Decentralized Approximate Bayesian Inference for Distributed Sensor Network

Decentralized Approximate Bayesian Inference for Distributed Sensor Network

Behnam Gholami, Sejong Yoon and **Vladimir Pavlovic**
Rutgers, The State University of New Jersey
110 Frelinghuysen Road
Piscataway, NJ 08854-8019
{bb510, sjyoon, vladimir}@cs.rutgers.edu

Abstract

Bayesian models provide a framework for probabilistic modelling of complex datasets. Many such models are computationally demanding, especially in the presence of large datasets. In sensor network applications, statistical (Bayesian) parameter estimation usually relies on decentralized algorithms, in which both data and computation are distributed across the nodes of the network. In this paper we propose a framework for decentralized Bayesian learning using Bregman Alternating Direction Method of Multipliers (B-ADMM). We demonstrate the utility of our framework, with Mean Field Variational Bayes (MFVB) as the primitive for distributed affine structure from motion (SfM).

Introduction

The traditional setting for many machine learning algorithms is the one where the model (e.g., a classifier or a regressor, typically parametric in some sense) is constructed from a body of data by processing this body in either batch or online fashion. The model itself is centralized and the algorithm has access to all model parameters and all data points. However, in many application scenarios today it is not reasonable to assume access to all data points because they could be distributed over a network of sensors or processing nodes. In those settings collecting and processing data in a centralized fashion is not always feasible because of several critical challenges.

First, in applications such as networks of cameras mounted on vehicles, the networks are constrained by severe capacity and energy constraints, considerably limiting the node communications (Radke 2010; Tron and Vidal 2011). Second, in many distributed sensor network applications such as health care, ecological monitoring, or smart homes, collecting all data at a single location may not be feasible because of its sheer volume as well as potential privacy concerns. Lastly, many sensor network tasks need to be performed in real time. Hence, processing data after collecting all information from different nodes prohibits performing the tasks in real time. The size of the centralized data would incur an insurmountable computational burden on the algorithm, preventing real-time or anytime (Zilberstein and Russell 1993) processing often desired in large sensing systems (Giannakis et al. 2015).

Distributed sensor networks provide an application setting in which distributed optimization tasks (including machine learning) that deal with some of the aforementioned challenges are frequently addressed (Radke 2010; Boyd et al. 2010). However, they are traditionally considered in a non-Bayesian (often deterministic) fashion. Moreover, the data is often assumed to be complete (not missing) across the network and in individual nodes. As a consequence, these approaches usually obtain parameter point estimates by minimizing a loss function based on the complete data and dividing the computation into subset-specific optimization problems.

A more challenging yet critical problem is to provide full posterior distributions for the parameters estimated in the aforementioned distributed settings. Such posteriors have the major advantage of characterizing the uncertainty in parameter learning and predictions, absent from traditional distributed optimization approaches. Another drawback of such approaches is that they traditionally rely on batch processing within individual nodes, unable to seamlessly deal with streaming data frequently present in sensing networks. However, both the sequential inference and the data completion would be naturally handled via the Bayesian analysis (Broderick et al. 2013), if one could obtain full posterior parameter estimates in this distributed setting. We also want our distributed Bayesian framemwork to work not only on discrete variables (Paskin and Guestrin 2004) but also in continuous cases.

In a recent work, Yoon and Pavlovic (2012) proposed a new method that estimates parametric probabilistic models with latent variables in a distributed network setting. The performance of this model was demonstrated to be on par with the centralized model, while it could efficiently deal with the distributed missing data. Nevertheless, the approach has several drawbacks.

First, its use of the Maximum Likelihood (ML) estimation increases the risk of overfitting, which is particularly pronounced in the distributed setting where each node works with a subset of the full data. Second, the approach cannot provide a measure of uncertainty around the estimated parameters that may be crucial in many applications, e.g., in online learning for streaming data or in assessing confidence

of predictions.

In this paper we propose a Distributed Mean Field Variational Inference (D-MFVI) algorithm for Bayesian Inference in a large class of graphical models. The goal of our framework is to learn a single consensus Bayesian model by doing local Bayesian inference and in-network information sharing without the need for centralized computation and/or centralized data gathering. In particular, we demonstrate D-MFVI on the Bayesian Principle Component Analysis (BPCA) problem and then apply this model to solve the distributed structure-from-motion task in a camera network.

Bregman Alternative Direction Method of Multipliers (B-ADMM)

