Lifelong Generative Modelling Using Dynamic Expansion Graph Model

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

Variational Autoencoders (VAEs) suffer from degenerated performance, when learning several successive tasks. This is caused by catastrophic forgetting. In order to address the knowledge loss, VAEs are using either Generative Replay (GR) mechanisms or Expanding Network Architectures (ENA). In this paper we study the forgetting behaviour of VAEs using a joint GR and ENA methodology, by deriving an upper bound on the negative marginal log-likelihood. This theoretical analysis provides new insights into how VAEs forget the previously learnt knowledge during lifelong learning. The analysis indicates the best performance achieved when considering model mixtures, under the ENA framework, where there are no restrictions on the number of components. However, an ENA-based approach may require an excessive number of parameters. This motivates us to propose a novel Dynamic Expansion Graph Model (DEGM). DEGM expands its architecture, according to the novelty associated with each new databases, when compared to the information already learnt by the network from previous tasks. DEGM training optimizes knowledge structuring, characterizing the joint probabilistic representations corresponding to the past and more recently learned tasks. We demonstrate that DEGM guarantees optimal performance for each task while also minimizing the required number of parameters. Supplementary materials (SM) and source code are available[1].

1 Introduction

The Variational Autoencoder (VAE) (Kingma and Welling 2013) is a popular generative deep learning model with remarkable successes in learning unsupervised tasks by inferring probabilistic data representations (Chen et al. 2018), for disentangled representation learning (Higgins et al. 2017; Ye and Bors 2021d) and for image reconstruction tasks (Ye and Bors 2020c, 2021b,c,d). Training a VAE model involves maximizing the marginal log-likelihood $\log p_\theta(\mathbf{x}) = \log \int p_\theta(\mathbf{x}|\mathbf{z})p(\mathbf{z})\mathrm{d}\mathbf{z}$ which is intractable during optimization due to the integration over the latent space defined by the variables $\mathbf{z}$. VAEs introduce using a variational distribution $q_\omega(\mathbf{z}|\mathbf{x})$ to approximate the posterior and the model is trained by maximizing a lower bound, called Evidence Lower Bound (ELBO), (Kingma and Welling 2013) :

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

where $p_\theta(\mathbf{x}\,|\,\mathbf{z})$ and $p(\mathbf{z}) = \mathcal{N}(0, I)$ are the decoding and prior distribution, respectively, while $KL[\cdot]$ represents the Kullback–Leibler divergence. Defining a tighter ELBO to the marginal log-likelihood, achieved by using a more expressive posterior (Kim and Pavlovic 2020; Maaløe et al. 2016), importance sampling (Burda, Grosse, and Salakhutdinov 2015; Domke and Sheldon 2018) or through hierarchical variational models (Molchanov et al. 2019; Vahdat and Kautz 2020), has been successful for improving the performance of VAEs. However, these approaches can only guarantee a tight ELBO for learning a single domain and have not yet been considered for lifelong learning (LLL), which involves learning sequentially several tasks associated with different databases. VAEs, similarly to other deep learning methods (Guo et al. 2020), suffer from catastrophic forgetting (French 1999), when learning new tasks, leading to degenerate performance on the previous tasks. One direct way enabling VAE for LLL is the Generative Replay (GR) process (Ramapuram, Gregorova, and Kalousis 2020).

Let us consider a VAE model to be trained on a sequence of t tasks. After the learning of i-th task is finished, the GR process allows the model to generate a pseudo dataset $\tilde{\mathbf{X}}^i$ which will be mixed with the incoming data set $\mathbf{X}^{new}$ to form a joint dataset for the $(i+1)$-th task learning. Usually, the distribution of $\{\tilde{\mathbf{X}}^i, \mathbf{X}^{new}\}$ does not match the real data distribution exactly and the optimal parameters $\{\theta^*, \omega^*\}$ are estimated by maximizing ELBO, on samples $\mathbf{x}'$ drawn from $\{\tilde{\mathbf{X}}^i, \mathbf{X}^{new}\}$. $\mathcal{L}_{ELBO}(\cdot)$ is not a tight ELBO in Eq. (1) by using the model's parameters $\{\theta^*, \omega^*\}$ which actually are not optimal for the real sample log-likelihood $\log p_\theta(\mathbf{x})$ (See Proposition 6 in Appendix-I from SM[1]). In this paper, we aim to evaluate the tightness between $\log p_\theta(\mathbf{x})$ and $\mathcal{L}_{ELBO}(\mathbf{x}'; \theta^*, \omega^*)$, by developing a novel upper bound to the negative marginal log-likelihood, called Lifelong ELBO (LELBO). LELBO involves the discrepancy distance (Mansour, Mohri, and Rostamizadeh 2009) between the target and the evolved source distributions, as well as the accumulated errors, caused when learning each new task. This analysis provides insights into how the VAE model is losing previously learnt knowledge during LLL. We also generalize the

[1] https://github.com/dtuzi123/Expansion-Graph-Model

proposed theoretical analysis to ENA models, which leads to a novel dynamic expansion graph model (DEGM) enabled with generating graph structures linking the existing components and a newly created component, benefiting on the transfer learning and the reduction of the model's size. We list our contributions as :

- This is the first research study to develop a novel theoretical framework for analyzing VAE's forgetting behaviour during LLL.
- We develop a novel generative latent variable model which guarantees the trade-off between the optimal performance for each task and the model's size during LLL.
- We propose a new benchmark for the probability density estimation task under the LLL setting.

2 Related works

Recent efforts in LLL focus on regularization based methods (Jung, Jung, and Kim 2016; Li and Hoiem 2017), which typically penalize significant changes in the model's weights when learning new tasks. Other methods rely on memory systems such as using past learned data to guide the optimization (Chaudhry et al. 2018; Guo et al. 2020; Pan et al. 2020), using Generative Adversarial Nets (GANs) or VAEs (Achille et al. 2018; Ramapuram, Gregorova, and Kalousis 2020; Ye and Bors 2021g; Shin et al. 2017; Ye and Bors 2020a,b, 2021a) aiming to reproduce previously learned data samples in order to attempt to overcome forgetting. However, most of these models focus on predictive tasks and the lifelong generative modelling remains an unexplored area.

Prior works for continuously learning VAEs are divided into two branches: Generative Replay (GR) and Expanding Network Architectures (ENA). GR was used in VAEs for the first time in (Achille et al. 2018) while (Ramapuram, Gregorova, and Kalousis 2020) extends the GR mechanism within a Teacher-Student framework, called the Lifelong Generative Modelling (LGM). A major limitation for GR is its inability of learning a long sequence of data domains. This is due to its fixed model capacity while having to retrain the generator frequently (Ye and Bors 2020a). This issue is relieved by using ENA (Lee et al. 2020), inspired by a network expansion mechanism (Rao et al. 2019), or by employing a combination between ENA and GR mechanisms (Ye and Bors 2021f,e). These methods significantly relieve forgetting but would suffer from informational interference when learning a new task (Riemer et al. 2019).

The tightness on ELBO is key to improving VAE's performance and one possible way is to use the Importance Weighted Autoencoder (IWELBO) (Burda, Grosse, and Salakhutdinov 2015) in which the tightness is controlled by the number of weighted samples considered. Other approaches focus on the choice of the approximate posterior distribution, including by using normalizing flows (Kingma et al. 2016; Rezende and Mohamed 2015), employing implicit distributions (Mescheder, Nowozin, and Geiger 2017) and using hierarchical variational inference (Huang et al. 2019). The IWELBO bound can be used with any of these approaches to further improve their performance (Sobolev and Vetrov 2019). Additionally, online variational inference (Nguyen et al. 2017) has been used in VAEs, but require to store the past samples for computing the approximate posterior, which is intractable when learning an infinite number of tasks. The tightness of ELBO under LLL was not studied in any of these works.

