Learning Latent Variable Models from Non-Stationary Data Streams

Andrés R. Masegosa ANDRESMASEGOSA@UAL.ES
Darío Ramos-López DRAMOSLOPEZ@UAL.ES
Antonio Salmerón ANTONIO.SALMERON@UAL.ES
Department of Mathematics
Unversity of Almería
Almería, Spain

Helge Langseth HELGE.LANGSETH@NTNU.NO
Department of Computer Science
Norwegian University of Science and Technology
Trondheim, Norway

Thomas D. Nielsen TDN@CS.AAU.DK
Department of Computer Science
Aalborg University
Aalborg, Denmark

Editor:

Abstract

In many modern data analysis problems, the available data is not static, but instead comes in a streaming fashion. Making inferences based on a data stream is challenging for several reasons. First of all, it requires continuous model updating and the ability to handle a posterior distribution conditioned on an unbounded data set. Secondly, the underlying data distribution may *drift* from one time step to another, and the classic i.i.d. (or data exchangeability) assumption does not hold any more. In this paper, we present a Bayesian approach which addresses these issues for general latent variable models within the conjugate exponential family. Our proposal makes use of a novel scheme based on hierarchical (non-conjugate) priors to explicitly model temporal changes of the model parameters, which induces an exponential forgetting mechanism with adaptive forgetting rates. A variational inference scheme is derived which maintains the computational efficiency of variational methods over conjugate models. The approach is validated on four different domains (energy, finance, geolocation, and text) using four real-world data sets.

Keywords: Latent Variable Models, Non-Stationary Data Streams, Concept Drift, Variational Inference, Power Priors, Exponential Forgetting

1. Introduction

Latent variable models (LVMs) (Bishop, 1998; Blei, 2014) are probabilistic models built to uncover hidden patterns in a data set. Usually, we are interested in both local patterns, which are specific for each sample of the data, and global patterns, that are shared among all the samples. These hidden patterns are modeled by means of a set of local and global random latent (unobserved) variables,

respectively, and the observed data is assumed to be generated from distributions conditioned on these latent variables. Figure 1 (a) illustrates this kind of models.

LVMs include popular models like LDA (Blei et al., 2003) models to uncover the hidden topics in a text corpora, mixture of Gaussian models to discover hidden clusters in data (Bishop, 2006), probabilistic principal component analysis for revealing a low-dimensional representation of the data (Tipping and Bishop, 1999), models with hierarchical latent variables to capture drift in data streams (Borchani et al., 2015; Masegosa et al., 2017a), and so on. Comprehensive descriptions of these models can be found in, e.g., (Bishop, 2006; Koller and Friedman, 2009; Murphy, 2012).

In recent years, the development of learning methods for LVMs that scale to massive data sets has received a lot of attention (Hoffman et al., 2013; Masegosa et al., 2017b; Hasenclever et al., 2017; Minsker et al., 2017). But in many modern machine learning applications, the presence of massive data sets is not the only issue. Many data sets are only available in a streaming fashion, where new data samples are continuously arriving. LVMs should be updated accordingly to capture the distribution and hidden patterns in the current data. However, in many domains the data stream are non-stationary and may exhibit both gradual and abrupt changes in the underlying generative process, a situation also known in the literature as *concept drift* (Gama et al., 2014).

A natural way to deal with these *drifts* in a data stream is to introduce temporal transition models for the parameters of the LVM (Blei and Lafferty, 2006; Perrone et al., 2017). The problem is that these approaches introduce non-conjugate relationships between the global parameters of the extended temporal model. As happens, for example, in Blei and Lafferty (2006) where the Dirichelet prior over topics is conditioned to the Dirichlet posterior over topics in the previous time step. In general, previous attempts to introduce a temporal dynamics on LVMs rely on ad-hoc definitions of transition models which are specific for every LVM at hand, and, also, ad-hoc developments of inference schemes able to deal with these complex temporally extended models.

The method presented in this paper is inspired by previous work on *Bayesian recursive estimation* (Ozkan et al., 2013; Kárnỳ, 2014), *power priors* (Ibrahim and Chen, 2000), and *exponential forgetting* (Honkela and Valpola, 2003). Our approach starts showing how these previously published methods are directly related to one another through the concept of *maximum entropy transition models*. We also show that this scheme can be used as a general approach to define temporal transitions between the global parameters of LVMs.

However, these aforementioned methods only work for slowly changing processes, where the rate of change anticipated by the model needs to be controlled by a quantity that must be set manually. The solution proposed in this paper, on the other hand, can accommodate both gradual and abrupt concept drifts by explicitly modeling the *rate of change* of the data stream as an unobserved mechanism using a fully Bayesian approach. A posterior distribution over the *rate of change* is also provided at every time step, revealing to the user hidden information about the pattern of change in the data stream.

With the explicit modeling of the rate of change, the resulting model class will (generally) not be part of the conjugate exponential family. We then develop an approximate variational inference scheme, based on a novel lower-bound of the data log-likelihood function, which also ensures the computational efficiency required by high-velocity data stream settings. The appropriateness of the approach is investigated through experiments using both synthetic and real-life data (covering energy, finance, geolocation, and text), showing promising results. The proposed method is released as part of an open-source toolbox for scalable probabilistic machine learning (`http://www.amidsttoolbox.com`) (Masegosa et al., 2017c).

2. Preliminaries

2.1 The Probabilistic Model

We shall initially focus on probabilistic models with the structure shown in Figure 1 (a), which is the standard structure of a latent variable model (LVM) (Bishop, 1998; Blei, 2014). This model includes the observed data $\boldsymbol{x} = \boldsymbol{x}_{1:N}$, global hidden variables (or parameters) $\boldsymbol{\beta} = \boldsymbol{\beta}_{1:M}$, a set of local hidden variables $\boldsymbol{z} = \boldsymbol{z}_{1:N}$, and a vector of fixed (hyper) parameters denoted by $\boldsymbol{\alpha}$. Notice how the dynamics of the process is not included in the model of Figure 1 (a); the model will be set in the context of data streams in Section 6, where we extend it to incorporate explicit dynamics over the (global) parameters to capture concept drifts in the data stream.

The joint distribution factorizes into a product of local terms and a global term,

$$p(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{\beta}|\boldsymbol{\alpha}) = p(\boldsymbol{\beta}|\boldsymbol{\alpha}) \prod_{n=1}^{N} p(\boldsymbol{x}_n, \boldsymbol{z}_n|\boldsymbol{\beta}).$$

The LVMs considered in this work belongs to the so-called *conjugate exponential family* (Barndorff-Nielsen, 2014). This model family has been largely studied in the statistics field and cover a wide range of probability distributions and density functions such as Multinomial, Normal, Gamma, Dirichlet, Beta, etc. According to this assumption, the functional form of the conditional distribution of the local variables $(\boldsymbol{x}_n, \boldsymbol{z}_n)$ given the global hidden variables $\boldsymbol{\beta}$ has the well-known *exponential family* form (Barndorff-Nielsen, 2014),

$$\ln p(\boldsymbol{x}_n, \boldsymbol{z}_n|\boldsymbol{\beta}) = \ln h(\boldsymbol{x}_n, \boldsymbol{z}_n) + \boldsymbol{\beta}^T t(\boldsymbol{x}_n, \boldsymbol{z}_n) - a_l(\boldsymbol{\beta}), \tag{1}$$

where the scalar functions $h(\cdot)$ and $a_l(\cdot)$ are the base measure and the log-normalizer, respectively; the vector function $\boldsymbol{t}(\cdot)$ is the *sufficient statistics* vector. The prior distribution $p(\boldsymbol{\beta})$ also belongs to the exponential family, and has the following structure,

$$\ln p(\boldsymbol{\beta}) = \ln h(\boldsymbol{\beta}) + \boldsymbol{\alpha}^T t(\boldsymbol{\beta}) - a_g(\boldsymbol{\alpha}) \tag{2}$$

were the sufficient statistics are $t(\boldsymbol{\beta}) = (\boldsymbol{\beta}, -a_l(\boldsymbol{\beta}))$ and the hyperparameter $\boldsymbol{\alpha}$ has two components $\boldsymbol{\alpha} = (\boldsymbol{\alpha}_1, \alpha_2)$, the first component $\boldsymbol{\alpha}_1$ has the same dimension as $\boldsymbol{\beta}$ and encodes the prior belief about the distribution over $\boldsymbol{\beta}$. The second component $\alpha_2 > 0$ is a scalar and encodes the strength in our prior belief (Bernardo and Smith, 2009). This second parameter is also known in the literature as the *equivalent sample size* (ESS) of the prior distribution (Heckerman et al., 1995)

Our inference goal is to approximate the posterior distribution of the hidden variables given the observations, $p(\boldsymbol{\beta}, \boldsymbol{z}|\boldsymbol{x})$. For the sake of simplicity, here we restrict a bit further our model class [1] and assume it satisfies the so-called *complete conditional* assumption (Hoffman et al., 2013). This assumption states that the conditional distribution over $\boldsymbol{\beta}$ and $\boldsymbol{z}$ given the rest of variables has the same functional form as the priors,

$$\ln p(\boldsymbol{\beta}|\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{\alpha}) = \ln h(\boldsymbol{\beta}) + \eta_g(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{\alpha})^T t(\boldsymbol{\beta}) - a_g(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{\alpha})$$
$$\ln p(\boldsymbol{z}_n|\boldsymbol{x}_n, \boldsymbol{\beta}) = h(\boldsymbol{z}_n) + \eta_l(\boldsymbol{x}_n, \boldsymbol{\beta})^T t(\boldsymbol{z}_n) - a_l(\eta_l(\boldsymbol{x}_n, \boldsymbol{\beta})),$$

where the vector function $\eta(\cdot)$ denotes the *natural parameter vectors* of these conditional probability distributions.

1. All the presented approach also applies to the more general *conjugate exponential family*.

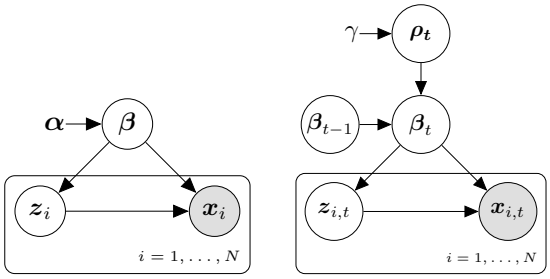


Figure 1: Left figure displays the core of the probabilistic model examined in this paper. Right figure includes a temporal evolution model for $\boldsymbol{\beta}_t$ as described in Section 6.

By Equations (1) and (2), the natural parameter vector of $p(\boldsymbol{\beta}|\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{\alpha})$ can be expressed as

$$\eta_g(\boldsymbol{x}, \boldsymbol{z}, \boldsymbol{\alpha}) = (\boldsymbol{\alpha}_1 + \sum_{n=1}^{N} t(\boldsymbol{x}_n, \boldsymbol{z}_n), \alpha_2 + N). \tag{3}$$

In Bayesian settings, computing the full posterior reduces to updating the natural parameters of the prior. In that sense, the *equivalent sample size* of the posterior is equal to the *equivalent sample size* of the prior plus the size of the observations.

2.2 Variational Inference

Variational inference is a deterministic technique for finding tractable posterior distributions, denoted by q, which approximates the Bayesian posterior, $p(\boldsymbol{\beta}, \boldsymbol{z}|\boldsymbol{x})$, that is often intractable to compute. More specifically, by letting $\mathcal{Q}$ be a set of possible approximations of this posterior, variational inference solves the following optimization problem for any model in the conjugate exponential family

$$\min_{q(\boldsymbol{\beta},\boldsymbol{z})\in\mathcal{Q}} KL\left(q(\boldsymbol{\beta}, \boldsymbol{z}) \,||\, p(\boldsymbol{\beta}, \boldsymbol{z}|\boldsymbol{x})\right), \tag{4}$$

where KL denotes the Kullback-Leibler divergence between two probability distributions.

In the *mean field variational* approach the approximation family $\mathcal{Q}$ is assumed to fully factorize. Following the notation of Hoffman et al. (2013), we have that

$$q(\boldsymbol{\beta}, \boldsymbol{z}|\boldsymbol{\lambda}, \boldsymbol{\phi}) = q(\boldsymbol{\beta}|\boldsymbol{\lambda}) \prod_{n=1}^{N} q(\boldsymbol{z}_n|\boldsymbol{\phi}_n).$$

Furthermore, each factor variational distribution is assumed to belong the same family as the model's complete conditionals,

$$\ln q(\boldsymbol{\beta}|\boldsymbol{\lambda}) = h(\boldsymbol{\beta}) + \boldsymbol{\lambda}^T t(\boldsymbol{\beta}) - a_g(\boldsymbol{\lambda})$$
$$\ln q(\boldsymbol{z}_n|\boldsymbol{\phi}_n) = h(\boldsymbol{z}_n) + \boldsymbol{\phi}_n^T t(\boldsymbol{z}_n) - a_l(\boldsymbol{\phi}_n).$$

As can be seen, $\boldsymbol{\lambda}$ parameterizes the variational distribution of $\boldsymbol{\beta}$, while $\boldsymbol{\phi}$ has the same role for the variational distribution of $\boldsymbol{z}$.