ADMM has been successfully applied in a broad range of machine learning applications (Boyd et al. 2010). ADMM is canonically used for optimizing the following objective function subject to an equality constraint:

$$\arg\min_{x,z} f(x) + g(z), \quad s.t.\ Ax + Bz = c, \tag{1}$$

where $x \in \mathbb{R}^D, z \in \mathbb{R}^M$, and f and g are convex functions, A, B and c are some fixed terms. ADMM iteratively optimizes the augmented Lagrangian of (1), defined as:

$$L_p(x, z, y) = f(x) + g(z) + \langle y, Ax + Bz - c\rangle + \eta/2\|Ax + Bz - c\|_2^2, \tag{2}$$

where y is the dual variable, $\eta > 0$ is a penalty parameter, and the goal of quadratic penalty term is to penalize the violation of the equality constraint. The optimization is typically accomplished in a three-step update:

$$x_{t+1} = \arg\min_x f(x) + \langle y_t, Ax + Bz_t - c\rangle + \eta/2\|Ax + Bz_t - c\|_2^2, \tag{3}$$

$$z_{t+1} = \arg\min_z g(z) + \langle y_t, Ax_{t+1} + Bz - c\rangle + \eta/2\|Ax_{t+1} + Bz - c\|_2^2, \tag{4}$$

$$y_{t+1} = y_t + \eta(Ax_{t+1} + Bz_{t+1} - c), \tag{5}$$

Bregman ADMM (B-ADMM) replaces the quadratic penalty term in ADMM by a Bregman divergence (Wang and Banerjee 2014). This generalization of Euclidean metric will become essential when dealing with densities in the exponential family and the D-MFVI. More precisely, the quadratic penalty term in the x and z updates will be replaced by a Bregman divergence in B-ADMM:

$$x_{t+1} = \arg\min_x f(x) + \langle y_t, Ax + Bz_t - c\rangle + \eta B_\phi(c - Ax, Bz_t), \tag{6}$$

$$z_{t+1} = \arg\min_z g(z) + \langle y_t, Ax_{t+1} + Bz - c\rangle + \eta B_\phi(Bz, c - Ax_{t+1}), \tag{7}$$

$$y_{t+1} = y_t + \eta(Ax_{t+1} + Bz_{t+1} - c), \tag{8}$$

where $B_\phi : \Theta \times \Theta \to \mathbb{R}_+$ is the Bregman divergence with Bregman function ϕ (ϕ is a strictly convex function on a closed convex set Θ) that is defined as:

$$B_\phi(x, y) = \phi(x) - \phi(y) - \langle \nabla\phi(y), x - y\rangle, \tag{9}$$

where ∇ denotes the gradient operator.

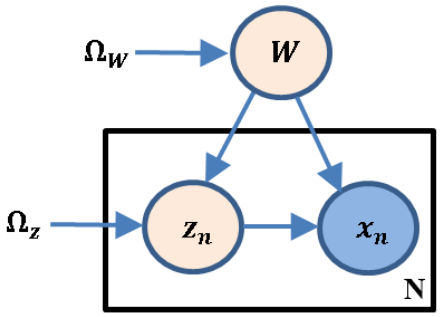


Figure 1: A graphical representation of the model of Eq. 10. Blue-shaded circle denotes observation.

Distributed Mean Field Variational Inference (D-MFVI)

We first explain a general parametric Bayesian model in a centralized setting. Then, we derive its distributed form.

Centralized Setting

Consider a data set X of observed D-dimensional vectors $X = \{x_i \in \mathbb{R}^D\}_{i=1}^N$ with the corresponding local latent variables $Z = \{z_i \in \mathbb{R}^M\}_{i=1}^N$, a global latent variable $W \in \mathbb{R}^p$ and a set of fixed parameters $\Omega = [\Omega_z, \Omega_w]$. The main assumption of our class of models is the factorization of the joint distribution of the observations, the global and the local variables into a global term and a product of local terms:

$$P(X, Z, W|\Omega) = P(W|\Omega_w)\prod_{i=1}^N P(x_n|z_n, W)P(z_n|\Omega_z). \tag{10}$$

The graphical representation of this class of models is shown in Fig 1. Given the observations, the goal is to compute (the approximation of) the posterior distribution of the latent variables, $P(W, Z|X, \Omega)$. For ease of computation, we use an exponential family assumption of the conditional distribution of a latent variable given the observation and the other latent variables:

$$P(W|X, Z, \Omega_w) = h(W)\exp\left\{\psi_w(X, Z, \Omega_w)^\top \mathcal{T}(W) - \mathcal{A}_w\big(\psi_w(X, Z, \Omega_w)\big)\right\}, \tag{11}$$

$$P(Z|X, W, \Omega_z) = \prod_{n=1}^N h(z_n)\exp\left\{\psi_{z_n}(x_n, W, \Omega_z)^\top \mathcal{T}(z_n) - \mathcal{A}_z\big(\psi_{z_n}(X, W, \Omega_z)\big)\right\}, \tag{12}$$

where $h(.)$ denotes the *base measure*, $\mathcal{A}(.)$ denotes the *log partition function*, and $\psi(.)$ and $\mathcal{T}(.)$ denote the *natural parameter* and the *sufficient statistics*, respectively. The assumed class of models contains many well known statistical models such as Bayesian PCA and Bayesian Mixture of PCA (Ghahramani and Beal 2000), Latent Dirichlet Allocation (Blei, Ng, and Jordan 2003), Bayesian Gaussian Mixture model (Attias 2000), Hidden Markov model (Fox et al. 2011; Paisley and Carin 2009), etc.

In many probabilistic models, due to the intractability of computing the exact posterior distribution of the latent variables given the observations, one is often required to employ approximate inference algorithms. In this work we use MFVI, which roots our strategy for distributed inference. The details of this approach are described below.

