Lifelong Variational Autoencoder via Online Adversarial Expansion Strategy

Fei Ye and Adrian G. Bors

Department of Computer Science, University of York, York YO10 5GH, UK
fy689@york.ac.uk, adrian.bors@york.ac.uk

Abstract

The Variational Autoencoder (VAE) suffers from a significant loss of information when trained on a non-stationary data distribution. This loss in VAE models, called catastrophic forgetting, has not been studied theoretically before. We analyse the forgetting behaviour of a VAE in continual generative modelling by developing a new lower bound on the data likelihood, which interprets the forgetting process as an increase in the probability distance between the generator's distribution and the evolved data distribution. The proposed bound shows that a VAE-based dynamic expansion model can achieve better performance if its capacity increases appropriately considering the shift in the data distribution. Based on this analysis, we propose a novel expansion criterion that aims to preserve the information diversity among the VAE components, while ensuring that it acquires more knowledge with fewer parameters. Specifically, we implement this expansion criterion from the perspective of a multi-player game and propose the Online Adversarial Expansion Strategy (OAES), which considers all previously learned components as well as the currently updated component as multiple players in a game, while an adversary model evaluates their performance. The proposed OAES can dynamically estimate the discrepancy between each player and the adversary without accessing task information. This leads to the gradual addition of new components while ensuring the knowledge diversity among all of them. We show theoretically and empirically that the proposed extension strategy can enable a VAE model to achieve the best performance given an appropriate model size.

Introduction

The Variational Autoencoder (VAE) (Kingma and Welling 2013) is one of the most popular deep generative models, which defines an encoding-decoding process for images. A VAE consists of two modules : an inference model mapping an image $\mathbf{x}$ to a low-dimensional latent variable $\mathbf{z}$ and a decoder recovering $\mathbf{x}$ from $\mathbf{z}$. The VAE is a likelihood-based model which is optimized by maximizing the sample log-likelihood $\log p_\theta(\mathbf{x}) = \log \int p_\theta(\mathbf{x}\,|\,\mathbf{z})p(\mathbf{z})\mathrm{d}\mathbf{z}$. However, this objective function is intractable for optimization since it requires integrating over all $\mathbf{z}$. A VAE introduces a lower bound to the sample log-likelihood, called Evidence Lower Bound (ELBO), used as its objective function :

$$\begin{aligned}&\log p_\theta(\mathbf{x}) \geq \mathbb{E}_{q_\omega(\mathbf{z}|\mathbf{x})}\left[\log p_\theta\left(\mathbf{x}\,|\,\mathbf{z}\right)\right]\\&- D_{KL}\left[q_\omega\left(\mathbf{z}\,|\,\mathbf{x}\right)\,||\,p\left(\mathbf{z}\right)\right] := \mathcal{L}_{ELBO}\left(\mathbf{x};\{\theta,\omega\}\right)\,,\end{aligned} \tag{1}$$

where $p(\mathbf{z}) = \mathcal{N}(0, I)$ and $p_\theta(\mathbf{x}\,|\,\mathbf{z})$ are the prior and decoding distribution, respectively. $D_{KL}[\cdot]$ is the Kullback–Leibler divergence. Existing works aiming for the improvement of VAE are mainly deriving a tighter ELBO to the data log-likelihood, which is implemented by using importance sampling (Burda, Grosse, and Salakhutdinov 2015; Domke and Sheldon 2018), a more expressive posterior (Kim and Pavlovic 2020; Maaløe et al. 2016; Kim and Pavlovic 2020), or hierarchical variational models (Molchanov et al. 2019; Vahdat and Kautz 2020).

However, these methods can only guarantee a tight ELBO for a static data domain and do not consider the circumstances of lifelong learning. Recently, (Ye and Bors 2022f)

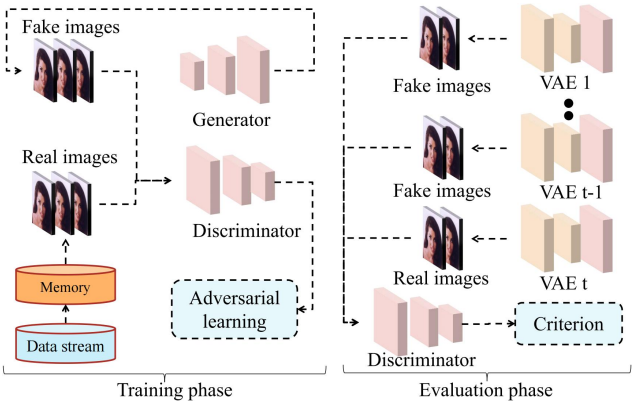


Figure 1: The scheme for the proposed Online Adversarial Expansion Strategy (OAES). We assume that a VAE-based Dynamic Expansion Model has already trained t components. At each training step, the generator, discriminator, and the current component (t) are trained on the memory buffer while all other components are frozen. At the evaluation phase, we treat the generation of all previously learnt components (1,$\cdots$,t-1) and the current component (t) as real and fake images, which are then fed into the discriminator for deciding the model's expansion.

provided for the first time a theoretical analysis for the forgetting behaviour of VAEs. However, this theoretical analysis requires two strong assumptions : a Gaussian decoder and knowing the task information. These two assumptions are not guaranteed in a more realistic continual learning scenario called Task-Free Continual Learning (TFCL) (Aljundi, Kelchtermans, and Tuytelaars 2019), where task identity is unavailable. In this paper, we focus on the realistic CL scenario and develop a novel theoretical framework for the VAE, which overcomes the limitations of the previous work (Ye and Bors 2022f). The proposed theoretical framework interprets the forgetting process of VAEs as an increase in the Jensen-Shannon divergence (JS) between the generator and evolved data distribution, which provides new insights into the forgetting behaviour of VAEs under TFCL. Furthermore, the proposed theoretical analysis shows that a VAE-based Dynamic Expansion Model (DEM), which appropriately increases its capacity when facing the data distribution shift, can significantly improve its performance. This takes place while also maintaining the knowledge diversity among components while inducing a compact model structure without sacrificing the performance (**Theorem 3**).

Inspired by the proposed theoretical analysis, we aim to learn a diverse VAE-based Dynamic Expansion Model (VAE-DEM) for TFCL. Unlike other dynamic expansion criteria (Rao et al. 2019), which recognise input shifts as expansion signals, we implement a dynamic criterion derived from a novel perspective, that of a multi-player game. More specifically, we can consider all the previously learned VAE components and the currently updated VAE component as multiple players while an adversary discriminator estimates the discrepancy between each player and the adversary, as an evaluator. In contrast to traditional adversarial learning (Goodfellow et al. 2014), which learns a static data domain with only two players, we propose the Online Adversarial Expansion Strategy (OAES), which can learn a non-stationary data distribution and dynamically add new players during training. OAES consists of two stages, as shown in Fig. 1. In the first stage (training), we use an episode memory to store some past samples that are used to train a generator and a discriminator using adversarial learning (Eq. (14)) while training the current VAE component ('VAE t') using VAE loss (Eq. (1)) as well. In the second phase (evaluation), we treat the data generations by all players ('VAE 1',...,'VAE t-1') and by the adversary ('VAE t') as the real and fake images, which are fed into the discriminator to produce $t-1$ pairs of probability measures. We evaluate the difference between each pair of measures as the discrepancy value for each player and the adversary, which guides us to dynamically add a new adversary ('VAE t+1') and transfer ('VAE t') to the player if ('VAE t') learns sufficient novel knowledge. Such a strategy promotes information diversity among components during expansion leading to learning a compact and diverse VAE-DEM for TFCL. Extensive experiments show that OAES can significantly improve the performance of VAE-DEM for TFCL with a minimum number of components.