To solve the minimization problem in Equation (4), the variational approach exploits the transformation

$$\ln p(\boldsymbol{x}) = \mathcal{L}(\boldsymbol{\lambda}, \boldsymbol{\phi}|\boldsymbol{x}, \boldsymbol{\alpha}) + KL\left(q(\boldsymbol{\beta}, \boldsymbol{z}|\boldsymbol{\lambda}, \boldsymbol{\phi}) \,||\, p(\boldsymbol{\beta}, \boldsymbol{z}|\boldsymbol{x})\right), \tag{5}$$

where $\mathcal{L}(\cdot|\cdot)$ is a *lower bound* of $\ln p(\boldsymbol{x})$ since KL is non-negative. This lower bound has the following form

$$\mathcal{L}(\boldsymbol{\lambda}, \boldsymbol{\phi}|\boldsymbol{x}, \boldsymbol{\alpha}) = \mathbb{E}_q[\ln p(\boldsymbol{x}|\boldsymbol{z}, \boldsymbol{\beta})] - \mathbb{E}_q[KL\left(q(\boldsymbol{z}|\boldsymbol{\phi}) \,||\, p(\boldsymbol{z}|\boldsymbol{\beta})\right)] - KL\left(q(\boldsymbol{\beta}|\boldsymbol{\lambda}) \,||\, p(\boldsymbol{\beta}|\boldsymbol{\alpha})\right)] \tag{6}$$

We introduce $\boldsymbol{x}$ and $\boldsymbol{\alpha}$ in $\mathcal{L}$'s notation to make explicit the function's dependency on $\boldsymbol{x}$, the data sample, and $\boldsymbol{\alpha}$, the natural parameters of the prior over $\boldsymbol{\beta}$. As $\ln p(\boldsymbol{x})$ is constant, minimizing the KL term is equivalent to maximizing the lower bound. Equation (6) shows the trade-off involved in the lower-bound. The first term measures the model's fit to the data, and favors variational posterior mass concentrated around the maximum likelihood estimate. The second and third terms are regulariser terms, and favor variational posteriors close to their respective prior distributions.

This lower bound can be maximized, for example, by a coordinate ascent method, that iteratively updates each individual variational distributions while holding the others fixed. As shown in (Hoffman et al., 2013), these iterative updating equations has the following closed-form solutions,

$$\boldsymbol{\lambda} = \boldsymbol{\alpha} + \sum_{n=1}^{N} \mathbb{E}_{\boldsymbol{\phi}_n}[(t(\boldsymbol{x}_n, \boldsymbol{z}_n), 1)] \tag{7}$$

$$\boldsymbol{\phi}_n = \mathbb{E}_{\boldsymbol{\lambda}}[\eta_l(\boldsymbol{x}_n, \boldsymbol{\beta})] \tag{8}$$

where $\mathbb{E}_{\boldsymbol{\lambda}}[\cdot]$ and $\mathbb{E}_{\boldsymbol{\phi}_n}[\cdot]$ denote the expected value according to $q(\boldsymbol{\beta}|\boldsymbol{\lambda})$ and $q(\boldsymbol{z}_n|\boldsymbol{\phi}_n)$, respectively. If the number of data points is large, alternative scalable methods can also be used (Hoffman et al., 2013; Masegosa et al., 2017b).

2.3 Streaming Variational Bayes

In this paper, we envision a situation where the data stream is defined by sequence of batches of data generated at discrete points in time. As new batches arrive, we want to update the posterior distribution over the global parameters of the model. The streaming variational Bayes (SVB) algorithm by Broderick et al. (2013) tries to address this problem by using a Bayesian recursive updating approach,

$$p(\boldsymbol{\beta}|\boldsymbol{x}_1, \ldots, \boldsymbol{x}_t) \propto p(\boldsymbol{\beta}|\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{t-1}) \int p(\boldsymbol{x}_t, \boldsymbol{z}_t|\boldsymbol{\beta}) d\boldsymbol{z}_t.$$

So, updating the posterior at time t reduces to a problem of computing a posterior over $\boldsymbol{\beta}$ conditioned to the data $\boldsymbol{x}_t$ and given a prior equal to the posterior in the previous time step $p(\boldsymbol{\beta}|\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{t-1})$.

SVB translates the above recursive updating approach to the variational settings described in the previous sections. Firstly, it approximates the posterior in the previous time step with a variational approximation, $p(\boldsymbol{\beta}|\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{t-1}) \approx q(\boldsymbol{\beta}|\boldsymbol{\lambda}_{t-1})$, and, then, solves the following optimization problem to get a new variational approximation to the posterior at time t by solving the following variational problem,

$$\arg\min_{\boldsymbol{\lambda}_t, \boldsymbol{\phi}_t} \mathcal{L}(\boldsymbol{\lambda}_t, \boldsymbol{\phi}_t|\boldsymbol{x}_t, \boldsymbol{\lambda}_{t-1})$$

3. Related Work

Concept drift in data streams has been extensively studied in the machine learning literature, specially in the context of classification and clustering models (Gaber et al., 2005; Aggarwal, 2007; Gama and Rodrigues, 2009; Gama et al., 2014). One the main techniques employed to address this problem has been *exponential forgetting* (Aggarwal, 2013; Papadimitriou et al., 2005). Under this approach, new data samples are assigned a weight equal to one, but this weight is exponentially decreased after every time step. In that way, old data samples are less relevant than newer data samples when learning the model, accounting for potential drift in the data stream.

Bayesian modeling of non-stationary (i.e., with concept drift) data streams for general probabilistic models has been much less studied. An online variational inference method, which exponentially forgets the variational parameters associated with old data, was proposed by Honkela and Valpola (2003). This approach suffers from the problem of setting an optimal exponential forgetting rate, which must be manually set by the user. A recent proposal, called population variational Bayes (PVB) was introduced by McInerney et al. (2015), which directly builds on the stochastic variational inference (SVI) algorithm (Hoffman et al., 2013). SVI assumes the existence of a fixed data set observed in a sequential manner, and in particular that this data set has a known finite size. This is unrealistic when modeling data streams. PVB addresses this problem by using the frequentist notion of a population distribution, $\mathbf{F}$, which is assumed to generate the data stream by repeatedly sampling M data points at a time. M parameterizes the size of the population, and helps control the variance of the population posterior. By artificially having a high variance in the posterior (i.e. by setting a small M value), PVB is able to accommodate drift in the data set. Unfortunately, M must be specified by the user. No clear rule exists regarding how to set it, and McInerney et al. (2015) show that its optimal value may differ from one data stream to another. The streaming variational Bayes (SVB) algorithm by Broderick et al. (2013) also tries to address the problem of Bayesian inference in the data streams. SVB builds on a Bayesian recursive updating approach, but it assumes data exchangeability and does nor provide any mechanism for dealing with concept drift.

The so-called *power prior* approach (Ibrahim and Chen, 2000) has been independently studied in the context of data aggregation for Bayesian modeling. Power priors provide a sound mechanism for Bayesian updating in the light of new data, and partial forgetting of old data. This approach enjoys nice theoretical properties (Ibrahim et al., 2003) but it depends again on a hyperparameter, which must be set by the user to control the forgetting rate.

A time series based modeling approach for concept drift using *implicit transition models* was pursued by Ozkan et al. (2013); Kárnỳ (2014). Unfortunately, the implicit transition model also depends on a hyper-parameter determining the forgetting-factor, which has to be manually set.

Many other works have proposed ad-hoc extension for specific LVMs which are able to deal with non-stationary data streams (Shi and Zhu, 2014; Williamson et al., 2010a). A remarkable effort has been given to dynamic extension of LDA models (Blei and Lafferty, 2006; Williamson et al., 2010b; Perrone et al., 2017), but none of them is applicable to general conjugate exponential family models. Moreover, most of them rely on complex inference mechanisms outside the class of the nicely behaved variational inference over conjuagate exponential models, where solutions to the gradients of the evidence lower function can be computed in closed-form.

Our approach provides a learning framework in the context of non-stationary data streams which is applicable to any LVMs belonging to the conjugate exponential family, and which does not required to manually set hyperparameters for defining the degree of forgetting. By applying a pure

Bayesian approach, our method provides at every time step a posterior probability over the optimal forgetting rate to better accommodate the current data. This approach builds on a novel interpretation of the *exponential forgetting* mechanism (Aggarwal, 2013; Papadimitriou et al., 2005) as an *implicit transition model* (Ozkan et al., 2013). We also show that this *implicit transition models* can be expressed in the form of a *power prior* (Ibrahim and Chen, 2000). Once expressed in that way, we place a hierarchical prior over the exponential forgetting rate, and derive a novel variational inference scheme which maintains the computational efficiency of variational methods over conjugate exponential models.

4. Exponential Forgetting in Bayesian Learning with Data Streams

As stated in Section 1 and Section 3, exponential forgetting is a classic technique used in machine learning and data mining to gradually forget past data and put more focus on more recent data samples when performing online learning. In machine learning this idea is usually implemented by exponentially down-weighting the loss function term associated with each data sample, so that data samples closer in time have more impact on the model that older data samples (Gaber et al., 2005).

In a probabilisitic settings, exponential forgetting is achieved by using a log-likelihood function with the form

$$\ln p(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_t|\boldsymbol{\beta}) = \sum_{i=1}^{t} \rho^{t-i} \ln p(\boldsymbol{x}_i|\boldsymbol{\beta}) + cte,$$

where $\rho \in [0, 1]$ is the exponential decay weight.

Similarly, in Bayesian learning settings, we can use this scheme to compute the posterior,

$$p(\boldsymbol{\beta}|\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_t, \rho) \propto p(\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_t|\boldsymbol{\beta}, \rho)p(\boldsymbol{\beta}) = p(\boldsymbol{x}_t|\boldsymbol{\beta})p(\boldsymbol{x}_{t-1}|\boldsymbol{\beta})^{\rho} \cdots p(\boldsymbol{x}_1|\boldsymbol{\beta})^{\rho^{t-1}}p(\boldsymbol{\beta}).$$

This scheme also applies to a variational learning letting by considering this exponential down-weighted likelihood instead of the standard data likelihood, as used in (Honkela and Valpola, 2003). Then the *lower bound* function has the following form,

$$\mathcal{L}_{\rho}(\boldsymbol{\lambda}, \boldsymbol{\phi}|\boldsymbol{x}, \boldsymbol{\alpha}_u) = \mathbb{E}_q \left[\sum_{i=1}^{t} \rho^{t-i} \ln p(\boldsymbol{x}_i|\boldsymbol{z}_i, \boldsymbol{\beta}) \right] - KL\left(q(\boldsymbol{\beta}, \boldsymbol{z}|\boldsymbol{\lambda}, \boldsymbol{\phi}) \,||\, p(\boldsymbol{\beta}, \boldsymbol{z}|\boldsymbol{\alpha}_u)\right). \tag{9}$$

The updating equation of the coordinate gradient ascent algorithm described in Equation (7) can now be expressed as follows (Winn and Bishop, 2005),

$$\boldsymbol{\lambda} = \boldsymbol{\alpha}_u + \sum_{i=1}^{t} \rho^{t-i} \mathbb{E}_{\boldsymbol{\phi}_i}[(t(\boldsymbol{x}_i, \boldsymbol{z}_i), 1)]. \tag{10}$$

The main point here is to highlight how, at the convergence point, the variational solution $\boldsymbol{\lambda}$ exponentially down-weights old data samples.

Exponential forgetting also addresses the problem of Bayesian learning over unbounded data streams. According to Equation (7), the $\boldsymbol{\lambda}$ parameter has two components, $\boldsymbol{\lambda}_1$ a component with the same dimension as $\boldsymbol{\beta}$, and a second component λ_2 which is a scalar that corresponds to the *equivalent sample size* of the variational posterior $q(\boldsymbol{\beta}|\boldsymbol{\lambda})$. If we denote by $\lambda_{2,t}$ the equivalent

sample size of the variational posterior after seeing t samples, then this value can be computed as

$$\lambda_{2,t} = \alpha_2 + \sum_{i=1}^{t} \rho^{(i-1)}.$$

If $\rho < 1$, then $\lambda_{2,t}$ converges to a finite number,

$$\lim_{t\to\infty} \lambda_{2,t} = \alpha_2 + \frac{1}{1-\rho}, \tag{11}$$

avoiding the problem of having a degenerated Bayesian posterior distribution in the presence of an unbounded data stream. As noted in (Olesen et al., 1992; Ozkan et al., 2013), this schema approximates a posterior distribution conditioned on the last $\frac{1}{1-\rho}$ data samples of the stream.

4.1 Exponential Forgetting in SVI and PVB

Stochastic varitional inference (SVI) (Hoffman et al., 2013) is a widely used variational learning algorithm for dealing with large data sets. As commented above, Population Variational Bayes (McInerney et al., 2015) is a simple modification of SVI used when the total size of the data set is unknown. When these algorithms are applied in data streaming settings, they use a constant learning rate ν,[2] and the sequential updating equation of the global variational parameters $\boldsymbol{\lambda}$ can be written as

$$\boldsymbol{\lambda}_t = (1-\nu)\boldsymbol{\lambda}_{t-1} + \nu(\boldsymbol{\alpha}_u + S\mathbb{E}_{\boldsymbol{\phi}_t}[(t(\boldsymbol{x}_t, \boldsymbol{z}_t), 1)]), \tag{12}$$

where S is equal the total size of the data set N in the case of SVI, or S is equal to the size of the population M in the case of PVB. By expanding this equation, we find that

$$\boldsymbol{\lambda}_t = (1-(1-\nu)^t)\boldsymbol{\alpha}_u + N\nu \sum_{i=1}^{t} (1-\nu)^{t-i} \mathbb{E}_{\boldsymbol{\phi}_i}[(t(\boldsymbol{x}_i, \boldsymbol{z}_i), 1)]. \tag{13}$$

The above equation highlights that SVI and PVB also exponentially down-weight old data samples, with a forgetting rate $\rho = 1 - \nu$ (compared the above equation with Equation (10)). Therefore, this is one of the mechanisms these two methods use to adapt to drifts in the data stream.

