Overcoming Catastrophic Forgetting by Bayesian Generative Regularization

Patrick H. Chen [1] **Wei Wei** [2] **Cho-jui, Hsieh** [1] **Bo Dai** [3]

Abstract

The streaming update of Bayesian posterior calculation provides us a natural way for continual learning. However, the naïve mean-field posterior parametrization for variational approximation is inappropiate in neural network, and thus, lock the full ability for preventing catastrophic forgetting. To resolve this issue, we introduce a generative regularization for all given classification models, which is implemented by leveraging energy-based models with contrastive loss, to obtain the sufficient features for valid decomposition in posterior approxiamtion. By combining discriminative and generative loss together, we empirically show that the proposed method outperforms state-of-the-art methods on a variety of tasks, avoiding catastrophic forgetting in continual learning. In particular, the proposed method outperforms baseline methods over 15% on the Fashion-MNIST dataset and 10% on the CUB dataset.

1. Introduction

Many real-world machine learning applications require classification models to learn a sequence of tasks in an incremental way. For each task, learning system could only access part of whole data and the previously seen data can not be assessed. For example, previous customer data usually can not be accessed due to increasingly more strict data regulations on the user privacy, such as GDPR (Voigt & Von dem Bussche, 2017). The labelled data of an existing task can be depleted when new tasks emerge (Sutton et al., 2014; Kirkpatrick et al., 2017). Thus, an intelligent agent for continual learning must not only adapt to newly incoming tasks but also perform well on entire set of all the existing tasks in an incremental way that avoids revisiting all previous data at each stage. Previous studies (McCloskey & Cohen, 1989; Ratcliff, 1990) found that conventional deep learning models fail to tackle continual learning with the phenomenon of **catastrophic forgetting**, where deep neural networks tend to lose the information of previous tasks (i.e. classification accuracy drops significantly) after a new task is introduced.

Apparently, in order to achieve continual learning, catastrophic forgetting is an important issue to be addressed. A common strategy is to fix parameters used in the previous tasks. When new tasks arrive, based on different criteria, each method can reuse part of the fixed parameters (Fernando et al., 2017), expand some model components (Rusu et al., 2016; Yoon et al., 2018), or search for the best new model architecture to process new task (Li et al., 2019). Alternatively, instead of fixing a model, memory-based methods store a subset of previous data and constrain the update of models by leveraging the distilled knowledge from previous tasks (Castro et al., 2018; Hou et al., 2018; Javed & Shafait, 2018; Li & Hoiem, 2017; Rebuffi et al., 2017; Shin et al., 2017). These methods demonstrate the capability of alleviating the forgetting in practice on several datasets, however, without investigating and explaining the potential cause of catastrophic forgetting. More importantly, model adaption methods come at the cost that the model size expands correspondingly to the number of new tasks; while keeping data directly violates GDPR regulation. These drawbacks make existing methods not applicable for large-scale real-world applications. Therefore, there is a need to investigate the cause of catastrophic forgetting for a principled algorithm under the memoryless, fixed model setup.

Most of the existing literature views the incremental training as a moving path in parameter space, then the catastrophic forgetting happens when the update direction obtained based on partial data leads an inappropriate solution. Therefore, it is natural to design the search and update directions in training to avoid the catastrophic forgetting (Kirkpatrick et al., 2017; Nguyen et al., 2018; Zenke et al., 2017; Smola et al., 2003). Variational Continual Learning (VCL) (Nguyen et al., 2018), as a representative algorithm, exploits the equivalent streaming update form of Bayesian posterior calculation, which by nature only uses part of data, and therefore can combat forgetting. In practice, the exact posterior is intractable, especially in the Bayesian neural network, then, variational methods are used for approximation. The VCL achieves good empirical performance on various bench-

[1]Department of Computer Science, UCLA, California, USA [2]Google Cloud, Sunnyvale, California, USA [3]Google Brain, Mountain View, California, USA. Correspondence to: Patrick H. Chen <patrickchen@g.ucla.edu>.

Proceedings of the 38^{th} International Conference on Machine Learning, PMLR 139, 2021.

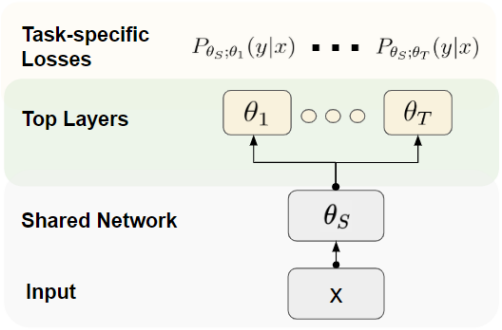

Figure 1. Illustration of the problem setup.

marks. However, VCL approxiamtes the posterior distribution by assuming parameters shared by all tasks to be *independent* of all task-specific parameters, which is difficult to satisfy, especially in neural network, as we will illustrate in Figure 1. Moreover, our experimental results demonstrate that discrminative VCL models tend to extract features from limited parts of an object, which is only useful particular current task, instead of diverse features from all different parts. Since the classifier is built on concentrated features, independence assumption in VCL is prone to errors as training in the subsequent tasks might make the model attend to other features which are not considered in the earlier tasks. These drawbacks of VCL motivate us to have a valid posterior approximation while encouraging models to focus on more diverse features.

Fortunately, we can fullfil these two desiderata by equipping the model with data generative regularization in the training process. The generative regularization is pushing the model to catch the characteristics of all parts of the object for generatation, so that the shared component will be sufficient features and stable across all tasks. Meanwhile, with the sufficient features, we can recover the independent condition as we discusse in Section 3.2. Our contributions can be summarised as follows:

- we analyze Bayesian approach in the continual learning setup and point out a deficiency of the parameter independence assumption;
- we propose to use energy-based model with Langevin dynamic sampling as an implicit regularization term in training discriminative task;
- we empirically show that the proposed variational learning with generative regularization works well on all benchmark datasets we consider.

