# Appendix for Learning latent representations across multiple data domains using Lifelong VAEGAN

Fei Ye and Adrian G. Bors

Dept. of Computer Science, University of York, York YO10 5GH, UK

## A Generalization Bounds for the generative replay mechanism

In the following, we extend the theory analysis on domain adaptation from [7] (Theorem 2) to the generative replay mechanism.

**Theorem 2.** *Let us consider two vector samples, one corresponding to the generated data* $\{\nu_{t'} \in \mathrm{R}^{\mathrm{s}} | \nu_{\mathrm{t}'} \sim \mathrm{p}(\tilde{\mathbf{x}}^{\mathrm{t}})\}$ *and another corresponding to the real data* $\{\nu_t \in \mathrm{R}^{\mathrm{s}} | \nu_{\mathrm{t}} \sim \mathrm{p}(\mathbf{x}^{\mathrm{t}})\}$ *of size* $n_t$ *and* $n_{t'}$*, respectively. Then let* $h^t(\cdot)$ *be a new learned model trained on* $\nu_{t'}$*. For any* $s' > s$ *and* $a' < \sqrt{2}$*, there is a constant* $n_0$ *depending on* $s'$ *satisfying that for any* $\delta > 0$ *and* $\min(\nu_t, n_{t'}) \geq n_0 \max(\delta^{-(s'+2)}, 1)$*. Then we have the following inequality, with the probability of at least* $1 - \delta$ *for any* $h^t$ *:*

$$E\left(h^t(\nu_t)\right) \leq E\left(h^t(\nu_{t'})\right) + \mathbf{W}\left(\nu_t, \nu_{t'}\right) + \sqrt{2\log\left(\frac{1}{\delta}\right)/a'}\left(\sqrt{\frac{1}{n_t}} + \sqrt{\frac{1}{n_{t'}}}\right) + D \tag{1}$$

*where* $E(h^t(\nu_t)) := \mathbb{E}_{\nu_t \sim p(\mathbf{x}^t)}[|h^t(\nu_t) - g(\nu_t)|]$,$E(h^t(\nu_{t'})) := \mathbb{E}_{\nu_{t'} \sim p(\tilde{\mathbf{x}}^t)}[|h^t(\nu_{t'}) - g(\nu_{t'})|]$ *denote the observed risk for* $\nu_t$ *and* $\nu_{t'}$*, respectively, and* $g(\cdot)$ *is the ground-truth labeling function.* $\mathbf{W}\left(\nu_t, \nu_{t'}\right)$ *is the Wassenstein distance between* $\nu_t$ *and* $\nu_{t'}$*. D is the combined error when we find the optimal model* $h^{t'} = \arg\min(E(h^t(\nu_t)) + E(h^t(\nu_{t'}))$*.*

This theorem clearly demonstrates that the performance degeneration of a new learned model $h^t$ depends on the empirical data distribution $p(\mathbf{x}^t)$. From Theorem 2, we can conclude that with the generative replay mechanism, the lifelong learning can be defined as a special domain adaptation case, in which the source and target domain are the empirical data distributions from the current task and the distribution approximated by the new learned model. As a direct consequence of Theorem 2, we have the following Lemma 2 :

**Lemma 2.** There is a bound on the accumulated errors across all tasks, learned from the given sequence of databases, during the lifelong learning :

$$\begin{aligned} &\textstyle\sum_{i=1}^{K} E\left(h^K(\nu_i)\right) \leq \sum_{i=1}^{K} E\left(h^K(\nu_{i^{(K)}})\right) + \\ &\mathbf{W}\left(\nu_i, \nu_{i^{(K)}}\right) + \sqrt{2\log\left(\frac{1}{\delta}\right)/a'}\left(\sqrt{\frac{1}{n_i}} + \sqrt{\frac{1}{n_{i^{(K)}}}}\right) + D_{(i^{(K-1)}, i^{(K)})}, \end{aligned} \tag{2}$$

where $E\left(h^K(\nu_{i^{(K)}})\right)$ denotes the observed risk on the probability measure $\nu_{i^{(K)}}$ formed by samples drawn from $p(\tilde{\mathbf{x}}^i)$, after they have been learned across K tasks. $D_{(i^{(K-1)},i^{(K)})}$ is the combined error of an optimal model

$$h^* = \arg\min(E\left(h^K(\nu_{i^{(K-1)}})\right) + E\left(h^K(\nu_{i^{(K)}})\right)) \tag{3}$$

**Proof.** From Theorem 2, we can derive the following :

$$\begin{aligned} &E\left(h^1(\nu_1)\right) \leq E\left(h^1(\nu_{1'})\right) + B_{1'} + D_{(1,1')} \\ &E\left(h^1(\nu_{1'})\right) \leq E\left(h^1(\nu_{1^2})\right) + B_{1^2} + D_{(1',1^2)} \\ &\dots \\ &E\left(h^1(\nu_{1^{K-1}})\right) \leq E\left(h^1(\nu_{1^K})\right) + B_{1^{K-1}} + D_{(1^{K-1},1^K)} \end{aligned} \tag{4}$$

And then we can have:

$$E\left(h^1(\nu_1)\right) \leq E\left(h^1(\nu_{1^K})\right) + B_{1^K} + C_{(1^{k-1},1^K)}$$

where

$$B_{1^k}=\mathbf{W}\left(\nu_1,\nu_{1^k}\right) + \sqrt{2\log\left(\frac{1}{\delta}\right)/a'}\left(\sqrt{\frac{1}{n_1}}+\sqrt{\frac{1}{n_{1^k}}}\right)$$

where $n_1$ and $n_{1^K}$ denote the sample size for $\nu_1$ and $\nu_{1^K}$, respectively.

And then we sum up all task risks, resulting in:

$$\begin{aligned} &\textstyle\sum_{i=1}^K E\left(h^K(\nu_i)\right) \leq \sum_{i=1}^K E\left(h^K(\nu_{i^{(K)}})\right) + \\ &\mathbf{W}\left(\nu_i,\nu_{i^{(K)}}\right) + \sqrt{2\log\left(\frac{1}{\delta}\right)/a'}\left(\sqrt{\frac{1}{n_i}}+\sqrt{\frac{1}{n_{i^{(K)}}}}\right) + D_{(i^{(K-1)},i^{(K)})}, \end{aligned} \tag{5}$$

## B The proof for Theorem 1 from the paper

In the paper, we define in equation (9) that $p(\tilde{\mathbf{x}}^t|\tilde{\mathbf{x}}^{t-1},\mathbf{x}^t)=\exp(-(\mathit{\Gamma}(p(\tilde{\mathbf{x}}^{t-1},\mathbf{x}^t),p(\tilde{\mathbf{x}}^t)))$ as a probability of $\tilde{\mathbf{x}}^t$ when observing $\tilde{\mathbf{x}}^{t-1}$ and $\mathbf{x}^t$, given that the proposed model aims to align two distributions $p(\tilde{\mathbf{x}}^{i-1},\mathbf{x}^i)$ and $p(\tilde{\mathbf{x}}^i)$ where $i>1$ at $i$-th task learning. We can have the joint distribution $p(\tilde{\mathbf{x}}^t,\tilde{\mathbf{x}}^{t-1},\mathbf{x}^t)=p(\tilde{\mathbf{x}}^t|\tilde{\mathbf{x}}^{t-1},\mathbf{x}^t)p(\tilde{\mathbf{x}}^{t-1},\mathbf{x}^t)$

The marginal probability is calculated by the following :

$$\begin{aligned} &p(\tilde{\mathbf{x}}^t) = \int\int p(\tilde{\mathbf{x}}^t|\tilde{\mathbf{x}}^{t-1},\mathbf{x}^t)p(\tilde{\mathbf{x}}^{t-1},\mathbf{x}^t)d\tilde{\mathbf{x}}^{t-1}d\mathbf{x}^t \\ &= \int\int p(\tilde{\mathbf{x}}^t|\tilde{\mathbf{x}}^{t-1},\mathbf{x}^t)p(\tilde{\mathbf{x}}^{t-1})p(\mathbf{x}^t)d\tilde{\mathbf{x}}^{t-1}d\mathbf{x}^t = \\ &\int\int\int\int p(\tilde{\mathbf{x}}^t|\tilde{\mathbf{x}}^{t-1},\mathbf{x}^t)p(\tilde{\mathbf{x}}^{t-1}|\tilde{\mathbf{x}}^{t-2},\mathbf{x}^{t-1})p(\tilde{\mathbf{x}}^{t-2})p(\mathbf{x}^t)p(\mathbf{x}^{t-1})d\tilde{\mathbf{x}}^{t-1}d\mathbf{x}^t d\tilde{\mathbf{x}}^{t-2}d\mathbf{x}^{t-1} \\ &= \int\dots\int p(\widehat{\mathbf{x}}_1)\prod_{i=0}^{t-2}p(\tilde{\mathbf{x}}^{t-i}|\tilde{\mathbf{x}}^{t-i-1},\mathbf{x}^{t-i})\prod_{i=0}^{t-2}p(\mathbf{x}^{t-i})d\tilde{\mathbf{x}}^1\dots d\tilde{\mathbf{x}}^{t-1}d\mathbf{x}^2\dots d\mathbf{x}^t \end{aligned} \tag{6}$$

This function describes how after the model would initially learn a distribution $p(\tilde{\mathbf{x}})$, then its knowledge can be refined to learn a much more complex distribution $p(\tilde{\mathbf{x}}^t)$ through the lifelong learning of multiple databases.