Mean Field Variational Inference (MFVI)

The goal of the Variational Inference (VI) is to approximate the true posterior distribution over the latent variables with a simpler distribution indexed by a set of free parameters that is the closest in KL divergence to the true posterior distribution (Jordan et al. 1999; Hoffman et al. 2013). MFVI is a subclass of VI that uses a family where all latent variables are independent of each other. More precisely, MFVI considers the following family of distributions as the approximate posterior distribution.

$$Q(Z, W) = \prod_{n=1}^{N} Q(z_n; \lambda_{z_n}) Q(W; \lambda_W), \tag{13}$$

where the form of $Q(z_n; \lambda_{z_n})$ and $Q(W; \lambda_W)$ are set to be in the same exponential family as the conditional distributions $P(W|X, Z, \Omega_w)$ (Eq. 11) and $P(Z|X, W, \Omega_z)$ (Eq. 12) and $\lambda_Z = \{\lambda_{z_n}\}_{n=1}^N$ and λ_W denote the variational parameters that are determined by maximizing the following variational objective function (that is equivalent to minimizing the $KL(Q(Z, W)||P(Z, W|X, \Omega_z, \Omega_w))$ (Davis et al. 2007).

$$\begin{aligned}
&\mathcal{L}(\lambda_Z, \lambda_W) = \mathbb{E}_Q\big[\log P(X, Z, W|\Omega_z, \Omega_w)\big] - \mathbb{E}_Q[\log Q] \\
&= \sum_{n=1}^{N} \mathbb{E}_{Q(z_n, W)}\big[\log P(x_n|z_n, W)\big] \\
&+ \sum_{n=1}^{N} \mathbb{E}_{Q(z_n)}\big[\log P(z_n|\Omega_z)\big] + \mathbb{E}_{Q(W)}\big[\log P(W|\Omega_w)\big] \\
&- \sum_{n=1}^{N} \mathbb{E}_{Q(z_n)}[\log Q(z_n)] - \mathbb{E}_{Q(W)}[\log Q(W)].
\end{aligned} \tag{14}$$

It should be noted that all terms of $\mathcal{L}(\lambda_Z, \lambda_W)$ are functions of the posterior parameters λ_Z, λ_W.

Distributed Setting

Consider $G = (V, E)$ as an undirected connected graph with vertices $i, j \in V$ and edges $e_{ij} = (i, j) \in E$ connecting the two vertices (Yoon and Pavlovic 2012). Each i-th node is directly connected with 1-hop neighbors in $\mathcal{B}_i = \{j|e_{ij} \in E\}$. Now, assume that each i-th node has its own set of data points $X_i = \{x_{in}|n = 1, ..., N_i\}$, local parameters $Z_i = \{z_{in}|n = 1, ..., N_i\}$ and global parameter W_i where $x_{in} \in \mathbb{R}^D$ is n-th data point and N_i is the number of samples collected in i-th node. Each i-th node infers the approximate posterior distribution over both global and local parameters locally, based on the available data in that node.

Computing the approximate posterior distribution of the local parameters ($\{Z_i\}_{i=1}^{|V|}$) is not an issue in the distributed setting due to the fact that the posterior distribution of each local latent variable z_n depends solely on the corresponding observation x_n and is independent of other observations (X_{-n}) conditioned on the global parameters. A naive approach for computing the posterior distribution of the global parameter (W) is to impose an additional *consensus* constraint on the global parameter in each node, $W_1 = W_2 = ... = W_{|V|}$. The details of this approach are described below.

However, in a Bayesian framework, the parameters are random variables and the notion of equality can be replaced with equivalency. This, however, leaves several options open (e.g., strict equality, equality in distribution, or almost sure equality). Here, we propose imposing equivalency in distribution, i.e., imposing equality constraints on the parameters of the posterior distribution of the global variable in each node, $\lambda_{W_1} = \lambda_{W_2} = ... = \lambda_{W_{|V|}}$.

Similar to prior work (Yoon and Pavlovic 2012), for decoupling purposes, we define a set of auxiliary variables ρ_{ij}, one for each edge e_{ij}. This now leads to the final distributed consensus MFVI formulation, which can be easily shown to be equivalent to the centralized MFVI optimization problem:

$$\begin{aligned}
[\widehat{\lambda}_Z, \widehat{\lambda}_W] = \underset{\lambda_{Z_i}, \lambda_{W_i}: i \in V}{\arg\min} \; &- \mathbb{E}_Q\big[\log P(X, Z, W|\Omega_z, \Omega_w)\big] \\
&+ \mathbb{E}_Q[\log Q(Z, W)], \\
s.t. \;\; \lambda_{W_i} = \rho_{ij}, \;\; &\rho_{ij} = \lambda_{W_j}, \;\; i \in V, j \in \mathcal{B}_i.
\end{aligned} \tag{15}$$

ADMM could be used to efficiently solve the above constrained optimization problem. More precisely, ADMM alternatively updates the variables in a block coordinate fashion by solving the augmented Lagrangian (using a linear and a quadratic penalty term) of Eq. 15.