2. Related Work

Continual learning by regularization. There are a rich body of methods solving catastrophic forgetting problem by introducing different regularizations. EWC (Kirkpatrick et al., 2017) aims to minimize the change of weights that are important to previous tasks through the estimation of diagonal empirical fisher information matrix. SI (Zenke et al., 2017) proposes to alleviate catastrophic forgetting by allowing individual synapse to estimate their importance for solving learned tasks, then penalizing changes on the important weights. IMM (Lee et al., 2017) trains individual models on each task and then carries out a second stage of training to combine them. VCL (Nguyen et al., 2018) takes a Bayesian point of view to model a sequential learning procedure. This line of research assumes a memoryless (i.e., no stored old-data) and fixed model (i.e., model architecture cannot be adjusted during training) setup to study inherent causes of catastrophic forgetting. Our work falls in this line of research, but is derived in a principled way, and we mainly compare our algorithm with state-of-the-art methods in this category.

Continual learning by model adaption. Another class of methods addresses the continual learning problem by allowing the model to expand its capacity, while keeping the parameters used to solve previous tasks fixed. PathNet (Fernando et al., 2017) selects paths between predefined modules, and tuning is allowed only when an unused module is selected. Dynamically expandable networks (DEN) (Yoon et al., 2018) selects whether to expand or duplicate layers based on certain criteria for an incoming new task. Similarly, Rusu et al. (2016) tries to leverage the strategies adopted in progressive networks to heal forgetting. Following this line of research, (Li et al., 2019) proposed to solve the continual learning by explicitly taking into account continual structure optimization via differentiable neural architecture search. Our main goal is to study the catastrophic forgetting problem given the constraint that the structure of underlying model is fixed, while this category is out of our consideration.

Memory-based approaches and generative models. Previous works also try to alleviate catastrophic forgetting by introducing memory systems which store previous data and replay the stored old examples with the new data (Farquhar & Gal, 2019; Li et al., 2018; Lopez-Paz & Ranzato, 2017; Rebuffi et al., 2017; Robins, 1995). Specifically, these approaches require to keep either a coreset data or a generative model to replay previous tasks in order to leverage the distilled knowledge from previous tasks (Castro et al., 2018; Hou et al., 2018; Javed & Shafait, 2018; Li & Hoiem, 2017; Rebuffi et al., 2017; Shin et al., 2017; Wu et al., 2019). In practice, these methods alleviate the forgetting phenomena if enough old-data recorded, but it will increase the data usage Since our method is related to generative models, we will also compare to one of the representative algorithms, variational generative replay (VGR) (Farquhar & Gal, 2019).

Energy-based model. Our method is partly based on applying energy-based models (EBMs). We refer readers to (LeCun et al., 2006) for a more comprehensive review. The

primary difficulty in training EBMs comes from estimation of the partition function. Our work follows the derivation in (Dai et al., 2019). We notice that some concurrent works have also pointed out the importance of generative capability in the training process (Du & Mordatch, 2019; Grathwohl et al., 2019), the motivation behind these works differ from us and their focus is not in overcoming catastrophic forgetting. Furthermore, empirical results showed that using only EBMs could not achieve the best performance. The proposed integration of Bayesian framework and generative capability significantly outperforms EBM alone.

3. Methods

In this section, we first clarify the problem setting in Section 3.1. Then, we analyze the drawback of the posterior approximation used in the original VCL (Nguyen et al., 2018), which motivates the generalization regularizer in Section 3.2. After providing a brief introduction to EBM in Section 3.3, we design the generation-regularized Bayesian EBM to combat catastrophic forgetting in Section 3.4.

3.1. Problem Statement

A given classification model M, with a set of parameters denoted as θ, consists of parameters shared across all tasks θ_S and parameters dedicated to specific tasks θ_t. Sequential tasks are denoted as $D_1, D_2, \ldots, D_T$, where each $D_t = (X_t, Y_t)$ defines a classification task with observations X_t and labels Y_t. In the canonical setup (Kirkpatrick et al., 2017; Nguyen et al., 2018), for each task t, only one dataset D_t can be used and all previous datasets $D_1, \ldots, D_{t-1}$ cannot be accessed. The goal of our work is to achieve good classification accuracy on each task after observing all T tasks. In addition, we do not allow the algorithm to change the pre-defined structure of the model M or introduce additional parameters in shared networks. An illustration of the problem formulation is shown in Figure 1.

3.2. Motivations

We first explain why Bayesian method is a good candidate to resolve the forgetting problem, and then point out what lacks in existing literature, which motives the EBM view with generalization regularization.

Following (Nguyen et al., 2018), we assume some prior distribution of model parameters $p_0(\theta)$ (e.g., $p_0(\theta)$ follows normal distribution). According to Bayes' rule, the posterior distribution after observing T datasets can be written as:

$$\begin{aligned} p(\theta|D_{1:T}) &\propto p(\theta)\prod_{t=1}^{T} p(D_t|\theta) \propto \Big(p(\theta)\prod_{t=1}^{T-1} p(D_t|\theta)\Big)p(D_T|\theta) \\ &= \Big(p(\theta)p(D_{1:T-1}|\theta)\Big)p(D_T|\theta) \\ &\propto p(\theta|D_{1:T-1})p(D_T|\theta). \end{aligned}$$

Therefore, we can see that if we have a good posterior approximation of previous tasks $p(\theta|D_{1:T-1})$, by Bayesian approach we can combine $p(\theta|D_{1:T-1})$ and likelihood of the current task $p(D_T|\theta)$ to obtain the posterior of model parameters $p(\theta|D_{1:T})$ that work well for all tasks. The above decomposition paves a natural way for Bayesian method to handle the continual learning setup. In general the posterior is intractable, however, we can approximate the true posterior $p(\theta|D_{1:t})$ of each task t by KL-divergence via variational inference, such that $\forall t = 1, 2, \ldots, T$,

$$q_t(\theta) = \operatorname*{argmin}_{q\in\mathbf{Q}} KL\big(q(\theta)\|\frac{1}{Z_t}q_{t-1}(\theta)p(D_t|\theta)\big),$$

where $q_t(\theta)$ and $q_{t-1}(\theta)$ are the approximated posterior up to timestamp $t-1$ and t, $\mathbf{Q}$ is a predefined approximate posteriors set and Z_t is a normalization constant which needs not to be computed. We then apply variational method to estimate the lower bound of $P(y|\theta, x)$ and arrive the following training loss for each task t:

$$\textstyle\sum_{n=1}^{B_t} \mathbb{E}_{\theta\sim q_t(\theta)}[-\log p(y_{t,n}|\theta, x_{t,n})] + KL\big(q_t(\theta)\|q_{t-1}(\theta)\big), \quad (1)$$

where B_t denotes the dataset size of task t. One can parametrize $p(y_{t,n}|\theta, x_{t,n})$ with Gaussian distribution and softmax upon the output of neural network, for regression and classification, respectively. The parameter of posterior can be trained end-to-end via reparametrization trick (Kingma & Welling, 2014).

