Supplemental Material : Lifelong Generative Modelling Using Dynamic Expansion Graph Model

December 11, 2021

Contents

A The proof of Theorem 1

Assumption 1 *Let $\mathcal{X}$ be a metric space that satisfies $\mathcal{L}(a,b) \leq \mathcal{L}(a,c) + \mathcal{L}(c,b)$ where the loss function $\mathcal{L}(\cdot)$ is a metric and $a, b, c \in \mathcal{X}$.*

Based on Assumption 1, we provide the detailed proof as follows :

Proof. Let $\mathcal{P}_i$ and $\tilde{\mathcal{P}}_i$ be two domains over $\mathcal{X}$. Then for $h_{\mathcal{P}_i}^\star = \arg\min_{h \in \mathcal{H}} \mathcal{R}_{\mathcal{P}_i}(h, f_{\mathcal{P}_i})$ and $h_{\tilde{\mathcal{P}}_i}^\star = \arg\min_{h \in \mathcal{H}} \mathcal{R}_{\tilde{\mathcal{P}}_i}(h, f_{\tilde{\mathcal{P}}_i})$ where $f_{\tilde{\mathcal{P}}_i} \in \mathcal{H}$ is the ground truth function (identity function under the encoder-decoding process) for $\tilde{\mathcal{P}}_i$.

Then according to the triangle inequality property of $\mathcal{L}$, applied twice, we have :

$$\mathcal{R}_{\mathcal{P}_i}(h, f_{\mathcal{P}_i}) \leq \mathcal{R}_{\mathcal{P}_i}(h, h_{\tilde{\mathcal{P}}_i}^*) + \mathcal{R}_{\mathcal{P}_i}(h_{\tilde{\mathcal{P}}_i}^*, h_{\mathcal{P}_i}^*) + \mathcal{R}_{\mathcal{P}_i}(h_{\mathcal{P}_i}^*, f_{\mathcal{P}_i}) \tag{1}$$

Eq. (1) holds because, after applying twice the triangle inequality, $\mathcal{L}(a,b) \leq \mathcal{L}(a,c) + \mathcal{L}(c,d) + L(d,b)$ where a, b, c, d are $h(\mathbf{x}), f_{\mathcal{P}_i}(\mathbf{x}), h_{\tilde{\mathcal{P}}_i}^*(\mathbf{x}), h_{\mathcal{P}_i}^*(\mathbf{x})$ and $\mathbf{x}$ is sampled from the same domain $\mathcal{P}_i$. According to the definition of discrepancy distance (See Definition 2 of the paper), we define the discrepancy distance between $\mathcal{P}_i$ and $\tilde{\mathcal{P}}_i$ as :

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

We rewrite the above equation as :

$$disc_{\mathcal{L}}(\mathcal{P}_i, \tilde{\mathcal{P}}_i) = \sup_{(h,h') \in \mathcal{H}} \left| \mathcal{R}_{\mathcal{P}_i}(h, h') - \mathcal{R}_{\tilde{\mathcal{P}}_i}(h, h') \right|. \tag{3}$$

We consider h' to be $h_{\tilde{\mathcal{P}}_i}^*$ in Eq. (3) and we have :

$$\sup_{(h,h') \in H} \left| \mathcal{R}_{\mathcal{P}_i}(h, h') - \mathcal{R}_{\tilde{\mathcal{P}}_i}(h, h') \right| \geq \left| \mathcal{R}_{\mathcal{P}_i}(h, h_{\tilde{\mathcal{P}}_i}^*) - \mathcal{R}_{\tilde{\mathcal{P}}_i}(h, h_{\tilde{\mathcal{P}}_i}^*) \right| \tag{4}$$

We also know that :

$$\mathcal{R}_{\mathcal{P}_i}(h, h_{\tilde{\mathcal{P}}_i}^*) \leq \left| \mathcal{R}_{\mathcal{P}_i}(h, h_{\tilde{\mathcal{P}}_i}^*) - \mathcal{R}_{\tilde{\mathcal{P}}_i}(h, h_{\tilde{\mathcal{P}}_i}^*) \right| + \mathcal{R}_{\tilde{\mathcal{P}}_i}(h, h_{\tilde{\mathcal{P}}_i}^*) \tag{5}$$

Therefore, we can replace the first term of the right hand side of Eq. (1) by the right hand side of Eq. (5), resulting in :

$$\mathcal{R}_{\mathcal{P}_i}(h, f_{\mathcal{P}_i}) \leq \mathcal{R}_{\tilde{\mathcal{P}}_i}(h, h_{\tilde{\mathcal{P}}_i}^*) + \left| \mathcal{R}_{\mathcal{P}_i}(h, h_{\tilde{\mathcal{P}}_i}^*) - \mathcal{R}_{\tilde{\mathcal{P}}_i}(h, h_{\tilde{\mathcal{P}}_i}^*) \right| + \mathcal{R}_{\mathcal{P}_i}(h_{\mathcal{P}_i}^*, h_{\tilde{\mathcal{P}}_i}^*) + \mathcal{R}_{\mathcal{P}_i}(h_{\mathcal{P}_i}^*, f_{\mathcal{P}_i}) \tag{6}$$

Then the second term, representing the absolute value of the difference, in RHS of Eq. (6) can be replaced by $disc_{\mathcal{L}}(\mathcal{P}_i, \tilde{\mathcal{P}}_i)$ from Eq. (4)), since the discrepancy distance between two distributions is an upper bound to this absolute value, resulting in :

$$\mathcal{R}_{\mathcal{P}_i}(h, f_{\mathcal{P}_i}) \leq \mathcal{R}_{\tilde{\mathcal{P}}_i}(h, h_{\tilde{\mathcal{P}}_i}^*) + disc_{\mathcal{L}}(\mathcal{P}_i, \tilde{\mathcal{P}}_i) + \mathcal{R}_{\mathcal{P}_i}(h_{\mathcal{P}_i}^*, h_{\tilde{\mathcal{P}}_i}^*) + \mathcal{R}_{\mathcal{P}_i}(h_{\mathcal{P}_i}^*, f_{\mathcal{P}_i}) \tag{7}$$

From **Definition 3** from the paper, we know that $disc_{\mathcal{L}}(\mathcal{P}_i, \tilde{\mathcal{P}}_i) \leq disc_{\mathcal{L}}^\star(\mathcal{P}_i, \tilde{\mathcal{P}}_i)$. This allows use to replace $disc_{\mathcal{L}}(\mathcal{P}_i, \tilde{\mathcal{P}}_i)$ by using $disc_{\mathcal{L}}^\star(\mathcal{P}_i, \tilde{\mathcal{P}}_i)$ in Eq. (7), resulting in :

$$\mathcal{R}_{\mathcal{P}_i}(h, f_{\mathcal{P}_i}) \leq \mathcal{R}_{\tilde{\mathcal{P}}_i}(h, h_{\tilde{\mathcal{P}}_i}^*) + disc_{\mathcal{L}}^\star(\mathcal{P}_i, \tilde{\mathcal{P}}_i) + \mathcal{R}_{\mathcal{P}_i}(h_{\mathcal{P}_i}^*, h_{\tilde{\mathcal{P}}_i}^*) + \mathcal{R}_{\mathcal{P}_i}(h_{\mathcal{P}_i}^*, f_{\mathcal{P}_i}) \tag{8}$$

Eq. (8) proves **Theorem 1** and a similar proof can be found in Theorem 8 from Mansour et al. [2009].

B The proof of Theorem 2

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 :

$$\frac{1}{t} \sum_{i=1}^{t} \mathcal{R}_{\mathcal{P}_i}(h, f_{\mathcal{P}_i}) \leq \mathcal{R}_{\mathbb{P}^{t-1} \otimes \tilde{\mathcal{P}}_t}\left(h, h_{\mathbb{P}^{t-1} \otimes \tilde{\mathcal{P}}_t}^*\right) + \mathcal{R}_A\left(\mathcal{P}_{(1:t)}, \mathbb{P}^{t-1} \otimes \tilde{\mathcal{P}}_t\right), \tag{9}$$

where $\mathcal{P}_{(1:t)}$ is the mixture distribution $\{\mathcal{P}_1 \otimes \mathcal{P}_2, \ldots, \otimes \mathcal{P}_t\}$. As it can be seen in Eq. (9), the performance on the target domain is largely depending on the discrepancy term even if $\mathcal{M}$ minimizes the source risk well. In the following, we

provide an analytical bound that considers all previously learnt distributions.

$$\frac{1}{t}\sum_{i=1}^{t} \mathcal{R}_{\mathcal{P}_i}\left(h, f_{\mathcal{P}_i}\right) \leq \mathcal{R}_{\mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t}\left(h, h^*_{\mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t}\right) + \mathrm{Err}^a + \mathrm{Err}^d, \tag{10}$$

where $\mathrm{Err}^d \geq 0$ evaluates the difference on the two risk terms, expressed by:

$$\sum_{k=1}^{t-1}\left\{\mathcal{R}_{\mathbb{P}^{(t-k)}}\left(h, h^*_{\mathbb{P}^{(t-l)}}\right) - \mathcal{R}_{\mathbb{P}^{(t-1-k)}\otimes\tilde{\mathcal{P}}_{(t-k)}}\left(h, h^*_{\mathbb{P}^{(t-1-k)}\otimes\tilde{\mathcal{P}}_{(t-k)}}\right)\right\}, \tag{11}$$

where $\mathcal{R}_{\mathbb{P}^0\otimes\tilde{\mathcal{P}}_1}(h, h^*_{\mathbb{P}^0\otimes\tilde{\mathcal{P}}_1}) = \mathcal{R}_{\tilde{\mathcal{P}}_1}(h, h^*_{\tilde{\mathcal{P}}_1})$. Err^a is the accumulated error term expressed by :

$$\sum_{k=1}^{t-2}\left\{\mathcal{R}_A\left(\mathbb{P}^{(t-1-k)}\otimes\tilde{\mathcal{P}}_{(t-k)}, \mathbb{P}^{(t-k)}\right)\right\} + \mathcal{R}_A\left(\mathcal{P}_{(1:t)}, \mathbb{P}^{(t-1)}\otimes\tilde{\mathcal{P}}_t\right) + \mathcal{R}_A\left(\tilde{\mathcal{P}}_1, \mathbb{P}^1\right). \tag{12}$$

From Eq. (12), we can observe that while learning more tasks (t is increased) during the lifelong learning, the gap on the GB for $\mathcal{M}$ tends to be large since Err^a increases. This can also explain why GR fails when learning a long sequence of tasks Ramapuram et al. [2020], Ye and Bors [2020]. Additionally, the term $\mathcal{R}_A(\mathbb{P}^{(t-1-k)}\otimes\tilde{\mathcal{P}}_{(t-k)}, \mathbb{P}^{(t-k)})$ and Err^d tend to be small when the discrepancy $disc^*_{\mathcal{L}}(\mathbb{P}^i, \mathbb{P}^{(i-1)}\otimes\tilde{\mathcal{P}}_i)$ is equal to 0 in each task learning ($i = 1, \ldots, t$). This is achieved by the optimal generator distribution that approximates $\mathbb{P}^{(i-1)}\otimes\tilde{\mathcal{P}}_i$ exactly in each task learning (See also other explanation in Proposition 3 in Appendix I).

Proof. Firstly, we can derive the bound according to Theorem 1 :

$$\mathcal{R}_{\mathcal{P}_{(i:t)}}\left(h, f_{\mathcal{P}_{(i:t)}}\right) \leq \mathcal{R}_{\mathbb{P}^{(t-1)}\otimes\tilde{\mathcal{P}}_t}\left(h, h^*_{\mathbb{P}^{(t-1)}\otimes\tilde{\mathcal{P}}_t}\right) + disc^*_{\mathcal{L}}\left(\mathcal{P}_{(1:t)}, \mathbb{P}^{(t-1)}\otimes\tilde{\mathcal{P}}_t\right) + \varepsilon\left(\mathcal{P}_{(1:t)}, \mathbb{P}^{(t-1)}\otimes\tilde{\mathcal{P}}_t\right) \tag{13}$$

where $\mathcal{P}_{(1:t)}$ represents the mixture distribution $\{\mathcal{P}_1 \otimes \mathcal{P}_2, \ldots, \otimes\mathcal{P}_t\}$. Let $\rho_{(1:t)}(\mathbf{x})$ represent the density function for $\mathcal{P}_{(1:t)}$ and $\rho_{(i)}(\mathbf{x})$ the density function for $\mathcal{P}_i$. Since $\mathcal{P}_{(1:t)}$ is the mixture distribution and its density is expressed by $\rho_{(1:t)}(\mathbf{x}) = \frac{1}{t}\sum_{i=1}^{t}\rho_{(i)}(\mathbf{x})$. We know that $\mathcal{R}_{\mathcal{P}_{(i:t)}}(h, f_{\mathcal{P}_{(1:t)}})$ can be rewritten as the integral form $\int \rho_{(i:t)}(\mathbf{x})\mathcal{L}(h, f_{P_{(1:t)}})\,\mathrm{d}\mathbf{x}$. We then take $\rho_{(1:t)}(\mathbf{x}) = \frac{1}{t}\sum_{i=1}^{t}\rho_{(i)}(\mathbf{x})$ in this integral form, resulting in :

$$\frac{1}{t}\sum_{i=1}^{t}\int \rho_{(i)}(\mathbf{x})\mathcal{L}(h, f_{P_{(1:t)}})\mathrm{d}\mathbf{x} \tag{14}$$

We assume that $\mathcal{P}_i$ is independent from $\mathcal{P}_j$, where $i \neq j$, which is a reasonable assumption, since each task is associated with a different dataset. Therefore, the true labeling function $f_{\mathcal{P}_{(1:t)}}$ can be represented by $f_{\mathcal{P}_i}$ under the target distribution $\mathcal{P}_i$ of the i-th task. Then we rewrite Eq. (14) as the expectation form $\frac{1}{t}\sum_{i=1}^{t}\mathcal{R}_{\mathcal{P}_i}(h, f_{\mathcal{P}_i})$.

Based on the above results, Eq. (13) is rewritten as :

$$\begin{aligned}
\frac{1}{t}\sum_{i=1}^{t}\mathcal{R}_{\mathcal{P}_i}(h, f_{\mathcal{P}_i}) \leq{}& \mathcal{R}_{\mathbb{P}^{(t-1)}\otimes\tilde{\mathcal{P}}_t}\left(h, h^*_{\mathbb{P}^{(t-1)}\otimes\tilde{\mathcal{P}}_t}\right) \\
&+ disc^*_{\mathcal{L}}\left(\mathcal{P}_{(1:t)}, \mathbb{P}^{(t-1)}\otimes\tilde{\mathcal{P}}_t\right) + \varepsilon\left(\mathcal{P}_{(1:t)}, \mathbb{P}^{(t-1)}\otimes\tilde{\mathcal{P}}_t\right)
\end{aligned} \tag{15}$$

In the following, we provide the derivations for Err^a and Err^d from Eq. (10). We consider to take $\mathbb{P}^{t-2}\otimes\tilde{\mathcal{P}}_{t-1}$ and $\mathbb{P}^{t-1}$ as the target and source domains, respectively. This is reasonable choice since we allow the $\mathbb{P}^{t-1}$ (the generator distribution of the model) to approximate $\mathbb{P}^{t-2}\otimes\tilde{\mathcal{P}}_{t-1}$. We derive the bound as :

$$\mathcal{R}_{\mathbb{P}^{t-2}\otimes\tilde{\mathcal{P}}_{t-1}}\left(h, h^*_{\mathbb{P}^{t-2}\otimes\tilde{\mathcal{P}}_{t-1}}\right) \leq \mathcal{R}_{\mathbb{P}^{t-1}}(h, h^*_{\mathbb{P}^{t-1}}) + disc^*_{\mathcal{L}}\left(\mathbb{P}^{t-2}\otimes\tilde{\mathcal{P}}_{t-1}, \mathbb{P}^{t-1}\right) + \varepsilon\left(\mathbb{P}^{t-2}\otimes\tilde{\mathcal{P}}_{t-1}, \mathbb{P}^{t-1}\right) \tag{16}$$

We then consider to take $\mathbb{P}^{t-3}\otimes\tilde{\mathcal{P}}_{t-2}$ and $\mathbb{P}^{t-2}$ as the target and source domains, respectively, and we derive the bound as :

$$\mathcal{R}_{\mathbb{P}^{t-3}\otimes\tilde{\mathcal{P}}_{t-2}}\left(h, h^*_{\mathbb{P}^{t-3}\otimes\tilde{\mathcal{P}}_{t-2}}\right) \leq \mathcal{R}_{\mathbb{P}^{t-2}}(h, h^*_{\mathbb{P}^{t-2}}) + disc^*_{\mathcal{L}}\left(\mathbb{P}^{t-3}\otimes\tilde{\mathcal{P}}_{t-2}, \mathbb{P}^{t-2}\right) + \varepsilon\left(\mathbb{P}^{t-3}\otimes\tilde{\mathcal{P}}_{t-2}, \mathbb{P}^{t-2}\right) \tag{17}$$

According to the Inductive Inference, we have :

$$\cdots$$

$$\cdots \tag{18}$$

$$\mathcal{R}_{\mathbb{P}^1 \otimes \tilde{\mathcal{P}}_2}\left(h, h^*_{\mathbb{P}^1}\right) \le \mathcal{R}_{\mathbb{P}^2}\left(h, h^*_{\mathbb{P}^2}\right) + disc^\star_{\mathcal{L}}\left(\mathbb{P}^1 \otimes \tilde{\mathcal{P}}_2, \mathbb{P}^2\right) + \varepsilon\left(\mathbb{P}^1 \otimes \tilde{\mathcal{P}}_2, \mathbb{P}^2\right)$$

$$\mathcal{R}_{\tilde{\mathcal{P}}_1}\left(h, h^*_{\tilde{\mathcal{P}}_1}\right) \le \mathcal{R}_{\mathbb{P}^1}\left(h, h^*_{\mathbb{P}^1}\right) + disc^\star_{\mathcal{L}}\left(\tilde{\mathcal{P}}_1, \mathbb{P}^1\right) + \varepsilon\left(\tilde{\mathcal{P}}_1, \mathbb{P}^1\right) \tag{19}$$

We then sum up all the above derivations in the inequality from Eq. (15) to Eq. (19), resulting in :

$$\frac{1}{t}\sum_{i=1}^{t}\left\{\mathcal{R}_{\mathcal{P}_i}\left(h, f_{\mathcal{P}_i}\right)\right\} + \mathcal{R}_{\tilde{\mathcal{P}}_1}\left(h, h^*_{\tilde{\mathcal{P}}_1}\right) + \sum_{k=1}^{t-2}\left\{\mathcal{R}_{\mathbb{P}^{t-1-k}\otimes\tilde{\mathcal{P}}_{t-k}}\left(h, h^*_{\mathbb{P}^{t-1-k}\otimes\tilde{\mathcal{P}}_{t-k}}\right)\right\} \le \sum_{k=1}^{t-2}\left\{\mathcal{R}_{\mathbb{P}^{t-k}}\left(h, h^*_{\mathbb{P}^{t-k}}\right)\right\}$$

$$+ \mathcal{R}_{\mathbb{P}^1}\left(h, h^*_{\mathbb{P}^1}\right) + \mathcal{R}_{\mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t}\left(h, h^*_{\mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t}\right) + \sum_{k=1}^{t-2}\left\{disc^\star_{\mathcal{L}}\left(\mathbb{P}^{t-1-k}\otimes\tilde{\mathcal{P}}_{t-k}, \mathbb{P}^{t-k}\right) + \varepsilon\left(\mathbb{P}^{t-1-k}\otimes\tilde{\mathcal{P}}_{t-k}, \mathbb{P}^{t-k}\right)\right\}$$

$$+ disc^\star_{\mathcal{L}}\left(\mathcal{P}_{(1:t)}, \mathbb{P}^{(t-1)}\otimes\tilde{\mathcal{P}}_t\right) + \varepsilon\left(\mathcal{P}_{(1:t)}, \mathbb{P}^{(t-1)}\otimes\tilde{\mathcal{P}}_t\right) + disc^\star_{\mathcal{L}}\left(\tilde{\mathcal{P}}_1, \mathbb{P}^1\right) + \varepsilon\left(\tilde{\mathcal{P}}_1, \mathbb{P}^1\right) \tag{20}$$

Then we move the second and third term in the left hand side to the right hand side in Eq.(20), resulting in :

$$\frac{1}{t}\sum_{i=1}^{t}\left\{\mathcal{R}_{\mathcal{P}_i}\left(h, f_{\mathcal{P}_i}\right)\right\} \le \mathcal{R}_{\mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t}\left(h, h^*_{\mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t}\right) + \mathcal{R}_{\mathbb{P}^1}\left(h, h^*_{\mathbb{P}^1}\right) - \mathcal{R}_{\tilde{\mathcal{P}}_1}\left(h, h^*_{\tilde{\mathcal{P}}_1}\right)$$

$$+ \sum_{k=1}^{t-2}\left\{\mathcal{R}_{\mathbb{P}^{t-k}}\left(h, h^*_{\mathbb{P}^{t-k}}\right) - \mathcal{R}_{\mathbb{P}^{t-1-k}\otimes\tilde{\mathcal{P}}_{t-k}}\left(h, h^*_{\mathbb{P}^{t-1-k}\otimes\tilde{\mathcal{P}}_{t-k}}\right)\right\}$$

$$+ \sum_{k=1}^{t-2}\left(disc^\star_{\mathcal{L}}\left(\mathbb{P}^{t-1-k}\otimes\tilde{\mathcal{P}}_{t-k}, \mathbb{P}^{t-k}\right)\right. \tag{21}$$

$$\left. + \varepsilon\left(\mathbb{P}^{t-1-k}\otimes\tilde{\mathcal{P}}_{t-k}, \mathbb{P}^{t-k}\right)\right) + disc^\star_{\mathcal{L}}\left(\mathcal{P}_{(1:t)}, \mathbb{P}^{(t-1)}\otimes\tilde{\mathcal{P}}_t\right)$$

$$+ \varepsilon\left(\mathcal{P}_{(1:t)}, \mathbb{P}^{(t-1)}\otimes\tilde{\mathcal{P}}_t\right) + disc^\star_{\mathcal{L}}\left(\tilde{\mathcal{P}}_1, \mathbb{P}^1\right) + \varepsilon\left(\tilde{\mathcal{P}}_1, \mathbb{P}^1\right)$$

Then we can rewrite Eq. (21) as :

$$\frac{1}{t}\sum_{i=1}^{t}\left\{\mathcal{R}_{\mathcal{P}_i}\left(h, f_{\mathcal{P}_i}\right)\right\} \le \mathcal{R}_{\mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t}\left(h, h^*_{\mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t}\right) + \mathcal{R}_{\mathbb{P}^1}\left(h, h^*_{\mathbb{P}^1}\right) - \mathcal{R}_{\tilde{\mathcal{P}}_1}\left(h, h^*_{\tilde{\mathcal{P}}_1}\right)$$

$$+ \underbrace{\sum_{k=1}^{t-1}\left\{\mathcal{R}_{\mathbb{P}^{t-k}}\left(h, h^*_{\mathbb{P}^{t-k}}\right) - \mathcal{R}_{\mathbb{P}^{t-1-k}\otimes\tilde{\mathcal{P}}_{t-k}}\left(h, h^*_{\mathbb{P}^{t-1-k}\otimes\tilde{\mathcal{P}}_{t-k}}\right)\right\}}_{\text{Err}^{\text{a}}} \tag{22}$$

$$+ \underbrace{\sum_{k=1}^{t-2}\left\{\mathcal{R}_A\left(\mathbb{P}^{t-1-k}\otimes\tilde{\mathcal{P}}_{t-k}, \mathbb{P}^{t-k}\right)\right\} + \mathcal{R}_A\left(\mathcal{P}_{(1:t)}, \mathbb{P}^{(t-1)}\otimes\tilde{\mathcal{P}}_t\right) + \mathcal{R}_A\left(\tilde{\mathcal{P}}_1, \mathbb{P}^1\right)}_{\text{Err}^{\text{d}}}$$

where $\mathcal{R}_{\mathbb{P}^0\otimes\tilde{\mathcal{P}}_1}\left(h, h^*_{\mathbb{P}^0\otimes\tilde{\mathcal{P}}_1}\right) = \mathcal{R}_{\tilde{\mathcal{P}}_1}\left(h, h^*_{\tilde{\mathcal{P}}_1}\right)$ and this proves Theorem 2.