Using conjugate exponential family for prior and likelihood distributions, each coordinate descent update in MFVI can be done in closed form. However, the penalty terms would be quadratic in the norm difference of $(\lambda_{W_i} - \rho_{ij})$, which may result in the non-analytic updates for $\{\lambda_{W_i}\}_{i=1}^{|V|}$. Note that updating $\{\lambda_{Z_i}\}_{i=1}^{|V|}$ can still be done in closed form as they do not appear in the equality constraints.

To solve Eq. 15 efficiently, we propose to use B-ADMM rather than standard ADMM. Since the global parameters are the parameters of the natural exponential family distributions, we propose to use the *log partition function* $\mathcal{A}_w(.)$ of the global parameter as the Bregman function.

It is worth noting that $\mathcal{A}_w(.)$ is not a stricly convex function in general, but, it is strictly convex if the exponential family is *minimal*[1]. We can always achieve this by reparametrization. Hence, using the minimal representation of the exponential families, it is easy to show that the coordinate descent steps Eq. 16 and Eq. 17 of B-ADMM for solving Eq. 15 have an analytic solution. Based on the proposed

[1] An exponential family is minimal if the functions $\psi(.)$ and the statistics $\mathcal{T}(.)$ each are linearly independent.

Bregman function, we obtain the updates for B-ADMM as

$$
\begin{aligned}
&[\lambda_Z^{(t+1)}, \lambda_W^{(t+1)}] = \\
&\underset{\lambda_{Z_i}, \lambda_{W_i}: i \in V}{\arg\min} \; - \sum_{i=1}^{|V|} \sum_{n=1}^{N_i} \mathbb{E}_{Q(z_{in}, W_i)} \big[\log P(x_{in} | z_{in}, W_i) \big] \\
&- \sum_{i=1}^{|V|} \sum_{n=1}^{N} \mathbb{E}_{Q(z_{in})} \big[\log P(z_{in} | \Omega_z) \big] \\
&- \frac{1}{|V|} \sum_{i=1}^{|V|} \mathbb{E}_{Q(W_i)} \big[\log P(W_i | \Omega_w) \big] \\
&+ \sum_{i=1}^{|V|} \sum_{n=1}^{N_i} \mathbb{E}_{Q(z_{in})} [\log Q(z_{in})] \\
&+ \frac{1}{|V|} \sum_{i=1}^{|V|} \mathbb{E}_{Q(W_i)} [\log Q(W_i)] \\
&+ \sum_{i \in V} \sum_{j \in \mathcal{B}_i} \left(\gamma_{ij1}^{(t)\top} (\lambda_{W_i} - \rho_{ij}^{(t)}) + \gamma_{ij2}^{(t)\top} (\rho_{ij}^{(t)} - \lambda_{W_j}) \right) \\
&+ \eta \sum_{i \in V} \sum_{j \in \mathcal{B}_i} B_{\mathcal{A}_w} (\lambda_{W_i}, \rho_{ij}^{(t)}) \qquad (16)
\end{aligned}
$$

$$
\begin{aligned}
\rho^{(t+1)} = \underset{\rho}{\arg\min} \; & \eta \sum_{i \in V} \sum_{j \in \mathcal{B}_i} B_{\mathcal{A}_w} (\rho_{ij}, \lambda_{W_i}^{(t+1)}) \\
&+ \sum_{i \in V} \sum_{j \in \mathcal{B}_i} \left(\gamma_{ij1}^{(t)\top} (\lambda_{W_i}^{(t+1)} - \rho_{ij}) + \gamma_{ij2}^{(t)\top} (\rho_{ij} - \lambda_{W_j}^{(t+1)}) \right), \qquad (17)
\end{aligned}
$$

$$
\gamma_{ijk}^{(t+1)} = \gamma_{ijk}^{(t)} + \eta (\lambda_{W_i}^{(t+1)} - \rho_{ij}^{(t+1)}), \qquad (18)
$$

where $i \in V, j \in \mathcal{B}_i$ and γ_{ijk} with $k = 1, 2$ are the Lagrange multipliers.

The scalar value η is the penalty parameter that should be determined in advanced (Boyd et al. 2010) or set separately to improve convergence properties of B-ADMM. $B_{\mathcal{A}_w}(.,.)$ denotes the Bregman divergence induced by $\mathcal{A}_w(.)$ and is defined as:

$$
B_{\mathcal{A}_w}(x, y) = \mathcal{A}_w(x) - \mathcal{A}_w(y) - \langle x - y, \nabla \mathcal{A}_w(y) \rangle, \quad (19)
$$

where $x^{(t)}$ denotes the value of the parameter x at iteration t. Since Bregman divergences are not necessarily convex in the second argument, we cannot use the same $B_{\mathcal{A}_w}(\lambda_{W_i}, \rho_{ij}^{(t)})$ for the Bregman penalization term in Eq. 17. Hence, Wang and Banerjee proposed to use the Bregman divergence with reverese of the parameters $(B_{\mathcal{A}_w}(\rho_{ij}, \lambda_{W_i}^{(t+1)}))$ and they proved the convergence of the new update equations.