Despite that Bayesian method looks promising, we need to point out one important deficit in VCL (Nguyen et al., 2018). VCL assumes the shared model parameters θ_S are *independent* of the individual head network θ_t and thus the posterior function $p(\theta|D_{1:t})$ of the task t could be decomposed into:

$$p(\theta|D_{1:t}) = p(\theta_t|D_{1:t})p(\theta_S|D_{1:t}), \quad (2)$$

where $\theta = \{\theta_S, \theta_t\}$. VCL then applies Bayeisan approach on approximating $p(\theta_S|D_{1:t})$ and fix θ_t after training each task t. However, the independence assumption between θ_t and θ_S is not true in general, especially in neural network where the head parameter θ_t highly depends on the shared layers. The correct factorization of posterior function should be

$$p(\theta|D_{1:t}) = p(\theta_t|D_{1:t}; \theta_S)p(\theta_S|D_{1:t}), \quad (3)$$

where the dependence between θ_S and θ_t exists. Thus, in order to correctly apply Bayesian framework, we may use a sufficient feature for θ_S such that the equation (2) becomes a valid reduction.

We provide a sufficient condition under which the decomposition (2) is valid.

Proposition 1. *If the causal joint distribution is*

$$p(\theta_t, \theta_S, D_{1:t}) = p(\theta_S)p(D_{1:t}\theta_S)p(\theta_t|D_{1:t}),$$

we have the posterior

$$q(\theta_S, \theta_t | D_{1:t}) = p(\theta_S | D_{1:t}) p(\theta_t | D_{1:t}),$$

therefore, the decomposition (2) *becomes valid.*

Proof. The conclusion can be verified straightforwardly.

$$\begin{aligned} q(\theta_S, \theta_t | D_{1:t}) &= \frac{p(\theta_S) p(D_{1:t} | \theta_S) p(\theta_t | D_{1:t})}{p(D_{1:t})} \\ &= p(\theta_S | D_{1:t}) p(\theta_t | D_{1:t}), \end{aligned}$$

from which we obtain the conditional independence property $\theta_S \perp \theta_t | D_{1:t}$. □

The condition in Proposition 1 inspires us the generative requirement on the intermediate layer of the model (i.e., θ_S) to revalid the decomposition (2). Bringing generative power ((i.e., $p(D_{1:t}\theta_S)$) into the play could exploit variational Bayesian inference overcome forgetting better.

The remaining question is

how do we equip the underlying model with generative power without adding more number of parameters?

We answer this question by resorting to the Energy-based model (EBM). Essentially, the neural network $p(y|x, \theta)$ can be understood as a EBM, which automatically has the discriminative and generative ability, although in most of the training for supervised tasks, the generative ability is just simply ignored. From this perspective, we will complement the existing discriminative loss in the training with an additional generative loss term to ensure the causal condition in (1), and eventually guarantee the decomposition in Bayesian inference is valid. Therefore, it totally release the power of Bayesian inference to overcome catastrophic forgetting. In the following sections, we illustrate how generative power of EBM can be fit into the Bayesian method.

3.3. Energy-based Model

For any given discriminative model $f_\theta(x)$ (e.g., deep neural networks for classification tasks) parameterized by θ as

$$p(y|x) = \frac{\exp(y^\top f_\theta(x))}{Z_x}$$

with $Z_x(\theta) = \sum_{y \in \mathcal{Y}} \exp(y^\top f_\theta(x))$, it can be view as EBM with energy function $y^\top f_\theta(x)$. Then, obviously, by redefine the partition function, we obtain the joint distribution with generative ability:

$$p_\theta(x, y) = \frac{\exp(y^T f_\theta(x))}{Z(\theta)}, \tag{4}$$

where $Z(\theta) = \sum_y \int \exp(y^T f_\theta(x)) dx$. In this work, $f_\theta(\cdot)$ is a neural network parameterized by θ. We can train the joint EBM by maximum likelihood estimation:

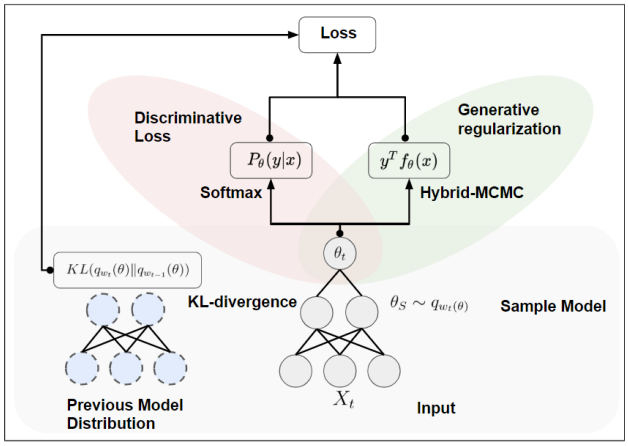

Figure 2. Illustration of the proposed method.

