Distributed Modelling of Big Dynamic Data with Generalized Linear Models

Kamil Dedecius* and Vladimíra Sečkárová*†
* Institute of Information Theory and Automation, Academy of Sciences of the Czech Republic
Pod Vodárenskou věží 4, 182 08 Prague, Czech Republic Email: {dedecius,seckarov}@utia.cas.cz
† Department of Probability and Mathematical Statistics, Faculty of Mathematics and Physics
Charles University in Prague, Sokolovská 83, 186 75 Prague, Czech Republic

***Abstract*—The big data, characterized by high volume, velocity and variety, often arise in a dynamic way, requiring fast online processing. This contribution proposes a new information-theoretic method for parallel dynamic statistical modelling of such data with a network of (potentially cooperating) processing units and an optional fusion center. The concept strongly exploits the principles of the Bayesian information processing, allowing its abstract formulation for arbitrary distributions. As a particular case, we specialize to the popular exponential family posterior distributions, arising either directly or indirectly from modelling with generalized linear models. Still, the applicability is considerably wider.**

I. Introduction

The big data are often defined via the 3Vs model as “high volume, high velocity and/or high variety information assets that require new forms of processing to enable enhanced decision making, insight discovery and process optimization.” [1] Nowadays data scale far beyond terabytes. Their velocity may necessarily need fast processing of massive amounts of streaming data of various relational and non-relational types, almost inevitably contamined by noise and outliers. Examples of systems producing large data are, e.g., the networked systems in general, big experimental physical systems like the Large Hadron Collider (LHC, 13 petabytes of data in 2010 [2]), e-commerce and industrial systems etc.

While the field of statistical analyses of big static data is already a well-established research discipline both with and without approaches rooted in the classical statistics, e.g. [3], the need for online processing of big dynamic data has received its initial impetus relatively very recently. Similarly to the static field, the arising methods for modelling, classification, prediction and other purposes are often based on Monte Carlo [4], variational Bayesian inference [5], conditional density filtering [6] and others. This contribution focuses on fast distributed statistical processing (modelling) of dynamic data with generalized linear models (GLMs) using both exact and approximate Bayesian inference. The important features of the proposed method, connected with the underlaying Bayesian framework are, among others, fixed low memory requirements, relatively simple and therefore fast recursions with, if needed, simple feasible approximation for GLMs and the ability to incorporate new information either in one-by-one manner or in blocks, similarly to the divide and recombine strategy [7]. The popular forgetting technique [8] is employed to guarantee adaptivity of the algorithm to potentially time-varying parameters with unknown evolution model. The method is inspired by distributed modelling with consensus and diffusion strategies (e.g. [9], [10]), in the Bayesian realm proposed by the authors [11].

Our setting, an example of which is depicted in Figure 1, is based on two or three types of network elements. First, the database, either centralized or distributed, storing data for modelling. The data items (vectors, database rows) are provided only once and only to one *processing node*. The processing nodes use the data for modelling and potentially exchange relevant information arising from this modelling among themselves. The nodes may also preprocess (e.g. decompress or parse) the raw data, access different instances/sockets of a distributed database to save communication resources etc. Finally, the network may contain the *fusion center*, where all the information from the processing nodes is merged. The fusion center is naturally a potential single point of failure. A certain degree of cooperation among processing nodes provides robustness by redundancy and, as the big data character usually guarantees fast convergence of the nodes' partial information due to the limit theorems, in certain (e.g. industrial) applications the fusion center can be avoided. The paper is methodologically oriented, hence it does not discuss any integration with software systems like Hadoop and others.

II. Principles of Bayesian Estimation

We consider modelling of an observed univariate or multivariate real random variable y based on the observed explanatory (usually real multivariate) variable x. The statistical approach to modelling builds upon a probabilistic model in the form of a probability distribution with a probability density function (pdf) $f(y|x, \Theta)$ where Θ is the latent parameter set

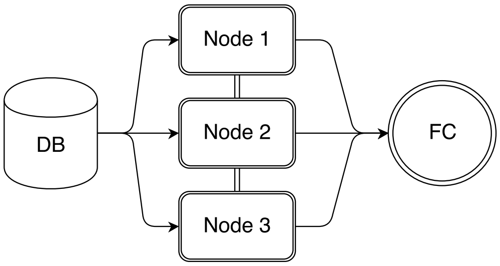

Fig. 1. Example of network topology. Three processing nodes query the database (DB) and provide their results to the fusion center (FC). Additionally, Node 2 cooperates with both other nodes. There is no cooperation between Node 1 and Node 3.

(shortly parameter) to be estimated. Unlike the frequentist statistical inference based, e.g., on maximum likelihood, the Bayesian inference performs with the aid of the prior distribution $\pi(\Theta)$, representing the a priori available knowledge about Θ. This pdf is updated by new evidence via the Bayes' rule

$$\pi(\Theta|x, y) = \frac{f(y|x,\Theta)\pi(\Theta)}{\int f(y|x,\Theta)\pi(\Theta)d\Theta}, \tag{1}$$

yielding the posterior pdf of Θ given the data x, y.

We focus on the popular class of GLMs, where the expected value of y depends on x and a parameter Θ through a known link function $g : \mathbb{R}^n \to \mathbb{R}$ [12],

$$\mathbb{E}[y|x,\Theta] = g^{-1}(\gamma) = g^{-1}(x^\intercal\theta), \qquad \theta \in \Theta,$$

where $\gamma = x^\intercal\theta$ is the linear predictor. The variance is typically a scalar function of the mean,

$$\mathrm{var}(y|x,\Theta) = V\left(g^{-1}(x^\intercal\theta)\right).$$

In the sequel, we cover the cases when the conjugate prior exists (linear regression with g^{-1} being identity mapping) and when it does not.

