Proceedings of the 18th World Congress
The International Federation of Automatic Control
Milano (Italy) August 28 - September 2, 2011

Scalable Statistical Monitoring of Fleet Data

Eric Chu [*], **Dimitry Gorinevsky** [**], and **Stephen Boyd** [*]

[*] *Electrical Engineering Department, Stanford University*
e-mail: {echu508, boyd}@stanford.edu
[**] *Mitek Analytics LLC, Palo Alto, CA*
e-mail: dimitry@mitekan.com

Abstract: This paper considers the problem of fitting regression models to historical fleet data with mixed effects, which arises in the context of statistical monitoring of data from a fleet (population) of similar units. A fleet-wide extension of the multivariable statistical process control approach is used to monitor for three different types of faults: a performance anomaly, a performance shift, and an anomalous unit. Our formulation requires the solution of a least-squares problem with very large numbers of both regressors (variables) and data measurements. For problems of interest, this least-squares problem cannot be solved using standard methods. We propose a method for solving the problem that is scalable to extremely large datasets, even ones that do not fit in to the memory of a single computer system. Our method can be parallelized, but also works serially on a single processor. This approach is demonstrated in a simulated example for monitoring a fleet of aircraft from historical cruise flight data.

1. INTRODUCTION

1.1 Population monitoring problems

This paper considers statistical monitoring of data generated by a population of N homogeneous units such as vehicles or aircrafts. We have a dataset

$$\left\{ \{x_i(t), y_i(t)\}_{t=1}^{T_i} \right\}_{i=1}^{N}, \tag{1}$$

where i numbers the unit in the population, and the integer t is the occasion (time) when the data from unit i is collected. For each unit i, there are T_i occasions (time samples). The independent variables $x_i(t) \in \mathbf{R}^n$ are the input measurements; these can be thought of as measurements of the operating conditions of unit i at some time t. The dependent variables $y_i(t) \in \mathbf{R}^m$ are the output measurements; these can be thought of as measurements of the performance (quality) of unit i at time t.

This paper discusses *population-wide statistical monitoring*. The goal is to detect and report three types of anomalies in the data (1). These anomalies are:

A1 a performance anomaly in unit i at time t,
A2 a performance shift in unit i at time t,
A3 an anomalous unit, *i.e.*, one that consistently performs differently from the rest of the population.

The entire historical dataset (1) is used for the monitoring computations.

In the special case of $N = 1$ (a single unit in the population), anomalies A1 and A2 are addressed by well known methods of *multivariable statistical process control* (MSPC) such as Hotelling T^2 statistics (also known as a multivariable Shewhart chart) and multivariate *exponentially weighted moving average* (EWMA) methods, *e.g.*, see NIST [2010].

The special case of $T_i = 1$ (a single measurement for each unit) can be addressed by computing the T^2 statistics for each unit. The standard monitoring approach is to build a regression model from the series of the input/output data and then use MSPC methods to monitor model prediction residuals.

This paper extends the MSPC approach to the entire population. One challenge is to monitor the data simultaneously in time and across the population. Another challenge is that population-wide multivariable data sets can be very large and impossible to keep and process in computer memory. The main contribution of this paper is in addressing these two challenges and presenting a scalable statistical monitoring approach for population-wide data.

Section 4 demonstrates the proposed approach in the aircraft fleet monitoring example. In this motivating example, subscript i refers to the tail number of an aircraft in the fleet; t is the consecutive flight number when the data is collected; and T_i is the total number of flights in the database for aircraft i. In this context, anomaly A1 means that an abnormal event occurred in aircraft i during flight t. Anomaly A2 means that a shift of the aircraft performance persists through the recent flights. Anomaly A3 means that performance of aircraft i is consistently different from the rest of the fleet.

[⋆] This work was supported in part by NASA Grant NNX07AEIIA and NASA Contract NNA08BC21C.