In the case of the PVB algorithm, the parameter M helps to adapt to drifts in the data set through the effect is has in computing $\boldsymbol{\phi}_i$, as discussed in McInerney et al. (2015). However, when the model does not contain local random variables, the variational parameters $\boldsymbol{\phi}_i$ do not exist and, then, the size of the population does not play any role in adapting to drifts in the data stream.

5. MaxEntropy Transition Models

5.1 Transitioning Model Parameters

In order to extend the model in Figure 1 (a) to data streams, we may introduce a transition model $p(\boldsymbol{\beta}_t|\boldsymbol{\beta}_{t-1})$ to explicitly model the evolution of the parameters over time, enabling the estimation of the predictive density at time t:

$$p(\boldsymbol{\beta}_t|\boldsymbol{x}_{1:t-1}) = \int p(\boldsymbol{\beta}_t|\boldsymbol{\beta}_{t-1}) p(\boldsymbol{\beta}_{t-1}|\boldsymbol{x}_{1:t-1}) d\boldsymbol{\beta}_{t-1}. \tag{14}$$

2. It is usually set to small values like 0.1 or 0.01.

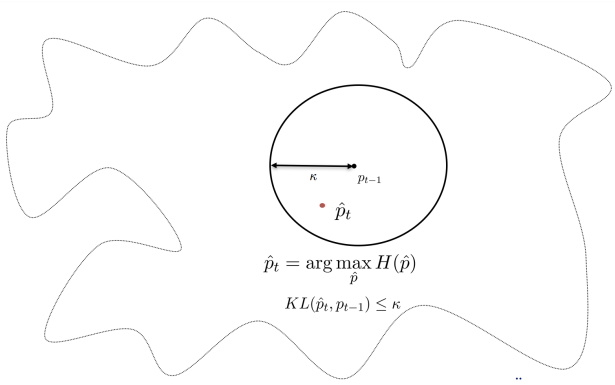


Figure 2: MaxEntropy Transition Models

However, this approach introduces two problems. First of all, in non-stationary domains we may not have a single transition model or the transition model may be unknown. Secondly, if we seek to position the model within the conjugate exponential family in order to be able to compute the gradients of $\mathcal{L}$ in closed-form, we need to ensure that the distribution family for $\boldsymbol{\beta}_t$ is its own conjugate distribution, thereby severely limiting model expressivity (we can, e.g., not assign a Dirichlet distribution to $\boldsymbol{\beta}_t$).

Rather than explicitly modeling the evolution of the $\boldsymbol{\beta}_t$ parameters as in Equation (14), we instead follow the approach of Kárnỳ (2014) and Ozkan et al. (2013) who define the time evolution model implicitly by constraining the maximum KL divergence over consecutive parameter distributions. Specifically, by defining

$$p_\delta(\boldsymbol{\beta}_t|\boldsymbol{x}_{1:t-1}) = \int \delta(\boldsymbol{\beta}_t - \boldsymbol{\beta}_{t-1}) p(\boldsymbol{\beta}_{t-1}|\boldsymbol{x}_{1:t-1}) d\boldsymbol{\beta}_{t-1} \tag{15}$$

one can restrict the space of possible distributions $p(\boldsymbol{\beta}_t|\boldsymbol{x}_{1:t-1})$, supported by an unknown transition model, by the constraint

$$KL\left(p(\boldsymbol{\beta}_t|\boldsymbol{x}_{1:t-1}) \,||\, p_\delta(\boldsymbol{\beta}_t|\boldsymbol{x}_{1:t-1})\right) \leq \kappa. \tag{16}$$

Kárnỳ (2014) and Ozkan et al. (2013) seek to approximate $p(\boldsymbol{\beta}_t|\boldsymbol{x}_{1:t-1})$ by the distribution $\hat{p}(\boldsymbol{\beta}_t|\boldsymbol{x}_{1:t-1})$ having maximum entropy under the constraint in (16); for continuous distributions the maximum entropy can be formulated relative to an uninformative prior density $p_u(\boldsymbol{\beta}_t)$, which corresponds to the Kullbach-Leibler divergence between the two distributions. This approach ensures that we will not underestimate the uncertainty in the parameter distribution and the particular solution being sought takes the form

$$\hat{p}(\boldsymbol{\beta}_t|\boldsymbol{x}_{1:t-1}, \rho_t) \propto p_\delta(\boldsymbol{\beta}_t|\boldsymbol{x}_{1:t-1})^{\rho_t} p_u(\boldsymbol{\beta}_t)^{(1-\rho_t)}, \tag{17}$$

where $0 \leq \rho_t \leq 1$ is indirectly defined by (16), and therefore depends on the user defined parameter κ.

In our streaming data setting we follow *assumed density filtering* (Lauritzen, 1992) and the SVB approach (Broderick et al., 2013) and employ the approximation $p(\boldsymbol{\beta}_{t-1}|\boldsymbol{x}_{1:t-1}) \approx q(\boldsymbol{\beta}_{t-1}|\boldsymbol{\lambda}_{t-1})$, where $q(\boldsymbol{\beta}_{t-1}|\boldsymbol{\lambda}_{t-1})$ is the variational distribution calculated in the previous time step. Using this approximation in (14) and (15), we can express p_δ in terms of $\boldsymbol{\lambda}_{t-1}$ in which case (17) becomes

$$\hat{p}(\boldsymbol{\beta}_t|\boldsymbol{\lambda}_{t-1}, \rho_t) \propto p_\delta(\boldsymbol{\beta}_t|\boldsymbol{\lambda}_{t-1})^{\rho_t} p_u(\boldsymbol{\beta}_t)^{(1-\rho_t)}, \tag{18}$$

which we use as the prior density for time step t. Now, if $p_u(\boldsymbol{\beta}_t)$ belong to the same family as $q(\boldsymbol{\beta}_{t-1}|\boldsymbol{\lambda}_{t-1})$, then $\hat{p}(\boldsymbol{\beta}_t|\boldsymbol{\lambda}_{t-1}, \rho_t)$ will stay within the same family and have natural parameters $\rho_t\boldsymbol{\lambda}_{t-1} + (1-\rho_t)\boldsymbol{\alpha}_u$, where $\boldsymbol{\alpha}_u$ are the natural parameters of $p_u(\boldsymbol{\beta}_t)$. Thus, under this approach, the transitioned posterior remains within the same exponential family, so we can enjoy the full flexibility of the conjugate exponential family (i.e. computing gradients of the $\mathcal{L}$ function in closed form), an option that would not be available if one were to explicitly specify a transition model as in Equation (14).

So, at each time step, we simply have to solve the following variational problem, where only the prior changes with respect to the original SVB approach,

$$\arg\max_{\boldsymbol{\lambda}_t, \boldsymbol{\phi}_t} \mathcal{L}(\boldsymbol{\lambda}_t, \boldsymbol{\phi}_t|\boldsymbol{x}_t, \rho_t\boldsymbol{\lambda}_{t-1} + (1-\rho_t)\boldsymbol{\alpha}_u). \tag{19}$$

We shall refer to the method outlined in this section as SVB with *power priors* (SVB-PP). The term *power priors* (Ibrahim and Chen, 2000) will be clear in Section 5.3.

5.2 Exponential Forgetting as MaxEntropy Transition Models

In this section we show that the exponential forgetting mechanism used in Bayesian learning settings described in Section 4 is a MaxEntropy transition model with constant forgetting rate ρ .

The updating equation detailed in Equation (7) to optimize the lower-bound function described in Equation (6) can be easily adapted to optimize the lower-bound associated to the MaxEntropy transition models given in Equation (19). This new updating equation for MaxEntropy transition models can be expressed as follows,

$$\boldsymbol{\lambda}_t = \mathbb{E}_{\boldsymbol{\phi}_t}[(t(\boldsymbol{x}_t, \boldsymbol{z}_t), 1)] + \rho\boldsymbol{\lambda}_{t-1} + (1-\rho)\boldsymbol{\alpha}_u. \tag{20}$$

Expanding the above equation we have

$$\boldsymbol{\lambda}_t = \sum_{i=1}^{t} \rho^{t-i}\mathbb{E}_{\boldsymbol{\phi}_i}[(t(\boldsymbol{x}_i, \boldsymbol{z}_i), 1)] + (1-\rho^t)\boldsymbol{\alpha}_u, \tag{21}$$

where we can see the scheme of exponentially down-weighting old data samples as in Equation (10). The only difference between the above equation and Equation (10) is therefore in the use of the prior term. When $\rho < 1$, ρ^t converges to zero, and they become identical in the limit. Therefore, it is clear that the classic technique of exponential forgetting, which was usually supported by heuristic arguments, has a sound interpretation in terms of MaxEntropy transition models.

5.3 Power Priors as MaxEntropy Transition Models

Power priors (Ibrahim and Chen, 2000) is a widely used class of informative priors for dealing with situations in which historical data are available. Let $\boldsymbol{x}_0$ denote a previously obtained data set, and let $\boldsymbol{x}_1$ be our current data set. According to the *power priors* scheme (Ibrahim and Chen, 2000), the posterior probability over the model parameters should be computed as follows

$$p(\boldsymbol{\beta}|\boldsymbol{x}_1, \boldsymbol{x}_0, \rho) \propto p(\boldsymbol{x}_1|\boldsymbol{\beta})p(\boldsymbol{x}_0|\boldsymbol{\beta})^\rho p(\boldsymbol{\beta}), \tag{22}$$

where $\rho \in [0, 1]$ is a scalar parameter down-weighting the likelihood of historical data relative to the likelihood of the current data.

Check this one. We define PP in 22, then say say that "this approach"

As stated in the following lemma, power priors can also be interpreted as maximum entropy transition models.

Lemma 1 *The Bayesian updating scheme described by Figure 1 (b) and Equation* (17)*, but with* ρ_t *fixed to a constant value, is equivalent to the recursive application of the Bayesian updating scheme of power priors given in Equation* (22)*.*

Proof Translate the recursive Bayesian updating approach of power priors into an equivalent two time slice model, where $\boldsymbol{\beta}_0$ is given a prior distribution p and $p(\boldsymbol{\beta}_1|\boldsymbol{\beta}_0)$ is a Dirac delta function. The distribution $p(\boldsymbol{\beta}_1|\boldsymbol{x}_0, \boldsymbol{x}_1, \rho)$ in this model is equivalent to $p(\boldsymbol{\beta}|\boldsymbol{x}_1, \boldsymbol{x}_0, \rho)$, which, in turn, is equivalent (up to proportionality) to $p(\boldsymbol{x}_1|\boldsymbol{\beta}_1)\hat{p}(\boldsymbol{\beta}_1|\boldsymbol{x}_0, \rho_t)$. Note that the last $\hat{p}$ term can alternatively be expressed as $\hat{p}(\boldsymbol{\beta}_1|\boldsymbol{x}_0, \rho_t) \propto p_\delta(\beta_1|\boldsymbol{x}_0)^\rho p(\beta_1)^{1-\rho} \propto p_\delta(\boldsymbol{x}_0|\beta_1)^\rho p(\beta_1)$. ■

This connection allows us to introduce well known results of power priors (Ibrahim et al., 2003),

$$p(\boldsymbol{\beta}|\boldsymbol{x}_1, \boldsymbol{x}_0, \rho) = \arg\min_{r \in \mathcal{P}} \quad \rho\ KL\left(r\ ||\ p(\boldsymbol{\beta}|\boldsymbol{x}_1, \boldsymbol{x}_0, \rho = 1)\right) + (1-\rho)\ KL\left(r\ ||\ p(\boldsymbol{\beta}|\boldsymbol{x}_1, \boldsymbol{x}_0, \rho = 0)\right)$$

where $\mathcal{P}$ denotes the set of all possible densities over $\boldsymbol{\beta}$. I.e. "*power priors minimize the convex combination of KL divergences between two extremes: one in which no historical data is used and the other in which the historical data and current data are given equal weight.*"

This paragraph reads a bit strange, I think. Optimality in what way is a natural question at this point. Should we restructure, by first giving the result in the equation below, then use the quote as a "discussion"?

6. Hierarchical Power Priors

6.1 A Hierarchical Prior over the forgetting rate ρ

In the approach taken by Ozkan et al. (2013) (and, by extension, SVB-PP), the forgetting factor ρ_t is user-defined. In this paper, we instead pursue a (hierarchical) Bayesian approach and introduce a prior distribution over ρ_t allowing the distribution over ρ_t (and thereby the forgetting mechanism) to adapt to the data stream. In this section we extend the model in Figure 1 (a) to also account for the dynamics of the data stream being modeled. We shall here assume that only the parameters $\boldsymbol{\beta}$ in Figure 1 (a) are time-varying, which we will indicate with the subscript t, i.e., $\boldsymbol{\beta}_t$. The resulting model can be illustrated as in Figure 1 (b). We shall refer to models of this type as *hierarchical power prior* (HPP) models.

We will show in Section 6.3 that the exponential and normal distributions, both of which truncated to the interval [0,1], are valid alternatives as prior distributions, $p(\rho_t|\gamma)$. The densities of these distributions have the following forms,

$$p(\rho_t|\gamma) = \frac{\gamma \exp(-\gamma\rho_t)}{1 - \exp(-\gamma)}, \quad 0 \leq \rho_t \leq 1 \tag{23}$$

$$p(\rho_t|\mu, \sigma) = \frac{\exp\left(-(\rho_t - \mu)^2/(2\sigma^2)\right)}{\sqrt{2\pi\sigma^2}\left(\Phi(\frac{1-\mu}{\sigma}) - \Phi(\frac{-\mu}{\sigma})\right)}, \quad 0 \leq \rho_t \leq 1 \tag{24}$$

where Φ represents the cumulative distribution of the standard normal distribution, $\mu \in \mathbb{R}$ (can be outside the interval $[0, 1]$) and $\sigma > 0$. Since the natural parameters of the normal distribution are $\left(\mu/\sigma^2, -1/(2\sigma^2)\right)$, it is sometimes convenient to talk in terms of the precision $\eta = 1/\sigma^2$ (the reciprocal of the variance). Using the precision, the natural parameters become $(\mu\eta, -\eta/2)$. Notice that the precision η appears in both components.

