Available online at www.sciencedirect.com
SCIENCE DIRECT®
ELSEVIER
Computational Statistics & Data Analysis 48 (2005) 125–138
COMPUTATIONAL STATISTICS & DATA ANALYSIS
www.elsevier.com/locate/csda

DPLS and PPLS: two PLS algorithms for large data sets

Ruy L. Milidiú, Raúl P. Rentería*

Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro, Rua Marquês de São Vicente 225, Gávea, Rio de Janeiro, Brazil

Received 13 October 2003; received in revised form 13 October 2003

Available online 18 November 2003

Abstract

Two enhancements to the PLS regression algorithm are presented. The first, direct PLS (DPLS), offers a direct approximate formulation for the calculation of the required eigenvectors when dealing with more than one dependent variable. The second enhancement is parallel PLS (PPLS), a parallel version of the PLS algorithm restricted to the case of only one dependent variable for the regression model. In the experiments, DPLS shows a 40% faster running time, while the PPLS produces a speedup of 3 for the first four machines in a computer cluster architecture.

Keywords: PLS; Parallelism; DPLS; PPLS; NIPALS; Large data set

1. Introduction

The PLS algorithm (Wold, 1966; Wold et al., 1983) has been widely used as a chemometric tool for near-infrared spectral analysis (Haaland and Thomas, 1988). The simplicity of the technique and robustness of the generated model also make the partial least-squares approach a powerful tool for factor analysis, being applied to many other areas such as process monitoring, marketing analysis and image processing (Morineau and Tenenhaus, 1999; Milidiú et al., 1999). In this paper we propose two enhancements to PLS regression aimed at efficiency. The first is a direct PLS (DPLS) formulation for the case of more than one dependent variable, also referred as PLS2. The second

* Corresponding author. PUC-Rio, Fundação Padre Leonel Franca, 11 andar, Rua Marquês de São Vicente 225, Gávea, 22453900 Rio de Janeiro, Brazil.
E-mail addresses: milidiu@inf.puc-rio.br (R.L. Milidiú), renteria@inf.puc-rio.br (R.P. Rentería).

0167-9473/$ - see front matter © 2003 Elsevier B.V. All rights reserved.
doi:10.1016/j.csda.2003.10.006

is a parallel PLS (PPLS) devised for large data sets in the case of only one dependent variable, also known as PLS1.

To measure the performance of DPLS, we report some experiments with nine (Kalivas, 1997) data sets. This direct PLS model shows a competitive quality: similar prediction errors and corresponding number of factors are observed. Moreover, since it is not a convergence-dependent technique it shows a 40% faster running time. Regarding PPLS performance, it shows with a relatively small data set an efficiency above 74% when using four nodes of our computer cluster.

In Section 2, we present the DPLS formulation. In Section 3, PPLS, our parallel approach is presented.

2. DPLS

2.1. Modeling

In classical PLS modeling, the eigenvectors of the mixed independent and dependent variables matrix $X^\top YY^\top X$ must be computed in the case of two or more dependent variables (PLS2). This can be accomplished, for example, using NIPALS, the power method (Wu et al., 1997) or neural techniques such as Hebbian learning (Haykin, 1999). DPLS (Milidiú and Rentería, 2001; Milidiú et al., 2001), or DPLS, provides a new method, yet approximate, for this calculation not relying on any convergence criteria.

2.2. The algorithm

Let G denote the matrix $X^\top YY^\top X$. For each factor, we must find the eigenvector w of G associated with the largest eigenvalue.

When Y has only one dependent variable ($l = 1$), then G has rank 1 and a corresponding non-normalized eigenvector is simply $X^\top Y$. This eigenvector corresponds to the unique non-zero eigenvalue $(X^\top Y)^\top (X^\top Y)$. On the other hand, when $l \geqslant 2$ that is not true anymore. Nevertheless, one can decompose G as follows:

$$G = X^\top (Y_1 Y_1^\top + Y_2 Y_2^\top + \cdots + Y_l Y_l^\top) X$$

and then,

$$G = X^\top Y_1 Y_1^\top X + \cdots + X^\top Y_l Y_l^\top X, \tag{1}$$

where Y_i $(1 \leqslant i \leqslant l)$ corresponds to the ith column of Y. For a simpler notation, we write (1) as

$$G = G_1 + G_2 + \cdots + G_l,$$

where $G_i = X^\top Y_i Y_i^\top X$ for $i = 1, \ldots, l$.

In order to find w, the power method suggests that one multiplies an initial random vector by G and normalize it until convergence. It is acceptable that among all matrices G_i, the eigenvector $w_{(1)}$ with the largest eigenvalue $\lambda_{(1)}$ will have a greater influence