Biometrika (2023), **110**, 4, *pp.* 841–858 https://doi.org/10.1093/biomet/asad010
Advance Access publication 20 February 2023

Statistical inference for streamed longitudinal data

By LAN LUO
Department of Statistics and Actuarial Science, University of Iowa, 241 Schaeffer Hall, Iowa City, Iowa 52242, U.S.A.
lan-luo@uiowa.edu

JINGSHEN WANG
Division of Biostatistics, University of California, Berkeley, 2121 Berkeley Way, Berkeley, California 94720, U.S.A.
jingshenwang@berkeley.edu

and EMILY C. HECTOR
Department of Statistics, North Carolina State University, 2311 Stinson Drive, Raleigh, North Carolina 27695, U.S.A.
ehector@ncsu.edu

Summary

Modern longitudinal data, for example from wearable devices, may consist of measurements of biological signals on a fixed set of participants at a diverging number of time-points. Traditional statistical methods are not equipped to handle the computational burden of repeatedly analysing the cumulatively growing dataset each time new data are collected. We propose a new estimation and inference framework for dynamic updating of point estimates and their standard errors along sequentially collected datasets with dependence, both within and between the datasets. The key technique is a decomposition of the extended inference function vector of the quadratic inference function constructed over the cumulative longitudinal data into a sum of summary statistics over data batches. We show how this sum can be recursively updated without the need to access the whole dataset, resulting in a computationally efficient streaming procedure with minimal loss of statistical efficiency. We prove consistency and asymptotic normality of our streaming estimator as the number of data batches diverges, even as the number of independent participants remains fixed. Simulations demonstrate the advantages of our approach over traditional statistical methods that assume independence between data batches. Finally, we investigate the relationship between physical activity and several diseases through analysis of accelerometry data from the National Health and Nutrition Examination Survey.

Some key words: Generalized method of moments; Online learning; Quadratic inference function; Scalable computing; Serial dependence.

1. Introduction

Traditionally, longitudinal studies have collected a small number of repeated measurements from a relatively large number of individuals, with the ultimate goal of drawing

statistical inference for this ever-growing set of participants. With the advent of modern technologies such as smartphones and wearable devices, the data collection paradigm has shifted to an infinite-horizon setting in which data are collected on a fixed number of participants in perpetuity. This new setting offers the possibility of discovering new patterns in human daily life specific to a set of individuals, opening the door to targeted biomedical interventions for population subgroups.

A typical example is that of wearable device data which are frequently uploaded to a smartphone application and summarized into a few health metrics such as steps and distance walked, change in heart rate over time and sleep history. A biomedical research question might focus on the relationship between physical activity and covariates such as phenotype across a fixed set of users, and pose a parametric model to study this relationship. Each time users upload data to the application, new data become available for answering this research question, but the memory and computational burden of storing and analysing the entire cumulative dataset, consisting of all observations up to the latest upload, can be astronomical. Instead of reanalysing the entire dataset each time new data become available, it is preferable to update parameter estimates from previous batches of data with the new batch in a computationally efficient approach.

Numerous statistical and computational challenges arise when considering intensively measured longitudinal data on a limited set of participants. The sheer size and complexity of the data frequently prohibit statistical analyses of the entire dataset because of computational or modelling challenges. Distributed and online approaches have gained popularity in the statistical community as viable alternatives to whole-data approaches, with numerous solutions proposed for independent and dependent data alike (Xie & Singh, 2013; Zhou & Song, 2017; Jordan et al., 2019; Hector & Song, 2020; Duan et al., 2022; Luo & Li, 2022; Hector et al., 2023). Technically speaking, the challenge in infinite-horizon settings is to derive updating rules for point estimates and measures of uncertainty, such as standard errors, across sequentially collected datasets, referred to as data batches, where both within- and between-batch dependences exist. The online paradigm is a natural framework for processing data batches that are collected serially; it also offers a technical advantage over the distributed-computing paradigm: while both require only the storing of summary statistics, the large-sample theory in a distributed setting is typically established under the assumption that the number of participants in each data batch diverges, which is clearly not aligned with our infinite-horizon setting with a finite number of participants.

The majority of efforts at developing procedures that allow for quick updates of parameter estimates fall in the field of online learning. This line of research dates back five decades to when Robbins & Monro (1951) proposed a stochastic approximation algorithm that laid a foundation for the popular stochastic gradient descent algorithm (Sakrison, 1965). The stochastic gradient descent algorithm and its variants have been extensively studied for online estimation and prediction (Toulis & Airoldi, 2017), but the work of developing online statistical inference remains unexplored. Recently, Fang (2019) proposed a perturbation-based resampling method to construct confidence intervals for stochastic gradient descent, but it does not achieve desirable statistical efficiency and may produce misleading inference in the case of large regression parameters. In addition to the stochastic gradient descent types of recursive algorithms, several cumulative updating methods have been proposed to specifically perform sequential updating of regression coefficient estimators, including the online least squares estimator for the linear model (Stengel, 1994; Chen et al., 2006) and the cumulative estimating equation estimator and cumulatively updated estimating equation estimator proposed by Schifano et al. (2016) for nonlinear models. Both

the cumulative estimating equation and the cumulatively updated estimating equation are developed under a mechanism similar to meta-analysis, and estimation consistency is established under a strong regularity condition that the number of data batches is much smaller than the sample size of each data batch (Lin & Xi, 2011; Schifano et al., 2016). Recently, Luo & Song (2020) proposed a renewable estimation and incremental inference method that is asymptotically equivalent to the maximum likelihood estimators obtained from the cumulative dataset as the number of independent participants goes to infinity. More importantly, this method overcomes the unnatural constraint on the number of data batches versus the sample size of each batch. It does not, however, allow for dependence between the data batches.

Indeed, most of the aforementioned online algorithms were developed under the assumption that samples collected at different time-points are independently generated from the same underlying model. A prominent concern in mobile health data analyses is that there exists a nonnegligible degree of correlation across different sampling points. Ignoring or misspecifying this correlation structure might not affect estimation consistency unless there are complicated missing data patterns (Liang & Zeger, 1986), but it may lead to a loss of statistical efficiency and therefore produce misleading inference in real-time decision-making. In the presence of nontrivial dependence between observations in streaming data, Cappé (2011) proposed an online expectation-maximization algorithm for parameter estimation in hidden Markov models, but did not provide a method for inference. Luo & Song (2023) developed a real-time inference method in linear state-space mixed models with batch-varying effects. To the best of our knowledge, real-time statistical inference with dependent and dynamic data batches in a general framework remains largely unexplored. Unlike in the independent setting, real-time statistical inference cannot be done through a simple linear aggregation of inferential quantities, such as information matrices, as in Luo & Song (2020). Instead, the within-individual correlation induces a huge nondiagonal covariance matrix of dimension $n \times n$, where n is the number of cumulative repeated measurements per individual, which grows rapidly with time. In addition to the nontrivial challenge of incorporating dependence between data batches, another issue is that existing methods model the data-generating process using a parametric model with fixed, common parameters that do not change over time. This assumption is too restrictive for high-throughput data. In the illustrative example of § 5, the association between explanatory variables, such as sex, and outcomes, such as physical activity, collected from wearable devices, may change dynamically over time rather than remain constant. Failing to account for these intrinsic dynamics can lead to severely misleading estimation and inference. A model that can handle local time dynamics is highly desirable, but so far lacking in the literature (Schifano et al., 2016; Luo & Song, 2020; Luo et al., 2023)

In this paper, we propose a new framework for online updating of point estimates and uncertainty quantification for time-varying effects in infinite-horizon longitudinal data settings. We derive a new result that shows how the extended inference function vector of the quadratic inference function of Qu et al. (2000) constructed over the cumulative longitudinal data elegantly decomposes into a sum of summary statistics over data batches. We further demonstrate how it can be computed recursively by using only the summary statistic from the previous data batch, a formulation which lends itself naturally to online updating over dependent data batches. To account for local dynamics, we incorporate an exponential weight function to dynamically adjust the weights applied to historical data batches. This approach is unique in that, in contrast to existing nonparametric approaches, it employs a one-sided kernel that uses only data prior to the current observations for weighting. We

solve the substantial technical challenge of deriving the asymptotic convergence rate of our online dynamic estimator under the infinite-horizon longitudinal data setting with finite sample size. In doing so, we move away from traditional assumptions of independence and embrace new frameworks that reflect local dynamics in streaming data environments. The resulting method leverages the dependence between data collected sequentially on the same set of participants to estimate dynamic associations with improved statistical efficiency. Because our framework does not require the sample size of each data batch to be large, our approach can be adopted for individual-level analysis as a special case. As a concrete example, suppose we have access to a single data stream collected from an individual with a fitness tracker or watch. At each updating time-point, the longitudinal outcome variables capturing activity counts and the covariates capturing heart rate, and blood oxygen saturation levels are sequentially updated. Then our approach can be applied to estimate the dynamically evolving associations between the covariates and outcomes.

