Variational Auto-Regressive Gaussian Processes for Continual Learning

Sanyam Kapoor [1]   **Theofanis Karaletsos** [2]   **Thang D. Bui** [3]

Abstract

Through sequential construction of posteriors on observing data online, Bayes' theorem provides a natural framework for continual learning. We develop *Variational Auto-Regressive Gaussian Processes* (VAR-GPs), a principled posterior updating mechanism to solve sequential tasks in continual learning. By relying on sparse inducing point approximations for scalable posteriors, we propose a novel auto-regressive variational distribution which reveals two fruitful connections to existing results in Bayesian inference, expectation propagation and orthogonal inducing points. Mean predictive entropy estimates show VAR-GPs prevent catastrophic forgetting, which is empirically supported by strong performance on modern continual learning benchmarks against competitive baselines. A thorough ablation study demonstrates the efficacy of our modeling choices.

1. Introduction

Continual Learning (CL) is the constant development of complex behaviors by building upon previously acquired skills (Ring, 1994; Thrun, 1998); humans and other animals exhibit knowledge acquisition for continual skill development (Hoppitt & Laland, 2013). To the contrary, modern artificial intelligence methods based on supervised machine learning rely on a stronger assumption of all representative information being available at once, i.e. *i.i.d* data. Many systems, however, violate this assumption. A hospital may not have legal access to past patient data to provide automated diagnosis, real-time learning systems may be limited by compute to utilize all available data, and a mobile device may prefer on-device learning and inference to guarantee user privacy. In each of these examples, the learning algo-
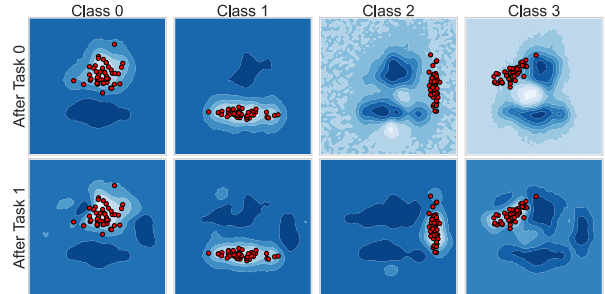
(a) VAR-GP (ours)
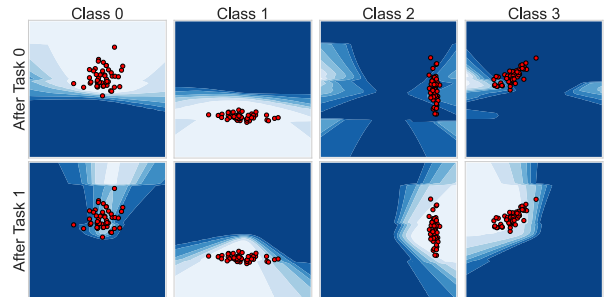
(b) VCL (with coreset size 10)

Figure 1. By first, only training on classes $0/1$ (**Task 0**), and next, only training on classes $2/3$ (**Task 1**), we show the posterior predictive density surface (brighter is higher) for a four-way classifier on the synthetic dataset (marked ●) from Figure 2. (a) VAR-GPs preserve information from Task 0 even after training on Task 1, i.e. prevent *catastrophic forgetting* (seen as bright regions around data from Task 0). (b) VCL is overconfident in its predictions and is structurally less stable (seen as large variations in predictive probabilities between Task 0 and Task 1).

rithm does not observe *i.i.d* data, but only new parts of the data space. We still want to improve our models using new data without compromising existing performance.

Conventional training methods, however, tend to perform poorly when it comes to balancing *rigidity*: the inability to adapt to new experience, and *plasticity*: the tendency to forget past experience. For example, neural network training is known to be vulnerable to *catastrophic forgetting*, causing any data distribution shift to override past learning (McCloskey & Cohen, 1989; Ratcliff, 1990). Many approaches have been pursued to address these issues, and among them, the Bayesian framework is arguably the most

general and principled. By allowing us to sequentially construct posteriors after observing data online, Bayes' theorem provides a coherent framework to build continual learning algorithms, and avoid pitfalls like catastrophic forgetting; this principle is seen in Sato (2001) for online model selection, and in Broderick et al. (2013) for inference on streaming data. More recently, much literature has been devoted to scalable posterior updating schemes with neural networks — regularization using a Laplace approximation with a diagonal Fisher information matrix (Kirkpatrick et al., 2017), using approximate path integral of gradient vector fields (Zenke et al., 2017), using recursive variational posterior approximations (Nguyen et al., 2018; Swaroop et al., 2019; Ahn et al., 2019), targeting adaptive capacity in Bayesian neural networks (Kessler et al., 2019), and using episodic memory (Lopez-Paz & Ranzato, 2017; Rebuffi et al., 2017). Alternatively, the effectiveness of Gaussian Process (GP) priors is demonstrated by Csató & Opper (2002) for online regression, and by Bui et al. (2017a) for streaming data. Despite favorable properties, GPs remain under-explored for modern continual learning tasks.

We take a step in this direction and propose *Variational Auto-Regressive Gaussian Processes* (VAR-GPs) for continual learning. By utilizing scalable sparse approximations for Gaussian processes (Titsias, 2009; Hensman et al., 2013; 2015) and advances in variational inference (Hoffman et al., 2013), we make the following key contributions — i) a generalized continual variational lower bound for sequential datasets with a natural interpretation, ii) a novel auto-regressive variational distribution for continual learning with GPs, iii) two fruitful connections to results in Bayesian inference, Expectation Propagation (EP) and orthogonal inducing point approximation, and iv) evidence on the effectiveness of hyper-priors.

The rest of the paper is organized as follows — Section 2 puts prior work in context and highlights the key similarities or differences. Section 3 summarizes the relevant background needed to develop VAR-GPs. Section 4 derives VAR-GPs for continual learning and describes connections to two existing results in Bayesian inference. Section 5 develops an intuitive characterization of how VAR-GPs avoid catastrophic forgetting, supported by empirical results and thorough ablation studies. Finally, Section 6 concludes with a discussion on limitations and future directions.

2. Related Work

Our work aligns with the key desiderata for continual learning outlined by Farquhar & Gal (2018) and van de Ven & Tolias (2019). In particular, unlike prior work (Kirkpatrick et al., 2017; Zenke et al., 2017; Nguyen et al., 2018; Swaroop et al., 2019) where task identity is revealed by

a multi-head architecture, we tackle a considerably harder and more realistic continual learning setup where task identity is unknown and rely on approximate Bayesian inference for all learning. While the broader literature focuses on many aspects of CL like vanilla transfer learning (Li & Hoiem, 2017), adaptive model capacity, and episodic memory, we emphasize on developing an automated posterior updating mechanism for continual learning.

Continual learning with neural networks relies on regularization of the parameters through the previous approximate posterior (or the prior in absence of learning) - *Elastic Weight Consolidation* (EWC) (Kirkpatrick et al., 2017) uses Laplace approximations with a diagonal Fisher Matrix, *Synaptic Intelligence* (SI) (Zenke et al., 2017) uses approximate path integral of gradient vector fields, and *RWalk* (Chaudhry et al., 2018) unifies these methods from a KL-divergence perspective. Ritter et al. (2018) use Kronecker-factored approximated Laplace approximation for tractability instead. By directly estimating a recursive approximate variational posterior, *Improved Variational Continual Learning* (VCL) (Nguyen et al., 2018; Swaroop et al., 2019) proves to be one of the most competitive methods, and forms our main baseline. Although, in principle VCL relies on regularizing only w.r.t the last constructed posterior, in practice episodic memory (Swaroop et al., 2019; Shin et al., 2017; Rebuffi et al., 2017) of real samples is used to limit catastrophic forgetting. We avoid this ad-hoc data subset selection problem by naturally incorporating past tasks into our learning objective.