978-3-902661-93-7/11/$20.00 © 2011 IFAC

10.3182/20110828-6-IT-1002.01620

1.2 Regression formulation

Population-wide monitoring uses a linear regression model that relates the inputs $x_i(t)$ and outputs $y_i(t)$ in (1) by the equation

$$y_i(t) = \beta_i x_i(t) + a_i(t) + r_i(t), \qquad (2)$$

where $\beta_i \in \mathbf{R}^{m\times n}$ and $a_i(t) \in \mathbf{R}^m$ are parameters of the regression model, and $r_i(t) \in \mathbf{R}^m$ is the residual for the model for unit i at time t. The parameter matrix β_i (which does not depend on time) gives the specialized regression model for unit i. The parameter vectors $a_i(t)$ can describe offsets (also called biases or fixed effects), and presumably are chosen to be slowly varying; time-variation in $a_i(t)$ models shifts or trends in performance. Finally, $r_i(t)$ is the residual, which can be interpreted as noise or model fit error. Our regression model (2) is linear in $x_i(t)$; but as in all regression applications, components of the regressor vector $x_i(t)$ can be nonlinear functions of some 'raw' explanatory parameters.

In this paper, we choose the regression parameters so as to minimize a quadratic objective function,

$$\text{minimize } J^{\text{res}} + \kappa J^{\text{shift}} + \mu J^{\text{unit}}, \qquad (3)$$

where κ and μ are positive weights, and the objective terms are as follows:

$$J^{\text{res}} = \sum_{i=1}^{N}\sum_{t=1}^{T_i} \|r_i(t)\|_2^2$$

is the total square residual,

$$J^{\text{shift}} = \sum_{i=1}^{N}\sum_{t=2}^{T_i} \|a_i(t) - a_i(t-1)\|_2^2$$

is the sum of squares of the change in offset, and

$$J^{\text{unit}} = \sum_{i=1}^{N} \|\beta_i - \bar\beta\|_F^2$$

is the variance of the parameter matrices. Here, $\|\cdot\|_F$ denotes the Frobenius norm, which is the square root of the sum of the squares of the entires: $\|A\|_F = (\sum_{i,j} A_{ij}^2)^{1/2}$, and $\bar\beta = (1/N)\sum_{i=1}^N \beta_i$ is the average of the parameter matrices. The variables in this optimization problem are the parameter matrices $\beta_i \in \mathbf{R}^{m\times n}$, $i = 1,\dots,N$, and the offsets $a_i(t) \in \mathbf{R}^m$, $t = 1,\dots,T_i$, $i = 1,\dots,N$.

We will denote the solution to (3) as $\bar\beta^\star$, $\beta_i^\star$, and $a_i^\star(t)$, $t = 1,\dots,T_i$ and $i = 1,\dots,N$. We denote the associated residuals as $r_i^\star(t)$.

The three objective terms are directly related to the three types of anomalies we wish to detect.

- If $r_i^\star(t)$ is large, then we have anomaly A1.
- If $a_i^\star(t)$ is large, then we have anomaly A2.
- If $\beta_i^\star - \bar\beta^\star$ is large, then we have anomaly A3.

We will later say more precisely what 'large' means.

The problem (3) involves a large number of measurements and variables. The dataset (1) contains a total number of (scalar) output measurements

$$M^{\text{tot}} = m\sum_{i=1}^{N} T_i.$$

In the regression problem (3), there are mnN (scalar) unknowns in β_i, and $m\sum_{i=1}^N T_i$ (scalar) unknowns in the offsets $a_i(t)$. The total number of (scalar) variables is therefore

$$N^{\text{tot}} = m\left(nN + \sum_{i=1}^{N} T_i\right).$$

Note that the number of measurements and variables can become very large for problems of interest.

The focus of this paper is on efficiently solving problem (3) for large values of M^{tot} and N^{tot} and using the regression results to monitor the three different anomalies.

1.3 Previous work

The above formulation is a slight variation on standard mixed effect regression for longitudinal data. In a fixed effects regression problem, the main regression model is assumed to be the same across the population so that $\beta_i = \beta_j$ for all i, j; in addition, the offsets $a_i(t)$ are constant for each unit i but vary across the population.

The regression models that involve cross-sectional data depending on time and on unit (individual) in a population are well known and used in econometrics, sociology, agriculture, biology, and medicine. Analysis of longitudinal data is discussed in the books Sayrs [1989], Hsiao [1986], and Greene [2007]. Multilevel regression is discussed in the texts Goldstein [1999] and de Leeuw and Meijer [2007].

Several computational tools exist for solving these types of regression problems. These include the software packages `GLLAMM` and `RRGibbs` which uses Gibbs sampling to approximate the posterior density of the regressors and reduce the scale of the regression solution [Rabe-Hesketh et al., 2008, Meyer, 2007, 2002]. The commercial analytics software package SAS has the ability to perform fixed effects regression [Allison, 2006]. There are statistical functions in R for solving mixed-effects regression [R, 2010, Fox, 2002].