We summarize our contributions as follows : (1) We propose a novel theoretical framework that provides new insights into the forgetting behaviour of the VAE model under TFCL; (2) The theoretical analysis can be used in realistic continual learning scenarios without the need to know task boundaries; (3) We extend the proposed theoretical framework to analyze the forgetting behaviour of existing VAE models (**Appendix-F** from Supplemental Material (SM)); (4) Inspired by the theoretical analysis, we propose a plug-and-play dynamic extension strategy which can be used in any VAE model; (5) To the best of our knowledge, this is the first work to propose a novel solution for model expansion in TFCL from the perspective of an adversarial criterion; (6) The proposed approach achieves state of the art performance in both classification and generative tasks.

Supplementary materials (SM) and source code are available[1].

Related Work

Continual learning. A natural approach to relieve forgetting in CL is to build a memory-based replay system, which stores some past training samples from each task and replays them during the subsequent task learning (Bang et al. 2021, 2022). Memory-based approaches can further improve the performance by combining with regularization methods (Kirkpatrick et al. 2017; Kemker et al. 2018; Martens and Grosse 2015; Aljundi et al. 2019b; Chaudhry et al. 2019, 2018; Lopez-Paz and Ranzato 2017; Derakhshani et al. 2021; Shi et al. 2021; Wang et al. 2021; Nguyen et al. 2017; Ahn et al. 2019). In addition, training a generator such as a Generative Adversarial Net (GAN) (Goodfellow et al. 2014) or a Variational Autoencoder (VAE) (Kingma and Welling 2013), used to produce generative samples corresponding to past tasks, was shown to effectively relieve forgetting in CL (Ramapuram, Gregorova, and Kalousis 2020; Rao et al. 2019; Ye and Bors 2021a, 2020a, 2022f, 2023, 2022a, 2021b, 2022d, 2020b). The other approach in CL is to dynamically build new hidden layers, which would preserve the best performance for past tasks (Ye and Bors 2022c; Hung et al. 2019; Li and Hoiem 2017; Polikar et al. 2001; Rao et al. 2019; Rusu et al. 2016; Wen, Tran, and Ba 2020; Xiao et al. 2014; Ye and Bors 2020c, 2021a, 2023; Zhou, Sohn, and Lee 2012).

Task-Free Continual Learning. TFCL defines a realistic situation in CL, which has attracted recently significant attention. The first work in TFCL (Aljundi, Kelchtermans, and Tuytelaars 2019) trains a classifier with a memory buffer. This approach was extended to train both VAEs and classifiers through a retrieval mechanism that selectively stores the most perturbed samples, called the Maximal Interfered Retrieval (MIR) (Aljundi et al. 2019a). The approach from (Aljundi et al. 2019b) further treats the sample selection of the memory buffer as a constrained optimization problem, called the Gradient Sample Selection (GSS). More recently, (De Lange and Tuytelaars 2021) propose a new *learner-evaluator* framework which manages a balanced memory buffer, called the Continual Prototype Evolution (CoPE). The Gradient-based Memory EDiting (GMED) (Jin et al. 2021) modifies the memorized samples such that it increases

[1]https://github.com/dtuzi123/OAES

the loss in the upcoming model updates. However, these memory-based methods are not scalable for learning infinite data streams due to their fixed memory capacity. The Dynamic Expansion Model (DEM) can solve these limitations by dynamically expanding the model's capacity to deal with incoming samples (Rao et al. 2019; Lee et al. 2020; Ye and Bors 2022e). However, the expansion mechanism in these approaches relies on the sample log-likelihood evaluation (Rao et al. 2019) or the Dirichlet process (Lee et al. 2020), which do not have theoretical guarantees.

Variational Autoencoder. The tightness of the VAE objective function (ELBO) is crucial for improving the performance of the VAE. One possible approach is to use the Importance Weighted Autoencoder (IWELBO) (Burda, Grosse, and Salakhutdinov 2015), which generates a set of weighted samples for the given input resulting in a tighter ELBO. Another approach aims to use a more informative approximate posterior distribution, such as the Normalizing Flows (Kingma et al. 2016; Rezende and Mohamed 2015), Implicit Distributions (Mescheder, Nowozin, and Geiger 2017) or the Hierarchical Variational Inference (Huang et al. 2019). Moreover, these approaches can further improve performance by integrating the IWELBO loss into their primary objective function. In addition, online variation inference (Nguyen et al. 2017) was used in the VAE framework, but it requires to store a subset of training samples for computing the approximate posterior which is impractical for learning an unlimited number of tasks. Moreover, the study of the ELBO's tightness under TFCL has not been explored before.

Preliminary

We first introduce the learning setting of TFCL and then the probabilistic representation of each data domain.

Definition 1 ***(The stream of data samples.)*** *Let us define $D_k^S = \{\mathbf{x}_j^S\}_{j=1}^{n(S,k)}$ and $D_k^T = \{\mathbf{x}_j^T\}_{j=1}^{n(T,k)}$ be the unlabelled training and testing sets of the k-th domain/dataset, where $\mathbf{x}_j^S$ is the unlabelled data sample. $n(S,k)$ and $n(T,k)$ represent the total number of samples for D_k^S and D_k^T, respectively. D_k^S can be divided into $C(S,k)$ parts according to the category or the task information, expressed as $\{D^S(1,k), \cdots, D^S(C(S,k),k)\}$. In TFCL, where there are no task labels, each $D^S(i,k)$ is represented by its probabilistic representation $\mathbb{P}^S_{(i,k)}$. Let us define a data stream in a class-incremental manner :*

$$W = \bigcup_{j=1}^{C(S,k)} D^S(j,k)\,, \tag{2}$$

At the i-th training step ($\mathcal{T}_i$), the model only accesses a data batch $\mathcal{B}_i \in D_k^S$ with the batch size of 10. Once the model finishes all training steps, its performance is evaluated on the testing set $\mathcal{D}_k^T$.

In the following we define a VAE model and a fixed-length memory buffer which continually stores training samples from W during training.

Definition 2 ***(VAE model.)*** *Let $\mathcal{V}^i$ be a single model updated at $\mathcal{T}_i$, where i represents the index of the training step. $\mathcal{V}^i$ consists of an inference model $q_{\omega^i}(\mathbf{z}\,|\,\mathbf{x})$ and the generator $p_{\theta^i}(\mathbf{x}\,|\,\mathbf{z})$. Let $\mathbb{P}_{\theta^i}$ represent the generator's distribution.*

Definition 3 ***(Memory buffer.)*** *Let $\mathcal{M}_i$ denote a memory buffer updated at $\mathcal{T}_i$. Let $|\mathcal{M}_i|$ and $|\mathcal{M}|^{max}$ represent the number of memorized samples and the maximum memory size, respectively. Let $\mathbb{P}_{\mathcal{M}_i}$ denote the distribution of the memory buffer $\mathcal{M}_i$ updated at $\mathcal{T}_i$.*

Theoretical Framework

Forgetting Analysis of A Single VAE Model

The VAE is a likelihood model and a higher log-likelihood estimation indicates a good performance of the VAE (Chen et al. 2018), which, however, is only evaluated on a static distribution. In this section, we develop a novel lower bound to the sample log-likelihood, which can be used to analyze the VAE's performance on a non-stationary data distribution.