## C The proof for Lemma 1

In order to have $p(\tilde{\mathbf{x}}^t) = \prod_{i=1}^{t} p(\mathbf{x}^i)$, we must firstly satisfy the following condition:

$$p(\tilde{\mathbf{x}}^t|\tilde{\mathbf{x}}^{t-1}, \mathbf{x}^t) = 1 \Rightarrow p(\tilde{\mathbf{x}}^t) = p(\tilde{\mathbf{x}}^{t-1}, \mathbf{x}^t) \tag{7}$$

where the right hand side can be decomposed as $p(\tilde{\mathbf{x}}^{t-1})p(\mathbf{x}^t)$ since $p(\tilde{\mathbf{x}}^{t-1})$ is independent from $p(\mathbf{x}^t)$. We further decompose $p(\tilde{\mathbf{x}}^{t-1})$ as $p(\tilde{\mathbf{x}}^{t-2})p(\mathbf{x}^{t-1})$ if satisfy $p(\tilde{\mathbf{x}}^{t-1}|\tilde{\mathbf{x}}^{t-2}, \mathbf{x}^{t-1}) = 1$. By considering all decomposition, we have:

$$\begin{aligned} &\prod_{i=0}^{t-2} p(\tilde{\mathbf{x}}^{t-i}|\tilde{\mathbf{x}}^{t-i-1}, \mathbf{x}^{t-i}) = 1 \\ &p(\tilde{\mathbf{x}}^1) = p(\mathbf{x}^1) \Rightarrow p(\tilde{\mathbf{x}}^t) = p(\mathbf{x}^1, \ldots, \mathbf{x}^t) \end{aligned} \tag{8}$$

## D The proof for Theorem 3.

In this case, we only consider two separate underlying generative factors $\mathbf{z}^t$ and $\mathbf{z}^{t-1}$ and define the latent variable model $p(\mathbf{x}^t, \tilde{\mathbf{x}}^t, \mathbf{z}^t, \mathbf{z}^{t-1}) = p(\mathbf{x}^t, \tilde{\mathbf{x}}^{t-1}|\mathbf{z}^t, \mathbf{z}^{t-1})p(\mathbf{z}^t)$ $p(\mathbf{z}^{t-1})$. It can be easily extended to multiple variables. The marginal likelihood is calculated as:

$$\begin{aligned} p(\mathbf{x}^t, \tilde{\mathbf{x}}^{t-1}) &= \int\int p(\mathbf{x}^t, \mathbf{z}^t)p(\tilde{\mathbf{x}}^{t-1}, \mathbf{z}^{t-1})d\mathbf{z}^t d\mathbf{z}^{t-1} \\ &= \int p(\mathbf{x}^t, \mathbf{z}^t)d\mathbf{z}^t \cdot \int p(\tilde{\mathbf{x}}^{t-1}, \mathbf{z}^{t-1})d\mathbf{z}^{t-1} \end{aligned} \tag{9}$$

where we assume $p(\mathbf{x}^t, \mathbf{z}^t)$ is independent from $p(\tilde{\mathbf{x}}^{t-1}, \mathbf{z}^{t-1})$. The marginal log-likelihood function is derived as:

$$\begin{aligned} \log p(\mathbf{x}^t, \tilde{\mathbf{x}}^{t-1}) &= \log\left(\int p(\mathbf{x}^t, \mathbf{z}^t)\ d\mathbf{z}^t \cdot \int p(\tilde{\mathbf{x}}^{t-1}, \mathbf{z}^{t-1})\ d\mathbf{z}^{t-1}\right) \\ &= \log\left(\int p(\mathbf{x}^t, \mathbf{z}^t)d\mathbf{z}^t\right) + \log\left(\int p(\tilde{\mathbf{x}}^{t-1}, \mathbf{z}^{t-1})d\mathbf{z}^{t-1}\right) \\ &= \log\left(\int p(\mathbf{x}^t, \mathbf{z}^t)\frac{q(\mathbf{z}^{t+}|\mathbf{x}^t)}{q(\mathbf{z}^{t+}|\mathbf{x}^t)}d\mathbf{z}^t\right) + \\ &\log\left(\int p(\tilde{\mathbf{x}}^{t-1}, \mathbf{z}^{t-1})\frac{q(\mathbf{z}^{t-1}|\tilde{\mathbf{x}}^{t-1})}{q(\mathbf{z}^{t-1}|\tilde{\mathbf{x}}^{t-1})}d\mathbf{z}^{t-1}\right) \\ &= \log \mathbb{E}_{q(\mathbf{z}^{t+}|\mathbf{x}^t)}\left[\frac{p(\mathbf{x}^t, \mathbf{z}^t)}{q(\mathbf{z}^t|\mathbf{x}^t)}\right] + \log \mathbb{E}_{q(\mathbf{z}^{t-1}|\tilde{\mathbf{x}}^{t-1})}\left[\frac{p(\tilde{\mathbf{x}}^{t-1}, \mathbf{z}^{t-1})}{q(\mathbf{z}^{t-1}|\tilde{\mathbf{x}}^{t-1})}\right] \end{aligned} \tag{10}$$

where $q(\mathbf{z}^t|\mathbf{x}^t)$ and $q(\mathbf{z}^{t-1}|\tilde{\mathbf{x}}^{t-1})$ are variational distributions. Then we derive a lower bound on the model log-likelihood by using the Jensens inequality.

$$\log p(\mathbf{x}^t, \tilde{\mathbf{x}}^{t-1}) \geq \mathbb{E}_{q(\mathbf{z}^t|\mathbf{x}^t)}\left[\log\frac{p(\mathbf{x}^t, \mathbf{z}^t)}{q(\mathbf{z}^t|\mathbf{x}^t)}\right] + \mathbb{E}_{q(\mathbf{z}^{t-1}|\tilde{\mathbf{x}}^{t-1})}\left[\log\frac{p(\tilde{\mathbf{x}}^{t-1}, \mathbf{z}^{t-1})}{q(\mathbf{z}^{t-1}|\tilde{\mathbf{x}}^{t-1})}\right] \tag{11}$$

Then we decompose the two terms from the right hand side, while we omit the superscript for $\mathbf{z}$ for the sake of simplicity :

$$\begin{aligned}\log p(\mathbf{x}^t)p(\tilde{\mathbf{x}}^{t-1}) &\geq \mathbb{E}_{q(\mathbf{z}|\mathbf{x}^t)}\left[\log \frac{p(\mathbf{x}^t|\mathbf{z})p(\mathbf{z})}{q(\mathbf{z}|\mathbf{x}^t)}\right] + \mathbb{E}_{q(\mathbf{z}|\tilde{\mathbf{x}}^{t-1})}\left[\log \frac{p(\tilde{\mathbf{x}}^{t-1}|\mathbf{z})p(\mathbf{z})}{q(\mathbf{z}|\tilde{\mathbf{x}}^{t-1})}\right] \\ &= \mathbb{E}_{q(\mathbf{z}|\mathbf{x}^{t+})}\left[\log p(\mathbf{x}^t|\mathbf{z})\right] + \mathbb{E}_{q(\mathbf{z}|\mathbf{x}^t)}\left[\log \frac{p(\mathbf{z})}{q(\mathbf{z}|\mathbf{x}^t)}\right] + \\ &\mathbb{E}^{t-1}_{q(\mathbf{z}|\tilde{\mathbf{x}})}\left[\log p(\tilde{\mathbf{x}}^{t-1}|\mathbf{z})\right] + \mathbb{E}_{q(\mathbf{z}|\tilde{\mathbf{x}}^{t-1})}\left[\log \frac{p(\mathbf{z})}{q(\mathbf{z}|\tilde{\mathbf{x}}^{t-1})}\right] \\ &= \mathbb{E}_{q(\mathbf{z}|\mathbf{x}^t)}\left[\log p(\mathbf{x}^t|\mathbf{z})\right] - D_{KL}(q(\mathbf{z}|\mathbf{x}^t)||p(\mathbf{z})) + \\ &\mathbb{E}_{q(\mathbf{z}|\tilde{\mathbf{x}}^{t-1})}\left[\log p(\tilde{\mathbf{x}}^{t-1}|\mathbf{z})\right] - D_{KL}(q(\mathbf{z}|\tilde{\mathbf{x}}^{t-1})||p(\mathbf{z}))\end{aligned} \tag{12}$$

In practice, we can implement variational distributions by using a single probabilistic encoder, and this can have many advantages. For instance, the latent space can capture specific task information in several subspaces and capture the shared information between different domains in the same subspace. In addition, we can easily perform many down-stream tasks such as inference and reconstruction across domains.