Figure 3 plots different densities of both distributions for different values of the parameters. The Truncated Exponential allows to model a uniform prior and priors that either favors ρ_t values close to 1 (i.e. non-forgetting past data) or ρ_t values close to 0 (i.e. forgetting past data). The Truncated

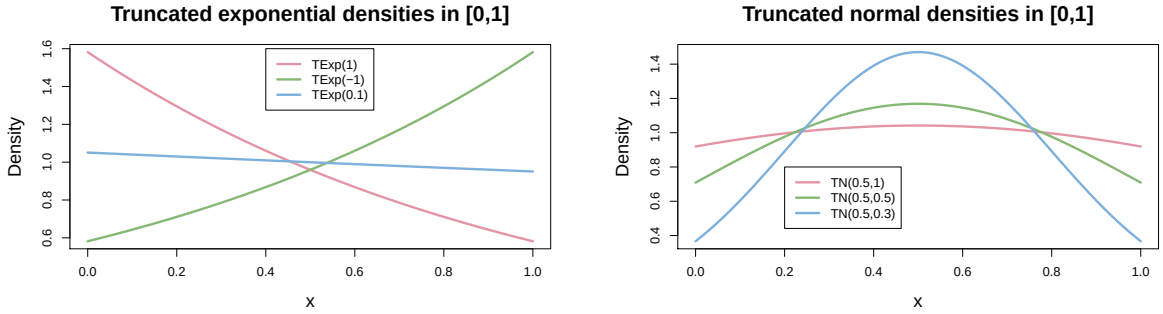


Figure 3: Density functions for the Truncated Exponential and the Truncated Normal distributions, respectively, for different values of their parameters.

Normal distribution, when using mean equal to 0.5, tends to favor non-extreme ρ_t values (i.e. partial forgetting of past data), where the variance parameter defines the strength of this belief.

For later use, we also detail here the equation for computing the expected value of ρ_t for both distributions:

$$\mathbb{E}[\rho_t|\gamma] = \frac{1}{(1-e^{-\gamma})} - \frac{1}{\gamma}, \tag{25}$$

$$\mathbb{E}_q[\rho_t|\mu,\sigma] = \mu + \sigma \frac{\phi(\frac{-\mu}{\sigma}) - \phi(\frac{1-\mu}{\sigma})}{\Phi(\frac{1-\mu}{\sigma}) - \Phi(\frac{-\mu}{\sigma})}, \tag{26}$$

where γ is the mean parameter parameter of the truncated exponential; μ and σ are the parameters of the truncated normal distribution in $[0,1]$, and ϕ and Φ are respectively the probability density function and the cumulative distribution function of the standard normal distribution.

6.2 The double lower-bound

For updating the model distributions we pursue a variational approach, where we seek to maximize the evidence lower bound $\mathcal{L}$ in Equation (5) for time step t. However, since the model in Figure 1 (b) does not define a conjugate exponential distribution due to the introduction of $p(\rho_t|\gamma)$, we cannot maximize $\mathcal{L}$ directly. Instead we will derive a (double) lower bound $\hat{\mathcal{L}}$ (with $\hat{\mathcal{L}} \leq \mathcal{L}$) and use this lower bound as a proxy for the updating rules of the variational posteriors.

First of all, by instantiating the lower bound $\mathcal{L}_{HPP}(\boldsymbol{\lambda}_t, \boldsymbol{\phi}_t, \boldsymbol{\omega}_t|\boldsymbol{x}_t, \boldsymbol{\lambda}_{t-1})$ in Equation (5) for the HPP model we obtain

$$\begin{aligned}\mathcal{L}_{HPP}(\boldsymbol{\lambda}_t, \boldsymbol{\phi}_t, \boldsymbol{\omega}_t|\boldsymbol{x}_t, \boldsymbol{\lambda}_{t-1}) = \mathbb{E}_q[\ln p(\boldsymbol{x}_t|\boldsymbol{Z}_t, \boldsymbol{\beta}_t)] - \mathbb{E}_q[KL\left(q(\boldsymbol{Z}_t|\boldsymbol{\phi}_t)\,||\,p(\boldsymbol{Z}_t|\boldsymbol{\beta}_t)\right)] \\ - \mathbb{E}_q[KL\left(q(\boldsymbol{\beta}_t|\boldsymbol{\lambda}_t)\,||\,\hat{p}(\boldsymbol{\beta}_t|\boldsymbol{\lambda}_{t-1}, \rho_t)\right)] \\ - KL\left(q(\rho_t|\boldsymbol{\omega}_t)\,||\,p(\rho_t|\gamma)\right)\end{aligned} \tag{27}$$

where $\boldsymbol{\omega}_t$ is the variational parameter for the variational distribution for ρ_t. For ease of presentation we shall sometimes drop from $\mathcal{L}_{HPP}(\boldsymbol{\lambda}_t, \boldsymbol{\phi}_t, \boldsymbol{\omega}_t|\boldsymbol{x}_t, \boldsymbol{\lambda}_{t-1})$ the subscript as well as the explicit specification of the parameters when this is otherwise clear from the context.

We now define $\hat{\mathcal{L}}_{HPP}(\boldsymbol{\lambda}_t, \boldsymbol{\phi}_t, \boldsymbol{\omega}_t | \boldsymbol{x}_t, \boldsymbol{\lambda}_{t-1})$ as

$$\begin{aligned}\hat{\mathcal{L}}_{HPP}(\boldsymbol{\lambda}_t, \boldsymbol{\phi}_t, \boldsymbol{\omega}_t | \boldsymbol{x}_t, \boldsymbol{\lambda}_{t-1}) = \mathbb{E}_q[\ln p(\boldsymbol{x}_t | \boldsymbol{Z}_t, \boldsymbol{\beta}_t)] - \mathbb{E}_q[KL\left(q(\boldsymbol{Z}_t|\boldsymbol{\phi}_t) \,||\, p(\boldsymbol{Z}_t|\boldsymbol{\beta}_t)\right)] \\ - \mathbb{E}_q[\rho_t] KL\left(q(\boldsymbol{\beta}_t|\boldsymbol{\lambda}_t) \,||\, p(\boldsymbol{\beta}_t|\boldsymbol{\lambda}_{t-1})\right) \\ - (1 - \mathbb{E}_q[\rho_t]) KL\left(q(\boldsymbol{\beta}_t|\boldsymbol{\lambda}_t) \,||\, p(\boldsymbol{\beta}_t|\boldsymbol{\alpha}_u)\right) \\ - KL\left(q(\rho_t|\boldsymbol{\omega}_t) \,||\, p(\rho_t|\gamma)\right) \end{aligned} \tag{28}$$

which provides a lower bound for $\mathcal{L}$.

Theorem 1 *$\hat{\mathcal{L}}_{HPP}$ gives a lower bound for $\mathcal{L}_{HPP}$:*

$$\hat{\mathcal{L}}_{HPP}(\boldsymbol{\lambda}_t, \boldsymbol{\phi}_t, \boldsymbol{\omega}_t | \boldsymbol{x}_t, \boldsymbol{\lambda}_{t-1}) \leq \mathcal{L}_{HPP}(\boldsymbol{\lambda}_t, \boldsymbol{\phi}_t, \boldsymbol{\omega}_t | \boldsymbol{x}_t, \boldsymbol{\lambda}_{t-1}).$$

Proof We start by looking at the difference between the two bounds $\mathcal{L}_{HPP}$ and $\hat{\mathcal{L}}_{HPP}$, which is given by the log-normalizer of $\hat{p}(\boldsymbol{\beta}_t|\boldsymbol{\lambda}_{t-1}, \rho_t)$:

$$\begin{aligned}\hat{\mathcal{L}}_{HPP} - \mathcal{L}_{HPP} = \mathbb{E}_q[\rho_t a_g(\boldsymbol{\lambda}_{t-1}) + (1-\rho_t) a_g(\boldsymbol{\alpha}_u) \\ + a_g(\rho_t \boldsymbol{\lambda}_{t-1} + (1-\rho_t)\boldsymbol{\alpha}_u)]\end{aligned} \tag{29}$$

Next, observe that $a_g(\rho_t \boldsymbol{\lambda}_{t-1} + (1-\rho_t)\boldsymbol{\alpha}_u) \leq \rho_t a_g(\boldsymbol{\lambda}_{t-1}) + (1-\rho_t) a_g(\boldsymbol{\alpha}_u)$ because the log-normalizer a_g is always a convex function (Wainwright et al., 2008), and the result follows. Full details are given in the supplementary material. ■

Even though Equation (28) defines an alternative objective function, when we compare this double lower bound with Equation (6) we can observe that the double lower bound still have the intuitive interpretation of the standard lower bound in terms of data fitting and Kullback-Leibler (KL) regularization. The only difference is that the KL regularization term associated to $q(\boldsymbol{\beta}_t|\boldsymbol{\lambda}_t)$ appears now as a convex combination of two KL terms, one regularizing with respect to $p(\boldsymbol{\beta}_t|\boldsymbol{\lambda}_{t-1})$ and the other with respect to $p(\boldsymbol{\beta}_t|\boldsymbol{\alpha}_u)$, whith $\mathbb{E}_q[\rho_t]$ acting as a combination-factor.

Rather than seeking to maximize $\mathcal{L}$ we will instead maximize $\hat{\mathcal{L}}$, see Equation (29). Thus, maximizing $\hat{\mathcal{L}}$ wrt. the variational parameters $\boldsymbol{\lambda}_t$ and $\boldsymbol{\phi}$ also maximizes $\mathcal{L}$. By the same observation, we also have that the (natural) gradients are consistent relative to the two bounds:

Corollary 1

$$\begin{aligned}\nabla^{nat}_{\boldsymbol{\lambda}_t} \hat{\mathcal{L}}_{HPP}(\boldsymbol{\lambda}_t, \boldsymbol{\phi}_t, \boldsymbol{\omega}_t | \boldsymbol{x}_t, \boldsymbol{\lambda}_{t-1}) &= \nabla^{nat}_{\boldsymbol{\lambda}_t} \mathcal{L}_{HPP}(\boldsymbol{\lambda}_t, \boldsymbol{\phi}_t, \boldsymbol{\omega}_t | \boldsymbol{x}_t, \boldsymbol{\lambda}_{t-1}) \\ &= \nabla^{nat}_{\boldsymbol{\lambda}_t} \mathcal{L}(\boldsymbol{\lambda}_t, \boldsymbol{\phi}_t | \boldsymbol{x}_t, \mathbb{E}_q[\rho_t]\boldsymbol{\lambda}_{t-1} + (1 - \mathbb{E}[\rho_t])\boldsymbol{\alpha}_u)\end{aligned}$$

The same result holds for $\boldsymbol{\phi}_t$.

Proof Follows immediately from Equation (29) because the difference does not depend of $\boldsymbol{\lambda}_t$ and $\boldsymbol{\phi}_t$. ■

Thus, updating the variational parameters $\boldsymbol{\lambda}_t$ and $\boldsymbol{\phi}_t$ in HPP models can be done as for regular conjugate exponential models of the form in Figure 1. A pseudo-code description of the algorithm can be found in Algorithm 1 when ρ_t is assumed to follow a Truncted Exponential distribution.

In order to update $\boldsymbol{\omega}_t$ we rely on $\hat{\mathcal{L}}$, which we can maximize using the natural gradient wrt. $\boldsymbol{\omega}_t$ (Sato, 2001) and which can be calculated in closed form for a restricted distribution family for ρ_t, as stated in the following result.

We could make Algorithm 1 more general by stating that the expectation of rho is

Algorithm 1 SVB with Hierarchical Power Priors and Truncated Exponential (SVB-HPP-Exp)

Input: A data batch $\boldsymbol{x}_t$, the variational posterior in previous time step $\boldsymbol{\lambda}_{t-1}$.
Output: $(\boldsymbol{\lambda}_t, \boldsymbol{\phi}_t, \boldsymbol{\omega}_t)$, a new update of the variational posterior.