3 Preliminary

In this paper, we address a more general lifelong unsupervised learning problem where the task boundaries are provided only during the training. For a given sequence of tasks $\{\mathcal{T}_1, \ldots, \mathcal{T}_N\}$ we consider that each $\mathcal{T}_i$ is associated with an unlabeled training set Q_i^S and an unlabeled testing set Q_i^T. The model only sees a sequence of training sets $\{Q_1^S, \ldots, Q_N^S\}$ while it is evaluated on $\{Q_1^T, \ldots, Q_N^T\}$. Let us consider the input data space $\mathcal{X} \in \mathbb{R}^d$ of dimension d, and $\mathcal{P}_i$ the probabilistic representation of the testing set Q_i^T. We desire to evaluate the quality of reconstructing data samples $\mathbf{x} \in \mathcal{X}$, by a model using the square loss (SL) function $\|\mathbf{x} - h(\mathbf{x})\|^2$, where h is a hypothesis function in a space of hypotheses $\{h \in \mathcal{H} \mid \mathcal{H} : \mathcal{X} \to \mathcal{X}\}$. For the image space, the loss is represented by $\sum_{i=1}^{d} (\mathbf{x}[i] - h(\mathbf{x})[i])^2$, where $[i]$ represents the entry for the i-th dimension.

Definition 1 ***(Single model.)*** *Let $\mathcal{M} = \{f_\omega, g_\theta\}$ be a single model consisting of an encoder $f_\omega \colon \mathcal{X} \to \mathcal{Z}$ for representing $q_\omega(\mathbf{z} \,|\, \mathbf{x})$, and a decoder $g_\theta : \mathcal{Z} \to \mathcal{X}$ for modelling $p_\theta(\mathbf{x} \,|\, \mathbf{z})$. The latent variable $\mathbf{z} = f_\omega^\mu(\mathbf{x}) + f_\omega^\delta(\mathbf{x}) \odot \gamma$, $\gamma \sim \mathcal{N}(0, I)$ is reparameterized by the mean $f_\omega^\mu(\mathbf{x})$ and variance $f_\omega^\delta(\mathbf{x})$, implemented by a network $f_\omega(\mathbf{x})$. $\{\omega^t, \theta^t\}$ are the parameters of the model $\mathcal{M}^t$, where t represents the number of tasks considered for training the model. Let $g_\theta(f_\omega) \colon \mathcal{X} \to \mathcal{X}$ be the encoding-decoding process for $\mathcal{M}$.*

Definition 2 ***(Discrepancy distance.)*** *We implement $h \in \mathcal{H}$ by $g_\theta(f_\omega)$ evaluated on the error function $\mathcal{L} \colon \mathcal{X} \times \mathcal{X} \to \mathbb{R}_+$ which is bounded, $\forall (\mathbf{x}, \mathbf{x}') \in \mathcal{X}^2, \mathcal{L}(\mathbf{x}, \mathbf{x}') \le U$ for some $U > 0$. We define the error function as the SL function $\mathcal{L}(\mathbf{x}, \mathbf{x}') = \|\mathbf{x} - \mathbf{x}'\|^2, (\mathbf{x}, \mathbf{x}') \in \mathcal{X}$. A risk for $h(\cdot)$ on the target distribution $\mathcal{P}_i$ of the i-th domain (task) is defined as $\mathcal{R}_{\mathcal{P}_i}(h, f_{\mathcal{P}_i}) = \mathbb{E}_{\mathbf{x} \sim \mathcal{P}_i} \mathcal{L}(h(\mathbf{x}), f_{\mathcal{P}_i}(\mathbf{x}))$, where $f_{\mathcal{P}_i} \in \mathcal{H}$ is the true labeling function for $\mathcal{P}_i$. The discrepancy distance on two domains $\{\mathcal{P}, \mathbb{P}\}$ over $\mathcal{X}$, is defined as:*

$$
\begin{aligned}
disc_{\mathcal{L}}\left(\mathcal{P}, \mathbb{P}\right) = \sup_{(h,h') \in \mathcal{H}} \Big| & \mathbb{E}_{\mathbf{x} \sim \mathcal{P}} \left[\mathcal{L}\left(h'\left(\mathbf{x}\right), h\left(\mathbf{x}\right)\right)\right] \\
& - \mathbb{E}_{\mathbf{x} \sim \mathbb{P}} \left[\mathcal{L}\left(h'\left(\mathbf{x}\right), h\left(\mathbf{x}\right)\right)\right] \Big| .
\end{aligned} \tag{2}
$$

Definition 3 ***(Empirical discrepancy distance.)*** *In practice, we usually get samples of size $m_{\mathcal{P}}$ and $m_{\mathbb{P}}$ considering $U_{\mathcal{P}}$ and $U_{\mathbb{P}}$, respectively, and these samples form the empirical distributions $\hat{\mathcal{P}}$ and $\hat{\mathbb{P}}$, corresponding to $\mathcal{P}$ and $\mathbb{P}$. Then, the discrepancy can be estimated by using finite samples :*

$$
\begin{aligned}
disc_{\mathcal{L}}\big(\mathcal{P}, \mathbb{P}\big) \le disc_{\mathcal{L}}\big(\hat{\mathcal{P}}, \hat{\mathbb{P}}\big) + 8\big(\mathrm{Re}_{U_{\mathcal{P}}}\left(\mathcal{H}\right) + \mathrm{Re}_{U_{\mathbb{P}}}\left(\mathcal{H}\right)\big) \\
+ 3M\left(\sqrt{\frac{\log\left(\frac{4}{\delta}\right)}{2m_{\mathcal{P}}}} + \sqrt{\frac{\log\left(\frac{4}{\delta}\right)}{2m_{\mathbb{P}}}}\right),
\end{aligned} \tag{3}
$$

which holds with probability $1 - \delta, \delta \in (0, 1)$, where $M > 0$, and $\mathrm{Re}_{U_{\mathbb{P}}}$ is the Rademacher complexity (See Appendix-H from SM[1]). We use $disc_{\mathcal{L}}^{\star}(\cdot)$ to represent the right-hand side (RHS) of Eq. (3).

4 The theoretical framework

4.1 Generalization bounds for a single model

Let us consider $\mathbb{P}^i$ the approximation distribution for the generated data by $g_{\theta^i}(\cdot)$ of $\mathcal{M}^i$, which was trained on a sequence of domains $\{Q_1^S,\ldots,Q_i^S\}$ and $\tilde{\mathcal{P}}_i$ represents the probabilistic representation for Q_i^S. Training $\mathcal{M}^{i+1}$ using GR, for the $(i+1)$-th task, requires the minimization of $\mathcal{L}^\star$ (implemented as the negative ELBO) :

$$\mathcal{M}^{i+1} = \mathop{\arg\min}_{\omega^{(i+1)},\theta^{(i+1)}} \mathcal{L}^\star\left(\mathbb{P}^{i+1}, \mathbb{P}^i \otimes \tilde{\mathcal{P}}_{i+1}\right), \tag{4}$$

where $\otimes$ represents the mixing distribution $\mathbb{P}^i \otimes \tilde{\mathcal{P}}_{i+1}$, formed by samples uniformly drawn from both $\mathbb{P}^i$ and $\tilde{\mathcal{P}}_{i+1}$, respectively. Eq. (4) can be treated as a recursive optimization problem as i increases from 1 to t. The learning goal of $\mathcal{M}^{i+1}$ is to approximate the distribution $\mathbb{P}^{i+1} \approx \mathbb{P}^i \otimes \tilde{\mathcal{P}}_{i+1}$ by minimizing $\mathcal{L}^\star(\cdot)$, when learning $(i+1)$-th task. During the LLL, the errors corresponding to the initial tasks Q_i^S, $i<t$, would increase, leading to a degenerated performance on its corresponding unseen domain, defined by its performance on the testing set Q_i^T. One indicator for the generalization ability of a model $\mathcal{M}$ is to predict its performance on a testing data set by achieving a certain error rate on a training data set (Kuroki et al. 2019). In this paper, we develop a new theoretical analysis that can measure the generalization of a model under LLL where the source distribution is evolved over time. Before we introduce the Generalization Bound (GB) for -ELBO, we firstly define the GB when considering a VAE learning a single task in Theorem 1 and then when learning several tasks in Theorem 2.