Model capacity has been investigated by *Progressive Neural Networks* (Rusu et al., 2016; Schwarz et al., 2018), which adapt by rewiring neural networks. Kessler et al. (2019) provide a fully Bayesian treatment through Indian Buffet Process priors. While, assessing model capacity required for both batch and continual learning remains an open problem, including in GP-based models, we emphasize only on an automated inference scheme in this work.

Csató & Opper (2002); Csató (2002) are seminal contributions towards utilizing Gaussian processes (GPs) in online regression. In the spirit of streaming variational Bayes framework (Broderick et al., 2013), Bui et al. (2017a) demonstrate the effectiveness of GPs for a single-task online regression problems. Moreno-Muñoz et al. (2019) adapt recursive construction from VCL (Nguyen et al., 2018) to multi-task GPs. To the best of our knowledge, no prior work addresses modern continual learning benchmarks and desiderata with GPs alone. More recently, FCRL (Titsias et al., 2020) applies functional regularization to neural networks using sparse GPs, unifying memory-based (Lopez-Paz & Ranzato, 2017; Rebuffi et al., 2017) and Bayesian methods through inducing points selection. Unlike FCRL, our proposed approach naturally

models cross-task covariances through structured approximations, and allows joint optimization of both the variational approximation and inducing inputs using a unified objective.

3. Background

Exact Gaussian processes We assume basic familiarity with Gaussian processes (GPs) (Rasmussen & Williams, 2005). By expressing priors over the function space, GPs provide a flexible non-parametric framework to perform probabilistic inference. For a dataset $\mathcal{D}$ of size N, we model the relationship between a collection of inputs $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^N$ and targets $\mathbf{y} = \{y_i\}_{i=1}^N$ using a Gaussian Process prior $f \mid \theta \sim \mathcal{GP}\left(\mu_\theta(\cdot), k_\theta(\cdot, \cdot)\right)$ and a likelihood model $p(\mathbf{y} \mid f(\mathbf{X}))$. The prior is fully defined by a mean function μ_θ and a covariance function k_θ, where θ is a small set of hyperparameters. Often, we will denote the covariance matrix as $\mathbf{K}_{\mathbf{X},\mathbf{Z}}$ to be explicit about the two sets of inputs $\mathbf{X}$ and $\mathbf{Z}$ that generate the matrix, keeping dependence on θ implicit. As typical in literature, we set the mean function to zero.

The intertwined goals of exact inference are to infer the posterior distribution $p(f \mid \mathcal{D}, \theta)$ and obtain the marginal likelihood $p(\mathbf{y} \mid \mathbf{X}, \theta) = \int p(\mathbf{y} \mid f(\mathbf{X})) p(f \mid \theta) df$. The posterior distribution can be used to obtain the predictive distribution for a novel input $\mathbf{x}_\star$, as $p(\mathbf{y}_\star \mid \mathbf{x}_\star, \mathcal{D}, \theta) = \int p(\mathbf{y}_\star \mid f(\mathbf{x}_\star)) p(f \mid \mathcal{D}, \theta) df$, while the marginal likelihood can be used for model selection. For instance, a Gaussian likelihood model with a diagonal covariance $p(\mathbf{y} \mid f, \mathbf{X}) = \mathcal{N}(\mathbf{y}; f(\mathbf{X}), \sigma_y^2 \mathbf{I})$ leads to closed-form expressions for predictive distribution, which is a Gaussian defined by the following mean and variance,

$$\mu_\star = \mathbf{K}_\star^\top (\mathbf{K} + \sigma_y^2 \mathbf{I})^{-1} \mathbf{y} , \tag{1}$$

$$\sigma_\star^2 = \mathbf{K}_{\star\star} - \mathbf{K}_\star^\top (\mathbf{K} + \sigma_y^2 \mathbf{I})^{-1} \mathbf{K}_\star , \tag{2}$$

where $\mathbf{K}_{\star\star} = k_\theta(\mathbf{x}_\star, \mathbf{x}_\star)$, and the covariance matrices are given by $(\mathbf{K}_\star)_i = k_\theta(\mathbf{x}_i, \mathbf{x}_\star)$, and $(\mathbf{K})_{ij} = k_\theta(\mathbf{x}_i, \mathbf{x}_j)$. This model can be extended for K-way classification by building K independent functions and using a *softmax* likelihood function such that class probabilities are given by $p(y = k \mid f_1, \ldots, f_K, \mathbf{x}) \propto \exp\{f_k(\mathbf{x})\}$.

Unfortunately, exact inference is analytically and computationally intractable for many models and datasets of interest. Even for the closed-form solution above for GP regression, the computational complexity is $\mathcal{O}\left(N^3\right)$ for inference & learning, and $\mathcal{O}\left(N^2\right)$ for every subsequent prediction. We will rely on sparse variational approximations to sidestep both intractabilities.

Sparse Variational Gaussian processes Among various sparse approximations of Gaussian processes (Quiñonero-

Candela & Rasmussen, 2005; Bui et al., 2017b), our continual learning algorithm builds on inducing-point methods (Titsias, 2009; Hensman et al., 2013; 2015). Following Hensman et al. (2015), we introduce M *inducing outputs* $\mathbf{u} = \{u_i\}_{i=1}^M$, and corresponding *inducing inputs* $\mathbf{Z} = \{\mathbf{z}_i\}_{i=1}^M$. We view $\mathbf{u}, \mathbf{Z}$ not simply as a collection of points, but $\mathbf{u}$ as values of a continuous function f evaluated at $\mathbf{Z}$, i.e. $\mathbf{u} = f(\mathbf{Z})$. By noting that the underlying function can be decomposed as $f = \{f_{\neq \mathbf{u}}, \mathbf{u}\}$, the joint distribution can be equivalently written as,

$$\begin{aligned} p(\mathbf{y}, f \mid \mathbf{X}, \theta) = {}& p(\mathbf{y} \mid f, \mathbf{X}) \\ & p(f_{\neq \mathbf{u}} \mid \mathbf{u}, \theta) p(\mathbf{u} \mid \mathbf{Z}, \theta) . \end{aligned} \tag{3}$$

The structured variational approximation is judiciously chosen to be $q(f) = p(f_{\neq \mathbf{u}} \mid \mathbf{u}, \theta) q(\mathbf{u})$. This leads to a cancellation of difficult terms, yielding the following variational lower bound,

$$\mathcal{F}(q, \theta) = \mathbb{E}_{q(f)}\left[\log \frac{p(\mathbf{y} \mid f) p(f_{\neq \mathbf{u}} \mid \mathbf{u}, \theta) p(\mathbf{u} \mid \mathbf{Z}, \theta)}{p(f_{\neq \mathbf{u}} \mid \mathbf{u}, \theta) q(\mathbf{u})}\right]$$

$$\begin{aligned} \mathcal{F}(q, \theta) = {}& \sum_{i=1}^N \mathbb{E}_{q(f)}\left[\log p\left(y_i \mid f, \mathbf{x}_i\right)\right] \\ & - \mathcal{KL}\left[q(\mathbf{u}) \,\|\, p(\mathbf{u} \mid \theta)\right] . \end{aligned} \tag{4}$$

The first term in this bound remains intractable for a general likelihood and large N. However, it can be approximated by simple Monte Carlo with reparameterisation gradients and data subsampling (Jordan et al., 1999; Hoffman et al., 2013; Hensman et al., 2013). Bui et al. (2017a) extend this bound for streaming data, and show good performance on time series regression. This approach, however, employs a single set of inducing points which is *shared* across multiple time steps or tasks and, as a result, the inducing points are either too rigid to adapt, or too quickly moving to regions of new data. We demonstrate this pitfall in Section 5. Equipped with this background, we are now ready to derive the proposed approximation, VAR-GPs.