Algorithm 1 Gibbs-Langevin Dynamic Sampling

Input: Buffer B, storing previous sampled data
Output: Sampled data x_S, y_S and buffer B
$x_0 \sim B$
for $s = 1$ **to** S **do**
  $y_s \sim p(y|x_s)$
  $x_s = x_{s-1} + \frac{1}{2}\eta_s \nabla_x [y_s^T f(x_{s-1})] + \epsilon$,
  $\epsilon \sim N(0, \eta_s)$
  $\eta_s = \frac{1}{s}$
end for
Add x_S, y_S into B.
Return x_S, y_S, B.

$$\begin{aligned} \max_\theta p_\theta(x, y) &= \max_\theta \log p_\theta(x, y) \\ &= \max_\theta y^T f_\theta(x) - \log Z(\theta). \end{aligned} \tag{5}$$

However, directly solving MLE of general EBMs is intractable due to the log-partition function $\log Z(\theta)$. To alleviate the computation, Contrastive Divergence (CD) is proposed in (Hinton, 2002). CD estimates the gradient of the MLE of EBM as:

$$\nabla_\theta \log p_\theta(x, y) = \mathbb{E}_D \left[y^T \nabla_\theta f_\theta(x) \right] - \mathbb{E}_{p_\theta(x,y)} \left[y^T \nabla_\theta f_\theta(x) \right], \tag{6}$$

where $p_\theta(x, y)$ denotes the underlying distribution from EBM. The second term $\mathbb{E}_{p_\theta(x,y)} \left[y^T \nabla_\theta f_\theta(x) \right]$ can be calculated as firstly sample (batch of) data x_t, y_t by using Langevin dynamic sampling shown in Algorithm 1, and then calculate the $y_t^T f(x_t)$ to stochastically get the estimated value of $\mathbb{E}_{p_\theta(x,y)} \left[y^T \nabla_\theta f_\theta(x) \right]$.

3.4. Bayesian Inference as Learning with Generative Regularization

With the formulation of generative loss, instead of interpreting $p(D_t|\theta)$ as a discriminative model $p(y_t|x_t; \theta)$ in eq

Algorithm 2 Algorithm of Bayesian Generative Regularization (BGR) at task t.

Input: Dataset of task t D_t, Posterior distribution of previous tasks $q_{t-1}(\theta)$, Number of training epochs E and learning rate β
Output: Posterior distribution $q_t(\theta)$ of learned model
$q_t(\theta) = q_{t-1}(\theta)$
for $epoch = 1$ **to** E **do**
  $x_b, y_b \sim D_t$
  $\theta \sim q_t(\theta)$
  Generate sample x_t, y_t by Algorithm 1
  Calculate gradient $\nabla_\theta L(\theta; p_\theta)$ via Theorem 2.
  $q_t(\theta)$ = $q_t(\theta)$ - $\beta \nabla_\theta L(\theta; p_\theta)$
end for
Return $q_t(\theta)$

(1), we have $p(D_t|\theta)$ to be a generative model as $p_\theta(x, y)$. Now, instead of lower-bounding $P(y|\theta, x)$, we estimate the lower bound of $P(x, y, \theta)$ and the core training objective of task t in variational method changes from eq (1) into :

$$\min_{q_t \in \mathbf{Q}} \mathbb{E}_{q_t, D_t}\big[-\log p_\theta(x, y) + KL\big(q_t(\theta|D_{1:t}) \| q_{t-1}(\theta|D_{1:t-1})\big)\big],$$

where $\mathbf{Q}$ is the functional space of posterior distribution. For simplicity, we follow the literature to assume $\mathbf{Q}$ to represent mean-field distribution, and we generate a model parameter θ by sampling it from q_t. Recall that $p(x, y) = p(y|x)p(x)$, thus we can rewrite the objective as

$$\begin{aligned}\min_{q_t \in \mathbf{Q}} \mathbb{E}_{q_t, D_t}\big[&-(1-\lambda)\log p_\theta(y, x) - \lambda \log p_\theta(y|x) \\ &- \lambda \log p_\theta(x) + KL(q_t(\theta|D_{1:t}) \| q_{t-1}(\theta|D_{1:t-1}))\big], \quad (7)\end{aligned}$$

where the $\log p_\theta(y|x)$ can be understood as the common discriminative loss, while both $\log p_\theta(x)$ and $\log p_\theta(x, y)$ can be understood as generative regularizations that match the empirical joint distribution and marginal distribution simultaneously. Then, we apply Contrastive Divergence to optimize (7), which providing the estimation of gradient of $p_\theta(x, y)$. In fact, new objective is also related to contrastive loss (Chen et al., 2020; Dai et al., 2019), where we use the synthesis samples as negative samples. Here we give a derivation of unbiased gradient estimator of $\log p_\theta(x)$ in the following theorem.

Theorem 1. *Given a discriminative model $f_\theta(x)$, the unbiased gradient estimator of the corresponding Energy-based model term $\log p_\theta(x)$ is given by the following estimator:*

$$\nabla_\theta \log p_\theta(x) = \mathbb{E}_{p_\theta(y|x)}[y^T \nabla_\theta f_\theta(x)] - \mathbb{E}_{p_\theta(x,y)}[y^T \nabla_\theta f_\theta(x)].$$

Proof. The proof is postponed to Appendix A. □

Based on this theorem, we could obtain the derivative of the objective in eq (7) by using eq (6) and Theorem 1, and we summarize it in the following theorem.

Theorem 2. *The estimation of gradient of loss used in training the proposed method eq (7) is given by*

$$\begin{aligned}\nabla_\theta L(\theta; p_\theta) \triangleq\ & \frac{1}{\lambda} \nabla_\theta KL(q_t(\theta|D_{1:t}) \| q_{t-1}(\theta|D_{1:t-1})) \\ & - \nabla_\theta \log p_\theta(y|x) \\ & + \frac{1}{\lambda}(\mathbb{E}_{p_\theta(x,y)}\big[y^T \nabla_\theta f_\theta(x)\big] - \mathbb{E}_{D_t}[y^T \nabla_\theta f_\theta(x)]).\end{aligned}$$

Proof. The proof is postponed to Appendix B. □

The overall illustration of losses used in this work is summarized in Figure 2. The first term of the gradient estimation corresponds to the weighted KL-divergence between posterior approximation of task t and $t-1$. The second term is the common NLL loss used in training deep neural networks. The calculation of these two terms corresponds to the gradient of forward neural network computation, and thus it could be obtained by back-propagation of underlying model f_θ. The rest two terms correspond to the weighted generative capability. In this paper, we treat the generative term as a regularization term. λ represents the importance balance of the generative regularization and discriminative NLL loss. We named the proposed method **Bayesian Generative Regularization**, and the overall algorithm is summarized in Algorithm 2.

