Computational Statistics and Data Analysis xx (xxxx) xxx–xxx

Contents lists available at ScienceDirect

Computational Statistics and Data Analysis

journal homepage: www.elsevier.com/locate/csda

Q1 Estimating random-intercept models on data streams

Q2 L. Ippel *,[1], M.C. Kaptein, J.K. Vermunt

Tilburg University, Warandelaan 2, PO Box 90153, 5000LE Tilburg, The Netherlands

ARTICLE INFO

Article history:
Received 12 October 2015
Received in revised form 14 June 2016
Accepted 17 June 2016
Available online xxxx

Keywords:
Data streams
Expectation–Maximization algorithm
Multilevel models
Online learning
Random-intercept model

ABSTRACT

Multilevel models are often used for the analysis of grouped data. Grouped data occur for instance when estimating the performance of pupils nested within schools or analyzing multiple observations nested within individuals. Currently, multilevel models are mostly fit to static datasets. However, recent technological advances in the measurement of social phenomena have led to data arriving in a continuous fashion (i.e., data streams). In these situations the data collection is never "finished". Traditional methods of fitting multilevel models are ill-suited for the analysis of data streams because of their computational complexity. A novel algorithm for estimating random-intercept models is introduced. The Streaming EM Approximation (SEMA) algorithm is a fully-online (row-by-row) method enabling computationally-efficient estimation of random-intercept models. SEMA is tested in two simulation studies, and applied to longitudinal data regarding individuals' happiness collected continuously using smart phones. SEMA shows competitive statistical performance to existing static approaches, but with large computational benefits. The introduction of this method allows researchers to broaden the scope of their research, by using data streams.

1. Introduction

In social sciences we often encounter grouped data, such as pupils grouped within school classes (e.g., Barrett et al., 2013), multiple observations grouped within individuals (Killingsworth and Gilbert, 2010), or voters grouped within geographical regions (Gelman, 2007). Such data are typically analyzed using multilevel (or hierarchical) models in which batches of group-level parameters are treated as randomly drawn from an underlying distribution. In this paper we will use the formulation of "observations nested within individuals", although the method we present does not restrict itself to this type of nesting.

Multilevel models have various advantages over more traditional methods of analysis, such as aggregated analysis, in which the within-group structure is ignored, or group-specific analysis, in which information about the other groups is ignored. That is, they

1. contain fewer parameters than group-specific models,
2. allow for generalization of results to a wider population of groups, and
3. allow information to be shared between groups (Raudenbush and Bryk, 2002; Steenbergen and Jones, 2002).

* Corresponding author.
E-mail addresses: G.J.E.Ippel@tilburguniversity.edu (L. Ippel), m.kaptein@tilburguniversity.edu (M.C. Kaptein), j.k.vermunt@tilburguniversity.edu (J.K. Vermunt).

[1] Online supplementary material is available (see Appendix A).

http://dx.doi.org/10.1016/j.csda.2016.06.008
0167-9473/

Please cite this article in press as: Ippel, L., et al., Estimating random-intercept models on data streams. Computational Statistics and Data Analysis (2016), http://dx.doi.org/10.1016/j.csda.2016.06.008

The latter property in particular makes multilevel analysis interesting when the focus is on obtaining group-level predictions, since multilevel modeling yields smaller out-of-sample prediction error than predictions derived from either an aggregate or a group-specific analysis (see e.g., Morris and Lysy, 2012).

Current (maximum-likelihood) methods for fitting multilevel models use iterative algorithms such as Newton–Raphson or Expectation–Maximization (EM, Dempster et al., 1977) to maximize the likelihood. Alternatively, but not considered in this paper, one could use a Bayesian framework with MCMC sampling (for more details see, e.g., Browne and Goldstein, 2010). However, each of these methods requires multiple passes through the full dataset to obtain parameter estimates. Even though fitting a multilevel model once, on a moderately sized dataset does often not require excessive computation time, such ways of fitting multilevel models can become infeasible when a dataset is extremely large, or in the situation where the data collection is never "finished" because more data present themselves over time.