C The proof of Lemma 1

According to the bound on the KL divergence :

$$\frac{1}{t}\sum_{i=1}^{t}\mathbb{E}_{\mathcal{P}_i}KL(q_{\omega^t}(\mathbf{z}\mid\mathbf{x}_i^T)\mid\mid p(\mathbf{z})) \leq \mathbb{E}_{\mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t}KL(q_{\omega^t}(\mathbf{z}\mid\tilde{\mathbf{x}}^t)\mid\mid p(\mathbf{z}))$$
$$+\left|\mathbb{E}_{\mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t}KL(q_{\omega^t}(\mathbf{z}\mid\tilde{\mathbf{x}}^t)\mid\mid p(\mathbf{z})) - \frac{1}{t}\sum_{i=1}^{t}\mathbb{E}_{\mathcal{P}_i}KL(q_{\omega^t}(\mathbf{z}\mid\mathbf{x}_i^T)\mid\mid p(\mathbf{z}))\right| \tag{23}$$

We also know that $\mathcal{L}_{ELBO}(\mathbf{x};\{\theta,\omega\})$ is expressed as :

$$\mathcal{L}_{ELBO}(\mathbf{x};\{\theta,\omega\}) := \mathbb{E}_{q_\omega(\mathbf{z}\mid\mathbf{x})}[\log p_\theta(\mathbf{x}\mid\mathbf{z})] - KL[q_\omega(\mathbf{z}\mid\mathbf{x})\mid\mid p(\mathbf{z})], \tag{24}$$

When the decoder models a Gaussian distribution, $\log p_\theta(\mathbf{x}\mid\mathbf{z})$ can be represented as :

$$\log p_\theta(\mathbf{x}\mid\mathbf{z}) = -\frac{1}{2\sigma_\theta^2(\mathbf{z})}\|\mathbf{x}-\mu_\theta(\mathbf{z})\|^2 - \frac{1}{2}\log 2\pi\sigma_\theta^2(\mathbf{z}) \tag{25}$$

where $\sigma_\theta(\mathbf{z})$ and $\mu_\theta(\mathbf{z})$ are the variance and mean of Gaussian distribution, obtained by the decoder. $\|\cdot\|^2$ represents the reconstruction error (square loss). We implement the decoder by a Gaussian distribution with the fixed variance $\mathcal{N}(\mu_\theta(\mathbf{z}),\sigma\mathbf{I})$ where $\mu_\theta(\mathbf{z})$ is a deep convolutional neural network and $\mathbf{I}$ is the identity matrix. Therefore, Eq. (25) is represented by the fixed variance σ :

$$\log p_\theta(\mathbf{x}\mid\mathbf{z}) = -\frac{1}{2\sigma^2}\|\mathbf{x}-\mu_\theta(\mathbf{z})\|^2 - \frac{1}{2}\log 2\pi\sigma^2 \tag{26}$$

Since h is the hypothesis of the model $(\mathcal{M}^t)$, implemented as a encoding-decoding process, we have

$$\mathcal{L}_{ELBO}(\mathbf{x}_i^T;h) = -\frac{1}{2\sigma^2}\mathcal{L}(h(\mathbf{x}_i^T),f_{\mathcal{P}_i}(\mathbf{x}_i^T)) - \frac{1}{2}\log 2\pi\sigma^2 - KL(q_{\omega^t}(\mathbf{z}\mid\mathbf{x}_i^T)\mid\mid p(\mathbf{z}))$$

.

Then we focus on the negative ELBO

$$-\mathcal{L}_{ELBO}(\mathbf{x}_i^T;h) = \frac{1}{2\sigma^2}\mathcal{L}(h(\mathbf{x}_i^T),f_{\mathcal{P}_i}(\mathbf{x}_i^T)) + \frac{1}{2}\log 2\pi\sigma^2 + KL(q_{\omega^t}(\mathbf{z}\mid\mathbf{x}_i^T)\mid\mid p(\mathbf{z}))$$

.

And we know that $\mathcal{R}_{\mathcal{P}_i}(h,f_{\mathcal{P}_i}) = \mathbb{E}_{\mathbf{x}\sim\mathcal{P}_i}\mathcal{L}(h(\mathbf{x}_i^T),f_{\mathcal{P}_i}(\mathbf{x}_i^T))$ and we have :

$$\mathbb{E}_{\mathbf{x}\sim\mathcal{P}_i}\left[-\mathcal{L}_{ELBO}(\mathbf{x}_i^T;h)\right] = \mathbb{E}_{\mathbf{x}_i^T\sim\mathcal{P}_i}\left\{\frac{1}{2\sigma^2}\mathcal{L}(h(\mathbf{x}_i^T),f_{\mathcal{P}_i}(\mathbf{x}_i^T)) + KL(q_{\omega^t}(\mathbf{z}\mid\mathbf{x}_i^T)\mid\mid p(\mathbf{z}))\right\} + \frac{1}{2}\log 2\pi\sigma^2 \tag{27}$$

We observe that $\frac{1}{2}\log 2\pi\sigma^2$ and $\frac{1}{2\sigma^2}$ are constants. In order to simplify the notations, we set $\sigma = \frac{1}{\sqrt{2}}$. Therefore, Eq. (27) is rewritten as :

$$\mathbb{E}_{\mathbf{x}_i^T\sim\mathcal{P}_i}\left[-\mathcal{L}_{ELBO}(\mathbf{x}_i^T;h)\right] = \mathbb{E}_{\mathbf{x}_i^T\sim\mathcal{P}_i}\{\mathcal{L}(h(\mathbf{x}_i^T),f_{\mathcal{P}_i}(\mathbf{x}_i^T)) + KL(q_{\omega^t}(\mathbf{z}\mid\mathbf{x}_i^T)\mid\mid p(\mathbf{z}))\} + \frac{1}{2}\log\pi \tag{28}$$

Eq. (28) can be seen as the average ELBO for all samples. We then take Eq. (23) in Eq.(7) in the paper, we have :

$$\frac{1}{t}\sum_{i=1}^{t}\left\{\mathbb{E}_{\mathbf{x}_i^T\sim\mathcal{P}_i}\{\mathcal{L}(h(\mathbf{x}_i^T),f_{\mathcal{P}_i}(\mathbf{x}_i^T)) + KL(q_{\omega^t}(\mathbf{z}|\mathbf{x}_i^T)\mid\mid p(\mathbf{z}))\}\right\} \leq \mathbb{E}_{\mathbf{x}^t\sim\mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t}\left\{\mathcal{L}\left(h(\tilde{\mathbf{x}}^t),h^*_{\mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t}(\tilde{\mathbf{x}}^t)\right)\right.$$
$$+ KL(q_{\omega^t}(\mathbf{z}\mid\tilde{\mathbf{x}}^t)\mid\mid p(\mathbf{z}))\Big\}$$
$$+ |KL_1 - KL_2| + \mathcal{R}_A\left(\mathcal{P}_{(1:t)},\mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t\right) \tag{29}$$

It notes that we can add the constant $\frac{1}{2}\log\pi$ in both sides of Eq. (29). According to Eq. (28), we can rewrite Eq. (29)

as :

$$\frac{1}{t}\sum_{i=1}^{t}\mathbb{E}_{\mathbf{x}_i^T \sim \mathcal{P}_i}[-\mathcal{L}_{ELBO}\left(\mathbf{x}_i^T; h\right)] \leq \mathbb{E}_{\mathbf{x}^t \sim \mathbb{P}^{t-1} \otimes \tilde{\mathcal{P}}_t}\left[-\mathcal{L}_{ELBO}\left(\tilde{\mathbf{x}}^t; h\right)\right] + |KL_1 - KL_2| + \mathcal{R}_A\left(\mathcal{P}_{(1:t)}, \mathbb{P}^{t-1} \otimes \tilde{\mathcal{P}}_t\right).$$

(30)

This proves Lemma 1. We can further replace the latest term in the right hand side of Eq. (30) by $\mathrm{Err}^a + \mathrm{Err}^d$ (See details in the proof of Theorem 2), resulting in :

$$\frac{1}{t}\sum_{i=1}^{t}\mathbb{E}_{\mathbf{x}_i^T \sim \mathcal{P}_i}[-\mathcal{L}_{ELBO}\left(\mathbf{x}_i^T; h\right)] \leq \mathbb{E}_{\mathbf{x}^t \sim \mathbb{P}^{t-1} \otimes \tilde{\mathcal{P}}_t}\left[-\mathcal{L}_{ELBO}\left(\tilde{\mathbf{x}}^t; h\right)\right] + |KL_1 - KL_2| + \mathrm{Err}^a + \mathrm{Err}^d.$$

(31)

D   The proof of Theorem 3

Firstly, $\mathcal{R}_C$ can be easily proved since the task was trained only once and we simply derive the bound between the training sets and testing sets.

For the components that are trained more than once, we firstly consider the c_i'-th component and $a(i,j)$-th task. This can be easily generalized to other components and other tasks. We firstly consider to take $\mathcal{P}_{a(i,j)}$ and $\mathbb{P}_{a(i,j)}^0$ as the target and source distribution and we have a bound :

$$\mathcal{R}_{\mathcal{P}_{a(i,j)}}\left(h_{c_i'}, f_{\mathcal{P}_{a(i,j)}}\right) \leq \mathcal{R}_{\mathbb{P}_{a(i,j)}^0}\left(h_{c_i'}, h_{\mathbb{P}_{a(i,j)}^0}^*\right) + disc_{\mathcal{L}}^{\star}\left(\mathcal{P}_{a(i,j)}, \mathbb{P}_{a(i,j)}^0\right) + \varepsilon\left(\mathcal{P}_{a(i,j)}, \mathbb{P}_{a(i,j)}^0\right)$$

(32)

We can observe that $\mathbb{P}_{a(i,1)}^0$ represent the training set $\tilde{\mathcal{P}}_{a(i,j)}$. Then we consider to take $\mathbb{P}_{a(i,j)}^0$ and $\mathbb{P}_{a(i,j)}^1$ as the target and source domains, respectively. Then we have the bound as :

$$\mathcal{R}_{\mathbb{P}_{a(i,j)}^0}\left(h_{c_i'}, f_{\mathbb{P}_{a(i,j)}^0}\right) \leq \mathcal{R}_{\mathbb{P}_{a(i,j)}^1}\left(h_{c_i'}, h_{\mathbb{P}_{a(i,j)}^1}^*\right) + disc_{\mathcal{L}}^{\star}\left(\mathbb{P}_{a(i,j)}^0, \mathbb{P}_{a(i,j)}^1\right) + \varepsilon\left(\mathbb{P}_{a(i,j)}^0, \mathbb{P}_{a(i,j)}^1\right)$$

(33)

Similarly, we have the following bounds :

$$\mathcal{R}_{\mathbb{P}_{a(i,j)}^1}\left(h_{c_i'}, h_{\mathbb{P}_{a(i,j)}^1}^*\right) \leq \mathcal{R}_{\mathbb{P}_{a(i,j)}^2}\left(h_{c_i'}, h_{\mathbb{P}_{a(i,j)}^2}^*\right) + disc_{\mathcal{L}}^{\star}\left(\mathbb{P}_{a(i,j)}^1, \mathbb{P}_{a(i,j)}^2\right) + \varepsilon\left(\mathbb{P}_{a(i,j)}^1, \mathbb{P}_{a(i,j)}^2\right)$$

(34)

$$\dots$$

$$\mathcal{R}_{\mathbb{P}_{a(i,j)}^{c(i,j)-2}}\left(h_{c_i'}, f_{\mathbb{P}_{a(i,j)}^{c(i,j)-2}}\right) \leq \mathcal{R}_{\mathbb{P}_{a(i,j)}^{c(i,j)-1}}\left(h_{c_i'}, h_{\mathbb{P}_{a(i,j)}^{c(i,j)-1}}^*\right) + disc_{\mathcal{L}}^{\star}\left(\mathbb{P}_{a(i,j)}^{c(i,j)-2}, \mathbb{P}_{a(i,j)}^{c(i,j)-1}\right) + \varepsilon\left(\mathbb{P}_{a(i,j)}^{c(i,j)-2}, \mathbb{P}_{a(i,j)}^{c(i,j)-1}\right)$$

(35)

$$\mathcal{R}_{\mathbb{P}_{a(i,j)}^{c(i,j)-1}}\left(h_{c_i'}, f_{\mathbb{P}_{a(i,j)}^{c(i,j)-1}}\right) \leq \mathcal{R}_{\mathbb{P}_{a(i,j)}^{c(i,j)}}\left(h_{c_i'}, h_{\mathbb{P}_{a(i,j)}^{c(i,j)}}^*\right) + disc_{\mathcal{L}}^{\star}\left(\mathbb{P}_{a(i,j)}^{c(i,j)-1}, \mathbb{P}_{a(i,j)}^{c(i,j)}\right) + \varepsilon\left(\mathbb{P}_{a(i,j)}^{c(i,j)-1}, \mathbb{P}_{a(i,j)}^{c(i,j)}\right)$$

(36)

Then we sum up all the above relationships, resulting in :

$$\begin{aligned}
\mathcal{R}_{\mathcal{P}_{a(i,j)}}\left(h_{c_i'}, f_{\mathcal{P}_{a(i,j)}}\right) &\leq \mathcal{R}_{\mathbb{P}_{a(i,j)}^{c(i,j)}}\left(h_{c_i'}, h_{\mathbb{P}_{a(i,j)}^{c(i,j)}}^*\right) \\
&+ \sum_{k=-1}^{c(i,j)-1}\left\{disc_{\mathcal{L}}^{\star}\left(\mathbb{P}_{a(i,j)}^k, \mathbb{P}_{a(i,j)}^{k+1}\right) + \varepsilon\left(\mathbb{P}_{a(i,j)}^k, \mathbb{P}_{a(i,j)}^{k+1}\right)\right\}8i
\end{aligned}$$

(37)

where we also use $\mathbb{P}_{a(i,j)}^{-1}$ represent $\mathcal{P}_{a(i,j)}$. RHS of Eq. (37) is an upper bound to the target risk for a single task $\mathcal{P}_{a(i,j)}$ modelled by using the c_i'-th component. In the following, we consider all components C' that are trained more than once :

$$\begin{aligned}
\sum_{i=1}^{|C'|}\sum_{j=1}^{\tilde{a}_i}\mathcal{R}_{\mathcal{P}_{a(i,j)}}\left(h_{c_i'}, f_{\mathcal{P}_{a(i,j)}}\right) &\leq \sum_{i=1}^{|C'|}\sum_{j=1}^{\tilde{a}_i}\left\{\mathcal{R}_{\mathbb{P}_{a(i,j)}^{c(i,j)}}\left(h_{c_i'}, h_{\mathbb{P}_{a(i,j)}^{c(i,j)}}^*\right)\right. \\
&\left. + \sum_{k=-1}^{c(i,j)-1}\left(disc_{\mathcal{L}}^{\star}\left(\mathbb{P}_{a(i,j)}^k, \mathbb{P}_{a(i,j)}^{k+1}\right) + \varepsilon\left(\mathbb{P}_{a(i,j)}^k, \mathbb{P}_{a(i,j)}^{k+1}\right)\right)\right\}
\end{aligned}$$

(38)

6

RHS of Eq. (38) is still an upper bound to the target risk of tasks modelled by the components that trained more than once. Therefore, $\mathcal{R}_{A'}$ in the paper, can be expressed by RHS of Eq. (38), which proves Theorem 3.

Based on the results from Theorem 3, we provide the additional analysis of the results of Theorem 3 in the following. Firstly, we rewrite Eq.(11) of the paper by :

$$
\frac{1}{t}\sum_{i=1}^{|C|}\left\{\mathcal{R}_{\mathcal{P}_{a_i}}\left(h_{c_i}, f_{\mathcal{P}_{a_i}}\right)\right\} + \frac{1}{t}\sum_{i=1}^{|C'|}\sum_{j=1}^{\tilde{a}_i}\left\{\mathcal{R}_{\mathcal{P}_{a(i,j)}}\left(h_{c_i'}, f_{\mathcal{P}_{a(i,j)}}\right)\right\} \le \sum_{i=1}^{|C|}\left\{\mathcal{R}_{\tilde{\mathcal{P}}_{a_i}}\left(h_{c_i}, h^*_{\tilde{\mathcal{P}}_{a_i}}\right) + \mathcal{R}_A\left(\mathcal{P}_{a_i}, \tilde{\mathcal{P}}_{a_i}\right)\right\}
$$

$$
+ \sum_{i=1}^{|C'|}\sum_{j=1}^{\tilde{a}_i}\left\{\mathcal{R}_{\mathbb{P}^{c(i,j)}_{a(i,j)}}\left(h_{c_i'}, h^*_{\mathbb{P}^{c(i,j)}_{a(i,j)}}\right) + \sum_{k=-1}^{c(i,j)-1}\left(\mathcal{R}_A\left(\mathbb{P}^k_{a(i,j)}, \mathbb{P}^{k+1}_{a(i,j)}\right)\right)\right\}.
$$

$$(39)$$

We consider an extreme case where $\mathbf{M}$ only has a single component after LLL, ($|C'| = 1$ and $|C| = 0$). Then the first term in RHS of Eq. (39) disappears and resulting in :

$$
\frac{1}{t}\sum_{i=1}^{|C'|}\sum_{j=1}^{\tilde{a}_i}\mathcal{R}_{\mathcal{P}_{a(i,j)}}\left(h_{c_i'}, f_{\mathcal{P}_{a(i,j)}}\right) \le \sum_{i=1}^{|C'|}\sum_{j=1}^{\tilde{a}_i}\left\{\mathcal{R}_{\mathbb{P}^{c(i,j)}_{a(i,j)}}\left(h_{c_i'}, h^*_{\mathbb{P}^{c(i,j)}_{a(i,j)}}\right) + \sum_{k=-1}^{c(i,j)-1}\left\{\mathcal{R}_A\left(\mathbb{P}^k_{a(i,j)}, \mathbb{P}^{k+1}_{a(i,j)}\right)\right\}\right\}.
$$

$$(40)$$

In this case, learning early tasks ($a(i,j)$ is small) tends to increase the number of error terms more than when learning more recent tasks ($a(i,j)$ is large), This is caused by the number of accumulated error terms $\mathcal{R}_A(\cdot)$ controlled by the times of GR processes $c(i,j) = t - a(i,j)$. In the opposite case where the number of components (K) is equal to the number of tasks (t), $|C'| = 0$ and the mixture model has not accumulated errors. The GB for this case is :

$$
\frac{1}{t}\sum_{i=1}^{|C|}\mathcal{R}_{\mathcal{P}_{a_i}}\left(h_{c_i}, f_{\mathcal{P}_{a_i}}\right) \le \sum_{i=1}^{|C|}\left\{\mathcal{R}_{\tilde{\mathcal{P}}_{a_i}}\left(h_{c_i}, h^*_{\tilde{\mathcal{P}}_{a_i}}\right) + \mathcal{R}_A\left(\mathcal{P}_{a_i}, \tilde{\mathcal{P}}_{a_i}\right)\right\}.
$$

$$(41)$$

where $|C| = K = t$. Then the lifelong learning problem is transformed to be the generalization problem under the generative modelling. This motivates us to propose a novel dynamic mixture model which would not accumulate errors during LLL.

E The proof of Lemma 2

Similarly to the proof for Theorem 3, we firstly consider the components that are trained only once :

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

$$(42)$$

Then we add the KL divergence term and D_{diff} term in Eq. (42), resulting in :

$$
\mathcal{R}_C = \sum_{i=1}^{|C|}\left\{\mathcal{R}_{\tilde{\mathcal{P}}_{a_i}}\left(h_{c_i}, h^*_{\tilde{\mathcal{P}}_{a_i}}\right) + \mathbb{E}_{\tilde{\mathcal{P}}_{a_i}}KL\left(p_{c_i}(\mathbf{z}\mid\mathbf{x}^S_{a_i})\,\|\,p(\mathbf{z})\right) + D_{diff}\left(\mathbf{x}^T_{a_i}, \mathbf{x}^S_{a_i}\right) + \mathcal{R}_A\left(\mathcal{P}_{a_i}, \tilde{\mathcal{P}}_{a_i}\right)\right\},
$$

$$(43)$$

where $p_{c_i}(\mathbf{z}\mid\cdot)$ represents the variational distribution modelled by the inference model of the c_i-th component. $D_{diff}(\cdot,\cdot)$ is defined as :

$$
D_{diff}(\mathbf{x}^T_{a_i}, \mathbf{x}^S_{a_i}) = \left|\mathbb{E}_{\mathcal{P}_{a_i}}KL(q_{\omega_{c_i}}(\mathbf{z}\mid\mathbf{x}^T_i)\,\|\,p(\mathbf{z})) - \mathbb{E}_{\tilde{\mathcal{P}}_{a_i}}KL(q_{\omega_{c_i}}(\mathbf{z}\mid\mathbf{x}^S_i)\,\|\,p(\mathbf{z}))\right|
$$

$$(44)$$

We can rewrite Eq. (43) as the negative ELBO form :

$$
\mathcal{R}_C = \sum_{i=1}^{|C|}\left\{\mathbb{E}_{\tilde{\mathcal{P}}_i}\{-\mathcal{L}_{ELBO}\left(\mathbf{x}^S_{a_i}; h_{c_i}\right)\} + D_{diff}\left(\mathbf{x}^T_{a_i}, \mathbf{x}^S_{a_i}\right) + \mathcal{R}_A\left(\mathcal{P}_{a_i}, \tilde{\mathcal{P}}_{a_i}\right)\right\},
$$

$$(45)$$

where $\mathcal{L}_{ELBO}(\cdot; h_{c_i})$ represents the ELBO estimated by the c_i-th component. Secondly, we consider the components that are trained more than once :

$$
\mathcal{R}_{A'} = \sum_{i=1}^{|C'|}\sum_{j=1}^{\tilde{a}_i}\left\{\mathcal{R}_{\mathbb{P}^{c(i,j)}_{a(i,j)}}\left(h_{c_i'}, f_{\mathbb{P}^{c(i,j)}_{a(i,j)}}\right) + \mathcal{R}_A\left(\mathcal{P}_{a(i,j)}, \mathbb{P}^{c(i,j)}_{a(i,j)}\right)\right\},
$$

$$(46)$$

We then add the KL divergence term and D_{diff} term in eq.(46), resulting in :

$$\mathcal{R}_{A'} = \sum_{i=1}^{|C'|} \sum_{j=1}^{\tilde{a}_i} \left\{ \mathcal{R}_{\mathbb{P}_{a(i,j)}^{c(i,j)}} \left(h_{c_i'}, f_{\mathbb{P}_{a(i,j)}^{c(i,j)}} \right) + \mathbb{E}_{\mathbb{P}_{a(i,j)}^{c(i,j)}} KL \left(p_{c_i'}(\mathbf{z} \mid \mathbf{x}_{a(i,j)}^t) \parallel p(\mathbf{z}) \right) \right.$$
$$\left. + D_{diff} \left(\mathbf{x}_{a(i,j)}^T, \mathbf{x}_{a(i,j)}^t \right) + \mathcal{R}_A \left(\mathcal{P}_{a(i,j)}, \mathbb{P}_{a(i,j)}^{c(i,j)} \right) \right\}, \tag{47}$$

where the conditional distribution $p_{c_i'}(\mathbf{z}|\mathbf{x}_{a(i,j)}^t)$ is modelled by the inference model of the c_i'-th component. Then we rewrite Eq. (47) as the negative ELBO form :

$$\mathcal{R}_{A'} = \sum_{i=1}^{|C'|} \sum_{j=1}^{\tilde{a}_i} \mathbb{E}_{\mathbb{P}_{a(i,j)}^{c(i,j)}} \left\{ -\mathcal{L}_{ELBO} \left(\mathbf{x}_{a(i,j)}^t; h_{c_i'} \right) + D_{diff} \left(\mathbf{x}_{a(i,j)}^T, \mathbf{x}_{a(i,j)}^t \right) + \mathcal{R}_A \left(\mathcal{P}_{a(i,j)}, \mathbb{P}_{a(i,j)}^{c(i,j)} \right) \right\}, \tag{48}$$

We summarize all KL divergence and D_{diff} terms :

$$\sum_{i=1}^{|C|} \left\{ \mathbb{E}_{\tilde{\mathcal{P}}_{a_i}} KL \left(p_{c_i}(\mathbf{z} \mid \mathbf{x}_{a_i}^S) \parallel p(\mathbf{z}) \right) + D_{diff} \left(\mathbf{x}_{a_i}^T, \mathbf{x}_{a_i}^S \right) \right\}$$
$$+ \sum_{i=1}^{|C'|} \sum_{j=1}^{\tilde{a}_i} \left\{ \mathbb{E}_{\mathbb{P}_{a(i,j)}^{c(i,j)}} KL \left(p_{c_i'}(\mathbf{z} \mid \mathbf{x}_{a(i,j)}^t) \parallel p(\mathbf{z}) \right) + D_{diff} \left(\mathbf{x}_{a(i,j)}^T, \mathbf{x}_{a(i,j)}^t \right) \right\}, \tag{49}$$