4. Experiments

4.1. Datasets

We evaluate the proposed method on four datasets.

Permuted-MNIST Permuted-MNIST is a very popular benchmark dataset in the continual learning literature. The dataset received at each time step D_t consists of labeled MNIST images whose pixels have undergone a fixed random permutation.

Split-MNIST This experiment was used by (Zenke et al., 2017). Five binary classification tasks from the MNIST dataset arrive in sequence: 0/1, 2/3, 4/5, 6/7, and 8/9.

Permuted-MNIST: The dataset received at each time step D_t consists of labeled MNIST images whose pixels have undergone a fixed random permutation. **Split-MNIST:** This experiment was used by (Zenke et al., 2017). Five binary classification tasks from the MNIST dataset arrive in sequence: 0/1, 2/3, 4/5, 6/7, and 8/9.

Fashion-MNIST Fashion-MNIST (Xiao et al., 2017), similar to MNIST dataset, consists of a training set of 60,000 examples and a test set of 10,000 examples. Each example is a 28 x 28 grayscale image, associated with a label from

10 classes. This dataset represents more realistic features of real-world images and thus becomes an increasingly popular benchmark. For this task, we follow the Split-MNIST setup to split the classes into sequence: 0/1 (T-shirt/Trouser), 2/3 (Pullover/Dress), 4/5 (Coat/Sandal), 6/7 (Shirt/Sneaker), and 8/9 (Bag/Ankle boot).

CUB To further validate the proposed method could work on real-world color images, we perform experiments on Caltech-UCSD Birds (CUB) dataset. CUB is an image dataset with photos of 200 bird species. We select top 100 classes with more training images and then split thsese 100 classes into 10 continual learning tasks randomly. Each task consists of 5 binary classification in order. Detailed processing of the dataset is described in the supplementary.

4.2. Baseline Methods

We compare our method to the following baseline methods.

- SGD: simply trains each task in an incremental setup without any regularization. It serves as the bottom line of all the methods.
- All-data: trains the tasks jointly assuming all datasets are available. At each step, a random dataset is sampled and then a batch of data is sampled from the dataset. It serves as the upper bound and indicates the difficulty of the classification task.
- EWC (Kirkpatrick et al., 2017): builds the importance estimation on top of diagonal Laplace propagation by calculating the diagonal of empirical Fisher information.
- VCL (Nguyen et al., 2018; Swaroop et al., 2019): conducts variational inference from Bayesian point of view of continual learning. VCL is reported as the most competitive method under our problem setup. In particular, we implement the improved version of VCL (Swaroop et al., 2019).
- VGR (Farquhar & Gal, 2019): extends VCL by augmenting a GAN generative model to record the replay data.

Detailed processing of the dataset, implementation of the baseline methods and hyperparameters of the proposed method are described in the supplementary.

4.3. Results and Analysis

The evaluation metric used is average classification accuracy over all observed tasks. We first summarize accuracy of each method after observing all tasks in Table 1. Our proposed method is named **Bayesian Generative Regularization (BGR)**. We notice that "All-data" achieves high accuracy for almost all datasets. Accuracy on CUB drops a bit as there are certain species of birds which are difficult to classify it correctly. This shows that all classification tasks are not difficult when all data are provided. The challenges are indeed faced when continual learning setup comes in and causes forgetting. In the table, we see that BGR outperforms baselines in all tasks. In particular, the improvement is significant on Fashion-MNIST and CUB dataset which contain more real-world alike objects. BGR increase about 15% accuracy in Fashion-MNIST and 10% in CUB datasets.

	Permuted	Split	Fashion	CUB
All-data	99.3	99.1	99.3	88.3
SGD	37	90	74.6	65.2
EWC	87.5	97.4	82.2	67.2
VCL	92.3	**98.2**	76.9	67.4
VGR	70.5	97.7	86.2	76.8
BGR	**92.7**	**98.2**	**97.2**	**78.8**

Table 1. Summarization of overall performance on continual learning tasks. Results shown in the table are average classification accuracy (in %) of each task.

	Fashion	CUB
SGD	74.6	65.2
All-data	99.3	91.0
GEN	87.9	74.0
GEN-L2	90.9	72.8
VCL	76.9	67.4
BGR	**97.2**	**78.8**

Table 2. Ablation study of overall performance on Fashion-MNIST and CUB datasets. Results shown in the table are average classification accuracy (in %) of each task.

In addition to accuracy after observing all tasks, we are also interested in individual performance after observing each new incoming task. Average classification accuracy of each time step of Permuted-MNIST and Split-MNIST are shown in Figure 3 and Figure 4. We can observe that despite the performance of SGD-only training drops abruptly, all other methods performs relatively steady over all time steps, and BGR stands out in the later time steps. For real-world objects as Fashion-MNIST and CUB, results are shown in Figure 5 and Figure 6. These two tasks contain more difficult classification tasks and thus it's more challenging when posed as continual learning setup. The difficulty of each task might be very different hence the accuracy fluctuates. Consequently, the curve won't be as smooth as previous two datasets. Nevertheless, we again observe that BGR has a relatively steady performance over the baseline methods.

We also want to point out that since BGR contains generative capability, indeed we could sample images from the trained model. However, our main focus is not generative model but overcoming forgetting, so the generated images might not

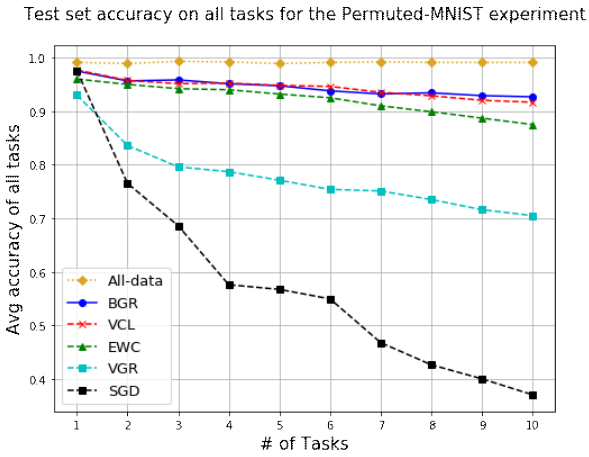

Figure 3. Detailed Classification Results of Permuted-MNIST.