A. Dynamic inference under conjugate priors

If the prior distribution $\pi(\Theta)$ is conjugate to the model $f(y|x,\Theta)$, then the posterior $\pi(\Theta|x,y)$ belongs to the same family of distributions and can serve as the prior for the subsequent update [13]. This principle is particularly useful in dynamical problems, as newly obtained data $\{x_k, y_k\}_{k=1,2,\ldots}$ can be recursively incorporated into the pdf of Θ. The existence of conjugate priors is guaranteed for the exponential family distributions with pdfs of the form (k omitted)

$$f(y|x,\Theta) = \exp\left\{\eta^\intercal T(x,y) - \Psi(\eta) + \phi(x,y)\right\} \tag{2}$$

where $\eta = \eta(\Theta)$ is the natural parameter, $T(x,y)$ denotes the sufficient statistic, $\Psi(\eta) = \log\int\exp\left(\eta^\intercal T(x,y) + \phi(x,y)\right)dy$ is the log-partition (normalization, cumulant) function and $\phi(x,y)$ is a known function. The role of the dimension-preserving sufficient statistic is to encompass all information contained in data x, y necessary to estimate the parameter.

The conjugate prior pdf is also an exponential family distribution taking a similar form

$$\pi(\Theta) = \pi(\Theta|\xi,\nu) = \exp\left\{\eta^\intercal\xi - \nu\Psi(\eta) + l(\xi,\nu)\right\}$$

where the hyperparameters $\nu \in \mathbb{R}_+$ and ξ with $\dim(\xi) = \dim(T(x,y))$ serve as the sufficient statistics.

The Bayesian update (1) than reduces to the update of the hyperparameters,

$$\begin{aligned}\xi_k &= \xi_{k-1} + T(x_k, y_k)\\ \nu_k &= \nu_{k-1} + 1.\end{aligned}$$

The big data systems often generate a block of m data pairs $\{x_\kappa, y_\kappa\}_{\kappa=1,\ldots,m}$ within the processing period $(k-1, k]$. Then, the data is stored in the sufficient statistic T_k and incorporated into the prior using

$$\begin{aligned}\xi_k &= \xi_{k-1} + T_k(x,y)\\ \nu_k &= \nu_{k-1} + m.\end{aligned} \tag{3}$$

This corresponds to m consecutive Bayesian updates between $k-1$ and k.

B. Dynamic Analytical Approximate Inference

Often, the GLM does not possess a suitable conjugate prior $\pi(\Theta)$ due to its structure. Two prominent examples of this situation are the logistic regression model for binomial data $y \in \{0,1\}$ with the logit link function of the form

$$g(p_k) = \log\left(\frac{p_k}{1-p_k}\right) = x_k^\intercal\theta,$$

and the probit model where

$$g(p_k) = \Phi^{-1}(p_k).$$

In both cases

$$p_k = \mathbb{E}\left[y_k|x_k,\Theta\right].$$

$\Phi(p_k)$ is the standard normal cumulative distribution function.

The Bayesian inference of GLMs without conjugate priors inevitably calls for approximations. The Markov chain Monte Carlo (MCMC) methods simulating samples from the posterior dominate this field, however, they are expensive in terms of computational time and their use in dynamical problems is rather complicated [14]. Instead, one can advance with the maximum a posteriori (MAP) estimation with Laplace approximation [15] of the posterior $\pi(\Theta|x_{1:k}, y_{1:k})$. That is, fitting a normal pdf of the form

$$\pi'(\Theta|x_{1:k}, y_{1:k}) \sim \mathcal{N}(\mu_k, \Sigma_k) \tag{4}$$

centered at the mode μ_k of the posterior $\pi(\Theta|x_{1:k}, y_{1:k})$ (hence MAP) and with the covariance given by

$$\Sigma_k^{-1} = -\frac{\partial^2}{\partial\Theta^2}\log\pi(\Theta|x_{1:k}, y_{1:k}).$$

The most demanding task is finding the mode μ_k in dynamic estimation. In this respect, one should benefit from two aspects: (i) under time-invariance or slow variability of Θ, the (recursive) estimation stabilizes once the mode is found; (ii) a good prior is invaluable. Its good initial fitting (e.g., using MLE) immediately improves the ongoing estimation.

C. Time-Varying Parameters

Often, the dynamic environment is not parameter-invariant. If the explicit model of parameter variation is known, the class of the state-space models dominated by the Kalman filter is usually employed. Otherwise the methods known as forgetting (or time discounting) are exploited. The simplest – exponential forgetting [8] – flattens the prior pdf[1] using a forgetting factor $\lambda \in (0,1)$, usually very slightly lower than 1,

$$\pi(\Theta|\cdot) \leftarrow \pi(\Theta|\cdot)^\lambda. \tag{5}$$

In terms of hyperparameters of the conjugate prior pdf,

$$\xi_{k-1} \leftarrow \lambda\xi_{k-1} \qquad \text{and} \qquad \nu_{k-1} \leftarrow \lambda\nu_{k-1}.$$

If the Laplacian approximation is used, then the exponential forgetting flattens the prior covariance matrix, yielding the pdf $\mathcal{N}(\mu_{k-1}, \lambda^{-1}\Sigma_{k-1})$.

More elaborated method for independent tracking of elements of Θ and preventing covariance blow-up was proposed by Dedecius et al. [16], where the current state of the art is referred, too.

[1] Sometimes, the posterior pdf is flattened instead of the prior.

TABLE I. WEIGHTS BEFORE NORMALIZATION. $n^i = \mathrm{card}(\mathcal{N}^i)$, σ^2 IS THE SCALAR VARIANCE.

Method	Rule
Uniform	$\omega_j^i = 1/n^i$
Laplacian	$\omega_j^i = 1/n^{\max}$
Maximum degree	$\omega_j^i = 1/N$
Metropolis	$\omega_j^i = 1/\max(n^i, n^j)$
Relative degree	$\omega_j^i = n^j / \left(\sum_{k\in\mathcal{N}^i} n^k\right)$
Rel. degree-noise variance	$\omega_j^i = n^j \sigma_j^2 / \left(\sum_{k\in\mathcal{N}^i} n^k \sigma_k^2\right)$

III. Distributed Estimation of Big Dynamic Data

This section, building on the exposed theory, develops the distributed modelling framework. First, we present the fusion method in the whole generality and then specialize on fusion under conjugacy and fusion of approximate posterior pdfs.