2. Streaming inference for longitudinal data

2.1. *Problem set-up*

For scalars $s_1, \ldots, s_\ell$, vectors $a_\ell \in \mathbb{R}^v$ and matrices $A_\ell \in \mathbb{R}^{q_\ell \times v}$, where $\ell = 1, \ldots, L$, define the column-stacking operation on scalars, vectors and matrices by

$$(s_\ell)_{\ell=1}^{L} = \begin{pmatrix} s_1 & \ldots & s_L \end{pmatrix}^{\mathrm{T}} \in \mathbb{R}^L, \quad (a_\ell^{\mathrm{T}})_{\ell=1}^{L} = \begin{pmatrix} a_1 & \ldots & a_L \end{pmatrix}^{\mathrm{T}} \in \mathbb{R}^{L \times v},$$
$$(A_\ell)_{\ell=1}^{L} = \begin{pmatrix} A_1^{\mathrm{T}} & \ldots & A_L^{\mathrm{T}} \end{pmatrix}^{\mathrm{T}} \in \mathbb{R}^{\sum_{\ell=1}^{L} q_\ell \times v},$$

respectively. Suppose we collect data batches $\mathcal{D}_{ij} = \{y_{ij}, X_{ij}\}$ sequentially at deterministic updating time-points t_j $(j = 1, 2, \ldots, b)$ on the same set of independent participants $i = 1, \ldots, m$, where $y_{ij} \in \mathbb{R}^{n_j}$ is the vector of n_j longitudinal measurements on the same outcome variable in batch j, and $X_{ij} = (x_{i,kj}^{\mathrm{T}})_{k=1}^{n_j} \in \mathbb{R}^{n_j \times p}$ is the corresponding covariate matrix of p explanatory variables with $x_{i,kj} = (x_{i,1kj}, \ldots, x_{i,pkj}) \in \mathbb{R}^p$ for $k = 1, \ldots, n_j$ and $j = 1, \ldots, b$. Let $\mathcal{D}_{ib}^{\star} = \{\mathcal{D}_{i1}, \ldots, \mathcal{D}_{ib}\}$ denote the cumulative dataset up to batch b in participant i, and let $N_b = \sum_{j=1}^{b} n_j$ denote the corresponding aggregated response dimension. For ease of exposition, we assume an equal batch size n_j and an equal number of repeated measurements N_b for all participants. The spacing between time-points is also assumed to be the same for all participants. We consider the marginal generalized linear model with outcome $y_i = (y_{i1}^{\mathrm{T}}, \ldots, y_{ib}^{\mathrm{T}})^{\mathrm{T}} \in \mathbb{R}^{N_b}$ and p covariates $X_i = (X_{ij})_{j=1}^{b} \in \mathbb{R}^{N_b \times p}$:

$$E(y_i \mid X_i) = \mu_i = (\mu_{i,kj})_{k,j=1}^{n_j,b} = \left[h\{x_{i,kj}^{\mathrm{T}} \beta(t_j)\}\right]_{k,j=1}^{n_j,b} \in \mathbb{R}^{N_b} \quad (i = 1, \ldots, m),$$

where $\beta(\cdot)$ is the regression coefficient function and $h(\cdot)$ is a known link function. The regression coefficient $\beta(\cdot)$ is assumed to be a smooth function that captures local dynamics of the outcomes and explanatory variables. We consider a batch-varying coefficient, denoted by $\beta(t_j) \in \mathbb{R}^p$, for the batch of data collected at time t_j. For notational simplicity, we use $\beta_j \in \mathbb{R}^p$ to denote the true value of the batch-specific coefficient and use β as generic notation for the coefficient function.

The covariance of the outcome is $\mathrm{cov}(y_i \mid X_i) \propto A_i^{1/2} R(\alpha) A_i^{1/2}$, where $A_i = \mathrm{diag}\{v(\mu_{i,kj})\}_{k,j=1}^{n_j,b}$ is a diagonal matrix with a known variance function $v(\cdot)$, and $R(\alpha)$ is a working correlation matrix that is fully characterized by a correlation parameter α. Clearly,

the dimensions of y_i, X_i, A_i and $R(\cdot)$ depend on the number of batches b, which is allowed to diverge. We use the subscript i to refer to the cumulative data on participant i up to batch b; the dependence on b is suppressed for parsimony of notation, but we will give reminders of this fact when relevant. To study the relationship between the longitudinal outcomes and covariates, we focus on estimation and inference for β, with α treated as a nuisance parameter.

Because of the longitudinal nature of the infinite-horizon setting, we model the longitudinal measurements through a first-order autoregressive working correlation structure. The first-order autoregressive process is one of the most widely used correlation models for longitudinal and time series correlations, because it provides a natural description of the exponential decay in correlation between measurements that occurs over time. Specifically, this process assumes that the correlation between two longitudinal outcomes Y_{it} and Y_{is}, measured at time-points t and s, respectively, satisfies a serial structure given by $\text{corr}(Y_{it}, Y_{is}) = \alpha^{|t-s|}$ for $t \neq s$ and $\alpha \in (-1, 1)$.

2.2. *Offline approaches to longitudinal data analysis*

Full likelihood-based estimation and inference for β depends on the specification of an N_b-dimensional multivariate likelihood and parameterization of all moments up to order N_b, which poses modelling and computational challenges. While these have proven trivial to overcome for multivariate Gaussian likelihoods, the specification of the likelihood for non-Gaussian outcomes typically relies on copulas (Song, 2007; Joe, 2014), which are computationally burdensome. Alternative quasilikelihood-based approaches, such as the composite likelihood (Lindsay, 1988; Varin et al., 2011), prove a necessary alternative, but frequently at the cost of statistical efficiency. The generalized estimating equations of Liang & Zeger (1986) avoid the need for a likelihood by directly specifying an estimating equation for β, with α estimated via a method-of-moments approach. Up to time t_b, a generalized estimating equations estimator of β is the solution to the weighted generalized estimating equation based on data $\{\mathcal{D}^\star_{ib}\}_{i=1}^m$, denoted by $\psi^\star_b(\beta, \alpha; \{\mathcal{D}^\star_{ib}\}_{i=1}^m) = \sum_{i=1}^m D_i^{\mathrm{T}} \Sigma_i^{-1} W_b (y_i - \mu_i) = 0$, where $D_i = \nabla_\beta \mu_i = (D_{ij})_{j=1}^b \in \mathbb{R}^{N_b \times p}$ and $\Sigma_i = A_i^{1/2} R(\alpha) A_i^{1/2}$ with $A_i = \text{diag}\{A_{ij}\}_{j=1}^b \in \mathbb{R}^{N_b \times N_b}$. Here, we introduce an additional weighting matrix $W_b = \text{diag}\{W_{bj}\}_{j=1}^b \in \mathbb{R}^{N_b \times N_b}$ to the original generalized estimating equation framework. This matrix dynamically adjusts the weights assigned to data batches collected at different time-points. In particular, we define $W_{bj} = q^{t_b - t_j} I_{n_j}$ for $0 < q < 1$; with this weight function, observations in batches that are further away from batch b receive less weight. We remind the reader that the dimensions of D_i, A_i, W_b, $R(\cdot)$ and Σ_i depend on b, which is allowed to diverge. The generalized estimating equations have enjoyed widespread popularity in the analysis of longitudinal data owing to their ease of implementation and desirable statistical properties: when the correlation structure is correctly specified by $R(\alpha)$, the generalized estimating equations estimator is consistent and semiparametrically efficient, i.e., as efficient as the quasilikelihood. Even when the correlation structure is misspecified, the generalized estimating equations estimator remains consistent. Unfortunately, there exist simple cases for which the estimator of the correlation parameter α does not exist (Crowder, 1995). Generally, estimation of α is cumbersome since the target of inference is β, and a preferable approach would bypass estimation of the correlation structure altogether.

The quadratic inference function of Qu et al. (2000) avoids estimation of α through a clever substitution of a linear expansion of known basis matrices for the inverse of the working correlation matrix in the generalized estimating equations. The formulation

of the quadratic inference function is based on an approximation to the inverse of the working correlation matrix by $R^{-1}(\alpha) \approx \sum_{s=1}^{S} \gamma_s M_s$, where $\gamma_1, \ldots, \gamma_S$ are unknown constants, possibly dependent on α, and $M_1, \ldots, M_S \in \mathbb{R}^{N_b \times N_b}$ are known basis matrices with elements 0 and 1, which are determined by a given correlation structure in $R(\alpha)$. Plugging this expansion into the generalized estimating equations leads to $\psi_b^\star(\beta, \alpha; \{\mathcal{D}_{ib}^\star\}_{i=1}^m) = \sum_{i=1}^m \sum_{s=1}^S \gamma_s D_i^{\mathrm{T}} A_i^{-1/2} M_s A_i^{-1/2} W_b(y_i - \mu_i) = 0$, which can be expressed as a linear combination of the extended inference function vector

$$U_b^\star(\beta) = \sum_{i=1}^m U_{ib}^\star(\beta) = \sum_{i=1}^m \begin{pmatrix} D_i^{\mathrm{T}} A_i^{-1/2} M_1 A_i^{-1/2} W_b(y_i - \mu_i) \\ \vdots \\ D_i^{\mathrm{T}} A_i^{-1/2} M_S A_i^{-1/2} W_b(y_i - \mu_i) \end{pmatrix} \in \mathbb{R}^{pS}. \tag{1}$$

Again, we introduce an additional weighting matrix W_b to the original quadratic inference function framework to dynamically adjust weights assigned to batches collected at different time-points.