Theorem 1 *Let $\mathcal{P}_i$ and $\tilde{\mathcal{P}}_i$ be two domains over $\mathcal{X}$. Then for $h^*_{\mathcal{P}_i} = \arg\min_{h\in\mathcal{H}}\mathcal{R}_{\mathcal{P}_i}(h, f_{\mathcal{P}_i})$ and $h^*_{\tilde{\mathcal{P}}_i} = \arg\min_{h\in\mathcal{H}}\mathcal{R}_{\tilde{\mathcal{P}}_i}(h, f_{\tilde{\mathcal{P}}_i})$ where $f_{\tilde{\mathcal{P}}_i} \in \mathcal{H}$ is the ground truth function (identity function under the encoder-decoding process) for $\tilde{\mathcal{P}}_i$, we can define the GB between $\mathcal{P}_i$ and $\tilde{\mathcal{P}}_i$:*

$$\begin{aligned}\mathcal{R}_{\mathcal{P}_i}\big(h, f_{\mathcal{P}_i}\big) \le \mathcal{R}_{\tilde{\mathcal{P}}_i}\big(h, h^*_{\tilde{\mathcal{P}}_i}\big) + disc^\star_{\mathcal{L}}\big(\mathcal{P}_i, \tilde{\mathcal{P}}_i\big)\\ + \mathcal{R}_{\mathcal{P}_i}\big(h^*_{\mathcal{P}_i}, f_{\mathcal{P}_i}\big) + \mathcal{R}_{\mathcal{P}_i}\big(h^*_{\mathcal{P}_i}, h^*_{\tilde{\mathcal{P}}_i}\big),\end{aligned} \tag{5}$$

where the last two terms represent the optimal combined risk denoted by $\varepsilon(\mathcal{P}_i, \tilde{\mathcal{P}}_i)$, and we have :

$$\mathcal{R}_{\tilde{\mathcal{P}}_i}\big(h, h^*_{\tilde{\mathcal{P}}_i}\big) = \mathbb{E}_{\mathbf{x}\sim\tilde{\mathcal{P}}_i}\mathcal{L}\big(h(\mathbf{x}), h^*_{\tilde{\mathcal{P}}_i}(\mathbf{x})\big). \tag{6}$$

See the proof in Appendix-A from SM[1]. We use $\mathcal{R}_A(\mathcal{P}_i, \tilde{\mathcal{P}}_i)$ to represent $disc^\star_{\mathcal{L}}(\mathcal{P}_i, \tilde{\mathcal{P}}_i) + \varepsilon(\mathcal{P}_i, \tilde{\mathcal{P}}_i)$. Theorem 1 explicitly defines the generalization error of $\mathcal{M}$ trained on the source distribution $\tilde{\mathcal{P}}_i$. With Theorem 1, we can extend this GB to ELBO and the marginal log-likelihood evaluation when the source distribution evolves over time.

Theorem 2 *For a given sequence of tasks $\{\mathcal{T}_1,\ldots,\mathcal{T}_t\}$, we derive a GB between the target distribution and the evolved source distribution during the t-th task learning :*

$$\begin{aligned}\frac{1}{t}\sum_{i=1}^{t}\mathcal{R}_{\mathcal{P}_i}\big(h, f_{\mathcal{P}_i}\big) \le \mathcal{R}_{\mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t}\big(h, h^*_{\mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t}\big)\\ + \mathcal{R}_A\big(\mathcal{P}_{(1:t)}, \mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t\big),\end{aligned} \tag{7}$$

where $\mathcal{P}_{(1:t)}$ is the mixture distribution $\{\mathcal{P}_1\otimes\mathcal{P}_2,\ldots,\otimes\mathcal{P}_t\}$.

See the proof in Appendix-B from SM[1].

Remark. Theorem 2 has the following observations:

- The performance on the target domain depends mainly on the discrepancy term even if $\mathcal{M}$ minimizes the source risk, from the first term of RHS of Eq. (7).
- In the GR process, $\mathbb{P}^{t-1}$ is gradually degenerated as t increases due to the repeated retraining (Ye and Bors 2020a), which leads to a large discrepancy distance term.

We also extend the idea from Theorem 2 to derive GBs for GANs, which demonstrates that the discrepancy distance between the target and the generator's distribution plays an important role for the generalization performance of GANs under the LLL setting, exhibiting similar forgetting behaviour as VAEs (See details in Appendix-G from SM[1]). In the following, we extend this GB to $\mathcal{L}^\star$.

Lemma 1 *Let us consider the random samples $\mathbf{x}_i^T \sim \mathcal{P}_i$, for $i = 1,\ldots,t$. The sample log-likelihood and its ELBO for all $\{\mathcal{P}_1,\ldots,\mathcal{P}_t\}$ can be represented by $\sum_{i=1}^t \log p_\theta(\mathbf{x}_i^T)$ and $\sum_{i=1}^t \mathcal{L}_{ELBO}(h, \mathbf{x}_i^T)$. Let $\tilde{\mathbf{x}}^t$ represent the random sample drawn from $\mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t$. We know that $KL(q_{\omega^t}(\mathbf{z}\,|\,\mathbf{x}_i^T)\;||\;p(\mathbf{z})) \neq KL(q_{\omega^t}(\mathbf{z}\,|\,\tilde{\mathbf{x}}^t)\;||\;p(\mathbf{z}))$ if $q_{\omega^t}(\mathbf{z}\,|\,\mathbf{x}_i^T) \neq q_{\omega^t}(\mathbf{z}\,|\,\tilde{\mathbf{x}}^t)$, and we have :*

$$\begin{aligned}&\frac{1}{t}\sum_{i=1}^{t}\mathbb{E}_{\mathcal{P}_i}KL\big(q_{\omega^t}(\mathbf{z}\mid\mathbf{x}_i^T)\;||\;p(\mathbf{z})\big) \le\\ &\mathbb{E}_{\mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t}KL\big(q_{\omega^t}(\mathbf{z}\mid\tilde{\mathbf{x}}^t)\;||\;p(\mathbf{z})\big) + \big|KL_1 - KL_2\big|,\end{aligned} \tag{8}$$

where $q_{\omega^t}(\cdot)$ represents the inference model for $\mathcal{M}^t$. KL_1 and KL_2 represent the left-hand side term (LHS) and the first term of the RHS of Eq. (8), respectively. Since ELBO consists of a negative reconstruction error term, a KL divergence term and a constant ($-\frac{1}{2}\log\pi$) (Doersch 2016), when the decoder models a Gaussian distribution with a diagonal covariance matrix (the diagonal element is $1/\sqrt{2}$), we derive a GB on -ELBO by combining (7) and (8) :

$$\begin{aligned}&\frac{1}{t}\sum_{i=1}^{t}\mathbb{E}_{\mathcal{P}_i}\Big[-\mathcal{L}_{ELBO}\big(\mathbf{x}_i^T; h\big)\Big] \le \mathcal{R}_A\big(\mathcal{P}_{(1:t)}, \mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t\big)\\ &+ \mathbb{E}_{\mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t}\Big[-\mathcal{L}_{ELBO}\big(\tilde{\mathbf{x}}^t; h\big)\Big] + \big|KL_1 - KL_2\big|,\end{aligned} \tag{9}$$

where $\mathbf{x}_i^T \sim \mathcal{P}_i$ and $\tilde{\mathbf{x}}^t \sim \mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t$.

See the proof in Appendix-C from SM[1]. We call the RHS of Eq. (9) as Lifelong ELBO (LELBO), denoted as $\mathcal{L}_{LELBO}$ which is a bound for an infinite number of tasks ($t\to\infty$). This bound shows the behaviour of $\mathcal{M}$ when minimizing -ELBO when learning each task. $\mathcal{L}_{LELBO}$ is also an upper bound to $-\sum_{i=1}^t \mathbb{E}_{\mathcal{P}_i}\left[\log p(\mathbf{x}_i^T)\right]/t$, estimated by $\mathcal{M}^t$.