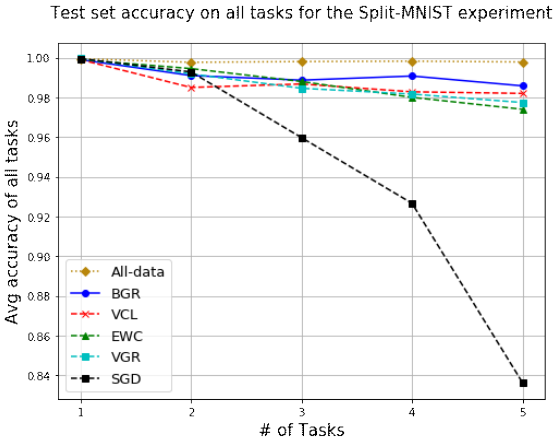

Figure 4. Detailed Classification Results of Split-MNIST.

be realistic as the state-of-the-art generative models. The generative power used rather as a regularization to make the model robust to continual learning setup. We have attached some examples of generated images of MNIST and Fashion-MNISt dataset in the supplementary.

4.4. Ablation Study

Despite we have validated the performance of the proposed method, we are not sure if the gain comes from generative regularization, Bayesian method or indeed the better estimation of the posterior by combining two approaches. Therefore, we conduct ablation analysis on Fashion-MNIST and CUB to verify the importance of each component. Notice that BGR without the generative regularization would simply become the VCL method. To test the generative component without Bayesian framework, we will remove model parameter sampling procedure and KL-divergence term. This leads to normal training of the classifier with

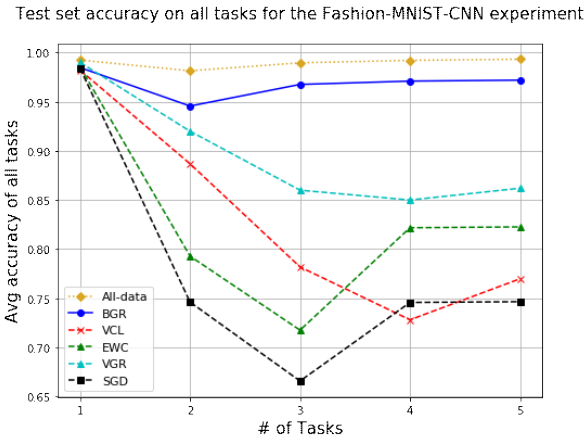

Figure 5. Detailed Classification Results of Fashion-MNIST.

NLL loss plus the generative loss from EBM. We denote this setup as GEN. We also try to apply the GEN with L2 regularization which resembles the KL divergence term in our formulation. We denote such method as GEN-L2. Results of all the methods are summarized in Table 2.

From Table 2, we could observe that generative term itself is very useful to overcome the catastrophic forgetting. Compared to the performance of VCL, GEN could achieve more than 5% performance gain on CUB, and more than 10% on Fashion-MNIST. Generative capability indeed provides a more robust model in continual learning setup and this validates our initial intuition that knowing the complete formulation of the object would make model perform better. On the other hand, results show that adding L2 regularization on top of generative term is not necessarily helpful. Even when it's effective, the performance gain is rather limited.

However, we also notice that the generative term alone cannot reach a performance comparable to the proposed method. Furthermore, we could observe the synergy effect that the sum of the performance gain from VCL and GEN together could not reach the performance of BGR. This implies that in BGR, Bayesian framework and generative term are not working independently. Generative capability implicitly helps to capture the relationship between θ_t and θ_S better with more diverse feature. So when two approaches are combined, we could get an approximation of posterior $p(\theta|D_{1:t})$ with more information on the object without introducing more model or data complexity.

4.5. Comparison to Generative Model

It is also important to compare BGR and VGR, which is a representative algorithm using generative models, in details since both are generative models. Results are shown in Figure 7. The performance on CUB and Split-MNIST are

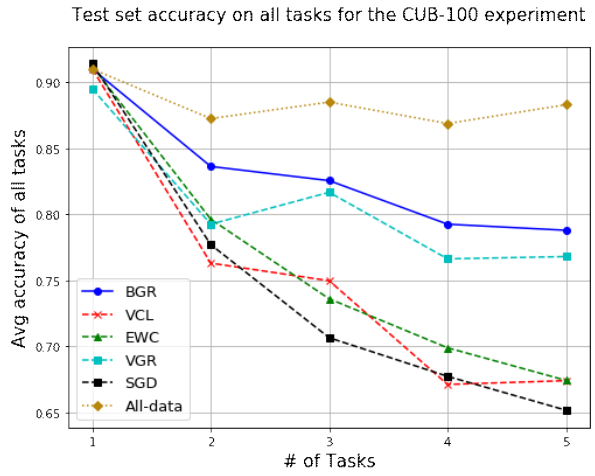

Figure 6. Detailed Classification Results of CUB.

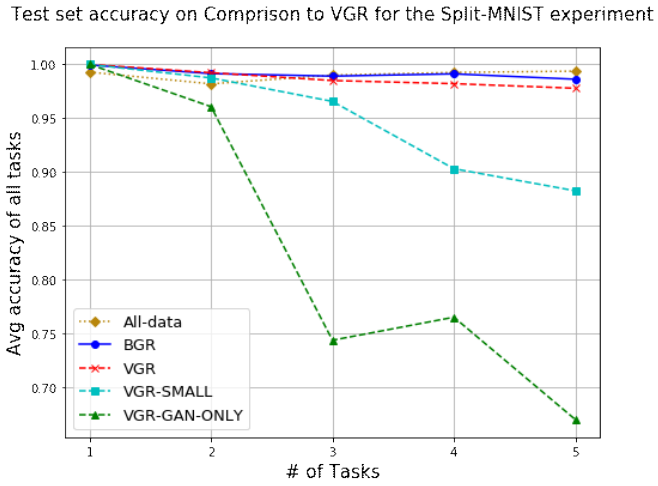

Figure 7. Detailed Comparison of VGR on Split-MNIST.