Clearly, estimation of the second-order moments of y_i is no longer required in the quadratic inference function since $\gamma_1, \ldots, \gamma_S$ are not involved in the construction of $U_b^\star(\beta)$. The extended inference function vector in (1) is an over-identified estimating function: $pS = \dim\{U_b^\star(\beta)\} > \dim(\beta) = p$. To obtain an estimator of β, following the generalized method of moments of Hansen (1982), the quadratic inference function estimator is defined as $\hat{\beta}_b^\star = \arg\min_\beta Q_b^\star(\beta)$ with

$$Q_b^\star(\beta) = U_b^\star(\beta)^{\mathrm{T}} \left\{ V_b^\star(\beta) \right\}^{-1} U_b^\star(\beta), \tag{2}$$

where $V_b^\star(\beta) = \sum_{i=1}^m U_{ib}^\star(\beta) U_{ib}^\star(\beta)^{\mathrm{T}}$ is the sample covariance matrix of $U_b^\star(\beta)$. The correlation parameter α is not involved in (2), so that the estimation of β using $\hat{\beta}_b^\star$ yields substantial computational gains over the generalized estimating equations when N_b is large. A suitable approximation to the inverse of a first-order autoregressive working correlation structure is $R^{-1}(\alpha) \approx \gamma_1 M_1 + \gamma_2 M_2$, where $M_1 = I_{N_b}$ is the $N_b \times N_b$ identity matrix and M_2 is a matrix with 1 on the two main off-diagonals and 0 elsewhere (Qu et al., 2000). While a third basis matrix is sometimes used to capture edge effects, using these two basis matrices for the first-order autoregressive process gives satisfactory efficiency gains in practice (Song, 2007, Ch. 5). As will be shown in § 2.3, the use of these basis matrices allows for an elegant decomposition of the covariance matrix, leading to substantial computational gains.

It is well known that the quadratic inference function estimator $\hat{\beta}_b^\star$ is consistent even if the correlation structure imposed by the choice of $M_1, \ldots, M_S$ is misspecified, and that it is semiparametrically efficient when the correlation structure is correctly specified (Qu et al., 2000). In addition, it has been shown both theoretically and numerically that the estimation efficiency of the quadratic inference function estimator $\hat{\beta}_b^\star$ is higher than that of the generalized estimating equations estimator under correlation misspecification (Qu et al., 2000; Song et al., 2009). According to standard generalized method of moments theory (Hansen, 1982), misspecification of the correlation structure does not impact estimation consistency, only estimation efficiency. The use of the quadratic inference function reduces the difficulty of computation on the cumulative data $\{\mathcal{D}_{ib}^\star\}_{i=1}^m$ compared with likelihood- and quasilikelihood-based approaches by avoiding estimation of the nuisance parameter related to second-order moments of the outcome. Nonetheless, it does not provide a satisfactory

solution to the tremendous memory and computational costs incurred by the analysis of the cumulative data.

2.3. *A new decomposition for quadratic inference functions with data batches*

We derive a new result that shows how the extended inference function vector of the quadratic inference function decomposes into a sum of summary statistics over a sequence of data batches. The basic idea is to partition the first-order autoregression basis matrices M_1 and M_2 of dimension $N_b \times N_b$ by data batches such that (i) each submatrix is of dimension $n_j \times n_j$ and (ii) more importantly, despite their dimension depending on the data batch size n_j, these submatrices share the same structure across the different data batches. This allows us to write $U_b^\star(\beta)$ as a summation of inference functions across batches rather than as a function of the cumulative dataset. Specifically, we first partition the extended inference function vector $U_b^\star(\beta)$ into two subvectors based on the basis matrices: let $U_b^\star(\beta)^{(1)} \in \mathbb{R}^p$ denote the subvector corresponding to the identity basis matrix M_1, and let $U_b^\star(\beta)^{(2)} \in \mathbb{R}^p$ denote the subvector that involves M_2. The extended inference function vector up to data batch b can be written as

$$U_b^\star(\beta) = \begin{pmatrix} U_b^\star(\beta)^{(1)} \\ U_b^\star(\beta)^{(2)} \end{pmatrix} = \sum_{i=1}^m \begin{pmatrix} U_{ib}^\star(\beta)^{(1)} \\ U_{ib}^\star(\beta)^{(2)} \end{pmatrix} = \sum_{i=1}^m \begin{pmatrix} D_i^{\mathrm{T}} A_i^{-1/2} M_1 A_i^{-1/2} W_b(y_i - \mu_i) \\ D_i^{\mathrm{T}} A_i^{-1/2} M_2 A_i^{-1/2} W_b(y_i - \mu_i) \end{pmatrix}.$$

Since M_1 is an identity matrix of dimension $N_b \times N_b$ that is equivalent to the case of independent data batches, $U_b^\star(\beta)^{(1)} = \sum_{i=1}^m \sum_{j=1}^b q^{t_b - t_j} U_{ij}(\beta)^{(1)}$ is a linear aggregation of the inference functions corresponding to each data batch $\mathcal{D}_{ij}$ $(j = 1, \ldots, b)$ for each subject $i = 1, \ldots, m$. Decomposition of M_2 over data batches, however, is not trivial. As an illustration of the proposed decomposition, we consider a simple case involving two data batches with $n_1 = 2$ and $n_2 = 3$; a decomposition of $M_2 \in \mathbb{R}^{5\times 5}$ takes the form

$$M_2 = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix} = \left(\begin{array}{cc|ccc} 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ \hline 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{array}\right) = \begin{pmatrix} M_{21} & B_1 \\ B_2 & M_{22} \end{pmatrix},$$

where $M_{21} \in \mathbb{R}^{2\times 2}$ and $M_{22} \in \mathbb{R}^{3\times 3}$ share the same structure and $B_2 = B_1^{\mathrm{T}}$. More generally, let $M_{2j} \in \mathbb{R}^{n_j \times n_j}$ denote the diagonal blocks of M_2 corresponding to the jth data batch, and let $B_j \in \mathbb{R}^{n_j \times n_{j+1}}$ be the off-diagonal blocks with 1 as the $(n_j, 1)$ entry and 0 elsewhere. Then $U_b^\star(\beta)^{(2)}$ decomposes as

$$\begin{aligned} \sum_{i=1}^m U_{ib}^\star(\beta)^{(2)} &= \sum_{i=1}^m D_i^{\mathrm{T}} A_i^{-\frac{1}{2}} M_2 A_i^{-\frac{1}{2}} W_b(y_i - \mu_i) \\ &= \sum_{i=1}^m \begin{pmatrix} D_{i1} \\ \vdots \\ D_{ib} \end{pmatrix}^{\mathrm{T}} \begin{pmatrix} A_{i1} & & \\ & \ddots & \\ & & A_{ib} \end{pmatrix}^{-\frac{1}{2}} \begin{pmatrix} M_{21} & B_1 & \\ & \ddots & \\ & B_b^{\mathrm{T}} & M_{2,b} \end{pmatrix} \begin{pmatrix} A_{i1} & & \\ & \ddots & \\ & & A_{ib} \end{pmatrix}^{-\frac{1}{2}} \begin{pmatrix} W_{b,1}(y_{i1} - \mu_{i1}) \\ \vdots \\ W_{b,b}(y_{ib} - \mu_{ib}) \end{pmatrix} \\ &= \sum_{i=1}^m \sum_{j=1}^b q^{t_b - t_j} U_{ij}(\beta)^{(2)} + \sum_{i=1}^m \sum_{j=1}^{b-1} q^{t_b - t_{j+1}} U_{i,j,j+1}(\beta) + \sum_{i=1}^m \sum_{j=1}^{b-1} q^{t_b - t_j} U_{i,j+1,j}(\beta), \end{aligned} \tag{3}$$

with

$$\begin{aligned} U_{ij}(\beta)^{(1)} &= D_{ij}^{\mathrm{T}} A_{ij}^{-1/2} M_1 A_{ij}^{-1/2}(y_{ij}-\mu_{ij}), \quad U_{ij}(\beta)^{(2)} = D_{ij}^{\mathrm{T}} A_{ij}^{-1/2} M_2 A_{ij}^{-1/2}(y_{ij}-\mu_{ij}), \\ U_{i,j,j+1}(\beta) &= D_{ij}^{\mathrm{T}} A_{ij}^{-1/2} B_j A_{i,j+1}^{-1/2}(y_{i,j+1}-\mu_{i,j+1}), \\ U_{i,j+1,j}(\beta) &= D_{i,j+1}^{\mathrm{T}} A_{i,j+1}^{-1/2} B_j^{\mathrm{T}} A_{ij}^{-1/2}(y_{ij}-\mu_{ij}). \end{aligned} \tag{4}$$

The corresponding negative gradient matrices are denoted by

$$\begin{aligned} S_{ij}(\beta)^{(1)} &= D_{ij}^{\mathrm{T}} A_{ij}^{-1/2} M_{1j} A_{ij}^{-1/2} D_{ij}, \qquad S_{ij}(\beta)^{(2)} = D_{ij}^{\mathrm{T}} A_{ij}^{-1/2} M_{2j} A_{ij}^{-1/2} D_{ij}, \\ S_{i,j,j+1}(\beta) &= D_{ij}^{\mathrm{T}} A_{ij}^{-1/2} B_j A_{i,j+1}^{-1/2} D_{i,j+1}, \quad S_{i,j+1,j}(\beta) = D_{i,j+1}^{\mathrm{T}} A_{i,j+1}^{-1/2} B_j^{\mathrm{T}} A_{ij}^{-1/2} D_{ij}. \end{aligned}$$

Thus, we have shown that $U_b^{\star}(\beta)^{(2)}$ elegantly decomposes into estimating functions for within-batch dependencies, through $U_{ij}(\beta)^{(2)}$, and between-batch dependencies, through $U_{i,j,j+1}(\beta)$ and $U_{i,j+1,j}(\beta)$. This decomposition effectively breaks down the massive correlation matrices of dimension $N_b \times N_b$ into smaller matrices of dimension $n_j \times n_j$. While this construction might seem more appealing, we have not yet reduced the computational and memory burdens associated with $Q_b^{\star}(\beta)$ in (2). Indeed, despite the construction in (3), (2) must be solved each time a new data batch arrives, since it depends on the cumulative data $\{\mathcal{D}_{ib}^{\star}\}_{i=1}^{m}$ up to batch b. In § 2.4 we show how to use the decomposition in (3) to avoid processing the cumulative data.