Theorem 1 *Let $p_{\theta^i}(\mathbf{x})$ be a probability density function for a single model $\mathcal{V}^i$ updated at $\mathcal{T}_i$. Let $\mathbb{P}_i^W$ denote a distribution of all visited data batches $\{\mathcal{B}_1, \cdots, \mathcal{B}_i\}$ drawn from W at $\mathcal{T}_i$. Let $p_{\mathcal{M}_i}(\mathbf{x})$ and $p_{W^i}(\mathbf{x})$ denote the density functions for $\mathbb{P}_{\mathcal{M}_i}$ and $\mathbb{P}_i^W$, respectively. We then derive a lower bound for a single VAE model trained on $\mathcal{M}_i$ at $\mathcal{T}_i$ as :*

$$\begin{aligned}\mathbb{E}_{\mathbb{P}_i^W}\big[\log p_{\theta^i}(\mathbf{x})\big] \geq \mathbb{E}_{\mathbb{P}_{\mathcal{M}_i}}\big[\log p_{\theta^i}(\mathbf{x})\big] - D_{JS}\big(\mathbb{P}_i^W \,||\, \mathbb{P}_{\mathcal{M}_i}\big) \\ - \mathcal{F}_{\mathrm{DL}}\big(\mathbb{P}_{\mathcal{M}_i}, \mathbb{P}_i^W, \mathbb{P}_{\theta^i}\big) + \mathcal{F}_{\mathrm{dis}}\big(\mathbb{P}_i^W, \mathbb{P}_{\mathcal{M}_i}\big)\,,\end{aligned} \tag{3}$$

where we have :

$$\begin{aligned}\mathcal{F}_{\mathrm{DL}}\big(\mathbb{P}_{\mathcal{M}_i}, \mathbb{P}_i^W, \mathbb{P}_{\theta^i}\big) \triangleq \big|D_{KL}\big(\mathbb{P}_{\mathcal{M}_i} \,||\, \mathbb{P}_{\theta^i}\big) \\ - D_{KL}\big(\mathbb{P}_i^W \,||\, \mathbb{P}_{\theta^i}\big)\big|\,,\end{aligned} \tag{4}$$

$$\begin{aligned}\mathcal{F}_{\mathrm{dis}}\big(\mathbb{P}_i^W, \mathbb{P}_{\mathcal{M}_i}\big) \triangleq \mathbb{E}_{\mathbb{P}_I^W}\big[p_{W^i}(\mathbf{x}) \log p_{W^i}(\mathbf{x})\big] \\ - \mathbb{E}_{\mathbb{P}_{\mathcal{M}_i}}\big[p_{\mathcal{M}_i}(\mathbf{x}) \log p_{\mathcal{M}_i}(\mathbf{x})\big]\,.\end{aligned} \tag{5}$$

We can observe that $\mathcal{F}_{\mathrm{dis}}(\mathbb{P}_i^W, \mathbb{P}_{\mathcal{M}_i})$ is constant if and only if $\mathbb{P}_i^W$ and $\mathbb{P}_{\mathcal{M}_i}$ are fixed. $\mathcal{F}_{\mathrm{dis}}(\mathbb{P}_{W^i}, \mathbb{P}_{\mathcal{M}_i})$ is bounded by $|D_{KL}(\mathbb{P}_i^W \,||\, \mathbb{P}_{\mathcal{M}_i}) - D_{KL}(\mathbb{P}_{\mathcal{M}_i} \,||\, (\mathbb{P}_i^W)|$. From Eq. (3)*, we can estimate the sample log-likelihood of $\mathbb{P}_i^W$ by ELBO :*

$$\begin{aligned}\mathbb{E}_{\mathbb{P}_i^W}\big[\log p_{\theta^i}(\mathbf{x})\big] \geq \mathbb{E}_{\mathbb{P}_{\mathcal{M}_i}}\big[\mathcal{L}_{ELBO}(\mathbf{x}; \theta^i, \omega^i)\big] \\ - D_{JS}\big(\mathbb{P}_i^W \,||\, \mathbb{P}_{\mathcal{M}_i}\big) - \mathcal{F}_{\mathrm{DL}}\big(\mathbb{P}_{\mathcal{M}_i}, \mathbb{P}_i^W, \mathbb{P}_{\theta^i}\big) \\ + \mathcal{F}_{\mathrm{dis}}\big(\mathbb{P}_i^W, \mathbb{P}_{\mathcal{M}_i}\big)\,,\end{aligned} \tag{6}$$

where D_{JS} is the Jensen-Shannon divergence. We then find that Eq. (6) *can be recovered to a standard ELBO (Eq.* (1)*) if and only if $\mathbb{P}_i^W$ is equal to $\mathbb{P}_{\mathcal{M}_i}$. We provide the proof in* ***Appendix-B*** *from SM*[1].

Remark. We have several observations from Theorem 1 : (1) The term $D_{JS}(\mathbb{P}_i^W \,||\, \mathbb{P}_{\mathcal{M}_i})$ in Eq. (6) plays an important role for the generalization performance. Reducing $D_{JS}(\mathbb{P}_i^W \,||\, \mathbb{P}_{\mathcal{M}_i})$ would lead to increasing the right-hand-side (RHS) of Eq. (6) and therefore the model $\mathcal{V}^i$ can have a good performance on $\mathbb{P}_i^W$. (2) A large $D_{JS}(\mathbb{P}_i^W \,||\, \mathbb{P}_{\mathcal{M}_i})$ indicates a significant reduction in the RHS of Eq. (6), resulting in a poor performance on $\mathbb{P}_i^W$. This usually happens when the memory buffer does not store sufficient information about $\mathbb{P}_i^W$, due to the forgetting process; (3) Unlike the theoretical analysis from (Ye and Bors 2022f, 2021a) which

relies on the task information, Eq. (3) can analyze the forgetting behaviour of a single VAE model without accessing any task information at each training step. In the following, we evaluate the generalization performance achieved by a single model on the target set.

Lemma 1 *Let $\{D^T(1,k), \cdots, D^T(C(T,k),k)\}$ be several target sets, where each $D^T(j,k)$ is represented by the probabilistic representation $\mathbb{P}^T_{(j,k)}$. We derive a lower bound to the sample log-likelihood for a single VAE model $\mathcal{V}^i$ at $\mathcal{T}_i$:*

$$\sum_{j=1}^{C(T,k)} \Big\{ \mathbb{E}_{\mathbb{P}^T_{(j,k)}} \big[\log p_{\theta^i}(\mathbf{x}) \big] \Big\} \geq \sum_{j=1}^{C(T,k)} \Big\{ \mathcal{F}_{\text{dis}}\big(\mathbb{P}^T_{(j,k)}, \mathbb{P}_{\mathcal{M}_i}\big) + \mathbb{E}_{\mathbb{P}_{\mathcal{M}_i}} \big[\mathcal{L}_{ELBO}(\mathbf{x}; \theta^i, \omega^i) \big] - D_{JS}\big(\mathbb{P}^T_{(j,k)} \,||\, \mathbb{P}_{\mathcal{M}_i}\big) - \mathcal{F}_{\text{DL}}\big(\mathbb{P}_{\mathcal{M}_i}, \mathbb{P}^T_{(j,k)}, \mathbb{P}_{\theta^i}\big) \Big\}, \tag{7}$$

The proof is to sum up the the bound of all target sets according to Eq. (3).

Remark. We have several observations from Lemma 1 : (1) Eq. (7) indicates that encouraging the sample diversity in the memory would relieve forgetting by minimizing the JS divergence terms since the diversity can allow $\mathcal{M}_i$ to store the information corresponding to all target sets, empirically demonstrated in (Bang et al. 2021); (2) In practice, a single model has significant limitations when aiming to learn infinite data streams or a data stream involving more underlying data distributions ($C(T,k)$ is large). In addition, a single model would also suffer from interference between the old and newly seen samples (Lee, Goldt, and Saxe 2021) . In the following, we provide the theoretical analysis and show how the Dynamic Expansion Model (DEM) can overcome the limitations of a single model.

Forgetting Analysis of DEM

The DEM can dynamically adapt its network architecture according to the complexity of the data stream. In this section, we theoretically demonstrate that the DEM can achieve better generalization performance than a single model.

Definition 4 ***Dynamic expansion model (DEM).*** *Let us define a DEM, $\mathbf{V} = \{\mathcal{V}_1^{c_1}, \cdots, \mathcal{V}_t^{c_t}\}$ with t components where the superscript c_i denotes that the i-th component ($\mathcal{V}_i^{c_i}$) finished its training and froze at $\mathcal{T}_{c_i}$. Each component $\mathcal{V}_i^{c_i}$ has already preserved the knowledge of the memory buffer $\mathcal{M}_{c_i}$ with the parameters $\{\theta_i^{c_i}, \omega_i^{c_i}\}$.*

In the following, we derive a new lower bound to analyze the forgetting behaviour of a dynamic expansion model during the training.