similar. The biggest difference is that for VGR to work, it needs to use GAN with much larer parameters. BGR uses only 220k parameters on Split-MNIST and if we choose GAN in VGR with similar size, the perform drops to 88 (VGR-SMALL). Also notice that in VGR paper (Farquhar & Gal, 2019), authors continue to use classifier from previous task. If we re-initiate the classifier every time a new task comes in and train the model with replayed data, then the performance drops to 66 (VGR-GAN-ONLY). This again shows that generative capability works better when combined with discriminative classifier as in BGR. On the other hand, the benefit of using separate generative model as in VGR is that the training speed is fast. BGR takes 3097 seconds to finish and VGR only takes 564 seconds. BGR could use smaller model size but it takes longer training time as monte carlo sampling is time-consuming.

(a) Top 20% salient points of models trained with SGD. Salient points concentrate on most discriminative part of the digits and model suffers from catastrophic forgetting. Accuracy drops from 99.7% to 54.2% after training on another task.

(b) Top 20% salient points of models trained with the proposed method. EBM provides a generative capability so the salient points scatter equally over the whole stroke of digits. Accuracy drops only from 99.7% to 95.0% after training on another task. This shows the importance of generative term in overcoming catastrophic forgetting.

Figure 8. Illustration of importance of learning diverse features by proposed generative term in the model.

4.6. Comparison to Memory-based Methods

Our work focused on memory-less setup to study how generative capability can empower the model to overcome catastrophic forgetting. However, many large-scale catastrophic forgetting problems are solved by memory-based solutions. Memory-based method allows the algorithm store some data from previous tasks and re-use the data to fine-tune the model in the later stages. It's thus hard to directly compare BGR with memory-based methods since the underlying assumptions are different. However, we provide some results on MNIST and FashionMNIST for the completeness, and this also gives reader a change to observe the trade-off between memory and memory-less methods. We run the code of GDumb method (Prabhu et al., 2020)[1], and found out that indeed it achieves better MNIST results (98.5) over us (98.2) with memory $k = 5000$. However, it does not perform equivalently well on FashionMNIST. It requires memory $k = 7500$ to achieve similar performance (97.2) whereas BGR requires no extra storage.

[1] https://github.com/drimpossible/GDumb

4.7. Extension to Class Incremental Learning Setup

With the advance of catastrophic forgetting research, many other learning scenarios are introduced. Class incremental Learning (Prabhu et al., 2020; Farquhar & Gal, 2019), a popular setup, consider a combination of single-head and multi-head setup. It's a single-head setup but each time we will only be given a subset of classes as in multi-head setup. In (Farquhar & Gal, 2019), authors pointed out that this task is challenging for methods without memory systems. Indeed, directly applying the proposed method cannot yield a good result. However, in this section we show that the proposed BGR can be combined with memory-based methods to achieve better results. Specifically, we consider a combination of BGR with an meta-learning method iTAML (Rajasegaran et al., 2020). Under the original learning scheme in iTAML, we additionally add the generative regularization term into the training objective, and observe that on splitMNIST dataset, the performance improves from 97.8 (iTAML) to 98.4 (iTAML+Ours), which shows that BGR can indeed be used as a regularization on top of state-of-the-art memory-based methods. Further study of how the proposed BGR could be combined with other methods is an interesting future direction.

4.8. Qualitative Analysis: Generative Capability Learns Diverse Features

Since we add the generative regularization in the objective function, we are interested in what types of features we learn in the proposed method. In order to generate an object, generative model should not only recognize certain parts of the object but also capture most variations of it. Thus, we hypothesize that generative capability learn a more diverse feature instead of concentrated discriminative features. Such holistic feature capture should prevent model from focusing on only part of object and alleviate the catastrophic forgetting. To get a qualitative assessment, we performed the Integrated Gradients method (Sundararajan et al., 2017) to investigate which pixels of the image contribute most to the output of the model[2]. The salient points are marked with red dots. These points represent the part of the object wit strongest response to the feature extraction process (i.e., activations in the neural network).

As shown in Figure 8a, instead of understanding full stroke of the drawing, training the pair discriminatively with NLL loss makes the model focusing on certain part of the underlying object. Specifically, salient points of the digit 7 are spread mostly on top horizontal stroke, and salient points of the digit 9 centered on lower left curved stroke. Admittedly, these salient points mark the most critical difference between the shape of these two digits so discriminative models can exploit such informative feature to succeed in classification task. However, since not many features are extracted in the first task, when the model moves to the next task, the discriminative model might focus on a very different set of features such that minor adjustment of model parameters will cause the desired feature extraction in task 1 lost. On the other hand, as shown in the in Figure 8b, salient points of BGR with generative capability will be be equally distributed to different parts of the stroke. And the accuracy drops only from 99.7% to 95.0% after training on another task. This shows the importance of generative term in overcoming catastrophic forgetting.

[2] https://github.com/chihkuanyeh/saliency_evaluation

5. Conclusions

In this paper, we use Energy-based Model to provide the generative loss as a regularization term in order to overcome the catastrophic forgetting. Energy-based model with hybrid monte carlo sampling process can equip the underlying model with the generative capability. Experimental results show that when generative capability is combined with Bayesian inference framework, it can alleviate catastrophic forgetting significantly without modifying underlying model architecture. The proposed BGR outperforms state-of-the-art method on Fashion-MNIST dataset about 15% accuracy and CUB dataset about 10%.

Acknowledgement

Part of this work was done during PHC's internship at Google. CJH and PHC are partially supported by NSF under IIS-1901527, IIS-2008173 and IIS-2048280.

References

Castro, F. M., Marín-Jiménez, M. J., Guil, N., Schmid, C., and Alahari, K. End-to-end incremental learning. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 233–248, 2018.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pp. 1597–1607. PMLR, 2020.

Dai, B., Liu, Z., Dai, H., He, N., Gretton, A., Song, L., and Schuurmans, D. Exponential family estimation via adversarial dynamics embedding. *arXiv preprint arXiv:1904.12083*, 2019.