Thorough, we consider a network of N processing nodes obtaining data from the database, and optionally a fusion center. $\mathcal{N}^i$ denotes the set of ith node's neighbors from which it can directly obtain information (the posterior pdfs π_j), including node i itself. $j \in \mathcal{N}^i$ does not necessarily imply $i \in \mathcal{N}^j$. The degree of belief of the ith node into information of a node $j \in \mathcal{N}^i$ is quantified by the weight (probability) $\omega_j^i \in [0,1]$; these weights sum to unity for all $j \in \mathcal{N}^i$. The fusion center, if present, does not pull data from the database. Instead, it accesses the processing nodes and combines their information. Similarly to the nodes, it quantifies the degree of belief into their information by the weights $\omega_j^{FC} \in [0,1]$, either summing to unity for all j or, under conjugacy, possibly equal to one to combine all information provided by the database to the network. Throughout, we consider static uniform weights for simplicity, other (also static) choices are given in Tab. I, see, e.g. [17] and the references therein. The extension to dynamic weights may become necessary if the communication lines or the nodes themselves corrupt the data by additional noise, failures or if the processing speed differs among nodes, e.g. m in (3) do not agree for all $i = 1, \ldots, N$.

A. Fusion Method

The fusion method is based on the idea of seeking the distribution $\tilde{\pi}_i$ of a node i, closest to all the posteriors π_j from the neighborhood $\mathcal{N}^i$ (hence including π_i), penalized by weights ω_j^i. The information theory advocates the use of information divergences as the proper measures of distributions dissimilarity. From the broad variety of divergences, for instance the Chernoff's and Rényi's α-divergences [18], [19], Csiszár's f-divergences [20] or the Bregman divergence [21], we choose the Kullback-Leibler (KL) divergence [22] for reasons that will become clear shortly.

The KL divergence is a particular case belonging to several important divergence families, including those mentioned. Given two pdfs f and g of a random variable z, their KL divergence is the nonnegative functional

$$\begin{aligned}\mathcal{D}(f||g) &= \mathbb{E}_{f(z)}\left[\log \frac{f(z)}{g(z)}\right] \\ &= \int f(z) \log \frac{f(z)}{g(z)} dz. \end{aligned} \tag{6}$$

Clearly $\mathcal{D}(f||g) = 0$ if and only if $f = g$ almost everywhere. Being a premetric, the KL divergence is not symmetric, $\mathcal{D}(f||g) \neq \mathcal{D}(g||f)$ nor does it satisfy the triangle inequality. From the Bayesian viewpoint it is essential that this divergence conforms to the conditional probabilities,

$$\begin{aligned}&\mathcal{D}\left(f(\Theta,z)||g(\Theta,z)\right) = \mathcal{D}\left(f(\Theta|z)f(z)||g(\Theta|z)g(z)\right) \\ &= \mathbb{E}_{f(\Theta,z)}\left[\log\frac{f(\Theta|z)}{g(\Theta|z)} + \log\frac{f(z)}{g(z)}\right] \\ &= \mathcal{D}\left(f(z)||g(z)\right) + \mathbb{E}_{f(z)}\left[\mathcal{D}(f(\Theta|z)||g(\Theta|z)\right].\end{aligned}$$

Assuming the models $f(\Theta|z)$ and $g(\Theta|z)$ identical, the divergence is driven by the information carried by the prior distribution. This is widely used by the principle of minimum discrimination information and in a sense coincides with the minimum cross-entropy.

Our fusion method is based on finding such a pdf $\tilde{\pi}_i(\Theta|\cdot)$ that satisfies the criterion

$$\min_{\tilde{\pi}_i} \sum_{j\in\mathcal{N}^i} \omega_j^i \mathcal{D}\left(\tilde{\pi}_i(\Theta|\cdot)\middle|\middle|\pi_j(\Theta|\cdot)\right) \tag{7}$$

where $\pi_j(\Theta|\cdot)$ is the posterior pdf of the jth node and $\tilde{\pi}_i(\Theta|\cdot)$ is the fused pdf optimal in the KL sense (7). Using the definition (6), we obtain

$$\begin{aligned}&\sum_{j\in\mathcal{N}^i} \omega_j^i \int \tilde{\pi}_i(\Theta|\cdot) \log \frac{\tilde{\pi}_i(\Theta|\cdot)}{\pi_j(\Theta|\cdot)} d\Theta \\ &= \int \tilde{\pi}_i(\Theta|\cdot) \log \frac{\tilde{\pi}_i(\Theta|\cdot)}{\prod_{j\in\mathcal{N}^i} \pi_j(\Theta|\cdot)^{\omega_j^i}} d\Theta \\ &= \mathcal{D}\left(\tilde{\pi}_i \middle|\middle| \prod_{j\in\mathcal{N}^i} \pi_j^{\omega_j^i}\right).\end{aligned}$$

Since the minimum of the KL divergence is attained when its arguments agree, the optimal pdf is the weighted geometric mean of neighbors' pdfs,

$$\tilde{\pi}_i(\Theta|\cdot) \propto \prod_{j\in\mathcal{N}^i} \left[\pi_j(\Theta|\cdot)\right]^{\omega_j^i}. \tag{8}$$

The same reasoning shows that using the alternative order of the KL divergence arguments in (7) yields the convex combination of neighbors' pdfs. The difference can be summarized as follows:

- The *weighted geometric mean* is zero-forcing in the sense that when the known second argument of the KL divergence tends to 0, the first must do as well, otherwise $\mathcal{D}(\cdot||\cdot) \to \infty$.
- The *convex combination* is zero-avoiding, that is, when the known first argument tends to zero, the second one does not need to.

Pragmatically, when the random variables obey a single model, the difference between both methods should vanish due to the equality of convex combination and geometric mean under identical arguments. With real data (observations), it vanishes in expected value. However, from the computational viewpoint, the geometric mean is notably preferred when the fused pdfs are from the exponential family, since the immediate result is again an exponential family distribution of the same type. The convex combination yields a finite mixture pdf, which would need further approximation by a single KL optimal pdf.

A particularly appealing feature of (8) is its resemblance to the Bayes' rule. Indeed, we can see the ith node's own posterior π_i as the prior information, updated by the "evidence" contained in π_j from the neighbors $j \in \mathcal{N}^i$, all terms being weighted.