The intuition behind the proposed Bregman function is as follows: based on the fact that the Bregman divergence (using log partition function as Bregman function) between two parameters λ, λ' of the same (minimal) exponential family $\mathcal{P}(x)$ is equivalent to the reverse KL divergence between the exponential families (Davis et al. 2007) and assuming that ρ_{ij} is the natural parameter of the same exponential family as $Q_{\lambda_{W_i}}(.)$, penalizing the deviation of the posterior parameter λ_{W_i} from the parameter ρ_{ij} using the Bregman divergence $B_{\mathcal{A}_w}(\lambda_{W_i}, \rho_{ij})$ is equivalent to penalizing the deviation of the approximate posterior distribution $Q_{\lambda_{W_i}}(W_i)$ from the distribution $Q_{\rho_{ij}}(.)$ in KL sense. We can write this formally as

$$
\begin{aligned}
B_{\mathcal{A}_w}(\lambda, \lambda') &= \mathcal{A}_w(\lambda) - \mathcal{A}_w(\lambda') - \langle \lambda - \lambda', \nabla \mathcal{A}_w(\lambda') \rangle \\
&= KL(\mathcal{P}_{\lambda'}(x), \mathcal{P}_{\lambda}(x)) \qquad (20)
\end{aligned}
$$

using the notations from previous section.

Case Study: Distributed Bayesian PCA (D-BPCA)

In what follows, we derive D-MFVI in the context of Bayesian PCA (Ghahramani and Beal 2000). Consider the latent factor model with $P(z_n) = \mathcal{N}(z_n; 0, I)$. In Probabilistic PCA (PPCA) model, the observed variable x_n is then defined as a linear transformation of z_n with additive Gaussian noise ϵ: $x_n = W z_n + \mu + \epsilon$, where $W \in \mathbb{R}^{D \times M}$, $\mu \in \mathbb{R}^D$ and ϵ is a zero-mean Gaussian-distributed vector with precision matrix $\tau^{-1} I$, where I denotes the identity matrix. The likelihood distribution is:

$$
P(x_n | z_n, W, \mu, \tau) = \mathcal{N}(x_n; W z_n + \mu, \tau^{-1} I), \qquad (21)
$$

where $n = 1, ..., N$. Based on the above probabilistic formulation of PCA, we can obtain a Bayesian treatment of PCA by first introducing a prior distribution $P(\mu, W, \tau)$ over the parameters of the model. Second, we compute the corresponding posterior distribution $P(\mu, W, \tau | X)$ by multiplying the prior by the likelihood function given by Eq. 21, and normalizing.

There are two issues that must be addressed in this framework: (i) The choice of the prior distribution, and (ii) The formulation of a tractable procedure for computing the posterior distribution. Typically, in BPCA, the prior distributions over parameters are defined such that they are independent of each other apriori $P(\mu, W, \tau) = P(\mu) P(W) p(\tau)$. For simplicity, in this paper we assume that the data noise precision τ is a fixed but unknown parameter. We define an independent Gaussian prior over each row of W as

$$
P(W | \alpha) = \prod_{d=1}^{D} (\frac{\alpha_d}{2\pi})^{M/2} \exp\{-\frac{1}{2} \alpha_d (w_d - \bar{w}_d)^\top (w_d - \bar{w}_d)\},
$$

where w_d is the d-th row of W, $\{\bar{w}_d\}_{d=1}^D$ and $\{\alpha_d\}_{d=1}^D$ are the mean and the precision hyperparameters respectively. Furthermore, we consider another Gaussian distribution as the prior for μ, $P(\mu) = \mathcal{N}(\bar{\mu}, \theta^{-1} I)$, where $\bar{\mu}$ and θ are the mean and the precision hyperparameters respectively. Due to the intractability of computing the exact posterior distribution $\in \mathbb{R}^{D \times M}$, we use MFVI. In order to apply MFVI to Bayesian PCA we assume a fully factorized posterior Q of the form

$$
Q(W, Z, \mu) = \prod_{d=1}^{D} \prod_{m=1}^{M} Q(w_{dm}) \prod_{n=1}^{N} Q(z_n) \prod_{d=1}^{D} Q(\mu_d).
$$

Due to the use of conjugate priors for W, Z and μ, the posterior distributions are Gaussian ($Q(w_{dm}) \sim \mathcal{N}(m_{dm}^w, \lambda_{dm}^w)$,

$Q(\mu_d) \sim \mathcal{N}(m_d^\mu, \lambda_{dm}^\mu)$, and $Q(z_n) \sim \mathcal{N}(m_n^z, \Lambda_n^{-1}))$ and their update is equivalent to re-estimation of the corresponding means and variances.