2.4. *The streaming inference framework*

Instead of processing the cumulative dataset $\{\mathcal{D}_{ib}^{\star}\}_{i=1}^{m}$ once through equation (2), we propose a recursive updating procedure for online estimation and inference. In our proposed streaming updating procedure, let $\tilde{\beta}_b$ denote the online estimator up to data batch b. We initialize $\tilde{\beta}_1$ by the offline quadratic inference function estimator with the first data batch, i.e., $\hat{\beta}_1 = \arg\min_{\beta} Q_1(\beta)$. When data batches $\{\mathcal{D}_{ib}\}_{i=1}^{m}$ are collected, we update the previous estimator $\tilde{\beta}_{b-1}$ to $\tilde{\beta}_b$ using only summary statistics from previous data batches $\{\mathcal{D}_{i,b-1}^{\star}\}_{i=1}^{m}$ and the raw data in the current data batch $\{\mathcal{D}_{ib}\}_{i=1}^{m}$. After completing the updating, individual-level data in $\{\mathcal{D}_{ib}\}_{i=1}^{m}$ are no longer accessible for the sake of storage. We can only access the updated estimate $\tilde{\beta}_b$, and summary statistics are carried forward for future updating.

For clarity of exposition, we begin our derivation with two data batches $\mathcal{D}_{i1}$ and $\mathcal{D}_{i2}$ that are collected sequentially at time-points t_1 and t_2, respectively, where $\mathcal{D}_{i2}$ arrives after $\mathcal{D}_{i1}$ for $i = 1, \ldots, m$. Following Qu et al. (2000), the quadratic inference function estimator $\hat{\beta}_1 = \arg\min_{\beta} Q_1(\beta)$ satisfies $S_1^{\mathrm{T}}(\hat{\beta}_1)\{V_1(\hat{\beta}_1)\}^{-1} U_1(\hat{\beta}_1) = 0$, where $Q_1(\beta) = U_1^{\mathrm{T}}(\beta) V_1^{-1}(\beta) U_1(\beta)$, $U_1(\beta)$ is the extended inference function for the first data batch, $S_1(\beta) = -\nabla_{\beta} U_1(\beta)$ is the negative gradient of $U_1(\beta)$ and $V_1(\beta) = \sum_{i=1}^{m} U_{i1}(\beta) U_{i1}(\beta)^{\mathrm{T}}$ is the sample variability matrix of $U_1(\beta)$. We require a moderately large m or n_1 to compute the initial estimator $\hat{\beta}_1$. When $\{\mathcal{D}_{i2}\}_{i=1}^{m}$ arrives, we can obtain the offline quadratic inference function estimator $\hat{\beta}_2^{\star}$ based on the cumulative dataset $\{\mathcal{D}_{i2}^{\star}\}_{i=1}^{m}$ by solving the estimating equation $S_2^{\star}(\hat{\beta}_2^{\star})^{\mathrm{T}}\{V_2^{\star}(\hat{\beta}_2^{\star})\}^{-1} U_2^{\star}(\hat{\beta}_2^{\star}) = 0$, where each of the building blocks can be decomposed as

follows:

$$
\begin{aligned}
U_{i2}^{\star}(\hat{\beta}_2^{\star}) &= q^{t_2-t_1} U_{i1}(\hat{\beta}_2^{\star}) + \begin{pmatrix} U_{i2}(\hat{\beta}_2^{\star})^{(1)} \\ U_{i2}(\hat{\beta}_2^{\star})^{(2)} + U_{i,12}(\hat{\beta}_2^{\star}) + q^{t_2-t_1} U_{i,21}(\hat{\beta}_2^{\star}) \end{pmatrix}, \\
S_{i2}^{\star}(\hat{\beta}_2^{\star}) &= q^{t_2-t_1} S_{i1}(\hat{\beta}_2^{\star}) + \begin{pmatrix} S_{i2}(\hat{\beta}_2^{\star})^{(1)} \\ S_{i2}(\hat{\beta}_2^{\star})^{(2)} + S_{i,12}(\hat{\beta}_2^{\star}) + q^{t_2-t_1} S_{i,21}(\hat{\beta}_2^{\star}) \end{pmatrix}, \\
U_2^{\star}(\hat{\beta}_2^{\star}) &= \sum_{i=1}^{m} U_{i2}^{\star}(\hat{\beta}_2^{\star}), \quad V_2^{\star}(\hat{\beta}_2^{\star}) = \sum_{i=1}^{m} U_{i2}^{\star}(\hat{\beta}_2^{\star})\{U_{i2}^{\star}(\hat{\beta}_2^{\star})\}^{\mathrm{T}}, \quad S_2^{\star}(\hat{\beta}_2^{\star}) = \sum_{i=1}^{m} S_{i2}^{\star}(\hat{\beta}_2^{\star}).
\end{aligned} \tag{5}
$$

Even though the quantities in (5) admit recursive updating forms, plugging the newly obtained estimator $\hat{\beta}_2^{\star}$ into old summary statistics, such as U_1, requires access to historical raw data $\{\mathcal{D}_{i1}\}_{i=1}^{m}$. This incurs both a data storage burden and a recomputation burden. To avoid reusing historical raw data, we do not carry out calculations retrospectively. To derive an online streaming estimation procedure, we take a first-order Taylor expansion of the terms $U_1(\hat{\beta}_2^{\star})$ and $S_1(\hat{\beta}_2^{\star})$ about $\hat{\beta}_1$, to obtain

$$
\begin{aligned}
\frac{n_1}{N_2} U_1(\hat{\beta}_2^{\star}) &= \frac{n_1}{N_2} U_1(\hat{\beta}_1) + \frac{n_1}{N_2} S_1(\hat{\beta}_1)(\hat{\beta}_1 - \hat{\beta}_2^{\star}) + O_{\mathrm{p}}\Big(\frac{n_1}{N_2}\|\hat{\beta}_1 - \hat{\beta}_2^{\star}\|^2\Big), \\
\frac{n_1}{N_2} S_1(\hat{\beta}_2^{\star}) &= \frac{n_1}{N_2} S_1(\hat{\beta}_1) + O_{\mathrm{p}}\Big(\frac{n_1}{N_2}\|\hat{\beta}_1 - \hat{\beta}_2^{\star}\|\Big).
\end{aligned} \tag{6}
$$

The error terms $O_{\mathrm{p}}(n_1\|\hat{\beta}_1 - \hat{\beta}_2^{\star}\|/N_2)$ and $O_{\mathrm{p}}(n_1\|\hat{\beta}_1 - \hat{\beta}_2^{\star}\|^2/N_2)$ in (6) may be asymptotically ignored if N_2 is large enough, and $\|\beta_2 - \beta_1\| = o(1)$, which will be discussed further in § 3. We drop these higher-order terms and propose an online streaming quadratic inference function estimator $\tilde{\beta}_2$ as a solution to the estimating equation $\tilde{S}_2^{\mathrm{T}} \tilde{V}_2^{-1} \tilde{U}_2 = 0$, where $\tilde{U}_2 = q^{t_2-t_1} U_1(\hat{\beta}_1) + q^{t_2-t_1} S_1(\hat{\beta}_1)(\hat{\beta}_1 - \tilde{\beta}_2) + U_2(\tilde{\beta}_2)$ is the adjusted inference function and $\tilde{S}_2 = q^{t_2-t_1} S_1(\hat{\beta}_1) + S_2(\tilde{\beta}_2)$ and $\tilde{V}_2 = \sum_{i=1}^{m} \tilde{U}_{i2}\tilde{U}_{i2}^{\mathrm{T}}$ are the aggregated negative gradient and sample variability matrices of $\tilde{U}_2$, respectively. In contrast to the independent case, the sample variability matrix does not take a linear aggregation form. Through this approximation, we avoid retrospective calculations of $U_1(\hat{\beta}_2^{\star})$ and $S_1(\hat{\beta}_2^{\star})$ with $\{\mathcal{D}_{i1}\}_{i=1}^{m}$. Therefore, we can update $\hat{\beta}_1$ to $\tilde{\beta}_2$ without reaccessing individual-level data in $\{\mathcal{D}_{i1}\}_{i=1}^{m}$. In addition, we can solve for $\tilde{\beta}_2$ via the Newton–Raphson algorithm. Specifically, at the $(r+1)$th iteration,