While these packages can solve mixed regression problems of moderate size, they do not scale well to large problems. For least-squares regression with a relatively small number of parameters, large amounts of data can be processed by employing iterative least squares, yet problems with large number of regression parameters might be hard to scale. For several hundreds or thousands of regressors, the normal equations of the least square problem can be solved using standard linear algebra codes. For more regressors and a sparse problem, an iterative method such as LSQR [Paige and Saunders, 1982] might be a solution. These methods can be scaled to an extent via brute force—by use of powerful computers with more memory. For example, SAS takes about an hour to solve a problem with 100,000 regressors on a custom computer system.

In all these cases, it is required that the regression parameters fit in to the memory of a single computer. In the more general case, if the parameters do not fit in memory, scalability is achieved by distributing the data, regression parameters, and computations over several computers or over several iterations on the same computer, or both. Though there is much work on distributed computing, there seems to be little prior work on a scalable solution of mixed regression problems with longitudinal data. Distributed regression algorithms are considered in Guestrin et al. [2004], Bhaduri and Kargupta [2008], and Bazerque et al. [2010]. These algorithms are motivated by sensor networks and the need to distribute the estimation computations over the network, rather than scalability needs.

There seems to be little related work on population-wide monitoring. Our formulation is related to data monitoring methods known as profile monitoring that have been used with two-level regression models. Profile monitoring involves fitting parametric or nonparametric models to longitudinal or profile data and monitoring these models. Some applications are discussed in Wang and Tsung [2005], Shiau et al. [2009], Jensen et al. [2006], and Mosesova et al. [2006]. In Wang and Tsung [2005], the authors propose an SPC method to monitor processes with large amounts of data. Their motivation is related to ours, yet they only consider problems with a relatively small number of regression variables.

This paper proposes a scalable, distributed formulation to solve the two-level, mixed-effects regression problem for a large-scale problem. In Section 4, we apply it in an example with $M^{\text{tot}} = 1,000,000$ (1 million pieces of data) and $N^{\text{tot}} = 1,000,800$ (about 1 million regression parameters). The data can be easily distributed over multiple computers and a fast solution of the regression problem is provided. Interestingly enough, the proposed approach is fast even when used on a single machine that iteratively reads and processes in-memory chunks of a large dataset on the disk. For this example a solution is computed in about two seconds on a single machine, much faster and beyond the scale allowed by the existing software packages.

While most previous monitoring work has focused on problems with large datasets, our algorithm is able to solve problems with large datasets *and* many regressors. Computing the regression parameters quickly over the large population-wide datasets enables monitoring of the residuals and trends for the fitted models.

The contribution of this paper, then, is a *scalable* regression algorithm with mixed effects and longitudinal data that can be used for fleet-wide monitoring.

2. SCALABLE SOLUTION

Since the objective in (3) is quadratic, the regression parameters $\bar{\beta}$, β_i, and $a_i(t)$, $t = 1, \ldots, T_i$ and $i = 1, \ldots, N$ can be found by solving the linear system of normal equations

$$\mu(\beta_i - \bar{\beta})^T + X_i X_i^T \beta_i^T + X_i \alpha_i^T = X_i Y_i^T \tag{4a}$$

$$X_i^T \beta_i^T + \alpha_i^T + \kappa D_i^T D_i \alpha_i^T = Y_i^T, \tag{4b}$$

for $i = 1, \ldots, N$, where D_i is an appropriately sized finite difference matrix and $\bar{\beta} = (1/N)\sum_{i=1}^{N} \beta_i$.

The data matrices X_i and Y_i, $i = 1, \ldots, N$, and the regression parameter matrix α_i, $i = 1, \ldots, N$ are defined as follows:

$$\begin{aligned} X_i &= [\, x_i(1)\ x_i(2)\ \cdots\ x_i(T_i)\,], \\ Y_i &= [\, y_i(1)\ y_i(2)\ \cdots\ y_i(T_i)\,], \\ \alpha_i &= [\, a_i(1)\ a_i(2)\ \cdots\ a_i(T_i)\,]. \end{aligned}$$

The normal equations (4) have a block sparsity structure. If the equations fit into computer memory, their solution can be found directly using a sparse solver. However, if N is large enough, we can no longer simultaneously solve for β_i and α_i.