We also know that :

$$\frac{1}{t} \sum_{i=1}^{t} \mathbb{E}_{\mathcal{P}_i} KL \left(p(\mathbf{z} \mid \mathbf{x}_i^T) \parallel p(\mathbf{z}) \right) \leq \frac{1}{t} \sum_{i=1}^{|C|} \left\{ \mathbb{E}_{\tilde{\mathcal{P}}_{a_i}} KL \left(p(\mathbf{z} \mid \mathbf{x}_{a_i}^S) \parallel p(\mathbf{z}) \right) + D_{diff} \left(\mathbf{x}_{a_i}^T, \mathbf{x}_{a_i}^S \right) \right\}$$
$$+ \frac{1}{t} \sum_{i=1}^{|C'|} \sum_{j=1}^{\tilde{a}_i} \left\{ \mathbb{E}_{\mathbb{P}_{a(i,j)}^{c(i,j)}} KL \left(p(\mathbf{z} \mid \mathbf{x}_{a(i,j)}^t) \parallel p(\mathbf{z}) \right) + D_{diff} \left(\mathbf{x}_{a(i,j)}^T, \mathbf{x}_{a(i,j)}^t \right) \right\} \tag{50}$$

where we omit the subscript for $p(\mathbf{z} \mid \cdot)$ for simplicity. We then consider the inequality from Eq. (50) into Eq. (11) of the paper and we have :

$$\frac{1}{t} \sum_{i=1}^{|C|} \mathbb{E}_{\mathcal{P}_{a_i}} \left\{ -\mathcal{L}_{ELBO} \left(\mathbf{x}_{a_i}^T; h_{c_i} \right) \right\} + \frac{1}{t} \sum_{i=1}^{|C'|} \sum_{j=1}^{\tilde{a}_i} \mathbb{E}_{\mathcal{P}_{a(i,j)}} \left\{ -\mathcal{L}_{ELBO} \left(\mathbf{x}_{a(i,j)}^T; h_{c_i'} \right) \right\} \leq \frac{1}{t} \sum_{i=1}^{|C|} \left\{ \mathbb{E}_{\tilde{\mathcal{P}}_{a_i}} \left\{ -\mathcal{L}_{ELBO} \left(\mathbf{x}_{a_i}^S; h_{c_i} \right) \right\} \right\}$$
$$+ \frac{1}{t} \sum_{i=1}^{|C'|} \sum_{j=1}^{|A_{c_i'}'|} \left\{ \mathbb{E}_{\mathbb{P}_{a(i,j)}^{c(i,j)}} \left\{ -\mathcal{L}_{ELBO} \left(\mathbf{x}_{a(i,j)}^t; h_{c_i'} \right) \right\} \right\} + \frac{1}{t} \{ \mathcal{R}_{A'}^{II} + \mathcal{R}_C^{II} + D_{diff}* \} \tag{51}$$

This proves Lemma 2. We should also observe that $D_{diff}*$ is expressed by :

$$D_{diff}* = \sum_{i=1}^{|C|} \left\{ D_{diff} \left(\mathbf{x}_{a_i}^T, \mathbf{x}_{a_i}^S \right) \right\} + \sum_{i=1}^{|C'|} \sum_{j=1}^{\tilde{a}_i} \left\{ D_{diff} \left(\mathbf{x}_{a(i,j)}^T, \mathbf{x}_{a(i,j)}^t \right) \right\}, \tag{52}$$

Based on the above results, in the following, we derive the risk bound of the mixture model to NLL :

$$\frac{1}{t} \sum_{i=1}^{|C|} \mathbb{E}_{\mathcal{P}_{a_i}} \left\{ -\log p_{c_i} \left(\mathbf{x}_{a_i}^T \right) \right\} + \frac{1}{t} \sum_{i=1}^{|C'|} \sum_{j=1}^{\tilde{a}_i} \left\{ -\log p_{c_i'} \left(\mathbf{x}_{a(i,j)}^T \right) \right\} \leq \frac{1}{t} \sum_{i=1}^{|C|} \left\{ \mathbb{E}_{\tilde{\mathcal{P}}_{a_i}} \left\{ -\mathcal{L}_{ELBO} \left(\mathbf{x}_{a_i}^S; h_{c_i} \right) \right\} \right\}$$
$$+ \frac{1}{t} \sum_{i=1}^{|C'|} \sum_{j=1}^{|A_{c_i'}'|} \left\{ \mathbb{E}_{\mathbb{P}_{a(i,j)}^{c(i,j)}} \left\{ -\mathcal{L}_{ELBO} \left(\mathbf{x}_{a(i,j)}^t; h_{c_i'} \right) \right\} \right\} + \frac{1}{t} \left\{ \mathcal{R}_{A'}^{II} + \mathcal{R}_C^{II} + D_{diff}* \right\} \tag{53}$$

where $\log p_{c_i}(\cdot)$ represents the sample log-likelihood (model likelihood) under the c_i-th component.

F   Generalize existing ELBOs to LELBO

In this section, we generalize previous well know ELBOs to LELBO.

F.1   Importance sampling

The main idea of Importance Weighted Autoencoder (IWELBO) Burda et al. [2015] is to allow the recognition network to generate multiple samples during the optimization leading to a better modelling of the posterior probabilities. The corresponding ELBO for sampling K' samples is defined as :

$$\mathcal{L}_{ELBO_{K'}}\left(\mathbf{x};\mathcal{M}\right)=\mathbb{E}_{\mathbf{z}_1,..,\mathbf{z}_{K'}\sim q(\mathbf{z}|\mathbf{x})}\left[\log\frac{1}{K'}\sum_{i=1}^{K'}\frac{p\left(\mathbf{x},\mathbf{z}_i\right)}{q\left(\mathbf{z}_i|\mathbf{x}\right)}\right] \tag{54}$$

where K' is the number of weighted samples and $K'=1$ is equivalent to the standard ELBO. In order to calculate $w_i = p(\mathbf{x},\mathbf{z}_i)/q(\mathbf{z}_i|\mathbf{x})$ in practice, we rewrite w_i as $\exp(\log w_i)$. By calculating the right hand side of Eq. (54) requires to estimate each individual $\mathbb{E}_{z_i}\log w_i$ which is a standard ELBO.

In the following, we extend this IWELBO to the LLL setting. From Lemma 1, we know that :

$$\frac{1}{t}\sum_{i=1}^{t}\mathbb{E}_{\mathbf{x}_i^T\sim\mathcal{P}_i}\Big[-\mathcal{L}_{ELBO}\left(\mathbf{x}_i^T;h\right)\Big]\leq\mathbb{E}_{\mathbf{x}^t\sim\mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t}\Big[-\mathcal{L}_{ELBO}\left(\tilde{\mathbf{x}}^t;h\right)\Big]$$
$$+|KL_1-KL_2|+\mathrm{Err}^a+\mathrm{Err}^d. \tag{55}$$

where $\mathcal{L}_{ELBO}(\cdot)$ has the form according to Kingma and Welling [2013]. We can rewrite the above equation, by considering importance sampling, as :

$$\frac{1}{t}\sum_{i=1}^{t}\mathbb{E}_{\mathbf{x}_i^T\sim\mathcal{P}_i}\Big[-\log p\left(\mathbf{x}_i^T\right)\Big]\leq\mathbb{E}_{\mathbf{x}^t\sim\mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t}\Big[-\mathbb{E}_{\mathbf{z}\sim q(\mathbf{z}|\mathbf{x})}\Big[\log\frac{p\left(\tilde{\mathbf{x}}^t,\mathbf{z}\right)}{q\left(\mathbf{z}\mid\mathbf{x}\right)}\Big]\Big]$$
$$+|KL_1-KL_2|+\mathrm{Err}^a+\mathrm{Err}^d. \tag{56}$$

According to $-\log p(\mathbf{x})\leq-\mathcal{L}_{ELBO_{K'+1}}(\mathbf{x};\mathcal{M})\leq-\mathcal{L}_{ELBO_{K'}}(\mathbf{x};\mathcal{M})$ Burda et al. [2015], we have $-\mathcal{L}_{ELBO_{K'+1}}(\tilde{\mathbf{x}}^t;\mathcal{M})\leq-\mathcal{L}_{ELBO_{K'}}(\tilde{\mathbf{x}}^t;\mathcal{M})$, based on the assumation that $\mathbb{P}^{t-1}$ is fixed. We note that $\mathcal{M}$ represents the model and h is the hypothesis of $\mathcal{M}$. We assume that when the $h^\star$ is an optimal solution for $-\log p(\tilde{\mathbf{x}}^t)$ and we have $-\log p(\tilde{\mathbf{x}}^t)\leq-\mathcal{L}_{ELBO}(\tilde{\mathbf{x}}^t;h^\star)\leq-\mathcal{L}_{ELBO_{K'}}(\tilde{\mathbf{x}}^t;h)$. In the following, we rewrite Eq. (55) by using $h^\star$, resulting in :

$$\frac{1}{t}\sum_{i=1}^{t}\mathbb{E}_{\mathbf{x}_i^T\sim\mathcal{P}_i}\Big[-\mathcal{L}_{ELBO}\left(\mathbf{x}_i^T;h^\star\right)\Big]\leq\mathbb{E}_{\mathbf{x}^t\sim\mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t}\Big[-\mathcal{L}_{ELBO}\left(\tilde{\mathbf{x}}^t;h^\star\right)\Big]$$
$$+|KL_1-KL_2|+\mathrm{Err}^a+\mathrm{Err}^d. \tag{57}$$

We observe that $|KL_1-KL_2|$ of Eq. (57) is estimated by $h^\star$ ($\{\theta_\star,\omega_\star\}$ are the corresponding model parameters). We can replace the first term in RHS of Eq. (57) by using $\mathcal{L}_{ELBO_{K'}}(\tilde{\mathbf{x}}^t;h)$, resulting in :

$$\frac{1}{t}\sum_{i=1}^{t}\mathbb{E}_{\mathbf{x}_i^T\sim\mathcal{P}_i}\Big[-\mathcal{L}_{ELBO}\left(\mathbf{x}_i^T;h^\star\right)\Big]\leq\mathbb{E}_{\mathbf{x}^t\sim\mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t}\Big[-\mathcal{L}_{ELBO_{K'}}(\tilde{\mathbf{x}}^t;h)\Big]$$
$$+|KL_1-KL_2|+\mathrm{Err}^a+\mathrm{Err}^d. \tag{58}$$

LHS of Eq. (58) is an upper bound to $\frac{1}{t}\sum_{i=1}^{t}\mathbb{E}_{\mathbf{x}_i^T\sim\mathcal{P}_i}[-\log p_{\theta^\star}(\mathbf{x}_i^T)]$ and we rewrite Eq. (58) as :

$$\frac{1}{t}\sum_{i=1}^{t}\mathbb{E}_{\mathbf{x}_i^T\sim\mathcal{P}_i}\Big[-\log p_{\theta^\star}(\mathbf{x}_i^T)\Big]\leq\mathbb{E}_{\mathbf{x}^t\sim\mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t}\Big[-\mathcal{L}_{ELBO_{K'}}(\tilde{\mathbf{x}}^t;h)\Big]$$
$$+|KL_1-KL_2|+\mathrm{Err}^a+\mathrm{Err}^d. \tag{59}$$

We then decompose the first term in RHS of Eq. (59), and we have :

$$\frac{1}{t}\sum_{i=1}^{t}\mathbb{E}_{\mathbf{x}_i^T\sim\mathcal{P}_i}\Big[-\log p_{\theta^\star}\left(\mathbf{x}_i^T\right)\Big]\leq\mathbb{E}_{\mathbf{x}^t\sim\mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t}\Big[-\mathbb{E}_{\mathbf{z}_1,...,\mathbf{z}_{K'}\sim q(\mathbf{z}|\mathbf{x})}\Big[\log\frac{1}{K'}\sum_{i=1}^{K'}\frac{p_\theta\left(\tilde{\mathbf{x}}^t,\mathbf{z}_i\right)}{q_\omega\left(\mathbf{z}_i\mid\mathbf{x}\right)}\Big]\Big]$$
$$+|KL_1-KL_2|+\mathrm{Err}^a+\mathrm{Err}^d. \tag{60}$$

If we do not consider Err^a and Err^d as in Lemma 1 (See details in Appendix C), Eq. (60) can be :

$$\frac{1}{t}\sum_{i=1}^{t}\mathbb{E}_{\mathbf{x}_i^T\sim\mathcal{P}_i}\left[-\log p\left(\mathbf{x}_i^T\right)\right]\leq\mathbb{E}_{\tilde{\mathbf{x}}^t\sim\mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t}\left[-\mathbb{E}_{\mathbf{z}_1\ldots,\mathbf{z}_{K'}\sim q(\mathbf{z}|\mathbf{x})}\left[\log\frac{1}{K'}\sum_{i=1}^{K'}\frac{p\left(\tilde{\mathbf{x}}^t,\mathbf{z}_i\right)}{q\left(\mathbf{z}_i\mid\mathbf{x}\right)}\right]\right] \quad (61)$$

$$+\left|KL_1-KL_2\right|+\mathcal{R}_A\left(\mathcal{P}_{(1:t)},\mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t\right).$$

where we omit the subscript (the model's parameters) for Eq. (61) for the sake of simplification. It notes that $h\in\mathcal{H}$ is the model and its parameters are $\{\theta,\omega\}$ optimized by Eq.(60). We call RHS of Eq. (60) as $\mathcal{L}_{LELBO_{K'}}$, and when $K'=1$, $\mathcal{L}_{LELBO_{K'}}$ is equal to $\mathcal{L}_{LELBO}$ (RHS of Eq. (56)). Based on the assumption that $\mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t$ is fixed, we have $\mathcal{L}_{LELBO_{K'+1}}\leq\mathcal{L}_{LELBO_{K'}}$. We can observe that the tightness of ELBO on the marginal log-likelihood of the source distribution $\mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t$ can not guarantee a tight GB on the marginal log-likelihood of the target distribution. However, the tightness of GB is largely depending on the discrepancy distance between the evolved source and target distribution.

F.2 Hierarchical Variational Inference

In this section, we review the Hierarchical latent variable model and extend it to IELBO.

Auxiliary Deep Generative Models (ADGM) Maaløe et al. [2016] ADGM is a classical hierarchical latent variable model. ADGM introduces an auxiliary variable $\mathbf{a}$ into the variational distribution $q(\mathbf{a},\mathbf{z}\mid\mathbf{x})=q(\mathbf{z}\mid\mathbf{a},\mathbf{x})q(\mathbf{a}\mid\mathbf{x})$ and its ELBO is expressed as :

$$\log p\left(\mathbf{x}\right)=\log\iint p\left(\mathbf{x},\mathbf{a},\mathbf{z}\right)d\mathbf{a}d\mathbf{z}\geq\mathbb{E}_{q(\mathbf{a},\mathbf{z}|\mathbf{x})}\left[\log\frac{p\left(\mathbf{a}\mid\mathbf{z},\mathbf{x}\right)p\left(\mathbf{x}\mid\mathbf{z}\right)p\left(\mathbf{z}\right)}{q\left(\mathbf{a}\mid\mathbf{x}\right)q\left(\mathbf{z}\mid\mathbf{a},\mathbf{x}\right)}\right]=\mathcal{L}_{ADGM}\left(\mathcal{M},\mathbf{x}\right). \quad (62)$$

We decompose $\mathcal{L}_{ADGM}(\mathbf{x};\mathcal{M})$ as :

$$\mathcal{L}_{ADGM}\left(\mathcal{M},\mathbf{x}\right)=\mathbb{E}_{q(\mathbf{a},\mathbf{z}|\mathbf{x})}\left[\log p\left(\mathbf{x}\mid\mathbf{z}\right)\right]+\mathbb{E}_{q(\mathbf{a},\mathbf{z}|\mathbf{x})}\left[\log\frac{p\left(\mathbf{a}\mid\mathbf{z},\mathbf{x}\right)p\left(\mathbf{z}\right)}{q\left(\mathbf{a}\mid\mathbf{x}\right)q\left(\mathbf{z}\mid\mathbf{a},\mathbf{x}\right)}\right] \quad (63)$$

We consider the following inequality :

$$\frac{1}{t}\sum_{i=1}^{t}\mathbb{E}_{\mathbf{x}_i^T\sim\mathcal{P}_i}\mathbb{E}_{q\left(\mathbf{a},\mathbf{z}|\mathbf{x}_i^T\right)}\left[\log\frac{p\left(\mathbf{a}\mid\mathbf{z},\mathbf{x}_i^T\right)p\left(\mathbf{z}\right)}{q\left(\mathbf{a}\mid\mathbf{x}_i^T\right)q\left(\mathbf{z}\mid\mathbf{a},\mathbf{x}_i^T\right)}\right]\leq\mathbb{E}_{\tilde{\mathbf{x}}^t\sim\mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_i}\mathbb{E}_{q(\mathbf{a},\mathbf{z}|\mathbf{x}^t)}\left[\log\frac{p\left(\mathbf{a}\mid\mathbf{z},\tilde{\mathbf{x}}^t\right)p\left(\mathbf{z}\right)}{q\left(\mathbf{a}\mid\tilde{\mathbf{x}}^t\right)q\left(\mathbf{z}\mid\mathbf{a},\tilde{\mathbf{x}}^t\right)}\right]$$

$$+\left|\mathbb{E}_{\tilde{\mathbf{x}}^t\sim\mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_i}\mathbb{E}_{q(\mathbf{a},\mathbf{z}|\tilde{\mathbf{x}}^t)}\left[\log\frac{p\left(\mathbf{a}\mid\mathbf{z},\tilde{\mathbf{x}}^t\right)p\left(\mathbf{z}\right)}{q\left(\mathbf{a}\mid\tilde{\mathbf{x}}^t\right)q\left(\mathbf{z}\mid\mathbf{a},\tilde{\mathbf{x}}^t\right)}\right]-\frac{1}{t}\sum_{i=1}^{t}\mathbb{E}_{\mathbf{x}_i^T\sim\mathcal{P}_i}\mathbb{E}_{q\left(\mathbf{a},\mathbf{z}|\mathbf{x}_i^T\right)}\left[\log\frac{p\left(\mathbf{a}\mid\mathbf{z},\mathbf{x}_i^T\right)p\left(\mathbf{z}\right)}{q\left(\mathbf{a}\mid\mathbf{x}_i^T\right)q\left(\mathbf{z}\mid\mathbf{a},\mathbf{x}_i^T\right)}\right]\right|. \quad (64)$$

We name the second term of RHS of Eq. (64) as $|\mathbf{E}_1^*-\mathbf{E}_2^*|$. By considering Lemma 1, we have :

$$(1/t)\sum_{i=1}^{t}\mathbb{E}_{\mathbf{x}_i^T\sim\mathcal{P}_i}\left[-\mathcal{L}_{ADGM}\left(\mathbf{x}_i^T\right)\right]\leq\mathbb{E}_{\tilde{\mathbf{x}}^t\sim\mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t}\left[-\mathbb{E}_{q(\mathbf{a},\mathbf{z}|\tilde{\mathbf{x}}^t)}\left[\log\frac{p\left(\mathbf{a}\mid\mathbf{z},\tilde{\mathbf{x}}^t\right)p\left(\tilde{\mathbf{x}}^t\mid\mathbf{z}\right)p\left(\mathbf{z}\right)}{q\left(\mathbf{a}\mid\tilde{\mathbf{x}}^t\right)q\left(\mathbf{z}\mid\mathbf{a},\tilde{\mathbf{x}}^t\right)}\right]\right]$$

$$+\left|\mathbf{E}_1^*-\mathbf{E}_2^*\right|+\text{Err}^a+\text{Err}^d. \quad (65)$$

Importance Weighted Hierarchical Variational Inference (IWHVI) Sobolev and Vetrov [2019]. IWHVI introduces a multisample generalization for the hierarchical variable model by using two auxiliary distributions, $q(\varsigma\mid\mathbf{z})$, $\tau(\psi\mid\mathbf{z},\mathbf{x})$ and its ELBO is expressed as :

$$\log p\left(\mathbf{x}\right)\geq\mathbb{E}_{q(\mathbf{z}|\mathbf{x})}\log\frac{p\left(\mathbf{x},\mathbf{z}\right)}{q\left(\mathbf{z}\mid\mathbf{x}\right)}\geq\mathbb{E}_{q(\mathbf{z},\psi_0|\mathbf{x})}\mathbb{E}_{\tau(\psi_{1:K}|\mathbf{z},\mathbf{x})}\mathbb{E}_{q(\zeta_{1:L}|\mathbf{z})}\log\frac{p\left(\mathbf{x}\mid\mathbf{z}\right)\frac{1}{L}\sum_{k=1}^{L}\frac{p(\mathbf{z},\zeta_k)}{p(\zeta_k,\mathbf{z})}}{\frac{1}{K+1}\sum_{k=0}^{K}\frac{q(\mathbf{z},\psi_k|\mathbf{x})}{\tau(\psi_k|\mathbf{z},\mathbf{x})}}=\mathcal{L}_{IWHVI}\left(\mathcal{M},\mathbf{x}\right). \quad (66)$$

From Eq. (66), we have $-\log p(\mathbf{x})\leq-\mathcal{L}_{\mathbf{ELBO}}(\mathbf{x};\mathcal{M})\leq-\mathcal{L}_{\mathbf{IWHVI}}(\mathbf{x};\mathcal{M})$. Therefore, according to Lemma 1, we generalize Eq. (66) to LLL setting :

$$\frac{1}{t}\sum_{i=1}^{t}\mathbb{E}_{\mathbf{x}_i^T\sim\mathcal{P}_i}\left\{-\log p\left(\mathbf{x}_i^T\right)\right\}\leq\mathbb{E}_{\tilde{\mathbf{x}}^t\sim\mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t}\left\{-\mathcal{L}_{IWHVI}\left(\tilde{\mathbf{x}}^t;h\right)\right\}$$

$$+\left|KL_1-KL_2\right|+\text{Err}^a+\text{Err}^d. \quad (67)$$

Additionally, Eq (67) can be extended to the IWELBO bound :

$$(1/t) \sum_{i=1}^{t} \mathbb{E}_{\mathbf{x}_i^T \sim \mathcal{P}_i} \{-\log p\left(\mathbf{x}_i^T\right)\} \le \mathbb{E}_{\mathbf{x}^t \sim \mathbb{P}^{t-1} \otimes \tilde{\mathcal{P}}_t} \left\{ -\mathbb{E} \log \frac{1}{K'} \sum_{i=1}^{K'} \frac{p\left(\mathbf{x}|\mathbf{z}_i\right) \frac{1}{L} \sum_{j=1}^{L} \frac{p\left(\mathbf{z}_i, \zeta_{(i,j)}\right)}{q\left(\zeta_{(i,j)}|\mathbf{z}_i\right)}}{\frac{1}{K+1} \sum_{k=0}^{K} \frac{q\left(\mathbf{z}_i, \psi_{(i,k)}|\mathbf{x}\right)}{\tau\left(\psi_{(i,k)}|\mathbf{x}, \mathbf{z}_i\right)}} \right\} \qquad (68)$$

$$+ |KL_1 - KL_2| + \mathrm{Err}^a + \mathrm{Err}^d.$$

We call the first term of RHS of Eq. (68) as $\mathcal{L}_{LIWHVI_{K'}}$. Each $\psi_{(i,0)}$ is sampled from $q(\psi \mid \mathbf{x})$ and each $\psi_{(i,k)}$ is sampled from $\tau\left(\psi \mid \mathbf{z}_i, \mathbf{x}\right)$. $\zeta_{(i,j)}$ is sampled from $q\left(\zeta \mid \mathbf{z}_i\right)$ and $\mathbf{z}_i$ is sampled from $q\left(\mathbf{z} \mid \mathbf{x}, \psi_{(i,0)}\right)$. Based on the assumption that $\mathbb{P}^{t-1}$ is fixed and $|KL_1 - KL_2| = 0$, we have $\mathcal{L}_{LIWHVI_{K'}} \ge \mathcal{L}_{LIWHVI_{K'+1}}$.

Ladder Variational Autoencoders (LVA) Sønderby et al. [2016]. LVA introduces a new inference model into the VAE framework, which allows to generate the latent variable $\mathbf{z}_i$ in each i-th layer and its approximate posterior is expressed as :

$$q\left(\mathbf{z} \mid \mathbf{x}\right) = q\left(\mathbf{z}_1 \mid \mathbf{x}\right) \prod_{i=2}^{L} q\left(\mathbf{z}_i \mid \mathbf{z}_{i-1}\right). \qquad (69)$$

We can also extend ELBO frm LVAE to the LLL setting, resulting in :

$$\frac{1}{t} \sum_{i=1}^{t} \mathbb{E}_{\mathbf{x}_i^T \sim \mathcal{P}_i} \{-\mathcal{L}_{ELBO}\left(\mathbf{x}_i^T; h\right)\} \le \mathbb{E}_{\mathbf{x}^t \sim \mathbb{P}^{t-1} \otimes \tilde{\mathcal{P}}_t} \{-\mathcal{L}_{LVA}\left(\tilde{\mathbf{x}}^t; h\right)\}$$
$$+ |KL_1 - KL_2| + \mathrm{Err}^a + \mathrm{Err}^d. \qquad (70)$$

We use the approximate posterior (Eq. (69)) in $|KL_1 - KL_2|$.

G Forgetting behaviour of other generative models

In this section, we extend the proposed GB to other types of generative models including GANs and Energy-based GANs.