$$
\tilde{\beta}_2^{(r+1)} = \tilde{\beta}_2^{(r)} + \big\{(\tilde{S}_2^{(r)})^{\mathrm{T}}(\tilde{V}_2^{(r)})^{-1}\tilde{S}_2^{(r)}\big\}^{-1}(\tilde{S}_2^{(r)})^{\mathrm{T}}(\tilde{V}_2^{(r)})^{-1}\tilde{U}_2^{(r)},
$$

where

$$
\begin{aligned}
\tilde{U}_{i2}^{(r)} &= q^{t_2-t_1} U_{i1}(\hat{\beta}_1) + q^{t_2-t_1} S_{i1}(\hat{\beta}_1)(\hat{\beta}_1 - \tilde{\beta}_2^{(r)}) \\
&\quad + \begin{pmatrix} U_{i2}(\tilde{\beta}_2^{(r)})^{(1)} \\ U_{i2}(\tilde{\beta}_2^{(r)})^{(2)} + U_{i,12}(\tilde{\beta}_2^{(r)}) + q^{t_2-t_1} U_{i,21}(\tilde{\beta}_2^{(r)}) \end{pmatrix}, \\
\tilde{S}_{i2}^{(r)} &= q^{t_2-t_1} S_{i1}(\hat{\beta}_1) + \begin{pmatrix} S_{i2}(\tilde{\beta}_2^{(r)})^{(1)} \\ S_{i2}(\tilde{\beta}_2^{(r)})^{(2)} + S_{i,12}(\tilde{\beta}_2^{(r)}) + q^{t_2-t_1} S_{i,21}(\tilde{\beta}_2^{(r)}) \end{pmatrix}, \\
\tilde{U}_2(\hat{\beta}_2^{(r)}) &= \sum_{i=1}^{m} \tilde{U}_{i2}^{(r)}, \quad \tilde{V}_2(\hat{\beta}_2^{(r)}) = \sum_{i=1}^{m} \tilde{U}_{i2}^{(r)}\{\tilde{U}_{i2}^{(r)}\}^{\mathrm{T}}, \quad \tilde{S}_2^{(r)} = \sum_{i=1}^{m} \tilde{S}_{i2}^{(r)}.
\end{aligned}
$$

Remarkably, owing to the decomposition of the inference functions in (5) and the Taylor approximations in (6), individual-level data in $\mathcal{D}_1$ are not used except for the last observation $\{x_{in_1}, y_{in_1}\}_{i=1}^m$, which appears in the between-batch terms $U_{i,12}(\tilde{\beta}_2)$, $U_{i,21}(\tilde{\beta}_2)$, $S_{i,12}(\tilde{\beta}_2)$ and $S_{i,21}(\tilde{\beta}_2)$; see (4). This allows us to avoid storing historical raw data while still accounting for and leveraging dependence between data batches for more efficient inference.

Generalizing the above procedure to a general streaming data setting where we want to update $\tilde{\beta}_{b-1}$ to $\tilde{\beta}_b$, the online estimator $\tilde{\beta}_b$ of β is the solution to the incremental estimating equation

$$\tilde{S}_b^{\mathrm{T}} \tilde{V}_b^{-1} \tilde{U}_b = 0, \tag{7}$$

where the building blocks are updated in a similar way to (6):

$$\begin{aligned}
\tilde{U}_{ib} &= q^{t_b - t_{b-1}} \tilde{U}_{i,b-1} + q^{t_b - t_{b-1}} \tilde{S}_{i,b-1}(\tilde{\beta}_{b-1} - \tilde{\beta}_b) \\
&\quad + \begin{pmatrix} U_{ib}(\tilde{\beta}_b)^{(1)} \\ U_{ib}(\tilde{\beta}_b)^{(2)} + U_{i,b-1,b}(\tilde{\beta}_b) + q^{t_b - t_{b-1}} U_{i,b,b-1}(\tilde{\beta}_b) \end{pmatrix}, \\
\tilde{S}_{ib} &= q^{t_b - t_{b-1}} \tilde{S}_{i,b-1} + \begin{pmatrix} S_{ib}(\tilde{\beta}_b)^{(1)} \\ S_{ib}(\tilde{\beta}_b)^{(2)} + S_{i,b-1,b}(\tilde{\beta}_b) + q^{t_b - t_{b-1}} S_{i,b,b-1}(\tilde{\beta}_b) \end{pmatrix}, \\
\tilde{U}_b &= \sum_{i=1}^m \tilde{U}_{ib}, \quad \tilde{V}_b = \sum_{i=1}^m \tilde{U}_{ib} \tilde{U}_{ib}^{\mathrm{T}}, \quad \tilde{S}_b = \sum_{i=1}^m \tilde{S}_{ib}.
\end{aligned}$$

Solving (7) can be done via the Newton–Raphson algorithm with the $(r+1)$th iteration taking the form

$$\tilde{\beta}_b^{(r+1)} = \tilde{\beta}_b^{(r)} + \{\tilde{S}_b^{(r)\mathrm{T}} (\tilde{V}_b^{(r)})^{-1} \tilde{S}_b^{(r)}\}^{-1} \tilde{S}_b^{(r)\mathrm{T}} (\tilde{V}_b^{(r)})^{-1} \tilde{U}_b^{(r)},$$

where we do not need to access the entire raw dataset except for the observations in the current batch $\mathcal{D}_{ib}$ and the last observation in data batch $\mathcal{D}_{i,b-1}$. Instead, we use only the previous estimate $\tilde{\beta}_{b-1}$, as well as the summary statistics $\{\tilde{U}_{i,b-1}, \tilde{S}_{i,b-1}\}_{i=1}^m$ from the historical data up to time-point t_{b-1}. In addition, since we use aggregated quantities over b batches, denoted by $\{\tilde{S}_b, \tilde{V}_b, \tilde{U}_b\}$, to construct the estimating equation, even with a small n_j $(j \geqslant 2)$ the updated matrix $\tilde{S}_b^{\mathrm{T}} \tilde{V}_b^{-1} \tilde{S}_b$ is positive definite as well.

Finally, we propose an adaptive tuning procedure for selecting the weighting parameter q. Let $\mathcal{C}_q$ denote a candidate set for q. At the updating time-point t_b, we compute $\tilde{\beta}_b(q)$ for all $q \in \mathcal{C}_q$ and choose the q that minimizes the quadratic inference function constructed with raw data in batch b and the last observation in batch $b-1$ only, as shown in (8). Specifically, let

$$\begin{aligned}
U_{ib}(\tilde{\beta}_b, q) &= \begin{pmatrix} U_{ib}(\tilde{\beta}_b)^{(1)} \\ U_{ib}(\tilde{\beta}_b)^{(2)} + U_{i,b-1,b}(\tilde{\beta}_b) + q^{t_b - t_{b-1}} U_{i,b,b-1}(\tilde{\beta}_b) \end{pmatrix}, \\
U_b(\tilde{\beta}_b, q) &= \sum_{i=1}^m U_{ib}(\tilde{\beta}_b, q), \quad V_b(\tilde{\beta}_b, q) = \sum_{i=1}^m U_{ib}(\tilde{\beta}_b, q)\{U_{ib}(\tilde{\beta}_b, q)\}^{\mathrm{T}}.
\end{aligned}$$

We propose to select q using

$$q_b^{\text{opt}} = \underset{q \in \mathcal{C}_q}{\arg\min}\ U_b(\tilde{\beta}_b, q)^{\mathrm{T}} \{V_b(\tilde{\beta}_b, q)\}^{-1} U_b(\tilde{\beta}_b, q). \tag{8}$$

3. Large-sample properties

We establish large-sample properties of our proposed online estimator $\tilde{\beta}_b$ in (7) under the condition that n_j is finite for every $j = 1, \ldots, b$, but the number of data batches b tends to infinity. The technical difficulty arises from the fact that n_j is finite and the convergence is driven by the number of iterative steps indexed by b. We first define population quantities of interest: let the sensitivity and variability matrices for batch b be denoted by $\mathbb{S}\{\beta(t_b)\} = n_b^{-1}\mathbb{E}[S_{ib}\{\beta(t_b)\}] = n_b^{-1}\mathbb{E}[-\partial U_{ib}\{y_i; X_i, \beta(t_b)\}/\partial\beta(t_b)^{\mathrm{T}}]$ and $\mathbb{V}\{\beta(t_b)\} = n_b^{-1}\mathbb{E}[U_{ib}\{y_i; X_i, \beta(t_b)\}U_{ib}\{y_i; X_i, \beta(t_b)\}^{\mathrm{T}}]$. We consider the following set of assumptions.

Assumption 1. For participant $i = 1, \ldots, m$, the expectation $\mathbb{E}[n_b^{-1} U_{ib}\{y_i; X_i, \beta(t_b)\}] = 0$ if and only if $\beta(t_b) = \beta_b$. Furthermore, $\mathbb{E}[\|n_b^{-1} U_{ib}\{y_i; X_i, \beta(t_b)\}\|^r] < \infty$ $(r = 1, 2)$ for all updating time-points.

Assumption 2. The matrix $H_b = N_b^{-1}\tilde{S}_b^{\mathrm{T}}\tilde{V}_b^{-1}\tilde{S}_{b-1}$ is positive definite for $b \geqslant 2$.

Assumption 3. For all updating time-points considered, the function of the inference function vector $U_{ib}^{\star}\{y_i; X_i, \beta(t_b)\}$ is twice continuously differentiable in $\beta(t_b)$, and the sensitivity matrix $\mathbb{S}\{\beta(t_b)\} = n_b^{-1}\mathbb{E}[S_b\{\mathcal{D}_b; \beta(t_b)\}]$ is of full column rank.