Theorem 2 *Let $\mathbb{P}_i^W$ represent the distribution of all visited data batches $\{\mathcal{B}_1, \cdots, \mathcal{B}_i\}$ at $\mathcal{T}_i$ where each data batch $\mathcal{B}_j$ is denoted by the probabilistic representation $\mathbb{P}_j^{\mathcal{B}}$. Let $\mathbf{V} = \{\mathcal{V}_1^{c_1}, \cdots, \mathcal{V}_t^{c_t}\}$ be a dynamic mixture model trained on $\mathcal{M}_i$ at $\mathcal{T}_i$ where $c_t = i$. We derive a lower bound as :*

$$\sum_{j=1}^{i} \Big\{ \mathbb{E}_{\mathbb{P}_j^{\mathcal{B}}} \big[\log p_{\Theta^i}(\mathbf{x}) \big] \Big\} \geq \sum_{j=1}^{i} \Big\{ \mathcal{F}_s(\mathbb{P}_j^{\mathcal{B}}, \mathbf{V}) \Big\}, \tag{8}$$

where Θ^i represent the parameters of $\mathbf{V}$ and $\mathcal{F}_s(\cdot,\cdot)$ is the component selection function defined as :

$$\mathcal{F}_s(\mathbb{P}_j^{\mathcal{B}}, \mathbf{V}) \triangleq \arg\max_{c_1,\cdots,c_t} \Big\{ \mathbb{E}_{\mathbb{P}_{\mathcal{M}_{c_i}}} \big[\mathcal{L}_{ELBO}(\mathbf{x}; \theta_i^{c_i}, \omega_i^{c_i}) \big] - D_{JS}\big(\mathbb{P}_j^{\mathcal{B}} \,||\, \mathbb{P}_{\mathcal{M}_{c_i}}\big) - \mathcal{F}_{\text{DL}}\big(\mathbb{P}_{\mathcal{M}_{c_i}}, \mathbb{P}_j^{\mathcal{B}}, \mathbb{P}_{\theta_i^{c_i}}\big) + \mathcal{F}_{\text{dis}}\big(\mathbb{P}_j^{\mathcal{B}}, \mathbb{P}_{\mathcal{M}_{c_i}}\big) \Big\}. \tag{9}$$

Eq. (9) *can be seen as an optimal component selection function which always returns the component with the highest selectivity function value.*

Remark. We have several observations from Theorem 2 : (1) Since each component $\mathcal{V}_j^{c_j}$ preserved the information of the associated memory buffer $\mathcal{M}_{c_j}$, $\mathbf{V}$ would capture more information about W when compared with a single VAE model. (2) By increasing the number of components in $\mathbf{V}$ we improve the performance since more components capture more underlying distributions and thus increase RHS of Eq. (8); In the following, we study the generalization performance of $\mathbf{V}$ on target sets by deriving a new lower bound.

Lemma 2 *Let $\{D^T(1,k), \cdots, D^T(C(T,k),k)\}$ be several target sets where each target set $D^T(c,k)$ can be divided into several data batches $\{\mathcal{B}^T(c,1), \cdots, \mathcal{B}^T(c,n(T,c,k))\}$ where $n(T,c,k)$ is the total number of data batches for $D^T(c,k)$. Let $\mathbb{P}_T^{\mathcal{B}}(c,j)$ represent the probabilistic representation of the data batch $\mathcal{B}^T(c,j)$. We suppose that $\mathbf{V}$ has already learnt t components trained on $\mathcal{M}_i$ at $\mathcal{T}_i$. The generalization performance on all target sets, achieved by $\mathbf{V}$ at $\mathcal{T}_i$, is defined as :*

$$\sum\nolimits_{c=1}^{C(T,k)} \Big\{ \sum\nolimits_{j=1}^{n(T,c,k)} \Big\{ \mathbb{E}_{\mathbb{P}_T^{\mathcal{B}}(c,j)} \big[\log p_{\Theta^i}(\mathbf{x}) \big] \Big\} \Big\} \geq \sum\nolimits_{c=1}^{C(T,k)} \Big\{ \sum\nolimits_{j=1}^{n(T,c,k)} \Big\{ \mathcal{F}_s(\mathbb{P}_T^{\mathcal{B}}(c,j), \mathbf{V}) \Big\} \Big\}, \tag{10}$$

Similar to the conclusion of Theorem 2, increasing the number of components in $\mathbf{V}$ leads to a better generalisation performance on all target sets. In practice, we use the sample log-likelihood comparison for component selection, which would introduce additional errors compared to using the optimal component selection Eq. (9) (see details in **Appendix-C** from SM[1]). In addition, we also extend our theoretical analysis to the existing VAE models in **Appendix-F** as well as to general continual learning with explicit task boundary in **Appendix-G** from SM[1].

Theoretical Analysis for The Component Diversity

In this section, we study how component diversity can influence the trade-off between the model's complexity and generalization performance.

Assumption 1 *Let us consider that $\mathbf{V}$ has already learnt t components at $\mathcal{T}_i$. Under the optimal component selection (Eq.* (8)*), we can treat the DEM $\mathbf{V}$ as a single model that has been trained on all memory buffers $\{\mathcal{M}_{c_1}, \cdots, \mathcal{M}_{c_t}\}$. Let $\mathbb{P}_{\mathcal{M}_{c_{1:t}}}$ represent the distribution of all memories data.*

Methods	Split MNIST			Split Fashion			Split MNIST-Fashion			Cross-domain		
	Log	Memory	N	Log	Memory	N	Log	Memory	N	Log	Memory	N
VAE-reservoir	-144.17	3.0K	1	-276.60	3.0K	1	-240.02	3.0K	1	-239.42	3.0K	1
VAE-ELBO-MIR (Aljundi et al. 2019a)	-143.27	3.0K	1	-274.72	3.0K	1	-238.68	3.0K	1	-237.93	3.0K	1
VAE-ELBO-Random	-150.79	3.0K	1	-280.54	3.0K	1	-247.46	3.0K	1	-239.71	3.0K	1
LIMix (Ye and Bors 2021a)	-146.23	2.0K	30	-262.52	2.0K	30	-238.63	2.0K	30	-226.63	2.0K	30
CNDPM (Lee et al. 2020)	-120.71	2.0K	30	-257.56	2.0K	30	-236.79	2.0K	30	-218.15	2.0K	30
VAE-ELBO-OCM (Ye and Bors 2022b)	-132.07	1.6K	1	-250.74	1.6K	1	-215.62	2.0K	1	-201.31	2.0K	1
VAE-IWVAE50-OCM (Ye and Bors 2022b)	-127.11	1.6K	1	-247.90	1.6K	1	-224.34	2.0K	1	-204.35	2.0K	1
Dynamic-ELBO-OCM (Ye and Bors 2022b)	-115.89	1.6K	5	-237.69	1.8K	10	-187.49	1.9K	10	-177.29	2.0K	11
OAES-ELBO	**-103.93**	1.5K	5	**-231.10**	1.5K	10	**-171.62**	1.9K	8	**-165.29**	2.0K	11

Table 1: The log-likelihood estimation on all testing samples by using the IWVAE bound with 1000 importance samples.