4. Variational Auto-Regressive Gaussian Processes

Consider datasets $\{\mathcal{D}^{(1)}, \ldots, \mathcal{D}^{(T)}\}$, of sizes $\{N_1, \ldots, N_T\}$ respectively, for T different but related tasks. These tasks are observed sequentially and only once. We want a model which performs well not only on the current task, but also sustains performance on previous tasks. In subsequent notations, we identify task-specific quantities with corresponding task numbers, $\{1, \ldots, T\}$.

4.1. Learning the First Task

We first extend the model in (3) to incorporate a prior over the hyperparameters $p(\theta)$ and

choose the corresponding variational posterior $q_1(f, \theta) = p\left(f_{\neq \mathbf{u}_1} \mid \mathbf{u}_1, \theta\right) q(\mathbf{u}_1) q_1(\theta)$. While the correlation between $\mathbf{u}$ and θ is ignored in the approximate posterior, the dependencies between θ and the remaining function values are retained through the conditional prior term. Learning the first task using $\mathcal{D}^{(1)}$ is thus a direct extension of the sparse variational approach in Section 3, leading to the following bound,

$$
\begin{aligned}
\mathcal{F}(q_1) = \sum_{i=1}^{N_1} & \mathbb{E}_{q_1(f, \theta)} \left[\log p\left(y_i^{(1)} \mid f, \mathbf{x}_i^{(1)} \right) \right] \\
& - \mathcal{KL}\left[q_1(\theta) \mid\mid p(\theta) \right] \\
& - \mathbb{E}_{q_1(\theta)} \left[\mathcal{KL}\left[q(\mathbf{u}_1) \mid\mid p(\mathbf{u}_1 \mid \theta) \right] \right] .
\end{aligned}
\tag{5}
$$

We stress that quantifying uncertainty in hyperparameters through $q_1(\theta)$ is important for continual learning, as supported by our ablation studies in Section 5.3.

4.2. Generalized Continual Variational Lower Bound

Instead of reusing and updating the same set of inducing inputs and outputs for all subsequent tasks as in Bui et al. (2017a), we introduce a separate set of inducing inputs and outputs, $\{\mathbf{Z}_t, \mathbf{u}_t\}$, for each task. Consider the t-th task and $t > 1$, using notation in a spirit similar to (3), the function can be split into, $f = \{f_{\neq \mathbf{u}_{\leq t}}, \mathbf{u}_{<t}, \mathbf{u}_t\}$ that describes a decomposition in terms of all past inducing outputs $\mathbf{u}_{<t}$ and current inducing outputs $\mathbf{u}_t$. This leads to an approximate running joint for $\mathcal{D}^{(t)}$ conditioned on past data as,

$$
\begin{aligned}
p(\mathbf{y}^{(t)}, f, \theta \mid \mathbf{X}^{(t)}, \mathcal{D}^{(<t)}) \approx \prod_{i=1}^{N_t} & p(y_i^{(t)} \mid f, \mathbf{x}_i^{(t)}) \\
& p(f_{\neq \mathbf{u}_{\leq t}} \mid \mathbf{u}_{\leq t}, \theta) \\
& p(\mathbf{u}_t \mid \mathbf{Z}_t, \mathbf{u}_{<t}, \theta) \\
& q(\mathbf{u}_{<t} \mid \mathbf{Z}_{<t}, \theta) \\
& q_{t-1}(\theta) .
\end{aligned}
\tag{6}
$$

We mirror the form of the prior in the approximate posterior, in a similar fashion to the first task, to obtain the following variational distribution,

$$
\begin{aligned}
q_t(f, \theta) = &\, p(f_{\neq \mathbf{u}_{\leq t}} \mid \mathbf{u}_{\leq t}, \theta) q(\mathbf{u}_t \mid \mathbf{Z}_t, \mathbf{u}_{<t}, \mathbf{Z}_{<t}, \theta) \\
& q(\mathbf{u}_{<t} \mid \mathbf{Z}_{<t}, \theta) q_t(\theta) .
\end{aligned}
\tag{7}
$$

This structured form leads to the cancellation of $p(f_{\neq \mathbf{u}_{\leq t}} \mid \mathbf{u}_{\leq t}, \theta)$ and $q(\mathbf{u}_{<t} \mid \mathbf{Z}_{<t}, \theta)$ to arrive at the generalized continual variational lower bound,

$$
\begin{aligned}
\mathcal{F}(q_t) = \sum_{i=1}^{N_t} & \mathbb{E}_{q_t(f, \theta)} \left[\log p(y_i^{(t)} \mid f, \mathbf{x}_i^{(t)}) \right] \\
& - \mathcal{KL}\left[q_t(\theta) \mid\mid q_{t-1}(\theta) \right] \\
& - \mathbb{E}_{q_t(\theta) q(\mathbf{u}_{<t} \mid \mathbf{Z}_{<t}, \theta)} \left[\mathfrak{D}_t \right] ,
\end{aligned}
\tag{8}
$$

where,

$$
\mathfrak{D}_t = \mathcal{KL}\left[q(\mathbf{u}_t \mid \mathbf{Z}_t, \mathbf{u}_{<t}, \mathbf{Z}_{<t}, \theta) \mid\mid p(\mathbf{u}_t \mid \mathbf{Z}_t, \mathbf{u}_{<t}, \mathbf{Z}_{<t}, \theta) \right] .
$$

We emphasize that any dependence on $\mathbf{u}_t$ is accompanied by corresponding $\mathbf{Z}_t$ as presented in (1) and (2), but may often keep it implicit for concise notation.

As our learning objective, the maximization of (8) takes a natural interpretation — we maximize the likelihood of current data $\mathcal{D}^{(t)}$, subject to a $\mathcal{KL}$-regularization that balances past posterior and new data. The regularization term involves hyperparameters and current inducing outputs. In practice, tempering the hyperparameter distribution may prove helpful under misspecified models (Wenzel et al., 2020; Wilson & Izmailov, 2020), which is equivalent to scaling $\mathcal{KL}\left[q_t(\theta) \mid\mid q_{t-1}(\theta) \right]$ by a positive scalar β. The new inducing points $\{\mathbf{Z}_t, \mathbf{u}_t\}$ are used to explain new parts of the data space while the old ones $\{\mathbf{Z}_{<t}, \mathbf{u}_{<t}\}$ aim to preserve past experience, as demonstrated in Figure 1. Next, we develop this intuition further, and experimentally demonstrate in Section 5.1.