## E The proof for Lemma 3.

**Lemma 3.** From Theorem 2 and Theorem 3, we can derive a lower bound on the ELBO at t-th task learning, as expressed by:

$$\begin{aligned}\mathcal{L}(\theta,\xi;\mathbf{x}^1,..,\mathbf{x}^t) \geq \mathcal{L}(\theta,\xi;\mathbf{x}^t,\tilde{\mathbf{x}}^{t-1}) - \mathrm{W}(v,v') - \sqrt{2\log\left(\frac{1}{\delta}\right)/a'}\left(\sqrt{\frac{1}{n}}+\sqrt{\frac{1}{n'}}\right) \\ - D^*\end{aligned} \tag{13}$$

where $\nu \in \mathbf{R}^s, \nu' \in \mathbf{R}^s$ are formed by $n$ and $n'$ numbers of drawn samples from $p(\mathbf{x}^t)p(\tilde{\mathbf{x}}^{t-1})$ and $\prod_i^t p(\mathbf{x}^i)$, respectively, where $n$ and $n'$ denote the sample size.

**Proof.** We consider the negative EBLO $-\mathcal{L}(\theta,\xi;\mathbf{x}^1,..,\mathbf{x}^t)$ as the observed risk for $v$ and $-\mathcal{L}(\theta,\xi;\mathbf{x}^t,\tilde{\mathbf{x}}^{t-1})$ as the observed risk for $v'$. The $h^t$ is expressed as the proposed model that measures the ELBO. For any $s' > s$ and $a' < \sqrt{2}$, there exists some constant $n_0$ depending on $s'$ satisfying that for any $\delta > 0$ and $\min(n,n') \geq n_0 \max(\delta^{-(s'+2)},1)$. Then with the probability at least $1-\delta$ for all $h^t$, we can have:

$$\begin{aligned}-\mathcal{L}(\theta,\xi;\mathbf{x}^1,..,\mathbf{x}^t) \leq -\mathcal{L}(\theta,\xi;\mathbf{x}^t,\tilde{\mathbf{x}}^{t-1}) \\ +\mathrm{W}(v,v')+\sqrt{2\log\left(\frac{1}{\delta}\right)/a'}\left(\sqrt{\frac{1}{n}}+\sqrt{\frac{1}{n'}}\right) + D^*\end{aligned} \tag{14}$$

where $D^*$ is the combined error of an optimal model $h^*$ that minimizes the errors $(-\mathcal{L}(\theta,\xi;\mathbf{x}^1,..,\mathbf{x}^t) - \mathcal{L}(\theta,\xi;\mathbf{x}^t,\tilde{\mathbf{x}}^{t-1}))$. Then Both sides are multiplied by -1, resulting in:

$$\mathcal{L}(\theta,\xi;\mathbf{x}^1,..,\mathbf{x}^t) \geq \mathcal{L}(\theta,\xi;\mathbf{x}^t,\tilde{\mathbf{x}}^{t-1}) - \mathrm{W}(v,v') - \sqrt{2\log\left(\frac{1}{\delta}\right)/a'}\left(\sqrt{\frac{1}{n}}+\sqrt{\frac{1}{n'}}\right) - D^* \tag{15}$$

This result shows that we can derive a lower bound on real sample log-likelihood $\log p_\theta(\mathbf{x}^1,\ldots,\mathbf{x}^t) \geq \mathcal{L}(\theta,\xi;\mathbf{x}^t,\tilde{\mathbf{x}}^{t-1})$. We also show the connection between domain adaptation and generative replay mechanism such that $\prod_i^t p(\mathbf{x}^i)$ and $p(\mathbf{x}^t)p(\tilde{\mathbf{x}}^{t-1})$ can be seen as the source domain and target domain under the context of domain adaptation.

In the following, we provide the quantitative results for Lemma 3. We train the proposed model under the MNIST to Fashion lifelong learning setting. In order investigate the convergence of the proposed algorithm, we calculate the $\mathcal{L}(\theta,i;\mathbf{x}^t,\widetilde{\mathbf{x}}^{t-1})$ and $\mathcal{L}(\theta,\xi;\mathbf{x}^1,\ldots,\mathbf{x}^t)$ during the second task learning. The results are provided in Figure 1, where ELBO1 and ELBO2 - A denote $\mathcal{L}(\theta,i;\mathbf{x}^t,\widetilde{\mathbf{x}}^{t-1})$ and $\mathcal{L}(\theta,\xi;\mathbf{x}^1,\ldots,\mathbf{x}^t) - \sqrt{2\log\left(\frac{1}{\delta}\right)/a'}\left(\sqrt{\frac{1}{n}}+\sqrt{\frac{1}{n'}}\right)$, respectively. We can observe that $\sqrt{2\log\left(\frac{1}{\delta}\right)/a'}\left(\sqrt{\frac{1}{n}}+\sqrt{\frac{1}{n'}}\right)$ can be calculated explicitly. However, if we can calculate the $-\mathrm{W}(v,v') - \sqrt{\frac{1}{n'}}) - C$, explicitly. Then $\mathcal{L}(\theta,\xi;\mathbf{x}^1,\ldots,\mathbf{x}^t)$ is bounded by the right hand side of equation (15). Lemma 3 shows that maximizing sample log-likelihood is equal to minimizing the two terms (one is the distance between empirical and the approximated distributions while the second is the combined error $C$)

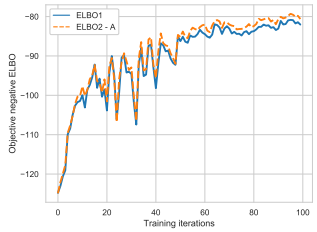



**Fig. 1.** ELBO calculated during the lifelong learning of MNIST to Fashion.

## F The derivation of $\mathcal{L}_{\mathbf{VAE}}$.

In this case, we only consider to model a single task, then we have:

$$\log p(\mathbf{x}) = \log \mathbb{E}_{q_{\varsigma,\varepsilon,\delta}(\mathbf{z},\mathbf{a},\mathbf{c}|\mathbf{x})}\left[\frac{p_\theta(\mathbf{x},\mathbf{z},\mathbf{a},\mathbf{c})}{q_{\varsigma,\varepsilon,\delta}(\mathbf{z},\mathbf{a},\mathbf{c}|\mathbf{x})}\right] \tag{16}$$

Then according to Jensens' inequality, we have:

$$\log p(\mathbf{x}) \geq \mathbb{E}_{q_{\varsigma,\varepsilon,\delta}(\mathbf{z},\mathbf{a},\mathbf{c}|\mathbf{x})}\left[\log\frac{p_\theta(\mathbf{x},\mathbf{z},\mathbf{a},\mathbf{c})}{q_{\varsigma,\varepsilon,\delta}(\mathbf{z},\mathbf{a},\mathbf{c}|\mathbf{x})}\right] \tag{17}$$

$$\begin{aligned}\mathcal{L}_{\text{VAE}}(\theta,\varsigma,\varepsilon,\delta) &= \mathbb{E}_{q_{\varsigma,\varepsilon,\delta}(\mathbf{z},\mathbf{a},\mathbf{c}|\mathbf{x})}\log\left[\frac{p_\theta(\mathbf{x},\mathbf{z},\mathbf{a},\mathbf{c})}{q_{\varsigma,\varepsilon,\delta}(\mathbf{z},\mathbf{a},\mathbf{c}|\mathbf{x})}\right] \\ &= \mathbb{E}_{q_\delta(\mathbf{c}|\mathbf{x})q_{,\varepsilon}(\mathbf{a}|\mathbf{z})q_\varsigma(\mathbf{z}|\mathbf{x})}\log\left[\frac{p_\theta(\mathbf{x}|\mathbf{z},\mathbf{a},\mathbf{c})p(\mathbf{a}|\mathbf{z})p(\mathbf{z})p(\mathbf{c})}{q_\delta(\mathbf{c}|\mathbf{x})q_{,\varepsilon}(\mathbf{a}|\mathbf{z})q_\varsigma(\mathbf{z}|\mathbf{x})}\right] \\ &= \mathbb{E}_{q_{\varsigma,\varepsilon,\delta}(\mathbf{z},\mathbf{a},\mathbf{c}|\mathbf{x})}\log[p_\theta(\mathbf{x}|\mathbf{z},\mathbf{a},\mathbf{c})] - D_{KL}[q_\varsigma(\mathbf{z}|\mathbf{x})||p(\mathbf{z})] \\ &- \mathbb{E}_{q_\delta(\mathbf{c}|\mathbf{x})}D_{KL}[q_\varepsilon(\mathbf{a}|\mathbf{z})||p(\mathbf{a}|\mathbf{z})] - D_{KL}[q_\delta(\mathbf{c}|\mathbf{x})||p(\mathbf{c})]\end{aligned} \tag{18}$$

where we have separated the Kullback-Leibler (KL) divergence components for the continuous $\mathbf{z}$ space, as well as for the discrete and domain spaces $\mathbf{c}$ and $\mathbf{d}$, respectively. Meanwhile, $\theta,\varsigma,\varepsilon,\delta$ represent the parameters of the corresponding networks.