Theorem 3 *Based on Assumption 1, we derive a lower bound for* $\mathbf{V}$ *on all target sets at* $\mathcal{T}_i$ *as :*

$$\sum_{j=1}^{C'}\Big\{\mathbb{E}_{\mathbb{P}^T_{(j,k)}}\big[\log p_{\Theta^i}(\mathbf{x})\big]\Big\} \geq \sum_{j=1}^{C'}\Big\{\mathcal{F}_{\text{dis}}\big(\mathbb{P}^T_{(j,k)}, \mathbb{P}_{\mathcal{M}_{c_{1:t}}}\big) + \mathbb{E}_{\mathbb{P}_{\mathcal{M}_{c_{1:t}}}}\big[\mathcal{L}_{ELBO}(\mathbf{x};\Theta^i,\Omega^i)\big] - D_{JS}\big(\mathbb{P}^T_{(j,k)} \,||\, \mathbb{P}_{\mathbb{P}_{\mathcal{M}_{c_{1:t}}}}\big) - \mathcal{F}_{\text{DL}}\big(\mathbb{P}_{\mathcal{M}_{c_{1:t}}}, \mathbb{P}^T_{(j,k)}, \mathbb{P}_{\Theta^i}\big)\Big\}, \tag{11}$$

where $C' = C(T,k)$ *and* $\mathbb{P}_{\Theta^i}$ *is the distribution of samples uniformly drawn from generators (decoders) of* $\mathbf{V}$ *at* $\mathcal{T}_i$*.*

Eq. (11) indicates that $\mathbf{V}$ can achieve an excellent performance by minimising the JS divergence terms. In practice, using a large number of components may not always ensure good performance for $\mathbf{V}$, as some components would model the same underlying data distribution and ignore other distributions (See details in **Appendix-D** from SM[1]). The diversity of knowledge among the components plays a vital role in the trade-off between the model complexity and its generalisation performance. However, existing DEM models (Rao et al. 2019; Lee et al. 2020) cannot guarantee this optimal trade-off because they do not consider the knowledge diversity during the expansion process. This inspires us to develop a novel dynamic expansion mechanism with theoretical guarantees, described in the next section.

Methodology

Based on the analysis of **Theorem 3**, we desire to train a diverse and compact VAE-DEM by ensuring the knowledge diversity among its components. In this section, we first introduce a new dynamic expansion criterion and then implement it using the proposed OAES.

Dynamic Expansion Criterion

Let us consider that $\mathbf{V}$ has already learnt t components at $\mathcal{T}_i$. According to the theoretical analysis from **Theorem 3**, we aim to promote the knowledge diversity among components, which can be realized as an optimization function :

$$\mathbb{P}_{\theta_t^{i\star}} = \arg\max_{i=c',\cdots,n}\Big\{\sum\nolimits_{u=1}^{t-1}\Big\{D_p(\mathbb{P}_{\theta_u^{c_u}} \,||\, \mathbb{P}_{\theta_t^i})\Big\}\Big\}, \tag{12}$$

where $c' = c_{t-1}+1$ and n is the total number of training steps. i is the index of the training step, beginning from $c_{t-1}+1$ (the initial index of the training step for the t-th component) to n. $D_p(\cdot\,||\,\cdot)$ is an arbitrary probability measure which can be the JS divergence or Wasserstein distance. In this paper, we employ the Wasserstein distance in Eq. (12) since it is more robust than the JS divergence (Arjovsky, Chintala, and Bottou 2017). Eq. (12) aims to find an optimal model's distribution $\mathbb{P}_{\theta_t^{i\star}}$ that maximizes the Wasserstein distance between each previously learnt model's distribution $\mathbb{P}_{\theta_u^{c_u}}$ and itself. However, directly optimizing Eq. (12) in TFCL is infeasible because it requires fulfilling all training steps. Therefore, we propose a new dynamic expansion criterion to implement the goal of Eq. (12), by involving a threshold $\beta \in [0, 20]$ at $\mathcal{T}_i$:

$$\min\Big\{D_p(\mathbb{P}_{\theta_1^{c_1}} \,||\, \mathbb{P}_{\theta_t^i}), \cdots, D_p(\mathbb{P}_{\theta_{t-1}^{c_{t-1}}} \,||\, \mathbb{P}_{\theta_t^i})\Big\} \geq \beta\,. \tag{13}$$

If Eq. (13) is satisfied, we add a new component $\mathcal{V}_{t+1}^{i+1} \in \mathbf{V}$ at the next training step ($\mathcal{T}_{i+1}$), while $\mathcal{V}_t^i$ is frozen to increase the diversity among components. The threshold β controls the trade-off between the model's complexity and its generalization performance while also avoiding passing through all training steps. When decreasing β, the model $\mathbf{V}$ tends to create more components during the training, which would improve the performance but would require more parameters. In contrast, a large threshold β would encourage $\mathbf{V}$ to use fewer components, leading to degenerated performance. The theoretical analysis for the choice of β can be found in **Appendix-E** from SM[1].

Online Adversarial Expansion Strategy (OAES)

The proposed criterion (Eq. (13)) employs the Wasserstein distance for assessing the similarity between two probability distributions. However, evaluating Wasserstein distance in the high-dimensional image space still requires enormous computational resources. To address this issue, instead of directly estimating Wasserstein distance, we formulate the expansion strategy as a multi-player game and introduce the proposed OAES to solve this game. OAES consists of two stages, as shown in Fig. 1 : training and evaluation stages..

Training stage : Let G_{ε^i} and D_{ψ^i} represent a generator and a discriminator, trained on the memory buffer $\mathcal{M}_i$ at $\mathcal{T}_i$. The

Methods	**Split MNIST**	**Split CIFAR10**	**Split CIFAR100**
finetune*	19.75 ± 0.05	18.55 ± 0.34	3.53 ± 0.04
GEM*	93.25 ± 0.36	24.13 ± 2.46	11.12 ± 2.48
iCARL*	83.95 ± 0.21	37.32 ± 2.66	10.80 ± 0.37
reservoir*	92.16 ± 0.75	42.48 ± 3.04	19.57 ± 1.79
MIR*	93.20 ± 0.36	42.80 ± 2.22	20.00 ± 0.57
GSS*	92.47 ± 0.92	38.45 ± 1.41	13.10 ± 0.94
CoPE-CE*	91.77 ± 0.87	39.73 ± 2.26	18.33 ± 1.52
CoPE*	93.94 ± 0.20	48.92 ± 1.32	21.62 ± 0.69
ER + GMED†	82.67 ± 1.90	34.84 ± 2.20	20.93 ± 1.60
ER_a + GMED†	82.21 ± 2.90	47.47 ± 3.20	19.60 ± 1.50
CURL*	92.59 ± 0.66	-	-
CNDPM*	93.23 ± 0.09	45.21 ± 0.18	20.10 ± 0.12
Dynamic-OCM	94.02 ± 0.23	49.16 ± 1.52	21.79 ± 0.68
OAES	**94.69** ± 0.18	**52.16** ± 0.25	**26.01** ± 1.02

Table 2: Classification accuracy results for five independent runs when testing various models on three datasets. * and † denote the results cited from (De Lange and Tuytelaars 2021) and (Jin et al. 2021), respectively.

objective function (Wasserstein GAN loss) for training G_{ε^i} and D_{ψ^i} at $\mathcal{T}_i$ is defined as (Gulrajani et al. 2017) :

$$\min_{\mathbb{P}_{\varepsilon^i}} \max_{D_{\psi^i} \in \Theta} \mathbb{E}_{\mathbf{x}_j \sim \mathcal{M}_i} \left[D_{\psi^i}(\mathbf{x}_j) \right] - \mathbb{E}_{\mathbf{x}' \sim \mathbb{P}_{\varepsilon^i}} \left[D_{\psi^i}(\mathbf{x}') \right] + \gamma \mathbb{E}_{\widehat{\mathbf{x}} \sim \mathbb{P}_{\widehat{\mathbf{x}}}} \left[\left(\left\| \nabla_{\widehat{\mathbf{x}}} D_{\psi^i}(\widehat{\mathbf{x}}) \right\|_2 - 1 \right)^2 \right], \quad (14)$$

where $\widehat{\mathbf{x}}$ is an interpolated image produced by $\widehat{\mathbf{x}} = a\mathbf{x}_i + (1-a)\mathbf{x}'$ where a is drawn from a uniform distribution $U(0,1)$ and $\mathbb{P}_{\widehat{\mathbf{x}}}$ is the distribution of the interpolated images. Different from WGAN (Gulrajani et al. 2017) which is trained on a static dataset, we train G_{ε^i} and D_{ψ^i} on the evolved memory buffer $\mathcal{M}_i$ in an online fashion.