Distributed formulation

The distributed MFVI algorithm can be directly applied to this BPCA model. Specifically, W and μ are global latent variables, and $\{z_n\}_{n=1}^N$ are the local latent variables. The basic idea is to assign each subset of samples to each node in the network, and do inference locally in each node. By considering $\Xi_i = \{(m_{dm}^w)_i, (\lambda_{dm}^w)_i, (m_d^\mu)_i, (\lambda_{dm}^\mu)_i, (m_n^z)_i, (\Lambda_n)_i\}$ as the set of parameters for node i, the D-MFVI optimization now becomes

$$\widehat{\Xi} = \underset{\Xi_i : i \in V}{\arg\min}\ \mathbb{E}_{Q(W_i,\mu_i,Z_i)}[\log Q(W_i, \mu_i, Z_i)] - \sum_{i=1}^{|V|} \mathbb{E}_{Q(W_i,\mu_i,Z_i)}\big[\log P(X_i, Z_i, W_i | \tau, \alpha, \theta, \bar{\mu}, \bar{w}_1, ..., \bar{w}_d)\big]$$

$$s.t. \quad (m_d^\mu)_i = (\rho_d^\mu)_{ij},\ \ (\rho_d^\mu)_{ij} = (m_d^\mu)_j,$$
$$(m_{dm}^w)_i = (\rho_{dm}^w)_{ij},\ \ (\rho_{dm}^w)_{ij} = (m_{dm}^w)_j,$$
$$(\lambda_d^\mu)_i = (\phi_d^\mu)_{ij},\ \ (\phi_d^\mu)_{ij} = (\lambda_d^\mu)_j,$$
$$(\lambda_{dm}^w)_i = (\phi_{dm}^w)_{ij},\ \ (\phi_{dm}^w)_{ij} = (\lambda_{dm}^w)_j,$$

where $i \in V, j \in \mathcal{B}_i$ and $\{(\rho_d^\mu)_{ij}, (\phi_d^\mu)_{ij}, (\rho_{dm}^w)_{ij}, (\phi_{dm}^w)_{ij}\}$ are auxiliary variables. Due to the lack of space, we explain how to specify hyperparameters $\tau, \alpha, \theta, \bar{\mu}, \bar{w}_1, ..., \bar{w}_d$ and present the coordinate descent update rules for solving the above problem in the extended version of this paper (Gholami, Yoon, and Pavlovic 2015). Generalizing our distributed BPCA (D-BPCA) to deal with missing data is straightforward and follows Ilin and Raiko (2010).

Experimental Results

In this section, we first demonstrate the general convergence properties of the D-BPCA algorithm on synthetic data. We then apply our model to a set of Structure from Motion (SfM) problems. We compared our distributed algorithm with traditional SVD, Centralized PCA (PPCA) (Tipping and Bishop 1999), Distributed PPCA (D-PPCA) (Yoon and Pavlovic 2012), and Centralized BPCA (BPCA) (Ilin and Raiko 2010).

Empirical Convergence Analysis

We generated synthetic data using a generative PPCA model in Fig. 1 to show the convergence of D-BPCA in various settings. Based on the results, D-BPCA is robust to topology of the network, the number of nodes in the network, choice of the parameter η, and both data missing-at-random (MAR) and missing-not-at-random (MNAR) (Ilin and Raiko 2010) cases. A detailed analysis of this evaluation can be found in the extended version (Gholami, Yoon, and Pavlovic 2015).

D-BPCA for Structure from Motion (SfM)

In affine SfM, based on a set of 2D points observed from multiple cameras (or views), the goal is to estimate the corresponding 3D location of those points, hence the 3D structure of the observed object as well as its motion (or, equivalently, the motion of the cameras used to view the object). A canonical way to solve this problem is the factorization approach (Tomasi and Kanade 1992).

More precisely, by collecting all the 2D points into a measurement matrix X of size $\#points \times (2 \times \#frames)$, we can factorize it into a $\#points \times 3$ 3D structure matrix W and a $3 \times (2 \times \#frames)$ motion matrix Z. SVD can be used to find both W and Z in a centralized setting. PPCA and D-PPCA can also estimate W and Z using the EM algorithm (Yoon and Pavlovic 2012).

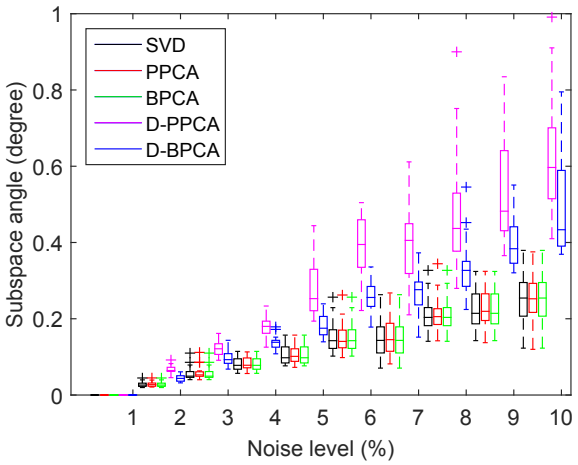
Equivalently, the estimates of W and Z can also be found using our D-BPCA where W is treated as the latent global structure and Z is the latent local camera motion. It is worth noting that D-PPCA can only provide the uncertainty around the motion matrix Z, while D-BPCA obtains additional estimates of the variance of the 3D structure W. We now show that our D-BPCA can be used as an effective framework for distributed affine SfM. For all SfM experiments, the network has the ring topology, with $\eta = 10$.

We equally partitioned the frames into 5 nodes to simulate 5 cameras, the convergence was set to 10^{-3} relative change in objective of (15). We computed maximum subspace angle between the ground truth 3D coordinates and the estimated 3D structure matrix as the measure of performance (for BPCA and D-BPCA, we used the posterior mean of 3D structure matrix for the subspace angle calculation).