Note that the normal equations for the N units are only coupled through the average parameter matrix, $\bar{\beta}$. If $\bar{\beta}$ were known, the solutions could be found independently for each unit as

$$\beta_i^T = Q_i^{-1}(X_i(I - P_i^{-1})Y_i^T + \mu\bar{\beta}^T) \tag{5a}$$

$$\alpha_i^T = P_i^{-1}(Y_i^T - X_i^T \beta_i^T) \tag{5b}$$

where matrices $P_i \in \mathbf{R}^{T_i \times T_i}$ and $Q_i \in \mathbf{R}^{n \times n}$ for unit i are defined through the data X_i, Y_i, and the bi-diagonal time-difference operator D_i as

$$\begin{aligned} P_i &= I + \kappa D_i^T D_i \\ Q_i &= \mu I + X_i(I - P_i^{-1})X_i^T. \end{aligned} \tag{6}$$

For smaller problems, the system of normal equations in (4) can be solved on a single machine. For larger regression problems, the sizes of the regression variables $a_i(t)$ and β_i and of the problem data can make solving (4) computationally intensive.

For example, consider the dataset of section 4 with $M^{\text{tot}} = 1,000,000$ and $N^{\text{tot}} = 1,000,800$. A naive approach to solving (3) requires forming the normal equations in (4), which costs $O((N^{\text{tot}})^2 M^{\text{tot}})$ flops, and solving the equations, which requires $O((N^{\text{tot}})^3)$ flops. Since $M^{\text{tot}} \approx N^{\text{tot}}$, solving (4) requires $O((N^{\text{tot}})^2 M^{\text{tot}})$ flops.

For other applications, m and n might be larger, and N and T_i might be significantly larger, with the data stored in a large database system. Therefore, there is a need for an algorithm that solves (3) and scales with the data.

A scalable solution can be obtained by observing that (5) expresses α_i and β_i as affine functions of $\bar{\beta}$. Substituting (5) into the equation $\bar{\beta} = (1/N)\sum_{i=1}^{N} \beta_i$ and solving for $\bar{\beta}$ yields

$$\bar{\beta}^T = R^{-1} \sum_{i=1}^{N} Q_i^{-1} X_i (I - P_i^{-1}) Y_i^T, \tag{7}$$

where the matrix R is defined as

$$R = NI - \mu \sum_{i=1}^{N} Q_i^{-1}.$$

The proposed scalable solution method works as follows.

Step 1. For each unit i, local data X_i and Y_i are used to compute the matrices $Q_i^{-1} \in \mathbf{R}^{n \times n}$, as described in

(6), and $Q_i^{-1}X_i(I - P_i^{-1})Y_i^T \in \mathbf{R}^{n\times m}$. The matrix Q_i is invertible since it is symmetric positive definite. Forming Q_i requires approximately $(4n^2 + 4nk^2)T$ flops, where $k = 3$ is the bandwidth of P_i, and we assume $T_i = T$ for $i = 1, \ldots, N$. Since P_i is tridiagonal and positive definite, it is quickly invertible. Computing Q_i^{-1} will be dominated by the factorization step which costs $(1/3)n^3$ flops. Once we have Q_i^{-1}, computing the second matrix reduces to matrix multiplies and costs approximately $2mn^2 + (4mn + 4mk^2)T$ flops. These operations have linear cost in T (in fact, these matrices can be formed by running sums). The total flop count for this step is multiplied by the number of units N. Since each unit computes the matrices independently, this step can be implemented iteratively or in parallel without the need to access all unit data simultaneously. If each unit computes these matrices locally and in parallel, we amortize the factor of N across the population, and this step takes on the order of $n^3 + T$ flops (assuming n^3, T, and N are of the same order).

Step 2. We gather the matrices computed in the previous step, Q_i^{-1} and $Q_i^{-1}X_i(I - P_i^{-1})Y_i^T$, in a central process; a total of $n(n + m)$ matrix entries for each unit i. These matrices are used to compute $\bar{\beta}$ in accordance with (7). This step is dominated by inverting R, which costs $(1/3)n^3$, and computing sums of the gathered matrices. Since we collect sums of the matrices obtained for different units in (7), computing $\sum_{i=1}^N Q_i^{-1}X_i(I - P_i^{-1})Y_i^T$ has linear cost in the population size N. This step takes on the order of $n^3 + N$ flops.