Evaluation stage : At this stage, we evaluate the discrepancy between each previously learnt component and the current component by comparing the discriminator's outputs and therefore Eq. (13) is reformulated as :

$$\min \left\{ C_{\psi^i}(\mathbb{P}_{\theta_1^{c_1}}, \mathbb{P}_{\theta_t^i}), \cdots, C_{\psi^i}(\mathbb{P}_{\theta_{t-1}^{c_{t-1}}}, \mathbb{P}_{\theta_t^i}) \right\} \geq \beta, \quad (15)$$

where $C_{\psi^i}(\mathbb{P}_{\theta_1^{c_1}}, \mathbb{P}_{\theta_t^i})$ is defined as :

$$C_{\psi^i}(\mathbb{P}_{\theta_1^{c_1}}, \mathbb{P}_{\theta_t^i}) = \frac{1}{n'} \sum_{j=1}^{n'} \left| D_{\psi^i}(\mathbf{x}'_j) - D_{\psi^i}(\mathbf{x}_j) \right| \quad (16)$$

where $\mathbf{x}'_j \sim \mathbb{P}_{\theta_1^{c_1}}$ and $\mathbf{x}_j \sim \mathbb{P}_{\theta_t^i}$ are treated as the real and fake images in the context of the adversarial criterion. $|\cdot|$ is the absolute value and $n' = 128$ is the number of samples. A small $\left| D_{\psi^i}(\mathbf{x}'_j) - D_{\psi^i}(\mathbf{x}_j) \right|$ indicates that $\mathbb{P}_{\theta_1^{c_1}}$ is similar to $\mathbb{P}_{\theta_t^i}$ since $\mathbb{P}_{\theta_t^i}$ is trained to approximate $\mathbb{P}_{\mathcal{M}_i}$. The implementation for the OAES is explained in the following.

Algorithm Implementation

In this section, we provide the algorithm implementation of OAES (See the pseudocode in **Appendix-A** from SM[1]), which is summarized into three stages :

Methods	M-S	Param	M-C	Param	Split IM	Param
ER	10.89	161M	15.28	161M	25.10	125M
ER + GMED	16.23	161M	21.26	161M	27.26	125M
CoPE	22.45	161M	26.85	161M	26.37	125M
CNDPM	47.64	237M	66.25	185M	27.98	102M
OAES	**55.35**	157M	**72.56**	173M	**29.62**	78M

Table 3: Classification accuracy of various models in the cross-domain setting.

Stage 1 . Memory updating : Let $|\mathcal{M}|$ be the number of samples in the memory buffer and $|\mathcal{M}|^{max}$ be the maximum memory buffer size. The memory buffer $\mathcal{M}_i$ when reaching $|\mathcal{M}|^{max}$ at the i-th training step, is updated by removing the earliest stored samples, while adding newly given samples.

Stage 2 . Training the component : Let us suppose that we have already learnt t components $\mathbf{V} = \{\mathcal{V}_1^{c_1}, \cdots, \mathcal{V}_t^{c_t}\}$ at $\mathcal{T}_i$. We only train $\mathcal{V}_t^{c_t}$ on $\mathcal{M}_i$ at $\mathcal{T}_i$ by using Eq. (1) to avoid forgetting previously learnt knowledge. In addition, we train the generator G_{ε^i} and the discriminator D_{ψ^i} on $\mathcal{M}_i$ using adversarial loss (Eq. (14)).

Stage 3 . Check the expansion : When reaching $|\mathcal{M}|^{max}$ we check the expansion criterion using Eq. (15) (OAES evaluation stage). If Eq. (15) is satisfied, we add a new component $\mathcal{V}_{t+1} \in \mathbf{V}$ and clear up the memory $\mathcal{M}_i$ in order to learn statistically non-overlapping data in the following training step. We return to **Stage 1** for next training step ($\mathcal{T}_{i+1}$).

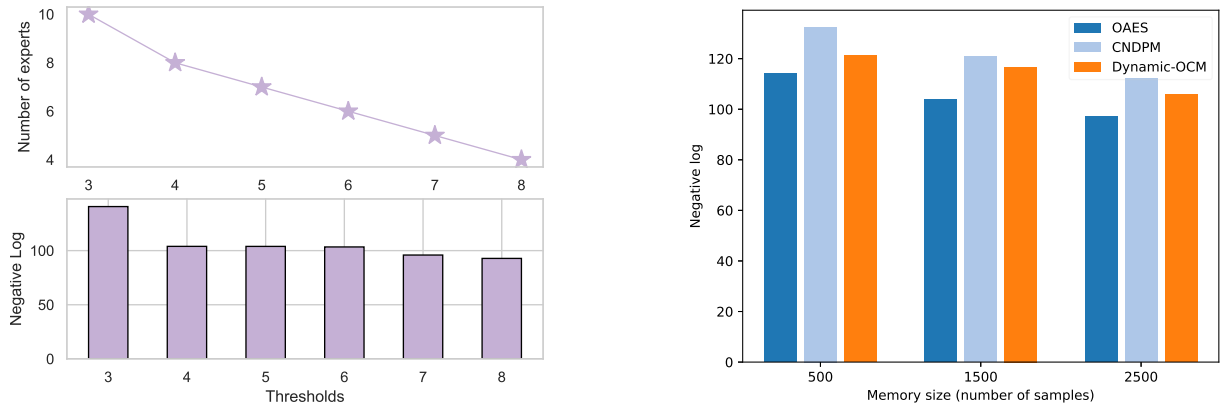
Experiments

Experiment Setting and Datasets

Datasets. For the generative modelling task, we have the following datasets : (1) **Split MNIST/Fashion.** We split MNIST/Fashion (LeCun et al. 1998) into ten parts according to the class. (2) **Split MNIST-Fashion.** We combine Split MNIST and Split Fashion in a class-incremental manner; (3) **Cross-Domain.** We consider to combine Split MNIST-Fashion and OMNIGLOT (Lake, Salakhutdinov, and Tenenbaum 2015). See more details in **Appendix-H2** from SM[1].

Criteria. The task classification task is employed for testing the accuracy. For testing the generative modelling task, we estimate the sample log-likelihood (Log) by using IWVAE bound (Burda, Grosse, and Salakhutdinov 2015), considering 1000 importance samples, as in (Ye and Bors 2022b).

Baseline. We introduce several baselines which are used for density estimation (Ye and Bors 2022b): (1) VAE-ELBO-OCM : A single VAE model with ELBO using the Online Cooperative Memorization (OCM) (Ye and Bors 2022b). (2) VAE-IWVAE50-OCM : A single VAE model with IWVAE using the OCM where the number of importance samples is 50. (3) VAE-ELBO-Random : A single VAE model with a memory that randomly removes samples when it reaches the maximum memory size. (4) Dynamic-ELBO-OCM : A mixture model with ELBO using OCM (Ye and Bors 2022b). (5) CNDPM (Lee et al. 2020); (6) LIMix (Ye and Bors 2021a) : we assign an episodic memory with a fixed buffer size for the LIMix model used for TFCL. The



(a) The effects of changing β in Eq. (13).

(b) Changing the memory size.

Figure 2: Ablation study results. (a) The performance and the number of components for OAES on Split MNIST when changing β. (b) The performance of various models on Split MNIST when changing the memory buffer size.

maximum number of components for various models is set to 30 to avoid memory overload.