B. Fusion under Conjugacy

The cooperating neighbors fuse their results – the exponential family distributions with the pdfs of the form (2) – within the neighborhood using (8). In terms of hyperparameters, this becomes simply two convex combinations,

$$\begin{aligned} \xi_{k,i} &= \sum_{j\in\mathcal{N}^i} \omega_j^i \xi_{k,j} \\ \nu_{k,i} &= \sum_{j\in\mathcal{N}^i} \omega_j^i \nu_{k,j}. \end{aligned} \tag{9}$$

In the fusion center, the situation is similar:

$$\begin{aligned} \xi_k^{FC} &= \sum_{j=1}^{N} \omega_j^{FC} \xi_{k,j} \\ \nu_k^{FC} &= \sum_{j=1}^{N} \omega_j^{FC} \nu_{k,j}. \end{aligned} \tag{10}$$

C. Fusion of Approximate Posterior pdfs

Fusion of the approximating univariate or multivariate normal posterior pdfs is easily done by observing that (8) leads here to a convex combination of natural parameters $\eta_{k,j}$ of nodes $j \in \mathcal{N}^i$. The exponential family form (2) of the posterior Laplacian normal pdf (4) reads

$$\begin{aligned} \pi'(\Theta|\mu_k, \Sigma_k) = \exp &\left(\begin{bmatrix} \Sigma_k^{-1}\mu_k \\ -\frac{1}{2}\Sigma_k^{-1} \end{bmatrix}^{\mathsf{T}} \begin{bmatrix} \Theta \\ \Theta\Theta^{\mathsf{T}} \end{bmatrix} \right) \\ \times \exp &\left(-\frac{1}{2}\mu_k^{\mathsf{T}}\Sigma_k^{-1}\mu_k + \frac{1}{2}\log|\Sigma_k| - \frac{r}{2}\log 2\pi \right) \end{aligned} \tag{11}$$

where $r = \dim(\Theta)$; $\mu_k \in \mathbb{R}^r$ and $\Sigma_k \in \mathbb{R}^{r\times r}$ are the mean vector and the positive definite covariance matrix, respectively. The first matrix in the upper exponential function is the natural parameter η_k, the latter is the sufficient statistic $T_k(\Theta)$. From the relationship among the moment and cumulant generating functions and the log-partition function of the exponential family distributions it follows (e.g. [23])

$$\mathbb{E}\left[T_k(\Theta)\right] = \frac{\partial\Psi(\eta_k)}{\partial\eta_k} \quad \text{and} \quad \operatorname{cov}\left(T_k(\Theta)\right) = \frac{\partial^2\Psi(\eta_k)}{\partial\eta_k^2},$$

the derivatives being in the multivariate sense. That is, taking derivatives of the log-partition function (the second row in (11)) with respect to the natural parameter and rearrangement of variables yield the moments of the random variable Θ, which under normality coincide with the parameters of the posterior normal pdf,

$$\begin{aligned} \tilde{\mu}_{k,i} &= -\frac{1}{2}\tilde{\eta}_{2,k}^{-1}\tilde{\eta}_{1,k} = \tilde{\Sigma}_{k,i}\left(\sum_{j\in\mathcal{N}^i} \omega_j^i \Sigma_{k,j}^{-1}\mu_{k,j}\right) \\ \tilde{\Sigma}_{k,i} &= -\frac{1}{2}\tilde{\eta}_{2,k}^{-1} = \left(\sum_{j\in\mathcal{N}^i} \omega_j^i \Sigma_{k,j}^{-1}\right)^{-1}. \end{aligned} \tag{12}$$

The right-hand sides follow from the product (8). The resulting merged normal pdf at the ith node is $\mathcal{N}(\tilde{\mu}_{k,i}, \tilde{\Sigma}_{k,i})$. The fusion center merging equations are also of type (12) with the weights ω_j^{FC}.

A thorough inspection of the system (12) reveals that it indeed coincides with the weighted Bayesian update of the multivariate normal pdf.

```
Initialization:
forall the Processing units $i = 1, \ldots, N$ do
    Set prior hyperparameters
    Set forgetting factor
    if Cooperation used then
        Determine cooperating neighborhood $\mathcal{N}^i$
        Determine neighbors' weights $\omega_j^i \quad \forall j \in \mathcal{N}^i$
    end
    if Fusion center exists then
        Set weight $\omega_i^{FC}$ in FC
    end
end
Online steps: for Processing steps $k = 1, 2, \ldots$ do
    forall the Processing nodes $i = 1, \ldots, N$ do
        Pull new data $(x, y)$ from database
        Perform Bayes' rule (1)
        if Non-conjugate priors then
            Approximate posterior pdf (Section II-B)
        end
        if Cooperation used then
            Pull posterior pdfs from neighbors
            Fuse posterior pdfs using (9) or (12)
        end
        Perform forgetting
    end
    if Fusion center exists then
        Pull posterior pdfs from nodes
        Fuse posterior pdfs using (10) or (12)
    end
end
```

Algorithm 1: Big Data Modelling Algorithm

IV. Experiments

Below we provide three experimental verifications of the proposed method. For illustration, the identically initialized nodes access the database in a sequential order instead of randomly. By κ we denote an ordinal index for data simulation while k denotes the time index of data processing. That is, with 3 processing nodes, the network assimilates 3 data items in transition from the prior pdf (index $k-1$) to the posterior one (k). With a single node, this would be the case $m = 3$ in equation (3).

The merging is done after all nodes have posterior pdfs; the next updating round follows afterwards. This allows sticking with constant equal weights and leads to the most representative (while not necessarily the best achievable) results. Theoretically, the speed of data processing should be N times higher (N is the number of nodes) plus the communication and fusion time, but as the purpose of these examples is demonstration of the method on simple models, we avoid time analyses. The ipython notebook demonstrations used to produce the results, including short theory exposition are freely available at http://diffest.utia.cas.cz.

We stress that in the examples we use noninformative conjugate priors at the beginning of the estimation process. This simulates the initial lack of any available knowledge about parameters, which is the case of many real-time (e.g. industrial) systems and where the big dynamic data can be an important aspect. Of course, if there is any prior information, it could (or should) be used instead. The choice of conjugacy is twofold: first, they are mathematically convenient; moreover, one even can apply the same principles to mixture estimation. Second, when the sample size grows, most priors will lead to the same inference [25].