4.3. Distributional Choices

We now detail the parametrizations for all aforementioned distributions. All prior-related densities, i.e. $p(\mathbf{u}_1 \mid \theta)$, $p(\mathbf{u}_t \mid \mathbf{u}_{<t}, \theta)$, and $p(f_{\mathbf{u}_{\leq t}} \mid \mathbf{X}^{(t)}, \mathbf{u}_{\leq t}, \theta)$ can be computed by invoking the GP prior. All experiments use the Exponentiated Quadratic kernel such that θ includes the log-ARD lengthscales and a log-scale factor (see Appendix C.1). The prior over the log-hyperparameters for the first task $p(\theta)$ is assumed to be a standard Normal, $\mathcal{N}(\theta; \mathbf{0}, \mathbf{I})$.

The variational distribution of the log-hyperparameters is assumed to be a mean-field Gaussian, i.e. parametrized by a diagonal covariance, $q_t(\theta) = \mathcal{N}(\theta; \boldsymbol{\mu}_t, \text{diag}(\boldsymbol{\sigma_t}))$. These choices allow for closed-form $\mathcal{KL}$ computations in (5) and (8). The variational distribution over inducing outputs for the first task is parametrized as $q(\mathbf{u}_1) = \mathcal{N}(\mathbf{u}_1; \mathbf{m}_1, \boldsymbol{\Sigma}_1)$ for a set of M_1 inducing outputs, as standard in sparse variational GPs using mean $\mathbf{m}_1$ and covariance matrix $\boldsymbol{\Sigma}_1$.

One of our key contributions is an auto-regressive parametrization for the variational distribution given by $q(\mathbf{u}_{\leq t} \mid \theta) = q(\mathbf{u}_1) \prod_{j=2}^t q(\mathbf{u}_j \mid \mathbf{u}_{<j}, \theta)$, where $q(\mathbf{u}_t \mid \mathbf{u}_{<t}, \theta) = \mathcal{N}(\mathbf{u}_t; \mathbf{K}_{\mathbf{Z}_t, \mathbf{z}_{<t}} \mathbf{K}_{\mathbf{Z}_{<t}, \mathbf{Z}_{<t}}^{-1} \mathbf{u}_{<t} + \mathbf{m}_t, \boldsymbol{\Sigma}_t)$ relies on M_t inducing variables for all subsequent tasks $t > 1$. As a consequence of this structure, $\mathfrak{D}_t$ becomes independent of $\mathbf{u}_{<t}$ and avoids sampling variance introduced by samples from $q(\mathbf{u}_{<t} \mid \mathbf{Z}_{<t}, \theta)$. A simplifying trick to compute $q(\mathbf{u}_{<t} \mid \theta)$ using conditional Gaussian identities (see Appendix C.3). We further note that (i) the marginal density for inducing outputs $\mathbf{u}_t$ when $t > 1$ is non-Gaussian, and (ii) even though $\mathbf{Z}_t, \mathbf{m}_t, \boldsymbol{\Sigma}_t$ are kept

fixed after training on the t-th task, the marginal density $q(\mathbf{u}_t)$ can still change over time due to the change in $q(\theta)$.

Next, Section 4.4 provide two fruitful connections of the proposed structured variational approximation to existing literature.

4.4. Connections to Expectation Propagation and Orthogonal Inducing Points

Structured EP Factor Approximation leads to an Auto-regressive Approximate Posterior For the approximate running joint in (6), we can introduce an approximation to the posterior as,

$$q_t(f, \theta) \propto \left[\prod_{i=1}^{N_t} \mathbf{g}_i^{(t)}(\theta) \mathbf{h}_i^{(t)}(\mathbf{u}_t) \right] p(f_{\neq \mathbf{u}_{\leq t}} \mid \mathbf{u}_{\leq t}, \theta) \quad (9)$$
$$q(\mathbf{u}_{<t} \mid \theta) p(\mathbf{u}_t \mid \mathbf{u}_{<t}, \theta) q_{t-1}(\theta),$$

where the difficult likelihood term, $\prod_{i=1}^{N_t} p(y_i^{(t)} \mid f, \mathbf{x}_i^{(t)})$, is approximated by $\prod_{i=1}^{N_t} \mathbf{g}_i^{(t)}(\theta) \mathbf{h}_i^{(t)}(\mathbf{u}_t)$ such that $\mathbf{g}_i$ and $\mathbf{h}_i$ are the approximate contributions of each likelihood term to the posterior. Expectation Propagation (EP) (Minka, 2001) then proceeds by repeating the following steps to convergence: i) remove the approximate contributions $\mathbf{g}_i$ and $\mathbf{h}_i$ from the posterior to form the cavity for i-th datum, ii) merge the cavity with $p(y_i^{(t)} \mid f, \mathbf{x}_i^{(t)})$ to form the tilted distribution $\tilde{p}_i$, iii) minimize the divergence $\mathcal{KL}[\tilde{p}_i \mid\mid q_t]$ to find a new approximate posterior, and iv) obtain the new approximate factors $\mathbf{g}_i$ and $\mathbf{h}_i$ by removing the cavity from the new posterior.

We are, however, not interested in running EP, but only in the form of the approximate posterior induced by EP. Merging relevant terms in the approximate posterior as

$$q_t(f, \theta) \propto p(f_{\neq \mathbf{u}_{\leq t}} \mid \mathbf{u}_{\leq t}, \theta)$$
$$q(\mathbf{u}_t \mid \mathbf{u}_{<t}, \theta) q(\mathbf{u}_{<t} \mid \theta) \quad (10)$$
$$q_t(\theta),$$

where,

$$q_t(\theta) \propto q_{t-1}(\theta) \prod_{i=1}^{N_t} \mathbf{g}_i^{(t)}(\theta),$$

$$q(\mathbf{u}_t \mid \mathbf{u}_{<t}, \theta) \propto p(\mathbf{u}_t \mid \mathbf{u}_{<t}, \theta) \prod_{i=1}^{N_t} \mathbf{h}_i^{(t)}(\mathbf{u}_t).$$

We find that the auto-regressive factor is given by $p(\mathbf{u}_t \mid \mathbf{u}_{<t}, \theta) = \mathcal{N}(\mathbf{u}_t; \mathbf{A}_t \mathbf{u}_{<t}, \mathbf{C}_t)$, where we define $\mathbf{A}_t \triangleq \mathbf{K}_{\mathbf{Z}_t, \mathbf{Z}_{<t}} \mathbf{K}_{\mathbf{Z}_{<t}, \mathbf{Z}_{<t}}^{-1}$, and $\mathbf{C}_t \triangleq \mathbf{K}_{\mathbf{Z}_t, \mathbf{Z}_t} - \mathbf{K}_{\mathbf{Z}_t, \mathbf{Z}_{<t}} \mathbf{K}_{\mathbf{Z}_{<t}, \mathbf{Z}_{<t}}^{-1} \mathbf{K}_{\mathbf{Z}_{<t}, \mathbf{Z}_t}$.