Step 3. In the last step, $\bar{\beta}$ is broadcast (scattered) by the central computational process. It is used to compute α_i and β_i independently for each unit in accordance with (5). These computations can be distributed and performed in parallel for each unit and only require matrix multiplication. This step takes on the order of T flops (if parallelized).

The main computational bottleneck of the described algorithm might be inverting Q_i for a large number n of input regressors (say, over 100,000). The main data transfer bottleneck could be in gathering $n(n+m)$ matrix entries for each unit, which could be a large number if n is large. Most practical applications, however, have n, the number of input regressors, on the order of 100.

Instead of the naive $O((N^{\rm tot})^2 M^{\rm tot})$ flops needed to solve for $\bar{\beta}^\star$, $\beta_i^\star$, and $a_i^\star(t)$, $i = 1, \ldots, N$, a distributed regression scheme allows the solution to be found on the order of $n^3 + T + N$ flops—linear in the size of the data, assuming n is not too large.

3. MONITORING APPROACH

This section describes the monitoring approach that extends standard multivariate statistical process control (MSPC) to fleet data.

The proposed population-wide monitoring formulation is an extension of standard process control methods. Our method provides three data monitors to detect the anomalies A1, A2, A3 introduced in section 1.1.

A1: Performance anomaly in unit i at time t. To detect abnormal performance of unit i at time t, we compute the Hotelling T^2 statistic for the residual for unit i at time t:

$$T^{\rm res} = (r_i^\star(t) - \bar{r}^\star)^T(\Sigma^{\rm res})^{-1}(r_i^\star(t) - \bar{r}^\star), \qquad (8)$$

where

$$\bar{r}^\star = (1/N^{\rm res})\sum_{i=1}^{N}\sum_{t=1}^{T_i} r_i^\star(t)$$

is the empirical average residual of the training set,

$$\Sigma^{\rm res} = (1/N^{\rm res})\sum_{i=1}^{N}\sum_{t=1}^{T_i}(r_i^\star(t) - \bar{r}^\star)(r_i^\star(t) - \bar{r}^\star)^T$$

is the empirical residual covariance matrix of the training set, and $N^{\rm res} = \sum_{i=1}^N T_i$. Since we can shift $a_i(t)$ by a constant $\bar{r}^\star$ without changing the objective value of (3), then $\bar{r}^\star = 0$.

The standard monitoring approach is to use the F-distribution to find the threshold that corresponds to a particular confidence level [NIST, 2010]. This threshold is given by

$$L^{\rm res} = \frac{m(N^{\rm res}+1)(N^{\rm res}-1)}{N^{\rm res}(N^{\rm res}-m)}F_\alpha(m, N^{\rm res} - m), \qquad (9)$$

where $F_\alpha(m, N^{\rm res} - m)$ is the F-distribution with parameters m and $N^{\rm res} - m$ and α is the desired confidence level. If $T^{\rm res} > L^{\rm res}$, then we guess that a performance anomaly has occurred in unit i at time t.

A2: Performance shift in unit i at time t. To detect a performance shift in the offset $a_i(t)$, we compute the Hotelling T^2 statistic for $a_i^\star(t)$:

$$T^{\rm shift} = (a_i^\star(t) - \bar{a}^\star)^T(\Sigma^{\rm shift})^{-1}(a_i^\star(t) - \bar{a}^\star), \qquad (10)$$

where

$$\bar{a}^\star = (1/N^{\rm shift})\sum_{i=1}^{N}\sum_{t=1}^{T_i} a_i^\star(t)$$

is the empirical average of $a_i^\star(t)$, and

$$\Sigma^{\rm shift} = (1/N^{\rm shift})\sum_{i=1}^{N}\sum_{t=1}^{T_i}(a_i^\star(t) - \bar{a}^\star)(a_i^\star(t) - \bar{a}^\star)^T$$

is the empirical covariance matrix. The scalar value $N^{\rm shift}$ is defined as $N^{\rm shift} = \sum_{i=1}^N T_i$. The threshold $L^{\rm shift}$ is computed as in (9) but with $N^{\rm shift}$ in place of $N^{\rm res}$. If $T^{\rm shift} > L^{\rm shift}$, we guess that a performance shift has occurred in unit i at time t.