The generalization of LELBO. From Eq. (9), we can generalize LELBO to other VAE variants under LLL, including the auxiliary deep generative models (Maaløe et al. 2016) and hierarchical variational inference (Sobolev and Vetrov 2019) (See details in Appendix-F from SM[1]). IWELBO bound (Burda, Grosse, and Salakhutdinov 2015) is an extension of ELBO by generating multiple weighted samples under the importance sampling (Domke and Sheldon 2018). We generalize the IWELBO bounds to the LLL setting as:

$$\frac{1}{t}\sum_{i=1}^{t}\mathbb{E}_{\mathbf{x}_i^T\sim\mathcal{P}_i}\left[-\log p\left(\mathbf{x}_i^T\right)\right]\leq$$
$$\mathbb{E}_{\tilde{\mathbf{x}}^t\sim\mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t}\left[-\mathbb{E}_{\mathbf{z}_1\ldots,\mathbf{z}_{K'}\sim q(\mathbf{z}|\mathbf{x})}\left[\log\frac{1}{K'}\sum_{i=1}^{K'}\frac{p\left(\tilde{\mathbf{x}}^t,\mathbf{z}_i\right)}{q\left(\mathbf{z}_i\mid\mathbf{x}\right)}\right]\right]$$
$$+\left|KL_1-KL_2\right|+\mathcal{R}_A\left(\mathcal{P}_{(1:t)},\mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t\right)\,. \tag{10}$$

See the derivation in Appendix-F.1 from SM[1]. We consider $\mathbf{z}_{1:K'}=\{\mathbf{z}_1,\ldots,\mathbf{z}_{K'}\}$ and omit the subscript for $q(\cdot)$. K' is the number of weighted samples (Domke and Sheldon 2018). We call RHS of Eq. (10) as $\mathcal{L}_{LELBO_{K'}}$, and $\mathcal{L}_{LELBO_{K'=1}}=\mathcal{L}_{LELBO}$.

Remark. We have several conclusions from Eq. (10) :

- Based on the assumption that $\mathbb{P}^{t-1}$ is fixed and $|KL_1-KL_2|=0$, we have $\mathcal{L}_{LELBO_{K'+1}}\leq\mathcal{L}_{LELBO_{K'}}$.
- The tightness of ELBO on $\mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t$ (the second term of RHS of Eq. (10)) can not guarantee a tight bound on the testing data log-likelihood since the RHS of Eq. (10) contains the discrepancy distance term and other error terms.

A tight GB can be achieved by reducing the discrepancy distance term by training a powerful generator that approximates the target distributions well, for example by using the Autoencoding VAE (Cemgil et al. 2020) or adversarial learning (Goodfellow et al. 2014), which would fail when learning several entirely different domains due to the fixed model's capacity and the mode collapse (Srivastava et al. 2017). In the following section, we show how we can achieve a tight GB by increasing the model's capacity through an expansion mechanism.

4.2 Generalization bounds for ENA

For a given mixture model $\mathbf{M}=\{\mathcal{M}_1,\ldots,\mathcal{M}_K\}$, each component $\mathcal{M}_i$ can be trained with GR. In order to assess the trade-off between performance and complexity, we assume that $\mathbb{P}^{(i,s)}$ is the generator distribution of the s-th component which was trained on a number of i tasks. Suppose that the j-th task was learnt by the s-th component of the mixture and its approximation distribution $\mathbb{P}_j^{(m,s)}$ is formed by the sampling process $\mathbf{x}\sim\mathbb{P}^{(i,s)}$ if $I_{\mathcal{T}}(\mathbf{x})=j$, where $I_{\mathcal{T}}\colon\mathcal{X}\to\mathcal{T}$ is the function that returns the true task label for the sample $\mathbf{x}$, and m represents the number of times $\mathcal{M}_s$ was used with GR for the j-th task. We omit the component index s for $\mathbb{P}_j^{(m,s)}$ for the sake of simplification and let $\mathbb{P}_t^0$ represent $\tilde{\mathcal{P}}_t$. In the following, we derive a GB for a mixture model $\mathbf{M}$ with K components.

Theorem 3 *Let $C=\{c_1,\ldots,c_m\}$ represent a set, where each item c_i indicates that the c_i-th component ($\mathcal{M}_{c_i}^1$) is only trained once during LLL. We use $A=\{a_1,\ldots,a_m\}$ to represent the task label set for C, where a_i is associated to c_i. Let $C'=\{c_1',\ldots,c_k'\}$ represent a set where c_i' indicates that the c_i'-th component $\mathcal{M}_{c_i'}$ is trained more than once and is associated with a task label set $A_{c_i'}'=\{a(i,1),\ldots,a(i,n)\}$. Let $\tilde{C}=\{c(i,1),\ldots,c(i,n)\}$ be a set where $c(i,j)$ denotes the number of times $\mathcal{M}_{c_i'}$ was used for $a(i,j)$-th task. We have $|C|+|C'|=K$, $|A_{c_i'}'|>1$, where K is the number of components in the mixture model and $|\cdot|$ is the cardinality of a set. Let $\tilde{A}=\{\tilde{a}_1,\ldots,\tilde{a}_k\}$ represent a set where each $\tilde{a}_i$ denotes the number of tasks modelled by the probabilistic representations of the c_i'-th component $\tilde{a}_i=|A_{c_i'}'|$. We derive the bound for $\mathbf{M}$ during the t-th task learning :*

$$\begin{aligned}&\frac{1}{t}\sum\nolimits_{i=1}^{|C'|}\left\{\sum\nolimits_{j=1}^{\tilde{a}_i}\left\{\mathcal{R}_{\mathcal{P}_{a(i,j)}}\left(h_{c_i'},f_{\mathcal{P}_{a(i,j)}}\right)\right\}\right\}+\\&\frac{1}{t}\sum\nolimits_{i=1}^{|C|}\left\{\mathcal{R}_{\mathcal{P}_{a_i}}\left(h_{c_i},f_{\mathcal{P}_{a_i}}\right)\right\}\leq\frac{1}{t}\mathcal{R}_C+\frac{1}{t}\mathcal{R}_{A'}\end{aligned} \tag{11}$$

where each $h_{c_i}\in\mathcal{H}$ and $h_{c_i'}\in\mathcal{H}$ represent the hypothesis of the c_i-th and c_i'-th component in the mixture, respectively. $\mathcal{R}_C$ is the error evaluated by the components that are trained only once :

$$\mathcal{R}_C=\sum\nolimits_{i=1}^{|C|}\left\{\mathcal{R}_{\tilde{\mathcal{P}}_{a_i}}\left(h_{c_i},h^*_{\tilde{\mathcal{P}}_{a_i}}\right)+\mathcal{R}_A\left(\mathcal{P}_{a_i},\tilde{\mathcal{P}}_{a_i}\right)\right\}, \tag{12}$$

and $\mathcal{R}_{A'}$ is the accumulated error evaluated by the components that are trained more than once :

$$\begin{aligned}\mathcal{R}_{A'}=\sum\nolimits_{i=1}^{|C'|}&\left\{\sum\nolimits_{j=1}^{\tilde{a}_i}\left\{\mathcal{R}_{\mathbb{P}_{a(i,j)}^{c(i,j)}}\left(h_{c_i'},h^*_{\mathbb{P}_{a(i,j)}^{c(i,j)}}\right)\right.\right.\\&\left.\left.+\mathcal{R}_A\left(\mathcal{P}_{a(i,j)},\mathbb{P}_{a(i,j)}^{c(i,j)}\right)\right\}\right\},\end{aligned} \tag{13}$$

and after decomposing the last term it becomes

$$\begin{aligned}\mathcal{R}_{A'}=\sum\nolimits_{i=1}^{|C'|}&\left\{\sum\nolimits_{j=1}^{\tilde{a}_i}\left\{\mathcal{R}_{\mathbb{P}_{a(i,j)}^{c(i,j)}}\left(h_{c_i'},h^*_{\mathbb{P}_{a(i,j)}^{c(i,j)}}\right)\right.\right.\\&\left.\left.+\sum\nolimits_{k=-1}^{c(i,j)-1}\left\{\mathcal{R}_A\left(\mathbb{P}_{a(i,j)}^k,\mathbb{P}_{a(i,j)}^{k+1}\right)\right\}\right\}\right\}.\end{aligned} \tag{14}$$

The proof is provided in Appendix-D from SM[1].