$\boldsymbol{\lambda}_t \leftarrow \boldsymbol{\lambda}_{t-1}$.
$\mathbb{E}_q[\rho_t] \leftarrow 0.5$.
Randomly initialize $\boldsymbol{\phi}_t$.
repeat
  $(\boldsymbol{\lambda}_t, \boldsymbol{\phi}_t) = \arg\min_{\boldsymbol{\lambda}_t, \boldsymbol{\phi}_t} \mathcal{L}(\boldsymbol{\lambda}_t, \boldsymbol{\phi}_t | \boldsymbol{x}_t, \mathbb{E}_q[\rho_t]\boldsymbol{\lambda}_{t-1} + (1 - \mathbb{E}[\rho_t])\boldsymbol{\alpha}_u)$
  $\omega_t = KL\left(q(\boldsymbol{\beta}_t|\boldsymbol{\lambda}_t) \,||\, p_u(\boldsymbol{\beta}_t)\right) - KL\left(q(\boldsymbol{\beta}_t|\boldsymbol{\lambda}_t) \,||\, p_\delta(\boldsymbol{\beta}_t|\boldsymbol{\lambda}_{t-1})\right) + \gamma$
  Update $\mathbb{E}_q[\rho_t]$ according to Equation (25) or Equation (26).
until convergence
return $(\boldsymbol{\lambda}_t, \boldsymbol{\phi}_t, \omega_t)$

Lemma 2 *Assuming that the first component of the sufficient statistics function for ρ_t is the identity function, i.e. $\boldsymbol{t}_1(\rho_t) = \rho_t$, we have*

$$\begin{aligned}\frac{\partial^{nat}\hat{\mathcal{L}}}{\partial \boldsymbol{\omega}_{t,1}} =& KL\left(q(\boldsymbol{\beta}_t|\boldsymbol{\lambda}_t) \,||\, p_u(\boldsymbol{\beta}_t)\right) - KL\left(q(\boldsymbol{\beta}_t|\boldsymbol{\lambda}_t) \,||\, p_\delta(\boldsymbol{\beta}_t|\boldsymbol{\lambda}_{t-1})\right) + \gamma_1 - \boldsymbol{\omega}_{t,1} \\ \frac{\partial^{nat}\hat{\mathcal{L}}}{\partial \boldsymbol{\omega}_{t,k}} =& \gamma_k - \boldsymbol{\omega}_{t,k} \quad (k \neq 1)\end{aligned} \tag{30}$$

Proof Based on a straightforward algebraic derivation of the gradient using standard properties of the exponential family. Full details are given in the supplementary material. ■

From the above lemma we can easily deduce that the truncated exponential distributions, whose sufficient statistics are $t(\rho_t) = \rho_t$, and the truncated normal distribution, whose sufficient statistics are $t(\rho_t) = (\rho_t, \rho_t^2)^T$, satisfied the criteria to be considered as hierarchical priors for ρ_t.

The problem of the above result is that for $k \neq 1$, the optimal $\boldsymbol{\omega}_{t,k}$ is just equal to the the prior value, i.e. $\boldsymbol{\omega}_{t,k} = \gamma_k$. In the case of the Truncated Normal, which has a two-dimensional natural parameter vector, it would imply that the variance of the posterior $q(\rho_t|\boldsymbol{\omega}_t)$, denoted by σ_q^2, will be equal to the variance of prior, denoted by σ_p^2, which has to be set manually[3]. To address the issue of having to manually fixed the variance of the Truncated Normal prior, σ_p^2, we employ an empirical Bayes approach and consider σ_p^2 as another free parameter of the double lower bound we want to optimize. So, we need to compute the gradient of the double lower bound w.r.t. this parameter,

$$\begin{aligned}\frac{\partial \hat{\mathcal{L}}}{\partial \sigma_p^2} &= \frac{\partial \gamma_1}{\partial \sigma_p^2}\frac{\partial \hat{\mathcal{L}}}{\partial \gamma_1} + \frac{\partial \gamma_2}{\partial \sigma_p^2}\frac{\partial \hat{\mathcal{L}}}{\partial \gamma_2} \\ &= -\frac{\mu_p}{\sigma_p^4}(\mathbb{E}[\rho_t|\mu_q, \sigma_q^2] - \mathbb{E}[\rho_t|\mu_p, \sigma_p^2]) + \frac{1}{2\sigma_p^4}(\mathbb{E}[\rho_t^2|\mu_q, \sigma_q^2] - \mathbb{E}[\rho_t^2|\mu_p, \sigma_p^2]),\end{aligned}$$

where $\gamma = (\mu_p/\sigma_p^2, -1/(2\sigma_p^2))$ is the natural parameter vector of the Truncated Normal prior, and μ_p and μ_q denote the mean of the Truncated Normal prior and posterior over ρ_t, respectively. We set μ_p to 0.5 trying to define a a non-informative and symmetric prior.

3. We dropped the t-index in σ_q^2 for simplicity.

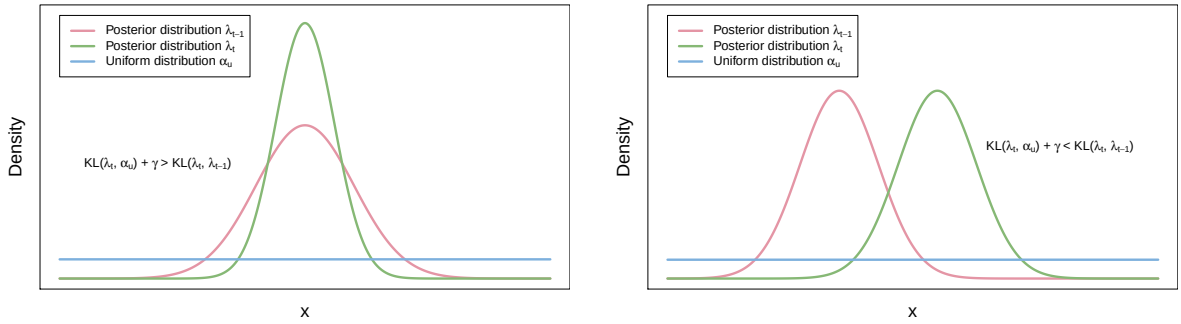


Figure 4: Example with Gaussian posterior distribution. Two possible situations: no concept drift (left) when λ_t is closer to λ_{t-1} than to α_u (in terms of KL distance); otherwise (right) there is concept drift. See Section 6.3 for details.

Note that in this case we have a plain gradient not a natural gradient wrt σ_p^2. Also note that there is no closed-form solution for the stationary point of σ_p^2. To optimize along this direction, we use a simple gradient ascent approach with backtracking line-search to set the learning rate.

6.3 Towards a Measure of Concept Drift

Observe that the form of the natural gradient of $\boldsymbol{\omega}_t$ given in Lemma 2 has an intuitive semantic interpretation in terms of measure of concept drift. If we follow a coordinate ascent algorithm, at every iteration we should set

$$\boldsymbol{\omega}_t = KL\left(q(\boldsymbol{\beta}_t|\boldsymbol{\lambda}_t) \,||\, p_u(\boldsymbol{\beta}_t)\right) - KL\left(q(\boldsymbol{\beta}_t|\boldsymbol{\lambda}_t) \,||\, p_\delta(\boldsymbol{\beta}_t|\boldsymbol{\lambda}_{t-1})\right) + \gamma \tag{31}$$

Specifically, using the constant γ as a threshold, we see that if the uniform prior $p_u(\boldsymbol{\beta}_t)$ is closer to the the variational posterior at time t, in terms of KL distance, than the variational posterior at the previous time step (i.e. $KL\left(q(\boldsymbol{\beta}_t|\boldsymbol{\lambda}_t) \,||\, p_u(\boldsymbol{\beta}_t)\right) + \gamma < KL\left(q(\boldsymbol{\beta}_t|\boldsymbol{\lambda}_t) \,||\, p_\delta(\boldsymbol{\beta}_t|\boldsymbol{\lambda}_{t-1})\right)$), then we will get a negative value for $\boldsymbol{\omega}_t$.

This in turn implies that $\mathbb{E}_q[\rho] < 0.5$, according to Equation (25) and Equation (26), (plotted in Figure 5), which means that we have a higher degree of forgetting for past data. If $\boldsymbol{\omega}_t > 0$ then $\mathbb{E}_q[\rho] > 0.5$, and less past data is forgotten. Figure 5 (left) graphically illustrates this trade-off. And Figure 4 shows a particular example of this situation using Gaussian posterior distributions.

The difference between the use of Truncated Normal over a Truncated Exponential is that with the former the relation between the $\boldsymbol{\omega}_t$ value and the $\mathbb{E}_q[\rho_t]$ value can tuned by a change in the precision of the truncated normal prior, as it is graphically illustrated in Figure 5 (right). By using a prior with a higher precision, we impose a stronger belief about that ρ_t values are neither close to 1 nor 0. In this way, the approach has the possibility to enforce smooth drift regimes.

6.4 The Multiple Hierarchical Power Prior Model

In this section we propose a modification of our HPP model to deal with complex concept drift patterns which involve only a part of the model. For example, let us consider the application of a LDA model for tracking over time the evolution of the topics in a text corpora. Under these settings, a drift could eventually affect only a subset of the topics. Using our current approach we might detect this drift and forget past of the data to adapt to the new situation and learn the new topics.

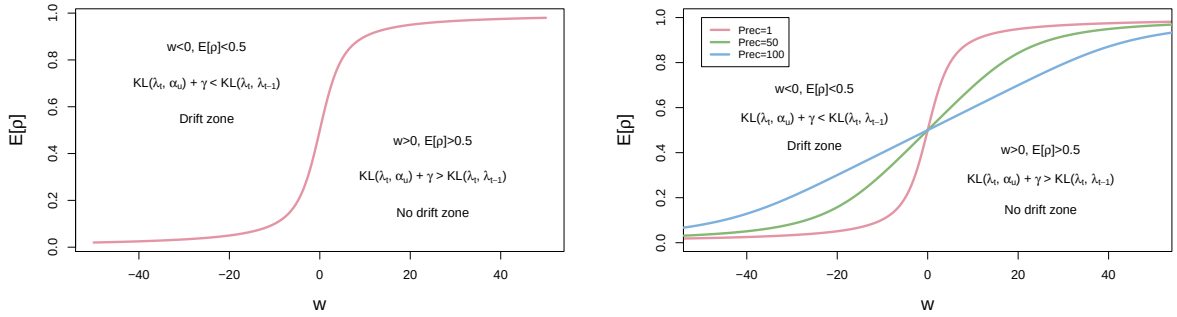


Figure 5: Relationship between $\boldsymbol{\omega}_t$ and $\mathbb{E}_q[\rho_t]$ according to Equation (25) (left) and Equation (26) (right). See Section 6.3 for details.

However, if some topics have not changed we are losing information that we could provide better estimations for these topics.

We propose an immediate extension of HPP which include multiple power priors $\rho_t^{(i)}$, one for each global parameter β_i. In this model the $\rho_t^{(i)}$'s are pair-wise independent. The latter ensures that optimizing the $\hat{\mathcal{L}}$ can be performed as above, since the variational distribution for each $\rho_t^{(i)}$ can be updated independently of the other variational distributions over $\rho_t^{(j)}$, for $j \neq i$. This extended model allows local model substructures to have different forgetting mechanisms, thereby extending the expressivity of the model. We shall refer to this extended model as a *multiple hierarchical power prior* (MHPP) model.

7. Experiments

7.1 Experimental Set-up

In this section we will evaluate the following methods:

1. Streaming variational Bayes (**SVB**) as described in Section 2.3.

2. Four versions of Population Variational Bayes (**PVB**)[4]: Population-size M equal a fixed value ($M = 1\,000$ in Section 7.2 and $M = 10\,000$ (or $1\,000$ for LDA) in Section 7.3). Learning-rate $\nu = 0.1$ or $\nu = 0.01$. Mini-batch size was set 1000 (100 for LDA).

3. Two versions of the SVB method with power priors (**SVB-PP**) or fixed exponential forgetting (as described in Section 5.1) with: $\rho = 0.9$ or $\rho = 0.99$.

4. Three version of our method based on the SVB method with adaptive exponential forgetting using hierarchical power priors (as described in Section 6):

   - **SVB-HPP-Exp** using a single shared ρ with a Truncated Exponential distribution as prior over ρ with $\gamma = 0.1$ (i.e. close to uniform).
   - **SVB-MHPP-Exp** using separate $\rho^{(i)}$ for each parameters (as described in Section 6.4) with Truncated Exponential distributions as priors over each $\rho^{(i)}$ with $\gamma = 0.1$.

4. We do not compare with SVI, because SVI is a special case of PVB when M is equal to the total size of the stream.

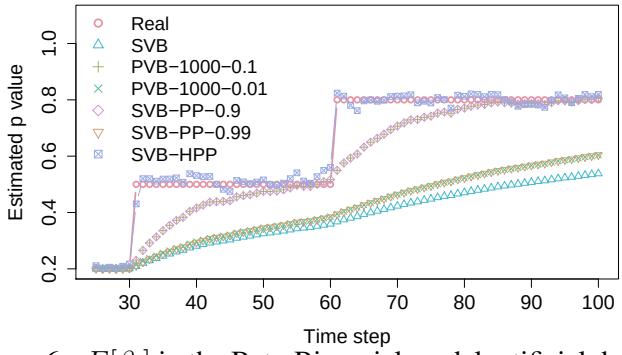


Figure 6: $E[\beta_t]$ in the Beta-Binomial model artificial data set

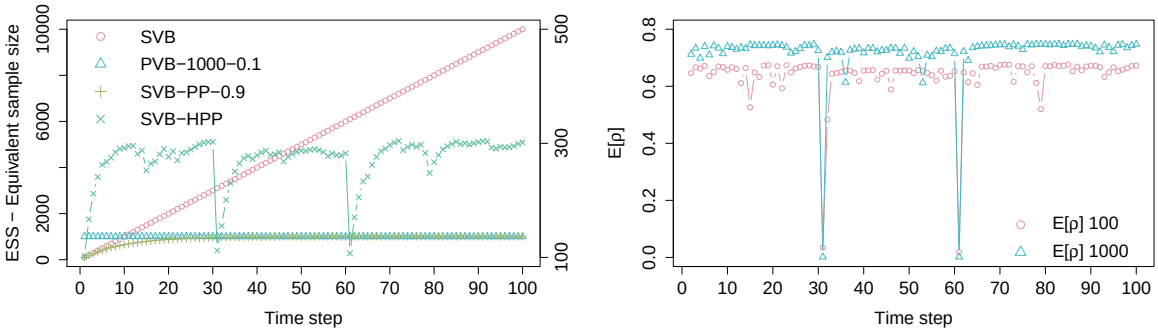


Figure 7: Results for the Beta-Binomial model artificial data set. Left figire: The equivalent sample, ESS_t, for the different methods; the values for SVB-HPP is shown on the right y-axis. Right figure: The expected values of ρ_t, $\mathbb{E}_q[\rho_t]$, for batches of size 100 and 1000, respectively.