G.1 Generative Adversarial Nets (GANs)

The discrepancy distance has been used in GANs Adlam et al. [2019] but its usage has not been explored within the LLL setting. Following from Adlam et al. [2019], we define $\mathcal{L}_{\mathcal{H}} = \{\mathcal{L}(h(\mathbf{x}), h'(\mathbf{x})) : h, h' \in \mathcal{H}\}$ as the family of discriminators which is used in the discrepancy distance (See Definition 2 in the paper). The errors of the model on the target distribution are bounded, as in Adlam et al. [2019] :

$$\mathcal{R}_{\mathcal{P}_i}\left(h, f_{\mathcal{P}_i}\right) \le \mathcal{R}_{\mathbb{P}^1}\left(h, f_{\mathcal{P}_i}\right) + disc_{\mathcal{L}}^{\star}\left(\mathcal{P}_1, \mathbb{P}^i\right), \qquad (71)$$

where we assume that $f_{\mathcal{P}_i}$ is the true labeling function for the i-th task. The detailed proof can be found in Adlam et al. [2019]. $\mathbb{P}^1$ represents the generator distribution of a GAN model trained on the i-th task learning. However, Eq. (71) is only applied for a single domain. In the following, we generalize Eq. (71) to the LLL setting, inspired by the proposed GB.

In order to overcome the forgetting, GANs are trained in a Self-Supervised (GR process) fashion in which the generator and discriminator are retrained on its generations. In this case, the previously generated samples and samples from the current task are treated as real images while the generated images during the training are treated as fake images. In the following, we derive a GB between the target distribution and the evolved source distribution.

Proposition 1 *Let a GAN model be trained on a sequence of t tasks by using the GR process, then we derive a GB for this GAN between the target and the evolved source distribution during the t-th task learning :*

$$\mathcal{R}_{\mathcal{P}_{(1:t)}}\left(h, f_{\mathcal{P}_{(1:t)}}\right) \le \mathcal{R}_{\mathbb{P}^t}\left(h, f_{\mathcal{P}_{(1:t)}}\right) + disc_{\mathcal{L}}^{\star}\left(\mathcal{P}_{(1:t)}, \mathbb{P}^t\right), \qquad (72)$$

where $\mathcal{P}_{(1:t)}$ represents the mixture distribution formed by samples draw from $\{\mathcal{P}_1, \ldots, \mathcal{P}_t\}$. $\mathbb{P}^t$ represents the generator distribution of a GAN model trained on the t-th task learning. In the following, we derive a GB to exhibit how a GAN model lose previously learnt knowledge for each task learning.

Proposition 2 *For a given a sequence of t tasks, a GAN model is trained with the GR process. We derive a GB on the risk of all tasks between the target distribution and the evolved distribution at the t-th task learning :*

$$\sum_{k=1}^{t} \mathcal{R}_{\mathcal{P}_k}\left(h, f_{\mathcal{P}_k}\right) \leq \sum_{k=1}^{t} \left\{ \mathcal{R}_{\mathbb{P}_k^t}\left(h, f_{\mathcal{P}_k}\right) + \sum_{j=k}^{t} \{disc_{\mathcal{L}}^{\star}\left(\mathbb{P}_k^{j-1}, \mathbb{P}_k^j\right)\} + disc_{\mathcal{L}}^{\star}\left(\mathbb{P}_k^{k-1}, \mathcal{P}_i\right) \right\}, \tag{73}$$

Eq. (73) explicitly measures the degenerated performance for each task learning, caused by the discrepancy distance between two distributions $\mathbb{P}_i^{j-1}$ and $\mathbb{P}_i^j$. From Eq. (73), we find that as learning more tasks (t is increased), The generalization performance of the GAN model would be gradually degenerated since the discrepancy distance terms are increased (The second term in the right hand side of Eq. (73).

Proof. In order to measure the forgetting behaviour of a GAN for each task learning, we need to define the individual approximation distribution related to each task. Let us define the approximation distribution $\mathbb{P}_i^j$ formed by the sampling process $\mathbf{x} \sim \mathbb{P}^j$ if $I_{\mathcal{T}}(\mathbf{x}) = i$. $\mathbb{P}_i^j$ is the probabilistic representation of the generated data related to the i-th task where j represents that the GAN model has been trained on j number of tasks. we use $\mathbb{P}_i^{(i-1)}$ represent $\tilde{\mathcal{P}}_i$ for simplicity. Therefore, for the i-th task, we can have the following bound :

$$\mathcal{R}_{\mathbb{P}_i^{(i-1)}}\left(h, f_{\mathcal{P}_i}\right) \leq \mathcal{R}_{\mathbb{P}_i^i}\left(h, f_{\mathcal{P}_i}\right) + disc_{\mathcal{L}}^{\star}\left(\mathbb{P}_i^i, \mathbb{P}_i^{(i-1)}\right), \tag{74}$$

and

$$\mathcal{R}_{\mathbb{P}_i^i}\left(h, f_{\mathcal{P}_i}\right) \leq \mathcal{R}_{\mathbb{P}_i^{(i+1)}}\left(h, f_{\mathcal{P}_i}\right) + disc_{\mathcal{L}}^{\star}\left(\mathbb{P}_i^{(i+1)}, \mathbb{P}_i^i\right), \tag{75}$$

In the following, we treat $\mathbb{P}_i^{(i+1)}$ as the target distribution and $\mathbb{P}_i^{(i+2)}$ as the source distribution, we have :

$$\mathcal{R}_{\mathbb{P}_i^{(i+1)}}\left(h, f_{\mathcal{P}_i}\right) \leq \mathcal{R}_{\mathbb{P}_i^{(i+2)}}\left(h, f_{\mathcal{P}_i}\right) + disc_{\mathcal{L}}^{\star}\left(\mathbb{P}_i^{(i+2)}, \mathbb{P}_i^{(i+1)}\right), \tag{76}$$

We repeat this process, resulting in :

$$\mathcal{R}_{\mathbb{P}_i^{((i+2)}}\left(h, f_{\mathcal{P}_i}\right) \leq \mathcal{R}_{\mathbb{P}_i^{((i+3)}}\left(h, f_{\mathcal{P}_i}\right) + disc_{\mathcal{L}}^{\star}\left(\mathbb{P}_i^{(i+3)}, \mathbb{P}_i^{(i+2)}\right)$$
$$\cdots$$
$$\cdots \tag{77}$$
$$\mathcal{R}_{\mathbb{P}_i^{t-1}}\left(h, f_{\mathcal{P}_i}\right) \leq \mathcal{R}_{\mathbb{P}_i^t}\left(h, f_{\mathcal{P}_i}\right) + disc_{\mathcal{L}}^{\star}\left(\mathbb{P}_i^t, \mathbb{P}_i^{t-1}\right)$$

We then sum up all inequalities, resulting in :

$$\mathcal{R}_{\mathbb{P}_i^{(i-1)}}\left(h, f_{\mathcal{P}_i}\right) \leq \mathcal{R}_{\mathbb{P}_i^t}\left(h, f_{\mathcal{P}_i}\right) + \sum_{j=i}^{t} disc_{\mathcal{L}}^{\star}\left(\mathbb{P}_i^{j-1}, \mathbb{P}_i^j\right), \tag{78}$$

We can observe that the left hand side of Eq. (78) is also an upper bound to the target risk of the model at the i-th task :

$$\mathcal{R}_{\mathcal{P}_i}\left(h, f_{\mathcal{P}_i}\right) \leq \mathcal{R}_{\mathbb{P}_i^{(i-1)}}\left(h, f_{\mathcal{P}_i}\right) + disc_{\mathcal{L}}^{\star}\left(\mathbb{P}_i^{(i-1)}, \mathcal{P}_i\right), \tag{79}$$

By comparing Eq. (79) and Eq. (78), we have a GB for the i-th task :

$$\mathcal{R}_{\mathcal{P}_i}\left(h, f_{\mathcal{P}_i}\right) \leq \mathcal{R}_{\mathbb{P}_i^t}\left(h, f_{\mathcal{P}_i}\right) + \sum_{j=i}^{t} \{disc_{\mathcal{L}}^{\star}\left(\mathbb{P}_i^{j-1}, \mathbb{P}_i^j\right)\} + disc_{\mathcal{L}}^{\star}\left(\mathbb{P}_i^{i-1}, \mathcal{P}_i\right), \tag{80}$$

Then we can easily obtain a GB for all tasks based on Eq. (80).

$$\sum_{k=1}^{t} \mathcal{R}_{\mathcal{P}_k}\left(h, f_{\mathcal{P}_k}\right) \leq \sum_{k=1}^{t} \left\{ \mathcal{R}_{\mathbb{P}_k^t}\left(h, f_{\mathcal{P}_k}\right) + \sum_{j=k}^{t} \{disc_{\mathcal{L}}^{\star}\left(\mathbb{P}_k^{j-1}, \mathbb{P}_k^j\right)\} + disc_{\mathcal{L}}^{\star}\left(\mathbb{P}_k^{k-1}, \mathcal{P}_i\right) \right\}, \tag{81}$$

This proves Proposition 2.

G.2 Energy-based GANs

Energy function was firstly used in GANs Zhao et al. [2017] in 2016, called EBGAN. Different from GANs, EBGAN introduces using a discriminator consisting of an auto-encoder to calculate the energy value which is the reconstruction loss. It shows that EBGAN assigns higher energy to fake images and low energy to real images. We implement $h \in \mathcal{H}$ as the discriminator. EBGAN in lifelong learning can be seen as an advanced Self-Supervised VAEs without the KL divergence term since its generator can produce sharper images than VAEs. Then we can derive a GB for EBGAN, similar to that defined by Theorem 2.

$$\frac{1}{t} \sum\nolimits_{i=1}^{t} \mathcal{R}_{\mathcal{P}_i}\left(h, f_{\mathcal{P}_i}\right) \leq \mathcal{R}_{\mathbb{P}^{t-1} \otimes \tilde{\mathcal{P}}_t}\left(h, h^*_{\mathbb{P}^{t-1} \otimes \tilde{\mathcal{P}}_t}\right) + \mathcal{R}_A\left(\mathcal{P}_{(1:t)}, \mathbb{P}^{t-1} \otimes \tilde{\mathcal{P}}_t\right), \tag{82}$$

Different from Theorem 2, $\mathbb{P}^{t-1}$ is approximated by the generator of EBGAN trained on the $(t-1)$ number of tasks. Since EBGAN does not have the KL regularization and therefore the generator tends to generate more realistic samples when comparing with the generator of VAEs.

H   Risk bound estimation from finite samples

In this section, we introduce how to estimate the risk bound from finite samples.

Definition 1 *(Rademacher complexity). Let $\mathcal{H}$ represent a hypothesis class, For a given unlabeled sample $U = \{\mathbf{x}_i\}_{i=1}^{m}$, the Rademacher complexity of $\mathcal{H}$ with respect to the sample U is defined as follows :*

$$\mathrm{Re}_U\left(\mathcal{H}\right) = \underset{\mathcal{K}}{\mathbb{E}}\left[\sup_{h \in \mathcal{H}} \frac{2}{m} \sum_{i=1}^{m} \mathcal{K}_i h\left(\mathbf{x}_i\right)\right] \tag{83}$$

where $\mathcal{K}_i$ is an independent uniform random variable within $\{-1, +1\}$. The Rademacher complexity for the whole hypothesis class is defined as :

$$\mathrm{Re}_n\left(\mathcal{H}\right) = \mathbb{E}_{U \sim (D)^n} \mathrm{Re}_U\left(\mathcal{H}\right) \tag{84}$$

Lemma 3 Let $\mathcal{H}$ represent a hypothesis class and let $\mathcal{L}_{\mathcal{H}} = \{\mathcal{X} \to \mathcal{L}(h'(\mathbf{x}), h(\mathbf{x})), (h', h) \in \mathcal{H}\}$. Let $\mathcal{L} : \mathcal{X} \times \mathcal{X} \to \mathbb{R}_+$ be a loss function, satisfying $\forall (\mathbf{x}, \mathbf{x}') \in \mathcal{X}, \mathcal{L}(\mathbf{x}, \mathbf{x}') > M$ and M is the positive value. Let $\mathcal{P}$ be a distribution over $\mathcal{X}$ and let $\tilde{\mathcal{P}}$ be the corresponding empirical distribution formed by a sample $U = \{\mathbf{x}_1, \dots, \mathbf{x}_m\}$. Then for any $\delta \in (0, 1)$, with the probability $1 - \delta$, we have :

$$disc_{\mathcal{L}}\left(\mathcal{P}, \tilde{\mathcal{P}}\right) \leq \mathrm{Re}_U\left(\mathcal{L}_{\mathcal{H}}\right) + 3M\sqrt{\frac{\log \frac{2}{\delta}}{2m}} \tag{85}$$

From Lemma 3, we can generalize the bound in case of more general loss function $\forall (\mathbf{x}', \mathbf{x}) \in \mathcal{X}^2, \mathcal{L}_q(\mathbf{x}', \mathbf{x}) = |\mathbf{x}' - \mathbf{x}|^q$ for some q. In the following, we show how to estimate the discrepancy distance by using the finite number of samples according to Mansour et al. [2009].

Corollary 1 *Let $\mathcal{P}$ and $\mathbb{P}$ represent two domains over $\mathcal{X}$, respectively. Let $U_{\mathcal{P}}$ and $U_{\mathbb{P}}$ represent samples of size $m_{\mathcal{P}}$ and $m_{\mathbb{P}}$, drawn independently from $U_{\mathcal{P}}$ and $U_{\mathbb{P}}$. Let $\hat{\mathcal{P}}$ and $\hat{\mathbb{P}}$ represent the empirical distributions for $U_{\mathcal{P}}$ and $U_{\mathbb{P}}$. let $\mathcal{L}(\mathbf{x}', \mathbf{x}) = |\mathbf{x}' - \mathbf{x}|^2$ be a loss function, satisfying $\forall (\mathbf{x}, \mathbf{x}') \in \mathcal{X}, \mathcal{L}(\mathbf{x}, \mathbf{x}') > M$ and M is the positive value. Then with probability $1 - \delta$, we have :*

$$disc_{\mathcal{L}}\left(\mathcal{P}, \mathbb{P}\right) \leq disc_{\mathcal{L}}\left(\hat{\mathcal{P}}, \hat{\mathbb{P}}\right) + 4q\left(\mathrm{Re}_{U_{\mathcal{P}}}\left(\mathcal{H}\right) + \mathrm{Re}_{U_{\mathbb{P}}}\left(\mathcal{H}\right)\right)$$
$$+ 3M\left(\sqrt{\frac{\log\left(\frac{4}{\delta}\right)}{2m_{\mathcal{P}}}} + \sqrt{\frac{\log\left(\frac{4}{\delta}\right)}{2m_{\mathbb{P}}}}\right) \tag{86}$$

We use $disc_{\mathcal{L}}^{\star}(\mathcal{P}, \mathbb{P})$ to represent the right hand side of eq.(86).

Corollary 1 can allow us to estimate the risk bound by using finite samples. In the following, we derive the risk bound for the model based on the empirical data distribution.

I   Self-Supervised VAEs

Self-Supervised technologies have been used in the semi-supervised learning Zhai et al. [2019], multi-task learning Doersch and Zisserman [2017] and generative models Chen et al. [2019], Tran et al. [2019], Cao et al. [2020]. In the context of lifelong learning, self-supervised learning is used to overcome catastrophic forgetting Seff et al. [2017] that this paper focuses on.

The learning process of the Self-Supervised VAEs is training a model $\mathcal{M}$ on the samples generated by its decoder $g_\theta(.)$ and the samples from the current task. For instance, at the i-th task learning, we get samples from the mixture distribution $\mathbb{P}^{(i-1)} \otimes \tilde{\mathcal{P}}_i$ where $\mathbb{P}^{(i-1)}$ is the distribution for $\mathcal{M}^{(i-1)}$, which was trained with the $(i-1)$-th task. We assume that at the i-th task learning, we can generate a joint sample $(\mathbf{x}'_{(i-1)}, \mathbf{x}_i^S)$ from $\mathbb{P}^{(i-1)}$ and $\tilde{\mathcal{P}}_i$, respectively. Then the model can be represented by $p_\theta(\mathbf{x}'_{(i-1)}, \mathbf{x}_i^S, \mathbf{z}'_{(i-1)}, \mathbf{z}_i^S) = p_\theta(\mathbf{x}'_{(i-1)}, \mathbf{x}_i^S | \mathbf{z}'_{(i-1)}, \mathbf{z}_i^S) p(\mathbf{z}'_{(i-1)}, \mathbf{z}_i^S)$ where $\mathbf{z}'_{(i-1)}$ and $\mathbf{z}_i^S$ are two latent variables corresponding to different observed variables. We aim to maximize the sample log-likelihood, expressed as :

$$
\begin{aligned}
\log p_\theta(\mathbf{x}'_{(i-1)}, \mathbf{x}_i^S) &= \log \int \int p_\theta(\mathbf{x}'_{(i-1)}, \mathbf{x}_i^S | \mathbf{z}'_{(i-1)}, \mathbf{z}_i^S) p(\mathbf{z}'_{(i-1)}, \mathbf{z}_i^S) d\mathbf{z}'_{(i-1)} d\mathbf{z}_i^S \\
&\geq \int \int \log p_\theta(\mathbf{x}'_{(i-1)}, \mathbf{x}_i^S | \mathbf{z}'_{(i-1)}, \mathbf{z}_i^S) p(\mathbf{z}'_{(i-1)}, \mathbf{z}_i^S) d\mathbf{z}'_{(i-1)} d\mathbf{z}_i^S
\end{aligned}
\tag{87}
$$

Since $\mathbf{x}'_{(i-1)}$ and $\mathbf{x}_i^S$ are independent, we can rewrite Eq.(87) as :

$$
\log p_\theta(\mathbf{x}'_{(i-1)}) + \log p_\theta(\mathbf{x}_i^S) \geq \int \log p_\theta(\mathbf{x}'_{(i-1)} | \mathbf{z}'_{(i-1)}) p(\mathbf{z}'_{(i-1)}) d\mathbf{z}'_{(i-1)} + \int \log p_\theta(\mathbf{x}_i^S | \mathbf{z}_i^S) p(\mathbf{z}_i^S) d\mathbf{z}_i^S
\tag{88}
$$

By considering to use the variational distributions Kingma and Welling [2013], we rewrite eq.(88) as :

$$
\begin{aligned}
\log p_\theta(\mathbf{x}'_{(i-1)}) + \log p_\theta(\mathbf{x}_i^S) \geq\; & \mathbb{E}_{q_{\omega^i}(\mathbf{z}'_{(i-1)} | \mathbf{x}'_{(i-1)})}[\log p_\theta(\mathbf{x}'_{(i-1)} | \mathbf{z}'_{(i-1)})] + KL[q_{\omega^i}(\mathbf{z}'_{(i-1)} | \mathbf{x}'_{(i-1)}) || p(\mathbf{z}'_{(i-1)})] \\
& + \mathbb{E}_{q_{\omega^i}(\mathbf{z}_i^S | \mathbf{x}_i^S)}[\log p_\theta(\mathbf{x}_i^S | \mathbf{z}_i^S)] + KL[q_{\omega^i}(\mathbf{z}_i^S | \mathbf{x}_i^S) || p(\mathbf{z}_i^S)]
\end{aligned}
\tag{89}
$$

where $p(\mathbf{z}'_{(i-1)})$ and $p(\mathbf{z}_i^S)$ are normal distributions. Eq.(89) is used for training the VAEs model in a Self-Supervised fashion at the i-th task learning. In order to simplify the loss function used for VAEs under LLL, we uniformly generate samples from a mixture distribution $\mathbb{P}^i \otimes \tilde{\mathcal{P}}_i$. Then the loss function can be defined as :

$$
\log p_\theta(\tilde{\mathbf{x}}^i) \geq \mathbb{E}_{q_{\omega^i}(\mathbf{z} | \tilde{\mathbf{x}}^i)}[\log p_\theta(\tilde{\mathbf{x}}^i | \mathbf{z})] + KL[q_{\omega^i}(\mathbf{z} | \tilde{\mathbf{x}}^i) || p(\mathbf{z})]
\tag{90}
$$

where $\tilde{\mathbf{x}}^i$ is sampled from $\mathbb{P}^i \otimes \tilde{\mathcal{P}}_i$. Based on this objective function, we analyze how the Self-Supervised VAEs can achieve an optimal solution in lifelong generative modelling.

Proposition 3 *Let $\tilde{\mathcal{P}}_{(1:t)} = \{\tilde{\mathcal{P}}_1 \otimes \tilde{\mathcal{P}}_2, \ldots, \otimes \tilde{\mathcal{P}}_t\}$ be a mixture distribution. The Self-Supervised VAEs approximates $\tilde{\mathcal{P}}_{(1:t)}$ exactly when VAEs approximates each target distribution exactly in each task learning.*

When satisfying Proposition 3, the GB from Theorem 2 in the paper is redefined as :

$$
\frac{1}{t} \sum_{i=1}^{t} \mathcal{R}_{\mathcal{P}_i}(h, f_{\mathcal{P}_i}) \leq \frac{1}{t} \sum_{i=1}^{t} \{\mathcal{R}_{\tilde{\mathcal{P}}_i}(h, h_{\tilde{\mathcal{P}}_i}^*)\} + \mathcal{R}_A(\mathcal{P}_{(1:t)}, \tilde{\mathcal{P}}_{(1:t)}).
\tag{91}
$$

From eq.(91), the lifelong learning problem is transformed into the multiple source-target generalization problem. In the following, we assume that the random variables $\{\mathbf{x}_1^T, \ldots, \mathbf{x}_t^T\}$ are sampled from $\{\mathcal{P}_1, \ldots, \mathcal{P}_t\}$ and $\{\mathbf{x}_1^S, \ldots, \mathbf{x}_t^S\}$ are sampled from $\{\tilde{\mathcal{P}}_1, \ldots, \tilde{\mathcal{P}}_t\}$. We can define the KL divergence as follows :

$$
\begin{aligned}
\frac{1}{t} \sum_{i=1}^{t} \left\{ \mathbb{E}_{\mathcal{P}_i} KL\left(p\left(\mathbf{z} | \mathbf{x}_i^T\right) || p\left(\mathbf{z}\right)\right) \right\} \leq\; & \frac{1}{t} \sum_{i=1}^{t} \left\{ \mathbb{E}_{\tilde{\mathcal{P}}_i} KL\left(p\left(\mathbf{z} | \mathbf{x}_i^S\right) || p\left(\mathbf{z}\right)\right) \right\} \\
& + \left| \frac{1}{t} \sum_{i=1}^{t} \left\{ \mathbb{E}_{\mathcal{P}_i} KL\left(p\left(\mathbf{z} | \mathbf{x}_i^T\right) || p\left(\mathbf{z}\right)\right) \right\} - \frac{1}{t} \sum_{i=1}^{t} \left\{ \mathbb{E}_{\tilde{\mathcal{P}}_i} KL\left(p\left(\mathbf{z} | \mathbf{x}_i^S\right) || p\left(\mathbf{z}\right)\right) \right\} \right|
\end{aligned}
\tag{92}
$$

where the latest term in RHS of Eq. (92) is defined as $D_{diff}(\mathbf{x}_i^T, \mathbf{x}_i^S)$. According to the definition of NLL, we have :

$$\frac{1}{t}\sum_{i=1}^{t}\mathbb{E}_{\mathcal{P}_i}\{-\log p\left(\mathbf{x}_i^T\right)\}\leq\frac{1}{t}\sum_{i=1}^{t}\mathbb{E}_{\mathcal{P}_i}\{-\mathcal{L}_{ELBO}\left(\mathbf{x}_i^T;h\right)\}. \tag{93}$$

In the following, we consider to sum up Eq. (92) and Eq. (91), resulting in :

$$\frac{1}{t}\sum_{i=1}^{t}\mathbb{E}_{\mathcal{P}_i}\{-\mathcal{L}_{ELBO}\left(\mathbf{x}_i^T;h\right)\}\leq\frac{1}{t}\sum_{i=1}^{t}\left\{\mathbb{E}_{\tilde{\mathcal{P}}_i}\{-\mathcal{L}_{ELBO}\left(\mathbf{x}_i^S;h\right)+D_{diff}\left(\mathbf{x}_i^T,\mathbf{x}_i^S\right)\}\right\}+\mathcal{R}_A\left(\mathcal{P}_{(1:t)},\tilde{\mathcal{P}}_{(1:t)}\right). \tag{94}$$

Eq. (94) is derived based on the formulation of ELBO (See details in Lemma 1 of the paper). It can be observed that LHS of Eq. (94) is RHS of Eq. (93) and we have :

$$\frac{1}{t}\sum_{i=1}^{t}\mathbb{E}_{\mathcal{P}_i}\{-\log p\left(\mathbf{x}_i^T\right)\}\leq\frac{1}{t}\sum_{i=1}^{t}\left\{-\mathcal{L}_{ELBO}\left(\mathbf{x}_i^S;h\right)+D_{diff}\left(\mathbf{x}_i^T,\mathbf{x}_i^S\right)\right\}+\mathcal{R}_A\left(\mathcal{P}_{(1:t)},\tilde{\mathcal{P}}_{(1:t)}\right). \tag{95}$$

Eq. (95) explicitly defines a GB for the VAE mode that is trained on all training sets or the generator distribution that approximates the distribution of all training samples, exactly.
Proof. We begin with the first task in which $\mathcal{M}$ is trained on $\tilde{\mathcal{P}}_1$ and the optimal generator distribution $\mathbb{P}^1=\tilde{\mathcal{P}}_1$. In the second task learning, $\mathcal{M}^1$ is trained on the mixture distribution $\mathbb{P}^1\otimes\tilde{\mathcal{P}}_2$. Since $\mathbb{P}^1=\tilde{\mathcal{P}}_1$, we replace $\mathbb{P}^1\otimes\tilde{\mathcal{P}}_2$ by $\tilde{\mathcal{P}}_1\otimes\tilde{\mathcal{P}}_2$ and the optimal generator distribution $\mathbb{P}^2=\tilde{\mathcal{P}}_1\otimes\tilde{\mathcal{P}}_2$. By using the mathematical recursion, we have $\mathbb{P}^t=\tilde{\mathcal{P}}_{(1:t)}$ if VAEs approximates the target distribution exactly in each task learning.