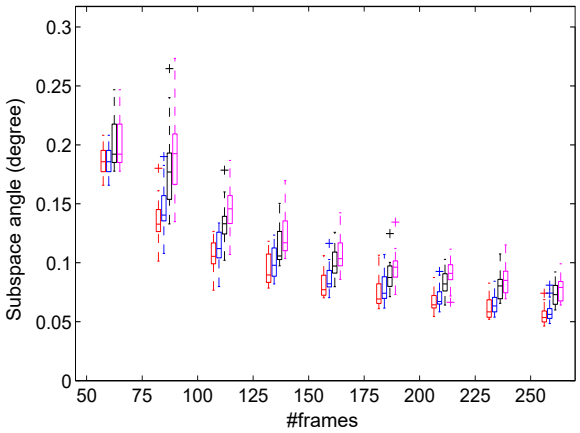
Synthetic Data (Cube) Similar to the synthetic experiments in Yoon and Pavlovic (2012), we used a rotating unit cube and 5 cameras facing the cube in a 3D space to generate synthetic data. In contrast to the setting in Yoon and Pavlovic (2012), we rotated the cube every 3° over 150° clockwise to obtain additional views necessary for our online learning evaluation, i.e. in this setting, each camera observed 50 frames. Fig. 2a shows the performance of different models in the case of noisy data (over 20 independent runs with 10 different noise levels). As can be seen from the figure, in the case of noisy data, D-BPCA consistently outperforms D-PPCA, thanks in part to improved robustness to overfitting.

For the MAR experiment, we randomly discarded 20% of data points over ten independent runs. The average errors were 1.41° and 1.01° for D-PPCA and D-BPCA, respectively. The same experiment was done for the more challenging MNAR with the missing data generated by a realistic visual occlusion process (hence, non-random). This yielded errors of 17.66° for D-PPCA and 14.12° for D-BPCA respectively. Again, D-BPCA resulted in consistently lower errors than D-PPCA, although the error rates were higher in the more difficult MNAR setting.

One particular advantage of the D-BPCA over D-PPCA and the SVD counterparts is its ability to naturally support online Bayesian estimation in the distributed sensing network. We first used 10 frames in each camera as the first minibatch of data. Then, we repeatedly added 5 more frames to each camera in subsequent steps. Results over 10 different runs with 1% noise in the data are given in Fig. 2b. Note



(a) Noisy data experiment



(b) Online data experiment. Legend: Red = BPCA, Blue = Online BPCA, Black = D-BPCA, Magenta = Online D-BPCA

Figure 2: Results for the cube synthetic data (crosses denote outliers).

Table 1: Results of Caltech dataset. All results ran 20 independent initializations.

Object	BallSander	BoxStuff	Rooster	Standing	StorageBin
# Points	62	67	189	310	102
# Frames	30	30	30	30	30
Subspace angle between centralized SVD SfM and D-PPCA (degree)					
Mean	1.4934	1.4402	1.4698	2.6305	0.4513
Variance	0.4171	0.4499	0.9511	1.7241	1.2101
Subspace angle between centralized SVD SfM and D-BPCA (degree)					
Mean	**0.9910**	**0.9879**	**1.3855**	**0.9621**	**0.4203**
Variance	0.0046	0.0986	0.0080	0.0033	0.0044

Table 2: Missing data Results of Caltech dataset. All results used 20 independent initializations. Results provide variances over various missing value settings. Scores are subspace angles between fully observable centralized SVD versus D-PPCA / D-BPCA.

		MAR	MNAR
D-PPCA	Mean	4.0609	9.4920
	Variance	1.2976	5.9624
D-BPCA	Mean	**2.2012**	**7.2187**
	Variance	1.3179	5.2853

Table 3: Subspace angles (degree) between fully observable centralized SVD and D-PPCA / D-BPCA for Hopkins dataset. All results ran 5 independent initializations.

		No-missing	MAR
D-PPCA	Mean	3.9523	13.4753
	Variance	3.3119	12.9832
D-BPCA	Mean	**0.7975**	**6.4372**
	Variance	0.5684	5.0689

that due to the non-Bayesian nature of the D-PPCA model, it cannot easily be applied in the online setting. Fig. 2b demonstrates that the subspace angle error of the online D-BPCA closely follows centralized BPCA in accuracy.

Real Data. We applied our model to the Caltech 3D Objects on Turntable dataset (Moreels and Perona 2007) and Hopkins155 dataset (Tron and Vidal 2007) to demonstrate its usefulness for real data. Following Yoon and Pavlovic (2012), we used a subset of the dataset which contains images of 5 objects for Caltech dataset, and 90 single-object sequences for Hopkins155 dataset. For both datasets, we used the same setup as Yoon and Pavlovic (2012). The subspace angles between the structure inferred using the traditional centralized SVD and the D-PPCA and D-BPCA for Caltech dataset are available in Table 1. We ran 20 independent initializations to obtain the mean and variance. 10% MAR and

MNAR results are also provided in the Table 2. We report the average over all 5 objects for this part. As can be seen, the D-BPCA performance is better than D-PPCA.