Density Estimation

The results of various models on the density estimation task are shown in Table 1, where 'Memory' and 'N' denotes the memory size and the number of components. The threshold β for Split MNIST, Split Fashion, Split MNIST-Fashion and Cross-domain is 4.2, 3, 4 and 4.2, respectively. We observe that dynamic expansion models usually outperform static models while using a small memory buffer, especially in the cross-domain setting involving multiple data domains. These results demonstrate that DEM provides a better generalization performance than a single model when the data stream involves more underlying data distributions, which is theoretically explained in **Theorem 2**. In addition, the OAES-ELBO outperforms other DEM baselines in all settings. Compared with the Dynamic-ELBO-OCM, which performs the sample selection for the memory buffer, the OAES-ELBO requires less training while achieving better performance since it does not require the sample selection.

Classification Task

We replace each component using a conditional VAE or train a classifier along with each VAE component in order to test the classification task performance. We employ the learning setting and network architecture from (De Lange and Tuytelaars 2021). We adopt Split MNIST, Split CIFAR10 and Split CIFAR100 from (De Lange and Tuytelaars 2021) for the classification tasks. The details of all classification baselines and the threshold β are provided in **Appendix-H3** from SM[1]. The results for Split MNIST, Split CIFAR10 and Split CIFAR100 are reported in Table 2. We also consider Split MiniImageNet (Split IM) (Vinyals et al. 2016) which divides MiniImageNet into 20 tasks, where each task collects the images of five classes (Aljundi et al. 2019a).

In the following we consider evaluating our model in the more challenging setting where a data stream involves multiple data domains. First, we create a data stream named M-S, combining Split MNIST and SVHN. Then, we create another data stream M-C, which combines Split MNIST and Split CIFAR10. The memory buffer size is 1000 for Split M-S and Split M-C, and the results are reported in Table 3, where 'Param' denotes the number of parameters. Together with the results from Tables 2 and 3, we show that DEM methods outperform other baselines on all datasets. In addition, the proposed OAES achieves better performance while using fewer parameters than other baselines,according to the results from Table 3.

Ablation Study

The impact of the threshold β : We consider different values for β in Eq. (13) when training OAES on Split MNIST and the results are reported in Fig. 2a where 'Negative log' denotes the negative sample log-likelihood. We can observe that a small β leads to training more components while improving the performance. In contrast, a large β leads to fewer components in OAES.

The impact of the memory buffer size : We train various models under Split MNIST by using different memory buffer sizes and the results are shown in Fig. 2b. These results show that a large-scale memory buffer can improve the performance of all DEM models. The proposed OAES outperforms other baselines, especially when the memory buffer size is small (500 samples).

More ablation studies are provided in **Appendix-I** from SM[1].

Conclusion

In this paper, we develop a novel theoretical framework for VAEs, which interprets their forgetting process, when used in continual learning, as an increase in the JS divergence between the generator distribution and the evolved data distribution. Based on the theoretical analysis, we propose a novel dynamic expansion strategy that provides appropriate signals for the VAE-based Dynamic Expansion Model (DEM) expansion. The proposed model is shown to outperform other models in continual learning applications while ensuring a minimal architecture.

References

Ahn, H.; Cha, S. u.; Lee, D.; and Moon, T. 2019. Uncertainty-based Continual Learning with Adaptive Regularization. In *Advances in Neural Information Processing Systems (NeurIPS)*, 4394–4404.

Aljundi, R.; Belilovsky, E.; Tuytelaars, T.; Charlin, L.; Caccia, M.; Lin, M.; and Page-Caccia, L. 2019a. Online Continual Learning with Maximal Interfered Retrieval. In *Advances in Neural Information Processing Systems (NeurIPS)*, 11872–11883.

Aljundi, R.; Kelchtermans, K.; and Tuytelaars, T. 2019. Task-free continual learning. In *Proc. of IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 11254–11263.

Aljundi, R.; Lin, M.; Goujaud, B.; and Bengio, Y. 2019b. Gradient based sample selection for online continual learning. In *Proc. Neural Inf. Proc. Systems (NeurIPS)*, 11817–11826.

Arjovsky, M.; Chintala, S.; and Bottou, L. 2017. Wasserstein Generative Adversarial Networks. In *Proc. Int. Conf. on Machine Learning (ICML), vol. PMLR 70*, 214–223.

Bang, J.; Kim, H.; Yoo, Y.; Ha, J.-W.; and Choi, J. 2021. Rainbow memory: Continual learning with a memory of diverse samples. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 8218–8227.

Bang, J.; Koh, H.; Park, S.; Song, H.; Ha, J.-W.; and Choi, J. 2022. Online Continual Learning on a Contaminated Data Stream with Blurry Task Boundaries. In *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, 9275–9284.

Burda, Y.; Grosse, R.; and Salakhutdinov, R. 2015. Importance weighted autoencoders. In *Proc. Int. Cont. of Learning Representations (ICLR), arXiv preprint arXiv:1509.00519*.

Chaudhry, A.; Ranzato, M.; Rohrbach, M.; and Elhoseiny, M. 2018. Efficient lifelong learning with A-GEM. In *Proc. Int. Conf. on Learning Representations (ICLR), arXiv preprint arXiv:1812.00420*.

Chaudhry, A.; Rohrbach, M.; Elhoseiny, M.; Ajanthan, T.; Dokania, P. K.; Torr, P. H. S.; and Ranzato, M. 2019. On Tiny Episodic Memories in Continual Learning. *arXiv preprint arXiv:1902.10486*.

Chen, L.; Dai, S.; Pu, Y.; Li, C.; Su, Q.; and Carin, L. 2018. Symmetric variational autoencoder and connections to adversarial learning. In *Proc. Int. Conf. on Artificial Intel. and Statistics (AISTATS) 2018, vol. PMLR 84*, 661–669.

De Lange, M.; and Tuytelaars, T. 2021. Continual prototype evolution: Learning online from non-stationary data streams. In *Proc. of the IEEE/CVF International Conference on Computer Vision (CVPR)*, 8250–8259.

Derakhshani, M. M.; Zhen, X.; Shao, L.; and Snoek, C. 2021. Kernel continual learning. In *International Conference on Machine Learning, vol. PMLR 139*, 2621–2631.

Domke, J.; and Sheldon, D. R. 2018. Importance weighting and variational inference. In *Advances in Neural Information Processing Systems (NeurIPS)*, 4470–4479.

Goodfellow, I. J.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative adversarial nets. In *Advances in Neural Inf. Proc. Systems (NIPS)*, 2672–2680.

Gulrajani, I.; Ahmed, F.; Arjovsky, M.; Dumoulin, V.; and Courville, A. C. 2017. Improved training of Wasserstein GANs. In *Proc. Advances in Neural Inf. Proc. Systems (NIPS)*, 5767–5777.

Huang, C.-W.; Sankaran, K.; Dhekane, E.; Lacoste, A.; and Courville, A. 2019. Hierarchical importance weighted autoencoders. In *Int. Conf. on Machine Learning (ICML), vol. PMLR 97*, 2869–2878.

Hung, C.-Y.; Tu, C.-H.; Wu, C.-E.; Chen, C.-H.; Chan, Y.-M.; and Chen, C.-S. 2019. Compacting, Picking and Growing for Unforgetting Continual Learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 13647–13657.

Jin, X.; Sadhu, A.; Du, J.; and Ren, X. 2021. Gradient-based Editing of Memory Examples for Online Task-free Continual Learning. In *Advances in Neural Information Processing Systems (NeurIPS), arXiv preprint arXiv:2006.15294*.

Kemker, R.; McClure, M.; Abitino, A.; Hayes, T.; and Kanan, C. 2018. Measuring catastrophic forgetting in neural networks. In *Proc. of AAAI Conference on Artificial Intelligence*, 3390–3398.