A3: An anomalous unit i. To detect an anomalous unit that is performing differently than the population, we compute the Hotelling T^2 statistic for $\beta_i^\star$:

$$T^{\rm unit} = \left(\mathbf{vec}(\beta_i^\star - \bar{\beta}^\star)\right)^T(\Sigma^{\rm unit})^{-1}\left(\mathbf{vec}(\beta_i^\star - \bar{\beta}^\star)\right), \qquad (11)$$

where

$$\Sigma^{\rm unit} = (1/N^{\rm unit})\sum_{i=1}^{N}\left(\mathbf{vec}(\beta_i^\star - \bar{\beta}^\star)\right)\left(\mathbf{vec}(\beta_i^\star - \bar{\beta}^\star)\right)^T$$

is the empirical covariance of $\mathbf{vec}(\beta_i^\star - \bar{\beta}^\star)$ and $N^{\rm unit} = N$. The **vec** operation vectorizes a matrix by stacking its columns together. The threshold $L^{\rm unit}$ is computed as in (9) but with N^{unit} and mn in place of N^{res} and m, respectively. If $T^{\text{unit}} > L^{\text{unit}}$, we guess that unit i is anomalous.

4. EXAMPLE AND RESULTS

To demonstrate the power of the proposed approach, we apply it to simulated aircraft fleet data. The motivation behind this example is to monitor a fleet of aircraft for incipient anomalies. The simulated data closely resembles the *flight operations quality assurance* (FOQA) data that is collected by airlines to improve fleet safety and maintenance.

Linear regression modeling for monitoring aircraft FOQA data is discussed in Chu et al. [2010], where a detailed nonlinear simulation is used to generate the data. In a follow-on to this paper we plan to report results of applying linear regression models with fixed effects in monitoring actual FOQA data for a fleet of many aircraft making hundreds of flights.

The example we discuss below describes a realistic model for monitoring the angle-of-attack channel in many flights of different aircraft. The variables have been scaled and do not correspond to particular engineering measurement units.

From aircraft dynamics, we know that the angle-of-attack in the cruise regime can be approximately explained by the balance between the aircraft weight moment and the moment of aerodynamic forces with respect to a fixed aircraft center. The varying component of the weight moment is created by the aircraft mass deviating (varying) from the mean because of fuel consumption during the flight and also because of flight-to-flight load variation.

Assuming that the center of gravity of the varying component of the mass is fixed, the weight moment is proportional to the mass variation. For small angle-of-attack, the aerodynamic pitching moment is proportional to the dynamic pressure. The aerodynamic coefficient combines the fixed part, the part proportional to the angle-of-attack, and the parts proportional to the deflections of elevator and stabilizer flight control surfaces.

The linear regression model of the channel has angle-of-attack times dynamic pressure as the scalar output $y_i(t)$. It is explained using four regressors: mass variation, dynamic pressure, the stabilizer deflection angle times dynamic pressure, and the elevator deflection angle times dynamic pressure. These regressors are components of the vector $x_i(t) \in \mathbf{R}^4$. Our model considers the average values of the regression variables in the cruise segment of the flight: the scalar output value $y_i(t)$ and the regressor vector $x_i(t)$.

We simulate the angle-of-attack channel with

$$y_i(t) = \beta_i x_i(t) + a_i(t) + r_i(t),$$

where $\beta_i = \bar{\beta} + \Delta_i$ with $\mathbf{vec}(\Delta_i) \sim \mathcal{N}(0, \Sigma_\Delta)$, $x_i(t)$ is drawn from the distribution $\mathcal{N}(\bar{x}, \Sigma_x)$, $r_i(t)$ is noise drawn from the distribution $\mathcal{N}(\bar{r}, \Sigma_r)$, and $a_i(t) = 0$ for flights without faults. Note that the index t does not refer to time, but rather to a single flight.