Assumption 4. For every individual i, the vector $(y_{i,kj} - \mu_{i,kj})_{k,j=1}^{n_j,b}$ forms a ρ-mixing stochastic process. If $\rho(l)$ $(l = 1, 2, \ldots)$ denote the mixing coefficients for $l = 1, 2, \ldots$, then $\sum_l \rho(l) < \infty$.

Assumption 5. The time-varying coefficient $\beta(t)$ is twice differentiable with respect to t with bounded derivative. During the time period $\mathcal{T}$ under consideration, defined on a compact set, $\sup_{t\in\mathcal{T}} \beta(t)$ is bounded. The adjacent batches share similar time-varying coefficients in the sense that $b \sup_{1\leqslant j\leqslant b} |\beta(t_j) - \beta(t_{j-1})| < \infty$.

Assumption 6. The number of time-points N_b observed in the algorithm running period and the tuning parameter q satisfy $-1/\log q \to 0$ and, as $b \to \infty$, $N_b \to \infty$, $-N_b/\log q \to \infty$ and $\log N_b/\{-N_b/(\log q)^3\}^{1/2} \to 0$.

Assumption 7. For all updating time-points considered, the variability matrix $\mathbb{V}\{\beta(t_b)\}$ is positive definite.

Assumptions 1–3 and 7 are regularity assumptions required to establish asymptotic consistency and asymptotic normality of the generalized method of moments estimator (Hansen, 1982). The matrix H_b in Assumption 2 approximates the sample version of the covariance matrix of $\tilde{\beta}_b$ and its positive definiteness is needed to ensure the feasibility of statistical inference. Assumption 4 is a reasonable condition for temporal time series for which dependence decays to zero with high polynomial rates over time. Assumption 5 is a smoothness condition commonly adopted in the literature on varying-coefficient models

(Fan & Zhang, 1999). By restricting the maximal change of the coefficient $\beta(t)$ for adjacent time-points during the batch updating period, we assume that the conditional distribution of the outcome given the covariates changes gradually over time. This is a reasonable expectation given our motivating wearable device application. In practice, if the local dynamics of $\beta(t)$ do not exhibit the smooth change required by the assumption, our tuning parameter selection procedure aims to adapt to these changes, and chooses a rather small q so that the data batches collected prior to these changes play a less important role. Assumption 6 imposes restrictions on the smoothing parameter q and the number of updates b. The smoothing parameter is chosen at a faster rate than in standard nonparametric regression problems so that the smoothing bias vanishes, allowing our procedure to yield valid statistical inference.

Theorem 1. *Under Assumptions 1–6, the online estimator* $\tilde{\beta}_b$ *in* (7) *is consistent, that is,* $\tilde{\beta}_b \to \beta_b$ *in probability as* $N_b = \sum_{j=1}^{b} n_j \to \infty$.

Theorem 2. *Under Assumptions 1–7, the online estimator* $\tilde{\beta}_b$ *in* (7) *is asymptotically normally distributed, that is,* $(-N_b/\log q)^{1/2}(\tilde{\beta}_b - \beta_b) \to \mathcal{N}\{0, \mathbb{J}^{-1}(\beta_b)\}$ *in distribution as* $N_b = \sum_{j=1}^{b} n_j \to \infty$, *where* $\mathbb{J}(\beta_b) = \mathbb{S}^{\mathrm{T}}(\beta_b)\mathbb{V}^{-1}(\beta_b)\mathbb{S}(\beta_b)$ *is the Godambe information matrix.*

The proofs of these theorems are given in the Supplementary Material. Importantly, the asymptotic covariance matrix of the online estimator $\tilde{\beta}_b$ in Theorem 2 is exactly the same as that of the offline estimator $\hat{\beta}_b^{\star}$. This implies that the proposed online estimator achieves the same asymptotic distribution as its offline counterpart. Without reaccessing previous historical data, we use aggregated inferential matrices $\tilde{S}_b$ and $\tilde{V}_b$ to estimate the Godambe information matrix by $\tilde{\mathbb{J}}(\beta_b) = (-\log q \tilde{S}_b^{\mathrm{T}} \tilde{V}_b^{-1} \tilde{S}_b / N_b)$. Then the estimated variance of $\tilde{\beta}_b$ is $\tilde{\mathrm{var}}(\tilde{\beta}_b) = \{-\log q \tilde{\mathbb{J}}(\beta_b)/N_b\}^{-1} = (\tilde{S}_b^{\mathrm{T}} \tilde{V}_b^{-1} \tilde{S}_b)^{-1}$.

4. Simulations

4.1. *Simulation setting*

In this section, we examine the finite-sample performance of the proposed streaming estimator $\tilde{\beta}_b$ in (7) and its estimated covariance $\tilde{\mathrm{var}}(\tilde{\beta}_b)$ in Theorem 2 through simulations. In all numerical experiments, we consider a sequence of equally spaced updating time-points $t_1, t_2, \ldots, t_b$ at which the data batches are collected. We assume that at a certain time t_j, a batch of n_j observations is collected, and the weight applied to observations in this batch is q^{b-j}, where q can be either a fixed value in $(0, 1)$ or adaptively chosen at batch b using q_b^{opt} in (8). All simulations are run on one central processing unit with 1 GB of random-access memory. We consider two sets of simulations in the linear and logistic models here. An additional simulation in the Poisson regression setting, reported in the Supplementary Material, corroborates the findings of the linear and logistic regression simulations. This Poisson simulation mimics the data analysed in § 5.

We simulate $b = 200$ batches of size $n_j = 20$. The covariates $X_{ij} = (x_{i,kj}^{\mathrm{T}})_{k=1}^{n_j}$ consist of an intercept and two longitudinal covariates independently simulated from a standard n_j-variate normal distribution. We allow the effect of the first covariate to be batch-heterogeneous. The tuning parameter q_j^{opt} is selected using (8) from the candidate set $\mathcal{C}_q = \exp(-ab^{0.3})$, where a is a sequence of 20 evenly spaced scalars in $[0.1, 1]$. We report the

Table 1. *Simulation metrics for the streaming estimator of the first simulation, with $m = 100$ and $b = 200$ batches of size $n_j = 20$*

Covariate	Error $\times 10^{-1}$	Deviation $\times 10^{-1}$	Bias $\times 10^{-3}$	Coverage	Length $\times 10^{-1}$
Intercept	1.24	1.24	4.30	0.95	6.72
X_1	0.25	0.25	0.44	0.97	1.67
X_2	0.25	0.25	−0.77	0.94	1.39

Error, root mean squared error; Deviation, empirical standard error; Bias, averaged bias; Coverage, coverage probability of the 95% confidence interval; Length, length of the 95% confidence interval.

root mean squared error, empirical standard error, bias, 95% confidence interval coverage and length of $\tilde{\beta}_b$ in the last batch averaged over 500 simulation replicates. In the Supplementary Material, plots of the estimated heterogeneous regression coefficient in all batches and simulations visually confirm that estimation consistency is stable across batches.

To evaluate the statistical performance of our streaming estimator $\tilde{\beta}_b$, we make comparisons with an online estimator derived under an independence working correlation matrix, i.e., only one basis matrix M_1 corresponding to the identity matrix; it is computed using the same selection procedure for q_j^{opt} as our streaming estimator. To evaluate the computational efficiency of our streaming estimator $\tilde{\beta}_b$, we compare it with an offline estimator in the logistic model. The mean elapsed times for our streaming estimator are 3, 7, 10 and 84 times faster than the offline estimator with $b = 5, 10, 15$ and 100, respectively. The simulation results of these statistical and computational performance comparisons are reported in the Supplementary Material.

4.2. *Linear regression simulation results*

In the first simulation, we consider the linear regression setting $E(y_{i,kj} \mid x_{i,kj}) = x_{i,kj}^{\mathrm{T}}\beta_j$, where $x_{i,kj}$ is the p-dimensional covariate vector for participant i at the kth observation in batch j, for $k = 1, \ldots, n_j$, $j = 1, \ldots, b$ and $i = 1, \ldots, m$ with $m = 100$. We take the covariate effects to be $\beta_j = \{0.2, \sin(2\pi j/b), 0.5\}^{\mathrm{T}}$. We simulate $y_i = (y_{i,kj})_{k,j=1}^{n_j,b}$ from a normal distribution with mean $X_{ij}\beta_j$ and covariance Σ jointly over b batches to control the correlation structure across batches, where Σ corresponds to a first-order autoregressive covariance structure with variance $\sigma^2 = 4$ and correlation $\rho = 0.8$. Table 1 reports the evaluation metrics for $\tilde{\beta}_b$.

From Table 1 it can be seen that the bias of $\tilde{\beta}_b$ is negligible and the root mean squared error of $\tilde{\beta}_b$ approximates the empirical standard error, meeting our expectations from the theoretical results on consistency and asymptotic normality. We observe appropriate 95% confidence interval coverage, supporting the inferential properties of our estimator in finite-b settings.