Remark. We have several observations from **Theorem 3** :

- If $|C'|=1$ and $|C|=0$, then the term $\mathcal{R}_C$ in Eq. (11) would disappear while $\mathcal{R}_A'$ would accumulate additional error terms, according to Eq. (14).
- In contrast, if $|C|=t$, then the GB from Eq. (11) is reduced to $\mathcal{R}_C$, where the number of components K is equal to the number of tasks and there are no accumulated error terms, leading to a SM[1]all gap on GB.
- When $|C|$ increases, the gap on GB tends to be small and the model's complexity tends to be large because the accumulated error term will be reduced ($|C'|=K-|C|$ in Eq. (14)) while K increases.
- If a single component learns multiple tasks ($|C'|=1$), then GB on the initial tasks ($a(i,j)$ is small), tends to have more accumulated error terms compared to the GB on the latest given tasks ($a(i,j)$ is large), shown by the number of accumulated error terms $\mathcal{R}_A(\cdot,\cdot)$ in Eq. (14), controlled by $c(i,j)=t-a(i,j)$.

In the following, we extend GB from $\mathcal{L}$ to $\mathcal{L}^\star$.

Lemma 2 *We derive a GB for the marginal log-likelihood during the t-th task learning for* $\mathbf{M}$*:*

$$\frac{1}{t}\sum\nolimits_{i=1}^{t}\mathbb{E}_{\mathcal{P}_i}\Big[-\log p\big(\mathbf{x}_i^T\big)\Big] \le \frac{1}{t}\Big(\mathcal{R}_{A'}^{II}+\mathcal{R}_{C}^{II}+D^{\star}_{diff}\Big)+$$
$$\frac{1}{t}\sum\nolimits_{i=1}^{|\mathrm{C}'|}\Big\{\sum\nolimits_{j=1}^{\tilde{a}_i}\Big\{\mathbb{E}_{\mathbb{P}^{c(i,j)}_{a(i,j)}}\Big[-\mathcal{L}_{ELBO}\Big(\mathbf{x}^t_{a(i,j)};h_{c'_i}\Big)\Big]\Big\}$$
$$+\sum\nolimits_{i=1}^{|\mathrm{C}|}\Big\{\mathbb{E}_{\tilde{\mathcal{P}}_{a_i}}\Big[-\mathcal{L}_{ELBO}\big(\mathbf{x}^S_{a_i};h_{c_i}\big)\Big]\Big\}\Big\}, \tag{15}$$

where we omit the component's index for each $\log p(\mathbf{x}_i^T)$ *for the sake of simplification, we use* $\mathcal{R}_C^{II}$ *and* $\mathcal{R}_{A'}^{II}$ *to represent the second terms in the RHS's from Eq.* (12) *and* (13)*, respectively, and* $D^{\star}_{diff}$ *represents the absolute difference on the KL divergence (details in Appendix-E from SM*[1]*). Each* $\mathbf{x}^S_{a_i}$ *is drawn from* $\tilde{\mathcal{P}}_{a_i}$ *and each* $\mathbf{x}^t_{a(i,j)}$ *is drawn from* $\mathbb{P}^{c(i,j)}_{a(i,j)}$ *modelled by the* c'_i*-th component in* $\mathbf{M}$*.* $\mathcal{L}_{ELBO}(\mathbf{x}^S_{a_i};h_{c_i})$ *is the ELBO estimated by the* c_i*-th component.*

Lemma 2 provides an explicit way to measure the gap between ELBO and the model likelihood for all tasks using the mixture model. When $|C'|=0$, $D^{\star}_{diff}=0$ and the discrepancy $disc^{\star}_{\mathcal{L}}(\mathcal{P}_{a_i},\mathbb{P}^0_{a_i})$ is very small, this bound is tight.

5 Dynamic expansion graph model (DEGM)

According to Theorem 3, achieving an optimal GB requires each mixture component to model a unique task only. However, adding dynamically a new component whenever learning a new task, leads to ever-increasing memory and computation requirements. For addressing the trade-off between task learning effectiveness and memory efficiency, we propose a novel expansion mechanism. This would selectively allow the newly created component to reuse some of the parameters and thus transfer information from existing components, according to a knowledge similarity criterion.

5.1 Basic and specific nodes in DEGM

A component trained during LLL, with independent parameters, is called a basic node and can be transferred to be used in other tasks. Therefore, a basic node can be seen as a knowledge source for other processing nodes in DEGM. Meanwhile, we also have specific nodes associated with the novel information acquired from a new task $\mathcal{T}_{(t+1)}$, after also considering reusing the information from the basic nodes.

Let $q_{\omega_i}(\mathbf{z}\,|\,\mathbf{x})$ and $p_{\theta_i}(\mathbf{x}\,|\,\mathbf{z})$ represent the encoding and decoding distributions, respectively, as in Eq. (1). We implement the basic node using paired sub-models, for encoding and decoding information. We consider two sub-inference models, $f_{\tilde{\omega}_i}\colon \mathcal{X}\to\tilde{\mathcal{Z}}$ and $f_{\omega'_i}\colon\tilde{\mathcal{Z}}\to\mathcal{Z}$ for modelling $q_{\omega_i}(\mathbf{z}\,|\,\mathbf{x})$, expressed by $f_{\tilde{\omega}_i}\circ f_{\omega'_i}\colon\mathcal{X}\to\mathcal{Z}$, where $\tilde{\mathcal{Z}}$ is an intermediate latent representation space with the dimension larger than $\mathcal{Z}$, $|\tilde{\mathcal{Z}}|>|\mathcal{Z}|$. We use two networks, $g_{\tilde{\theta}_i}\colon\mathcal{X}\to\tilde{\mathcal{X}}$ and $g_{\theta'_i}\colon\tilde{\mathcal{X}}\to\mathcal{X}$, for modelling $p_{\theta_i}(\mathbf{x}\,|\,\mathbf{z})$ which is expressed by $g_{\tilde{\theta}_i}\circ g_{\theta'_i}\colon\mathcal{Z}\to\mathcal{X}$, where $\tilde{\mathcal{X}}$ is an intermediate representation space, $|\tilde{\mathcal{X}}|<|\mathcal{X}|$. Since a basic node has two connectable sub-models $\{f_{\tilde{\omega}_i},g_{\tilde{\theta}_i}\}$, building a specific j-th node only requires two separate sub-models $\{f_{\omega'_j},g_{\theta'_j}\}$ which would be connected with the sub-models $\{f_{\tilde{\omega}_i},g_{\tilde{\theta}_i}\}$ from all basic nodes, $i=1,\dots,K$ to form a graph structure in DEGM. In the following section, we describe how DEGM expands its architecture during LLL.