Proposition 4 *For a given sequence of tasks $\{\mathcal{T}_1,\ldots,\mathcal{T}_t\}$, the sample log-likelihood of a single model $\mathbf{M}$ evaluated on the target distribution $\{\mathcal{P}_1,\ldots,\mathcal{P}_t\}$ during the t-th task learning is bounded by the accumulated errors and the generalization errors.*

$$\frac{1}{t}\sum_{i=1}^{t}\mathbb{E}_{\mathcal{P}_i}\{-\log p\left(\mathbf{x}_t^T\right)\}\leq\mathbb{E}_{\mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t}\{-\mathcal{L}_{ELBO}\left(\mathbf{x}^t;h\right)\}+\underbrace{\frac{1}{t}\sum_{i=1}^{t}\{D_{diff}\left(\mathbf{x}_i^S,\mathbf{x}^t\right)\}+\mathcal{R}_A\left(\tilde{\mathcal{P}}_{(1:t)},\mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t\right)}_{\textit{Accumulated error}}$$

$$+\underbrace{\frac{1}{t}\sum_{i=1}^{t}\{D_{diff}\left(\mathbf{x}_i^S,\mathbf{x}_i^T\right)\}+\mathcal{R}_A\left(\tilde{\mathcal{P}}_{(1:t)},\mathcal{P}_{(1:t)}\right)}_{\textit{Generalization error}}$$

$$\tag{96}$$

We have the accumulated error Err^c and the generalization error term Err^h :

$$\mathrm{Err}^c=\frac{1}{t}\sum_{i=1}^{t}\{D_{diff}\left(\mathbf{x}_i^S,\mathbf{x}^t\right)\}+\mathcal{R}_A\left(\tilde{\mathcal{P}}_{(1:t)},\mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t\right) \tag{97}$$

$$\mathrm{Err}^h=\frac{1}{t}\sum_{i=1}^{t}\{D_{diff}\left(\mathbf{x}_i^S,\mathbf{x}_i^T\right)\}+\mathcal{R}_A\left(\tilde{\mathcal{P}}_{(1:t)},\mathcal{P}_{(1:t)}\right) \tag{98}$$

Then, Eq. (96) is rewritten as :

$$\frac{1}{t}\sum_{i=1}^{t}\mathbb{E}_{\mathcal{P}_i}\{-\log p\left(\mathbf{x}_t^T\right)\}\leq\mathbb{E}_{\mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t}\{-\mathcal{L}_{ELBO}\left(\mathbf{x}^t;h\right)\}+\mathrm{Err}^c+\mathrm{Err}^h \tag{99}$$

Eq. (99) explicitly defines a GB for the negative sample log-likelihood for $\{\mathcal{P}_1,\ldots,\mathcal{P}_t\}$ where Err^c and Err^h are two error terms. We show that the gap on this GB depends not only on Err^c but also on Err^h.
Proof. Since the model $\mathcal{M}$ is trained on a sequence of distributions $\{\tilde{\mathcal{P}},\ldots,\tilde{\mathcal{P}}_t\}$, we would like to derive the GB between the model distribution and $\{\tilde{\mathcal{P}},\ldots,\tilde{\mathcal{P}}_t\}$ as :

$$\frac{1}{t}\sum_{i=1}^{t}\mathcal{R}_{\tilde{\mathcal{P}}_i}\left(h,f_{\tilde{\mathcal{P}}_i}\right)\leq\mathcal{R}_{\mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t}\left(h,h_{\mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t}^*\right)+\mathcal{R}_A\left(\tilde{\mathcal{P}}_{(1:t)},\mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t\right), \tag{100}$$

where $\tilde{\mathcal{P}}_{(1:t)}$ represents the mixture distribution $\{\tilde{\mathcal{P}}_1 \otimes \cdots \otimes \tilde{\mathcal{P}}_t\}$. As similar to the proof from Lemma 1 of the paper, we add the KL divergence term in both sides in eq.(100), resulting in :

$$\frac{1}{t}\sum_{i=1}^{t}\{\mathcal{R}_{\tilde{\mathcal{P}}_i}\left(h, f_{\tilde{\mathcal{P}}_i}\right) + KL\left(q\left(\mathbf{z}|\mathbf{x}_i^S\right)||p\left(\mathbf{z}\right)\right)\} \le \mathcal{R}_{\mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t}\left(h, h^*_{\mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t}\right) + \mathbb{E}_{\mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t}KL(p(\mathbf{z}|\tilde{\mathbf{x}}^t)||p(\mathbf{z}))$$

$$+ \frac{1}{t}\sum_{i=1}^{t}\{+D_{diff}\left(\mathbf{x}_i^S, \mathbf{x}^t\right)\} + \mathcal{R}_A\left(\tilde{\mathcal{P}}_{(1:t)}, \mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t\right), \tag{101}$$

We can rewrite Eq.(101) According to ELBO, resulting in :

$$\frac{1}{t}\sum_{i=1}^{t}\mathbb{E}_{\tilde{\mathcal{P}}_i}\{-\mathcal{L}_{ELBO}\left(\mathbf{x}_i^S; h\right)\} \le \mathbb{E}_{\mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t}\{-\mathcal{L}_{ELBO}\left(\mathbf{x}^t; h\right)\} + \frac{1}{t}\sum_{i=1}^{t}\{D_{diff}\left(\mathbf{x}_i^S, \mathbf{x}^t\right)\} + \mathcal{R}_A\left(\tilde{\mathcal{P}}_{(1:t)}, \mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t\right), \tag{102}$$

Secondly, we consider an ideal situation in which the model $\mathcal{M}$ approximates the learning target distribution exactly in each task learning. Then, we can derive the GB for the risk between $\{\mathcal{P}_1, \ldots, \mathcal{P}_t\}$ and $\{\tilde{\mathcal{P}}_1, \ldots, \tilde{\mathcal{P}}_t\}$:

$$\frac{1}{t}\sum_{i=1}^{t}\mathbb{E}_{\mathcal{P}_i}\{-\mathcal{L}_{ELBO}\left(\mathbf{x}_i^T; h\right)\} \le \frac{1}{t}\sum_{i=1}^{t}\mathbb{E}_{\tilde{\mathcal{P}}_i}\{-\mathcal{L}_{ELBO}\left(\mathbf{x}_i^S; h\right)\} + \frac{1}{t}\sum_{i=1}^{t}\{D_{diff}\left(\mathbf{x}_i^S, \mathbf{x}_i^T\right)\} + \mathcal{R}_A\left(\tilde{\mathcal{P}}_{(1:t)}, \mathcal{P}_{(1:t)}\right), \tag{103}$$

We also know that the left hand side of Eq. (103) is a bound for $\frac{1}{t}\sum_{i=1}^{t}\mathbb{E}_{\mathcal{P}_i}\left\{-\log p\left(\mathbf{x}_t^T\right)\right\}$, expressed as :

$$\frac{1}{t}\sum_{i=1}^{t}\mathbb{E}_{\mathcal{P}_i}\left\{-\log p\left(\mathbf{x}_t^T\right)\right\} \le \frac{1}{t}\sum_{i=1}^{t}\mathbb{E}_{\mathcal{P}_i}\{-\mathcal{L}_{ELBO}\left(\mathbf{x}_i^T; h\right)\} \tag{104}$$

By comparing eq.(104) and eq.(103), we have :

$$\frac{1}{t}\sum_{i=1}^{t}\mathbb{E}_{\mathcal{P}_i}\left\{-\log p\left(\mathbf{x}_t^T\right)\right\} \le \frac{1}{t}\sum_{i=1}^{t}\{\mathbb{E}_{\tilde{\mathcal{P}}_i}\{-\mathcal{L}_{ELBO}\left(\mathbf{x}_i^S; h\right)\}\} + \frac{1}{t}\sum_{i=1}^{t}\{D_{diff}\left(\mathbf{x}_i^S, \mathbf{x}_i^T\right)\} + \mathcal{R}_A\left(\tilde{\mathcal{P}}_{(1:t)}, \mathcal{P}_{(1:t)}\right) \tag{105}$$

Then we add the last two terms in the right hand side of eq.(103) to both sides of eq.(102), resulting in :

$$\frac{1}{t}\sum_{i=1}^{t}\{\mathbb{E}_{\tilde{\mathcal{P}}_i}\{-\mathcal{L}_{ELBO}\left(\mathbf{x}_i^S; h\right)\}\} + \frac{1}{t}\sum_{i=1}^{t}\{D_{diff}\left(\mathbf{x}_i^S, \mathbf{x}_i^T\right)\} + \mathcal{R}_A\left(\tilde{\mathcal{P}}_{(1:t)}, \mathcal{P}_{(1:t)}\right) \le \mathbb{E}_{\mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t}\{-\mathcal{L}_{ELBO}\left(\mathbf{x}^t; h\right)\}$$

$$+ \frac{1}{t}\sum_{i=1}^{t}\{D_{diff}\left(\mathbf{x}_i^S, \mathbf{x}^t\right)\} + \mathcal{R}_A\left(\tilde{\mathcal{P}}_{(1:t)}, \mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t\right) + \frac{1}{t}\sum_{i=1}^{t}\{D_{diff}\left(\mathbf{x}_i^S, \mathbf{x}_i^T\right)\} + \mathcal{R}_A\left(\tilde{\mathcal{P}}_{(1:t)}, \mathcal{P}_{(1:t)}\right) \tag{106}$$

It can be observed that the right hand side of eq.(105) is equal to the left hand side of eq.(106). By comparing eq.(105) and eq.(106), we have :

$$\frac{1}{t}\sum_{i=1}^{t}\mathbb{E}_{\mathcal{P}_i}\left\{-\log p\left(\mathbf{x}_i^T\right)\right\} \le \mathbb{E}_{\mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t}\{-\mathcal{L}_{ELBO}\left(\mathbf{x}^t; h\right)\} + \frac{1}{t}\sum_{i=1}^{t}\{D_{diff}\left(\mathbf{x}_i^S, \mathbf{x}^t\right)\}$$

$$+ \mathcal{R}_A\left(\tilde{\mathcal{P}}_{(1:t)}, \mathbb{P}^{t-1}\otimes\tilde{\mathcal{P}}_t\right) + \frac{1}{t}\sum_{i=1}^{t}\{D_{diff}\left(\mathbf{x}_i^S, \mathbf{x}_i^T\right)\} + \mathcal{R}_A\left(\tilde{\mathcal{P}}_{(1:t)}, \mathcal{P}_{(1:t)}\right) \tag{107}$$

This proves Proposition 4.

Proposition 5 *For a given mixture model* **M***, The sample log-likelihood of* **M** *evaluated on the target distribution*

$\{\mathcal{P}_1, \ldots, \mathcal{P}_t\}$ *during the* t*-th task learning is bounded by :*

$$\frac{1}{t}\sum_{i=1}^{t}\mathbb{E}_{\mathcal{P}_i}\{-\mathcal{L}_{ELBO}\left(\mathbf{x}_i^T;h\right) \le \frac{1}{t}\sum_{i=1}^{|C'|}\sum_{j=1}^{\tilde{a}_i}\mathbb{E}_{\mathbb{P}_{a(i,j)}^{c(i,j)}}\left\{-\mathcal{L}_{ELBO}\left(\mathbf{x}_{a(i,j)}^t;h\right) + \mathcal{R}_A\left(\tilde{\mathcal{P}}_{a(i,j)}, \mathbb{P}_{a(i,j)}^{c(i,j)}\right)\right\}$$

$$+\frac{1}{t}\sum_{i=1}^{|C|}\mathbb{E}_{\tilde{\mathcal{P}}_{a_i}}\{-\mathcal{L}_{ELBO}\left(\mathbf{x}_{a_i}^S;h\right)\} + \frac{1}{t}\sum_{i=1}^{|C'|}\sum_{j=1}^{\tilde{a}_i}\{D_{diff}\left(\mathbf{x}_{a(i,j)}^S, \mathbf{x}_{a(i,j)}^t\right)\} \qquad (108)$$

$$+\frac{1}{t}\sum_{i=1}^{t}D_{diff}\left(\mathbf{x}_i^T, \mathbf{x}_i^S\right) + \mathcal{R}_A\left(\mathcal{P}_{(1:t)}, \widetilde{\mathcal{P}}_{(1:t)}\right)$$

From eq.(108), it can be observed that the generalization error term $\frac{1}{t}\sum_{i=1}^{t}D_{diff}\left(\mathbf{x}_i^T, \mathbf{x}_i^S\right) + \mathcal{R}_A\left(\mathcal{P}_{(1:t)}, \widetilde{\mathcal{P}}_{(1:t)}\right)$ is the same for both the single model $\mathcal{M}$ and the mixture model $\mathbf{M}$. This means that the mixture model can obtain a tight GB by reducing the accumulated error term during lifelong learning.

Proof. Firstly, we consider to take $\{\tilde{\mathcal{P}}_1, \ldots, \tilde{\mathcal{P}}_t\}$ as the target distribution and we can derive the GB for $\mathbf{M}$ as :

$$\frac{1}{t}\sum_{i=1}^{|C|}\{\mathcal{R}_{\tilde{\mathcal{P}}_{a_i}}\left(h, f_{\tilde{\mathcal{P}}_{a_i}}\right)\} + \frac{1}{t}\sum_{i=1}^{|C'|}\sum_{j=1}^{\tilde{a}_i}\{\mathcal{R}_{\tilde{\mathcal{P}}_{a(i,j)}}\left(h, f_{\tilde{\mathcal{P}}_{a(i,j)}}\right)\} \le \frac{1}{t}\sum_{i=1}^{|C'|}\sum_{j=1}^{\tilde{a}_i}\left\{\mathcal{R}_{\mathbb{P}_{a(i,j)}^{c(i,j)}}\left(h, f_{\mathbb{P}_{a(i,j)}^{c(i,j)}}\right)\right.$$

$$\left. +\mathcal{R}_A\left(\tilde{\mathcal{P}}_{a(i,j)}, \mathbb{P}_{a(i,j)}^{c(i,j)}\right)\right\} \qquad (109)$$

$$+\frac{1}{t}\sum_{i=1}^{|C|}\{\mathcal{R}_{\tilde{\mathcal{P}}_{a_i}}\left(h, f_{\tilde{\mathcal{P}}_{a_i}}\right)\}$$

We then add the KL divergence in both sides of eq.(109), resulting in :

$$\frac{1}{t}\sum_{i=1}^{|C|}\{\mathcal{R}_{\tilde{\mathcal{P}}_{a_i}}\left(h, f_{\tilde{\mathcal{P}}_{a_i}}\right) + KL\left(q\left(\mathbf{z}|\mathbf{x}_{a_i}^S\right)||p(\mathbf{z})\right)\} + \frac{1}{t}\sum_{i=1}^{|C'|}\sum_{j=1}^{\tilde{a}_i}\{\mathcal{R}_{\tilde{\mathcal{P}}_{a(i,j)}}\left(h, f_{\tilde{\mathcal{P}}_{a(i,j)}}\right) + KL\left(q\left(\mathbf{z}|\mathbf{x}_{a(i,j)}^S\right)||p(\mathbf{z})\right)\} \le$$

$$\frac{1}{t}\sum_{i=1}^{|C'|}\sum_{j=1}^{\tilde{a}_i}\left\{\mathcal{R}_{\mathbb{P}_{a(i,j)}^{c(i,j)}}\left(h, f_{\mathbb{P}_{a(i,j)}^{c(i,j)}}\right) + \mathcal{R}_A\left(\tilde{\mathcal{P}}_{a(i,j)}, \mathbb{P}_{a(i,j)}^{c(i,j)}\right) + KL\left(q\left(\mathbf{z}|\mathbf{x}_{a(i,j)}^t\right)||p(\mathbf{z})\right)\right\}$$

$$+\frac{1}{t}\sum_{i=1}^{|C|}\{\mathcal{R}_{\tilde{\mathcal{P}}_{a_i}}\left(h, f_{\tilde{\mathcal{P}}_{a_i}}\right)\} + KL\left(q\left(\mathbf{z}|\mathbf{x}_{a_i}^S\right)||p(\mathbf{z})\right) + \frac{1}{t}\sum_{i=1}^{|C'|}\sum_{j=1}^{\tilde{a}_i}\{D_{diff}\left(\mathbf{x}_{a(i,j)}^S, \mathbf{x}_{a(i,j)}^t\right)\}$$

$$(110)$$

We rewrite eq.(110) as the ELBO from as :

$$\frac{1}{t}\sum_{i=1}^{|C|}\mathbb{E}_{\tilde{\mathcal{P}}_{a_i}}\{-\mathcal{L}_{ELBO}\left(\mathbf{x}_{a_i}^S;h\right)\} + \frac{1}{t}\sum_{i=1}^{|C'|}\sum_{j=1}^{\tilde{a}_i}\mathbb{E}_{\tilde{\mathcal{P}}_{a(i,j)}}\{-\mathcal{L}_{ELBO}\left(\mathbf{x}_{a(i,j)}^S;h\right)\} \le$$

$$\frac{1}{t}\sum_{i=1}^{|C'|}\sum_{j=1}^{\tilde{a}_i}\mathbb{E}_{\mathbb{P}_{a(i,j)}^{c(i,j)}}\left\{-\mathcal{L}_{ELBO}\left(\mathbf{x}_{a(i,j)}^t;h\right) + \mathcal{R}_A\left(\tilde{\mathcal{P}}_{a(i,j)}, \mathbb{P}_{a(i,j)}^{c(i,j)}\right)\right\} \qquad (111)$$

$$+\frac{1}{t}\sum_{i=1}^{|C|}\mathbb{E}_{\tilde{\mathcal{P}}_{a_i}}\{-\mathcal{L}_{ELBO}\left(\mathbf{x}_{a_i}^S;h\right)\} + \frac{1}{t}\sum_{i=1}^{|C'|}\sum_{j=1}^{\tilde{a}_i}\{D_{diff}\left(\mathbf{x}_{a(i,j)}^S, \mathbf{x}_{a(i,j)}^t\right)\}$$

In the following, we consider to take $\{\mathcal{P}_1, \ldots, \mathcal{P}_t\}$ as the target distribution and $\{\tilde{\mathcal{P}}_1, \ldots, \tilde{\mathcal{P}}_t\}$ as the source distribution. Then we have :

$$\frac{1}{t}\sum_{i=1}^{t}\mathbb{E}_{\mathcal{P}_i}\{-\mathcal{L}_{ELBO}\left(\mathbf{x}_i^T;h\right) \le \frac{1}{t}\sum_{i=1}^{t}\mathbb{E}_{\tilde{\mathcal{P}}_i}\{-\mathcal{L}_{ELBO}\left(\mathbf{x}_i^S;h\right) + D_{diff}\left(\mathbf{x}_i^T, \mathbf{x}_i^S\right)\} + \mathcal{R}_A\left(\mathcal{P}_{(1:t)}, \tilde{\mathcal{P}}_{(1:t)}\right). \qquad (112)$$

We then add the latest two term of RHS of Eq. (112) into both sides of Eq. (111), resulting in :

$$
\frac{1}{t}\sum_{i=1}^{t} D_{diff}\left(\mathbf{x}_i^T, \mathbf{x}_i^S\right) + \mathcal{R}_A\left(\mathcal{P}_{(1:t)}, \widetilde{\mathcal{P}}_{(1:t)}\right) \frac{1}{t}\sum_{i=1}^{|C|} \mathbb{E}_{\tilde{\mathcal{P}}_{a_i}}\{-\mathcal{L}_{ELBO}\left(\mathbf{x}_{a_i}^S; h\right)\} + \frac{1}{t}\sum_{i=1}^{|C'|}\sum_{j=1}^{\tilde{a}_i} \mathbb{E}_{\tilde{\mathcal{P}}_{a(i,j)}}\{-\mathcal{L}_{ELBO}\left(\mathbf{x}_{a(i,j)}^S; h\right)\} \leq
$$

$$
\frac{1}{t}\sum_{i=1}^{|C'|}\sum_{j=1}^{\tilde{a}_i} \mathbb{E}_{\mathbb{P}_{a(i,j)}^{c(i,j)}}\left\{-\mathcal{L}_{ELBO}\left(\mathbf{x}_{a(i,j)}^t; h\right) + \mathcal{R}_A\left(\tilde{\mathcal{P}}_{a(i,j)}, \mathbb{P}_{a(i,j)}^{c(i,j)}\right)\right\}
$$

$$
+ \frac{1}{t}\sum_{i=1}^{|C|} \mathbb{E}_{\tilde{\mathcal{P}}_{a_i}}\{-\mathcal{L}_{ELBO}\left(\mathbf{x}_{a_i}^S; h\right)\} + \frac{1}{t}\sum_{i=1}^{|C'|}\sum_{j=1}^{\tilde{a}_i}\{D_{diff}\left(\mathbf{x}_{a(i,j)}^S, \mathbf{x}_{a(i,j)}^t\right)\}
$$

$$
+ \frac{1}{t}\sum_{i=1}^{t} D_{diff}\left(\mathbf{x}_i^T, \mathbf{x}_i^S\right) + \mathcal{R}_A\left(\mathcal{P}_{(1:t)}, \widetilde{\mathcal{P}}_{(1:t)}\right)
$$

$$(113)$$

It can be noted that LHS of Eq. (113) is equal to RHS of Eq. (112), therefore, RHS of Eq. (113) is also an upper bound to LHS of Eq. (112).

$$
\frac{1}{t}\sum_{i=1}^{t} \mathbb{E}_{\mathcal{P}_i}\{-\mathcal{L}_{ELBO}\left(\mathbf{x}_i^T; h\right) \leq \frac{1}{t}\sum_{i=1}^{|C'|}\sum_{j=1}^{\tilde{a}_i} \mathbb{E}_{\mathbb{P}_{a(i,j)}^{c(i,j)}}\left\{-\mathcal{L}_{ELBO}\left(\mathbf{x}_{a(i,j)}^t; h\right) + \mathcal{R}_A\left(\tilde{\mathcal{P}}_{a(i,j)}, \mathbb{P}_{a(i,j)}^{c(i,j)}\right)\right\}
$$

$$
+ \frac{1}{t}\sum_{i=1}^{|C|} \mathbb{E}_{\tilde{\mathcal{P}}_{a_i}}\{-\mathcal{L}_{ELBO}\left(\mathbf{x}_{a_i}^S; h\right)\} + \frac{1}{t}\sum_{i=1}^{|C'|}\sum_{j=1}^{\tilde{a}_i}\{D_{diff}\left(\mathbf{x}_{a(i,j)}^S, \mathbf{x}_{a(i,j)}^t\right)\} \quad (114)
$$

$$
+ \frac{1}{t}\sum_{i=1}^{t} D_{diff}\left(\mathbf{x}_i^T, \mathbf{x}_i^S\right) + \mathcal{R}_A\left(\mathcal{P}_{(1:t)}, \widetilde{\mathcal{P}}_{(1:t)}\right)
$$

Eventually, we know that LHS of Eq. (114) is an upper bound to NLL, then RHS of Eq. (114) is also an upper bound to NLL.

$$
\frac{1}{t}\sum_{i=1}^{t} \mathbb{E}_{\mathcal{P}_i}\{-\log p\left(\mathbf{x}_i^T\right) \leq \frac{1}{t}\sum_{i=1}^{|C'|}\sum_{j=1}^{\tilde{a}_i} \mathbb{E}_{\mathbb{P}_{a(i,j)}^{c(i,j)}}\left\{-\mathcal{L}_{ELBO}\left(\mathbf{x}_{a(i,j)}^t; h\right) + \mathcal{R}_A\left(\tilde{\mathcal{P}}_{a(i,j)}, \mathbb{P}_{a(i,j)}^{c(i,j)}\right)\right\}
$$

$$
+ \frac{1}{t}\sum_{i=1}^{|C|} \mathbb{E}_{\tilde{\mathcal{P}}_{a_i}}\{-\mathcal{L}_{ELBO}\left(\mathbf{x}_{a_i}^S; h\right)\} + \frac{1}{t}\sum_{i=1}^{|C'|}\sum_{j=1}^{\tilde{a}_i}\{D_{diff}\left(\mathbf{x}_{a(i,j)}^S, \mathbf{x}_{a(i,j)}^t\right)\} \quad (115)
$$

$$
+ \frac{1}{t}\sum_{i=1}^{t} D_{diff}\left(\mathbf{x}_i^T, \mathbf{x}_i^S\right) + \mathcal{R}_A\left(\mathcal{P}_{(1:t)}, \widetilde{\mathcal{P}}_{(1:t)}\right)
$$

This proves Proposition 5.