Now, consider a Gaussian factor approximation $\mathbf{H}_t(\mathbf{u}_t) = \prod_{i=1}^{N_t} \mathbf{h}_t(\mathbf{u}_t) = \mathcal{N}(\mathbf{u}_t; \mu, \Sigma)$. By multi-

plying $\mathbf{H}_t(\mathbf{u}_t)$ with $p(\mathbf{u}_t \mid \mathbf{u}_{<t}, \theta)$ and renormalizing, we arrive at $q(\mathbf{u}_t \mid \mathbf{u}_{<t}, \theta) = \mathcal{N}(\mathbf{u}_t; \tilde{\mu}, \tilde{\Sigma})$, where $\tilde{\Sigma}^{-1} = \Sigma^{-1} + \mathbf{C}_t^{-1}$ and $\tilde{\Sigma}^{-1} \tilde{\mu} = \Sigma^{-1} \mu + \mathbf{C}_t^{-1} \mathbf{A}_t \mathbf{u}_{<t}$. The conditional posterior mean can be reformulated as $\tilde{\mu} = \mathbf{A}_t \mathbf{u}_{<t} + (\mathbf{I} + \Sigma \mathbf{C}_t^{-1})^{-1} (\mu - \mathbf{A}_t \mathbf{u}_{<t})$, so that we can parameterize the conditional posterior instead of the factor $\mathbf{H}_t(\mathbf{u}_t)$, giving us $q(\mathbf{u}_t \mid \mathbf{u}_{<t}, \theta) = \mathcal{N}(\mathbf{u}_t; \mathbf{A}_t \mathbf{u}_{<t} + \mathbf{m}, \Sigma)$. This is equivalent to our auto-regressive parametrization.

Equivalence to Orthogonal Inducing Points Our auto-regressive parametrization is also exactly equivalent to the *orthogonal inducing points* formulation when $T = 2$ (Shi et al., 2020, §3.3). A variational approximation over two set of orthogonal inducing points, $\mathbf{u}$ and $\mathbf{v}$, is presented as

$$q(\mathbf{u}, \mathbf{v}) = \mathcal{N}\left(\begin{bmatrix} \mathbf{u} \\ \mathbf{v} \end{bmatrix}; \begin{bmatrix} \mathbf{m}_\mathbf{u} \\ \mathbf{K}_{\mathbf{vu}} \mathbf{K}_{\mathbf{uu}}^{-1} \mathbf{m}_\mathbf{u} \end{bmatrix}, \boldsymbol{\Sigma} \right), \quad (11)$$

where,

$$\boldsymbol{\Sigma} \triangleq \begin{bmatrix} \boldsymbol{\Sigma}_\mathbf{u} & \boldsymbol{\Sigma}_\mathbf{u} \mathbf{K}_{\mathbf{uu}}^{-1} \mathbf{K}_{\mathbf{uv}} \\ \mathbf{K}_{\mathbf{vu}} \mathbf{K}_{\mathbf{vu}}^{-1} \boldsymbol{\Sigma}_\mathbf{u} & \boldsymbol{\Sigma}_\mathbf{v} + \mathbf{K}_{\mathbf{vu}} \mathbf{K}_{\mathbf{uu}}^{-1} \boldsymbol{\Sigma}_\mathbf{u} \mathbf{K}_{\mathbf{uu}}^{-1} \mathbf{K}_{\mathbf{uv}} \end{bmatrix}.$$

By appealing to conditional Gaussian identities, the joint can be factored as $q(\mathbf{u}, \mathbf{v}) = q(\mathbf{u}) q(\mathbf{v} \mid \mathbf{u})$, such that $q(\mathbf{u}) = \mathcal{N}(\mathbf{u}; \mathbf{m}_\mathbf{u}, \boldsymbol{\Sigma}_\mathbf{u})$, and $q(\mathbf{v} \mid \mathbf{u}) = \mathcal{N}(\mathbf{v}; \mathbf{K}_{\mathbf{vu}} \mathbf{K}_{\mathbf{uu}}^{-1} \mathbf{u} + \mathbf{m}_\mathbf{v}, \boldsymbol{\Sigma}_\mathbf{v})$. This conditional is exactly equivalent to our proposal when there are two tasks, i.e. $\mathbf{u} \Leftrightarrow \mathbf{u}_1$ and $\mathbf{v} \Leftrightarrow \mathbf{u}_2$. Intuitively, the second set of inducing variables $\mathbf{u}_2$ attempt to explain the data space that is not well-explained by the first set $\mathbf{u}_1$. Shi et al. (2020) briefly discussed using more than two sets of inducing points for a fixed dataset but did not investigate further due to implementation complexity and the potential small gain beyond two sets. Unlike the batch setting of Shi et al. (2020), our work extends the idea of using many sets of such inducing points to the continual learning setting.

5. Experiments

Through our experiments, we highlight the qualitative characteristics of the derived generalized learning objective (8), and provide evidence for the competitiveness of VAR-GPs compared to our main baseline Improved Variational Continual Learning (VCL) (Nguyen et al., 2018; Swaroop et al., 2019), among others. A thorough ablation study demonstrates the efficacy of our modeling choices. The full reference implementation of VAR-GPs in PyTorch (Paszke et al., 2019) in publicly available at u.perhapsbay.es/vargp-code.

To mimic a real continual learning setting, the model only observes $\mathcal{D}^{(t)}$ for training and is tested on all the tasks seen
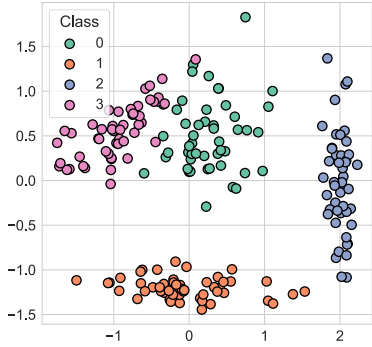
Figure 2. A synthetic toy dataset used for the four-way classification problem in Figure 1.

so far using $\mathcal{D}^{(\leq t)}$. We track validation accuracy on a subset of the training set for early stopping. To optimize both variational objectives (5) and (8), we use a mini-batch size of 512 and 3 samples from variational distribution with the Yogi optimizer (Zaheer et al., 2018). Predictive distributions are estimated via 10 Monte-Carlo samples. The training for the t-th task is summarized in Algorithm 1.

Algorithm 1 VAR-GP per-task training

Input: Learning rate η, Batch size B, Number of inducing points M, Maximum epochs E, Task dataset $\mathcal{D}^{(t)}$, hyperparameters: KL tempering factor β, Early stopping patience of K epochs and tolerance δ
Output: Per-task variational approximations for kernel hyper-parameters θ and inducing outputs $\mathbf{u}_t$, and inducing inputs $\mathbf{Z}_t$
Initialize $\mathbf{Z}_t \in \mathbb{R}^{M \times D} \subset \mathbf{X}^{(t)} \in \mathbb{R}^{N_t \times D}$
for e in $1 \ldots E$ **do**
    for $\{\mathbf{x}_i, y_i\}_{i=1}^{B} \subset \mathcal{D}^{(t)}$ **do**
        Compute $\mathcal{F}(q_t)$ — using (5) for the first task, and (8) for all subsequent ones.
        Update θ, $\mathbf{m}_t$, $\boldsymbol{\Sigma}_t$ and $\mathbf{Z}_t$ with learning rate η and tempering factor β
    end for
    Compute validation accuracy A_e
    if $e > K$ **and** $|A_e - A_{e-K}| < \delta$ **then**
        `break`
    end if
end for

All covariance matrices in the variational distributions are modeled as a lower triangular Cholesky factors, described in Appendix C.2, initialized at an identity matrix for numerical stability, with diagonals constrained to be positive during optimization using a `softplus` transform. Further, as described in Appendix C.1, the kernel parameters θ, including lengthscales and scale factors modeled as `log`-transforms to maintain positivity. Appendix D.1 lists the

search space for all hyper-parameters. Results report the mean and one standard deviation of five independent trials. Next, we describe the datasets used for experiments.