Our simulations used the following parameters

$$\bar{\beta} = [\,0.80\ \ -2.70\ \ -0.63\ \ 0.46\,],$$

$$\Sigma_\Delta = \begin{bmatrix} 0.04 & 0.12 & -0.02 & 0.02 \\ 0.12 & 0.84 & -0.09 & 0.10 \\ -0.02 & -0.09 & 0.03 & 0.00 \\ 0.02 & 0.10 & 0.00 & 0.05 \end{bmatrix},$$

$$\bar{x} = \begin{bmatrix} 0.95 \\ -1.22 \\ -2.79 \\ 7.11 \end{bmatrix}, \qquad \Sigma_x = \begin{bmatrix} 0.25 & -0.02 & 0.12 & -0.04 \\ -0.02 & 0.45 & 0.03 & -0.52 \\ 0.12 & 0.03 & 1.05 & -1.26 \\ -0.04 & -0.52 & -1.26 & 3.89 \end{bmatrix},$$

$$\bar{r} = -0.03, \qquad \Sigma_r = 0.83.$$

We used the above model to simulate a fleet of $N = 200$ aircraft with $T_i = 5000$ flights for each aircraft and generate $m = 1$ output measurements and $n = 4$ input measurements for each aircraft and each flight. This simulated data corresponds to $M^{\text{tot}} = 1,000,000$ (scalar) measurements in the dataset (components of all $y_i(t)$) and $N^{\text{tot}} = 1,000,800$ variables.

In our simulation experiments, we seed three types of faults for selected aircraft i:

(1) anomaly A1—a performance anomaly, where $r_i(t)$ is large at the last flight t of aircraft i,
(2) anomaly A2—a performance shift, where $a_i(t)$ ramps up to some large value over successive flights of aircraft i,
(3) anomaly A3—an anomalous aircraft, where β_i has a large deviation $\Delta_i = \beta_i - \bar{\beta}$.

The fault magnitudes for these anomalies are chosen to be approximately two times larger than the respective T^2 decision boundary, which is found by evaluating L^{res}, L^{shift}, and L^{unit} using the appropriate parameters and $\alpha = 0.99$. We seed anomalies in 6 of the 200 aircraft, two anomalies of each type.

We use the quadratic formulation of (3) to demonstrate the capabilities of the proposed fleet monitoring approach highlighted in section 3; the parameters $\kappa = 50$ and $\mu = 0.3$ are used. (These are empirically chosen to give the best results.)

Monitoring for performance anomaly, A1. While we could compute (8) for all flights t, we only compute T^{res} (the Hotelling T^2 statistic) for the last flight of each aircraft in this example. The top plot in Figure 1 shows the values of T^{res} (8) evaluated at the last flight for all 200 aircraft. The dashed line indicates L^{res}. The (green) circles indicate aircraft with seeded performance anomalies; they are above the decision boundary, which means the anomalies have been detected.

Monitoring for a performance shift, A2. The middle plot in Figure 1 shows the values of T^{shift} (10) evaluated at the last flight for all 200 aircraft. The dashed line indicates L^{shift}. The (red) squares indicate aircraft with seeded performance shifts; these are also successfully detected.

Monitoring for an anomalous aircraft i, A3. The bottom plot in Figure 1 shows the values of T^{unit} (11) evaluated for all 200 aircraft. The dashed line indicates L^{unit}.

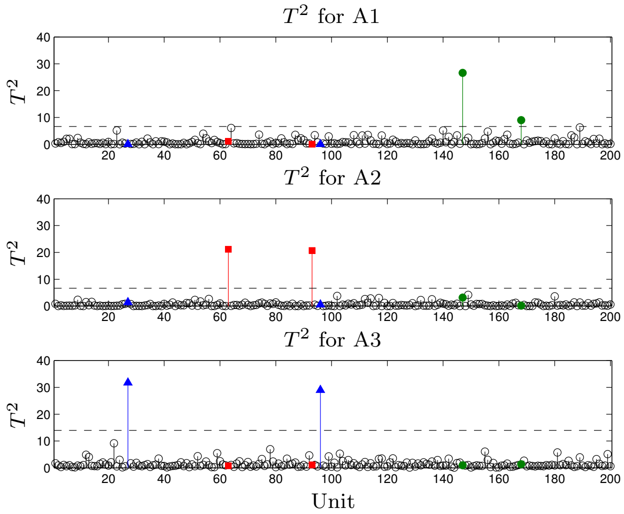

Fig. 1. Hotelling T^2 values for 200 aircraft. The (green) circles show seeded A1 anomalies; (red) squares show seeded A2 anomalies; (blue) triangles show seeded A3 anomalies.

The (blue) triangles indicate aircraft with seeded performance offsets and are well above the decision boundary.