- **SVB-MHPP-Norm** using also separate $\rho^{(i)}$ for each parameters but with Truncated Normal distributions as priors over each $\rho^{(i)}$. In this case we use $\mu_p = 0.5$ and learn the variance σ_p^2 using the empirical Bayes approach described at the end of Section 6.2.

The underlying variational engine is the VMP algorithm (Winn and Bishop, 2005) for all models; VMP was terminated after 100 iterations or if the relative increase in the lower bound fell below 0.01% (0.0001% for LDA). All priors were uninformative, using either flat Gaussians, flat Gamma priors or uniform Dirichlet priors. For the LDA model (Blei et al., 2003), we use standard priors for this model which include a Dirichlet prior over topics with $\alpha = \frac{1}{|V|}$ where $|V|$ denotes the size of the vocabulary, and another Dirichlet prior over topics assignments with $\alpha = 0.1$. Variational parameters were randomly initialized using the same seed for all methods.

7.2 Evaluation using an Artificial Data Set

First, we illustrate the behavior of the different approaches in a controlled experimental setting: We produced an artificial data stream by generating 100 samples (i.e., $|\boldsymbol{x}_t| = 100$) from a Binomial distribution at each time step. We artificially introduce concept drift by changing the parameter p of the Binomial distribution: $p = 0.2$ for the first 30 time steps, then $p = 0.5$ for the following 30 time steps, and finally $p = 0.8$ for the last 40 time steps. The data stream was modelled using a Beta-Binomial model.

Parameter Estimation: Figure 6 shows the evolution of $\mathbb{E}_q[\beta_t]$ for the different methods. We recognize that SVB simply generates a running average of the data, as it is not able to adapt to the concept drift. The results from PVB depend heavily on the learning rate ν, where the higher learning rate, which results in the more aggressive forgetting, works better in this example. Recall, though, that ν needs to be hand-tuned to achieve an optimal performance. As expected, the choice of the size of the population M for SVB does not have an impact, because the present model has no local hidden variables (cf. Section 4.1). SVB-PP produces results almost identical to PVB when ρ matches the learning rate of PVB (i.e., $\rho = 1 - \nu$). Finally, SVB-HPP provides the best results, almost mirroring the true model.

Equivalent Sample Size (ESS): Figure 7 (left) gives the evolution of the equivalent sample size, ESS_t, for the different methods [5]. The ESS of PVB is always given by the constant M. For SVB, the ESS monotonically increases as more data is seen, while SVB-PP exhibits convergence to the limiting value computed in Equation (11). A different behaviour is observed for SVB-HPP: It is automatically adjusted. Notice that the values for this model is to be read off the alternative y-axis. We can detect the the concept drift, by identifying where the ESS rapidly declines.

Evolution of Expected Forgetting factor: In Figure 7 (right) the series denoted "$E[\rho] - 100$" shows the evolution of $\mathbb{E}_q[\rho_t]$ for the artificial data set. Notice how the model clearly identifies abrupt concept drift at time steps $t = 30$ and $t = 60$. The series denoted "$E[\rho] - 1000$" illustrates the evolution of the parameter when we increase the batch size to 1000 samples. We recognize a more confident assessment about the absence of concept drift as more data is made available.

7.3 Evaluation using Real Data Sets

7.3.1 DATA, MODELS AND EVALUATION CRITERIA

For this evaluation we consider four real data sets from four different domains:

Electricity Market (Harries, 1999): The data set describes the electricity market of two Australian states. It contains 45312 instances of 6 attributes, including a class label comparing the change of the electricity price related to a moving average of the last 24 hours. Each instance in the data set represents 30 minutes of trading; during our analysis we created batches such that $\boldsymbol{x}_t$ contains all information associated with month t.

The data is analyzed using a Bayesian linear regression model. The binary class label is assumed to follow a Gaussian distribution in order to fit within the conjugate model class. Similarly, the marginal densities of the predictive attributes are also assumed to be Gaussian. The regression coefficients are given Gaussian prior distributions, and the variance is given a Gamma prior. Note that the overall distribution does not fall inside the conditional conjugate exponential family (Hoffman et al., 2013), hence we do not apply SVI (and PVB) in this setting.

GPS (Zheng et al., 2008, 2009, 2010): This data set contains 17 621 GPS trajectories (time-stamped x and y coordinates), totalling more than 4.5 million observations. To reduce the data-size we kept only one out of every ten measurements. We grouped the data so that $\boldsymbol{x}_t$ contains all data collected during hour t of the day, giving a total of 24 batches of this stream.

Here we employ a model with one independent Gaussian mixture model per day of the week, each mixture with 5 components. This enables us to track changes in the users' profiles across hours of the day, and also to monitor how the changes are affected by the day of the week.

5. For this model, ESS is simply computed by summing up the components of the λ_t defining the Beta posterior.

Finance (Borchani et al., 2015): The data contains monthly aggregated information about the financial profile of around $50\,000$ customers over 62 (non-consecutive) months. Three attributes were extracted per customer, in addition to a class-label telling whether or not the customer will default within the next 24 months.

We fit a naïve Bayes model to this data set, where the distribution at the leaf-nodes is 5-component mixture of Gaussians distribution. The distribution over the mixture node is shared by all the attributes, but not between the two classes of customers.

NIPS (Perrone et al., 2017): This dataset consist of the abstracts of published papers in the NIPS conference, between 1987 and 2015 (5804 documents in total). The data were pre-processed by choosing the most relevant individual terms across the whole dataset. This was done by ordering the words (11463 in total) by their importance in the dataset, using the TF-IDF metric (term frequency-inverse document frequency). The top 10 words after this filtering were *'policy','image','kernel','network','neurons','training','graph','images','matrix'* and *'tree'*. While the last 5 words in the ranking were *'ralf', 'ciated', 'havior', 'references'* and *'abstract'*. Only the top 100 words were kept, according to this criterion. In that way, we remove words that are not significant to track the concept drift in this data set. The documents were grouped by year, yielding a total of 29 batches of documents of different sizes. A LDA model (Blei et al., 2003) with ten topics was employed to analyze the vocabulary and to detect changes in the evolution of the major topics of the papers of this conference every year. Note that the temporal extension of this model involves dealing with dynamics at the Dirichlet distributions over the topics. As commented in Section 3, there have been many previous approaches trying to deal with this problem (Blei and Lafferty, 2006; Williamson et al., 2010b; Perrone et al., 2017), but none of them is applicable to general conjugate exponential family models and, in general, rely on much more complex inference schemes.

To evaluate the different methods discussed, we look at the test marginal log-likelihood (TMLL). Specifically, each data batch is randomly split in a train data set, $\boldsymbol{x}_t$, and a test data set, $\tilde{\boldsymbol{x}}_t$, containing two thirds and one third of the data batch, respectively. Then, TMLL_t is computed as $\text{TMLL}_t = \frac{1}{|\tilde{\boldsymbol{x}}_t|} \int p(\tilde{\boldsymbol{x}}_t, \boldsymbol{z}_t|\boldsymbol{\beta}_t) p(\boldsymbol{\beta}_t|\boldsymbol{x}_t) d\boldsymbol{z}_t d\boldsymbol{\beta}_t$[6].

A detailed description of all the models, including their structure and their variational families, is given at the supplementary material.

7.3.2 DISCUSSION

In this first part, we want to highlight how the basic versions of SVB-HPP and SVB-MHPP outperforms the rest of the approaches in most of the cases.

Figure 8 shows for each method the difference between its TMLL_t and that obtained by SVB (which is considered the baseline method). To improve readability, we only plot the results of the best performing method inside each group of methods. Figure 9 shows the development of $\mathbb{E}_q[\rho_t]$ over time for SVB-HPP-Exp, SVB-MHPP-Exp and SVB-MHPP-Norm. For SVB-HPP-TExp we only have one ρ_t-parameter, and its value is given by the solid line. SVB-MHPP utilizes one $\rho^{(i)}$ for each variational parameter.[7] In this case, we plot $\mathbb{E}_q[\rho_t^{(i)}]$ at each point in time to indicate the variability between the different estimates throughout the series. We also report the average of the

6. For LDA, $|\tilde{\boldsymbol{x}}_t|$ refers to the number of words in the test set, we then compute the so-called *per-word perplexity*.

7. The numbers of variational parameters are 14, 78, 33 and 10 for the Electricity, GPS, Financial and NIPS model, respectively.

$\mathbb{E}_q[\rho_t^{(i)}]$ values at every time step. Finally, we compute each method's aggregated test marginal log-likelihood measure $\sum_{t=1}^{T} \text{TMLL}_t$, and report these values in Table 1.

DATA SET	SVB	PVB				SVB-PP		SVB-HPP	SVB-MHPP	
		(1)	(2)	(3)	(4)	$\rho = 0.9$	$\rho = 0.99$	EXP	EXP	NORM
ELECTRICITY	-44.91	-51.01	-52.19	-51.11	-61.70	-43.92	-44.80	-40.05	-40.02	**-39.91**
GPS	-1.98	-2.10	-2.77	-1.97	-4.49	-1.94	-1.97	-1.97	**-1.86**	**-1.86**
FINANCE	-19.84	-22.29	-22.57	-20.40	-20.73	**-19.05**	-19.78	-19.83	-19.83	-19.82
NIPS	-4.07	-4.04*	-4.21*	-4.01	-4.12	-4.02	-4.06	-4.01	**-4.00**	**-4.00**

PVB PARAMETERS: (1) $M = 10k, \nu = 0.1$; (2) $M = 10k, \nu = 0.01$; (3) $M = |\boldsymbol{x}_t|, \nu = 0.1$; (4) $M = |\boldsymbol{x}_t|, \nu = 0.01$.
*: FOR NIPS, $M = 1k$ WAS USED IN (1) AND (2).

Table 1: Aggregated Test Marginal Log-Likelihood. See text for discussion.

For the electricity data set, we can see that the two proposed methods (SVB-HPP and SVB-MHPP) perform best. All models are comparable during the first nine months, which is a period where our models detect no or very limited concept drift (cf. top right plot of Figure 8). However, after this period, both SVB-HPP and SVB-MHPP detects substantial drift, and is able to adapt better than the other methods, which appear unable to adjust to the complex concept drift structure in the latter part of the data. SVB-HPP and SVB-MHPP continue to behave at a similar level, mainly because when drift happens it typically includes a high proportion of the parameters of the model.

For the GPS data set, we can observe how the SVB-MHPP is superior to the rest of the methods, particularly towards the end of the series. When looking at Figure 8 (middle right panel), we can see that a significative proportion of the model parameters are drifting (i.e., $\mathbb{E}_q[\rho_t^{(i)}] \leq 0.05$) at all times, while another proportion of the parameters show a quite stable behavior (ρ-values above 0.9). This complex pattern is not captured well by SVB-HPP, which ends up assuming no concept drift after the initial time-step.

The financial data set shows a different behavior. During the first months, no major differences among methods can be found. But after month 30, SVB-PP with $\rho = 0.9$ is superior. Looking at the $E[\rho_t^{(i)}]$-values of SVB-MHPP, we observe that there is significant concept drift in some of the parameters over the first few months. However, only a few parameters exhibit noteworthy drift after the first third of the sequence. Apparently, the simple SVB-PP approach has the upper hand when the drift is constant and fairly limited, at least when the optimal forgetting factor ρ has been identified.

In the case of the NIPS data set, we see again as HPP approaches captures the drift in the data. SVB-HPP hardly detects any drift in the first 20 years and performs quite similarly to SVB (i.e. relative performance close to zero). But in the last 10 years SVB-HPP clearly outperforms SVB because it detects two strong drifts at years 23 and 29. So, at these time steps the whole LDA model is almost retrained from scratch. In this case, SVB-MHPP is able to capture more fine-grained drifts in the data. Mainly, it detects changes in some topics while other topics remain constant over time. This allows SVB-MHPP to outperform SVB-HPP during some periods.

We have also observed there are no major differences between SVB-MHPP-Exp and SVB-MHPP-Norm. So, it seems that inclusion of alternative priors it is not having a big impact into the performance, at least if the variance parameter of the Truncated Normal is fixed automatically using an empirical Bayes approach. But this is something that may require further investigations.