## G The derivation of $\mathcal{L}_{\text{VAE}}(\theta_t,\varsigma_t,\varepsilon_t,\delta_t)$.

From Theorem 2, we can define the following latent variable model :

$$\begin{aligned}&p(\tilde{\mathbf{x}}^{t-1},\mathbf{x}^t,\mathbf{z}^{t+1},\mathbf{a}^{t+1},\mathbf{c}^{t+1},\mathbf{z}^t,\mathbf{a}^t,\mathbf{c}^t) = \\ &p(\tilde{\mathbf{x}}^{t-1}|\mathbf{z}^{t+1},\mathbf{a}^{t+1},\mathbf{c}^{t+1})p(\mathbf{z}^{t+1},\mathbf{a}^{t+1},\mathbf{c}^{t+1})p(\mathbf{x}^t|\mathbf{z}^t,\mathbf{a}^t,\mathbf{c}^t)p(\mathbf{z}^t,\mathbf{a}^t,\mathbf{c}^t)\end{aligned} \tag{19}$$

The marginal log-likelihood can be rewritten as :

$$\begin{aligned}&\log p(\tilde{\mathbf{x}}^{t-1},\mathbf{x}^t) = \\ &\log\int\int\int p(\tilde{\mathbf{x}}^{t-1}|\mathbf{z}^{t+1},\mathbf{a}^{t+1},\mathbf{c}^{t+1})p(\mathbf{z}^{t+1},\mathbf{a}^{t+1},\mathbf{c}^{t+1})d\mathbf{z}^{t+1}d\mathbf{a}^{t+1}d\mathbf{c}^{t+1} \\ &+\log\int\int\int p(\mathbf{x}^t|\mathbf{z}^t,\mathbf{a}^t,\mathbf{c}^t)p(\mathbf{z}^t,\mathbf{a}^t,\mathbf{c}^t)d\mathbf{z}^t d\mathbf{a}^t d\mathbf{c}^t \\ =&\log\mathbb{E}_{q_{\varsigma,\varepsilon,\delta}(\mathbf{z}^{t+1},\mathbf{a}^{t+1},\mathbf{c}^{t+1}|\mathbf{x}^t)}\left[\frac{p_\theta(\tilde{\mathbf{x}}^{t-1},\mathbf{z}^{t+1},\mathbf{a}^{t+1},\mathbf{c}^{t+1})}{q_{\varsigma,\varepsilon,\delta}(\mathbf{z}^{t+1},\mathbf{a}^{t+1},\mathbf{c}^{t+1}|\tilde{\mathbf{x}}^{t-1})}\right] + \\ &\log\mathbb{E}_{q_{\varsigma,\varepsilon,\delta}(\mathbf{z}^t,\mathbf{a}^t,\mathbf{c}^t|\mathbf{x}^t)}\left[\frac{p_\theta(\mathbf{x}^t,\mathbf{z}^t,\mathbf{a}^t,\mathbf{c}^t)}{q_{\varsigma,\varepsilon,\delta}(\mathbf{z}^t,\mathbf{a}^t,\mathbf{c}^t|\mathbf{x}^t)}\right]\end{aligned} \tag{20}$$

Then according to Jensens' inequality, we can rewrite the above equation as.

$$\begin{aligned}\log p(\tilde{\mathbf{x}}^{t-1},\mathbf{x}^t) \geq\ &\mathbb{E}_{q_{\varsigma,\varepsilon,\delta}(\mathbf{z},\mathbf{a},\mathbf{c}|\tilde{\mathbf{x}}^{t-1})}\left[\log\frac{p_\theta(\tilde{\mathbf{x}}^{t-1}|\mathbf{z},\mathbf{a},\mathbf{c})}{q_{\varsigma,\varepsilon,\delta}(\mathbf{z},\mathbf{a},\mathbf{c}|\tilde{\mathbf{x}}^{t-1})}\right] \\ &+\mathbb{E}_{q_{\varsigma,\varepsilon,\delta}(\mathbf{z},\mathbf{a},\mathbf{c}|\mathbf{x}^t)}\left[\log\frac{p_\theta(\mathbf{x}^t|\mathbf{z},\mathbf{a},\mathbf{c})}{q_{\varsigma,\varepsilon,\delta}(\mathbf{z},\mathbf{a},\mathbf{c}|\mathbf{x}^t)}\right]\end{aligned} \tag{21}$$

where the superscripts of all latent variables are omitted for simplicity. Then we can decompose the right hand side of the above equation as follows :

$$\begin{aligned}\log[p(\tilde{\mathbf{x}}^{t-1})p(\mathbf{x}^t)] \geq\ & \mathbb{E}_{\mathbf{z}\sim q_\varsigma(\mathbf{z}|\tilde{\mathbf{x}}^{t-1}),\mathbf{a}\sim q_\varepsilon(\mathbf{a}|\tilde{\mathbf{x}}^{t-1}),\mathbf{c}\sim q_\delta(\mathbf{c}|\tilde{\mathbf{x}}^{t-1})}\left[\log p_\theta(\tilde{\mathbf{x}}^{t-1}|\mathbf{z},\mathbf{a},\mathbf{c})\right] \\ & - D_{KL}\left[q_\varsigma(\mathbf{z}|\tilde{\mathbf{x}}^{t-1})||p(\mathbf{z})\right] - \mathbb{E}_{q_\varsigma(\mathbf{z}|\tilde{\mathbf{x}}^{t-1})}D_{KL}\left[q_\varepsilon(\mathbf{a}|\mathbf{z})||p(\mathbf{a}|\mathbf{z})\right] \\ & - D_{KL}\left[q_\delta(\mathbf{c}|\tilde{\mathbf{x}}^{t-1})||p(\mathbf{c})\right] \\ & + \mathbb{E}_{\mathbf{z}\sim q_\varsigma(\mathbf{z}|\mathbf{x}^t),\mathbf{a}\sim q_\varepsilon(\mathbf{a}|\mathbf{x}^t),\mathbf{c}\sim q_\delta(\mathbf{c}|\mathbf{x}^t)}\left[\log p_\theta(\mathbf{x}^t|\mathbf{z},\mathbf{a},\mathbf{c})\right] \\ & - D_{KL}\left[q_\varsigma(\mathbf{z}|\mathbf{x}^t)||p(\mathbf{z})\right] - \mathbb{E}_{q_\varsigma(\mathbf{z}|\mathbf{x}^t)}D_{KL}\left[q_\varepsilon(\mathbf{a}|\mathbf{z})||p(\mathbf{a}|\mathbf{z})\right] \\ & - D_{KL}\left[q_\delta(\mathbf{c}|\mathbf{x}^t)||p(\mathbf{c})\right]\end{aligned} \tag{22}$$