A. Linear regression

This example demonstrates the linear regression with three non-cooperating processing nodes and a fusion center. The data sample consists of 5000 data generated by the model

$$y_\kappa = x_\kappa^\intercal \theta_\kappa + \varepsilon_\kappa = \theta_{0,\kappa} + \theta_1 x_\kappa + \varepsilon_\kappa, \qquad \varepsilon_\kappa \sim \mathcal{N}(0, \sigma^2),$$

where $\sigma^2 = 1$, the uniformly distributed regressors $x_\kappa \sim \mathcal{U}(0, 250)$ are ordered ascendantly and the regression coefficients

$$\theta_{0,\kappa} = 1 + 0.15 \sin\left(\frac{\pi}{2} \cdot \frac{\kappa}{5000}\right) \quad \text{and} \quad \theta_1 = 0.5 \ \forall \kappa.$$

The goal is to estimate $\Theta = \{\theta_k, \sigma^2\} = \{\theta_{0,k}, \theta_1, \sigma^2\}$.

Since the model is normal, we exploit the conjugate normal inverse-gamma prior distribution $\mathcal{N}i\mathcal{G}(\xi, \nu)$ with the symmetric positive definite information matrix ξ and the degrees of freedom ν as hyperparameters [8]. The Bayes' rule (1) updates the hyperparameters according to (3) with forgetting (5)

$$\begin{aligned} \xi_k &= \lambda \xi_{k-1} + \begin{bmatrix} y_k \\ x_k \end{bmatrix} \begin{bmatrix} y_k \\ x_k \end{bmatrix}^\intercal \\ \nu_k &= \lambda \nu_{k-1} + 1. \end{aligned}$$

The estimators easily follow from partitioning the information matrix ξ into blocks connected with y and x (indices k omitted)

$$\xi \equiv \begin{bmatrix} \xi_{yy} & \xi_{yx}^\intercal \\ \xi_{yx} & \xi_{xx} \end{bmatrix}, \qquad \xi_{yy} \in \mathbb{R}^1.$$

Then the point estimator of θ, its covariance matrix and the point estimator of the noise variance σ^2 read

$$\begin{aligned} \hat{\theta} &= \xi_{xx}^{-1} \xi_{yx}, \qquad \operatorname{cov}(\hat{\theta}) = \xi_{yy}^{-1} \\ \hat{\sigma}^2 &= \frac{\xi_{yy} - \xi_{yx}^\intercal \xi_{xx}^{-1} \xi_{yx}}{\nu + 2}. \end{aligned} \tag{13}$$

Merging at the fusion center has directly the form (10).

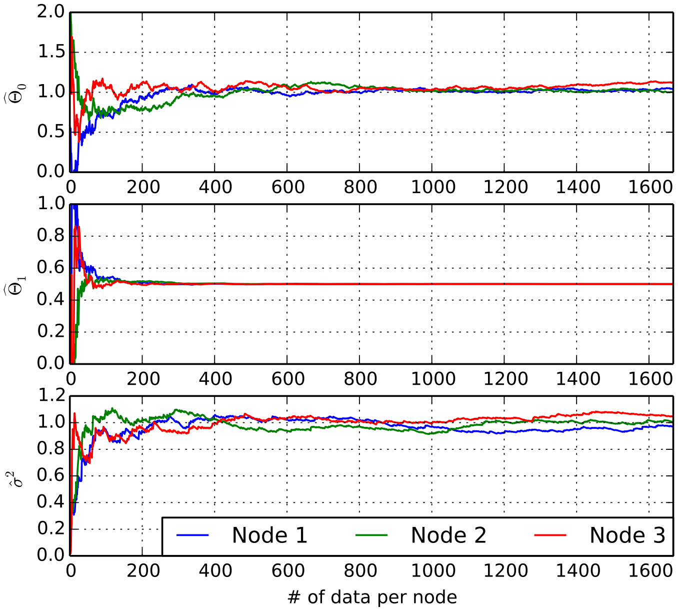

Fig. 2. Evolution of estimates of $\theta_{0,k}, \theta_1$ and σ^2 – noncooperating nodes.

TABLE II. MSE OF NODES' ESTIMATES. ASTERISK (*) INDICATES BETTER ESTIMATION IN PROCESSING NODE THAN IN THE FUSION CENTER.

	θ_0	θ_1	σ^2
Node 1	54.907	29.404	18.699
Node 2	34.173	38.001	12.913
Node 3	12.673*	14.927	11.333
FC	18.247	10.659	7.160

In our experiments, we set $\lambda = 0.998$ for tracking of the slowly varying parameter θ_0 and started from a flat prior with diagonal information matrix with $\operatorname{diag} \xi_0 = [0.1, 0.01]$ and $\nu_0 = 5$ degrees of freedom. Both uniform weights $\omega_j^{FC} = \frac{1}{3}$ and $\omega_j^{FC} = 1$ lead to the same point estimates, the latter with smaller variance of regression coefficient estimates. The progress of point estimates for all three processing nodes is depicted in Fig. 2. The evolution of point estimates, accompanied by 2 standard deviations band for $\hat{\theta}_0$ and $\hat{\theta}_1$ at the fusion center (for $\omega_j^{FC} = 1$) shows Fig. 3. Tab. II shows the mean squared errors (MSE) of estimates at the processing nodes and at the fusion center, which produces better results than all processing nodes with the only exception of $\hat{\theta}_0$ (indicated with an asterisk).

B. Logistic Regression – Simulated Data

The simulation example studies the effects of the proposed algorithm on a logistic regression with static parameters. We consider the model

$$y_\kappa \sim \text{Bernoulli}(p_\kappa) \quad \text{with} \quad p_\kappa = \frac{1}{1 + \exp(-\gamma_\kappa)}$$

where

$$\gamma_\kappa = x_\kappa^\intercal \theta = \theta_0 + \theta_1 x_\kappa, \qquad \kappa = 1, \ldots, 5000,$$

$\Theta = \theta = [\theta_0, \theta_1]^\intercal = [-1.6, 0.03]^\intercal$ and $x_\kappa \sim \mathcal{U}(18, 60)$ are uniformly distributed discrete integers. The goal is to estimate θ using a network of 5 processing nodes in a nearly total