Proposition 6 *Let $\mathcal{P}_i$ be a target distribution and let $\mathbb{P}^i$ be an approximation distribution that partly overlaps with $\mathcal{P}_i$. Let $\{\theta^*, \omega^*\}$ be the optimal parameters for a VAE model trained on samples $\mathbf{x}'$ drawn from $\mathbb{P}^i$. Then $\mathcal{L}_{ELBO}(\{\theta^*, \omega^*\}, \mathbf{x})$ is not a tight ELBO for $\log p_\theta(\mathbf{x})$ by using the model's parameters $\{\theta^*, \omega^*\}$.*

Proof Let $\{\theta', \omega'\}$ be an optimal parameter set for a VAE model trained on real samples $\mathbf{x}$ drawn from $\mathcal{P}_i$ and we have $\mathcal{L}_{ELBO}(\{\theta', \omega'\}, \mathbf{x}) > \mathcal{L}_{ELBO}(\{\tilde{\theta}, \tilde{\omega}\}, \mathbf{x}), \tilde{\theta} \neq \theta', \tilde{\omega} \neq \omega'$. Since the model's parameters $\{\theta^*, \omega^*\}$ are optimized by maximizing ELBO on samples draw from $\mathbb{P}_i$ and are not equal to $\{\theta', \omega'\}$. Therefore we have $\mathcal{L}_{ELBO}(\{\theta', \omega'\}, \mathbf{x}) > \mathcal{L}_{ELBO}(\{\theta^*, \omega^*\}, \mathbf{x})$ and $\mathcal{L}_{ELBO}(\{\theta^*, \omega^*\}, \mathbf{x})$ is not a tight ELBO to the real sample log-likelihood $\log p_\theta(\mathbf{x})$.

In order to investigate the results for Proposition 6, we train a single model with GR under MNIST, Fashion, IFashion (MIF) lifelong learning (The detailed setting is provided in Appendix L.2). We also train a single model that can access the real training samples from previous tasks during LLL and this model can be seen to be optimal for the real sample log-likelihood. We evaluate the average target risk (estimated by the negative ELBO on all testing samples) for each training epoch and the results are reported in Fig. 1 where "Single" and "Single*" represent the single model with GR and the single model that can access the real training samples from previous tasks, respectively. It notes that "Single"
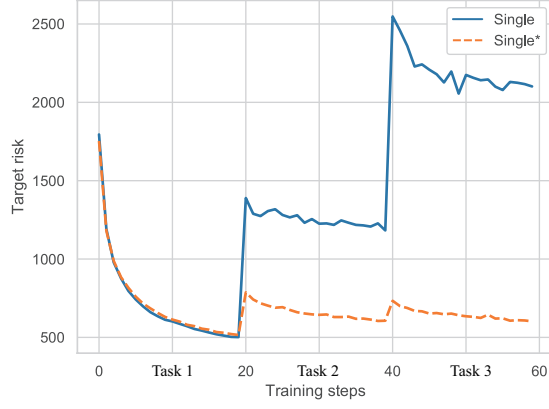
Figure 1: The target risk for a single model under MNIST, Fashion, IFashion lifelong learning.

is trained on the approximation distribution instead of the distributon of real training sets. It shows that, as learning more tasks, a VAE model with GR process gradually loses performance on the real data distribution. This demonstrates that the optimal parameters $\{\theta^*, \omega^*\}$ achieved from the approximation distribution are not optimal for the real sample log-likelihood, which empirically proves Proposition 6.

J The main objective function, algorithm and pipeline for DEGM

In this section, we introduce additional information for the proposed DEGM.

J.1 Derivation of $\mathcal{L}_{MELBO}$

As illustrated in Fig.2 from the paper, where we show the graph structure implementing DEGM, we have K number of basic nodes in DEGM. When building a new specific node for learning a new task, we derive the main objective function showing as follows.

Theorem 4. When a new specific node $((t+1)$-th node) is built for learning the $(t+1)$-th task. This specific node connected with all basic nodes can be seen as a sub-graph model which can be trained by a valid lower bound (ELBO).

Proof. We start by considering the KL divergence Doersch [2016] :

$$KL\left[Q\left(\mathbf{z}\right)||p\left(\mathbf{z}|\mathbf{x}\right)\right] = \mathbb{E}_{z \sim Q(\mathbf{z})}\left[\log Q\left(\mathbf{z}\right) - \log p\left(\mathbf{z}|\mathbf{x}\right)\right] \tag{116}$$

where $Q(\mathbf{z})$ is the variational distribution. We can rewrite the above equation as :

$$KL\left[Q\left(\mathbf{z}\right)||p\left(\mathbf{z}|\mathbf{x}\right)\right] = \mathbb{E}_{z \sim Q(\mathbf{z})}\left[\log Q\left(\mathbf{z}\right) - \log p\left(\mathbf{x}|\mathbf{z}\right) - \log p\left(z\right)\right] + \log p\left(\mathbf{x}\right) \tag{117}$$

And we have :

$$\log p\left(\mathbf{x}\right) - KL\left[Q\left(\mathbf{z}\right)||p\left(\mathbf{z}|\mathbf{x}\right)\right] = \mathbb{E}_{z \sim Q(\mathbf{z})}\left[\log p\left(\mathbf{x}|\mathbf{z}\right)\right] - KL\left[Q\left(\mathbf{z}\right)||p\left(\mathbf{z}\right)\right] \tag{118}$$

where the right hand side is also called evidence lower bound (ELBO). We particularly focus on the KL term $KL\left[Q\left(\mathbf{z}\right)||p\left(\mathbf{z}\right)\right]$ which has the following from :

$$KL\left[Q\left(\mathbf{z}\right)||p\left(\mathbf{z}\right)\right] = \int q\left(\mathbf{z}\right)\log\frac{q\left(\mathbf{z}\right)}{p\left(\mathbf{z}\right)}d\mathbf{z} \tag{119}$$

where $q(\mathbf{z})$ is the density of $Q(\mathbf{x})$. Since we have K components and we consider the $q(\mathbf{z})$ to be mixture density function $q\left(\mathbf{z}\right) = \sum_{i=1}^{w}\pi_i q_{\tilde{\omega}_{\mathcal{SG}(i)}} \circ q_{\omega'(t+1)}\left(\mathbf{z}|\mathbf{x}\right)$ where π_i is the weight. We then rewrite Eq. (119) as :

$$\begin{aligned}
KL\left[Q\left(\mathbf{z}\right)||p\left(\mathbf{z}\right)\right] &= \int \left(\sum_{i=1}^{K}\pi_i q_{\tilde{\omega}_{\mathcal{GI}(i)}} \circ q_{\omega'(t+1)}\left(\mathbf{z}|\mathbf{x}\right)\right)\log\frac{q\left(\mathbf{z}\right)}{p\left(\mathbf{z}\right)}d\mathbf{z} \\
&= \sum_{i=1}^{K}\pi_i \int q_{\tilde{\omega}_{\mathcal{GI}(i)}} \circ q_{\omega'(t+1)}\left(\mathbf{z}|\mathbf{x}\right)\log\frac{q\left(\mathbf{z}\right)}{p\left(\mathbf{z}\right)}d\mathbf{z}
\end{aligned} \tag{120}$$

We then add the term $q_{\tilde{\omega}_{\mathcal{SG}(i)}} \circ q_{\omega'(t+1)}\left(\mathbf{z}|\mathbf{x}\right)/q_{\tilde{\omega}_{\mathcal{SG}(i)}} \circ q_{\omega'(t+1)}\left(\mathbf{z}|\mathbf{x}\right)$ to Eq. (120), resulting in :

$$KL\left[Q\left(\mathbf{z}\right)||p\left(\mathbf{z}\right)\right] = \sum_{i=1}^{K}\pi_i \int q_{\tilde{\omega}_{\mathcal{GI}(i)}} \circ q_{\omega'(t+1)}\left(\mathbf{z}|\mathbf{x}\right)\log\{\frac{q\left(\mathbf{z}\right)}{p\left(\mathbf{z}\right)} \times \frac{q_{\tilde{\omega}_{\mathcal{GI}(i)}} \circ q_{\omega'(t+1)}\left(\mathbf{z}|\mathbf{x}\right)}{q_{\tilde{\omega}_{\mathcal{GI}(i)}} \circ q_{\omega'(t+1)}\left(\mathbf{z}|\mathbf{x}\right)}\}d\mathbf{z} \tag{121}$$

We rewrite the above equation as :

$$\begin{aligned}
&\sum_{i=1}^{K}\pi_i \int q_{\tilde{\omega}_{\mathcal{GI}(i)}} \circ q_{\omega'(t+1)}\left(\mathbf{z}|\mathbf{x}\right) \times \log\{\frac{q_{\tilde{\omega}_{\mathcal{GI}(i)}} \circ q_{\omega'(t+1)}\left(\mathbf{z}|\mathbf{x}\right)}{p\left(\mathbf{z}\right)} \times \frac{q\left(\mathbf{z}\right)}{q_{\tilde{\omega}_{\mathcal{GI}(i)}} \circ q_{\omega'(t+1)}\left(\mathbf{z}|\mathbf{x}\right)}\}d\mathbf{z} = \\
&\sum_{i=1}^{K}\pi_i \int q_{\tilde{\omega}_{\mathcal{GI}(i)}} \circ q_{\omega'(t+1)}\left(\mathbf{z}|\mathbf{x}\right) \times \left(\log\frac{q_{\tilde{\omega}_{\mathcal{GI}(i)}} \circ q_{\omega'(t+1)}\left(\mathbf{z}|\mathbf{x}\right)}{p\left(\mathbf{z}\right)} + \log\frac{q\left(\mathbf{z}\right)}{q_{\tilde{\omega}_{\mathcal{GI}(i)}} \circ q_{\omega'(t+1)}\left(\mathbf{z}|\mathbf{x}\right)}\right)d\mathbf{z}
\end{aligned} \tag{122}$$

Then we can rewrite the above equation as KL terms :

$$\begin{aligned}
KL\left[Q\left(\mathbf{z}\right)||p\left(\mathbf{z}|\mathbf{x}\right)\right] &= \sum_{i=1}^{K}\left(\pi_i KL\left[Q_{\tilde{\omega}_{\mathcal{GI}(i)} \circ \omega'(t+1)}\left(\mathbf{z}|\mathbf{x}\right)||p\left(\mathbf{z}\right)\right]\right) \\
&\quad - \sum_{i=1}^{K}\left(\pi_i KL\left[Q_{\tilde{\omega}_{\mathcal{GI}(i)} \circ \omega'(t+1)}\left(\mathbf{z}|\mathbf{x}\right)||p\left(\mathbf{z}\right)\right]\right)
\end{aligned} \tag{123}$$

where $q_{\tilde{\omega}_{\mathcal{GI}(i)}} \circ q_{\omega'(t+1)}\left(\mathbf{z}|\mathbf{x}\right)$ is the density form of $Q_{\tilde{\omega}_{\mathcal{GI}(i)} \circ \omega'(t+1)}\left(\mathbf{z}|\mathbf{x}\right)$. We take the expression of $KL\left[Q\left(\mathbf{z}\right)||p\left(\mathbf{z}|\mathbf{x}\right)\right]$ to Eq. (118), resulting in :

$$\begin{aligned}
\log p\left(\mathbf{x}\right) - KL\left[Q\left(\mathbf{z}\right)||p\left(\mathbf{z}|\mathbf{x}\right)\right] &= \mathbb{E}_{z \sim Q(\mathbf{z})}\left[\log p\left(\mathbf{x}|\mathbf{z}\right)\right] - \sum_{i=1}^{K}\left(\pi_i KL\left[Q_{\tilde{\omega}_{\mathcal{GI}(i)} \circ \omega'(t+1)}\left(\mathbf{z}|\mathbf{x}\right)||p(\mathbf{z})\right]\right) \\
&\quad + \sum_{i=1}^{K}\left(\pi_i KL\left[Q_{\tilde{\omega}_{\mathcal{GI}(i)} \circ \omega'(t+1)}\left(\mathbf{z}|\mathbf{x}\right)||Q\left(\mathbf{z}\right)\right]\right)
\end{aligned} \tag{124}$$

where we move the last term of the right hand side to the left hand side, resulting in :

$$\log p\left(\mathbf{x}\right) - KL\left[Q\left(\mathbf{z}\right) || p\left(\mathbf{z}|\mathbf{x}\right)\right] - \sum\nolimits_{i=1}^{K}\left(\pi_i KL\left[Q_{\tilde{\omega}_{\mathcal{GI}(i)}\circ\omega'_{(t+1)}}\left(\mathbf{z}|\mathbf{x}\right)||Q\left(\mathbf{z}\right)\right]\right) =$$
$$\mathbb{E}_{z\sim Q(\mathbf{z})}\left[\log p\left(\mathbf{x}|\mathbf{z}\right)\right] - \sum\nolimits_{i=1}^{K}\left(\pi_i KL\left[Q_{\tilde{\omega}_{\mathcal{GI}(i)}\circ\omega'_{(t+1)}}\left(\mathbf{z}|\mathbf{x}\right)||p\left(\mathbf{z}\right)\right]\right) \tag{125}$$

We know that KL terms are always larger or equal to zero and the right hand side of Eq. (125) is a lower bound to the sample log-likelihood. Finally, the objective function for training DEGM is to maximize this lower bound :

$$\mathbb{E}_{z\sim Q(\mathbf{z})}\left[\log p\left(\mathbf{x}|\mathbf{z}\right)\right] - \sum\nolimits_{i=1}^{K}\left(\pi_i KL\left[Q_{\tilde{\omega}_{\mathcal{GI}(i)}\circ\omega'_{(t+1)}}\left(\mathbf{z}|\mathbf{x}\right)||p\left(\mathbf{z}\right)\right]\right) \tag{126}$$

where in the first term, $\mathbf{z}$ is sampled from the mixture distribution $Q\left(\mathbf{z}\right) = \sum_{i=1}^{K}\pi_i Q_{\tilde{\omega}_{\mathcal{GI}(i)}\circ\omega'_{(t+1)}}\left(\mathbf{z}|\mathbf{x}\right)$. In order to reuse the parameters and transferable information from all basic nodes, we consider the following implementations for the decoder. When calculating the first term, we build an input layer on the top of the decoder. This input layer is used as the identity function and is connected with each sub-decoder $g_{\tilde{\theta}_{\mathcal{GI}(i)}}(\mathbf{z})$ of each basic node, represented as one layer or module in a single decoder. Then we obtain the intermediate data representation $\tilde{\mathbf{x}} = \sum_{i=1}^{K}\pi_i g_{\tilde{\theta}_{\mathcal{GI}(i)}}(\mathbf{z})$ which is used as the input for the newly created sub-decoder $g_{\theta'_{(t+1)}}(\tilde{\mathbf{x}})$. Therefore, we treat the intermediate data representation $\tilde{\mathbf{x}}$ as the information between two layers in a single decoder and do not consider $\tilde{\mathbf{x}}$ to be the random variable. In this case we rewrite the decoding distribution $p(\mathbf{x}|\mathbf{z})$ as :

$$p(\mathbf{x}|\mathbf{z}) = p_{\theta'_{(t+1)}\circ\{\tilde{\theta}_{\mathcal{GI}(1)},...,\tilde{\theta}_{\mathcal{GI}(K)}\}}\left(\mathbf{x}\mid\mathbf{z}\right)$$

. Therefore, we rewrite Eq. (126) as :

$$\mathcal{L}_{MELBO}\left(\mathbf{x};\mathcal{M}_{(t+1)}\right) = \mathbb{E}_{Q(\mathbf{z})}\left[\log p_{\theta'_{(t+1)}\circ\{\tilde{\theta}_{\mathcal{GI}(1)},...,\tilde{\theta}_{\mathcal{GI}(K)}\}}\left(\mathbf{x}\mid\mathbf{z}\right)\right]$$
$$- \sum_{i=1}^{K}\left(\pi_i KL\left[q_{\tilde{\omega}_{\mathcal{GI}(i)}}\circ q_{\omega'_{(t+1)}}\left(\mathbf{z}\mid\mathbf{x}\right)||p\left(\mathbf{z}_i\right)\right]\right) \tag{127}$$

Eq.(127) shows that we can implement existing variational inference mechanisms such as using a more expressive posterior Maaløe et al. [2016], Kingma et al. [2016], Sobolev and Vetrov [2019], important sampling Burda et al. [2015] and Semi-Implicit Variational Inference Yin and Zhou [2018], Molchanov et al. [2019] in our framework to enable for lifelong learning.

J.2 The pipeline and algorithm of DEGM

In order to conveniently understand the learning procedure of DEGM, we show the processing pipeline in Figure 2. Once the t-th task was finished, we get a group of samples, denoted as $\tilde{Q}_{(t+1)}^{S}$ from $\mathcal{T}_{t+1}$. Then we calculate the edge values, represented by $\{\pi_1,\ldots,\pi_K\}$. If $\min(\mathcal{K}) > \tau$, then we build a basic node which will be added into $\mathcal{G}$ for learning the $(t+1)$-th task, otherwise, we build a specific node based on members of $\mathcal{G}$ for learning the $(t+1)$-th task. From this pipeline, we observe that the edge values regularize the latent representations and the intermediate data representations in the inference and generation process, respectively. We also provide the pseudocode in Algorithm 1.

J.3 The component selection for model evaluation during the testing phase

Let us consider DEGM model with t trained nodes after LLL. We introduce the cluster assignment $\mathbf{u}$ in the DEGM and the probability density of DEGM on n samples is represented by :

$$p\left(\mathbf{x}\right) = \prod\nolimits_{i=1}^{n}\sum\nolimits_{j=1}^{t} p\left(\mathbf{x}_i\mid\mathbf{u}_{(i,j)}\right) p\left(\mathbf{u}_{(i,j)}\right),\ \mathbf{u}_{(i,j)}\in\{0,1\}. \tag{128}$$

We particularly focus on the posterior $p(\mathbf{u}_{(i,j)}|\mathbf{x}_i)$ which can be rewritten by the Bayes' theorem :

$$p\left(\mathbf{u}_{(i,j)}\mid\mathbf{x}_i\right) = \frac{p\left(\mathbf{x}_i\mid\mathbf{u}_{(i,j)}\right) p\left(\mathbf{u}_{(i,j)}\right)}{p\left(\mathbf{x}_i\right)} = \frac{p\left(\mathbf{x}_i\mid\mathbf{u}_{(i,j)}\right) p\left(\mathbf{u}_{(i,j)}\right)}{\sum_{k=1}^{t} p\left(\mathbf{u}_{(i,k)}\right) p\left(\mathbf{x}_i\mid\mathbf{u}_{(i,k)}\right)}, \tag{129}$$

where the prior is $p(\mathbf{u}_{(i,j)}) = 1/t$. Since $\log_a$ is a function monotonic increasing if $a > 1$, we can replace each $p(\mathbf{x}_i\mid\mathbf{u}_{(i,j)})$ by $\log p(\mathbf{x}_i\mid\mathbf{u}_{(i,j)})$ estimated by $\mathcal{L}_{ELBO}(,\mathbf{x}_i;\mathcal{M}_j)$ for the elements of $\mathcal{G}$ on the sample $\mathbf{x}$, and by $\mathcal{L}_{MELBO}(\mathbf{x}_i;\mathcal{M}_j)$

Algorithm 1: The training algorithm for DEGM

Input: All training databases
Output: The model's parameters

1 **for** $i < taskCount$ **do**
2   **if** $i == 1$ **then**
3   Build a basic component B_1 and added to $\mathcal{G}$;
4         $isBasic = True$;
5   **end**
6   **for** $index < batchCount$ **do**
7         $\mathbf{x}_{batch} \sim Q_i^S$ a batch of images sampled from the i-th task;
8   **if** $isBasic == True$ **then**
9   Update $\{\omega_i, \theta_i\}$ by maximizing $\mathcal{L}_{ELBO}(\mathbf{x}_{batch}; \mathcal{M}_i)$;
10   **end**
11   **else**
12   Update $\{\omega_i', \theta_i'\}$ by maximizing $\mathcal{L}_{MELBO}(\mathbf{x}_{batch}; \mathcal{M}_i)$;
13   **end**
14   **end**
15   **Expansion mechanism**;
16     $\mathbf{x}_{new} \sim Q_{i+1}^S$ collect 1000 number of samples from the next task;
17   **Calculate the importance of each basic node to the new task**;
18   **for** $k < K$ **do**
19         $ks_k = |\mathcal{L}_{ELBO}(B_k) - \mathbb{E}_{\mathbf{x} \sim \mathbf{x}_{new}} \mathcal{L}_{ELBO}(\mathbf{x}; B_k)|$;
20   **end**
21     $\mathcal{K} = \{ks_1, \ldots, ks_K\}$;
22   **if** $\min\{\mathcal{K}\} \leq \tau$ **then**
23   **The construction of the basic node**;
24         $V(i + 1, \mathcal{GI}(j)) = 0, j = 1, \ldots, K$;
25   Build a basic component $B_{(K+1)}$ and added to $\mathcal{G}$;
26         $isBasic = True$;
27         $K = K + 1$;
28   **end**
29   **else**
30   **The construction of the sub-graph structure**;
31         $V(i + 1, \mathcal{GI}(i)) = (w^* - ks_i)/\sum_{j=1}^{K}(w^* - ks_j), w^* = \sum_{j=1}^{K} ks_j, i = 1, \ldots, K$;
32         $\sum_{j=1}^{K} \pi_i f_{\tilde{\omega}_{\mathcal{GI}(j)}} \circ f_{\omega'_{(i+1)}}(\mathbf{x})$ build the inference model ;
33         $\mathbf{z} = \sum_{j=1}^{K} \pi_j \mathbf{z}_j$ obtain the latent representation;
34         $\tilde{\mathbf{x}} = \sum_{j=1}^{K} \pi_j g_{\tilde{\theta}_{\mathcal{GI}(j)}}(\mathbf{z})$ obtain the intermediate representations from sub-decoder of all basic nodes;
35         $g_{\theta'_{(i+1)}}(\tilde{\mathbf{x}})$ build a sub-decoder ;
36   Add $\mathcal{M}_{i+1}$ in $\mathcal{S}$;
37         $isBasic = False$;
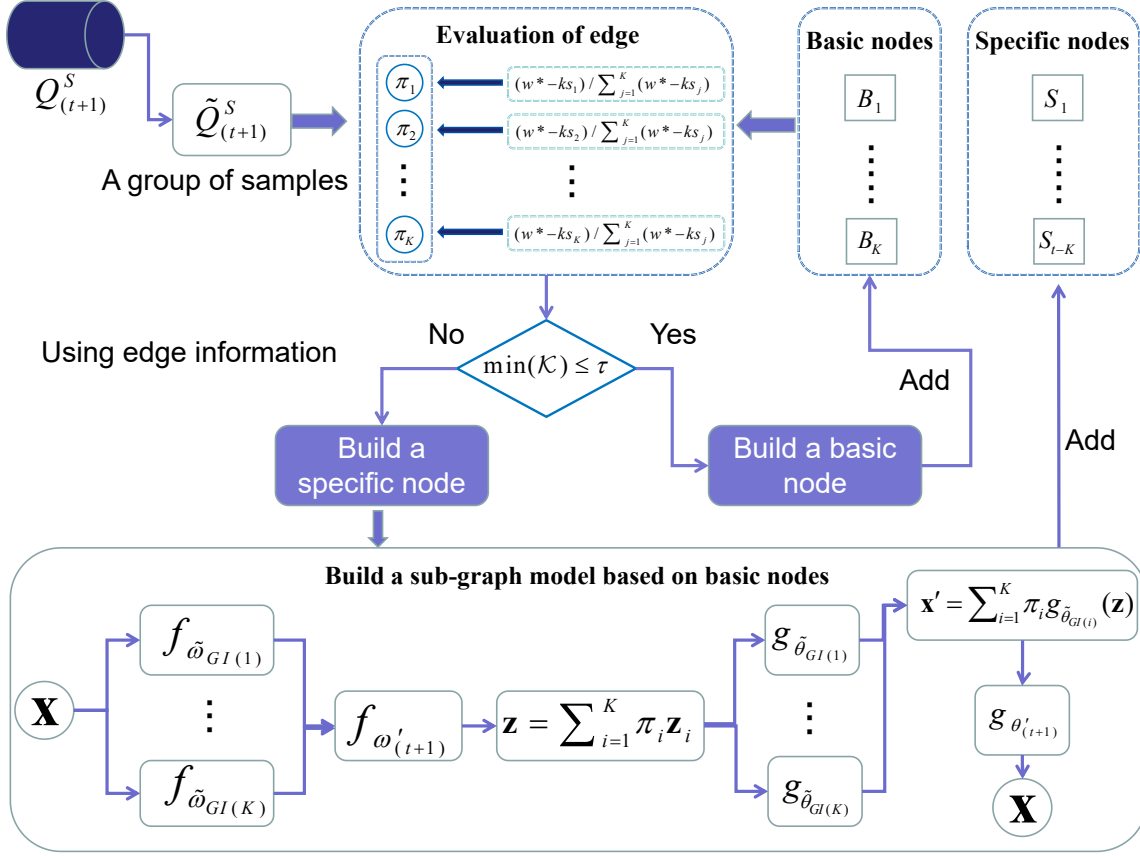38   **end**
39 **end**

Figure 2: The processing pipeline of the proposed DEGM.

from (127) for elements of $\mathcal{S}$, as described in Section 5.3 of the paper. This selection process can allow DEGM to infer a related component without having task labels.