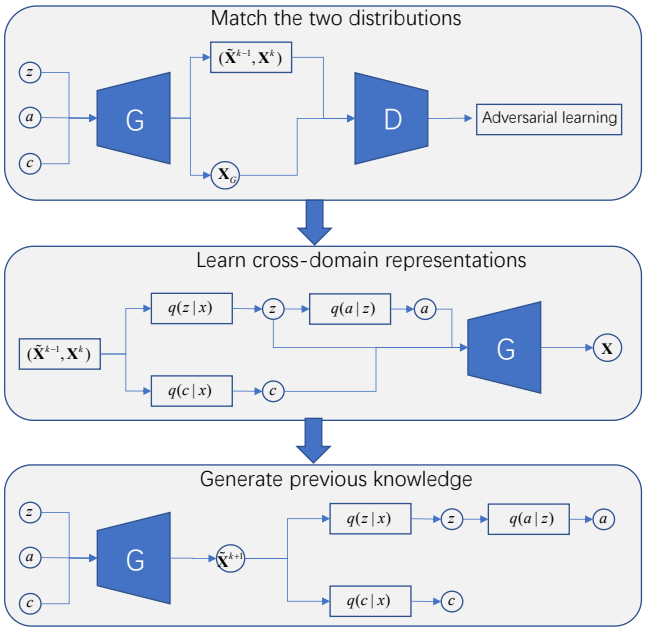



**Fig. 2.** The structure of the supervised learning.

We ignore the subscripts for the sake of simplicity and rewrite the above equation as:

$$\begin{aligned}\log[p(\tilde{\mathbf{x}}^{t-1})p(\mathbf{x}^t)] = & \sum^2 \mathbb{E}_{\mathbf{z}\sim q_\varsigma(\mathbf{z}|\mathbf{x}),\mathbf{a}\sim q_\varepsilon(\mathbf{a}|\mathbf{x}),\mathbf{c}\sim q_\delta(\mathbf{c}|\mathbf{x})}\left[\log p_\theta(\mathbf{x}|\mathbf{z},\mathbf{a},\mathbf{c})\right] \\ & - D_{KL}\left[q_\varsigma(\mathbf{z}|\mathbf{x})||p(\mathbf{z})\right] - \mathbb{E}_{q_\varsigma(\mathbf{z}|\mathbf{x})}D_{KL}\left[q_\varepsilon(\mathbf{a}|\mathbf{z})||p(\mathbf{a}|\mathbf{z})\right] \\ & - D_{KL}\left[q_\delta(\mathbf{c}|\mathbf{x})||p(\mathbf{c})\right]\end{aligned} \tag{23}$$

Where we use $\sum^2$ to denote the joint model log-likelihood which includes both data from a new database as well as data generated by the generator and corresponding to the previously learnt databases. In practice, we sample a batch of images from both the true data distribution $p(\mathbf{x}^t)$ and from the previously learnt distributon $p(\tilde{\mathbf{x}}^{t-1})$ for estimating the gradients of the data with respect to the model parameters in the Stochastic Gradient Descent (SGD) training.

# H The pseudocode and learning process for the supervised algorithm.

The pseudocode of the proposed algorithm is provided in Algorithm 1. The learning procedure is illustrated in Fig. 2. Two objective functions, adversarial loss and log-likelihood maximization, are employed to train the generator and inference models, respectively. Once the learning for the current task is fulfilled, the generator starts generating replay data samples while the inference models infer the latent variables from the generated images.

**Algorithm 1** The supervised training algorithm for L-VAEGAN.

1:Sample $X^T = \{x_1^T, x_2^T, ...., x_N^T\}$ from the T-th task
2:Sample $Y^T = \{y_1^T, y_2^T, ...., y_N^T\}$ from the T-th task
3:Assign $A^T = \{a_1^1, a_2^1, ...., a_N^1\}$ for the T-th task
4:Sample $\{X^1, .., X^{T-1}\} = \{x_1^1, x_2^2, ...., x_N^{T-1}\}$ from the previous task
5:Obtain $\{Y^1, .., Y^{T-1}\} = \{y_1^1, y_2^2, ...., y_N^{T-1}\}$ inferred by the encoder
6:Obtain $\{A^1, .., A^{T-1}\} = \{a_1^1, a_2^2, ...., a_N^{T-1}\}$ inferred by the encoder
7 : $X_{\text{Joint}} = X^T \cup \{X^1, .., X^{T-1}\}$
8 : $Y_{\text{Joint}} = Y^T \cup \{Y^1, .., Y^{T-1}\}$
9 : $A_{\text{Joint}} = A^T \cup \{A^1, .., A^{T-1}\}$
10:**While** $epoch < epoch^{\max}$ **do**
11 : **While** $batch < batch^{\max}$ **do** minibatch procedure
12 : $x_{batch} = Select(epoch, X_{\text{Joint}})$ batch samples
13 : $y_{batch} = Select(epoch, Y_{\text{Joint}})$ batch samples
14 : $a_{batch} = Select(epoch, A_{\text{Joint}})$ batch samples
15 : **Wake phase :**
16 : Train the generator and discriminator by optimizing $\mathrm{L}_{GAN}^{G}(\theta_t, \omega_t)$
17 : **Dreaming phase :**
18 : Train the generator and encoders by optimizing $\mathrm{L}_{VAE}^{J}(\theta_t, \varsigma_t, \varepsilon_t, \delta_t)$
19 : Train the class-specific and domain-specific encoders by $\mathrm{L}_a, \mathrm{L}_c$
20 : **End**
21:**End**

# I Ablation study

In this section, we investigate the importance of various model characteristics for the lifelong unsupervised representation learning.

## I.1 The choice of the latent variables

Firstly, we consider that we train the proposed framework with only a single latent variable **z** as the baseline. Afterwards, we train the proposed framework with two inference models as explained in Section 5.3 for comparison. We would like to investigate whether the proposed approach can accurately infer the task ID for the given data samples without performance loss. The average reconstructions across all testing data is reported in Table 1. We observe that the performance of the task inference model does not deteriorate while the model

is learning the information from several databases. Then we perform the task inference experiments and the results are reported in Table 2. We find that the task-inference model can infer accurate task ID for the given data. This result also demonstrates that the latent variable $\mathbf{z}$ captures the task and domain information, which enables the task-inference model $q_\varepsilon(\mathbf{a}|\mathbf{z})$ to make accurate predictions.

| MNIST and Fashion | | | | |
|---|---|---|---|---|
| Methods | Lifelong | Dataset | Reco | Acc |
| L-VAEGAN | M-F | MNIST | 4.75 | 92.53 |
| Baseline | M-F | MNIST | 4.71 | 91.29 |
| L-VAEGAN | M-F | Fashion | 17.44 | 67.66 |
| Baseline | M-F | MNIST | 16.54 | 67.97 |
| L-VAEGAN | F-M | MNIST | 4.92 | 93.29 |
| Baseline | F-M | MNIST | 5.14 | 92.34 |
| L-VAEGAN | F-M | Fashion | 13.16 | 66.97 |
| Baseline | F-M | MNIST | 14.78 | 66.45 |

**Table 1.** Quantitative evaluation on the representation learning ability of various methods

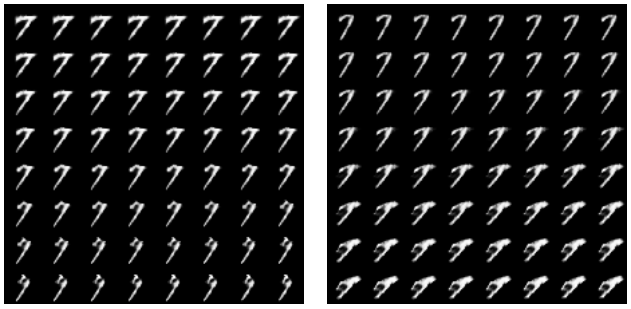
| MNIST and Fashion | | | |
|---|---|---|---|
| Methods | Lifelong | Dataset | Acc |
| L-VAEGAN | M-F | MNIST | 91.26 |
| L-VAEGAN | M-F | Fashion | 91.12 |
| L-VAEGAN | F-M | MNIST | 94.25 |
| L-VAEGAN | F-M | Fashion | 97.48 |

**Table 2.** Task inference accuracy on MNIST and Fashion.

### I.2 Enforcing the disentanglement between z and c

In this section, we investigate the effectiveness of the disentanglement between $\mathbf{z}$ and $\mathbf{c}$. We train the proposed model with three latent vectors under the lifelong supervised learning setting. After training, the inference model $q_\omega(\mathbf{c}|\mathbf{x})$ is used to make predictions. Then we change one dimension of the latent vector $\mathbf{z}$ inferred by $q_\varsigma(\mathbf{z}|\mathbf{x})$ while fixing the others. We present the results in Figure 3. We observe

that the latent variable $\mathbf{z}$ only represents the hand writing styles instead of digital types in the images.



(a) MNIST. (b) MNIST.

**Fig. 3.** The reconstruction results on MNIST when changing a single continuous latent variable and fixing all the others. We change a latent variable from -2 to 2 for MNIST and from -1 to 1 for SVHN.