Synthetic Classification Dataset Visualized in Figure 2, we use a synthetic 2-D dataset with four classes in the range $x, y \in [-3., 3.]$ for qualitative assessments. We observe classes in pairs $0/1$ (Task 0) and $2/3$ (Task 1), each only once. We do not use any tempering for this dataset, i.e. $\beta = 1$.

Split MNIST Following Zenke et al. (2017), we consider the full 10-way classification task at each time step but receive a dataset $\mathcal{D}^{(t)}$ of only a subset of MNIST digits in the sequence $0/1$, $2/3$, $4/5$, $6/7$, and $8/9$. 10000 training samples are cumulatively set aside for validation set across all tasks. We allocate 60 inducing points for each task, with a learning rate of 0.003, and $\beta = 10.0$. We remind the reader that unlike prior work which uses a multi-head model with task information, we only use a single-head to report the classification test accuracy, making the benchmark considerably harder and more representative of reality.

Permuted MNIST In this benchmark, we receive a dataset $\mathcal{D}^{(t)}$ of MNIST digits at each time step t, such that the pixels undergo an unknown but fixed permutation. 10000 samples are set aside for validation. We allocate 100 inducing points for each task, with a learning rate of 0.0037, and $\beta = 1.64$. The first task is fixed to be the unpermuted MNIST to provide an upper bound on the performance of subsequent tasks. While prior work (Zenke et al., 2017; Kirkpatrick et al., 2017) uses this benchmark as an indicator of representational capacity of neural networks, we use this to test performance under distributional shift.

5.1. Qualitative Analysis

Before presenting benchmark results, we seek to intuitively understand how VAR-GPs learn and avoid *catastrophic forgetting*. By modeling the cross-task covariances through $\mathfrak{D}_t$ in (8), we are able to bias the learning such that under the given budget of inducing points, VAR-GPs retain information about the previous tasks. This is exemplified in Figure 1. Observing predictive density plots for Classes 0 and 1 after "Task 1", we notice that the high density regions from "Task 0" are preserved by VAR-GPs, whereas VCL suffers from large variations in the predictions.

This characteristic is not restricted to the toy dataset. Using mean predictive entropy estimates to quantify uncertainty over the test set, we visualize the degree of forgetting for benchmark datasets in Figure 3. Even as tasks progress, VAR-GPs demonstrate information preservation from old tasks by keeping entropy low. In contrast, often being overconfident in predictions, VCL keeps entropy low only for the old tasks at the expense of new ones.
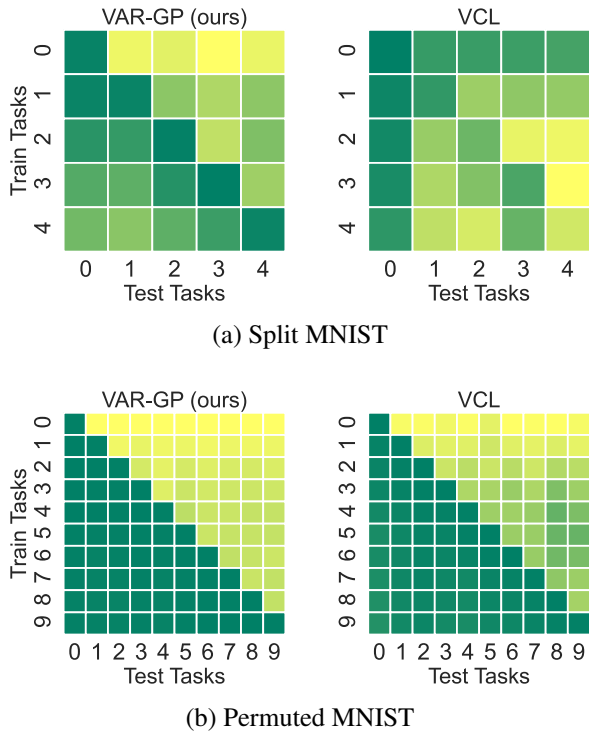
(a) Split MNIST

(b) Permuted MNIST

Figure 3. As we continually train on tasks (shown along the y-axis), we evaluate the mean predictive entropy on the test set for all tasks (shown along the x-axis). In other words, the upper triangular region shows tasks which are not yet seen during training. The values are normalized by the entropy of a random ten-way classifier, $\log 10$. Brighter regions correspond to a higher entropy, i.e. larger predictive uncertainty. (**Top**) For Split MNIST, we observe that VAR-GPs lead to lower forgetting rates through lower predictive entropy in the lower triangular regions, and higher uncertainties in upper triangular regions. VCL (Swaroop et al., 2019), on the other hand fails to improve much on subsequent tasks. (**Bottom**) For Permuted MNIST, we see that while both VAR-GPs and VCL maintain reasonably low catastrophic forgetting on the tasks seen so far, VCL tends to be overconfident on unseen tasks, i.e. low predictive entropy in the upper triangular region.

Finally, an inspection of the optimized inducing points reveals that VAR-GPs tend to focus on covering the regions of the current task, as one would naturally expect. Figure 4 showcases this behavior for Split MNIST. We anticipate such space covering behavior from our learning objective (8) due to the cross-correlations modeled in the regularization term $\mathfrak{D}_t$.

5.2. Benchmark Results

With a qualitative picture of how VAR-GPs learn favorably for sequential tasks in continual learning, we validate its performance on the benchmark datasets of Split MNIST and Permuted MNIST.
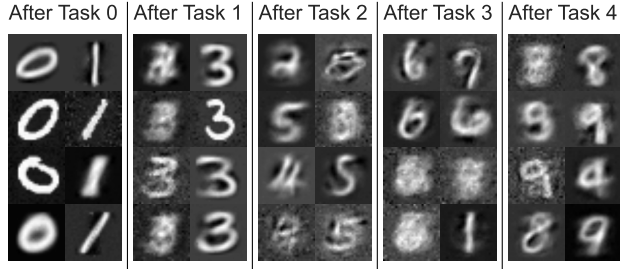

Figure 4. We visualize the inducing points learned after each task in Split MNIST, progressing from left to right. This figure reveals that VAR-GPs spread inducing points to cover the input space for each task.

Figure 5 shows how the cumulative performance across all tasks seen so far evolves as the number of tasks increase. VAR-GPs perform favorably by performing consistently better throughout. One of the most competitive methods, VCL and its variants involving episodic memory through coresets (Swaroop et al., 2019) tend to deteriorate much faster, hinting at catastrophic forgetting. Quantitatively, we find the final mean test accuracy achieved by VAR-GPs after training on five Split MNIST tasks is 90.57%, which is significantly better than the best performing VCL variant at 81.90%. Similarly, VAR-GPs achieve a mean test accuracy of 97.2% after training on ten tasks of Permuted MNIST, as compared to 95.06% by the best variant of VCL. Comparisons against other continual learning algorithms are summarized in Table 1.