J.4 Limitations of the proposed theoretical framework

The primary limitation of the proposed theoretical framework is that it only supports the Gaussian decoder since ELBO can be decomposed by a negative reconstruction error term ($\mathcal{L}$ loss) and the KL divergence term. Our future work will focus on extending the proposed theoretical framework for other types of distributions for the decoder such as Bernoulli distribution.

One potential negative societal impacts of our work is that the proposed DEGM would cause data leakage. For example, in many lifelong learning tasks, one potential requirement is to allow the data for each task to appear once for privacy protection.

J.5 The difference between related work and DEGM

In this section, we discuss the difference between several related works and DEGM. Firstly, our work is related to Rao et al. [2019] that expands its network architectures on the inference models only. Rao et al. [2019] also uses the generative replay mechanism, which would lead to the degenerated performance on previously learnt samples due to the network forgetting. This is not the case in DEGM since we do not use the generative replay mechanism. To compare with CN-DPM Lee et al. [2020] that also uses the expansion mechanism only, DEGM can reuse prior knowledge when learning novel samples by forming a graph structure in which the independent basic nodes are used to model entirely different tasks while the specific nodes are built based on the information flow from basic nodes for learning related tasks. However, in both studies Lee et al. [2020] and Rao et al. [2019] perform task-free continual learning, which is not the main task in our work since we address a more general lifelong learning problem where the task boundaries are provided only at the training stage.

Our work is more related to LIMix Ye and Bors [2021] that introduces a mixture model with an infinite number of components. LIMix has only a single shared encoder and decoder, which has limitations when learning several entirely different tasks since the shared parameters preserve the knowledge from the first task only. DEGM can have several

basic nodes which aim to learn several entirely different tasks and can be used for transferring knowledge when learning novel tasks. Additionally, LIMix introduces a new theoretical analysis for the forgetting behaviour of the model under lifelong learning, which, however, is only applied in the supervised learning task. The proposed theoretical framework in our work studies the forgetting behaviour of the model under the unsupervised lifelong learning setting.

K   Unsupervised lifelong learning benchmark

K.1   Network architecture and hyperaprameter setting

For the general network architecture for VAEs, the inference and generator models are implemented by two fully connected networks where each network contains two layers and each layer has 200 hidden units. We also extend VAEs by using two stochastic layers where the latent dimension for the first and second stochastic layer is 100 and 50, respectively. The VAE with two stochastic layers is named as ELBO-GR*. If VAE uses only a single stochastic layer, then the latent dimension is 100. When the important sampling is used in various models such as VAEs, we call it as IWELBO-GR-K' where K' represents the number of weighted samples used in the objective function. The GPU used for the experiment is GeForce GTX 1080. The operating system is Ubuntu 18.04.5.

For the DEGM model, the inference model and the generator in DEGM are implemented by four sub-models and each sub-model has only one layer with 200 units. Therefore, the inference and generator model in each specific node are two sub-models built based on all basic components members. As similar to a single model, DEGM can be extended with the important sampling framework. For implementation, if a basic component is used for training, we can easily use the IWELBO objective function (See Eq. (54) in Appendix F.1). If a specific node is used for training, we rewrite the objective function as the IWELBO bound form as :

$$\mathcal{L}_{MELBO_{K'}}\left(\mathbf{x}; \mathcal{M}\right) = \mathbb{E}_{\mathbf{z}_1,..,\mathbf{z}_{K'} \sim Q(\mathbf{z})} \left[\log \frac{1}{K'} \sum_{i=1}^{K'} \frac{p\left(\mathbf{x}, \mathbf{z}_i\right)}{Q\left(\mathbf{z}_i\right)} \right] \tag{130}$$

The above equation is extended by the IWELBO bound with Eq. (118).

CN-DPM is originally designed for the task-free task. In order allow CN-DPM in a more general lifelong unsupervised learning task, we implement a variant of CN-DPM, namely CN-DPM*. We replace the Dirichlet process in CN-DPM* by dynamically creating a new component after the task switch since the Dirichlet process can allow CN-DPM to be a large model. As similar to CN-DPM, CN-DPM* has a single shared module which is frozen after the first task learning and a new component will be built based on this shared module. In order to set the optimal setting for LIMix Ye and Bors [2021], the number of components in LMix matches the number of tasks while each component models a certain task only.

We train various models using Adam Kingma and Ba [2015] optimization algorithm with the learning rate of 0.0001. It notes that the decoder models Bernoulli distribution and we use the binary cross entropy as the reconstruction error term in ELBO. The batch size and the number of training epochs for each task learning are 64 and 500, respectively.

K.2   Results on cross-domain setting (COFMI)

We search the threshold τ of 35-40 for Split MNIST/Fashion and 80-100 for COFMI, respectively. DEGM based methods has four basic nodes and one specific node after Split MNIST and Split Fashion lifelong learning setting.

We report the results under COFMI in Table 1. The weight matrix $\mathcal{V}$ of DEGM-IWELBO-50 under COFMI lifelong learning is shown in Fig. 3. DEGM-IWELBO-50, DEGM-IWELBO-5, DEGM-ELBO has 4 basic and 1 specific nodes after COFMI lifelong learning.

K.3   Results on Split MNIST and Split Fashion

In this section, we perform five independent runs for Split MNIST and Split Fashion. After the training, we calculate the average result as well as the standard deviation (std). The results for Split MNIST and Split Fashion are reported in Table 2 and Table 3, respectively.
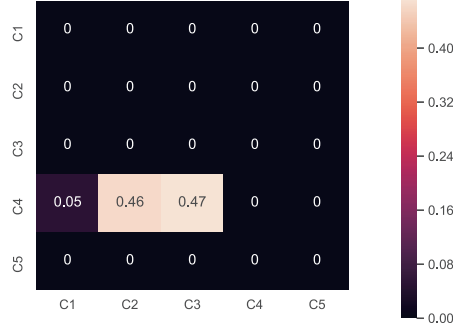
Figure 3: **V** of DEGM-IWVAE-50 under COFMI lifelong learning. "C1" represents the first component and "C4" is a basic node that connects previous three basic nodes ("C1", "C2", "C3").

Mwthods	Caltech 101	OMNIGLOT	Fashion	MNIST	IFashion	Average
ELBO-GR	-163.68	-136.97	-247,91	-101.75	-237,03	-177,47
IWELBO-GR-50	-153.65	-131.37	-243.62	-97,29	-234.58	-172.10
IWELBO-GR-5	-166.05	-134.07	-245.78	-99.43	-235.73	-176.21
ELBO-GR*	-175.10	-140.05	-247.54	-102.73	-237.06	-180.50
IWELBO-GR*-50	-215.16	-144.42	-246.35	-102.82	-236.12	-188.97
CN-DPM*-IWELBO-50	-136.22	-150.31	-259.12	-131.34	-243.97	-184.19
LIMix-IWELBO-50	-137.32	-150.79	258.69	-131.25	-243.50	-184.32
DEGM-ELBO	-137.72	-116.07	-233.32	-122.33	-234.62	-168.81
DEGM-IWELBO-50	-133.93	-112.15	-230.63	-107.44	-232.17	**-163.27**
DEGM-IWELBO-5	-137.18	-113.25	-231.88	-109.21	-233.43	-164.99

Table 1: The estimation of the sample log-likelihood under COFMI lifelong learning.

Mwthods	Run 1	Run 2	Run 3	Run 4	Run 5	Average
ELBO-GR	-97.97	-98.16	-98.70	-97.91	-98.39	-98.23 $\pm$ (0.28 std)
IWELBO-GR-50	-93.61	-93.69	-93.66	-93.64	-93.27	-93.57 $\pm$ (0.15 std)
IWELBO-GR-5	-95.96	-95.68	-95.95	-95.68	-95.73	-95.80 $\pm$ (0.12 std)
ELBO-GR*	-97.92	-98.79	-97.55	-98.45	-99.11	-98.36 $\pm$ (0.56 std)
IWELBO-GR*-50	-91.15	-91.33	-91.24	-91.23	-91.18	-91.23 $\pm$ (0.06 std)
CN-DPM*-IWELBO-50	-95.94	-96.35	-95.90	-95.74	-95.63	-95.91 $\pm$ (0.24 std)
LIMix-IWELBO-50	-95.57	-95.80	-95.84	-95.84	-95.63	-95.74 $\pm$ (0.11 std)
DEGM-ELBO	-93.85	-87.03	-93.15	-93.71	-93.51	-92.25 $\pm$ (2.62 std)
DEGM-IWELBO-50	-89.41	-89.58	-89.60	-85.78	-85.83	**-88.04** $\pm$ (1.82 std)
DEGM-IWELBO-5	-91.54	-91.70	-91.42	-91.21	-91.32	-91.44 $\pm$ (0.17 std)

Table 2: Five independent runs for the estimation of the sample log-likelihood under Split MNIST setting.

Mwthods	Run 1	Run 2	Run 3	Run 4	Run 5	Average
ELBO-GR	-241.00	-240.38	-240.26	-240.82	-240.44	-240.58 $\pm$ (0.28 std)
IWELBO-GR-50	-236.48	-236.65	-236.98	-236.70	-236.49	-236.66 $\pm$ (0.18 std)
IWELBO-GR-5	-237.86	-238.19	-238.31	-238.00	-238.00	-238.08 $\pm$ (0.15 std)
ELBO-GR*	-242.54	-242.27	-244.76	-242.38	-247.61	-243.91 $\pm$ (2.06 std)
IWELBO-GR*-50	-236.71	-236.77	-237.04	-236.88	-237.08	-236.90 $\pm$ (0.14 std)
CN-DPM*-IWELBO-50	-237.60	-237.13	-237.81	-237.38	-237.43	-237.47$\pm$ (0.22 std)
LIMix	-237.28	-237.50	-237.39	-237.37	-237.87	-237.48$\pm$ (0.20 std)
DEGM-ELBO	-240.59	-237.68	-239.45	-237.60	-237.39	-238.54 $\pm$ (1.26 std)
DEGM-IWELBO-50	-234.17	-232.72	-234.13	-233.15	-234.63	**-233.76** $\pm$ (0.70 std)
DEGM-IWELBO-5	-236.51	-234.46	-236.59	235.00	-237.08	-235.93 $\pm$ (1.00 std)

Table 3: Five independent runs for the estimation of the sample log-likelihood under Split Fashion setting.

L Additional experiment results

L.1 Quantitative evaluation

Baselines: We introduce several baselines and one baseline is the proposed mixture model but we use few epochs (five epochs) for training when the new node is built on other basic nodes, namely DEGM-1. The second baseline is 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 task.

- **Hyperparameter setting and GPUs** The SGD optimization algorithm for each model is using the Adam Kingma and Ba [2015], with a learning rate of 0.0002 while the other hyperparameters are set to their default. The GPU used for the experiment is GeForce GTX 1080.

Setting. To measure the reconstruction quality, we calculate Squared Loss (SL) as the criterion. We train various models under MNIST LeCun et al. [1998], SVHN Netzer et al. [2011], Fashion Xiao et al. [2017], InverseFashion (IFashion), Rated MNIST (RMNIST) and Cifar10 Krizhevsky and Hinton [2009], (MSFIRC) lifelong learning setting. We also evaluate the performance of various models under CelebA Liu et al. [2015], CACD Chen et al. [2014], 3D-Chair Aubry et al. [2014], Ommiglot Lake et al. [2015], ImageNet* Krizhevsky et al. [2012], Car Yang et al. [2015], Zappos Yu and Grauman [2017], CUB Wah et al. [2010] (CCCOSCZC) lifelong learning setting.

In the following, we provide details about the setting in the experiments. For lifelong learning setting of MSFIRC and CCCOSCZC, we train model for each task for 20 training epochs. The SGD optimization algorithm for each model is using the Adam Kingma and Ba [2015] where the learning rate is 0.0002 and other hyperparameters are default. In the following, we provide the details for datasets. For CelebA, CACD, we randomly choose 10000 samples as the testing set and the remaining samples are used as the training set. For 3D-chair, we randomly choose 1000 samples as the testing set and the remaining samples as the training set. For ImageNet, we randomly choose 10000 and 50000 samples as the testing set and the training set, respectively. For CUB, we randomly choose 1000 samples as the testing set and the remaining samples as the training set.

We consider the threshold of 400 and 600 for DEGM on the MSFIRC and CCCOSCZC, respectively. We report the results in Table 4 and Table 5, respectively. The interference can be observed by BE and CN-DPM* which gives the degenerated performance on the last task since Cifar10 is different from the first task. In contrast, DEGM avoids interference by building a basic component to learn Cifar10 database.
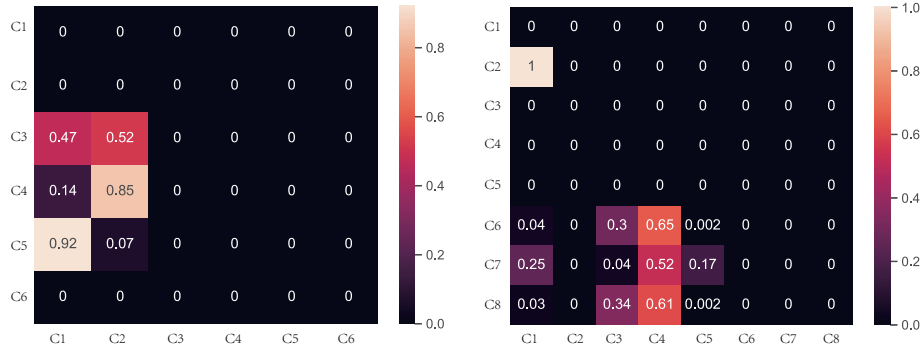
(a) $\mathbf{V}$ of DEGM after the MSFIRC lifelong learning. (b) $\mathbf{V}$ of DEGM after the CCCOSCZC lifelong learning.

Figure 4: The graph adjacency matrix $\mathbf{V}$, characterizing the connections between different nodes of DEGM after lifelong learning.

We show the adjacency matrix of the nodes from the graph $\mathbf{V}$ of DEGM after MSFIRC lifelong learning in Fig. 4a where "C1" represents the first node. We can observe that DEGM creates three basic nodes and three specific nodes, respectively. Three specific nodes have edges from the first and second basic nodes. We also show $\mathbf{V}$ of DEGM after CCCOSCZC lifelong learning in Fig. 4b. DEGM creates basic nodes when learning the first, third, fourth and fifth task. We also show the edge information between members of $\mathcal{S}$ and members of $\mathcal{G}$ in Fig. 5 where red colour represents the basic nodes and blue colour represents the specific nodes.
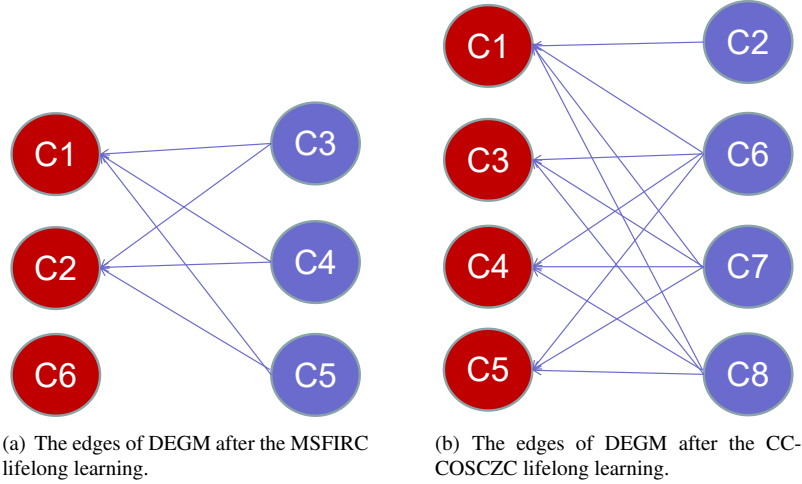
(a) The edges of DEGM after the MSFIRC lifelong learning.

(b) The edges of DEGM after the CC-COSCZC lifelong learning.

Figure 5: Edge information of DEGM after lifelong learning. The red and blue represent the basic and specific node, respectively.

Criteria	Dataset	BE	LGM	DEGM	DEGM-2	CN-DPM*
	MNIST	26.3	685.3	22.3	22.3	21.9
	SVHN	47.0	941.7	30.1	29.0	39.3
	Fashion	43.8	663.4	37.7	27.4	36.6
SL	IFashion	45.9	1148.4	35.6	27.4	38.4
	RMNIST	27.9	704.2	20.2	22.1	25.3
	Cifar10	994.4	1241.1	615.3	608.1	892.1
	Average	197.5	897.4	126.9	122.7	175.6

Table 4: The results under MSFIRC lifelong learning.

L.2   Results for generalization bounds

To estimate the discrepancy, we train an auxiliary model on the distribution $\mathbb{P}^i \otimes \tilde{\mathcal{P}}_{(i+1)}$ for each $(i+1)$-th task learning. Then we calculate the discrepancy by using Definition 3 of the paper for each training epoch.

In the following, we train a single model with GR under MNIST, Fashion and IFashion (MFI) lifelong learning. The images from all databases have pixel values within the range $[0, 255]$. We implement the decoder by a neural network that outputs the mean vector of a Gaussian distribution with the diagonal covariance matrix (diagonal element is 1.0). We estimate the reconstruction error term by using :

$$\log p_\theta \left(\mathbf{x} \,|\, \mathbf{z} \right) = -\frac{1}{2\sigma^2} \|\mathbf{x} - \mu_\theta \left(\mathbf{z} \right)\|^2 - \frac{1}{2} \log 2\pi\sigma^2 \tag{131}$$

It can be noted that we also normalize this reconstruction error by dividing 28×28, as done in Chen et al. [2016]. We evaluate the average risk and the discrepancy distance for each training epoch (See details in Lemma 1 from the paper). To calculate $|KL_1 - KL_2|$ and discrepancy distance, we randomly choose 10000 number of samples from $\mathcal{P}_{(1:t)}$ and $\mathbb{P}^{t-1} \otimes \tilde{\mathcal{P}}_t$ at the t-th task learning, respectively. The results are presented in Fig. 6a where the source risk is continuously decreased while the discrepancy is increased as learning more tasks. We also evaluate the risk, $|KL_1 - KL_2|$ and discrepance distance on MNIST, shown in Fig. 6b. In order to select the generated images that are belonging to MNIST, we train a task-specific classifier that predicts the task label for giving samples.

In the following, we investigate the results for Lemma 2 of the paper. We consider a sequence of MNIST, Fashion, IFashion (MFI) learning tasks. We consider a mixture model $\mathbf{M} = \{\mathcal{M}_1, \mathcal{M}_2\}$ consisting of two components after LLL in which $\mathcal{M}_1$ is fixed after the first task learning while $\mathcal{M}_2$ is used to learn Fashion and IFashion, respectively. We also consider to learn a single model $\mathcal{M}$ with GR under MFI lifelong learning. We evaluate the average target risk (NLL estimated by ELBO) for each training epoch and the results are reported in Fig. (7) where "single" and "mixture" represent $\mathcal{M}$ and $\mathbf{M}$, respectively. As shown from the results, the expansion mechanism can achieve a tight GB, as discussed in Lemma 2 of the paper.

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

Table 5: The results under CCCOSCZC lifelong learning.

(a) Evaluation on all tasks.
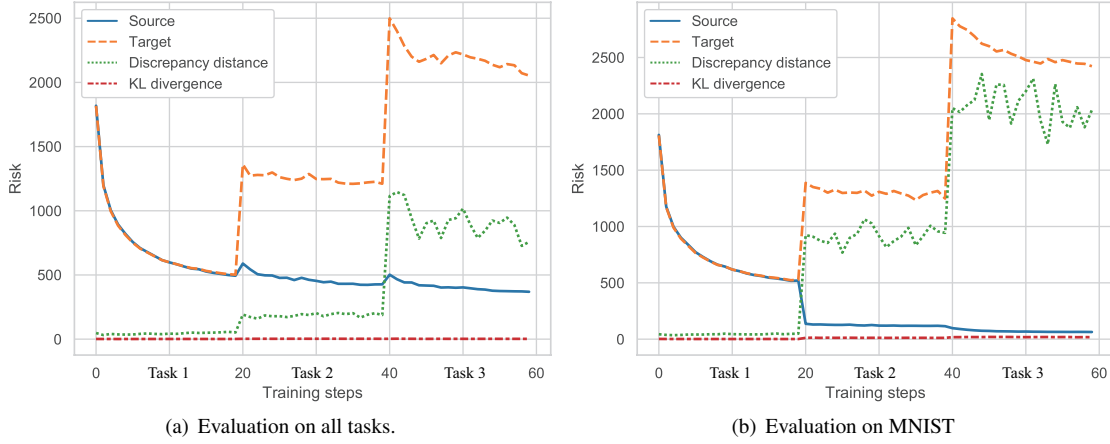
(b) Evaluation on MNIST

Figure 6: The risk, $|KL_1 - KL_2|$ and discrepancy distance estimated by a single VAE model with GR under MFI lifelong learning.

In the following, we provide additional empirical results for the theoretical analysis. It notes that in the following experiments, we resize all databases as $32 \times 32 \times 3$ and the pixel values of all images are normalized as $[0, 1]$.

We evaluate the target risk (square loss) of a single model on four datasets (MNIST, SVHN, Fashion, IFashion) under MSFI (See Theorem-2). We plot the results in Fig. 8-a where "All" represents the accumulated target risks for all tasks (left hand side of Equation-7 in the paper). In the next, we train a single model under MSFI and SFMI setting, respectively. We plot the target risk on MNIST in Fig. 8-b where "Early" and "Recent" denote that the MNIST is used as the first and third task, respectively. It observes that the model tends to forget early tasks than recent tasks, as demonstrated in Equation-14 of the paper and discussed in Theorem 3. We also evaluate the accumulated target risk estimated by DEGM under MSFI lifelong learning, which is shown in Fig. 8-c. It observes that there have no accumulated errors for each task during lifelong learning since the number of components match the number of task, as discussed in Theorem 3.

In the following, we investigate the performance of DEGM and a single model when changing the order of tasks. We randomly generate three different orders : Fashion, SVHN, MNIST, Cifar10, IMNIST, IFashion (FSMCII); Cifar10, IMNIST, Fashion, MNIST, SVHN, IFashion (CIFMSI); IFashion, IMNIST, Cifar10, Fashion, SVHN, MNIST (IICFSM). We evaluate the accumulated target risk for all tasks and the results are presented in Fig. 9 where "order1", "order2" and "order3" denote FSMCII, CIFMSI and IICFSM, respectively. It observes that a single model is sensitive to the choice of orders of tasks since the final accumulated target risk for all tasks are different when training the model under different orders of tasks. Although CN-DPM can avoid the forgetting during the training, the performance is still changed when changing the order of tasks. This is mainly because newly created components only reuse the transferable information from the first task and would lead to negative transfer when learning an entire different task. However, the proposed DEGM is robust to the change of orders of tasks.
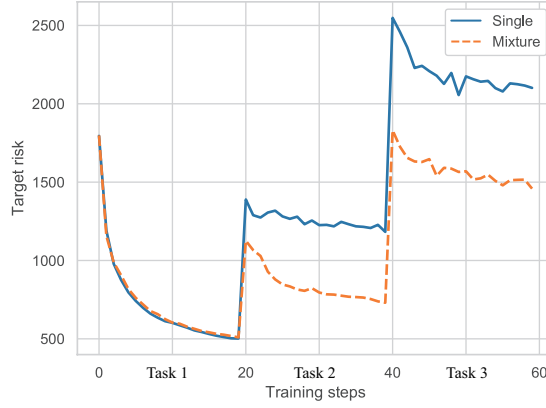
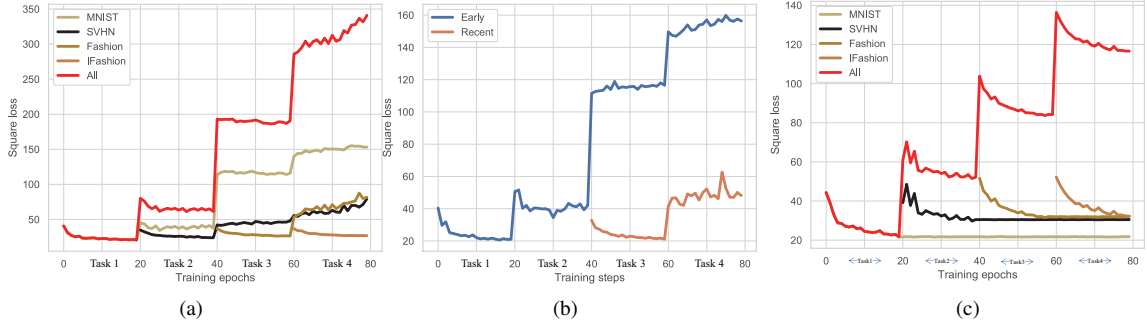Figure 7: Target risk on the single model and the mixture model.