5.2 Training sub-graph structures in DEGM

Let us assume that we have trained t nodes after learning t tasks, where K nodes, $K<t$, represent basic nodes $\mathcal{G}=\{B_1,\dots,B_K\}$, and $(t-K)$ nodes belong to specific nodes $\mathcal{S}=\{S_1,\dots,S_{(t-K)}\}$. Let $\mathcal{GI}(\cdot)$ and $\mathcal{SI}(\cdot)$ be the functions that return the node index for $\mathcal{G}$ and $\mathcal{S}$. Each $B_i\in\mathcal{G}$ has four sub-models $\{f_{\tilde{\omega}_{i^*}},f_{\omega'_{i^*}},g_{\tilde{\theta}_{i^*}},g_{\theta'_{i^*}}\}$ where $i^*=\mathcal{GI}(i)$, and each $S_i\in\mathcal{S}$ has only two sub-models $\{f_{\omega'_{i'}},g_{\theta'_{i'}}\}$, where $i'=\mathcal{SI}(i)$. Let us consider $\mathbf{V}\in\mathbb{R}^{t\times t}$ an adjacency matrix representing the directed graph edges from $\mathcal{S}$ to $\mathcal{G}$. $V(i,j)$ is the directed edge from nodes i to j, and $\mathbf{V}$ is used for expanding the architecture whenever necessary. After learning t-th task, we set a new task $\mathcal{T}_{t+1}$ for training the mixture model with Q^S_{t+1}. We evaluate the efficiency of using each element of $B_i\in\mathcal{G}$, $i=1,\dots,K$ by calculating $\mathcal{L}_{ELBO}(\mathbf{x}_j;B_i)$ on $\mathbf{x}_j\sim Q^S_{t+1}$, $j=1,\dots,n$, ($n=1000$ in experiments). For assessing the novelty of a given task $\mathcal{T}_{t+1}$, with respect to the knowledge already acquired, we consider the following criterion :

$$ks_i=\big|\mathcal{L}_{ELBO}(B_i)-\mathbb{E}_{\mathbf{x}\sim Q^S_{(t+1)}}\mathcal{L}_{ELBO}(\mathbf{x};B_i)\big|, \tag{16}$$

where $i=1,\dots,K$ and $\mathcal{L}_{ELBO}(B_i)$ is the best log-likelihood estimated by B_i on its previously assigned task and we form $\mathcal{K}=\{ks_1,\dots,ks_K\}$. Similar log-likelihood evaluations were used for selecting components in (Lee et al. 2020; Rao et al. 2019). However, in our approach we develop a graph-based structure by defining Basic and Specific nodes based on analyzing $\mathcal{K}$, as explained in the following.

Building a Basic node. A Basic node is added to the DEGM model when the incoming task is assessed as completely novel. If $\min(\mathcal{K})>\tau$, where τ is a threshold, then we set $V(t+1,\mathcal{GI}(i))=0$, $i=1,\dots,K$ and DEGM builds a basic node which is added to $\mathcal{G}$. During the $(t+1)$-th task learning, we only optimize the parameters of the $(t+1)$-th component by using the loss function from Eq. (1) with the given task' dataset.

Building a Specific node. A Specific node is built when the incoming task is related to the already learned knowledge, encoded by the basic nodes. If $\min(\mathcal{K})\le\tau$, then we update $\mathbf{V}$ by calculating the importance weight $V(t+1,\mathcal{GI}(i))=(w^*-ks_i)/\sum_{j=1}^{K}(w^*-ks_j)$, $w^*=\sum_{j=1}^{K}ks_j$, $i=1,\dots,K$, where we denote $\pi_i=V(t+1,\mathcal{GI}(i))$ for simplification. According to the updated $\mathbf{V}$, we built a new sub-inference model $f_{\omega'_{(t+1)}}$, based on a set of sub-models $\{f_{\tilde{\omega}_{i^*}}\,|\,i^*=\mathcal{GI}(i),i=1,\dots,K\}$, as $\sum_{i=1}^{K}\pi_i f_{\tilde{\omega}_{i^*}}\odot f_{\omega'_{(t+1)}}(\mathbf{x})$, which represents $\mathbf{z}=\sum_{i=1}^{K}\pi_i\mathbf{z}_i$, where each $\mathbf{z}_i=f_{\tilde{\omega}_{i^*}}\odot f_{\omega'_{(t+1)}}(\mathbf{x})$ is weighted by π_i. In Fig. 1, we show the structure of the decoder, where an identity function implemented by the input layer distributes the latent variable $\mathbf{z}$ to each $g_{\tilde{\theta}_{i^*}}(\mathbf{z}),i^*=\mathcal{GI}(1),\dots,\mathcal{GI}(K)$, leading to $\tilde{\mathbf{x}}=\sum_{i=1}^{K}\pi_i g_{\tilde{\theta}_{i^*}}(\mathbf{z})$, where the intermediate feature information from $\mathcal{G}$ is weighted by π_i. We then build a new sub-decoder $g_{\theta'_{(t+1)}}(\tilde{\mathbf{x}})$, that takes $\tilde{\mathbf{x}}$ as the input and outputs the reconstruction of $\mathbf{x}$ enlarging $\mathcal{S}$ with $S_{t-K+1} \in \mathcal{S}$. The procedure for building the graph and how a new node connects to the elements from $\mathcal{G}$, as a sub-graph in DEGM, is shown in Fig. 1. The importance of processing modules during LLL was considered in (Aljundi, Kelchtermans, and Tuytelaars 2019; Jung et al. 2020). However, DEGM is the first model where this mechanism is used for the dynamic expansion of a graph model. Additionally, different from existing methods, the importance weighting approach proposed in this paper regularizes the transferable information during both the inference and generation processes. In the following, we propose a new objective function for the training of a Specific node, which also guarantees a lower bound to the marginal log-likelihood.

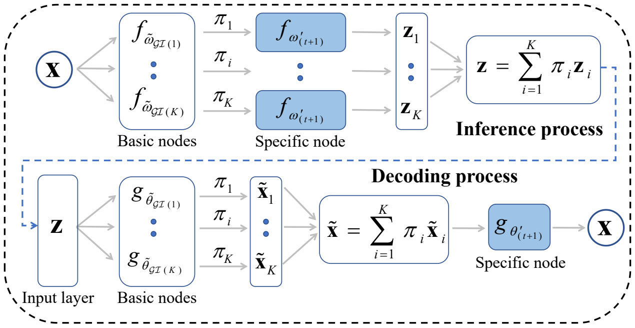


Figure 1: The graph structure in DEGM where an image is firstly processed by K Basic nodes to which the newly created Specific node connects during the inference process. This procedure is also performed at the decoding process.

Theorem 4 *A Specific node is built for learning the $(t+1)$-th task, which forms a sub-graph structure and can be trained by using a valid lower bound (ELBO) (See details in Appendix-J.1 from SM*[1]*) :*

$$\begin{aligned} &\mathcal{L}_{MELBO}(\mathbf{x};\mathcal{M}_{(t+1)}) =: \\ &\mathbb{E}_{Q(\mathbf{z})}\left[\log p_{\theta'_{(t+1)} \circ \{\tilde{\theta}_{\mathcal{GI}(1)},\ldots,\tilde{\theta}_{\mathcal{GI}(K)}\}}(\mathbf{x} \mid \mathbf{z})\right] \\ &-\sum\nolimits_{i=1}^{K} \pi_i KL\left(Q_{\tilde{\omega}_{\mathcal{GI}(i)} \circ \omega'_{(t+1)}}(\mathbf{z} \mid \mathbf{x}) \,||\, p\left(\mathbf{z}_i\right)\right), \end{aligned} \tag{17}$$

where $q_{\tilde{\omega}_{\mathcal{GI}(i)}} \circ q_{\omega'_{(t+1)}}(\mathbf{z} \,|\, \mathbf{x})$ is the density function form of $Q_{\tilde{\omega}_{\mathcal{GI}(i)} \circ \omega'_{(t+1)}}(\mathbf{z} \,|\, \mathbf{x})$.

We implement the variational distribution $Q(\mathbf{z})$ by $\sum_{i=1}^{K} \pi_i Q_{\tilde{\omega}_{\mathcal{GI}(i)} \circ \omega'_{(t+1)}}(\mathbf{z} \,|\, \mathbf{x})$, which is a mixture inference model. The difference in Eq. (17) from $q_\omega(\mathbf{z} \,|\, \mathbf{x})$ in Eq. (1) is that $Q(\mathbf{z})$ is much more knowledge expressive by reusing the transferable information from previously learnt knowledge while also reducing the computational cost, by only updating the components $\{\omega'_{(t+1)}, \theta'_{(t+1)}\}$ when learning the $(t+1)$-th task. The first term in the RHS of Eq. (17) is the negative reconstruction error evaluated by a single decoder. The second term consists of the sum of all KL terms weighted by their corresponding edge values π_i.