Kim, M.; and Pavlovic, V. 2020. Recursive Inference for Variational Autoencoders. In *Advances in Neural Information Processing Systems (NeurIPS)*, 19632–19641.

Kingma, D. P.; Salimans, T.; Jozefowicz, R.; Chen, X.; Sutskever, J.; and Welling, M. 2016. Improved variational inference with inverse autoregressive flow. In *Proc. Advances in Neural Inf. Proc. Systems (NIPS)*, 4743–4751.

Kingma, D. P.; and Welling, M. 2013. Auto-encoding variational Bayes. *arXiv preprint arXiv:1312.6114*.

Kirkpatrick, J.; Pascanu, R.; Rabinowitz, N.; Veness, J.; Desjardins, G.; Rusu, A. A.; Milan, K.; Quan, J.; Ramalho, T.; Grabska-Barwinska, A.; Hassabis, D.; Clopath, C.; Kumaran, D.; and Hadsell, R. 2017. Overcoming catastrophic forgetting in neural networks. *Proc. of the National Academy of Sciences (PNAS)*, 114(13): 3521–3526.

Lake, B. M.; Salakhutdinov, R.; and Tenenbaum, J. B. 2015. Human-level concept learning through probabilistic program induction. *Science*, 350(6266): 1332–1338.

LeCun, Y.; Bottou, L.; Bengio, Y.; and Haffner, P. 1998. Gradient-based learning applied to document recognition. *Proc. of the IEEE*, 86(11): 2278–2324.

Lee, S.; Goldt, S.; and Saxe, A. 2021. Continual learning in the teacher-student setup: Impact of task similarity. In *Proc. Int. Conf. on Machine Learning (ICML), vol. PMLR 139*, 6109–6119.

Lee, S.; Ha, J.; Zhang, D.; and Kim, G. 2020. A Neural Dirichlet Process Mixture Model for Task-Free Continual Learning. In *Int. Conf. on Learning Representations (ICLR), arXiv preprint arXiv:2001.00689*.

Li, Z.; and Hoiem, D. 2017. Learning without forgetting. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 40(12): 2935–2947.

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

Maaløe, L.; Sønderby, C. K.; Sønderby, S. K.; and Winther, O. 2016. Auxiliary deep generative models. In *Proc. Int. Conf. on Machine Learning (ICML), vol. PMLR 48*, 1445–1453.

Martens, J.; and Grosse, R. B. 2015. Optimizing Neural Networks with Kronecker-factored Approximate Curvature. In *Proc. of Int. Conf. on Machine Learning (ICML), vol. PMLR 37*, 2408–2417.

Mescheder, L.; Nowozin, S.; and Geiger, A. 2017. Adversarial Variational Bayes: Unifying variational autoencoders and generative adversarial networks. In *Proc. Int. Conf. on Machine Learning (ICML), vol. PMLR 70*, 2391–2400.

Molchanov, D.; Kharitonov, V.; Sobolev, A.; and Vetrov, D. 2019. Doubly semi-implicit variational inference. In *Proc. Int. Conf. on Artificial Intelligence and Statistics (AISTATS), vol. PMLR 89*, 2593–2602.

Nguyen, C. V.; Li, Y.; Bui, T. D.; and Turner, R. E. 2017. Variational continual learning. In *Proc. Int. Conf. on Learning Representations (ICLR), arXiv preprint arXiv:1710.10628*.

Polikar, R.; Upda, L.; Upda, S. S.; and Honavar, V. 2001. Learn++: An incremental learning algorithm for supervised neural networks. *IEEE Trans. on Systems Man and Cybernetics, Part C*, 31(4): 497–508.

Ramapuram, J.; Gregorova, M.; and Kalousis, A. 2020. Lifelong Generative Modeling. *Neurocomputing*, 404: 381–400.

Rao, D.; Visin, F.; Rusu, A. A.; Teh, Y. W.; Pascanu, R.; and Hadsell, R. 2019. Continual Unsupervised Representation Learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, 7645–7655.

Rezende, D. J.; and Mohamed, S. 2015. Variational inference with normalizing flows. In *Proc. Int. Conf. on Machine Learning (ICML), vol. PMLR 37*, 1530–1538.

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

Shi, Y.; Yuan, L.; Chen, Y.; and Feng, J. 2021. Continual learning via bit-level information preserving. In *Proc. of IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 16674–16683.

Vahdat, A.; and Kautz, J. 2020. NVAE: A Deep Hierarchical Variational Autoencoder. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, 19667–19679.

Vinyals, O.; Blundell, C.; Lillicrap, T.; Kavukcuoglu, K.; and Wierstra, D. 2016. Matching networks for one shot learning. *Advances in neural information processing systems (NIPS)*, 29: 3637–3645.

Wang, S.; Li, X.; Sun, J.; and Xu, Z. 2021. Training networks in null space of feature covariance for continual learning. In *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 184–193.

Wen, Y.; Tran, D.; and Ba, J. 2020. BatchEnsemble: an Alternative Approach to Efficient Ensemble and Lifelong Learning. In *Proc. Int. Conf. on Learning Representations (ICLR), arXiv preprint arXiv:2002.06715*.

Xiao, T.; Zhang, J.; Yang, K.; Peng, Y.; and Zhang, Z. 2014. Error-driven incremental learning in deep convolutional neural network for large-scale image classification. In *Proc. of ACM Int. Conf. on Multimedia*, 177–186.

Ye, F.; and Bors, A. 2022a. Lifelong Teacher-Student Network Learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(10): 6280–6296.

Ye, F.; and Bors, A. G. 2020a. Learning Latent Representations Across Multiple Data Domains Using Lifelong VAEGAN. In *Proc. of European Conference on Computer Vision (ECCV), vol. LNCS 12365*, 777–795.

Ye, F.; and Bors, A. G. 2020b. Lifelong learning of interpretable image representations. In *Proc. Int. Conf. on Image Processing Theory, Tools and Applications (IPTA)*, 1–6.

Ye, F.; and Bors, A. G. 2020c. Mixtures of variational autoencoders. In *Proc. Int. Conf. on Image Processing Theory, Tools and Applications (IPTA)*, 1–6.

Ye, F.; and Bors, A. G. 2021a. Lifelong Infinite Mixture Model Based on Knowledge-Driven Dirichlet Process. In *Proc. of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 10695–10704.

Ye, F.; and Bors, A. G. 2021b. Lifelong Twin Generative Adversarial Networks. In *Proc. IEEE Int. Conf. on Image Processing (ICIP)*, 1289–1293.

Ye, F.; and Bors, A. G. 2022b. Continual Variational Autoencoder Learning via Online Cooperative Memorization. In *Proc. of European Conference on Computer Vision (ECCV), vol. LNCS 13683*, 531–549.

Ye, F.; and Bors, A. G. 2022c. Deep Mixture Generative Autoencoders. *IEEE Transactions on Neural Networks and Learning Systems*, 33(10): 5789–5803.

Ye, F.; and Bors, A. G. 2022d. Dynamic Self-Supervised Teacher-Student Network Learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1–19.

Ye, F.; and Bors, A. G. 2022e. Learning an evolved mixture model for task-free continual learning. In *Proc. IEEE Int. Conf. on Image Processing (ICIP)*, 1936–1940.

Ye, F.; and Bors, A. G. 2022f. Lifelong Generative Modelling Using Dynamic Expansion Graph Model. In *Proc. AAAI Conf. on Artificial Intelligence*, 8857–8865.

Ye, F.; and Bors, A. G. 2023. Lifelong Mixture of Variational Autoencoders. *IEEE Transactions on Neural Networks and Learning Systems*, 34(1): 461–474.

Zhou, G.; Sohn, K.; and Lee, H. 2012. Online incremental feature learning with denoising autoencoders. In *Proc. Int. Conf. on Artificial Intelligence and Statistics (AISTATS), vol. PMLR 22*, 1453–1461.