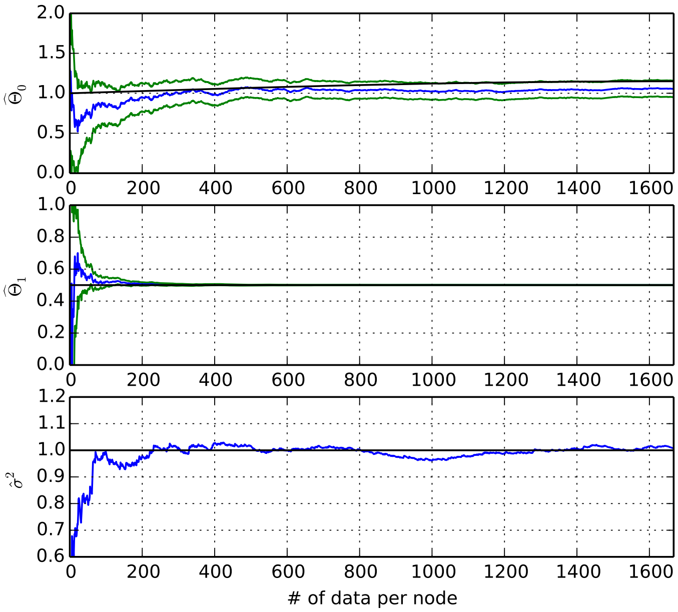

Fig. 3. Evolution of estimates of θ_0, θ_1 and σ^2 – fusion center, $\omega_j^{FC} = 1$. Black lines indicate true values.

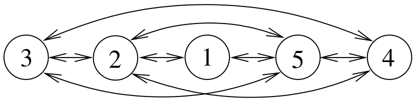

Fig. 4. Network layout (logistic regression)

graph, Fig. 4. The mode of the posterior distribution $\pi(\theta|\cdot)$ is approximated by a single Newton-Raphson optimization step,

$$\mu_k \leftarrow \mu_k - \frac{\partial_\theta \pi(\theta|x_{1:k}, y_{1:k})}{\partial_\theta^2 \pi(\theta|x_{1:k}, y_{1:k})}, \tag{14}$$

where ∂_θ and ∂_θ^2 denote the first and the second derivatives and the estimate $\hat{\theta}_{k-1}$ is taken as the initial value of μ_k. This approximation is crude in the initial phase under flat prior, still one can expect stable estimation after hitting the neighborhood of true time-invariant θ.

The weights are uniform, i.e., the nodes j within a neighborhood $\mathcal{N}^i$ have the same ω_i^j. The estimation starts with a relatively flat zero-centered normal prior with the diagonal covariance matrix with elements equal to 10. With respect to the problem setting and network topology, we expect quite stable estimation with similar properties in all five nodes. Therefore, the fusion center is not employed for its negligible contribution.

The estimation results in terms of the MSE of θ_0 and θ_1 for all five nodes are given in Table III. Compared to the no-cooperation scenario, only one node has worse MSE (indicated with an asterisk). The logarithm of the Brier score indicates slight improvement in prediction quality, which is also evident in the initial phase, see Figures 6 versus 7. The evolution of point estimates $\hat{\theta}_0$ and $\hat{\theta}_1$ with 2 standard deviations band is depicted in Fig. 5 for Node 1. In other nodes it is nearly identical. Clearly, the estimation quickly stabilises, after 200 incorporated data it remains almost invariant.

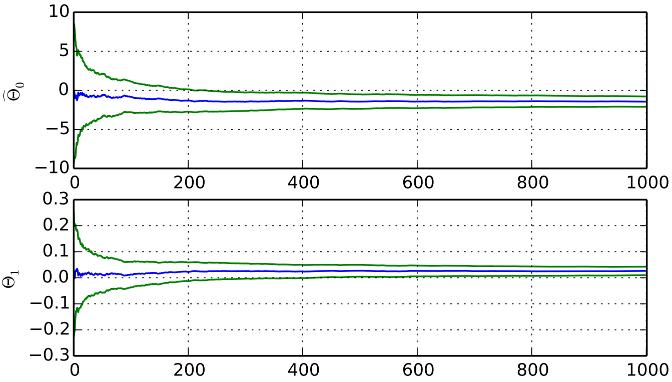

Fig. 5. Evolution of Node 1 estimates of θ_0 and θ_1 ± 2 standard deviations (with cooperation).

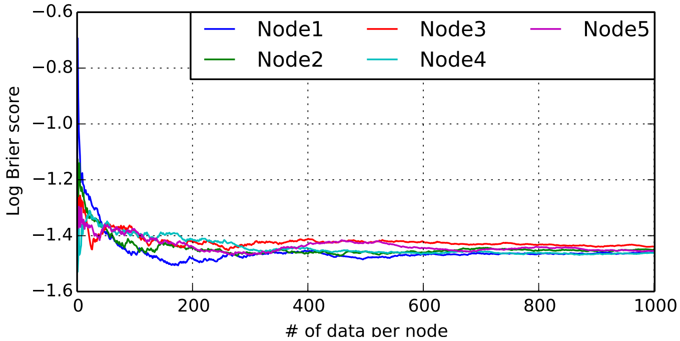

Fig. 6. Evolution of the log Brier score of all five nodes (estimation with cooperation).

C. Logistic Regression – Real Data

The second example with the same initial setting and the same approximation of the posterior mode considers astroparticle data from Jan Conrad of Uppsala University, Sweden [24]. This time, the regressor x_k is a vector of length 5 with the first element being 1 for the intercept. Figure 8 suggests that data standardisation might have a positive impact on modelling, however, we avoid it to simulate online processing. The data were randomly shuffled to suppress the effect of the original dataset ordering. The setting of the network is identical to the previous example. The resulting Brier score for all five nodes is given in Tab. V. Its development in time shows Fig. 10. Apparently, the estimation quality stabilizes with the number

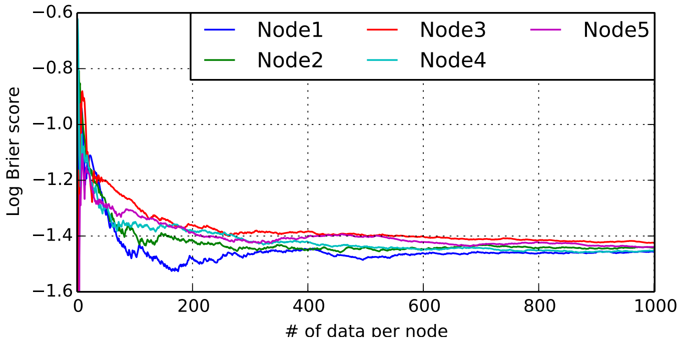

Fig. 7. Evolution of the log Brier score of all five nodes (estimation without cooperation).