Model selection. We evaluate the negative marginal log-likelihood, (Eq. (1) for Basic nodes, and Eq. (17) for Specific nodes) for testing data samples after LLL. Then we choose the node with the highest likelihood for the evaluation. This mechanism can allow DEGM to infer an appropriate node without task labels (See details in Appendix-J.3 from SM[1]).

6 Experimental results

6.1 Unsupervised lifelong learning benchmark

Setting. We define a novel benchmark for the log-likelihood estimation under LLL, explained in Appendix-K from SM[1]. We consider learning multiple tasks defined within a single domain, such as MNIST (LeCun et al. 1998) and Fashion (Xiao, Rasul, and Vollgraf 2017). Following from (Burda, Grosse, and Salakhutdinov 2015) we divide MNIST and Fashion into five tasks (Zenke, Poole, and Ganguli 2017), called Split MNIST (S-M) and Split- Fashion (S-F). We use the cross-domain setting where we aim to learn a sequence of domains, called COFMI, consisting of: Caltech 101 (Fei-Fei, Fergus, and Perona 2007), OMNIGLOT (Lake, Salakhutdinov, and Tenenbaum 2015), Fashion, MNIST, InverseFashion (IFashion) where each task is associated with a distinct dataset from all others. All databases are binarized.

Baselines. We adapt the network architecture from (Burda, Grosse, and Salakhutdinov 2015) and consider several baselines. A single VAE with GR is called ELBO-GR and when considering IWELBO bounds it becomes IWELBO-GR-K' where K' represent the number of weighted samples. We call DEGM with ELBO and IWELBO bounds as DEGM-ELBO and DEGM-IWELBO-K', respectively. We also compare with LIMix (Ye and Bors 2021e) and implement CN-DPM (Lee et al. 2020) with the optimal setting, namely CN-DPM* (See details in Appendix-L.1 from SM[1]).

Methods	S-M	S-F	COFMI
ELBO-GR	-98.23	-240.58	-177.47
IWELBO-GR-50	-93.57	-236.66	-172.10
IWELBO-GR-5	-95.80	-238.08	-176.21
ELBO-GR*	-98.36	-243.91	-180.50
IWELBO-GR*-50	-91.23	-236.90	-188.9
CN-DPM*-IWELBO-50	-95.91	-237.47	-184.19
LIMix-IWELBO-50	-95.74	-237.48	-184.32
DEGM-ELBO	-93.51	-238.54	-168.91
DEGM-IWELBO-50	**-88.04**	**-233.76**	**-163.27**
DEGM-IWELBO-5	-91.44	-235.93	-164.99

Table 1: Results for Split MNIST, Split Fashion and COFMI.

Results. The testing data log-likelihood is estimated by the IWELBO bounds (Burda, Grosse, and Salakhutdinov 2015) with $K' = 5000$. we perform five independent runs for S-M/S-F and COFMI data. The average results are reported in Table 1 where '*' denotes that the model uses two stochastic layers (See details in Appendix-K.1 from SM[1]) which can further improve the performance with the IWELBO bound, according to IWELBO-GR*-50. The proposed DEGM-IWELBO-50 obtains the best results when using the IWELBO bound, for both S-M and S-F settings. The proposed DEGM also outperforms other baselines on COFMI, which represents a more challenging task than S-M/S-F. The detailed results for each task are reported in Appendix-K.2 from SM[1], showing that ELBO-GR* and IWELBO-GR*-50 tend to degenerate the performance on the early tasks under the cross-domain learning setting when compared with VAEs that do not use two stochastic layers. Details, such as the number of Basic and Specific nodes used are provided in Appendix-K.2 from SM[1].

6.2 Comparing to lifelong learning models

Baselines. The first baseline consists of dynamically creating a new VAE to adapt to a new task, namely DEGM-2, which is a strong baseline and would achieve the best performance for each new task. Meanwhile, Batch Ensemble (BE) (Wen, Tran, and Ba 2020) is designed for classification tasks. We implement each component of BE as a VAE. DEGM is trained by using ELBO without considering the IWELBO bound aiming for using a small network. The number of parameters required by various models is listed in Appendix-L.5 from SM[1].

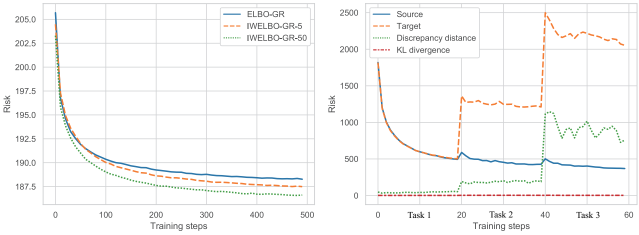
We train various models under CCCOSCZC lifelong learning setting, where each task is associated with a dataset: CelebA (Liu et al. 2015), CACD (Chen, Chen, and Hsu 2014), 3D-Chair (Aubry et al. 2014), Ommiglot (Lake, Salakhutdinov, and Tenenbaum 2015), ImageNet* (Krizhevsky, Sutskever, and Hinton 2012), Car (Yang et al. 2015), Zappos (Yu and Grauman 2017), CUB (Wah et al. 2010) (detailed dataset setting is provided in Appendix-L.1 from SM[1]). The square loss (SL) is used to evaluate the reconstruction quality and other criteria are reported in Appendix-L.3 from SM[1]. The threshold for adding a new component for DEGM is $\tau = 600$ on CCCOSCZC and the results are reported in Table 2. We can observe that the proposed DEGM outperforms other existing lifelong learning models and achieves a close result to DEGM-2 which trains individual VAEs for each task and requires more parameters. Visual results are shown in Appendix-L.7 from SM[1].

Dataset	BE	LIMix	LGM	DEGM	DEGM-2	CN-DPM*
CelebA	213.9	214.2	535.6	229.2	217.0	215.4
CACD	414.9	353.5	814.3	368.3	281.95	347.3
3D-Chair	649.1	353.1	2705.9	324.0	291.46	513.8
Omniglot	875.1	351.1	5958.9	225.6	195.7	343.2
ImageNet*	758.4	778.5	683.1	689.6	652.8	769.1
Car	745.1	688.19	583.7	588.8	565.9	709.8
Zappos	451.1	283.4	431.2	263.4	275.8	280.7
CUB	492.0	400.7	330.2	461.3	569.6	638.6
Average	575.0	427.8	1505.4	393.8	381.3	477.2

Table 2: Results under CCCOSCZC lifelong learning.

6.3 Empirical results for theoretical analysis

We train a VAE on the binarized Caltech 101 database and use it to generate a pseudo set of images consistent with the dataset. Then, ELBO-GR, IWELBO-GR-K', where $K' \in \{5, 50\}$ corresponds to the number of weighted samples used for training on the joint dataset consisting of the pseudo set and a training set from a second task (Fashion). We evaluate the average target risk (LHS of Eq. (10)) for these models in order to investigate the tightness between the negative log-likelihood (NLL) and LELBO, since NLL is a lower bound to LELBO. The IWELBO bounds with 5000 weighted samples results are shown in Fig. 2a. Although, Lemma 1 considers the Gaussian decoder, VAEs with a Bernoulli decoder, corresponding to the IWELBO bound, indicates that IWELBO-GR-50 is a lower bound to IWELBO-GR-5 and ELBO-GR, which empirically proves $\mathcal{L}_{LELBO_{50}} \leq \mathcal{L}_{LELBO_5}$ when the generator distribution is fixed and $|KL_1 - KL_2| = 0$, as discussed in Lemma 1.



(a) Target risk as in Eq. (10). (b) Evaluation of Eq. (9).

Figure 2: The estimation of the target and source risks.