(a)   (b)   (c)

Figure 8: "a" shows the target risk (LHS of Eq.(7) in Theorem-2 of the paper)) across four tasks under MSFI lifelong learning. "b" shows the target risk (LHS of Eq.(7) in Theorem-2 of the paper) on MNIST under MSFI lifelong learning. "c" shows the target risk (LHS of Eq.(7) in Theorem-2 of the paper) estimated by the proposed mixture model under (MSFI) lifelong learning. It notes that all results are the accumulated target error and we do not calculate the average result.

(a)   (b)   (c)
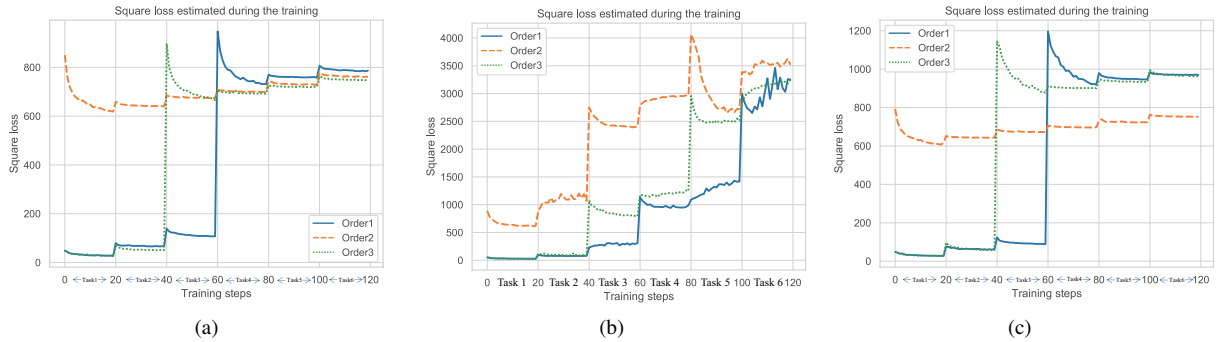
Figure 9: The accumulated target risks of DEGM, a single model and CN-DPM* with different orders of tasks. "a", "b" and "c" represent the results achieved by DEGM, a single model and CN-DPM*, respectively.

L.3 Evaluation by using other criterion

In addition to SL and NLL, we introduce to use the structural similarity index measure (SSIM) Hore and Ziou [2010] and the Peak-Signal-to-Noise Ratio (PSNR) Hore and Ziou [2010] as criteria. We report the results evaluated by the above criterion in Table 6 and Table 7.

Criteria	SL					SSMI					PSNR				
	BE	LGM	DEGM	DEGM-2	CN-DPM*	BE	LGM	DEGM	DEGM-2	CN-DPM*	BE	LGM	DEGM	DEGM-2	CN-DPM*
MNIST	26.3	685.3	22.3	22.3	21.9	0.88	0.19	0.90	0.90	0.90	21.0	7.0	21.8	21.8	21.8
SVHN	47.0	941.7	30.1	29.0	39.3	0.58	0.20	0.66	0.67	0.61	13.7	5.0	15.5	15.7	14.3
Fashion	43.8	663.4	37.7	27.4	36.6	0.68	0.15	0.72	0.79	0.73	18.4	3.7	19.0	20.6	19.2
IFashion	45.9	1148.4	35.6	27.4	38.4	0.72	0.11	0.76	0.81	0.76	18.2	5.0	19.4	20.6	19.1
RMNIST	27.9	704.2	20.2	22.1	25.3	0.87	0.20	0.91	0.90	0.89	16.5	7.0	22.2	21.8	21.2
Cifar10	994.4	1241.1	615.3	608.1	892.1	0.29	0.23	0.49	0.50	0.34	16.5	15.4	18.9	18.9	17.0
Average	197.5	897.4	126.9	122.7	175.6	0.67	0.18	0.74	0.76	0.70	18.1	7.2	19.5	19.9	18.8

Table 6: The performance of various models under the MSFIRC learning setting.

Criteria	SL					SSMI					PSNR				
	BE	LGM	DEGM	DEGM-2	CN-DPM*	BE	LGM	DEGM	DEGM-2	CN-DPM*	BE	LGM	DEGM	DEGM-2	CN-DPM*
CelebA	213.9	535.6	229.2	217.0	215.4	0.69	0.48	0.66	0.69	0.69	23.5	19.3	23.2	23.4	23.5
CACD	414.9	814.3	368.3	281.95	347.3	0.57	0.47	0.62	0.68	0.63	20.6	17.33	21.2	22.4	21.4
3D-Chair	649.1	2705.9	324.0	291.46	513.8	0.73	0.42	0.84	0.86	0.79	19.0	13.54	22.4	23.1	20.5
Omniglot	875.1	5958.9	225.6	195.7	343.2	0.73	0.22	0.92	0.93	0.89	17.9	9.2	24.0	24.6	22.1
Sub-ImageNet	758.4	683.1	689.6	652.8	769.1	0.37	0.42	0.41	0.43	0.37	18.5	18.9	19.0	19.2	18.5
Car	745.1	583.7	588.8	565.9	709.8	0.39	0.48	0.47	0.49	0.42	18.0	19.0	19.0	19.2	18.2
Zappos	451.1	431.2	263.4	275.8	280.7	0.68	0.60	0.75	0.74	0.73	20.0	20.2	22.4	22.3	22.1
CUB	492.0	330.2	461.3	569.6	638.6	0.35	0.48	0.45	0.43	0.35	19.0	20.9	19.3	18.6	18.0
Average	575.0	1505.4	393.8	381.3	477.2	0.60	0.45	0.64	0.66	0.61	19.6	17.3	21.3	21.6	20.5

Table 7: The performance of various models under the CCCOSCZC learning setting.

L.4 Ablation study

First, we evaluate the effectiveness of the proposed dynamic expansion mechanism used in our model. We train all models under MSFIRC lifelong learning and present the results in Fig 10a where we compare the average score, the training times and the memory use. It observes that even if the DEGM uses few training epochs but does not degenerate the performance. In addition, DEGM outperforms CN-DPM* that only transfers features from a single shared model. Furthermore, the proposed DEGM can achieve a similar performance as the baseline that trains individual VAEs for each task.
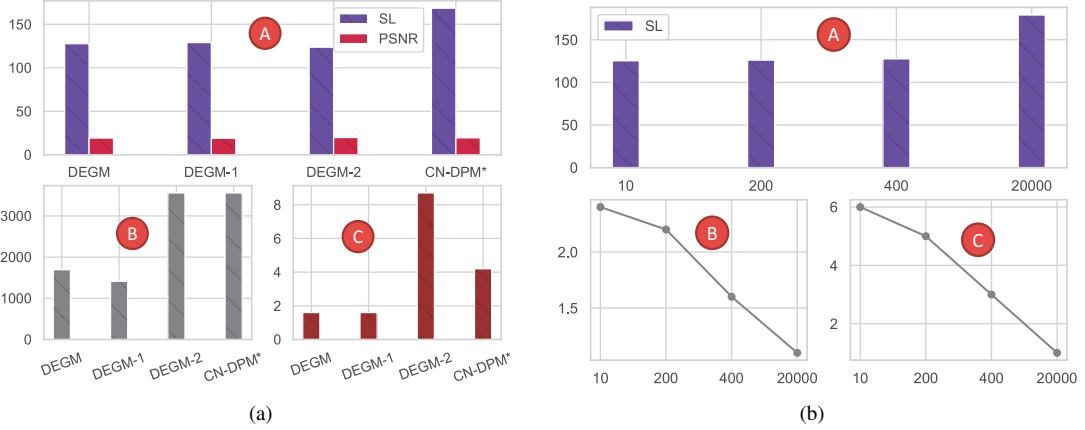

Figure 10: The results for various baselines under MSFIRC. In the left chart, "A" represent the average square loss and PSNR for all tasks. "B" represent the overall training times (seconds). "C" represents the number of parameters (10^8) for various baselines. In the right chart, "A" represent the average square loss for all tasks. "B" represent number of parameters (10^8). "C" represents the number of basic components. X-axis represents different threshold values thr

The effects of thresholds in ORVAE In the following, we investigate the performance of DEGM with different thresholds τ under MSFIRC lifelong learning and report results in Fig. 10b. As reduce τ, DEGM tends to increase the number of basic components and gradually improve performance. It observes that the choice of $\tau = 400$ can trade-off between the

performance and the model's complexity since it does not significantly improve the performance when decreasing thr from 400. In the next, in order to evaluate the effects of the proposed expansion mechanism and the adaptive weight, we introduce several new baselines in the following.

DEGM-4: This baseline generates information flows from all learned components to a new component. For instance, if a new component receives the information flow from members of $\mathcal{S}$, we will sum up the latent codes and intermediate representations from the sub-inference and sub-decoder of these components. DEGM-4 does not use the adaptive weight.

DEGM-5: We implement this baseline by considering to create the edges without using the adaptive weight. The training process for this baseline is described as follows: Once the t-th task learning was finished, As similar done to DEGM, we have a set of K measures, denoted by $KS = \{ks_1, \ldots, ks_K\}$, which can be used to build the edges from a new node to members of $\mathcal{G}$. We set a threshold τ which is used to update $\mathbf{V}$ such that if each $ks_i < \tau$, then $\mathbf{V}(t+1, \mathcal{GI}(i)) = 1$, otherwise $\mathbf{V}(t+1, \mathcal{GI}(i)) = 0$. This means that if $ks_i > \tau$, then the construction of a new component does not reuse the information and parameters from the i-th component in DEGM.

DEGM-6: For this baseline, we consider the adaptive weight for each edge is equal. This means that the importance of each basic component is treated as the same for a new task.

DEGM-7: We implement this baseline by considering to create only a single edge for a new component to a certain basic component that has the maximum sample log-likelihood for data of the new task.

CN-DPM*-1; This baseline builds a new components and creates connections with previously learned components, as similar to DEGM-4.

CN-DPM*-2: We implement this baseline by using the large model which contains 1.3×10^9 number of parameters.

We report the results in Table 8. It observes that the adaptive weights in the expansion mechanism can further improve the performance. We also find that although CN-DPM*-2 uses the more parameters, our DEGM still outperforms CN-DPM* by a large margin.

Criteria	Dataset	DEGM	DEGM-4	CN-DPM*-1	CN-DPM*-2	DEGM-5	DEGM-6	DEGM-7
	MNIST	22.30	22.18	22.12	22.67	22.88	21.48	22.35
	SVHN	30.18	30.73	40.53	38.74	30.56	29.44	29.20
	Fashion	37.73	41.22	45.03	38.51	37.65	37.26	41.29
SL	IFashion	35.62	49.26	36.19	36.90	41.27	37.15	43.95
	RMNIST	20.23	60.72	24.79	24.39	27.86	25.73	25.78
	Cifar10	615.34	610.38	929.55	877.35	617.34	617.36	614.48
	Average	**126.90**	135.58	183.03	173.09	129.59	128.07	129.51

Table 8: The results of various models under MSFIRC lifelong learning.

L.5 The number of parameters used in various methods

We list the number of parameters used in various methods in Table 9 and Table 10, respectively. We can observe that the proposed DEGM architecture requires less parameters than other parameters.

Model	LGM	BE	DEGM	DEGM-2	CN-DPM*
N	1.5×10^8	9.4×10^8	1.6×10^8	8.7×10^8	4.2×10^8

Table 9: The number of parameters of various models under MSFIR learning setting.

Model	LGM	BE	DEGM	DEGM-2	CN-DPM*	LIMix
N	1.9×10^9	3.9×10^9	3.2×10^8	1.3×10^9	9.4×10^9	9.4×10^9

Table 10: The number of parameters of various models under CCCOSCZC learning setting.

L.6 Visual results

We show the reconstructions from DEGM under MSFIRC and CCCOSCZC lifelong learning in Fig. 11 and Fig. 12, respectively.

| (a) MNIST. | (b) SVHN. | (c) Fashion. | (d) IFasion. | (e) RMNIST. | (f) Cifar10. |

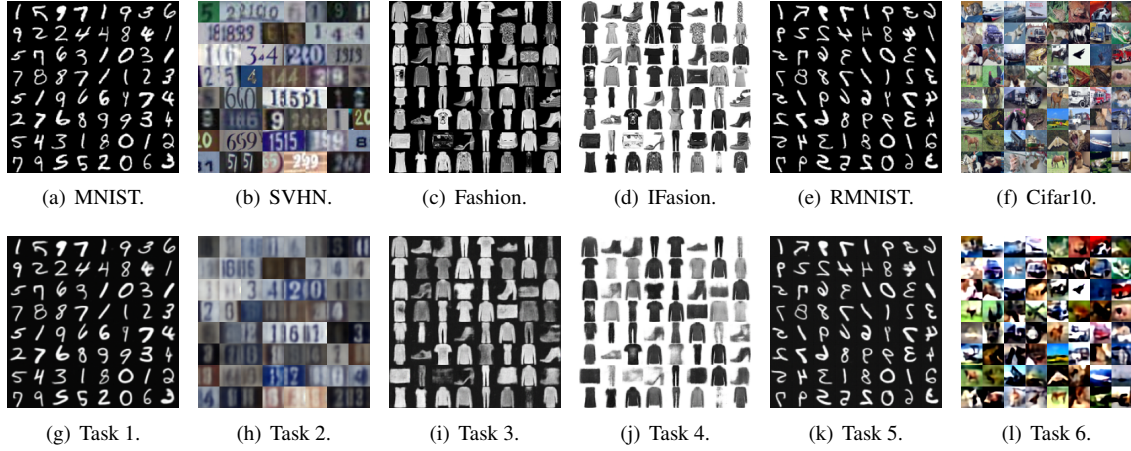| (g) Task 1. | (h) Task 2. | (i) Task 3. | (j) Task 4. | (k) Task 5. | (l) Task 6. |

Figure 11: Image reconstructions when using DEGM under the MSFIRC lifelong learning. The first row represents testing data samples and the second row are their reconstructions using DEGM.

| (a) Task1. | (b) Task2. | (c) Task3. | (d) Task4. | (e) Task5. | (f) Task6. | (g) Task7. |

| (h) Reconstruction of Task1. | (i) Reconstruction of Task2. | (j) Reconstruction of Task3. | (k) Reconstruction of Task4. | (l) Reconstruction of Task5. | (m) Reconstruction of Task6. | (n) Reconstruction of Task7. |

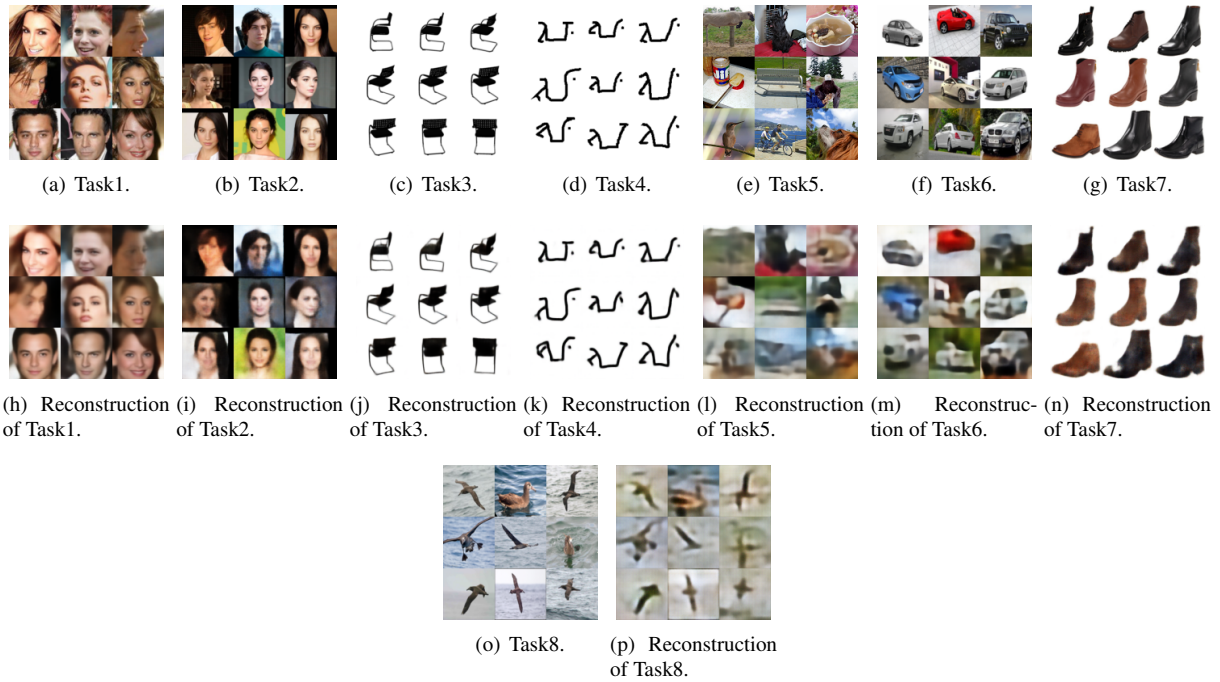| (o) Task8. | (p) Reconstruction of Task8. |

Figure 12: Image reconstructions when using DEGM under the CCCOSCZC lifelong learning. The first row represents testing data samples and the second row are their reconstructions using DEGM.

References

Ben Adlam, Corinna Cortes, Mehryar Mohri, and Ningshan Zhang. Learning GANs and ensembles using discrepancy. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, pages 1–12, 2019. 11

Mathieu Aubry, Daniel Maturana, Alexei A. Efros, Bryan C. Russell, and Josef Sivic. 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)*, pages 3762–3769, 2014. 27

Yuri Burda, Roger Grosse, and Ruslan Salakhutdinov. Importance weighted autoencoders. *arXiv preprint arXiv:1509.00519*, 2015. 9, 21

Bing Cao, Han Zhang, Nannan Wang, Xinbo Gao, and Dinggang Shen. Auto-GAN: self-supervised collaborative learning for medical image synthesis. In *Proc. of the AAAI Conf. on Artificial Intelligence*, pages 10486–10493, 2020. 14

Bor-Chun Chen, Chu-Song Chen, and Winston H. Hsu. Cross-age reference coding for age-invariant face recognition and retrieval. In *Proc. European Conf on Computer Vision (ECCV), vol. LNCS 8694*, pages 768–783, 2014. 27

Ting Chen, Xiaohua Zhai, Marvin Ritter, Mario Lucic, and Neil Houlsby. Self-supervised GANs via auxiliary rotation loss. In *Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12154–12163, 2019. 14

Xi Chen, Diederik P Kingma, Tim Salimans, Yan Duan, Prafulla Dhariwal, John Schulman, Ilya Sutskever, and Pieter Abbeel. Variational lossy autoencoder. *arXiv preprint arXiv:1611.02731*, 2016. 28

Carl Doersch. Tutorial on variational autoencoders. *arXiv preprint arXiv:1606.05908*, 2016. 20

Carl Doersch and Andrew Zisserman. Multi-task self-supervised visual learning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2051–2060, 2017. 14

Alain Hore and Djemel Ziou. Image quality metrics: PSNR vs. SSIM. In *Proc. Int. Conf. on Pattern Recognition (ICPR)*, pages 2366–2369, 2010. 31

D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In *Proc. Int. Conf. on Learning Representations (ICLR), arXiv preprint arXiv:1412.6980*, 2015. 24, 27

Diederik P. Kingma and Max Welling. Auto-encoding variational Bayes. *arXiv preprint arXiv:1312.6114*, 2013. 9, 14

Durk P Kingma, Tim Salimans, Rafal Jozefowicz, Xi Chen, Ilya Sutskever, and Max Welling. Improved variational inference with inverse autoregressive flow. In *Advances in Neural Information Processing Systems (NIPS)*, volume 29, pages 4743–4751, 2016. 21

Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. Technical report, Citeseer, 2009. 27

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In *Proc. Advances in Neural Inf. Proc. Systems (NIPS)*, pages 1097–1105, 2012. 27

Brenden M Lake, Ruslan Salakhutdinov, and Joshua B Tenenbaum. Human-level concept learning through probabilistic program induction. *Science*, 350(6266):1332–1338, 2015. 27

Yann LeCun, Leon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proc. of the IEEE*, 86(11):2278–2324, 1998. 27

Soochan Lee, Junsoo Ha, Dongsu Zhang, and Gunhee Kim. A neural Dirichlet process mixture model for task-free continual learning. In *Proc. Int. Conf. on Learning Representations (ICLR), arXiv preprint arXiv:2001.00689*, 2020. 23

Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaogang Tang. Deep learning face attributes in the wild. In *Proc. of IEEE Int. Conf. on Computer Vision (ICCV)*, pages 3730–3738, 2015. 27

Lars Maaløe, Casper Kaae Sønderby, Søren Kaae Sønderby, and Ole Winther. Auxiliary deep generative models. In *Proc. Int. Conf. on Machine Learning (ICML), vol. PMLR 48*, pages 1445–1453, 2016. 10, 21

Yishay Mansour, Mehryar Mohri, and Afshin Rostamizadeh. Domain adaptation: Learning bounds and algorithms. In *Proc. Conf. on Learning Theory (COLT), arXiv preprint arXiv:2002.06715*, 2009. 2, 13

Dmitry Molchanov, Valery Kharitonov, Artem Sobolev, and Dmitry Vetrov. Doubly semi-implicit variational inference. In *Proc. Int. Conf. on Artificial Intelligence and Statistics (AISTATS), vol. PMLR 89*, pages 2593–2602, 2019. 21

Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y. Ng. Reading digits in natural images with unsupervised feature learning. In *Proc. NIPS Workshop on Deep Learning and Unsupervised Feature Learning*, 2011. 27

Jason Ramapuram, Magda Gregorova, and Alexandros Kalousis. Lifelong generative modeling. *Neurocomputing*, 404: 381–400, 2020. 3

Dushyant Rao, Francesco Visin, Andrei A. Rusu, Yee W. Teh, Razvan Pascanu, and Raia Hadsell. Continual unsupervised representation learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 1–11, 2019. 23

Ari Seff, Alex Beatson, Daniel Suo, and Han Liu. Continual learning in generative adversarial nets. *arXiv preprint arXiv:1705.08395*, 2017. 14

Artem Sobolev and Dmitry Vetrov. Importance weighted hierarchical variational inference. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, pages 1–13, 2019. 10, 21

Casper Kaae Sønderby, Tapani Raiko, Lars Maaløe, Søren Kaae Sønderby, and Ole Winther. Ladder variational autoencoders. In *Advances in neural information processing systems*, pages 3738–3746, 2016. 11

Ngoc-Trung Tran, Viet-Hung Tran, Bao-Ngoc Nguyen, Linxiao Yang, and Ngai-Man (Man) Cheung. Self-supervised gan: Analysis and improvement with multi-class minimax game. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper/2019/file/d04cb95ba2bea9fd2f0daa8945d70f11-Paper.pdf. 14

Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The Caltech-UCSD birds-200 dataset. Technical Report CNS-TR-2010-001, California Institute of Technology, 2010. 27

Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-MNIST: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017. 27

Linjie Yang, Ping Luo, Chen Change Loy, and Xiaoou Tang. A large-scale car dataset for fine-grained categorization and verification. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 3973–3981, 2015. 27

Fei Ye and Adrian G. Bors. Lifelong infinite mixture model based on knowledge-driven dirichlet process. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10695–10704, October 2021. 23, 24

Fei Ye and Adrian G.Bors Bors. Learning latent representations across multiple data domains using lifelong vaegan. In *Proc. of European Conference on Computer Vision (ECCV), vol. LNCS 12365*, pages 777–795, 2020. 3

Mingzhang Yin and Mingyuan Zhou. Semi-implicit variational inference. In *Proc. of Int. Conf. on Machine Learning (ICML), vol. PMLR 80*, pages 5660–5669, 2018. 21

Aron Yu and Kristen Grauman. Semantic jitter: Dense supervision for visual comparisons via synthetic images. In *Proc. IEEE Int. Conf. on Computer Vision (ICCV)*, pages 5571–5580, 2017. 27

Xiaohua Zhai, Avital Oliver, Alexander Kolesnikov, and Lucas Beyer. S4l: Self-supervised semi-supervised learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1476–1485, 2019. 14

Junbo Zhao, Michael Mathieu, and Yann LeCun. Energy-based generative adversarial network. In *Proc. Int. Conf. on Learning Representations (ICLR), arXiv preprint arXiv:1609.03126*, 2017. 13