Simulation metrics for the proposed estimator $\tilde{\beta}_b$ under an independence working correlation structure are presented in the Supplementary Material. Despite achieving nominal coverage, this estimator with a misspecified correlation structure has 95% confidence intervals for covariates X_1 and X_2 that are, on average, approximately 20% longer. Indeed, by accounting for dependence, the streaming estimator is more efficient, highlighting the statistical efficiency gains of our approach. Furthermore, to show the effect of violating Assumption 5, we include in the Supplementary Material an additional simulation with a nonsmoothly varying coefficient.

Table 2. *Simulation metrics for the streaming estimator in the second simulation, with $m = 100$ and $b = 200$ batches of size $n_j = 20$*

Covariate	Error $\times 10^{-1}$	Deviation $\times 10^{-1}$	Bias $\times 10^{-3}$	Coverage	Length $\times 10^{-1}$
Intercept	1.04	1.04	1.42	0.95	4.97
X_1	0.36	0.36	0.93	0.97	1.71
X_2	0.46	0.46	3.57	0.96	2.09

4.3. *Logistic regression simulation results*

In the second simulation, we consider the marginal logistic regression setting $E(y_{i,kj} \mid x_{i,kj}) = \exp(x_{i,kj}^{\mathrm{T}}\beta_j)/\{1 + \exp(x_{i,kj}^{\mathrm{T}}\beta_j)\}$, where $x_{i,kj}$ is the p-dimensional covariate vector for participant i at observation k in batch j, for $k = 1, \ldots, n_j$, $j = 1, \ldots, b$ and $i = 1, \ldots, m$ with $m = 100$. We take the covariate effects to be $\beta_j = \{0.2, 4j(1 - j/b)/b, 0.5\}^{\mathrm{T}}$. We simulate $y_i = (y_{i,kj})_{k,j=1}^{n_j,b}$ from a multivariate Bernoulli distribution with mean $\exp(x_{i,kj}^{\mathrm{T}}\beta_j)/\{1 + \exp(x_{i,kj}^{\mathrm{T}}\beta_j)\}$ using the `R` (R Development Core Team, 2023) package `SimCorMultRes` with latent first-order autoregressive covariance structure with variance $\sigma^2 = 4$ and correlation $\rho = 0.8$. Table 2 reports the evaluation metrics for $\tilde{\beta}_b$. The root mean squared error approximates the empirical standard error well and the bias is negligible. We observe appropriate 95% confidence interval coverage. We again conclude that the asymptotic properties of $\tilde{\beta}_b$ in Theorems 1 and 2 appear to hold in finite-b settings. This simulation supports the use of our streaming estimator in marginal generalized linear models.

Simulation metrics for the online estimator with misspecified independence working structure are presented in the Supplementary Material. This estimator has 95% confidence intervals for covariates X_1 and X_2 that are, on average, approximately 5% longer, corroborating our findings from the first simulation that our streaming estimator results in more efficient inference by leveraging the dependence structure of the outcomes. We emphasize that it is difficult to simulate Bernoulli outcomes with an exact autoregressive correlation structure, so the apparently small gain in efficiency is likely due to this imprecision, and would be more significant for outcomes with an exact autoregressive correlation structure.

5. Analysis of accelerometer data

We return to the motivating example introduced in § 1. Leroux et al. (2019) and the accompanying `R` package `rnhanesdata` provide the data and a detailed pipeline for processing and analysing the National Health and Nutrition Examination Survey accelerometry data. Following their proposed pre-processing pipeline, we analyse accelerometer activity counts for $m = 1642$ study participants. For each participant $i = 1, \ldots, m$, the outcome consists of activity counts for 1440 minutes per day over seven days, yielding a total of 10 080 longitudinal outcomes. The outcome data for one participant are visualized in Fig. 1.

Using our proposed streaming estimator, we investigate the association between activity counts and diseases, adjusting for covariates through the Poisson regression model $\log\{E(y_{i,kj} \mid x_{i,kj})\} = \alpha_j + x_{i,kj}^{\mathrm{T}}\beta_j$ $(k = 1, \ldots, n_j; j = 1, \ldots, b)$, with batches of size $n_j = 120$, corresponding to two hours, for $j = 1, \ldots, 84$, with $q_j^{\mathrm{opt}} = 10^{-5}$ considered fixed. The choice of this window size is based on preliminary analysis of the data, which shows that the activity counts have low variability over two-hour intervals. The covariates $x_{i,kj}$ are body mass index, BMI, with mean 28.7 and standard deviation 5.7; presence of coronary heart disease,

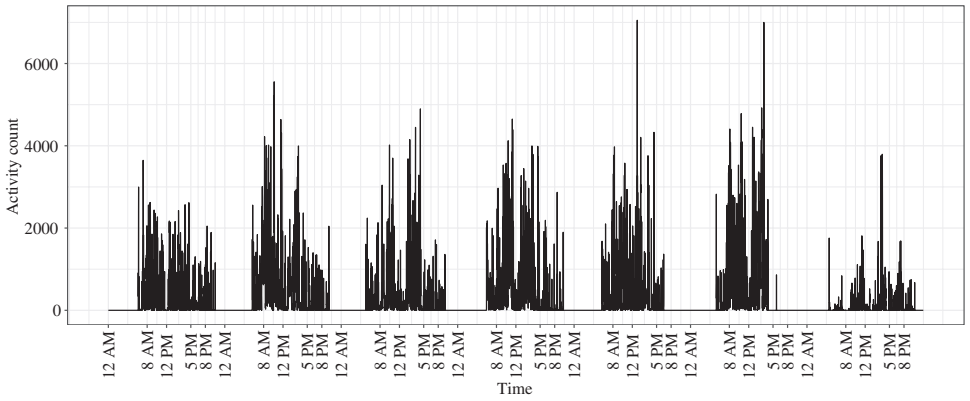

Fig. 1. Outcome data for one study participant. The vertical axis represents the activity count measured by the accelerometer, and the horizontal axis shows the time of day over seven consecutive days.

CHD, where 0 indicates no and 1 indicates yes; presence of coronary heart failure, CHF, where 0 indicates no and 1 indicates yes; presence of cancer, where 0 indicates no and 1 indicates yes; presence of stroke, where 0 indicates no and 1 indicates yes; presence of diabetes, where 0 indicates no and 1 indicates yes; sex, where 0 indicates male and 1 female; education, where 0 indicates high school or less and 1 indicates more than high school; and self-reported mobility difficulties, where 0 indicates none and 1 some. These are all baseline covariates that do not vary across the seven-day window.

The estimated regression coefficients along with 95% confidence intervals are visualized in Fig. 2; trace plots of p-values, and estimated effect signs across time are presented in the Supplementary Material. The trends observed over time across the week are consistent with our intuition: we would expect to see cyclical associations that reflect diurnal and nocturnal activity patterns. This is especially evident for the intercept, which appears to capture intrinsic variations of activity over time that are not captured by our covariates. On the whole, covariate effects are primarily negative across the week, with the exception of education. Indeed, it is not surprising that diseases such as coronary heart disease and coronary heart failure are negatively associated with physical activity. Women also appear to be less physically active than men throughout the week. Moreover, the magnitude of the sex effect is greater in the early morning, indicating that males are physically more active than females during these time periods.

Trace plots of p-values are also useful for identifying time periods in which a certain disease or confounder may be more important. Unsurprisingly, participants with mobility difficulties are significantly less physically active than those without through most of the day, except around midnight when participants are presumably asleep. There appears to be a negative association between coronary heart disease and activity during the night, suggesting that participants with coronary heart disease may be more restless during the night. Sex appears to be more strongly associated with physical activity in the mornings than in the afternoons.

The Poisson simulation setting in the Supplementary Material supports our discussion of significance levels in this data analysis. Nonetheless, we recommend caution when interpreting confidence intervals that have been computed multiple times for different batches, because Type I errors may not be well controlled owing to multiple testing. In practice, α