TABLE III. MSE OF NODES' ESTIMATES UNDER COOPERATION AND WITHOUT IT. ASTERISK (*) INDICATES BETTER ESTIMATION WITHOUT COOPERATION.

	θ_0 **coop.**	θ_0 **no coop.**	θ_1 **coop.**	θ_1 **no coop.**
Node 1	1.320e-01	1.623e-01	6.027e-05	7.980e-05
Node 2	1.270e-01	8.346e-02*	5.917e-05	5.783e-05*
Node 3	1.297e-01	4.045e-01	5.883e-05	3.012e-04
Node 4	1.297e-01	3.984e-01	5.883e-05	2.882e-04
Node 5	1.302e-01	2.326e-01	5.927e-05	5.986e-05

TABLE IV. LOG BRIER SCORE OF NODES 1 – 5.

	Node 1	**Node 2**	**Node 3**	**Node 4**	**Node 5**
No coop.	0.233	0.237	0.241	0.234	0.237
Cooperation	0.232	0.235	0.237	0.232	0.234

of data incorporated. Evolution of estimates including the 2-standard deviation band is depicted for the first node in Fig. 9, the other nodes are very similar.

A very interesting empirical result is the estimation stabilisation. Naturally, the estimation of GLMs with nontrivial link functions is very sensitive to the character of modelled data. If the dataset does not carry enough information about the parameters, e.g. due to the low number of data in the initial phase of modelling or their low excitation, numerical difficulties are almost likely to occur in the frequentist approach. In the Bayesian paradigm, this can be resolved using informative priors. However, our example has shown that even with a flat prior one can obtain reasonable results and avoid numerical issues when the cooperation is used, because a potentially diverging node is corrected by its neighbors. Running the same example without cooperation mostly leads to numerical failure of at least one node in the first steps.

D. Discussion

The three experiments demonstrate the potential of the proposed information fusion method. The resulting estimation has mostly better quality, often significantly. In the case of GLMs with nontrivial link function (here, the logistic regression), the cooperation among processing nodes leads to numerical stabilization of the estimation. For illustration, the nodes deal with a simple data representation and perform simple approximations, where the parallelization need not lead to faster processing. In practice, the big dynamic data come in various forms, often needing a computationally demanding preprocessing. Also the MAP approximation of the mode in GLMs may be insufficiently precise, which would call for expensive numerical methods. Under such circumstances, the distributed processing may be invaluable.

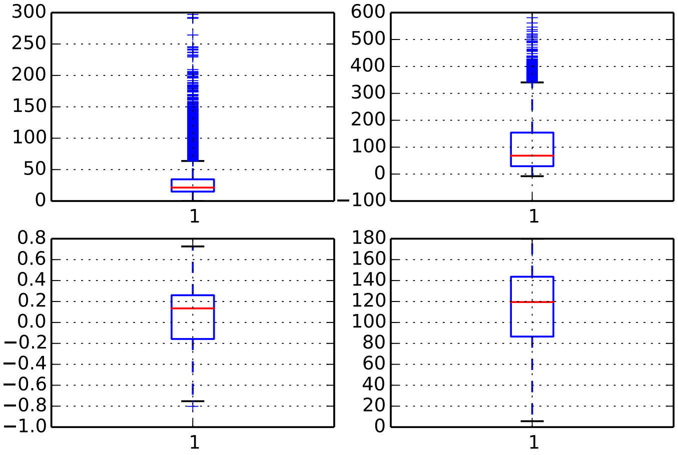

Fig. 8. Boxplots of regressors $x_{1,k}$ to $x_{4,k}$ (row-wise).

TABLE V. LOG BRIER SCORE OF NODES 1 – 5.

	Node 1	**Node 2**	**Node 3**	**Node 4**	**Node 5**
Cooperation	0.033	0.041	0.037	0.039	0.035

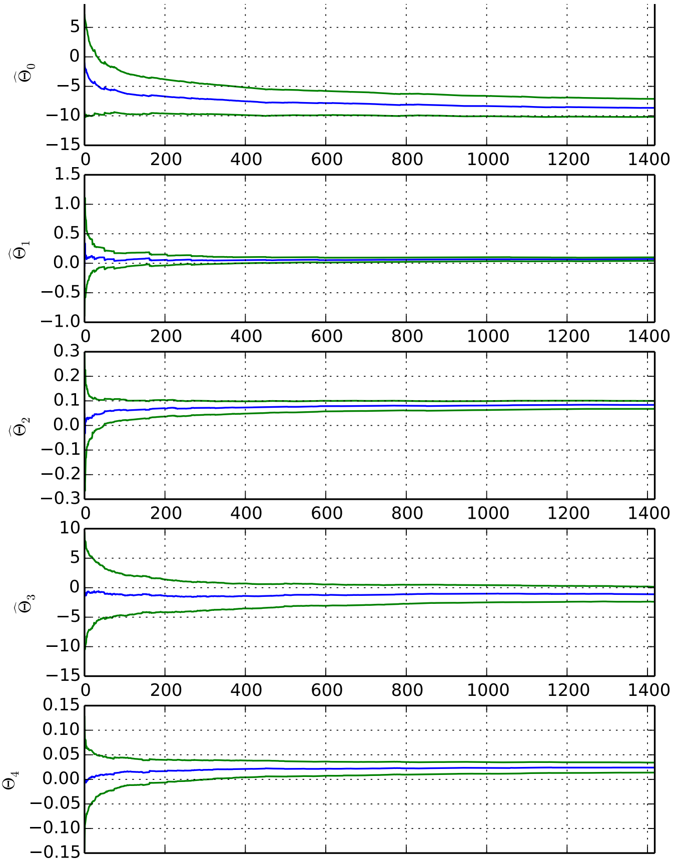

Fig. 9. Evolution of Node 1 estimates of $\theta_0, \ldots, \theta_4$ with ± 2 standard deviations (with cooperation).