Du, Y. and Mordatch, I. Implicit generation and generalization in energy-based models. *arXiv preprint arXiv:1903.08689*, 2019.

Farquhar, S. and Gal, Y. A unifying bayesian view of continual learning. *arXiv preprint arXiv:1902.06494*, 2019.

Fernando, C., Banarse, D., Blundell, C., Zwols, Y., Ha, D., Rusu, A. A., Pritzel, A., and Wierstra, D. Pathnet: Evolution channels gradient descent in super neural networks. *CoRR*, abs/1701.08734, 2017. URL http://arxiv.org/abs/1701.08734.

Grathwohl, W., Wang, K.-C., Jacobsen, J.-H., Duvenaud, D., Norouzi, M., and Swersky, K. Your classifier is secretly an energy based model and you should treat it like one. *arXiv preprint arXiv:1912.03263*, 2019.

Hinton, G. E. Training products of experts by minimizing contrastive divergence. *Neural computation*, 14(8):1771–1800, 2002.

Hou, S., Pan, X., Change Loy, C., Wang, Z., and Lin, D. Lifelong learning via progressive distillation and retrospection. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 437–452, 2018.

Javed, K. and Shafait, F. Revisiting distillation and incremental classifier learning. In *Asian Conference on Computer Vision*, pp. 3–17. Springer, 2018.

Kingma, D. P. and Welling, M. Stochastic gradient vb and the variational auto-encoder. In *Second International Conference on Learning Representations, ICLR*, volume 19, 2014.

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*, 114(13):3521–3526, 2017.

LeCun, Y., Chopra, S., Hadsell, R., Ranzato, M., and Huang, F. A tutorial on energy-based learning. *Predicting structured data*, 1(0), 2006.

Lee, S., Kim, J., Ha, J., and Zhang, B. Overcoming catastrophic forgetting by incremental moment matching. *CoRR*, abs/1703.08475, 2017. URL http://arxiv.org/abs/1703.08475.

Li, X., Zhou, Y., Wu, T., Socher, R., and Xiong, C. Learn to grow: A continual structure learning framework for overcoming catastrophic forgetting. *CoRR*, abs/1904.00310, 2019. URL http://arxiv.org/abs/1904.00310.

Li, Y., Li, Z., Ding, L., Pan, Y., Huang, C., Hu, Y., Chen, W., and Gao, X. Supportnet: solving catastrophic forgetting in class incremental learning with support data. *arXiv preprint arXiv:1806.02942*, 2018.

Li, Z. and Hoiem, D. Learning without forgetting. *IEEE transactions on pattern analysis and machine intelligence*, 40(12):2935–2947, 2017.

Lopez-Paz, D. and Ranzato, M. Gradient episodic memory for continual learning. In *Advances in Neural Information Processing Systems*, pp. 6467–6476, 2017.

McCloskey, M. and Cohen, N. J. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of learning and motivation*, volume 24, pp. 109–165. Elsevier, 1989.

Neal, R. M. et al. Mcmc using hamiltonian dynamics.

Nguyen, C. V., Li, Y., Bui, T. D., and Turner, R. E. Variational continual learning. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=BkQqq0gRb.

Prabhu, A., Torr, P. H., and Dokania, P. K. Gdumb: A simple approach that questions our progress in continual learning. In *European Conference on Computer Vision*, pp. 524–540. Springer, 2020.

Rajasegaran, J., Khan, S., Hayat, M., Khan, F. S., and Shah, M. itaml: An incremental task-agnostic meta-learning approach. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13588–13597, 2020.

Ratcliff, R. Connectionist models of recognition memory: constraints imposed by learning and forgetting functions. *Psychological review*, 97(2):285, 1990.

Rebuffi, S.-A., Kolesnikov, A., Sperl, G., and Lampert, C. H. icarl: Incremental classifier and representation learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pp. 2001–2010, 2017.

Robins, A. Catastrophic forgetting, rehearsal and pseudorehearsal. *Connection Science*, 7(2):123–146, 1995.

Rusu, A. A., Rabinowitz, N. C., Desjardins, G., Soyer, H., Kirkpatrick, J., Kavukcuoglu, K., Pascanu, R., and Hadsell, R. Progressive neural networks. *CoRR*, abs/1606.04671, 2016. URL http://arxiv.org/abs/1606.04671.

Shin, H., Lee, J. K., Kim, J., and Kim, J. Continual learning with deep generative replay. In *Advances in Neural Information Processing Systems*, pp. 2990–2999, 2017.

Smola, A. J., Vishwanathan, V., and Eskin, E. Laplace propagation. In *NIPS*, pp. 441–448, 2003.

Sundararajan, M., Taly, A., and Yan, Q. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 3319–3328. JMLR. org, 2017.

Sutton, R. S., Whitehead, S. D., et al. Online learning with random representations. In *Proceedings of the Tenth International Conference on Machine Learning*, pp. 314–321, 2014.

Swaroop, S., Nguyen, C. V., Bui, T. D., and Turner, R. E. Improving and understanding variational continual learning. *arXiv preprint arXiv:1905.02099*, 2019.

Voigt, P. and Von dem Bussche, A. The eu general data protection regulation (gdpr). *A Practical Guide, 1st Ed., Cham: Springer International Publishing*, 2017.

Wu, Y., Chen, Y., Wang, L., Ye, Y., Liu, Z., Guo, Y., and Fu, Y. Large scale incremental learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 374–382, 2019.

Xiao, H., Rasul, K., and Vollgraf, R. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms, 2017.

Yoon, J., Yang, E., Lee, J., and Hwang, S. J. Lifelong learning with dynamically expandable networks. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=Sk7KsfW0-.

Zagoruyko, S. and Komodakis, N. Wide residual networks. *arXiv preprint arXiv:1605.07146*, 2016.

Zenke, F., Poole, B., and Ganguli, S. Continual learning through synaptic intelligence. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, pp. 3987–3995. JMLR.org, 2017. URL http://dl.acm.org/citation.cfm?id=3305890.3306093.