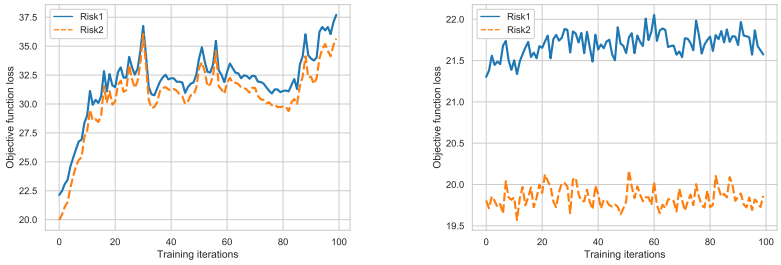
### I.3 Generalization bounds for the generative replay

In the following we provide numerical results for the generalization bounds for the generative replay mechanism described in the Section 3.3 of the paper. Firstly, we train the proposed model under the MNIST-Fashion lifelong learning, where we separately calculate the observed risk $E_t(h^1)$ (the objective function loss) for the drawing samples from $p(\mathbf{x}^1)$ and the risk $E_{1'}(h^1)$ for samples draw from $p(\tilde{\mathbf{x}}^1)$ during the second task learning. We plot the results in Figure 4-a, where risk1 and risk2 denote $E_{1'}(h^1)$ and $E_1(h^1)$, respectively. We find that $E_{1'}(h^1)$ is closer to $E_1(h^1)$ and still a bound on $E_1(h^1)$ during the course of training. And then we observe that both $E_1(h^1)$ and $E_{1'}(h^1)$ are increased due to the model's capacity (the learning adapts the weights of the model in order to perform the task associated to both MNIST and Fashion databases).

We also train the model under the MNIST-MNIST lifelong learning, where the dataset associated with the second task is comprised of generative replay data samples produced by the model trained on the first task. Figure 4-b provides the numerical results. We observe that $E_{1'}(h^1)$ is a bound on $E_1(h^1)$ and this bound is gradually slightly increased during the course of training. The reason is that the model is gradually adapting $p(\tilde{\mathbf{x}}^1)$ to the underlying distribution, and the bound is depending on the distance between $p(\tilde{\mathbf{x}}^1)$ and $p(\mathbf{x}^1)$.

### I.4 Is the two-step optimization necessary?

The proposed two-step optimization algorithm contains two independent optimization paths. However, if the proposed model would use only one of the optimization paths, then it would not be able to learn the representation of




(a) MNIST-Fashion (b) MNIST-MNIST

**Fig. 4.** Observed risks during the lifelong learning.

data on one hand, or it would lack the ability to generate higher-quality replay samples on the other hand. In order to investigate these assumptions, we firstly assume that the proposed model is only trained through the "wake" phase. In this situation, the inference network would not be trained and therefore would not learn data representations. On the other hand, we consider to train the proposed model by using only the "dreaming" phase as our baseline. We report the results in Table 3. From these results we observe that without the "wake" phase, the proposed model can not learn good latent representations when compared with the model trained with both "wake" and "dreaming" phases. The reason for this is that the log-likelihood optimization can not provide high-quality generative replay samples and this would result in a deterioration of the performance.

**Table 3.** The reconstruction error and classification accuracy after MNIST to Fashion lifelong learning.

| MNIST and Fashion | | | | |
|---|---|---|---|---|
| Methods | Lifelong | Dataset | Reco | Acc |
| L-VAEGAN | M-F | MNIST | **4.75** | **92.53** |
| baseline | M-F | MNIST | 8.94 | 90.13 |
| L-VAEGAN | M-F | Fashion | **17.44** | **67.66** |
| baseline | M-F | Fashion | 21.35 | 63.89 |
| L-VAEGAN | F-M | MNIST | **4.92** | **93.29** |
| baseline | F-M | MNIST | 8.32 | 89.56 |
| L-VAEGAN | F-M | Fashion | **13.16** | **66.97** |
| baseline | F-M | Fashion | 19.98 | 61.49 |

### I.5 Reducing the memory use

Instead of generating a collection of data samples from the generator before the next task learning, we can use a small buffer to preserve the current model's parameters before the next task learning. Then the preserved model is used to

sample a batch of images, which is used in the next task learning. The learning structure is shown in Figure 5, where the buffer is always fixed when increasing the number of tasks to be learnt. After the current task learning, the old model parameters stored in the buffer will be replaced by the current model parameters. And then in the new task learning, this buffer is used to generate a batch of images from the stored model. The buffer used in our model can achieve a similar performance without the need to increase the required memory when increasing the number of tasks to be learnt. This mechanism provides a reduced memory requirement in the proposed model.

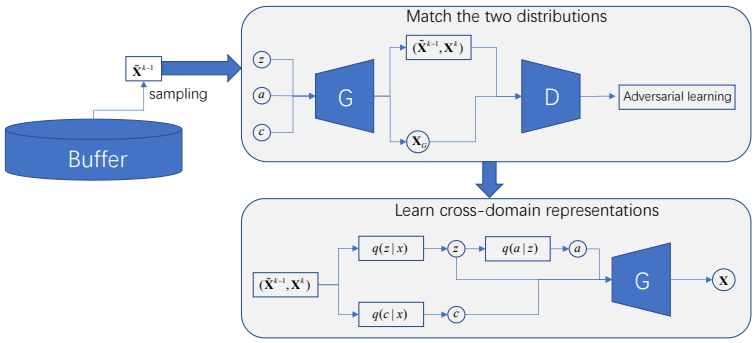
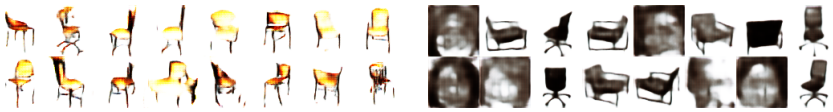



**Fig. 5.** The structure of the proposed model with buffer.

### I.6 Is the Generative Replay Mechanism (GRM) important?

In the following experiments we consider the proposed model without using the generative replay mechanism as the baseline in unsupervised experiments. We use the same hyperparameter setting for the baseline and for the approach proposed in the paper. We train the baseline under the CelebA to 3D-Chair lifelong learning and the results are shown in Figure 6. It can be observed from these results, that when we would not use GRM, the model quickly forgets the knowledge learned from the previous databases and cannot give appropriate image generations and accurate reconstructions for the images from previous tasks.

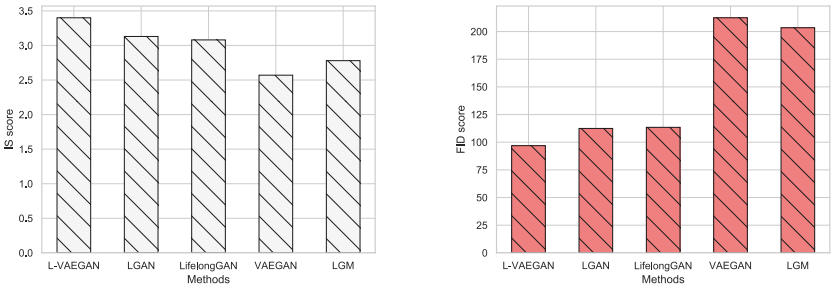


(a) Generations. (b) Reconstructions.

**Fig. 6.** Reconstruction and generation results when considering CelebA to 3D-chair lifelong learning without using the generative replay mechanism.

## J Image quality evaluation using the FID and IS score

In the following, we introduce to use the Inception score (IS) [8] and Fréchet Inception Distance (FID) [1] in order to evaluate the quality of generated image results. We train various methods considering the Cifar10 [3] to MNIIST database lifelong learning. After training, we calculate the IS score on 5,000 generated images, some of which are shown in Figure 7-a, where we compare our results with four popular lifelong learning approaches : LGAN [10], LifelongGAN [11], VAEGAN [4] and LGM [6]. The visual results are reported in Figure 8. We can observe that LifelongGAN [11] requires to use previous real data samples to prevent forgetting, when is applied in generation tasks. The results show that GAN based lifelong approaches achieve higher IS score than VAE based methods and this it can be observed in the quality of the images generated, where VAEs usually generate blurred images. The approach proposed in this paper not only produces higher-quality generative replay images but also learns representations of data that other GAN based lifelong learning approaches can not model. We also train various methods under the CelebA to CACD lifelong learning setting. The FID scores are calculated between 5,000 target images and 5,000 generated images, which are displayed in Figure 7-b. We sample 5,000 images from both CelebA and CACD databases as target images for calculating FID.



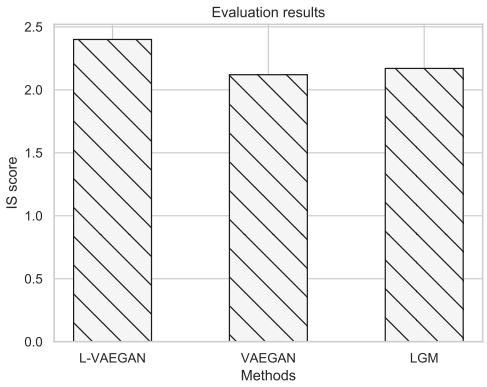

(a) IS evaluation. (b) FID evaluation.