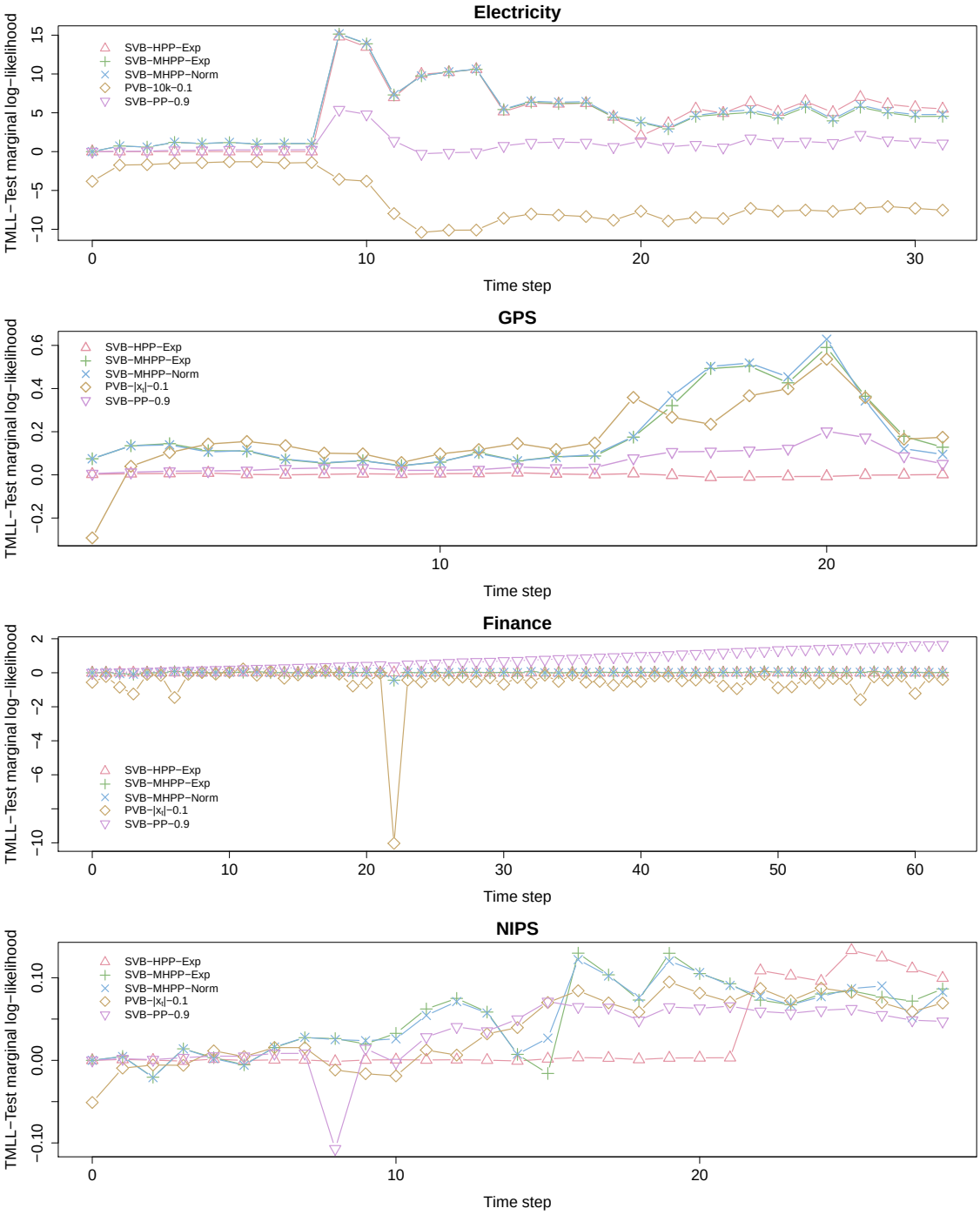


Figure 8: Results of the TMLL_t improvement over SVB for the competing methods, for the four real data sets. See text for discussion.

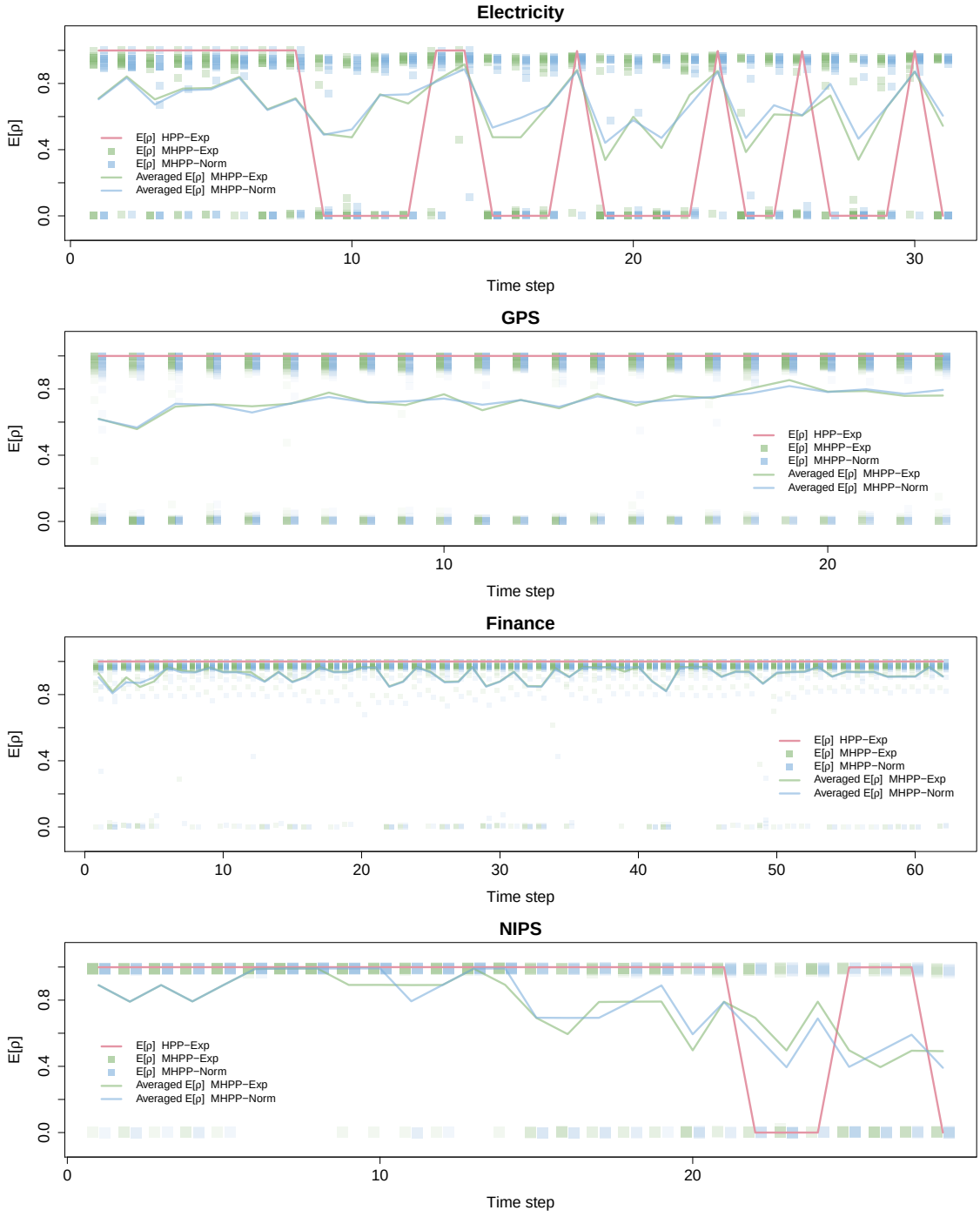


Figure 9: Evolution of $\mathbb{E}_q[\rho_t]$ for SVB-HPP and SVB-MHPP.

We conclude this section by highlighting that the performance of SVB-PP and PVB depend heavily on the hyper-parameters of the model, cf. Table 1. As an example, consider SVB-PP and how its performance varies by changing the ρ parameter. Similarly, PVB's performance is sensitive

both to ν (see in particular the results for the GPS data) and M (financial data). These hyper-parameters are hard to fix, as their optimal values depend on data characteristics (see Broderick et al. (2013); McInerney et al. (2015) for similar conclusions). We therefore believe that the fully Bayesian formulation is an important strong point of our approach.

8. Conclusions and Future Work

In this paper we have introduced a novel Bayesian approach for learning general latent variable models from non-stationary data streams. For this purpose, we introduce max-entropy transition models as a general method for transitioning the global parameters of a latent variable model. We also show that previous approaches like exponential forgetting and power priors can be seen as specific cases of this general transition model. But these approaches are only able to model slowly changing data streams. Our approach is able to handle both abrupt and gradual drifts in the data stream by explicitly modeling the rate of change of the data stream. For this purpose we introduce a novel hierarchical prior which allows to adapt to the different drifts one can encounter in a data stream. We then develop an efficient variational inference scheme that optimizes a novel lower bound of the likelihood function.

As future work we aim to provide a sound approach to semantically characterize concept drift by inspecting the $\mathbb{E}[\rho_t^{(i)}]$ values provided by SVB-MHPP. And to investigate the effects in the variational approximation introduced by the use of the double lower bound approximation.

Acknowledgments

This work was partly carried out as part of the AMIDST project. AMIDST has received funding from the European Union's Seventh Framework Programme for research, technological development and demonstration under grant agreement no 619209. Furthermore, this research has been partly funded by the Spanish Ministry of Economy and Competitiveness, through projects TIN2015-74368-JIN, TIN2013-46638-C3-1-P, TIN2016-77902-C3-3-P and by ERDF funds. AM, DRL and AS thank the support from CDTIME. DRL thanks also to CEIMAR. This work is an extended version of a previously published conference paper (Masegosa et al., 2017a).

Appendix A.

Appendix A. Proof of Theorem 1 and Lemma 2

Proof [Proof of Theorem 1] In the specification of $\mathcal{L}$ we have that $\mathbb{E}_q[\ln \hat{p}(\boldsymbol{\beta}_t|\boldsymbol{\lambda}_{t-1}, \rho_t)]$ (defined in Equation (7)) can be expanded as (ignoring the base measure) :

$$\mathbb{E}_q[(\rho_t\boldsymbol{\lambda}_{t-1} + (1-\rho_t)\boldsymbol{\alpha}_u)\boldsymbol{t}(\boldsymbol{\beta}_t) - a_g(\rho_t\boldsymbol{\lambda}_{t-1} + (1-\rho_t)\boldsymbol{\alpha}_u)].$$

Since a_g is convex we have

$$a_g(\rho_t\boldsymbol{\lambda}_{t-1} + (1-\rho_t)\boldsymbol{\alpha}_u) \leq \rho_t a_g(\boldsymbol{\lambda}_{t-1}) + (1-\rho_t)a_g(\boldsymbol{\alpha}_u),$$

which combined with Equation (10) gives

$$\begin{aligned}&\mathbb{E}_q[\ln p(\boldsymbol{x}_t, \boldsymbol{Z}_t|\boldsymbol{\beta}_t)] + \mathbb{E}_q[(\rho_t\boldsymbol{\lambda}_{t-1} + (1-\rho_t)\boldsymbol{\alpha}_u)\boldsymbol{t}(\boldsymbol{\beta}_t)\\ &- \rho_t a_g(\boldsymbol{\lambda}_{t-1}) - (1-\rho_t)a_g(\boldsymbol{\alpha}_u)] + \mathbb{E}_q[p(\rho_t|\boldsymbol{\gamma})]\\ &- \mathbb{E}_q[\ln q(\boldsymbol{Z}_t|\boldsymbol{\phi}_t)] - \mathbb{E}_q[q(\boldsymbol{\beta}_t|\boldsymbol{\lambda}_t)] - \mathbb{E}_q[q(\rho_t|\boldsymbol{\omega}_t)] \leq \mathcal{L}.\end{aligned}$$

Lastly, by exploiting the mean field factorization of q and using the exponential family form of $p_\delta(\boldsymbol{\beta}_t|\boldsymbol{\lambda}_{t-1})$ and $p_u(\boldsymbol{\beta}_t)$ we get the desired result. ■

Proof [Proof of Lemma 2] Firstly, by ignoring the terms in $\hat{\mathcal{L}}$ (Equation (28)) that do not involve $\boldsymbol{\omega}_t$ we get

$$\begin{aligned}\hat{\mathcal{L}}(\boldsymbol{\omega}_t) &= \mathbb{E}_q[\rho_t](\mathbb{E}_q[\ln(p_\delta(\boldsymbol{\beta}_t|\boldsymbol{\lambda}_{t-1})) - \mathbb{E}_q[\ln p_u(\boldsymbol{\beta}_t)]) + \mathbb{E}_q[p(\rho_t|\boldsymbol{\gamma})] - \mathbb{E}_q[q(\rho_t|\boldsymbol{\omega}_t)]\\ &= \mathbb{E}_q[\rho_t](\mathbb{E}_q[\ln(p_\delta(\boldsymbol{\beta}_t|\boldsymbol{\lambda}_{t-1})) - \mathbb{E}_q[\ln p_u(\boldsymbol{\beta}_t)]) + \boldsymbol{\gamma}^T\mathbb{E}_q[t[\rho_t]] - (\boldsymbol{\omega}_t^T\mathbb{E}_q[t[\rho_t]] - a_g(\boldsymbol{\omega}_t)) + \textit{cte}\end{aligned}$$

As we have assumed that the sufficient statistics function $\boldsymbol{t}(\rho_t)$ for $p(\rho_t|\boldsymbol{\gamma})$ and $q(\boldsymbol{\beta}_t|\boldsymbol{\lambda}_t)$ contains the identity function ($\boldsymbol{t}_1(\rho_t) = \rho_t$) we have

$$\hat{\mathcal{L}}(\boldsymbol{\omega}_t) = \begin{pmatrix}\mathbb{E}_q[\rho_t]\\ \mathbb{E}_q[t_{\neq 1}(\rho_t)]\end{pmatrix}^T \begin{pmatrix}(\mathbb{E}_q[\ln(p_\delta(\boldsymbol{\beta}_t|\boldsymbol{\lambda}_{t-1})) - \mathbb{E}_q[\ln p_u(\boldsymbol{\beta}_t)]) + \boldsymbol{\gamma}_1 - \boldsymbol{\omega}_1\\ \boldsymbol{\gamma}_{\neq 1} - \boldsymbol{\omega}_{\neq 1}\end{pmatrix} - a_g(\boldsymbol{\omega}_t) + \textit{cte}$$

where the sub-index $\neq 1$ refers to those sub-indexes different from 1.