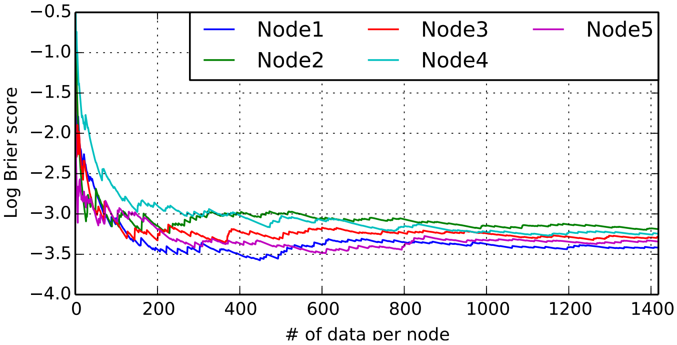

Fig. 10. Evolution of the log Brier score of all five nodes (estimation with cooperation).

V. Conclusion and future work

The paper proposes a new Bayesian method for distributed cooperative modelling of big dynamic data under the existence of both conjugate and non-conjugate prior distributions. The only imposed requirement allowing tractable information fusion is that the posterior pdf is an exponential family distribution. The method is very scalable and versatile. It allows various topologies of the network, presence of absence of the fusion center and, most importantly, cooperation among information processing nodes. The consistent Bayesian paradigm provides means for dealing with dynamic parameters, which is demonstrated by the basic exponential forgetting.

Besides the dynamic weights, the future work comprises extension to the mixture-based modelling, variational inference methods including variational message passing, and Monte Carlo-based estimation of complicated models. A perspective field seems to be also the computationally demanding inference of non-probabilistic models termed approximate Bayesian computation (ABC). With respect to the information-theoretical setting we conjecture these extensions to be straightforward.

Acknowledgement

The work of the first author is supported by the Czech Science Foundation, grant no. 14-06678P. The second author is supported by grant no. 13-13502S. All demonstrations including short theory exposition and source codes are freely available as ipython notebooks (including rendered version) at http://diffest.utia.cas.cz.

References

[1] M. A. Beyer and D. Laney, "The importance of big data: A definition," *Stamford, CT: Gartner*, 2012.

[2] G. Brumfiel, "Down the petabyte highway," *Nature*, vol. 469, no. 20, pp. 282–283, 2011.

[3] B. Ratner, *Statistical and machine-learning data mining: techniques for better predictive modeling and analysis of big data*. CRC Press, 2011.

[4] A. Kleiner, A. Talwalkar, P. Sarkar, and M. Jordan, "The big data bootstrap," *arXiv:1206.6415*, 2012.

[5] T. Broderick, N. Boyd, A. Wibisono, A. C. Wilson, and M. Jordan, "Streaming variational Bayes," in *Advances in Neural Information Processing Systems*, 2013, pp. 1727–1735.

[6] R. Guhaniyogi, S. Qamar, and D. B. Dunson, "Bayesian conditional density filtering for big data," *arXiv:1401.3632*, 2014.

[7] S. Guha, R. Hafen, J. Rounds, J. Xia, J. Li, B. Xi, and W. S. Cleveland, "Large complex data: divide and recombine (d&r) with rhipe," *Stat*, vol. 1, no. 1, pp. 53–67, 2012.

[8] V. Peterka, "Bayesian approach to system identification," In *Trends and Progress in System Identification*, P. Eykhoff, Ed. Oxford, U.K.: Pergamon, pp. 239–304, 1981.

[9] I. D. Schizas, G. Mateos, and G. B. Giannakis, "Distributed LMS for consensus-based in-network adaptive processing," *IEEE Trans. Signal Process.*, vol. 57, no. 6, pp. 2365–2382, Jun. 2009.

[10] S.-Y. Tu and A. H. Sayed, "Diffusion strategies outperform consensus strategies for distributed estimation over adaptive networks," *IEEE Trans. Signal Process.*, vol. 60, no. 12, pp. 6217–6234, 2012.

[11] K. Dedecius and V. Sečkárová, "Dynamic diffusion estimation in exponential family models," *IEEE Signal Process. Lett.*, vol. 20, no. 11, pp. 1114–1117, Nov. 2013.

[12] D. K. Dey, S. K. Ghosh, and B. K. Mallick, *Generalized linear models: A Bayesian perspective*. CRC Press, 2000.

[13] H. Raiffa and R. Schlaifer, *Applied Statistical Decision Theory)*. Harvard University Press, Jan. 1961.

[14] W. R. Gilks, S. Richardson, and D. J. Spiegelhalter, *Markov Chain Monte Carlo in Practice*. Chapman & Hall/CRC, Dec. 1995.

[15] A. Gelman, J. B. Carlin, H. S. Stern, and D. B. Rubin, *Bayesian Data Analysis, Second Edition*. Chapman & Hall/CRC, Jul. 2003.

[16] K. Dedecius, I. Nagy, and M. Kárný, "Parameter tracking with partial forgetting method," *International Journal of Adaptive Control and Signal Processing*, vol. 26, no. 1, pp. 1–12, 2012.

[17] F. S. Cattivelli and A. H. Sayed, "Diffusion LMS strategies for distributed estimation," *IEEE Trans. Signal Process.*, vol. 58, no. 3, pp. 1035–1048, Mar. 2010.

[18] H. Chernoff *et al.*, "A measure of asymptotic efficiency for tests of a hypothesis based on the sum of observations," *The Annals of Mathematical Statistics*, vol. 23, no. 4, pp. 493–507, 1952.

[19] A. Rényi, "On measures of entropy and information," In *Fourth Berkeley Symposium on Mathematical Statistics and Probability*, vol. 547, 1961, pp. 547–561.

[20] I. Csiszár, "Eine informationstheoretische Ungleichung und ihre Anwendung auf den Beweis der Ergodizität von Markoffschen Ketten," *Publ. Math. Inst. Hungar. Acad.*, vol. 8, pp. 85–108, 1963.

[21] L. M. Bregman, "The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming," *USSR Computational Mathematics and Mathematical Physics*, vol. 7, pp. 200–217, 1967.

[22] S. Kullback and R. A. Leibler, "On information and sufficiency," *The Annals of Mathematical Statistics*, vol. 22, no. 1, pp. 79–86, 1951.

[23] E. L. Lehmann and G. Casella, *Theory of Point Estimation (Springer Texts in Statistics)*. Springer, Aug. 1998.

[24] C.-W. Hsu, C.-C. Chang, C.-J. Lin *et al.*, "A practical guide to support vector classification," 2003.

[25] C.P. Robert, *The Bayesian Choice*. Springer, 2007.