**Fig. 7.** IS and FID evaluations.

(a) Real testing samples (b) Generations. (c) Reconstructions.

**Fig. 8.** The generation and reconstruction produced by L-VAEGAN after CIFAR10 to MNIST lifelong learning.

In order to further compare the representation ability of the proposed methods with other approaches, we represent and reconstruct 5,000 images from Cifar10 database and then calculate the IS score as a measure of image quality. The results are provided in Figure 9. We are not considering comparisons with GAN based approaches because these methods can not provide reconstructions of original images. These results demonstrate that the proposed L-VAEGAN learns better lifelong representations than other VAE based lifelong approaches.




(a) IS score.

**Fig. 9.** IS evaluations.

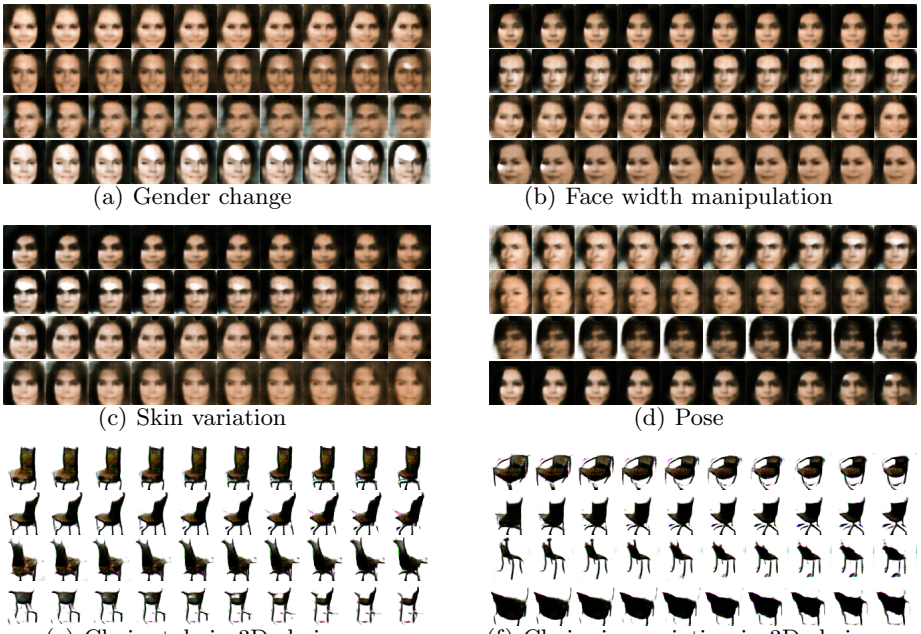
## K Additional experimental results

In this section, we present additional results to those shown and discussed in Section 6, "Experimental results" from the paper.

### K.1 Unsupervised learning

In the following we present a series of additional results, which add to those provided in Section 6.1, "Unsupervised learning" of the paper. The results from Figures 12a, 12b and 12c contain examples of real images from CelebA to CACS dataset, their generations and reconstructions, respectively. The results from Figures 13a, 13b and 13c contain examples of real images from CelebA to 3D-Chair dataset, their generations and reconstructions, respectively. These results are supplementary to the results shown in Figure 7 from page 7 in the paper. We also show that the proposed approach is able to discover four and two disentangled representations for CelebA and 3D-chair, respectively, which are illustrated in Figure 10. The results from Figure 10 show visual disentangled results, which display variations in the gender of the person whose face is shown in the image from Figure 10a, of the width of the faces in Figure 10b, skin colour variations in Figure 10c and face orientation, as shown in Figure 10d. The visual disentangled

results in the 3D Chairs show variations in the chair style in Figure 10e and in the 3D chair size in Figure 10f.



(a) Gender change (b) Face width manipulation

(c) Skin variation (d) Pose

(e) Chair style in 3D chairs (f) Chair size variation in 3D chairs

**Fig. 10.** Results when manipulating latent variables under the CelebA to 3D-Chair lifelong learning procedure. We change a single latent variables in the latent space from -3.0 to 3.0 while fixing the other latent variables.

In the following we evaluate further results for interpolating in the latent space between different domains. These results are additional to those discussed in Section 6.1 from the paper and add to the results presented in Figures 8, 10 and 11 from the paper. The visual results are provided in Figure 11.

### K.2 Lifelong learning of several databases

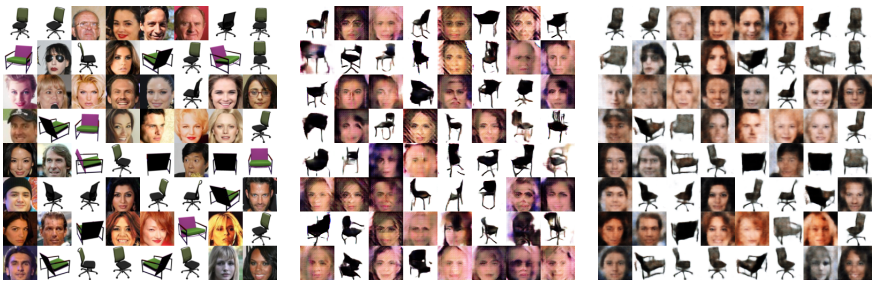
In the following we provide results when considering lifelong training using the proposed L-VAEGAN model on 4 databases: MNIST, SVHN, Fashion and InverseFashion lifelong learning, where each database is trained for 100 epochs. We evaluate the classification accuracy and average reconstruction errors of all MNIST testing samples during the lifelong learning in order to measure the loss of information. The plots showing the classification and image reconstruction are provided in Figures 14a and 14b, respectively. We observe that the proposed L-VAEGAN approach performs well when learning the first three databases while is losing some information storage capacity when training during the following stages. These results show the limitations of the generative replay mechanisms when learning a long series of tasks by training consecutively with several

(a) CelebA-CACD (b) CelebA-3DChair

**Fig. 11.** Interpolation results after lifelong learning.

(a) Real samples. (b) Generations. (c) Reconstructions.

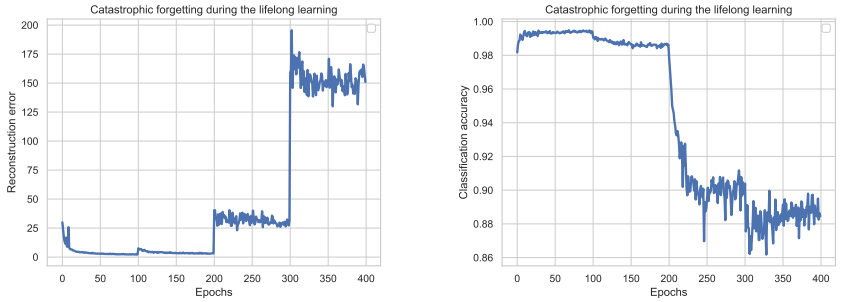
**Fig. 12.** The reconstruction and generation results on under the CelebA to CACD lifelong learning.



(a) Real samples. (b) Generations. (c) Reconstructions.

**Fig. 13.** The reconstruction and generation results on under the CelebA to 3D-Chair lifelong learning.

databases. A set of original images are shown in Figure 15a, while their reconstructions and generations are shown in Figures 15b and 15c. From these results we can observe that the L-VAEGAN can give higher-quality reconstructions even if learning four different tasks in a sequential manner.




(a) Reconstruction errors (b) Classification accuracy

**Fig. 14.** Forgetting curves during MNIST-SVHN-Fashion-IFashion lifelong learning.

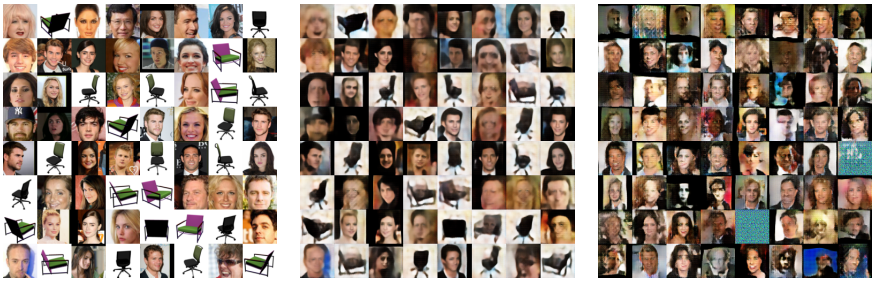
(a) Testing Images (b) Reconstructions (c) Generation

**Fig. 15.** The generation and reconstruction after lifelong learning.

We also train the model on the high dimensional datasets under the lifelong learning framework. The results are shown in Figures 16a, 16b and 16c for a set of original images, their reconstructions and corresponding generations, respectively. From these results we can observe that the generator network can not produce all images from the three different domains. This may due to the model collapse problem. The proposed approach still provides reasonable reconstructions for the given inputs, which demonstrates that it learns reasonable latent representations from previously learnt distributions of the 3D-Chair dataset.