Using the standard equality of exponential family distributions, $\mathbb{E}_q[\boldsymbol{t}(\rho_t)] = \nabla_{\boldsymbol{\omega}_t} a_g(\boldsymbol{\omega}_t)$, we have

$$\nabla_{\omega_t}\hat{\mathcal{L}} = \nabla^2_{\omega_t} a_g(\boldsymbol{\omega}_t) \begin{pmatrix}\mathbb{E}_q[\ln(p_\delta(\boldsymbol{\beta}_t|\boldsymbol{\lambda}_{t-1})) - \ln p_u(\boldsymbol{\beta}_t)] + \boldsymbol{\gamma}_1 - \boldsymbol{\omega}_{t,1}\\ \boldsymbol{\gamma}_{\neq 1} - \boldsymbol{\omega}_{t,\neq 1}\end{pmatrix}$$

We can now find the natural gradient by premultiplying $\nabla_{\omega_t}\hat{\mathcal{L}}$ by the inverse of the Fisher information matrix, which for the exponential family corresponds to the inverse of the Hessian of the log-normalizer:

$$\begin{aligned}\hat{\nabla}_{\omega_t}\hat{\mathcal{L}} &= (\nabla^2_{\omega_t} a_g(\boldsymbol{\omega}_t))^{-1}\nabla_{\omega_t}\hat{\mathcal{L}}\\ &= \begin{pmatrix}\mathbb{E}_q[\ln(p_\delta(\boldsymbol{\beta}_t|\boldsymbol{\lambda}_{t-1})) - \ln p_u(\boldsymbol{\beta}_t)] + \boldsymbol{\gamma}_1 - \boldsymbol{\omega}_{t,1}\\ \boldsymbol{\gamma}_{\neq 1} - \boldsymbol{\omega}_{t,\neq 1}\end{pmatrix}\end{aligned}$$

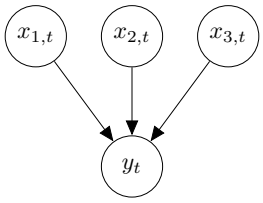
Lastly, by introducing $q(\boldsymbol{\beta}_t|\boldsymbol{\lambda}_t) - q(\boldsymbol{\beta}_t|\boldsymbol{\lambda}_t)$ inside the expectation we get the difference in Kullbach-Leibler divergence $KL\left(q(\boldsymbol{\beta}_t|\boldsymbol{\lambda}_t)\,||\,p_u(\boldsymbol{\beta}_t)\right) - KL\left(q(\boldsymbol{\beta}_t|\boldsymbol{\lambda}_t)\,||\,p_\delta(\boldsymbol{\beta}_t|\boldsymbol{\lambda}_{t-1})\right)$. ■

Appendix B. Experimental Evaluation

B.1 Probabilistic Models

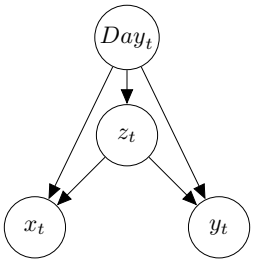
We provide a (simplified) graphical description of the probabilistic models used in the experiments. We also detail the distributional assumptions of the parameters, which are then used to define the variational approximation family.

ELECTRICITY MODEL



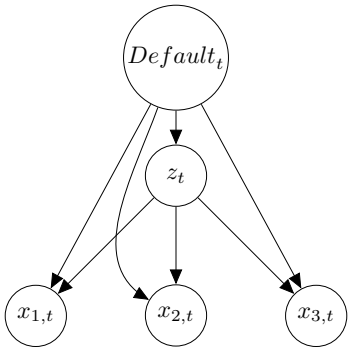
$$
\begin{aligned}
(\mu_i, \gamma_i) &\sim NormalGamma(1, 1, 0, 1e-10) \\
\gamma &\sim Gamma(1, 1) \\
b_i &\sim \mathcal{N}(0, +\infty) \\
x_{i,t} &\sim \mathcal{N}(\mu_i, \gamma_i) \\
y_t &\sim \mathcal{N}\left(b_0 + \sum_i b_i x_{i,t}, \gamma\right)
\end{aligned}
$$

GPS MODEL



$$
\begin{aligned}
p &\sim Dirichlet(1, \ldots, 1) \\
p_k &\sim Dirichlet(1, \ldots, 1) \\
(\mu^{(x)}_{j,k}, \gamma^{(x)}_{j,k}) &\sim NormalGamma(1, 1, 0, 1e-10) \\
(\mu^{(y)}_{j,k}, \gamma^{(y)}_{j,k}) &\sim NormalGamma(1, 1, 0, 1e-10) \\
Day_t &\sim Multinomial(p) \\
(z_t | Day_t = k) &\sim Multinomial(p_k) \\
(x_t | z_t = j, Day_t = k) &\sim \mathcal{N}(\mu^{(x)}_{j,k}, \gamma^{(x)}_{j,k}) \\
(y_t | z_t = j, Day_t = k) &\sim \mathcal{N}(\mu^{(y)}_{j,k}, \gamma^{(y)}_{j,k})
\end{aligned}
$$

FINANCIAL MODEL



$$
\begin{aligned}
p &\sim Dirichlet(1, \ldots, 1) \\
p_k &\sim Dirichlet(1, \ldots, 1) \\
(\mu_{i;j,k}, \gamma_{i;j,k}) &\sim NormalGamma(1, 1, 0, 1e-10) \\
Default_t &\sim Binomial(p) \\
(z_t | Default_t = k) &\sim Multinomial(p_k) \\
(x_{i,t} | z_t = j, Day_t = k) &\sim \mathcal{N}(\mu_{i;j,k}, \gamma_{i;j,k})
\end{aligned}
$$

References

Charu C Aggarwal. *Data streams: models and algorithms*, volume 31. Springer Science & Business Media, 2007.

Charu C Aggarwal. *Managing and mining sensor data*. Springer Science & Business Media, 2013.

Ole Barndorff-Nielsen. *Information and exponential families: in statistical theory*. John Wiley & Sons, 2014.

José M Bernardo and Adrian FM Smith. *Bayesian Theory*, volume 405. John Wiley & Sons, 2009.

Christopher M Bishop. Latent variable models. In *Learning in graphical models*, pages 371–403. Springer, 1998.

Christopher M Bishop. *Pattern recognition and machine learning*. springer, 2006.

David M Blei. Build, compute, critique, repeat: Data analysis with latent variable models. *Annual Review of Statistics and Its Application*, 1:203–232, 2014.

David M Blei and John D Lafferty. Dynamic topic models. In *Proceedings of the 23rd international conference on Machine learning*, pages 113–120. ACM, 2006.

David M Blei, Andrew Y Ng, and Michael I Jordan. Latent Dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022, 2003.

Hanen Borchani, Ana M Martínez, Andrés R Masegosa, Helge Langseth, Thomas D Nielsen, Antonio Salmerón, Antonio Fernández, Anders L Madsen, and Ramón Sáez. Modeling concept drift: A probabilistic graphical model based approach. In *International Symposium on Intelligent Data Analysis*, pages 72–83. Springer, 2015.

Tamara Broderick, Nicholas Boy, Andre Wibisono, Ashia C. Wilson, and Michael I. Jordan. Streaming variational Bayes. In *Advances in Neural Information Processing Systems 26*, pages 1727–1735. Curran Associates, Inc., 2013.

Mohamed Medhat Gaber, Arkady Zaslavsky, and Shonali Krishnaswamy. Mining data streams: a review. *ACM Sigmod Record*, 34(2):18–26, 2005.

João Gama, Indrė Žliobaitė, Albert Bifet, Mykola Pechenizkiy, and Abdelhamid Bouchachia. A survey on concept drift adaptation. *ACM Computing Surveys*, 46(4):44:1–44:37, 2014.

João Gama and Pedro Pereira Rodrigues. An overview on mining data streams. In *Foundations of Computational, IntelligenceVolume 6*, pages 29–45. Springer, 2009.

Michael Harries. Splice-2 comparative evaluation: Electricity pricing. NSW-CSE-TR-9905, School of Computer Siene and Engineering, The University of New South Wales, 1999.

Leonard Hasenclever, Stefan Webb, Thibaut Lienart, Sebastian Vollmer, Balaji Lakshminarayanan, Charles Blundell, and Yee Whye Teh. Distributed Bayesian learning with stochastic natural gradient expectation propagation and the posterior server. *Journal of Machine Learning Research*, 18(106):1–37, 2017.

David Heckerman, Dan Geiger, and David M Chickering. Learning Bayesian networks: The combination of knowledge and statistical data. *Machine learning*, 20(3):197–243, 1995.

Matthew D. Hoffman, David M. Blei, Chong Wang, and John Paisley. Stochastic variational inference. *Journal of Machine Learning Research*, 14:1303–1347, 2013.

Antti Honkela and Harri Valpola. On-line variational Bayesian learning. In *4th International Symposium on Independent Component Analysis and Blind Signal Separation*, pages 803–808, 2003.

Joseph G Ibrahim and Ming-Hui Chen. Power prior distributions for regression models. *Statistical Science*, pages 46–60, 2000.

Joseph G Ibrahim, Ming-Hui Chen, and Debajyoti Sinha. On optimality properties of the power prior. *Journal of the American Statistical Association*, 98(461):204–213, 2003.

Miroslav Kárnỳ. Approximate Bayesian recursive estimation. *Information Sciences*, 285:100–111, 2014.

Daphne Koller and Nir Friedman. *Probabilistic graphical models: principles and techniques*. MIT press, 2009.

Steffen L Lauritzen. Propagation of probabilities, means, and variances in mixed graphical association models. *Journal of the American Statistical Association*, 87(420):1098–1108, 1992.

Andrés Masegosa, Thomas D Nielsen, Helge Langseth, Darío Ramos-López, Antonio Salmerón, and Anders L Madsen. Bayesian models of data streams with hierarchical power priors. In *International Conference on Machine Learning*, pages 2334–2343, 2017a.

Andrés R Masegosa, Ana M Martinez, Helge Langseth, Thomas D Nielsen, Antonio Salmerón, Darío Ramos-López, and Anders L Madsen. Scaling up Bayesian variational inference using distributed computing clusters. *International Journal of Approximate Reasoning*, 88:435–451, 2017b.

Andrés R Masegosa, Ana M Martínez, Darío Ramos-López, Rafael Cabañas, Antonio Salmerón, Thomas D Nielsen, Helge Langseth, and Anders L Madsen. Amidst: a Java toolbox for scalable probabilistic machine learning. *arXiv preprint arXiv:1704.01427*, 2017c.

James McInerney, Rajesh Ranganath, and David Blei. The population posterior and Bayesian modeling on streams. In *Advances in Neural Information Processing Systems 28*, pages 1153–1161. Curran Associates, Inc., 2015.

Stanislav Minsker, Sanvesh Srivastava, Lizhen Lin, and David B. Dunson. Robust and scalable Bayes via a median of subset posterior measures. *Journal of Machine Learning Research*, 18 (124):1–40, 2017. URL `http://jmlr.org/papers/v18/16-655.html`.

Kevin P Murphy. *Machine learning: a probabilistic perspective*. MIT press, 2012.

Kristian G Olesen, Steffen L Lauritzen, and Finn V Jensen. aHUGIN: A system creating adaptive causal probabilistic networks. In *Proceedings of the Eighth international conference on Uncertainty in artificial intelligence*, pages 223–229. Morgan Kaufmann Publishers Inc., 1992.

E. Ozkan, V. Smidl, S. Saha, C. Lundquist, and F. Gustafsson. Marginalized adaptive particle filtering for nonlinear models with unknown time-varying noise parameters. *Automatica*, 49(6): 1566–1575, 2013. ISSN 0005-1098.

Spiros Papadimitriou, Jimeng Sun, and Christos Faloutsos. Streaming pattern discovery in multiple time-series. In *Proceedings of the 31st international conference on Very large data bases*, pages 697–708. VLDB Endowment, 2005.

Valerio Perrone, Paul A Jenkins, Dario Spano, and Yee Whye Teh. Poisson random fields for dynamic feature models. *Journal of Machine Learning Research*, 18(127):1–45, 2017.

Masa-Aki Sato. Online model selection based on the variational Bayes. *Neural Computation*, 13 (7):1649–1681, 2001.

Tianlin Shi and Jun Zhu. Online bayesian passive-aggressive learning. In *Icml*, pages 378–386, 2014.

Michael E Tipping and Christopher M Bishop. Probabilistic principal component analysis. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 61(3):611–622, 1999.

Martin J Wainwright, Michael I Jordan, et al. Graphical models, exponential families, and variational inference. *Foundations and Trends® in Machine Learning*, 1(1–2):1–305, 2008.

Sinead Williamson, Peter Orbanz, and Zoubin Ghahramani. Dependent Indian buffet processes. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 924–931, 2010a.

Sinead Williamson, Chong Wang, Katherine Heller, and David Blei. The IBP compound Dirichlet process and its application to focused topic modeling. In *Proceedings of ICML*, 2010b.

John M. Winn and Christopher M. Bishop. Variational message passing. *Journal of Machine Learning Research*, 6:661–694, 2005.

Yu Zheng, Quannan Li, Yukun Chen, Xing Xie, and Wei-Ying Ma. Understanding mobility based on gps data. In *Proceedings of the 10th International Conference on Ubiquitous Computing*, UbiComp '08, pages 312–321, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-136-1. doi: 10.1145/1409635.1409677.

Yu Zheng, Lizhu Zhang, Xing Xie, and Wei-Ying Ma. Mining interesting locations and travel sequences from gps trajectories. In *Proceedings of the 18th International Conference on World Wide Web*, WWW '09, pages 791–800, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-487-4. doi: 10.1145/1526709.1526816.

Yu Zheng, Xing Xie, and Wei-Ying Ma. Geolife: A collaborative social networking service among user, location and trajectory. *IEEE Data Eng. Bull.*, 33(2):32–39, 2010.