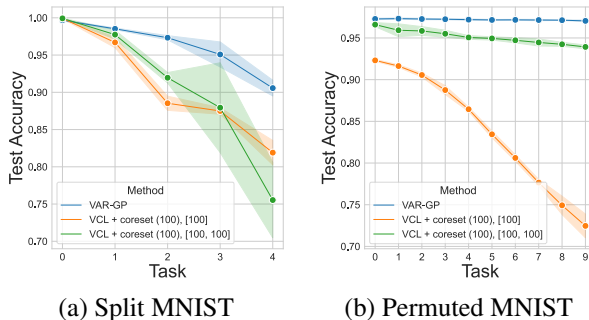

(a) Split MNIST   (b) Permuted MNIST

Figure 5. Comparing VAR-GPs to variants of VCL (Nguyen et al., 2018; Swaroop et al., 2019) as we see more tasks (shown along the X-axis). We compute the average test accuracy (show along Y-axis) across all tasks seen so far, including the current one. The plots show mean and one standard deviation across five independent trials. VAR-GPs preserve information from old tasks, avoid catastrophic forgetting and consequently perform much better on newer tasks without compromising on old ones. See Table 1, for all numerical results.

5.3. Ablations

Finally, we validate the specific modeling choices in VAR-GPs via a thorough ablation study.

Table 1. The final average test accuracy (in %) after sequential training on benchmark dataset tasks —five for Split MNIST and ten for Permuted MNIST are noted below. We provide the mean and one standard deviation (where available) over five independent trials. *Split MNIST results for SI (Zenke et al., 2017) and EWC (Kirkpatrick et al., 2017) are taken from VCL (Swaroop et al., 2019), however, are not directly comparable as they use a multi-head setup making evaluation easier; CS = coreset.; [100] and [100, 100] represent hidden layer sizes of the neural networks.

Method	Split MNIST	Permuted MNIST
SI	98.9*	86.02
EWC	63.1*	84.11
VCL, [100]	19.90 ± 0.14	82.94 ± 0.85
+ CS(50)	76.21 ± 2.02	85.80 ± 0.41
+ CS(100)	81.90 ± 1.64	86.45 ± 0.26
VCL, [100, 100]	19.91 ± 7.79	94.31 ± 1.05
+ CS(50)	71.89 ± 5.06	95.43 ± 0.39
+ CS(100)	75.54 ± 1.06	95.06 ± 0.22
VAR-GP (ours)	**90.57** ± 1.06	**97.20** ± 0.08
+ Block Diag.	78.64 ± 1.41	96.31 ± 0.42
+ MLE Hypers	10.09 ± 0.40	10.07 ± 0.15
+ Global	39.31 ± 0.28	46.02 ± 1.09

Block-Diagonal Variational Distribution Instead of the auto-regressive posterior $q(\mathbf{u}_t \mid \mathbf{u}_{<t}, \theta) = \mathcal{N}(\mathbf{u}_t; \mathbf{K}_{\mathbf{Z}_t, \mathbf{z}_{<t}} \mathbf{K}_{\mathbf{Z}_{<t}, \mathbf{z}_{<t}}^{-1} \mathbf{u}_{<t} + \mathbf{m}_t, \mathbf{\Sigma}_t)$, we choose a simpler variational distribution given by $q'(\mathbf{u}_t \mid \mathbf{u}_{<t}, \theta) = \mathcal{N}(\mathbf{u}_t; \mathbf{m}_t, \mathbf{\Sigma}_t)$, keeping everything else the same. Effectively, removing the conditioning induces a block diagonal structure in the full covariance matrix across the inducing variables where each block represents the covariance among the inducing points for a given task. This also decouples the inducing points and the hyperparameters in the approximate posterior. We hypothesize that this is detrimental to performance. Figure 6 shows that the long-term continual learning performance tends to deteriorate faster, hinting at *greater* catastrophic forgetting.

Global Inducing Points In an alternative model, we completely do away with the auto-regressive nature of the variational distribution and just rely on a single set of inducing points at each time step $q_t(f, \theta) = q_t(\theta)p(f_{\neq \mathbf{u}_t} \mid \mathbf{u}_t, \theta)q(\mathbf{u}_t)$. Bui et al. (2017a) rely on this variational distribution for the inducing points and an MLE estimate of the hyperparameters for streaming regression tasks. See Appendix B.1 for precise modeling details and derivations. The experimental evidence in Figure 6, however, shows that such variational approximation is poor for large scale continual classification tasks.

MLE Hyperparameters Quantifying uncertainty about the hyperparameters considerably helps the model to perform well across tasks without a detrimental effect on the old ones. In this section, we validate this hypothesis by simply switching off the $\mathcal{KL}$-divergence term for the hyperparameters in both (5) and (8). Instead, we rely on a point estimate of the hyperparameters and use the maximum likelihood estimate at each step. The stark performance comparison is shown in Figure 6. The hyper-parameters are stuck in a local minimum and virtually never recover for subsequent tasks.

Retraining Old Inducing Points We also investigate a variant of VAR-GP named Re-VAR-GP where unlike earlier, we retrain the old inducing points $\mathbf{Z}_{<t}$. This changes the form of the variational lower bound. The precise details and implications are discussed in Appendix B.2, owing to which we do not pursue this approach further.
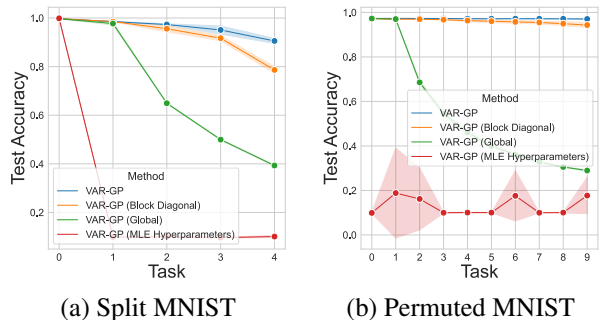

(a) Split MNIST   (b) Permuted MNIST

Figure 6. A thorough ablation study reveals the significance of our modeling choices. Evidently, VAR-GPs are able to sustain long-term continual learning performance than any other variant. See Section 5.3 for a detailed description of the ablations conducted.

6. Discussion

In this work, we develop VAR-GPs, a principled Bayesian inference scheme for continual learning. Using a sparse inducing point approximation, we propose a structured variational approximation to the true posterior for Gaussian processes, which reveals two fruitful connections to Expectation Propagation and Orthogonal Inducing Points. The resulting lower bound provides a natural learning objective to update the belief over underlying functions as new data arrives. A qualitative characterization of how VAR-GPs learn is presented, backed by strong empirical results on modern continual learning benchmarks. Further, a thorough ablation study establishes the efficacy of modeling choices.

In its current form, the number of inducing points grows linearly in the number of tasks. Consequently, the inference grows cubic in the number of tasks. This computational complexity limits the scaling of VAR-GPs to only a

moderate number of continual learning tasks. Nevertheless, accommodating a post-hoc information distillation procedure in the learning process can prove helpful for scalability, and remains an open research problem. While Gaussian processes provide a flexible approach towards priors in the functional space, the representational power of VAR-GPs can be improved by the use of deep kernel learning. This, however, leads to new regularization challenges to be tackled (see Appendix B.3). Contemporary approaches relying on a memory buffer are also amenable to the proposed inference scheme, since the inducing points provide an automated data selection procedure. Finally, model-based reinforcement learning, where dataset shift is unavoidable, is a promising application space for VAR-GPs.

Through VAR-GPs, we remain optimistic that Gaussian processes provide an effective foundation for predictions under uncertainty in continual learning. More broadly, the characteristics highlighted by our learning objective (8) serve as a principled starting point to guide future research in continual learning objectives.

