic programming neural network approach

GPNN was developed to improve upon the trial-and-error process of choosing an optimal architecture for a pure feed-forward back propagation neural network (NN) [19]. Optimization of NN architecture using genetic programming (GP) was first proposed by Koza and Rice [9]. The goal of this approach is to use the evolutionary features of genetic programming to evolve the architecture of an NN. The use of binary expression trees allow for the flexibility of the GP to evolve a tree-like structure that adheres to the components of an NN. Fig. 1 shows an example of a binary expression tree representation of an NN generated by GPNN. The GP is constrained such that it uses standard GP operators but retains the typical structure of a feed-forward NN. While GP could be implemented without constraints, the goal was to evolve NN since they were being explored as a tool for genetic epidemiology. Thus, we wanted to make an improvement to a method already being used. A set of rules is defined prior to network evolution to ensure that the GP tree maintains a structure that represents an NN. The rules used for this GPNN implementation are consistent with those described by Koza and Rice [9]. The flexibility of the GPNN allows optimal network architectures to be generated that consist of the appropriate inputs, connections, and weights for a given data set.

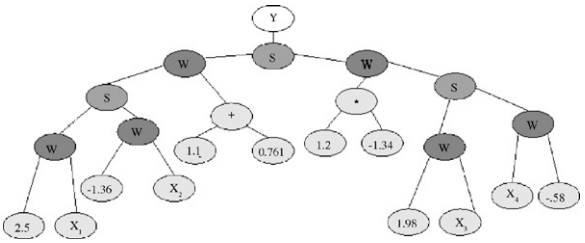

Fig. 1. An example of an NN evolved by GPNN. The Y is the output node, S indicates the activation function, W indicates a weight, and X_1–X_4 are the NN inputs.

The GPNN method has been described in detail [19]. The steps of the GPNN method are shown in Fig. 2 and described in brief as follows. First, GPNN has a set of parameters that

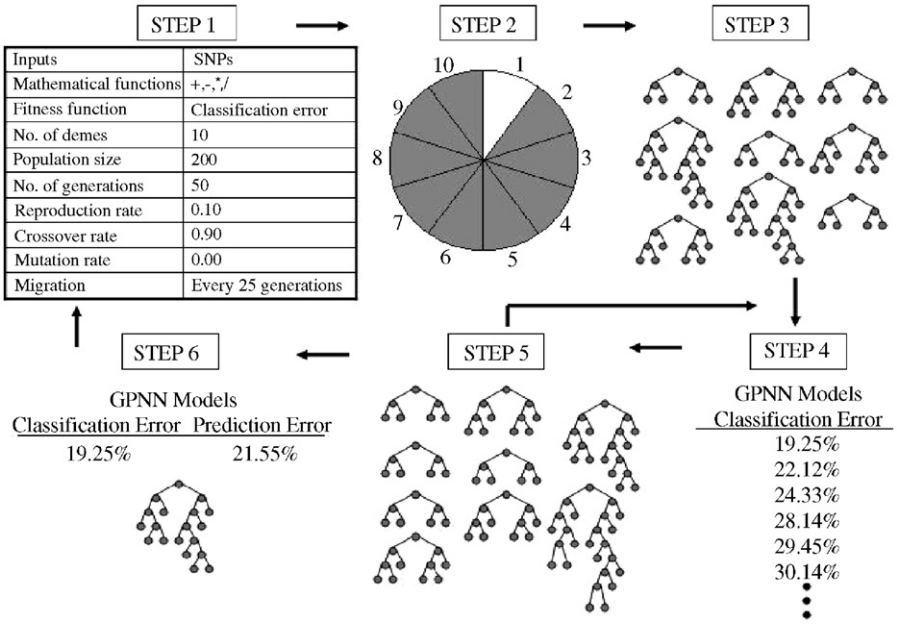

Fig. 2. The steps of the GPNN algorithm.