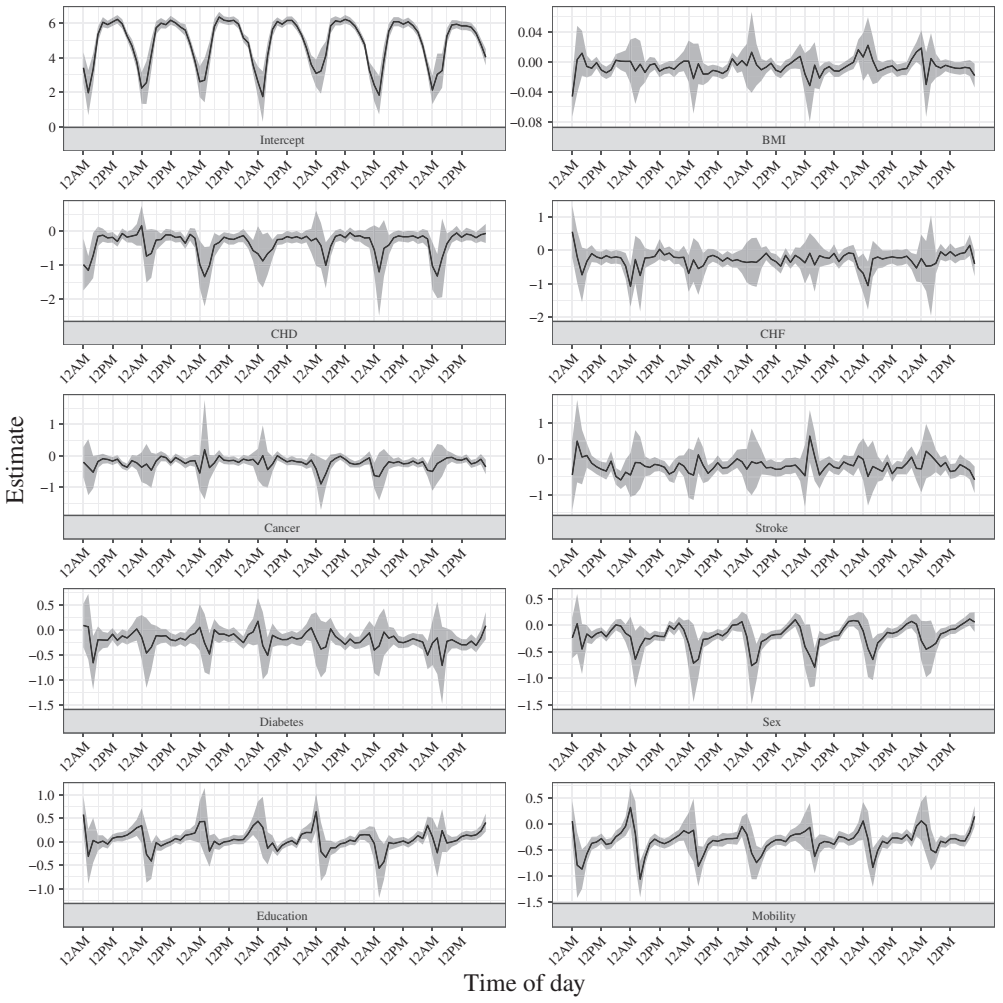

Fig. 2. Trace plots of estimated covariate effects across seven days, along with 95% confidence intervals.

spending functions (DeMets & Lan, 1994) may be of use for inference at multiple timepoints. Finally, our approach has been developed under the quadratic inference function framework, an extension of the quasilikelihood approach, rather than maximum likelihood estimation. The resulting sandwich estimator of the covariance automatically accounts for potential overdispersion in count data.

6. Discussion

The proposed streaming approach is derived with two basis matrices that approximate the inverse of a first-order autoregressive working correlation matrix, i.e., $R(\alpha)^{-1} = \gamma_1 M_1 + \gamma_2 M_2$ where M_1 is an identity matrix, while M_2 has 1 on the two main off-diagonals and 0 elsewhere. We do not assume that the underlying data-generating process is exactly first-order autoregressive. If, however, the true correlation matrix R^{-1} is in the set of matrices $\{\gamma_1 M_1 + \gamma_2 M_2 : \gamma_1, \gamma_2 \in \mathbb{R}\}$, then our approach is fully efficient (Qu et al., 2000). For example, if the true correlation structure is independent, our approach is statistically efficient;

this can be obtained from our basis matrix approximation by setting $\gamma_2 = 0$. In this work we have chosen to use the first-order autoregressive structure, because it is the most suitable for modelling the correlation between longitudinal measurements collected by wearable devices.

To extend our current framework, more basis matrices can be added to accommodate more correlation structures. Similar recursive procedures may be derived with a different set of basis matrices; the main idea remains to partition the basis matrices by data batches so that a matrix of dimension $N \times N$ decomposes into a summation of submatrices of dimension $n_j \times n_j$. Increasing the number of basis matrices will lead to a computationally more intensive algorithm that may depend on more data from previous batches.

In practice, the choice of the data batch size n_j should be based on the data pattern and practical needs. First, if the covariate curve is expected to fluctuate substantially over time, and more local dynamics need to be captured, then a smaller n_j and an adaptively chosen q are more appropriate. There is, however, a trade-off between capturing local dynamics and computational efficiency; if n_j is very small, say $n_j = 1$, choosing q adaptively at every single data point will be computationally intensive. The choice of n_j also depends on a preferred updating frequency. If we wish to obtain the updated results with a relatively low frequency, we could use a larger batch size which is computationally more efficient.

Acknowledgement

We thank the participants of the National Health and Nutrition Examination Survey. We are grateful to the associate editor and reviewers for their comments, which have led to substantial improvements of the manuscript. Correspondence should be addressed to the third author.

Supplementary material

The Supplementary Material includes proofs of the theorems, additional simulation and data analysis results, and an `R` package.

References

Cappé, O. (2011). Online EM algorithm for hidden Markov models. *J. Comp. Graph. Statist.* **20**, 728–49.

Chen, Y., Dong, G., Han, J., Pei, J., Wah, B. W. & Wang, J. (2006). Regression cubes with lossless compression and aggregation. *IEEE Trans. Know. Data Eng.* **18**, 1585–99.

Crowder, M. (1995). On the use of a working correlation matrix in using generalised linear models for repeated measures. *Biometrika* **82**, 407–10.

DeMets, D. L. & Lan, K. G. (1994). Interim analysis: The alpha spending function approach. *Statist. Med.* **13**, 1341–52.

Duan, R., Ning, Y. & Chen, Y. (2022). Heterogeneity-aware and communication-efficient distributed statistical inference. *Biometrika* **109**, 67–83.

Fan, J. & Zhang, W. (1999). Statistical estimation in varying coefficient models. *Ann. Statist.* **27**, 1491–518.

Fang, Y. (2019). Scalable statistical inference for averaged implicit stochastic gradient descent. *Scand. J. Statist.* **46**, 987–1002.

Hansen, L. P. (1982). Large sample properties of generalized method of moments estimators. *Econometrica* **50**, 1029–54.

Hector, E. C., Luo, L. & Song, P. X.-K. (2023). Parallel-and-stream accelerator for computationally fast supervised learning. *Comp. Statist. Data Anal.* **177**, 107587.

Hector, E. C. & Song, P. X.-K. (2020). A distributed and integrated method of moments for high-dimensional correlated data analysis. *J. Am. Statist. Assoc.* **116**, 805–18.

Joe, H. (2014). *Dependence Modeling with Copulas*. Boca Raton, Florida: Chapman & Hall/CRC.
Jordan, M. I., Lee, J. D. & Yang, Y. (2019). Communication-efficient distributed statistical inference. *J. Am. Statist. Assoc.* **114**, 668–81.
Leroux, A., Di, J., Smirnova, E., Mcguffey, E. J., Cao, Q., Bayatmokhtari, E., Tabacu, L., Zipunnikov, V., Urbanek, J. K. & Crainiceanu, C. (2019). Organizing and analyzing the activity data in NHANES. *Statist. Biosci.* **11**, 262–87.
Liang, K.-Y. & Zeger, S. L. (1986). Longitudinal data analysis using generalized linear models. *Biometrika* **73**, 13–22.
Lin, N. & Xi, R. (2011). Aggregated estimating equation estimation. *Statist. Interface* **4**, 73–83.
Lindsay, B. G. (1988). Composite likelihood methods. *Contemp. Math.* **80**, 220–39.
Luo, L. & Li, L. (2022). Online two-way estimation and inference via linear mixed-effects models. *Statist. Med.* **41**, 5113–33.
Luo, L. & Song, P. X.-K. (2020). Renewable estimation and incremental inference in generalized linear models with streaming datasets. *J. R. Statist. Soc.* B **82**, 69–97.
Luo, L. & Song, P. X.-K. (2023). Multivariate online regression analysis with heterogeneous streaming data. *Can. J. Statist.* **51**, 111–33.
Luo, L., Zhou, L. & Song, P. X.-K. (2023). Real-time regression analysis of streaming clustered data with possible abnormal data batches. *J. Am. Statist. Assoc.* **118**, 2029–44.
Qu, A., Lindsay, B. G. & Li, B. (2000). Improving generalised estimating equations using quadratic inference functions. *Biometrika* **87**, 823–36.
R Development Core Team (2023). *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing. ISBN 3-900051-07-0, http://www.R-project.org.
Robbins, H. & Monro, S. (1951). A stochastic approximation method. *Ann. Math. Statist.* **22**, 400–7.
Sakrison, D. J. (1965). Efficient recursive estimation: Application to estimating the parameter of a covariance function. *Int. J. Eng. Sci.* **3**, 461–83.
Schifano, E. D., Wu, J., Wang, C., Yan, J. & Chen, M.-H. (2016). Online updating of statistical inference in the big data setting. *Technometrics* **58**, 393–403.
Song, P. X.-K. (2007). *Correlated Data Analysis: Modeling, Analytics, and Applications*. New York: Springer.
Song, P. X.-K., Jiang, Z., Park, E. & Qu, A. (2009). Quadratic inference functions in marginal models for longitudinal data. *Statist. Med.* **28**, 3683–96.
Stengel, R. F. (1994). *Optimal Control and Estimation*. New York: Dover Publications.
Toulis, P. & Airoldi, E. M. (2017). Asymptotic and finite-sample properties of estimators based on stochastic gradients. *Ann. Statist.* **45**, 1694–727.
Varin, C., Reid, N. & Firth, D. (2011). An overview of composite likelihood methods. *Statist. Sinica* **21**, 5–42.
Xie, M. & Singh, K. (2013). Confidence distribution, the frequentist distribution estimator of a parameter: A review. *Int. Statist. Rev.* **81**, 3–39.
Zhou, L. & Song, P. X.-K. (2017). Scalable and efficient statistical inference with estimating functions in the MapReduce paradigm for big data. *arXiv:* 1709.04389.

[*Received on* 10 *August* 2021. *Editorial decision on* 6 *February* 2023]