Average maximum subspace angle between D-PPCA, D-BPCA and SVD for all selected 90 objects without missing data and with 10% MAR are shown in Table 3 (we did not perform MNAR experiments on Hopkins due to the fact that the ground truth occlusion information is not provided with the dataset). For this dataset, D-BPCA consistently has better performance than D-PPCA. It should be noted that although the subspace angle error is very large for MAR case for both D-PPCA and D-BPCA, 3D structure estimates were similar to that of SVD up to the orthogonal ambiguity.

Conclusion

In this paper we introduced a general approximate inference approach using Mean Field Variational Inference for learning parameters of traditional centralized probabilistic models in a distributed setting. The main idea is to split the data into different nodes, impose consensus constraints on the posterior parameters of each node, and solve the constrained variational inference using Bregman ADMM. We illustrated this approach with BPCA for SfM application. Experimental results showed that Bayesian approaches show substantial improvements over the traditional ML approach with an additional benefit of natural online learning.

References

Attias, H. 2000. A variational Bayesian framework for graphical models. In *Advances in Neural Information Processing Systems (NIPS)*.

Blei, D.; Ng, A.; and Jordan, M. 2003. Latent Dirichlet allocation. *Journal of Machine Learning Research* 3:993–1022.

Boyd, S.; Parikh, N.; Chu, E.; Peleato, B.; and Eckstein, J. 2010. Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers. *Foundations and Trends in Machine Learning* 3(1):1–122.

Broderick, T.; Boyd, N.; Wibisono, A.; Wilson, A.; and Jordan, M. 2013. Streaming Variational Bayes. In *Advances in Neural Information Processing Systems (NIPS)*.

Davis, J. V.; Kulis, B.; Jain, P.; Sra, S.; and Dhillon, I. S. 2007. Information-theoretic metric learning. In *International Conference on Machine Learning (ICML)*, 209–216.

Fox, E.; Sudderth, E.; Jordan, M.; and Willsky. 2011. A sticky HDP-HMM with application to speaker diarization. *Annals of Applied Statistics* 5:1020–1056.

Ghahramani, Z., and Beal, M. 2000. Variational inference for Bayesian mixtures of factor analysers. In *Advances in Neural Information Processing Systems (NIPS)*.

Gholami, B.; Yoon, S.; and Pavlovic, V. 2015. D-MFVI: distributed mean field variational inference using bregman ADMM. *CoRR* http://arxiv.org/abs/1507.00824.

Giannakis, G. B.; Ling, Q.; Mateos, G.; Schizas, I. D.; and Zhu, H. 2015. Decentralized learning for wireless communications and networking. *CoRR* http://arxiv.org/abs/1503.08855.

Hoffman, M.; Blei, D.; Wang, C.; and Paisly, J. 2013. Stochastic Variational Inference. *Journal of Machine Learning Research* 14:1303–1347.

Ilin, A., and Raiko, T. 2010. Practical approaches to principal component analysis in the presence of missing values. *Journal of Machine Learning Research* 11:1957–2000.

Jordan, M.; Ghahramani, Z.; Jaakkola, T.; and Saul, L. 1999. An introduction to variational methods for graphical models. *Machine Learning* 37(2):183–233.

Moreels, P., and Perona, P. 2007. Evaluation of Features Detectors and Descriptors based on 3D Objects. *International Journal of Computer Vision* 73(3):263–284.

Paisley, J., and Carin, L. 2009. Nonparametric factor analysis with beta process priors. In *International Conference on Machine Learning (ICML)*.

Paskin, M. A., and Guestrin, C. E. 2004. Robust Probabilistic Inference in Distributed Systems. In *Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence (UAI)*, UAI '04, 436–445. Arlington, Virginia, United States: AUAI Press.

Radke, R. J. 2010. A Survey of Distributed Computer Vision Algorithms. In Nakashima, H.; Aghajan, H.; and Augusto, J. C., eds., *Handbook of Ambient Intelligence and Smart Environments*. Springer Science+Business Media, LLC.

Tipping, M. E., and Bishop, C. M. 1999. Probabilistic Principal Component Analysis. *Journal of the Royal Statistical Society, Series B* 61:611–622.

Tomasi, C., and Kanade, T. 1992. Shape and motion from image streams under orthography: a factorization method. *International Journal of Computer Vision* 9:137–154. 10.1007/BF00129684.

Tron, R., and Vidal, R. 2007. A Benchmark for the Comparison of 3-D Motion Segmentation Algorithms. In *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 1–8.

Tron, R., and Vidal, R. 2011. Distributed Computer Vision Algorithms Through Distributed Averaging. In *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 57–63.

Wang, H., and Banerjee, A. 2014. Bregman Alternating Direction Method of Multipliers. In *Advances in Neural Information Processing Systems (NIPS)*.

Yoon, S., and Pavlovic, V. 2012. Distributed probabilistic learning for camera networks with missing data. In *Advances in Neural Information Processing Systems (NIPS)*.

Zilberstein, S., and Russell, S. 1993. Anytime sensing, planning and action: A practical model for robot control. In *International Joint Conference on Artificial Intelligence (IJCAI)*, 1402–1407.