Acknowledgements

References

Ahn, H., Cha, S., Lee, D., and Moon, T. Uncertainty-based continual learning with adaptive regularization. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, 2019. 2

Broderick, T., Boyd, N., Wibisono, A., Wilson, A. C., and Jordan, M. I. Streaming variational bayes. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States*, 2013. 2

Bui, T. D., Nguyen, C. V., and Turner, R. E. Streaming sparse gaussian process approximations. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, 2017a. 2, 3, 4, 8

Bui, T. D., Yan, J., and Turner, R. E. A unifying framework for Gaussian process pseudo-point approximations using power expectation propagation. *The Journal of Machine Learning Research*, (1), 2017b. 3

Chaudhry, A., Dokania, P. K., Ajanthan, T., and Torr, P. H. Riemannian walk for incremental learning: Understanding forgetting and intransigence. In *Proceedings of the European Conference on Computer Vision*, 2018. 2

Csató, L. *Gaussian processes: iterative sparse approximations*. PhD thesis, Aston University, 2002. 2

Csató, L. and Opper, M. Sparse on-line Gaussian processes. *Neural computation*, (3), 2002. 2

Farquhar, S. and Gal, Y. Towards robust evaluations of continual learning. *arXiv preprint arXiv:1805.09733*, 2018. 2

Hensman, J., Fusi, N., and Lawrence, N. D. Gaussian processes for big data. In *Proceedings of the Twenty-Ninth Conference on Uncertainty in Artificial Intelligence, UAI 2013, Bellevue, WA, USA, August 11-15, 2013*, 2013. 2, 3

Hensman, J., de G. Matthews, A. G., and Ghahramani, Z. Scalable variational gaussian process classification. In *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2015, San Diego, California, USA, May 9-12, 2015*, JMLR Workshop and Conference Proceedings, 2015. 2, 3

Hoffman, M. D., Blei, D. M., Wang, C., and Paisley, J. Stochastic variational inference. *The Journal of Machine Learning Research*, (1), 2013. 2, 3

Hoppitt, W. and Laland, K. N. *Social learning: an introduction to mechanisms, methods, and models*. 2013. 1

Jordan, M. I., Ghahramani, Z., Jaakkola, T. S., and Saul, L. K. An introduction to variational methods for graphical models. *Machine learning*, (2), 1999. 3

Kessler, S., Nguyen, V., Zohren, S., and Roberts, S. Hierarchical Indian buffet neural networks for Bayesian continual learning. *arXiv*, 2019. 2

Kirkpatrick, J., Pascanu, R., Rabinowitz, N., Veness, J., Desjardins, G., Rusu, A. A., Milan, K., Quan, J., Ramalho, T., Grabska-Barwinska, A., et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, (13), 2017. 2, 6, 8

Li, Z. and Hoiem, D. Learning without forgetting. *IEEE transactions on Pattern Analysis and Machine Intelligence*, (12), 2017. 2

Lopez-Paz, D. and Ranzato, M. Gradient episodic memory for continual learning. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, 2017. 2

McCloskey, M. and Cohen, N. J. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of learning and motivation*. 1989. 1

Minka, T. P. Expectation propagation for approximate bayesian inference. In *Proceedings of the 17th Conference in Uncertainty in Artificial Intelligence, University of Washington, Seattle, Washington, USA, August 2-5, 2001*, 2001. 5

Moreno-Muñoz, P., Artés-Rodríguez, A., and Álvarez, M. A. Continual multi-task Gaussian processes. *arXiv preprint arXiv:1911.00002*, 2019. 2

Murphy, K. P. *Machine learning: a probabilistic perspective*. 2012. 14

Nguyen, C. V., Li, Y., Bui, T. D., and Turner, R. E. Variational continual learning. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*, 2018. 2, 5, 7

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Köpf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, 2019. 5

Quiñonero-Candela, J. and Rasmussen, C. E. A unifying view of sparse approximate Gaussian process regression. *Journal of Machine Learning Research*, (Dec), 2005. 3

Rasmussen, C. E. and Williams, C. K. I. *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. 2005. 3

Ratcliff, R. Connectionist models of recognition memory: constraints imposed by learning and forgetting functions. *Psychological review*, (2), 1990. 1

Rebuffi, S., Kolesnikov, A., Sperl, G., and Lampert, C. H. icarl: Incremental classifier and representation learning. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, 2017. 2

Ring, M. B. *Continual learning in reinforcement environments*. PhD thesis, University of Texas at Austin, 1994. 1

Ritter, H., Botev, A., and Barber, D. Online structured laplace approximations for overcoming catastrophic forgetting. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, 2018. 2

Rusu, A. A., Rabinowitz, N. C., Desjardins, G., Soyer, H., Kirkpatrick, J., Kavukcuoglu, K., Pascanu, R., and Hadsell, R. Progressive neural networks. *arXiv preprint arXiv:1606.04671*, 2016. 2

Sato, M.-A. Online model selection based on the variational Bayes. *Neural computation*, (7), 2001. 2

Schwarz, J., Czarnecki, W., Luketina, J., Grabska-Barwinska, A., Teh, Y. W., Pascanu, R., and Hadsell, R. Progress & compress: A scalable framework for continual learning. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, Proceedings of Machine Learning Research, 2018. 2

Shi, J., Titsias, M. K., and Mnih, A. Sparse orthogonal variational inference for gaussian processes. In *The 23rd International Conference on Artificial Intelligence and Statistics, AISTATS 2020, 26-28 August 2020, Online [Palermo, Sicily, Italy]*, Proceedings of Machine Learning Research, 2020. 5

Shin, H., Lee, J. K., Kim, J., and Kim, J. Continual learning with deep generative replay. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, 2017. 2

Swaroop, S., Nguyen, C. V., Bui, T. D., and Turner, R. E. Improving and understanding variational continual learning. *arXiv:1905.02099 [cs, stat]*, 2019. 2, 5, 7, 8

Thrun, S. Lifelong learning algorithms. In *Learning to learn*. 1998. 1

Titsias, M. Variational learning of inducing variables in sparse Gaussian processes. In *Artificial Intelligence and Statistics*, 2009. 2, 3

Titsias, M. K., Schwarz, J., de G. Matthews, A. G., Pascanu, R., and Teh, Y. W. Functional regularisation for continual learning with gaussian processes. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*, 2020. 2

van de Ven, G. M. and Tolias, A. S. Three scenarios for continual learning. *arXiv preprint arXiv:1904.07734*, 2019. 2

Wenzel, F., Roth, K., Veeling, B. S., Swiatkowski, J., Tran, L., Mandt, S., Snoek, J., Salimans, T., Jenatton, R., and Nowozin, S. How good is the bayes posterior in deep neural networks really? In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, Proceedings of Machine Learning Research, 2020. 4

Wilson, A. G. and Izmailov, P. Bayesian deep learning and a probabilistic perspective of generalization. *arXiv preprint arXiv:2002.08791*, 2020. 4

Wilson, A. G., Hu, Z., Salakhutdinov, R., and Xing, E. P. Deep kernel learning. In *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics, AISTATS 2016, Cadiz, Spain, May 9-11, 2016*, JMLR Workshop and Conference Proceedings, 2016. 13

Zaheer, M., Reddi, S. J., Sachan, D. S., Kale, S., and Kumar, S. Adaptive methods for nonconvex optimization. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, 2018. 6

Zenke, F., Poole, B., and Ganguli, S. Continual learning through synaptic intelligence. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, Proceedings of Machine Learning Research, 2017. 2, 6, 8