We also train a VAE whose decoder outputs the mean vector of a Gaussian distribution with the Identity matrix as its covariance, under MNIST, Fashion and IFashion LLL, where pixel values of all images are within $(0, 255)$. The reconstruction error ELBO is normalized by dividing with the image size (28×28), as in (Park, Kim, and Kim 2019). We evaluate the risk and the discrepancy distance for each training epoch, according to Eq. (9) from Lemma 1 and the results are provided in Fig. 2b, where the source risk (the first term in RHS of Eq. (9)) keeps stable and the discrepancy distance $disc_{\mathcal{L}}(\cdot)$, Eq. (2), represented within $\mathcal{R}_A(\cdot)$, increases while learning more tasks. The 'KL divergence,' calculated as $|KL_1 - KL_2|$, shown in Fig. 2b increases slowly. This demonstrates that the discrepancy distance plays an important role on shrinking the gap for the GB. An ablation study, demonstrating the effectiveness of the proposed expansion mechanism, is provided in Appendix-L.4 from SM[1].

7 Conclusion

In this paper we analyze the forgetting behaviour of VAEs by finding an upper bound on the negative marginal log-likelihood, called LELBO. This provides insights into the generalization performance on the target distribution when the source distribution evolves continuously over time during lifelong learning. We further develop a Dynamic Expansion Graph Model (DEGM), which adds new Basic and Specific components to the network, depending on a knowledge novelty criterion. DEGM can significantly reduce the accumulated errors caused by the forgetting process. The empirical and theoretical results verify the effectiveness of the proposed DEGM methodology.

References

Achille, A.; Eccles, T.; Matthey, L.; Burgess, C.; Watters, N.; Lerchner, A.; and Higgins, I. 2018. Life-long disentangled representation learning with cross-domain latent homologies. In *Advances in Neural Inf. Proc. Systems (NIPS)*, 9873–9883.

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

Aubry, M.; Maturana, D.; Efros, A. A.; Russell, B. C.; and Sivic, J. 2014. Seeing 3D chairs: exemplar part-based 2D-3D alignment using a large dataset of CAD models. In *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 3762–3769.

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

Cemgil, T.; Ghaisas, S.; Dvijotham, K.; Gowal, S.; and Kohli, P. 2020. The Autoencoding Variational Autoencoder. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, 15077–15087.

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*.

Chen, B.-C.; Chen, C.-S.; and Hsu, W. H. 2014. Cross-Age Reference Coding for Age-Invariant Face Recognition and Retrieval. In *Proc. European Conf on Computer Vision (ECCV), vol. LNCS 8694*, 768–783.

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.

Doersch, C. 2016. Tutorial on variational autoencoders. *arXiv preprint arXiv:1606.05908*.

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

Fei-Fei, L.; Fergus, R.; and Perona, P. 2007. Learning generative visual models from few training examples: An incremental Bayesian approach tested on 101 object categories. *Computer Vision and Image Understanding*, 106: 59–70.

French, R. M. 1999. Catastrophic forgetting in connectionist networks. *Trends in cognitive sciences*, 3(4): 128–135.

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.

Guo, Y.; Liu, M.; Yang, T.; and Rosing, T. 2020. Improved Schemes for Episodic Memory-based Lifelong Learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, 1023–1035.

Higgins, I.; Matthey, L.; Pal, A.; Burgess, C.; Glorot, X.; Botvinick, M.; Mohamed, S.; and Lerchner, A. 2017. β-VAE: Learning basic visual concepts with a constrained variational framework. In *Proc. Int. Conf. on Learning Representations (ICLR)*, 1–13.

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.

Jung, H.; Jung, M.; and Kim, J. 2016. Less-forgetting learning in deep neural networks. *arXiv preprint arXiv:1607.00122*.

Jung, S.; Ahn, H.; Cha, S.; and Moon, T. 2020. Continual Learning with Node-Importance based Adaptive Group Sparse Regularization. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, 3647–3658.

Kim, M.; and Pavlovic, V. 2020. Recursive Inference for Variational Autoencoders. In *Advances in Neural Information Processing Systems*, volume 33, 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*.

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2012. ImageNet classification with deep convolutional neural networks. In *Proc. Advances in Neural Inf. Proc. Systems (NIPS)*, 1097–1105.

Kuroki, S.; Charoenphakdee, N.; Bao, H.; Honda, J.; Sato, I.; and Sugiyama, M. 2019. Unsupervised domain adaptation based on source-guided discrepancy. In *Proc. AAAI Conf. on Artificial Intelligence*, volume 33, 4122–4129.

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.; Ha, J.; Zhang, D.; and Kim, G. 2020. A Neural Dirichlet Process Mixture Model for Task-Free Continual Learning. In *Proc. 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.

Liu, Z.; Luo, P.; Wang, X.; and Tang, X. 2015. Deep learning face attributes in the wild. In *Proc. of IEEE Int. Conf. on Computer Vision (ICCV)*, 3730–3738.

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.

Mansour, Y.; Mohri, M.; and Rostamizadeh, A. 2009. Domain adaptation: Learning bounds and algorithms. In *Proc. Conf. on Learning Theory (COLT), arXiv preprint arXiv:2002.06715*.

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*.

Pan, P.; Swaroop, S.; Immer, A.; Eschenhagen, R.; Turner, R.; and Khan, M. E. E. 2020. Continual Deep Learning by Functional Regularisation of Memorable Past. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, 4453–4464.

Park, Y.; Kim, C.; and Kim, G. 2019. Variational Laplace autoencoders. In *Proc. Int. Conf. on Machine Learning (ICML), vol. PMLR 97*, 5032–5041.

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.

Riemer, M.; Cases, I.; Ajemian, R.; Liu, M.; Rish, I.; Tu, Y.; ; and Tesauro, G. 2019. Learning to Learn without Forgetting By Maximizing Transfer and Minimizing Interference. In *Proc. Int. Conf. on Learning Representations (ICLR), arXiv preprint arXiv:1810.11910*.

Shin, H.; Lee, J. K.; Kim, J.; and Kim, J. 2017. Continual learning with deep generative replay. In *Advances in Neural Information Proc. Systems (NIPS)*, 2990–2999.

Sobolev, A.; and Vetrov, D. 2019. Importance Weighted Hierarchical Variational Inference. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, 601–613.

Srivastava, A.; Valkov, L.; Russell, C.; Gutmann, M. U.; and Sutton, C. 2017. VEEGAN: Reducing mode collapse in GANs using implicit variational learning. In *Advances in Neural Inf. Proc. Systems (NIPS)*, 3308–3318.

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

Wah, C.; Branson, S.; Welinder, P.; Perona, P.; and Belongie, S. 2010. The Caltech-UCSD Birds-200 dataset. Technical Report CNS-TR-2010-001, California Institute of Technology.

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, H.; Rasul, K.; and Vollgraf, R. 2017. Fashion-MNIST: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*.

Yang, L.; Luo, P.; Change Loy, C.; and Tang, X. 2015. A large-scale car dataset for fine-grained categorization and verification. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 3973–3981.

Ye, F.; and Bors, A. 2021a. Lifelong Teacher-Student Network Learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

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. 2021b. Deep Mixture Generative Autoencoders. *IEEE Transactions on Neural Networks and Learning Systems*, 1–15.

Ye, F.; and Bors, A. G. 2021c. InfoVAEGAN: Learning Joint Interpretable Representations by Information Maximization and Maximum Likelihood. In *Proc. IEEE Int. Conf. on Image Processing (ICIP)*, 749–753.

Ye, F.; and Bors, A. G. 2021d. Learning joint latent representations based on information maximization. *Information Sciences*, 567: 216–236.

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

Ye, F.; and Bors, A. G. 2021f. Lifelong Mixture of Variational Autoencoders. *IEEE Transactions on Neural Networks and Learning Systems*, 1–14.

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

Yu, A.; and Grauman, K. 2017. Semantic Jitter: Dense Supervision for Visual Comparisons via Synthetic Images. In *Proc. IEEE Int. Conf. on Computer Vision (ICCV)*, 5571–5580.

Zenke, F.; Poole, B.; and Ganguli, S. 2017. Continual learning through synaptic intelligence. In *Proc. of Int. Conf. on Machine Learning (ICML), vol. PLMR 70*, 3987–3995.