5. CONCLUSION

We have presented an approach to monitor a population of similar units from their historical performance data. Central to the approach is a large-scale regression fit of similar models to data from the units taking into account variation between the individual units and in time. It can be thought of as a form of collaborative filtering.

The regression fit problem (3) can be solved efficiently in one iteration by partial minimizations and Gaussian elimination. This solution can be carried for each unit in parallel (or sequentially) and is scalable to very large datasets for unlimited number of units. This approach is also distributed and fully scalable (computational complexity grows linearly with the amount of data).

We have demonstrated how the regression problem solution allows us to perform population-wide monitoring and detect three types of anomalies in the data: performance anomalies, performance shifts, and anomalous units behaving differently from the rest of the population. The formulated approach effectively allows simultaneous monitoring of an infinite number of units.

REFERENCES

P. D. Allison. Fixed effects regression methods in SAS®. In *Thirty-first Annual SAS® Users Group International Conference*, San Francisco, CA, March 2006.

J. A. Bazerque, G. Mateos, and G. B. Giannakis. Distributed lasso for in-network linear regression. In *2010 IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 2978–2981, Dallas, TX, March 2010.

K. Bhaduri and H. Kargupta. A scalable local algorithm for distributed multivariate regression. *Statistical Analysis and Data Mining*, 1(3), 2008.

E. Chu, D. Gorinevsky, and S. Boyd. Detecting aircraft performance anomalies from cruise flight data. In *AIAA Infotech@Aerospace*, Atlanta, GA, April 2010. AIAA-2010-3307.

J. de Leeuw and E. Meijer, editors. *Handbook of Multilevel Analysis*. Springer, 1st edition, 2007.

J. Fox. Linear mixed models. Appendix to An R and S-PLUS Companion to Applied Regression. Available at `http://cran.r-project.org/doc/contrib/Fox-Companion/appendix-mixed-models.pdf`, 2002.

H. Goldstein. *Multilevel Statistical Models*. London: Institute of Education, 1999.

W. H. Greene. *Econometric Analysis*. Prentice Hall, 2007.

C. Guestrin, R. Thibaux, P. Bodik, M. A. Paskin, and S. Madden. Distributed regression: an efficient framework for modeling sensor network data. In *Information Processing in Sensor Networks (IPSN 2004)*, Urbana-Champaign, IL, April 2004.

C. Hsiao. *Analysis of Panel Data*. New York: Cambridge University Press, 1986.

W. A. Jensen, J. B. Birch, and W. H. Woodall. Profile monitoring via linear mixed models. Technical Report 06-02, Virginia Tech, 2006.

K. Meyer. RRGIBBS — A program for simple random regression analyses via gibbs sampling. In *Seventh World Congress on Genetics Applied to Livestock Production*, Montpellier, France, August 2002.

K. Meyer. RRGIBBS: A program for simple random regression analyses via gibbs sampling. Available at `http://didgeridoo.une.edu.au/km/rrgibbs.php`, June 2007.

S. Mosesova, H. Chipman, R. J. MacKay, and S. H. Steiner. Profile monitoring using mixed-effects models. Technical Report 06-06, University of Waterloo, 2006.

NIST. NIST/SEMATECH e-handbook of statistical methods. Available at `http://www.itl.nist.gov/div898/handbook/`, September 2010.

C. C. Paige and M. A. Saunders. LSQR: An algorithm fo sparse linear equations and sparse least squares. *TOMS*, 8(1):43–71, 1982.

R. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2010. URL `http://www.R-project.org`.

S. Rabe-Hesketh, A. Skrondal, and A. Pickles. GLLAMM: Stata programs for estimating, predicting, and simulating generalized linear latent and mixed models. Available at `http://www.gllamm.org/`, January 2008.

L. W. Sayrs. Pooled time series analysis. In *Sage University Paper Series on Quantitative Applications in the Social Sciences*, volume 07-070. Beverly Hills: Sage, 1989.

J. H. Shiau, H. Huang, S. Lin, and M. Tsai. Monitoring nonlinear profiles with random effects by nonparametric regression. *Communications in Statistics—Theory and Methods*, 38(10):1664–1679, 2009.

K. Wang and F. Tsung. Using profile monitoring techniques for a data-rich environment with huge sample size. *Quality and Reliability Engineering International*, 21:677–688, 2005.