Recent technological developments have, however, led to the increased availability of these so-called data streams: i.e., datasets which are continuously augmented with new data points. Such data streams often have a grouped (or nested) structure. Examples include fraud detection using credit card transactions, where transactions are nested within credit cards (Patidar and Sharma, 2011), telephone communication analysis, where calls are nested within telephone registrations (Cortes et al., 2000), and consumer behavior tracking in e-commerce, where purchased items or visited web pages are nested within customers (Lee et al., 2001). In order to obtain up-to-date predictions of the individual-level effects, the parameters of the model of interest should be updated as data points come in, and the updated model parameters should be used for prediction purposes. When applied to streaming data, these traditional methods have to repeatedly cycle through all available data points, each time a new data point arrives, in order to obtain up-to-date parameter estimates. Additionally, even if the dataset is no longer augmented, but static and (extremely) large, it is often computationally preferable to analyze the dataset in smaller batches, or even a data point at a time (Ng and McLachlan, 2003; Thiesson et al., 2001). We propose an adaption of the EM algorithm for the estimation of random-intercept models, to resolve the problem of analyzing grouped data in a data stream or when the dataset is extremely large.

The resulting Streaming EM Approximation algorithm (henceforth referred to as SEMA) falls within the framework of online learning methods (Gaber et al., 2005). A key feature of online learning is that the data are summarized into a few summary statistics which contain all relevant information of previous data points (Opper, 1998). SEMA is an approximate EM method, because unlike the EM algorithm which uses all the data to update the model parameters, we only use a single data point, some summary statistics on the individual level, and the previous estimates of the model parameters, to update the model parameters. Because SEMA does not require all the data to be in memory, SEMA is more appropriate to deal with data streams than the conventional EM algorithm.

Related methods for speeding up the EM algorithm have been proposed for dealing with large (static) datasets, for example, Berlinet and Roland (2012) discussed methods to speed up the convergence rate of the conventional EM algorithm. Wolfe et al. (2008) presented an (offline) parallel version of the EM algorithm and McLachlan and Peel (2000, ch. 12) described various possible adaptations of EM methods for large datasets. Various online adaptations of the EM algorithm for different applications have also been proposed, for example, for mixture models (see, e.g., Cappé and Moulines, 2009; Liu et al., 2006; McLachlan and Peel, 2000; Wolfe et al., 2008) and for latent variable models (Cappé and Moulines, 2009). Instead of speeding up the EM algorithm, Steiner and Hudec (2007) proposed a method to scale down the data prior to using the EM algorithm. We add to this existing literature by proposing an EM approximation for the estimation of models based on data streams consisting of *dependent* observations. The method we propose stores information on the level of individuals, instead of the level of observations, and updates the estimates in a single pass over the data, making it suitable for both data streams and extremely large datasets.

The remainder of this article is organized as follows. In the next section, we illustrate the computational advantages of streaming estimation using the simple example of the estimation of a sample mean. Next, we discuss the estimation of random-intercept models using the EM algorithm, and show how this algorithm can be modified into a streaming version, leading to SEMA. Subsequently we evaluate SEMA in two simulation studies. In the first simulation study we evaluate the accuracy of the estimates of the model parameters, and of the individual-level effects. In the second study we evaluate three alternative implementations of SEMA to improve the estimates both of the model parameters and of the individual-level effects. The first alternative uses a small part of the data to obtain better starting values, the second implementation cycles through all individuals at given intervals, and the last implementation is a combination of the previous two. In Section 5 we illustrate the use of SEMA in an application using real data on respondents' happiness, in which nested data, collected using a smart-phone application, "arrived" in a stream. In Section 6 we detail some theoretical characteristics of SEMA, and we discuss a convergence diagnostic to evaluate the estimated model parameters of SEMA. In the following section, we extend the random-intercept model to include additional fixed covariates. The last section discusses the main results of the simulation studies and presents directions for future work.

2. From offline to online data analysis

Before introducing SEMA, we first explain the key changes involved when moving from the *offline* analysis of static datasets to the *online* analysis of data streams. This conceptual shift is easily illustrated by examining the computation

Please cite this article in press as: Ippel, L., et al., Estimating random-intercept models on data streams. Computational Statistics and Data Analysis (2016), http://dx.doi.org/10.1016/j.csda.2016.06.008