### K.3 Transfer metric and transfer learning

By using the generative replay mechanism, the proposed approach can accelerate the training speed for learning the next task by transferring the previously



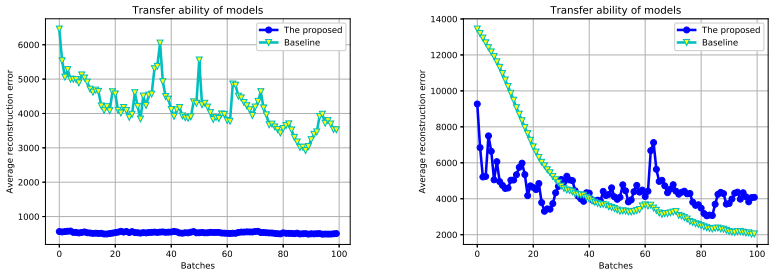
(a) Testing images (b) Reconstructed images (c) Generated images

**Fig. 16.** The generations and reconstructions after lifelong learning.

learned knowledge when learning a new task. If the current task is related to previously learnt data distributions, the model should adapt to the new task quickly. In order to measure such transfer ability in the network, we consider defining a performance score calculated by testing the data from each task in the beginning stage of training :

$$p_{k,i} = \frac{1}{N}\sum_{j=1}^{N}\phi(x_{k,j}, f_{\theta_{k,i}}(x_{k,j})) \tag{24}$$

where $p_{k,i}$ is the performance score evaluated by the model updated after $i$-th batch learning in $k$-th task, defining the corresponding database. $x_{k,j}$ is the $j$-th testing sample of the $k$-th task. $\phi$ is the performance metric which can be either the Mean Square Error (MSE) or the classification accuracy, depending on the type of task being learnt. $f_{\theta_{k,i}}(\cdot)$ is the model updated at $i$-th batch learning for $k$-th task. This performance criterion has the ability to compare the learning transfer ability.




(a) Average reconstruction errors on CACD during the lifelong CelebA to CACD

(b) Average reconstruction errors on 3D-Chair during the lifelong CelebA to 3D-Chair

**Fig. 17.** The transfer ability, calculated using (24), for the L-VAEGAN model under the CelebA to CACD, and for CelebA to 3D-Chair lifelong learning. The average reconstruction errors are calculated based on samples from CACD and 3D-Chair datasets during the second task learning, respectively.

In the following we train the proposed model under the CelebA to CACD and CelebA to 3D-Chair lifelong learning frameworks, respectively. We consider that the baseline is our model to be trained only on the CACD and 3D-chair datasets. During the training, we evaluate the performance score $p_{k,i}$ from equation (24) for each batch learning and we use the average reconstruction error as the performance metric $\phi(\cdot)$. The results are shown in Figures 17a and 17b for the CelebA to CACD database and CelebA to 3D-Chair, respectively. From Figure 17a we observe that the model gives reasonable reconstruction errors in the initial training phase of the second task. However, the baseline learns data samples rather slowly. This is due to the fact the CACD and CelebA are both human face datasets, which means that they share similar facial feature information with each other. So the model can quickly adapt to the new task as we can observe from the decreasing of the average reconstruction errors during the learning steps. From Figure 17a we observe that the proposed L-VAEGAN approach achieves lower reconstruction errors than the baseline in the beginning stage of the training procedure. Then the baseline learns faster than the proposed approach. The reason behind this is that the human face image dataset shares few features with the 3D-chair images, which have completely distinct appearance. The knowledge learned by the CelebA cannot have a positive transferable effect when learning an entirely different dataset.

### K.4 Lifelong semi-supervised learning

**Table 4.** Semi-supervised classification error results on MNIST database, under the MNIST to Fashion lifelong learning.

| Methods | Lifelong? | Error |
|---|---|---|
| L-VAEGAN | Yes | 4.34 |
| LGAN [9] | Yes | 5.46 |
| Neural networks (NN) [2] | No | 10.7 |
| (CNN) [2] | No | 6.45 |
| TSVM [2] | No | 5.38 |
| CAE [2] | No | 4.77 |
| M1+TSVM [2] | No | 4.24 |
| M2 [2] | No | 3.60 |
| M1+M2 [2] | No | 2.40 |
| Semi-VAE [5] | No | 2.88 |

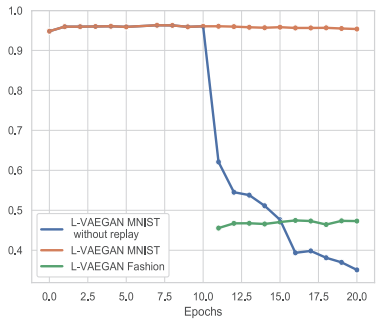



**Fig. 18.** The accuracy during the semi-supervised training on the testing data samples during the lifelong learning. The model is trained for 10 epochs for each task.

During the semi-supervised training we consider only a small number of labelled images from each database. In the following experiments we divide MNIST and Fashion datasets into two subsets each, representing labelled and unlabeled data. We consider a total of 1,000 and 10,000 labelled images for MNIST and Fashion datasets, respectively, with an equal number of data in each class for the labelled set. We train the proposed L-VAEGAN model with 10 epochs for

learning each database and the resulting classification plots are shown in Fig. 18. We observe that without generative replay samples, the model under the semi-supervised setting suffers from catastrophic forgetting.

The classification results for lifelong learning when using L-VAEGAN compared to other semi-supervised learning methods are provided in Table 4. These results show that the proposed approach outperforms LGAN [9], under the semi-supervised learning setting, and achieves competitive results when compared to the state-of-the art models which are not trained using lifelong learning.

## References

1. Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S.: Gans trained by a two time-scale update rule converge to a local Nash equilibrium. In: Proc. Advances in Neural Information Processing Systems (NIPS). pp. 6626–6637 (2017)
2. Kingma, D.P., Mohamed, S., Rezende, D.J., Welling, M.: Semi-supervised learning with deep generative models. In: Proc. Advances in Neural Inf. Proc. Systems (NIPS). pp. 3581–3589 (2014)
3. Krizhevsky, A., Hinton, G.: Learning multiple layers of features from tiny images. Tech. rep. (2009)
4. Mescheder, L., Nowozin, S., Geiger, A.: Adversarial variational Bayes: Unifying variational autoencoders and generative adversarial networks. In: Proc. Int. Conf. on Machine Learning (ICML), vol. PMLR 70. pp. 2391–2400 (2017), https://arxiv.org/abs/1701.04722
5. Narayanaswamy, S., Paige, T.B., Van de Meent, J.W., Desmaison, A., Goodman, N., Kohli, P., Wood, F., Torr, P.: Learning disentangled representations with semi-supervised deep generative models. In: Proc. Advances in Neural Inf. Proc. Systems (NIPS). pp. 5925–5935 (2017)
6. Ramapuram, J., Gregorova, M., Kalousis, A.: Lifelong generative modeling. In: Proc. Int. Conf. on Learning Representations (ICLR), arXiv preprint arXiv:1705.09847 (2017)
7. Redko, I., Habrard, A., Sebban, M.: Theoretical analysis of domain adaptation with optimal transport. In: Joint European Conference on Machine Learning and Knowledge Discovery in Databases. pp. 737–753. Springer (2017)
8. Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., Chen, X.: Improved techniques for training GANs. In: Proc. Advances in Neural Inf. Proc. Systems (NIPS). pp. 2234–2242 (2016)
9. Shin, H., Lee, J.K., Kim, J., Kim, J.: Continual learning with deep generative replay. In: Proc. Advances in Neural Inf. Proc. Systems (NIPS). pp. 2990–2999 (2017)
10. Zenke, F., Poole, B., Ganguli, S.: Continual learning through synaptic intelligence. In: Proc. of Int. Conf. on Machine Learning, vol. PLMR 70. pp. 3987–3995 (2017)
11. Zhai, M., Chen, L., Tung, F., He, J., Nawhal, M., Mori, G.: Lifelong gan: Continual learning for conditional image generation. In: Proc. of the IEEE Int. Conf. on Computer Vision (ICCV). pp. 2759–2768 (2019)